

TUTORIAL PASSO A PASSO
PARA DOMINAR O LINUX

GUIA COMPLETO PARA SE
TORNAR UM PROFISSIONAL LINUX

ESCRITO POR
CHRISTOPHER NEGUS
com a colaboração de
CHRISTINE BRESNAHAN

Linux[®]

A BÍBLIA

O MAIS ABRANGENTE E DEFINITIVO GUIA SOBRE LINUX

DESENVOLVA HABILIDADES EM
DESKTOP E SERVIDOR LINUX

AVANCE PARA O NÍVEL DA
COMPUTAÇÃO CORPORATIVA

TORNE-SE UM ADMINISTRADOR DE
SISTEMA OU UM USUÁRIO AVANÇADO



ALTA BOOKS
EDITORA

TRADUÇÃO
DA 8ª EDIÇÃO

A compra deste conteúdo não prevê atendimento e fornecimento de suporte técnico operacional, instalação ou configuração do sistema de leitor de ebooks. Em alguns casos, e dependendo da plataforma, o suporte poderá ser obtido com o fabricante do equipamento e/ou loja de comércio de ebooks.

Linux[®] A Bíblia

Tradução da 8ª Edição

Linux[®] A Bíblia

Tradução da 8a Edição

Christopher Negus

com a colaboração de
Christine Bresnahan



ALTA BOOKS
E D I T O R A
Rio de Janeiro, 2014

Como sempre, eu dedico este livro a minha esposa, Sheree.

— Chris Negus

Gostaria de dedicar este livro a minha família
e ao Senhor Deus Todo Poderoso.

“Mas esforçai-vos e não desfaleçam as vossas mãos; porque a
vossa obra tem uma recompensa.” 2 Crônicas 15:7

— Christine Bresnaham

Sobre os autores

Chris Negus passou os últimos três anos como instrutor para a Red Hat, Inc. ensinando os profissionais de TI a se tornarem Red Hat Certified Engineers (RHCE). As certificações de Chris incluem RHCE, Red Hat Certified Instructor (RHCI) e Red Hat Certified Examiner (RHCX). Ele também tem as certificações Red Hat Enterprise Virtualization (RHCVA) e Red Hat Enterprise Deployment and Systems Management.

Antes de ingressar na Red Hat, Chris escreveu e coescreveu dezenas de livros sobre Linux e UNIX, incluindo *Red Hat Linux Bible* (todas as edições), *CentOS Bible*, *Fedora Bible*, *Linux Troubleshooting Bible*, *Linux Toys* e *Linux Toys II*. Recentemente, Chris foi coautor de vários livros para a série de ferramentas Linux para usuários avançados: *Fedora Linux Toolbox*, *SUSE Linux Toolbox*, *Ubuntu Linux Toolbox*, *Mac OS X Toolbox* e *BSD UNIX Toolbox*.

Por oito anos, Chris trabalhou com a organização da AT&T que desenvolveu o UNIX, antes de se mudar para Utah a fim de contribuir para o projeto UnixWare da Novell, no início da década de 1990. Quando não está escrevendo sobre Linux, Chris gosta de jogar futebol e simplesmente passar o tempo com sua família.

Christine Bresnahan começou a trabalhar com computadores há mais de 25 anos na indústria de TI como uma administradora de sistemas. Christine é atualmente professora adjunta da Ivy Tech Community College, em Indianápolis, Indiana, ensinando administração de sistemas Linux, segurança do Linux e segurança do Windows.

Sobre o editor técnico

Richard Blum trabalha na indústria de TI há mais de 20 anos, como administrador de sistemas e administrador de redes. Blum publicou vários livros sobre Linux e Open Source. Ele administrou servidores UNIX, Linux, Novell e Microsoft, assim como ajudou a projetar e manter uma rede de 3.500 usuários utilizando switches e roteadores Cisco. Ele utiliza servidores Linux e scripts shell para executar monitoramento automatizado da rede e do sistema e escreve scripts shell na maioria dos ambientes comuns de shell Linux. Rich também é instrutor online de Linux e vários cursos de programação web que são utilizados por faculdades e universidades em todo o mundo. Quando ele não está sendo um nerd de computador, toca contrabaixo em duas bandas diferentes da igreja e gosta de passar o tempo com sua esposa, Barbara, e suas três filhas, Katie, Jane e Jessica.

Linux – A Bíblia, Tradução da 8a Edição Copyright © 2014 Starlin Alta Editora e Consultoria Eireli.
ISBN: 978-85-7608-774-8

Translated From Original Linux Bible (8rd Edition) ISBN: 978-1-11821-854-9. Original English language edition Copyright © 2012 by Pearson Education, Inc. All rights reserved including the right of reproduction in whole or in part in any form. This translation is published by Cisco Press, Inc. Portuguese language edition Copyright © 2014 by Starlin Alta Editora e Consultoria Eireli. All rights reserved including the right of reproduction in whole or in part in any form.

Todos os direitos reservados e protegidos por Lei. Nenhuma parte deste livro, sem autorização prévia por escrito da editora, poderá ser reproduzida ou transmitida.

Erratas: No site da editora relatamos, com a devida correção, qualquer erro encontrado em nossos livros (Procure pelo nome do livro).

Marcas Registradas: Todos os termos mencionados e reconhecidos como Marca Registrada e/ou Comercial são de responsabilidade de seus proprietários. A Editora informa não estar associada a nenhum produto e/ou fornecedor apresentado no livro.

Impresso no Brasil, 2014

Vedada, nos termos da lei, a reprodução total ou parcial deste livro.

Produção Editorial

Editora Alta Books

Gerência Editorial

Anderson Vieira

Editoria de Atualização

Vanessa Gomes

Supervisão Gráfica

Angel Cabeza

Supervisão de Qualidade Editorial

Sergio Luiz de Souza

Supervisão de Texto

Jaciara Lima

Conselho de Qualidade Editorial

Anderson Vieira

Angel Cabeza

Jaciara Lima

Sergio Luiz de Souza

Design Editorial

Auleriano Messias

Equipe Editorial

Claudia Braga

Cristiane Santos

Daniel Siqueira

Evellyn Pacheco

Livia Brazil

Milena Souza

Thiê Alves

Tradução

Edson Furmankiewicz

Revisão Gramatical

Milena Dias de Paula

Revisão Técnica

Allan Trabuco

Técnico em Processamento de Dados, amante de tecnologias, entusiasta do software livre e cofundador da Kylver Technologies.

Diagramação

Futura Editoração

Produção de ePub

Tatiana Medeiros

Dados internacionais de Catalogação na Publicação (CIP)

N394b Negus, Christopher.

Linux – a Bíblia / Christopher Negus: com a colaboração de Christine Bresnahan. – 8. ed. – Rio de Janeiro, RJ: Alta Books, 2014.

852 p.: il.; 24 cm.

Inclui material on-line.

Modo de acesso: <www.altabooks.com.br>

Inclui índice e apêndice.

Tradução de: Linux Bible.

ISBN 978-85-7608-774-8

1. Linux (Sistema operacional de computador). 2. Linux (Sistema operacional de computador) – Configurações. 3. Linux (Sistema operacional de computador) – Gerência. 4. Cliente/servidor (Computadores). I. Bresnahan, Christine. II. Título.

CDU 004.451.9LINUX

Índice para catálogo sistemático:

1. Sistemas operacionais específicos: Linux 004.451.9LINUX

(Bibliotecária responsável: Sabrina LEal Araujo – CRB 10/1507)



Rua Viúva Cláudio, 291 – Bairro Industrial do Jacaré

CEP: 20970-031 – Rio de Janeiro – Tels.: 21 3278-8069/8419 Fax: 21 3277-1253

www.altabooks.com.br – e-mail: altabooks@altabooks.com.br

www.facebook.com/altabooks – www.twitter.com/alta_books

Agradecimentos

Desde que fui contratado pela Red Hat, Inc. há três anos, tenho tido contato com os melhores desenvolvedores, testadores, profissionais de suporte e instrutores de Linux do mundo. Não posso agradecer a todos individualmente, portanto, em vez disso, saúdo a cultura da cooperação e da excelência, que serve para aprimorar minhas próprias habilidades em Linux todos os dias. Não falo bem da Red Hat porque trabalho lá; trabalho lá porque a Red Hat ganhou sua reputação como a principal força no desenvolvimento do Linux.

Dito isso, há algumas pessoas na Red Hat a quem eu gostaria de agradecer. Discussões com Victor Costea, Andrew Blum e outros instrutores da Red Hat ajudaram-me a adaptar meus modos de pensar sobre como as pessoas aprendem Linux. Em meu novo papel na Red Hat como escritor para o portal do cliente, meu gerente Sam Folk-Williams me permitiu ampliar minhas habilidades em áreas como virtualização, solução de problemas e ajuste de desempenho.

Quando precisava de ajuda para terminar este livro no prazo, Christine Bresnahan foi uma excelente adição à equipe. Christine escreveu um novo material com conteúdo atualizado para os capítulos de segurança deste livro, trazendo suas sólidas habilidades em ensino de Linux e redação para este projeto.

Quanto às pessoas na Wiley, sou particularmente grato pela paciência. Mary James e Linda Harrison gentilmente me guiaram ao longo de uma agenda exigente. Obrigado a Richard Blum por sua edição técnica completa... dando-me um viés Ubuntu a este livro mais centrado no Red Hat. Obrigado a Margot Maley Hutchison e Maloney Maureen da Waterside Productions pela contratação do livro para mim junto à Wiley.

Por fim, obrigado à minha esposa, Sheree, por compartilhar sua vida comigo e fazer um ótimo trabalho na criação de Seth e Caleb.

— **Christopher Negus**

Muito obrigado à fantástica equipe da John Wiley & Sons pelo seu excelente trabalho neste projeto. Obrigado a Mary James, a editora de aquisições, por me oferecer a oportunidade de trabalhar neste livro. Também sou grato a Linda Harrison, a editora de desenvolvimento, por manter as coisas nos trilhos e tornar este livro mais apresentável. Obrigado a Linda por todo seu trabalho duro e sua dedicação. O editor técnico, Rich Blum, fez um trabalho maravilhoso de duplo controle em todo o livro, além de fazer sugestões para melhorar o conteúdo. Obrigado a Nancy Rapoport, a editora, por sua infinita paciência e diligência para tornar nosso trabalho legível.

Também gostaria de agradecer a Carole McClendon da Waterside Productions, Inc. por me dar essa oportunidade e por me ajudar na minha carreira de escritora. Também gostaria de agradecer a meu marido, Timothy, por seu incentivo, paciência e disposição para ouvir, mesmo quando ele não tinha a mínima ideia do que eu estava falando.

— **Christine Bresnahan**

Sumário Resumido

Introdução

Parte I: Começando

Capítulo 1: Começando com o Linux

Capítulo 2: Criando o desktop perfeito em Linux

Parte II: Tornando-se um usuário avançado do Linux

Capítulo 3: Utilizando o shell

Capítulo 4: Movendo-se pelo sistema de arquivos

Capítulo 5: Trabalhando com arquivos de texto

Capítulo 6: Gerenciando processos em execução

Capítulo 7: Escrevendo scripts de shell simples

Parte III: Tornando-se um administrador de sistema Linux

Capítulo 8: Aprendendo administração de sistema

Capítulo 9: Instalando o Linux

Capítulo 10: Obtendo e gerenciando software

Capítulo 11: Gerenciando contas de usuário

Capítulo 12: Gerenciando discos e sistemas de arquivos

Parte IV: Tornando-se um administrador de servidor Linux

Capítulo 13: Entendendo administração de servidores

Capítulo 14: Administrando redes

Capítulo 15: Iniciando e parando serviços

Capítulo 16: Configurando um servidor de impressão

Capítulo 17: Configurando um servidor web

Capítulo 18: Configurando um servidor FTP

Capítulo 19: Configurando um servidor de compartilhamento de arquivos do Windows (Samba)

Capítulo 20: Configurando um servidor de arquivos NFS

Capítulo 21: Solução de problemas do Linux

Parte V: Aprendendo técnicas de segurança do Linux

Capítulo 22: Entendendo a segurança básica do Linux

Capítulo 23: Entendendo a segurança avançada do Linux

Capítulo 24: Aprimorando a segurança do Linux com o SELinux

Capítulo 25: Protegendo o Linux em uma rede

Parte VI: Apêndices

Apêndice A: Mídia

Sumário

Introdução

Parte I Começando

Capítulo 1: Começando com o Linux

Entendendo o que é o Linux

Entendendo como o Linux difere de outros sistemas operacionais

Explorando a história do Linux

- A cultura de fluxo livre do UNIX no Bell Labs

- O UNIX comercializado

 - Chega a Berkeley Software Distribution

 - O UNIX Laboratory e a Comercialização

- O GNU faz a transição do UNIX para a liberdade

- O BSD perde um pouco da sua força

- Linus constrói a peça que faltava

- Definição de código-fonte aberto da OSI

Entendendo como as distribuições Linux surgiram

- Escolhendo uma distribuição Red Hat

 - Usando o Red Hat Enterprise Linux

 - Usando o Fedora

- Escolhendo o Ubuntu ou outra distribuição Debian

Encontrando oportunidades profissionais com Linux hoje

- Entendendo como as empresas fazem dinheiro com Linux

- Tornando-se um profissional certificado em Red Hat

 - Tópicos do RHCSA

 - Tópicos RHCE

Resumo

Capítulo 2: Criando o desktop perfeito em Linux

Entendendo a tecnologia de desktop do Linux

Começando com o GNOME Fedora Desktop Live CD

Usando o desktop GNOME 3

- Depois que o computador inicia

 - Navegando com o mouse

- Navegando com o teclado
- Configurando o desktop GNOME 3
- Estendendo o desktop GNOME 3
 - O uso de extensões GNOME Shell
 - Usando o GNOME Tweak Tool
- Começando com aplicativos desktop
 - Gerenciando arquivos e pastas com o Nautilus
 - Instalando e gerenciando software adicional
 - Reproduzindo música com o Rhythmbox
- Parando o desktop GNOME 3
- Usando o desktop GNOME 2
 - Utilizando o gerenciador de janelas Metacity
 - Alterando a aparência do GNOME
 - Usando os painéis GNOME
 - Usando os menus Applications e System
 - Adicionando um miniaplicativo (applet)
 - Adicionando outro painel
 - Adicionando um launcher de aplicativo
 - Adicionando uma gaveta
 - Alterando as propriedades do painel
 - Efeitos 3D com o AIGLX
- Resumo
- Exercícios

Parte II: Tornando-se um usuário avançado do Linux

Capítulo 3: Utilizando o shell

- Sobre Shells e Janelas de Terminal
 - Usando o prompt de shell
 - Usando uma janela terminal
 - Usando consoles virtuais
- Escolhendo Seu Shell
- Executando comandos
 - Entendendo a sintaxe de comando
 - Localizando comandos
- Lembrando comandos com o histórico de comandos
 - Edição da linha de comando
 - Completamento de linha de comando
 - Recuperação de linhas de comando
- Conectando e expandindo comandos

- Redirecionamento entre os comandos
- Comandos sequenciais
- Comandos em segundo plano
- Expandindo comandos
- Expandindo expressões aritméticas
- Expandindo variáveis
- Usando variáveis de shell
 - Criação e uso de aliases
 - Encerrando o shell
- Criando Seu Ambiente de Shell
 - Configurando seu shell
 - Configurando seu prompt
 - Adicionando variáveis de ambiente
- Obtendo Informações Sobre Comandos
- Resumo
- Exercícios

Capítulo 4: Movendo-se pelo sistema de arquivos

- Usando comandos básicos do sistema de arquivos
- Usando Metacaracteres e Operadores
 - Utilizando metacaracteres para correspondência de arquivo
 - Utilizando metacaracteres para redirecionamento de arquivos
 - Uso de caracteres de expansão
- Listando arquivos e diretórios
- Entendendo Permissões e Posse de Arquivos
 - Alterando permissões com chmod (números)
 - Alterando permissões com chmod (letras)
 - Configurando a permissão de arquivo padrão com umask
 - Alterando a posse de arquivo
- Movendo, copiando e excluindo arquivos
- Resumo
- Exercícios

Capítulo 5: Trabalhando com arquivos de texto

- Editando Arquivos com vim e vi
 - Iniciando com o vi
 - Adicionando texto
 - Movendo-se pelo texto
 - Excluindo, copiando e alterando texto

Colando texto

Repetindo comandos

Saindo do vi

Outras maneiras de se mover por um arquivo

Procurando texto

Usando o modo ex

Aprendendo mais sobre o vi e o vim

Localizando arquivos

Usando locate para localizar arquivos por nome

Procurando arquivos com find

Localizando arquivos por nome

Localizando arquivos por tamanho

Localizando arquivos por usuário

Localizando arquivos por permissão

Localizando arquivos por data e hora

Usando not e or ao localizar arquivos

Localizando arquivos e executando comandos

Pesquisando o conteúdo de arquivos com grep

Resumo

Exercícios

Capítulo 6: Gerenciando processos em execução

Entendendo Processos

Listando Processos

Listando processos com ps

Listando e alterando processos com top

Listando processos com o System Monitor

Gerenciando Processos em Primeiro e Segundo Planos

Iniciando processos em segundo plano

Utilizando comandos em primeiro e em segundo plano

Eliminando e Repriorizando Processos

Eliminando processos com kill e killall

Usando kill para sinalizar processos por PID

Usando killall para sinalizar processos por nome

Configurando a prioridade sobre o processador com nice e renice

Resumo

Exercícios

Capítulo 7: Escrevendo Scripts de Shell Simples

Entendendo Scripts do Shell

- Executando e depurando scripts de shell

- Entendendo variáveis de shell

 - Parâmetros de shell posicionais especiais

 - Lendo parâmetros

 - Expansão de parâmetros no bash

- Fazendo aritmética em scripts de shell

- Usando construções de programação em scripts de Shell

 - As instruções “if...then”

 - O comando case

 - O loop “for...do”

 - Os loops “while...do” e “until...do”

- Experimentando alguns programas úteis de manipulação de texto

 - O general regular expression parser

 - Remova seções de linhas de texto (cut)

 - Traduza ou exclua caracteres (tr)

 - O editor de fluxo (sed)

- Usando scripts de shell simples

 - Lista telefônica

 - Script de backup

Resumo

Exercícios

Parte III: Tornando-se um administrador de sistema Linux

Capítulo 8: Aprendendo administração de sistema

- Entendendo a administração do sistema

- Usando ferramentas de administração gráfica

- Usando a conta do usuário root

 - Tornando-se root a partir do shell (comando su)

 - Ganhando acesso administrativo com sudo

- Explorando comandos administrativos, arquivos de configuração e arquivos de log

 - Comandos administrativos

 - Arquivos de configuração administrativa

 - Arquivos de log administrativos

- Usando outras contas administrativas

- Verificando e configurando o hardware

 - Verificando seu hardware

 - Gerenciando hardware removível

Trabalhando com módulos carregáveis

Listando os módulos carregados

Carregando módulos

Removendo módulos

Resumo

Exercícios

Capítulo 9: Instalando o Linux

Escolhendo um Computador

Instalando o Fedora a partir de um Live CD

Instalando o Red Hat Enterprise Linux a partir de Mídia de Instalação

Instalando o Linux na Empresa

Explorando Tópicos Comuns Sobre a Instalação

Atualizando ou instalando a partir do zero

Inicialização dual

Instalando o Linux para executar virtualmente

Usando opções de inicialização da instalação

Opções de inicialização para desabilitar recursos

Opções de inicialização para problemas de vídeo

Opções de inicialização para tipos especiais de instalação

Opções de inicialização para kickstarts e repositórios remotos

Opções de inicialização diversas

Usando armazenamento especializado

Particionando discos rígidos

Entendendo os diferentes tipos de partições

Particionando durante a instalação do Fedora

Razões para esquemas de particionamento diferentes

As dicas para criar partições

Usando o gerenciador de inicialização GRUB

Usando o GRUB Legacy (versão 1)

Usando o GRUB 2

Resumo

Exercícios

Capítulo 10: Obtendo e gerenciando software

Gerenciando software com o PackageKit

Procurando pacotes

Instalando e removendo pacotes

Indo além do PackageKit

Entendendo o empacotamento de software RPM do Linux

Entendendo pacotes RPM

O que há em um RPM?

De onde vêm os RPMs?

Instalando RPMs

Gerenciando pacotes RPM com o YUM

Entendendo como funciona o yum

1. Verificando /etc/yum.conf

2. Verificando /etc/sysconfig/rhn/up2date (RHEL somente)

3. Verificando arquivos /etc/yum.repos.d/ *.repo

4. Baixando pacotes RPM e metadados de um repositório YUM

5. Pacotes RPM instalados para o sistema de arquivos Linux

6. Armazene o repositório de metadados YUM no banco de dados RPM local

Usando o YUM com repositórios de software de terceiros

Gerenciando software com o comando YUM

Procurando por pacotes

Instalando e removendo pacotes

Atualizando pacotes

Atualizando grupos de pacotes

Mantendo o banco de dados e o cache de pacotes RPM

Baixando RPMs de um repositório yum

Instalando, consultando e verificando software com o comando rpm

Instalando e removendo pacotes com o comando rpm

Consultando informações do rpm

Verificando pacotes RPM

Gerenciando software na empresa

Resumo

Exercícios

Capítulo 11: Gerenciando contas de usuário

Criando contas de usuário

Adicionando usuários com o useradd

Configurando padrões de usuário

Modificando usuários com usermod

Excluindo usuários com userdel

Entendendo contas de grupo

Usando contas de grupo

Criando contas de grupo

Gerenciando usuários na empresa

Definindo permissões com listas de controle de acesso

Configurando ACLs com setfacl

Definindo ACLs padrão

Habilitando ACLs

Adicionando diretórios para os usuários colaborarem

Criando diretórios de colaboração em grupo (bit set GID)

Criando diretórios de exclusão restrita (sticky bit)

Centralizando contas de usuário

Usando a janela Authentication Configuration

Resumo

Exercícios

Capítulo 12: Gerenciando discos e sistemas de arquivos

Entendendo armazenamento em disco

Particionando discos rígidos

Visualizando partições de disco

Criando um disco de uma única partição

Criando um disco de múltiplas partições

Usando partições LVM

Verificando uma LVM existente

Criando volumes lógicos LVM

Aumentando volumes lógicos LVM

Montando sistemas de arquivos

Sistemas de arquivos suportados

Ativando áreas de troca

Desativando a área de troca

Utilizando o arquivo fstab para definir sistemas de arquivos montáveis

Utilizando o comando mount para montar sistemas de arquivos

Montando uma imagem de disco em loopback

Usando o comando umount

Usando o comando mkfs para criar um sistema de arquivos

Resumo

Exercícios

Parte IV: Tornando-se um administrador de servidor Linux

Capítulo 13: Entendendo administração de servidores

Começando com administração do servidor

Passo 1: Instale o servidor

Passo 2: Configure o servidor

Usando arquivos de configuração

Verificando a configuração padrão

Passo 3: Inicie o servidor

Passo 4: Proteja o servidor

Proteção por senha

Firewalls

TCP Wrappers

SELinux

Configurações de segurança em arquivos de configuração

Passo 5: Monitore o servidor

Configure o registro em log

Execute relatórios de atividade do sistema

Mantenha o software de sistema atualizado

Verifique sinais de invasão do sistema de arquivos

Gerenciando o acesso remoto com o serviço Secure Shell

Iniciando o serviço openssh-server

Usando ferramentas de cliente SSH

Usando SSH para login remoto

Usando SSH para execução remota

Copiando arquivos entre sistemas com scp e rsync

Cópia interativa com sftp

Utilizando autenticação baseada em chave (sem senha)

Configurando o registro em log do sistema

Ativando o log do sistema com rsyslog

Entendendo o arquivo rsyslog.conf

Entendendo o arquivo de log de mensagens

Configurando e usando um servidor de logs com rsyslogd

Observando logs com logwatch

Verificando recursos do sistema com sar

Verificando o espaço do sistema

Exibindo espaço em disco do sistema com df

Verificando o uso do disco com du

Descobrimo o consumo em disco com find

Resumo

Exercícios

Capítulo 14: Administrando redes

Configurando uma rede para desktops

Verificando suas placas de rede

- Verificando sua rede a partir do NetworkManager
- Verificando sua rede a partir da linha de comando
- Configurar interfaces de rede
- Configurando uma conexão de rede proxy
- Configurando redes para servidores
 - Utilizando system-config-network
 - Escolhendo a configuração do dispositivo
 - Escolhendo a configuração do DNS
 - Entendendo os arquivos de configuração de rede
 - Arquivos de configuração de placas de rede
 - Outros arquivos de rede
 - Configurando aliases de placas de rede
 - Configurando agregação de canais Ethernet
 - Definindo rotas personalizadas
- Configurando redes na empresa
 - Configurando o Linux como um roteador
 - Configurando o Linux como um servidor DHCP
 - Configurando o Linux como um servidor de DNS
 - Configurando o Linux como um servidor proxy
 - Configurando VLANs no Linux
- Resumo
- Exercícios

Capítulo 15: Iniciando e parando serviços

- Entendendo o daemon init do Linux
 - Entendendo os daemons de inicialização clássicos
 - Entendendo o daemon Upstart init
 - Aprendendo noções básicas sobre o daemon Upstart init
 - Aprendendo a retrocompatibilidade de Upstart com SysVinit
 - Entendendo systemd init
 - Aprendendo noções básicas sobre systemd
 - Aprendendo a retrocompatibilidade de systemd com SysVinit
- Auditando serviços
 - Auditando o daemon SysVinit clássico
 - Auditando o daemon Upstart init
 - Auditando init systemd
- Parando e iniciando serviços
 - Parando e iniciando o daemon SysVinit clássico
 - Parando e iniciando o daemon Upstart init
 - Parando e iniciando o daemon systemd

- Parando um serviço com systemd
- Iniciando um serviço com systemd
- Reiniciando um serviço com systemd
- Recarregando um serviço com systemd

Configurando serviços persistentes

- Configurando serviços persistentes do daemon SysVinit clássico
- Configurando serviços persistentes do daemon Upstart init
- Configurando serviços persistentes de systemd init
 - Ativando um serviço com systemd
 - Desativando (removendo) um serviço com systemd

Configurando um runlevel ou uma target unit padrão

- Configurando o nível de execução padrão do SysVdaemon init clássico
- Configurando o nível de execução padrão do daemon Upstart init
- Configurando a target unit padrão de systemd init

Adicionando serviços novos ou personalizados

- Adicionando novos serviços ao SysVdaemon init clássico
 - Passo 1: Crie um arquivo de script de serviço novo ou personalizado
 - Passo 2: Mova o script de serviço
 - Passo 3: Adicione o serviço a runlevels

Acrescentando novos serviços ao daemon Upstart init

Acrescentando novos serviços a systemd init

- Passo 1: Criar um arquivo de configuração de unidade de serviço novo ou personalizado
- Passo 2: Mover o arquivo de configuração de unidade de serviço
- Passo 3: Adicionar o serviço ao diretório Wants

Resumo

Exercícios

Capítulo 16: Configurando um servidor de impressão

Sistema comum de impressão UNIX

Configurando impressoras

- Adicionando uma impressora automaticamente
- Usando a administração baseada na web do CUPS
- Usando a janela Printer Configuration
 - Configurando impressoras locais com a janela Printer Configuration
 - Configurando impressoras remotas
 - Adicionando uma impressora CUPS remota
 - Adicionando uma impressora remota UNIX (LDP/LPR)
 - Adicionando uma impressora ao Windows (SMB)

Trabalhando com impressão CUPS

- Configurando o servidor CUPS (cupsd.conf)
- Iniciando o servidor CUPS
- Configurando opções de impressora CUPS manualmente
- Usando comandos de impressão
 - Imprimindo com lpr
 - Listando o status com lpc
 - Removendo trabalhos de impressão com lprm
- Configurando servidores de impressão
 - Configurando uma impressora CUPS compartilhada
 - Configurando uma impressora compartilhada na rede Samba
 - Entendendo smb.conf para impressão
 - Configurando clientes SMB
- Resumo
- Exercícios

Capítulo 17: Configurando um servidor web

- Entendendo o servidor web Apache
- Obtendo e instalando o servidor web
 - Entendendo o pacote httpd
 - Instalando o Apache
- Iniciando o Apache
 - Tornando o Apache seguro
 - Permissões e posse de arquivos no Apache
 - Apache e iptables
 - Apache e SELinux
 - Entendendo os arquivos de configuração do Apache
 - Usando diretivas
 - Entendendo as configurações padrão
 - Adicionando um host virtual ao Apache
 - Permitindo que os usuários publiquem seu próprio conteúdo web
 - Protegendo seu tráfego na web com SSL/TLS
 - Entendendo como o SSL é configurado
 - Gerando uma chave SSL e um certificado autoassinado
 - Gerando uma solicitação de assinatura de certificado (Certificate Signing Request –CSR)
 - Verificando erros de configuração
 - Erros de acesso proibido e erros internos do servidor
- Resumo
- Exercícios

Capítulo 18: Configurando um servidor FTP

Entendendo o FTP

Instalando o servidor FTP vsftpd

Iniciando o serviço vsftpd

Protegendo seu servidor FTP

- Abrindo seu firewall para FTP

- Permitindo acesso FTP no TCP wrappers

- Configurando o SELinux para seu servidor FTP

- Relacionando as permissões de arquivos Linux com o vsftpd

Configurando seu servidor FTP

- Configurando o acesso do usuário

- Permitindo upload

- Configurando vsftpd para a internet

Usando clientes FTP para se conectar ao servidor

- Acessando um servidor FTP a partir do Firefox

- Acessando um servidor FTP com o comando lftp

- Usando o cliente gFTP

Resumo

Exercícios

Capítulo 19: Configurando um servidor de compartilhamento de arquivos do Windows (Samba)

Entendendo o Samba

Instalando o Samba

Iniciando e parando o Samba

- Iniciando o serviço Samba (smb)

- Iniciando o servidor de nomes NetBIOS (nmbd)

- Parando os serviços do Samba (SMB) e do NetBIOS (BNM)

Protegendo o Samba

- Configurando firewalls para o Samba

- Configurando o SELinux para o Samba

 - Configurando os booleanos do SELinux para o Samba

 - Definindo contextos de arquivo do SELinux para o Samba

- Configurando permissões de host/usuário do Samba

Configurando o Samba

- Utilizando system-config-samba

 - Escolhendo as configurações do servidor Samba

 - Configurando contas de usuários do Samba

 - Criando uma pasta compartilhada Samba

 - Verificando o compartilhamento Samba

- Configurando o Samba no arquivo smb.conf
 - Configurando a seção [global]
 - Configurando a seção [homes]
 - Configurando a seção [printers]
 - Criando diretórios compartilhados personalizados
- Acessando compartilhamentos do Samba
 - Acessando compartilhamentos do Samba no Linux
 - Acessando compartilhamentos do Samba no Windows
- Usando o Samba na empresa
- Resumo
- Exercícios

Capítulo 20: Configurando um servidor de arquivos NFS

- Instalando um servidor NFS
- Iniciando o serviço NFS
- Compartilhando sistemas de arquivos NFS
 - Configurando o arquivo etc/exports
 - Hostnames em /etc/exports
 - Opções de acesso em /etc/exports
 - Opções de mapeamento de usuário em /etc/exports
 - Exportando os sistemas de arquivos compartilhados
- Protegendo seu servidor NFS
 - Abrindo seu firewall para NFS
 - Permitindo o acesso NFS em TCP wrappers
 - Configurando o SELinux para seu servidor NFS
- Usando sistemas de arquivos NFS
 - Visualizando compartilhamentos NFS
 - Montando manualmente um sistema de arquivos NFS
 - Montagem de um sistema de arquivos NFS no momento da inicialização
 - Montando sistemas de arquivos noauto
 - Usando as opções de montagem
 - Usando o autofs para montar sistemas de arquivos NFS sob demanda
 - Automontando o diretório /net
 - Automontando diretórios iniciais
- Desmontando sistemas de arquivos NFS
- Resumo
- Exercícios

Capítulo 21: Solução de problemas do Linux

Solucionando problemas de inicialização

 Começando pela BIOS

 Solucionando problemas de configuração da BIOS

 Solucionando problemas de ordem de inicialização

 Solucionando problemas do carregador de inicialização GRUB

 Iniciando o kernel

 Solucionando problemas do processo init

 Solucionando problemas do script rc.sysinit

 Solucionando problemas de processos de runlevel

Solucionando problemas de pacotes de software

 Corrigindo bancos de dados e cache RPM

Solucionando problemas rede

 Solucionando problemas de conexões de saída

 Visualize as placas de rede

 Verifique as conexões físicas

 Verifique as rotas

 Verifique a conversão de hostname

 Solucionando problemas de conexões de entrada

 Verifique se realmente o cliente pode acessar seu sistema

 Verifique se o serviço está disponível para o cliente

 Verifique o firewall no servidor

 Verifique o serviço no servidor

Solucionando problemas de memória

 Descobrimos problemas de memória

 Verificando problemas de memória

 Lidando com problemas de memória

Solucionando problemas no modo de recuperação

Resumo

Exercícios

Parte V: Aprendendo técnicas de segurança do Linux

Capítulo 22: Entendendo a segurança básica do Linux

Introdução ao Ciclo de Vida do Processo de Segurança

Examinando a fase de planejamento

 Escolhendo um modelo de controle de acesso

 Controle de acesso discricionário

 Controle de acesso mandatário

 Controle de Acesso Baseado em Papéis

 Usando listas de verificação de segurança

- Matriz de controle de acesso
- Listas de verificação de segurança da indústria
- Entrando na fase de execução
 - Implementação da segurança física
 - Implementando recuperação após desastre
 - Protegendo contas de usuário
 - Um usuário por conta de usuário
 - Nenhum login na conta root
 - Configurando as datas de expiração nas contas temporárias
 - Removendo contas de usuários não utilizadas
 - Protegendo senhas
 - Escolhendo boas senhas
 - Configurando e alterando senhas
 - Impondo melhores práticas de senha
 - Entendendo os arquivos de senha e hashes de senha
 - Protegendo o sistema de arquivos
 - Gerenciando permissões perigosas do sistema de arquivos
 - Protegendo os arquivos de senha
 - Bloqueando o sistema de arquivos
 - Gerenciando softwares e serviços
 - Removendo softwares e serviços não utilizados
 - Atualizando pacotes de software
 - Implementação avançada
- Trabalhando na fase de monitoramento
 - Monitorando arquivos de log
 - Monitorando contas de usuário
 - Detectando novas contas e privilégios falsificados
 - Detectando senhas ruins de contas
 - Monitorando o sistema de arquivos
 - Verificando pacotes de software
 - Verificando o sistema de arquivos
 - Detectando vírus e rootkits
 - Detectando uma invasão
- Trabalhando na fase de Auditoria/Revisão
 - Realizando revisões de conformidade
 - Realizando revisões de segurança
- Resumo
- Exercícios

Implementando a segurança do Linux com criptografia

Entendendo o hashing

Entendendo criptografia/decriptografia

Entendendo as cifras criptográficas

Entendendo assinaturas digitais

Implementando a criptografia no Linux

Garantindo a integridade dos arquivos

Criptografando um sistema de arquivos Linux

Criptografando um diretório do Linux

Criptografando um arquivo do Linux

Criptografando várias coisas no Linux

Implementando a segurança do Linux com PAM

Entendendo o processo de autenticação PAM

Entendendo contextos PAM

Entendendo flags de controle PAM

Entendendo módulos PAM

Compreendendo os arquivos de configuração de evento de sistema PAM

Administrando o PAM no sistema Linux

Gerenciando arquivos de configuração de aplicativos compatíveis com PAM

Gerenciando os arquivos de configuração de eventos de sistema PAM

Implementando limites de recursos com PAM

Implementando restrições de tempo com PAM

Impondo boas senhas com PAM

Incentivando o uso de sudo com PAM

Bloqueando contas com PAM

Obtendo mais informações sobre o PAM

Resumo

Exercícios

Capítulo 24: Aprimorando a segurança do Linux com o SELinux

Entendendo os benefícios do SELinux

Entendendo como o SELinux funciona

Entendendo o Type Enforcement

Entendendo a Multi-Level Security

Implementando modelos de segurança do SELinux

Entendendo os modos operacionais do SELinux

Compreendendo contextos de segurança do SELinux

Entendendo os tipos de política do SELinux

Entendendo os pacotes de regras do SELinux

Configurando o SELinux

- Definindo o modo operacional do SELinux

- Definindo o tipo de política do SELinux

- Gerenciando os contextos de segurança do SELinux

 - Gerenciando o contexto de segurança do usuário

 - Gerenciando o contexto de segurança de arquivo

 - Gerenciando o contexto de segurança do processo

- Gerenciando pacotes de regras de política do SELinux

- Gerenciando o SELinux via booleanos

Monitoramento e solução de problemas no SELinux

- Entendendo o registro em log do SELinux

 - Revisando mensagens SELinux no log de auditoria

 - Revisando mensagens SELinux no log de mensagens

- Solucionando problemas no registro em log do SELinux

- Solucionando problemas comuns do SELinux

 - Usar um diretório não padrão para um serviço

 - Usar uma porta não padrão para um serviço

 - Movendo arquivos e perdendo rótulos de contexto de segurança

 - Booleanos definidos incorretamente

Juntando tudo

Obtendo informações adicionais sobre o SELinux

Resumo

Exercícios

Capítulo 25: Protegendo o Linux em uma rede

Auditando serviços de rede

- Avaliando o acesso aos serviços de rede

 - Usando nmap para criar uma lista de serviços de rede

 - Usando nmap para auditar anúncios dos serviços de rede

- Controlando o acesso aos serviços de rede

Trabalhando com firewalls

- Entendendo firewalls

- Implementando firewalls

 - Entendendo o utilitário iptables

 - Usando o utilitário iptables

Resumo

Exercícios

Parte VI: Apêndices

Apêndice A: Mídia

Obtendo o Fedora

Obtendo o Red Hat Enterprise Linux

Obtendo o Ubuntu

Criando CDs e DVDs Linux

Gravando CDs/DVDs no Windows

Gravando CDs/DVDs em um sistema Mac OS X

Gravando CDs/DVDs no Linux

Gravando CDs a partir de um desktop Linux

Gravando CDs a partir de uma linha de comando do Linux

Iniciando o Linux a partir de um pen drive USB

Apêndice B: Respostas dos Exercícios

Capítulo 2: Criando o desktop Linux perfeito

Capítulo 3: Utilizando o shell

Capítulo 5: Trabalhando com arquivos de texto

Capítulo 6: Gerenciando processos em execução

Capítulo 7: Escrevendo scripts de shell simples

Capítulo 8: Aprendendo administração de sistema

Capítulo 9: Instalando o Linux

Capítulo 10: Obtendo e gerenciando software

Capítulo 11: Gerenciando contas de usuário

Capítulo 12: Gerenciando discos e sistemas de arquivos

Capítulo 13: Entendendo administração de servidores

Capítulo 14: Administrando redes

Capítulo 15: Iniciando e parando serviços

Capítulo 17: Configurando um servidor web

Capítulo 19: Configurando um servidor de compartilhamento de arquivos do Windows (Samba)

Capítulo 22: Entendendo a segurança básica do Linux

Capítulo 23: Entendendo a segurança avançada do Linux

Capítulo 24: Aprimorando a segurança do Linux com o SELinux

Capítulo 25: Protegendo o Linux em uma rede

Introdução

Você não pode aprender Linux sem usá-lo.

Cheguei a essa conclusão depois de mais de uma década ensinando as pessoas a aprenderem Linux. Você não pode simplesmente ler um livro, você não pode simplesmente ouvir uma palestra. Você precisa de alguém para guiá-lo e você precisa mergulhar no assunto e praticar.

Em 1999, a Wiley publicou a primeira edição do livro *Linux – A Bíblia*. O enorme sucesso me deu a oportunidade de me tornar um autor de tempo integral e independente sobre o Linux. Por cerca de uma década, escrevi dezenas de livros sobre Linux e explorei as melhores maneiras de explicar Linux a partir da quietude no meu pequeno escritório doméstico.

Em 2008, peguei a estrada. Fui contratado pela Red Hat, Inc. como um instrutor em tempo integral, ensinando Linux para administradores de sistema profissionais que procuram a certificação Red Hat Certified Engineer (RHCE). Em meus três anos como instrutor de Linux, aperfeiçoei minhas habilidades de ensino na frente de pessoas vivas cuja experiência em Linux variava de zero a um profissional experiente.

Neste livro, espero aplicar minha experiência em texto transformando você de alguém que nunca usou o Linux em alguém com as habilidades para se tornar um profissional em Linux.

Agora em sua oitava edição, este livro adquiriu um escopo muito amplo em edições anteriores. O livro cobria várias distribuições do Linux, descrevia como executar aplicativos, tocava na questão da administração de sistema e fornecia uma entrada para o desenvolvimento de software em Linux. Apesar de ter sido eleito um dos cinco melhores livros de todos os tempos do Linux há três anos, *Linux – A Bíblia* tinha perdido seu caminho.

Esta oitava edição de *Linux – A Bíblia* representa uma grande revisão. Quase todo o conteúdo anterior foi reescrito ou, na maioria dos casos, completamente substituído. O novo foco do livro pode ser resumido das seguintes maneiras:

- **Iniciante a profissional certificado:** Desde que você tenha usado um computador, mouse e teclado, você pode começar com este livro. Nós lhe dizemos como obter o Linux e como começar a usá-lo, passamos por temas críticos e, por fim, chegamos a como dominar a administração e segurança do sistema.
- **Focado no administrador de sistema:** Quando tiver concluído a leitura deste livro, você não só vai saber como usar o Linux mas como modificá-lo e mantê-lo. Todos os tópicos necessários para se tornar um Red Hat Certified Engineer são abordados neste livro.
- **Ênfase nas ferramentas de linha de comando:** Embora a interface gráfica de janelas para gerenciar o Linux tenha melhorado significativamente nos últimos anos, muitos recursos avançados só podem ser utilizados digitando comandos e editando arquivos de configuração manualmente. Nós ensinamos como se tornar proficiente com o shell de linha de comando do Linux.

- **Destinado a menos distribuições do Linux:** Em edições anteriores, descrevi cerca de 18 diferentes distribuições do Linux. Com apenas algumas exceções notáveis, distribuições do Linux mais populares são ou são baseadas no Red Hat (Red Hat Enterprise Linux, Fedora, CentOS etc) ou no Debian (Ubuntu, Linux Mint, Knoppix etc). Concentrei-me no Red Hat, porque é onde estão os trabalhos com Linux mais bem remunerados; abordo um pouco o Ubuntu porque é com ele que muitos dos maiores fãs do Linux começam.
- **Muitas, muitas demos e exercícios:** Em vez de apenas dizer o que o Linux faz, eu realmente mostro o que ele faz. Então, para garantir que aprendeu o assunto, você tem a oportunidade de experimentar os exercícios sozinhos. Todos os procedimentos e exercícios foram testados para funcionar no Fedora ou no Red Hat Enterprise Linux. Muitos vão funcionar no Ubuntu também.

Como este livro está organizado

O livro está organizado para que você possa começar logo no início com o Linux e crescer para se tornar um administrador de sistema Linux profissional e um “power user”, um usuário avançado.

A Parte I, “Começando”, inclui dois capítulos destinados a ajudar você a entender o que é o Linux e apresenta um desktop Linux:

- O Capítulo 1, “Começando com o Linux”, aborda temas como o que é o sistema operacional Linux, de onde ele vem e como começar a usá-lo.
- O Capítulo 2, “Criando o desktop perfeito em Linux”, fornece informações sobre como você pode criar um sistema desktop e usar alguns dos recursos de desktop mais populares.

A Parte II, “Tornando-se um usuário avançando do Linux”, fornece detalhes em profundidade sobre como usar o shell do Linux, trabalhar com sistemas de arquivos, manipular arquivos de texto, gerenciar processos e usar scripts de shell:

- O Capítulo 3, “Utilizando o Shell”, inclui informações sobre como acessar um shell, executar comandos, recuperar comandos (usando o histórico) e usar o completamento de comando com a tecla Tab. O capítulo também descreve como usar variáveis, aliases e páginas do manual.
- O Capítulo 4, “Movendo-se pelo sistema de arquivos”, inclui comandos para listar, criar, copiar e mover arquivos e diretórios. Tópicos mais avançados neste capítulo incluem a segurança do sistema de arquivos, tais como a posse de arquivo, permissões e listas de controle de acesso.
- O Capítulo 5, “Trabalhando com arquivos de texto”, inclui tudo, desde editores de textos básicos até ferramentas para encontrar arquivos e pesquisar texto dentro de arquivos.
- O Capítulo 6, “Gerenciando processos em execução”, descreve a forma de ver os processos que estão em execução no sistema e alterar esses processos. Maneiras de alterar processos incluem eliminar, pausar e enviar outros tipos de sinais.
- O Capítulo 7, “Escrevendo scripts de shell simples”, inclui comandos e funções de shell que você pode reunir em um arquivo para ser executado como um único comando.

Na Parte III, “Tornando-se um administrador de sistema Linux”, você aprende a administrar sistemas Linux:

- O Capítulo 8, “Aprendendo administração de sistema”, fornece informações básicas sobre ferramentas gráficas, comandos e arquivos de configuração para a administração de sistemas Linux.
- O Capítulo 9, “Instalando o Linux”, aborda as tarefas de instalações comuns, como o particionamento de disco e a seleção inicial do pacote de software, assim como ferramentas de instalação avançadas, como a instalação a partir de arquivos kickstart.
- O Capítulo 10, “Obtendo e gerenciando software”, fornece uma compreensão de como os pacotes de software funcionam e como obtê-los e gerenciá-los.
- O Capítulo 11, “Gerenciando contas de usuário”, discute ferramentas para adicionar e excluir usuários e grupos, bem como a forma de centralizar o gerenciamento de conta de usuário.
- O Capítulo 12, “Gerenciando discos e sistemas de arquivos”, fornece informações sobre a adição de partições, criar e montar sistemas de arquivos, bem como trabalhar com gerenciamento de volume lógico.

Na Parte IV, “Tornando-se um administrador de servidor Linux”, você aprenderá a criar servidores de rede poderosos e as ferramentas necessárias para gerenciá-los:

- O Capítulo 13, “Entendendo administração de servidores”, aborda o registro em log remoto, ferramentas de monitoramento e o processo de inicialização do Linux.
- O Capítulo 14, “Administrando redes”, discute a configuração de rede.
- O Capítulo 15, “Iniciando e parando serviços”, fornece informações sobre como iniciar e parar serviços de rede.
- O Capítulo 16, “Configurando um servidor de impressão”, descreve como configurar impressoras para usar localmente no seu sistema Linux ou através da rede a partir de outros computadores.
- O Capítulo 17, “Configurando um servidor web”, descreve como configurar um servidor Web Apache.
- O Capítulo 18, “Configurando um servidor FTP”, aborda os procedimentos para a configuração de um servidor FTP que pode ser usado para permitir que outros baixem arquivos de seu sistema Linux através da rede.
- O Capítulo 19, “Configurando um servidor de compartilhamento de arquivos do Windows (Samba)”, abrange configuração do servidor de arquivos Windows com o Samba.
- O Capítulo 20, “Configurando um servidor de arquivos NFS”, descreve como usar os recursos de rede do sistema de arquivos para compartilhar pastas de arquivos entre sistemas em uma rede.
- O Capítulo 21, “Solução de problemas do Linux”, abrange ferramentas populares para a solução de problemas no seu sistema Linux.

Na Parte V, “Aprendendo técnicas de segurança do Linux”, você aprende a proteger seus sistemas e serviços Linux:

- O Capítulo 22, “Entendendo a segurança básica do Linux”, aborda os conceitos e técnicas básicas de segurança.
- O Capítulo 23, “Entendendo segurança avançada do Linux”, fornece informações sobre o uso de Pluggable Authentication Modules (PAM) e ferramentas de criptologia para reforçar a segurança e autenticação do sistema.
- O Capítulo 24, “Aprimorando a segurança do Linux com o SELinux”, mostra como ativar Security Enhanced Linux (SELinux) para garantir os serviços do sistema.
- O Capítulo 25, “Protegendo Linux na rede”, abrange recursos de segurança de rede, como firewalls iptables, para garantir a segurança dos serviços do sistema.

A Parte VI contém dois apêndices para ajudar você a obter o máximo de sua compreensão do Linux. O Apêndice A, “Mídia”, fornece orientação sobre download de distribuições do Linux. O Apêndice B, “Respostas dos Exercícios”, fornece soluções de exemplo para os exercícios incluídos nos capítulos 2 a 25.

Convenções utilizadas neste livro

Ao longo do livro, uma tipografia especial indica código e comandos. Comandos e códigos são mostrados em uma fonte monoespaçada:

```
This is how code looks.
```

No caso de um exemplo incluir tanto a entrada como a saída, a fonte monoespaçada é ainda usada, mas a entrada é apresentada em negrito para diferenciar um do outro. Eis um exemplo:

```
$ ftp ftp.handsonhistory.com  
Name (home:jake): jake  
Password: *****
```

Quanto aos estilos no texto:

- Novos termos e palavras importantes aparecem em *itálico* quando introduzidos.
- Teclas de atalho aparecem assim: Ctrl+A
- Nomes de arquivos, URLs e código dentro do texto aparecem assim:
`persistence.properties`.

Os seguintes itens chamam a atenção para pontos que são particularmente importantes.

ta

caixa de Nota fornece informações extras para as quais você precisa dedicar uma atenção especial.

ca

caixa de Dica mostra uma maneira especial de realizar uma tarefa em particular.

enção

caixa de Atenção alerta para você tomar cuidado especial ao executar um procedimento ou seu hardware ou software pode ser danificado.

Mergulhando no Linux

Se você é iniciante em Linux, pode ter vagas ideias sobre o que ele é e de onde veio. Você já deve ter ouvido algo sobre ele ser “free” no sentido de “gratuito” ou no sentido de “livre” (como a liberdade de usá-lo como você quiser). Antes de começar a pôr as mãos no Linux (o que faremos em breve), o Capítulo 1 procura responder algumas de suas perguntas sobre as origens e os recursos do Linux.

Dedique um tempo para estudar este livro a fim de entender o Linux e como você pode fazê-lo trabalhar para atender suas necessidades. Esse é o seu convite para entrar e dar o primeiro passo para se tornar um especialista em Linux!

visite o site *Linux Bible*

Para encontrar links para as várias distribuições do Linux, dicas sobre como obter a certificação Linux e correções para o livro à medida que se tornam disponíveis, visite <http://www.wiley.com/WileyCDA/>.

Linux[®] A Bíblia

Tradução da 8ª Edição

Parte I

Começando

NESTA PARTE

Capítulo 1

Começando com o Linux

Capítulo 2

Criando o Desktop Perfeito em Linux

Começando com o Linux

NESTE CAPÍTULO

Aprendendo o que é Linux

Aprendendo de onde o Linux veio

Escolhendo distribuições Linux

Explorando oportunidades profissionais com o Linux

Tomando-se certificado em Linux

O Linux é um dos avanços tecnológicos mais importantes do século XXI. Além de seu impacto sobre o crescimento da internet e do seu lugar como uma tecnologia capacitadora para uma série de dispositivos baseados em computador, o desenvolvimento do Linux tem sido um exemplo de como projetos colaborativos podem ultrapassar o que as pessoas e empresas individuais podem fazer sozinhas.

O Google roda milhares e milhares de servidores Linux para fornecer sua tecnologia de busca. Seus telefones Android são baseados no Linux. Da mesma forma, quando você baixa e executa o Google Chrome OS, você tem um navegador que é apoiado por um sistema operacional Linux.

O Facebook constrói e implanta em seu site usando o que é conhecido como uma pilha LAMP (Linux, servidor web Apache, banco de dados MySQL e linguagem de script web PHP) — todos são projetos open source, isto é, baseados em código livre. Na verdade, o próprio Facebook usa um modelo de desenvolvimento de código-fonte aberto, tornando o código-fonte para os aplicativos e ferramentas que guiam o Facebook disponíveis para o público. Esse modelo tem ajudado o Facebook a eliminar bugs rapidamente, obter contribuições de todo o mundo e impulsionar o crescimento exponencial dessa rede social.

Organizações financeiras que têm trilhões de dólares dependendo da velocidade e da segurança de seus sistemas operacionais também se baseiam muito no Linux. Essas incluem as bolsas de valores de Nova York, Chicago e Tóquio.

A ampla adoção do Linux em todo o mundo criou uma demanda enorme pelo seu conhecimento. Este capítulo inicia você em um caminho para se tornar um especialista em Linux, ajudando-o a entender o que o Linux é, de onde veio e quais são suas oportunidades por se tornar proficiente nesse sistema. O restante deste livro fornece atividades práticas para lhe ajudar a ganhar essa experiência.

Entendendo o que é o Linux

Linux é um sistema operacional de computador. Um sistema operacional consiste no software que gerencia seu computador e permite que você execute aplicativos nele. O que caracteriza os sistemas operacionais Linux e similares é:

- Detectar e preparar hardware — Quando o sistema Linux inicializa (quando você liga seu computador), ele olha para os componentes em seu computador (CPU, disco rígido, placas de rede etc.) e carrega os softwares (drivers e módulos) necessários para acessar dispositivos de hardware específicos.
- Gerenciar processos — O sistema operacional deve manter o controle de vários processos em execução ao mesmo tempo e decidir quais têm acesso à CPU e quando. O sistema também deve oferecer formas de iniciar, parar e alterar o status dos processos.
- Gerenciar memória — Memória RAM e espaço de troca (memória estendida) precisam ser alocados para aplicativos conforme eles precisam de memória. O sistema operacional decide como as solicitações de memória são manipuladas.
- Fornecer interfaces de usuário — Um sistema operacional deve fornecer maneiras de acessar o sistema. Os primeiros sistemas Linux eram acessados a partir de um interpretador de linha de comando chamado shell. Hoje, as interfaces gráficas desktop estão comumente disponíveis.
- Controlar sistemas de arquivos — Estruturas do sistema de arquivos são incorporadas ao sistema operacional (ou carregadas como módulos). O sistema operacional controla a posse e acesso aos arquivos e diretórios que contêm os sistemas de arquivos.
- Proporcionar acesso e autenticação de usuário — Criar contas de usuários e definir limites para eles é uma característica básica do Linux. Separar contas de usuário e de grupo permite o controle de seus próprios arquivos e processos.
- Oferecer utilitários administrativos — No Linux, existem centenas (talvez milhares) de comandos e janelas gráficas para fazer coisas como adicionar usuários, gerenciar discos, monitorar a rede, instalar softwares e, de maneira geral, proteger e gerenciar seu computador.
- Iniciar serviços — Para utilizar impressoras, tratar mensagens de log e fornecer uma variedade de sistemas e serviços de rede, processos rodam em segundo plano, à espera de pedidos por vir. Há muitos tipos de serviços que são executados no Linux e ele fornece diferentes maneiras de iniciar e parar esses serviços. Em outras palavras, enquanto o Linux possui navegadores para exibir páginas da web, ele também pode ser aquele que serve essas páginas a outros. Recursos de servidores populares incluem servidores de web, e-mail, banco de dados, impressora, arquivo, DNS e DHCP.
- Ferramentas de programação — Uma grande variedade de utilitários de programação para criar aplicativos e bibliotecas a fim de implementar interfaces especiais estão disponíveis com o Linux.

Como um gerenciador de sistemas Linux, você precisa aprender a trabalhar com as características

descritas anteriormente. Embora muitos recursos possam ser gerenciados usando interfaces gráficas, uma compreensão da linha de comando do shell é fundamental para alguém que administra sistemas Linux.

Os modernos sistemas Linux agora vão muito além do que podiam os primeiros sistemas UNIX (em que o Linux se baseava). Recursos avançados no Linux, frequentemente usados em grandes empresas, incluem os seguintes:

- **Clustering** — O Linux pode ser configurado para trabalhar em “clusters”, ou aglomerados de computadores, de modo a fazer vários sistemas aparecerem como um sistema para o mundo exterior. Os serviços podem ser configurados para se distribuírem entre os nós do cluster, enquanto, para aqueles que os utilizam, pareçam estar funcionando sem interrupção.
- **Virtualização** — Para gerenciar recursos de computação de forma mais eficiente, o Linux pode funcionar como um servidor de virtualização. Nesse servidor, você pode executar outros sistemas Linux, Microsoft Windows, BSD, ou outros sistemas operacionais como convidados virtuais. Para o mundo exterior, cada um dos convidados virtuais aparece como um computador separado. O KVM e o Xen são duas tecnologias em Linux para a criação de máquinas virtuais. Red Hat Enterprise Virtualization é um produto da Red Hat, Inc. para gerenciar múltiplos servidores de virtualização, convidados virtuais e armazenamento de dados.
- **Computação em tempo real** — O Linux pode ser configurado para computação em tempo real, em que os processos de alta prioridade possam esperar uma atenção rápida e previsível.
- **Armazenamento especializado** — Em vez de apenas armazenar dados no disco rígido do computador, várias interfaces especializadas de armazenamento local e em rede estão disponíveis em Linux. Dispositivos compartilhados de armazenamento em Linux incluem o iSCSI, o Fibre Channel e o InfiniBand.

Muitos desses tópicos avançados não são abordados neste livro. Mas os recursos abordados aqui para usar o shell, trabalhar com discos, iniciar e parar serviços e configurar uma grande variedade de servidores devem servir como uma base para trabalhar com esses recursos avançados.

Entendendo como o Linux difere de outros sistemas operacionais

Se você é novo no Linux, é provável que tenha usado um sistema operacional Microsoft Windows ou Apple Mac OS. Embora o Mac OS X tenha suas raízes em um sistema operacional de software livre, referido como Berkeley Software Distribution (mais sobre isso mais adiante), os sistemas operacionais da Microsoft e da Apple são considerados sistemas operacionais proprietários. Isso significa que:

- Você não pode ver o código usado para criar o sistema operacional.
- Você, portanto, não pode alterar o sistema operacional em seus níveis mais básicos se ele não atender suas necessidades — e você não pode usar o sistema operacional para construir seu próprio sistema operacional a partir do código-fonte.

- Você não pode verificar o código para encontrar erros, explorar vulnerabilidades de segurança, ou simplesmente saber o que o código está fazendo.
- Você pode não ser capaz de facilmente conectar seu próprio software com o sistema operacional se os criadores desse sistema não quiserem expor as interfaces de programação que você precisa para o mundo exterior.

Você pode ler as declarações sobre o software proprietário e dizer: “O que me importa? Eu não sou um desenvolvedor de software. Eu não quero ver ou mudar o modo como meu sistema operacional é construído.”

Isso pode ser verdade. Mas o fato de que outros podem ter software livre e de código-fonte aberto e usá-lo como quiserem tem impulsionado o crescimento explosivo da internet, telefones celulares (pense no Android), dispositivos de computação especiais (pense no Tivo) e centenas de empresas de tecnologia. O software livre baixou os custos de computação e permitiu uma explosão de inovação.

Talvez você não queira usar o Linux — de forma semelhante a que o Google, o Facebook e outras empresas têm usado — para construir a fundação de uma empresa de bilhões de dólares. Mas essas e outras empresas que agora contam com o Linux para conduzir suas infraestruturas de computador estão precisando de cada vez mais pessoas com as habilidades para rodar esses sistemas.

Você pode se perguntar como um sistema de computador que é tão poderoso e flexível tornou-se livre também. Para entender como isso aconteceu, você precisa ver de onde o Linux veio. Assim, a próxima seção deste capítulo descreve o caminho estranho e sinuoso do movimento do software livre que levou ao Linux.

Explorando a história do Linux

Algumas histórias do Linux começam com essa mensagem postada por Linus Torvalds no newsgroup `comp.os.minix` em 26 de agosto de 1991

(<http://groups.google.com/group/comp.os.minix/msg/b813d52cbc5a044b>):

Linus Benedict Torvalds

Olá pessoal por aí usando minix -

Estou criando um sistema operacional (livre) (apenas um hobby, não será grande e profissional como o gnu) para clones AT 386(486). Ele vem crescendo desde abril e está começando a ficar pronto. Eu gostaria de qualquer feedback das pessoas sobre o que gostaram ou não no minix, uma vez que meu OS se parece um pouco com ele (mesmo layout físico do sistema de arquivos (devido a razões práticas, entre outras coisas)... Quaisquer sugestões serão bem-vindas, mas não prometo que vou implementá-las. :-)

Linus (torvalds@kruuna.helsinki.fi)

P.S.: Sim — não contém nenhum código minix e tem um fs multi-threaded. NÃO é portátil [sic] (usa alternância de tarefas de 386 etc) e provavelmente nunca vai suportar outra coisa senão discos rígidos AT, já que isso é tudo o que tenho. :-)

O Minix era um sistema operacional tipo UNIX que rodava em PCs no início da década de 1990. Assim

como o Minix, o Linux também era um clone do sistema operacional UNIX. Com poucas exceções, como o Microsoft Windows, sistemas de computadores mais modernos (incluindo Mac OS X e Linux) eram provenientes de sistemas operacionais UNIX, criados originalmente pela AT&T.

Para apreciar verdadeiramente como um sistema operacional livre poderia ter sido projetado com base em um sistema proprietário dos Laboratórios Bell da AT&T, ajuda entender a cultura em que o UNIX foi criado e a cadeia de eventos que tornaram possível reproduzir livremente a essência desse sistema.

ta

saber mais sobre como o Linux foi criado, pegue o livro *Just For Fun: The Story of an Accidental Revolution*, de Linus Torvalds (Harper Collins Publishing, 2001).

A cultura de fluxo livre do UNIX no Bell Labs

Desde o início, o sistema operacional UNIX foi criado e nutrido em um ambiente de comunidade. Sua criação não foi impulsionada por necessidades do mercado, mas por um desejo de superar os obstáculos na produção de programas. A AT&T, que detinha a marca UNIX originalmente, acabou transformando o UNIX em um produto comercial, mas a essa altura, muitos dos conceitos (e até mesmo grande parte do código inicial) que tornavam o UNIX especial tinham caído em domínio público.

Se você não tem idade suficiente para lembrar quando a AT&T se dividiu, em 1984, você pode não se lembrar de uma época em que a AT&T era “a” companhia telefônica. Até o início de 1980, a AT&T não precisava pensar muito em competição, porque se você queria um telefone nos Estados Unidos, tinha de recorrer à AT&T. Ela se dava ao luxo de financiar projetos de pesquisa pura. A Meca para tais projetos era o Bell Laboratories, em Murray Hill, Nova Jersey.

Depois que um projeto chamado Multics falhou por volta de 1969, os funcionários da Bell Labs Ken Thompson e Dennis Ritchie decidiram sair e criar por conta própria um sistema operacional que oferecesse um melhor ambiente para o desenvolvimento de software. Naquela época, a maioria dos programas era escrita em cartões perfurados que tinham de ser inseridos em lotes em computadores mainframe. Em uma palestra em 1980 sobre “A Evolução do Sistema de Compartilhamento de Tempo do UNIX”, Dennis Ritchie resumiu o espírito que iniciou o UNIX:

O que queríamos preservar era não só um bom ambiente para fazer programação, mas um sistema em torno do qual um companheirismo poderia se formar. Sabíamos, por experiência, que a essência da computação em comunidade da maneira proporcionada pelo acesso remoto e o compartilhamento de tempo de máquinas não é apenas para digitar programas em um terminal em vez de um furador de papel, mas para encorajar a comunicação de perto.

A simplicidade e o poder do projeto UNIX começou a quebrar as barreiras que, até esse ponto, impediam os desenvolvedores de software. A fundação do UNIX foi criada com vários elementos-chave:

- O sistema de arquivos UNIX — Porque incluía uma estrutura que permitia níveis de subdiretórios (os quais, para os usuários de desktop de hoje, parecem pastas dentro de pastas), o UNIX poderia ser usado para organizar os arquivos e diretórios de forma intuitiva. Além disso, os complexos

métodos de acessar discos, fitas e outros dispositivos foram significativamente simplificados, representando esses dispositivos como arquivos de dispositivos individuais que você também podia acessar como itens em um diretório.

- Redirecionamento de entrada/saída — Os primeiros sistemas UNIX também incluíam redirecionamento de entrada. A partir de uma linha de comando, os usuários UNIX podiam direcionar a saída de um comando para um arquivo usando uma tecla de seta para a direita (>). Mais tarde, o conceito de redirecionamento (usando o caractere |, conhecido como “pipe” em inglês) foi ampliado fazendo a saída de um comando ser direcionada para a entrada de outro. Por exemplo, o comando a seguir concatena (`cat`) o `arquivo1` e `arquivo2`, ordena (`sort`) as linhas nesses arquivos alfabeticamente, pagina o texto classificado para a impressão (`pr`) e direciona a saída para a impressora padrão do computador (`lpr`):

```
$ cat file1 file2 | sort | pr | lpr
```

Esse método de direcionar entrada e saída permitiu aos desenvolvedores criar seus próprios utilitários especializados que podiam ser vinculados com utilitários existentes. Essa modularidade tornou possível que lotes de código fossem desenvolvidos por muitas pessoas diferentes. Um usuário poderia simplesmente montar as peças para o que ele precisava.

- Portabilidade — Simplificar a experiência de usar UNIX também o tornou extremamente portátil para rodar em diferentes computadores. Por ter drivers de dispositivo (representados por arquivos na árvore de arquivos), o UNIX poderia apresentar uma interface para aplicações de tal forma que os programas não precisavam saber sobre os detalhes do hardware subjacente. Para mais tarde portar o UNIX para outro sistema, os desenvolvedores só precisavam mudar os drivers. Os programas aplicativos não precisavam ser alterados para rodar em um hardware diferente!

Para tornar a portabilidade uma realidade, era necessária uma linguagem de programação de alto nível a fim de implementar o software necessário. Para esse fim, Brian Kernighan e Dennis Ritchie criaram a linguagem de programação C. Em 1973, o UNIX foi reescrito em C. Hoje, o C ainda é o principal idioma usado para criar o kernel dos sistemas operacionais Unix (e Linux).

Como Ritchie chegou a dizer em uma palestra, em 1979 (<http://.bell-labs.com/who/dmr/hist.html>):

Hoje, o único programa UNIX importante ainda escrito em assembler é o assembler em si, praticamente todos os programas utilitários estão em C e por isso são a maioria dos programas, embora também existam locais com muitos em Fortran, Pascal e Algol 68. Parece certo que a maior parte do sucesso do UNIX resulta da legibilidade, modificabilidade e portabilidade do seu software que, por sua vez, resulta da sua expressão em linguagens de alto nível.

Se você é um entusiasta do Linux e está interessado nos recursos de seus primeiros dias que sobreviveram até hoje, uma leitura interessante é a reimpressão do primeiro manual programador UNIX feito por Dennis Ritchie (datado em 3 de novembro de 1971). Você pode encontrá-lo no site de Dennis Ritchie: <http://cm.bell-labs.com/cm/cs/who/dmr/1stEdman.html>. A forma dessa documentação é a das “páginas man” (página do manual) do UNIX — que ainda é o principal formato

para documentar comandos de sistema operacional UNIX e Linux e ferramentas de programação de hoje em dia.

O que fica claro quando você lê a documentação inicial e os relatos do sistema UNIX é que o desenvolvimento foi um processo de fluxo livre, sem ego, e se dedicou a tornar o UNIX excelente. Esse processo levou a um compartilhamento de código (dentro e fora da Bell Labs), o que permitiu o rápido desenvolvimento de um sistema operacional UNIX de alta qualidade. Isso também levou a um sistema operacional que a AT&T teria dificuldade em recuperar mais tarde.

O UNIX comercializado

Antes da alienação de participação societária da AT&T em 1984, quando ela foi dividida em AT&T e sete empresas “Baby Bell”, a AT&T foi proibida de vender sistemas de computador. As empresas que mais tarde se tornariam Verizon, Qwest e Alcatel-Lucent eram todas parte da AT&T. Como resultado desse monopólio do sistema de telefonia, o governo dos EUA estava preocupado com a possibilidade de que uma AT&T irrestrita pudesse dominar a incipiente indústria de computadores.

Por causa disso, a AT&T foi impedida de vender computadores diretamente aos clientes antes de sua alienação, o código-fonte UNIX foi licenciado para universidades por um valor simbólico. Não havia um sistema operacional UNIX para venda pela AT&T que você não tivesse de compilar por sua própria conta.

Chega a Berkeley Software Distribution

Em 1975, o UNIX V6 tornou-se a primeira versão UNIX disponível para uso generalizado fora da Bell Laboratories. A partir desse código-fonte UNIX inicial, a primeira variante importante do UNIX foi criada na Universidade da Califórnia, em Berkeley, e foi batizada como Berkeley Software Distribution (BSD).

Durante a maior parte da década seguinte, as versões BSD e UNIX da Bell Labs tomaram direções distintas. O BSD continuou à frente no estilo fluxo livre e compartilhado, que era a marca do UNIX inicial da Bell Labs, enquanto a AT&T começava a direcionar o UNIX para a comercialização. Com a formação de um UNIX Laboratory separado, que saiu de Murray Hill e se estabeleceu em Summit, Nova Jersey, a AT&T iniciou suas tentativas de comercializar o UNIX. Em 1984, a alienação societária imposta pelo governo espreitava a AT&T e ela estava pronta para realmente começar a comercializar o UNIX.

O UNIX Laboratory e a Comercialização

O UNIX Laboratory era considerado uma joia que não conseguia encontrar um lar ou uma maneira de dar lucro. Enquanto se movia entre a Bell Laboratories e outras áreas da AT&T, seu nome mudou várias vezes. Ele é provavelmente melhor lembrado pelo nome que tinha quando começou sua proliferação a partir da AT&T: UNIX System Laboratories (USL).

O código-fonte do UNIX que saiu da USL, o legado que agora pertence em parte à Santa Cruz Operation (SCO), tem sido usado como base para os processos da SCO contra os principais fornecedores do Linux (como a IBM e a Red Hat, Inc.). Por causa disso, acho que os esforços de USL que contribuíram para o sucesso do Linux são esquecidos pela maioria das pessoas.

Durante a década de 1980, naturalmente, muitas empresas de informática tinham medo de que a recém-vendida AT&T representasse uma ameaça maior para o controle da indústria de computadores do que seria uma empresa iniciante, em Redmond, Washington. Para acalmar os temores da IBM, Intel, Digital Equipment Corporation e outras empresas de informática, o UNIX Lab fez os seguintes compromissos para garantir a igualdade de condições:

- Código-fonte único — Em vez de produzir seu próprio conjunto de caixas UNIX, a AT&T continuou a vender apenas o código-fonte e torná-lo disponível igualmente a todos os licenciados. Cada empresa, então, portaria o UNIX para seu próprio equipamento. Mas foi só em 1992, quando o laboratório foi desmembrado como uma joint venture com a Novell (chamada Univel) e então acabou sendo vendido para a Novell, que um conjunto comercial em caixa do UNIX (chamado UnixWare) foi produzido diretamente a partir daquele código-fonte.
- Interfaces publicadas — Para criar um ambiente de justiça e comunidade para seus OEMs (fabricantes de equipamentos originais), a AT&T começou a padronizar o que os diferentes fornecedores de UNIX tinham de ser capazes de fazer para continuar a ser chamado de UNIX. Para esse fim, o Portable Operating System Interface (POSIX) e a AT&T UNIX System V Interface Definition (SVID) eram especificações que os fornecedores de Unix poderiam usar para criar sistemas compatíveis com UNIX. Esses mesmos documentos também serviram como roteiros para a criação do Linux.

ta

uma postagem inicial no grupo de notícias por e-mail, Linus Torvalds fez um pedido de uma cópia, de preferência de, do padrão POSIX. Acho que ninguém da AT&T esperava alguém ser realmente capaz de escrever seu próprio código-fonte do UNIX a partir dessas interfaces, sem o uso de qualquer parte do seu código-fonte UNIX.

- Abordagem técnica — Mais uma vez, até o fim da USL, a maioria das decisões sobre a direção do UNIX eram feitas com base em considerações técnicas. O gerenciamento foi promovido entre o pessoal técnico e, até onde eu sei, nunca houve qualquer conversa sobre escrever um software para quebrar o de outras empresas ou restringir o sucesso de parceiros da USL.

Quando por fim a USL começou a contratar especialistas em marketing e criar um produto desktop UNIX para usuários finais, o Microsoft Windows já tinha uma firme predominância no mercado de desktops. Também, porque a direção do UNIX tinha ido sempre no sentido do código-fonte de licenciamento destinado a sistemas de computação de grande porte, a USL tinha dificuldades para precificar seus produtos. Por exemplo, em um software que ela estava incluindo no UNIX, a USL era obrigada a pagar taxas de licenciamento por computador que se baseavam em mainframes de US\$ 100.000 em vez de computadores de US\$ 2.000. Adicione a isso o fato de que não havia programas aplicativos disponíveis para o UnixWare e você pode ver por que o esforço falhou.

Mas um marketing de sucesso de sistemas UNIX da época estava acontecendo com outras empresas de informática. A SCO tinha encontrado um nicho de mercado, principalmente vendendo versões para PC do UNIX executando terminais burros em pequenos escritórios. A Sun Microsystems estava vendendo um grande número de estações de trabalho UNIX (originalmente baseadas no BSD, mas fundidas com o UNIX no SVR4) para programadores e aplicações de tecnologia sofisticada (como bolsas de valores).

Outros sistemas UNIX comerciais também foram surgindo na década de 1980. Essa nova afirmação de propriedade do UNIX estava começando a cobrar seu preço ao espírito das contribuições abertas. Começaram a surgir processos para proteger o código-fonte e marcas comerciais do UNIX. Em 1984, esse novo UNIX restritivo deu origem a uma organização que pavimentou uma estrada para o Linux: a Free Software Foundation (Fundação do Software Livre).

O GNU faz a transição do UNIX para a liberdade

Em 1984, Richard M. Stallman iniciou o projeto GNU (<http://www.gnu.org>), recursivamente chamado pela frase GNU is Not UNIX (GNU Não é UNIX). Como um projeto da Free Software Foundation (FSF), o GNU era para se tornar uma recodificação de todo o sistema operacional UNIX que poderia ser distribuída gratuitamente.

A página do projeto GNU (<http://www.gnu.org/gnu/thegnuproject.html>) conta a história de como ele surgiu, nas próprias palavras de Stallman. Ele também expõe os problemas que as empresas particulares de software impunham sobre os desenvolvedores que queriam compartilhar, criar e inovar.

Embora reescrever milhões de linhas de código possa parecer assustador para uma ou duas pessoas, distribuir o esforço entre dezenas ou mesmo centenas de programadores tornou o projeto possível. Lembre-se de que o UNIX foi projetado para ser construído em partes separadas que poderiam ser unidas. Como eles estavam reproduzindo comandos e utilitários com interfaces conhecidas, esse esforço poderia ser facilmente dividido entre muitos desenvolvedores.

Descobriu-se não apenas que os mesmos resultados podiam ser alcançados com um código totalmente novo, mas também que, em alguns casos, o código era melhor do que as versões originais do UNIX. Como todo mundo podia ver o código sendo produzido para o projeto, o código mal escrito podia ser corrigido rapidamente ou substituído ao longo do tempo.

Se você está familiarizado com o UNIX, tente procurar os milhares de pacotes de software GNU para seu comando UNIX favorito a partir do Free Software Directory (<http://directory.fsf.org/GNU>), Provavelmente, você vai encontrá-lo lá, junto com muitos outros projetos de software disponível como suplementos.

Com o tempo, o termo *software livre* (*free software*) tem sido quase sempre substituído pelo *software de código-fonte aberto* (*open source software*). O “software livre” é o preferido pela Free Software Foundation, enquanto o software de código-fonte aberto é promovido pela Open Source Initiative (<http://www.opensource.org>).

Para acomodar ambos os campos, algumas pessoas usam o termo *Free and Open Source Software* (FOSS) em seu lugar. Um princípio subjacente do FOSS, porém, é que, embora você seja livre para usar o software que quiser, você tem alguma responsabilidade para disponibilizar as melhorias que fez no código para outros usuários. Dessa forma, toda a comunidade pode se beneficiar do seu trabalho, assim como você se beneficiou do trabalho dos outros.

Para definir claramente como software open source deve ser tratado, o projeto de software GNU criou a GNU Public License, ou GPL. Embora muitas outras licenças de software cubram abordagens ligeiramente diferentes para proteger o software livre, a GPL é a mais conhecida — e é a única que cobre

o próprio kernel do Linux. Características básicas da GNU Public License incluem as seguintes:

- Direitos do autor — O autor original mantém os direitos para seu software.
- Distribuição livre — As pessoas podem utilizar o software GNU em seu próprio software, alterar e redistribuí-lo como bem quiserem. Essas pessoas têm de incluir o código-fonte na sua distribuição (ou torná-lo facilmente disponível).
- Direitos autorais mantidos — Mesmo que você reempacote e revenda o software, o acordo GNU original deve ser mantido com o software, o que significa que todos os seus destinatários futuros têm a oportunidade de alterar o código-fonte, assim como você fez.

Não há garantia em software GNU. Se algo sair errado, o desenvolvedor original do software não tem nenhuma obrigação de corrigir o problema. Mas muitas organizações, grandes e pequenas, oferecem pacotes de suporte pago para o software quando ele é incluído na sua distribuição Linux ou outro software de código-fonte aberto. (Veja a seção “Definição de Código-fonte Aberto da OSI”, mais adiante neste capítulo, para uma definição mais detalhada do software de código aberto.)

Apesar de seu sucesso na produção de milhares de utilitários UNIX, o projeto GNU em si não conseguiu produzir uma peça fundamental do código: o kernel. Suas tentativas de construir um kernel de código-fonte aberto com o projeto GNU Hurd (<http://www.gnu.org/software/hurd>) foram malsucedidas.

O BSD perde um pouco da sua força

O projeto de software que teve a chance de bater o Linux em ser o primeiro de código-fonte aberto foi o antigo e venerável projeto BSD. No final da década de 1980, os desenvolvedores do BSD da Universidade da Califórnia (UC) em Berkeley perceberam que já haviam reescrito a maior parte do código-fonte do UNIX que tinham recebido uma década antes.

Em 1989, a Universidade de Berkeley distribuiu seu próprio código UNIX como Net/1 e mais tarde (em 1991) como Net/2. Logo que a UC Berkeley começou a preparar um completo sistema operacional do tipo UNIX livre de todo o código da AT&T, esta entrou com uma ação judicial em 1992 contra a Universidade. A ação alegava que o software foi escrito usando segredos comerciais obtidos a partir do sistema UNIX da AT&T.

É importante notar aqui que os desenvolvedores do BSD tinham reescrito completamente o código protegido por direitos autorais da AT&T. Os direitos autorais eram o principal meio que a AT&T usava para proteger seus direitos sobre o código UNIX. Alguns acreditam que se a AT&T tivesse patenteado os conceitos abordados nesse código, poderia não haver um sistema operacional Linux (ou qualquer clone do UNIX) hoje.

A ação foi abandonada quando a Novell comprou o UNIX System Laboratories da AT&T em 1994. Mas, durante esse período crítico, havia medo e dúvidas sobre a legalidade do código BSD, o suficiente para que o impulso que o BSD havia ganho a essa altura na jovem comunidade de código-fonte aberto fosse perdido. Muitas pessoas começaram a procurar outra alternativa de código-fonte aberto. O momento era propício para um estudante universitário da Finlândia que estava trabalhando em seu próprio kernel.

ta

, as versões BSD estão disponíveis a partir de três grandes projetos: FreeBSD, NetBSD e OpenBSD. As pessoas geralmente caracterizam o FreeBSD como o mais fácil de usar, o NetBSD como disponível para o maior número de formas de hardware de computador e o OpenBSD como o mais fanaticamente seguro. Muitas pessoas mais preocupadas com a segurança ainda preferem o BSD ao Linux. Além disso, por causa de seu licenciamento, o código pode ser utilizado por fabricantes de software proprietário, como a Microsoft e a Apple, que não desejam compartilhar o código do seu sistema operacional com os outros. O Mac OS X é construído sobre um derivado do BSD.

Linus constrói a peça que faltava

Linus Torvalds começou a trabalhar no Linux em 1991, quando ainda era um estudante da Universidade de Helsinki, na Finlândia. Ele queria criar um kernel tipo UNIX para poder usar o mesmo tipo de sistema operacional que ele usava na escola em seu PC em casa. Na época, Linus estava usando Minix, mas ele queria ir além do que os padrões Minix permitiam.

Como observado anteriormente, Linus anunciou a primeira versão pública do kernel do Linux para o grupo de notícias `comp.os.minix`, em 26 de agosto de 1991, embora Torvalds ache que a primeira versão não chegou a sair até meados de setembro do mesmo ano.

Embora Torvalds afirmasse que o Linux foi escrito para o processador 386 e provavelmente não era portátil, outros continuaram a incentivar (e contribuir para) uma abordagem mais portátil nas primeiras versões do Linux. Em 5 de outubro, o Linux 0.02 foi lançado com grande parte do código assembly original reescrito na linguagem de programação C, o que tornou possível começar a portá-lo para outras máquinas.

O kernel do Linux foi a última — e mais importante — parte do código necessária para completar um sistema operacional do tipo UNIX completo sob a GPL. Então, quando as pessoas começaram a montar as distribuições, foi o nome Linux e não GNU que pegou. Algumas distribuições, como o Debian, porém, referem-se a si mesmas como distribuições GNU/Linux. (A não inclusão da palavra GNU no título ou subtítulo de um sistema operacional Linux também é motivo de muitas reclamações públicas de alguns membros do projeto GNU. Ver <http://www.gnu.org>.)

Hoje, o Linux pode ser descrito como um sistema operacional de código-fonte aberto tipo UNIX, que reflete uma combinação de conformidade de padrões com o SVID, o POSIX e o BSD. O Linux continua a apontar para a conformidade com o POSIX, bem como com as normas estabelecidas pelo proprietário da marca UNIX, The Open Group (<http://www.unix.org>).

A organização sem fins lucrativos Open Source Development Labs, rebatizada como Linux Foundation após a fusão com o Free Standards Group (<http://www.linuxfoundation.org>), que emprega Linus Torvalds, gerencia hoje a direção dos esforços de desenvolvimento do Linux. Sua lista de patrocinadores é como um “Quem é Quem” dos fornecedores de sistema e aplicativos Linux comerciais, incluindo IBM, Red Hat, SUSE, Oracle, HP, Dell, Computer Associates, Intel, Cisco Systems, entre outros. O principal objetivo da Linux Foundation é proteger e acelerar o crescimento do Linux, fornecendo proteção jurídica e padrões de desenvolvimento de software para desenvolvedores Linux.

Embora grande parte dos esforços do Linux seja voltada para a computação corporativa, grandes

aprimoramentos também continuam na área do desktop. Os ambientes de desktop KDE e GNOME aprimoram continuamente a experiência do Linux para usuários casuais. Ambientes de trabalho “leves” mais recentes, como o Xfce e o LXDE agora oferecem alternativas eficientes que hoje trazem o Linux para milhares de usuários de netbooks.

Linus Torvalds continua a manter e aprimorar o kernel do Linux.

ta

uma história mais detalhada do Linux, consulte o livro *Open Sources: Voices from the Open Source Revolution* (O’Reilly, 1999). A primeira edição está disponível online em <http://oreilly.com/catalog/opensources/book/toc.html>.

Definição de código-fonte aberto da OSI

O Linux fornece uma plataforma que permite que os desenvolvedores de software alterem o sistema operacional como quiserem e obtenham uma ampla gama de formas de ajuda para criar os aplicativos que precisam. Um dos cães de guarda do movimento de código-fonte aberto é a Open Source Initiative (OSI, <http://www.opensource.org>).

Embora o objetivo principal de software de código-fonte aberto seja disponibilizar o código-fonte, outros objetivos também são definidos pela OSI. A maioria das regras a seguir para licenças de código-fonte aberto aceitáveis servem para proteger a liberdade e a integridade do código-fonte aberto:

- Distribuição gratuita — Uma licença de código-fonte aberto não pode exigir uma taxa de quem revende o software.
- Código-fonte — O código-fonte deve ser incluído com o software e não pode haver restrições à sua redistribuição.
- Obras derivadas — A licença deve permitir a modificação e a redistribuição do código sob os mesmos termos.
- Integridade do código-fonte do autor — A licença pode exigir que aqueles que usam o código-fonte removam o nome ou a versão do projeto original se eles alterarem o código-fonte.
- Não discriminação contra pessoas ou grupos — A licença deve permitir que todas as pessoas possam usar o código-fonte.
- Nenhuma discriminação contra os campos de atividade — A licença não pode restringir um projeto de usar o código-fonte porque é comercial ou porque está associado a um campo de trabalho de que o fornecedor do software não gosta.
- Distribuição da licença — Nenhuma licença adicional deve ser necessária para usar e redistribuir o software.
- A licença não deve ser específica para um produto — A licença não pode restringir o código-fonte a uma distribuição de software específico.

- A licença não deve restringir outro software — A licença não pode impedir que alguém inclua o software de código-fonte aberto na mesma mídia que o software de código-fonte não aberto.
- A licença deve ser tecnologicamente neutra — A licença não pode restringir os métodos em que o código-fonte pode ser redistribuído.

Licenças de código-fonte aberto usadas por projetos de desenvolvimento de software devem atender a esses critérios para serem aceitas como software de código-fonte aberto pela OSI. Mais de 40 diferentes licenças são aceitas pela OSI para serem usadas a fim de marcar o software como o “OSI Certified Open Source Software”. Além da GPL, outras populares licenças aprovadas pela OSI incluem:

- LGPL — A GNU Lesser General Public License (LGPL) é frequentemente usada para a distribuição de bibliotecas das quais outros programas de aplicação dependem.
- BSD — A licença Berkeley Software Distribution permite a redistribuição do código-fonte, com a exigência de que o código-fonte mantenha o aviso de direitos autorais BSD e não use os nomes dos colaboradores para apoiar ou promover software derivado sem autorização por escrito. A principal diferença da GPL, porém, é que o BSD não requer que as pessoas que modificam o código passem essas mudanças para a comunidade. Como resultado, os fornecedores de software proprietário, como a Apple e a Microsoft têm utilizado código BSD em seus próprios sistemas operacionais.
- MIT — A licença MIT é como a licença BSD, exceto que não inclui a exigência de suporte e promoção.
- Mozilla — A licença Mozilla abrange o uso e redistribuição de código-fonte associado com o navegador Firefox e outros softwares relacionados ao projeto Mozilla (<http://www.mozilla.org>). É uma licença muito mais longa que as outras já mencionadas, pois contém mais definições de como os colaboradores e aqueles que reutilizam o código-fonte devem se comportar. Isso inclui a apresentação de um arquivo de mudanças ao fazer modificações, e que aqueles que fazem suas próprias adições ao código para a redistribuição devem estar cientes dos problemas de patentes ou outras restrições associadas ao seu código.

O resultado final do código-fonte aberto é um software que tem uma maior flexibilidade para crescer e menos fronteiras na forma como pode ser usado. Muitos acreditam que o fato de que um grande número de pessoas examinam o código-fonte de um projeto resulta em software de maior qualidade para todos. Como o defensor do código-fonte aberto Eric S. Raymond diz em uma frase muito citada: “Muitos olhos fazem todos os bugs emergirem”.

Entendendo como as distribuições Linux surgiram

Ter pacotes de código-fonte espalhados pela internet que podem ser compilados e empacotados em um sistema Linux funcionou bem para os geeks. Os usuários mais casuais de Linux, porém, precisavam de uma maneira mais simples de montar um sistema Linux. Para responder a essa necessidade, alguns dos melhores nerds começaram a construir suas próprias distribuições Linux.

Uma distribuição Linux é composta dos componentes necessários para criar um sistema de trabalho Linux

e os procedimentos necessários para ter os componentes instalados e funcionando. Tecnicamente, o Linux é realmente apenas o que é referido como kernel. Antes de o kernel poder ser útil, você deve ter outros softwares, como comandos básicos (utilitários GNU), serviços que pretende oferecer (como login remoto ou servidores web) e, possivelmente, uma interface de desktop e aplicativos gráficos. Então, você deve ser capaz de juntar tudo isso e instalá-lo no disco rígido do seu computador.

O Slackware (<http://www.slackware.com>) é uma das mais antigas distribuições Linux que ainda hoje estão sendo desenvolvidas. Ele tornou o Linux amigável para usuários menos técnicos, distribuindo software já compilado e agrupado em pacotes (os pacotes de componentes de software eram chamados de tarballs). Costumava-se usar comandos básicos do Linux na época para fazer coisas como formatar o disco, ativar o *swap* (troca de dados entre a memória e o disco) e criar contas de usuário.

Em pouco tempo, muitas outras distribuições Linux foram criadas, algumas delas para atender a necessidades especiais, como o KNOPPIX (um live CD¹ Linux), o Gentoo (um Linux elegante e customizável) e o Mandrake (mais tarde chamado Mandriva, que foi uma das várias distribuições de Linux para desktop). Mas duas principais distribuições cresceram para se tornarem a base para muitas outras distribuições: o Red Hat Linux e o Debian.

Escolhendo uma distribuição Red Hat

Quando a Red Hat Linux apareceu no final da década de 1990, rapidamente se tornou a mais popular distribuição Linux por várias razões:

- Gerenciamento de pacotes RPM — Embora os tarballs sejam bons para instalar software em seu computador, eles não funcionam tão bem quando você quer atualizar, remover ou mesmo saber mais sobre esse software. O Red Hat criou o formato de empacotamento RPM; assim, um pacote de software pode conter não só os arquivos a serem compartilhados, mas também informações sobre a versão do pacote, quem o criou, quais arquivos eram documentação ou arquivos de configuração e quando foram criados. Ao instalar software empacotado em formato RPM, essas informações sobre cada pacote de software podem ser armazenadas em um banco de dados RPM local. Tornou-se fácil encontrar o que foi instalado, atualizá-lo ou removê-lo.
- Instalação simples — O instalador Anaconda tornou muito mais simples instalar o Linux. Como um usuário, você poderá percorrer algumas perguntas simples, na maioria dos casos aceitando padrões para instalar o Red Hat Linux.
- Administração gráfica — O Red Hat adicionou simples ferramentas gráficas para configurar impressoras, adicionar usuários, definir a hora e a data, e fazer outras tarefas administrativas básicas. Como resultado, os usuários de desktop poderiam usar um sistema Linux, mesmo sem ter de executar comandos.

Durante anos, o Red Hat Linux foi a distribuição Linux preferida para profissionais e entusiastas do Linux. A Red Hat, Inc. cedia o código-fonte, bem como o código compilado, prontos para executar versões do Red Hat Linux (conhecidos como os binários). Mas, conforme as necessidades da sua comunidade de usuários Linux e os grandes clientes corporativos começavam a se distanciar, a Red Hat abandonou o Red Hat Linux e começou a desenvolver dois sistemas operacionais em vez disso: O Red Hat Enterprise Linux e o Fedora.

Usando o Red Hat Enterprise Linux

Em março de 2012, a Red Hat, Inc. tornou-se a primeira empresa de software de código-fonte aberto a gerar mais de um bilhão de dólares em receitas anuais. Ela alcançou seu objetivo construindo um conjunto de produtos em torno do Red Hat Enterprise Linux (RHEL) que atendem às necessidades dos ambientes de computação corporativos mais exigentes.

Enquanto outras distribuições Linux focaram sistemas desktop ou computação em pequenas empresas, o RHEL trabalhou sobre os recursos necessários para lidar com aplicações de missão crítica para empresas e governo. Ele construiu sistemas que poderiam acelerar transações para as maiores bolsas de valores do mundo e serem implantados como clusters e hosts virtuais.

Em vez de apenas vender o RHEL, a Red Hat oferece um ecossistema de benefícios aos clientes Linux. Para usar o RHEL, os clientes compram assinaturas que eles podem usar para implantar qualquer versão do RHEL que desejam. Se os clientes retirarem de serviço um sistema RHEL, eles podem usar a assinatura para implantar outro sistema.

Diferentes níveis de suporte estão disponíveis para o RHEL, dependendo das necessidades dos clientes. Os clientes podem ter a certeza de que, juntamente com o suporte, eles podem obter hardware e software de terceiros que sejam certificados para funcionar com o RHEL. Eles podem obter consultores e engenheiros da Red Hat para ajudá-los a montar os ambientes de computação ideais e também podem obter treinamento e exames de certificação para seus funcionários (veja a discussão da certificação RHCE mais adiante neste capítulo).

A Red Hat também acrescentou outros produtos, como extensões naturais ao Red Hat Enterprise Linux. O JBoss é um produto de *middleware*² para implantar aplicativos baseados em Java na internet ou nas intranets das empresas. A Red Hat Enterprise Virtualization é composta de hosts de virtualização, gerentes e computadores convidados que permitem instalar, executar, gerenciar, migrar e desativar grandes ambientes de computação virtual.

Há aqueles que tentaram clonar o RHEL, usando seu código-fonte livremente disponível, reconstruindo-o e reempacotando-o sob uma nova marca. O CentOS é uma distribuição patrocinada pela Comunidade Linux, que é construída a partir do código-fonte do RHEL. Da mesma forma, o Oracle Linux é construído a partir do código-fonte do RHEL, mas atualmente oferece um kernel incompatível. Apesar disso, o RHEL ainda é de longe o principal sistema operacional de computador no mundo corporativo.

Eu escolhi usar o Red Hat Enterprise Linux para muitos dos exemplos deste livro, porque, se você quer uma carreira trabalhando em sistemas Linux, há uma demanda enorme por aqueles que podem administrar sistemas RHEL. Se você está começando com Linux, porém, o Fedora pode proporcionar um excelente ponto de entrada gratuito para as mesmas habilidades que você precisa ter para usar e administrar sistemas RHEL.

Usando o Fedora

Enquanto o RHEL é a distribuição Linux comercial, estável e suportada, o Fedora é a distribuição Linux gratuita e moderna que é patrocinada pela Red Hat, Inc. O Fedora é o sistema Linux que a Red Hat usa para envolver a comunidade de desenvolvimento do Linux e incentivar aqueles que querem um Linux gratuito para uso pessoal.

O Fedora inclui mais de 16.000 pacotes de software, muitos dos quais se mantêm atualizados com a mais

recente tecnologia de código-fonte aberto disponível. Como um usuário, você pode experimentar as mais recentes interfaces desktop, de servidor e administrativas de Linux no Fedora gratuitamente. Como um desenvolvedor de software, você pode criar e testar seus aplicativos usando a última versão do kernel Linux e ferramentas de desenvolvimento.

Como o foco do Fedora é a mais recente tecnologia, ele se concentra menos na estabilidade. Portanto, é de se esperar que você tenha algum trabalho extra para que tudo funcione e que nem todo o software esteja totalmente “pronto”.

Mas eu recomendo que você use o Fedora para a maioria dos exemplos deste livro, pelas seguintes razões:

- O Fedora é usado como um campo de provas para o Red Hat Enterprise Linux. A Red Hat testa muitas novas aplicações no Fedora antes de implantá-las no RHEL. Usando o Fedora, você vai aprender as habilidades de que precisa para trabalhar com recursos à medida que eles vão sendo desenvolvidos para o Red Hat Enterprise Linux.
- Para aprendizagem, o Fedora é mais conveniente do que o RHEL, mas ainda inclui muitas das mais avançadas ferramentas prontas para uso corporativo que existem no RHEL.
- O Fedora é *free*, não só no sentido de “livre”, mas também no sentido de “gratuito”.

O Fedora é extremamente popular entre aqueles que desenvolvem software de código-fonte aberto. Mas nos últimos anos, outra distribuição Linux tem atraído a atenção de muitas pessoas que estão começando com o Linux: o Ubuntu.

Escolhendo o Ubuntu ou outra distribuição Debian

Como o Red Hat Linux, a distribuição Debian GNU/Linux foi uma distribuição Linux inicial que se destacou no empacotamento e gerenciamento de software. O Debian usa o formato de pacote deb e ferramentas para gerenciar todos os pacotes de software em seus sistemas. O Debian também tem uma reputação de estabilidade.

Muitas distribuições Linux se originam do Debian, de acordo com a DistroWatch (<http://distrowatch.com>), mais de 120. Distribuições populares baseadas no Debian incluem o Linspire, o Xandros, o Knoppix, o MEPIS, o Damn Small Linux e muitas outras. Mas o derivado Debian que alcançou o maior sucesso é o Ubuntu (<http://www.ubuntu.com>).

Ao se basear no estável desenvolvimento e empacotamento de software do Debian, a distribuição Ubuntu Linux foi capaz de avançar e adicionar os recursos que faltavam ao Debian. Em busca de trazer novos usuários para o Linux, o projeto Ubuntu adicionou um instalador gráfico simples e ferramentas gráficas fáceis de usar. Ele também focaliza sistemas desktop com funcionalidade completa e ainda oferece pacotes populares de servidor.

O Ubuntu também foi inovador na criação de novas formas de executar o Linux. Usando live CDs oferecidos pelo Ubuntu, você pode tê-lo instalado e funcionando em poucos minutos. Muitas vezes, nesses CDs também vinham aplicações de código-fonte aberto, como navegadores web e processadores de texto, que realmente funcionavam no Windows. Isso facilitou a transição do Windows para o Linux para algumas pessoas.

Se você estiver usando o Ubuntu, sem problemas. A maioria dos temas abordados neste livro vai funcionar tão bem no Ubuntu como no Fedora ou no RHEL. Porém, quando entramos em algumas das seções de servidor no livro, você pode encontrar algum conteúdo do mundo corporativo, o qual pode não corresponder exatamente com o que você encontra no Fedora ou no RHEL.

Encontrando oportunidades profissionais com Linux hoje

Se você quer desenvolver uma ideia para um projeto de pesquisa relacionado a computadores ou uma empresa de tecnologia, por onde começar? Com uma ideia. Depois disso, você olha para as ferramentas que precisa explorar e, por fim, cria a sua visão. Então, procura outros para ajudá-lo durante o processo de criação.

Hoje, os custos de abrir uma empresa como a Google ou Facebook incluem apenas um computador, uma conexão com a internet e café suficiente para mantê-lo acordado escrevendo código a noite inteira. Se você tem sua ideia de como mudar o mundo, o Linux e milhares de pacotes de software estão disponíveis para ajudá-lo a construir seus sonhos. O mundo do código-fonte aberto também é acompanhado pelas comunidades de desenvolvedores, administradores e usuários que estão disponíveis para ajudá-lo.

Se você quiser se envolver com um projeto de código-fonte aberto existente, os projetos estão sempre procurando pessoas para escrever código, testar software ou escrever documentação. Nesses projetos, você vai encontrar pessoas que usam o software ou trabalham nele e geralmente estão dispostos a partilhar sua experiência para ajudá-lo.

Mas queira você desenvolver o próximo grande projeto de software de código-fonte aberto, ou simplesmente adquirir as habilidades necessárias para competir com os milhares de trabalhos bem-remunerados de administrador ou de desenvolvimento em Linux, ele vai ajudar você a saber como instalar, proteger e manter sistemas Linux.

Então, quais são as perspectivas de carreira em Linux? A pesquisa intitulada “Linux Adoption Trends 2012: A Survey of Enterprise End Users” (Tendências de Adoção de Linux 2012: Um Levantamento dos Usuários Finais Corporativos), da Linux Foundation (<http://www.linuxfoundation.org/publications/linux-foundation/linux-adoption-trends-end-user-report-2012>), entrevistou mais de 400 trabalhadores de organizações com mais de 500 funcionários e mais de 500 milhões de dólares em vendas anuais. Eis o que a Linux Foundation descobriu:

- Aumento do uso Linux — Mais de 80% das empresas esperam que seu uso do Linux aumente ao longo dos próximos cinco anos.
- Mais Linux para **big data** — Mais de 70% das empresas esperam adicionar mais sistemas Linux para lidar com *big data* (em comparação com cerca de 36% mais sistemas Windows e 29% mais sistemas UNIX). *Big data* refere-se a enormes quantidades de informações complexas e de difícil manejo que precisam ser armazenadas e gerenciadas.
- Precisa-se de mais especialistas em Linux! — Além de algumas preocupações com a

interoperabilidade com plataformas existentes (37%), a próxima maior preocupação dos entrevistados com o Linux era serem capazes de encontrar talento para suportar esses sistemas.

A mensagem principal a considerar a partir dessa pesquisa é que o Linux continua a crescer e criar demandas de especialistas nele. As empresas que começaram a usar Linux continuaram a avançar com ele. Aqueles que utilizam o Linux continuam a expandir seu uso e acham que a redução de custos, segurança e flexibilidade que ele oferece continuam a fazer dele um bom investimento .

Entendendo como as empresas fazem dinheiro com Linux

Entusiastas do código-fonte aberto acreditam que melhores softwares podem resultar de um modelo aberto de desenvolvimento do que de modelos proprietários. Portanto, em teoria, qualquer empresa criando um software para uso próprio pode economizar dinheiro adicionando suas contribuições as dos outros a fim de obter um produto final muito melhor para ela mesma.

As empresas que querem ganhar dinheiro com a venda de software precisam ser mais criativas do que eram antigamente. Embora você possa vender o software que cria incluindo um software GPL, o código-fonte dele deve ser passado para frente. Naturalmente, outros podem então recompilar esse produto, basicamente utilizando e até mesmo revendendo-o sem custos. Eis algumas maneiras como as empresas estão lidando com essa questão:

- Assinaturas de software — A Red Hat, Inc. vende seus produtos Red Hat Enterprise Linux com base em uma assinatura. Por uma determinada quantia de dinheiro por ano, você obtém o código binário para rodar o Linux (assim você não tem que compilar por conta própria), suporte garantido, ferramentas para monitoramento de hardware e software no seu computador, acesso à base de conhecimento da empresa e outros recursos.

Apesar de o projeto da Red Hat Fedora incluir grande parte do mesmo software e também estar disponível em forma binária, não há garantias associadas com o software ou futuras atualizações dele. Um pequeno escritório ou um usuário pessoal pode correr o risco de usar o Fedora (que é em si mesmo um sistema operacional excelente), mas uma grande empresa que está executando aplicações de missão crítica provavelmente acabará investindo alguns dólares no RHEL.

- Treinamento e certificação — Com o uso do sistema Linux crescendo em uso governamental e em grandes negócios, são necessários profissionais para suportar esses sistemas. A Red Hat oferece cursos de formação e exames de certificação para ajudar os administradores de sistema a se tornarem proficientes usando sistemas Red Hat Enterprise Linux. Em particular, as certificações Red Hat Certified Engineer (RHCE) e Red Hat Certified System Administrator (RHCSA) tornaram-se populares (<http://www.redhat.com/certification>). Mais sobre RHCE/RHCSA certificações mais tarde neste capítulo.

Outros programas de certificação são oferecidos pelo Linux Professional Institute (<http://www.lpi.org>), CompTIA (<http://www.comptia.org>) e Novell (<http://www.novell.com>). LPI e CompTIA são associações profissionais da indústria de computadores. Novell centra seus cursos de treinamento e certificação em seus produtos SUSE Linux.

- **Recompensas** — Recompensas de software são uma maneira fascinante de as empresas de software de código-fonte aberto fazerem dinheiro. Digamos que você está usando o pacote de software XYZ e precisa de um novo recurso imediatamente. Ao pagar uma recompensa de software para o projeto em si, ou para outros desenvolvedores, você pode ter as melhorias que você precisa deslocadas para o início da fila. O software que você paga permanecerá coberto pela sua licença de código-fonte aberto, mas você terá os recursos de que precisa provavelmente por menos do que uma fração do custo da construção do projeto a partir zero.
- **Doações** — Muitos projetos de código-fonte aberto aceitam doações de pessoas físicas ou empresas de desenvolvimento de código-fonte aberto que usam código a partir de seus projetos. Surpreendentemente, muitos projetos de código-fonte aberto suportam um ou dois desenvolvedores e funcionam exclusivamente com base em doações.
- **Estojo, caneca e camiseta** — Muitos projetos de código-fonte aberto têm lojas online onde você pode comprar CDs (algumas pessoas ainda gostam de CDs físicos e cópias impressas de documentos) e uma variedade de canecas, camisetas, mouse pads e outros souvenirs. Se você realmente ama um projeto, compre uma camiseta!

Essa não é de forma alguma uma lista exaustiva, porque formas mais criativas estão sendo inventadas todos os dias para apoiar aqueles que criam software de código-fonte aberto. Lembre-se de que muitas pessoas se tornaram colaboradoras e mantenedoras de software de código-fonte aberto porque precisavam ou queriam o software. As contribuições que elas fazem de graça valem a pena pelo retorno que elas obtêm de outras pessoas que fazem o mesmo.

Tornando-se um profissional certificado em Red Hat

Embora este livro não seja focado em se tornar certificado em Linux, ele aborda atividades que você precisa para ser capaz de dominar a forma como passar em exames populares de certificação do Linux. Em particular, a maior parte do que é coberto nos exames Red Hat Certified Engineer (RHCE) e Red Hat Certified System Administrator (RHCSA) é descrita neste livro.

Se você estiver procurando por um trabalho como um profissional de TI em Linux, muitas vezes a certificação RHCSA ou RHCE é listada como uma exigência ou, pelo menos, uma preferência dos empregadores. O exame RHCSA (EX200) fornece a certificação básica, abordando temas como configuração de discos e sistemas de arquivos, adição de usuários, criação de um site simples para web e um servidor FTP e adição de espaço de troca. O exame de RHCE (EX300) testa para configurações de servidor mais avançadas, bem como um conhecimento avançado de recursos de segurança, como SELinux e firewalls (iptables).

Aqueles de nós que têm ensinado cursos RHCE/RHCSA e aplicado exames (como eu fiz por três anos) não estão autorizados a dizer exatamente o que está no exame. Mas a Red Hat dá uma visão geral de como os exames funcionam, bem como uma lista de possíveis tópicos que serão abordados. Você pode encontrar os objetivos do exame nos seguintes sites:

- **RHCSA** —

<http://www.redhat.com/training/courses/ex200/examobjective>

- **RHCE** — <http://www.redhat.com/training/courses/ex300/examobjective>

Como os objetivos do exame declaram, os RHCSA e RHCE são baseados em desempenho, o que significa que você recebe tarefas e deve executá-las em um Red Hat Enterprise Linux real, como faria no trabalho. Você é classificado de acordo com seu desempenho nos resultados dessas tarefas.

Se você pretende fazer os exames, verifique constantemente as páginas de objetivos deles, pois eles mudam de tempos em tempos. Tenha em mente também que o RHCSA é uma certificação independente; entretanto, você deve passar nos exames RHCSA e RHCE para obter uma certificação RHCE. Muitas vezes, os dois exames são aplicados no mesmo dia.

Você pode se inscrever em cursos de treinamento e exames RHCSA e RHCE em <http://training.redhat.com>. Cursos de treinamento e exames são dados em grandes cidades em todos os Estados Unidos e ao redor do mundo. As habilidades de que você precisa para completar as provas são descritas nas seções a seguir.

Tópicos do RHCSA

Como observado anteriormente, os tópicos do exame RHCSA cobrem habilidades básicas de administração do sistema. Esses são os temas atuais listados no site de objetivos do exame RHCSA (novamente, verifique constantemente o site de objetivos do exame em questão para ficar atualizado com eventuais alterações):

- Entender as ferramentas essenciais — Espera-se que você tenha um conhecimento prático do shell de comando (bash), incluindo como usar a sintaxe de comando adequada e fazer redirecionamento de entrada/saída (< > >>). Você precisa saber como fazer o login em sistemas remotos e locais. Espere ter de criar, mover, copiar, linkar, excluir e alterar permissões e posse de arquivos. Da mesma forma, você deve saber como procurar informações nas páginas do manual (“páginas man”) e em `/usr/share/doc`.
- Operar sistemas em execução — Nessa categoria, você deve entender o processo de inicialização do Linux, entrar no modo monousuário, desligar, reiniciar e mudar níveis de execução. Você deve ser capaz de iniciar e parar as máquinas virtuais e serviços de rede, bem como encontrar e interpretar arquivos de log.
- Configurar o armazenamento local — A criação de partições de disco inclui a criação de volumes físicos e a configuração deles a fim de utilizá-los para gerenciamento de volume lógico (*logical volume management*, LVM) ou criptografia (LUKS). Você também deve ser capaz de criar essas partições como sistemas de arquivos ou espaço de troca que podem ser montados ou ativados no momento da inicialização.
- Criar e configurar sistemas de arquivos — Criar e montar automaticamente diferentes tipos de sistemas de arquivos, incluindo sistemas de arquivos regulares do Linux (ext2, ext3 ou ext4), de arquivos criptografados com LUKS e de arquivos de rede (NFS e CIFS). Criar diretórios de colaboração usando o recurso de bit de ID de grupo e listas de controle de acesso (*access control lists*, ACLs). Você também deve ser capaz de usar o LVM para aumentar o tamanho de um volume lógico.
- Implementar, configurar e manter sistemas — Isso cobre uma variedade de tópicos, incluindo

configuração de redes, criação de tarefas `cron`, a definição do nível de execução padrão e a instalação de sistemas RHEL. Você também deve ser capaz de configurar um servidor HTTP e um servidor FTP simples. Para pacotes de software, você deve ser capaz de instalar os pacotes da Red Hat Network, um repositório remoto, ou o sistema de arquivos local. Por fim, você deve ser capaz de instalar corretamente um novo kernel e escolher esse ou algum outro kernel para iniciar quando o sistema for iniciado.

- Gerenciar usuários e grupos — Você deve saber adicionar, excluir e alterar contas de usuário e de grupo. Outro tópico que você deve saber é o envelhecimento de senha, usando o comando `chage`. Você também deve saber configurar um sistema de autenticação mediante conexão com um servidor de diretório LDAP.
- Gerenciar a segurança — Você deve ter um entendimento básico de como configurar um firewall (`system-config-firewall` ou `iptables`) e como usar o SELinux.

A maioria desses tópicos é abordada neste livro. Consulte a documentação da Red Hat (<http://docs.redhat.com>) sob o título Red Hat Enterprise Linux para descrições de características não encontradas neste livro. Em particular, o Guia de Implantação contém descrições de muitos dos tópicos relacionados com o RHCSA.

Tópicos RHCE

Os tópicos do exame RHCE cobrem configurações de servidor mais avançadas, juntamente com uma variedade de recursos para garantir a segurança desses servidores. Mais uma vez, verifique o site de objetivos do exame RHCE para informações mais atualizadas sobre os tópicos que você deve estudar para o exame.

O requisito de configuração e gerenciamento do sistema para o exame RHCE cobre uma variedade de tópicos, incluindo os seguintes:

- Roteamento de tráfego IP — Definir rotas estáticas para endereços de rede específicos.
- Firewalls — Bloquear ou permitir o tráfego para as portas selecionadas em seu sistema que oferecem serviços como web, FTP e NFS, assim como bloquear ou permitir o acesso a serviços com base no endereço IP do originador.
- Ajuste do Kernel — Configurar parâmetros ajustáveis do kernel usando o arquivo `/etc/sysctl.conf` e o comando `sysctl`
- Configure iSCSI — Configurar o sistema como um iniciador de iSCSI que monta um destino iSCSI no momento da inicialização.
- Relatórios do sistema — Usar recursos como `sar` para informar sobre o uso de memória do sistema, acesso a disco, tráfego de rede e uso do processador.
- Script de shell — Criar um script de shell simples para receber entrada e produzir saída de várias maneiras.
- Log remoto — Configurar o recurso `rsyslogd` para coletar as mensagens de log e distribuí-las para um servidor remoto de registro em log. Além disso, configurar um servidor remoto de

registro em log para coletar mensagens de log de clientes.

- SELinux — Com Security Enhanced Linux em modo imposição, certificar-se de que todas as configurações do servidor descritas na próxima seção estão devidamente protegidas com o SELinux.

Para cada um dos serviços de rede na lista que se segue, certifique-se de que você pode seguir os passos para instalar os pacotes exigidos pelo serviço, configurar o SELinux para permitir o acesso a ele, defini-lo para iniciar no momento da inicialização, garantir o serviço por host ou por usuário (usando iptables, TCP Wrappers ou recursos fornecidos pelo próprio serviço) e configurá-lo para a operação básica. Estes são os serviços:

- Servidor web — Configure um servidor Apache (HTTP/HTTPS). Você deve ser capaz de criar uma máquina virtual, implantar um script CGI, usar diretórios privados e permitir que um determinado grupo Linux gerencie o conteúdo.
- Servidor DNS — Configure um servidor DNS (pacote bind) para agir como um servidor de nomes somente cache que pode encaminhar consultas DNS com outro servidor DNS. Não há necessidade de configurar zonas de mestre ou escravo.
- Servidor FTP — Configure um servidor FTP para fornecer downloads anônimos.
- Servidor NFS — Configure um servidor NFS para compartilhar diretórios específicos com sistemas específicos de clientes a fim de que eles possam ser usados para colaboração em grupo.
- Servidor de compartilhamento de arquivos Windows — Configure o Linux (Samba) para fornecer compartilhamentos SMB a máquinas e usuários específicos. Configure os compartilhamentos para a colaboração em grupo.
- Servidor de e-mail — Configure o postfix ou sendmail para aceitar e-mails recebidos de fora do host local. Transmita correio para um host inteligente.
- Servidor Secure Shell — Configure o serviço SSH (sshd) para permitir acesso remoto ao seu sistema local, bem como autenticação baseada em chave. Ou, de outro modo, configure o arquivo `sshd.conf` conforme necessário.
- Servidor NTP — Configure um servidor Network Time Protocol (ntpd) para sincronizar o tempo com outros pares NTP.

Embora existam outras tarefas no exame RHCE, como acabamos de observar, tenha em mente que a maioria das tarefas exige que você configure servidores e, então, garanta a segurança desses servidores usando qualquer técnica que seja necessária. Essas podem incluir regras de firewall (iptables), o SELinux, TCP Wrappers, ou quaisquer recursos integrados em arquivos de configuração para o serviço específico.

Resumo

O Linux é um sistema operacional que é construído por uma comunidade de desenvolvedores de software

ao redor do mundo e liderada por seu criador, Linus Torvalds. Ele é originalmente derivado do sistema operacional UNIX, mas cresceu além do UNIX em popularidade e poder ao longo dos anos.

A história do sistema operacional Linux remonta ao início dos sistemas UNIX, que foram distribuídos gratuitamente para escolas e melhorados por iniciativas como a Berkeley Software Distribution (BSD). A Free Software Foundation ajudou a fazer muitos dos componentes necessários para criar um sistema operacional do tipo UNIX totalmente livre. O kernel do Linux em si foi o último grande componente necessário para completar o trabalho.

A maioria dos projetos de software são protegidos por um dos conjuntos de licenças que estão sob o guarda-chuva da Open Source Initiative. A mais importante delas é a GNU Public License (GPL). Normas, como a Linux Standard Base, e organizações e empresas de classe mundial Linux (como a Red Hat, Inc.) tornam possível que o Linux continue sendo um sistema operacional estável e produtivo no futuro.

Aprender os princípios de como usar e administrar um sistema Linux irá lhe servir bastante em qualquer aspecto do trabalho com ele. Os capítulos a seguir fornecem uma série de exercícios com os quais você pode testar seu conhecimento. Portanto, pelo resto deste livro, você aprenderá melhor tendo um sistema Linux à sua frente, pois poderá trabalhar com os exemplos em cada capítulo e completar os exercícios com sucesso.

O próximo capítulo descreve como começar com o Linux descrevendo como obter e utilizar um sistema desktop Linux.

¹ N.T.: Live CD é um CD que contém um sistema operacional (GNU/Linux, BSD ou outro) que não precisa ser instalada no disco rígido do usuário, uma vez que o sistema operacional completo é executado diretamente a partir do CD e da memória RAM. (Fonte: http://pt.wikipedia.org/wiki/Live_CD)

² Middleware ou mediador, no campo da computação distribuída, é um programa de computador que faz a mediação entre o software e as demais aplicações. (Fonte: <http://pt.wikipedia.org/wiki/Middleware>)

Criando o desktop perfeito em Linux

NESTE CAPÍTULO

Entendendo o X Window System e os ambientes de desktop

Executando o Linux a partir de um Live CD

Navegando na área de trabalho do GNOME 3

Adicionando extensões ao GNOME 3

Usando o Nautilus para gerenciar arquivos no GNOME 3

Trabalhando com a área de trabalho do GNOME 2

Ativando efeitos 3D em GNOME 2

Usar o Linux como seu sistema desktop cotidiano está se tornando cada vez mais fácil. Como tudo em Linux, você tem escolhas. Existem ambientes de desktop com todas as funcionalidades, como o GNOME e o KDE, ou desktops “leves”, como o LXDE ou o Xfce. Há ainda gerenciadores de janelas independentes mais simples.

Depois de ter escolhido um desktop, você vai descobrir que quase todo tipo importante de aplicativo desktop que você tem em um sistema Windows ou Mac terá aplicações equivalentes em Linux. Para aplicações que não estão disponíveis em Linux, muitas vezes você pode executar um aplicativo Windows no Linux usando software de compatibilidade para Windows.

O objetivo deste capítulo é familiarizá-lo com os conceitos relacionados com os sistemas de desktop Linux e depois dar-lhe dicas para trabalhar com eles. Você irá:

- Entender os recursos e tecnologias de desktop que estão disponíveis no Linux
- Visitar as principais características do ambiente desktop GNOME
- Aprender dicas e truques para obter o máximo de sua experiência com o desktop GNOME

Para usar as descrições deste capítulo, recomendo que você tenha um sistema Fedora rodando à sua frente. Você pode obter o Fedora de muitas maneiras, incluindo:

- **Rodando o Fedora a partir de um Live CD** — Consulte o Apêndice A para obter informações sobre como baixar e gravar um Fedora Live CD para que possa iniciá-lo sem precisar instalar em um disco rígido para usá-lo com este capítulo.
- **Instalando o Fedora permanentemente** — Instale o Fedora em seu disco rígido e inicie a partir daí (como descrito no Capítulo 9, “Instalando o Linux”).

Como a versão atual do Fedora usa a interface GNOME 3, a maioria dos procedimentos descritos aqui funcionará com outras distribuições Linux que tenham GNOME 3 disponível. Se você estiver usando o Red Hat Enterprise Linux (que, como desde o RHEL 6, usa o GNOME 2), adicionei descrições do GNOME 2 que você também pode experimentar.

ta

Ubuntu usa seu próprio desktop Unity como padrão, em vez do GNOME, mas você pode adicionar o GNOME e usá-lo como o ambiente desktop para o Ubuntu 11.10 e versões posteriores. Versões mais antigas do Ubuntu usam o GNOME 2 por padrão.

Você pode adicionar o GNOME 3 a uma versão do Ubuntu, fazendo o seguinte a partir de seu sistema Ubuntu: abra o Ubuntu Software Center, instale o GNOME Shell, faça logout, volte para a tela de login e selecione o GNOME como sua opção de desktop e faça login. Além de algumas diferenças no conjunto padrão de aplicativos disponíveis, as descrições do GNOME 3 neste livro devem funcionar no Ubuntu exatamente como apresentadas.

Entendendo a tecnologia de desktop do Linux

Os modernos sistemas de desktop oferecem janelas gráficas, ícones e menus que são operados usando mouse e teclado. Se você tem menos de 30 anos de idade, pode pensar que não há nada de especial nisso. Mas os primeiros sistemas Linux não tinham interfaces gráficas disponíveis. Além disso, hoje, muitos servidores Linux adaptados para tarefas especiais (por exemplo, funcionando como um servidor web ou um servidor de arquivos) não têm software de desktop instalado.

Quase toda grande distribuição Linux que oferece interfaces desktop se baseia no X Window System (<http://www.x.org>). O X Window System fornece uma base sobre a qual diferentes tipos de ambientes de desktop ou gerenciadores de janelas simples podem ser construídos.

O X Window System (às vezes chamado simplesmente de X) foi criado antes que o Linux existisse e antecede até o Microsoft Windows. Ele foi construído para ser um *framework* “leve”, uma base simples de desktop em rede.

O X funciona em uma espécie de modelo cliente/servidor invertido. O servidor X roda no sistema local, fornecendo uma interface com tela, teclado e mouse. Clientes X (como processadores de texto, leitores de música ou visualizadores de imagem) podem ser carregados a partir do sistema local ou de qualquer sistema em sua rede ao qual o servidor X dê permissão para fazer isso.

O X foi criado em um momento em que os terminais gráficos (clientes “magros”) simplesmente gerenciavam o teclado, o mouse e o monitor. Aplicações, armazenamento em disco e poder de processamento ocorriam, todos, em grandes computadores centrais. Portanto, os aplicativos rodavam em

máquinas maiores, mas eram exibidos e gerenciados através da rede no cliente magro. Mais tarde, os clientes magros foram substituídos por computadores pessoais. A maioria dos aplicativos cliente rodava localmente, usando o poder de processamento local, espaço em disco, memória e outros recursos de hardware, e ao mesmo tempo não permitindo a execução de aplicativos que não se iniciaram a partir do sistema local.

O X em si oferece um fundo cinza básico e um simples cursor de mouse em forma de “X”. Não há menus, painéis ou ícones em uma tela básica do X. Se você tivesse que carregar um cliente X (como uma janela de terminal ou um processador de texto), ele apareceria na tela do X sem borda em torno dele para mover, minimizar ou fechar a janela. Esses recursos são adicionados por um gerenciador de janelas.

Um gerenciador de janelas adiciona a capacidade de gerenciar as janelas no desktop e costuma oferecer menus para carregar aplicativos e trabalhar com o desktop. Um desktop completo inclui um gerenciador de janelas, mas também adiciona menus, painéis e, de maneira geral, uma interface de programação de aplicativo que é usada para criar aplicativos que rodam bem juntos.

Assim, de que maneira o entendimento de como as interfaces desktop funcionam no Linux pode ajudá-lo quando se trata de usar o Linux? Eis algumas:

- Como os ambientes de desktop Linux não são obrigados a rodar um sistema Linux, um sistema Linux pode ter sido instalado sem um desktop. Ele pode oferecer apenas uma interface de linha de comando baseada em texto simples. Você pode optar por adicionar um desktop mais tarde. Instalado o desktop, você pode escolher se quer iniciá-lo junto com seu computador ou iniciá-lo conforme necessário.
- Para um sistema Linux muito simples, como um feito para rodar em computadores menos poderosos, você pode escolher um gerenciador de janelas eficiente, mas menos rico em recursos, (como o `twm` ou o `fluxbox`) ou um ambiente desktop leve (como o LXDE ou o Xfce).
- Para computadores mais robustos, você pode escolher ambientes de desktop mais poderosos (como o GNOME e o KDE) que podem fazer coisas como prestar atenção a eventos que podem acontecer (como a inserção de uma unidade flash USB) e responder a esses eventos (como abrir uma janela para visualizar o conteúdo do disco).
- Você pode ter vários ambientes de desktop instalados e pode escolher qual carregar ao fazer login. Dessa forma, diferentes usuários no mesmo computador podem usar diferentes ambientes de desktop.

Muitos diferentes ambientes de desktop estão disponíveis para escolha em Linux. Eis alguns exemplos:

- **GNOME** — O GNOME é o ambiente de desktop padrão para o Fedora, o Red Hat Enterprise Linux e muitos outros. É considerado um desktop profissional, com mais foco na estabilidade do que em efeitos visuais.
- **K Desktop Environment** — O KDE é provavelmente o segundo desktop mais popular para Linux. Ele tem mais penduricalhos que o GNOME e oferece aplicativos mais integrados. O KDE também está disponível com o Fedora, RHEL, Ubuntu e muitos outros sistemas Linux.
- **Xfce** — O desktop Xfce foi um dos primeiros ambientes de desktop “leves”. É bom para usar em computadores antigos ou menos poderosos. Ele está disponível com o RHEL, o Fedora, o Ubuntu

e outras distribuições Linux.

- **LXDE** — O Lightweight X11 Desktop Environment (LXDE) foi projetado para ser um desktop de rápido desempenho e economia de energia. Muitas vezes, o LXDE é usado em dispositivos menos caros, como netbooks e em mídia ao vivo (como um live CD ou live pen drive). É o desktop padrão para a distribuição do Live CD KNOPPIX. Embora o LXDE não venha incluído no RHEL, você pode experimentá-lo com o Fedora ou o Ubuntu.

O GNOME foi originalmente concebido para se parecer com o desktop MAC OS, enquanto o KDE deveria emular o ambiente de desktop do Windows. Como é o desktop mais popular e um dos mais frequentemente usados em sistemas Linux comerciais, a maioria dos procedimentos e exercícios de desktop deste livro usa o desktop GNOME. Usar o GNOME, porém, ainda lhe dá a opção de várias distribuições Linux diferentes.

Começando com o GNOME Fedora Desktop Live CD

Um live CD é a maneira mais rápida de ter um sistema Linux instalado e funcionando para você poder começar a experimentar. Com um live CD de Linux, você pode fazer o Linux assumir a operação do seu computador temporariamente, sem prejudicar o conteúdo de seu disco rígido.

Se você tiver o Windows instalado, o Linux irá simplesmente ignorá-lo e usar a si mesmo para controlar seu computador. Quando terminar de usar o live CD Linux, você pode reiniciar o computador, retirar o CD e voltar a rodar qualquer que seja o sistema operacional instalado no disco rígido.

Para experimentar um desktop GNOME, juntamente com as descrições desta seção, sugiro que você obtenha um Fedora Live CD (conforme descrito no Apêndice A). Como um live CD faz todo seu trabalho a partir do CD e na memória, ele vai rodar mais lento do que um sistema Linux instalado. Além disso, embora você possa alterar os arquivos, adicionar software e de outras maneiras configurar seu sistema, por padrão, o trabalho que você faz desaparece quando você reinicia, a menos que você salve os dados em seu disco rígido ou mídia de armazenamento externa.

O fato de que as alterações feitas no ambiente do live CD desapareçam na reinicialização é muito bom para experimentar o Linux, mas não é tão bom se você quer um desktop ou sistema de servidor para uso contínuo. Por essa razão, recomendo que se você tiver um computador extra, instale o Linux permanentemente no disco rígido dele para usar com o resto deste livro (como descrito no Capítulo 9).

Uma vez que você tem um live CD na mão, faça o seguinte para começar:

1. **Arrume um computador.** Se você tem um PC padrão (de 32 ou 64 bits) com uma unidade de CD/DVD e pelo menos 1 GB de memória (RAM) e 400 MHz, você está pronto para começar.
2. **Inicie o live CD.** Insira o live CD na unidade de CD do computador e reinicie o computador (desligue-o e ligue-o novamente). Dependendo da ordem de boot definida em seu computador, o live CD pode iniciar diretamente pela BIOS (o código que controla o computador antes de o sistema operacional ser iniciado).

ta

em vez de iniciar o live CD, iniciar o sistema operacional instalado, você precisará dar um passo adicional para iniciar o live CD. Reinicie novamente e quando você vir a tela da BIOS, procure algumas palavras que dizem algo como “Boot Order”. As instruções na tela podem dizer para pressionar a tecla de função F12 ou F1. Pressione a tecla imediatamente a partir da tela da BIOS. Então, você verá uma tela que exibe as opções disponíveis. Realce uma opção para CD/DVD e pressione Enter para iniciar o live CD. Se você não vir o disco aí, você pode precisar ir para a configuração da BIOS e ativar a unidade de CD/DVD lá.

3. **Inicie o Fedora.** Se o CD for capaz de inicializar, você verá uma tela de inicialização. No Fedora, com Start Fedora destacado, pressione Enter para iniciar o live CD.
4. **Comece usando o desktop.** Para o Fedora, o live CD inicia diretamente em um desktop GNOME 3 por padrão. Em alguns casos, se o computador não atender às especificações mínimas, o Fedora carrega o GNOME 2 no lugar do 3.

Você pode agora avançar para a próxima seção, “Usando o Desktop GNOME 3” (que inclui informações sobre o uso do GNOME 3 no Fedora e outros sistemas operacionais). A seção seguinte dessa cobrirá o desktop GNOME 2.

Usando o desktop GNOME 3

O desktop GNOME 3 oferece uma mudança radical em relação a seus homólogos GNOME 2.x. Enquanto o GNOME 2.x é prático, o GNOME 3 é elegante. Com o GNOME 3, um desktop Linux agora se parece mais como as interfaces gráficas em dispositivos móveis, com menos foco em múltiplos botões de mouse e combinações de teclas e mais em movimentos do mouse e operações de um clique.

Em vez da aparência estruturada e rígida, o desktop GNOME 3 parece expandir-se à medida que você exige mais dele. Quando um novo aplicativo é executado, seu ícone é adicionado ao Dash. Conforme você usa o espaço de trabalho seguinte, um novo se abre, pronto para que sejam colocados mais aplicativos.

Depois que o computador inicia

No caso de um live CD, o sistema vai direto para o desktop e você ganha o nome de usuário Live System User. No caso de um sistema já instalado, você verá a tela de login, com contas de usuário no sistema pronto para selecionar e digitar uma senha. A Figura 2.1 é um exemplo da tela de login do Fedora.

no desktop GNOME a partir do Fedora.

Há muito pouco no GNOME 3 quando você começa. A barra superior tem a palavra “Activities” à esquerda, um relógio no meio e alguns ícones à direita para coisas como ajustar o volume de áudio, verificar sua conexão de rede e visualizar o nome do usuário atual.

Navegando com o mouse

Para começar, tente navegar pelo desktop GNOME 3 com o mouse:

1. Alterne entre atividades e janelas. Mova o cursor do mouse para o canto superior esquerdo da tela, ao lado do botão Activities. Cada vez que você muda para lá, sua tela muda entre mostrar as janelas que você está usando ativamente e um conjunto de atividades disponíveis. (Isso tem o mesmo efeito de pressionar o botão Windows.)
2. Abra janelas a partir da barra de aplicativos. Clique para abrir alguns aplicativos a partir do Dash à esquerda (Firefox, File Manager, Shotwell ou outros). Mova o mouse para o canto superior esquerdo de novo e alterne entre exibir todas as janelas minimizadas (tela Overview) e mostrá-las sobrepostas (em tamanho completo). A Figura 2.2 mostra um exemplo do modo de exibição de janelas em miniatura.
3. Abra os aplicativos a partir da lista Applications. Na tela Overview, selecione o botão de Application no topo da página. A visualização muda para um conjunto de ícones que representam os aplicativos instalados em seu sistema, como mostrado na Figura 2.3.

ando todas as janelas minimizadas no desktop.

ando a lista de aplicativos disponíveis.

4. **Ver aplicativos adicionais.** A partir da tela Applications, há várias maneiras de mudar o ponto de vista de seus aplicativos, bem como diferentes formas de carregá-los:
 - **Rolagem** — Para ver os ícones que representam os aplicativos que não estão na tela, use o mouse para agarrar e mover a barra de rolagem à direita. Se tiver um mouse de roda, você pode usá-lo para rolar os ícones.
 - **Grupos de aplicativos** — Selecione um grupo de aplicativos à direita (Accessories, Games, Graphics etc.) para ver os aplicativos que estão somente nesse grupo.
 - **Carregando um aplicativo** — Para iniciar o aplicativo desejado, clique no ícone para abrir o aplicativo no espaço de trabalho atual. Se o mouse tiver um botão do meio, você pode clicar com esse botão em um aplicativo para abri-lo em um novo espaço de trabalho. Clique com o botão direito do mouse para abrir um menu contendo instâncias abertas desse aplicativo, você pode selecionar uma opção para abrir uma seleção New Window e uma opção para adicionar ou remover o aplicativo de Favorites (para o ícone do aplicativo aparecer no Dash). A Figura 2.4 mostra um exemplo do menu.

e com o botão do meio do mouse para exibir o menu de seleção de um aplicativo.

5. Abra outros aplicativos. Inicie os aplicativos adicionais. Note que quando você abrir um novo aplicativo, um ícone que o representa aparece na barra Dash à esquerda. Eis algumas outras formas de iniciar aplicativos:
 - **Ícone de aplicativo** — Clique em qualquer ícone de aplicativo para abrí-lo.
 - **Soltar ícones Dash no espaço de trabalho** — Do ponto de vista do Windows, você pode arrastar qualquer ícone de aplicativo a partir do Dash, segurando a tecla Ctrl e arrastando o ícone para qualquer um dos espaços de trabalho em miniatura à direita.
6. Use múltiplos espaços de trabalho. Mova o mouse para o canto superior esquerdo de novo para mostrar uma visão de todas as janelas minimizadas. Observe todos os aplicativos à direita comprimidos em uma pequena representação de um espaço de trabalho enquanto um espaço de trabalho adicional está vazio. Arraste e solte duas das janelas para o espaço do desktop vazio. A Figura 2.5 mostra a aparência dos espaços de trabalho em miniatura. Observe que um espaço de trabalho adicional vazio é criado cada vez que o último vazio é usado. Você pode arrastar e soltar as janelas em miniatura entre qualquer espaço de trabalho e selecionar o espaço de trabalho para visualizá-lo.

didada que novos desktops são usados, os adicionais aparecem à direita.

7. Use o menu de janela. Mova o mouse para o canto superior esquerdo da tela a fim de retornar ao espaço de trabalho ativo (vista da janela grande). Clique com o botão direito do mouse na barra de título de uma janela para ver o menu dela. Tente essas ações desse menu:
 - **Minimize** — Tira temporariamente a janela de vista.
 - **Maximize** — Expande a janela para o tamanho máximo.
 - **Move** — Muda a janela para o modo de movimentação. Mover o mouse move a janela. Clique para fixar a janela em um ponto.
 - **Resize** — Altera a janela para o modo de redimensionamento. Mover o mouse redimensiona a janela. Clique para manter o tamanho.
 - **Opções dos espaços de trabalho** — Várias opções permitem que você use os espaços de trabalho de diferentes maneiras. Escolha fazer a janela ficar sempre por cima de outras janelas, visível em cada espaço de trabalho ou apenas no espaço de trabalho atual. Ou mova a janela para outro espaço de trabalho, o de cima ou o de baixo.

Se você não se sentir confortável navegando pelo GNOME 3 com o mouse, ou se você não tiver um mouse, a próxima seção ajuda você a navegar pelo desktop a partir do teclado.

Navegando com o teclado

Se preferir manter as mãos no teclado, você pode trabalhar com o desktop GNOME 3 diretamente a partir do teclado de várias maneiras, incluindo as seguintes:

- **Tecla Windows** — Pressione a tecla Windows no teclado. Na maioria dos teclados de PC, essa é a tecla com o logotipo do Microsoft Windows, ao lado da tecla Alt. Isso alterna entre os pontos de vista de minijanela (Overview) e janela ativa (espaço de trabalho atual). Muitas pessoas usam bastante essa tecla.
- **Selecione diferentes pontos de vista** — A partir da visualização Windows ou Applications, segure Ctrl+Alt+Tab para ver um menu de diferentes pontos de vista (ver Figura 2.6). Ainda segurando as teclas Ctrl+Alt Tab, pressione mais uma vez para destacar um dos ícones a seguir no menu e solte para selecioná-lo:
 - **Top Bar** — Mantém a visualização atual.
 - **Dash** — Destaca o primeiro aplicativo na barra de aplicativos à esquerda. Use as setas para mover para cima e para baixo esse menu e pressione Enter para abrir o aplicativo destacado.
 - **Windows** — Seleciona o ponto de vista de janelas.
 - **Applications** — Seleciona o ponto de vista de aplicativos.

- **Search** — Destaca a caixa de pesquisa. Digite algumas letras para mostrar somente ícones de aplicativos que contêm as letras digitadas. Quando você tiver digitado letras suficientes para identificar o aplicativo que você deseja, pressione Enter para iniciar o aplicativo.

ione Ctrl+Alt+Tab a fim de exibir áreas de desktop adicionais para selecionar.

- **Selecione uma janela ativa** — Retorna a qualquer um dos seus espaços de trabalho (pressione a tecla Windows, se você já não estiver em um espaço de trabalho ativo). Pressione Alt+Tab para ver uma lista de todas as janelas ativas (ver Figura 2.7). Continue a segurar a tecla Alt ao pressionar a tecla Tab (ou a seta para a direita ou para a esquerda) a fim de realçar o aplicativo que você deseja na lista de janelas de aplicativos de desktop ativos. Se um aplicativo tiver várias janelas abertas, pressione Alt + ` (crase) para escolher uma das subjanelas. Solte a tecla Alt para selecioná-la.

ione Alt+Tab para selecionar o aplicativo em execução que você quer ativar.

- **Carregue um comando ou um aplicativo** — A partir de qualquer espaço de trabalho ativo, você pode carregar um comando Linux ou um aplicativo gráfico. Eis alguns exemplos:
 - **Aplicações** — Na tela Overview, pressione Ctrl+Alt+Tab e depois continue a pressionar Tab até que o ícone de engrenagens (Applications) seja destacado; então solte as teclas Ctrl+Alt. A visualização Applications aparece, com o primeiro ícone em destaque. Use a tecla Tab ou as teclas de seta (para cima, para baixo, para a direita e para a esquerda) a fim de destacar o ícone do aplicativo que você quer e pressione Enter.
 - **Caixa de comando** — Se você sabe o nome de um comando que deseja executar, pressione Alt+F2 para exibir uma caixa de comando. Digite o nome do comando na caixa (experimente **system-config-date** para ajustar a data e hora, por exemplo).
 - **Caixa de pesquisa** — Na tela Overview, pressione Ctrl+Alt+Tab e depois continue a pressionar Tab até o ícone de lupa (Search) ser realçado, então solte as teclas Ctrl+Alt. Na caixa de pesquisa agora destacada, digite algumas letras do nome de um aplicativo ou a descrição (digite **scr** para ver o que acontece). Continue digitando até que o aplicativo desejado seja destacado (nesse caso, Screenshot) e pressione Enter para iniciá-lo.
 - **Dash** — Na tela Overview, pressione Ctrl+Alt+Tab e então continue a pressionar Tab até que o ícone de estrela (Dash) seja destacado; em seguida, solte a tecla Ctrl+Alt. A partir do Dash, mova para cima e para baixo a fim de realçar um aplicativo que você pretende carregar e pressione Enter.
 - **Escape** — Quando você estiver preso em uma ação que não deseja concluir, tente pressionar a tecla Esc. Por exemplo, depois de pressionar Alt+F2 (para introduzir um comando), abrir um ícone da barra superior, ou ir a uma página de visão geral, pressionar Esc retorna para a janela ativa no desktop ativo.

Espero que agora você se sinta confortável navegando pelo desktop do GNOME 3. A seguir, você pode experimentar executar alguns aplicativos desktop úteis e divertidos do GNOME 3.

Configurando o desktop GNOME 3

Muito do que você precisa que o GNOME 3 faça por você é configurado automaticamente. Mas há alguns ajustes que você vai querer fazer para que o desktop fique da maneira como prefere. A maioria dessas atividades de configuração está disponível na janela System Settings (ver Figura 2.8). Abra o ícone System Settings a partir da tela Applications.

ando as configurações do desktop a partir da janela System Settings.

Eis algumas sugestões para configurar um desktop GNOME 3:

- **Configurando a rede** — Uma conexão de rede com fio geralmente é configurada automaticamente quando você carrega seu sistema Fedora. Para redes sem fio, você provavelmente terá de selecionar sua rede sem fio e adicionar uma senha quando solicitado. Um ícone na barra superior permite fazer qualquer configuração de rede com ou sem fio que você precise fazer. Consulte o Capítulo 14, “Administração de Redes”, para obter mais informações sobre configuração de rede.
- **Personal settings** — Ferramentas nesse grupo permitem alterar o fundo do desktop (Background), usar diferentes contas online (Online Accounts) e definir o idioma e o formato de data e moeda com base na região (Region e Language) e bloqueio de tela (Screen). Para alterar seu fundo, abra a janela System Settings, selecione Background e então selecione um dos papéis de parede disponíveis. Para adicionar seu próprio fundo, baixe um papel de parede do seu gosto para sua pasta Pictures e clique na caixa de Wallpapers para colocá-la na pasta Pictures e escolha a imagem que você quiser.
- **Bluetooth** — Se seu computador tiver hardware Bluetooth, você pode ativar esse dispositivo para se comunicar com outros dispositivos Bluetooth.
- **Printers** — Em vez de usar a janela System Settings para configurar uma impressora, consulte o Capítulo 16, “Configurando um servidor de impressão”, para obter informações sobre como configurar uma impressora usando o serviço CUPS.
- **Sound** — Clique no botão configurações de som para ajustar os dispositivos de entrada e saída de som em seu sistema.
- **Removable media** — Para configurar o que acontece quando CDs, DVDs, leitores de música ou outras mídias removíveis são inseridas em seu computador, selecione o ícone Removable Media. Consulte o Capítulo 8, “Aprendendo administração de sistema”, para obter informações sobre configuração de mídia removível.

Estendendo o desktop GNOME 3

Se o shell do GNOME 3 não faz tudo o que você quer, não se desespere. Você pode adicionar extensões para fornecer funcionalidades adicionais ao GNOME 3. Também há uma ferramenta GNOME Tweak que permite alterar configurações avançadas no GNOME 3.

O uso de extensões GNOME Shell

Extensões de shell para o GNOME estão disponíveis para alterar a maneira como o desktop GNOME aparece e se comporta. Visite o site GNOME Shell Extensions (<http://extensions.gnome.org>)

de seu navegador Firefox em seu desktop GNOME 3 e ele informará qual extensão você tem instalada e quais estão disponíveis para instalar.

Como a página de extensões sabe quais extensões você tem e a versão do GNOME 3 que está em execução, ela pode apresentar somente as extensões que são compatíveis com seu sistema. Muitas das extensões ajudam-no a adicionar de volta recursos do GNOME 2, incluindo:

- **Menu Applications** — Adiciona um menu Applications ao painel superior, exatamente como era no GNOME 2.
- **Indicador de status de locais** — Adiciona um menu de status de sistemas, semelhante ao menu Places no GNOME 2, para que você navegue rapidamente para pastas úteis em seu sistema.
- **Lista de janelas** — Adiciona uma lista de janelas ativas ao painel superior, semelhante à Window List que aparecia no painel inferior do GNOME 2.

Para instalar uma extensão, basta selecioná-la na lista a fim de ver a página dela e, então, clicar no botão nessa página para ativá-lo. Clique em Install quando for perguntado se deseja baixar e instalar a extensão. A extensão é então adicionada ao seu desktop.

A Figura 2.9 mostra um exemplo do menu Applications (o ícone em forma de pé do GNOME), a Window List (mostrando vários ícones de aplicativos ativos) e o Places Status Indicator (com pastas exibidas a partir de um menu drop-down).

sões adicionam recursos ao desktop GNOME 3.

Mais de 100 extensões de shell para o GNOME já estão disponíveis e mais estão sendo adicionadas o tempo todo. Outras extensões populares incluem Notifications Alert (que o alertam de mensagens não lidas), o Presentation Mode (que impede que o protetor de tela seja ativado quando você está fazendo uma apresentação) e Music Integration (que integra tocadores de músicas populares ao GNOME 3 para que você seja alertado sobre músicas sendo tocadas).

Como o site de extensões pode monitorar suas extensões, você pode clicar no botão Installed extensions no topo da página e ver todas as extensões que estão instaladas. Você pode ativar e desativar extensões e até excluí-las permanentemente.

Usando o GNOME Tweak Tool

Se você não gosta do jeito que algumas das características internas do GNOME 3 se comportam, é possível alterar muitas delas com o GNOME Tweak Tool. Essa ferramenta não é instalada por padrão com o Fedora GNOME Live CD, mas você pode adicioná-lo ao instalar o pacote `gnome-tweak-tool` (Consulte o Capítulo 10, “Obtendo e gerenciando o software”, a fim de obter informações sobre como instalar pacotes de software no Fedora.)

Uma vez instalado, o GNOME Tweak Tool está disponível clicando-se no ícone Advanced Settings a partir de tela Applications. Comece com a categoria Desktop para considerar o que você pode querer mudar no GNOME 3. A Figura 2.10 mostra a Tweak Tool (janela Advanced Settings) mostrando as configurações do desktop.

ando as configurações de desktop com o GNOME Tweak Tool (Advanced Settings).

Se você está acostumado a colocar arquivos e pastas na sua área de trabalho, pode clicar na opção “Have file manager handle the desktop” (Deixar o gerenciador de arquivos lidar com a área de trabalho). Você será imediatamente capaz de abrir pastas e arrastar e soltar arquivos para o desktop. Se as fontes forem muito pequenas para você, selecione a categoria Fonts e arraste o fator de escala de texto para aumentar o tamanho da fonte. Ou altere fontes individualmente para documentos, títulos de janelas ou fontes monoespaciais.

Em Shell settings, você pode alterar como as informações do relógio são exibidas na barra superior ou configurar o que acontece quando você fecha a tampa do laptop (suspender, hibernar e assim por diante). Para alterar a aparência da área de trabalho, selecione a categoria Theme e mude o tema Icon e o tema GTK+ como você quiser a partir das caixas suspensas.

Começando com aplicativos desktop

O desktop GNOME 3 do live CD do Fedora vem com alguns aplicativos interessantes que você pode começar a usar imediatamente. Para usar o GNOME 3 como desktop todos os dias, você deve instalá-lo permanentemente no disco rígido do seu computador e adicionar os aplicativos de que precisa (um processador de texto, um editor de imagem, um aplicativo de desenho etc.). Se você estiver apenas começando, as seções a seguir listam alguns aplicativos interessantes para experimentar.

Gerenciando arquivos e pastas com o Nautilus

Para mover, copiar, excluir, renomear e de outra maneira organizar os arquivos e pastas no GNOME 3, você pode usar o gerenciador de arquivos Nautilus. O Nautilus vem com o desktop GNOME e funciona como os gerenciadores de arquivos que você pode usar no Windows ou no Mac.

Para abrir o Nautilus, clique no ícone Files a partir do Dash do GNOME ou da lista de aplicativos. Sua conta de usuário começa com um conjunto de pastas destinadas a armazenar os tipos mais comuns de conteúdo: Músicas, Fotos, Vídeos etc. Esses são todos armazenados no que é referido como seu diretório Home. A Figura 2.11 mostra o Nautilus aberto em um diretório home.

icie arquivos e pastas a partir da janela do Nautilus.

Quando quiser salvar os arquivos que você baixou da Internet ou criou com um processador de texto, você pode organizá-los nessas pastas. Você pode criar novas pastas, conforme necessário, arrastar e soltar arquivos e pastas para copiar, movê-los e excluí-los.

Como o Nautilus não é muito diferente da maioria dos gerenciadores de arquivo que você já usou em outros sistemas de computador, este capítulo não entra em detalhes sobre como usar o arrastar e soltar e percorrer pastas para encontrar seu conteúdo. Mas quero fazer algumas observações que podem não ser óbvias sobre como usar o Nautilus:

- **Pasta Home** — Você tem controle total sobre os arquivos e pastas que cria em sua pasta pessoal. A maioria das outras partes do sistema de arquivos não é acessível a você como um usuário regular.
- **Organização do sistema de arquivos** — Embora apareça sob o nome Home, a pasta home encontra-se na verdade no sistema de arquivos sob a pasta `/home` em uma pasta nomeada com base no seu nome de usuário — por exemplo, `/home/liveuser`. Nos próximos capítulos, você vai aprender como o sistema de arquivos é organizado (especialmente em relação ao shell de comando do Linux).
- **Trabalhando com arquivos e pastas** — Clique com o botão direito do mouse em um arquivo ou ícone de pasta para ver como você pode agir sobre ela. Por exemplo, você pode copiar, cortar, mover para a lixeira (excluir), ou abrir qualquer ícone de arquivo ou pasta.
- **Criando pastas** — Para criar uma nova pasta, clique com o botão direito do mouse em uma janela de pasta e selecione **Create New Folder**. Digite o novo nome da pasta sobre o destaque **Untitled Folder** e pressione **Enter** para nomear a pasta.
- **Acessando conteúdo remoto** — O Nautilus pode exibir conteúdo de servidores remotos, bem como o sistema de arquivos local. No Nautilus, selecione **Connect to Server** no menu **File**. Você pode se conectar a um servidor remoto via **SSH (Secure Shell)**, **FTP com login**, **FTP Público**, **compartilhamento Windows**, **WebDav (HTTP)**, ou **Secure WebDav (HTTPS)**. Adicione o nome de usuário e senha apropriados conforme necessário e o conteúdo do servidor remoto aparece na janela do Nautilus. A Figura 2.12 mostra um exemplo de uma janela do Nautilus exibindo pastas a partir de um servidor remoto por meio do protocolo SSH.

se pastas remotas usando o recurso Nautilus Connect to Server.

Instalando e gerenciando software adicional

O Fedora Live CD vem com um navegador (Firefox), um gerenciador de arquivos (Nautilus) e alguns outros aplicativos comuns. Mas existem muitos outros aplicativos úteis que, por causa de seu tamanho, simplesmente não caberiam em um live CD. Se você instalar o live CD em seu disco rígido (como descrito no Capítulo 9), certamente vai querer acrescentar alguns outros softwares.

ta

tente instalar um software se você estiver executando o live CD. Como o espaço gravável de um live CD usa a memória virtual (RAM), esse espaço é limitado e pode facilmente acabar. Além disso, quando reiniciar o sistema, qualquer coisa que você instalar terá desaparecido.

Quando o Fedora é instalado, ele é automaticamente configurado para conectar seu sistema ao enorme repositório de software Fedora que está disponível na internet. Desde que você tenha uma conexão internet, você pode executar a ferramenta Add/Remove software para baixar e instalar qualquer um dos milhares de pacotes do Fedora.

Apesar de toda a facilidade para o gerenciamento de software no Fedora (os recursos `yum` e `rpm`) ser descrita em detalhes no Capítulo 10, “Obtendo e gerenciando software”, você pode começar a instalar alguns pacotes de software sem saber muito sobre a forma como o recurso funciona. Comece indo para a tela de aplicativos e abrindo a janela Add/Remove Software.

Com a janela Add/Remove Software aberta, você pode selecionar os aplicativos que deseja instalar, pesquisar (digite o nome na caixa Find), escolher uma categoria, ou classificar pacotes e selecionar a partir de uma lista (por coleções, novos pacotes ou pacotes selecionados). A Figura 2.13 mostra uma pesquisa para o pacote LibreOffice Writer, com ele selecionado e pronto para ser instalado.

e instale o software a partir do repositório Fedora.

Com os pacotes que deseja selecionados, você pode ler uma descrição deles e até mesmo ir para suas respectivas homepages para ler mais sobre o assunto. Quando estiver pronto, clique em Apply para instalar o pacote e quaisquer pacotes dependentes necessários para fazê-lo funcionar.

Ao pesquisar e instalar alguns aplicativos de desktop comuns, você deve ser capaz de começar a utilizar o desktop de forma eficaz. Consulte o Capítulo 10 para obter detalhes sobre como adicionar repositórios de software e usar os comandos `yum` e `rpm` para gerenciar o software no Fedora e no RHEL.

Reproduzindo música com o Rhythmbox

O Rhythmbox é o tocador de música que vem no Fedora GNOME Live CD. Você pode iniciar o Rhythmbox do Dash do GNOME 3 e então imediatamente reproduzir CDs de música, podcasts ou shows de rádio da internet. Você pode importar arquivos de áudio nos formatos WAV e Ogg Vorbis, ou adicionar plug-ins para MP3 ou outros formatos de áudio.

A Figura 2.14 apresenta um exemplo da janela Rhythmbox com vários CDs de áudio importados.

de música, podcasts e rádios da internet a partir do Rhythmbox.

Eis algumas maneiras como você pode começar com o Rhythmbox:

- **Rádio** — Dê um duplo clique na opção Radio sob Library. Então, escolha uma estação de rádio na lista que aparece à direita.
- **Podcasts** — Procure podcasts na internet e encontre a URL daquele que lhe interessa. Clique com o botão direito do mouse na entrada Podcasts e selecione New Podcast Feed. Cole ou digite a URL do podcast e clique em Add. Uma lista de podcasts do site selecionado aparece à direita. Dê um duplo clique sobre o que você quer ouvir.
- **CDs de áudio** — Insira um CD de áudio e pressione Play quando ele aparecer na janela do Rhythmbox. O Rhythmbox também permite copiar e gravar CDs de áudio.
- **Arquivos de áudio** — O Rhythmbox pode tocar arquivos WAV e Ogg Vorbis. Ao adicionar plug-ins, você pode tocar muitos outros formatos de áudio, incluindo MP3. Como há questões de patentes relacionadas com o formato MP3, a capacidade de reproduzir arquivos MP3 não vem incluída no Fedora. No Capítulo 10, descrevo como obter softwares de que você precisa mas que não estão no repositório de sua distribuição Linux.

Existem plug-ins para o Rhythmbox que servem para obter a arte da capa, mostrar informações sobre artistas e músicas, adicionar suporte a serviços de música (como Last.fm e Magnatune) e buscar letras de música.

Parando o desktop GNOME 3

Quando você terminar uma sessão do GNOME 3, selecione o botão de usuário no canto superior direito da barra superior. A partir daí, você pode optar por sair, suspender a sessão ou mudar para uma conta de usuário diferente sem fazer logout.

Usando o desktop GNOME 2

O desktop GNOME 2 é a interface de desktop padrão utilizada por todo o Red Hat Enterprise Linux 6. Ele é bem conhecido, estável e, talvez, um pouco entediante.

Os desktops GNOME 2 fornecem os menus, painéis, ícones e espaços de trabalho padrões. Se você estiver usando um sistema Red Hat Enterprise Linux até o RHEL6 ou uma antiga distribuição Fedora ou Ubuntu, provavelmente está olhando para um desktop GNOME 2. Assim, esta seção oferece um tour pelo GNOME 2, juntamente com algumas oportunidades para enfeitá-lo um pouco.

Na época em que escrevíamos isto, o GNOME 2.38 era a versão mais recente disponível no Red Hat

Enterprise Linux, embora a distribuição que você está usando possa ou não incluir essa versão. Os recentes lançamentos do GNOME incluem avanços em efeitos 3D (ver “Efeitos 3D com o AIGLX” mais adiante neste capítulo) e recursos de usabilidade melhorados.

Para utilizar sua área de trabalho do GNOME, você deve familiarizar-se com os seguintes componentes:

- **Metacity (gerenciador de janelas)** — O gerenciador de janelas padrão para o GNOME 2.2 é o Metacity. As opções de configuração do Metacity permitem controlar coisas como temas, bordas de janelas e controles usados em seu desktop.
- **Compiz (gerenciador de janelas)** — Você pode ativar esse gerenciador de janelas do GNOME para fornecer efeitos de desktop 3D.
- **Nautilus (gerenciador de arquivos/shell gráfico)** — Quando você abre uma pasta (clcando duas vezes no ícone Home no desktop, por exemplo), a janela do Nautilus se abre e exibe o conteúdo da pasta selecionada. O Nautilus também pode exibir outros tipos de conteúdo, como pastas compartilhadas de computadores Windows na rede (usando o SMB).
- **Painéis GNOME (carregador de aplicativo/tarefa)** — Esses painéis, que definem uma linha superior e uma inferior na sua tela, são projetados para tornar mais conveniente para você iniciar os aplicativos que usa, gerenciar a execução deles e trabalhar com múltiplos desktops virtuais. Por padrão, o painel superior contém botões de menu (Applications, Places e System), carregadores de aplicativos desktop (o programa de e-mail Evolution e o navegador Firefox), um alternador de espaço de trabalho (para gerenciar quatro desktops virtuais) e um relógio. Os ícones aparecem no painel quando você precisa de atualizações de software ou o SELinux detecta um problema. O painel inferior tem um botão Show Desktop, listas de janelas, uma lata de lixo e um alternador de espaço de trabalho.
- **Área do Desktop** — As janelas e ícones que você usa são dispostos no desktop, que suporta arrastar e soltar entre aplicativos, um menu de desktop (clique com o botão direito do mouse para vê-lo) e ícones para carregar aplicativos. Um ícone de computador consolida unidades de CD, unidades de disquete e o sistema de arquivos e recursos de rede compartilhados em um só lugar.

O GNOME também inclui um conjunto de janelas de preferências que permitem configurar diferentes aspectos do seu desktop. Você pode mudar fundos, cores, fontes, atalhos de teclado e outras características relacionadas com a aparência e o comportamento do desktop. A Figura 2.15 mostra como o desktop GNOME 2 aparece na primeira vez que você faz login, com algumas janelas adicionadas à tela.

O desktop mostrado na Figura 2.15 é para o Red Hat Enterprise Linux. As seções a seguir fornecem detalhes sobre o uso do desktop GNOME 2.

Utilizando o gerenciador de janelas Metacity

O gerenciador de janelas Metacity parece ter sido escolhido como o gerenciador de janelas padrão para o GNOME por causa de sua simplicidade. O criador do Metacity se refere a ele como um “gerenciador de janelas chato para o adulto em você” e, então, passa a comparar outros gerenciadores de janelas com cereais de açúcar colorido, enquanto o Metacity é caracterizado como um cereal comum de milho.

Para usar efeitos 3D, a melhor solução é usar o gerenciador de janelas Compiz, descrito mais adiante neste capítulo.

Não há realmente muito que você possa fazer com o Metacity (exceto fazer seu trabalho de forma eficiente). Você atribui novos temas ao Metacity e muda cores e decorações de janelas por meio das preferências do GNOME (descritas mais adiante). Existem alguns poucos temas do Metacity, mas esperamos que o número cresça.

As funções básicas do Metacity que podem interessar a você são atalhos do teclado e o alternador de espaço de trabalho. A Tabela 2.1 mostra os atalhos de teclado para lidar com o gerenciador de janelas Metacity.

Tabela 2.1 Atalhos de Teclado

Função	Tecla de atalho
Alternar entre os aplicativos, sem ícones de pop-up	Alt+Shift+Esc
Alternar entre painéis	Alt+Ctrl+Shift+Tab
Mostrar o menu	Esc

Existem muitos outros atalhos de teclado que você pode usar com o gerenciador de janelas. Selecione System ⇒ Preferences ⇒ Keyboard Shortcuts para ver uma lista de atalhos, como os seguintes:

- **Executar diálogo** — Para executar um comando a fim de iniciar um aplicativo de desktop pelo nome de comando, pressione Alt+F2. Na caixa de diálogo que aparece, digite o comando e pressione Enter. Por exemplo, digite **gedit** para executar um editor de texto gráfico simples.
- **Bloqueio de tela** — Se você quer se afastar de sua tela e bloqueá-la, pressione Ctrl+Alt+L. Você

terá de digitar sua senha de usuário para abrir a tela novamente.

- **Exibir o menu principal** — Para abrir um aplicativo a partir do menu Applications, Places ou System, pressione Alt+F1. Então, use as teclas de seta para cima e para baixo a fim de selecionar uma opção a partir do menu atual ou as teclas de seta para a direita e para a esquerda a fim de selecionar a partir de outros menus.
- **Print Screen** — Pressione a tecla Print Screen para capturar uma tela de todo o desktop. Pressione Alt+Print Screen para capturar uma tela da janela atual.

Outra característica de interesse do Metacity é o alternador de espaço de trabalho. Quatro espaços de trabalho virtuais aparecem no Workspace Switcher no painel do GNOME 2. Você pode fazer o seguinte com o Workspace Switcher:

- **Escolher o espaço de trabalho atual** — Quatro espaços de trabalho virtuais aparecem no Workspace Switcher. Clique em qualquer um deles para torná-lo seu espaço de trabalho atual.
- **Mover janelas para outros espaços de trabalho** — Clique em qualquer janela, cada uma representada por um retângulo minúsculo em um espaço de trabalho, para arrastar e soltar para outro espaço de trabalho. Da mesma forma, você pode arrastar um aplicativo a partir da Window List para movê-lo para outro espaço de trabalho.
- **Adicionar mais espaços de trabalho** — Clique com o botão direito do mouse no Workspace Switcher e selecione Preferences. Você pode adicionar mais espaços de trabalho (até 32).
- **Nomear espaços de trabalho** — Clique com o botão direito do mouse no Workspace Switcher e selecione Preferences. Clique no painel Workspaces para alterar os nomes de espaços de trabalho para qualquer nome que você escolher.

Você pode visualizar e alterar informações sobre controles e configurações do Metacity usando a janela do `gconf-editor` (digite `gconf-editor` em uma janela Terminal). Como a janela diz, não é a maneira recomendada para alterar as preferências, por isso, quando possível, você deve mudar o desktop por meio das preferências do GNOME 2. Entretanto, o `gconf-editor` é uma boa maneira de ver as descrições de cada recurso do Metacity.

A partir da janela do `gconf-editor`, selecione `apps ⇒ metacity` e depois escolha `general`, `global_keybindings`, `keybindings_commands`, `window_keybindings` e `workspace_names`. Clique em cada chave para ver seu valor, juntamente com descrições breves e longas da chave.

Alterando a aparência do GNOME

Você pode alterar a aparência geral do seu desktop GNOME selecionando `System ⇒ Preferences < Appearance`. A partir da janela Appearance Preferences, selecione uma das três diferentes abas:

- **Theme** — Para o desktop GNOME 2, estão disponíveis temas inteiros que mudam as cores, ícones, fontes e outros aspectos do desktop. Junto com o desktop GNOME, vêm vários temas diferentes, os quais você pode simplesmente selecionar a partir dessa guia para usar. Ou clique em `Get more themes online` para escolher entre uma variedade de temas disponíveis.

- **Background** — Para alterar o fundo de tela, selecione a partir de uma lista de fundos dessa guia e aplique imediatamente o efeito escolhido. Para adicionar um fundo diferente, coloque o fundo que você quer em seu sistema (talvez baixe um selecionando [Get more backgrounds online](#) e o colocando na pasta `Pictures`). Então, clique em `Add` e selecione a imagem da sua pasta `Pictures`.
- **Fonts** — Fontes diferentes podem ser selecionadas para usar por padrão em aplicativos e documentos, no desktop, na barra de título da janela e para largura fixa.

Usando os painéis GNOME

Os painéis GNOME são colocados na parte superior e na parte inferior do desktop GNOME. A partir desses painéis, você pode iniciar aplicativos (botões ou menus), ver os programas que estão ativos e monitorar como o sistema está funcionando. Existem também muitas maneiras de mudar os painéis superior e inferior — adicionando aplicativos ou monitores, alterando o posicionamento ou o comportamento do painel, por exemplo.

Clique com o botão direito do mouse em qualquer espaço aberto em cada painel para ver o menu do painel. A Figura 2.16 mostra o menu do painel na parte superior.

Menu do painel do GNOME

A partir do menu do painel do GNOME, você pode escolher uma entre várias funções, incluindo:

■ Use the menus

- O menu Applications exibe a maioria dos aplicativos e ferramentas de sistema que você irá usar a partir do desktop.
- O menu Places permite selecionar lugares para ir, como a pasta Desktop, a pasta inicial, mídia removível ou locais de rede.
- O menu System permite alterar as preferências e configurações do sistema, bem como obter outras informações sobre o GNOME.

■ **Add to Panel** — Adicionar um applet, menu, carregador, gaveta ou botão.

■ **Properties** — Muda a posição, tamanho e propriedades de fundo do painel.

■ **Delete This Panel** — Exclui o painel atual.

■ **New Panel** — Adiciona painéis ao desktop em diferentes estilos e localizações.

Você também pode trabalhar com itens em um painel. Por exemplo, você pode:

■ **Mover itens** — Para mover um item em um painel, clique com o botão direito do mouse, selecione Move e então arraste e solte para uma nova posição.

■ **Redimensionar itens** — Você pode redimensionar alguns elementos, tais como a lista de janelas, clicando em uma borda e arrastando-a para o novo tamanho.

■ **Usar a lista de janelas** — Tarefas em execução no desktop aparecem na área da lista de janelas. Clique em uma tarefa para minimizar ou maximizar.

As seções a seguir descrevem algumas coisas que você pode fazer com o painel do GNOME.

Usando os menus Applications e System

Clique em Applications no painel e você verá categorias de aplicativos e ferramentas de sistema que você pode selecionar. Clique no aplicativo que você deseja iniciar. Para adicionar um item de um menu a fim de poder iniciar a partir do painel, arraste e solte o item que você quer para o painel.

Você pode adicionar itens aos menus do seu GNOME 2. Para fazer isso, clique com botão direito em qualquer um dos nomes de menu e selecione Edit Menus. A janela que aparece permite adicionar ou excluir menus associados com os menus Applications e System. Você também pode adicionar itens para carregar a partir desses menus, selecionando New Item e digitando o nome, comando e comentário para o item.

Adicionando um miniaplicativo (applet)

Você pode executar vários pequenos aplicativos, chamados *applets*, diretamente no painel do GNOME. Esses aplicativos são capazes de mostrar informações que você queira ver constantemente ou apenas proporcionar alguma diversão. Para ver quais miniaplicativos estão disponíveis e para adicionar os miniaplicativos que você quer ao seu painel, execute os seguintes passos:

1. Clique com o botão direito do mouse em uma área vazia no painel de modo que o menu Panel apareça.
2. Clique em Add to Panel. Uma janela Add to Panel aparece.
3. Selecione uma opção entre as várias dezenas de applets, incluindo um relógio, pesquisa de dicionário, cotações da bolsa de valores (*stock ticker*) e previsão do tempo. O applet selecionado aparece no painel, pronto para você usar.

A Figura 2.17 mostra (da esquerda para a direita) o applet dos “olhinhos”, o monitor de sistema, a previsão do tempo, o terminal e Wanda, o peixe.

Mover applets no painel facilita o acesso a eles.

Depois que um applet está instalado, clique com o botão direito do mouse no painel para ver quais opções estão disponíveis. Por exemplo, selecione Preferences para o stock ticker e você pode adicionar ou excluir ações cujos preços você queira monitorar. Se você não gostar da localização do applet, clique com o botão direito do mouse, clique em Move, deslize o mouse até o applet estar onde você o quer (mesmo para outro painel) e clique para definir sua localização.

Se você não quiser mais que um miniaplicativo apareça no painel, clique com o botão direito do mouse nele e depois clique em Remove From Panel. O ícone que representa o applet desaparece. Se achar que falta espaço em seu painel, você pode adicionar um novo painel a outra parte da tela, como descrito na próxima seção.

Adicionando outro painel

Se ficar sem espaço nos painéis superior ou inferior, você pode adicionar mais painéis ao seu desktop. É possível ter vários painéis em seu desktop GNOME 2. Você pode adicionar painéis que se estendem ao longo da parte superior, ao longo da parte inferior ou ao longo de um dos lados da tela. Para adicionar um painel, faça o seguinte:

1. Clique com o botão direito do mouse em uma área vazia no painel de modo que o menu Panel apareça.
2. Clique em New Panel. Um novo painel aparece no lado direito da tela.
3. Clique com o botão direito do mouse em um espaço aberto no novo painel e selecione Properties.
4. A partir de Panel Properties, selecione onde você deseja que o painel fique a partir da caixa Orientation (Top, Bottom, Left ou Right – Em cima, Em Baixo, Esquerda ou Direita).

Depois de adicionar um painel, você pode adicionar applets ou carregadores de aplicativos a ele como você fez com o painel padrão. Para remover um painel, clique com o botão direito do mouse nele e selecione Delete This Panel.

Adicionando um launcher de aplicativo

Ícones em seu painel representam um navegador web e aplicativos de escritório diversos. Você pode adicionar seus próprios ícones para carregar aplicativos do painel também. Para adicionar um novo launcher de aplicativo no painel, faça o seguinte:

1. Clique com o botão direito do mouse em uma área vazia no painel.
2. Clique em Add to Panel ⇒ Application Launcher a partir do menu. Todas as categorias de aplicativos de seus menus Applications e System aparecem.
3. Selecione a seta ao lado da categoria de aplicativo que você quer e então, selecione Add.

Um ícone representando o aplicativo aparece no painel.

Para iniciar o aplicativo que você acabou de adicionar, basta clicar no ícone que aparece no painel.

Se o aplicativo que você deseja carregar não estiver em um dos seus menus, você pode construir um carregador da seguinte maneira:

1. Clique com o botão direito do mouse em uma área vazia no painel.
2. Clique em Add to Panel ⇒ Custom Application Launcher ⇒ Add. A janela Create Launcher aparece.
3. Forneça as seguintes informações para o aplicativo que você quer adicionar:
 - **Type** — Selecione Application (para carregar um aplicativo gráfico normal) ou Application in Terminal. Use Application in Terminal se o aplicativo for baseado em texto ou for o ncurses. (Aplicativos escritos usando a biblioteca ncurses rodam em uma janela de Terminal, mas oferecem um mouse e controles pelo teclado.)
 - **Name** — Um nome para identificar o aplicativo (isso aparece na dica de ferramenta quando o mouse está sobre o ícone).
 - **Command** — A linha de comando que é executada quando o aplicativo é carregado. Use o caminho completo e quaisquer opções necessárias.
 - **Comment** — Um comentário que descreve o aplicativo. Ele também aparece quando você mais tarde move o mouse sobre o carregador.
4. Clique na caixa Icon (que pode exibir No Icon). Selecione um dos ícones mostrados e clique em OK. Alternativamente, você pode navegar pelo sistema de arquivos para escolher um ícone.
5. Clique em OK.

O aplicativo agora deve aparecer no painel. Clique nele para iniciar o aplicativo.

ta

As imagens disponíveis para representar seu aplicativo estão contidas no diretório `/usr/share/pixmaps`. Esses arquivos estão nos formatos `.png` ou `.xpm`. Se não houver um ícone no diretório que você quer usar, crie seu próprio (em ambos os formatos) e o atribua ao aplicativo.

Adicionando uma gaveta

Uma gaveta é um ícone em que você pode clicar para exibir outros ícones representando menus, applets e carregadores; ele funciona como um painel. Essencialmente, qualquer item que você possa adicionar a um painel, também poderá adicionar a uma gaveta. Ao adicionar uma gaveta ao painel do GNOME, você pode incluir vários applets e carregadores que, juntos, ocupam o espaço de apenas um ícone. Clique na gaveta para mostrar os applets e carregadores, como se estivessem sendo puxadas para fora de um ícone de gaveta no painel.

Para adicionar uma gaveta ao seu painel, clique com o botão direito do mouse no painel e selecione Add

to Panel ⇒ Drawer. A gaveta é exibida no painel. Clique com o botão direito do mouse e adicione applets ou carregadores à gaveta como você faria com um painel. Clique no ícone novamente para fechar a gaveta.

A Figura 2.18 mostra uma parte do painel com uma gaveta aberta que inclui um ícone para o carregamento de uma previsão do tempo, notas e cotações da bolsa de valores.

one carregadores ou applets a uma gaveta no painel do GNOME 2.

Alterando as propriedades do painel

Você pode alterar a orientação, o tamanho, a política de ocultamento e as propriedades de fundo dos painéis do desktop. Para abrir a janela Panel Properties que se aplica a um painel específico, clique com o botão direito do mouse em um espaço livre no painel e escolha Properties. A janela Panel Properties que aparece inclui os seguintes valores:

- **Orientation** — Mova o painel para locais diferentes da tela clicando em uma nova posição.
- **Size** — Selecione o tamanho do painel, escolhendo a altura em pixels (48 pixels por padrão).
- **Expand** — Marque essa caixa de seleção para que o painel se expanda até preencher todo o lado, ou desmarque a caixa de seleção para deixar o painel com o tamanho dos applets que ele contém.
- **AutoHide** — Selecione se um painel é automaticamente oculto (aparecendo somente quando o ponteiro do mouse estiver na área).
- **Show Hide buttons** — Escolha se os botões Hide/Unhide (com setas pixmap sobre eles) aparecem nas bordas do painel.
- **Arrows on hide buttons** — Se você selecionar Show Hide Buttons, você pode optar por ter setas nesses botões.
- **Background** — A partir da guia de fundo, você pode atribuir uma cor ao fundo do painel, atribuir uma imagem de pixmap, ou apenas deixar o padrão (que é baseado no tema atual do sistema). Clique na caixa de seleção Background Image se você quiser selecionar uma imagem para o fundo e, então, selecione uma imagem, como um “ladrilho” de `/usr/share/backgrounds/tiles` ou outro diretório.

ca

ormalmente ligo o recurso de AutoHide e desligo os botões Hide. Usar AutoHide oferece mais espaço de desktop para trabalhar. Quando você move o mouse para a borda, onde o painel está, o painel aparece — assim você não precisa dos botões Hide.

Efeitos 3D com o AIGLX

Diversas iniciativas têm feito progressos nos últimos anos para trazer efeitos de desktop 3D para o Linux. Ubuntu, OpenSuse e Fedora utilizam o AIGLX (<http://fedoraproject.org/wiki/RenderingProject/aiglx>).

O objetivo do projeto Accelerated Indirect GLX (AIGLX) é adicionar efeitos 3D a sistemas desktop de

uso rotineiro. Ele faz isso por meio da implementação dos efeitos acelerados do OpenGL (<http://opengl.org>) utilizando a implementação do OpenGL em código-fonte aberto chamada Mesa (<http://www.mesa3d.org>).

Atualmente, AIGLX suporta um conjunto limitado de placas de vídeo e implementa apenas alguns efeitos 3D, mas dá uma boa ideia dos recursos sofisticados para “encher os olhos” que estão em desenvolvimento.

Se a sua placa de vídeo foi corretamente detectada e configurada, você pode ser capaz de simplesmente ligar o recurso de efeitos de desktop para ver os efeitos que foram implementados até o momento. Para ativar os Desktop Effects, selecione System ⇒ Preferences ⇒ Desktop Effects. Quando a janela Desktop Effects aparecer, selecione Compiz. (Se essa opção não estiver disponível, instale o pacote Compiz.)

A ativação do Compiz faz o seguinte:

- **Starts Compiz (Iniciar o Compiz)** — Interrompe o gerenciador de janelas atual e inicia o gerenciador de janelas Compiz.
- **Enables the Windows Wobble When Moved effect (Ativa o efeito Tremular Janela quando movido)** — Com esse efeito, quando você agarra a barra de título da janela para movê-la, ela tremula ao ser movida. Menus e outros itens que abrem no desktop também tremulam.
- **Enables the Workspaces on a Cube effect (Habilitar os espaços de trabalho em um efeito Cubo)** — Arraste uma janela no desktop para a direita ou para a esquerda e o desktop irá rodar como um cubo, com cada um dos espaços de trabalho do seu desktop aparecendo como um lado do cubo. Solte a janela no espaço de trabalho onde você quer que ele fique. Você também pode clicar no applet Switcher Workspace no painel inferior para girar o cubo a fim de exibir diferentes espaços de trabalho.

Outros bons efeitos de desktop resultam ao usar as teclas Alt+Tab para alternar entre diferentes janelas em execução. Conforme você pressiona Alt+Tab, uma miniatura de cada janela rola pela tela quando a janela que ela representa é realçada.

A Figura 2.19 mostra um exemplo de um desktop com o Compiz AIGLX ativado. A figura ilustra uma janela de navegador sendo movida de um espaço de trabalho para outro, e os espaços de trabalho girando sobre um cubo.

espaços de trabalho em um cubo com efeitos de desktop AIGLX ativado.

Eis alguns efeitos interessantes que você pode obter com seu desktop AIGLX 3D:

- **Girar cubo** — Segure Ctrl+Alt e pressione as teclas de seta para a direita e para a esquerda. O cubo do desktop gira para cada sucessivo espaço de trabalho (para frente ou para trás).
- **Girar cubo lentamente** — Segure as teclas Ctrl+Alt, pressione e segure o botão esquerdo do mouse e mova o mouse na tela. O cubo se move lentamente com o mouse entre os espaços de trabalho.
- **Redimensionar e separar as janelas** — Se seu desktop estiver bagunçado, segure Ctrl+Alt e pressione a seta para cima. As janelas irão encolher e se separar no desktop. Ainda segurando Ctrl+Alt, use as teclas de seta para destacar a janela que você deseja e solte as teclas para fazer essa janela vir para o primeiro plano.
- **Alternar entre as janelas** — Segure a tecla Alt e pressione a tecla Tab. Você vai ver versões reduzidas de todas as suas janelas em uma faixa no meio da tela, com a janela atual destacada no meio. Ainda segurando a tecla Alt, pressione Tab ou Shift+Tab para avançar ou retroceder pelas janelas. Solte as teclas quando a que você quer estiver realçada.
- **Redimensionar e separar os espaços de trabalho** — Segure Ctrl+Alt e pressione a tecla de seta para baixo a fim de ver imagens reduzidas dos espaços de trabalho mostradas em uma faixa. Ainda segurando Ctrl+Alt, use as teclas de seta para a direita e para a esquerda a fim de mover-se entre os diferentes espaços de trabalho. Solte as teclas quando o espaço de trabalho que você quer é realçado.
- **Enviar a janela atual para o espaço de trabalho seguinte** — Segure Ctrl+Alt+Shift juntas e pressione as teclas de seta para a esquerda e para a direita. O espaço de trabalho seguinte à esquerda ou à direita, respectivamente, aparece no desktop atual.
- **Deslizar janelas** — Pressione e segure o botão esquerdo do mouse sobre a barra de título da janela e pressione as teclas de seta para a esquerda, para a direita, para cima ou para baixo a fim de deslizar a janela atual pela tela.

Se ficar cansado de janelas tremulando e cubos girando, você pode facilmente desativar os efeitos 3D AIGLX e voltar a usar o Metacity como gerenciador de janelas. Selecione System ⇒ Preferences ⇒ Desktop novamente e desative o botão Enable Desktop Effects para desabilitar o recurso.

Se você tiver uma placa de vídeo suportada, mas achar que não é capaz de ativar os efeitos de desktop, verifique se o servidor X foi iniciado corretamente. Em particular, certifique-se de que o arquivo `/etc/X11/xorg.conf` está devidamente configurado. Certifique-se de que `dri` e `glx` estão carregados na seção Module. Além disso, adicione uma seção de extensões em qualquer lugar do arquivo (geralmente no final do arquivo) que aparece da seguinte maneira:

```
Section "extensions"  
Option "Composite"  
EndSection
```

Outra opção é adicionar a seguinte linha ao arquivo `/etc/X11/xorg.conf` na seção `Device`:

```
Option "XAANoOffscreenPixmaps"
```

A opção `XAANoOffscreenPixmaps` irá melhorar o desempenho. Verifique seu arquivo `/var/log/Xorg.log` para se certificar de que os recursos `DRI` e `AIGLX` foram iniciados corretamente. As mensagens nesse arquivo também podem ajudar a depurar outros problemas.

Resumo

O ambiente desktop GNOME tornou-se o padrão para muitos sistemas Linux, incluindo o Fedora e o RHEL. O GNOME 3 (usado agora no Fedora) é um desktop moderno e elegante, projetado para combinar com os tipos de interfaces disponíveis em muitos dos dispositivos móveis de hoje. O desktop GNOME 2 (utilizado no RHEL 6) proporciona uma experiência de desktop mais tradicional.

Além dos desktops GNOME, existem outros ambientes de desktop populares e úteis que você pode experimentar. O K Desktop Environment (KDE) oferece muito mais penduricalhos que o GNOME e é usado por padrão em diversas distribuições Linux. Distribuições em netbooks e Live CD às vezes usam os desktops LXDE ou Xfce.

Agora que você tem uma ideia de como obter e usar um desktop Linux, é hora de começar a se aprofundar nas interfaces administrativas mais profissionais. O Capítulo 3 apresenta a interface de shell de linha de comando do Linux.

Exercícios

Use esses exercícios para testar sua habilidade no uso de um desktop GNOME. Você pode usar um desktop GNOME 2.x (Red Hat Enterprise Linux até RHEL 6.x) ou GNOME 3.x (Fedora 16 ou versão posterior, ou Ubuntu 11.10 ou versão posterior). Se você empacar em algum exercício, as soluções para as tarefas (tanto no GNOME 2 como no GNOME 3) estão no Apêndice B.

1. Obtenha um sistema Linux com um desktop GNOME 2 ou GNOME 3. Inicie o sistema e faça login em um desktop GNOME.
2. Inicie o navegador Firefox e vá para a página inicial do GNOME (<http://gnome.org>).
3. Escolha um fundo de que você goste no site de arte do GNOME (<http://art.gnome.org/backgrounds>), baixe-o para a pasta Pictures e selecione-o como seu fundo atual.

4. Inicie uma janela Nautilus File Manager e a mova para o segundo espaço de trabalho em seu desktop.
5. Encontre a imagem que você baixou para usar como fundo do desktop e abra-a em qualquer visualizador de imagens.
6. Alterne entre os espaços de trabalho com o Firefox nele e aquele com o gerenciador de arquivos Nautilus.
7. Abra uma lista de aplicativos instalados em seu sistema e selecione um visualizador de imagens para abrir a partir dessa lista. Tente usar o mínimo possível de cliques ou teclas.
8. Altere a visualização das janelas no seu espaço de trabalho atual para visualizações menores de modo que possa ver todas ao mesmo tempo e alternar entre elas. Selecione qualquer janela que você gostaria de tornar sua janela atual.
9. No seu desktop, usando apenas o teclado, carregue um tocador de música.
10. Capture uma tela do seu desktop, usando apenas o teclado.

Parte II

Tornando-se um usuário avançado do Linux

NESTA PARTE

Capítulo 3

Utilizando o Shell

Capítulo 4

Movendo-se pelo Sistema de Arquivos

Capítulo 5

Trabalhando com Arquivos de Texto

Capítulo 6

Gerenciando Processos em Execução

Capítulo 7

Escrevendo Scripts de Shell Simples

Utilizando o Shell

NESTE CAPÍTULO

Entendendo o shell do Linux

Usando o shell a partir de consoles ou terminais

Usando comandos

Usando o histórico de comandos e o completamento de comando com `tab`

Conectando e expandindo comandos

Entendendo variáveis e aliases

Tomando as configurações de shell permanentes

Usando páginas `man` e outras documentações

Antes de ícones e janelas tomarem as telas de computador, você digitava comandos para interagir com a maioria dos computadores. Em sistemas UNIX, do qual o Linux se originou, o programa usado para interpretar e gerenciar comandos era referido como *shell*.

Independentemente da distribuição Linux que você esteja usando, você sempre pode contar com o fato de que o shell está disponível para você. Ele fornece uma maneira de criar arquivos de script executáveis, executar programas, trabalhar com sistemas de arquivos, compilar código de computador e gerenciar o computador. Embora o shell seja menos intuitivo do que interfaces gráficas (GUIs) comuns, a maioria dos especialistas considera o shell do Linux como sendo muito mais poderoso do que as GUIs. Os shells já existem há muito tempo e muitos recursos avançados foram construídos neles.

O shell do Linux ilustrado neste capítulo é chamado de *shell bash*, que é uma abreviação de Bourne Again Shell. O nome deriva do fato de que o `bash` é compatível com um dos primeiros shells UNIX: o shell Bourne (em homenagem a seu criador Stephen Bourne e representado pelo comando `sh`).

Embora o `bash` seja incluído na maioria das distribuições e seja considerado um padrão, outros shells estão disponíveis, incluindo o shell C (`csh`), que é popular entre os usuários de BSD UNIX, e o shell Korn (`ksh`), que é popular entre os usuários de Unix System V. O Ubuntu usa o shell `dash`, por padrão, que é projetado para executar mais rápido do que o shell `bash`. O Linux também tem um shell `tcsh` (um shell C aprimorado) e um shell `ash` (outro shell sócia do shell Bourne).

É muito provável que a distribuição Linux que você está usando tenha mais de um shell instalado por padrão e esteja disponível para seu uso. Este capítulo, porém, se concentra principalmente no shell bash. Isso porque as distribuições Linux apresentadas neste livro, Fedora e Red Hat Enterprise Linux, usam, ambas, o shell bash por padrão.

Eis algumas razões importantes para aprender a usar o shell:

- Você saberá se virar em qualquer Linux ou outro sistema tipo UNIX. Por exemplo, eu posso fazer login no meu servidor web do Red Hat Enterprise Linux, meu servidor multimídia de casa, ou no Mac da minha esposa e explorar e usar qualquer um desses sistemas de computador a partir de um shell. Posso até mesmo fazer login e executar comandos no meu celular Android. Todos eles executam sistemas Linux ou similares internamente.
- Recursos especiais de shell permitem coletar entrada de dados e direcionar saída de dados entre comandos e o sistema de arquivos do Linux. Para economizar digitação, você pode encontrar, editar e repetir comandos a partir do histórico de comandos do seu shell. Muitos usuários avançados raramente usam uma interface gráfica, fazendo a maior parte de seu trabalho a partir de um shell.
- Você pode reunir os comandos em um arquivo usando construções de programação, tais como testes condicionais, loops e instruções case para rapidamente fazer operações complexas que seriam trabalhosas para redigitar repetidamente. Programas compostos de comandos que são armazenados e executados a partir de um arquivo são referidos como *scripts de shell*. A maioria dos administradores de sistemas Linux utiliza scripts de shell para automatizar tarefas como fazer backup de dados, monitorar arquivos de log ou verificar a saúde do sistema.

O shell é um interpretador de linguagem de comandos. Se você já usou os sistemas operacionais da Microsoft, vai ver que o uso de um shell no Linux é semelhante — mas geralmente muito mais poderoso — ao interpretador usado para executar comandos no DOS ou na interface de comando CMD. Você pode usar o Linux alegremente a partir de uma interface de desktop gráfica, mas, a medida que for crescendo em Linux, vai precisar usar o shell em algum ponto para rastrear um problema ou administrar alguns recursos.

Como usar o shell não é óbvio no começo, mas com a ajuda certa, você pode aprender rapidamente muitos dos recursos mais importantes dele. Este capítulo é seu guia para trabalhar com comandos de sistema, processos e sistema de arquivos do Linux a partir do shell. Ele descreve o ambiente de shell e ajuda você a adaptá-lo às suas necessidades.

Sobre Shells e Janelas de Terminal

Há várias maneiras de se chegar a uma interface de shell no Linux. Três das mais comuns são o prompt de shell, a janela de terminal e o console virtual, sobre os quais você aprenderá mais nas próximas seções.

Para começar a utilizar esta seção, inicie seu sistema Linux. Na sua tela, você deverá ver um prompt de login em texto simples semelhante ao seguinte:

joe login:

Ou você verá uma tela de login gráfica.

Em qualquer um dos casos, você deve fazer login com uma conta de usuário regular. Se você tiver um prompt de login de texto simples, continue na seção “Usando o prompt de shell”. Se você entrar por meio de uma tela gráfica, vá para “Usando uma janela de terminal” para ver como acessar um shell a partir do desktop. Em ambos os casos, você pode acessar mais shells, como descrito na seção “Usando consoles virtuais”.

Usando o prompt de shell

Se seu sistema Linux não tem interface gráfica do usuário (ou uma que não está funcionando no momento), você provavelmente verá um prompt de shell depois de fazer login. Digitar comandos do shell poderá ser seu principal meio de usar o sistema Linux.

O prompt padrão para um usuário comum é simplesmente um sinal de cifrão:

```
$
```

O padrão de solicitação para o usuário root é um sinal de jogo da velha (também chamado de *cerquilha*):

```
#
```

Na maioria dos sistemas Linux, os prompts \$ e # são precedidos por seu nome de usuário, o nome do sistema e o nome do diretório atual. Por exemplo, uma tela de login para o usuário chamado jake em um computador chamado pine com `/usr/share/` como o diretório de trabalho atual apareceria como:

```
[jake@pine share]$
```

Você pode mudar o prompt para exibir os caracteres que lhe agradam e até ler trechos de informações sobre o sistema — por exemplo, você pode usar o diretório de trabalho atual, a data, o nome do computador local ou qualquer sequência de caracteres como seu prompt. Para configurar seu prompt, consulte a seção “Configurando o Prompt” mais adiante neste capítulo.

Apesar de um enorme número de recursos estar disponível com o shell, é fácil começar apenas digitando alguns comandos. Tente alguns comandos mostrados no restante desta seção para familiarizar-se com seu ambiente atual de shell.

Nos exemplos que se seguem, os símbolos de cifrão (\$) e cerquilha (#) indicam um prompt. Enquanto um \$ indica que o comando pode ser executado por qualquer usuário, um # normalmente significa que você deve executar o comando como root — muitas ferramentas administrativas exigem permissão de root para ser capaz de executá-las. O prompt é seguido pelo comando que você digita (e então você pressiona Enter). As linhas que se seguem exibem a saída resultante do comando.

Usando uma janela terminal

Com a interface gráfica do desktop rodando, você pode abrir um programa emulador de terminal (às vezes referido como janela de terminal) para iniciar um shell. A maioria das distribuições Linux facilita o acesso a um shell a partir da interface gráfica. Eis duas maneiras comuns para carregar uma janela de terminal a partir de um desktop Linux:

- **Clique com o botão direito do mouse na área de trabalho.** No menu de contexto que aparece, se você vê `Open in Terminal`, `Shells`, `New Terminal`, `Terminal Window`, `Xterm` ou algum item similar, selecione-o para iniciar uma janela de terminal. (Algumas distribuições desabilitam esse recurso.)
- **Clique no menu do painel.** Muitos desktops Linux incluem um painel na parte superior ou inferior da tela a partir do qual você pode carregar aplicativos. Por exemplo, em alguns sistemas que usam o desktop GNOME, você pode selecionar `Applications` ⇒ `System Tools` ⇒ `Terminal` para abrir uma janela de terminal.

Em todos os casos, você deve ser capaz de digitar um comando como se fosse a partir de um shell sem GUI. Há vários emuladores de terminal disponíveis no Linux. No Fedora, Red Hat Enterprise Linux (RHEL) e outras distribuições Linux que usam o desktop GNOME, a janela padrão do emulador de terminal é um terminal GNOME (representado pelo comando `gnome-terminal`).

O Terminal GNOME suporta muitas funções além do shell básico. Por exemplo, você pode cortar e colar texto para ou de uma janela do Terminal GNOME, alterar fontes, definir um título, escolher cores ou imagens para usar como fundo e definir quanto texto salvar quando o ele rola para fora da tela.

Para experimentar alguns recursos do Terminal do GNOME, inicie um sistema Fedora ou RHEL e faça login no desktop. Então, siga este procedimento:

1. **Select Applications** ⇒ **System Tools** ⇒ **Terminal**. A janela de terminal deve abrir no seu desktop.
2. **Selecione Edit** ⇒ **Profiles**, então, com **Default** destacado, **selecione Edit**.
3. **Na guia General**, **desmarque a caixa “Use the system fixed width font”** (“Usar a fonte de largura fixa do sistema”).
4. **A partir da janela Choose A Terminal Font**, **experimente uma fonte diferente e selecione OK**. A nova fonte aparece na janela de terminal.
5. **Selecione de novo a caixa “Use system fixed width font”**. Isso vai levar você de volta à fonte original.
6. **Na guia Colors**, **limpe a caixa “Use colors from system theme”** (“Usar as cores do tema do sistema”). A partir daí você pode experimentar diferente fontes e cores de fundo.
7. **Selecione de novo a caixa “Use colors from system theme” para restaurar as cores padrão**.
8. **Vá para a janela Profile**. Há outros recursos que você pode querer experimentar, como definir uma imagem de fundo, tornar o fundo transparente ou definir quantos dados rolados

são mantidos.

9. **Feche a janela Profile quando terminar.** Agora você está pronto para usar sua janela de terminal.

Se estiver usando o Linux a partir de um desktop gráfico, provavelmente você acessará o shell a partir de uma janela de terminal na maioria das vezes.

Usando consoles virtuais

A maioria dos sistemas Linux que incluem uma interface de desktop inicia vários consoles virtuais para rodar no computador. Consoles virtuais são uma maneira de ter várias sessões de shell abertas ao mesmo tempo, além da interface gráfica que você está usando.

Você pode alternar entre os consoles virtuais, segurando as teclas Ctrl e Alt e pressionando uma tecla de função entre F1 e F7. Por exemplo, no Fedora, pressione Ctrl+Alt+F1 (ou F2, F3, F4 e assim por diante até F7 na maioria dos sistemas Linux) para exibir um dos sete consoles virtuais. O primeiro espaço de trabalho virtual no Fedora é onde a interface gráfica está e os seis consoles virtuais seguintes são consoles virtuais baseados em texto. Você pode voltar para a interface gráfica (se estiver em execução), pressionando Ctrl+Alt+F1. (Em alguns sistemas, a interface gráfica roda no console virtual 7 ou 5. Portanto, você deve voltar à interface gráfica, pressionando Ctrl+Alt+F5 ou Ctrl+Alt+F7).

Experimente isso agora. Segure as teclas Ctrl+Alt e pressione F3. Você deverá ver um prompt de login em texto simples. Faça login usando seu nome de usuário e senha. Experimente alguns comandos. Quando terminar, digite exit para sair do shell. Então, pressione Ctrl+Alt+F1 para voltar para sua interface de desktop gráfica. Você pode ir e voltar livremente entre esses consoles gráficos.

Escolhendo Seu Shell

Na maioria dos sistemas Linux, o shell padrão é o shell bash. Para descobrir qual é seu shell de login padrão, digite os seguintes comandos:

```
$ who am i
chris pts/0          2011-11-26 07:19 (:0.0)
$ grep chris /etc/passwd
cnegus:x:13597:13597:Chris Negus:/home/cnegus:/bin/bash
```

O comando `who am i` exibe seu nome de usuário e o comando `grep` (substituindo `chris` pelo seu nome) exibe a definição de sua conta de usuário no arquivo `/etc/passwd`. O último campo nessa entrada exibe que o shell bash (`/bin/bash`) é o shell padrão (aquele que inicia quando você faz login ou abre uma janela de terminal).

É possível, embora não provável, que você tenha um conjunto padrão de shells diferente. Para tentar um shell diferente, basta digitar o nome do shell (exemplos incluem `ksh`, `tcsh`, `csch`, `sh`, `dash` e outros, assumindo que eles estão instalados). Você pode experimentar alguns comandos nesse shell e digitar `exit` quando terminar para voltar ao shell bash.

Você pode optar por usar shells diferentes, pelas seguintes razões:

- Você está acostumado a usar sistemas UNIX System V (muitas vezes `ksh` por padrão) ou Sun Microsystems e outras distribuições baseadas no UNIX Berkeley (frequentemente `csh` por padrão) e você se sente mais à vontade usando shells padrão a partir desses ambientes.
- Você deseja executar scripts de shell que foram criados para um ambiente de shell específico e precisa executar o shell para o qual eles foram feitos a fim de poder testar ou usar esses scripts em seu shell atual.
- Você simplesmente prefere os recursos de um shell aos dos outros. Por exemplo, um membro do meu grupo de usuários do Linux prefere `ksh` ao `bash`, porque não gosta do modo como os aliases são usados com o `bash`.

Embora a maioria dos usuários do Linux prefira um shell ou outro, quando você sabe como usar um shell, você pode aprender rapidamente qualquer um dos outros consultando ocasionalmente a página `man` do shell (por exemplo, digite `man bash`). As páginas `man` (descritas mais adiante na seção “Obtendo Informações sobre os Comandos”) fornecem a documentação para os comandos, formatos de arquivos e outros componentes no Linux. A maioria das pessoas usa o `bash` só porque não têm um motivo especial para usar um shell diferente. O resto desta seção descreve o shell `bash`.

O `Bash` inclui recursos originalmente desenvolvidos para os shells `sh` e `ksh` nos primeiros sistemas UNIX, bem como alguns recursos do `csh`. Espere ver o `bash` como o shell padrão na maioria dos sistemas Linux que você usar, com a exceção de alguns sistemas Linux especializados (como alguns que são executados em dispositivos embarcados) que podem exigir um shell menor que precisa de menos memória e requer menos recursos. A maioria dos exemplos neste capítulo se baseia no shell `bash`.

ca

a pena conhecer o shell `bash`, não só porque ele é o padrão na maioria das instalações, mas também porque é o que você vai usar na maioria dos exames de certificação do Linux.

Executando comandos

A maneira mais simples de executar um comando é simplesmente digitar o nome do comando a partir de um shell. A partir da área de trabalho, abra uma janela de terminal. Então, digite o seguinte comando:

```
$ date
Sat Nov 26 08:04:00 EST 2011
```

O comando `date`, sem opções ou argumentos, exibe dia, mês, data, hora, fuso horário e ano como no exemplo acima. Eis alguns comandos que você pode experimentar:

```
$ pwd
/home/chris
$ hostname
mydesktop
$ ls
Desktop    Downloads  Pictures    Templates
Documents  Music      Public      Videos
```

O comando `pwd` exibe o diretório de trabalho atual. O comando `hostname` exibe o nome de host do seu computador. O comando `ls` lista os arquivos e diretórios no diretório atual. Embora muitos comandos possam ser executados simplesmente digitando seus nomes, é mais comum digitar **mais** após o comando para modificar seu comportamento. Os caracteres e palavras que você pode digitar depois de um comando são chamados de opções e argumentos.

Entendendo a sintaxe de comando

A maioria dos comandos tem uma ou mais *opções* que você pode adicionar para mudar o comportamento deles. Em geral, as opções consistem em uma única letra, precedida por um hífen. Mas você pode agrupar opções de uma letra ou preceder cada uma com um hífen, para usar mais de uma opção de cada vez. Por exemplo, os dois seguintes usos de opções para o comando `ls` são os mesmos:

```
$ ls -l -a -t
$ ls -lat/
```

Em ambos os casos, o comando `ls` é executado com as opções `-l` (listagem longa) `-a` (exibe arquivos de ponto ocultos) e `-t` (lista por tempo).

Alguns comandos incluem opções que são representados por uma palavra inteira. Para instruir um comando a usar uma palavra inteira como uma opção, você geralmente a precede com um hífen duplo (`--`). Por exemplo, para usar a opção de ajuda em muitos comandos, você digita `--help` na linha de comando. Sem o hífen duplo, as letras `h`, `e`, `l` e `p` devem ser interpretadas como opções separadas. (Há alguns comandos que não seguem a convenção de hífen duplo, usando um único hífen antes de uma palavra, mas a maioria dos comandos usará hífen duplos para as opções de palavras.)

ta

pode usar a opção `--help` com a maioria dos comandos para ver as opções e argumentos que eles suportam: exemplo, experimente digitar `hostname --help`.

Muitos comandos também aceitam argumentos depois que certas opções são inseridas ou no final da linha de comando inteira. Um *argumento* é um fragmento extra de informações, como um nome de arquivo, diretório, nome de usuário, dispositivo ou outro item que informa ao comando o objeto sobre o qual ele deve atuar. Por exemplo, `cat /etc/passwd` exibe o conteúdo do arquivo `/etc/passwd` em sua tela. Nesse caso, `/etc/passwd` é o argumento. Em geral, você pode ter quantos argumentos quiser, limitado apenas pelo número total de caracteres permitidos em uma linha de comando.

Há casos em que um argumento está associado a uma opção. Nesses casos, o argumento deve ser imediatamente seguido da opção. Com opções de uma única letra, o argumento geralmente vem depois de um espaço. Para opções de uma palavra, o argumento muitas vezes vem depois de um sinal de igual (=). Eis alguns exemplos:


```
$ ls --hide=Desktop
Documents Music Public Videos
Downloads Pictures Templates
```

No exemplo anterior, a opção `--hide` instrui o comando `ls` a não exibir o arquivo ou diretório chamado `Desktop` ao listar o conteúdo do diretório. Note que o sinal de igual vem imediatamente depois da opção (sem espaço) e, então, o argumento (novamente, sem espaço).

Eis um exemplo de uma opção de uma única letra que é seguida por um argumento:

```
$ tar -cvf backup.tar /home/chris
```

No exemplo do comando `tar` anterior, as opções instruem o comando a criar (c) um arquivo (f) chamado `backup.tar` que inclui todo o conteúdo do diretório `/home/chris` e seus subdiretórios e exibe mensagens verbosas (v) conforme o backup é feito. Como `backup.tar` é um argumento para a opção `f`, `backup.tar` deve vir imediatamente após a opção.

Eis alguns comandos que você pode experimentar. Veja como eles se comportam de maneira diferente com diferentes opções:

```
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos
$ ls -a
. Desktop .gnome2_private .lesshst Public
.. Documents .gnote .local Templates
.bash_history Downloads .gnupg .mozilla Videos
.bash_logout .emacs .gstreameer-0.10 Music .xsession-errors
.bash_profile .esd_auth .gtk-bookmarks Pictures .zshrc
.bashrc .fsync.log .gvfs Pictures
$ uname
Linux
$ uname -a
Linux myhost.example.com 2.6.32-131.17.1.el6.x86_64 #1 SMP Thu Sep 29
10:24:25 EDT 2011 x86_64 x86_64 x86_64 GNU/LinuxLinux
$ date
Sun Dec 4 09:08:38 EST 2011
$ date +%d/%m/%y
04/12/11
$ date +%A, %B %d, %Y
Sunday, December 04, 2011
```

O comando `ls`, por si só, exibe todos os arquivos e diretórios regulares no diretório atual. Ao adicionar `-a`, você também pode ver os arquivos ocultos no diretório (aqueles começando com um ponto). O comando `uname` exibe o tipo de sistema que está rodando (Linux). Quando você adiciona `-a`, também pode ver o `hostname`, a compilação e a versão do kernel.

O comando `date` tem alguns tipos especiais de opções. Por si só, `date` simplesmente imprime a data e a hora atuais, como mostrado acima. Mas o comando `date` suporta uma opção `+` de formato especial, que permite exibir a data em diferentes formatos. Digite `date --help` para ver os diferentes indicadores de formato que você pode usar.

Experimente os comandos `id` e `who` para ter uma ideia do seu ambiente atual do Linux, como descrito nos parágrafos a seguir.

Quando você efetua login em um sistema Linux, ele vê você como tendo uma identidade particular, o que inclui seu nome de usuário, o nome do seu grupo, seu ID de usuário e seu ID do grupo. O Linux também monitora sua sessão de login: sabe quando você fez login, quanto tempo o computador está inativo e onde

you se conectou.

Para saber informações sobre sua identidade, use o comando `id` da seguinte maneira:

```
$ id
uid=501(chris) gid=501(chris) groups=105(sales), 7(lp)
```

Nesse exemplo, o nome de usuário é `chris`, o qual é representado pela identificação numérica de usuário (`uid`) 501. O grupo primário de `chris` também se chama `chris`, seu ID de grupo (`gid`) é 501. É normal para usuários do Fedora e do Red Hat Enterprise Linux ter o mesmo nome de grupo primário que seu nome de usuário. O usuário `chris` também pertence a outros grupos chamados `sales` (`gid` 105) e `lp` (`gid` 7). Esses nomes e números representam as permissões que `chris` tem para acessar os recursos do computador.

ta

istribuições Linux que tenham Security Enhanced Linux (SELinux) ativado, como o Fedora e RHEL, mostram informações adicionais no final da saída `id`. Essa saída deve ser algo como o seguinte:

```
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

maioria dos usuários não precisa se preocupar com o SELinux, a menos que eles estejam configurando servidores. Consulte o Capítulo 24, “Aumentando a segurança do Linux com SELinux”, se quiser saber mais sobre o SELinux.

Você pode ver informações sobre sua sessão atual usando o comando `who`. No exemplo a seguir, a opção `-u` diz para adicionar informações sobre o tempo ocioso e o ID do processo `-H` pede que um cabeçalho seja impresso:

```
$ who -uH
NAME     LINE    TIME          IDLE          PID          COMMENT
chris    tty1    Jan 13 20:57  .            2013
```

A saída desse `who` mostra que o usuário `chris` está logado no `tty1` (que é o primeiro console virtual no monitor conectado ao computador) e sua sessão de login começou às 20:57 em 13 de janeiro. O tempo `IDLE` mostra quanto tempo o shell ficou aberto sem qualquer comando ser digitado (o ponto indica que ele está ativo no momento). `PID` mostra o ID do processo do shell de login do usuário. `COMMENT` iria mostrar o nome do computador remoto de onde o usuário teria se logado, caso isso acontecesse a partir de outro computador da rede, ou o nome do display X local se o usuário estivesse usando uma janela de terminal (por exemplo: 0,0).

Localizando comandos

Agora que já digitou alguns comandos, talvez você esteja querendo saber onde eles estão localizados e como o shell encontra esses comandos digitados. Para localizar comandos que você digita, o shell procura no que é chamado de “caminho” (*path*). Para comandos que não estão em seu caminho, você

pode digitar a identidade completa da localização deles.

Se você sabe o diretório que contém o comando que deseja executar, uma maneira de executá-lo é digitar o caminho completo, ou absoluto, para o comando. Por exemplo, você executa o comando `date` do diretório `/bin`, digitando

```
$ /bin/date
```

Obviamente isso pode ser inconveniente, especialmente se o comando reside em um diretório com um caminho longo. A melhor maneira é ter comandos armazenados em diretórios bem conhecidos e depois adicioná-los para a variável de ambiente `PATH` do seu shell. O caminho consiste em uma lista de diretórios em que os comandos que você insere são verificados sequencialmente. Para ver seu caminho atual, digite o seguinte:

```
$ echo $PATH
```

```
/usr/local/bin:/usr/bin:/bin:/usr/local/sbin:/usr/sbin:/sbin:/sbin:  
/home/cn/bin:/sbin
```

Os resultados exibem um caminho padrão comum para um usuário normal do Linux. Os diretórios na lista de caminho são separados por dois-pontos. A maioria dos comandos do usuário que vêm com Linux são armazenadas nos diretórios `/bin`, `/usr/bin` ou `/usr/local/bin`. Os diretórios `/sbin` e `/usr/sbin` contêm comandos administrativos (alguns sistemas Linux não colocam esses diretórios em caminhos de usuários regulares). O último diretório mostrado é o `bin` no diretório `home` do usuário.

ca

Se você quiser adicionar seus próprios comandos ou scripts shell, coloque-os no diretório `bin` no seu diretório inicial (por exemplo, `/home/cn/bin` para o usuário chamado `cn`). Esse diretório é automaticamente adicionado ao seu caminho em alguns sistemas Linux, embora possa ser necessário que você crie esse diretório ou adicione-o ao seu `PATH` em outros sistemas Linux. Então, se você adicionar o comando para o diretório `bin` com permissão de execução, pode começar a usá-lo simplesmente digitando o nome do comando no prompt do seu shell. Para tornar os comandos disponíveis para todos os usuários, adicione-os em `/usr/local/bin`.

Ao contrário de alguns outros sistemas operacionais, por padrão, o Linux não verifica o diretório atual para um arquivo executável antes de procurar o caminho. Ele imediatamente começa a procurar o caminho, e executáveis no diretório atual são executados somente se eles estiverem na variável `PATH` ou se você der o endereço absoluto (como `/home/chris/scriptx.sh`) ou relativo (por exemplo, `./scriptx.sh`) deles.

A ordem do diretório de caminho é importante. Os diretórios são verificados da esquerda para a direita. Assim, nesse exemplo, se houver um comando chamado `foo` localizado em ambos os `/bin` e `/usr/bin` diretórios, o que está em `/bin` é executado. Para executar outro comando `foo`, digite o caminho completo para o comando ou altere sua variável `PATH`. (Alterar seu `PATH` e adicionar diretórios para ele são descritos mais adiante neste capítulo.)

Nem todos os comandos que você executa estão localizados em diretórios em sua variável `PATH`. Alguns comandos estão predefinidos no shell. Outros comandos podem ser anulados criando aliases que definem quaisquer comandos e opções que você quer que o comando execute. Há também maneiras de definir uma função que consiste em uma série armazenada de comandos. Eis a ordem em que o shell verifica os comandos digitados:

1. **Aliases.** Nomes definidos pelo comando `alias` que representam um determinado comando e um conjunto de opções. Digite **alias** para ver quais aliases estão definidos. Muitas vezes, os aliases permitem que você defina um nome curto para um comando longo e complicado. (Descrevo como criar seus próprios aliases mais adiante neste capítulo.)
2. **Palavra reservada do shell.** Palavras reservadas pelo shell para uso especial. Muitas dessas são palavras que você usaria na função de digitação de programação, como `do`, `while`, `case` e `else`. (Abordaremos algumas dessas palavras reservadas no Capítulo 7, “Escrevendo scripts de shell simples.”)
3. **Função.** Um conjunto de comandos que são executados juntos no shell atual.
4. **Comando predefinido.** Um comando predefinido no shell. Como resultado, não há nenhuma representação do comando no sistema de arquivos. Alguns dos comandos mais comuns que você vai usar são os de shell predefinidos, tais como `cd` (para mudar de diretórios), `echo` (para a saída de texto para a tela), `exit` (para sair de um shell), `fg` (para trazer um comando executado em segundo plano para o primeiro plano), `history` (para ver uma lista de comandos que foram executados anteriormente), `pwd` (para listar o diretório de trabalho atual), `set` (para definir as opções de shell) e `type` (para exibir a localização de um comando).
5. **Comando do sistema de arquivos.** Esse comando é armazenado e executado dentro do sistema de arquivos do computador. (Esses são os comandos que são indicados pelo valor da variável `PATH`.)

Para saber de onde um determinado comando é retirado, você pode usar o comando `type`. (Se você estiver usando um shell que não seja `bash`, use o comando `which`.) Por exemplo, para descobrir onde o comando shell `bash` está localizado, digite o seguinte:

```
$ type bash
bash is /bin/bash
```

Experimente essas poucas palavras com o comando `type` para ver outros locais dos comandos: `que`, `which`, `case` e `return`. Se um comando reside em vários locais, você pode adicionar a opção `-a` para obter todos os locais conhecidos do comando. Por exemplo, o comando `type -a ls` deve mostrar um alias e localização do sistema de arquivos para o comando `ls`.

ca

vezes você executa um comando e recebe uma mensagem de erro informando que o comando não foi localizado ou que a permissão para executá-lo foi negada. Se o comando não foi encontrado, verifique se você digitou-o

atamente e se está localizado na sua variável `PATH`. Se a permissão para executar o comando foi negada, ele não está na variável `PATH`, mas pode não ser executável. Lembre-se também que o uso de maiúscula ou minúscula é importante, por isso, se digitar `CAT` ou `Cat` não vai encontrar o comando `cat`.

Se um comando não está na sua variável `PATH`, você pode usar o comando `locate` para tentar encontrá-lo. Usando `locate`, é possível pesquisar qualquer parte do sistema que seja acessível para você (alguns arquivos são acessíveis apenas para o usuário `root`). Por exemplo, se quisesse encontrar a localização do comando `chage`, você pode digitar o seguinte:

```
$ locate chage
/usr/bin/chage
/usr/sbin/lchage
/usr/share/man/fr/man1/chage.1.gz
/usr/share/man/it/man1/chage.1.gz
/usr/share/man/ja/man1/chage.1.gz
/usr/share/man/man1/chage.1.gz
/usr/share/man/man1/lchage.1.gz
/usr/share/man/pl/man1/chage.1.gz
/usr/share/man/ru/man1/chage.1.gz
/usr/share/man/sv/man1/chage.1.gz
/usr/share/man/tr/man1/chage.1.gz
```

Observe que `locate` não só encontrou o comando `chage`, mas também encontrou o comando `lchage` e uma variedade de páginas `man` associadas com `chage` para diferentes idiomas. O comando `locate` procura em todo seu sistema de arquivos e não apenas em diretórios que contêm comandos.

Nos próximos capítulos, você aprenderá a usar comandos adicionais. Por enquanto, quero que você se familiarize com a maneira como o shell em si funciona. Então, discuto em seguida os recursos para lembrar e completar comandos, utilizar variáveis e criar aliases.

Lembrando comandos com o histórico de comandos

Ser capaz de repetir um comando que você executou anteriormente em uma sessão de shell pode ser conveniente. Às vezes, uma linha de comando é muito longa e complexa e poder recuperá-la a partir de uma sessão anterior pode poupar alguns problemas. Felizmente, alguns recursos do shell permitem recuperar e editar linhas de comandos anteriores, ou completar uma linha de comando parcialmente digitada.

O *histórico do shell* é uma lista dos comandos que você inseriu anteriormente. Usando o comando `history` em um shell `bash`, é possível ver seus comandos anteriores. Assim, utilizando vários recursos de shell, você pode chamar novamente linhas de comando individuais a partir dessa lista e alterá-las como preferir.

O restante desta seção descreve como editar a linha de comando, como completar partes das linhas de

comando e como chamar novamente e trabalhar com a lista de histórico.

Edição da linha de comando

Se você digitar algo errado em uma linha de comando, o shell `bash` garante que você não tem que apagar toda a linha e começar de novo. Da mesma forma, você pode chamar uma linha de comando prévia novamente e alterar os elementos para criar um novo comando.

Por padrão, o shell `bash` usa edição de linha de comando que é baseada no editor de texto `emacs`. (Digite **man emacs** para ler sobre isso, se isso lhe interessar.) Se estiver familiarizado com o `emacs`, você provavelmente já conhece a maioria das teclas de atalho descritas aqui.

ca

Preferir o comando `vi` para editar linhas de comando de shell, você pode fazer isso facilmente. Adicione a seguinte linha ao arquivo `.bashrc` no seu diretório inicial:

```
set -o vi
```

Próxima vez que abrir um shell, você pode usar os comandos `vi` para editar as suas linhas de comando.

Para fazer a edição, você pode utilizar uma combinação de teclas control, teclas meta e teclas de seta. Por exemplo, `Ctrl+F` significa segurar a tecla `Ctrl` e pressionar **f**. `Alt+F` significa segurar a tecla `Alt` e pressionar **f**. (Em vez da tecla `Alt`, o teclado pode usar uma tecla `Meta` ou a tecla `Esc`. Em um teclado `Windows`, você pode usar a tecla `Windows`.)

Para experimentar um pouco de edição de linha de comando, digite o seguinte:

```
$ ls /usr/bin | sort -f | less
```

Esse comando lista o conteúdo do diretório `/usr/bin`, classifica-o em ordem alfabética (independentemente do caso) e redireciona a saída para `less`. O comando `less` exibe a primeira página de saída, após a qual você pode ir através do resto da saída uma linha (pressione `Enter`) ou uma página (pressione a barra de espaço) de cada vez. Basta pressionar **q** quando terminar. Agora, suponha que você queira mudar `/usr/bin` para `/bin`. Você pode usar as etapas a seguir para alterar o comando:

1. **Pressione a tecla da seta para cima (↑)**. Isso exibe o comando mais recente de seu histórico shell.
2. **Pressione `Ctrl+A`**. Isso move o cursor para o início da linha de comando.
3. **Pressione `Ctrl+F` ou a tecla de seta para a direita (→)**. Repita esse comando algumas vezes para posicionar o cursor sob a primeira barra (/).
4. **Pressione `Ctrl+D`**. Repita esse comando quantas vezes for necessário para eliminar `/usr` da linha.

5. **Pressione Enter.** Isso executa a linha de comando.

À medida que edita uma linha de comando, você pode digitar caracteres regulares para adicioná-los a ela a qualquer momento. Os caracteres aparecem no local do cursor de texto. Você pode usar as teclas de seta para a direita e para a esquerda a fim de mover o cursor ao longo da linha de comando. Você também pode pressionar as teclas de seta para cima e para baixo a fim de percorrer os comandos anteriores na lista do histórico e assim selecionar uma linha de comando para edição. (Veja a “Recuperação de linhas de comando” para obter detalhes sobre como recuperar comandos da lista de histórico.)

Há muitas combinações de teclas que você pode usar para editar suas linhas de comando. A Tabela 3.1 lista as teclas que você pode usar para se mover ao longo da linha de comando.

TABELA 3.1 Teclas de atalho para navegar pelas linhas de comando

Atalho	Nome completo	Significado
l+F	Caractere para frente	Move o cursor um caractere para frente.
l+B	Caractere para trás	Move o cursor um caractere para trás.
+F	Palavra para frente	Move o cursor uma palavra para frente.
+B	Palavra para trás	Move o cursor uma palavra para trás.
l+A	Início da linha	Move o cursor para o início da linha atual.
l+E	Fim da linha	Move o cursor para o final da linha atual.
l+L	Limpar tela	Limpa a tela e move o cursor para a parte superior esquerda da tela.

A combinação de teclas na Tabela 3.2 pode ser usada para editar linhas de comando.

TABELA 3.2 Tecla de atalho para editar linhas de comando

Atalho	Nome Completo	Significado
l+D	Excluir atual	Exclui o caractere atual.
l+Backspace	Excluir anterior	Exclui o caractere anterior.
l+T	Transpor caractere	Troca de posição entre o caractere anterior e o atual.
+T	Transpor palavras	Troca de posição entre as palavras atuais e as anteriores.
+U	Palavra em maiúscula	Converte os caracteres da palavra atual em letras maiúsculas.
+L	Palavra em minúscula	Converte os caracteres da palavra atual em letras minúsculas.
+C	Palavra com inicial maiúscula	Converte o caractere inicial da palavra atual em letra maiúscula.
l+V	Inserir caractere especial	Adiciona um caractere especial. Por exemplo, para

Use as teclas na Tabela 3.3 para cortar e colar texto em uma linha de comando.

TABELA 3.3 Teclas para cortar e colar texto em linhas de comando

Atalho	Nome Completo	Significado
I+K	Corta fim da linha	Recorta o texto no final da linha.
I+U	Corta o início da linha	Recorta o texto no início da linha.
I+W	Corta a palavra anterior	Recorta a palavra localizada antes do cursor.
+D	Corta a próxima palavra	Recorta a palavra após o cursor.
I+Y	Cola texto recente	Cola o texto mais recentemente recortado.
+Y	Cola texto anterior	Volta ao texto previamente recortado e cola-o.
I+C	Exclui a linha inteira	Exclui a linha inteira.

Completamento de linha de comando

Para poupar algumas teclas, o shell bash oferece várias maneiras de completar os valores parcialmente digitados. Para tentar completar um valor, digite os primeiros caracteres e então pressione Tab. Eis alguns dos valores que você pode digitar parcialmente a partir de um shell bash:

- **Comando, alias, ou função** — Se o texto digitado começa com caracteres regulares, o shell tenta completar o texto com um comando, um alias ou um nome de função.
- **Variável** — Se o texto digitado começa com um cifrão (\$), o shell completa-o com uma variável do shell atual.
- **Nome de usuário** — Se o texto digitado começa com um til ~, o shell completa-o com um nome de usuário. Como resultado, ~username indica o diretório home do usuário chamado.
- **Hostname** — Se o texto digitado começa com um sinal de arroba (@), o shell completa-o com um hostname extraído do arquivo /etc/hosts.

ca

adicionar nomes de máquinas a partir de um arquivo adicional, você pode definir a variável `HOSTFILE` ao longo do arquivo. O arquivo deve estar no mesmo formato do `/etc/hosts`.

Eis alguns exemplos da conclusão de comando. (Quando você vir <Tab>, isso significa pressionar a tecla Tab no teclado.) Digite o seguinte:

```
$ echo $OS<Tab>
```



```
$ cd ~ro<Tab>
```

```
$ fing <Tab>
```

O primeiro exemplo faz com que `$OS` expanda-se para a variável `$OSTYPE`. No próximo exemplo, `~ro` expande-se para o diretório inicial do usuário `root` (`~root/`). Então, `fing` expande-se para o comando `finger`.

Pressionar `Tab` duas vezes oferece algumas possibilidades maravilhosas. Há momentos em que estão disponíveis várias conclusões possíveis para a sequência de caracteres digitada. Nesses casos, você pode verificar as maneiras como o texto pode ser expandido pressionando `Tab` duas vezes no ponto em que você quer completar o código.

O que vem a seguir mostra o resultado que você obterá se verificasse as possíveis conclusões sobre `$P`:

```
$ echo $P<Tab><Tab>
```

```
$PATH $PPID $PS1 $PS2 $PS4 $PWD
```

```
$ echo $P
```

Nesse caso, existem seis possíveis variáveis que começam com `$P`. Depois que as possibilidades são exibidas, a linha de comando original retorna, pronta para ser completada como você escolher. Por exemplo, se você digitou um outro `P` e depois bateu `Tab` novamente, a linha de comando seria completada com `$PPID` (a única possibilidade exclusiva).

Recuperação de linhas de comando

Depois de digitar uma linha de comando, ela inteira é salva na lista de histórico do seu shell. A lista é armazenada no shell atual até você sair dele. Depois disso, ela é gravada em um arquivo de histórico, a partir do qual qualquer comando pode ser recuperado para ser executado novamente na sua próxima sessão. Depois que um comando é recuperado, você pode modificar a linha de comando, como descrito anteriormente.

Para ver seu histórico, use o comando `history`. Digite o comando sem opções ou seguido por um número para os comandos mais recentes. Por exemplo:

```
$ history 8
```

```
382 date
```

```
383 ls /usr/bin | sort -a | more
```

```
384 man sort
```

```
385 cd /usr/local/bin
```

```
386 man more
```

```
387 useradd -m /home/chris -u 101 chris
```

```
388 passwd chris
```

```
389 history 8
```

Um número precede cada linha de comando na lista. Você pode recuperar um desses comandos usando um ponto de exclamação (`!`). Tenha em mente que, ao usar um ponto de exclamação, o comando roda cegamente, sem lhe dar uma oportunidade para confirmar o comando que você está referenciando. Há

várias maneiras de executar um comando imediatamente a partir dessa lista, incluindo as seguintes:

- `! n` — Executa o número de comando. Substitua o *n* pelo número da linha de comando e essa linha é executada. Por exemplo, aqui está como repetir o comando `date` indicado como número de comando 382 na listagem do histórico anterior:

```
$ !382
date
Fri Oct 29 21:30:06 PDT 2011
```

- `!!` — Executar comando `previous`. Executa a linha de comando anterior. Veja como você iria imediatamente executar o mesmo comando `date`

```
$ !!
date
Fri Oct 29 21:30:39 PDT 2012
```

- `!?string?` - Executa o comando que contém a string. Isso executa o comando mais recente que contém uma determinada string de caracteres. Por exemplo, você pode executar o comando `date` novamente apenas procurando por parte dessa linha de comando da seguinte maneira:

```
$ !?dat?
date
Fri Oct 29 21:32:41 PDT 2011
```

Em vez de apenas executar uma linha de comando `history` imediatamente, você pode lembrar de uma determinada linha e editá-la. Você pode usar as seguintes teclas ou combinações de teclas para fazer isso, como exhibe a Tabela 3.4.

TABELA 3.4 Combinações de Tecla para Usar o Histórico de Comandos

Tecla(s)	Nome da função	Descrição
Teclas de seta (↑ e ↓)	Passo a passo	Pressione as teclas de seta para cima e para baixo a fim de percorrer cada linha de comando em sua lista de histórico para chegar ao que você deseja. (Ctrl+P e Ctrl+N fazem as mesmas funções, respectivamente.)
Ctrl+R	Pesquisa incremental inversa	Depois de pressionar essas teclas, você insere uma string de pesquisa para fazer uma pesquisa inversa. À medida que digita a string, aparece uma linha de comando correspondente que você pode executar ou editar.
Ctrl+S	Pesquisa incremental para frente	O mesmo que a função anterior, mas procura para frente. (Isso pode não funcionar em todos os casos.)
Ctrl+P	Pesquisa inversa	Depois de pressionar essas teclas, você insere uma string de pesquisa para fazer uma pesquisa inversa. Digite uma string e pressione Enter para ver a linha de comando mais recente que inclui essa string.
Ctrl+N	Pesquisa para frente	O mesmo que a função anterior, mas procura para

Outra maneira de trabalhar com a sua lista de histórico é usar o comando `fc`. Digite `fc` seguido de um número de linha do histórico e essa linha de comando é aberta em um editor de texto (`vi` por padrão, digite `:wq` para salvar e sair ou `:q!` para simplesmente sair se você empacar no `vi`). Faça as alterações que você quer. Quando você sai do editor, o comando é executado. Você também pode dar um intervalo de números de linha (por exemplo, `fc 100 105`). Todos os comandos são abertos no seu editor de textos e então executados um depois do outro quando você encerra o editor.

Depois de fechar seu shell, a lista de histórico é armazenado no `.bash_history` em seu diretório home. Por padrão, até 1.000 comandos são armazenados no histórico para você.

ta

mas pessoas desativam o recurso de histórico para o usuário `root`, definindo o `HISTFILE` para `/dev/null` simplesmente deixando `HISTSIZE` em branco. Isso evita que informações sobre as atividades do usuário `root` sejam potencialmente exploradas. Se você é um usuário administrativo com privilégios de `root`, você também pode considerar a possibilidade de esvaziar seu arquivo ao sair, pelas mesmas razões. Além disso, como o histórico do shell é armazenado permanentemente quando ele é fechado corretamente, você pode impedir que o histórico do shell seja armazenado, eliminando um shell. Por exemplo, para matar um shell com o processo de ID 1234, você pode usar `kill -9 1234` a partir de qualquer shell.

Conectando e expandindo comandos

Um recurso verdadeiramente poderoso do shell é a capacidade de redirecionar a entrada e saída de comandos para e de outros comandos e arquivos. Para permitir agrupar comandos, o shell utiliza metacaracteres. Um *metacaractere* é um caractere digitado que tem um significado especial para o shell para conectar comandos ou solicitar expansão.

Metacaracteres incluem o caractere de barra vertical ou *pipe* (`|`), o “e” comercial (`&`), ponto e vírgula (`;`), parêntese direito (`)`, parêntese esquerdo (`(`), sinal de menor que (`<`) e maior que (`>`). As próximas seções descrevem como usar metacaracteres na linha de comando para mudar a maneira como os comandos se comportam.

Redirecionamento entre os comandos

O metacaractere de redirecionamento (`|`) conecta a saída de um comando à entrada de outro comando. Isso permite que você tenha um comando funcionando sobre alguns dados e, então, o próximo comando lidando com os resultados. Eis um exemplo de uma linha de comando que inclui barras verticais:

```
$ cat /etc/passwd | sort | less
```

Esse comando lista o conteúdo do arquivo `/etc/passwd` e redireciona a saída para o comando `sort`.

O comando `sort` leva os nomes que começam cada linha do arquivo `/etc/passwd`, classificando-os em ordem alfabética, e redireciona a saída para o comando `less` (para a página por meio da saída).

As barras verticais são uma excelente ilustração de como o UNIX, o predecessor do Linux, foi criado como um sistema operacional composto de blocos de construção. Uma prática padrão no UNIX era conectar utilitários de maneiras diferentes para realizar trabalhos. Por exemplo, antes dos processadores de texto gráficos, os usuários criavam arquivos de texto simples que incluíam macros para indicar a formatação. Para ver como o documento realmente aparecia, eles utilizavam um comando como o seguinte:

```
$ gunzip < /usr/share/man/man1/grep.1.gz | nroff -c -man | less
```

Nesse exemplo, o conteúdo da página `man grep (grep.1.gz)` é direcionado para o comando `gunzip` a ser extraído. A saída de `gunzip` é redirecionada para o comando `nroff` para formatar a página `man` usando a macro `manual (-man)`. A saída é redirecionada para o comando `less` para exibir a saída. Como o arquivo que está sendo exibido está em texto simples, você poderia ter substituído qualquer número de opções para trabalhar com o texto antes de exibi-lo e também classificar, alterar ou excluir algum conteúdo ou criar texto de outros documentos. A chave é que, em vez de todos esses recursos estarem em um programa, você obtém resultados a partir do redirecionamento da entrada e saída entre vários comandos.

Comandos sequenciais

Eventualmente você pode querer que uma sequência de comandos seja executada, com um comando sendo completado antes de o comando seguinte começar. Você pode fazer isso digitando vários comandos na mesma linha de comando e separando-os com ponto e vírgulas (;):

```
$ date ; troff -me verylargedocument | lpr ; date
```

Nesse exemplo, eu estava formatando um documento enorme e quis saber quanto tempo isso levaria. O primeiro comando (`date`) exibiu a data e hora antes de a formatação ter começado. O comando `troff` formatou o documento e depois redirecionou a saída para a impressora. Quando a formatação terminou, a data e a hora foram impressas novamente (então eu sabia quanto tempo o comando `troff` levou para completar).

Outro comando útil para adicionar ao final de uma longa linha de comando é `mail`. Você pode adicionar o seguinte ao final de uma linha de comando:

```
; mail -s "Finished the long command" chris@example.com
```

Assim, por exemplo, uma mensagem de e-mail é enviada para o usuário que você escolher, após a conclusão do comando.

Comandos em segundo plano

Alguns comandos podem demorar um pouco para serem concluídos. Às vezes você talvez não queira que seu shell espere que um comando termine. Nesses casos, você pode fazer os comandos executarem em segundo plano, usando o E comercial (&).

Comandos de formatação de texto (como `nroff` e `troff`, descrito anteriormente) são exemplos frequentemente executados em segundo plano para formatar um documento grande. Você também pode criar seus próprios scripts que são executados em segundo plano para verificar continuamente por certos acontecimentos, como o disco rígido encher ou usuários específicos efetuarem login.

Eis um exemplo de um comando a ser executado em segundo plano:

```
$ troff -me verylargedocument | lpr &
```

Não feche o shell até que o processo seja concluído, ou ele será destruído. Outras maneiras de gerenciar processos em execução em primeiro e segundo planos são descritas no Capítulo 6, “Gerenciando processos em execução”.

Expandindo comandos

Com a substituição de comando, você pode ter a saída de um comando interpretada pelo shell em vez de pelo próprio comando. Dessa maneira, você pode tornar a saída padrão de um comando em um argumento para outro comando. As duas maneiras de substituição de comando são `$(comando)` e ``comando`` (crases, não aspas simples).

O comando nesse caso pode incluir opções, metacaracteres e argumentos. Eis um exemplo de como usar a substituição de comando:

```
$ vi $(find /home | grep xyzzy)
```

Nesse exemplo, a substituição de comando é feita antes de o comando `vi` ser executado. Primeiro, o comando `find` começa no diretório `/home` e imprime todos os arquivos e diretórios abaixo desse ponto no sistema de arquivos. A saída é redirecionada para o comando `grep`, que filtra todos os arquivos, exceto para aqueles que incluem a string `xyzzy` em seu nome. Por fim, o comando `vi` abre todos os nomes que incluem `xyzzy` para a edição (um de cada vez). (Se executar isso e não estiver familiarizado com o `vi`, você pode digitar: `:q!` para sair do arquivo.)

Esse exemplo em particular é útil se você quiser editar um arquivo cujo nome você conhece, mas não a localização. Desde que a string não seja comum, você pode encontrar e abrir cada ocorrência de um nome de arquivo existente abaixo de um ponto escolhido no sistema de arquivos. (Em outras palavras, não use `grep` a partir do sistema de arquivos raiz ou você vai encontrar e tentar editar vários milhares de arquivos.)

Expandindo expressões aritméticas

Pode haver ocasiões em que você deseja passar resultados aritméticos para um comando. Há duas formas que você pode utilizar para expandir uma expressão aritmética e passá-la para o shell: `$ [expressão]`

ou `$ (expressão)` .Exemplo:

```
$ echo "I am $(2012 - 1957) years old."  
I am 55 years old.
```

O shell interpreta a primeira expressão aritmética (2012 - 1957) e depois passa essas informações para o comando `echo`. O comando `echo` exibe o texto, com os resultados da aritmética (55) inseridos.

Eis um exemplo da outra forma:

```
$ echo "There are $(ls | wc -w) files in this directory."  
There are 14 files in this directory.
```

Isso lista o conteúdo do diretório atual (`ls`) e executa o comando de contagem de palavras para contar o número de arquivos encontrados (`wc -w`). O número resultante (14 nesse caso) é ecoado de volta com o resto da sentença sendo mostrado.

Expandindo variáveis

Variáveis que armazenam informações dentro do shell podem ser expandidas usando o metacaractere cifrão (`$`). Quando você expande uma variável de ambiente em uma linha de comando, o valor da variável é impresso em vez do próprio nome da variável, como a seguir:

```
$ ls -l $BASH  
-rwxr-xr-x 1 root root 625516 Dec 5 11:13 /bin/bash
```

Usando `$BASH` como um argumento para `ls -l` causa uma listagem do comando `bash` para ser impressa.

Usando variáveis de shell

O próprio shell armazena informações que podem ser úteis para a sessão de shell do usuário naquilo que é chamado de *variáveis*. Exemplos de variáveis incluem `$SHELL` (que identifica o shell que você está usando), `$PS1` (que define a sua prompt de shell) e `$MAIL` (que identifica a localização da caixa de correio).

Você pode ver todas as variáveis definidas para seu shell atual digitando o comando `set`. Um subconjunto de variáveis locais é chamado de *variáveis de ambiente*, que são exportadas para quaisquer novos shells abertos a partir do shell atual. Digite `env` para ver variáveis de ambiente.

Você pode digitar `echo $VALOR`, em que o **VALOR** é substituído pelo nome de uma variável de ambiente especial que você deseja listar. E como há sempre múltiplas maneiras de fazer qualquer coisa no Linux, você também pode digitar `declare` para obter uma lista das variáveis de ambiente atuais e seus valores, juntamente com uma lista de funções de shell.

Além daqueles que você mesmo define, arquivos de sistema definem variáveis que guardam as coisas, tais como locais de arquivos de configuração, caixas de correio e diretórios de caminho. Elas também podem armazenar valores para prompts de shell, o tamanho da lista de histórico e o tipo do sistema operacional. Você pode consultar o valor de qualquer uma dessas variáveis precedendo-a com um cifrão (\$) e colocá-la em qualquer lugar na linha de comando. Por exemplo:

```
$ echo $USER  
chris
```

Esse comando imprime o valor da variável de `USER`, que contém seu nome de usuário (Chris). Substitua qualquer outro valor para `USER` para imprimir seu valor.

Quando você inicia um shell (entrando no meio de um console virtual ou abrindo uma janela de terminal), muitas variáveis de ambiente já estão definidas. A Tabela 3.5 exibe algumas variáveis que são definidas quando você usa um shell `bash` ou que podem ser definidas por você para utilizar com recursos diferentes.

TABELA 3.5 Variáveis de Ambiente Shell Comuns

Variável	Descrição
<code>SH</code>	Contém o caminho completo do comando <code>bash</code> . Esse é geralmente <code>/bin/bash</code> .
<code>SH_VERSION</code>	Um número da versão atual do comando <code>bash</code> .
<code>ID</code>	Esse é o número de ID de usuário efetivo do usuário atual. Ele é atribuído quando o shell inicia, com base na entrada do usuário no arquivo <code>/etc/passwd</code> .
<code>EDIT</code>	Se definido, indica o editor de texto usado pelo comando <code>fc</code> para editar comandos do histórico. Se essa variável não estiver definida, o comando <code>vi</code> é usado.
<code>STFILE</code>	A localização do seu arquivo de histórico. Ele é geralmente localizado em <code>\$HOME/.bash_history</code> .
<code>STFILESIZE</code>	O número de entradas de histórico que pode ser armazenado. Depois que esse número é alcançado, os comandos antigos são descartados. O valor padrão é 1000.
<code>STCMD</code>	Isso retorna o número do comando atual na lista de histórico.
<code>ME</code>	Esse é seu diretório inicial. É seu diretório de trabalho atual cada vez que você entrar ou digitar o comando <code>cd</code> com qualquer opção.
<code>STTYPE</code>	Um valor que descreve a arquitetura do computador em que o sistema Linux está executando. Para PCs compatíveis com Intel, esse valor é <code>i386</code> , <code>i486</code> , <code>i586</code> , ou <code>i686</code> , ou algo como <code>i386-linux</code> . Para máquinas AMD de 64 bits, o valor é <code>x86_64</code> .
<code>IL</code>	Essa é a localização do seu arquivo de caixa de correio. O arquivo é geralmente seu nome de usuário no <code>/var/spool/mail</code> directory.
<code>DPWD</code>	O diretório que era o de trabalho antes de você mudar para o atual.

TYPE	Um nome que identifica o sistema operacional atual. Para o Linux Fedora, o valor OSTYPE é tanto linux como linux-gnu, dependendo do tipo de shell que você está usando. (O bash também pode ser executado em outros sistemas operacionais.)
TH	Uma lista de diretórios separados por dois-pontos usada para encontrar os comandos que você digita. O valor padrão para usuários normais é: /bin:/usr/bin:/usr/local/bin:/usr/bin/X11:/usr/X11R6/bin:~/bin. Você precisa digitar o caminho completo ou o caminho relativo para um comando que deseja executar mas que não está em seu PATH. Para o usuário root, o valor também inclui /sbin, /usr/sbin e /usr/local/sbin.
ID	O processo de identificação do comando que iniciou o shell atual (por exemplo, a janela de terminal contendo o shell).
PROMPT_COMMAND	Pode ser definido como um nome de comando que é executado cada vez antes de seu prompt de shell ser exibido. Definir PROMPT_COMMAND=date lista a data/hora atual antes do prompt aparecer.
1	Configura o valor de seu prompt de shell. Há muitos itens que podem ser interpretados no seu prompt (data, hora, nome de usuário, nome de host e assim por diante). Às vezes, um comando requer avisos adicionais, que você pode definir com as variáveis PS2, PS3, assim por diante.
D	Esse é o diretório que é atribuído como seu diretório atual. Esse valor muda a cada vez que você muda diretórios usando o comando cd.
NDOM	Acessar essa variável faz com que um número aleatório seja gerado. O número está entre 0 e 99999.
CONDS	O número de segundos desde o momento em que o shell foi iniciado.
LVL	O número de níveis de shell associados com a sessão de shell atual. Quando você efetuar login no shell, o SHLVL é 1. Cada vez que você iniciar um novo comando bash (como, por exemplo, usar su para se tornar um novo usuário, ou simplesmente digitando bash), esse número é incrementado.
OUT	Pode ser configurado como um número que representa o número de segundos que o shell pode permanecer inativo sem receber entrada. Depois que o número de segundos é alcançado, o shell é encerrado. Esse é um recurso de segurança que torna menos provável que shells não supervisionados sejam acessados por pessoas não autorizadas. (Isso deve ser configurado no shell de login para realmente fazer com que o shell efetue o logout do usuário.)

Criação e uso de aliases

Usando o comando `alias`, você pode efetivamente criar um atalho para qualquer comando e as opções que deseja executar mais tarde. Você pode adicionar e listar aliases com o comando `alias`. Considere os seguintes exemplos do uso de `alias` a partir de um shell bash:

```
$ alias p='pwd ; ls -CF'
$ alias rm='rm -i'
```


No primeiro exemplo, a letra `p` é atribuída para executar o comando `pwd`, depois para executar `ls -CF` para imprimir o diretório de trabalho atual e listar seu conteúdo em forma de coluna. O segundo exemplo executa o comando `rm` com a opção `-i` cada vez que você simplesmente digitar `rm`. (Esse é um alias que costuma ser configurado automaticamente para o usuário `root`. Em vez de apenas excluir arquivos, você será solicitado para cada exclusão de arquivo individual. Isso impede que a remoção automática de todos os arquivos de um diretório ao digitar algo por engano como `rm`.)

Enquanto você estiver no shell, pode verificar quais aliases são definidos digitando o comando `alias`. Se você quiser remover um alias, digite **`unalias`**. (Lembre-se de que se o `alias` é definido em um arquivo de configuração, ele será definido novamente quando você abrir outro shell.)

Encerrando o shell

Para sair do shell quando terminar, digite **`exit`**, ou pressione `Ctrl+D`. Se você for para o shell a partir de uma janela de terminal e estiver usando o shell original a partir dessa janela, sair faz com que a janela Terminal se feche. Se você estiver em um console virtual, o shell se fecha e você é levado de volta para um prompt de login.

Se você tiver vários shells abertos a partir da mesma sessão de shell, sair de um shell simplesmente faz você voltar ao shell que carregou o shell atual. Por exemplo, o comando `su` abre um shell como um novo usuário. Sair desse shell simplesmente traz você de volta ao shell original.

Criando Seu Ambiente de Shell

Você pode sintonizar seu shell para ajudá-lo a trabalhar de maneira mais eficiente. Você pode definir aliases que criam atalhos para suas linhas de comando e variáveis de ambiente favoritas para armazenarem informações. Adicionando essas configurações aos arquivos de configuração do shell, você pode ter as configurações disponíveis sempre que abrir um shell.

Configurando seu shell

Vários arquivos de configuração suportam a maneira como o shell se comporta. Alguns dos arquivos são executados para cada usuário e cada shell, enquanto outros são específicos do usuário que cria o arquivo de configuração. A Tabela 3.6 exibe os arquivos que são de interesse para qualquer pessoa usando o shell `bash` no Linux.

TABELA 3.6 Arquivos de Configuração do Bash

Arquivo	Descrição
<code>tc/profile</code>	Configura as informações de ambiente do usuário para cada usuário. É executado quando você faz login pela primeira vez. Esse arquivo fornece valores para seu caminho, além de definir as variáveis de ambiente para coisas como o local de sua caixa de correio e o tamanho dos arquivos de seu histórico. Por fim, o arquivo <code>/etc/profile</code> reúne as configurações do

shell de arquivos de configuração no diretório `/etc/profile.d`.

<code>tc/bashrc</code>	Executa para cada usuário que roda o shell bash, sempre que um shell bash é aberto. Ele configura o prompt padrão e pode adicionar um ou mais aliases. Valores nesse arquivo podem ser substituídos por informações no arquivo <code>~/ .bashrc</code> de cada usuário.
<code>. bash_profile</code>	Utilizado por cada usuário para inserir informações que são específicas do seu uso do shell. É executado apenas uma vez: quando o usuário faz login. Por padrão, configura algumas variáveis de ambiente e executa o arquivo <code>.bashrc</code> do usuário. Esse é um bom lugar para adicionar variáveis de ambiente, porque, uma vez configuradas, elas são herdadas por shells futuros.
<code>. bashrc</code>	Contém informações específicas dos shells bash do usuário. Ele é lido quando você efetua o login e também todas as vezes que abre um novo shell bash. Esse é o melhor local para adicionar aliases para o shell usar.
<code>. bash_logout</code>	Executa cada vez que você faz logout (sai do último shell bash). Por padrão, ele simplesmente limpa sua tela.

Para alterar os arquivos `/etc/profile` ou `/etc/bashrc`, você deve ser o usuário `root`. Os usuários podem alterar as informações nos arquivos `$HOME/.bash_profile`, `$HOME/.bashrc` e `$HOME/.bash_logout` em seus próprios diretórios iniciais.

Até que aprenda a usar o editor `vi`, descrito no Capítulo 5, “Trabalhando com arquivos de texto”, você pode usar um editor simples chamado `nano` para editar arquivos de texto simples. Por exemplo, digite o seguinte para editar e adicionar coisas ao seu arquivo `$HOME/.bashrc`:

```
$ nano $HOME/.bashrc
```

Com o arquivo aberto no `nano`, mova o cursor para a parte inferior do arquivo (usando a tecla de seta para baixo). Digite a linha que você quer (por exemplo, digite `alias d="date +%D"`). Para salvar o arquivo, pressione `Ctrl+O` (a letra `O`) e para sair pressione `Ctrl+X`. Da próxima vez que você entrar ou abrir um novo shell, será capaz de usar o novo alias (nesse caso, apenas digite `d`). Para disponibilizar no shell atual a nova informação que você acabou de adicionar ao arquivo, digite o seguinte:

```
$ source $HOME/.bashrc
```

As próximas seções fornecem ideias sobre itens que você pode adicionar aos arquivos de configuração do shell. Na maioria dos casos, você pode adicionar esses valores para o arquivo `.bashrc` em seu diretório `home`. Mas se você administra um sistema, pode querer configurar alguns desses valores como padrão para todos os usuários do seu sistema Linux.

Configurando seu prompt

Seu prompt consiste em um conjunto de caracteres que aparecem sempre que o shell está pronto para aceitar um comando. A variável de ambiente `PS1` define o que contém o aviso e é com o que você interage a maior parte do tempo. Se o shell requer entrada adicional, ele usa os valores de `PS2`, `PS3` e

Quando seu sistema Linux é instalado, um prompt costuma ser configurado para conter mais do que apenas um sinal de cifrão ou de jogo da velha. Por exemplo, no Fedora ou no Red Hat Enterprise Linux, o prompt é configurado para incluir as seguintes informações: seu nome de usuário, o nome do host e o nome de base do seu diretório de trabalho atual. Essas informações são cercadas por colchetes e seguidas por um sinal de cifrão (para usuários normais) ou um sinal de cerquilha (para o usuário root). Eis um exemplo desse prompt:

```
[chris@myhost bin]$
```

Se você mudar de diretório, o nome bin mudará para o nome do novo diretório. Da mesma forma, se estivesse conectado como um usuário diferente ou a um host diferente, essas informações mudariam.

Você pode utilizar vários caracteres especiais (indicados ao adicionar uma barra invertida para uma variedade de letras) para incluir diferentes informações no seu prompt. Os caracteres especiais podem ser utilizados para produzir o número do terminal, a data e a hora, bem como outras informações. A Tabela 3.7 fornece alguns exemplos (você pode encontrar mais na página `man` do `bash`).

TABELA 3.7 Caracteres para Adicionar Informações ao Prompt do Bash

Caractere especial	Descrição
	Exibe o número do comando no histórico atual. Isso inclui todos os comandos anteriores armazenados para seu nome de usuário.
	Exibe o número do comando atual. Isso inclui somente os comandos do shell ativo.
	Exibe o prompt do usuário (\$) ou root (#), dependendo de quem é o usuário.
	Exibe apenas o nome de base do diretório de trabalho atual. Por exemplo, se o diretório atual for <code>/var/spool/mail</code> , esse valor simplesmente aparece como <code>mail</code> .
	Precede uma sequência de caracteres não imprimíveis. Isso pode ser usado para adicionar uma sequência de controle de terminal no prompt para coisas como mudar cores, adicionar efeitos de piscar ou colocar caracteres em negrito. (Seu terminal determina as sequências exatas disponíveis.)
	Segue uma sequência de caracteres não imprimíveis.
	Exibe uma barra invertida.
	Exibe o nome do dia, o mês e o número do dia da data atual. Por exemplo: <code>Sat Jan 23</code>
	Exibe o hostname do computador que executa o shell.
	Faz com que uma nova linha ocorra.
<code>nn</code>	Exibe o caractere que se relaciona com o número octal substituindo <code>nnn</code> .
	Exibe o nome do shell atual. Para o shell <code>bash</code> , o valor seria <code>bash</code> .
	Imprime o horário atual em horas, minutos e segundos (por exemplo, <code>10:14:39</code>).
	Imprime seu nome de usuário atual.

ca

Se você está definindo seu prompt temporariamente digitando no shell, deve colocar o valor de `PS1` entre aspas. Por exemplo, você pode digitar `export PS1="[\t \w]\$ "` para ver um prompt parecido com este: `0:26:32 /var/spool]$`.

Para fazer uma alteração permanente em seu prompt, adicione o valor de `PS1` para seu `.bashrc` em seu diretório home (supondo que você está usando o shell bash). Pode já haver um valor `PS1` no arquivo que você pode modificar. Consulte o Bash Prompt HOWTO (<http://www.tldp.org/HOWTO/Bash-Prompt-HOWTO>) para obter informações sobre alteração de cores, comandos e outros recursos do prompt do shell bash.

Adicionando variáveis de ambiente

Você pode querer considerar a adição de algumas variáveis de ambiente para seu `.bashrc`. Elas podem ajudar a tornar o trabalho com o shell mais eficiente e efetivo:

- `TMOUT` — Define o tempo que o shell pode estar inativo antes que o bash saia automaticamente. O valor é o número de segundos em que o shell não recebeu entrada. Isso pode ser um bom recurso de segurança, se você deixar sua mesa de trabalho enquanto ainda está conectado ao Linux. Então, para não ser desconectado enquanto está trabalhando, você pode querer definir o valor para algo como `TMOUT=1800` (para permitir 30 minutos de tempo ocioso). Você pode usar qualquer sessão de terminal para fechar o shell atual após um determinado número de segundos — por exemplo, `TMOUT=30`.
- `PATH` — Conforme descrito anteriormente, a variável `PATH` define os diretórios que estão sendo buscadas para os comandos que você usa. Se você costuma usar diretórios de comandos que não estão no seu caminho, pode adicioná-los permanentemente. Para fazer isso, adicione a variável `PATH` para seu `.bashrc`. Por exemplo, para adicionar um diretório chamado `/getstuff/bin`, adicione o seguinte:

```
PATH=$PATH:/getstuff/bin ; export PATH
```

Esse primeiro exemplo lê todos os diretórios do caminho atual para o novo `PATH` (`$PATH`) e `/getstuff/bin`, adiciona o diretório `/getstuff/bin` e exporta o novo `PATH`.

ênção

Muitas pessoas adicionam o diretório atual ao seu `PATH` acrescentando um diretório identificado simplesmente por ponto (`.`), como segue:

```
PATH=./$PATH ; export PATH
```

permite que você execute comandos no seu diretório atual antes de avaliar qualquer outro comando no caminho (e as pessoas podem estar acostumadas a fazer se usam DOS). Mas o risco de segurança com esse mecanismo é que você pode estar em um diretório que contém um comando que você não pretende executar a partir desse diretório. Por exemplo, uma pessoa mal-intencionada poderia colocar um comando `ls` em um diretório em vez de listar o conteúdo de seu diretório, faz algo desonesto. Por causa disso, a prática de adicionar o ponto aminho é altamente desencorajada.

- **WHATEVER** — Você pode criar suas próprias variáveis de ambiente para criar atalhos no seu trabalho. Escolha qualquer nome que não está sendo utilizado e atribua um valor útil. Por exemplo, se você faz um monte de trabalho com arquivos no diretório `/work/time/files/info/memos`, você pode definir a seguinte variável:

```
M=/work/time/files/info/memos ; export M
```

Você poderia fazer daquele o seu diretório atual, digitando `cd $M`. Você pode executar um programa a partir desse diretório chamado de `hotdog`, digitando `$M/hotdog`. Você pode editar a partir daí um arquivo chamado `bun` digitando `vi $M/bun`.

Obtendo Informações Sobre Comandos

Quando você começa a usar o shell, isso pode ser intimidante. Tudo o que você vê é um prompt. Como você sabe quais comandos estão disponíveis, quais as opções eles usam ou como usar os recursos avançados? Felizmente, há uma grande quantidade de ajuda. Eis alguns lugares que você pode conferir para complementar o que aprendeu neste capítulo.

- **Verifique o PATH** — Digite `echo $PATH`. Você visualiza uma lista dos diretórios contendo comandos imediatamente acessíveis. A listagem do conteúdo desses diretórios exibem a maioria dos comandos padrão do Linux. Por exemplo:

```
$ ls /bin
arch      dd          fusemount  loadkeys   mv          rnano       taskset
awk       df          gawk       login      nano        rpm         tcsh
basename dmesg      gettext    ls          netstat     rvi         touch
bash      dnsdomainname  grep       lsblk      nice        rview       true
cat       domainname    gtar       lscgroup   nisdomainname  sed         umount
chgrp     echo         gunzip     lssubsys   ping        setfont     uname
chmod     ed           gzip       mail       ping6       setserial   unlink
chown     egrep        hostname   mailx      ps          sh          usleep
cp        env          ipcalc     mkdir      pwd         sleep       vi
cpio      ex           kbd_mode   mknod      readlink    sort        view
csh       false       keyctl     mktemp     red         stty        scat
cut       fgrep        kill       more       redhat_lsb_init  su          ssh
dash      find         link       mount      rm          sync
date      findmnt      ln          mountpoint  rmdir       tar
```

- **Use o comando help** — Alguns comandos são predefinidos no shell; portanto, não aparecem em um diretório. O comando `help` lista os comandos e programas de opções disponíveis com cada um deles. (Digite `help | less` para percorrer a lista.) Para obter ajuda com um determinado comando interno, digite `help comando`, substituindo `comando` pelo nome que lhe interessa. O comando de `help` funciona apenas com o shell `bash`.

- **Use `--help` com o comando** — Muitos comandos incluem uma opção `--help` que você pode usar para obter informações sobre como o comando é usado. Por exemplo, se você digitar `date --help | less`, o resultado mostra não apenas opções, mas também formatos de tempo que você pode usar com o comando `date`. Outros comandos simplesmente usam uma opção `-h`, como `fdisk -h`.
- **Use o comando `info`** — O comando `info` é outra ferramenta para exibir informações sobre comandos do shell. O comando `info` pode mover-se entre uma hierarquia de nós para encontrar informações sobre os comandos e outros itens. Nem todos os comandos têm informações disponíveis no banco de dados, mas às vezes podem ser encontradas mais informações nele do que em uma página `man`.
- **Use o comando `man`** — Para saber mais sobre um determinado comando, digite **`man comando`**. (Substitua **`comando`** pelo nome do comando que você deseja.) Uma descrição do comando e das suas opções aparece na tela.

Páginas `man` são a maneira mais comum de obter informações sobre comandos, bem como sobre outros componentes básicos de um sistema Linux. As páginas `man` são divididas nas categorias listadas na Tabela 3.8. Como um usuário regular, você estará mais interessado nas páginas `man` na seção 1. Como administrador do sistema, você também estará interessado nas seções 4 e 8, e, ocasionalmente, na seção 4. Programadores estarão interessados nas páginas `man` das seções 2 e 3.

TABELA 3.8 Seções das Páginas Man

Numero da seção	Nome da seção	Descrição
	Comandos do usuário	Comandos que podem ser executados a partir do shell por um usuário regular (em geral, nenhum privilégio administrativo é necessário).
	Chamadas de sistema	Funções de programação usadas em um aplicativo para fazer chamadas ao kernel.
	Funções da biblioteca C	Funções de programação que oferecem interfaces para bibliotecas de programação específicas (tais como as de determinadas interfaces gráficas ou outras bibliotecas que operam no espaço do usuário).
	Dispositivos e arquivos especiais	Nós do sistema de arquivos que representam dispositivos de hardware (como terminais ou unidades de CD) ou dispositivos de software (como geradores de números aleatórios).
	Formatos de arquivos e convenções	Tipos de arquivos (como um arquivo gráfico ou de processamento de texto) ou arquivos de configuração específicos (como o arquivo <code>passwd</code> ou <code>group</code>).
	Jogos	Jogos disponíveis no sistema.
	Diversos	Visão geral de tópicos como protocolos, sistemas de arquivos, padrões de conjunto de caracteres etc.
	Ferramentas e daemons de	Comandos que exigem <code>root</code> ou outros privilégios

Opções para o comando `man` permitem pesquisar o banco de dados da página `man` ou exibir páginas `man` na tela. Eis alguns exemplos de comandos e opções de `man`:

```
$ man -k passwd
...
passwd          (1) - update user's authentication tokens
passwd          (5) - password file
$ man passwd
$ man 5 passwd
```

Usando a opção `-k`, você pode procurar o nome e as seções de resumo de todas as páginas do `man` instaladas no sistema. Há cerca de uma dúzia de páginas `man` que incluíam `passwd` no nome ou descrição de um comando.

Digamos que as duas páginas `man` que estou interessado são o comando `passwd` (na seção 1 das páginas `man`) e o arquivo `passwd` (na seção 5) das páginas `man`. Como simplesmente digitar `man password` exibiu a página da seção 1, eu teria de pedir explicitamente a página `man` da seção 5 se quisesse ver isso no lugar (`man 5 passwd`).

Enquanto está exibindo uma página `man`, você pode ver diferentes partes do arquivo usando as teclas Page Down e Page Up (para mover uma página de cada vez). Use a tecla Enter ou as teclas de seta para cima e para baixo para mover-se uma linha de cada vez. Pressione uma barra (/) e, então, digite um termo para pesquisá-lo no documento. Pressione **n** para repetir a pesquisa para frente ou **N** para repetir a pesquisa para trás. Para sair da página `man`, digite **q**.

Resumo

Para se tornar um especialista em Linux, você deve ser capaz de usar o shell para digitar comandos. Esse capítulo focaliza o shell `bash`, que é o mais comumente utilizado com sistemas Linux. Neste capítulo, você aprendeu como os comandos são estruturados e como muitos recursos especiais, tais como variáveis, completamento de comando e aliases são utilizados.

O próximo capítulo descreve como mover-se pelo sistema de arquivos do Linux a partir da linha de comando do shell.

Exercícios

Use esses exercícios para testar seus conhecimentos de como usar o shell. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostradas no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. A partir do Desktop, alterne para o segundo console virtual e efetue login com sua conta de usuário. Execute alguns comandos. Então, saia do shell e volte ao desktop.

2. Abra uma janela de terminal e mude a cor da fonte para vermelho e o fundo para amarelo.
3. Encontre o local do comando `mount` e a página `man tracepath`.
4. Digite os três seguintes comandos e depois recupere e altere os comandos como descrito:

```
$ cat /etc/passwd
$ ls $HOME
$ date
```

- Use o recurso de recordar linha de comando para lembrar o comando `cat` e mudar `/etc/passwd` para `/etc/group`.
 - Lembre-se do comando `ls`, determine a maneira de listar os arquivos por tempo (utilizando a página de `man`) e adicione essa opção para a linha de comando `ls $HOME`.
 - Adicione indicadores de formato para o comando `date` para exibir a saída de data como **dia/mês/ano**.
5. Execute o seguinte comando, digitando o mínimo de caracteres possível (usando completamente de comando com a tecla `Tab`):

```
basename /usr/share/doc/.
```
 6. Use o comando `cat` para listar o conteúdo do arquivo `/etc/services` e redirecione esses conteúdos ao comando `less` para que você possa folheá-lo (pressione `q` para sair quando você tiver terminado).
 7. Execute o comando `date` de tal maneira que a saída do comando produza o dia, mês, data e ano. Faça com que isso seja lido por outra linha de comando, resultando em um texto que aparece da seguinte maneira (sua data, naturalmente, será diferente): `Today is Thursday, December 08, 2011`.
 8. Usando variáveis, descubra como seu `hostname`, seu nome de usuário, seu `shell` e seus diretórios iniciais estão atualmente configurados.
 9. Crie um alias chamado `mypass` que exibe o conteúdo do arquivo `/etc/passwd` em sua tela de tal maneira que ele esteja disponível a cada vez que você entrar ou abrir um novo `shell` de sua conta de usuário.
 10. Mostre a página `man` para a chamada de sistema `mount`.

Movendo-se pelo sistema de arquivos

NESTE CAPÍTULO

Aprendendo sobre o sistema de arquivos Linux

Listando atributos de arquivo e diretório

Criando arquivos e diretórios

Listando e alterando permissões e posse

Fazendo cópias e movendo arquivos

O sistema de arquivos Linux é a estrutura em que todas as informações sobre seu computador estão armazenadas. Na verdade, uma das propriedades que definem os sistemas UNIX, em que se baseia o Linux, é que quase tudo o que você precisa identificar em seu sistema (dados, comandos, links simbólicos, dispositivos e diretórios) é representado por itens nos sistemas de arquivos. Saber onde as coisas estão e entender como manipular o sistema de arquivos a partir do shell, são habilidades fundamentais em Linux.

No Linux, os arquivos são organizados dentro de uma hierarquia de diretórios. Cada diretório pode conter arquivos e outros diretórios. Você pode referenciar qualquer arquivo ou diretório usando um caminho completo (por exemplo, `/home/joe/myfile.txt`) ou um caminho relativo (por exemplo, se `/home/joe` fosse seu diretório atual, você poderia simplesmente referenciar o arquivo como `myfile.txt`).

Se você mapeasse os arquivos e diretórios no Linux, ele seria parecido com uma árvore de cabeça para baixo. No alto, está o diretório raiz (que não deve ser confundido com o usuário `root`), que é representado por uma única barra (`/`). Abaixo disso, está um conjunto de diretórios comuns no sistema Linux, como `bin`, `dev`, `home`, `lib` e `tmp`, para citar alguns. Cada um desses diretórios, bem como diretórios adicionados ao diretório-raiz, pode conter subdiretórios.

A Figura 4.1 ilustra como o sistema de arquivos Linux é organizado como uma hierarquia. Para demonstrar como os diretórios são conectados, a figura mostra um diretório `/home` que contém um subdiretório para o usuário `joe`. Dentro do diretório `joe` estão o `Desktop`, `Documents` e outros subdiretórios. Para referenciar um arquivo chamado `memo1.doc` no diretório `memos`, você pode digitar o caminho completo `/home/joe/Documents/memos/memo1.doc`. Se seu diretório atual for `/home/joe/Documents/memos`, consulte o arquivo simplesmente como `memo1.doc`.

tema de arquivos Linux é organizado como uma hierarquia de diretórios.

Eis alguns dos diretórios do Linux que podem lhe interessar:

- `/bin` — Contém comandos comuns de usuários Linux, como `ls`, `sort`, `date` e `chmod`.
- `/boot` — Contém o kernel inicializável do Linux e os arquivos de configuração do carregador de inicialização (GRUB).
- `/dev` — Contém arquivos que representam os pontos de acesso a dispositivos em seus sistemas. Esses incluem dispositivos `terminal` (`tty*`), disquetes (`fd*`), discos rígidos (`hd*` ou `sd*`), RAM (`ram*`) e CD-ROM (`cd*`). Os usuários podem acessar esses dispositivos diretamente através desses arquivos de dispositivo, mas alguns aplicativos ocultam dos usuários finais o nome dos dispositivos reais.
- `/etc` — Contém os arquivos de configuração administrativa. A maioria desses arquivos são de texto simples que, desde que o usuário tenha a devida permissão, podem ser editados em qualquer editor de texto.
- `/home` — Contém diretórios atribuídos a cada usuário regular com uma conta de login. (O usuário `root` é uma exceção, usando `/root` como seu diretório inicial.)
- `/media` — Fornece uma localização padrão para dispositivos de automontagem (mídia removível em particular). Se o produto possuir um nome de volume, esse nome é normalmente usado como ponto de montagem. Por exemplo, um drive USB com um nome de volume `myusb` seria montado em `/media/myusb`.
- `/lib` — Contém bibliotecas compartilhadas requeridas por aplicativos em `/bin` e `/sbin` para inicializar o sistema.
- `/mnt` — Um ponto de montagem comum para muitos dispositivos, antes de ser suplantado pelo diretório `/media` padrão. Alguns sistemas Linux inicializáveis ainda usam esse diretório para montar partições de disco rígido e sistemas de arquivos remotos. Muitas pessoas ainda usam esse diretório para montar temporariamente sistemas de arquivos locais ou remotos que não são montados de forma permanente.
- `/misc` — Um diretório usado às vezes para montar sistemas de arquivos mediante solicitação.
- `/opt` — Estrutura de diretórios disponíveis para armazenar software suplementar.
- `/proc` — Contém informações sobre os recursos do sistema.
- `/root` — Representa o diretório do usuário `root`. O diretório inicial de `root` não reside sob `/home` por razões de segurança.

- `/sbin` — Contém comandos administrativos e processos de daemon.
- `/tmp` — Contém arquivos temporários usados pelos aplicativos.
- `/usr` — Contém a documentação do usuário, jogos, arquivos gráficos (X11), bibliotecas (`lib`) uma variedade de outros comandos e arquivos que não são necessários durante o processo de inicialização.
- `/var` — Contém diretórios de dados utilizados por vários aplicativos. Em particular, esse é o lugar onde você coloca os arquivos que você compartilha, como um servidor de FTP (`/var/ftp`) ou um servidor web (`/var/www`). Ele também contém todos os arquivos de log do sistema (`/var/log`) e arquivos de spool `/var/spool` (como `mail`, `cups` e `news`).

Os sistemas de arquivos nos sistemas operacionais DOS ou Microsoft Windows diferem da estrutura de arquivo do Linux, como explica o quadro “Sistemas de arquivos Linux *versus* sistemas de arquivos baseados no Windows”.

Sistemas de arquivos Linux versus sistemas de arquivos baseados no Windows

Embora semelhante em muitos aspectos, o sistema de arquivos Linux tem algumas diferenças marcantes em relação aos sistemas de arquivos usados nos sistemas operacionais MS-DOS e Windows. Eis algumas:

- Nos sistemas operacionais MS-DOS e Windows, as letras de unidade representam dispositivos de armazenamento diferentes (por exemplo, A: é uma unidade de disquete e C: é um disco rígido). No Linux, todos os dispositivos de armazenamento estão conectados com a hierarquia do sistema de arquivos. Assim, o fato de que tudo de `/usr` pode estar em um disco rígido separado ou que `/mnt/remotel` é um sistema de arquivos de outro computador é invisível para o usuário.
- Barras, em vez de barras invertidas, são utilizadas para separar nomes de diretório no Linux. Portanto, `C:\home\joe` em um sistema da Microsoft é `/home/joe` em um sistema Linux.
- Nomes de arquivos quase sempre têm sufixos em DOS (como `.txt` para arquivos de texto ou `.doc` para arquivos de processamento de texto). Embora às vezes você possa usar essa convenção em Linux, sufixos de três caracteres não têm significado exigido em Linux. Ele podem ser úteis para identificar um tipo de arquivo. Muitos aplicativos Linux e ambientes de desktop usam sufixos de arquivo para determinar o conteúdo de um arquivo. No Linux, porém, as extensões de comando do DOS como `.com`, `.exe` e `.bat` não significam necessariamente um executável. (Sinalizadores, ou flags, de permissão tornam os arquivos Linux executáveis.)
- Cada arquivo e diretório em um sistema Linux tem permissões e posses associadas a ele. A

segurança varia entre sistemas Microsoft. Como o DOS e o Microsoft Windows começaram como sistemas de um único usuário, a posse de arquivo não foi construída para esses sistemas quando foram concebidos. Distribuições posteriores adicionaram recursos como atributos de arquivo e de pasta para resolver esse problema.

Usando comandos básicos do sistema de arquivos

Quero apresentá-lo a alguns comandos simples relacionados com o sistema de arquivos para começar. Se você quiser acompanhar, faça login e abra um shell. Ao efetuar login em um sistema Linux e abrir um shell, você é levado para seu diretório inicial. Como um usuário de Linux, a maioria dos arquivos que você salva e com os quais você trabalha provavelmente estará no diretório ou subpastas que você cria. A Tabela 4.1 mostra comandos para criar e usar arquivos e diretórios.

TABELA 4.1 Comandos para Criar e Usar Arquivos

Comando	Resultado
<code>cd</code>	Muda para outro diretório.
<code>pwd</code>	Imprime o nome do diretório de trabalho atual.
<code>mkdir</code>	Cria um diretório.
<code>chmod</code>	Altera a permissão sobre um arquivo ou diretório.
<code>ls</code>	Lista o conteúdo de um diretório.

Um dos comandos mais básicos que você utiliza a partir do shell é `cd`. O comando `cd` pode ser usado sem opções (para levá-lo para seu diretório inicial) ou com caminhos completos ou relativos. Considere os seguintes comandos:

```
$ cd /usr/share/
```

```
$ pwd
```

```
/usr/share
```

```
$ cd doc
```

```
/usr/share/doc
```

```
$ cd
```

```
$ pwd
```

```
/home/chris
```

A opção `/usr/share` representa o *caminho absoluto* para um diretório no sistema. Como começa com uma barra (/), esse caminho diz ao shell para iniciar na raiz do sistema de arquivos e levá-lo para o

diretório `share` que existe no diretório `usr`. A opção `doc` para o comando `cd` disse para procurar um diretório chamado `doc` que é relativo ao diretório atual. Portanto, isso tornou `/usr/share/doc` seu diretório atual.

Depois disso, digitando `cd` apenas, você volta para seu diretório inicial. Se você já se perguntou onde você está no sistema de arquivos, o comando `pwd` pode ajudá-lo. Eis algumas outras opções interessantes do comando `cd`:

```
$ cd ~
$ pwd
/home/chris
$ cd ~/Music
$ pwd
/home/chris/Music
$ cd ../../../../usr
$ pwd
/usr
```

O til (~) representa seu diretório inicial. Assim, `cd ~` leva você para lá. Também é possível usar o til para referenciar diretórios relativos ao seu diretório inicial, como `/home/chris/Music` com `~/Music`. Enquanto digitar um nome como uma opção leva você para um diretório abaixo do diretório atual, você pode usar dois pontos (..) para ir para um diretório acima do atual. O exemplo mostrado o leva três níveis de diretório acima (até /) e, então, o leva para o diretório `/usr`.

Os passos a seguir o guiam pelo processo de criação de diretórios dentro do seu diretório inicial e mostram como você pode se mover pelos seus diretórios, com uma menção à configuração das permissões de arquivo apropriadas:

1. Vá para seu diretório inicial. Para tanto, simplesmente digite `cd` em um shell e pressione Enter. (Para outras formas de referenciar seu diretório inicial, consulte o quadro “Identificando diretórios”.)
2. Para certificar-se de que está no seu diretório inicial, digite `pwd`. Quando faço isso, obtenho a seguinte resposta (a sua refletirá seu diretório inicial) :

```
$ pwd
/home/joe
```

3. Crie um novo diretório chamado `test` em seu diretório inicial, como segue:

```
$ mkdir test
```

4. Verifique as permissões do diretório:

```
$ ls -ld test
```

Essa listagem mostra que `test` é um diretório (`d`). O `d` é seguido pelas permissões (`rwxr-xr-x`), que serão explicadas mais adiante na seção “Entendendo as permissões e posse de arquivos”. As demais informações indicam o proprietário (`joe`), o grupo (`sales`) e a data em que os arquivos no diretório foram modificados mais recentemente (24 de janeiro às 12:17).

ta

edora e no Red Hat Enterprise Linux, quando você adiciona um novo usuário, o usuário é atribuído a um grupo de nome, por padrão. Por exemplo, no texto anterior, o usuário `joe` seria atribuído ao grupo `joe`. Essa designação para a atribuição de grupos é referida como esquema de grupo privado do usuário.

Por enquanto, digite o seguinte:

```
$ chmod 700 test
```

Esse passo altera as permissões do diretório para dar-lhe acesso completo e nenhum acesso a todos os outros. (As novas permissões devem exibir `rwX - - - - -`.)

5. Torne o diretório `test` seu diretório atual da seguinte maneira:

```
$ cd test
```

```
$ pwd
```

```
/home/joe/test
```

Se você acompanhou, nesse momento um subdiretório do seu diretório inicial chamado `test` é seu diretório de trabalho atual. Você pode criar arquivos e diretórios no diretório `test`, juntamente com as descrições no restante deste capítulo.

Usando Metacaracteres e Operadores

Quer você esteja listando, movendo, copiando, removendo ou de outra forma atuando sobre os arquivos em seu sistema Linux, alguns caracteres especiais, chamados de metacaracteres e operadores, ajudam-no a trabalhar com arquivos de forma mais eficiente. Metacaracteres podem ajudá-lo a localizar um ou mais arquivos sem digitar completamente o nome de cada arquivo. Operadores permitem que você direcione a informação de um comando ou arquivo para outro comando ou arquivo.

Utilizando metacaracteres para correspondência de arquivo

Para poupar-lhe alguns pressionamentos de tecla e para que você possa facilmente referenciar um grupo de arquivos, o shell `bash` permite usar metacaracteres. Sempre que precisar referenciar um arquivo ou diretório, como para listá-lo, abri-lo ou excluí-lo, você pode usar metacaracteres para localizar os

arquivos que deseja. Eis alguns metacaracteres úteis para corresponder nomes de arquivo:

- `*` — Localiza qualquer número de caracteres.
- `?` — Localiza qualquer caractere.
- `[...]` — Localiza qualquer um dos caracteres entre os colchetes, o que pode incluir uma gama de letras ou números separados por hífen.

Experimente alguns desses metacaracteres de correspondência de arquivo primeiro indo para um diretório vazio (como o diretório `test` descrito na seção anterior) e criando alguns arquivos vazios:

```
$ touch apple banana grape grapefruit watermelon
```

O comando `touch` cria arquivos vazios. Os comandos a seguir mostram como usar metacaracteres de shell com o comando `ls` para localizar nomes de arquivos. Tente os seguintes comandos para ver se você tem as mesmas respostas:

```
$ ls a*
apple
$ ls g*
grape grapefruit
$ ls g*t
grapefruit
$ ls *e*
apple grape grapefruit watermelon
$ ls *n*
banana watermelon
```

O primeiro exemplo localiza qualquer arquivo que começa com um `a` (`apple`). O próximo exemplo combina todos os arquivos que começam com `g` (`grape`, `grapefruit`). Em seguida, os arquivos que começam com `g` e terminam em `t` são localizados (`grapefruit`). Depois, qualquer arquivo que contém um `e` no nome é localizado (`apple`, `grape`, `grapefruit`, `watermelon`). Por fim, qualquer arquivo que contém um `n` é localizado (`banana`, `watermelon`).

Eis alguns exemplos de correspondência de padrões com o ponto de interrogação (`?`):

```
$ ls ???e
apple grape
$ ls g???e*
grape grapefruit
```

O primeiro exemplo localiza qualquer arquivo de cinco caracteres que termina em e (apple, grape). O segundo localiza qualquer arquivo que começa com g e tem como seu quinto caractere um e (grape, grapefruit).

Os exemplos a seguir usam colchetes para fazer a correspondência de padrões:

```
$ ls [abw]*
```

```
apple banana watermelon
```

```
$ ls [agw]*[ne]
```

```
apple grape watermelon
```

No primeiro exemplo, qualquer arquivo começando com a, b ou w é localizado. No segundo, qualquer arquivo que começa com a, g, ou w e também termina com um n ou um e, é localizado. Você também pode incluir intervalos entre colchetes. Por exemplo:

```
$ ls [a-g]*
```

```
apple banana grape grapefruit
```

Aqui, todos os nomes que começam com uma letra de a a g são localizados.

Utilizando metacaracteres para redirecionamento de arquivos

Comandos recebem dados da entrada padrão e os enviam para a saída padrão. Utilizando redirecionamentos (descritos anteriormente), você pode direcionar a saída padrão de um comando para a entrada padrão de outro. Com os arquivos, você pode usar os sinais menor que (<) e maior que (>) para direcionar dados de e para arquivos. Eis os caracteres de redirecionamento de arquivo:

- < — Direciona o conteúdo de um arquivo para o comando. Na maioria dos casos, essa é ação padrão esperada pelo comando e o uso do caractere é opcional; utilizar `less bigfile` é o mesmo que `less < bigfile`.
- > — Direciona a saída padrão de um comando para um arquivo. Se o arquivo existir, o conteúdo dele será sobrescrito.
- 2> — Direciona mensagens de erro padrão (*standard error*) para o arquivo.
- &> — Direciona tanto a saída padrão como o erro padrão para o arquivo.
- >> — Direciona a saída de um comando para um arquivo, adicionando a saída ao final do arquivo existente.

Eis alguns exemplos de linhas de comando em que a informação é dirigida de e para arquivos:

```
$ mail root < ~/.bashrc
```



```
$ man chmod | col -b > /tmp/chmod
```

```
$ echo "I finished the project on $(date)" >> ~/projects
```

No primeiro exemplo, o conteúdo do arquivo `.bashrc` no diretório inicial é enviado em uma mensagem de mail para o usuário `root` do computador. A segunda linha de comando formata a página `man chmod` (usando o comando `man`), remove os espaços extras (`col -b`) e envia a saída para o arquivo `/tmp/chmod` (apagando o arquivo `/tmp/chmod` anterior, se ele existir). O comando final faz com que o seguinte texto seja adicionado ao arquivo `project` do usuário:

```
I finished the project on Sat Jan 22 13:46:49 PST 2011
```

Outro tipo de redirecionamento, referido como *here text* (também chamado *here document*), permite que você digite o texto que pode ser usado como entrada padrão para um comando. *Here documents* envolvem inserir dois sinais de menor que (`<<`) após um comando, seguidos de uma palavra. Toda digitação após a palavra é tomada como entrada do usuário até que a palavra seja repetida em uma linha sozinha. Eis um exemplo:

```
$ mail root cnegus rjones bdecker <<thetext
> I want to tell everyone that there will be a 10 a.m.
> meeting in conference room B. Everyone should attend.
>
> -- James
> thetext
$
```

Esse exemplo envia uma mensagem de e-mail para os usuários `root`, `cnegus`, `rjones`, e `bdecker`. O texto inserido entre `<<thetext` e `thetext` torna-se o conteúdo da mensagem. Uma forma comum de usar *here text* é com um editor de texto para criar ou adicionar um arquivo a partir de dentro de um script:

```
/bin/ed /etc/resolv.conf <<resendit
a
nameserver 100.100.100.100
.
w
q
resendit
```

Com essas linhas adicionadas a um script executado pelo usuário `root`, o editor de texto `ed` adiciona o endereço IP de um servidor DNS ao arquivo `/etc/resolv.conf`.

Uso de caracteres de expansão

Usando chaves ({ }), você pode expandir um conjunto de caracteres em todos os nomes de arquivo, nomes de diretórios ou outros argumentos que você dá aos comandos. Por exemplo, se quiser criar um conjunto de arquivos, como memo1 a memo5, você pode fazer isso da seguinte maneira:

```
$ touch memo{1,2,3,4,5}
$ ls
memo1  memo2  memo3  memo4  memo5
```

Os itens que são expandidos não precisam ser números ou mesmo dígitos individuais. Por exemplo, você poderia usar intervalos de números ou dígitos. Você também pode usar qualquer string de caracteres, desde que os separe com vírgulas. Eis alguns exemplos:

```
$ touch {John,Bill,Sally}-{Breakfast,Lunch,Dinner}
$ ls
Bill-Breakfast Bill-Lunch      John-Dinner Sally-Breakfast
Sally-Lunch
Bill-Dinner      John-Breakfast John-Lunch  Sally-Dinner
$ rm -f {John,Bill,Sally}-{Breakfast,Lunch,Dinner}
$ touch {a..f}{1..5}
$ ls
a1 a3 a5 b2 b4 c1 c3 c5 d2 d4 e1 e3 e5 f2 f4
a2 a4 b1 b3 b5 c2 c4 d1 d3 d5 e2 e4 f1 f3 f5
```

No primeiro exemplo, o uso de dois conjuntos de chaves significa que John, Bill e Sally têm nomes de arquivo associados a Breakfast, Lunch e Dinner. Se tivesse cometido um erro, eu poderia facilmente recuperar o comando e mudar touch para rm -f para excluir todos os arquivos. No exemplo a seguir, o uso de dois pontos entre as letras A e F e os números 1 e 5 especifica os intervalos a serem utilizados. Observe os arquivos que foram criados a partir desses poucos caracteres.

Listando arquivos e diretórios

O comando ls é o mais comumente usado para listar informações sobre arquivos e diretórios. Muitas opções disponíveis com o comando ls permitem reunir diferentes conjuntos de arquivos e diretórios, bem como ver diferentes tipos de informações sobre eles.

Por padrão, quando você digita o comando ls, o resultado exibe todos os arquivos e diretórios não ocultos contidos no diretório atual. Quando você digita ls, porém, muitos sistemas Linux (incluindo o Fedora e o RHEL) atribuem um alias ls para adicionar opções. Para ver se ls é alias, digite o seguinte:

```
$ alias ls
alias ls='ls --color=auto'
```

A opção --color=auto faz com que diferentes tipos de arquivos e diretórios sejam exibidos em cores diferentes. Assim, voltando ao diretório \$HOME/test criado no início do capítulo, adicione alguns tipos de arquivo diferentes e depois veja como eles aparecem com o comando ls.

```
$ cd $HOME/test
```

```

$ touch scriptx.sh apple
$ chmod 755 scriptx.sh
$ mkdir Stuff
$ ln -s apple pointer_to_apple
$ ls
apple pointer_to_apple scriptx.sh Stuff

```

Embora você não possa ver isso no exemplo de código anterior, o diretório `docs` aparece em azul, `pointer_to_apple` (um link simbólico) aparece em azul e `scriptx.sh` (que é um arquivo executável) aparece em verde. Todos os outros arquivos regulares aparecem em preto. Digitar `ls -l` para ver uma longa lista desses arquivos pode tornar esses diferentes tipos de arquivo ainda mais claros:

```

$ ls -l
total 4
-rw-rw-r--. 1 joe joe    0 Dec 18 13:38 apple
lrwxrwxrwx. 1 joe joe    5 Dec 18 13:46 pointer_to_apple -> apple
-rwxr-xr-x. 1 joe joe    0 Dec 18 13:37 scriptx.sh
drwxrwxr-x. 2 joe joe 4096 Dec 18 13:38 Stuff

```

Ao examinar a longa listagem mostrada, note que o primeiro caractere de cada linha exibe o tipo de arquivo. Um hífen (-) indica um arquivo regular, um `d` indica um diretório e um `l` (L minúsculo) indica um link simbólico. Um arquivo executável (um script ou arquivo binário que funciona como um comando) tem os bits de execução ativados (x). Veja mais sobre bits de execução na seção “Entendendo Permissões e Posse de Arquivos” mais adiante.

A seguir, você deve se familiarizar com o conteúdo do seu diretório inicial. Use as opções `-l` e `-a` para `ls`.

```

$ ls -la /home/joe
total 158
drwxrwxrwx 2  joe  sales   4096   May 12 13:55 .
drwxr-xr-x 3  root  root    4096   May 10 01:49 ..
-rw----- 1  joe  sales  2204   May 18 21:30 .bash_history
-rw-r--r-- 1  joe  sales    24   May 10 01:50 .bash_logout
-rw-r--r-- 1  joe  sales   230   May 10 01:50 .bash_profile
-rw-r--r-- 1  joe  sales  4096   May 10 01:50 .kde
-rw-rw-r-- 1  joe  sales 149872 May 11 22:49 letter

^      ^      ^      ^      ^      ^      ^
col 1  col 2  col 3  col 4  col 5      col 6      col 7

```

Exibindo uma lista completa (opção `-l`) do conteúdo de seu diretório inicial, é mostrado mais sobre o tamanho de arquivos e diretórios. A linha `total` exibe a quantidade total de espaço em disco utilizado pelos arquivos na lista (158 kilobytes neste exemplo). Diretórios como o diretório atual (`.`) e o diretório pai (`..`) — o diretório acima do diretório atual — são apontados como diretórios com a letra `d` no início de cada entrada. Cada diretório começa com um `d` e cada arquivo começa com um traço (-).

Os nomes de arquivos e diretórios são mostrados na coluna 7. Neste exemplo, um ponto (`.`) representa `/home/joe` e dois pontos (`..`) representam `/home` — o diretório pai de `/joe`. A maioria dos arquivos desse exemplo é de arquivos ponto (`.`), que são usados para armazenar propriedades da interface gráfica (diretório `.kde`) ou propriedades de shell (arquivos `.bash`). O único arquivo não

ponto nessa lista é um chamado `letter`. A coluna 3 exibe o proprietário do diretório ou arquivo. O diretório `/home` é possuído por `root` e tudo o mais é de posse do usuário `joe`, que pertence ao grupo `sales` (grupos são listados na coluna 4).

Além do `d` ou `-`, cada linha da coluna 1 contém as permissões definidas para esse arquivo ou diretório. Outras informações na listagem incluem o número de links não simbólicos para o item (coluna 2), o tamanho de cada arquivo em bytes (coluna 5), bem como a data e a hora em que cada arquivo foi mais recentemente modificado (coluna 6).

Eis alguns outros fatos sobre arquivos e listas de diretórios:

- O número de caracteres exibidos para um diretório (4096 bytes nesses exemplos) reflete o tamanho do arquivo que contém informações sobre o diretório. Embora esse número possa crescer acima de 4096 bytes para um diretório que contém bastantes arquivos, esse número não reflete o tamanho dos arquivos contidos nesse diretório.
- O formato da coluna de hora e data pode variar. Em vez de exibir “May 12”, a data pode ser apresentada como “2011/05/12”, dependendo da distribuição e da configuração de idioma (variável `LANG`).
- Ocasionalmente, em vez de ver o bit de execução (`x`) configurado em um arquivo executável, você pode ver um `s` no lugar. Com um `s` aparecendo dentro das permissões do proprietário (`-rwsr-xr-x`), ou do grupo (`-rwxr-sr-x`), ou ambas (`-rwsr-sr-x`), o aplicativo pode ser executado por qualquer usuário, mas a posse do processo em execução é atribuída ao usuário/grupo do aplicativo em vez do usuário que inicia o comando. Isso é referido como um programa `set UID` ou `set GID`, respectivamente. Por exemplo, o comando `mount` tem o conjunto de permissões `-rwsr-xr-x`. Isso permite que qualquer usuário execute `mount` para listar sistemas de arquivos montados (embora você ainda tenha de ser `root` para usar `mount` a fim de realmente montar sistemas de arquivos, na maioria dos casos).
- Se um `t` aparecer no final de um diretório, isso indica que o *sticky bit* está configurado para esse diretório (por exemplo, `drwxrwxr-t`). Ao configurar o *sticky bit* em um diretório, o proprietário desse diretório pode permitir que outros usuários e grupos adicionem arquivos a ele, mas impede que os usuários excluam os arquivos uns dos outros nesse diretório. Com um conjunto `GID` atribuído a um diretório, todos os arquivos criados nele são atribuídos ao mesmo grupo que o grupo desse diretório. (Se você vir um `S` maiúsculo ou `T` em vez de bits de execução em um diretório, isso significa que a permissão `set GID` ou *stick bit*, respectivamente, foi configurada, mas por alguma razão o bit de execução também não foi ativado.)
- Se você vir um sinal de mais no final dos bits de permissão (por exemplo, `-rw-rw-r--+`), isso significa que os atributos estendidos, como listas de controle de acesso (ACLs) ou SELinux, estão configuradas no arquivo.

Identificando diretórios

Quando precisar identificar seu diretório inicial em uma linha de comando do shell, você pode usar o seguinte:

- `$HOME` — Essa variável de ambiente armazena o nome do diretório inicial.
- `~` — O til (`~`) representa seu diretório inicial na linha de comando.

Você também pode usar o til para identificar o diretório inicial de outro usuário. Por exemplo, `~joe` seria expandido para o diretório inicial `joe` (provavelmente `/home/joe`). Então, se eu quisesse ir para o diretório `/home/joe/test` `cd ~joe/test`, eu poderia digitar `cd joe/test` para chegar lá.

Outras maneiras especiais de identificar diretórios no shell incluem o seguinte:

- `.` — Um único ponto (`.`) referencia o diretório atual.
- `..` — Dois pontos (`..`) referenciam um diretório diretamente acima do diretório atual.
- `$PWD` — Essa variável de ambiente referencia o diretório de trabalho atual.
- `$OLDPWD` — Essa variável de ambiente referencia o diretório de trabalho anterior antes de você mudar para o atual. (Digitar `cd -` retorna para o diretório representado por `$OLDPWD`).

Como mencionei anteriormente, existem muitas opções úteis para o comando `ls`. Volte para o diretório `$HOME/test` em que você esteve trabalhando. Eis alguns exemplos de opções de `ls`. Não se preocupe se o resultado não corresponder exatamente ao que está no seu diretório neste momento.

Qualquer arquivo ou diretório começando com um ponto (`.`) é considerado um arquivo oculto e não é exibido por padrão com `ls`. A opção `-a` permite que você veja os arquivos. A opção `-t` exibe arquivos na ordem em que eles foram mais recentemente modificados. Com a opção `-F`, uma barra (`/`) aparece no final do nome dos diretórios, um asterisco (`*`) é adicionado aos arquivos executáveis e um sinal de arroba (`@`) é mostrado ao lado de links simbólicos.

Para mostrar arquivos ocultos e não ocultos:

```
$ ls -a
.  apple  docs  grapefruit  pointer_to_apple  .stuff  watermelon
.. banana  grape  .hiddendir  script.sh        .tmpfile
```

Para listar todos os arquivos pela data/hora da última modificação:

```
$ ls -at
.tmpfile .hiddendir ..          docs          watermelon
banana  script.sh
.       .stuff    pointer_to_apple  grapefruit  apple      grape
```

Para listar arquivos e anexar indicadores do tipo arquivo:

```
$ ls -F
```

```
apple banana docs/ grape grapefruit pointer_to_apple@ script.sh*  
watermelon
```

Para evitar a exibição de determinados arquivos ou diretórios quando você usa `ls`, use a opção `--hide=`. No próximo conjunto de exemplos, qualquer arquivo começando com `g` não aparece na saída. A opção `-d` em um diretório exibe informações sobre o diretório em vez de mostrar os arquivos e diretórios que esse diretório contém. A opção `-R` lista todos os arquivos no diretório atual, bem como quaisquer arquivos ou diretórios que estão associados ao diretório original. A opção `-S` lista arquivos por tamanho.

Para não incluir todos os arquivos que começam com a letra `g` na lista:

```
$ ls --hide=g*
```

```
apple banana docs pointer_to_apple script.sh watermelon
```

Para listar informações sobre um diretório em vez dos arquivos que ele contém:

```
$ ls -ld $HOME/test/
```

```
drwxrwxr-x. 4 joe joe 4096 Dec 18 22:00 /home/joe/test/
```

Para criar múltiplas camadas de diretório (`-p` é necessário):

```
$ mkdir -p $HOME/test/documents/memos/
```

Para listar todos os arquivos e diretórios recursivamente a partir do diretório atual para baixo:

```
$ ls -R
```

```
...
```

Para listar os arquivos por tamanho:

```
$ ls -S
```

```
...
```

Entendendo Permissões e Posse de Arquivos

Depois de ter trabalhado com Linux por um tempo, é quase certo que você obterá uma mensagem `Permission denied`. Permissões associadas a arquivos e diretórios no Linux foram projetadas para impedir que os usuários acessem arquivos privados de outros usuários e para proteger arquivos importantes do sistema.

Os nove bits atribuídos para cada permissão de arquivo definem o acesso que você e outros têm ao seu

arquivo. Bits de permissão para um arquivo regular aparecem como `-rwxrwxrwx`. Esses bits são usados para definir quem pode ler, gravar ou executar o arquivo.

ta

um arquivo regular, um traço aparece na frente do indicador de permissões de nove bits. Em vez de um traço, pode ver um `d` (para diretório), `l` (para link simbólico), `b` (para dispositivo de bloco), `c` (para dispositivo de caracteres), `s` (para soquete), ou `p` (para pipe nomeado).

Das permissões de nove bits, os três primeiros bits se aplicam à permissão do proprietário, os três seguintes se aplicam ao grupo atribuído ao arquivo e os três últimos se aplicam a todos os outros. O `r` significa *permissão de leitura*, o `w` significa *permissão de gravação* e `x` significa *permissão de execução*. Se aparecer um traço em vez da letra, significa que a permissão está desativada para esse bit associado de leitura, gravação ou execução.

Como os arquivos e diretórios são diferentes tipos de elementos, permissões de leitura, gravação e execução sobre arquivos e diretórios significam coisas diferentes. A Tabela 4.2 explica o que você pode fazer com cada um deles.

TABELA 4.2 Configurando as Permissões de Leitura, Gravação e Execução

Permissão	Arquivo	Diretório
<code>l</code>	Visualiza o que está no arquivo.	Visualiza os arquivos e subpastas que ele contém.
<code>w</code>	Altera o conteúdo do arquivo, o renomeia ou o exclui.	Adiciona arquivos ou subdiretórios ao diretório. Remove arquivos ou diretórios do diretório.
<code>x</code>	Executa o arquivo como um programa.	Muda para o diretório como o diretório atual, pesquisa no diretório ou executa um programa a partir do diretório. Acessa metadados (tamanho do arquivo, registros de data e hora etc.) de arquivos no diretório.

Como observado anteriormente, você pode ver a permissão para qualquer arquivo ou diretório digitando o comando `ls -ld`. O arquivo ou diretório nomeado aparece como os mostrados neste exemplo:

```
$ ls -ld ch3 test
-rw-rw-r-- 1 joe sales 4983 Jan 18 22:13 ch3
drwxr-xr-x 2 joe sales 1024 Jan 24 13:47 test
```

A primeira linha exhibe que o arquivo `ch3` tem permissão de leitura e gravação para o proprietário e o grupo. Todos os outros usuários têm permissão de leitura, o que significa que eles podem visualizar o arquivo, mas não alterar ou remover seu conteúdo. A segunda linha exhibe o diretório `test` (indicado pela letra `d` antes dos bits de permissão). O proprietário tem permissões de leitura, gravação e execução, enquanto o grupo e os outros usuários têm apenas permissões de leitura e execução. Como resultado, o proprietário pode adicionar, alterar ou excluir arquivos no diretório e todos os outros só podem alterar

esse diretório ou ler e listar seu conteúdo. (Se não tivesse usado as opções `-d` para `ls`, você teria listado arquivos no diretório `test` em vez das permissões nesse diretório.)

Alterando permissões com `chmod` (números)

Se possuir um arquivo, você pode usar o comando `chmod` para alterar as permissões sobre ele como quiser. Um método de fazer isso, a cada permissão (leitura, gravação e execução) é atribuída um número — $r=4$, $w=2$ e $x=1$ — é usar o número total de cada conjunto para estabelecer a permissão. Por exemplo, para tornar as permissões abertas para você mesmo como proprietário, você deve configurar o primeiro número como 7 ($4+2+1$) e, então, você daria ao grupo e aos outros, permissão de somente leitura, configurando o segundo e o terceiro números como 4 ($4+0+0$), de modo que o número final é 744. Qualquer combinação de permissões pode resultar de 0 (sem permissão) a 7 (plena permissão).

Eis alguns exemplos de como alterar permissões sobre um arquivo (chamado `file`) e o que a permissão resultante seria:

O comando `chmod` a seguir resulta nesta permissão: `rw-rw-rwx`

```
# chmod 777 file
```

O comando `chmod` a seguir resulta nesta permissão: `rw-r-xr-x`

```
# chmod 755 file
```

O comando `chmod` a seguir resulta nesta permissão: `rw-r--r--`

```
# chmod 644 file rw-r--r-
```

O comando `chmod` a seguir resulta nesta permissão: `-----`

```
# chmod 000 file -----
```

O comando `chmod` também pode ser utilizado de forma recursiva. Por exemplo, digamos que você queira dar a uma estrutura de diretório inteira a permissão 755 (`rw-r-xr-x`), começando no diretório `$HOME/myapps`. Para fazer isso, você pode usar a opção `-R`, como segue:

```
$ chmod -R 755 $HOME/myapps
```

Todos os arquivos e diretórios abaixo, incluindo o diretório `myapps` em seu diretório inicial, terão o conjunto de permissões 755. Como a abordagem numérica para configurar permissões altera todos os bits de permissão de uma só vez, é mais comum o uso de letras para recursivamente alterar os bits de permissão de um conjunto grande de arquivos.

Alterando permissões com `chmod` (letras)

Você também pode ativar e desativar as permissões de arquivo usando os sinais de mais (+) e menos (-), respectivamente, junto com letras para indicar o que muda e para quem. Usando letras, para cada arquivo você pode mudar a permissão para o usuário (u), grupo (g), outros (o) e todos os usuários (a). O que você mudaria inclui os bits de leitura (r), gravação (w) e execução (x). Por exemplo, comece com um arquivo que tem todas as permissões abertas (rwxrwxrwx). Execute os seguintes comandos `chmod` usando as opções do sinal de menos. As permissões resultantes são mostradas à direita de cada comando:

O comando `chmod a` a seguir resulta nesta permissão: r-xr-xr-x

```
chmod a-w file
```

O comando `chmod a` a seguir resulta nesta permissão: rwxrwxrwx-

```
chmod o-x file
```

O comando `chmod a` a seguir resulta nesta permissão: rwx-----

```
chmod go-rwx file
```

Da mesma forma, os exemplos que se seguem começam com todas as permissões fechadas (-----). O sinal positivo é usado com `chmod` para ativar permissões:

O comando `chmod a` a seguir resulta nesta permissão: rw-----

```
$ chmod u+rw files
```

O comando `chmod a` a seguir resulta nesta permissão: --x--x--x

```
$ chmod a+x files
```

O comando `chmod a` a seguir resulta nesta permissão: r-xr-x---

```
$ chmod ug+rx files
```

Usar letras para alterar a permissão recursivamente com `chmod` geralmente funciona melhor do que usar números, porque você pode alterar os bits de forma seletiva, em vez de mudar todos os bits de permissão de uma vez. Por exemplo, digamos que você deseja remover a permissão de gravação para “outros” sem alterar nenhum outro bit de permissão em um conjunto de arquivos e diretórios. Você poderia fazer o seguinte:

```
$ chmod -R o-w $HOME/myapps
```

Este exemplo remove recursivamente as permissões de gravação para os “outros” em todos os arquivos e diretórios abaixo do diretório `myapps`. Se você tivesse usado números como 644, a permissão de execução seria desativada para diretórios; usando 755, a permissão de execução seria ativada para arquivos regulares. Usando `o-w`, apenas um bit é desativado e todos os outros bits são deixados intactos.

Configurando a permissão de arquivo padrão com umask

Quando você cria um arquivo como um usuário comum, este recebe a permissão `rw-rw-r--` por padrão. Um diretório recebe a permissão `rw-rwxr-x`. Para o usuário `root`, as permissões sobre arquivo e diretório são `rw-r--r--` e `rw-r-xr-x`, respectivamente. Esses valores padrão são determinados pelo valor de `umask`. Digite **umask** para ver qual é seu valor de `umask`. Por exemplo:

```
$ umask
```

```
0002
```

Se você ignorar o primeiro zero por um momento, o valor `umask` mascara o que é considerado como sendo permissões completamente abertas para um arquivo `666` ou um diretório `777`. O valor `umask` de `002` resulta na permissão `755` sobre um diretório (`rw-rwxr-x`). Esse mesmo `umask` resulta em uma permissão `644` sobre um arquivo (`rw-rw-r--`). (Permissões de execução estão desativadas por padrão para arquivos regulares.)

Para alterar temporariamente seu valor de `umask`, execute o comando `umask`. Agora, experimente criar alguns arquivos e diretórios para ver como o valor `umask` afeta o modo como as permissões são configuradas. Por exemplo:

```
$ umask 777 ; touch file01 ; mkdir dir01 ; ls -ld file01 dir01
d-----. 2 joe joe 4096 Dec 19 11:03 dir01
-----. 1 joe joe  0 Dec 19 11:02 file01
$ umask 000 ; touch file02 ; mkdir dir02 ; ls -ld file02 dir02
drwxrwxrwx. 2 joe joe 4096 Dec 19 11:00 dir01/
-rw-rw-rw-. 1 joe joe  0 Dec 19 10:59 file01
$ umask 022 ; touch file03 ; mkdir dir03 ; ls -ld file03 dir03
drwxr-xr-x. 2 joe joe 4096 Dec 19 11:07 dir03
-rw-r--r--. 1 joe joe  0 Dec 19 11:07 file03
```

Se quiser alterar permanentemente seu valor de `umask`, adicione um comando `umask` ao arquivo `.bashrc` em seu diretório inicial (perto do final desse arquivo). Da próxima vez que você abrir um shell, o `umask` será configurado com o valor que você escolheu.

Alterando a posse de arquivo

Como um usuário comum, você não pode alterar a posse de um arquivo ou diretório para que eles pertençam a outro usuário. Você *pode* alterar a posse como usuário `root`. Por exemplo, digamos que você criou um arquivo chamado `memo.txt` enquanto era o usuário `root`, no diretório inicial do usuário `joe`. Veja como você pode alterá-lo para ser possuído por `joe`:

```
# chown joe /home/joe/memo.txt
```

```
# ls -l /home/joe/memo.txt
```

```
-rw-r--r--. 1 joe root 0 Dec 19 11:23 /home/joe/memo.txt
```

Note que o comando `chown` alterou o usuário `joe`, mas deixou o grupo como `root`. Para alterar o usuário e o grupo para `joe`, você pode digitar o seguinte no lugar:

```
# chown joe:joe /home/joe/memo.txt
# ls -l /home/joe/memo.txt
-rw-r--r--. 1 joe joe 0 Dec 19 11:23 /home/joe/memo.txt
```

O comando `chown` também pode ser usado recursivamente. Usar a opção recursiva (`-R`) é útil se você precisar alterar uma estrutura de diretório inteira para ser possuída por um determinado usuário. Por exemplo, se você inseriu um pen drive, que é montado no diretório `/media/myusb`, e quiser dar a posse plena sobre o conteúdo dessa unidade para o usuário `joe`, você pode digitar o seguinte:

```
# chown -R joe:joe /media/myusb
```

Movendo, copiando e excluindo arquivos

Comandos para mover, copiar e excluir arquivos são relativamente simples. Para alterar a localização de um arquivo, use o comando `mv`. Para copiar um arquivo de um local para outro, use o comando `cp`. Para excluir um arquivo, use o comando `rm`. Esses comandos podem ser usados para agir sobre arquivos e diretórios individuais, ou recursivamente para agir sobre muitos arquivos e diretórios de uma só vez. Eis alguns exemplos:

```
$ mv abc def
$ mv abc ~
$ mv /home/joe/mymemos/ /home/joe/Documents/
```

O comando `mv` primeiro move o arquivo `abc` para o arquivo `def` no mesmo diretório (essencialmente o renomeando), enquanto o segundo move o arquivo `abc` para seu diretório inicial (`~`). O próximo comando move o diretório `mymemos` (e todo seu conteúdo) para o diretório `/home/joe/Documents`.

Por padrão, o comando `mv` sobrescreve todos os arquivos existentes, se houver arquivos com o mesmo nome no destino. Mas muitos sistemas Linux fornecem um alias ao comando `mv` para que ele use a opção `-i` (que faz o `mv` perguntar antes se você quer sobrescrever arquivos existentes). Eis como verificar se isso é verdade em seu sistema:

```
$ alias mv
alias mv='mv -i'
```

Eis alguns exemplos de como usar o comando `cp` para copiar arquivos de um local para outro:

```
$ cp abc def
$ cp abc ~
$ cp -r /usr/share/doc/bash-completion* /tmp/a/
```

```
$ cp -ra /usr/share/doc/bash-completion* /tmp/b/
```

O primeiro comando `copy` (`cp`) copia `abc` para o novo nome `def` no mesmo diretório, enquanto o segundo copia `abc` para seu diretório inicial (`~`), mantendo o nome `abc`. Os dois comandos de cópia recursiva (`-r`) copiam o diretório `bash-completion`, e todos os arquivos que ele contém, primeiro para os novos diretórios `/tmp/a/` e `/tmp/b/`. Se você executar `ls -l` nesses dois diretórios com a opção `archive` (`-a`), os registros de data/hora e as permissões são mantidos pela cópia, mas sem o `-a`, a data e a hora atuais são usadas e as permissões são determinadas por seu `umask`.

O comando `cp` normalmente também é usado na forma de um alias com a opção `-i`, para evitar sobrescrever os arquivos inadvertidamente.

Tal como acontece com os comandos `cp` e `mv`, `rm` também é geralmente usado com um alias para incluir a opção `-i`. Isso pode impedir o dano que possa vir de uma exclusão recursiva (`-r`) inadvertida. Eis alguns exemplos do comando `rm`:

```
$ rm abc
```

```
$ rm *
```

O primeiro comando removedor exclui o arquivo `abc`, o segundo remove todos os arquivos do diretório atual (exceto diretórios e/ou quaisquer arquivos que começam com um ponto). Se quiser excluir um diretório, você precisa usar a opção recursiva (`-r`) para `rm` ou, para um diretório vazio, você pode usar o comando `rmdir`. Considere os seguintes exemplos:

```
$ rmdir /home/joe/nothing/
```

```
$ rm -r /home/joe/bigdir/
```

```
$ rm -rf /home/joe/hugedir/
```

O comando `rmdir` no código anterior só remove o diretório (`nothing`) se ele estiver vazio. O exemplo `rm -r` remove o diretório `bigdir` e todo seu conteúdo (arquivos e vários níveis de subdiretórios), mas pedirá confirmação antes de cada um ser removido. Ao adicionar a opção de força (`-f`), o diretório `hugedir` e todo seu conteúdo são imediatamente removidos, sem avisar.

Atenção

Se você sobrescreve a opção `-i` nos comandos `mv`, `cp` e `rm`, você arrisca excluir alguns (ou muitos) dos arquivos por engano. Usar curingas (como `*`) e nenhum `-i` torna os erros ainda mais prováveis. Dito isso, há momentos em que você não quer se incomodar em percorrer cada arquivo que exclui. Você tem outras opções:

- Como observado com a opção `-f`, você pode forçar `rm` a excluir sem perguntar. Uma alternativa é executar `rm`, `cp` ou `mv` com uma barra invertida na frente dele (`\rm bigdir`). A barra invertida faz com que qualquer comando execute sem alias.

- Outra alternativa com `mv` é usar a opção `-b`. Com `-b`, se um arquivo de mesmo nome existir no destino,

Resumo

Comandos para se mover pelo sistema de arquivos, copiar, mover e excluir arquivos estão entre os mais básicos que você precisa para rabalhar a partir do shell. Este capítulo abrange muitos comandos para mover e manipular arquivos, bem como comandos para alterar a posse e as permissões sobre arquivos e diretórios.

O próximo capítulo descreve os comandos de edição e pesquisa de arquivos. Esses comandos incluem os editores de texto `vim/vi`, o comando `find` e o comando `grep`.

Exercícios

Use esses exercícios para testar seus conhecimentos sobre maneiras eficientes de se mover pelo sistema de arquivos e trabalhar com arquivos e diretórios. Quando possível, tente usar atalhos para digitar o mínimo possível a fim de obter os resultados desejados. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostrados no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. Crie um diretório no seu diretório inicial chamado `projects`. No diretório `projects`, crie nove arquivos vazios com os nomes `house1`, `house2`, `house3` e assim por diante até `house9`. Assumindo que existem muitos outros arquivos no diretório, sugira um único argumento para `ls` que liste apenas os nove arquivos.
2. Crie o caminho de diretório `$HOME/projects/houses/doors/`. Crie os seguintes arquivos vazios dentro desse caminho de diretório (tente usar caminhos absolutos e relativos em relação ao seu diretório inicial):

```
$HOME/projects/houses/bungalow.txt  
$HOME/projects/houses/doors/bifold.txt  
$HOME/projects/outdoors/vegetation/landscape.txt
```
3. Copie os arquivos `house1` e `house5` para o diretório `$HOME/projects/houses/`.
4. Copie recursivamente o diretório `/usr/share/doc/initscripts*` para o diretório `$HOME/projects/`. Mantenha os registros de data/hora e as permissões.
5. Liste recursivamente o conteúdo do diretório `$HOME/projects/`. Redirecione a saída para o comando `less` para que você possa percorrê-la página por página.
6. Remova os arquivos `house6`, `house7` e `house8` sem ser perguntado se realmente deseja excluí-los.

7. Mova house3 e house4 para o diretório \$HOME/projects/houses/doors .
8. Exclua o diretório \$HOME/projects/houses/doors e seu conteúdo.
9. Altere as permissões sobre o arquivo \$HOME/projects/house2 de modo que ele possa ser lido e gravado pelo usuário que possui o arquivo, apenas lido pelo grupo e que os outros não tenham permissões sobre ele.
10. Altere recursivamente as permissões sobre o diretório \$HOME/projects/ de modo que ninguém tenha permissão de gravação sobre qualquer arquivo ou diretório abaixo desse ponto no sistema de arquivos.

Trabalhando com arquivos de texto

NESTE CAPÍTULO

Usando `vim` e `vi` para editar arquivos de texto

Procurando arquivos

Pesquisando em arquivos

Quando o sistema UNIX, em que o Linux foi baseado, foi criado, quase todas as informações eram gerenciadas no sistema em arquivos de texto simples. Assim, era fundamental para os usuários saber como usar as ferramentas para pesquisar, dentro de arquivos de texto simples, e por eles, e ser capaz de alterar e configurar esses arquivos.

Hoje, a maior parte da configuração de sistemas Linux ainda é feita por edição de arquivos de texto simples. Mesmo quando uma ferramenta gráfica está disponível para trabalhar com um arquivo de configuração, ela não fornece uma maneira de fazer tudo o que você quer fazer nesse arquivo. Como resultado, você precisará usar um editor de texto para configurar o arquivo. Da mesma forma, alguns tipos de arquivos de documento, como HTML e XML, também são arquivos de texto simples que podem ser editados manualmente.

Antes que você possa se tornar um administrador de sistema pleno, precisa ser capaz de usar um editor de texto simples. O fato de que a maioria dos servidores Linux profissionais não tem sequer uma interface gráfica disponível torna necessária a edição manual dos arquivos de configuração de texto simples.

Uma vez que você sabe como editar arquivos de texto, ainda pode achar que é difícil descobrir onde os arquivos que você precisa editar estão localizados. Com comandos como `find`, é possível pesquisar por arquivos com base em vários atributos (nome do arquivo, tamanho, data de modificação e posse, para citar alguns). Com o comando `grep`, você pode pesquisar dentro de arquivos de texto para encontrar termos de pesquisa específicos.

Editando Arquivos com `vim` e `vi`

É quase impossível usar o Linux por qualquer período de tempo e não precisar de um editor de texto, porque, como observado anteriormente, a maioria dos arquivos de configuração do Linux são arquivos de

texto simples que você vai precisar quase com certeza alterar manualmente em algum ponto.

Se estiver usando um desktop, você pode executar `gedit` (selecione Applications ⇒ Accessories ⇒ `gedit` Text Editor), o qual é bastante intuitivo para edição de texto. Também há um editor de texto simples que você pode executar a partir do shell chamado `nano`. Mas a maioria dos usuários do shell do Linux usa o comando `vi` ou `emacs` para editar arquivos de texto.

A vantagem do `vi` ou do `emacs` sobre um editor gráfico é que você pode usar o comando a partir de qualquer shell, terminal de caractere, ou uma conexão baseado em caracteres através de uma rede (usando `telnet` ou `ssh`, por exemplo) — nenhuma interface gráfica é necessária. Eles também contêm uma enorme quantidade de recursos, de modo que você pode continuar a crescer com eles.

Esta seção fornece um breve tutorial sobre o editor de texto `vi`, que você pode usar para editar manualmente um arquivo de texto a partir de qualquer shell. Ela também descreve as versões melhoradas do `vi` chamadas `vim`. (Se o `vi` não lhe agrada, veja o quadro “Explorando outros editores de texto” para outras opções.)

O editor `vi` é difícil de aprender no início, mas depois que o conhece, você nunca precisa usar um mouse ou uma tecla de função — você pode editar e mover-se rápida e eficientemente dentro de arquivos usando apenas o teclado.

Explorando outros editores de texto

Dzias de editores de texto estão disponíveis para uso com Linux. Existem algumas alternativas que podem estar na sua distribuição Linux. Você pode experimentá-las se achar o `vi` muito tedioso. Eis algumas das opções:

- `nano` — Um editor de texto simplificado popular que é usado com muitos Linuxes inicializáveis e outros ambientes Linux de espaço limitado. Por exemplo, o `nano` está disponível para editar arquivos de texto durante o processo de instalação do Gentoo Linux.
- `gedit` — O editor de texto do GNOME que funciona na desktop.
- `jed` — Esse editor orientado para tela foi criado para programadores. Usando cores, o `jed` pode realçar o código que você cria para que possa ler facilmente o código e identificar erros de sintaxe. Use a tecla `Alt` a fim de selecionar os menus para manipular o texto.
- `joe` — O editor `joe` é semelhante a muitos editores de texto para PC. Use `Control` e as teclas de seta para se mover. Pressione `Ctrl+C` para sair sem salvar ou `Ctrl+X` para salvar e sair.
- `kate` — Um editor elegante que acompanha o pacote `kdebase`. Ele tem muitos penduricalhos, como destacar diferentes tipos de linguagens de programação e controles para lidar com quebras de linha.

- `kedit` — Um editor de texto baseado em uma interface gráfica que acompanha o desktop KDE.
- `mcedit` — Nesse editor, teclas de função ajudam você a se mover pelo arquivo e a salvar, copiar, mover e excluir texto. Assim como o `jed` e o `joe`, o `mcedit` é orientado para tela.
- `nedit` — Um excelente editor para programadores. Você precisa instalar o pacote `nedit` opcional para obtê-lo.

Para usar o `ssh` para fazer login em outros computadores Linux em sua rede, você pode usar qualquer editor de texto disponível para editar arquivos. Se você usar o `ssh -X` para se conectar ao sistema remoto, um editor baseado em GUI irá aparecer em sua tela local. Quando nenhuma interface gráfica estiver disponível, você precisará de um editor de texto que funcione no shell, como o `vi`, o `jed` ou o `joe`.

Iniciando com o vi

Na maioria das vezes, você inicia o `vi` para abrir um arquivo específico. Por exemplo, para abrir um arquivo chamado `/tmp/test`, digite o seguinte comando:

```
$ vi /tmp/test
```

Se esse for um novo arquivo, você deverá ver algo semelhante ao seguinte:

```
~  
~  
~  
~  
~  
~  
"/tmp/test" [New File]
```

Uma caixa piscando no topo representa o lugar onde está o cursor. A linha inferior mantém você informado sobre o que está acontecendo com sua edição (aqui, você acabou de abrir um novo arquivo). No meio, há tils (~) como enchimento porque ainda não há texto no arquivo. Agora, eis a parte intimidante: não há dicas, menus ou ícones para lhe dizer o que fazer. Para piorar as coisas, você não pode simplesmente começar a digitar. Se digitar, o computador possivelmente emitirá um bip. E algumas pessoas se queixam de que o Linux não é amigável.

Primeiro, você precisa saber os dois modos de operação principais: comando e entrada. O editor `vi` sempre inicia no modo de comando. Antes de poder adicionar ou alterar o texto no arquivo, você tem de digitar um comando (uma ou duas letras e um número opcional) para falar ao `vi` o que você quer fazer. A distinção entre letras maiúsculas e minúsculas é importante, portanto, use-as exatamente como mostrado

nos exemplos!

ta

Red Hat Enterprise Linux, no Fedora e em outras distribuições Linux, para usuários regulares, o comando `vi` é tratado como um alias para executar o `vim`. Se digitar `alias vi`, você deve ver `alias vi='vim'`. A primeira diferença óbvia entre `vi` e `vim` é que qualquer tipo de arquivo conhecido de texto, como HTML, código C ou um arquivo de configuração comum, aparecerá colorido. As cores indicarão a estrutura do arquivo. Outras características do `vim` que não estão no `vi` incluem recursos como destaque visual e modo de tela dividida. Por padrão, o usuário não tem um alias de `vi` para `vim`.

Adicionando texto

Para entrar no modo de entrada, digite um comando de entrada. Para começar, digite qualquer um dos seguintes comandos. Quando você terminar de introduzir texto, pressione a tecla Esc (às vezes, duas vezes) para voltar ao modo de comando. Lembre-se da tecla Esc!

- `a` — O comando *add*. Com esse comando, é possível inserir um texto que começa à *direita* do cursor.
- `A` — Adiciona ao final do comando. Com esse comando, é possível inserir texto a partir do fim da linha atual.
- `i` — O comando *insert*. Com esse comando, é possível inserir texto que começa à *esquerda* do cursor.
- `I` — O comando *insert at beginning*. Com esse comando, é possível inserir texto que começa no início da linha atual.
- `o` — O comando *open below*. Esse comando abre uma linha abaixo da linha atual e coloca você no modo de inserção.
- `O` — O comando *open above*. Esse comando abre uma linha acima da linha atual e coloca você no modo de inserção.

ca

Quando você está no modo de inserção, `-- INSERT --` aparece na parte inferior da tela.

Digite algumas palavras e depois pressione Enter. Repita isso algumas vezes até que você tenha algumas linhas de texto. Quando terminar de digitar, pressione a tecla Esc para voltar ao modo de comando. Agora que você tem um arquivo com algum texto nele, tente mover-se pelo texto com as teclas ou letras descritas na próxima seção.

ca

Pre-se da tecla Esc! Ela sempre posiciona você de volta ao modo de comando. Há ocasiões em que você tem de pressionar Esc duas vezes.

Movendo-se pelo texto

Para mover-se pelo texto, você pode usar as setas para cima, para baixo, para a direita e para a esquerda. Mas muitas das teclas para mover-se pelo texto estão imediatamente abaixo dos seus dedos quando estes estão em posição de digitação:

- **Teclas de seta** — Move o cursor para cima, para baixo, para a esquerda ou para a direita no arquivo um caractere por vez. Para mover-se para a esquerda e para a direita, você também pode usar a barra de espaço e Backspace, respectivamente. Se você preferir manter seus dedos no teclado, mova o cursor com h (para a esquerda), l (para a direita), j (para baixo), ou k (para cima).
- **w** — Move o cursor para o início da próxima palavra (delimitada por espaços, tabulações ou pontuação).
- **W** — Move o cursor para o início da próxima palavra (delimitada por espaços ou tabulações).
- **b** — Move o cursor para o início da palavra anterior (delimitada por espaços, tabulações ou pontuação).
- **B** — Move o cursor para o início da palavra anterior (delimitada por espaços ou tabulações).
- **0** (zero) — Move o cursor para o início da linha atual.
- **\$** — Move o cursor para o final da linha atual.
- **H** — Move o cursor para o canto superior esquerdo da tela (primeira linha na tela).
- **M** — Move o cursor para o primeiro caractere da linha central na tela.
- **L** — Move o cursor para o canto esquerdo inferior da tela (última linha na tela).

Excluindo, copiando e alterando texto

A única outra edição que você precisa saber é como excluir, copiar ou alterar o texto. Os comandos x, d, y e c podem ser usados para excluir e alterar texto. Todos eles podem ser usados junto com teclas de movimento (setas, PgUp, PgDn, letras e teclas especiais) e números para indicar exatamente o que você está excluindo, copiando ou alterando. Considere os seguintes exemplos:

- **x** — Exclui o caractere sob o cursor.
- **X** — Exclui o caractere imediatamente antes do cursor.
- **d <?>** — Exclui algum texto.
- **c <?>** — Altera algum texto.
- **y <?>** — Copia (*yanks*) algum texto.

O <?> depois de cada letra na lista anterior identifica o lugar onde você pode utilizar um comando de movimento para escolher o que você está excluindo, alterando ou copiando. Por exemplo:

- `dw` — Exclui (d) uma palavra (w) após a posição atual do cursor.
- `db` — Exclui (d) uma palavra (b) antes da posição atual do cursor.
- `dd` — Exclui (d) toda a linha atual (d).
- `c$` — Altera (c) os caracteres a partir do caractere atual até o final da linha atual (\$) e entra no modo de entrada.
- `c0` — Altera (c) a partir do caractere anterior até o início da linha atual (0) e entra no modo de entrada.
- `c1` — Apaga (c) a letra atual (1) e entra no modo de entrada.
- `cc` — Apaga (c) a linha (c) e entra no modo de entrada.
- `yy` — Copia (y) a linha atual (y) para o buffer.
- `y)` — Copia (y) a sentença atual () , à direita do cursor, para o buffer.
- `y}` — Copia (y) o parágrafo atual (}), à direita do cursor, para o buffer.

Qualquer um dos comandos mostrados somente pode ser modificado usando números, como você pode ver nos exemplos a seguir:

- `3dd` — Exclui (d) três linhas (3d), começando na linha atual.
- `3dw` — Apaga (d) as próximas três palavras (3w).
- `5c1` — Altera (c) as próximas cinco (5) letras (1) (isto é, remove as letras e entra no modo de entrada).
- `12j` — Move-se para baixo (j) 12 linhas (12).
- `5cw` — Apaga (c) as próximas cinco (5) palavras (w) e entra no modo de entrada.
- `4y)` — Copia (y) as próximas quatro (4) sentenças ()).

Colando texto

Depois que o texto foi copiado para o buffer (apagando, alterando ou copiando-o), você pode colocá-lo de volta em seu arquivo usando a letra `p` ou `P`. Com ambos os comandos, o texto mais recentemente armazenado no buffer é copiado para o arquivo de diferentes maneiras.

- `P` — Coloca o texto copiado à esquerda do cursor se ele contiver letras ou palavras; coloca o texto copiado acima da linha atual se ele contiver linhas de texto.
- `p` — Coloca o texto em buffer à direita do cursor se ele contiver letras ou palavras; coloca o texto em buffer abaixo da linha atual se ele contiver linhas de texto.

Repetindo comandos

Depois de excluir, alterar ou colar texto, você pode repetir essa ação, digitando um ponto (.). Por exemplo, com o cursor no início do nome `Joe`, você digita `cw` e digita `Jim` para alterar `Joe` para `Jim`. Você pesquisa a próxima ocorrência de `Joe` no arquivo, posiciona o cursor no início desse nome e pressiona um ponto. A palavra muda para `Jim` e você pode pesquisar a próxima ocorrência.

Saindo do vi

Para concluir esse assunto, use os seguintes comandos para salvar ou fechar o arquivo:

- `ZZ` — Salva as mudanças atuais no arquivo e sai do `vi`.
- `:w` — Salve o arquivo atual, mas continua no modo de edição.
- `:wq` — O mesmo que `ZZ`.
- `:q` — Sai do arquivo atual. Isso só funciona se você não tiver nenhuma alteração não salva.
- `:q!` — Sai do arquivo atual e *não* salva as alterações feitas no arquivo.

ca

Se você realmente estragou o arquivo por engano, o comando `:q!` é a melhor maneira de sair e abandonar as suas alterações. O arquivo reverte para a versão mais recentemente alterada. Portanto, se você acabou de salvar com `:w`, está amarrado às mudanças até esse ponto. Se quiser apenas desfazer algumas edições, digite `u` para desfazer uma das alterações.

Você aprendeu alguns comandos de edição do `vi`. Descrevo mais comandos nas próximas seções. Primeiro, porém, considere as seguintes dicas para facilitar seus primeiros testes com o `vi`:

- **ESC** — Lembre-se de que ESC leva-o de volta ao modo de comandos. (Já vi pessoas pressionarem cada tecla no teclado tentando sair de um arquivo.) ESC seguido por `ZZ` faz com que você saia do modo de comandos, salve o arquivo e encerre.
- **u** — Pressione `u` para desfazer a última alteração que você fez. Continue pressionando `u` para desfazer cada sucessiva alteração anterior.
- **Ctrl+R** — Se decidir que não queria desfazer o comando de desfazer anterior (Undo), use `Ctrl+R` para refazer (Redo). Essencialmente, esse comando desfaz o que você desfez.
- **Caps Lock** — Cuidado com a possibilidade de pressionar Caps Lock por engano. Tudo que você digita no `vi` tem um significado diferente quando as letras estão em maiúsculas. Você não receberá um aviso de que está digitando em maiúsculas — as coisas apenas começam a parecer estranhas.
- **!:comando** — Você pode executar um comando enquanto estiver no `vi` usando `!` seguido por um nome de comando de shell. Por exemplo, digite `!date` para ver a data e hora atuais, digite

: !pwd para ver qual é o diretório atual, digite : !jobs para ver se você tem algum trabalho sendo executado em segundo plano. Quando o comando termina, pressione Enter e você volta a editar o arquivo. Você pode até mesmo usar essa técnica para iniciar um shell (: !bash) a partir do vi, executar alguns comandos de shell e, então, digitar exit para retornar ao vi. (Recomendo fazer um salvamento antes de sair para o shell, apenas para o caso de você esquecer-se de voltar ao vi.)

- **Ctrl+G** — Se você esquecer o que está editando, pressionar essas teclas exibe o nome do arquivo que está sendo editado e a linha atual em que você está na parte inferior da tela. Também exibe o número total de linhas no arquivo, a porcentagem do ponto em que você está no arquivo e o número da coluna em que o cursor está. Isso só o ajuda a se orientar depois que você parou para tomar uma xícara de café às três da madrugada.

Outras maneiras de se mover por um arquivo

Além dos poucos comandos para se mover por um arquivo descritos anteriormente, há outras maneiras de fazer isso em um arquivo vi. Para experimentá-las, abra um arquivo grande de teste. (Experimente copiar /var/log/messages para /tmp e abri-lo no vi. Eis alguns comandos de movimento que você pode utilizar:

- **Ctrl+f** — Pagina para frente, uma página por vez.
- **Ctrl+b** — Pagina para trás, uma página por vez.
- **Ctrl+d** — Desce meia página por vez.
- **Ctrl+u** — Sobe meia página por vez.
- **G** — Vai para a última linha do arquivo.
- **1G** — Vai para a primeira linha do arquivo. (Utilize qualquer número para ir para essa linha no arquivo.)
- **35G** — Vá para qualquer número da linha (35, neste caso).

Procurando texto

Para pesquisar a ocorrência seguinte ou anterior do texto no arquivo, utilize o caractere de barra (/) ou o ponto de interrogação (?). Depois da barra ou do ponto de interrogação, coloque um padrão (string de texto) para pesquisar esse padrão para frente ou para trás, respectivamente. Dentro da pesquisa, você também pode utilizar metacaracteres. Eis alguns exemplos:

- /hello — Pesquisa para frente a palavra hello.
- ?goodbye — Pesquisa para trás a palavra goodbye.
- /The.*foot — Pesquisa para frente uma linha que tem a palavra The nela e também, depois de

algum ponto, a palavra `foot`.

- `?[pP]rint` — Pesquisa para trás `print` ou `Print`. Lembre-se de que a distinção entre maiúsculas e minúsculas importa em Linux; portanto, utilize colchetes para pesquisar palavras que poderiam ter diferentes usos de maiúsculas ou minúsculas.

Depois de ter introduzido um termo de pesquisa, basta digitar `n` ou `N` para pesquisar para frente ou para trás o mesmo termo de novo, respectivamente.

Usando o modo `ex`

O editor `vi` foi originalmente baseado no editor `ex`, que não permite que você trabalhe no modo de tela cheia. Entretanto, permitia que você executasse comandos para localizar e alterar texto em uma ou mais linhas por vez. Quando você digita um caractere de dois-pontos e o cursor vai para o fundo da tela, você está essencialmente no modo `ex`. Eis alguns exemplos desses comandos `ex` para pesquisar e alterar texto. (Escolhi as palavras `Local` e `Remote` para pesquisar, mas você pode usar quaisquer palavras apropriadas.)

- `:g/Local` — Pesquisa a palavra `Local` e imprime todas as ocorrências dessa linha no arquivo. (Se houver mais de uma tela de resultados, a saída é redirecionada para o comando `more`.)
- `:s/Local/Remote` — Substitui `Remote` pela palavra `Local` na linha atual.
- `:g/Local/s//Remote` — Substitui a primeira ocorrência da palavra `Local` em cada linha do arquivo pela palavra `Remote`.
- `:g/Local/s//Remote/g` — Substitui todas as ocorrências da palavra `Local` pela palavra `Remote` em todo o arquivo.
- `:g/Local/s//Remote/gp` — Substitui todas as ocorrências da palavra `Local` pela palavra `Remote` em todo o arquivo e depois imprime cada linha de modo que você possa ver as mudanças (redirecionando-as através de `less` se a saída ocupar mais de uma página).

Aprendendo mais sobre o `vi` e o `vim`

Para saber mais sobre o editor `vi`, experimente digitar `vimtutor`. O comando `vimtutor` abre um tutorial no editor `vim` que guia você pelos comandos e recursos comuns que podem ser usados no `vim`.

Localizando arquivos

Até mesmo uma instalação básica do Linux pode ter milhares de arquivos instalados nela. Para ajudá-lo a encontrar arquivos em seu sistema, há comandos como `locate` (para encontrar comandos pelo nome), `find` (para encontrar arquivos com base em vários atributos diferentes) e `grep` (para pesquisar arquivos de texto a fim de encontrar linhas que contêm o texto de pesquisa).

Usando locate para localizar arquivos por nome

Na maioria dos sistemas Linux (Fedora e RHEL incluídos), o comando `updatedb` é executado uma vez por dia para coletar os nomes de arquivos em todo seu sistema Linux em um banco de dados. Ao executar o comando `locate`, você pode pesquisar o banco de dados para encontrar a localização de arquivos armazenados nele.

Eis algumas coisas que você deve saber sobre a pesquisa de arquivos usando o comando `locate`

- Há vantagens e desvantagens em usar `locate` para localizar nomes de arquivos em vez de `find`. Um comando `locate` localiza arquivos mais rápido, porque ele pesquisa em um banco de dados, em vez de ter de pesquisar o sistema de arquivos inteiro. A desvantagem é que o comando `locate` não pode encontrar todos os arquivos adicionados ao sistema desde a última vez que o banco de dados foi atualizado.
- Nem todo arquivo em seu sistema de arquivos está armazenado no banco de dados. O conteúdo do arquivo `/etc/updatedb.conf` limita os nomes que são coletados, podendo tipos de montagem, tipos de sistemas de arquivos, tipos de arquivo e pontos de montagem selecionados. Por exemplo, nomes de arquivo não são coletados dos sistemas de arquivos remotamente montados (`cifs`, `nfs` etc.) ou CDs ou DVDs montados localmente (`iso9660`). Caminhos que contêm arquivos temporários (`/tmp`) e arquivos de spool (`/var/spool/cups`) também são podados. Você pode adicionar itens para podar (ou remover alguns itens que você não quer que sejam podados) o banco de dados `locate` de acordo com suas necessidades. No RHEL, o arquivo `updatedb.conf` contém o seguinte:

```
PRUNE_BIND_MOUNTS = "yes"
PRUNEFS = "9p afs anon_inodefs auto autofs bdev binfmt_misc
cgroup cifs coda configfs cpuset debugfs devpts encryptfs exofs
fuse fusectl gfs gfs2 hugetlbfs inotifyfs iso9660 jffs2 lustre
mqueue ncpfs nfs nfs4 nfsd pipefs proc ramfs rootfs rpc_pipefs
securityfs selinuxfs sfs sockfs sysfs tmpfs ubifs udf usbfs"
PRUNENAMES = ".git .hg .svn"
PRUNEPATHS = "/afs /media /net /sfs /tmp /udev /var/cache/ccache
/var/spool/cups /var/spool/squid /var/tmp"
```

- Como um usuário comum, você não será capaz de ver todos os arquivos do banco de dados `locate` que você não podia ver no sistema de arquivos normalmente. Por exemplo, se você não puder digitar `ls` para visualizar arquivos no diretório `/root`, não será capaz de localizar os arquivos armazenados no diretório.
- Quando você procura por uma string, ela pode aparecer em qualquer lugar no caminho de um arquivo. Por exemplo, se pesquisar `passwd`, você pode descobrir `/etc/passwd`, `/usr/bin/passwd`, `/home/chris/passwd/pwdfiles.txt` e muitos outros arquivos com `passwd` no caminho.
- Se adicionar arquivos ao seu sistema depois de executar `updatedb`, você não será capaz de

localizá-los até que `updatedb` execute novamente (provavelmente nessa noite). Para fazer com que o banco de dados contenha todos os arquivos até o momento atual, você pode simplesmente executar `updatedb` a partir do shell como root.

Eis alguns exemplos de como usar o comando `locate` para pesquisar arquivos:

```
$ locate .bashrc
/etc/skel/.bashrc
/home/cnegus/.bashrc
# locate .bashrc
/etc/skel/.bashrc
/home/bill/.bashrc
/home/joe/.bashrc
/root/.bashrc
```

Quando executado como um usuário comum, `locate` só localiza `.bashrc` em `/etc/skel` e diretório inicial do usuário. Executado como root, o mesmo comando localiza arquivos `.bashrc` no diretório inicial de todo mundo.

```
$ locate muttrc
/usr/share/doc/mutt-1.5.20/sample.muttrc
...
$ locate -i muttrc
/etc/Muttrc
/etc/Muttrc.local
/usr/share/doc/mutt-1.5.20/sample.muttrc
...
```

Usando `locate -i`, nomes de arquivos são encontrados, independentemente do uso de maiúsculas ou minúsculas. No exemplo anterior, `Muttrc` e `Muttrc.local` foram encontrados com `-i` ao passo que não foram encontrados sem essa opção.

```
$ locate services
/etc/services
/usr/share/services/bmp.kmgio
/usr/share/services/data.kmgio
```

Ao contrário do comando `find`, que usa a opção `-name` para localizar nomes de arquivo, o comando

`locate` localiza a string que você insere se ela existir em qualquer parte do caminho do arquivo. Por exemplo, se pesquisar “services” usando o comando `locate`, você encontra os arquivos e diretórios que contêm a string de texto “services”.

Procurando arquivos com `find`

O comando `find` é o melhor comando para pesquisar arquivos no sistema de arquivos, com base em uma variedade de atributos. Depois de localizar os arquivos, você também pode agir sobre eles (usando as opções `-exec` ou `-okay`) executando todos os comandos que quiser sobre os arquivos.

Quando você executa `find`, ele procura seu sistema de arquivos “ao vivo”, o que faz com que se torne mais lento do que `locate`, mas lhe dá uma visualização atualizada dos arquivos em seu sistema Linux. Mas você também pode dizer para `find` começar em um determinado ponto no sistema de arquivos, assim, limitando a área do sistema de arquivos que está sendo pesquisada, a pesquisa pode ir mais rápido.

Quase todo o atributo de arquivo existente pode ser usado como uma opção de pesquisa. É possível pesquisar por nomes de arquivos, posse, permissão, tamanho, data/hora da última modificação e outros atributos. Você ainda pode usar combinações de atributos. Eis alguns exemplos básicos do uso do comando `find`:

```
$ find
$ find /etc
# find /etc
$ find $HOME -ls
```

Executado em uma linha sozinha, o comando `find` encontra todos os arquivos e diretórios abaixo do diretório atual. Se você deseja pesquisar a partir de um determinado ponto na árvore de diretórios, basta adicionar o nome do diretório que você deseja pesquisar (como `/etc`). Como um usuário comum, `find` não lhe dá permissão especial para encontrar arquivos que sejam legíveis somente pelo usuário `root`. Assim, `find` irá produzir um monte de mensagens de erro. Executado como usuário `root`, `find /etc` localizará todos os arquivos sob `/etc`.

Uma opção especial para comando `find` é `-ls`. Uma longa lista (permissão, posse, tamanho etc.) é impressa com cada arquivo quando você adiciona `-ls` ao comando `find`. Essa opção irá ajudá-lo em exemplos posteriores quando você quiser verificar se encontrou arquivos que contêm a posse, tamanho, data/hora da última modificação, ou outros atributos que você está tentando encontrar.

ta

como um usuário comum, você pesquisar uma área do sistema de arquivos sobre a qual você não tem plena permissão para acessar todos os arquivos nele contidos (como o diretório `/etc`), você pode receber um monte de mensagens de erro quando pesquisar com `find`. Para se livrar dessas mensagens, direcione erros padrão para `/dev/null`. Para fazer isso, adicione o seguinte ao final da linha de comando: `2> /dev/null. 0 2>`

Localizando arquivos por nome

Para localizar arquivos por nome, você pode usar as opções `-name` e `-iname`. A pesquisa é feita pelo nome de base do arquivo, os nomes de diretório não são pesquisados por padrão. Para tornar a pesquisa mais flexível, você pode usar caracteres de correspondência de arquivo, tais como asteriscos (*) e pontos de interrogação (?), como nos exemplos a seguir:

```
# find /etc -name passwd
/etc/pam.d/passwd
/etc/passwd
# find /etc -iname '*passwd*'
/etc/pam.d/passwd
/etc/passwd-
/etc/passwd.OLD
/etc/passwd
/etc/MYPASSWD
/etc/security/opasswd
```

Usando a opção `-name` e nenhum asterisco, o primeiro exemplo acima lista todos os arquivos no diretório `/etc` que têm exatamente o nome `passwd`. Usando `-iname` em vez disso, você pode localizar o arquivo com qualquer combinação de letras maiúsculas e minúsculas. Usando asteriscos, você pode localizar qualquer nome de arquivo que inclui a palavra `passwd`.

Localizando arquivos por tamanho

Se seu disco está cheio e você quer saber onde seus maiores arquivos estão, pode pesquisar seu sistema pelo tamanho do arquivo. A opção `-size` permite pesquisar por arquivos que têm exatamente o tamanho selecionado ou que são menores que ou maiores do que o tamanho selecionado, como você pode ver nos exemplos a seguir:

```
$ find /usr/share/ -size +10M
$ find /mostlybig -size -1M
$ find /bigdata -size +500M -size -5G -exec du -sh {} \;
4.1G /bigdata/images/rhel6.img
606M /NotBackedUp/Fedora-16-i686-Live-Desktop.iso
560M /NotBackedUp/dance2.avi
```

O primeiro exemplo no código anterior encontra arquivos maiores que 10 MB. O segundo encontra os

arquivos com menos de 1 MB. No terceiro exemplo, estou procurando imagens ISO e arquivos de vídeo que estão entre 500 MB e 5 GB. Isso inclui um exemplo da opção `-exec` (que descrevo mais adiante) para executar o comando `du` em cada arquivo, a fim de ver seu tamanho.

Localizando arquivos por usuário

Você pode pesquisar por um proprietário particular (`-user`) ou grupo (`-group`) ao tentar localizar arquivos. Usando `-not` e `-or` você pode refinar sua pesquisa por arquivos associados a usuários e grupos específicos, como pode ver nos exemplos a seguir:

```
$ find /home -user chris -ls
131077  4 -rw-r--r-- 1 cnegus  cnegus 379 Jun 29 2010 ./bashrc
$ find /home -user chris -or -user joe -ls
131077  4 -rw-r--r-- 1 cnegus  cnegus 379 Jun 29 2010 ./bashrc
181022  4 -rw-r--r-- 1 joe      joe      379 Jun 15 2010 ./bashrc
$ find /etc -group ntp -ls
131438  4 drwxrwxr-x 3 root    ntp     4096 Mar 9 22:16 /etc/ntp
$ find /var/spool -not -user root -ls
262100  0 -rw-rw---- 1 rpc      mail    0 Jan 27 2011 /var/spool/mail/rpc
278504  0 -rw-rw---- 1 joe      mail    0 Apr 3  2011 /var/spool/mail/joe
261230  0 -rw-rw---- 1 13599   mail    0 Dec 18 14:17 /var/spool/mail/bill
277373 2848 -rw-rw---- 1 chris   mail   8284 Mar 15  2011 /var/spool/mail/
chris
```

O primeiro exemplo gera uma listagem de todos os arquivos no diretório `/home` que pertencem ao usuário `chris`. O exemplo seguinte lista arquivos pertencentes a `chris` ou `joe`. O comando `find` de `/etc` revela todos os arquivos que têm `ntp` como sua atribuição primária de grupo. O último exemplo mostra todos os arquivos em `/var/spool` que não pertencem a `root`. Você pode ver os arquivos pertencentes a outros usuários na saída do exemplo.

Localizando arquivos por permissão

Localizar arquivos com permissão é uma excelente maneira de expor as questões de segurança em seu sistema ou descobrir problemas de acesso. Assim como você mudou as permissões em arquivos usando números ou letras (com o comando `chmod`), da mesma forma é possível localizar arquivos com base no número ou na letra das permissões, juntamente com as opções `-perm`. (Consulte o Capítulo 4, “Movendo-se pelo sistema de arquivos”, para ver como usar números e letras com `chmod` a fim de refletir as permissões de arquivo.)

Se você usa números para a permissão, como eu faço abaixo, lembre-se de que os três números representam as permissões para o usuário, o grupo e os outros. Cada um desses três números varia de nenhuma permissão (0) à permissão plena de leitura/gravação/execução (7), pela soma dos bits de leitura (4), gravação (2) e execução (1). Com um traço (-) na frente do número, todos os três bits indicados devem ser localizados; com um sinal positivo (+) na frente do mesmo, qualquer um dos números podem corresponder à pesquisa para localizar um arquivo. Os números completos e exatos devem corresponder se nem um hífen nem um sinal de mais for usado.

Considere os seguintes exemplos:

```
$ find /bin -perm 755 -ls
788884 28 -rwxr-xr-x 1 root    root      28176 Mar 10 2011 /bin/echo
$ find /home/chris/ -perm -222 -type d -ls
144503  4 drwxrwxrwx 8 cnegus  cnegus 4096 Jun 23 2011 /home/chris/OPENDIR
```

Pesquisando `-perm 755`, quaisquer arquivos ou diretórios com exatamente permissão `rwxr-xr-x` são localizados. Usando `-perm -222`, somente os arquivos que têm permissão de gravação para o usuário, o grupo e outros são localizados. Note que, nesse caso, a opção `-type d` é adicionada para localizar apenas diretórios.

```
$ find /myreadonly -perm +222 -type f
685035 0 -rw-rw-r-- 1 cnegus cnegus 0 Dec 30 16:34 /tmp/write/abc

$ find . -perm -002 -type f -ls
266230 0 -rw-rw-rw- 1 cnegus cnegus 0 Dec 30 16:28 ./LINUX_BIBLE/abc
```

Usando `-perm +222`, você pode localizar qualquer arquivo (`-type f`) que tenha a permissão de gravação ativada para o usuário, o grupo ou outros. Você pode fazer isso para ter certeza de que todos os arquivos são somente leitura em uma determinada parte do sistema de arquivos (no caso, sob o diretório `/myreadonly`). O último exemplo, `-perm +002`, é muito útil para localizar arquivos que têm permissão de gravação aberta para “outros”, independentemente da forma como os outros bits de permissão estão configurados.

Localizando arquivos por data e hora

Data e hora são armazenadas para cada arquivo quando ele é criado, acessado, quando seu conteúdo é modificado, ou quando seus metadados são alterados. Os metadados incluem o proprietário, o grupo, um registro de data/hora, o tamanho do arquivo, permissões e outras informações armazenadas no inode do arquivo. Você pode querer pesquisar alterações nos dados ou nos metadados dos arquivos por qualquer um dos seguintes motivos:

- Você simplesmente alterou o conteúdo de um arquivo de configuração e não se lembra de qual. Então você pesquisa em `/etc` para ver o que mudou nos últimos 10 minutos:

```
$ find /etc/ -mmin -10
```
- Você suspeita que alguém invadiu seu sistema três dias atrás. Então você pesquisa o sistema para ver se todos os comandos tiveram seu proprietário ou suas permissões alteradas nos últimos três dias:

```
$ find /bin /usr/bin /sbin /usr/sbin -ctime -3
```
- Você quer localizar arquivos no servidor FTP (`/var/ftp`) e no servidor web (`/var/www`) que não são acessados há mais de 300 dias, e assim ver se algo precisa ser excluído:

```
$ find /var/ftp /var/www -atime +300
```

Como pode constatar a partir dos exemplos, você pode pesquisar por alterações no conteúdo ou nos metadados ao longo de um determinado número de dias ou minutos. As opções de data/hora (`-atime`, `-ctime`, e `-mtime`) permitem pesquisar com base no número de dias desde quando cada arquivo foi acessado, alterado ou teve seus metadados modificados. As opções (`-amin`, `-cmin` e `-mmin`) fazem o mesmo em minutos.

Os números que você fornece como argumentos para as opções `min` e `time` são precedidos por um hífen (para indicar um tempo desde o momento atual até o número de minutos ou dias atrás) ou um sinal de mais (para indicar o tempo a partir do número de minutos ou dias atrás e mais antigos). Sem o hífen e o

sinal de mais, o número exato é procurado.

Usando *not* e *or* ao localizar arquivos

Com as opções `-not` e `-or`, é possível refinar ainda mais suas pesquisas. Talvez você precise localizar arquivos pertencentes a um determinado usuário, mas não atribuídos a um determinado grupo. Você pode querer arquivos maiores do que um determinado tamanho, mas menores do que outro tamanho. Ou você pode querer localizar arquivos pertencentes a qualquer um dos vários usuários. As opções `-not` e `-or` podem ajudá-lo a fazer isso. Considere os seguintes exemplos:

- Há um diretório compartilhado chamado `/var/allusers`. Essa linha de comando permite localizar arquivos pertencentes a `joe` ou `chris`.

```
$ find /var/allusers \( -user joe -o -user chris \) -ls
679967 0 -rw-r--r-- 1 chris chris 0 Dec 31 12:57
/var/allusers/myjoe
679977 1812 -rw-r--r-- 1 joe joe 4379 Dec 31 13:09
/var/allusers/dict.dat
679972 0 -rw-r--r-- 1 joe sales 0 Dec 31 13:02
/var/allusers/one
```

- Essa linha de comando procura arquivos pertencentes ao usuário `joe`, mas apenas aqueles que não estão atribuídos ao grupo `joe`:

```
$ find /var/allusers/ -user joe -not -group joe -ls
679972 0 -rw-r--r-- 1 joe sales 0 Dec 31 13:02 /var/allusers/one
```

- Você também pode adicionar múltiplas condições sobre suas pesquisas. Aqui, um arquivo deve pertencer a `joe` e também ter mais do que 1 MB de tamanho:

```
$ find /var/allusers/ -user joe -and -size +1M -ls
679977 1812 -rw-r--r-- 1 joe root 1854379 Dec 31 13:09
/var/allusers/dict.dat
```

Localizando arquivos e executando comandos

Um dos recursos mais poderosos do comando `find` é a capacidade de executar comandos sobre qualquer arquivo que você localiza. Com a opção `-exec`, o comando que você usa é executado sobre cada arquivo encontrado, sem parar para perguntar se está tudo bem. A opção `-ok` vai parar em cada arquivo localizado e perguntar se você deseja executar o comando sobre ele ou não.

A vantagem de `-ok` é que, se você estiver fazendo algo destrutivo, pode ter certeza de que irá confirmar a execução do comando sobre cada arquivo individualmente. A sintaxe para `-exec` e `-ok` é a mesma:

```
$ find [options] -exec command {} \;
```

Com `-exec` ou `-ok`, você executa `find` com qualquer opção que quiser para localizar os arquivos que lhe interessam. Então, digite a opção `-exec` ou `-ok`, seguida do comando que você deseja executar em cada arquivo. O conjunto de chaves indica onde ler na linha de comando em cada arquivo localizado. Cada arquivo pode ser incluído na linha de comando várias vezes, se você quiser. Para terminar a linha,

é preciso adicionar uma barra invertida e um ponto e vírgula (`\;`). Eis alguns exemplos:

- Esse comando procura qualquer arquivo chamado `iptables` sob o diretório `/etc` e inclui o nome na saída de um comando `echo`:

```
$ find /etc -iname iptables -exec echo "I found {}" \;  
I found /etc/bash_completion.d/iptables  
I found /etc/sysconfig/iptables  
I found /etc/rc.d/init.d/iptables
```

- Esse comando localiza todos os arquivos sob o diretório `/usr/share` que tem mais do que 5MB de tamanho. Em seguida, ele lista o tamanho de cada arquivo com o comando `du`. A saída é de `find`, então, ordenada por tamanho, do maior para o menor. Inserindo `-exec`, todas as entradas encontradas são processadas, sem confirmação:

```
$ find /usr/share -size +5M -exec du {} \; | sort -nr  
101608 /usr/share/icons/oxygen/icon-theme.cache  
42636 /usr/share/virtio-win/virtio-win-1.3.3.iso  
20564 /usr/share/fonts/cjkuni-uming/uming.ttc
```

- A opção `-ok` permite escolher, um de cada vez, se ou não o comando que você insere é executado sobre cada arquivo localizado. Por exemplo, você quer localizar todos os arquivos que pertencem a `joe` no diretório `/var/allusers` (e seus subdiretórios) e movê-los para o diretório `/tmp/joe`:

```
# find /var/allusers/ -user joe -ok mv {} /tmp/joe/ \;  
< mv ... /var/allusers/dict.dat > ?  
< mv ... /var/allusers/five > ? y
```

Observe no código anterior que você é questionado sobre cada arquivo localizado antes de ele ser movido para o diretório `/tmp/joe`. Você simplesmente digita `y` e pressiona Enter em cada linha para mover o arquivo, ou apenas pressiona Enter para ignorá-lo.

Para mais informações sobre o comando `find`, digite **man find**.

Pesquisando o conteúdo de arquivos com `grep`

Se quiser localizar os arquivos que contêm um termo de pesquisa específico, você pode usar o comando `grep`. Com `grep`, você pode pesquisar um único arquivo ou uma estrutura de diretórios inteira de arquivos recursivamente.

Ao pesquisar, você pode imprimir na tela (saída padrão) cada linha contendo o termo ou apenas listar o nome dos arquivos que contêm o termo de pesquisa. Por padrão, `grep` pesquisa fazendo distinção entre maiúsculas de minúsculas, embora você também possa pesquisar sem essa distinção.

Em vez de apenas pesquisar arquivos, você também pode usar `grep` para pesquisar a saída padrão. Assim, se um comando retorna um monte de texto e você quer localizar apenas as linhas que contêm determinado texto, pode usar `grep` para filtrar apenas o que lhe interessa.

Eis alguns exemplos de linhas de comando `grep`, usadas para localizar strings de texto em um ou mais arquivos:

```
$ grep desktop /etc/services
desktop-dna 2763/tcp      # Desktop DNA
desktop-dna 2763/udp      # Desktop DNA

$ grep -i desktop /etc/services
sco-dtmgr   617/tcp      # SCO Desktop Administration Server
sco-dtmgr   617/udp      # SCO Desktop Administration Server
airsync     2175/tcp     # Microsoft Desktop AirSync Protocol
...
```

No primeiro exemplo, um `grep` para a palavra `desktop` no arquivo `/etc/services` revelou duas linhas. Pesquisando de novo, utilizando `-i` para não diferenciar maiúsculas de minúsculas (como no segundo exemplo), foram reveladas 24 linhas de texto.

Para pesquisar por linhas que não contêm uma string de texto selecionada, use a opção `-v`. No exemplo a seguir, todas as linhas do arquivo `/etc/services` são exibidas, exceto aquelas que contêm o texto `tcp` (sem distinção entre maiúsculas e minúsculas):

```
$ grep -vi tcp /etc/services
```

Para fazer pesquisas recursivas, use a opção `-r` e um diretório como argumento. O exemplo a seguir inclui a opção `-l`, que apenas lista os arquivos que incluem o texto da pesquisa, sem mostrar as linhas reais de texto. Essa pesquisa revela arquivos que contêm o texto `peerdns` (sem distinção entre maiúsculas e minúsculas).

```
$ grep -rli peerdns /usr/share/doc/
/usr/share/doc/dnsmasq-2.48/setup.html
/usr/share/doc/initscripts-9.03.23/sysconfig.txt
```

O próximo exemplo pesquisa recursivamente o diretório `/etc/sysconfig` em busca do termo `root`. Ele lista todas as linhas em cada arquivo sob o diretório que contém esse texto. Para destacar o termo `root` em cada linha, a opção `--color` é adicionada. Por padrão, o termo localizado aparece em vermelho.

```
$ grep -ri --color root /etc/sysconfig/
```

Para pesquisar um termo na saída de um comando, você pode enviar a saída para o comando `grep`. Nesse exemplo, sei que os endereços IP são listados em linhas geradas pelo comando `ip` que incluem a string `inet`. Então, uso `grep` para exibir apenas essas linhas:

```
$ ip addr show | grep inet
```



```
inet 127.0.0.1/8 scope host lo
```

```
inet 192.168.1.231/24 brd 192.168.1.255 scope global wlan0
```

Resumo

Ser capaz de trabalhar com arquivos de texto simples é uma habilidade crucial para usar o Linux. Como os arquivos de configuração e tantos arquivos de documentos estão em formato de texto simples, você precisa se tornar proficiente com um editor de texto para efetivamente usar o Linux. Localizar nomes de arquivos e conteúdo em arquivos também é uma habilidade crucial. Neste capítulo, você aprendeu a usar os comandos `locate` e `find` para localizar arquivos e `grep` para pesquisar arquivos.

O próximo capítulo abrange uma variedade de formas de trabalhar com processos. Nele, você aprende a ver os processos que estão em execução, executar processos em primeiro e segundo planos e alterar processos (enviar sinais).

Exercícios

Use esses exercícios para testar seus conhecimentos em usar o editor de texto `vi` (ou `vim`), os comandos para localizar arquivos (`locate` e `find`) e os comandos para pesquisar arquivos (`grep`). Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostrados no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. Copie o arquivo `/etc/services` para o diretório `/tmp`. Abra o arquivo `/tmp/services` no `vim` e pesquise o termo `WorldWideWeb`. Mude isso para exibir `World Wide Web`.
2. Encontre o parágrafo seguinte em seu arquivo `/tmp/services` (se ele não estiver lá, escolha um parágrafo diferente) e mova-o para o final desse arquivo.

```
# Note that it is presently the policy of IANA to assign a
single well-known
# port number for both TCP and UDP; hence, most entries here
have two entries
# even if the protocol doesn't support UDP operations.
# Updated from RFC 1700, "Assigned Numbers" (October 1994).
Not all ports
# are included, only the more common ones.
```
3. Usando o modo `ex`, pesquise cada ocorrência do termo `tcp` (com distinção entre maiúsculas e minúsculas) no seu arquivo `/tmp/services` e mude-as para `WHATEVER`.
4. Como um usuário regular, pesquise sob o diretório `/etc` todos os arquivos chamados `passwd`. Redirecione mensagens de erro a partir da sua pesquisa para `/dev/null`.

5. Crie um diretório no seu diretório inicial chamado `TEST`. Crie três arquivos nesse diretório com os nomes `one`, `two` e `three`, e atribua permissões plenas de leitura/gravação/execução sobre esses arquivos para todo mundo (usuário, grupo e outros). Construa um comando `find` que localize esses arquivos e outros arquivos que têm permissões de gravação abertas para os “outros” a partir do seu diretório inicial e abaixo.
6. Localize arquivos no diretório `/usr/share/doc` que não tenham sido modificados há mais de 300 dias.
7. Crie um diretório `/tmp/FILES`. Localize todos os arquivos no diretório `/usr/share` que tenham mais do que 5MB e menos de 10MB e copie-os para o diretório `/tmp/FILES`.
8. Localize todos os arquivos no diretório `/tmp/FILES` e faça uma cópia de backup de cada arquivo no mesmo diretório. Use o nome de cada arquivo existente e apenas anexe `.mybackup` para criar cada arquivo de backup.
9. Instale o pacote `kernel-doc` no Fedora ou no Red Hat Enterprise Linux. Usando `grep`, pesquise dentro dos arquivos contidos no diretório `/usr/share/doc/kernel-doc*` pelo termo `e1000` (sem distinção entre maiúsculas e minúsculas) e liste o nome dos arquivos que contenham esse termo.
10. Pesquisa pelo termo `e1000` novamente no mesmo local, mas desta vez liste cada linha que contém o termo e destaque o termo com cores.

Gerenciando processos em execução

NESTE CAPÍTULO

Exibindo processos

Executando processos em primeiro e segundo planos

Eliminando e reiniciando processos

Além de ser um sistema operacional multiusuário, o Linux também é multitarefa. Ser *Multitarefa* significa que muitos programas podem ser executados nele ao mesmo tempo. Uma instância de um programa em execução é referida como um *processo*. O Linux fornece ferramentas para listar processos em execução, monitorar o uso do sistema e parar (ou eliminar) processos quando necessário.

A partir de um shell, você pode carregar processos e, então, pausar, parar ou eliminá-los. Você também pode colocá-los em segundo plano e trazê-los para o primeiro plano. Esse capítulo descreve ferramentas como `ps`, `top`, `kill`, `jobs` e outros comandos para listar e gerenciar processos.

Entendendo Processos

Um processo é uma instância em execução de um comando. Por exemplo, pode haver um comando `vi` no sistema. Mas se `vi` estiver sendo executado por 15 usuários diferentes, esse comando será representado por 15 diferentes processos em execução.

Um processo é identificado no sistema por aquilo que é referido como um ID de processo. Esse ID de processo é exclusivo do sistema atual. Em outras palavras, nenhum outro processo pode usar o mesmo número de ID de um processo que ainda estiver em execução. Mas depois que um processo é encerrado, outro processo pode reutilizar esse número.

Além de um número de ID, há outros atributos associados a um processo. Cada processo, quando executado, está associado a uma conta de usuário e a uma conta de grupo. Essa informação ajuda a determinar quais recursos do sistema o processo pode acessar. Por exemplo, os processos executados pelo usuário `root` tem muito mais acesso aos arquivos e recursos do sistema do que um processo executado por um usuário regular.

A capacidade de gerenciar os processos em seu sistema é fundamental para um administrador de sistema Linux. Há ocasiões em que “processos desenfreados” (*runaway processes*) podem matar o desempenho

do seu sistema. Encontrar e lidar com esses processos, com base em atributos como memória e uso de CPU, são abordados neste capítulo.

ta

Quando se quer obter informações sobre processos em execução, a maioria dessas informações é obtida a partir dos dados brutos armazenados no sistema de arquivos `/proc`. Cada processo armazena suas informações em um diretório `/proc`, nomeado de acordo com o ID desse processo. Você pode ver alguns desses dados brutos lendo o conteúdo de arquivos em um desses diretórios (usando comandos `cat` ou `less`).

Listando Processos

A partir da linha de comando, o comando `ps` é o mais antigo e comum para listar processos atualmente em execução no sistema. O comando `top` fornece uma abordagem mais orientada para a tela para a listagem de processos e também pode ser usado para alterar o estado do processo. Se estiver usando o desktop GNOME, você pode usar `gnome-system-monitor` para fornecer um meio gráfico de trabalhar com processos. Esses comandos são descritos nas próximas seções.

Listando processos com `ps`

O utilitário mais comum para verificar os processos em execução é o comando `ps`. Use-o para ver quais programas estão em execução, os recursos que eles estão usando e quem os está executando. O seguinte é um exemplo do comando `ps`:

```
$ ps u
USER  PID  %CPU  %MEM  VSZ   RSS  TTY   STAT  START  TIME  COMMAND
jake  2147  0.0   0.7   1836  1020  tty1  S+   14:50   0:00  -bash
jake  2310  0.0   0.7   2592   912  tty1  R+   18:22   0:00  ps u
```

Nesse exemplo, a opção `u` pede que sejam mostrados os nomes de usuário e outras informações, tais como o momento em que o processo foi iniciado e o uso de memória e CPU pelos processos associados com o usuário atual. Os processos descritos são associados com o terminal atual (`ttty1`). O conceito de terminal vem dos velhos tempos, quando as pessoas trabalhavam exclusivamente a partir de terminais de caracteres; por isso, um terminal geralmente representava uma única pessoa em uma única tela. Agora, você pode ter muitos “terminais” em uma única tela, abrindo múltiplos terminais virtuais ou janelas de terminal na área de trabalho.

Nessa sessão de shell, não há muito acontecendo. O primeiro processo exibe que o usuário chamado `jake` abriu um shell `bash` após efetuar login. O processo seguinte exibe que `jake` executou o comando `ps u`. O dispositivo terminal `ttty1` está sendo utilizado para a sessão de login. A coluna `STAT` representa o estado do processo, com `R` indicando um processo em execução e `S` representando um processo dormente.

ta

Os outros valores podem aparecer na coluna `STAT`. Por exemplo, um sinal de mais (+) indica que o processo está iniciado com as operações de primeiro plano.

A coluna `USER` exibe o nome do usuário que iniciou o processo. Cada processo é representado por um número de ID único denominado ID de processo (PID). Você pode usar o PID se alguma vez precisar eliminar um processo desenfreado ou enviar outro tipo de sinal para um processo. As colunas `%CPU` e `%MEM` mostram as percentagens de processador e de memória de acesso aleatório, respectivamente, que o processo está consumindo.

`VSZ` (*virtual set size* — tamanho da memória virtual) indica o tamanho do processo de imagem (em kilobytes) e `RSS` (*resident set size* — tamanho da memória residente) exibe o tamanho do programa na memória. Os tamanhos `VSZ` e `RSS` podem ser diferentes, porque `VSZ` é a quantidade de memória alocada para o processo, enquanto `RSS` é a quantidade que está realmente sendo usada.

`START` exibe a hora em que o processo começou a rodar e `TIME` exibe o tempo acumulado de uso do sistema. (Muitos comandos consomem pouco tempo da CPU, como refletido por `0:00` para processos que ainda não usaram nem um segundo inteiro de tempo da CPU.)

Muitos processos executados em um computador não estão associados a um terminal. Um sistema Linux normal tem muitos processos em execução em segundo plano. Processos do sistema em segundo plano executam tarefas como registrar em log a atividade do sistema ou ouvir dados provenientes da rede. Eles costumam ser iniciados quando o Linux inicializa e são executados continuamente até que o sistema seja desligado. Para percorrer todos os processos do usuário atual em execução no seu sistema Linux, adicione a barra vertical (|) e o comando `less` a `ps ux`:

```
$ ps ux | less
```

Para percorrer todos os processos em execução para todos os usuários em seu sistema, use o comando `ps aux` como segue:

```
$ ps aux | less
```

Uma barra vertical (localizada acima da barra invertida no teclado) permite direcionar a saída de um comando para ser a entrada do próximo comando. Nesse exemplo, a saída do comando `ps` (uma lista de processos) é direcionada para o comando `less`, o que permite que você veja essas informações página por página. Use a barra de espaço para ver página por página e digite `q` para terminar a lista. Você também pode usar as teclas de seta para mover uma linha de cada vez através da saída.

O comando `ps` também pode ser personalizado para exibir colunas selecionadas de informação e ordenar as informações por uma dessas colunas. Usando a opção `-o`, você pode usar palavras-chave para indicar as colunas que deseja listar com `ps`. Por exemplo, o exemplo a seguir lista todos os processos em execução (`-e`) e depois segue a opção `-o` com todas as colunas de informação que eu quero exibir, incluindo:

O ID do processo (`pid`), o nome do usuário (`user`), o ID do usuário (`uid`), o nome do grupo (`group`), o ID do grupo (`gid`), a memória virtual alocada (`vsz`), a memória residente usada (`rss`) e a linha de comando completa que foi executada (`comm`). Por padrão, a saída é ordenada pelo número de ID do processo.

```
$ ps -eo 'pid,user,uid,group,gid,vsz,rss,comm' | less
PID USER  GROUP      GID   VSZ   RSS COMMAND
  1 root    root        0 19324 1320  init
  2 root    root        0     0     0  kthreadd
```

Se você quiser ordenar por uma coluna específica, você pode usar a opção `sort=`. Por exemplo, para ver quais processos estão usando mais memória, eu ordenei pelo campo `rss`. Isso ordenará as linhas pelo uso da memória, do mais baixo para o mais alto. Como quero ver as maiores primeiro, coloquei um hífen na frente da opção para ordenar (`sort=-rss`).

```
$ ps -eo 'pid,user,group,gid,vsz,rss,comm' --sort=-rss | less
PID USER  GROUP      GID   VSZ   RSS COMMAND
12005 chris   chris      13597 1271008 522192 firefox
 5412 cnegus  cnegus    13597  949584 157268 thunderbird-bin
25870 cnegus  cnegus    13597 1332492 112952 swriter.bin
```

Consulte a página `man` do comando `ps` para obter informações sobre outras colunas de informação que você pode visualizar e usar como ordem de classificação.

Listando e alterando processos com `top`

O comando `top` oferece uma maneira orientada para tela de exibir os processos em execução no seu sistema. Com `top`, o padrão é exibir os processos com base em quanto tempo de CPU eles estão atualmente consumindo. Mas você também pode ordenar por outras colunas. Depois de identificar um processo malcomportado, você também pode usar `top` para eliminar (terminar completamente) ou repriorizar (“`renice`”) esse processo.

Se quiser ser capaz de eliminar ou repriorizar processos, você precisará executar `top` como o usuário `root`. Se só quiser exibir processos e, talvez, eliminar ou mudar seus próprios processos, você pode fazer isso como um usuário regular. A Figura 6.1 exibe um exemplo da janela de `top`:

ndo os processos em execução com `top`.

Informações gerais sobre o sistema aparecem na parte superior da saída de `top`, seguidas por informações sobre cada processo em execução (ou pelo menos tantos quantos caberem em sua tela). No topo, você pode ver há quanto tempo o sistema está ativo, quantos usuários estão conectados nele e quanta demanda tem havido sobre o ele no(s) último(s) 1, 5 e 10 minutos.

Outras informações gerais incluem quantos processos (tarefas) estão atualmente em execução, quanto da CPU está sendo usado e quanta memória RAM e swap estão disponíveis e sendo usadas. Depois das informações gerais, vêm as listas de cada processo, ordenadas pela porcentagem da CPU sendo usado por cada processo. Todas essas informações são exibidas novamente a cada 5 segundos, por padrão.

A lista que se segue inclui ações que você pode fazer com o `top` para mostrar informações de maneiras diferentes e modificar os processos em execução:

- Pressione **h** para ver as opções de ajuda e então pressione qualquer tecla para retornar à tela de `top`.
- Pressione **M** para ordenar por uso de memória em vez de CPU e então pressione **P** para voltar a ordenar por CPU.
- Pressione o número **1** para exibir o uso da CPU de todas as suas CPUs, se você tiver mais de uma CPU em seu sistema.
- Pressione **R** para ordenar inversamente sua saída.
- Pressione **u** e digite um nome de usuário para exibir processos apenas de um usuário específico.

Uma prática comum é usar `top` para encontrar processos que estão consumindo muita memória ou poder de processamento e, então, agir sobre esses processos de alguma forma. Um processo que consome CPU demais pode ser repriorizado para receber menos prioridade sobre os processadores. Um processo de memória consumindo muita memória pode ser eliminado. Com `top` em execução, eis aqui como repriorizar ou eliminar um processo:

- **Repriorizando um processo:** Observe o ID do processo que você deseja repriorizar e pressione **r**. Quando a mensagem do PID para repriorizar aparecer: digite o ID do processo que você deseja repriorizar. Quando solicitado a informar o valor para repriorizar o PID (Renice PID to value): digite um número entre -19 e 20 . (Consulte “Configurando a prioridade sobre o processador com `nice` e `renice`” neste capítulo para obter informações sobre os significados dos diferentes valores de `renice`.)
- **Eliminando um processo:** Observe o ID do processo que você deseja eliminar e pressione **k**. Digite **15** para terminar de forma limpa ou **9** para simplesmente eliminar o processo imediatamente. (Veja “Eliminando processos com `kill` e `killall`” mais adiante neste capítulo, para obter mais informações sobre o uso de sinais diferentes que você pode enviar para processos).

Listando processos com o System Monitor

Se você tem o desktop GNOME disponível em seu sistema Linux, System Monitor (`gnome-system-monitor`) está disponível para fornecer uma forma mais gráfica de exibir os processos em seu sistema. Você ordena os processos clicando em colunas e pode clicar com o botão direito do mouse nesses processos a fim de pará-los, eliminá-los ou repriorizá-los.

Para iniciar o System Monitor no desktop GNOME, selecione Applications ⇒ System Tools ⇒ System Monitor. Então, selecione a guia Processes. A Figura 6.2 exibe um exemplo da janela do System Monitor.

janela System Monitor para visualizar e alterar processos em execução.

Por padrão, apenas os processos em execução associados à sua conta de usuário são exibidos. Esses processos são listados alfabeticamente em primeiro lugar. Você pode reordenar os processos clicando em qualquer um dos cabeçalhos de campo (para frente e invertido). Por exemplo, clique no cabeçalho %CPU para ver quais processos estão consumindo mais poder de processamento. Clique no cabeçalho Memory para ver quais processos consomem mais memória.

Você pode mudar seus processos de várias maneiras clicando no nome de um deles e selecionando uma opção no menu que aparece (ver Figura 6.3, por exemplo).

orize, elimine ou pause um processo a partir da janela System Monitor.

Eis algumas das coisas que você pode fazer com um processo a partir do menu em que você clicou:

- **Stop Process** — Pausa o processo, assim nenhum processamento ocorre até que você selecione Continue Process. (Isso é o mesmo que pressionar Ctrl+Z em um processo a partir do shell.)
- **Continue Process** — Continua a execução de um processo pausado.
- **End Process** — Envia um sinal Terminate (15) para um processo. Na maioria dos casos, isso irá terminar o processo de forma limpa.
- **Kill Process** — Envia um sinal KILL (9) para um processo, devendo eliminá-lo imediatamente, independente de isso poder ou não ser feito de maneira limpa.
- **Change Priority** — Apresenta uma barra deslizante a partir da qual você pode repriorizar um processo. A prioridade normal é 0. Para obter uma prioridade de processador melhor, use um número negativo de -1 a -20. Para ter uma prioridade mais baixa do processador, use um número positivo (0 a 19). Somente o usuário root pode atribuir prioridades negativas, por isso, você precisa fornecer a senha de root quando solicitado, a fim de definir um valor negativo de nice.
- **Memory Maps** — Permite visualizar o mapa de memória do sistema para ver quais bibliotecas e outros componentes estão sendo mantidos na memória para o processo.
- **Open Files** — Permite ver quais arquivos estão sendo mantidos abertos pelo processo.

Você pode exibir a execução de processos associados com outros usuários além de você próprio. Para fazer isso, selecione qualquer processo na tela (basta clicar nele). Então, na barra de menus, selecione View ⇒ All Processes. Você só pode modificar os processos que não possui se for o usuário root ou se puder fornecer a senha de root quando solicitado depois de tentar mudar um processo.

Há momentos em que você não terá o luxo de trabalhar com uma interface gráfica. Para mudar os processos sem uma interface gráfica, há um conjunto de comandos e teclas que você pode usar para alterar, pausar ou eliminar processos em execução. Alguns deles são descritos a seguir.

Gerenciando Processos em Primeiro e Segundo Planos

Se estiver usando Linux em uma rede ou a partir um *terminal burro* (um monitor que permite entrada de texto apenas, com nenhum suporte de interface gráfica), seu shell pode ser tudo o que você tem. Você pode estar acostumado a um ambiente gráfico em que há um monte de programas ativos ao mesmo tempo para que você possa alternar entre eles conforme necessário. Essa coisa de shell pode parecer bem limitada.

Embora o shell `bash` não inclua uma interface gráfica para executar muitos programas de uma só vez, ele realmente o deixa mover programas ativos entre o primeiro e o segundo planos. Dessa forma, você pode ter um monte de coisas rodando e seletivamente escolher o que você quer tratar no momento.

Há várias maneiras de colocar um programa ativo em segundo plano. Uma delas é adicionar uma letra E comercial (&) ao final de uma linha de comando quando você executá-lo. Você também pode usar o comando `at` para executar comandos de tal maneira que eles não estejam conectados ao shell.

Para parar a execução de um comando e colocá-lo em segundo plano, pressione `Ctrl+Z`. Depois que o comando parar, você pode trazê-lo de volta para o primeiro plano para que seja executado (o comando `fg`) ou iniciá-lo rodando em segundo plano (o comando `bg`). Tenha em mente que qualquer comando executado em segundo plano pode vomitar saída durante comandos executados posteriormente a partir desse shell. Por exemplo, se a saída de um comando aparece a partir de um comando rodando em segundo plano durante uma sessão do `vi`, basta pressionar `Ctrl+L` para redesenhar a tela e se livrar dessa saída.

ca

evitar que a saída apareça, você deve fazer com que qualquer processo executando em segundo plano envie sua saída para um arquivo ou para `null` (acrescente `2> /dev/null` ao final da linha de comando).

Iniciando processos em segundo plano

Se tiver programas que deseja executar enquanto continua a trabalhar no shell, você pode colocá-los em segundo plano. Para colocar um programa em segundo plano no momento de executá-lo, insira uma letra E comercial (&) no final da linha de comando, assim:

```
$ find /usr > /tmp/allusrfiles &
[3] 15971
```

Esse exemplo de comando encontra todos os arquivos em seu sistema Linux (a partir de `/usr`), imprime os nomes de arquivo e coloca esses nomes no arquivo `/tmp/allusrfiles`. A letra E comercial (&) executa essa linha de comando em segundo plano. Observe que o número da tarefa (*job*), [3], e o número de ID do processo, 15971, são exibidos quando o comando é iniciado. Para verificar os comandos que você tem em execução em segundo plano, use o comando `jobs`, como segue:

```
$ jobs
[1] Stopped (tty output) vi /tmp/myfile
[2] Running find /usr -print > /tmp/allusrfiles &
[3] Running nroff -man /usr/man2/* >/tmp/man2 &
[4]- Running nroff -man /usr/man3/* >/tmp/man3 &
[5]+ Stopped nroff -man /usr/man4/* >/tmp/man4
```

A primeira tarefa exibe um comando de edição de texto (`vi`) que eu coloquei em segundo plano e parei pressionando `Ctrl+Z` enquanto eu estava editando. A tarefa 2 exibe o comando `find` que acabei de executar. As tarefas 3 e 4 mostram comandos `nroff` atualmente em execução em segundo plano. A tarefa 5 estava em execução no shell (primeiro plano) até que decidi que muitos processos estavam rodando e

pressionei Ctrl+Z para interrompê-la até que alguns processos fossem completados.

O sinal de mais (+) ao lado do número 5 mostra que esse processo foi o mais recentemente colocado em segundo plano. O sinal de menos (-) ao lado do número 4 mostra que esse processo foi colocado em segundo plano antes da tarefa em segundo plano mais recente. Como o trabalho 1 requer entrada de terminal, ele não pode ser executado em segundo plano. Como resultado, ele ficará parado (Stopped) até ser trazido para o primeiro plano novamente.

ca

Ver o ID do processo para a tarefa em segundo plano, acrescente uma opção `-l` (a letra L minúscula) ao comando `jobs`. Se digitar `ps`, você pode usar o ID do processo para descobrir qual comando é para uma tarefa de segundo plano específico.

Utilizando comandos em primeiro e em segundo plano

Continuando com o exemplo, você pode trazer qualquer um dos comandos na lista de tarefas para o primeiro plano. Por exemplo, para editar `myfile` novamente, digite:

```
$ fg %1
```

Como resultado, o comando `vi` se abre novamente. Todo o texto é como era quando você parou o trabalho `vi`.

enção

Quando se coloca um processador de texto, ou programa semelhante, em segundo plano, certifique-se de salvar seu trabalho. É fácil esquecer-se de que há um programa em segundo plano, e você perderá seus dados se sair ou o computador for reiniciado mais tarde.

Para referenciar uma tarefa em segundo plano (cancelar ou trazê-la para o primeiro plano), use um sinal de porcentagem (%), seguido pelo número da tarefa. Você também pode utilizar o seguinte para referenciar um trabalho em segundo plano:

- `%` — Referencia o comando mais recente colocado em segundo plano (indicado pelo sinal de mais quando você digita o comando `jobs`). Essa ação coloca o comando no primeiro plano.
- `%string` — Referencia uma tarefa em que o comando começa com uma determinada *string* de caracteres. A *string* não deve ser ambígua. (Em outras palavras, digitar `%vi` quando há dois comandos `vi` em segundo plano resulta em uma mensagem de erro.)
- `/?string` — Referencia uma tarefa em que a linha de comando contém uma **string** em qualquer ponto. A *string* não deve ser ambígua ou a correspondência falhará.

- %-- — Referencia a tarefa anterior parada antes da mais recentemente parada.

Se um comando estiver parado, você pode iniciá-lo novamente para rodar em segundo plano usando o comando `bg`. Por exemplo, considere a tarefa 5 da lista de tarefas no exemplo anterior:

```
[5]+ Stopped          nroff -man man4/* >/tmp/man4
```

Digite o seguinte:

```
$ bg %5
```

Depois disso, o trabalho é executado em segundo plano. Sua entrada `jobs` aparece assim:

```
[5] Running          nroff -man man4/* >/tmp/man4 &
```

Eliminando e Repriorizando Processos

Assim como pode mudar o comportamento de um processo utilizando ferramentas gráficas como o System Monitor (descrito anteriormente neste capítulo), você também pode usar ferramentas de linha de comando para eliminar um processo ou mudar sua prioridade de CPU. O comando `kill` pode enviar um sinal *kill* para qualquer processo a fim de encerrá-lo, supondo que você tem permissão para fazer isso. Ele também pode enviar sinais diferentes para um processo para de outra maneira alterar seu comportamento. Os comandos `nice` e `renice` podem ser usados para definir ou alterar a prioridade de um processo sobre um processador.

Eliminando processos com `kill` e `killall`

Embora geralmente usado para terminar um processo em execução, os comandos `kill` e `killall` podem realmente ser usados para enviar qualquer sinal válido para um processo em execução. Além de fazer um processo terminar, um sinal pode dizer para um processo reler os arquivos de configuração, pausar (parar) ou continuar depois de uma pausa, para citar algumas possibilidades.

Sinais são representados por números e nomes. Sinais que você pode enviar mais comumente a partir de um comando incluem `SIGKILL` (9), `SIGTERM` (15) e `SIGHUP` (1). O sinal padrão é `SIGTERM`, que tenta encerrar um processo de forma limpa. Para eliminar um processo imediatamente, você pode usar `SIGKILL`. O sinal `SIGHUP` instrui um processo a reler seus arquivos de configuração. `SIGSTOP` pausa um processo, enquanto `SIGCONT` continua um processo parado.

Diferentes processos respondem a diferentes sinais. Mas processos não podem bloquear os sinais `SIGKILL` e `SIGSTOP`. A Tabela 6.1 exibe exemplos de alguns sinais (digite `man 7 signals` para ler sobre outros sinais disponíveis):

TABELA 6.1 Os Sinais Disponíveis no Linux

al	Número	Descrição
----	--------	-----------

GHUP	1	Hangup detectado no terminal de controle ou eliminação do processo de controle.
GINT	2	Interrompe a partir do teclado.
GQUIT	3	Sai do teclado.
GABRT	6	Sinal de abortar a partir de <code>abort(3)</code> .
GKILL	9	Sinal de eliminação.
GTERM	15	Sinal de término.
GCONT	19,18,25	Continua, se parado.
GSTOP	17,19,23	Para o processo.

Observe que há vários números de sinais possíveis para `SIGCONT` e `SIGSTOP` porque diferentes números são usados em diferentes arquiteturas de computador. Para arquiteturas x86 e Power PC, use o valor médio. O primeiro valor normalmente funciona para Alphas e Sparcs, enquanto o último é para arquitetura MIPS.

Usando kill para sinalizar processos por PID

Utilizando comandos como `ps` e `top`, você pode encontrar processos para os quais deseja enviar um sinal. Então, você pode usar o ID de processo dele como uma opção para o comando `kill`, juntamente com o sinal que você deseja enviar.

Por exemplo, você executa o comando `top` e vê que o processo `bigcommand` é o que mais está consumindo suas capacidades de processamento:

```

  PID USER      PR  NI   VIRT  RES  SHR  S %CPU  %MEM    TIME+  COMMAND
 10432 chris    20   0   471m 121m  18m  S 99.9   3.2   77:01.76 bigcommand

```

Aqui, o processo `bigcommand` está consumindo 99,9% da CPU. Você decide que quer eliminá-lo para que outros processos tenham chance de usar a CPU. Se você usar o ID de processo do processo `bigcommand` em execução, eis alguns exemplos do comando `kill` que você poderia usar para eliminar esse processo:

```

$ kill 10432
$ kill -15 10432
$ kill -SIGKILL 10432

```

O sinal padrão enviado por `kill` é 15 (`SIGTERM`); portanto, os dois primeiros exemplos têm exatamente os mesmos resultados. Ocasionalmente, um `SIGTERM` não elimina um processo, de modo que um `SIGKILL` pode ser necessário para eliminá-lo. Em vez de `SIGKILL`, você poderia usar `-9`.

Outro sinal útil é `SIGHUP`. Alguns processos de servidor, tais como o processo `httpd`, que fornece serviços web, vai responder a um sinal `SIGHUP` (1) relendo seus arquivos de configuração. De fato, o comando `service httpd reload` efetivamente envia `SIGHUP` para processos `httpd` rodando no seu sistema para dizer-lhes que os arquivos de configuração precisam ser lidos novamente. Assim, se o processo `httpd` tinha um PID de 1833, você pode usar esse comando para fazê-lo ler os arquivos de

configuração novamente:

```
$ kill -1 1833
```

Usando killall para sinalizar processos por nome

Com o comando `killall`, você pode sinalizar processos por nome em vez de ID do processo. A vantagem é que você não tem de olhar para o ID de processo do processo que você quer eliminar. A desvantagem potencial é que você pode eliminar mais processos do que o desejado se não for cuidadoso. (Por exemplo, digitar `killall bash` pode eliminar vários shells que você não quer eliminar).

Assim como o comando `kill`, `killall` usa `SIGTERM` (sinal 15), se você não digitar explicitamente um número de sinal. Também como ocorre com `kill`, você pode enviar qualquer sinal que quiser para o processo que indicar com `killall`. Por exemplo, se vir um processo chamado `testme` rodando em seu sistema e quiser eliminá-lo, você pode simplesmente digitar o seguinte:

```
$ killall -9 testme
```

O comando `killall` pode ser particularmente útil se você quiser eliminar vários comandos que têm o mesmo nome.

Configurando a prioridade sobre o processador com nice e renice

Quando o kernel Linux tenta decidir quais processos em execução obtêm acesso às CPUs do sistema, uma das coisas que ele leva em conta é o valor de `nice` configurado sobre o processo. Cada processo em execução no sistema tem um valor de `nice` entre `-20` e `19`. Por padrão, o valor de `nice` é configurado como `0`. Eis alguns fatos sobre valores de `nice`:

- Quanto menor o valor de `nice`, mais acesso à CPU o processo terá. Em outras palavras, quanto menos prioridade um processo tiver, menos atenção da CPU ele receberá. Portanto, um valor de `nice` de `-20` recebe mais atenção do que um processo com um valor de `nice` de `19`.
- Um usuário comum pode definir valores de `nice` somente entre `0` e `19`. Valores negativos não são permitidos.
- Um usuário comum pode configurar o valor de `nice` mais alto, não mais baixo. Assim, por exemplo, se um usuário configurar o valor de `nice` de um processo como `10` e depois quiser configurá-lo de volta para `5`, essa ação falhará. Da mesma forma, qualquer tentativa de configurar um valor negativo falhará.
- Um usuário comum somente pode configurar o valor de `nice` nos processos dele próprio.
- O usuário `root` pode configurar o valor de `nice` em qualquer processo com qualquer valor válido, para cima ou para baixo.

Você pode usar o comando `nice` para executar um comando com um determinado valor de `nice`. Uma vez que um processo está em execução, você pode alterar o valor de `nice` usando o comando `renice`, juntamente com o ID de processo do processo, como no exemplo que se segue:

```
# nice +5 updatedb &
```

O comando `updatedb` é utilizado para gerar o banco de dados de localização manualmente coletando nomes de arquivos em todo o sistema de arquivos. Nesse caso, eu só queria que `updatedb` fosse executado em segundo plano (`&`) e não interrompesse as tarefas que estão sendo feitas por outros processos no sistema. Executei o comando `top` para me certificar de que o valor de `nice` foi configurado corretamente:

```
PID USER      PR  NI   VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
20284 root        25   5  98.7m  932  644 D  2.7  0.0   0:00.96 updatedb
```

Observe que, na coluna `NI`, o valor de `nice` é configurado como 5. Como o comando foi executado como usuário `root`, o usuário `root` pode diminuir o valor de `nice` mais tarde usando o comando `renice`. (Lembre-se de que um usuário comum não pode reduzir o valor de `nice` nem nunca configurá-lo como um número negativo.) Veja como você alteraria o valor de `nice` para o comando `updatedb` que acabamos de executar para `-5`

```
# renice -n -5 20284
```

Se você executou o comando `top` novamente, pode notar que o comando `updatedb` está no topo ou próximo do topo da lista de processos consumindo tempo de CPU porque você deu a ele prioridade para obter mais atenção da CPU.

Resumo

Mesmo em um sistema Linux em que não há muita atividade, normalmente há dezenas ou até mesmo centenas de processos em execução em segundo plano. Usando as ferramentas descritas neste capítulo, você pode visualizar e gerenciar os processos em execução no seu sistema. O gerenciamento de processos inclui visualizá-los de diferentes maneiras, executá-los em primeiro ou segundo plano, e eliminá-los ou repriorizá-los.

No próximo capítulo, você aprenderá a combinar comandos e funções de programação em arquivos que podem ser executados como scripts de shell.

Exercícios

Use esses exercícios para testar seus conhecimentos em visualização de processos em execução e, então, alterá-los mais tarde eliminando-os ou alterando a prioridade do processador (valor de `nice`). Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostradas no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. Liste todos os processos em execução no sistema, mostrando um conjunto completo de colunas. Redirecione essa saída para o comando `less` de modo que você possa percorrer a

lista de processos página por página.

2. Liste todos os processos em execução no sistema e ordene-os pelo nome do usuário que executa cada processo.
3. Liste todos os processos em execução no sistema e apresente as seguintes colunas de informação: ID do processo, nome do usuário, nome do grupo, tamanho da memória virtual, tamanho da memória residente e o comando.
4. Execute o comando `top` para ver os processos em execução no sistema. Ordene por uso da CPU e consumo da memória, do maior para o menor e vice-versa.
5. Inicie o processo de `gedit` a partir de seu desktop. Certifique-se de executá-lo sob o nome de usuário com que você efetuou login. Use a janela System Monitor para eliminar esse processo.
6. Execute o processo `gedit` novamente. Desta vez, usando o comando `kill`, envie um sinal para o processo `gedit` que o faça pausar (parar). Tente digitar algum texto na janela do `gedit` e certifique-se de que nenhum texto ainda aparece.
7. Use o comando `killall` para dizer ao comando `gedit` que você parou no exercício anterior para continuar trabalhando. Verifique se o texto que você digita depois que `gedit` foi interrompido agora aparece na janela.
8. Instale o comando `xeyes` (no Red Hat Enterprise Linux, ele está no pacote `xorg-x11-apps`). Execute o comando `xeyes` cerca de 20 vezes em segundo plano para que 20 janelas `xeyes` apareçam na tela. Mova o mouse ao redor e note os olhos observarem o ponteiro do mouse. Quando você se cansar dessa diversão, elimine todos os processos `xeyes` em um comando usando `killall`.
9. Como um usuário regular, execute o comando `gedit` para que ele inicie com um valor de `nice` de 5.
10. Usando o comando `renice`, altere o valor de `nice` do comando `gedit` que você acabou de iniciar para 7. Utilize qualquer comando que quiser para verificar se o valor atual de `nice` para o comando `gedit` já está configurado como 7.

Escrevendo Scripts de Shell Simples

NESTE CAPÍTULO

Trabalhando com scripts de shell

Aritmética em scripts de shell

Executando loops e cases em scripts de shell

Criando scripts de shell simples

Você nunca concluiria nenhum trabalho se precisasse digitar todos os comandos que devem ser executados em seu sistema Linux quando ele é iniciado. Da mesma forma, você pode trabalhar de maneira mais eficiente se agrupar conjuntos de comandos que executa o tempo todo. Os scripts de shell podem cuidar dessas tarefas.

Um script de shell é um grupo de comandos, funções, variáveis, ou qualquer outra coisa que você pode usar em um shell. Esses itens são digitados em um arquivo de texto simples que pode ser executado como um comando. A maioria dos sistemas Linux utiliza scripts de shell de inicialização do sistema durante a inicialização do sistema para executar comandos necessários para fazer os serviços funcionarem. Você pode criar seus próprios scripts de shell para automatizar tarefas que precisa fazer regularmente.

Este capítulo fornece uma visão geral rudimentar do funcionamento interno dos scripts de shell e como eles podem ser utilizados. Você aprenderá como scripts de shell são responsáveis pelas mensagens que rolam no console do sistema durante a inicialização e como scripts simples podem ser aproveitados para um recurso de agendamento (como `cron` ou `at`) a fim de simplificar as tarefas administrativas.

Entendendo Scripts do Shell

Você nunca teve uma tarefa que precisava realizar repetidamente e que exigia muito trabalho de digitação na linha de comando? Nunca pensou: “Será que eu não poderia escrever apenas um comando para fazer tudo isso?” Talvez um script de shell seja o que você está procurando.

Scripts de shell são os equivalentes a arquivos em lote do MS-DOS e podem conter longas listas de comandos, controle de fluxo complexo, avaliações aritméticas, variáveis e funções definidas pelo usuário

e testes de condição sofisticados. Scripts de shell são capazes de lidar com tudo, desde uma simples linha de comando até algo tão complexo quanto inicializar seu sistema Linux.

De fato, como você lerá neste capítulo, os sistemas Linux fazem exatamente isso. Eles usam scripts para verificar e montar todos os sistemas de arquivo, configurar seus consoles, configurar a rede, carregar todos os serviços do sistema e, por fim, fornecer uma tela de login. Embora dezenas de shells diferentes estejam disponíveis no Linux, o shell padrão é chamado `bash`, o Bourne Again Shell.

Executando e depurando scripts de shell

Uma das principais vantagens dos scripts de shell é que eles podem ser abertos em qualquer editor de texto para ver o que fazem. Mas a grande desvantagem é que os scripts de shell grandes ou complexos costumam executar mais lentamente do que os programas compilados. Há duas maneiras básicas de executar um script de shell:

- O nome do arquivo é usado como um argumento para o shell (como em `bash myscript`). Nesse método, o arquivo não precisa ser executável; ele apenas contém uma lista de comandos de shell. O shell especificado na linha de comando é utilizado para interpretar os comandos no arquivo de script. Isso é mais comum para tarefas simples e rápidas.
- O script de shell também pode ter o nome do interpretador colocado na primeira linha do script precedido por `#!` (como em `#!/bin/bash`) e ter o bit de execução do arquivo contendo o script configurado (usando `chmod +x nome_do_arquivo`). Você pode então executar o script como qualquer outro programa em seu caminho simplesmente digitando o nome do script na linha de comando.

Quando os scripts são executados de qualquer uma dessas duas maneiras, as opções para o programa podem ser especificadas na linha de comando. Qualquer coisa seguinte ao nome do script é referida como um *argumento de linha de comando*.

Como ocorre ao escrever qualquer software, não há substituto para um projeto claro e bem planejado e muitos comentários. O sinal de jogo da velha (`#`) precede os comentários e pode ocupar até uma linha inteira ou residir na mesma linha após o código de script. É melhor implementar scripts de shell mais complexos em etapas, certificando-se de que a lógica é correta em cada passo antes de continuar. Eis algumas boas dicas concisas para garantir que as coisas funcionem como esperado durante os testes:

- Em alguns casos, você pode acrescentar um comando `echo` no início das linhas dentro do corpo de um loop e colocá-lo entre aspas. Desta maneira, em vez de executar o código, você pode ver o que será executado sem fazer quaisquer alterações permanentes.
- Para alcançar o mesmo objetivo, você pode colocar comandos `echo` de teste por todo o código. Se essas linhas forem impressas, você sabe que o desvio lógico correto está sendo conduzido.
- Você pode usar `set -x` próximo do início do script para exibir cada comando que é executado ou carregar seus scripts usando:

```
$ bash -x myscript
```

- Como os scripts úteis têm uma tendência a crescer ao longo do tempo, manter seu código legível à medida em que progride é extremamente importante. Faça o máximo que puder para manter a lógica de seu código limpa e fácil de seguir.

Entendendo variáveis de shell

Muitas vezes, dentro de um script de shell, você quer reutilizar determinados itens de informação. Durante o curso de processamento do script de shell, o nome ou número representando essa informação pode mudar. Para armazenar as informações utilizadas por um script de shell, de tal maneira que ele possa ser reutilizado facilmente, você pode configurar variáveis. Os nomes de variáveis dentro de scripts de shell diferenciam maiúsculas e minúsculas e podem ser definidos da seguinte maneira:

```
NAME=value
```

A primeira parte de uma variável é o nome de variável e a segunda parte é o conjunto de valores para esse nome. Certifique-se de que `NAME` e `value` estejam grudados ao sinal de igual, sem espaços. Variáveis podem ser atribuídas a partir de constantes, como texto, números e sublinhados. Isso é útil para inicializar valores ou poupar trabalho de digitação para constantes longas. Os exemplos a seguir mostram variáveis definidas em uma string de caracteres (`CITY`) e um valor numérico (`PI`)

```
CITY="Springfield"  
PI=3.14159265
```

As variáveis podem conter a saída de um comando ou uma sequência de comandos. Você pode fazer isso precedendo o comando com um sinal de cifrão e parêntese de abertura e seguindo-o com um parêntese de fechamento. Por exemplo, `MYDATE=$(date)` atribui a saída do comando `date` à variável `MYDATE`. Colocar o comando entre crases (```) pode ter o mesmo efeito. Nesse caso, o comando `date` é executado quando a variável é configurada e não cada vez que a variável é lida.

Escapando caracteres de shell especiais

Lembre-se em mente que caracteres como cifrão (`$`), crase (```), asterisco (`*`), ponto de exclamação (`!`) e outros têm um significado especial para o shell, como você verá à medida que avançar neste capítulo. Em algumas ocasiões, você quer que o shell use um significado diferente desses caracteres e outras vezes não. Por exemplo, se digitasse `echo $HOME`, o shell pensaria que você quis exibir o nome de seu diretório inicial (armazenado na variável `HOME`) na tela (como `/home/chris`), porque um `$` indica que um nome de variável se segue esse caractere.

Se você quisesse exibir literalmente `$HOME`, precisaria “escapar” (isto é, tratar literalmente) o

Digitar `echo `\$HOME`` ou `echo \$HOME` exibiria `\$HOME` literalmente na tela. Portanto, se você quer que o shell interprete um único caractere literalmente, preceda-o com uma barra invertida (`\`). Para ter um conjunto de caracteres interpretados literalmente, coloque-os entre aspas simples (`'`).

O uso de aspas duplas é um pouco mais complicado. Coloque um conjunto de texto entre aspas duplas se quiser que todos exceto alguns caracteres sejam usados literalmente. Por exemplo, quando o texto colocado entre aspas, o cifrão (`$`), crases (```) e pontos de exclamação (`!`) são interpretados especialmente, mas outros caracteres (como um asterisco) não o são. Digite as duas linhas para ver a saída diferente (mostrada à direita):

```
echo `\$HOME` *** `date`      \$HOME *** `date`
echo "\$HOME `date`"         /home/chris *** Tue Mar 20 16:56:52 EDT 2012
```

Usar variáveis é uma ótima maneira de obter informações que podem mudar de um computador para outro ou de um dia para outro. O exemplo a seguir configura a saída do comando `uname -n` para a variável `MACHINE`. Então, uso parênteses para configurar `NUM_FILES` como o número de arquivos no diretório atual redirecionando (`|`) a saída do comando `ls` para o comando de contagem de palavras (`wc -l`).

```
MACHINE='uname -n'
NUM_FILES=$(/bin/ls | wc -l)
```

As variáveis também podem conter o valor de outras variáveis. Isso é útil quando você tem de preservar um valor que mudará a fim de poder usá-lo mais tarde no script. Aqui, `BALANCE` é configurado para o valor da variável `CurBalance`

```
BALANCE="$CurBalance"
```

ta

tribuir variáveis, use apenas o nome da variável (por exemplo, `BALANCE`). Ao referenciar uma variável, o que fica que você quer o *valor* dela, preceda-a com um sinal de cifrão (como em `$CurBalance`). O resultado e último é que você obtém o valor da variável e não o nome dela em si.

Parâmetros de shell posicionais especiais

Existem variáveis especiais que o shell atribui para você. Um conjunto de variáveis comumente usadas é chamado de *parâmetros posicionais* ou *argumentos de linha de comando* e é referenciado como `$0`, `$1`, `$2`, `$3...` `$n`. `$0` é especial e recebe o nome usado para chamar o script; os outros recebem os valores dos parâmetros passados na linha de comando, na ordem em que apareceram. Por exemplo, digamos que você tem um script de shell chamado `myscript` que contém o seguinte:

```
#!/bin/bash
```

```
# Script to echo out command-line arguments
echo "The first argument is $1, the second is $2."
echo "The command itself is called $0."
```

A listagem a seguir mostra o que aconteceria se você executasse esse comando com `foo` e `bar` como argumentos:

```
$ myscript foo bar
The first argument is foo, the second is bar.
The command itself is called /home/chris/bin/myscript.
```

Como você pode ver, o parâmetro posicional `$0` é o caminho completo ou caminho relativo para `myscript`, `$1` é `foo` e `$2` é `bar`.

Outra variável, `$#`, informa quantos parâmetros seu script recebeu. No exemplo, `$#` seria 2. A variável `$@` armazena todos os argumentos inseridos na linha de comando. Outra variável de shell especial particularmente útil é `$?`, que recebe o status de saída do último comando executado. Em geral, um valor de zero significa que o comando foi encerrado com sucesso e qualquer coisa diferente de zero indica um erro de qualquer tipo. Para uma lista completa das variáveis de shell especiais, consulte a página `man` do `bash`.

Lendo parâmetros

Usando o comando `read`, você pode pedir informações ao usuário e, então, armazenar e usar essas informações mais tarde em seu script. Eis um exemplo de um script que usa o comando `read`:

```
#!/bin/bash
read -p "Type in an adjective, noun and verb (past tense): " adj1
noun1 verb1
echo "He sighed and $verb1 to the elixir. Then he ate the $adj1
    $noun1."
```

Nesse script, depois de pedir um adjetivo, um substantivo e um verbo, o usuário deve inserir palavras que são então atribuídas às variáveis `adj1`, `noun1` e `verb1`. Essas três variáveis são incluídas em uma frase tola, que é exibida na tela. Se o script se chamasse `sillyscript`, eis um exemplo de como isso poderia ser executado:

```
$ sillyscript
Type in an adjective, noun and verb (past tense): hairy football
danced
He sighed and danced to the elixir. Then he ate the hairy football.
```

Expansão de parâmetros no bash

Como mencionado anteriormente, se quiser o valor de uma variável, você deve precedê-la com um `$` (por exemplo, `$CITY`). Essa é apenas uma abreviação para a notação `${CITY}`; as chaves são

utilizadas quando o valor do parâmetro tem de ser colocado ao lado de outro texto sem um espaço. O bash tem regras especiais que permitem expandir o valor de uma variável de diferentes maneiras. Conhecer todas as regras é provavelmente um exagero para uma rápida introdução a scripts de shell, mas a lista a seguir apresenta algumas construções comuns que provavelmente você verá em scripts de bash que encontrador no seu sistema Linux.

- `${var:-value}` — Se a variável estiver em branco ou vazia, expande isso para *valor*.
- `${var#pattern}` — Corta a correspondência mais curta para o *padrão* da frente do valor de *var*.
- `${var##pattern}` — Corta a correspondência mais longa para o *padrão* a partir da frente do valor de *var*.
- `${var%pattern}` — Corta a correspondência mais curta para o *padrão* a partir do final do valor de *var*.
- `${var%%pattern}` — Corta a correspondência mais longa para o *padrão* a partir do final do valor de *var*.

Experimente digitar os seguintes comandos a partir de um shell para testar como a expansão de parâmetros funciona:

```
$ THIS="Example"
$ THIS=${THIS:-"Not Set"}
$ THAT=${THAT:-"Not Set"}
$ echo $THIS
Example
$ echo $THAT
Not Set
```

Nos exemplos aqui, a variável `THIS` é configurada inicialmente com a palavra `Example`. Nas próximas duas linhas, as variáveis `THIS` e `THAT` são configuradas com seus valores atuais ou `Not Set`, se não estiverem atualmente configuradas. Observe que, como acabei de configurar `THIS` com a string `Example`, quando ecoo o valor de `THIS` ele aparece como `Example`. Mas `THAT` não foi configurado, ele aparece como `Not Set`.

ta
esto desta seção, mostro como variáveis e comandos podem aparecer em um script de shell. Para experimentar quer um desses exemplos, porém, você pode simplesmente digitá-los em um shell, como mostrado no exemplo rior.

No exemplo a seguir, `MYFILENAME` é definido como `/home/digby/myfile.txt`. Em seguida, a variável `FILE` é configurada como `myfile.txt` e `DIR` é configurada como `/home/digby`. Na variável `NAME`, o nome do arquivo é cortado para simplesmente `myfile`; portanto, na variável `EXTENSION`, a extensão do arquivo é configurada como `txt`. (Para experimentar estes, você pode

digitá-los na linha de comando, como no exemplo anterior, e, então, exibir o valor de cada variável para ver como ela está configurada.) Digite o código à esquerda. O material do lado direito descreve a ação.

```
MYFILENAME="/home/digby/myfile.txt" – Define o valor de
MYFILENAME
FILE=${MYFILENAME##*/} – FILE torna-se "myfile.txt"
DIR=${MYFILENAME%/ *} – DIR torna-se "/home/digby"
NAME=${FILE%.*} – NAME torna-se "myfile"
EXTENSION=${FILE##*.} – EXTENSION torna-se "txt"
```

Fazendo aritmética em scripts de shell

O bash usa *variáveis não tipadas*, o que significa que normalmente ele trata variáveis como strings ou texto, mas pode alterá-las no processo, se quiser. A menos que você diga a ele de outra maneira com `declare`, suas variáveis são apenas um monte de letras para o bash. Mas quando você começar a tentar fazer aritmética com elas, o bash as converte para números inteiros, se puder. Isso torna possível fazer alguma aritmética relativamente complexa em bash.

Aritmética inteira pode ser feita usando o comando `let` predefinido ou por meio dos comandos externos `expr` ou `bc`. Depois de configurar o valor da variável `BIGNUM` como `1024`, todos os três comandos que se seguem armazenariam o valor `64` na variável `RESULT`. O comando `bc` é um aplicativo de calculadora que está disponível na maioria das distribuições Linux. O último comando recebe um número aleatório entre 0 e 10 e ecoa os resultados de volta para você.

```
BIGNUM=1024
let RESULT=$BIGNUM/16
RESULT='expr $BIGNUM / 16'
RESULT='echo "$BIGNUM / 16" | bc'
let foo=$RANDOM%10; echo $foo
```

Outra maneira de aumentar uma variável é usar a notação `$(())` com `++I` adicionados para incrementar o valor de `I`. Experimente digitar o seguinte:

```
$ I=0
$ echo The value of I after increment is $((++I))
$ echo The value of I before and after increment is $((I++)) and $I
```

Repita qualquer um desses comandos para continuar a aumentar o valor de `$I`.

ta

quanto a maioria dos elementos de scripts de shell são relativamente livres (em que os espaços em branco, como caracteres de espaço ou tabulações, são insignificantes), `let` e `expr` tratam o espaçamento de maneira peculiar. O comando `let` insiste em nenhum espaço entre cada operando e o operador matemático, enquanto a

xe do comando `expr` exige um espaço em branco entre cada operando e seu operador. Em oposição a estes, não é exigente com relação a espaços, mas pode ser mais complicado de usar, porque faz aritmética de ponto ante.

Para ver uma lista completa dos tipos de aritmética que você pode executar usando o comando `let`, digite **help let** no prompt do bash.

Usando construções de programação em scripts de Shell

Uma das características que torna os scripts de shell tão poderosos é que sua implementação de construções de loop e execução condicional é semelhante à encontrada nos mais complexos scripts e linguagens de programação. Você pode utilizar vários tipos de loops diferentes, dependendo de suas necessidades.

As instruções “*if...then*”

A construção de programação mais comumente usada é a execução condicional, ou a instrução `if`. Ela é usada para executar as seguintes ações apenas sob certas condições. Há diversas variações de `if` para testar vários tipos de condições.

O primeiro exemplo de `if...then` testa se `VARIABLE` está configurada como o número `1`. Se estiver, então o comando `echo` é usado para dizer que ela está configurada como `1`. A instrução `fi` então indica que a instrução `if` está completa e que o processamento pode continuar.

```
VARIABLE=1
if [ $VARIABLE -eq 1 ] ; then
echo "The variable is 1"
fi
```

Em vez de usar `-eq`, é possível usar o sinal de igualdade (`=`), como mostrado no exemplo a seguir. O `=` funciona melhor para a comparação de valores de string, enquanto `-eq` costuma ser melhor para a comparação de números. Usando a instrução `else`, palavras diferentes podem ser ecoadas se o critério da instrução `if` não for atendido (`$STRING = "Friday"`). Tenha em mente que é uma boa prática colocar strings entre aspas.

```
STRING="Friday"
if [ $STRING = "Friday" ] ; then
echo "WhooHoo. Friday."
else
echo "Will Friday ever get here?"
fi
```

Você também pode inverter os testes com um ponto de exclamação (`!`). No exemplo a seguir, se `STRING` não for `Monday`, então `"At least it's not Monday"` é ecoado.

```

STRING="FRIDAY"
if [ "$STRING" != "Monday" ] ; then
    echo "At least it's not Monday"
fi

```

No exemplo a seguir, `elif` (que significa “*else if*”) é usado para testar uma condição adicional (por exemplo, se `filename` é um arquivo ou um diretório).

```

filename="$HOME"
if [ -f "$filename" ] ; then
    echo "$filename is a regular file"
elif [ -d "$filename" ] ; then
    echo "$filename is a directory"
else
    echo "I have no idea what $filename is"
fi

```

Como você pode ver nos exemplos anteriores, a condição que você está testando é colocada entre colchetes []. Quando uma expressão de teste é avaliada, ela retorna um valor de 0, o que significa que é verdade, ou um 1, o que significa que ela é falsa. Observe que as linhas de eco são recuadas. O recuo é opcional e feito apenas para facilitar a leitura do script.

A Tabela 7.1 lista as condições que são testáveis e é uma referência muito útil. (Se estiver com pressa, você pode digitar **help test** na linha de comando para obter as mesmas informações.)

TABELA 7.1 Operadores para Expressões de Teste

Operador	O que está sendo testado?
arquivo	O arquivo existe? (o mesmo que <code>-e</code>)
arquivo	O arquivo é um dispositivo de bloco especial?
arquivo	O caractere de arquivo é especial (por exemplo, um dispositivo de caracteres)? Utilizado para identificar linhas seriais e dispositivos terminais.
arquivo	O arquivo é um diretório?
arquivo	O arquivo existe? (o mesmo que <code>-a</code>)
arquivo	O arquivo existe e é regular (por exemplo, não é um pipe, soquete, redirecionamento, arquivo de link ou dispositivo)?
arquivo	O arquivo tem o bit set-group-id configurado (SGID)?
arquivo	O arquivo é um link simbólico? (o mesmo que <code>-L</code>)
arquivo	O arquivo tem sticky bit configurado?
arquivo	O arquivo é um link simbólico?

string	O comprimento da string é maior do que 0 byte?
arquivo	Você possui o arquivo?
arquivo	O arquivo é um pipe nomeado?
arquivo	O arquivo é legível por você?
arquivo	O arquivo existe e ele é maior que 0 byte?
arquivo	O arquivo existe e ele é um soquete?
fd	O descritor de arquivo está conectado a um terminal?
arquivo	O arquivo tem o bit set-user-id (SUID) configurado?
arquivo	O arquivo é gravável por você?
arquivo	O arquivo é executável por você?
string	O tamanho da string é de 0 (zero) bytes?
pr1 -a expr2	A primeira e a segunda expressões são verdadeiras?
pr1 -o expr2	Qual das duas expressões é verdadeira?
quivol -nt arquivo2	O primeiro arquivo é mais novo do que o segundo (utilizando o registro de data/hora de modificação)?
quivol -ot arquivo2	O primeiro arquivo é mais velho do que o segundo (utilizando o registro de data/hora de modificação)?
quivol -ef arquivo2	Os dois arquivos estão associados por um link (um link físico [hard link], ou um link simbólico)?
r1 = var2	A primeira variável é igual à segunda variável ?
r1 -eq var2	A primeira variável é igual à segunda variável ?
r1 -ge var2	A primeira variável é maior do que ou igual à segunda variável?
r1 -gt var2	A primeira variável é maior do que a segunda variável?
r1 -le var2	A primeira variável é menor do que ou igual à segunda variável?
r1 -lt var2	A primeira variável é menor do que a segunda variável?
r1 != var2	A primeira variável é diferente da segunda variável?
r1 -ne var2	A primeira variável é diferente da segunda variável?

Há também um método especial abreviado para a realização de testes que podem ser úteis para *ações de um único comando*. No exemplo a seguir, as duas barras verticais (| |) indicam que, se o diretório não existir (-d dirname), o script deve criar o diretório (mkdir \$dirname).

```
# [ test ] || ação
# Executa um comando simples se o teste for falso
dirname="/tmp/testdir"
[ -d "$dirname" ] || mkdir "$dirname"
```

Em vez de barras verticais, você pode usar dois caracteres “E comercial” (&&) para testar se algo é verdade. No exemplo a seguir, um comando está sendo testado para ver se ele inclui pelo menos três argumentos de linha de comando.

```
# [ teste ] && {acção}
# Executa um comando simples se o teste for verdadeiro
[ $# -ge 3 ] && echo "There are at least 3 command line arguments."
```

Você pode combinar os operadores && e || para fazer um rápido *if-then-else* de uma linha. O seguinte exemplo testa se o diretório representado por \$dirname já existe. Se ele existir, uma mensagem diz que o diretório já existe. Se ele existir não, a instrução cria o diretório:

```
# dirname=meudiretorio
# [ -e $dirname ] && echo $dirname já existe || mkdir $dirname
```

O comando case

Outra construção bastante utilizada é o comando `case`. Semelhante a um `switch` em linguagens de programação, esse comando pode substituir várias instruções `if` aninhadas. Eis a forma geral da instrução `case`:

```
case "VAR" in
  Resultado1)
    { corpo };;
  Resultado2)
    { corpo };;
  *)
    { corpo } ;;
esac
```

Entre outras coisas, você pode usar o comando `case` para ajudar em seus backups. A instrução `case` a seguir testa as três primeiras letras do dia atual (`case `date +%a` in`). Então, dependendo do dia, um diretório de backup específico (BACKUP) e uma unidade de fita (TAPE) são configurados.

```

# Nossa VAR não precisa ser uma variável,
# também pode ser a saída de um comando
# Realiza uma ação com base no dia da semana
case `date +%a` in
    "Mon")
        BACKUP=/home/myproject/data0
        TAPE=/dev/rft0
# Note o uso de dois pontos e vírgulas para terminar cada opção
        ;;
# Note o uso do "|" para querer dizer "ou"
    "Tue" | "Thu")
        BACKUP=/home/myproject/data1
        TAPE=/dev/rft1
        ;;
    "Wed" | "Fri")
        BACKUP=/home/myproject/data2
        TAPE=/dev/rft2
        ;;
# Não faz backups no final da semana.
*)
    BACKUP="none"
    TAPE=/dev/null
    ;;
esac

```

O asterisco (*) é utilizado como um “pega tudo”, semelhante à palavra-chave `default` na linguagem de programação C. Neste exemplo, se nenhuma das outras entradas for encontrada até o final do loop, o asterisco é usado e o valor de `BACKUP` se torna `none`. Observe o uso de `esac`, ou `case`, de trás para frente, para terminar a instrução `case`.

O loop “for...do”

Loops são usados para executar ações repetidamente até que uma condição seja atendida ou até que todos os dados tenham sido processados. Um dos loops mais usados é o `for...do`. Ele itera através de uma lista de valores, executando o corpo do loop para cada elemento na lista. A sintaxe e alguns exemplos são aqui apresentados:

```

for VAR in LISTA
do
    { corpo }
done

```

O loop `for` atribui os valores em `LIST` a `VAR`, um de cada vez. Então, para cada valor, o corpo em chaves entre `do` e `done` é executado. `VAR` pode ser qualquer nome de variável e `LIST` pode ser composta de praticamente qualquer lista de valores ou qualquer coisa que gere uma lista.

```

for NUMBER in 0 1 2 3 4 5 6 7 8 9
do
    echo O número é $NUMBER
done

for FILE in `/bin/ls`
do
    echo $FILE
done

```

Você também pode escrever dessa maneira, que é um pouco mais limpa:

```
for NAME in John Paul Ringo George ; do
echo $NAME é meu Beatle favorito
done
```

Cada elemento de `LIST` é separado do seguinte por espaços em branco. Isso pode causar problemas se você não tiver cuidado, pois alguns comandos, como `ls -l`, geram saída de múltiplos campos por linha, cada um separado por um espaço em branco. A string `done` termina a instrução `for`.

Se você é um fiel programador em C, o `bash` permite que use a sintaxe C para controlar seus loops:

```
LIMIT=10
# Parênteses duplos, e nenhum $ em LIMIT mesmo sendo uma variável!
for ((a=1; a <= LIMIT ; a++)) ; do
echo "$a"
done
```

Os loops “while...do” e “until...do”

Duas outras possíveis construções de iteração são o loop `while...do` e o loop `until...do`. A estrutura de cada uma é apresentada aqui:

```
while condição          until condição
do                      do
  { corpo }             { corpo }
done                   done
```

A instrução `while` executa enquanto a condição é verdadeira. A instrução `until` executa até que a condição seja verdade — em outras palavras, enquanto a condição é falsa.

Eis um exemplo de um loop `while` que irá gerar a saída do número 0123456789:

```
N=0
while [ $N -lt 10 ] ; do
echo -n $N
let N=$N+1
done
```

Outra maneira de produzir o número 0123456789 é usar um loop `until` como a seguir:

```
N=0
until [ $N -eq 10 ] ; do
echo -n $N
let N=$N+1
done
```

Experimentando alguns programas úteis de manipulação

de texto

O bash é grande e tem muitos comandos predefinidos, mas geralmente precisa de alguma ajuda para fazer algo realmente útil. Alguns dos programas comuns mais úteis que você verá sendo utilizados são `grep`, `cut`, `tr`, `awk` e `sed`. Tal como acontece com todas as melhores ferramentas do UNIX, a maioria desses programas é projetada para trabalhar com a entrada padrão e a saída padrão, de modo que você pode facilmente utilizá-los com redirecionamentos e scripts de shell.

O `general regular expression parser`

O nome *general regular expression parser* (`grep`) – *analisador de expressão geral regular* soa intimidante, mas `grep` é apenas uma maneira de localizar padrões em arquivos ou texto. Pense nisso como uma ferramenta de pesquisa útil. Ganhar experiência com expressões regulares é um grande desafio, mas depois de dominar o assunto, você pode fazer muitas coisas úteis com até as formas mais simples.

Por exemplo, você pode exibir uma lista de todas as contas de usuários regulares usando `grep` para pesquisar todas as linhas que contêm o texto `/home` no arquivo `/etc/passwd`, como a seguir:

```
$ grep /home /etc/passwd
```

Ou você pode localizar todas as variáveis de ambiente que começam com `HO` usando o seguinte comando:

```
$ env | grep ^HO
```

ta

tal `^` no código anterior é o caractere de circunflexo real, não o que você vai ver comumente para um backspace, Digite `^`, `H` e `O` (a letra maiúscula) para ver os itens que começam com as letras maiúsculas `HO`.

A fim de localizar uma lista de opções para usar com o comando `grep`, digite `man grep`.

Remova seções de linhas de texto (`cut`)

O comando `cut` pode extrair campos de uma linha de texto ou de arquivos e é muito útil para analisar arquivos de configuração do sistema em pedaços fáceis de digerir. Você pode especificar o separador de campo que pretende utilizar e os campos que você quer, ou pode quebrar uma linha com base em bytes.

O exemplo a seguir lista todos os diretórios de usuários em seu sistema. Essa linha de comando `grep` redireciona uma lista de usuários regulares do arquivo `/etc/passwd` e depois exibe o sexto campo (`-f6`) delimitado com um caractere de dois-pontos (`-d' : '`). O hífen no final instrui `cut` a ler a partir da entrada padrão (a partir do redirecionamento).

```
$ grep /home /etc/passwd | cut -d' :' -f6 -
```

Traduza ou exclua caracteres (tr)

O comando `tr` é um tradutor baseado em caracteres que pode ser usado para substituir um caractere ou conjunto de caracteres por outro ou para remover um caractere de uma linha de texto.

O exemplo a seguir traduz todas as letras maiúsculas para letras minúsculas e exibe as palavras `mixed upper and lower case` como resultado:

```
$ FOO="Mixed UPpEr aNd LoWeR cAsE"  
$ echo $FOO | tr [A-Z] [a-z]  
mixed upper and lower case
```

No próximo exemplo, o comando `tr` é usado em uma lista de nomes de arquivos para renomear os arquivos na lista de modo que quaisquer tabulações ou espaços (como indicado pela opção `[:blank:]`) contidos em um arquivo são traduzidos em sublinhados. Tente executar o seguinte código em um diretório de teste:

```
for file in * ; do  
f='echo $file | tr [:blank:] [_]'  
[ "$file" = "$f" ] || mv -i -- "$file" "$f"  
done
```

O editor de fluxo (sed)

O comando `sed` é um editor simples de script e, portanto, pode realizar apenas edições simples, como remover linhas de texto que correspondem a um determinado padrão, substituir um padrão de caracteres por outro e assim por diante. Para se ter uma melhor ideia de como scripts `sed` funcionam, não há substituto para a documentação online, mas aqui estão alguns exemplos de usos comuns.

Você pode usar o comando `sed` para fazer basicamente o que eu fiz anteriormente com o exemplo `grep`: pesquise no arquivo `/etc/passwd` a palavra `home`. Aqui, o comando `sed` varre todo o arquivo `/etc/passwd` `home`, pesquisa a palavra `home` e imprime qualquer linha que contém a palavra `home`.

```
$ sed -n '/home/p' /etc/passwd
```

Neste exemplo, `sed` pesquisa o arquivo `somefile.txt` e substitui cada ocorrência da string `Mac` por `Linux`. Observe que a letra `g` é necessária no fim do comando para fazer com que a substituição de todas as ocorrências de `Mac` em cada linha sejam alteradas para `Linux`. (Caso contrário, apenas a primeira ocorrência de `Mac` em cada linha é alterada.) A saída é, então, enviada para o arquivo `fixed_file.txt`. A saída de `sed` vai para `stdout` e, em seguida, esse comando redireciona a saída para um arquivo por segurança.

```
$ sed 's/Mac/Linux/g' somefile.txt > fixed_file.txt
```

Você pode obter o mesmo resultado usando um redirecionamento:


```
$ cat somefile.txt | sed 's/Mac/Linux/g' > fixed_file.txt
```

Procurando por um padrão e substituindo-o por um padrão nulo, você exclui o padrão original. Este exemplo examina os conteúdos do arquivo `somefile.txt` e substitui espaços em branco extras no final de cada linha (`s/ *$`) por nada (`//`). Os resultados vão para o arquivo `fixed_file.txt`.

```
$ cat somefile.txt | sed 's/ *$//' > fixed_file.txt
```

Usando scripts de shell simples

Às vezes o script mais simples pode ser o mais útil. Se você digitar a mesma sequência de comandos repetidamente, faz sentido armazenar esses comandos (uma vez!) em um arquivo. As seções a seguir oferecem um par de simples, mas úteis, scripts de shell.

Lista telefônica

Essa ideia foi transmitida de geração em geração de antigos hacks do UNIX. É realmente muito simples, mas emprega vários dos conceitos que acabamos de apresentar.

```
#!/bin/bash
# (@)/ph
# A very simple phonebook
# Type "ph new name number" to add to the list, or
# simply type "ph name" to obtain a telephone number
PHONELIST=~/.phonenumber.txt

# If there is no command line parameter ($#), there is
# a problem, so ask about what they are talking.
if [ $# -lt 1 ] ; then
    echo "What phone number do you want? "
    exit 1
fi

# Do you want to add a new phone number?
if [ $1 = "new" ] ; then
    shift
```

```

echo $* >> $PHONELIST
echo $* added to database
exit 0
fi

# No. But does the file still have something in it?
# This may be the first time that we use it, after all.
if [ ! -s $PHONELIST ] ; then
    echo "No number in the phone book yet! "
    exit 1
else
    grep -i -q "$*" $PHONELIST    # Search the file silently
    if [ $? -ne 0 ] ; then        # Did we find something?
        echo "Sorry, but that name was not found in the list."
        exit 1
    else
        grep -i "$*" $PHONELIST
    fi
fi
fi
exit 0

```

Então, se você criou o arquivo de lista telefônica como `ph` em seu diretório atual, pode digitar o seguinte a partir do shell para testar seu script `ph`:

```

$ chmod 755 ph

$ ./ph new "Mary Jones" 608-555-1212
Mary Jones 608-555-1212 added to database

$ ./ph Mary
Mary Jones 608-555-1212

```

O comando `chmod` torna o script `ph` executável. O comando `./ph` executa o comando `ph` a partir do diretório atual com a opção `new`. Isso acrescenta Mary Jones como o nome e 608-555-1212 como o número de telefone ao banco de dados (`$HOME/.phone.txt`). O comando `ph` em seguida pesquisa no banco de dados o nome de Mary e mostra a entrada de telefone para Mary. Se o script funcionar, adicione-o a um diretório em seu caminho (como `$HOME/bin`).

Script de backup

Como nada funciona para sempre e erros acontecem, os backups são apenas um fato da vida quando se trata de dados de computador. Esse simples script faz backup de todos os dados nos diretórios iniciais de todos os usuários de seu sistema Fedora ou RHEL.

```

#!/bin/bash
# (@)/my_backup
# Um script de backup muito simples
#
#
Change the TAPE device to match your system.
# Check /var/log/messages to determine your tape device.
# You may also need to add support scsi-tape to your kernel.
TAPE=/dev/rft0
#
Rewind the $TAPE tape device

```

```
mt $TAPE rew
# Get a list of home directories
HOMES='grep /home /etc/passwd | cut -f6 -d' : '
# Back up the data in this directory
tar cvf $TAPE $HOMES
# Rewind and eject the tape.
mt $TAPE rewoffl
```

Resumo

Escrever scripts de shell lhe dá a oportunidade de automatizar muitas de suas tarefas mais comuns de administração de sistema. Este capítulo cobriu comandos e funções comuns que você pode usar em scripts com o shell bash. Ele também forneceu alguns exemplos concretos de scripts para fazer backups e outros procedimentos.

No próximo capítulo, você deixa os recursos do usuário e passa a examinar tópicos relacionados com a administração do sistema. O Capítulo 8 aborda como se tornar o usuário root e como usar comandos administrativos, monitorar arquivos de log e trabalhar com arquivos de configuração.

Exercícios

Use estes exercícios para testar seus conhecimentos em escrever scripts de shell simples. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostradas no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. **Crie um script no seu diretório `$HOME/bin` chamado `myownscript`. Quando o script é executado, ele deve apresentar informações que se parecem com o seguinte:**

```
Today is Sat Dec 10 15:45:04 EST 2011.
```

```
You are in /home/joe and your host is abc.example.com.
```

Naturalmente, você precisa ler a data/hora atual, seu diretório de trabalho atual e seu hostname. Além disso, inclua comentários sobre o que o script faz e indique que o script deve ser executado com o shell `/bin/bash`.

2. **Crie um script que lê os três parâmetros posicionais na linha de comando, atribui os parâmetros aos nomes de variável `ONE`, `TWO` e `THREE`, respectivamente, e, então, gera essa informação no seguinte formato:**

```
There are X parameters that include Y.
```

```
The first is A, the second is B, the third is C.
```

Substitua `X` pelo número de parâmetros e `Y` por todos os parâmetros inseridos. Então, substitua `A` pelo conteúdo da variável `ONE`, `B` pela variável `TWO` e `C` pela variável `THREE`.

3. Crie um script que solicita aos usuários o nome da rua e da cidade em que eles cresceram. Atribua cidade e rua às variáveis chamadas `mytown` e `mystreet` e envie-as com uma frase que exiba (naturalmente, `$mystreet` e `$mytown` aparecerão como a cidade e a rua reais que o usuário digitar):

```
The street I grew up on was $mystreet and the town was $mytown
```

4. Crie um roteiro chamado `myos` que pergunta ao usuário, “Qual é seu sistema operacional favorito?” Envie para a saída uma frase insultante se o usuário digitar Windows ou Mac. Responda “Ótima escolha!” se o usuário digitar Linux. Para qualquer outra coisa diga “É <o que foi digitado> um sistema operacional?”
5. Crie um script que itera pelas palavras `alce`, `vaca`, `ganso` e `porco`, usando um loop `for`. Exiba cada uma dessas palavras no final da linha de “Tenho um... .”

Parte III

Tornando-se um administrador de sistema Linux

NESTA PARTE

Capítulo 8

Aprendendo administração de sistema

Capítulo 9

Instalando o Linux

Capítulo 10

Obtendo e gerenciando software

Capítulo 11

Gerenciando contas de usuário

Capítulo 12

Gerenciando discos e sistemas de arquivos

Aprendendo administração de sistema

NESTE CAPÍTULO

Fazendo administração gráfica

Utilizando o login de root

Entendendo comandos administrativos, arquivos de configuração e arquivos de log

Trabalhando com dispositivos e sistemas de arquivos

O Linux, como outros sistemas baseados em UNIX, foi concebido para ser utilizado por mais de uma pessoa ao mesmo tempo. Os recursos *multiusuário* permitem que muitas pessoas tenham contas em um único sistema Linux, com seus dados mantidos em segurança, longe de outros. A *multitarefa* permite que muitas pessoas executem vários programas no computador ao mesmo tempo, com cada pessoa sendo capaz de executar mais de um programa. Protocolos de rede sofisticados e aplicativos tornam possível para um sistema Linux estender seus recursos aos usuários e computadores em rede ao redor do mundo. A pessoa designada para gerenciar todos os recursos de um sistema Linux é chamada de *administrador do sistema*.

Mesmo que você seja a única pessoa a utilizar um sistema Linux, a administração do sistema ainda está configurada para ser separada do uso do computador. Para fazer a maioria das tarefas administrativas, você precisa estar conectado como *usuário root* (também chamado de *superusuário*) ou temporariamente obter a permissão de root. Os usuários comuns que não têm permissão de root não podem mudar ou, em alguns casos, nem mesmo ver, algumas das informações de configuração de um sistema Linux. Em particular, recursos de segurança como senhas armazenadas são protegidos da visão geral.

Como a administração do sistema Linux é um assunto extenso, este capítulo foca apenas os princípios gerais. Em particular, ele examina algumas das ferramentas básicas que você precisa para administrar um sistema Linux de um computador pessoal ou de um servidor de pequeno porte. Além do básico, este capítulo também ensina como trabalhar com sistemas de arquivos e monitorar a configuração e desempenho de seu sistema Linux.

Entendendo a administração do sistema

Separar o papel do administrador de sistema do papel de outros usuários tem vários efeitos. Para um sistema que é usado por muitas pessoas, limitar quem pode gerenciá-lo permite que você mantenha-o mais seguro. Um papel administrativo separado também impede que outras pessoas danifiquem acidentalmente seu sistema quando estão apenas usando-o para escrever um documento ou navegar na internet.

Se você é o administrador de um sistema Linux, geralmente faz login com uma conta de usuário regular e, então, solicita privilégios administrativos quando precisa deles. Isso geralmente é feito com uma das seguintes opções:

- **Comando `su`** — Muitas vezes `su` é usado para abrir um shell como usuário root. Com esse shell aberto, o administrador pode executar vários comandos e depois sair para voltar para um shell como um usuário regular.
- **Comando `sudo`** — Com `sudo`, um usuário regular tem privilégios de root, mas apenas quando executa o comando `sudo`. Depois de executar esse comando com `sudo`, o usuário é imediatamente retornado a um shell e estará atuando como o usuário regular novamente.
- **Janelas gráficas** — Muitas janelas gráficas de administração, as quais podem ser iniciadas a partir do menu System ou Applications, podem também ser iniciadas por um usuário regular. Quando privilégios de root são necessários, a senha de root será solicitada.

Tarefas que podem ser feitas apenas pelo usuário root tendem a ser aquelas que afetam o sistema como um todo ou sua segurança ou saúde. A seguir, apresentamos uma lista de características comuns que um administrador de sistema deverá gerenciar:

- **Sistemas de arquivos** — Quando você instala o Linux, a estrutura de diretórios está configurada para tornar o sistema utilizável. Mas se mais tarde você quiser adicionar armazenamento extra ou mudar o layout do sistema de arquivos, precisa de privilégios administrativos para fazer isso. Além disso, o usuário root tem permissão para acessar os arquivos de qualquer usuário. Como resultado, o usuário root pode copiar, mover ou alterar arquivos de qualquer outro usuário — um privilégio necessário para fazer cópias de backup do sistema de arquivos por questões de segurança.
- **Instalação de software** — Como os softwares maliciosos podem danificar seu sistema ou torná-lo inseguro, você precisa ter privilégios de root para instalar softwares de modo que estejam disponíveis para todos os usuários de seu sistema. Os usuários regulares ainda podem instalar algum software em seus próprios diretórios e podem listar informações sobre os softwares de sistema instalados.
- **Contas de usuário** — Somente o usuário root pode adicionar e remover contas de usuário e de grupo.
- **Interfaces de rede** — Costumava ser totalmente de responsabilidade do usuário root configurar interfaces de rede, bem como iniciar e parar essas interfaces. Agora, muitos desktops Linux permitem que usuários regulares iniciem e parem interfaces de rede a partir de sua área de trabalho usando o Network Manager.
- **Servidores** — Configurar servidores web, de arquivos, de nome de domínio, de e-mail e dezenas

de outros servidores requer privilégios de root, assim como iniciar e parar os serviços. Muitas vezes, serviços executados como usuários não root e conteúdo, como páginas da web, podem ser adicionados a servidores por usuários não-root se você configurar seu sistema para permitir isso.

- **Recursos de segurança** — A configuração de recursos de segurança, como firewalls e listas de acesso de usuário, geralmente é feita pelo usuário root. Também é de responsabilidade do usuário root monitorar como os serviços estão sendo usados e se certificar de que os recursos do servidor não se esgotem nem sejam abusados.

A maneira mais fácil de começar a administração do sistema é usando algumas ferramentas de administração gráfica.

Usando ferramentas de administração gráfica

Muitos sistemas Linux vêm com ferramentas gráficas simplificadas para administração. Se você é um usuário casual, essas ferramentas muitas vezes permitem que você faça tudo o que precisa para administrar seu sistema sem editar arquivos de configuração ou executar comandos de shell.

Um conjunto de ferramentas gráficas que vem com os sistemas Fedora e Red Hat Enterprise Linux pode ser carregado a partir do submenu Administration do menu System ou a partir da linha de comando. A maioria das ferramentas do Fedora e do RHEL que são carregadas a partir da linha de comando começa com a string `system-config` (como `system-config-network`).

Essas ferramentas `system-config` requerem permissão de root. Se estiver conectado como um usuário comum, você deve digitar a senha de root antes de a janela do aplicativo de interface gráfica do usuário (GUI) iniciar ou, como em certos casos, quando você solicitar para fazer alguma atividade especial.

Uma vez que você digitou a senha, procure um conjunto de chaves (RHEL 5), como ilustrado na Figura 8.1, ou um ícone de crachá amarelo (RHEL 6) no lado superior direito do painel, indicando que você tem autorização de root. Clique no ícone para remover a autorização. Caso contrário, a autorização desaparecerá após alguns minutos. Embora o ícone de chaves ou crachá seja exibido, você pode abrir qualquer aplicativo GUI administrativo sem ter de digitar a senha novamente.

cone de chaves aparece no painel superior, enquanto o privilégio está aberto para ferramentas de trabalho administrativas.

A lista a seguir descreve muitas das ferramentas gráficas que você pode usar para administrar um sistema Fedora ou Red Hat Enterprise Linux (alguns existem apenas no Fedora). Inicie essas janelas a partir do submenu Administration no menu System. O nome do pacote que deverá ser instalado para obter o recurso é mostrado entre parênteses. As seguintes ferramentas gráficas estão disponíveis no Fedora:

- **Domain Name System** (`system-config-bind`) — Cria e configura zonas se seu computador estiver atuando como um servidor DNS.
- **HTTP** (`system-config-httpd`) — Configura seu computador como um servidor Web Apache.
- **NFS** (`system-config-nfs`) — Configura diretórios de seu sistema para serem compartilhados com outros computadores em sua rede, utilizando o serviço NFS.
- **Bootloader** (`system-config-boot`) — Se tiver vários sistemas operacionais em seu computador, ou múltiplos kernels Linux disponíveis para inicializar no Linux, você pode usar a tela Boot Configuration para escolher qual inicializar por padrão. Por exemplo, é possível ter um Fedora, um SUSE e um Windows XP, todos no mesmo disco rígido. Você pode escolher qual iniciará automaticamente (depois de um determinado número de segundos), se nenhum for selecionado explicitamente.
- **Root Password** (`system-config-rootpassword`) — Altera a senha de root.
- **Display** (`system-config-display`) — Altera as configurações do desktop do seu X Window System, incluindo cor e resolução para seu monitor. Você também pode escolher as configurações da placa de vídeo e do monitor.
- **Samba NFS** (`system-config-samba`) — Configura o compartilhamento de arquivos Windows (SMB). (Para configurar outros recursos do Samba, você pode usar a janela SWAT).

As seguintes ferramentas gráficas estão disponíveis no Fedora e no Red Hat Enterprise Linux:

- **Services** (`system-config-services`) — Exiba e altere os serviços que estão em execução em seu sistema Fedora em diferentes níveis de execução da janela Service Configuration.
- **Add/Remove Software** (`PackageKit`) — Inicie a janela Add/Remove Software para encontrar, adicionar e remover software associado a repositórios de software configurados para seu sistema.
- **Authentication** (`authconfig-gtk`) — Altere a forma como os usuários são autenticados no sistema. Normalmente, Shadow Passwords (senhas de sombra) e MD5 Passwords (senhas MD5)

estão selecionadas. Mas se a rede suporta autenticação LDAP, Kerberos, SMB, NIS ou Hesiod, você pode selecionar para usar qualquer um desses tipos de autenticação.

- **Date & Time** (`system-config-date`) — Defina a data e a hora ou escolha ter um servidor NTP para manter a data/hora do sistema em sincronia.
- **Firewall** (`system-config-firewall`) — Configure o firewall para permitir ou negar serviços para computadores da rede.
- **Language** (`system-config-language`) — Selecione o idioma padrão usado para o sistema.
- **Logical Volume Management** (`system-config-lvm`) — Gerencie suas partições LVM.
- **Network** (`system-config-network`) — Gerencie suas interfaces de rede atuais e adicione interfaces.
- **Printing** (`system-config-printer`) — Configure impressoras locais e de rede.
- **SELinux Management** (`policycoreutils-gui`) — Defina os modos de imposição de segurança e a política padrão do SELinux.
- **Users & Groups** (`system-config-users`) — Adicione, visualize e altere contas de usuário e de grupo do seu sistema Fedora.

Outros utilitários administrativos estão disponíveis a partir do menu Applications no painel superior. Selecione o submenu System Tools para ver algumas das seguintes opções:

- **Configuration Editor** (`gconf-editor`) — Edite diretamente o banco de dados de configuração do GNOME.
- **Disk Usage Analyzer** (`gnome-utils`) — Exiba informações detalhadas sobre seus discos rígidos e dispositivos de armazenamento removíveis.
- **Disk Utility** (`gnome-disk-utility`) — Gerencie partições de disco e adicione sistemas de arquivos.
- **Kickstart** (`system-config-kickstart`) — Crie um arquivo de configuração de inicialização rápida que pode ser usado para instalar vários sistemas Linux sem interação do usuário.

Conforme você estudar o resto deste livro para configurar vários servidores Linux, descreverei como usar muitas dessas ferramentas. Quando quiser ir além de uma interface administrativa de apontar e clicar, você terá de aprender a obter privilégios de root a partir do shell, como descrito na próxima seção.

Usando a conta do usuário root

Todo sistema Linux começa com pelo menos uma conta de usuário administrativo (root) e, possivelmente,

uma ou mais contas de usuários regulares (que recebem um nome que você escolhe, ou um nome atribuído pela sua distribuição Linux). Na maioria dos casos, você faz login como um usuário comum e torna-se o usuário root para fazer uma tarefa administrativa.

O usuário root tem controle completo sobre a operação de seu sistema Linux. Esse usuário pode abrir qualquer arquivo ou executar qualquer programa. O usuário root também instala pacotes de software e adiciona contas de outras pessoas que usam o sistema.

ca

se no usuário root no Linux como similar ao usuário Administrador no Windows.

Ao instalar a maioria dos sistemas Linux (embora nem todos os sistemas), você adiciona uma senha ao usuário root. Você deve se lembrar e proteger essa senha — é necessário fazer login como root ou obter a permissão de root, enquanto você está conectado como outro usuário.

Para se familiarizar com a conta de usuário root, você pode simplesmente efetuar login como o usuário root. Recomendo tentar isso a partir de um console virtual. Para fazer isso, pressione Ctrl+Alt+F2. Quando você vir o prompt de login, digite **root** (pressione Enter) e digite a senha. Isso abrirá uma sessão de login para o root. Quando você terminar, digite **exit** e pressione Ctrl+Alt+F1 para voltar ao login de desktop regular.

Depois de você ter se conectado como root, o diretório inicial do usuário root é geralmente `/root` e ele junto com outras informações associadas com a conta de usuário root estão localizados no arquivo `/etc/passwd`. Eis como aparece a entrada de root no arquivo `/etc/passwd`:

```
root:x:0:0:root:/root:/bin/bash
```

Isso mostra que para o usuário chamado root, o ID do usuário é definido como 0 (root), o ID do grupo é definido como 0 (grupo root), o diretório inicial é `/root` e o shell para o usuário é `/bin/bash`. (O Linux usa o arquivo `/etc/shadow` para armazenar dados de senha criptografados, de modo que o campo de senha aqui contém um `x`.) Você pode alterar o diretório inicial ou o shell usado, editando os valores nesse arquivo. A melhor maneira de alterar esses valores, porém, é usar o comando `usermod` (consulte a seção “Modificando usuários com `usermod`”, no Capítulo 11, para obter mais informações).

Nesse ponto, qualquer comando que você executar a partir de seu shell será executado com privilégios de root. Portanto, tenha cuidado. Você tem muito mais poder para alterar (e danificar) o sistema do que você tem como um usuário regular. Novamente, digite **exit** quando terminar e, se estiver em um console virtual e tiver uma interface de desktop rodando em outro console, pressione Ctrl+Alt+F1 para voltar para a tela de login gráfica, se você estiver usando um sistema de desktop Linux.

ta

padrão, a conta root é desabilitada no Ubuntu. Isso significa que, mesmo que a conta exista, você não pode fazer login para usá-la ou usar `su` para se tornar o usuário root. Isso acrescenta um nível adicional de segurança ao Ubuntu e exige que você use o `sudo` antes de cada comando que deseja executar como usuário root.

Tornando-se root a partir do shell (comando su)

Embora você possa se tornar o superusuário efetuando login como root, às vezes isso não é conveniente. Por exemplo, você pode estar conectado a uma conta de usuário regular e só quer fazer uma rápida mudança administrativa em seu sistema sem precisar fazer logout e voltar a fazer login. Ou, você pode precisar fazer login na rede a fim de fazer uma mudança em um sistema Linux, mas achar que o sistema não permite que usuários root efetuem login a partir da rede (uma prática comum para sistemas Linux altamente seguros).

A solução é usar o comando `su`. A partir de qualquer janela de terminal ou shell, você pode simplesmente digitar o seguinte:

```
$ su
Password: *****
#
```

Quando for solicitado, digite a senha do usuário root. O prompt para o usuário regular (\$) muda para o prompt de superusuário (#). Nesse ponto, você tem permissão completa para executar qualquer comando e utilizar qualquer arquivo no sistema. Mas uma coisa que o comando `su` não faz quando usado dessa forma é ler o ambiente do usuário root. Como resultado, você pode digitar um comando que sabe que está disponível e obter a mensagem `Command Not Found` (comando não encontrado). Para corrigir esse problema, use o comando `su` com a opção traço (-), como a seguir:

```
$ su -
Password: *****
#
```

Você ainda precisará digitar a senha, mas depois disso, tudo o que acontece normalmente no login do usuário root acontece depois de o comando `su` ser concluído. Seu diretório atual será o diretório inicial do usuário root (provavelmente `/root`) e coisas como a variável `PATH` do usuário root serão usadas. Se você se tornar o usuário root simplesmente digitando `su`, em vez de `su -`, não vai poder mudar de diretório nem alterar o ambiente da sessão de login atual.

Você também pode usar o comando `su` para se tornar um usuário diferente de root. Isso é útil para solucionar um problema que está sendo experimentado por um usuário em particular, mas não por outros no computador (como uma incapacidade de imprimir ou enviar e-mail). Por exemplo, para ter as permissões de um usuário chamado `jsmith`, você digita o seguinte:

```
$ su - jsmith
```

Mesmo que você fosse o usuário root antes de digitar esse comando, depois disso você teria somente as permissões para abrir arquivos e executar programas que estão disponíveis para `jsmith`. Como usuário root, porém, depois de digitar o comando `su` para se tornar outro usuário, você não precisa de uma senha para continuar. Se digitar o comando como um usuário comum, você deve digitar a senha do novo

usuário.

Quando terminar de usar permissões de superusuário, volte para o shell anterior saindo do shell atual. Faça isso pressionando Ctrl+D ou digitando **exit**. Se você é o administrador de um computador que é acessível a vários usuários, não deixe um shell de root aberto na tela de outra pessoa — a menos que você queira dar a ela liberdade de fazer qualquer coisa que quiser no computador!

Permitindo acesso administrativo por meio da interface gráfica (GUI)

Como mencionado anteriormente, quando você executa ferramentas gráficas (GUI) como um usuário regular (a partir do Fedora, do Red Hat Enterprise Linux ou de alguns outros sistemas Linux), você será solicitado a fornecer a senha de root antes de ser capaz de acessar a ferramenta. Ao inserir a senha de root, você tem privilégios de administrador para essa tarefa. Nos casos do Red Hat Enterprise Linux e do Fedora, depois de digitar a senha, um ícone de crachá amarelo aparece no painel superior, indicando que a autorização de root ainda está disponível para outras ferramentas GUI executarem a partir dessa sessão de desktop.

Ganhando acesso administrativo com sudo

Um usuário especial também pode receber permissões administrativas para tarefas específicas sem receber a senha de root. O recurso `sudoers` é a forma mais comum de fornecer tal privilégio. Usando `sudoers` para quaisquer usuários ou grupos no sistema, você pode:

- Atribuir privilégios de root a qualquer comando que eles executem com `sudo`.
- Atribuir privilégios de administrador para um seletor conjunto de comandos.
- Dar aos usuários privilégios de root sem dizer-lhes a senha de root, porque eles só têm de fornecer sua própria senha de usuário para obter privilégios de root.
- Permitir que usuários, se você preferir, executem `sudo` sem digitar nenhuma senha.
- Controlar quais usuários executam comandos administrativos em seu sistema. (Usando `su`, tudo que você sabe é que alguém com a senha de root efetuou login, enquanto o comando `sudo` registra em log o usuário que executa o comando administrativo.)

Com o recurso `sudoers`, dar privilégios de root completos ou limitados a qualquer usuário requer simplesmente que o usuário seja adicionado ao arquivo `/etc/sudoers` e que seja definido qual privilégio você quer que ele tenha. Então, o usuário pode executar qualquer comando que tiver o privilégio de usar, precedendo-o com o comando `sudo`.

Eis um exemplo de como usar o recurso `sudo` para fazer com que o usuário chamado `joe` tenha plenos privilégios de root.

ca

Examinar o arquivo `sudoers` no Ubuntu, você verá que o usuário inicial no sistema já tem privilégios, por padrão,

enientes do fato de ser um membro do grupo `admin`. Para dar a qualquer outro usuário os mesmos privilégios, pode simplesmente acrescentar o usuário adicional ao grupo `admin` quando você executa `visudo`.

1. Como usuário `root`, edite o arquivo `/etc/sudoers` `visudo` com o comando `visudo`:

```
# /usr/sbin/visudo
```

Por padrão, o arquivo é aberto no `vi`, a não ser que a variável `EDITOR` esteja configurada como algum outro editor aceitável para `visudo` (por exemplo, `export EDITOR=gedit`). A razão para usar `visudo` é que o comando bloqueia o arquivo `/etc/sudoers` e faz uma verificação de sanidade básica nele para assegurar de que foi editado corretamente.

ta

Se você empacar aqui, tente executar o comando `vimtutor` para um rápido tutorial sobre como usar `vi` e `vim`.

2. Adicione a seguinte linha para permitir que `joe` tenha plenos privilégios de `root` sobre o computador:

```
joe    ALL=(ALL)    ALL
```

Essa linha faz com que `joe` forneça uma senha (sua própria senha, não a senha de `root`) para usar comandos administrativos. Para permitir que `joe` tenha esse privilégio sem o uso de uma senha, digite a seguinte linha no lugar:

```
joe    ALL=(ALL)    NOPASSWD: ALL
```

3. Salve as alterações no arquivo `/etc/sudoers` (no `vi`, digite **Esc** e então **:wq**).

O exemplo a seguir é de uma sessão do usuário `jake` depois que ele ganhou privilégios de `sudo`

```
[joe]$ sudo touch /mnt/testfile.txt
We trust you have received the usual lecture
from the local System Administrator. It usually
boils down to these two things:
#1) Respect the privacy of others.
#2) Think before you type.

Password: *****
[joe]$ ls -l /mnt/testfile.txt
-rw-r--r--. 1 root root 0 Jan 7 08:42 /mnt/testfile.txt
[jake]$ rm /mnt/testfile.txt
rm: cannot remove '/mnt/testfile.txt': Permission denied
[jake]$ sudo rm /mnt/textfile.txt
[jake]$
```

Nessa sessão, o usuário `jake` executa o comando `sudo` para criar um arquivo (`/mnt/textfile.txt`) em um diretório no qual ele não tem permissão de gravação. Ele recebe um aviso e é solicitado a fornecer sua senha (essa é a senha de `jake`, não a senha de `root`).

Mesmo depois que jake forneceu a senha, ele ainda deve usar o comando `sudo` para executar comandos administrativos subsequentes como `root` (o `rm` falha, mas o `sudo rm` funciona). Note que ele não é solicitado a fornecer uma senha para o segundo `sudo`. Isso ocorre porque depois de inserir sua senha com sucesso, ele pode inserir quantos comandos `sudo` quiser pelos próximos cinco minutos sem ter de digitá-la novamente. (Você pode alterar o valor do tempo limite para quanto quiser, definindo o valor `passwd_timeout` no arquivo `/etc/sudoers`.)

O exemplo anterior concede um simples privilégio administrativo do tipo “tudo ou nada” a joe. Mas o arquivo `/etc/sudoers` lhe dá uma quantidade incrível de flexibilidade ao permitir que usuários individuais e grupos utilizem aplicativos individuais ou grupos de aplicativos. Consulte as páginas `man` de `sudoers` e `sudo` para obter informações sobre como ajustar seu recurso `sudo`.

Explorando comandos administrativos, arquivos de configuração e arquivos de log

Você pode esperar encontrar muitos comandos, arquivos de configuração e arquivos de log nos mesmos lugares do sistema de arquivos, independentemente de qual distribuição Linux você está usando. As seções a seguir fornecem algumas indicações sobre onde procurar esses elementos importantes.

ta

erramentas administrativas gráficas para o Linux tomaram-se tão boas, por que você precisa saber sobre os arquivos administrativos? Por um lado, embora as ferramentas GUI difiram entre versões do Linux, muitos arquivos de configuração subjacentes são os mesmos. Portanto, se aprender a trabalhar com eles, você pode trabalhar com qualquer sistema Linux. Além disso, se um recurso quebrar ou se você precisa fazer algo que não é suportado pela interface gráfica, quando pedir ajuda, especialistas em Linux quase sempre lhe dizem como alterar o arquivo de configuração diretamente.

Comandos administrativos

Muitos comandos administrativos foram concebidos para serem usados somente pelo usuário `root`. Quando você efetua login como `root` (ou usa `su -` a partir do shell para se tornar `root`), sua variável `$PATH` está definida para incluir alguns diretórios que contêm comandos para o usuário `root`. Elas incluem o seguinte:

- `/sbin` — Contém comandos necessários para iniciar seu sistema, incluindo comandos para verificar sistemas de arquivos (`fsck`) e iniciar/parar serviços do sistema (`service`).
- `/usr/sbin` — Contém comandos para coisas como gerenciar contas de usuário (como `useradd`) e verificar processos que estão segurando arquivos abertos (como `lsof`). Comandos que são executados como processos `daemon` também estão contidos nesse diretório. Processos `daemon` são processos que rodam em segundo plano, à espera de requisições de serviços, como

os para acessar uma impressora ou uma página web. (Olhe para os comandos que terminam em `d`, como `sshd`, `pppd` e `cupsd`.)

Alguns comandos administrativos estão contidos em diretórios de usuários regulares (como `/bin` e `/usr/bin`). Isso é especialmente verdadeiro para os comandos que têm algumas opções disponíveis para todos. Um exemplo é o comando `/bin/mount`, que qualquer um pode usar para listar os sistemas de arquivos montados, mas somente `root` pode usar para montar sistemas de arquivos. (Alguns desktops, porém, são configurados para permitir que os usuários regulares usem `mount` para montar CDs, DVDs ou outras mídias removíveis)

ta

Consulte a seção “Montando sistemas de arquivos”, no Capítulo 12, para obter instruções sobre como montar um sistema de arquivos.

Para localizar comandos destinados primariamente ao administrador de sistema, consulte as páginas `man` da seção 8 (normalmente em `/usr/share/man/man8`). Elas contêm descrições e opções para a maioria dos comandos administrativos do Linux.

Alguns aplicativos de terceiros adicionam comandos administrativos aos diretórios que não estão em seu `PATH`. Por exemplo, um aplicativo pode colocar comandos no diretório `/usr/local/bin`, `/opt/bin`, `/usr/local/sbin`. Algumas distribuições Linux automaticamente adicionam diretórios ao seu `PATH`, geralmente antes de seus diretórios padrão `bin` e `sbin`. Assim, os comandos instalados para os diretórios não são apenas acessíveis, mas também podem substituir os comandos de mesmo nome em outras pastas.

Arquivos de configuração administrativa

Os arquivos de configuração são outros dos pilares da administração Linux. Quase tudo o que você configura para seu computador particular — contas de usuário, endereços de rede ou preferências de interface gráfica — é armazenado em arquivos de texto simples. Isso apresenta algumas vantagens e desvantagens.

A vantagem de arquivos de texto simples é que é fácil de lê-los e mudá-los. Qualquer editor de texto serve. A desvantagem, porém, é que, ao editar arquivos de configuração, nenhuma verificação de erro está acontecendo. Você tem de executar o programa que lê esses arquivos (como um `daemon` de rede ou o desktop X) para descobrir se configurou os arquivos corretamente. Enquanto alguns arquivos de configuração usam estruturas padrão, tais como XML, para armazenar informações, muitos não o fazem. Então, você precisa aprender as regras de estrutura específicas para cada arquivo de configuração. Uma vírgula ou uma aspa no lugar errado às vezes pode fazer uma interface toda falhar.

ta

Alguns pacotes de software oferecem um comando para testar a sanidade do arquivo de configuração associado a um

te antes de iniciar um serviço. Por exemplo, o comando `testparm` é usado com o Samba para verificar a validade de seu arquivo `smb.conf`. Outras vezes, o processo `daemon` fornecendo um serviço oferece uma opção para verificar seu arquivo de configuração. Por exemplo, execute `httpd -t` para verificar sua configuração de servidor web Apache antes de iniciar seu servidor web.

Ao longo deste livro, você vai encontrar as descrições dos arquivos de configuração que você precisa para configurar as diferentes características que compõem os sistemas Linux. Os dois principais locais de arquivos de configuração são seu diretório inicial (onde seus arquivos de configuração pessoal são mantidos) e o diretório `/etc` (que contém arquivos de configuração de todo o sistema).

A seguir, são descritos os diretórios (e subdiretórios) que contêm arquivos de configuração úteis. As descrições são seguidas por alguns arquivos de configuração individuais em `/etc` que são de interesse particular. Visualizar o conteúdo de arquivos de configuração do Linux pode ensinar muito sobre administração de sistemas Linux.

- `$HOME` — Todos os usuários armazenam informações em seus diretórios iniciais, que determinam como suas contas de login se comportam. Vários arquivos de configuração estão diretamente no diretório inicial de cada usuário (como `/home/joe`) e começam com um ponto (`.`), de tal modo que eles não aparecem no diretório de um usuário quando você usa um comando `ls` padrão (você precisa digitar `ls -a` para vê-los). Da mesma maneira, arquivos e diretórios ponto não vão aparecer na maioria das janelas do gerenciador de arquivos por padrão. Há arquivos ponto que definem o comportamento do shell de cada usuário, a aparência e comportamento do desktop e opções utilizadas com seu editor de texto. Há até arquivos, como aqueles no diretório `$HOME/.ssh` de cada usuário, que configuram permissões de rede para cada usuário. (Para ver o nome de seu diretório inicial, digite `echo $HOME` partir de um shell.)
- `/etc` — Esse diretório contém a maioria dos arquivos de configuração de sistema básicos do Linux. A Tabela 8.1 mostra alguns arquivos de configuração `/etc` de interesse.
- `/etc/cron*` — Diretórios nesse conjunto contêm arquivos que definem como o utilitário `crond` executa aplicativos com base em uma agenda diária (`cron.daily`), de hora em hora (`cron.hourly`), mensal (`cron.monthly`) ou semanal (`cron.weekly`).
- `/etc/cups` — Contém arquivos usados para configurar o serviço de impressão CUPS.
- `/etc/default` — Contém arquivos que definem valores padrão para diversas utilidades. Por exemplo, o arquivo para o comando `useradd` define o número do grupo padrão, o diretório inicial, a data de expiração de senha, o shell e o diretório esqueleto (`/etc/skel`) que são utilizados para criar uma nova conta de usuário.
- `/etc/httpd` — Contém uma variedade de arquivos usados para configurar o comportamento de seu servidor web Apache (especificamente, o processo `daemon httpd`. (Em alguns sistemas Linux, o diretório `/etc/apache` ou `/etc/apache2` é usado.)
- `/etc/init.d` — Contém as cópias permanentes dos scripts de nível de execução no estilo System V. Esses scripts são muitas vezes linkados a partir dos diretórios `/etc/rc?.d` para ter

cada serviço associado a um script iniciado ou parado para o nível de execução específico. O ? é substituído pelo número do nível de execução (de 0 a 6).

- `/etc/mail` — Contém arquivos usados para configurar o agente de transporte de e-mail `sendmail`.
- `/etc/pcmcia` — Contém arquivos de configuração que permitem que você tenha uma variedade de cartões PCMCIA configurados para seu computador. (Slots PCMCIA são as aberturas em seu laptop que permitem que você conecte ao computador cartões PCMCIA do tamanho de cartões de crédito. Você pode conectar dispositivos como modems e CD-ROMs externos. Com a disseminação dos dispositivos USB, hoje os slots PCMCIA são menos comuns do que eram.)
- `/etc/postfix` — Contém arquivos de configuração para o agente de transporte de e-mail `postfix`.
- `/etc/ppp` — Contém diversos arquivos de configuração usados para configurar o Point-to-Point Protocol (PPP), o qual permite fazer uma conexão discada com a internet. (O PPP era mais comumente utilizado quando os modems discados eram populares.)
- `/etc/rc?.d` — Há um diretório `rc?.d` separado para cada estado válido do sistema: `rc0.d` (estado desligado), `rc1.d` (estado monousuário), `rc2.d` (estado multiusuário), `rc3.d` (estado multiusuário mais rede), `rc4.d` (estado definido pelo usuário), `rc5.d` (estado multiusuário, rede e login GUI) e `rc6.d` (estado reboot).
- `/etc/security` — Contém arquivos que definem uma variedade de condições de segurança padrão para seu computador, basicamente definindo como a autenticação é feita. Esses arquivos são parte do pacote `pam` (módulos de autenticação acopláveis).
- `/etc/skel` — Todos os arquivos contidos nesse diretório são automaticamente copiados para o diretório inicial do usuário quando este é adicionado ao sistema. Por padrão, a maioria desses arquivos são arquivos ponto, como `.kde` (um diretório para definir padrões do desktop KDE) e `.bashrc` (para definir os valores padrão utilizados com o shell `bash`).
- `/etc/sysconfig` — Contém arquivos importantes do sistema de configuração que são criados e mantidos por vários serviços (incluindo `iptables`, `samba` e a maioria dos serviços de rede). Esses arquivos são essenciais para distribuições Linux, como o Fedora e o RHEL, que usam ferramentas de interface gráfica, mas não são usados em outros sistemas Linux.
- `/etc/xinetd.d` — Contém um conjunto de arquivos, em que cada um deles define um serviço de rede por demanda que o daemon `xinetd` escuta em uma porta específica. Quando o processo de daemon `xinetd` recebe uma requisição de um serviço, ele usa as informações contidas nesses arquivos para determinar qual daemon processar para começar a lidar com a requisição.

A seguir são apresentados alguns arquivos de configuração interessantes em `/etc`:

- `aliases` — Pode conter listas de distribuição utilizadas pelo serviço de e-mail Linux. (Esse arquivo pode estar localizado em `/etc/mail`.)

- `bashrc` — Define todos os padrões de nível de sistema para usuários do shell `bash`. (Isso pode ser chamado `bash.bashrc` em algumas distribuições Linux.)
- `crontab` — Configura a data e a hora para a execução de tarefas automatizadas e variáveis associadas com o recurso `cron` (como o `SHELL` e `PATH` associado com o `cron`).
- `csch.cshrc` (ou `cshrc`) — Configura padrões de nível de sistema para usuários do `csch` (shell C).
- `exports` — Contém uma lista de diretórios locais que estão disponíveis para serem compartilhados por computadores remotos usando o Network File System (NFS).
- `fstab` — Identifica os dispositivos de mídia de armazenamento comuns (disco rígido, disquete, CD-ROM etc.) e os locais onde são montados no sistema Linux. Isso é usado pelo comando `mount` para escolher quais sistemas de arquivos devem ser montados primeiro na inicialização do sistema.
- `group` — Identifica os nomes e IDs de grupo (GID) que estão definidos no sistema. As permissões de grupo no Linux são definidas pelo segundo de três conjuntos de bits `rwX` (leitura, gravação, execução) associados com cada arquivo e diretório.
- `gshadow` — Contém senhas de sombra para grupos.
- `host.conf` — Usado por aplicativos mais antigos para definir os locais em que os nomes de domínio (por exemplo, `redhat.com`) são procurados em redes TCP/IP (como a internet). Por padrão, o arquivo `hosts local` é procurado e, então, qualquer entrada de servidor de nomes em `resolv.conf`.
- `hosts` — Contém endereços IP e nomes de `host` que você pode alcançar a partir de seu computador. (Normalmente, esse arquivo é usado apenas para armazenar nomes de computadores em sua rede local ou pequena rede privada.)
- `hosts.allow` — Lista computadores `host` que estão autorizados a utilizar determinados serviços TCP/IP do computador local.
- `hosts.deny` — Lista computadores `host` que *não* estão autorizados a utilizar determinados serviços TCP/IP do computador local (embora esse arquivo seja usado se você criá-lo, ele não existe por padrão).
- `inittab` — Contém informações que definem quais programas iniciam e param quando o Linux é inicializado, é desligado ou entra em diferentes estados no meio disso. Esse arquivo de configuração é o primeiro a ser lido quando o Linux inicia o processo `init`.
- `mtab` — Contém uma lista de sistemas de arquivos que estão atualmente montados.
- `mtools.conf` — Contém configurações usadas por ferramentas DOS no Linux.
- `named.conf` — Contém configurações de DNS, se você estiver executando seu próprio servidor DNS.

- `nsswitch.conf` — Contém configurações de troca do serviço de nomes, para identificar de onde (host local ou via serviços de rede) vêm as informações de sistemas críticos (contas de usuário, mapeamentos de nome de host/endereço etc.).
- `ntp.conf` — Inclui informações necessárias para executar o Network Time Protocol (NTP).
- `passwd` — Armazena informações das contas de todos os usuários válidos no sistema. Também inclui outras informações, como o diretório inicial e o shell padrão. (Raramente inclui as senhas em si, que normalmente estão armazenados no arquivo `/etc/shadow`.)
- `printcap` — Contém definições para as impressoras configuradas para seu computador. (Se o arquivo `printcap` não existir, procure informações sobre a impressora no diretório `/etc/cups`.)
- `profile` — Define o ambiente no nível de sistema e os programas de inicialização para todos os usuários. Esse arquivo é lido quando o usuário efetua o login.
- `protocols` — Configura números e nomes de protocolo para uma variedade de serviços de internet.
- `rpc` — Define os nomes e números das chamadas de procedimento remoto.
- `services` — Define nomes e serviços de TCP/IP e UDP e suas atribuições de portas.
- `shadow` — Contém senhas criptografadas para usuários que estão definidos no arquivo `passwd`. (Isso é visto como uma forma mais segura de armazenar senhas do que a senha original criptografada no arquivo `passwd`. O arquivo `passwd` precisa ser lido publicamente, enquanto o arquivo `shadow` pode ser ilegível por todos, exceto o usuário `root`.)
- `shells` — Lista os interpretadores de linha de comando de shell (`bash`, `sh`, `csch` etc.) que estão disponíveis no sistema, bem como suas localizações.
- `sudoers` — Configura comandos que podem ser executados por usuários, que exceto por isso não tem permissão para executar o comando, usando o comando `sudo`. Em particular, esse arquivo é utilizado para fornecer a usuários selecionados permissão de `root`.
- `rsyslog.conf` — Define quais mensagens de log são coletadas pelo daemon `syslogd` e quais arquivos são armazenados. (Normalmente, as mensagens de log são armazenadas em arquivos contidos no diretório `/var/log`.)
- `termcap` — Lista definições para os terminais de caracteres, de modo que aplicativos baseados em caractere saibam quais recursos são suportados por um dado terminal. Terminais gráficos e aplicações tornaram esse arquivo obsoleto para a maioria das pessoas.
- `xinetd.conf` — Contém informações de configuração simples usados pelo processo daemon `xinetd`. Esse arquivo em sua maior parte aponta para o diretório `/etc/xinetd.d` para obter informações sobre serviços individuais.

Outro diretório, `/etc/X11` `etc/X11`, inclui subdiretórios que contêm, cada um, arquivos de todo o

sistema de configuração usados por gerenciadores de janelas X e gerenciadores diferentes disponíveis para Linux. O arquivo `xorg.conf` (configura o computador e o monitor para torná-los utilizáveis com o X) e os diretórios de configuração contendo arquivos usados por `xdm` e `xinit` para iniciar o X estão aqui.

Os diretórios relacionados aos gerenciadores de janela contêm arquivos que incluem os valores padrão que um usuário obterá se esse usuário iniciar um desses gerenciadores de janela no seu sistema. Gerenciadores de janelas que podem ter arquivos de todo o sistema de configuração nesses diretórios incluem `twm` (`twm/`) e `xfce` (`xdg/`).

ta

ns arquivos e diretórios em `/etc/X11` estão linkados a locais no diretório `/usr/X11R6`.

Arquivos de log administrativos

Uma das coisas que o Linux faz bem é monitorar a si próprio. Isso é uma coisa boa, quando você considera quanta coisa está acontecendo em um sistema operacional complexo. Às vezes você está tentando obter um novo recurso para trabalhar e ele falha sem explicar minimamente por que falhou. Outras vezes, você quer monitorar seu sistema para ver se há tentativas de acesso ilegal ao seu computador. Em qualquer uma dessas situações, você pode utilizar os arquivos de log para ajudar a monitorar o problema.

A principal utilidade para registro em log de erros e mensagens de depuração em Linux é o daemon `rsyslogd`. (Alguns sistemas Linux mais antigos usam `syslogd` e `daemons syslogd`).

O registro em log será feito de acordo com informações no arquivo `/etc/rsyslog.conf`. As mensagens são normalmente direcionadas para arquivos de log que geralmente estão no diretório `/var/log`. Eis alguns arquivos de log comuns:

- `boot.log` — Contém mensagens de boot sobre serviços à medida que eles iniciam.
- `messages` — Contém muitas mensagens informativas gerais sobre o sistema.
- `secure` — Contém mensagens relacionadas à segurança, como a atividade de login ou qualquer outro ato que autentica usuários.
- `XFree86.0.log` ou `Xorg.0.log` — Dependendo de qual servidor X você está usando, contém mensagens sobre a placa de vídeo, o mouse e a configuração do monitor.

Se você estiver usando o Fedora, o utilitário System Log Viewer é um bom caminho para percorrer seus arquivos de registro em log do sistema. No menu Applications, selecione System ⇒ Administration ⇒ Log File Viewer. Você não só pode ver os logs de boot, kernel, e-mail, segurança e outros logs de sistema, como também pode usar o painel de visualização para selecionar as mensagens de log de uma determinada data.

Consulte o Capítulo 13, “Entendendo a administração de servidor”, para obter informações sobre a

Usando outras contas administrativas

Você não ouve muito sobre outras contas de usuários administrativos (além de `root`) sendo utilizado com o Linux. Era uma prática relativamente comum nos sistemas UNIX ter vários logins administrativos diferentes que permitiam que tarefas administrativas fossem divididas entre vários usuários. Por exemplo, as pessoas sentadas perto de uma impressora podem ter permissões de `lp` para mover trabalhos de impressão para outra impressora se elas souberem que a primeira não está funcionando.

Em qualquer caso, logins administrativos estão disponíveis no Linux, porém, o login direto desses usuários está desabilitado por padrão. As contas são mantidas principalmente para fornecer posse de arquivos e processos associados com serviços particulares. Ao executar processos `daemon` sob logins administrativos separados, ter um desses processos dominado por um cracker não dá permissão de `root` ao cracker e a capacidade de acessar outros processos e arquivos. Considere os seguintes exemplos:

- `lp` — O usuário possui coisas como o arquivo de log de impressão `/var/log/cups` e vários arquivos de cache e spool de impressão. O diretório inicial para `lp` é `/var/spool/lpd`.
- `apache` — O usuário pode ser usado para criar arquivos de conteúdo e diretórios. É usado principalmente para executar processos do servidor web (`httpd`).
- `postfix` — O usuário possui vários diretórios e arquivos de spool do servidor de e-mail. O usuário executa os processos `daemon` usados para fornecer o serviço `postfix` (`master`).
- `uucp` — O usuário possui vários comandos `uucp` (antigamente utilizados como o principal método para comunicação serial discada), bem como arquivos de log em `/var/log/uucp`, arquivos de spool em `/var/spool`, comandos administrativos (tais como `uuchk`, `uucico`, `uuconv` e `uuxqt`) em `/usr/sbin` e comandos do usuário (`uucp`, `cu`, `uuname`, `uustat` e `uux`) em `/usr/bin`. O diretório inicial para o `uucp` é `/var/spool/uucp`. O serviço `uucp` é raramente usado.
- `bin` — O usuário possui muitos comandos em `/bin` em sistemas UNIX tradicionais. Esse não é o caso em alguns sistemas Linux (como o Fedora e o Gentoo) porque `root` possui a maioria dos arquivos executáveis. O diretório inicial de `bin` é `/bin`.
- `news` — O usuário pode fazer a administração de serviços de news da internet, dependendo de como você configura a permissão para `/var/spool/news` e outros recursos relacionados com news. O diretório inicial para `news` é `/etc/news`.

Por padrão, os logins administrativos na lista anterior estão desabilitados. Você precisa mudar o shell padrão de sua configuração atual (geralmente `/sbin/nologin` ou `/bin/false`) para um shell real (geralmente `/bin/bash`) para ser capaz de fazer login como esses usuários.

Verificando e configurando o hardware

Em um mundo perfeito, após a instalação e inicialização do Linux, todo seu hardware é detectado e disponível para acesso. Embora os sistemas Linux estejam rapidamente se aproximando desse mundo, há momentos em que você deve tomar medidas especiais para fazer o hardware do computador funcionar. Além disso, o crescente uso de dispositivos USB e FireWire removíveis (CDs, DVDs, pen drives, câmeras digitais e discos rígidos removíveis) tornou importante para o Linux fazer o seguinte:

- Gerenciar eficientemente o hardware removível
- Olhar para a mesma peça de hardware de maneiras diferentes (por exemplo, ser capaz de ver uma impressora não apenas como impressora, mas também como uma máquina de fax, um scanner e um dispositivo de armazenamento)

Recursos do kernel do Linux adicionados nos últimos anos tornaram possível mudar drasticamente a maneira como dispositivos de hardware são detectados e gerenciados. Recursos no kernel, ou estreitamente relacionados a ele, incluem Udev (para dinamicamente nomear e criar dispositivos de hardware removíveis) e HAL (para passar informações sobre alterações de hardware para o espaço do usuário).

Se tudo isso soar um pouco confuso, não se preocupe. Na verdade, isso é concebido para facilitar sua vida como um usuário Linux. O resultado final de funcionalidades incorporadas no kernel é que a manipulação de dispositivos no Linux tornou-se:

- **Mais automática** — Para a maioria dos hardwares, quando um dispositivo de hardware é conectado ou desconectado, ele é automaticamente detectado e identificado. Interfaces de acesso ao hardware são adicionadas, por isso ele é acessível para o Linux. Então, o fato de o hardware estar presente (ou ter sido removido) é passado para o nível de usuário, no qual os aplicativos ouvindo alterações de hardware estão prontos para montar o hardware e/ou iniciar um aplicativo (como um visualizador de imagens ou leitor de música).
- **Mais flexível** — Se você não gosta do que acontece automaticamente quando um item de hardware é conectado ou desconectado, é possível mudar isso. Por exemplo, recursos incorporados nos desktops GNOME e KDE permitem que você escolha o que acontece quando um CD de música ou um DVD de dados é inserido, ou quando uma câmera digital é conectada. Se preferir que um programa diferente seja carregado para lidar com isso, você pode facilmente fazer essa mudança.

Essa seção aborda várias questões relacionadas com fazer seu hardware funcionar corretamente no Linux. Primeiro, ele descreve como verificar informações sobre os componentes de hardware de seu sistema. Então, ele cobre como configurar o Linux para lidar com mídia removível. Finalmente, descreve como usar as ferramentas para carregar manualmente e trabalhar com drivers para hardware que não são detectados e corretamente carregados.

Verificando seu hardware

Quando seu sistema inicializa, o kernel detecta o hardware e carrega os drivers que permitem que o Linux funcione com esse hardware. Como as mensagens sobre a detecção de hardware passam

rapidamente pela tela quando você inicializa, para ver as mensagens de problemas potenciais, é preciso reexibi-las depois que o sistema inicializa.

Há duas maneiras de ver as mensagens de inicialização do kernel depois que o Linux inicializa. Qualquer usuário pode executar o comando `dmesg` para ver qual hardware foi detectado e quais drivers foram carregados pelo kernel durante a inicialização. À medida que novas mensagens são geradas pelo kernel, elas também são disponibilizadas para o comando `dmesg`. Uma segunda maneira de ver as mensagens em alguns sistemas Linux é exibindo o conteúdo do arquivo `/var/log/dmesg`, se ele existir.

ta

ois que seu sistema está funcionando, muitas mensagens do kernel são enviadas para o arquivo `/var/log/messages`. Assim, por exemplo, se você quiser ver o que acontece quando conecta um pen drive, pode digitar `tail -f /var/log/messages` e ver como dispositivos e pontos de montagem são os.

Eis um exemplo de uma saída do comando `dmesg` que foi reduzida para mostrar algumas informações interessantes:

```
$ dmesg | less
[ 0.000000] Linux version 3.1.0-7.fc16.i686
          (mockbuild@x86-11.phx2.fedoraproject.org)
          (gcc version 4.6.2 20111027 (Red Hat 4.6.2-1) (GCC) )
          #1 SMP Tue Nov 1 21:00:16 UTC 2011
[ 0.000000] DMI: Dell Inc. Precision WorkStation 490
          /0GU083, BIOS A06 08/20/2007
[0.000000] Kernel command line: initrd=initrd0.img
          root=live:CDLABEL=Fedora-16-i686-Live-Desktop.iso
          rootfstype=auto ro liveimg quiet rhgb rd.luks=0
          rd.md=0 rd.dm=0 BOOT_IMAGE=vmlinuz0
[ 0.056934] CPU0: Intel(R) Xeon(R) CPU E5320 @ 1.86GHz stepping 0b
[ 0.272025] Brought up 4 CPUs
[ 0.272029] Total of 4 processors activated (14895.38 BogomIPS).
[ 3.020618] Serial: 8250/16550 driver, 4 ports, IRQ sharing enabled
[ 3.041185] serial8250: ttyS0 at I/O 0x3f8 (irq = 4) is a 16550A
[ 3.061880] serial8250: ttyS1 at I/O 0x2f8 (irq = 3) is a 16550A
[ 3.145982] mousedev: PS/2 mouse device common for all mice
[ 3.538044] scsi 6:0:0:0: CD-ROM
          TSSTcorp DVD-ROM TS-H352C DE02 PQ: 0 ANSI: 5
[ 3.870128] input: ImPS/2 Generic Wheel Mouse
          as /devices/platform/i8042/seriol/input/input3
[ 26.964764] e1000: Intel(R) PRO/1000 Network Driver - version 7.3.21-k8-NAPI
[ 26.964767] e1000: Copyright (c) 1999-2006 Intel Corporation.
[ 26.964813] e1000 0000:0c:02.0: PCI INT A -> GSI 18 (level, low) -> IRQ 18
[ 27.089109] parport_pc 00:08: reported by Plug and Play ACPI
[ 27.089169] parport0: PC-style at 0x378 (0x778), irq 7 [PCSPP,TRISTATE]
[24179.176315] scsi 9:0:0:0: Direct-Access
          S31B1102 USB DISK 1100 PQ: 0 ANSI: 0 CCS
[24179.177466] sd 9:0:0:0: Attached scsi generic sg2 type 0
[24179.177854] sd 9:0:0:0: [sdb]
          8343552 512-byte logical blocks: (4.27 GB/3.97 GiB)
[24179.178593] sd 9:0:0:0: [sdb] Write Protect is off
```

A partir dessa saída, você vê primeiro a versão do kernel do Linux, seguida de informações sobre o computador (Dell Precision WorkStation) e opções de linha de comando do kernel. Em seguida, você pode ver o tipo dos processadores (Intel Xeon) e o número de CPUs (4). Depois disso, eu excluí as informações sobre os hardwares conectados ao computador: portas seriais, porta de mouse, unidade de

CD, placa de rede interface (e1000) e porta paralela. As últimas linhas refletem um pen drive de 4GB sendo conectado ao computador.

Se algo der errado ao detectar seu hardware ou drivers de carga, você pode consultar essas informações para ver o nome e o número do modelo de hardware que não está funcionando. Então, você pode procurar fóruns ou documentação sobre Linux para tentar resolver o problema.

Uma vez que seu sistema está instalado e funcionando, existem outros comandos que permitem que você examine para obter informações detalhadas sobre o hardware de seu computador. O comando `lspci` lista barramentos PCI em seu computador e os dispositivos conectados a eles. Eis um trecho de saída:

```
$ lspci
```

```
00:00.0 Host bridge: Intel Corporation 5000X Chipset Memory
ControllerHub
00:02.0 PCI bridge: Intel Corporation 5000 Series Chipset PCI
Express x4 Port 2
00:1b.0 Audio device: Intel Corporation 631xESB/632xESB
High Definition Audio Controller (rev 09)
00:1d.0 USB controller: Intel Corporation 631xESB/632xESB/3100
Chipset UHCI
USBController#1 (rev 09)
07:00.0 VGA compatible controller: nVidia Corporation NV44 [Quadro
NVS 285]
0c:02.0 Ethernet controller: Intel Corporation 82541PI
Gigabit Ethernet Controller (rev 05)
```

A ponte do servidor conecta o barramento local a outros componentes na ponte PCI. Reduzi a saída para mostrar informações sobre os diferentes dispositivos no sistema que lidam com vários recursos: som (dispositivo de áudio), pen drives e outros dispositivos USB (controlador USB), monitor (controlador compatível com VGA) e placas de rede com fio (controlador Ethernet). Se você tiver problemas em fazer qualquer um desses dispositivos funcionar, anotar os nomes e números de modelo lhe dá algo para procurar no Google.

Para obter uma saída mais detalhada de `lspci`, adicione uma ou mais opções `-v`. Por exemplo, usando `lspci -vvv`, recebi informações sobre meu controlador Ethernet, incluindo latência, capacidade do controlador e o driver do Linux (e1000) a ser utilizado para o dispositivo.

Se você estiver interessado especificamente em dispositivos USB, experimente o comando `lsusb`. Por padrão, `lsusb` lista informações sobre hubs USB do computador, juntamente com todos os dispositivos USB conectados às portas USB do computador:

```
$ lsusb
```

```
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
Bus 002 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 003 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 004 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
```

```
Bus 005 Device 001: ID 1d6b:0001 Linux Foundation 1.1 root hub
Bus 002 Device 002: ID 413c:2105 Dell Computer Corp. Model L100
Keyboard
Bus 002 Device 004: ID 413c:3012 Dell Computer Corp. Optical Wheel
Mouse
Bus 001 Device 005: ID 090c:1000 Silicon Motion, Inc. - Taiwan 64MB
QDI U2 DISK
```

A partir da saída anterior, você pode ver o modelo de um teclado, mouse e pen drive conectados ao computador. Tal como acontece com `lspci`, você pode adicionar uma ou mais opções `-v` para ver mais detalhes.

Para ver mais detalhes sobre seu processador, execute o comando `lscpu`. Esse comando fornece informações básicas sobre os processadores de seu computador.

```
$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
CPU(s):                4
On-line CPU(s) list:   0-3
Thread(s) per core:    1
Core(s) per socket:    4
...
```

A partir da amostragem de saída de `lscpu`, você pode ver que esse é um sistema de 64 bits (x86-64) que pode operar em 32 bits ou 64 bits e há quatro CPUs.

Gerenciando hardware removível

Sistemas Linux, como Red Hat Enterprise Linux, Fedora e outros que suportam ambientes de desktop KDE e GNOME completos, incluem ferramentas gráficas simples para configurar o que acontece quando você anexa dispositivos removíveis populares ao computador. Assim, com um desktop KDE ou GNOME em execução, basta conectar um dispositivo USB ou inserir um CD ou DVD e uma janela pode aparecer para lidar com esse dispositivo.

Embora diferentes ambientes de trabalho compartilhem muitos dos mesmos mecanismos subjacentes (em particular, o Udev) para detectar e nomear hardware removível, oferecem diferentes ferramentas para configurar como eles são montados ou usados. O Udev (usando o daemon `udev`) cria e remove dispositivos (diretório `/dev`) quando hardware é adicionado e removido do computador. A camada de abstração de hardware (HAL) fornece a plataforma global para descobrir e configurar hardware. Configurações que são de interesse para alguém usando um sistema desktop Linux, porém, podem ser configuradas com ferramentas de desktop fáceis de usar.

O gerenciador de arquivos Nautilus, usado com o desktop GNOME, permite que você defina o que acontece quando conecta dispositivos removíveis ou insere mídia removível no computador a partir da janela File Management Preferences. As descrições dessa seção são baseadas no GNOME 3.2 no Fedora 16.

A partir do desktop GNOME 3.2, selecione Activities ⇒ Applications ⇒ System Tools ⇒ System Settings ⇒ Removable Media. A Figura 8.2 mostra um exemplo dessa janela.

As configurações de mídia removível na janela Removable Media.

As configurações a seguir estão disponíveis a partir da janela Removable Media. Essas configurações relacionam-se à maneira como mídias removíveis são tratadas quando são inseridas ou conectadas. Na maioria dos casos, você será perguntado sobre como lidar com uma mídia que é inserida ou conectada.

- **CD de áudio** — Quando um CD de áudio é inserido, você pode optar por ser perguntado sobre o que fazer (o padrão), não fazer nada, abrir o conteúdo em uma janela da pasta ou selecionar a partir de vários leitores de CD de áudio a serem carregados para reproduzir o conteúdo. Rhythmbox (leitor de música), Audio CD Extractor (gravador de CD) e Brasserio (gravador de CD) estão entre as opções que você tem para lidar com um CD de áudio.
- **DVD de vídeo** — Quando um DVD de vídeo comercial é inserido, o sistema lhe pergunta o que o que você quer fazer com esse DVD. Você pode alterar esse padrão para carregar o Movie Player (Totem), o Brasero (gravador de DVD) ou outro tocador de mídia que você tenha instalado (como o MPlayer).
- **Tocador de música** — Quando a mídia inserida contém arquivos de áudio, você será perguntado sobre o que fazer. Você pode optar por fazer o Rhythmbox ou algum outro tocador de música começar a tocar os arquivos selecionando o tocador a partir dessa caixa.
- **Fotos** — Quando a mídia inserida (como um cartão de memória de uma câmera digital) contém imagens digitais, você é perguntado sobre o que fazer com essas imagens. Você pode optar por não fazer nada ou por abrir as imagens no visualizador de imagens Shotwell (o aplicativo padrão para visualização de imagens no ambiente GNOME), ou outro gerenciador de fotos instalado.
- **Software** — Quando a mídia inserida contém um aplicativo de execução automática, um prompt de execução automática será aberto. Para mudar esse comportamento (para não fazer nada ou abrir o conteúdo de mídia em uma pasta), você pode selecionar a partir dessa caixa.
- **Outras mídias** — Selecione a caixa Type sob o título Other Media para selecionar como as mídias menos utilizadas são manipuladas. Por exemplo, você pode selecionar quais ações são tomadas para lidar com DVDs de áudio ou discos Blu-ray, CDs, DVDs ou discos HD DVD. Você pode selecionar quais aplicativos carregar para disco de vídeo Blu-ray, leitores de e-book, vídeos HD DVD, CDs de fotos, Super Vídeo CDs e CDs de vídeo.

Note que as configurações descritas aqui estão em vigor somente para o usuário que está conectado. Se vários usuários têm contas de acesso, cada um pode ter sua própria maneira de lidar com mídia removível.

criptografar o DVD. Há questões jurídicas e opções de outros tocadores de filmes que você deve considerar se quiser reproduzir filmes em DVD comerciais no Linux.

As opções para conectar pen drives ou discos rígidos não estão listadas nessa janela, mas se você conectar uma dessas unidades em seu computador, os dispositivos serão criados automaticamente quando plugá-los (chamados `/dev/sda`, `/dev/sdb/sda`, `/dev/sdb` etc.). Quaisquer sistemas de arquivos encontrados nesses dispositivos são automaticamente montados em `/media` e você será perguntado se deseja abrir uma janela do Nautilus para visualizar arquivos nesses dispositivos. Isso é feito automaticamente, assim você não tem de fazer nenhuma configuração especial que isso aconteça.

Quando terminar de usar um pen drive, clique com o botão direito do mouse sobre o nome do dispositivo na janela do gerenciador de arquivos Nautilus e selecione Remove Drive. Essa ação desmonta a unidade e remove o ponto de montagem no diretório `/media`. Depois disso, você pode desligar em segurança o pen drive de seu computador.

Trabalhando com módulos carregáveis

Se você adicionou um hardware ao computador que não é detectado corretamente, você pode precisar carregar manualmente um módulo para esse hardware. O Linux vem com um conjunto de comandos para carregar, descarregar e obter informações sobre os módulos de hardware.

Os módulos do kernel estão instalados nos subdiretórios `/lib/modules/`. O nome de cada subdiretório é baseado no número de versão do kernel. Por exemplo, se o kernel fosse 3.1.0-7.fc16.i686, o diretório `lib/modules/3.1.0-7.fc16.i686` conteria drivers para esse kernel. Módulos nesses diretórios podem ser carregados e descarregados conforme são necessários.

Comandos para listar, carregar, descarregar e obter informações sobre os módulos estão disponíveis com o Linux. As próximas seções descrevem como usar esses módulos.

Listando os módulos carregados

Para ver quais módulos estão carregados no kernel em execução em seu computador, use o comando `lsmod`. Considere o seguinte exemplo:

```

# lsmod
Module                Size      Used by
snd_seq_oss           38912      0
snd_seq_midi_event    9344       1  snd_seq_oss
snd_seq               67728      4
snd_seq_oss,snd_seq_midi_event
snd_seq_device        8328       2  snd_seq_oss,snd_seq
.
.
.
autofs                16512      0
ne2k_pci              9056       0
8390                  13568      1  ne2k_pci
ohci1394              41860      0
ieee1394              284464     1  ohci1394
floppy                65712      0
sg                    36120      0
scsi_mod              124600     1  sg
parport_pc            39724      0
parport               47336     1  parport_pc
ext3                  128424     2
jbd                   86040     1  ext3

```

Esse resultado mostra uma variedade de módulos que foram carregados em um sistema Linux, incluindo vários para suportar o sistema de som ALSA, alguns dos quais oferecem compatibilidade com OSS (`snd_seq_oss`).

Para encontrar informações sobre qualquer um dos módulos carregados, use o comando `modinfo`. Por exemplo, você poderia digitar o seguinte:

```

# /sbin/modinfo -d snd-seq-oss
"OSS-compatible sequencer module"

```

Nem todos os módulos têm descrições disponíveis e, se nada estiver disponível, dados não serão retornados. Nesse caso, porém, o módulo `snd-seq-oss` é descrito como um módulo sequenciador compatível com OSS. Você também pode usar a opção `-a` para ver o autor do módulo, ou `-n` para ver o arquivo objeto que representa o módulo. As informações sobre o autor muitas vezes tem o endereço de e-mail do criador do driver, assim você pode contatá-lo, se tiver problemas ou dúvidas sobre o assunto.

Carregando módulos

Você pode carregar qualquer módulo (como `root`) que foi compilado e instalado (um subdiretório `/lib/modules`) em seu kernel em execução com o comando `modprobe`.

Uma razão comum para carregar um módulo é usar um recurso temporariamente (como carregar um módulo para dar suporte a um sistema de arquivos especial em um disquete que você deseja acessar). Outra razão para carregar um módulo é identificá-lo como um que vai ser utilizado por uma determinada parte do hardware que não pôde ser automaticamente detectada.

Eis um exemplo do comando `modprobe` sendo usado para carregar o módulo `parport`, que fornece as funções essenciais para compartilhar portas paralelas, com vários equipamentos:

```

# modprobe parport

```

Depois que `parport` é carregado, você pode carregar o módulo `parport_pc` para definir as portas no

estilo PC disponíveis através da interface. O módulo `parport_pc` permite definir opcionalmente os endereços e números IRQ associados com cada dispositivo que compartilha a porta paralela. Por exemplo:

```
# modprobe parport_pc io=0x3bc irq=auto
```

Nesse exemplo, um dispositivo é identificado como tendo um endereço de `0x3bc` e o IRQ para o dispositivo é autodetectado.

O comando `modprobe` carrega módulos temporariamente — eles desaparecem na próxima reinicialização. Para adicionar permanentemente o módulo ao sistema, adicione a linha de comando `modprobe` para um dos scripts de inicialização sendo executado no momento da inicialização.

Removendo módulos

Use o comando `rmmmod` para remover um módulo de um kernel em execução. Por exemplo, para remover o módulo `parport_pc` do kernel atual, digite o seguinte:

```
# rmmmod parport_pc
```

Se ele não estiver atualmente ocupado, o módulo `parport_pc` é removido do kernel. Se ele estiver ocupado, tente eliminar qualquer processo que possa estar usando o dispositivo. Então, execute `rmmmod` novamente. Às vezes, o módulo que você está tentando remover depende de outros módulos que podem ser carregados. Por exemplo, o módulo `usbcore` não pode ser descarregado, enquanto o módulo de impressora USB (`usb_lpr`) estiver carregado, como mostrado a seguir:

```
# rmmmod usbcore
```

```
ERROR: Module usbcore is in use by wacom,usb_lpr,ehci_hcd,ohci_hcd
```

Em vez de usar `rmmmod` para remover módulos, você pode usar o comando `modprobe -r`. Com `modprobe -r`, em vez de apenas remover o módulo desejado, você também pode remover módulos dependentes que não estão sendo utilizados por outros módulos.

Resumo

Muitos recursos do Linux, especialmente aqueles que podem danificar o sistema ou afetar os outros usuários, exigem que você obtenha privilégios de root. Este capítulo descreve diferentes formas de obter privilégios de root: login direto, o comando `su` ou o comando `sudo`. Ele também cobre algumas das principais responsabilidades de um administrador de sistema e componentes (arquivos de configuração, ferramentas gráficas etc.) que são fundamentais para o trabalho de um administrador de sistema.

O próximo capítulo descreve como instalar um sistema Linux. As abordagens para a instalação do Linux que são cobertas nesse capítulo incluem como instalar a partir de um Live CD e a partir da mídia de instalação.

Exercícios

Use esses exercícios para testar seus conhecimentos de administração de sistema e permitir que você explore as informações sobre o hardware do sistema. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostrados no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. A partir de um desktop GNOME, abra a janela Time. Verifique se seu fuso horário está definido corretamente.
2. Execute um comando `ps` para classificar todos os processos em execução no sistema por nome de usuário. Observe quais usuários executam quais processos.
3. Encontre todos os arquivos no diretório `/var/spool` que pertencem a outros usuários que não sejam o root e apresente uma longa listagem deles.
4. Torne-se o usuário root usando o comando `su -`. Para provar que você tem privilégios de root, crie um arquivo vazio ou texto simples chamado `/mnt/test.txt`. Saia do shell quando terminar. Se estiver usando o Ubuntu, primeiro você deve configurar sua senha de root (`sudo passwd root`).
5. Efetue login como um usuário comum e torne-se root usando `su -`. Edite o arquivo `/etc/sudoers` para permitir que sua conta de usuário regular tenha totais privilégios de root por meio do comando `sudo`.
6. Como o usuário ao qual você acabou de dar privilégio de sudoers, use o comando `sudo` para criar um arquivo chamado `/mnt/test2.txt`. Verifique se o arquivo está lá e se pertence ao usuário root.
7. Execute o comando `tail -f` no arquivo `/var/log/messages` e conecte um pen drive a uma porta USB de seu computador. Desmonte o dispositivo e remova-o, continuando a observar a saída em `/var/log/messages`.
8. Execute um comando para ver quais dispositivos USB estão conectados ao seu computador.
9. Finja que você adicionou uma placa de TV ao seu computador, mas o módulo necessário para usá-la (`bttv`) não foi corretamente detectado e carregado. Carregue você mesmo o módulo `bttv` e depois veja se ele foi carregado. Outros módulos foram carregados com ele?
10. Remova o módulo `bttv` juntamente com todos os outros módulos que foram carregados com ele. Liste seus módulos para garantir que isso foi feito.

Instalando o Linux

NESTE CAPÍTULO

Escolhendo um método de instalação

Instalando um sistema de boot único ou multi-boot

Executando uma instalação Live CD do Fedora

Instalando o Red Hat Enterprise Linux

Particionando o disco para instalação

Entendendo o gerenciador de inicialização GRUB

Instalar o Linux se tornou uma coisa muito fácil de fazer — se você está começando com um computador que atende aos requisitos mínimos (disco rígido, memória RAM, CPU etc.) e não se importa em apagar completamente seu disco rígido, veja como escolher o melhor método de instalação. A instalação é mais complexa se você quiser se desviar de uma instalação padrão. Portanto, este capítulo começa com uma simples instalação de um live CD e avança para tópicos de instalação mais complexos.

Para facilitar sua introdução no assunto de instalar o Linux, abordo aqui três maneiras diferentes de instalação e guio você ao longo de cada processo:

- **Instalando a partir de um Live CD** — Um Linux Live CD é um disco de somente leitura que contém tudo o que você precisa para iniciar um sistema operacional Linux. Com o Live CD, você pode ignorar totalmente o disco rígido do computador, na verdade, é possível executar um Live CD em um sistema sem disco rígido. Uma vez que você estiver executando o Live CD, alguns permitem a inicialização de um aplicativo que instala permanentemente o conteúdo do Live CD em seu disco rígido. O procedimento de primeira instalação neste capítulo mostra como instalar permanentemente o Linux a partir de um Live CD do Fedora.
- **Instalando a partir de um DVD de instalação** — Um DVD de instalação, que está disponível com o Fedora, RHEL e outras distribuições Linux, oferece maneiras mais flexíveis de instalar o Linux. Em particular, em vez de apenas copiar os conteúdos inteiros do Live CD para seu computador, com um DVD de instalação, você pode escolher exatamente qual pacote de software deseja. O segundo procedimento de instalação que mostro neste capítulo guia-o ao longo do processo de instalação a partir de um DVD do Red Hat Enterprise Linux 6.

- **Instalando na empresa** — Sentar-se à frente de um computador e clicar para responder a algumas perguntas sobre a instalação não é muito inconveniente se você estiver instalando um único sistema. Mas e se você precisar instalar dezenas ou centenas de sistemas Linux? E se você quiser instalar esses sistemas de maneiras particulares que precisam ser repetidas em várias instalações? A última seção deste capítulo descreve maneiras eficientes de instalar vários sistemas Linux usando recursos de instalação de rede e arquivos kickstart.

Para experimentar os procedimentos neste capítulo junto comigo, você deve ter à sua frente um computador que não se importe de apagar os dados completamente. Como alternativa, você pode usar um computador que tem outro sistema operacional instalado (como o Windows), desde que haja espaço suficiente em disco disponível não utilizado. Descrevo o procedimento e o risco de perda de dados, se você decidir criar um desses arranjos de “boot dual” (Linux e Windows).

Escolhendo um Computador

Você pode obter uma distribuição Linux que roda em dispositivos de mão ou um PC antigo em seu armário com tão pouco como 24MB de RAM e um processador 486. Para ter uma boa experiência de desktop PC com Linux, porém, você deve considerar o que quer ser capaz de fazer com ele quando estiver escolhendo seu computador.

É importante considerar as especificações básicas que você precisa de um computador do tipo PC para executar o Fedora e o Red Hat Enterprise Linux. Como o Fedora é usado como base para o Red Hat Enterprise Linux, os requisitos de hardware são semelhantes para o hardware básico de desktop e de servidor nas duas distribuições.

- **Processador** — Um Pentium de 400 MHz é o mínimo para a instalação de uma interface gráfica. Para a maioria dos aplicativos, um processador de 32 bits é bom (x86). Mas se quiser configurar o sistema para fazer virtualização, você precisa de um processador de 64 bits (x86_64).

ta

Se tiver uma máquina de 486 (pelo menos 100 MHz), considere experimentar o Damn Small Linux (<http://www.damnsmalllinux.org>) ou o Slackware (<http://www.slackware.org>). Ele não tem a mesma interface gráfica, mas você poderia fazer alguns dos exercícios de shell. Se você tiver um MacBook, talvez você queira experimentar uma versão GNOME do Ubuntu, que você pode obter em <https://help.ubuntu.com/community/MacBook>.

- **RAM** — O Fedora recomenda pelo menos 1GB de RAM, mas pelo menos 2GB ou 3GB seria muito melhor. Em meu desktop RHEL, estou executando um navegador, um processador de texto e um leitor de e-mail e estou consumindo mais de 2GB de RAM.
- **Unidade de DVD ou CD** — Você precisa ser capaz de iniciar o processo de instalação a partir de uma unidade de DVD, CD ou USB. Se você não puder iniciar a partir de um DVD ou CD, há maneiras de iniciar a instalação a partir de um disco rígido ou uma unidade USB ou usando uma instalação PXE. Após o processo de instalação ser iniciado, eventualmente mais softwares

podem ser obtidos a partir de locais diferentes (através da rede ou a partir do disco rígido, por exemplo).

ta

(pronuncia-se pixie) significa Preboot eXecution Environment. Você pode inicializar um computador cliente a partir de uma placa de interface de rede que seja compatível com o PXE. Se um servidor de boot PXE estiver disponível na rede, ele pode fornecer tudo que um computador cliente precisa para inicializar. O que ele inicializa pode ser um sistema operacional. Assim, com um boot PXE, é possível fazer uma instalação completa do Linux sem um CD, DVD ou qualquer outra mídia física.

- **Placa de rede** — Você precisa de um hardware de rede com ou sem fio para poder adicionar mais software ou obter atualizações. O Fedora oferece repositórios de software livre se você puder se conectar à internet. Para o RHEL, atualizações estão disponíveis como parte de uma assinatura paga.
- **Espaço em disco** — O Fedora recomenda pelo menos 10GB de espaço em disco para uma instalação de desktop típica, embora as instalações possam variar (dependendo do pacote que você escolher instalar) de 600MB (para um servidor mínimo e sem qualquer instalação de interface gráfica) a 7GB (para instalar todos os pacotes a partir da mídia de instalação). Considere a quantidade de dados que você precisa armazenar. Enquanto documentos podem consumir muito pouco espaço, vídeos podem consumir grandes quantidades de espaço. (Em comparação, você pode instalar o Damn Small Linux Live CD em disco com apenas cerca de 200MB de espaço em disco.)
- **Recursos de hardware especiais** — Alguns recursos do Linux requerem recursos de hardware especiais. Por exemplo, para usar o Fedora ou o RHEL como um host de virtualização usando o KVM, o computador deve ter um processador que suporte a virtualização. Esses incluem chips AMD-V ou Intel-VT.

Se você não tem certeza sobre o hardware presente no computador, há algumas maneiras de verificá-lo. Se você estiver executando o Windows, a janela Propriedades do Sistema pode lhe mostrar o processador que você tem, bem como a quantidade de RAM que está instalada. Como uma alternativa, com o Fedora Live CD inicializado, abra um shell e digite `dmesg | less` para ver uma lista de dispositivos detectados em seu sistema.

Com o hardware no lugar, você pode optar por instalar o Linux a partir de um Live CD ou da mídia de instalação, como descrito nas próximas seções.

Instalando o Fedora a partir de um Live CD

No Capítulo 1, você aprendeu a obter e inicializar um Linux Live CD. Este capítulo guia-o ao longo do processo de instalação de um Live CD Fedora; assim, ele é permanentemente instalado em seu disco rígido.

A simplicidade é a principal vantagem da instalação a partir de um Live CD. Essencialmente, você está apenas copiando o kernel, aplicativos e configurações a partir da imagem do CD para o disco rígido. Há menos decisões que você tem de tomar para fazer esse tipo de instalação, mas você não pode escolher exatamente quais pacotes de software instalar. Após a instalação, é possível adicionar e remover pacotes como quiser.

As primeiras decisões que você tem de tomar sobre a instalação do Live CD incluem onde você deseja instalar o sistema e se você quer manter os sistemas operacionais existentes quando a instalação for concluída:

- **Computador single-boot** — A maneira mais fácil de instalar o Linux é não ter de se preocupar com outros sistemas operacionais ou dados no computador e deixar o Linux substituir tudo. Quando você estiver pronto, o computador inicializará diretamente no Fedora.
- **Computador multi-boot** — Se já tiver o Windows instalado em um computador e não quiser apagá-lo, você pode instalar o Fedora junto com o Windows no sistema. Então, durante o boot, é possível escolher qual sistema operacional inicializar. Para ser capaz de instalar o Fedora em um sistema diferente do sistema operacional instalado, você deve ter espaço em disco disponível (fora da partição do Windows) ou ser capaz de encolher o sistema Windows para ganhar espaço livre suficiente para instalar o Fedora.
- **Bare metal ou sistema virtual** — A instalação do Fedora resultante pode ser instalada para inicializar diretamente a partir do hardware do computador ou a partir de um sistema operacional existente no computador. Se tiver um computador que está sendo executado como uma máquina virtual, você pode instalar o Fedora nesse sistema como um convidado virtual. Produtos de host de virtualização incluem o KVM, o Xen e o VirtualBox (para sistemas Linux e UNIX, bem como Windows e Mac), o Hyper-V (para sistemas Microsoft) e o VMWare (sistemas Linux e Microsoft). Você pode usar a imagem ISO do Fedora Live CD a partir do disco ou gravar um CD para iniciar uma instalação a partir do host hypervisor escolhido.

O procedimento a seguir guia você pelo processo de instalação do Fedora Live CD descrito no Capítulo 1 para seu computador local.

Atenção

Antes de iniciar, tenha em mente que, por padrão, esse procedimento apaga todos os dados de seu disco rígido. Se quiser impedir que uma área do disco (uma partição) seja substituída, você terá a oportunidade de fazer isso, desde que tenha espaço suficiente para concluir a instalação. Desconecte todas as unidades USB que tiver conectado ao computador, porque elas também serão substituídas.

1. **Obtenha o Fedora.** Escolha a imagem do Fedora Live CD que você deseja usar, transfira-a para seu sistema local e grave-a em um CD, como descrito no Capítulo 1.
2. **Inicialize o Live CD.** Quando você vir a tela de inicialização, selecione Start Fedora.
3. **Inicie a instalação.** Quando a tela do desktop aparecer, posicione o mouse sobre o ícone Install to Hard Drive e selecione-o. A Figura 9.1 mostra um exemplo de ícone Install to Hard Drive no desktop do Fedora Live CD.

4. **Selecione o teclado.** Quando solicitado, escolha o idioma/tipo de teclado que melhor se adapte ao seu sistema (como U.S. English ou Brasil ABNT2) e selecione Next. Você será perguntado sobre os tipos de dispositivos de armazenamento a usar.

ndo o processo de instalação a partir de um Live CD.

5. **Selecione armazenamento básico ou especializado.** Tradicionalmente, o sistema operacional inteiro do computador seria instalado no disco rígido local do computador (Basic Storage Device). Hoje é comum em grandes empresas instalar o sistema operacional também em um ou mais dispositivos de armazenamentos especiais, como redes de área de armazenamento (*storage area networks*, SANs), Fibre Channel over Ethernet, dispositivos de armazenamento de acesso direto (DASDs), dispositivos RAID de firmware ou dispositivos Multipath. Descreverei esses tipos de dispositivos na seção sobre a instalação do Red Hat Enterprise Linux. Para uma instalação simples de desktop ou do servidor pessoal, escolha Basic e selecione Next. Você será solicitado a fornecer seu hostname.
6. **Escolha um nome de máquina.** Digite um nome para seu computador na caixa Hostname. Esse pode ser um hostname simples (`myhost`) ou um nome de domínio totalmente qualificado (`myhost.example.com`). Quando terminar, selecione Next. Você será solicitado a informar seu fuso horário.
7. **Informe o fuso horário.** Escolha um fuso horário para sua máquina no mapa ou na lista mostrada. Então, verifique se o relógio do sistema utiliza Tempo Universal Coordenado (UTC) ou a hora local. Para um desktop, você geralmente escolhe seu fuso horário local. Servidores que são acessados a partir de vários fusos horários costumam configurar a data e a hora com o UTC, também chamado de Greenwich Mean Time (GMT). O site U.S. National Weather Service descreve como diferentes fusos horários se relacionam com UTC (<http://www.nhc.noaa.gov/aboututc.shtml>). Selecione Next para continuar.
8. **Defina a senha de root.** Você será solicitado (duas vezes) a definir a senha inicial de root do seu sistema. A senha deve ter pelo menos seis caracteres. Se a senha for considerada muito fácil (como uma palavra comum), você é informado disso e tem a oportunidade de mudá-la ou usá-la assim mesmo. Uma boa senha inclui uma mistura de letras (maiúsculas e minúsculas), números e sinais de pontuação e tem pelo menos oito caracteres. Selecione Next para continuar.
9. **Selecione o tipo de instalação.** Aqui você escolhe de onde virá o espaço para sua instalação. Se você quiser manter seus dados no disco rígido, é muito importante fazer a seleção certa. Por exemplo, se você tiver um sistema Windows instalado que deseja usar em um arranjo dual-boot, você precisa se certificar de não apagar a partição do Windows. Eis suas opções (junto com as seleções na parte inferior da tela, descritas ao lado):
 - **Use All Space** — Se não há problema em todo o seu disco ser apagado, escolha essa opção. É a seleção mais fácil e limpa, mas só deve ser selecionada se você não se importa de perder tudo que está atualmente instalado no computador.
 - **Replace Existing Linux Systems** — Todas as partições Linux (contendo sistemas de

arquivos Linux ext, partições de swap e semelhantes) são substituídos por essa seleção. Partições comumente usadas com sistemas Windows (VFAT, FAT32, ou NTFS) não são removidas. Use essa seleção se quiser manter um sistema Windows instalado em seu computador e potencialmente fazer dual boot.

- **Shrink Current System** — Para obter algum espaço livre a fim de fazer uma nova instalação, você pode tentar diminuir uma partição existente antes de continuar com a instalação.
- **Use Free Space** — Se não houver espaço livre suficiente em seus discos de sistema que esteja fora das partições existentes, você pode escolher usar esse espaço para sua instalação. Nesse caso, nenhuma partição existente (Windows ou Linux) será apagada ou reutilizada.
- **Create Custom Layout** — Escolha essa opção se, em vez de deixar o instalador particionar seu disco, você mesmo quer fazer isso. Se preferir, em vez dessa seleção, você pode escolher um dos outros tipos de instalação e então selecione a caixa Review and modify partitioning layout (Revise e modifique o layout do particionamento). Isso permite que você veja o que o instalador recomenda para o particionamento de disco, mas permite que você revise e modifique a recomendação, se quiser.

Outras opções nessa tela permitem fazer o seguinte:

- **Use LVM** — Marque essa caixa (ela aparece selecionada por padrão), se você quiser usar o recurso Logical Volume Management (LVM) para criar mais partições de disco. Essa opção acrescenta alguma complexidade ao seu particionamento, mas também oferece maior flexibilidade. Se optar por usar LVM, você tem maior flexibilidade para aumentar e diminuir sistemas de arquivos mais tarde do que se usar partições de disco padrão. Recomendo que você mantenha essa caixa selecionada. Consulte o Capítulo 12, “Gerenciando discos e sistemas de arquivos”, para obter informações sobre a criação de partições LVM.
- **Encrypt system** — Se escolher essa opção, o sistema de arquivos Linux será criptografado, de modo que você só poderá ver o conteúdo das partições se puder digitar uma senha quando o sistema tentar montá-los no momento da inicialização. Você será solicitado a fornecer essa senha, que deve ter pelo menos oito caracteres. Adicionar uma senha aqui protege a partição LVM, mas não protege com senha a partição `/boot`. Você pode escolher criptografar a partição `/boot` ou qualquer um dos volumes lógicos de LVM, se optar por analisar o esquema de particionamento.
- **Review and modify partitioning layout** — Recomendo que você sempre selecione essa caixa para poder ver o esquema de particionamento que está a ponto de habilitar, e alterá-lo, se quiser.

ionando seu disco.

A Figura 9.2 mostra um exemplo da tela de particionamento que aparece depois que você seleciona as opções Use All Space, Use LVM e Review and modify partitioning layout.

- **Modify partitioning** — Você pode modificar o particionamento como quiser. Detalhes sobre como modificar o particionamento a partir dessa tela, bem como a forma como o gerenciador de inicialização GRUB funciona, são descritos mais adiante nesta seção. Por enquanto, examine a Figura 9.2 e veja que há uma unidade de 160MB de disco rígido (`/dev/sda`), que tem três partições físicas (`sda1`, `sda2` e `sda3`). O Fedora usa o GRUB 2, que incorpora duas partições de inicialização: `sda1` é uma pequena partição contendo dados de inicialização brutos do stage2 (os dados de inicialização do GRUB 2 são muito grandes para caber no registro mestre de inicialização do disco, como o GRUB 1 faz) e `sda2` contém dados como o kernel do Linux e informações de configuração que são montadas em `/boot`.

No exemplo ilustrado na Figura 9.2, `sda3` consome todo o espaço restante (mais de 150GB) como um único volume físico LVM. Esse volume é aplicado ao grupo de volumes `vg_myhost`, que é dividido em um volume lógico de 50GB separado, chamado `lv_root`, montado sobre a raiz do sistema de arquivos (`/`) e um sistema de arquivos de 92GB, chamado `lv_home`, montado em `/home`. Também há uma partição de swap de cerca de 500MB. Você pode remover ou modificar as partições conforme descrito mais tarde neste capítulo.

Se tiver partições Windows em seu sistema que você deseja manter, você deve vê-las na tela. Você deve se certificar de que elas *não* sejam selecionadas para serem reformatadas. (Clique duas vezes na entrada e, então, na janela que aparece, desmarque a caixa `Format partition` para que essas partições não sejam formatadas.) O processo de instalação deve deixá-las intactas e elas devem estar disponíveis após a instalação do Linux.

Quando estiver satisfeito, clique em `Next`. Você receberá um prompt informando que o disco está prestes a ser sobrescrito. Essa é sua última chance de voltar atrás antes de seus dados serem sobrescritos no disco. Certifique-se de que o particionamento está como você quer, porque não há como voltar atrás!

10. **Reinicie quando terminar.** Nesse ponto, o disco é reparticionado, os sistemas de arquivos são formatados, a imagem do Linux é copiada para o disco rígido e as configurações necessárias são implementadas. Depois de fazer isso, clique em `Reboot`. Ejete o Live CD. O computador deve inicializar seu sistema Fedora recém-instalado. (Você pode precisar realmente desligar o computador para reinicializar.)
11. **Execute o firstboot.** A primeira vez que você iniciar o novo sistema, uma tela de `firstboot` permite que você faça algumas configurações iniciais. Eis o que você faz:
 - **Welcome** — Clique em `Next` na tela `Welcome` (Bem-Vindo).

- **License** — Leia as informações de licença e clique em Next.
- **Date and Time** — Selecione a data e hora atuais e clique em Forward. Se você deseja se conectar a um servidor NTP para sincronizar o horário com um servidor de data/hora, em vez de defini-lo manualmente, clique na caixa Synchronize date and time e use os servidores padrão do Fedora ou adicione outros servidores de data/hora que você quiser. Clique em Forward para prosseguir.
- **Create User** — É uma boa prática ter pelo menos um usuário regular (não-root) em todos os sistemas, pois o root deve ser utilizado apenas para tarefas administrativas e não para uso rotineiro do computador. Adicionar o nome de usuário completo, o nome de usuário abreviado e a senha. Para permitir que esse usuário faça tarefas administrativas sem saber a senha de root, selecione a caixa Add to Administrators group.
Normalmente, esse usuário é adicionado aos arquivos `passwd` e `shadow` locais para autenticação. Se sua localização utiliza a autenticação de rede central, clique no botão Use Network Login e selecione o tipo e a localização de seu servidor de autenticação. Se você quiser fazer o ajuste fino da nova conta de usuário, clique no botão Advanced. Isso permite que você use a janela User Manager para configurar o usuário (consulte o Capítulo 11 para detalhes sobre como usar essa janela para gerenciar contas de usuário).

Clique em Forward para continuar.

12. **Send hardware profile.** Para ajudar o projeto Fedora a saber o hardware em que o sistema deles está sendo instalado, você pode optar por enviar o perfil anonimamente para o projeto Fedora. Selecione Finish depois de fazer essa seleção.
13. **Logue-se e comece a usar o Fedora.** A tela de login aparece nesse ponto, permitindo que você faça login com a conta de usuário e a senha que você acabou de criar.
14. **Atualize os softwares.** Para manter seu sistema seguro e atualizado, uma das primeiras tarefas que você deve fazer depois de instalar o Fedora é obter as últimas versões do software que acabou de instalar. Se seu computador tiver uma conexão com a internet (conectar-se a uma rede com fio Ethernet ou selecionar uma rede sem fio acessível a partir do desktop resolve isso), você pode simplesmente abrir um terminal como root e digitar **yum update** para baixar e atualizar todos os pacotes a partir da internet. Se um novo kernel for instalado, você pode reiniciar seu computador para fazê-lo funcionar.

Nesse ponto, você pode começar a usar o desktop, conforme descrito no Capítulo 2. Você também pode usar o sistema para fazer os exercícios de qualquer um dos capítulos deste livro.

Instalando o Red Hat Enterprise Linux a partir de Mídia de Instalação

Além de oferecer um Live CD, a maioria das distribuições Linux oferece uma única imagem ou um conjunto de imagens que podem ser usadas para instalar a distribuição. Em vez de oferecer uma única imagem nessa mídia que é copiada para o disco rígido, o software é dividido em pacotes que você pode

selecionar para atender às suas necessidades. Um DVD de instalação completo, por exemplo, pode permitir que você instale qualquer coisa a partir de um servidor mínimo para um desktop completo.

Neste capítulo, eu uso um DVD de instalação Red Hat Enterprise Linux 6.2 como mídia de instalação. Reveja as informações de hardware e as descrições de dual boot na seção anterior antes de iniciar a instalação do RHEL.

Siga este procedimento para instalar o Red Hat Enterprise Linux a partir de um DVD de instalação:

1. **Obtenha a mídia de instalação.** O processo de download de imagens ISO de instalação do RHEL é descrito na página How to Download Red Hat Installation Files (Como baixar arquivos de instalação do Red Hat). Se ainda não é um cliente da Red Hat, você pode solicitar uma cópia de avaliação aqui:

<https://www.redhat.com/products/enterprise-linux/server/download.html>.

Isso requer que você crie uma conta Red Hat. Se isso não for possível, você pode baixar um DVD de instalação a partir de um site espelho do projeto CentOS para obter uma experiência semelhante:

<http://www.centos.org/modules/tinycontent/index.php?id=30>.

2. **Inicialize a mídia de instalação.** Insira o DVD no drive de DVD e reinicie seu computador. A tela Welcome aparece.
3. **Execute uma nova instalação.** Selecione a entrada Install or upgrade an existing system (Instale ou atualize um sistema existente) para fazer uma nova instalação do RHEL. Se precisar modificar o processo de instalação, você pode adicionar opções de inicialização pressionando a tecla Tab com uma entrada de inicialização em destaque e digitando as opções desejadas. Consulte o “Usando opções de inicialização de instalação”, mais adiante neste capítulo.
4. **Teste a mídia.** Você será solicitado a testar o disco de instalação do DVD quanto a erros. Você pode selecionar OK (para testá-lo) ou Skip (para pular o teste).
5. **Veja a página Welcome.** Selecione Next para continuar. Você será solicitado a selecionar o idioma.
6. **Selecione um idioma.** Selecione seu idioma e clique em Next. Você é convidado a identificar o tipo de teclado.
7. **Identifique o teclado.** Identifique o tipo de layout de teclado que você está usando (com base no idioma e país) e clique em Next. Você será solicitado a selecionar armazenamento básico ou especializado.
8. **Selecione o armazenamento.** Escolha Basic Storage para ter o RHEL instalado em seu disco rígido local ou Specialized Storage para usar um dispositivo de armazenamento de rede e clique em Next. Consulte “Utilizando armazenamento especializado”, mais adiante neste capítulo, para obter detalhes sobre os tipos de armazenamento de rede que estão disponíveis e o que informar se você selecionar Specialized Storage.
9. **Escolha um nome de máquina.** Digite um nome para seu computador na caixa Hostname.

Pode ser um hostname simples (`myhost`) ou um nome de domínio totalmente qualificado (`myhost.example.com`). Quando terminar, selecione Next. Você será solicitado a informar seu fuso horário.

10. **Informe o fuso horário.** Escolha um fuso horário para sua máquina no mapa ou na lista apresentada (como descrito na seção Fedora). Selecione Next para continuar.
11. **Defina a senha de root.** Você será solicitado (duas vezes) para definir a senha inicial de root de seu sistema (como descrito na seção Fedora). Selecione Next para continuar.
12. **Selecione o tipo de instalação.** Escolha o tipo de instalação e selecione Review and modify partitioning layout (como descrito na seção “Instalação do Fedora”). O particionamento que recebi por padrão é mostrado na Figura 9.3.

ionando durante a instalação do RHEL

A principal diferença entre as partições do disco na Figura 9.3 e o particionamento do Fedora é que RHEL 6.2 ainda usa o GRUB 1, enquanto que o Fedora usa o GRUB 2 (que exige uma pequena partição adicional para conter os dados de inicialização brutos estendidos em `sda1`). Para alterar esse particionamento, consulte o “Particionando discos rígidos”, mais adiante neste capítulo. Lembre-se de que depois que você selecionar salvar o novo particionamento, é tarde demais para recuperar dados das partições que você escolheu reformatar.

13. **Modifique o gerenciador de inicialização.** O gerenciador de inicialização GRUB é instalado no registro mestre de inicialização (*master boot record*, MBR) do primeiro disco em seu sistema (`/dev/sda`) por padrão. Você pode definir uma senha no gerenciador de inicialização (selecione a caixa Use a boot loader password e altere a senha), que faz com que você seja perguntado se deseja alterar as opções durante a inicialização. A Figura 9.4 mostra essa tela depois de você escolher adicionar um sistema Windows que está instalado no segundo disco (`/dev/sdb1`) ao menu de inicialização. Quando o sistema é inicializado, você será capaz de selecionar para iniciar o Linux (`/dev/sda1`) ou o Windows (`/dev/sdb1`).

onando o Windows ao gerenciador de inicialização para habilitar a inicialização dual

O RHEL 6.2 usa o GRUB 1, que é criado principalmente no arquivo `/boot/grub/grub.conf`. Consulte “Utilizando o gerenciador de inicialização GRUB”, mais adiante neste capítulo, para obter mais informações sobre o GRUB.

14. **Escolha pacotes de software.** Ao contrário dos Live CDs, você pode selecionar exatamente quais pacotes instalar a partir do DVD de instalação do RHEL. Você começa selecionando um grupo de pacotes (servidor básico, servidor de banco de dados, servidor web, servidor de gerenciamento de identidades, host de virtualização, desktop, workstation de desenvolvimento de software, ou mínimo).

Sugiro que você comece com uma instalação desktop e depois adicione pacotes do servidor à medida que precisar testá-los. Para refinar sua instalação, selecione o botão **Customize Now** na parte inferior da página para ver uma página que lhe permite escolher grupos de softwares adicionais e pacotes de software individuais.

Você pode conectar-se a repositórios de software adicionais, descritos no Capítulo 10, “Obtendo e gerenciando software”, que estão fora da distribuição RHEL para poder instalar pacotes de software adicionais. Depois de ter feito suas seleções, clique em **Next** para continuar. Depois que as dependências de software foram resolvidas, o instalador começa a instalar os pacotes.

15. **Conclua a instalação.** Quando a instalação estiver concluída, clique em **Reboot**. Ejete o DVD quando o sistema for reiniciado e o Red Hat Enterprise Linux iniciará a partir do disco rígido.
16. **Execute o firstboot.** Se você instalou uma interface de desktop, a tela **firstboot** aparece na primeira vez que você inicializa o sistema. Eis o que você faz:
 - **Welcome** — Clique em **Next** na tela **Welcome**.
 - **License** — Leia e concorde com as informações de licença e clique em **Next**.
 - **Set Up Software Updates** — Conforme descrito no Capítulo 10, para obter downloads e atualizações de software, você deve se inscrever no Red Hat Enterprise Linux. Essa tela descreve como se inscrever para atualizações de software.
 - **Create User** — Crie uma conta de usuário regular e clique em **Forward** para continuar. Você também tem a oportunidade de usar bancos de dados de autenticação de rede ou refinar as configurações de conta de usuário.
 - **Date and Time** — Selecione a data e hora e clique em **Next**. Se você deseja se conectar a um servidor NTP para sincronizar o horário com um servidor de data/hora, em vez de defini-lo manualmente, selecione a caixa **Synchronize date and time** e use os servidores padrão do Fedora ou adicione seus próprios. Clique em **Forward** para prosseguir.

- **y=** — Você pode escolher reservar uma certa quantidade de memória RAM para o recurso `kdump`. Se `kdump` estiver habilitado, a RAM reservada pode ser usada no caso de seu kernel travar para ter um lugar em que o dump do kernel possa ser armazenado. Sem `kdump`, não haveria maneira de diagnosticar um kernel travado.

Agora você deve ser capaz de entrar em seu sistema Red Hat Enterprise Linux. Uma das primeiras coisas que você deve fazer é registrar seu sistema na Red Hat Network e atualizar seu software. Consulte o Capítulo 10 para obter informações sobre como obter atualizações de software.

Instalando o Linux na Empresa

Se você for administrar dezenas, centenas ou até milhares de sistemas Linux em uma grande empresa, seria terrivelmente ineficiente ter de ir em cada computador para digitar e clicar em cada instalação. Felizmente, com o Red Hat Enterprise Linux e outras distribuições, você pode automatizar a instalação de tal forma que tudo o que você precisa fazer é ligar um computador e inicializar a partir da placa de rede dele para obter sua instalação Linux desejada.

Embora tenhamos focado a instalação do Linux a partir de um CD ou DVD, existem muitas outras maneiras de carregar uma instalação do Linux e muitas maneiras de completá-la. A lista de itens a seguir guia você pelo processo de instalação e descreve maneiras de mudar esse processo ao longo do caminho:

- **Carregue a mídia de instalação.** Você pode iniciar uma instalação a partir de qualquer mídia que possa ser inicializada a partir de um computador: CD, DVD, USB, disco rígido ou placa de rede com suporte PXE. O computador percorre sua ordem de inicialização e olha para o registro mestre de inicialização na mídia física ou procura um servidor PXE na rede.
- **Iniciar o kernel anaconda.** O trabalho do gerenciador de inicialização é apontar para o kernel especial (e, possivelmente, um disco de RAM inicial) que inicia o instalador Linux (chamado anaconda). Assim, qualquer uma das mídias descritas precisa simplesmente apontar para a localização do kernel e o disco de RAM inicial a fim de iniciar a instalação. Se os pacotes de software não estiverem na mesma mídia, o processo de instalação lhe perguntará onde obter os pacotes.
- **Adicione o kickstart ou outras opções de inicialização.** Opções de inicialização (descritas mais tarde neste capítulo) podem ser passadas para o kernel anaconda a fim de configurar como ele inicia. Uma opção suportada pelo Fedora e o RHEL permite que você passe a localização de um arquivo de kickstart para o instalador. O kickstart pode conter todas as informações necessárias para concluir a instalação: senha de root, particionamento, fuso horário etc., para configurar melhor o sistema instalado. Assim que o instalador iniciar, ele solicita as informações necessárias ou usa as respostas fornecidas no arquivo de kickstart.
- **Encontre pacotes de software.** Pacotes de software não precisam estar na mídia de instalação. Isso permite que você inicie uma instalação a partir de uma mídia de inicialização que contém apenas um kernel e o disco de RAM inicial. A partir do arquivo de kickstart ou de uma opção que você digita manualmente para o instalador, é possível identificar a localização do repositório onde estão os pacotes de software RPM. Essa localização pode ser um CD local (cdrom), site

(http), site de FTP (FTP), compartilhamento NFS (NFS), NFS ISO (nfsiso) ou disco local (HD).

- **Modifique a instalação com scripts de kickstart.** Scripts incluídos em um kickstart podem executar comandos que você escolhe, antes ou após a instalação, para configurar melhor o sistema Linux. Esses comandos podem adicionar usuários, alterar permissões, criar arquivos e diretórios, baixar arquivos da rede ou configurar o sistema instalado exatamente como você especificar.

Embora a instalação do Linux em ambientes corporativos esteja além do escopo deste livro, quero que você entenda as tecnologias que estão disponíveis quando quiser automatizar o processo de instalação do Linux. Eis algumas dessas tecnologias disponíveis para uso com o Red Hat Enterprise Linux, junto com links de onde você pode encontrar mais informações sobre eles:

- **Instale o servidor** — Se configurar um servidor de instalação, você não precisa transportar os pacotes de software para cada máquina em que instalar o RHEL. Essencialmente, você copia todos os pacotes de software a partir da mídia de instalação do RHEL para um servidor web (http), servidor FTP (FTP), ou servidor NFS (NFS) e aponta para a localização do servidor ao iniciar o instalador. O RHEL Installation Guide descreve como configurar um servidor de instalação (http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/htmlsingle/Installation_Guide/index.html#s1-steps-network-installs-x86).
- **Servidor PXE** — Se tiver um computador com uma placa de rede que suporta o PXE (como a maioria suporta), você pode configurar a BIOS de seu computador para inicializar a partir dessa placa de rede. Se você configurar um servidor PXE nessa rede, esse servidor pode apresentar um menu para o computador contendo as entradas para carregar um processo de instalação. O RHEL Installation Guide fornece informações sobre como configurar servidores PXE para instalação (http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/htmlsingle/Installation_Guide/index.html#ap-installserver).
- **Arquivos de kickstart** — Para automatizar totalmente a instalação, você cria o que é chamado de arquivo de kickstart. Ao passar um arquivo de kickstart como uma opção de inicialização para um instalador de Linux, você pode dar respostas a todas as perguntas de instalação que normalmente precisa responder. Ao instalar o RHEL, um arquivo de kickstart contendo respostas para todas as suas dúvidas sobre a instalação que você acabou de fazer está contido no arquivo `/root/anaconda-ks.cfg`. Você pode fornecer o arquivo para sua próxima instalação a fim de repetir a configuração de instalação ou usar esse arquivo como um modelo para diferentes instalações.
Consulte o RHEL Installation Guide para obter informações sobre como passar um arquivo de kickstart para o instalador do anaconda (http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/htmlsingle/Installation_Guide/index.html#ch-parmfiles-Kickstart_parameters) e sobre como criar seus próprios arquivos de kickstart (http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/htmlsingle/Installation_Guide/index.html#chkickstart2).

Explorando Tópicos Comuns Sobre a Instalação

Alguns tópicos sobre a instalação abordados anteriormente neste capítulo exigem maiores explicações para você poder aplicá-los completamente. Leia os tópicos desta seção para entender melhor temas especificamente relacionados com a instalação.

Atualizando ou instalando a partir do zero

Se você tiver uma versão anterior do Linux já instalada em seu computador, Fedora, RHEL, Ubuntu e outras distribuições Linux oferecem uma opção de atualização. Isso permite que você atualize todos os pacotes, por exemplo, de uma versão 1 da distribuição para a versão 2. Eis algumas regras gerais antes de fazer uma atualização:

- **Remova pacotes extras.** Se tiver pacotes de software de que você não precisa, remova-os antes de fazer uma atualização. Processos de atualização normalmente atualizam somente os pacotes que estão em seu sistema. Atualizações, em geral, fazem mais verificações e comparações do que instalações limpas; por isso, remover qualquer pacote pode economizar tempo no processo de atualização.
- **Verifique os arquivos de configuração.** Um procedimento de atualização do Linux costuma deixar cópias de arquivos de configuração antigos. Você deve verificar se os novos arquivos de configuração ainda funcionam para você.

ca

Instalar o Linux a partir do zero é mais rápido do que fazer uma atualização. Também resulta em um sistema Linux limpo. Portanto, se você não precisa dos dados em seu sistema (ou se você tiver um backup), recomendo que faça uma nova instalação. Daí, você pode restaurar seus dados para um sistema recém-instalado.

Algumas distribuições Linux, principalmente o Gentoo, adotaram a abordagem de fornecer atualizações contínuas. Em vez de obter uma nova versão a cada poucos meses, você simplesmente adquire pacotes atualizados continuamente à medida que se tornam disponíveis e os instala em seu sistema.

Inicialização dual

Ter vários sistemas operacionais inicializáveis no mesmo computador é possível. Você pode fazer isso usando múltiplas partições em um disco rígido e/ou vários discos rígidos e, então, instalando sistemas operacionais diferentes em partições diferentes. Desde que o gerenciador de inicialização contenha informações de inicialização para cada um dos sistemas operacionais instalados, você será capaz de escolher qual deles executar no momento da inicialização.

enção

Atualmente as ferramentas para redimensionar partições do Windows e configurar sistemas multi-boot tenham melhorado

Últimos anos, ainda há algum risco de perda de dados em sistemas de dual-boot Windows/Linux. Diferentes sistemas operacionais muitas vezes têm diferentes visões de tabelas de partição e registros mestres de inicialização, e pode fazer com que sua máquina não inicie (pelo menos temporariamente) ou causar a perda permanente de dados. Sempre faça backup de seus dados antes de tentar redimensionar um sistema de arquivos Windows (NTFS ou FAT) para criar espaço para o Linux.

Se o computador que você está usando já tem um sistema Windows nele, muito possivelmente o disco rígido inteiro é dedicado ao Windows. Embora você possa executar um Linux de boot, como o Knoppix ou o Damn Small Linux, sem tocar no disco rígido, para fazer uma instalação mais permanente você vai querer encontrar espaço em disco fora da instalação do Windows. Há algumas maneiras de fazer isso:

- **Adicione um disco rígido.** Em vez de mexer com sua partição do Windows, você pode simplesmente adicionar um disco rígido e dedicá-lo ao Linux.
- **Redimensione a partição do Windows.** Se tiver espaço disponível em sua partição Windows, você pode diminuir a partição a fim de disponibilizar espaço em disco para dedicar ao Linux. Ferramentas comerciais como o Partition Magic da Symantec (<http://www.symantec.com>) ou o Acronis Disk Director (<http://www.acronis.com>) estão disponíveis para redimensionar partições de disco e configurar um gerenciador de inicialização viável. Algumas distribuições Linux (particularmente sistemas inicializáveis usados como CDs de recuperação) incluem uma ferramenta chamada GParted, que é um clone de código-fonte aberto do Partition Magic (que inclui software do projeto Linux-NTFS para redimensionar partições NTFS do Windows).

ta

Use `yum install gparted` (no Fedora ou no RHEL) ou `apt-get install gparted` (no Ubuntu) para instalar o GParted. Execute `gparted` como root para iniciá-lo.

Antes de tentar redimensionar a partição do Windows, talvez você precise desfragmentá-la. Para desfragmentar o disco em alguns sistemas Windows, de forma que todo o espaço usado seja colocado em ordem no disco, abra Meu Computador, clique com botão direito no ícone do disco rígido (normalmente C:), selecione Propriedades, clique em Ferramentas e selecione Desfragmentar Agora.

A desfragmentação do seu disco pode ser um processo relativamente longo. O resultado da desfragmentação é que todos os dados em seu disco são contíguos, criando uma grande quantidade de espaço livre contíguo no fim da partição. Há casos em que você terá de realizar as seguintes tarefas especiais para tornar isso verdade:

- Se o arquivo de troca do Windows não for movido durante a desfragmentação, você deverá removê-lo. Então, depois de desfragmentar o disco novamente e redimensioná-lo, você precisará restaurar o arquivo de swap. Para remover o arquivo de troca, abra o Painel de controle, abra o ícone de Sistema e então clique na guia Desempenho e selecione Memória virtual. Para desativar o arquivo de troca, clique em Desativar a memória virtual.
- Se sua partição DOS tiver arquivos ocultos que estão no espaço que você está tentando liberar,

você precisa localizá-los. Em alguns casos, você não será capaz de excluí-los. Em outros casos, como os arquivos de troca criados por um programa, você seguramente pode excluir esses arquivos. Isso é um pouco difícil porque alguns arquivos não deveriam ser excluídos, como os arquivos de sistema do DOS. Você pode usar o comando `attrib -s -h` do diretório-raiz para lidar com arquivos ocultos.

Depois que seu disco estiver desfragmentado, você pode usar ferramentas comerciais descritas anteriormente (Partition Magic ou Acronis Disk Director) para reparticionar seu disco rígido a fim de liberar espaço para o Linux. Ou use a alternativa de código-fonte aberto GParted.

Depois de limpar espaço em disco suficiente para instalar o Linux (ver os requisitos de espaço em disco descritos anteriormente neste capítulo), você pode instalar o Fedora, o RHEL ou outra distribuição Linux. Ao configurar seu gerenciador de inicialização durante a instalação, você será capaz de identificar as partições de boot do Windows, do Linux e de quaisquer outras partições de boot e, assim, poderá selecionar qual inicializar ao ligar o computador.

Instalando o Linux para executar virtualmente

Usando a tecnologia de virtualização, como o KVM, VMWare, VirtualBox ou Xen, você pode configurar o computador para executar vários sistemas operacionais simultaneamente. Normalmente, você tem um host rodando o sistema operacional (como seu desktop Linux ou Windows) e configura sistemas operacionais convidados para executar nesse ambiente.

Se tiver um sistema Windows, você pode usar os produtos comerciais VMWare para rodar o Linux em seu desktop Windows. Obtenha uma versão de demonstração do VMWare Workstation (<http://www.vmware.com/tryvmware>) para ver se lhe agrada. Então, execute seus convidados virtuais instalados com o VMWare Player livre. Com uma versão completa do VMware Workstation, você pode executar múltiplas distribuições ao mesmo tempo.

Produtos de virtualização de código-fonte aberto que estão disponíveis com os sistemas Linux incluem VirtualBox (<http://www.virtualbox.org>), Xen (<http://www.xen.org>) e KVM (<http://www.linux-kvm.org>). O VirtualBox foi desenvolvido originalmente pela Sun Microsystems. O Xen tem sido popular junto aos sistemas da Red Hat, mas a Red Hat recentemente começou a se direcionar para a tecnologia KVM.

Usando opções de inicialização da instalação

Quando o kernel anaconda é carregado no momento da inicialização do RHEL ou do Fedora, as opções de inicialização fornecidas na linha de comando do kernel modificam o comportamento do processo de instalação. Interrompendo o gerenciador de inicialização antes de o kernel de instalação inicializar, você pode adicionar suas próprias opções de inicialização para controlar como a instalação se comporta.

Quando vir a tela de inicialização de instalação, dependendo do gerenciador de inicialização, pressione Tab ou alguma outra tecla para poder editar a linha de comando do kernel anaconda. A linha de identificação do kernel poderia ser algo como o seguinte:

```
vmlinuz initrd=initrd.img
```

O `vmlinuz` é o kernel compactado e `initrd.img` é o disco de RAM inicial (contendo módulos e outras ferramentas necessárias para iniciar o instalador). Para adicionar mais opções, basta digitá-las no final da linha e pressionar Enter.

Assim, por exemplo, se você tiver um arquivo de kickstart disponível a partir de `/root/ks.cfg` em um CD, o prompt de inicialização do anaconda para iniciar a instalação usando o arquivo de kickstart pode se parecer com o seguinte:

```
vmlinuz initrd=initrd.img ks=cdrom:/root/ks.cfg
```

As seções a seguir descrevem outras opções de inicialização do instalador que você pode achar úteis.

Opções de inicialização para desabilitar recursos

Às vezes, uma instalação do Linux falha porque o computador tem algum hardware que não está funcionando ou que não é suportado. Muitas vezes você pode contornar essas questões, passando opções para o instalador fazer coisas como desabilitar hardware selecionado quando você precisa selecionar seu próprio driver. A Tabela 9.1 fornece alguns exemplos:

Tabela 9.1 Opções de Inicialização para Desabilitar Recursos

Opção do instalador	Diz ao sistema
<code>firewire</code>	Não carregar suporte para dispositivos firewire
<code>dma</code>	Não carregar suporte DMA para discos rígidos
<code>ide</code>	Não carregar suporte para dispositivos IDE
<code>mpath</code>	Não habilitar o suporte para dispositivos multipath
<code>parport</code>	Não carregar suporte para portas paralelas
<code>pcmcia</code>	Não carregar suporte para controladores PCMCIA
<code>probe</code>	Não investigar hardware, em vez disso, solicite os drivers ao usuário
<code>scsi</code>	Não carregar suporte para dispositivos SCSI
<code>usb</code>	Não carregar suporte para dispositivos USB
<code>ipv6</code>	Não habilitar a rede IPV6
<code>net</code>	Não investigar dispositivos de rede
<code>numa=off</code>	Desativar o acesso à memória não uniforme (NUMA) para arquitetura AMD64
<code>acpi=off</code>	Desativar a Interface de Energia e Configuração Avançada (ACPI)

Opções de inicialização para problemas de vídeo

Se estiver tendo problemas com o monitor, você pode especificar as configurações de vídeo como observado na Tabela 9.2.

TABELA 9.2 Opções de Inicialização para Problemas de Vídeo

Opção de inicialização	Diz ao sistema
<code>vesa</code>	Usar o driver de vídeo padrão vesa
<code>resolution=1024x768</code>	Escolher a resolução exata para usar
<code>fb</code>	Não use o driver de <i>framebuffer</i> VGA 16
<code>ipddc</code>	Não investigar o DDC do monitor (a investigação pode travar o instalador)
<code>aphical</code>	Forçar uma instalação gráfica

Opções de inicialização para tipos especiais de instalação

Por padrão, a instalação é executada em modo gráfico com você sentado à frente do console respondendo perguntas. Se tiver um console somente texto, ou se a interface gráfica não estiver funcionando corretamente, você pode executar uma instalação no modo texto simples: ao digitar **text**, você faz com que a instalação seja executada em modo texto.

Se quiser começar a instalação em um computador, mas quiser responder as perguntas de instalação a partir de outro computador, você pode habilitar uma instalação vnc (Virtual Network Computing). Depois de começar esse tipo de instalação, você pode ir para outro sistema e abrir um vnc viewer, dando ao visualizador o endereço da máquina de instalação (como 192.168.0.99:1). A Tabela 9.3 fornece os comandos necessários, junto com o que dizer ao sistema para fazer.

TABELA 9.3 Opções de Inicialização para Instalações VNC

Opção de inicialização	Diz ao sistema
<code>c</code>	Execute a instalação como um servidor VNC.
<code>cconnect=hostname[:porta]</code>	Conecte-se ao hostname do cliente VNC e porta opcional.
<code>cpassword=<senha></code>	O cliente usa senha (no mínimo 8 caracteres) para conectar-se ao instalador.

Opções de inicialização para kickstarts e repositórios remotos

Você pode iniciar o processo de instalação a partir de uma mídia de instalação que contém pouco mais do que o kernel e o disco de RAM inicial. Se for esse o caso, você precisa identificar o repositório onde os pacotes de software residem. Você pode fazer isso por meio de um arquivo de kickstart ou identificando a localização dos repositórios de alguma maneira. Para forçar o instalador a solicitar o local do repositório (CD/DVD, disco rígido, NFS, ou URL), adicione `askmethod` às opções de inicialização de instalação.

Usando opções `repo=`, você pode identificar locais de repositório de software. Os exemplos a seguir mostram a sintaxe que deve ser utilizada para criar entradas `repo=`:

```
repo=hd:/dev/sda1:/myrepo
```

Repositório em */myrepo* na primeira partição do disco 1

```
repo=http://abc.example.com/myrepo
```

Repositório disponível a partir de /myrepo no servidor web.

`repo=ftp://ftp.example.com/myrepo`

Repositório disponível a partir de /myrepo no servidor server.

`repo=cdrom`

Repositório disponível a partir do CD or DVD local.

`repo=nfs::my nfs.example.com:/myrepo/`

Repositório disponível a partir de /myrepo no compartilhamento NFS.

`repo=nfsiso::nfs.example.com:/mydir/rhel6.iso`

Instalação da imagem ISO disponível a partir do servidor NFS.

Em vez de identificar o repositório diretamente, você pode especificá-lo dentro de um arquivo de kickstart. A seguir, são apresentados exemplos de algumas maneiras de identificar a localização de um arquivo de kickstart.

`ks=cdrom:/stuff/ks.cfg`

Obtenha o kickstart no CD/DVD.

`ks=hd:sda2:/test/ks.cfg`

Obtenha o kickstart no diretório de teste do disco rígido (sda2)

`ks=http://www.example.com/ksfiles/ks.cfg`

Obtenha o kickstart em um servidor web.

`ks=ftp://ftp.example.com/allks/ks.cfg`

Obtenha o kickstart em um servidor FTP.

`ks=nfs:my nfs.example.com:/someks/ks.cfg`

Obtenha o kickstart em um servidor NFS.

Opções de inicialização diversas

Aqui são apresentadas algumas outras opções que você pode passar para o instalador que não se encaixam em uma categoria.

Rescue

Em vez de instalar, executa o kernel para abrir o modo de recuperação do Linux.

mediacheck

Verifica se o CD/DVD de instalação contém erros de soma de verificação.

Para mais informações sobre como usar o instalador anaconda no modo de recuperação (para recuperar um sistema Linux corrompido), consulte o Capítulo 21, “Solucionando problemas do Linux”.

Usando armazenamento especializado

Em ambientes de computação de grandes empresas, é comum armazenar o sistema operacional e os dados fora do computador local. Em vez disso, alguns dispositivos de armazenamento especial, além do disco

rígido, são identificados para o instalador e podem ser utilizados durante a instalação.

Uma vez identificados, os dispositivos de armazenamento que você indicar durante a instalação podem ser usados da mesma maneira que os discos locais são usados. Você pode particioná-los e atribuir uma estrutura (sistema de arquivos, espaço de swap etc.) ou deixá-los intactos e simplesmente montá-los onde quer que os dados estejam disponíveis.

Os seguintes tipos de dispositivos de armazenamento especializados podem ser selecionados a partir da tela Specialized Storage Devices quando você instala o Red Hat Enterprise Linux, o Fedora ou outras distribuições Linux:

- **Firmware RAID** — Um dispositivo RAID de firmware é um tipo de dispositivo que tem ganchos na BIOS, permitindo que ele seja usado para inicializar o sistema operacional, se você quiser.
- **Dispositivos multipath** — Como o nome indica, os dispositivos multipath fornecem vários caminhos entre o computador e os dispositivos de armazenamento. Esses caminhos são agregados, assim esses dispositivos parecem ser um só para o sistema usar, enquanto a tecnologia subjacente oferece melhor desempenho, redundância, ou ambos. As conexões podem ser fornecidas por dispositivos iSCSI ou Fibre Channel over Ethernet (FCoE).
- **Outros dispositivos SAN** — Qualquer dispositivo que representa uma Storage Area Network (SAN).

Embora a configuração desses dispositivos de armazenamento especializados esteja além do escopo deste livro, saiba que se você estiver trabalhando em uma empresa onde dispositivos iSCSI e FCoE estão disponíveis, você pode configurar o sistema Linux para usá-los no momento da instalação. Os tipos de informação que você precisa para fazer isso incluem:

- **Dispositivos iSCSI** — Peça para seu administrador de armazenamento lhe fornecer o endereço IP de destino do dispositivo iSCSI e do tipo de autenticação de descoberta necessário para usar o dispositivo. O dispositivo iSCSI pode exigir credenciais.
- **Fibre Channel over Ethernet Devices (FcoE)** — Para FCoE, você precisa conhecer a interface de rede que está conectada ao switch FCoE. Você pode procurar essa interface para os dispositivos FCoE disponíveis.

Particionando discos rígidos

O(s) disco(s) rígido(s) em seu computador oferece(m) a área de armazenamento permanente para seus arquivos de dados, programas aplicativos e o sistema operacional em si. O particionamento é o ato de dividir um disco em áreas lógicas que podem ser trabalhadas separadamente. No Windows, você normalmente tem uma partição que consome todo o disco rígido. Mas, com o Linux, há várias razões pelas quais você pode querer ter múltiplas partições:

- **Vários sistemas operacionais** — Se instalar o Linux em um PC que já tem um sistema operacional Windows, você pode querer manter os dois sistemas operacionais no computador. Para todos os efeitos práticos, cada sistema operacional deve existir em uma partição completamente separada. Quando seu computador reinicializa, você pode escolher qual

sistema executar.

- Várias partições dentro de um sistema operacional — Para evitar que o sistema operacional inteiro fique sem espaço em disco, as pessoas costumam atribuir partições separadas para diferentes áreas do sistema de arquivos Linux. Por exemplo, se forem atribuídas partições separadas a `/home` e `/var`, um usuário glutão que enche a partição `/home` não impediria que daemons de registro em log continuassem a gravar arquivos de log no diretório `/var/log`.

O fato de ter várias partições também facilita certos tipos de backups (como um backup de imagem). Por exemplo, um backup de imagem `/home` seria muito mais rápido (e, provavelmente, mais útil) do que um backup de imagem do sistema de arquivos raiz (`/`).

- Tipos de sistemas de arquivos diferentes — Diferentes tipos de sistemas de arquivos têm estruturas diferentes. Sistemas de arquivos de tipos diferentes devem estar em suas próprias partições. Além disso, você pode precisar de sistemas de arquivos diferentes para ter opções de montagens diferentes para recursos especiais (tais como marcar como somente leitura ou definir quotas de usuário). Na maioria dos sistemas Linux, você precisa de pelo menos um tipo de sistema de arquivos para a raiz do sistema de arquivos (`/`) e um para a área de swap. Sistemas de arquivos em CDROM usam o tipo de sistema de arquivos `iso9660`.

ca

criar partições para o Linux, normalmente você vai atribuir o tipo de sistema de arquivos como o Linux nativo (quando o tipo `ext2`, `ext3` ou `ext4` na maioria dos sistemas Linux). Se os aplicativos que estão em execução exigirem recursos particularmente longos, tamanhos de arquivos grandes, ou muitos inodes (cada arquivo consome um inode), você pode querer escolher um tipo de sistema de arquivo diferente.

Por exemplo, se você configurar um servidor de notícias, ele pode usar muitos inodes para armazenar pequenos arquivos de news. Outra razão para usar um tipo de sistema de arquivos diferente é copiar uma fita de backup de imagem de outro sistema operacional para seu disco local (como um sistema de arquivos herdado de um sistema operacional OS/2 ou Minix).

Quando do Windows

Se você usou somente sistemas operacionais Windows antes, provavelmente teve seu disco rígido inteiro atribuído a `C:` e nunca pensou sobre partições. Com muitos sistemas Linux, você tem a oportunidade de visualizar e alterar o particionamento padrão com base em como deseja usar o sistema.

Durante a instalação, sistemas como Fedora e RHEL deixam você particionar seu disco rígido usando ferramentas gráficas de particionamento. As próximas seções descrevem como particionar seu disco durante uma instalação do Fedora. Consulte a seção “Dicas para criar partições” para ter algumas ideias sobre como criar partições de disco.

Entendendo os diferentes tipos de partições

Muitas distribuições Linux dão a opção de selecionar diferentes tipos de partições ao particionar o disco rígido durante a instalação. Tipos de partição incluem:

- Partições Linux — Use esta opção para criar uma partição para tipos de sistema de arquivos ext2, ext3, ext4 que é adicionada diretamente a uma partição no disco rígido (ou outra mídia de armazenamento).
- Partições LVM — Crie uma partição LVM se planeja criar ou adicionar a um grupo de volumes LVM. LVMs proporcionam mais flexibilidade para, mais tarde, aumentar, diminuir e mover partições do que as partições normais.
- Partições RAID — Crie duas ou mais partições RAID para criar um array RAID. Essas partições devem estar em discos separados para criar uma efetiva array RAID. Arrays RAID podem ajudar a melhorar o desempenho, confiabilidade, ou ambos uma vez que essas características se relacionam com ler, gravar e armazenar seus dados.
- Partições de swap — Crie uma partição de swap para estender a quantidade de memória virtual disponível em seu sistema.

As próximas seções descrevem como adicionar partições comuns Linux, LVM, RAID e partições de swap usando o instalador gráfico do Fedora. Se você ainda não tem certeza de quando deve usar esses diferentes tipos de partição, consulte o Capítulo 12 para obter mais informações sobre a configuração de partições de disco.

Particionando durante a instalação do Fedora

Durante a instalação, o Fedora lhe dá a oportunidade de mudar a forma como o disco rígido é particionado. Se optar por fazer um layout personalizado (ou revisar e modificar o particionamento atual), você tem a oportunidade de personalizar seu particionamento de disco. A partir dos discos rígidos que aparecem na tela do instalador, selecione Free (para criar uma nova partição usando o espaço disponível no disco) ou exclua uma ou mais partições, e então selecione Free para reutilizar esse espaço.

As próximas seções descrevem como criar diferentes tipos de partição.

Partições Linux ou partições de swap

Criar uma partição Linux padrão permite que você crie a partição e aplique o sistema de arquivos diretamente a ela. Selecione Create ⇒ Standard Partition e clique no botão Create. A Figura 9.5 mostra um exemplo da tela que aparece. Escolha o ponto de montagem para a partição (como / ou /boot), o tipo de sistema de arquivos (ext4 é o padrão) e o tamanho (em MB). A partição pode ter um tamanho fixo ou simplesmente ocupar todo o espaço disponível.

uma partição Linux normal no Fedora.

Para criar uma partição de swap, você pode seguir os passos utilizados para criar uma partição Linux. Mas em vez de selecionar um tipo de sistema de arquivos (ext3, ext4 etc.), você escolhe swap como tipo de sistema de arquivos. Nenhum ponto de montagem é necessário.

Partições LVM

O Logical Volume Manager (LVM) permite que você adicione uma camada de abstração e agrupe o espaço em disco que pode ser atribuído a um ou mais volumes lógicos. Depois de criar os volumes lógicos, você pode adicionar sistemas de arquivos a esses volumes e montá-los como faria com partições Linux normais. Mas se ficar sem espaço, você tem muito mais flexibilidade quando se trata de gerenciar essas partições.

As etapas descritas aqui detalham como criar uma partição LVM (referidas como um volume LVM físico), aplicar uma ou mais partições LVM a um grupo de volumes LVM e então criar volumes lógicos a partir desse grupo de volumes:

- 1. Selecione Create ⇒ LVM Physical Volume e clique no botão Create. Uma janela Add Partition aparece.**
- 2. Deixe o tipo de sistema de arquivos como “physical volume (LVM)” e escolha a quantidade de espaço para dedicar a ele. Algumas pessoas vão aplicar todo o espaço restante em seu disco para a partição LVM depois de criar as partições não LVM que desejam (como uma partição `/boot` separada e, possivelmente, uma partição de swap). Clique em OK para continuar.**
- 3. Com um ou mais volumes físicos LVM criados, clique em Create, selecione LVM Volume Group e clique em Create de novo. A janela Make LVM Volume Group aparece.**
- 4. Selecione um nome de grupo de volumes (`VolGroup` é usado por padrão) e um tamanho de extensão física (essa é a menor unidade de espaço que pode ser adicionada ou removida de um volume lógico) e escolha qual partição adicionar ao grupo (use o volume físico LVM que você criou na etapa anterior).**
- 5. Selecione Add e a janela Make Logical Volume aparece. É aqui que você cria os volumes lógicos reais em que cada sistema de arquivos residirá. A Figura 9.6 mostra as telas para a criação do grupo de volumes e um volume lógico. Na maioria dos casos, você não vai usar todo o espaço disponível, assim é possível aumentar suas partições mais tarde, se necessário. Nesse exemplo, o nome do volume lógico é `myhome`, o grupo é `VolGroup` e o dispositivo resultante representa o volume lógico `/dev/mapper/VolGroup-myhome`. Se o tipo de sistema de arquivos e o tamanho**

estiverem corretos, selecione OK para criar o volume lógico.

um grupo de volumes LVM e um volume lógico a partir de um volume físico LVM.

- 6. Se ainda houver espaço disponível a partir do grupo de volumes, você pode criar mais volumes lógicos a partir desse grupo, repetindo os mesmos passos. Clique em OK para retornar à tela de particionamento.**

Partições RAID

Um array RAID permite que você use vários discos, de modo que possa ter várias cópias de seus dados ou grave-os em vários discos (para melhorar o desempenho) ou ambos. Siga estes passos a fim de criar as partições de RAID que você precisa para criar um array RAID:

- 1. Selecione Create ⇒ RAID Partition e clique em Create. A janela Add Partition aparece.**
- 2. Supondo que você tenha vários discos disponíveis, o que é necessário para criar um array RAID adequado, escolha o primeiro disco, indique o tamanho da partição RAID e selecione OK.**
- 3. Repita os passos 1 e 2 para cada disco a fim de criar cada partição necessária no array. (O tamanho do array será baseado na menor das partições criadas; portanto, em geral, você quer que elas sejam do mesmo tamanho.)**
- 4. Selecione Create de novo, mas desta vez selecione RAID Device e clique em Create. A janela Make Raid Device aparece, parecida com a mostrada na Figura 9.7.**

em dispositivo RAID.

5. **Selecione o ponto de montagem, o tipo de sistema de arquivos e um dispositivo RAID (normalmente md0 para o primeiro dispositivo RAID). Além disso, selecione o nível de RAID (RAID0 divide os dados em vários discos para melhorar o desempenho; RAID1 faz com que os dados sejam espelhados nos membros do RAID; RAID5 é popular porque oferece redundância a um baixo custo). Então, verifique as partições para usar como membros do RAID e selecione OK. O novo array RAID aparece sob o título RAID Devices.**

Quando você terminar de particionar seu disco, selecione Next para salvar seu particionamento e continuar a instalação.

Razões para esquemas de particionamento diferentes

Existem diferentes opiniões a respeito de como dividir um disco rígido. Eis algumas questões a serem consideradas:

- Quer instalar outro sistema operacional? Se quiser o Windows em seu computador, juntamente com o Linux, você precisa de pelo menos uma partição Windows (tipo Win95, FAT16, VFAT ou NTFS), uma partição Linux (Linux ext4) e, normalmente, uma partição de swap.
- É um sistema multiusuário? Se você próprio estiver utilizando o sistema, provavelmente não precisa de muitas partições. Uma razão para particionar um sistema operacional é proteger o sistema inteiro contra falta de espaço em disco. Isso serve também para impor limites sobre o que uma pessoa pode usar em seu diretório inicial (embora as cotas de disco forneçam uma maneira mais refinada de limitar o uso de disco).
- Você tem vários discos rígidos? Você precisa de pelo menos uma partição por disco rígido. Se seu sistema tiver dois discos rígidos, você pode atribuir um para / e outro para /home (se você tem muitos usuários), ou /var (se o computador é um monte de servidores de compartilhamento de dados). Com uma partição /home separada, você pode instalar outro sistema Linux no futuro, sem mexer com os diretórios iniciais (e, presumivelmente, todos ou a maioria de seus dados de usuário).

As dicas para criar partições

Alterar suas partições de disco para lidar com vários sistemas operacionais pode ser muito complicado, em parte porque cada sistema operacional tem suas próprias ideias sobre como as informações de particionamento devem ser tratadas, bem como diferentes ferramentas para fazer isso. Eis algumas dicas para ajudar você fazer da forma correta:

- Se você estiver criando um sistema dual-boot, especialmente para um sistema Windows, tente instalar o sistema operacional Windows primeiro depois de particionar seu disco. Caso contrário, a instalação do Windows pode tornar inacessíveis as partições do Linux. Escolher um sistema de arquivos VFAT em vez de NTFS para o Windows também tornará o compartilhamento de arquivos entre os sistemas Windows e Linux mais fácil e confiável. (O suporte a partições NTFS no Linux melhorou muito nos últimos anos, mas nem todos os sistemas Linux incluem esse suporte).
- A página man do `fdisk` recomenda que você use ferramentas de particionamento que vêm com um sistema operacional a fim de criar partições para ele. Por exemplo, o `fdisk` do DOS sabe como criar partições das quais o DOS vai gostar, e o `fdisk` do Linux criará alegremente suas partições Linux. Depois de seu disco rígido estar configurado para dual boot, porém, você provavelmente não deve voltar a usar ferramentas de particionamento que funcionam somente para Windows. Use o `fdisk` Linux ou um produto feito para sistemas multi-boot (como o Partition Magic).
- Você pode ter até 63 partições em um disco rígido IDE. Um disco rígido SCSI pode ter até 15 partições. Em geral, você não precisa de tantas partições, mas, se precisar, use o LVM e crie quantos volumes lógicos quiser.

Se estiver usando Linux como um sistema desktop, você provavelmente não precisa de muitas partições diferentes. Mas há algumas boas razões para ter várias partições para sistemas Linux que são compartilhadas por um grande número de usuários ou que são servidores web públicos ou de arquivos. Ter várias partições no Fedora ou no RHEL, por exemplo, oferece as seguintes vantagens:

- Proteção contra ataques — Ataques de negação de serviço às vezes tomam ações que tentam encher seu disco rígido. Se as áreas públicas, como `/var`, estiverem em partições separadas, um ataque bem-sucedido pode encher uma partição sem desligar o computador inteiro. Como `/var` é o local padrão para servidores web e servidores FTP e supõe-se que ele armazene uma grande quantidade de dados, discos rígidos inteiros costumam ser atribuídos ao sistema de arquivos `/var` sozinho.
- Proteção contra sistemas de arquivos corrompidos — Se você tiver apenas um sistema de arquivos (`/`), a corrupção pode fazer com que todo o sistema Linux seja danificado. A corrupção de uma partição menor pode ser mais fácil de corrigir e muitas vezes permite que o computador permaneça em serviço enquanto a correção é feita.

A Tabela 9.4 lista alguns diretórios que você pode querer considerar criar em partições do sistema de arquivos separadas.

TABELA 9.4 Atribuindo Partições a Diretórios Particulares

Diretório	Explicação
<code>boot</code>	Às vezes, a BIOS de PCs mais antigos pode acessar apenas os primeiros 1.024 cilindros de seu disco rígido. Para certificar-se de que as informações em seu diretório <code>/boot</code> são acessíveis à

BIOS, crie uma partição de disco separada (de cerca de 500 MB) para `/boot`. Mesmo com vários kernels instalados, raramente há uma razão para `/boot` ser maior do que 500MB.

sr	Essa estrutura de diretórios contém a maioria dos aplicativos e utilitários disponíveis para usuários de Linux. A teoria original era que, se <code>/usr</code> estivesse em uma partição separada, você poderia montar o sistema de arquivos como somente leitura após o sistema operacional ser instalado. Isso impediria os invasores de substituir ou remover aplicativos importantes do sistema por suas próprias versões que poderiam causar problemas de segurança. Uma partição <code>/usr</code> separada também é útil se você tiver estações de trabalho sem disco rígido em sua rede local. Usando NFS, você pode compartilhar <code>/usr</code> pela rede com as estações de trabalho.
ar	Seu servidor de FTP (<code>/var/ftp</code>) e seu servidor web (<code>/var/www</code>) estão, por padrão em muitos sistemas Linux, armazenados em <code>/var</code> . Ter uma partição <code>/var</code> separada pode impedir que um ataque a esses recursos corrompa ou encha seu disco rígido inteiro.
ome	Como seus diretórios de conta de usuário estão localizados nesse diretório, ter uma conta <code>/home</code> separada pode impedir um usuário irresponsável de encher o disco rígido inteiro. Isso também separa convenientemente os dados do usuário, de um lado, dos dados do seu sistema operacional, de outro (para backups fáceis ou novas instalações). Muitas vezes, <code>/home</code> é criado como um volume LVM lógico, para permitir que ele seja ampliado com o aumento das demandas do usuário.
mp	Proteger <code>/tmp</code> do resto do disco rígido, colocando-o em uma partição separada, pode assegurar que os aplicativos que precisam gravar em arquivos temporários em <code>/tmp</code> sejam capazes de completar sua transformação, mesmo que o resto do disco fique cheio.

Embora as pessoas que usam sistemas Linux raramente vejam a necessidade de várias partições, aqueles que mantêm e, ocasionalmente, tem de recuperar grandes sistemas, são gratos quando o sistema que eles precisam corrigir tem várias partições. O fato de ter várias partições pode limitar os efeitos de danos deliberados (como os de ataques de negação de serviço), problemas de usuários errantes e corrupção acidental de arquivos.

Usando o gerenciador de inicialização GRUB

Um gerenciador de inicialização permite escolher quando e como iniciar os sistemas operacionais instalados nos discos rígidos de seu computador. O GRand Unified Bootloader (GRUB) é o gerenciador de inicialização mais popular usado para sistemas Linux instalados. Há duas versões principais do GRUB disponíveis hoje:

- GRUB Legacy (versão 1) — Até o momento, essa versão do GRUB é usada por padrão para

inicializar sistemas operacionais Red Hat Enterprise Linux (pelo menos até o RHEL 6.3). Ela também era usada com versões anteriores do Fedora e Ubuntu.

- GRUB 2 — As versões atuais do Ubuntu e Fedora usam o GRUB 2 como o gerenciador de inicialização padrão.

A versão do GRUB Legacy é descrita nas próximas seções. Depois disso, há uma descrição do GRUB 2.

ta

LINUX é outro gerenciador de inicialização que você vai encontrar com sistemas Linux. Os gerenciadores de inicialização SYSLINUX não são normalmente utilizados para sistemas Linux instalados, mas são comumente usados como o gerenciador de inicialização para CDs e DVDs do Linux. O SYSLINUX é particularmente bom para iniciar imagens de CD ISO9660 (isolinux) e pen drives (syslinux), para funcionar em hardware mais antigo ou para inicializar um sistema PXE (pxelinux) através da rede.

Usando o GRUB Legacy (versão 1)

Com vários sistemas operacionais instalados e várias partições configuradas, como é que o computador sabe qual o sistema operacional iniciar? Para selecionar e gerenciar qual partição é inicializada e como é inicializada, você precisa de uma rotina de inicialização. O gerenciador de inicialização que é instalado por padrão com o Red Hat Enterprise Linux é o Grand Unified Boot loader (GRUB).

O GRUB Legacy é um gerenciador de inicialização GNU

(<http://www.gnu.org/software/grub>), que oferece os seguintes recursos:

- Suporte para múltiplos formatos executáveis.
- Suporte a sistemas operacionais multi-boot (como o Fedora, RHEL, FreeBSD, NetBSD, OpenBSD e outros sistemas Linux).
- Suporte a sistemas operacionais não multi-boot (como Windows 95, Windows 98, Windows NT, Windows ME, Windows XP, Windows Vista, Windows 7 e OS/2) por meio de uma função de carregamento em cadeia. O carregamento em cadeia é o ato de carregar outra rotina de inicialização (presumivelmente a que é específica para o sistema operacional proprietário) a partir do GRUB para iniciar o sistema operacional selecionado.
- Suporte a vários tipos de arquivos.
- Suporte para a descompactação automática de imagens de inicialização.
- Suporte para fazer download de imagens de inicialização a partir de uma rede.

À época em que escrevíamos isso, o GRUB versão 1 era o gerenciador de inicialização usado no Red Hat Enterprise Linux. O GRUB versão 2 é usado no Fedora, no Ubuntu e em outras distribuições Linux. Esta seção descreve como usar o GRUB versão 1.

Para mais informações sobre como o GRUB funciona, na linha de comando digite `man grub` ou `info grub`. O comando `info grub` contém mais detalhes sobre o GRUB.

Inicializando com o GRUB Legacy

Ao instalar o Linux, você normalmente tem a opção de configurar as informações necessárias para iniciar o computador (com um ou mais sistemas operacionais) no gerenciador de inicialização padrão. O GRUB é muito flexível para configurar, por isso, ele parece diferente em diferentes distribuições do Linux.

Com o gerenciador de inicialização GRUB que vem com o Red Hat Enterprise Linux instalado no registro de inicialização mestre de seu disco rígido, quando o BIOS inicia o gerenciador de inicialização uma dessas várias coisas pode acontecer:

- **Padrão** — Se você não fizer nada, o sistema operacional padrão inicializará automaticamente depois de cinco segundos. (O tempo limite é definido pelo valor `timeout`, em segundos, no arquivo `grub.conf` ou `menu.lst`.)
- **Selecione um sistema operacional** — Pressione qualquer tecla antes de os cinco segundos expirarem e você verá uma lista de títulos para escolher. Os títulos podem representar um ou mais núcleos para o mesmo sistema Linux. Ou eles podem representar o Windows, o Ubuntu ou outros sistemas operacionais. Use as teclas de seta para cima e para baixo para destacar qualquer título e, então, pressione Enter para iniciar esse sistema operacional.
- **Edite o processo de inicialização** — Se você quiser alterar qualquer uma das opções utilizadas durante o processo de inicialização, use as teclas de seta para destacar o sistema operacional que você deseja e digite `e` para selecioná-lo. Siga o próximo procedimento para alterar suas opções de inicialização temporariamente.

Se quiser alterar as opções de inicialização para que elas tenham efeito cada vez que você iniciar o computador, consulte a seção sobre como alterar permanentemente as opções de inicialização. Alterar essas opções envolve editar o arquivo `/boot/grub/grub.conf`.

Alterando as opções de inicialização temporariamente

A partir da tela de boot GRUB Legacy, você pode escolher entre modificar ou adicionar opções de inicialização à sessão de inicialização atual. Em alguns sistemas Linux, o menu está escondido, assim você tem que pressionar a tecla Tab ou alguma outra tecla (antes de alguns segundos de o tempo limite ser excedido) para ver o menu. Então, selecione o sistema operacional que você deseja (usando as teclas de seta) e digite `e` (como descrito anteriormente).

Três linhas na tela de exemplo de edição do GRUB Legacy identificam o processo de inicialização do sistema operacional que você escolheu. Eis um exemplo dessas linhas (devido ao comprimento da linha do kernel, essa linha foi aqui dividida em três):

```
root (hd0,0)
kernel /vmlinuz-2.6.32-131.17.1.el6.x86_64 ro root=/dev/mapper/vg_
  myhost-lv_root
rd_NO_MD rd_NO_DM LANG=en_US.UTF-8 SYSFONT=latarcyrheb-sun16
KEYBOARDTYPE=pc
KEYTABLE=us rhgb quiet crashkernel=auto
```

A primeira linha (começando com `root`) mostra que a entrada para o gerenciador de inicialização GRUB está na primeira partição do primeiro disco rígido (`hd0, 0`). O GRUB representa o disco rígido como `hd`, independentemente de se tratar de um disco SCSI, um disco IDE ou outro tipo de disco. No GRUB Legacy, você simplesmente conta o número da unidade e o número da partição a partir de zero (0). A segunda linha do exemplo (começando com `kernel`) identifica a imagem de inicialização do kernel (`/boot/vmlinuz-2.6.32-131.17.1.el6.x86_64`) e várias opções. As opções identificam a partição como inicialmente sendo carregada `ro` (somente leitura) e a localização do sistema de arquivos raiz em uma partição com o rótulo que começa `root=/dev/mapper/vg_myhost-lv_root`. A terceira linha (começando com `initrd`) identifica a localização do disco de RAM inicial, que contém módulos adicionais e ferramentas necessárias durante o processo de inicialização.

Se você for alterar qualquer uma das linhas relacionadas com o processo de inicialização, provavelmente modificará apenas a segunda linha para adicionar ou remover opções de inicialização. Siga estes passos para fazer exatamente isso:

- 1. Depois de interromper o processo de inicialização do GRUB e digitar `e` para selecionar a entrada de inicialização que você quer, posicione o cursor na linha `kernel` e digite `e`.**
- 2. Adicione ou remova as opções depois do nome da imagem de inicialização. Você pode utilizar um conjunto mínimo de recursos de edição de linha de comando do shell `bash` para editar a linha. Você pode até mesmo utilizar a conclusão de comando (digite parte de um nome de arquivo e pressione `Tab` para completá-lo). Eis algumas opções que você pode querer adicionar ou excluir:**
 - Inicialize para um shell. Se tiver esquecido sua senha de `root` ou se seu processo de inicialização travar, você pode iniciar diretamente para um shell adicionando `init=/bin/sh` à linha de boot.
 - Selecione um nível de execução. Se desejar inicializar um nível de execução especial, você pode adicionar o nível de execução que deseja ao final da linha do kernel. Por exemplo, para fazer a inicialização do RHEL executar o nível 3 (multiusuário mais o modo de rede), adicione 3 ao final da linha do kernel. Você também pode inicializar no modo monousuário (1), no modo multiusuário (2) ou no modo gráfico do X (5). O nível 3 é uma boa escolha se sua GUI estiver temporariamente rompida. O nível 1 é bom se você esqueceu sua senha de `root`.
 - Observe as mensagens de inicialização. Por padrão, você verá uma tela enquanto o Linux inicializa. Se quiser ver as mensagens que mostram atividades que acontecem à medida que o sistema é inicializado, você pode remover a opção `rhgb quiet` da linha do kernel. Isso permite que você veja mensagens à medida que elas rolam na tela. Pressionar `Esc` durante a inicialização tem o mesmo resultado.
- 3. Pressione `Enter` para retornar à tela de edição.**
- 4. Digite `b` para inicializar o computador com as novas opções. Na próxima vez que você inicializar seu computador, as novas opções não serão salvas. Para adicionar opções de**

modo que elas sejam salvas permanentemente, consulte a próxima seção.

Alterando as opções de inicialização permanentemente

Você pode mudar as opções que têm efeito a cada vez que você inicializa o computador alterando o arquivo de configuração do GRUB. No RHEL e outros sistemas Linux, a configuração do GRUB concentra-se no arquivo `/boot/grub/grub.conf` ou `/boot/grub/menu.lst`.

O arquivo `/boot/grub/grub.conf` é criado quando você instala o Linux. Eis um exemplo desse arquivo para o RHEL:

```
# grub.conf generated by anaconda
##
Note that you do not have to rerun grub after making changes to this
file
# NOTICE: You have a /boot partition. This means that
#         all kernel and initrd paths are relative to /boot/, eg.
#         root (hd0,0)
#         kernel /vmlinuz-version ro root=/dev/mapper/vg_joke-lv_root
#         initrd /initrd-[generic]-version.img
$boot=/dev/sda
default=0
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Red Hat Enterprise Linux (2.6.32-131.17.1.el6.x86_64)
    root (hd0,0)
    kernel /vmlinuz-2.6.32-131.17.1.el6.x86_64 ro
        root=/dev/mapper/vg_myhost-lv_root rd_NO_MD rd_NO_DM
        LANG=en_US.UTF-8 SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc
        KEYTABLE=us rhgb quiet crashkernel=auto
    initrd /initramfs-2.6.32-131.17.1.el6.x86_64.img
title Windows XP
    rootnoverify (hd0,1)
    chainloader +1
```

A linha `default=0` indica que a primeira partição na lista (nesse caso, Red Hat Enterprise Linux) será a única que é inicializada por padrão. A linha `timeout=5` faz com que o GRUB faça uma pausa por cinco segundos antes de iniciar a partição padrão. (Esse é o tempo que você tem para pressionar e se quiser editar a linha de boot ou para pressionar as teclas de seta a fim de selecionar um sistema operacional diferente para inicializar.)

A linha `splashimage` procura na primeira partição do primeiro disco (`hd0, 0`) a partição de boot (nesse caso `/dev/sda1`). O GRUB carrega `splash.xpm.gz`, como a imagem na tela inicial (`/boot/grub/splash.xpm.gz`). A tela inicial aparece como o fundo da tela de inicialização.

ta

GRUB indica as partições de disco a usando a seguinte notação: `(hd0, 0)`. O primeiro número representa o disco e o segundo é a partição nesse disco. Portanto, `(hd0, 1)` é a segunda partição (1) no primeiro disco (0).

As duas partições de inicialização nesse exemplo são Red Hat Enterprise Linux e Windows XP. As linhas de título para cada uma dessas partições são seguidas pelo nome que aparece na tela de inicialização para representar cada partição.

Para o sistema RHEL, a linha `root` indica o local da partição de inicialização como a segunda partição no primeiro disco. Assim, para encontrar o kernel inicializável (`vmlinuz-*`) e a imagem de

inicialização do disco RAM inicial `initrd` que é carregada (`initrd-*`), o GRUB monta `hd0,0` como a raiz de todo o sistema de arquivos (que é representado por `/dev/mapper/vg_myhost-lv_root` e é montada como `/`). Há outras opções na linha `kernel`.

Para a partição do Windows XP, a linha `rootnoverify` indica que o GRUB não deve tentar montar a partição. Nesse caso, Windows XP é a primeira partição do primeiro disco rígido (`hd0,1`) ou `/dev/sda2`. Em vez de montar a partição e passar opções para o novo sistema operacional, a linha `chainloader +1` diz para GRUB passar a inicialização do sistema operacional para outro gerenciador de inicialização. O `+1` indica que o primeiro setor da partição é usado como gerenciador de inicialização. (Você poderia semelhantemente configurar para inicializar um sistema operacional Windows Vista ou Windows 7.)

ta
sistemas operacionais da Microsoft exigem que você use o `chainloader` para inicializá-los a partir do GRUB, que o GRUB não oferece suporte nativo para sistemas operacionais Windows.

Se fizer quaisquer alterações no arquivo `/boot/grub/grub.conf`, você não precisa carregar essas alterações. O GRUB automaticamente as captura quando você reiniciar seu computador.

Adicionando uma nova imagem de inicialização do GRUB

Você pode ter imagens diferentes de inicialização para os kernels que incluem recursos diferentes. Na maioria dos casos, a instalação de um novo pacote de kernel configura automaticamente `grub.conf` para usá-lo. Mas se você quiser adicionar um kernel manualmente, eis o procedimento para modificar o arquivo `grub.conf` no Red Hat Enterprise Linux para inicializar o kernel:

1. **Copie a nova imagem a partir do diretório em que ela foi criada (como `/usr/src/kernels/linux-2.6.25-11/arch/i386/boot`) para o diretório `/boot`. Nomeie o arquivo que reflete seu conteúdo, como `bz-2.6.25-11`. Por exemplo:**

```
# cd /usr/src/Linux-2.6.25.11/arch/i386/boot
# cp bzImage /boot/bz-2.6.25-11
```

2. **Adicione várias linhas ao arquivo `/boot/grub/grub.conf` para que a imagem possa ser iniciada no momento da inicialização, se ele for selecionada. Por exemplo:**

```
title Red Hat Enterprise Linux 6.3 (My own IPV6 build)
  root (hd0,4)
  kernel /bz-2.6.25-11 ro root=/dev/sda5
  initrd /initrd-2.6.25-11.img
```

3. **Reinicie seu computador.**
4. **Quando a tela de inicialização do GRUB aparecer, mova seu cursor para o título que representa o novo kernel e pressione Enter.**

A vantagem para essa abordagem, no lugar de copiar a nova imagem de inicialização sobre a antiga, é que se o kernel não conseguir inicializar, você sempre poderá voltar e reiniciar o kernel antigo. Quando você se sentir confiante de que o novo kernel está funcionando corretamente, pode usá-lo para substituir o kernel antigo ou talvez apenas tornar o novo kernel a definição de inicialização padrão.

Usando o GRUB 2

O GRUB 2 representa uma grande reformulação do projeto GRUB Legacy. Ele foi adotado como o gerenciador de inicialização padrão para o Fedora e o Ubuntu. Embora a principal função do GRUB 2 ainda seja encontrar e iniciar o sistema operacional que você deseja, agora há muito mais poder e flexibilidade incorporado às ferramentas e arquivos de configuração que o levam lá.

No GRUB 2, o arquivo de configuração agora se chama `/boot/grub2/grub.cfg` (no Fedora e outros sistemas Linux usando o GRUB 2). Tudo a partir do conteúdo de `grub.cfg`, até a maneira como `grub.cfg` é criado, é diferente do arquivo `grub.conf` do GRUB Legacy. Eis algumas coisas que você deve saber sobre o arquivo `grub.cfg`:

- Em vez de editar `grub.cfg` manualmente ou adicionar pacotes RPM de kernel a ele, `grub.cfg` é gerado automaticamente a partir do conteúdo do arquivo `/etc/default/grub` e do diretório `/etc/grub.d`. Você deve modificar ou adicionar esses arquivos para configurar o GRUB 2 por conta própria.
- O arquivo `grub.cfg` pode conter sintaxe de script, incluindo coisas como funções, loops e variáveis.
- Os nomes de dispositivos necessários para identificar a localização de núcleos de memória RAM e discos iniciais podem ser mais confiavelmente identificados usando rótulos ou Identificadores Universalmente Únicos (UUIDs). Isso evita a possibilidade de um dispositivo de disco como o `/dev/sda` ser alterado para `/dev/sdb` quando você adiciona um novo disco (o que faria com que o kernel não fosse encontrado).

Os comentários no arquivo `grub.cfg` indicam de onde o conteúdo veio. Por exemplo, a informação gerada a partir do arquivo `/etc/grub.d/00_header` vem logo após esta linha de comentário:

```
### BEGIN /etc/grub.d/00_header ###
```

No início da seção `00_header`, há algumas funções, como aquelas que carregam drivers para fazer seu monitor de vídeo funcionar. Depois disso, a maioria das seções no arquivo `grub.cfg` consiste em entradas de menu. O seguinte é um exemplo de um item de menu extraído do arquivo `grub.cfg` que você pode selecionar para iniciar o Fedora 16 quando o sistema inicializa:

```
menuentry 'Fedora (3.4.2-1.fc16.i686)' --class fedora
  --class gnu-linux --class gnu --class os {
  load_video
  set gfxpayload=keep
```

```

insmod gzio
insmod part_msdos
insmod ext2
set root='(hd0,msdos4)'
search --no-floppy --fs-uuid --set=root eb31517f-f404-410b-
937e-a6093b5a5380
echo `Loading Fedora (3.4.2-1.fc16.i686)`
linux /boot/vmlinuz-3.4.2-1.fc16.i686
root=UUID=eb31517f-f404-410b-937e-a6093b5a5380 ro rd.md=0
rd.lvm=0 rd.dm=0 KEYTABLE=us quiet SYSFONT=latarcyrheb-sun16
rhgb rd.luks=0 LANG=en_US.UTF-8
echo `Loading initial ramdisk ...`
initrd /boot/initramfs-3.4.2-1.fc16.i686.img
}

```

A entrada de menu para essa seleção aparece como `Fedora (3.4.2-1.fc16.i686)` no menu de inicialização do GRUB 2. As entradas `--class` nessa linha permitem que o GRUB 2 agrupe as entradas de menu em classes (nesse caso, identifica-o como um tipo de sistema `fedora`, `gnu-linux`, `gnu`, `os`). As linhas seguintes carregam drivers de vídeo e drivers do sistema de arquivos. Depois disso, as linhas identificam a localização do sistema de arquivos raiz.

Nesse ponto, o GRUB 2 exibe uma mensagem `Loading Fedora (3.4.2-1.fc16.i686)`. Agora obtemos informações que são semelhantes ao que seria de se esperar em uma entrada do GRUB Legacy. A linha `linux` identifica a localização do kernel (`boot/vmlinuz-3.4.2-1.fc16.i686`), seguida por opções que são passadas para o kernel. Então, depois de uma mensagem `Loading initial ramdisk ...`, o local do disco de RAM inicial é identificado e esse arquivo é carregado.

Há muitos mais recursos no GRUB 2 que você pode aprender se quiser se aprofundar em seu gerenciador de inicialização do sistema. A melhor documentação para o GRUB 2 está disponível no sistema Fedora: digite `info grub2` no shell. A entrada de `info` para o GRUB 2 fornece muitas informações para iniciar sistemas operacionais diferentes, escrever seus próprios arquivos de configuração e trabalhar com arquivos de imagem e com outros recursos do GRUB, bem como configurar variáveis de ambiente.

Resumo

Apesar de todas as distribuições do Linux incluírem um método de instalação diferente, você precisa fazer muitas atividades comuns, independentemente do sistema Linux instalado. Para cada sistema Linux, é preciso lidar com questões de particionamento de disco, opções de inicialização e configurar gerenciadores de inicialização.

Neste capítulo, você aprendeu os procedimentos de instalação do Fedora (usando uma instalação de Live CD) e do Red Hat Enterprise (a partir da mídia de instalação). O capítulo também abordou temas especiais de instalação, incluindo o uso de opções de boot e particionamento de disco. Agora, com seu sistema Linux instalado, o Capítulo 10 descreve como começar a gerenciar o software nele.

Exercícios

Use esses exercícios para testar seus conhecimentos de instalação do Linux. Recomendo que você faça-os em um computador que não tenha nenhum sistema operacional ou dados que você não queira perder (em outras palavras, que você não se importa de apagar). Se você tiver um computador que permite a instalação de sistemas virtuais, essa também é uma forma segura de fazer os exercícios. Esses exercícios foram testados usando um Fedora 16 Live CD e um DVD de instalação do RHEL 6.2 Server.

1. **Inicie a instalação a partir de um Fedora Live CD usando o maior número possível das opções.**
2. **Depois de ter instalado completamente o Fedora, atualize todos os pacotes no sistema.**
3. **Inicie a instalação a partir de uma instalação de DVD do RHEL, mas faça isso de tal modo que a instalação seja executada em modo texto. Conclua a instalação da maneira que você escolher.**
4. **Inicie a instalação a partir de uma instalação de DVD do RHEL e configure o particionamento de disco da seguinte maneira: uma partição `/boot` de 400MB e uma partição LVM de 10 GB. Crie a partição LVM em um volume LVM físico e a atribua a um grupo de volumes chamado `tracker`. Crie três volumes lógicos a partir do grupo de volumes `tracker`: `/` (3G), `/var` (2G) e `/home` (3G). Deixe o resto como espaço não utilizado.**

Atenção

Completando o Exercício 4, você acabará excluindo todo o conteúdo do disco rígido. Se quiser apenas usar esse exercício para a prática de particionamento, você pode reiniciar seu computador antes de clicar em Next no final do procedimento, sem apagar seu disco rígido. Mas se quiser prosseguir e particionar seu disco, lembre-se de que todos os dados serão excluídos.

Obtendo e gerenciando software

NESTE CAPÍTULO

Usando o PackageKit para instalar software

Trabalhando com pacotes RPM

Usando o comando yum para gerenciar pacotes

Usando o comando rpm para trabalhar com pacotes

Instalando software na empresa

Em distribuições Linux, como Fedora e Ubuntu, você não precisa saber muito sobre como o software é empacotado e gerenciado para obter aquele que você quer. Essas distribuições possuem excelentes ferramentas de instalação de softwares, que automaticamente apontam para grandes repositórios deles. Apenas alguns cliques e você está usando o programa em pouco mais tempo do que leva para fazer o download.

O fato de que o gerenciamento de software Linux é tão fácil nos dias de hoje é um crédito para a comunidade Linux, que tem trabalhado diligentemente para criar formatos de pacotes, ferramentas de instalação complexas e pacotes de software de alta qualidade. Não só é fácil de obter o software, mas também, uma vez instalado, é fácil gerenciá-lo, atualizá-lo, consultá-lo e removê-lo.

Este capítulo começa descrevendo como instalar pacotes de software no Fedora usando a ferramenta de instalação gráfica PackageKit. Se você estiver apenas instalando alguns pacotes de software em seu próprio sistema desktop, pode não precisar de muito mais do que isso e ocasionais atualizações de segurança.

Para nos aprofundarmos no gerenciamento de software Linux, descrevo a seguir o que compõe os pacotes de software Linux (comparando os pacotes formatados `deb` e `rpm`), componentes de software de gerenciamento subjacentes e comandos (`yum` e `rpm`) para gerenciamento de software no Fedora e Red Hat Enterprise Linux. Isso é seguido por uma descrição de como gerenciar pacotes de software na computação corporativa.

Gerenciando software com o PackageKit

O PackageKit é um conjunto de programas que fornece janelas gráficas e outros componentes para instalação, remoção e outras tarefas de gerenciamento de software no Fedora e no Red Hat Enterprise Linux. O Ubuntu utiliza componentes do PackageKit, mas não toda a interface. Para usar o PackageKit a partir de um desktop GNOME 3 no Fedora, clique no ícone Add/Remove Software (veja no menu Applications).

A Figura 10.1 mostra um exemplo da janela PackageKit Add/Remove Software sendo usada para procurar por um pacote.

e e gerencie pacotes de software a partir do PackageKit.

Ativando repositórios e obtendo atualizações

Depois que você instalou o Fedora, o PackageKit está configurado para lidar com as atualizações de software e conectar-se a repositórios de programas do Fedora, conforme necessário. Para ver informações sobre como o PackageKit está configurado, a partir da janela PackageKit, selecione System ⇒ Software Sources. Você vai ver:

- **Update Settings** — A partir das definições sobre essa guia, você vê que o sistema verifica diariamente a existência de atualizações de pacotes de software e instala automaticamente apenas atualizações de segurança.
- **Software Sources** — Entradas dessa guia mostram repositórios de software que estão configurados em seu sistema. As com marcas de seleção ao lado estão habilitadas. Por padrão, o repositório principal o de atualizações do Fedora estão habilitados.

Se você acabou de instalar o Fedora, é uma boa ideia imediatamente verificar se há atualizações (especialmente se a versão que você instalou já existe há um tempo). Tudo o que você precisa é uma conexão com a internet para verificar se há atualizações (nenhuma configuração ou senha é necessária).

Para verificar os pacotes de software atualizados, clique no botão Check Now na parte inferior da guia Software Sources. A Figura 10.2 mostra um exemplo da janela Software Updates após uma nova instalação do Fedora Live CD, com centenas de pacotes disponíveis para atualização.

atualiza o software atual a partir da janela Software Update.

Neste exemplo, há 428 pacotes (187,5 MB) prontos para serem atualizados. Assim, o processo pode demorar um pouco. Você pode desmarcar as caixas ao lado de qualquer pacote que não quer atualizar. Quando estiver satisfeito com a lista, clique em Install Updates.

Os repositórios listados vêm de entradas nos arquivos do diretório `/etc/yum.repos.d`. As entradas padrão disponibilizam milhares de pacotes a partir dos principais repositórios. Mas você também pode adicionar repositórios de terceiros para obter um software especial ou software não livre que não vem com o Fedora. A seção “Usando repositórios de software de terceiros”, posteriormente neste capítulo, descreve como fazer isso.

Procurando pacotes

Para localizar pacotes a partir da janela Add/Remove Software, você pode usar a caixa de pesquisa (para encontrar por uma palavra do nome do pacote ou da descrição), selecione o pacote a partir dos grupos de software listados no lado esquerdo ou classifique-os do mais recente para o mais antigo. Clique no nome do pacote para ver uma descrição. Os ícones ao lado de cada pacote informam o status de cada um:

- **Caixa fechada** — O pacote está disponível, mas ainda não instalado.
- **Caixa aberta** — O pacote está instalado.
- **Caixa aberta com sinal de mais** — O pacote está programado para ser instalado. Você deve clicar em Apply para instalá-lo.

Para refinar sua busca de pacotes, você pode clicar no botão Filters no topo da janela. Em vez de exibir todos os pacotes de uma categoria ou pesquisa, posso selecionar Installed ⇒ Only Available para ver somente pacotes que não estão instalados. Ou selecione Graphical ⇒ Only Graphical para exibir apenas aplicativos gráficos.

Se tiver ativado repositórios não livres (pacotes que são de código aberto ou patenteados), você também pode ver ou excluir os pacotes no menu Filters. Apenas os pacotes mais recentes e aqueles que correspondem à arquitetura de seu computador são exibidos, mas você também pode mudar isso.

Instalando e removendo pacotes

Só para ver como funciona, tente instalar um jogo ou outro pacote de software que lhe interessa. Eis como:

1. **Localize um pacote.** Neste caso, quero um jogo; portanto, seleciono a entrada Games na

coluna esquerda. A lista de jogos aparece no painel direito.

2. **Selecione um pacote.** Percorra a lista até encontrar um jogo de que você goste e clique na caixa de seleção ao lado. Uma descrição e fatos sobre o pacote (licenciamento, repositório, origem, tamanho de download etc.) aparecem nos painéis inferiores.
3. **Inicie a instalação.** Clique em Apply para baixar e instalar todos os pacotes marcados. Se houver pacotes exigidos pelos pacotes selecionados para fazê-los funcionar (pacotes dependentes), aparecerá uma lista com eles. Clique em Continue para instalá-los também.
4. **Autentique.** Para instalar um pacote assinado, você precisa digitar sua senha (desde que você tenha atribuído tal privilégio a seu usuário) ou a senha de root, se você não fez isso. Isso impede que qualquer software seja instalado em seu sistema sem seu conhecimento. Digite sua senha e clique em Authenticate. Os pacotes são baixados e instalados.
5. **Comece a usar o pacote.** Como instalei o jogo bzflag, fui para o grupo de aplicativos Games (tecla Windows ⇒ Applications ⇒ Games), vi o ícone lá e o iniciei. Depois de algumas horas de batalhas de tanques online, voltei a escrever o livro.

Se você quiser remover um pacote mais tarde, basta procurar por ele. Quando ele for exibido, desmarque a caixa de seleção e clique em Apply. Depois que você se autenticou com sua senha, o pacote é removido.

Indo além do PackageKit

Se estiver gerenciando um único sistema desktop, você pode estar bastante satisfeito com os milhares de pacotes disponíveis com o Fedora. Salvo indicação contrária, todos os pacotes de software utilizados neste livro estão disponíveis a partir do repositório Fedora (desktop, administração do sistema, servidor, desenvolvimento de software e outros tipos de pacotes).

Mas aqui estão alguns exemplos de por que você pode querer ir além do que você pode fazer com a janela Add/Remove Software:

- **Mais repositórios** — O Fedora e o Red Hat Enterprise Linux distribuem apenas software de código aberto, livremente distribuído. Você pode querer instalar alguns softwares comerciais (como o Adobe Flash player) ou software não livre (disponível a partir de repositórios como rpmfusion.org).
- **Consultas mais complexas** — Usando comandos como yum e rpm, você pode obter informações detalhadas sobre os pacotes, grupos de pacotes e repositórios.
- **Validação de software** — Utilizando rpm e outras ferramentas, você pode verificar se um pacote assinado foi modificado antes de instalá-lo ou se qualquer um dos componentes de um pacote foi adulterado desde que ele foi instalado.
- **Gerenciamento de instalação de software** — Embora o PackageKit funcione bem se você estiver instalando o software em um único sistema, ele não serve bem para gerenciar software em vários sistemas. Outras ferramentas são construídas por cima do recurso rpm para fazer isso.

Antes de trabalhar com algumas das ferramentas de linha de comando para a instalação e gerenciamento

de software no Linux, a próxima seção descreve como os sistemas básicos de empacotamento e gerenciamento de pacotes funcionam no Linux. Em particular, focalizo o empacotamento RPM como ele é usado no Fedora, no Red Hat Enterprise Linux e em distribuições relacionadas.

Entendendo o empacotamento de software RPM do Linux

Nos primeiros sistemas Linux, se quisesse adicionar mais software, você pegava o código-fonte de um projeto que o produziu, compilava-o em binários executáveis e o colocava em seu computador. Se tivesse sorte, alguém já o teria compilado de uma forma que seria executado em seu computador.

A forma do pacote poderia ser um tarball, contendo arquivos executáveis (comandos), documentação, arquivos de configuração e bibliotecas. (Um tarball é um arquivo único em que vários arquivos são reunidos para facilitar o armazenamento ou distribuição.) Quando você instala o software a partir de um tarball, os arquivos deste talvez sejam distribuídos por seu sistema Linux em diretórios apropriados (`/usr/share/man`, `/etc`, `/bin` e `/lib`, para citar alguns).

Embora seja fácil criar um tarball e simplesmente copiar um conjunto de aplicativos em seu sistema Linux, esse método de instalação de software torna difícil:

- **Obter software dependente** — Você precisa saber se o software que está instalando depende de outro software que está sendo instalado para que ele funcione. Então você teria de rastrear esse software e instalá-lo também (o qual também pode ter suas próprias dependências).
- **Listar o software** — Mesmo se você soubesse o nome do comando, talvez não soubesse onde sua documentação ou seus arquivos de configuração estavam instalados quando você o procurava posteriormente.
- **Remover o software** — A menos que você mantivesse o tarball original ou uma lista de arquivos, você não saberia onde todos os arquivos estavam quando chegava a hora de removê-los. Mesmo se soubesse, você tinha de remover cada um individualmente.
- **Atualizar o software** — Tarballs não são projetados para manter metadados sobre o conteúdo que eles armazenam. Depois de instalar o conteúdo de um tarball, você pode não ter como dizer qual versão do software você está usando, o que torna difícil rastrear erros e obter novas versões dele.

Para lidar com esses problemas, os pacotes evoluíram de simples tarballs para pacotes mais complexos. Com apenas algumas poucas exceções (como o Gentoo, o Slackware e outros), a maioria das distribuições Linux se classificava em um de dois formatos de pacotes diferentes — DEB e RPM:

- **Pacotes DEB (.deb)** — O projeto Debian GNU/Linux criou os pacotes `.deb`, que são usados pelo Debian e outras distribuições baseadas em Debian (Ubuntu, Linux Mint, KNOPPIX etc.). Usando ferramentas como `apt-get` e `dpkg`, distribuições Linux podem instalar, gerenciar, atualizar e remover software.
- **Pacotes RPM (.rpm)** — Originalmente chamado Red Hat Package Manager, mas mais tarde renomeado recursivamente como RPM Package Manager, o RPM é o formato de pacotes preferido para as distribuições SUSE e Red Hat (RHEL e Fedora) e aquelas baseadas em

distribuições do Red Hat (CentOS, Oracle Linux etc.). O comando `rpm` foi a primeira ferramenta para gerenciar RPMs, mas mais tarde o `yum` foi adicionado para aumentar a facilidade do RPM.

Para o gerenciamento de sistemas de software individuais, há proponentes de ambos os lados do debate RPM *versus* DEB com argumentos válidos. Mas o RPM é o formato preferido para gerenciar instalação, atualização e manutenção de software de qualidade corporativa. Este capítulo centra-se no gerenciamento de pacotes RPM (Fedora e Red Hat Enterprise Linux) e no gerenciamento de software.

Entendendo pacotes RPM

Um pacote RPM é uma consolidação de arquivos necessários para fornecer um recurso, como um processador de texto, um visualizador de fotos ou um servidor de arquivos. Dentro de um RPM pode estar os comandos, arquivos de configuração e documentação que compõem o recurso de software. Mas um arquivo RPM também contém metadados que armazenam informações sobre o conteúdo do pacote, de onde o pacote veio, o que ele precisa para funcionar e outras informações.

O que há em um RPM?

Antes mesmo de olhar para dentro de um RPM, você pode dizer muito sobre ele a partir do próprio nome do pacote. Para descobrir o nome de um pacote RPM atualmente instalado em seu sistema (como o navegador Firefox), você pode digitar o seguinte a partir do shell no Fedora ou no Red Hat Enterprise Linux:

```
# rpm -q firefox
firefox-10.0.3-1.el6_2.x86_64
```

A partir daí, você pode dizer que o nome de base do pacote é `firefox`. O número da release é 10.0.3 (atribuído pelo criador inicial (*upstream producer*) do Firefox, o Mozilla Project). O número de versão (atribuído pelo empacotador, a Red Hat, cada vez que o pacote é reconstruído com o mesmo número de release) é 1. O pacote `firefox` foi construído para o Red Hat Enterprise Linux 6.2 (el6_2) e compilado para a arquitetura X86 de 64 bits (x86_64).

Quando o pacote `firefox` foi instalado, provavelmente foi copiado da mídia de instalação (como um CD ou DVD) ou baixado de um repositório YUM (mais sobre isso mais tarde). Se você tivesse recebido o arquivo RPM e ele residisse em um diretório local, o nome poderia aparecer como `firefox-10.0.3-1.el6_2.x86_64.rpm` e você poderia instalá-lo a partir daí. Independentemente de onde ele veio, uma vez instalado, o nome e outras informações sobre o pacote são armazenados em um banco de dados RPM na máquina local.

Para saber mais sobre o que está dentro de um pacote RPM, você pode usar outras opções além do comando `rpm` para consultar o banco de dados RPM local. Por exemplo:

```
# rpm -qi firefox
Name: firefox           Relocations: (not relocatable)
Version: 10.0.3        Vendor: Red Hat, Inc.
Release: 1.el6_2       Build Date: Tue 06 Mar 2012 04:41:17 AM EST
Install Date: Thu 22 Mar 2012 01:54:02 PM EDT
Build Host: x86-003.build.bos.redhat.com
Group: Applications/Internet Src RPM: firefox-10.0.3-1.el6_2.src.rpm
Size: 24510223         License: MPLv1.1 or GPLv2+ or LGPLv2+
Signature: RSA/8, Tue 13 Mar 2012 08:14:35 AM, Key ID
199e2f91fd431d51
Packager: Red Hat, Inc. <http://bugzilla.redhat.com/bugzilla>
URL: http://www.mozilla.org/projects/firefox/
Summary: Mozilla Firefox Web browser
Description:
Mozilla Firefox is an open-source web browser, designed for
standards
compliance, performance and portability.
```

Além das informações que você obtém a partir do nome do pacote em si, a opção `-qi` (*query information* / consultar informações) permite que você veja quem construiu o pacote (Red Hat, Inc.), quando ele foi construído e quando ele foi instalado. O grupo em que o pacote está (Applications/Internet), seu tamanho e o licenciamento são listados. Para saber mais sobre o pacote, a URL aponta para a página do projeto na internet e o Summary e a Description informam para que o pacote é utilizado.

De onde vêm os RPMs?

O software incluído com distribuições Linux ou construídos para trabalhar com essas distribuições vem de milhares de projetos de código aberto de todo o mundo. Esses projetos, conhecidos como *fornecedores de software upstream*, geralmente tornam o software disponível para quem os quiser, sob certas condições de licenciamento.

Uma distribuição Linux tem o código-fonte e o constrói na forma de arquivos binários. Então, a distribuição junta os binários com a documentação, arquivos de configuração e outros componentes disponibilizados pelo provedor *upstream*.

Depois de juntar todos os componentes no RPM, o pacote RPM é assinado (para que usuários possam testar a validade do pacote) e colocado em um repositório de RPMs para a distribuição e a arquitetura específicas (32-bit x86, x86 de 64 bits e assim por diante). O repositório é colocado em um CD ou DVD de instalação ou em um diretório que é disponibilizado como um servidor FTP, Web ou NFS.

Instalando RPMs

Quando você inicialmente instala um Fedora ou um Red Hat Enterprise Linux, muitos pacotes RPM individuais compõem essa instalação. Depois de instalar o Linux, você pode adicionar mais pacotes usando a interface gráfica do PackageKit (como descrito anteriormente). Consulte o Capítulo 9 para obter informações sobre a instalação do Linux.

A primeira ferramenta a ser desenvolvida para instalar pacotes RPM, porém, foi o comando `rpm`. Usando o `rpm`, você pode instalar, atualizar, consultar, validar e remover pacotes RPM. O comando, porém, tem algumas desvantagens importantes:

- **Dependências** — A maioria dos pacotes RPM dependem de algum outro software (biblioteca, arquivos executáveis etc.) que é instalado no sistema para o pacote funcionar. Quando você tenta instalar um pacote com o `rpm`, se um pacote dependente não estiver instalado, a instalação

falhará, dizendo quais componentes são necessários. Nesse ponto, você tem de escavar para encontrar o pacote que contém esse componente. Depois de encontrá-lo e tentar instalá-lo, esse pacote dependente pode ter dependências que você precisa instalar para fazê-lo funcionar. Essa situação foi carinhosamente chamada de “inferno das dependências” e foi muitas vezes usada como um exemplo de que pacotes DEB eram melhores do que RPMs. As ferramentas de empacotamento DEB foram feitas para resolver automaticamente as dependências de pacotes bem antes das ferramentas de empacotamento relacionadas com RPM fazerem isso.

- **Localização dos RPMs** — O comando `rpm` espera que você forneça a localização exata do arquivo RPM quando tenta instalá-lo. Em outras palavras, você teria de indicar `firefox-10.0.3-1.el6_2.x86_64.rpm` como uma opção se o RPM estivesse no diretório atual ou `http://example.com/firefox-10.0.3-1.el6_2.x86_64.rpm` se fosse em um servidor.

À medida que o Red Hat Linux e outros aplicativos baseados em RPM cresciam em popularidade, tornou-se evidente que algo tinha de ser feito para tornar o pacote de instalação mais conveniente. A resposta foi o recurso YUM.

Gerenciando pacotes RPM com o YUM

O projeto Yellowdog Updater Modified (YUM) foi criado para resolver a dor de cabeça de gerenciar dependências com pacotes RPM. Sua principal contribuição foi a de parar de pensar em pacotes RPM como componentes individuais e pensar neles como partes de repositórios de software maiores.

Com repositórios, o problema de lidar com dependências não recai na pessoa que instalava o software, mas na distribuição Linux ou no distribuidor de software independente que disponibiliza o software. Assim, por exemplo, caberia ao projeto Fedora se certificar de que todos os componentes necessários para cada pacote em sua distribuição Linux podem ser disponibilizados por algum outro pacote no repositório.

Repositórios também podem se basear um no outro. Assim, por exemplo, o repositório `rpmfusion.org` poderia supor que um usuário já teve acesso ao repositório Fedora principal. Portanto, se um pacote que está sendo instalado a partir de `rpmfusion.org` precisasse de uma biblioteca ou de comando a partir do repositório Fedora principal, o pacote Fedora poderia ser baixado e instalado ao mesmo tempo em que você instala o pacote `rpmfusion.org`.

Os repositórios yum podem ser colocados em um diretório em um servidor web (`http://`), servidor FTP (`ftp://`) ou mídia local, como um CD, DVD ou diretório local (`file://`). A localização desses repositórios, então, seria armazenada no sistema do usuário no arquivo `/etc/yum.conf` ou, mais geralmente, em arquivos de configuração separados no diretório `/etc/yum.repos.d`.

Entendendo como funciona o yum

A sintaxe básica do comando `yum` é:

```
# yum [options] command
```

Usando essa sintaxe, você pode encontrar e consultar as informações de pacotes, saber mais sobre grupos de pacotes, atualizar ou excluir pacotes, para citar alguns recursos. Com o repositório YUM e a configuração no lugar, um usuário pode instalar um pacote simplesmente digitando algo assim:

```
# yum install firefox
```

O usuário só precisa saber o nome do pacote (que pode ser consultado em diferentes formas, como descrito na seção “Pesquisando pacotes”, mais adiante neste capítulo). A instalação YUM encontra a versão mais recente desse pacote disponível no repositório, transfere-o para o sistema local e instala-o. A Figura 10.3 mostra o que acontece quando alguém instala um pacote usando o comando `yum`.

O resultado de um comando `yum install package` é que o pacote solicitado é copiado a partir do repositório `yum` para o sistema local. Os arquivos do pacote são colocados no sistema de arquivos onde são necessários (`/etc`, `/bin`, `/usr/share/man` etc.). As informações sobre o pacote são armazenadas no banco de dados RPM local, onde podem ser consultadas.

lades locais e remotas ao instalar um RPM com o YUM.

Para ganhar mais experiência com a instalação do YUM e ver onde há oportunidades para que você personalize como o YUM funciona em seu sistema, siga as descrições de cada fase do processo de instalação do yum ilustradas na Figura 10.3 e definidas aqui.

1. Verificando `/etc/yum.conf`

Quando qualquer comando `yum` é iniciado, ele verifica as configurações padrão no arquivo `/etc/yum.conf`. O arquivo `/etc/yum.conf` é de configuração básica do YUM. Você também pode identificar a localização dos repositórios aqui, embora o diretório `/etc/yum.repos.d` seja o local mais típico para identificar repositórios. Eis um exemplo de `/etc/yum.conf` em um sistema RHEL 6.2:

```
[main]
cachedir=/var/cache/yum/$basearch/$releasever
keepcache=0
debuglevel=2
logfile=/var/log/yum.log
exactarch=1
plugins=1
gpgcheck=1
```

As configurações em `yum.conf` dizem ao YUM onde manter os arquivos de cache (`/var/cache/yum`) e entradas de log (`/var/log/yum.log`) e se deve ou não manter arquivos de cache depois que um pacote é instalado (0 significa não). Você pode aumentar o valor `debuglevel` no arquivo `yum.conf` para acima de 2 se quiser ver mais detalhes em seus arquivos de log.

Em seguida, você pode ver se a arquitetura exata (x86, x86_64 etc.) deve ser combinada ao escolher pacotes para instalar (1 significa sim) e se deve ou não usar plugins (1 significa sim) para permitir coisas como listas negras, listas brancas ou conectar-se à Red Hat Network para encontrar pacotes.

Por fim, `gpgcheck` diz se você deve ou não validar cada pacote com uma chave que você recebe de quem construiu o RPM. Para pacotes que vêm com o Fedora ou o RHEL, a chave vem incluída na distribuição para verificar todos os pacotes. Mas se tentar instalar pacotes que não são de sua distribuição, você precisa importar a chave necessária para assinar os pacotes ou desativar esse recurso (`gpgcheck=0`).

Para encontrar outros recursos que você pode definir no arquivo `yum.conf`, digite **man yum.conf**.

2. Verificando `/etc/sysconfig/rhn/up2date` (RHEL somente)

Para sistemas Red Hat Enterprise Linux, em vez de apontar para um repositório de software público único (como o Fedora), você registra seu sistema na Red Hat Network e compra direitos para baixar o software a partir de diferentes canais.

Quando o sistema está registrado na Red Hat Network, a informação é adicionada ao arquivo `/etc/sysconfig/rhn/up2date` para dizer ao yum onde encontrar pacotes Red Hat Enterprise Linux (a partir de um Red Hat Network hospedado ou de um servidor RHN Satellite).

3. Verificando arquivos `/etc/yum.repos.d/*.repo`

Repositórios de software podem ser ativados copiando arquivos que terminam em `.repo` para o diretório `/etc/yum.repos.d/` que apontam para a localização de um ou mais repositórios. No Fedora, até seus repositórios básicos são ativados a partir dos arquivos `.repo` nesse diretório.

O seguinte é um exemplo de um arquivo de configuração yum simples chamado `/etc/yum.repos.d/myrepo.repo`:

```
[myrepo]
name=My repository of software packages
baseurl=http://myrepo.example.com/pub/myrepo
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/MYOWNKEY
```

Cada entrada no repositório começa com o nome do repositório entre colchetes. A linha `name` contém uma descrição legível por humanos do repositório. A linha `baseurl` identifica o diretório que contém os arquivos rpm, o que pode ser uma entrada `httpd://`, `ftp://` ou `file://`.

A linha `enabled` indica se ou não a entrada está ativa. 1 é ativa e 0 é inativa. Se não houver uma linha `enabled`, a entrada está ativa. As duas últimas linhas do código anterior indicam se ou não verificar as assinaturas em pacotes nesse repositório. A linha `gpgkey` mostra a localização da chave que é usada para verificar os pacotes nesse repositório.

Você pode ter quantos quiser repositórios habilitados, mas tenha em mente que quando você usa comandos yum, cada repositório é verificado e metadados sobre todos os pacotes são baixados para o sistema local que está executando o comando yum. Assim, para ser mais eficiente, não ative repositórios dos quais você não precisa.

4. Baixando pacotes RPM e metadados de um repositório YUM

Depois que yum sabe os locais dos repositórios, os metadados provenientes do diretório `repodata` de cada repositório são baixados para o sistema local. De fato, é a existência de um diretório `repodata` em um diretório de RPMs que indica que ele é um repositório yum.

As informações de metadados são armazenadas no sistema local no diretório `/var/cache/yum`. Quaisquer outras pesquisas sobre pacotes, grupos de pacotes ou outras informações do repositório são reunidos a partir dos metadados em cache por um tempo determinado.

Depois que o tempo limite é atingido, yum recupera novos metadados se o comando yum for executado.

Por padrão, o tempo de espera é de 90 minutos. Você pode alterar esse período definindo `metadata_expire` no arquivo `/etc/yum.conf`. Remova os caracteres de comentário dessa linha e altere o número de minutos.

Em seguida, o `yum` olha para os pacotes cuja instalação você solicitou e verifica se há pacotes dependentes exigidos. Com a lista de pacotes reunidos, o `yum` pergunta se está tudo certo para baixar todos os pacotes. Se você escolher sim, os pacotes são baixados para os diretórios de cache e instalados.

5. Pacotes RPM instalados para o sistema de arquivos Linux

Uma vez que todos os pacotes necessários são transferidos para os diretórios de cache, o comando `yum` roda comandos `rpm` para instalar cada pacote. Se um pacote contiver os scripts de pré-instalação (que podem criar uma conta de usuário especial ou criar diretórios), esses scripts são executados. O conteúdo dos pacotes é copiado para o sistema de arquivos (comandos, arquivos de configuração, documentos etc.). Então, os scripts de pós-instalação são executados. (Scripts de pós-instalação executam comandos adicionais necessários para configurar o sistema depois que cada pacote é instalado).

6. Armazene o repositório de metadados YUM no banco de dados RPM local

Os metadados contidos em cada pacote RPM instalado são, por fim, copiados para o banco de dados RPM local. O banco de dados RPM está contido nos arquivos que são armazenados no diretório `/var/lib/rpm`.

Depois que as informações sobre os pacotes instalados estão no banco de dados RPM local, você pode fazer vários tipos de consulta a esse banco de dados. Você pode ver quais pacotes estão instalados, listar os componentes desses pacotes e ver scripts ou mudar logs associados a cada pacote. Você pode até validar os pacotes instalados no banco de dados RPM para ver se alguém adulterou componentes instalados.

O comando `rpm` (descrito na seção “Instalando, consultando e verificando software com o comando `rpm`”, mais adiante neste capítulo) é a melhor ferramenta para consultar o banco de dados RPM. Você pode executar consultas individuais com `rpm` ou usá-lo em scripts para produzir relatórios ou executar consultas comuns repetidamente.

Agora que você entende o funcionamento básico do comando `yum`, seu sistema Fedora deve estar automaticamente configurado para se conectar ao repositório principal e ao de atualizações do Fedora. Você pode ir em frente e tentar algumas linhas de comando do `yum` para instalar pacotes agora. Ou você pode ativar repositórios YUM de terceiros para baixar software.

Usando o YUM com repositórios de software de terceiros

Os repositórios de software do Fedora e do Red Hat Enterprise Linux foram preparados para conter apenas software que atenda aos critérios que o tornam aberto e redistribuível. Em alguns casos, porém, você pode querer ir além desses repositórios. Antes de fazer isso, você deve compreender que alguns repositórios de terceiros:

- Têm exigências menos rigorosas para redistribuição e liberdade em relação às restrições de patentes do que os repositórios do Fedora e do RHEL;

- Podem introduzir alguns conflitos de software;
- Podem incluir o software que não é de código-fonte aberto e, embora possam ser livres para uso pessoal, podem não ser redistribuíveis;
- Podem retardar o processo de instalação de todos os seus pacotes (porque metadados são baixados para cada repositório que você tiver habilitado).

Por essas razões, recomendo que você não habilite todos os repositórios de software extra ou habilite somente o repositório RPM Fusion (<http://rpmfusion.org>) em primeiro lugar para o Fedora e o repositório EPEL (<http://fedoraproject.org/wiki/EPEL>) para o Red Hat Enterprise Linux. O RPM Fusion apresenta uma fusão de vários populares repositórios Fedora de terceiros (Freshrpms, Livna.org e Dribble). Veja a FAQ do repositório para obter detalhes (<http://rpmfusion.org/FAQ>). Para habilitar o repositório RPM Fusion no Fedora, faça o seguinte:

1. Abra uma janela de terminal.
2. Digite **su -** e insira a senha de root quando solicitado.
3. Digite o seguinte comando em uma linha sem espaço entre a barra e rpmfusion. (Tive de quebrar a linha em duas porque era muito grande para caber em uma linha na página impressa deste livro, então certifique-se de digitar o endereço completo em uma linha sem espaço):

```
# rpm -Uvh http://download1.rpmfusion.org/free/fedora/  
rpmfusion-free-release-stable.noarch.rpm
```

O repositório não livre do RPM Fusion contém coisas como codecs necessários para reproduzir muitos formatos populares de multimídia. Para habilitar o repositório não livre no Fedora, digite o seguinte (novamente, digite as seguintes linhas em uma única linha, sem espaço entre os dois):

```
# rpm -Uvh http://download1.rpmfusion.org/nonfree/fedora/  
rpmfusion-nonfree-release-stable.noarch.rpm
```

A maioria dos repositórios de terceiros que podem interessar-lhe contém software que não é de código-fonte aberto. Por exemplo, se você deseja instalar o Adobe Flash plug-in para Linux, baixe o pacote repositório YUM da Adobe e você pode usar o comando `yum` para instalar o plug-in Flash e obter atualizações mais tarde, executando o comando `yum update`, quando houver atualizações disponíveis.

Gerenciando software com o comando YUM

O comando `yum` tem dezenas de subcomandos que você pode usar para trabalhar com pacotes RPM em seu sistema. As próximas seções fornecem alguns exemplos de linhas de comando `yum` úteis para procurar, instalar, consultar e atualizar pacotes associados com seus repositórios YUM. Também há uma seção que descreve como remover os pacotes instalados com o comando `yum`.

ta

adados, descrevendo o conteúdo dos repositórios YUM, são baixados de cada um de seus repositórios YUM listados na primeira vez que você executa um comando `yum`. Metadados são baixados novamente após o tempo `data_expire` ser atingido (90 minutos, por padrão). Quanto mais repositórios YUM você habilita e quanto maiores forem, mais tempo esse download pode demorar. Você pode reduzir esse tempo de download aumentando o tempo de expiração (no arquivo `/etc/yum.conf`) ou não habilitando repositórios dos quais você não precise.

Procurando por pacotes

Usando diferentes subcomandos de pesquisa, você pode encontrar pacotes com base em palavras-chave, conteúdo ou outros atributos deles.

Digamos que você queira experimentar um editor de texto diferente, mas não consegue lembrar o nome daquele que você queria. Você pode usar o subcomando `search` para procurar o termo “editor” no nome ou na descrição:

```
# yum search editor
...
eclipse-veditor.noarch : Eclipse-based Verilog/VHDL plugin
ed.x86_64 : The GNU line editor
emacs.x86_64 : GNU Emacs text editor
```

A pesquisa revelou uma longa lista de pacotes contendo “editor” no nome ou na descrição. O que eu estava procurando chamava-se `emacs`. Para obter informações sobre o pacote, posso usar o subcomando `info`:

```
# yum info emacs
Name      : emacs
Arch      : x86_64
Epoch    : 1
Version   : 23.1
Release   : 21.el6_2.3
Size      : 2.2 M
Repo      : production-rhel-x86_64-workstation-6
Summary   : GNU Emacs text editor
License   : GPLv3+
Description : Emacs is a powerful, customizable, self-documenting,
           : modeless text editor. Emacs contains special code
           : editing features, a scripting language (elisp), and ...
```

Se você souber o comando, o nome do arquivo de configuração ou biblioteca que você quer, mas não souber qual pacote é, use o subcomando `provides` para procurá-lo. Aqui você pode ver que o comando `dvdrecord` é parte do pacote `wodim`:

```
# yum provides dvdrecord
wodim-1.1.9-11.el6.x86_64 : A command line CD/DVD recording program
Repo      : production-rhel-x86_64-workstation-6
Matched from:
Other     : dvdrecord
```

O subcomando `list` pode ser utilizado para listar nomes de pacotes de diferentes maneiras. Use-o com um nome de base do pacote para encontrar a versão e o repositório de um pacote. Você pode listar apenas os pacotes que estão disponíveis (`available`) ou instalados (`installed`) ou você pode listar todos (`all`) eles.

```

# yum list emacs
emacs.i686      1:23.3-9.fc16      updates
# yum list available
389-admin.i686      1.1.23-1.fc16      fedora
389-admin-console.noarch  1.1.8-2.fc16      updates
389-admin-console-doc.noarch  1.1.8-2.fc16      updates
...
# yum list installed
Installed Packages
ConsoleKit.i686      0.4.5-1.fc15      @koji-override-0/$releasever
ConsoleKit-libs.i686  0.4.5-1.fc15      @koji-override-0/$releasever
ConsoleKit-x11.i686  0.4.5-1.fc15      @koji-override-0/$releasever
# yum list all

```

Se encontrar um pacote, mas quiser ver os componentes de que ele depende, você pode usar o subcomando `deplist`. Com `deplist`, você pode ver os componentes (dependência), mas também o pacote em que esse componente vem (provedor). Usar `deplist` pode ajudar se nenhum pacote estiver disponível para fornecer uma dependência, mas você quiser saber qual é o componente para poder pesquisá-lo em outros repositórios. Considere o seguinte exemplo:

```

# yum deplist emacs | less
Finding dependencies:
package: emacs.x86_64 1:23.1-21.el6_2.3
  dependency: libc.so.6 (GLIBC_2.3.4) (64bit)
  provider: glibc.x86_64 2.12-1.7.el6
  dependency: libasound.so.2 (ALSA_0.9.0rc4) (64bit)

```

Instalando e removendo pacotes

O subcomando `install` permite instalar um ou mais pacotes, juntamente com todos os pacotes dependentes necessários. Com `yum install`, múltiplos repositórios podem ser pesquisados para satisfazer as dependências necessárias. Considere o seguinte exemplo de `yum install`:

```

# yum install emacs
...
=====
Package Arch Version Repository Size
=====
Installing:
emacs x86_64 1:23.1-21.el6_2.3 rhel-x86_64-workstation-6 2.2 M
Installing for dependencies:
emacs-common x86_64 1:23.1-21.el6_2.3 rhel-x86_64-ws-6 18 M
Transaction Summary
=====
Install 2 Package(s)
Total download size: 20 M
Installed size: 0
Is this ok [y/N]: y

```

Você pode ver aqui que o `emacs` requer que `emacs-common` seja instalado e, portanto, ambos são colocados na fila para instalação. Os dois pacotes juntos têm 20MB. Pressionar `y` instala-os. Você pode colocar um `-y` na linha de comando (logo após o comando `yum`) para evitar ter de pressionar `y` para instalar os pacotes, mas, pessoalmente, prefiro ver todos os pacotes que serão instalados antes de concordar com a instalação.

Você pode reinstalar um pacote previamente instalado se excluir por engano componentes dele. Se você tentar uma instalação regular, o sistema responderá com “nothing to do” (“nada a fazer”). Você deve, em

vez disso, usar o subcomando `reinstall`. Por exemplo, digamos que você instalou o pacote `zsh` e depois excluiu `/bin/zsh` por engano. Você pode restaurar os componentes em falta, digitando o seguinte:

```
# yum reinstall zsh
```

Você pode remover um único pacote com o subcomando `erase`. Por exemplo, para apagar o pacote `emacs`, você pode digitar o seguinte:

```
# yum erase emacs
Dependencies Resolved
=====
Package
Arch Version           Repository             Size
=====
Removing:
emacs x86_64 1:23.1-21.el6_2.3 @rhel-x86_64-workstation-6 11 M
Transaction Summary
=====
Remove      1 Package(s)
Installed size: 11 M
Is this ok [y/N]: y
```

Observe que, apesar de dois pacotes terem sido instalados quando o `emacs` foi instalado, somente o pacote `emacs` foi removido com o subcomando `erase`. Embora o `emacs` exija que o `emacs-common` seja instalado, o `emacs-common` não depende do `emacs` e poderia, portanto, permanecer no sistema sem quebrar as dependências. Executar `yum remove emacs-common` teria removido os dois pacotes.

Um método alternativo para remover um conjunto de pacotes que você instalou é usar o subcomando `history`. Usando `history`, você pode ver as atividades do seu `yum` e desfazer uma transação inteira. Em outras palavras, todos os pacotes que você instalou podem ser desinstalados utilizando a opção `undo` do subcomando `history`. Por exemplo:

```
# yum history
ID | Login user           | Date and time       | Action(s) | Altered
-----
96 | Chris Negus <cnegus> | 2012-04-10 06:25   | Install   | 2
...
# yum history info 96
Transaction ID : 96
...
Command Line   : install emacs
...
# yum history undo 96
Undoing transaction 96, from Tue Apr 10 06:25:41 2012
Install emacs-1:23.1-21.el6_2.3.x86_64 @rhel-x86_64-workstation-6
Dep-Install emacs-common-1:23.1-21.el6_2.3.x86_64 @rhel-workstation-6
Resolving Dependencies
...
```

Antes de desfazer a transação, você pode vê-la para saber exatamente quais pacotes estão envolvidos. Ver a transação pode salvá-lo de excluir por engano pacotes que você deseja manter. Desfazendo a transação 96, você pode remover os pacotes que foram instalados durante essa transação. Se você estiver tentando desfazer uma instalação que inclui dezenas ou mesmo centenas de pacotes, `undo` pode ser uma opção muito útil.

Atualizando pacotes

À medida que novas releases de um pacote se tornam disponíveis, elas, às vezes, são colocadas em repositórios de atualização separados ou simplesmente adicionadas ao repositório original. Caso várias versões de um pacote estejam disponíveis (quer no mesmo repositório ou outro repositório habilitado), o comando `yum` fornece a versão mais recente quando você instala um pacote. Se uma nova versão aparecer mais tarde, você pode baixar e instalar a nova versão do pacote usando o subcomando `update`.

O subcomando `check-update`, pode verificar se há atualizações. O subcomando `update` pode ser usado para atualizar um único pacote ou para obter atualizações para todos os pacotes que estão instalados atualmente e têm uma atualização disponível. Por exemplo:

```
# yum check-update
...
freetype.x86_64 2.3.11-6.el6_2.9 rhel-x86_server-6
libtiff.x86_64 3.9.4-5.el6_2 rhel-x86_server-6
# yum update
...
tzdata          noarch 2012b-3.el6 rhel-x86_64-workstation-6 441 k
tzdata-java     noarch 2012b-3.el6 rhel-x86_64-workstation-6 154 k
Transaction Summary
=====
Upgrade      38 Package(s)
Total download size: 50 M
Is this ok [y/N]: y
# yum update cups
```

O código anterior solicitou para atualizar o pacote `cups`. Se outros pacotes dependentes precisarem ser atualizados para atualizar `cups`, eles também serão baixados e instalados.

Atualizando grupos de pacotes

Para facilitar o gerenciamento de um conjunto de pacotes de uma só vez, o YUM suporta grupos de pacotes. Por exemplo, você pode instalar o GNOME Desktop Environment (para obter um desktop inteiro) ou Virtualization (a fim de obter os pacotes necessários para configurar o computador como uma máquina virtual). Você pode começar executando o subcomando `groupinstall` para ver uma lista de nomes de grupo:

```
# yum groupinstall | less
Installed Groups:
  Administration Tools
  Design Suite
  Dial-up Networking Support
...
Available Groups:
  Authoring and Publishing
  Base
  Books and Guides
...
```

Digamos que você queria experimentar um desktop diferente. Você vê o LXDE e quer saber o que está nesse grupo. Para saber, use o subcomando `groupinfo`:

```
# yum groupinfo LXDE
```

```
Group: LXDE
```

```
Description: LXDE is a lightweight X11 desktop environment...
```

```
Mandatory Packages:
```

```
lxde-common
```

```
lxmenu-data
```

```
...
```

Além de mostrar uma descrição do grupo, `groupinfo` mostra pacotes obrigatórios (aqueles que sempre são instalados com o grupo), pacotes padrão (aqueles que são instalados por padrão, mas podem ser excluídos) e pacotes opcionais (que fazem parte do grupo, mas não são instalados por padrão). Quando você usa ferramentas gráficas para instalar grupos de pacotes, pode desmarcar pacotes padrão ou marcar pacotes opcionais para definir se ou não eles são instalados com o grupo.

Se você decidir que quer instalar um grupo de pacotes, use o subcomando `groupinstall`:

```
# yum groupinstall LXDE
```

Esse `groupinstall` resultou na instalação de 23 pacotes do grupo e 9 pacotes extras. Se decidir que não gosta do grupo de pacotes, você pode removê-lo inteiramente de uma vez usando o subcomando `groupremove`:

```
# yum groupremove LXDE
```

Mantendo o banco de dados e o cache de pacotes RPM

Há vários subcomandos do `yum` que podem ajudar você a fazer as tarefas de manutenção, como verificar problemas com seu banco de dados RPM ou limpar o cache. O recurso YUM tem ferramentas para manter seus pacotes RPM e manter o software de seu sistema eficiente e seguro.

Limpar o cache é algo que é recomendável fazer de vez em quando. Se você decidir manter os pacotes baixados depois que eles forem instalados (eles são removidos por padrão, com base na configuração `keepcache=0` no arquivo `/etc/yum.conf`), seus diretórios de cache (em `/var/cache/yum`) podem ficar cheios. Metadados armazenados em diretórios de cache podem ser apagados, fazendo com que novos metadados sejam baixados de todos os repositórios YUM habilitados da próxima vez que o comando `yum` for executado. Eis algumas formas de limpar essas informações:

```
# yum clean packages
```

```
Cleaning repos: rhel-x86_64-server-6
```

```
7 package files removed
```

```
# yum clean metadata
```

```
43 metadata files removed
```

```
13 sqlite files removed
```

```
# yum clean all
```

```
Cleaning repos: rhel-x86_64-server-6
```

```
Cleaning up Everything
```


Embora improvável, é possível que o banco de dados RPM seja corrompido. Isso pode acontecer se algo inesperado ocorrer, como puxar o cabo de energia quando um pacote está parcialmente instalado. Você pode verificar o banco de dados RPM para procurar erros (`yum check`) ou apenas reconstruir os arquivos de banco de dados do RPM, como segue:

```
# yum check
check all
# yum clean rpmdb
Cleaning repos: rhel-x86_64-server-6
4 rpmdb files removed
```

Dos exemplos de `yum clean` nas três linhas de comando precedentes, todos removem os dados em cache dos subdiretórios `/var/cache/yum`, exceto para o exemplo `rpmdb`. Esse comando removeu os arquivos `db*` do diretório `/var/lib/rpm` (recriando os arquivos de banco de dados para limpar qualquer problema).

A `rpmdb` é uma das poucas opções do `yum` que é usada para trabalhar com o banco de dados RPM diretamente. Em geral, o `yum` é usado para manipular repositórios `yum`. O comando mais adequado para trabalhar com o banco de dados RPM local é o `rpm`.

Baixando RPMs de um repositório yum

Se quiser apenas examinar um pacote sem realmente instalá-lo, você pode usar o comando `yumdownloader`. Executar esse comando faz com que o pacote que você nomeou seja baixado a partir do repositório YUM e copiado para o diretório atual.

Por exemplo, para baixar a última versão do pacote de navegador Firefox do repositório YUM para o diretório atual, digite o seguinte:

```
# yumdownloader firefox
...
firefox-10.0.5.1.el6.2.x86_64.rpm
```

Com o pacote RPM agora residindo em seu diretório atual, você pode usar uma variedade de comandos `rpm` para consultar ou usar esse pacote de diferentes formas (como descrito na próxima seção).

Instalando, consultando e verificando software com o comando rpm

Há uma riqueza de informações sobre os pacotes instalados no banco de dados RPM local. O comando `rpm` contém dezenas de opções para que você possa encontrar informações sobre cada pacote, como os arquivos que ele contém, quem o criou, quando foi instalado, seu tamanho e muitos outros atributos. Como o banco de dados contém impressões digitais (`md5sums`) de cada arquivo em cada pacote, ele pode ser consultado com o RPM para descobrir se arquivos de qualquer pacote foram adulterados.

O comando `rpm` ainda pode fazer a instalação básica e atividades de atualização, embora a maioria das pessoas só use o `rpm` dessa forma quando há um pacote residindo no diretório local, pronto para ser instalado. Então, vamos pegar um em nosso diretório local para trabalhar. Digite o seguinte para baixar a última versão do pacote `zsh`:

```
# yumdownloader zsh
zsh-4.3.17-1.1.31-2.fc16.noarch | 2.3 MB 00:03
```

Com o pacote `zsh` copiado para seu diretório atual, experimente alguns comandos `rpm` nele.

Instalando e removendo pacotes com o comando rpm

Para instalar um pacote com o comando `rpm`, digite:

```
# rpm -i zsh-4.3.17-1.1.31-2.fc16.noarch.rpm
```

Observe que o nome do pacote inteiro é dado para instalar com o `rpm`, não apenas o nome de base do pacote. Se uma versão anterior do `zsh` estivesse instalada, você poderia atualizar o pacote usando `-U`. Muitas vezes, as pessoas usam `-h` e `-v` para obter sinais de jogo da velha impressos e uma saída mais detalhada durante a atualização:

```
# rpm -Uhv zsh-4.3.17-1.1.31-2.fc16.noarch.rpm
Preparing...      ##### [100%]
 1:zsh            ##### [100%]
```

Enquanto uma instalação (`-i`) só instalará um pacote se ele não estiver instalado, uma atualização (`-U`) instalará o pacote quer ele já esteja instalado ou não. Um terceiro tipo de instalação, chamado *freshen* (`-F`), instala um pacote apenas se existir uma versão anterior instalada no computador. Por exemplo:

```
# rpm -Fhv *.rpm
```

Você pode usar o comando *freshen* anterior se estiver em um diretório que contém milhares de RPMs, mas você só quer atualizar aqueles que já estavam instalados (em uma versão anterior) em seu sistema e pular aqueles que ainda não foram instalados.

Há algumas opções interessantes que você pode adicionar a qualquer das suas opções de instalação. A opção `--replacepkgs` permite a reinstalação de uma versão existente de um pacote (se, por exemplo, você tiver excluído por engano alguns componentes) e `--oldpackage` permite instalar uma versão anterior de um pacote.

```
# rpm -Uhv --replacepkgs emacs-common-23.3-9.fc16i686.rpm
# rpm -Uhv --oldpackage zsh-4.1.0-2.fc16.noarch.rpm
```

Você pode remover um pacote com a opção `-e`. Você só precisa do nome de base de um pacote para removê-lo. Por exemplo:

```
# rpm -e emacs
```

O comando `rpm -e emacs` seria bem-sucedido porque nenhum outro pacote depende do `emacs`. Mas ele deixaria para trás `emacs-common`, que foi instalado como uma dependência para o `emacs`. Se você tentou remover `emacs-common` primeiro, esse comando falharia com uma mensagem “Failed dependencies” (“Falha de dependências”).

Consultando informações do rpm

Depois que o pacote está instalado, você pode consultar informações sobre ele. Usando a opção `-q`, você pode ver informações sobre o pacote, incluindo uma descrição (`-qi`), uma lista de arquivos (`-ql`), a documentação (`-qd`) e os arquivos de configuração (`-qc`).

```
# rpm -qi zsh
```

```
Name : zsh
```

```
Version : 4.3.17
```

```
Release : 1.fc16
```

```
...
```

```
# rpm -ql zsh
```

```
/bin/zsh
```

```
/etc/skel/.zshrc
```

```
/etc/zlogin
```

```
/etc/zlogout
```

```
...
```

```
# rpm -qd zsh
```

```
/usr/share/doc/zsh-4.3.17/FAQ
```

```
/usr/share/doc/zsh-4.3.17/FEATURES
```

```
/usr/share/doc/zsh-4.3.17/README
```

```
# rpm -qc zsh
```

```
/etc/skel/.zshrc
```

```
/etc/zlogin
```

```
/etc/zlogout
```

Há opções para consultar qualquer informação contida em um RPM. Você pode encontrar o que precisa para um RPM ser instalado (`--requires`), que versão do software um pacote oferece (`--provides`), quais scripts são executados antes e depois de um RPM ser instalado ou removido (`--scripts`) e que mudanças foram feitas em um RPM (`--changelog`).

```
# rpm -q --requires emacs-common
```

```
/bin/sh
```

```
/sbin/install-info
```

```
/usr/bin/perl
```

```
...
```

```
# rpm -q --provides emacs-common
```

```

config(emacs-common) = 1:23.3-9.fc16
emacs-common = 1:23.3-9.fc16
# rpm -q --scripts httpd
# Add the "apache" user
/usr/sbin/useradd -c "Apache" -u 48 \
-s /sbin/nologin -r -d /var/www apache 2> /dev/null || :
postinstall scriptlet (using /bin/sh):
# Register the httpd service
/sbin/chkconfig --add httpd
...
# rpm -q --changelog httpd | less
* Mon Mar 05 2012 Jan Kaluza <jkaluza@redhat.com> - 2.2.22-2
- fix #751591 - start httpd after network filesystems
...

```

Nos dois exemplos anteriores, você pode ver que os scripts dentro do pacote `httpd` adicionam um usuário `apache` durante a instalação e ativam o serviço `httpd` com `chkconfig`. A opção `--changelog` permite ver por que foram feitas mudanças em cada versão do pacote. O texto `fix #` representa um erro corrigido que você pode verificar em <http://bugzilla.redhat.com>.

Usando um recurso chamado `--queryformat`, você pode consultar as diferentes tags de informações e gerar uma saída delas como quiser. Execute a opção `--querytags` para ver todas as tags que estão disponíveis:

```

# rpm --querytags | less
ARCH
ARCHIVESIZE
BASENAMES
BUGURL
...
# rpm -q binutils --queryformat "The package is %{NAME} \
    and the release is %{RELEASE}\n"
The package is binutils and the release is 6.fc16

```

Todas as consultas que você fez até agora têm sido no banco de dados RPM local. Ao adicionar `-p` para as opções de consulta, você pode consultar um arquivo RPM que reside em seu diretório local. A opção `-p` é uma ótima maneira de pesquisar um pacote que você recebeu de alguém a fim de investigar o que ele é antes de instalá-lo em seu sistema.

Se você não tiver o pacote `zsh`, baixe-o agora e copie-o para seu diretório local (`yumdownloader zsh`). Então, execute alguns comandos de consulta sobre ele.

```

# rpm -qip zsh-4.3.17-1.fc16.i686.rpm
# rpm -qlp zsh-4.3.17-1.fc16.i686.rpm
# rpm -qdp zsh-4.3.17-1.fc16.i686.rpm

```

```
# rpm -qcp zsh-4.3.17-1.fc16.i686.rpm
```

Verificando pacotes RPM

Usando a opção `-V`, você pode verificar os pacotes instalados em seu sistema para ver se os componentes foram alterados desde que os pacotes foram inicialmente instalados. Embora seja normal que arquivos de configuração mudem ao longo do tempo, não é normal que binários (os comandos em `/bin`, `/sbin` etc.) mudem após a instalação. Binários alterados são provavelmente uma indicação de que seu sistema foi invadido.

Nesse exemplo, vou instalar o pacote `zsh` e adulterá-lo. Se você quiser tentar, juntamente com os exemplos, certifique-se de remover ou reinstalar o pacote depois que terminar.

```
# rpm -i zsh-4.3.17-1.fc16.i686.rpm
# echo hello > /bin/zsh
# rm /etc/zshrc
# rpm -V zsh
S.5....T. /bin/zsh
missing c /etc/zshrc
```

Nessa saída, você pode ver que o arquivo `/bin/zsh` foi adulterado e que `/etc/zshrc` foi removido. Sempre que vir uma letra ou um número em vez de um ponto na saída de `rpm -V`, isso é uma indicação do que foi alterado. Letras que podem substituir os pontos (por ordem) são as seguintes:

```
S      file Size differs
M      Mode differs (includes permissions and file type)
5      MD5 sum differs
D      Device major/minor number mismatch
L      readLink(2) path mismatch
U      User ownership differs
G      Group ownership differs
T      mTime differs
P      caPabilities differ
```

Esses indicadores são da seção `Verify` da página `man rpm`. No meu exemplo, você pode ver que o tamanho do arquivo mudou (S), a `md5sum` comparada com a impressão digital do arquivo mudou (5) e a data/hora da última modificação (T) no arquivo é diferente.

Para restaurar o pacote ao seu estado original, use `rpm` com a opção `—replacepkgs`, como mostrado a seguir. (O comando `yum reinstall zsh` também funcionaria). Em seguida, verifique-o com `-V` novamente. O fato de não haver nenhuma saída de `-V` significa que todos os arquivos voltaram ao seu estado original.

```
# rpm -i --replacepks zsh-4.3.17-1.fc16.i686.rpm
# rpm -V zsh
```

Uma boa prática é fazer backup de seu banco de dados RPM (a partir de `/var/lib/rpm`) e copiá-lo para alguma mídia somente leitura (como um CD). Depois, quando você for verificar os pacotes que suspeita terem sido adulterados, você sabe que não está comparando com um banco de dados que também foi adulterado.

Gerenciando software na empresa

Nesse ponto, você deve ter um bom conhecimento de como instalar, consultar, remover ou manipular pacotes com ferramentas gráficas, o comando `yum` e o comando `rpm`. Quando você começar a trabalhar com arquivos RPM em uma grande empresa, precisará estender esse conhecimento.

Recursos usados para gerenciar pacotes RPM na empresa com o Red Hat Enterprise Linux oferecem um pouco mais de complexidade e muito mais poder. Em vez de ter um grande repositório de software, como o Fedora, o RHEL fornece implantação pela Red Hat Network, o que requer uma assinatura paga e direitos sobre uma variedade de canais de software (RHEL, Red Hat Enterprise Virtualization, Red Hat Cluster Suite etc.).

Em termos de computação corporativa, um dos grandes benefícios do projeto de pacotes RPM é que seu gerenciamento pode ser automatizado. Outros esquemas de empacotamento Linux permitem fazer os pacotes pararem e solicitarem a você informações quando eles começarem a ser instalados (tais como pedir um local de diretório ou um nome de usuário). Pacotes RPM são instalados sem interrupção, oferecendo algumas das seguintes vantagens:

- **Arquivos de kickstart** — Todas as perguntas que você responde durante uma instalação manual e todos os pacotes que seleciona podem ser adicionados em um arquivo chamado arquivo de kickstart. Quando você inicia um instalador do Fedora ou do Red Hat Enterprise Linux, você pode fornecer um arquivo de kickstart no prompt de inicialização. A partir desse ponto, o processo de instalação se desenvolve por conta própria.
- **PXE** — Você pode configurar um servidor PXE para permitir que computadores clientes inicializem um kernel anaconda (instalador) e selecione um arquivo de kickstart. Um computador completamente em branco com uma placa de rede que suporta inicialização PXE pode simplesmente inicializar a partir da sua placa de rede para carregar uma nova instalação. Em outras palavras, ligue o computador e se ele encontrar a placa de rede em sua ordem de boot, alguns minutos mais tarde você pode ter um sistema novo em folha instalado, configurado de acordo com suas especificações exatas e sem intervenção.
- **Servidor satélite (Spacewalk)** — Sistemas Red Hat Enterprise Linux podem ser implantados usando o que é chamado de Satellite Server (o projeto de código aberto é chamado Spacewalk). Integrados ao Satellite Server estão os mesmos recursos que você tem na Red Hat Network para gerenciar e implantar novos sistemas e atualizações. Sem se conectar diretamente, sistemas RHEL podem ser configurados para receber atualizações de software em horários definidos a partir do servidor de satélite. Conjuntos de pacotes chamados Errata, que corrigem problemas específicos,

podem ser rápida e automaticamente implantados nos sistemas que precisam deles.

Descrições de como usar arquivos de kickstart, servidores satélites e outros recursos de instalação prontos para empresas estão fora do escopo deste livro. Mas o conhecimento que você ganhou ao aprender a usar o YUM e o RPM servirá como uma base sólida para qualquer trabalho de instalação de software RHEL que fizer no futuro.

Resumo

O empacotamento de software no Fedora, no Red Hat Enterprise Linux e em sistemas relacionados é fornecido usando pacotes de software com base nas ferramentas RPM Package Manager (RPM). Há ferramentas gráficas fáceis de usar, como a janela PackageKit Add/Remove Software para encontrar e instalar pacotes. As principais ferramentas de linha de comando incluem os comandos `yum` e `rpm`.

Utilizando qualquer uma dessas ferramentas de gerenciamento de software, você pode instalar, consultar, verificar, atualizar e remover pacotes. Você também pode fazer tarefas de manutenção, como limpar os arquivos de cache e reconstruir o banco de dados RPM. Este capítulo descreve muitos dos recursos do PackageKit, YUM e RPM.

Com o sistema instalado e os pacotes de software que você precisa adicionados, é hora de configurar ainda mais seu sistema Fedora ou RHEL. Se você espera ter várias pessoas utilizando seu sistema, sua próxima tarefa poderia ser adicionar e também gerenciar contas de usuário no sistema. O Capítulo 11 descreve o gerenciamento de usuários no Fedora, RHEL e outros sistemas Linux.

Exercícios

Esses exercícios testam seu conhecimento para trabalhar com pacotes de software RPM no Fedora ou no Red Hat Enterprise Linux. Para fazer os exercícios, recomendo que você tenha um sistema Fedora à sua frente e uma conexão com a internet. (A maioria dos procedimentos irá funcionar igualmente bem em um sistema RHEL registrado.)

Você precisa ser capaz de alcançar os repositórios do Fedora (que devem ser configurados automaticamente). Se você empacar, soluções para as tarefas são mostradas no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. Pesquise o repositório YUM para localizar o pacote que fornece o comando `mogrify`.
2. Exiba informações sobre o pacote que fornece o comando `mogrify` e determine qual é a homepage (URL) desse pacote.
3. Instale o pacote contendo o comando `mogrify`.
4. Liste todos os arquivos de documentação contidos no pacote que fornece o comando `mogrify`.
5. Examine o `changelog` do pacote que fornece o comando `mogrify`.

6. Exclua o comando `mogrify` de seu sistema e verifique seu pacote no banco de dados RPM para ver se o comando está realmente faltando.
7. Reinstale o pacote que fornece o comando `mogrify` e certifique-se de que todo o pacote está intacto novamente.
8. Baixe o pacote que fornece o comando `mogrify` para seu diretório atual.
9. Exiba informações gerais sobre o pacote que você acabou de baixar, consultando o arquivo RPM do pacote no diretório atual.
10. Remova de seu sistema o pacote contendo o comando `mogrify`.

Gerenciando contas de usuário

NESTE CAPÍTULO

Trabalhando com contas de usuário

Trabalhando com contas de grupo

Configurando contas de usuário centralizadas

Adicionar e gerenciar usuários são tarefas comuns para os administradores de sistemas Linux. *Contas de usuário* mantêm fronteiras entre as pessoas que usam seus sistemas e entre os processos que são executados em seus sistemas. *Grupos* são uma forma de atribuir direitos ao seu sistema a vários usuários de uma vez.

Esta seção descreve não só como criar um novo usuário, mas também como criar configurações predefinidas e arquivos para configurar o ambiente do usuário. Usando ferramentas como os comandos `useradd` e `usermod`, você pode atribuir configurações, tais como a localização de um diretório, um shell padrão, um grupo padrão e valores específicos de ID de usuário e de ID de grupo.

Criando contas de usuário

Cada pessoa que usa o sistema Linux deve ter uma conta de usuário separada. Ter uma conta de usuário fornece a cada pessoa uma área para armazenar arquivos de maneira segura e um meio de adequar a interface com o usuário (GUI, caminho, variáveis de ambiente e assim por diante) para satisfazer a maneira como o usuário utiliza o computador.

Você pode adicionar contas de usuário à maioria dos sistemas Linux de diversas maneiras. Sistemas Fedora e Red Hat Enterprise Linux usam a janela User Manager (`system-config-users`) na área de trabalho. A partir do Red Hat Enterprise Linux, selecione System ⇒ Administration ⇒ Users & Groups e digite a senha de root quando solicitado. Quando a janela aparecer, selecione o botão Add User. A Figura 11.1 mostra um exemplo da janela do User Manager e o pop-up Add New User.

one contas de usuário e de grupo a partir da janela User Manager.

Agora, você está pronto para começar a adicionar uma nova conta de usuário ao seu sistema Linux. Eis os campos que você precisa preencher:

- **User Name** — O nome usado para fazer login como esse usuário. Quando você escolher um nome de usuário, não comece com um número (por exemplo, 26jsmith). Além disso, o melhor é usar no máximo oito letras, todas minúsculas, sem caracteres de controle ou espaços. O comando `useradd` permite usar até 32 caracteres, mas alguns aplicativos não podem lidar com nomes assim tão longos. Ferramentas como `ps` exibem IDs de usuário (UIDs) em vez de nomes, se os nomes forem muito longos. Ter usuários nomeados como `Jsmith` e `jsmith` pode causar confusão com programas (como o `sendmail`) que não fazem distinção entre maiúsculas e minúsculas.
- **Full Name** — O nome real do usuário, normalmente utilizado com letras maiúsculas e minúsculas, da mesma forma como o usuário escreveria na vida real. Tecnicamente, essa informação é armazenada no campo de comentário do arquivo `passwd`, mas, por convenção, a maioria dos sistemas Linux e UNIX espera que esse campo armazene o nome completo de cada usuário.
- **Password, Confirm Password** — A senha deve ter pelo menos oito caracteres e conter uma mistura de letras maiúsculas e minúsculas, números e sinais de pontuação. A senha não deve conter palavras reais, letras repetidas ou letras sequenciais do teclado. Como `root`, você receberá um aviso, mas ainda pode definir a senha da maneira que quiser. Um usuário normal, tentando mudar uma senha mais tarde, seria restringido a seguir o conjunto de normas.
- **Login Shell** — O shell `bash` é usado por padrão. Você pode, porém, selecionar a partir de uma lista de shells que estão instalados em seu sistema. O que quer que você defina será usado como o shell padrão quando o usuário fizer login ou abrir uma janela de terminal.
- **Create home directory** — Com essa caixa selecionada, um diretório é criado para o usuário se ele não existir. O diretório inicial, por padrão, será `/home/usuário`, em que *usuário* é substituído pelo nome de usuário. Você pode usar um local diferente, mas `/home` é o local padrão.
- **Create a private group for the user** — Com essa caixa de seleção marcada, o grupo principal atribuído a esse usuário será um novo grupo que tem mesmo nome que o usuário (neste caso, `mary`). Outros sistemas Linux atribuem todos os usuários a um grupo chamado `users` (user ID 100). Mas o RHEL e o Fedora usam um grupo privado dessa maneira.
- **Specify user ID manually** — Se nada for feito, o usuário recebe o próximo ID numérico. IDs de usuários começam em 500, embora, em alguns sistemas Linux, eles comecem em 1.000. Com a caixa marcada, você pode usar qualquer ID de usuário que quiser. Tenha em mente que se você especificar o ID de usuário manualmente, corre o risco de duplicar IDs de usuário. (O ID do usuário é o valor real que o sistema usa para atribuir aos arquivos e processos associados com o novo usuário.)

- Specify group ID manually — Como ocorre com o ID de usuário, marque essa caixa para definir um ID de grupo específico ou simplesmente usar o padrão, isto é, o próximo número de ID de grupo disponível. Da mesma maneira, especificar um ID de grupo corre o risco de duplicação.

Selecione OK para adicionar o usuário ao sistema. Uma entrada para a nova conta de usuário é adicionada ao arquivo `/etc/passwd` e a nova conta de grupo ao arquivo `/etc/group`. Descrevo estes mais adiante neste capítulo. Para modificar as informações do usuário mais tarde ou definir diferentes informações, clique duas vezes na conta de usuário que você deseja alterar. Uma janela User Properties aparece, como a mostrada na Figura 11.2.

fique contas de usuários existentes na janela User Properties.

A maioria das informações que você definiu anteriormente pode ser alterada a partir da guia User Data. Mas selecione a conta Info se você quiser que a conta expire em uma data específica ou para bloquear a senha de modo que, temporariamente, ninguém possa logar como o usuário. Na guia Password Info, você pode definir quando uma senha expira e, assim, forçar os usuários a alterarem suas senhas depois de um determinado período de tempo. A guia Groups permite que você selecione caixas de seleção para permitir que um usuário pertença a grupos diferentes.

Tudo descrito na janela User Manager também pode ser feito a partir da linha de comando. A próxima parte deste capítulo descreve como adicionar contas de usuário a partir da linha de comando com `useradd` ou alterá-las com o comando `usermod`.

Adicionando usuários com o `useradd`

Às vezes, um sistema Linux não vai ter um desktop disponível para usar a janela User Manager. Outras vezes, você pode achar que é mais conveniente adicionar vários usuários de uma vez com um script de shell. Em ambos os casos, há comandos para que você possa adicionar e modificar contas de usuário a partir da linha de comando.

O método mais simples para criar um novo usuário a partir do shell é o comando `useradd`. Depois de abrir uma janela de terminal com permissão de root, você simplesmente invoca `useradd` no prompt de comando, com os detalhes da nova conta como parâmetros.

O único parâmetro necessário é o nome de login do usuário, mas você provavelmente vai querer incluir alguma informação adicional. Cada item de informações de conta é precedido por um código de opção de uma única letra com um traço na frente. As opções disponíveis para `useradd` incluem as seguintes:

- `-c` "*comentário aqui*" — Forneça uma descrição da nova conta de usuário. Em geral, é o nome completo da pessoa. Substitua *comentário* pelo nome da conta do usuário (`-c Jake`). Use aspas para entrar várias palavras (por exemplo, `-c "Jake Jackson"`).
- `-d` *diretório_inicial* — Defina o diretório inicial para usar com a conta. O padrão é dar o mesmo nome usado para login e colocá-lo em `/home`. Substitua *diretório_inicial* pelo nome do diretório a usar (por exemplo, `-d /mnt/homes/jake`).
- `-D` — Em vez de criar uma nova conta, salve as informações fornecidas como as novas configurações padrão para todas as novas contas que forem criadas.
- `-e` *data_de_expiração* — Atribua a data de expiração da conta no formato AAAA-MM-DD. Substitua *data_de_expiração* pela data que você quiser usar. (Por exemplo, para expirar uma conta em 5 de maio de 2013, use `-e 2013-05-05`).

- `-f -1` — Defina o número de dias após a senha expirar até que a conta seja permanentemente desativada. O padrão, `-1`, desabilita a opção. Configurar isso como `0` desativa a conta imediatamente após a senha expirar. Substitua `-1` (isto é, menos um) pelo número desejado.
- `-g grupo` — Configure o grupo principal (que já deve existir no arquivo `/etc/group`) a que o novo usuário pertencerá. Substitua `grupo` pelo nome do grupo (por exemplo, `-g wheel`). Sem essa opção, um novo grupo é criado com o mesmo nome do usuário e esse novo grupo é usado como o principal do usuário.
- `-G lista_de_grupos` — Adicione o novo usuário à lista de grupos suplementares separada por vírgulas fornecida (por exemplo, `-G wheel, sales, tech, lunch`). (Se, depois, você usar `-G` com `usermod`, certifique-se de usar `-aG` e não apenas `-G`. Se você não fizer isso, grupos suplementares existentes serão removidos e os grupos que você fornece aqui são os únicos atribuídos.)
- `-k dir_esqueleto` — Defina o diretório esqueleto que contém arquivos de configuração inicial e scripts de login que devem ser copiados para o diretório inicial de um novo usuário. Esse parâmetro pode ser utilizado em conjunto com a opção `-m`. Substitua `dir_esqueleto` pelo nome do diretório a usar. (Sem essa opção, o diretório `/etc/skel` é utilizado.)
- `-m` — Crie Automaticamente o diretório inicial do usuário e copie os arquivos no diretório esqueleto (`/etc/skel`) para ele. (Essa é a ação padrão para o Fedora e o RHEL, por isso não é requerida. Não é o padrão do Ubuntu.)
- `-M` — Não crie o diretório inicial do novo usuário, mesmo que o comportamento padrão seja criá-lo.
- `-n` — Desative o comportamento padrão de criar um novo grupo que coincide com o nome e o ID de usuário do novo usuário. Essa opção está disponível em sistemas Fedora e RHEL. Outros sistemas Linux costumam atribuir um novo usuário ao grupo chamado `users`.
- `-o` — Use `-u uid` para criar uma conta de usuário que tem o mesmo UID que outro nome de usuário. (Isso efetivamente permite que você tenha dois nomes de usuário diferentes, com autoridade sobre o mesmo conjunto de arquivos e diretórios).
- `-p senha` — Digite uma senha para a conta que você está adicionando. Ela deve ser uma senha criptografada. Em vez de adicionar uma senha criptografada aqui, você pode simplesmente usar o comando `passwd usuário` depois de adicionar uma senha ao *usuário*. (Para gerar uma senha criptografada MD5, digite `openssl passwd`.)
- `-s shell` — Especifique o shell de comando a ser usado para essa conta. Substitua `shell` pelo shell de comando (por exemplo, `-s /bin/csh`).
- `-u id_do_usuario` — Especifique o número de ID do usuário para a conta (por exemplo, `-u 793`). Sem a opção `-u`, o comportamento padrão é atribuir automaticamente o próximo número disponível. Substitua `id_do_usuario` pelo número de ID.

Vamos criar uma conta para um novo usuário. O nome completo do usuário é Sara Green com um nome de login sara. Para começar, torne-se o usuário root e digite o seguinte comando:

```
# useradd -c "Sara Green" sara
```

Então, defina a senha inicial para **sara** usando o comando `passwd`. Você será solicitado a digitar a senha duas vezes:

```
# passwd sara
Changing password for user sara.
New password: *****
Retype new password: *****
```

ta

Os asteriscos neste exemplo representam a senha que você digita. Nada é realmente exibido quando você digita a senha. Também tenha em mente que executar `passwd` como usuário root permite que você adicione senhas curtas e senhas em branco que os usuários comuns não têm permissão para criar.

Ao criar a conta para Sara, o comando `useradd` executa diversas ações:

- Lê os arquivos `/etc/login.defs` e `/etc/default/useradd` para obter valores padrão para usar ao criar contas.
- Verifica parâmetros de linha de comando para saber quais valores padrão substituir.
- Cria uma entrada de novo usuário nos arquivos `/etc/passwd` e `/etc/shadow` com base em valores padrão e parâmetros de linha de comando.
- Cria qualquer entrada de novo grupo no arquivo `/etc/group`. (O Fedora cria um grupo usando o nome do novo usuário.)
- Cria um diretório inicial, com base no nome do usuário, no diretório `/home`.
- Copia quaisquer arquivos localizados dentro do diretório `/etc/skel` para o novo diretório. Normalmente, isso inclui scripts de login e de inicialização de aplicativo.

O exemplo anterior utiliza apenas algumas das opções de `useradd` disponíveis. A maioria das configurações das contas é atribuída utilizando os valores padrão. Você pode definir valores mais explicitamente, se quiser. Eis um exemplo que usa mais algumas opções para fazer isso:

```
# useradd -g users -G wheel,apache -s /bin/tcsh -c "Sara Green" sara
```

Nesse caso, `useradd` é instruído a tornar `users` o grupo primário a que `sara` pertence (`-g`), adicioná-la aos grupos `wheel` e `apache` e definir `tcsh` como seu shell de comando primário (`-s`). Um diretório inicial em `/home` sob o nome do usuário (`/home/sara`) é criado por padrão. Essa linha de comando

resulta na adição de uma linha semelhante à seguinte no arquivo `/etc/passwd`:

```
sara:x:502:507:Sara Green:/home/sara:/bin/tcsh
```

Cada linha no arquivo `/etc/passwd` representa um único registro de conta de usuário. Cada campo é separado do seguinte por um caractere de dois-pontos (:). A posição do campo na sequência determina o que ele é. Como você pode ver, o nome de login é o primeiro. O campo de senha contém um `x`, porque, nesse exemplo, o arquivo de senhas sombra é usado para armazenar dados de senha criptografados (em `/etc/shadow`).

O ID de usuário selecionado pelo `useradd` é 502. O ID de grupo primário é 507, o que corresponde a um grupo privado `sara` no arquivo `/etc/group`. O campo de comentário foi corretamente configurado como `Sara Green`, o diretório inicial foi automaticamente atribuído como `/home/sara` e o shell de comando foi atribuído como `/bin/tcsh`, exatamente como especificado com as opções `useradd`.

Deixando de fora muitas das opções (como eu fiz no primeiro exemplo de `useradd`), os padrões são atribuídos na maioria dos casos. Por exemplo, não usando `-g sales` ou `-G wheel, apache`, o nome grupo `mary` foi designado para o novo usuário. Alguns sistemas Linux (que não Fedora e o RHEL) definem `users` como o nome de grupo por padrão. Da mesma maneira, `-s /bin/tcsh` faz com que `/bin/bash` seja definido como o shell padrão.

O arquivo `/etc/group` contém informações sobre os diferentes grupos em seu sistema Linux e os usuários que pertencem a eles. Grupos são úteis para permitir que múltiplos usuários compartilhem o acesso aos mesmos arquivos enquanto nega acesso a outros. Eis a entrada `/etc/group` criada para `sara`:

```
sara:x:507:
```

Cada linha no arquivo de grupo contém o nome de um grupo, uma senha de grupo (normalmente preenchida com um `x`), o número de ID de grupo associado e uma lista de usuários nesse grupo. Por padrão, cada usuário é adicionado ao seu próprio grupo, começando com o próximo GID disponível, partindo de 500.

Configurando padrões de usuário

O comando `useradd` determina os valores padrão para novas contas lendo os arquivos `/etc/login.defs` e `/etc/default/useradd`. Você pode modificar esses padrões editando os arquivos manualmente com um editor de texto padrão. Embora `login.defs` seja diferente em diferentes sistemas Linux, o que segue é um exemplo que contém muitas das configurações que você pode encontrar em um arquivo `login.defs`:

```

PASS_MAX_DAYS      99999
PASS_MIN_DAYS      0
PASS_MIN_LEN       5
PASS_WARN_AGE      7
UID_MIN             500
UID_MAX            60000
GID_MIN            500
GID_MAX            60000
CREATE_HOME        yes

```

Todas as linhas ativas (sem um caractere de comentário no início da linha) contêm pares chave/valor. Por exemplo, a palavra-chave `PASS_MIN_LEN` é seguida de alguns espaços em branco e o valor 5. Isso diz a `useradd` que a senha do usuário deve ter pelo menos cinco caracteres. Outras linhas permitem que você personalize o intervalo válido de números de ID de usuário ou de grupos atribuídos automaticamente. (O Fedora começa em UID 500; sistemas anteriores começam com UID 100.) Uma seção de comentários, que explica o propósito dessa palavra-chave, precede cada palavra-chave (o que deixei de fora aqui para economizar espaço). Alterar um valor padrão é tão simples quanto editar o valor associado a uma palavra-chave e, então, salvar o arquivo antes de executar o comando `useradd`.

Se quiser ver as outras configurações padrão, você pode encontrá-las no arquivo `/etc/default/useradd`. Outra maneira de ver as configurações padrão é digitar o comando `useradd` com a opção `-D`, como segue:

```

# useradd -D
GROUP=100
HOME=/home
INACTIVE=-1
EXPIRE=
SHELL=/bin/bash
SKEL=/etc/skel

```

Você também pode usar a opção `-D` para mudar padrões. Quando executado com esse flag, `useradd` não realmente cria uma nova conta de usuário e, em vez disso, salva as opções adicionalmente fornecidas como os novos valores padrão em `/etc/default/useradd`. Nem todas as opções `useradd` podem ser usadas em conjunto com a opção `-D`. Você pode usar apenas as cinco opções listadas a seguir.

- `-b dir_inicial_padrao` — Defina o diretório padrão em que os diretórios iniciais dos usuários são criados. Substitua `dir_inicial_padrao` pelo nome do diretório a usar (por exemplo `-b /garage`). Normalmente, esse diretório é `/home`.
- `-e data_de_expiracao_padrao` — Defina a data de expiração padrão em que a conta de usuário será desativada. O valor `data_de_expiracao_padrao` deve ser substituído por uma data no formato AAAA-MM-DD (por exemplo `-e 2011-10-15`).
- `-f inativo_padrao` — Defina o número de dias em que a conta está desativada após a senha ter expirado. Substitua `inativo_padrao` por um número que representa o número de dias (por exemplo, `-f 7`).
- `-g grupo_padrao` — Defina o grupo padrão em que os novos usuários serão colocados. Em geral, `useradd` cria um novo grupo com o mesmo nome e número de ID que o usuário. Substitua

grupo_padrao pelo nome de grupo a utilizar (por exemplo `-g bears`).

- `-s shell_padrao` — Defina o shell padrão para novos usuários. Em geral, esse shell é `/bin/bash`. Substitua *shell_padrao* pelo caminho completo para o shell que você deseja como padrão para novos usuários (por exemplo `-s /bin/ash`).

Para definir qualquer um dos padrões, forneça a opção `-D` primeiro e depois adicione os padrões que você deseja configurar. Por exemplo, para configurar o local padrão do diretório inicial como `/home/everyone` e o shell padrão como `/bin/tcsh`, digite o seguinte:

```
# useradd -D -b /home/everyone -s /bin/tcsh
```

Além de configurar os padrões do usuário, um administrador pode criar arquivos padrão que são copiados para cada diretório inicial do usuário para uso. Esses arquivos podem incluir scripts de login e arquivos de configuração de shell (como `.bashrc`).

Outros comandos que são úteis para trabalhar com contas de usuários incluem `usermod` (para modificar as configurações de uma conta existente) e `userdel` (para excluir uma conta de usuário existente).

Modificando usuários com `usermod`

O comando `usermod` fornece um método simples e direto para mudar parâmetros de conta. Muitas das opções disponíveis com ele espelham aquelas encontradas em `useradd`. As opções que podem ser usadas com esse comando são as seguintes:

- `-c nome_de_usuario` — Altere a descrição associada com a conta de usuário. Substitua *nome de usuário* pelo nome da conta de usuário (`-c jake`). Use aspas para inserir várias palavras (por exemplo, `-c "Jake Jackson"`).
- `-d dir_inicial` — Mude o diretório inicial de uma conta. O padrão é dar o mesmo nome usado para login e colocá-lo em `home`. Substitua *diretorio_inicial* pelo nome do diretório que será usado (por exemplo, `-d /mnt/homes/jake`).
- `-e data_de_expiração` — Atribua uma nova data para a conta no formato AAAA-MM-DD. Substitua *data_de_expiração* pela data que você quiser usar. (Para 15 de outubro de 2013, uso `-e 2013-10-15`)
- `-f -1` — Altere o número de dias em que a conta será permanentemente desativada após a senha expirar. O padrão, `-1`, desabilita a opção. Configurar como `0` desativa a conta imediatamente depois que a senha tenha expirado. Substitua `-1` pelo número desejado.
- `-g grupo` — Altere o grupo primário (como listado no arquivo `/etc/group` em que o usuário vai estar. Substitua `group` pelo nome do grupo (por exemplo, `-g wheel`).
- `-G lista_de_grupos` — Defina os grupos secundários do usuário como uma lista fornecida de grupos separados por vírgula. Se o usuário já estiver em pelo menos um grupo além do privado

dele, você deve adicionar opção `-a` também (`-Ga`). Se não, o usuário pertencerá apenas ao novo conjunto de grupos e deixará de pertencer a quaisquer grupos anteriores.

- `-l nome_de_login` — Altere o nome de login da conta.
- `-L` — Bloqueie a conta colocando um ponto de exclamação no início da senha criptografada no arquivo `/etc/shadow`. Isso bloqueia a conta, enquanto ainda permite que você deixe a senha intacta (a opção `-U` a desbloqueia).
- `-m` — Disponível apenas quando `-d` é usado, isso faz com que o conteúdo do diretório inicial do usuário seja copiado para o novo diretório.
- `-o` — Use apenas com `-u uid` para remover a restrição de que UIDs devem ser únicos.
- `-s shell` — Especifique um shell de comando diferente a ser usado para essa conta. Substitua `shell` pelo comando de shell (por exemplo, `-s bash`).
- `-u id_de_usuario` — Altere o número de ID de usuário para a conta. Substitua `user_id` pelo número de ID (por exemplo, `-u 474`).
- `-U` — Desbloqueia a conta de usuário (removendo o ponto de exclamação no início da senha criptografada).

Os seguintes são exemplos do comando `usermod`:

```
# usermod -s /bin/csh chris
# usermod -Ga sales,marketing, chris
```

O primeiro exemplo muda o shell do usuário `chris` para o shell `csh`. No segundo exemplo, grupos suplementares são adicionados ao usuário `chris`. A opção `-a` (`-Ga`) garante que os grupos suplementares sejam adicionados a quaisquer grupos existentes do usuário `chris`. Se `-a` não for utilizado, grupos suplementares existentes para `chris` são apagados e a nova lista de grupos inclui os grupos suplementares unicamente atribuídos a esse usuário.

Excluindo usuários com `userdel`

Assim como `usermod` é usado para modificar as configurações de usuário e `useradd` é usado para criar usuários, `userdel` é usado para remover usuários. O comando a seguir remove o usuário `chris`:

```
# userdel -r chris
```

Aqui, o usuário `chris` é removido do arquivo `/etc/password`. A opção `-r` também remove o diretório inicial do usuário. Se você optar por não usar `-r`, como a seguir, o diretório inicial de `chris` não é removido:

```
# userdel chris
```

Tenha em mente que a simples remoção da conta do usuário não muda nada sobre os arquivos que ele deixa por todo o sistema (exceto aqueles que são excluídos quando você usa `-r`). No entanto, a posse dos arquivos deixados para trás aparece como pertencendo ao número de ID de usuário do proprietário anterior quando você executa `ls -l` nos arquivos.

Antes de excluir o usuário, você pode querer executar um comando `find` para localizar todos os arquivos que seriam deixados para trás por ele. Depois de excluir o usuário, você pode procurar por ID de usuário para encontrar arquivos deixados para trás. Eis aqui dois comandos `find` para fazer essas coisas:

```
# find / -user chris -ls
# find / -uid 504 -ls
```

Como os arquivos que não são atribuídos a nenhum nome de usuário são considerados um risco de segurança, é uma boa ideia localizá-los e atribuir-lhes uma conta de usuário real. Eis um exemplo de um comando `find` que localiza todos os arquivos no sistema de arquivos que não estão associados a nenhum usuário (os arquivos são listados por UID):

```
# find / -nouser -ls
```

Entendendo contas de grupo

Contas do grupo são úteis se você quiser compartilhar um conjunto de arquivos com múltiplos usuários. Você pode criar um grupo e alterar o conjunto de arquivos a serem associados com ele. O usuário `root` pode atribuir usuários a esse grupo para que eles possam ter acesso a arquivos com base na permissão do grupo. Considere o seguinte arquivo e diretório:

```
$ ls -ld /var/salesdocs /var/salesdocs/file.txt
drwxrwxr-x. 2 root sales 4096 Jan 14 09:32 /var/salesstuff/
-rw-rw-r--. 1 root sales 0 Jan 14 09:32 /var/salesstuff/file.txt
```

Examinando as permissões no diretório `/var/salesdocs` (`rw-rwxr-x`), você vê que o segundo conjunto de `rw` mostra que qualquer membro do grupo (`sales`) tem permissão para ler arquivos no diretório (`r` é *read*, ler), criar e excluir arquivos desse diretório (`w` é *write*, gravar) e mudar para esse diretório (`x` é *execute*, executar). O arquivo chamado `file.txt` pode ser lido e alterado pelos membros do grupo `sales` (com base no segundo `rw`).

Usando contas de grupo

Cada usuário é atribuído a um grupo primário. No Fedora e no RHEL, por padrão, esse grupo é um novo grupo com o mesmo nome do usuário. Assim, se o usuário se chamar `sara`, o grupo atribuído a ele também seria `sara`. O grupo primário é indicado pelo número no terceiro campo de cada entrada no arquivo `/etc/passwd`, por exemplo, o ID de grupo 507 aqui:

```
sara:x:502:507:Sara Green:/home/sara:/bin/tcsh
```

Essa entrada aponta para uma entrada no arquivo `/etc/group`:

```
sara:x:507:
```

Vamos voltar às contas de usuário e grupo `sara` para exemplos. Eis alguns fatos sobre o uso de grupos:

- Quando `sara` cria um arquivo ou diretório, por padrão, ele é designado para o grupo primário de `sara` (também chamado `sara`).
- O usuário `sara` pode pertencer a zero ou mais grupos suplementares. Se `sara` fosse um membro dos grupos nomeados `sales` e `marketing`, essas entradas poderiam parecer com o seguinte no arquivo `/etc/group`:

```
sales:x:302:joe,bill,sally,sara marketing:x:303:mike,terry,sara
```
- O usuário `sara` não pode adicionar-se a um grupo suplementar e nem adicionar outro usuário ao seu grupo `sara`. Só alguém com privilégios de administrador pode atribuir usuários aos grupos.
- Qualquer arquivo atribuído ao grupo `sales` ou `marketing` será acessível a `sara` com as permissões de grupo e outras permissões (o que fornecer o maior acesso). Se `sara` quiser criar um arquivo com os grupos `sales` ou `marketing` atribuídos a ela, pode usar o comando `newgrp`. Nesse exemplo, `sara` usa o comando `newgrp` para fazer `sales` tornar-se seu grupo primário temporariamente e cria um arquivo:

```
[sara]$ touch file1
[sara]$ newgrp sales
[sara]$ touch file2
[sara]$ ls -l file*
-rw-rw-r--. 1 sara sara 0 Jan 18 22:22 file1
-rw-rw-r--. 1 sara sales 0 Jan 18 22:23 file2 [sara]$ exit
```

Também é possível permitir que os usuários temporariamente se tornem um membro de um grupo com o comando `newgrp` sem realmente ser um membro desse grupo. Para fazer isso, alguém com permissão de `root` pode usar `gpasswd` para definir uma senha de grupo (como `gpasswd sales`). Depois disso, qualquer usuário pode digitar `newgrp sales` em um shell e usar temporariamente `sales` como seu grupo primário, simplesmente digitando a senha do grupo quando solicitado.

Criando contas de grupo

Como o usuário `root`, você pode criar novos grupos a partir da janela `User Manager` ou a partir da linha de comando com o comando `groupadd`. Além disso, como observado anteriormente, os grupos são criados automaticamente quando uma conta de usuário é criada.

Números de identificação de grupo de 0 a 499 são atribuídos a grupos administrativos especiais. Por exemplo, o grupo de root é associado com o GID 0. Grupos regulares começam em 500 para o Red Hat Enterprise Linux e o Fedora. Sobre os primeiros sistemas UNIX, GIDs passaram de 0 a 99. Outros sistemas Linux reservam os GIDs de 0 a 1000 para os grupos administrativos.

Eis alguns exemplos de como criar uma conta de grupo com o comando `groupadd`:

```
# groupadd kings
# groupadd -g 325 jokers
```

Nos exemplos apresentados, o grupo chamado `kings` é criado com o próximo ID de grupo disponível. Depois disso, o grupo `jokers` é criado usando o ID de grupo 325. Alguns administradores gostam de usar um número de grupo indefinido de menos de 500 para que o grupo que eles criam não interfira nas denominações de grupos acima de 500 (assim, os números de UID e GID podem crescer em paralelo).

Para alterar um grupo mais tarde, use o comando `groupmod`. Por exemplo:

```
# groupmod -g 330 jokers
# groupmod -n jacks jokers
```

No primeiro exemplo, o ID de grupo de `jokers` é alterado para 330. No segundo, o nome `jokers` é alterado para `jacks`. Se quiser definir qualquer um dos grupos como grupos complementares para um usuário, você pode usar o comando `usermod` (como descrito anteriormente neste capítulo).

Gerenciando usuários na empresa

O método básico do Linux para lidar com as contas de usuário e de grupo não mudou desde que os primeiros sistemas UNIX foram desenvolvidos décadas atrás. Mas à medida que os sistemas Linux passaram a ser usados de maneiras mais complexas, funcionalidades para gerenciar usuários, grupos e as permissões associadas a eles foram adicionadas ao modelo básico de usuário/grupo para que pudesse ser:

- Mais flexível — No modelo básico, apenas um usuário e um grupo podem ser atribuídos a cada arquivo. Além disso, usuários regulares não têm capacidade para atribuir permissões específicas a diferentes usuários ou grupos e têm muito pouca flexibilidade na configuração de arquivos/diretórios de colaboração. Melhorias nesse modelo permitem que usuários regulares criem diretórios especiais de colaboração (usando recursos como *sticky bit* e *set GID bit*). Usando listas de controle de acesso (ACLs), qualquer usuário também pode atribuir permissões específicas sobre arquivos e diretórios a todos os usuários e grupos que ele quiser.
- Mais centralizado — Quando você tem apenas um computador, armazenar informações de usuário para todos os usuários no arquivo `/etc/password` provavelmente não é uma dificuldade. Mas se você precisar autenticar o mesmo conjunto de usuários em milhares de sistemas Linux, centralizar essas informações pode poupar-lhe muito tempo e dor de cabeça. O Red Hat Enterprise Linux inclui recursos que permitem autenticar os usuários de servidores LDAP ou

servidores Microsoft Active Directories.

As próximas seções descrevem como usar recursos como listas de controle de acesso (ACLs) e diretórios compartilhados (*sticky bit* e *set GID bit*) a fim de fornecer maneiras poderosas de compartilhar seletivamente arquivos e diretórios. Então, eu descrevo como gerenciar contas de usuário a partir de servidores de autenticação centralizada utilizando a janela Authentication Configuration.

Definindo permissões com listas de controle de acesso

O recurso de lista de controle de acesso (ACL) foi criado para que os usuários regulares pudessem compartilhar seus arquivos e diretórios seletivamente com outros usuários e grupos. Com as ACLs, um usuário pode permitir que outros leiam, gravem e executem arquivos e diretórios sem deixar os elementos do sistema de arquivos expostos ou exigir que o usuário root altere o usuário ou grupo que lhes foi atribuído.

Eis algumas coisas a saber sobre ACLs:

- Para ACLs serem utilizadas, elas devem ser ativadas em um sistema de arquivos quando o sistema de arquivos é montado.
- No Fedora e no Red Hat Enterprise Linux, as ACLs são ativadas automaticamente em qualquer sistema de arquivos criado quando o sistema é instalado.
- Se criar um sistema de arquivos após a instalação (por exemplo, quando você adiciona um disco rígido), você precisa ter certeza de que a opção de montagem `acl` é usada quando o sistema de arquivos é montado (mais sobre isso mais adiante).
- Para adicionar ACLs a um arquivo, você pode usar o comando `setfacl`; para ver ACLs definidas em um arquivo, você pode usar o comando `getfacl`.
- Para definir ACLs em qualquer arquivo ou diretório, você deve ser o real proprietário (usuário) atribuído a ele. Em outras palavras, receber permissões de usuário ou grupo com `setfacl` não lhe permite mudar as ACLs definidas para esses arquivos.
- Como vários usuários e grupos podem ser atribuídos a um arquivo/diretório, a permissão real que um usuário tem é baseada em uma união de todas as designações de usuário/grupo a que eles pertencem. Por exemplo, se um arquivo tivesse permissão somente de leitura (`r--`) para o grupo sales e de leitura/gravação/execução (`rwX`) para o grupo market, e mary pertencesse a ambos, ela teria permissão `rwX`.

ta

CLs não estiverem habilitadas no sistema de arquivos que você está tentando usar com `setfacl`, consulte o "Habilitando ACLs", mais adiante, neste capítulo, para obter informações sobre como montar um sistema de arquivos com ACLs ativadas.

Configurando ACLs com setfacl

Usando o comando `setfacl`, você pode modificar as permissões (-m) ou remover permissões ACL (-x). Eis um exemplo da sintaxe do comando `setfacl`:

```
setfacl -m u:username:rw filename
```

No exemplo mostrado, a opção `modify` (-m) é seguida pela letra `u`, indicando que você está configurando permissões ACL para um usuário. Depois de um sinal de dois-pontos (:), você indica o nome do usuário, seguido por outro sinal de dois-pontos e as permissões que deseja atribuir. Como ocorre com o comando `chmod`, você pode atribuir permissões de leitura (`r`), gravação (`w`) e/ou execução (`x`) para o usuário ou grupo (no exemplo, é dada a permissão `rw` completa). O último argumento é substituído pelo nome do arquivo real que você está modificando.

A seguir, alguns exemplos do usuário `mary` usando o comando `setfacl` para adicionar a permissão para outros usuários e grupos em um arquivo:

```
[mary]$ touch /tmp/memo.txt
[mary]$ ls -l /tmp/memo.txt
-rw-rw-r--. 1 mary mary 0 Jan 21 09:27 /tmp/memo.txt
[mary]$ setfacl -m u:bill:rw /tmp/memo.txt
[mary]$ setfacl -m g:sales:rw /tmp/memo.txt
```

No exemplo anterior, `mary` criou um arquivo chamado `/tmp/memo.txt`. Usando o comando `setfacl`, ela modificou (-m) as permissões para o usuário chamado `bill`, de modo que agora ele tem permissões de leitura/gravação (`rw`) sobre esse arquivo. Em seguida, ela modificou as permissões para o grupo `sales` de modo que qualquer pessoa pertencente a esse grupo também teria permissões de leitura/gravação. Olhe para a saída de `ls -l` e `getfacl` nesse arquivo agora:

```
[mary]$ ls -l /tmp/memo.txt
-rw-rw-r--+ 1 mary mary 0 Jan 21 09:27 /tmp/memo.txt
[mary]$ getfacl /tmp/memo.txt
# file: tmp/memo.txt
# owner: mary
# group: mary
user::rw
user:bill:rw
group::rw
group:sales:rw
mask::rw
other::r--
```

A partir da saída de `ls -l`, observe o sinal de mais (+) na saída `rw-rw-r--+`. O sinal de mais indica que ACLs estão configuradas no arquivo, e você sabe executar o comando `getfacl` para ver como as ACLs estão configuradas. O resultado mostra `mary` como proprietária e como grupo (o mesmo que você

vê com `ls -l`), as permissões de usuários regulares (`rw-`) e as permissões sobre as ACLs para o usuário `bill` (`rw-`). O mesmo é verdadeiro para as permissões de grupo e as permissões do grupo `sales`. Outras permissões são `r--`.

A linha de máscara (perto do final do exemplo de `getfacl` anterior) requer uma discussão especial. Logo que você configura as ACLs sobre um arquivo, a permissão regular de um grupo sobre ele configura uma máscara de permissão máxima que o usuário ou grupo de ACL pode ter sobre um arquivo. Assim, mesmo que você forneça a um indivíduo mais permissões ACL do que as permissões do grupo permitem, as permissões efetivas do indivíduo não poderão exceder as permissões do grupo. Por exemplo:

]

```
[mary]$ chmod 644 /tmp/memo.txt
[mary]$ getfacl /tmp/memo.txt
# file: tmp/memo.txt
# owner: mary
# group: mary
user::rw-
user:bill:rw- #effective:r--
group::rw- #effective:r--
group:sales:rw- #effective:r--
mask::r--
other::r--
```

Note, no exemplo anterior, que apesar de o usuário `bill` e o grupo `sales` terem permissões `rw-`, suas permissões efetivas são `r--`. Então, `bill` ou qualquer um em `sales` não seriam capazes de alterar o arquivo, a menos que `mary` abra as permissões novamente (por exemplo, digitando `chmod 644 /tmp/memo.txt`).

Definindo ACLs padrão

Definir ACLs padrão em um diretório permite que suas ACLs sejam herdadas. Isso significa que, quando os novos arquivos e diretórios são criados no diretório, eles recebem as mesmas ACLs. Para definir uma permissão ACL de usuário ou grupo como o padrão, você adiciona um `d:` à designação do usuário ou do grupo. Considere o seguinte exemplo:

```
[mary]$ mkdir /tmp/mary
[mary]$ setfacl -m d:g:market:rwX /tmp/mary/
[mary]$ getfacl /tmp/mary/
# file: tmp/mary/
# owner: mary
# group: mary
user::rwx
group::rwx
other::r-x
```



```
default:user::rwx
default:group::rwx
default:group:sales:rwx
default:group:market:rwx
default:mask::rwx
default:other::r-x
```

Para se certificar de que a ACL padrão funcionou, crie um subdiretório. Então, execute `getfacl` novamente. Você vai ver que linhas padrão são adicionadas a `user`, `group`, `mask` e `other`, que são herdadas das ACLs do diretório.

```
[mary]$ mkdir /tmp/mary/test
[mary]$ getfacl /tmp/mary/test
# file: tmp/mary/test
# owner: mary
# group: mary
user::rwx
group::rwx
group:sales:rwx
group:market:rwx
mask::rwx
other::r-x
default:user::rwx
default:group::rwx
default:group:sales:rwx
default:group:market:rwx
default:mask::rwx
default:other::r-x
```

Observe que quando você cria um arquivo no diretório, as permissões herdadas são diferentes. Como um arquivo regular é criado sem permissão de execução, a permissão efetiva é reduzida a `rw-`:

```
[mary@cnegus~]$ touch /tmp/mary/file.txt
[mary@cnegus~]$ getfacl /tmp/mary/file.txt
# file: tmp/mary/file.txt
# owner: mary
# group: mary
user::rw-
group::
rwx #effective:rw
group:sales:rwx #effective:rw
group:market:rwx #effective:rw
mask::rw-
other::r--
```

Habilitando ACLs

Sistemas de arquivos Linux básicos têm apenas um usuário e um grupo designado para cada arquivo e diretório e não incluem suporte a ACLs por padrão. Os tipos de sistema de arquivos Linux ext (ext2, ext3 e ext4) podem adicionar suporte a ACLs por meio de uma opção de montagem. Para adicionar suporte a ACLs, você deve adicionar a opção de montagem `acl` ao montá-la. Você pode fazer isso de várias maneiras:

- Adicione a opção `acl` ao quinto campo da linha no arquivo `/etc/fstab` que monta automaticamente o sistema de arquivos quando o sistema é inicializado.
- Implante a linha `acl` no campo Default mount options no superbloco do sistema de arquivos, de modo que a opção `acl` seja usada se o sistema de arquivos for montado automaticamente ou manualmente.
- Adicione a opção `acl` à linha de comando de montagem quando você montar o sistema de arquivos manualmente com o comando `mount`.

Tenha em mente que em sistemas Fedora e Red Hat Enterprise Linux, você só precisa acrescentar a opção de montagem `acl` para os sistemas de arquivos que criar depois que o Linux foi instalado. O instalador anaconda automaticamente adiciona suporte ACL a cada sistema de arquivo que ele cria durante a instalação. Para verificar se a opção `acl` foi adicionada a um sistema de arquivos, determine o nome do dispositivo associado com o sistema de arquivos e execute o comando `tune2fs -l` para ver as opções de montagem implantadas. Por exemplo:

```
# mount | grep home
/dev/mapper/mybox-home on /home type ext4 (rw)
# tune2fs -l /dev/mapper/mybox-home | grep "mount options"
Default mount options: user_xattr acl
```

Primeiro, digitei o comando `mount` para ver uma lista de todos os sistemas de arquivos que estão montados atualmente, limitando a saída com a palavra `home` (porque eu estava procurando sistemas de arquivos montados em `/home`). Depois que vi o nome de dispositivo do sistema de arquivos, eu o usei como uma opção para `tune2fs -l` a fim de encontrar a linha de opções de montagem padrão. Lá, pude ver que as opções de montagem `user_xattr` (para atributos estendidos como SELinux) e `acl` estavam ambas implantadas no superbloco do sistema de arquivos, de modo que elas seriam usadas quando o sistema de arquivos fosse montado.

Se o campo Default mount options estiver em branco (como quando você acabou de criar um novo sistema de arquivos), você pode adicionar a opção de montagem `acl` usando o comando `tune2fs -o`. Por exemplo, eu criei um sistema de arquivos em um drive USB removível que foi designado como o dispositivo `/dev/sdc1`. Para implantar a opção de montagem `acl` e verificar se ela está lá, executei os seguintes comandos:

```
# tune2fs -o acl /dev/sdc1
# tune2fs -l /dev/sdc1 | grep "mount options"
Default mount options: acl
```

Você pode testar se isso funcionou remontando o sistema de arquivos e tentando usar o comando `setfacl` em um arquivo nesse sistema de arquivos.

Uma segunda maneira de adicionar suporte `acl` a um sistema de arquivos é adicionar a opção `acl` à linha no arquivo `/etc/fstab` que monta automaticamente o sistema de arquivos durante a inicialização. O seguinte é um exemplo de como seria uma linha que monta o sistema de arquivos `ext4` localizado no dispositivo `/dev/sdc1` no diretório `/var/stuff`:

```
/dev/sdc1 /var/stuff ext4 acl 1 2
```

Em vez da entrada padrão no quarto campo, eu adicionei `acl`. Se já houver opções definidas nesse campo, adicione uma vírgula após a última opção e acrescente `acl`. Da próxima vez que o sistema de arquivos for montado, as ACLs estarão habilitadas. Se o sistema de arquivos já estivesse montado, eu poderia digitar o comando `mount` a seguir como `root` para remontar o sistema de arquivos, usando `acl` ou quaisquer outros valores adicionados ao arquivo `/etc/fstab`:

```
# mount -o remount /dev/sdc1
```

Uma terceira maneira de adicionar suporte ACL a um sistema de arquivos é montar o sistema de arquivos manualmente e solicitar especificamente a opção de montagem `acl`. Então, se não houver nenhuma entrada para o sistema de arquivos no arquivo `/etc/fstab`, depois de criar o ponto de montagem (`/var/stuff`), digite o seguinte comando para montar o sistema de arquivos e incluir suporte ACL:

```
# mount -o acl /dev/sdc1 /var/stuff
```

Tenha em mente que o comando `mount` monta o sistema de arquivos apenas temporariamente. Quando o sistema for reinicializado, o sistema de arquivos não será montado novamente, a menos que você adicione uma entrada no arquivo `/etc/fstab`.

Adicionando diretórios para os usuários colaborarem

Há um conjunto especial de três bits de permissão que normalmente são ignorados quando você usa o comando `chmod` para alterar as permissões no sistema de arquivos. Esses bits podem definir permissões especiais sobre comandos e diretórios. O foco desta seção é definir os bits que ajudam a criar diretórios para uso colaborativo.

Como ocorre com os bits de leitura, gravação e execução para `user`, `group` e `other`, esses bits de permissão de arquivos especiais podem ser definidos com o comando `chmod`. Por exemplo, se você executar `chmod 775 /mnt/xyz`, a permissão implícita é, na verdade, `0775`. Para alterar permissões, você pode substituir o número 0 por qualquer combinação desses três bits (4, 2 e 1) ou você pode usar valores em vez de letras. (Consulte o Capítulo 4, “Movendo-se pelo sistema de arquivos”, se você precisar lembrar como as permissões funcionam.) As letras e os números são apresentados na Tabela 11.1.

Tabela 11.1 Comandos para Criar e Usar Arquivos

me	Valor numérico	Valor alfabético
set user ID	4	u+s
set group ID	2	g+s
sticky bit	1	o+t

Os bits nos quais você está interessado para a criação de diretórios de colaboração são os bits set group ID (2) e sticky bit (1). Se você estiver interessado em outros usos de bits de set user ID e set group ID, consulte a barra lateral “Usando os comandos de bit Set UID e Set GID”.

Criando diretórios de colaboração em grupo (bit set GID)

Ao criar um diretório set GID, todos os arquivos criados nele são atribuídos ao grupo designado para o próprio diretório. A ideia é ter um diretório onde todos os membros de um grupo podem compartilhar arquivos, mas ainda protegê-los de outros usuários. Eis um conjunto de passos para criar um diretório de colaboração para todos os usuários no grupo chamado sales, que eu criei:

- 1. Crie um grupo para uso colaborativo:**

```
# groupadd -g 301 sales
```
- 2. Adicione alguns usuários ao grupo que você quer que sejam capazes de compartilhar arquivos (eu usei mary):**

```
# usermod -aG sales mary
```
- 3. Crie o diretório colaborativo:**

```
# mkdir /mnt/salestools
```

Usando os comandos de bit set UID e set GID

Os bits set UID e set GID são usados em arquivos executáveis especiais que permitem configurar comandos para serem executados de forma diferente da maioria. Normalmente, quando um usuário executa um comando, ele é executado com as permissões do usuário. Em outras palavras, se eu executar o comando `vi como chris`, essa instância do comando `vi` teria as mesmas permissões para ler e gravar arquivos que o usuário `chris` tivesse.

Comandos com os bits set UID ou set GID definidos são diferentes. É o proprietário e o grupo designado para o comando, respectivamente, que determinam as permissões que o ele tem para acessar recursos no computador. Assim, um comando set UID pertencente ao usuário `root` seria executado com as permissões de `root`, um comando set GID pertencente ao grupo `apache` teria as permissões desse grupo.

Exemplos de aplicativos que ativam os bits set UID são os comandos `sue newgrp`. Em

ambos os casos, os comandos devem ser capazes de agir como o usuário root para fazer seus trabalhos. Mas, para realmente obter permissões de root, o usuário deve fornecer uma senha. Você pode dizer que `su` é um comando `set UID bit` por causa do `s`, em que o primeiro bit de execução (`x`) normalmente aparece assim:

```
$ ls /bin/su
-rwsr-xr-x. 1 root root 30092 Jan 30 07:11 su
```

4. Atribua o grupo sales ao diretório:

```
# chgrp sales /mnt/salestools
```

5. Altere a permissão sobre o diretório para 2775. Isso ativa o bit set group ID (2), `rw`x completo para `user(7)`, `rw`x para `group(7)`, e `r-x(5)` para `other`:

```
# chmod 2775 /mnt/salestools
```

6. Torne-se mary (execute `su - mary`). Como mary, crie um arquivo no diretório compartilhado e examine as permissões. Ao listar as permissões, você pode ver que o diretório é um diretório set GID, porque um `s` minúsculo aparece onde a permissão de execução do grupo deve aparecer (`rwxrwsr-x`):

```
# su - mary [mary]$ touch /mnt/salestools/test.txt
[mary]$ ls -ld /mnt/salestools/ /mnt/salestools/test.txt
drwxrwsr-x. 2 root sales 4096 Jan 22 14:32 /mnt/salestools/
-rw-rw-r--. 1 mary sales 0 Jan 22 14:32
/mnt/salestools/test.txt
```

Normalmente, um arquivo criado por mary teria o grupo mary atribuído a ele. Mas como `test.txt` foi criado em um diretório set group ID bit, o arquivo é atribuído ao grupo sales. Agora, qualquer um que pertence ao grupo sales pode ler ou gravar o arquivo, com base nas permissões do grupo.

Criando diretórios de exclusão restrita (sticky bit)

Um *diretório de exclusão restrita* é criado ativando o *sticky bit* de um diretório. O que torna um diretório de exclusão restrita diferente dos outros diretórios? Normalmente, se a permissão de gravação é aberta para um usuário em um arquivo ou diretório, o usuário é capaz de excluir o arquivo ou diretório. Mas em um diretório de exclusão restrita, a não ser que você seja o usuário root ou o proprietário do diretório, você nunca pode excluir os arquivos de outro usuário.

Normalmente, um diretório de exclusão restrita é usado como um lugar onde muitos usuários diferentes podem criar arquivos. Por exemplo, o diretório `/tmp` é de exclusão restrita:

```
$ ls -ld /tmp
drwxrwxrwt. 116 root root 36864 Jan 22 14:18 /tmp
```

Você pode ver que as permissões estão abertas, mas em vez de um `x` para o bit de execução de `other`, o `t` indica que o *sticky bit* está configurado. O exemplo a seguir cria um diretório de

exclusão restrita com um arquivo que ninguém tem permissão para gravar:

```
[mary]$ mkdir /tmp/mystuff
[mary]$ chmod 1777 /tmp/mystuff
[mary]$ cp /etc/services /tmp/mystuff/
[mary]$ chmod 666 /tmp/mystuff/services
[mary]$ ls -ld /tmp/mystuff /tmp/mystuff/services
drwxrwxrwt. 2 mary mary 4096 Jan 22 15:28 /tmp/mystuff/
-rw-rw-rw-. 1 mary mary 640999 Jan 22 15:28
/tmp/mystuff/services
```

Com as permissões definidas para 1777 no diretório `/tmp/mystuff`, você pode ver que todas as permissões estão abertas, mas um `t` aparece em vez do último bit de execução. Com o arquivo `/tmp/mystuff/services` aberto para a gravação, qualquer usuário pode abri-lo e alterar seu conteúdo. Mas como o arquivo está em um diretório sticky bit, apenas root e mary podem excluir o arquivo.

Centralizando contas de usuário

Embora a forma padrão de autenticação de usuários no Linux seja verificar as informações do usuário com o arquivo `/etc/passwd` e senhas no arquivo `/etc/shadow`, há outras maneiras de fazer a autenticação. Na maioria das grandes empresas, as informações de conta de usuário são armazenadas em um servidor de autenticação centralizada, de modo que cada vez que você instala um novo sistema Linux, em vez de adicionar contas de usuário a esse sistema, você faz o Linux consultar o servidor de autenticação quando alguém tenta fazer login.

Tal como acontece com a autenticação `passwd/shadow` local, configurar a autenticação centralizada exige fornecer dois tipos de informações: informações da conta (nome de usuário, IDs de usuário/grupo, diretório inicial, shell padrão etc.) e método de autenticação (diferentes tipos de senhas criptografadas, cartões inteligentes, leitura da retina etc.). O Linux oferece maneiras de configurar os tipos de informação.

Como eu espero que você um dia use suas habilidades em Linux para trabalhar em uma grande instalação do Linux, quero apresentar o conceito de autenticação centralizada. Por enquanto, vou apenas discutir como se conectar a servidores de autenticação existentes (em vez de criar seus próprios servidores) e fazer os usuários do sistema Linux que você configurou autenticarem-se nesses tipos de servidor.

Usando a janela Authentication Configuration

No Fedora e no Red Hat Enterprise Linux, há uma janela gráfica para configurar a autenticação centralizada chamada Authentication Configuration. Por meio dessa janela, você pode configurar onde seu sistema consegue informações de conta e que tipo de método de autenticação é usado para verificar usuários. Os tipos de banco de dados centralizado suportados incluem:

- LDAP— O Lightweight Directory Access Protocol (LDAP) é um protocolo popular para fornecer serviços de diretório (como agendas telefônicas, endereços e contas de usuário). É um padrão aberto que é configurado em muitos tipos de ambientes de computação.
- NIS— O Network Information Service (NIS) foi originalmente criado pela Sun Microsystems para propagar informações de sistema, como contas de usuário, configuração de host e outras, entre muitos sistemas UNIX. Como o NIS passa informações em texto claro, a maioria das empresas já utiliza os protocolos mais seguros LDAP ou Winbind para autenticação centralizada.
- Winbind— Selecionar Winbind a partir da janela Authentication Configuration permite autenticar seus usuários contra um servidor Microsoft Active Directory (AD). Muitas grandes empresas estendem sua configuração de autenticação de desktop para também fazer a configuração de servidor usando um servidor AD.

Para essa introdução à configuração de servidores de autenticação centralizada, você vai configurar um sistema Linux para autenticação contra um servidor LDAP, usando a janela Authentication Configuration no Red Hat Enterprise Linux.

Para começar, você precisa coletar informações sobre o serviço LDAP no seu local. Isso inclui informações sobre o banco de dados de contas e o método de autenticação:

- LDAP Search Base DN— Esse é o nome distinto do banco de dados LDAP utilizado para identificar a localização dos registros de conta de usuário. Muitas vezes, o nome é construído a partir do nome de domínio DNS da empresa. Por exemplo, `dc=example,dc=com`.
- LDAP server— O nome de host do servidor LDAP. Por exemplo, `ldap://ldap.example.com`.
- Use TLS to encrypt connections— Com essa opção selecionada, você também deve indicar a localização de um certificado de autoridade certificadora (AC), que será transferido para o sistema local a fim de ser usado para validar e criptografar as comunicações com o servidor LDAP. Certificados de Transport Layer Security (TLS) para uma organização são obtidos de autoridades de certificação, como a Verisign. Ou você pode criar certificados autoassinados.
- Authentication method— Em vez de usar senhas MD5 regulares, escolha uma senha LDAP ou Kerberos como método de autenticação com LDAP. Para Kerberos, você também deve fornecer informações sobre o servidor Kerberos, o que inclui o domínio Kerberos, KDCs e Admin Servers. Todas essas informações devem ser fornecidas pelos administradores que gerenciam os servidores Kerberos de sua empresa.

Para iniciar a janela Authentication Configuration a partir de um desktop Red Hat Enterprise Linux 6, selecione System ⇒ Administration ⇒ Authentication. Para um sistema padrão que só faz a autenticação (passwd/shadow) local, a janela aparece.

Para adicionar a autenticação LDAP, selecione a caixa User Account Database e selecione LDAP. Então, preencha as informações descritas na lista itemizada anterior.

Se o novo método de autenticação estiver configurado corretamente, você deve ser capaz de ir para um shell e validar isso. Se você souber de uma conta de usuário disponível no servidor LDAP, use o seguinte comando `getent`, para verificar se a conta está disponível:

```
# getent passwd jsmith
jsmith:x:13599:13600:John Smith:/home/jsmith:/bin/bash
```

Se você vir as informações da conta, então sabe que seu sistema foi capaz de recuperá-las a partir do servidor LDAP. A próxima coisa a verificar é se o método de autenticação está funcionando também. Para isso, você pode tentar se conectar como o usuário do console ou usando o comando `ssh`. Por exemplo:

```
$ ssh jsmith@localhost
```

Quando solicitado, digite o nome de usuário e a senha. Se funcionar, você sabe que tanto as informações de conta como as de autenticação que você inseriu para seu servidor LDAP estavam corretas.

Com a autenticação centralizada, considere centralizar também o diretório inicial dos usuários. Usando o automontador do Linux (serviço `autofs`), você pode configurar diretórios que são automaticamente montados quando cada usuário efetua login, independentemente da máquina em que eles estão se logando. (Veja o Capítulo 20, “Configurando um servidor de arquivos NFS”, para obter informações sobre como configurar um servidor NFS e configurar clientes para automontagem a partir desse servidor.)

Resumo

Ter contas de usuário separadas é o principal método de definição de fronteiras seguras entre as pessoas que usam seu sistema Linux. Usuários regulares normalmente podem controlar os arquivos e diretórios dentro de seus próprios diretórios iniciais, mas muito pouco fora desses diretórios.

Neste capítulo, você aprendeu a adicionar contas de usuário e de grupo, como modificá-las e até como estender contas de usuário e de grupo para além das fronteiras do arquivo `/etc/passwd`. Você também aprendeu que a autenticação pode ser feita acessando servidores LDAP centralizados.

O próximo capítulo introduz outro tema básico necessário para administradores de sistemas Linux: como gerenciar discos. Nesse capítulo, você aprenderá a particionar discos, adicionar sistemas de arquivos e montá-los, de modo que o conteúdo das partições de disco seja acessível para aqueles que utilizam o sistema.

Exercícios

Use esses exercícios para testar seus conhecimentos em adicionar e gerenciar contas de usuário e

de grupo no Linux. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostradas no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. **Adicione uma conta de usuário local a seu sistema Linux que tenha o nome de usuário `jbaxter` e o nome completo John Baxter e use `/bin/sh` como seu shell padrão. Deixe o UID ser atribuídos por padrão. Configure a senha para `jbaxter` como: `My1N1te0ut!`**
2. **Crie uma conta de grupo chamada `testing` que use o ID de grupo 315.**
3. **Adicione `jbaxter` ao grupo `testing` e o grupo `bin`.**
4. **Abra um shell como `jbaxter` (ou uma nova sessão ou usando um shell atual) e, temporariamente, faça o grupo `testing` ser seu grupo padrão, de modo que quando você digitar `touch /home/jbaxter/file.txt`, o grupo `testing` seja atribuído como o grupo do arquivo.**
5. **Observe qual ID de usuário foi atribuído a `jbaxter` e então exclua a conta do usuário sem excluir o diretório inicial atribuído a `jbaxter`.**
6. **Encontre todos os arquivos no diretório `/home` (e quaisquer subdiretórios) que estão atribuídos ao ID de usuário que, recentemente, pertencia ao usuário chamado `jbaxter`.**
7. **Copie o arquivo `/etc/services` para o diretório esqueleto padrão, de modo que ele apareça no diretório inicial de qualquer novo usuário. Então, adicione ao sistema um novo usuário chamado `MJones`, com o nome completo Mary Jones e um diretório inicial `/home/maryjones`.**
8. **Encontre todos os arquivos no diretório `/home` que pertencem a `MJones`. Há algum arquivo pertencente a `MJones` que você não esperava ver?**
9. **Faça login como `MJones` e crie um arquivo chamado `/tmp/maryfile.txt`. Usando ACLs, atribua ao usuário `bin` permissões de leitura/gravação sobre o arquivo. Então, atribua ao grupo `lp` permissões de leitura/gravação sobre o arquivo.**
10. **Ainda como `MJones`, crie um diretório chamado `/tmp/mydir`. Usando ACLs, atribua permissões padrão a esse diretório para que o usuário `adm` tenha permissões de leitura/gravação/execução sobre ele e todos os arquivos ou diretórios criados nele. Crie o diretório `/tmp/mydir/testing/` e o arquivo `/tmp/mydir/newfile.txt` e certifique-se de que o usuário `adm` também recebeu permissões completas de leitura/gravação/execução. (Note que, apesar da atribuição da permissão `rxw` ao usuário `adm`, a permissão efetiva sobre `newfile.txt` é apenas `rw`. O que você poderia fazer para garantir que `adm` recebesse também a permissão de execução?)**

Gerenciando discos e sistemas de arquivos

NESTE CAPÍTULO

Criando partições de disco

Criando volumes lógicos com LVM

Adicionando sistemas de arquivos

Montando sistemas de arquivos

Desmontando sistemas de arquivos

Seu sistema operacional, aplicativos e todos os dados precisam ser mantidos em algum tipo de armazenamento permanente, de modo que quando você desligar o computador, tudo ainda esteja lá quando o computador for ligado novamente. Tradicionalmente, esse armazenamento tem sido fornecido por um disco rígido de seu computador. Para organizar as informações sobre esse disco, ele é normalmente dividido em partições, com a maioria das partições recebendo uma estrutura chamada *sistema de arquivos*.

Este capítulo descreve como trabalhar com discos rígidos. Tarefas de disco rígido incluem particionar, acrescentar sistemas de arquivos e gerenciar sistemas de arquivos de várias maneiras.

Depois de cobrir partições básicas, descrevo como o gerenciamento de volume lógico (LVM) pode ser usado para tornar mais fácil aumentar, reduzir e administrar sistemas de arquivos de forma mais eficiente.

Entendendo armazenamento em disco

As noções básicas de como funciona o armazenamento de dados são as mesmas na maioria dos sistemas operacionais modernos. Quando você instala o sistema operacional, o disco é dividido em uma ou mais partições. Cada partição é formatada por um sistema de arquivos. No caso do Linux, algumas das partições podem ser especialmente formatadas para elementos, tais como área de troca (*swap*), ou volumes físicos LVM.

Enquanto os discos são usados para o armazenamento permanente, a memória de acesso aleatório (RAM) e a memória de troca (*swap*) são usadas para armazenamento temporário. Por exemplo, quando você executa um comando, ele é copiado do disco rígido para a memória RAM para que o processador do

computador (CPU) possa acessá-lo mais rapidamente.

Seu processador pode acessar dados muito mais rápido a partir da memória RAM do que a partir do disco rígido. Mas um disco é geralmente muito maior do que a RAM, e ela é muito mais cara e é apagada quando o computador é reiniciado. Pense em seu escritório como uma metáfora para RAM e disco. Um disco é como um armário no qual você armazena pastas de informações de que precisa. A RAM é como sua mesa, na qual você coloca a pasta de papéis enquanto a está usando, mas a coloca de volta no armário quando não está usando mais.

Se a RAM se encher, por estar executando muitos processos ou por um processo com um vazamento de memória, novos processos falharão se o sistema não tiver uma maneira de estender a memória do sistema. É aí que uma área de troca entra em ação. Um espaço de troca é uma partição do disco rígido em que seu computador pode “tirar” (*swap out*) os dados da memória RAM que não estão sendo usados no momento e, então, “devolver” (*swap in*) os dados para a memória RAM quando eles são novamente necessários. Embora seja melhor nunca exceder sua memória RAM (o desempenho cai quando você faz a troca), tirar os dados da memória é melhor do que deixar os processos simplesmente falharem.

Outra partição especial é um gerenciador de volumes lógicos (LVM) e volumes físicos. Volumes físicos LVM permitem criar conjuntos de espaço de armazenamento chamados *grupos de volumes*. A partir desses grupos de volumes, você tem muito mais flexibilidade para aumentar e diminuir volumes lógicos do que teria redimensionando partições de disco diretamente.

Para Linux, pelo menos uma partição de disco é necessária, atribuída à raiz (/) do sistema de arquivos Linux inteiro. Mas é mais comum ter partições separadas que são atribuídas a diretórios específicos, tais como /home, /var e/ou /tmp. Cada uma das partições está conectada ao sistema de arquivos Linux maior, montando-a em um ponto no sistema de arquivos onde você quer que a partição seja usada. Qualquer arquivo adicionado ao diretório do ponto de montagem de uma partição, ou em um subdiretório, é armazenado na partição.

ta

lavra *montagem* é usada para se referir à ação de conectar um sistema de arquivos de um disco rígido, unidade ou dispositivo de armazenamento de rede a um determinado ponto no sistema de arquivos. Essa ação é feita do comando `mount`, juntamente com opções para dizer ao comando onde o dispositivo de armazenamento e a qual diretório no sistema de arquivos ele deve ser conectado.

O negócio de conectar partições de disco ao sistema de arquivos Linux é feito de forma automática e invisível para o usuário final. Como isso acontece? Cada partição de disco regular criada quando você instala o Linux está associada a um nome de dispositivo. Uma entrada no arquivo `/etc/fstab` informa o nome do dispositivo Linux de cada partição e onde montá-lo (assim como outras informações). A montagem é feita quando o sistema é inicializado.

A maior parte deste capítulo se concentra em explicar como o disco de seu computador é particionado e conectado para formar seu sistema de arquivos Linux, bem como a forma de particionar discos, formatar sistemas de arquivos e espaço de troca e fazer esses itens serem usados durante a inicialização do sistema. O capítulo, então, explica como particionar e criar o sistema de arquivos manualmente.

ndo do Windows

o Linux, os sistemas de arquivos são organizados de maneira diferente de como o são em sistemas operacionais Microsoft Windows. Em vez de letras de unidade (por exemplo, A:, B:, ...), para cada disco local, sistema de arquivos de rede, CD-ROM ou outro tipo de mídia de armazenamento, tudo se encaixa perfeitamente na estrutura de diretórios do Linux.

Algumas unidades são conectadas (montadas) automaticamente no sistema de arquivos quando você insere uma mídia removível. Por exemplo, um CD pode ser montado em `/media/cdrom`. Se a unidade não for montada automaticamente, cabe a um administrador criar um ponto de montagem no sistema de arquivos e, então, conectar o disco a esse ponto.

Linux pode entender sistemas de arquivos `vfat`, que muitas vezes são o formato padrão quando você compra um pen drive USB. Um pen drive USB VFAT oferece uma boa maneira de compartilhar dados entre sistemas Linux e Windows. Suporte ao kernel Linux está disponível para sistemas de arquivos NTFS, que são normalmente utilizados com o Windows nesses dias. Mas o NTFS muitas vezes requer a instalação de drivers de kernel Linux adicionais.

Sistemas de arquivo VFAT costumam ser usados quando é preciso trocar arquivos entre diferentes tipos de sistemas operacionais. Como o VFAT era usado no MS-DOS e nos primeiros sistemas operacionais Windows, ele oferece um menor denominador comum para compartilhar arquivos com muitos tipos de sistemas (incluindo o Linux). O NTFS é o tipo de sistema de arquivos mais comumente usado com modernos sistemas Microsoft Windows.

Particionando discos rígidos

O Linux fornece várias ferramentas para gerenciar suas partições de disco rígido. Você precisa saber como particionar seu disco se quiser adicionar um disco a seu sistema ou alterar a configuração do disco existente.

Esta seção demonstra o particionamento de disco usando um pen drive USB removível de 8GB e um disco rígido fixo. Por segurança, uso um pen drive USB que não contém nenhum dado que eu queira manter para praticar particionamento.

Terar o particionamento pode tornar o sistema não inicializável!

Eu recomendo o uso do disco rígido principal de seu sistema para a prática de particionamento, porque um erro pode tornar seu sistema não inicializável. Mesmo se você usar um pen drive USB em separado para a prática, uma entrada errada em `/etc/fstab`

de travar seu sistema na reinicialização. Se depois de alterar as partições seu sistema parar na inicialização, consulte o Capítulo 21, “Solucionando problemas do Linux”, para obter informações sobre como corrigir o problema.

Visualizando partições de disco

Para visualizar as partições do disco, use o comando `fdisk` com a opção `-l`. Para RHEL 6, use `-c` (desativa o modo de compatibilidade DOS) e `-u` (mostra o tamanho em setores, não cilindros). O exemplo a seguir sobre o particionamento de um pen drive USB removível de 8GB:

```
# fdisk -c -u -l /dev/sdc
Disk /dev/sdc: 8021 MB, 8021606400 bytes
16 heads, 48 sectors/track, 20400 cylinders, total 15667200 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xc3072e18

   Device Boot      Start         End      Blocks   Id  System
/dev/sdc1             2048         4196351    2097152   83  Linux
/dev/sdc2          4196352         5220351     512000   82  Linux swap / Solaris
/dev/sdc3          5220352         7317503    1048576   c   W95 FAT32 (LBA)
/dev/sdc4          7317504         15667199    4174848   5   Extended
/dev/sdc5          7319552         7729151     204800   83  Linux
/dev/sdc6          7731200         15667199    3968000   8e  Linux LVM
```

Quando essa pen drive USB é inserido, ele é atribuído ao próximo dispositivo `sd` disponível. Nesse exemplo, `/dev/sdc` é definido como o nome do dispositivo (o terceiro disco no sistema). Sua unidade pode receber um nome de dispositivo diferente. Você pode ver que o pen drive USB tem 8021MB de tamanho e seis partições. Eis algumas coisas a procurar:

- Um dispositivo de armazenamento USB ou SCSI, representado por um dispositivo `sd`? (como `sda`, `sdb`, `sdc` e assim por diante) pode ter até 16 dispositivos menores (por exemplo, o dispositivo `/dev/sdc` principal e `/dev/sdc1` a `/dev/sdc15`). Assim, pode haver 15 partições no total.
- Para computadores x86, os discos podem ter até quatro partições primárias. Assim, para ter mais de quatro partições no total, pelo menos uma tem de ser uma partição estendida. Observe que `/dev/sdc4` é uma partição estendida que consome todo o espaço em disco não utilizado pelas primeiras três partições. Todas as partições depois disso são partições lógicas que usam o espaço da partição estendida.
- O campo `id` indica o tipo de partição. Observe que há uma mistura de partições Linux, FAT, de troca e Linux LVM.

Seu primeiro disco primário geralmente aparece como `/dev/sda`. Com instalações RHEL e Fedora, geralmente há pelo menos uma partição LVM, a partir da qual outras partições podem ser configuradas. Assim, a saída de `fdisk` pode ser tão simples como o seguinte:

```
# fdisk -cul /dev/sda
Disk /dev/sda: 500.1 GB, 500107862016 bytes
```

255 heads, 63 sectors/track, 60801 cylinders, total 976773168 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes

```
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x000ebb20

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1 *          2048        411647        204800   83   Linux
/dev/sda2            411648    976773119    488180736   8e   Linux LVM
```

A primeira partição tem aproximadamente 200MB e está montada no diretório `/boot`. O asterisco (*) na coluna de inicialização indica que a partição é inicializável (que é onde o kernel e outros componentes necessários para inicializar o sistema são armazenados). O restante do disco é consumido pela partição LVM, que é em última instância usada para criar volumes lógicos. Você pode descobrir mais sobre o LVM na seção “Usando partições LVM”, mais adiante neste capítulo.

Por enquanto, recomendo que você não mexa com o disco rígido intacto e encontre um pen drive USB que você não se importa de apagar. Você pode experimentar os comandos que eu demonstro nessa unidade.

Criando um disco de uma única partição

Para adicionar uma nova mídia de armazenamento (disco rígido, pen drive USB ou dispositivo similar) a seu computador para que ele possa ser usado pelo Linux, você precisa primeiro conectar o dispositivo de disco a seu computador e, então, particionar o disco. Eis o procedimento geral:

1. Instale o novo disco rígido ou insira o novo pen drive USB.
2. Particione o novo disco.
3. Crie o sistema de arquivos no novo disco.
4. Monte os sistemas de arquivos.

A maneira mais fácil de adicionar um disco ou pen drive ao Linux é dedicar o disco inteiro a uma única partição Linux. Mas você pode ter várias partições e atribuir cada uma a diferentes tipos de sistemas de arquivos e diferentes pontos de montagem, se quiser.

O processo a seguir conduz você pelo particionamento de um pen drive USB para ser usado em sistema Linux que tem apenas uma partição. Se tiver um pen drive USB (qualquer tamanho) que não se importe de apagar, você pode reproduzir esse procedimento à medida que o lê. A próxima seção descreve como particionar um disco com várias partições.

Atenção

Você cometerá um erro ao particionar seu disco com o `fdisk`, simplesmente digite `q` para sair sem salvar as alterações. Se você empacar no meio de uma operação, como a adição de uma partição, simplesmente complete a operação e, então, digite `q` depois de ver o prompt de comando.

1. Para um pen drive USB, basta conectá-lo a uma porta USB disponível. Daqui para frente,

uso um pen drive USB de 8 GB, mas você pode usar seu próprio pen drive USB de qualquer tamanho.

2. Determine o nome do dispositivo do disco rígido. Como usuário root de um shell, digite o seguinte comando `tail` e, então, insira o pen drive USB. Aparecem mensagens, indicando o nome do dispositivo da unidade que você acabou de conectar (pressione `Ctrl+C` para sair do comando `tail` quando terminar):

```
# tail -f /var/log/messages
```

```
scsi 6:0:0:0: Direct-Access S31B1102 USB DISK 1100 PQ: 0  
ANSI:
```

```
0 CCS
```

```
sd 6:0:0:0: Attached scsi generic sg2 type 0
```

```
sd 6:0:0:0: [sdc] 8342528 512-byte logical blocks: (4.27  
GB/3.97 GiB)
```

```
sd 6:0:0:0: [sdc] Write Protect is off
```

```
sd 6:0:0:0: [sdc] Mode Sense: 43 00 00 00
```

```
sd 6:0:0:0: [sdc] Assuming drive cache: write through
```

```
sd 6:0:0:0: [sdc] Assuming drive cache: write through sdc:  
sdc1
```

```
sd 6:0:0:0: [sdc] Assuming drive cache: write through
```

```
sd 6:0:0:0: [sdc] Attached SCSI removable disk
```

3. A partir da saída, você pode ver que o pen drive USB foi encontrado e atribuído a `/dev/sdc`. (O nome do seu dispositivo pode ser diferente!) Ele também contém uma partição: `/dev/sdc1`. Certifique-se de identificar o disco correto ou você pode perder todos os dados de discos que talvez você queira manter!
4. Se o pen drive USB for montado automaticamente, desmonte-o. Na sua área de trabalho, clique no ícone `Filesystem 8.0GB` que aparece e selecione `Unmount`. Ou, nesse caso, como root você poderia, em vez disso, digitar `umount /dev/sdc1`.
5. Use o comando `fdisk` para criar partições no disco novo. Por exemplo, se você estiver formatando o terceiro disco USB, SATA ou SCSI (`sdc`), pode digitar o seguinte:

```
# fdisk -c -u /dev/sdc
```

```
Command (m for help):
```

Agora você está no modo de comando `fdisk`, no qual pode usar o comando `fdisk` de uma única letra configurado para trabalhar com suas partições. (Adicionar as opções `-c -u` permite que você selecione o tamanho de cada partição com base em setores em vez de cilindros).

6. Se você começar com um novo pen drive USB, ele pode ter uma partição que é inteiramente dedicada a um sistema de arquivos compatível com Windows (como VFAT). Use `p` para ver todas as partições e `d` para excluir a partição. Eis a saída do comando quando fiz isso:

```

Command (m for help): p
...
  Device Boot   Start       End   Blocks  Id System
/dev/sdc1      2048    15667199  7832576   c  W95 FAT32 (LBA)
Command (m for help): d
Selected partition 1

```

7. Para criar uma nova partição, digite a letra **n**. Você será solicitado a informar o tipo de partição.
8. Escolha uma partição estendida (e) ou primária (p). Digite a letra **p** para escolher primária.
9. Digite o número da partição. Se você estiver criando a primeira partição (ou para somente uma partição), digite o número **1**. Você será solicitado a informar o primeiro setor para iniciar a partição.
10. Selecione o número do primeiro setor disponível (você pode simplesmente pressionar enter para escolher). Você será solicitado a informar o último setor.
11. Digite o tamanho da partição. Como você está apenas criando uma partição para consumir todo o disco, escolha o último setor disponível. Para fazer isso você pode simplesmente pressionar Enter para aceitar o padrão.
12. Verifique se a unidade está dividida como você quer pressionando **p**. (Sua saída será diferente, dependendo do tamanho de seu disco.)

```

Command (m for help): p
...
  Device Boot   Start       End   Blocks  Id System
/dev/sdb1      2048    15667199  7832576   83  Linux

```

13. Para fazer alterações na tabela de partições permanentemente, digite **w**. Isso gravará as alterações, experimente sincronizá-las com o kernel do Linux e encerrar o `fdisk`. Se vir uma mensagem como a seguinte, não se preocupe, você pode corrigir isso na próxima etapa:


```

WARNING: Re-reading the partition table failed with
error 16:
Device or resource busy.

```
14. Se `fdisk` não puder sincronizar a tabela de partições no disco com o kernel, o motivo mais provável é que uma partição do disco ainda está montada. Desmonte a partição e tente executar o seguinte comando para sincronizar a tabela de partição do disco com o kernel:

```
# partprobe /dev/sdc
```

Se `partprobe` não funcionar, reiniciar o computador vai garantir que o disco e o kernel estejam em sincronia.

15. Embora o particionamento esteja concluído, a nova partição ainda não está pronta para uso. Para isso, você tem de criar um sistema de arquivos na nova partição. Para criar um sistema de arquivos na nova partição do disco, use o comando `mkfs`. Por padrão, o comando cria um sistema de arquivos `ext2`, que é utilizável pelo Linux. Mas na maioria dos casos, você vai querer usar um sistema de arquivos de *journaling* (como `ext3` ou `ext4`). Para criar um sistema de arquivos `ext4` na primeira partição do terceiro disco rígido, digite o seguinte:


```
# mkfs -t ext4 /dev/sdc1
```

ca

pode usar outros comandos ou opções para esse comando, para criar tipos de sistema de arquivos. Por exemplo, use `mkfs.vfat` para criar um sistema de arquivos VFAT, `mkfs.msdos` para DOS ou `mkfs.reiserfs` para o tipo de sistema de arquivos Reiser. Você pode querer um sistema de arquivos VFAT se quiser compartilhar arquivos entre sistemas Linux, Windows e Mac.

16. Para ser capaz de usar o novo sistema de arquivos, você precisa criar um ponto de montagem e montá-lo na partição. Eis um exemplo de como fazer isso. Você, então, verifica se a montagem foi bem-sucedida.

```
# mkdir /mnt/test
# mount /dev/sdc1 /mnt/test
# df -h /mnt/test
Filesystem      Size  Used Avail Use% Mounted on
/dev/sdc1       7.4G   17M   7.0G   1% /mnt/test
# mount | grep sdc1
/dev/sdc1 on /mnt/test type ext4 (rw)
```

O comando `df` mostra que `/dev/sdc1` está montado em `/mnt/test` e que oferece cerca de 7,4GB de espaço em disco. O comando `mount` mostra todos os sistemas de arquivos montados, mas aqui usei `grep` para mostrar que `sdc1` está montado e é um tipo de sistema de arquivos `ext4`.

Todos os arquivos ou diretórios que você criar posteriormente no diretório `/mnt/test` e qualquer um de seus subdiretórios serão armazenados no dispositivo `/dev/sdc1`.

17. Quando terminar de utilizar a unidade, você pode desmontá-la com o comando `umount`, após o qual você pode remover a unidade com segurança (veja a descrição do comando `umount` mais adiante, se esse comando falhar):

```
# umount /dev/sdc1
```

18. Você não costuma configurar um pen drive USB para montar automaticamente toda vez que inicializar o sistema, pois ele é montado automaticamente quando você o conecta. Mas se decidir que quer fazer isso, edite `/etc/fstab` e adicione uma linha descrevendo o que e onde montar. Eis um exemplo de uma linha que você pode acrescentar:

```
/dev/sdc1 /mnt/test ext4 defaults 0 1
```

Nesse exemplo, a partição (`/dev/sdc1`) está montada no diretório `/mnt/test` como um sistema de arquivos `ext4`. A palavra-chave `defaults` faz com que a partição seja montada durante a inicialização. O número `0` instrui o sistema a não fazer backup de arquivos desse sistema de arquivos com o comando `dump` (`dump` é raramente usado hoje em dia, mas o campo está aqui). O `1` na última coluna indica ao sistema para verificar se há erros na partição após certo número de montagens.

Nesse ponto, você tem uma partição de disco permanentemente montada e funcionando. A próxima seção descreve como particionar um disco que tem várias partições.

Criando um disco de múltiplas partições

Agora que você entende o processo básico de particionar um disco e de adicionar um sistema de arquivos e torná-lo disponível (temporária e permanentemente), é hora de tentar um exemplo mais complexo. Levando esse mesmo pen drive USB de 8GB, segui o procedimento descrito mais adiante nesta seção para criar múltiplas partições em um disco.

Nesse procedimento, eu crio uma partição de 500MB (`sdC1` e `sdC2`), 300MB (`sdC3`), 350MB (`sdC5`) e 400MB (`sdC6`). O dispositivo `sdC4` é uma partição estendida, que consome todo o espaço restante do disco. O espaço das partições `sdC5` e `sdC6` é tomado da partição estendida.

Como antes, insira o pen drive USB e determine o nome do dispositivo (no meu caso, `/dev/sdc`). Além disso, certifique-se de desmontar as partições que são montadas automaticamente quando você insere o pen drive USB.

ca

Quando você indicar o tamanho de cada partição, digite o sinal de mais e o número de megabytes ou gigabytes que deseja atribuir à partição. Por exemplo, `+1024M` para criar uma partição de 1024 megabytes ou `+10G` para partição de 10 gigabytes. Certifique-se de lembrar do sinal de mais (+) e M ou G! Se você esquecer o M ou G, `sdC` vai pensar que você quer dizer setores e você vai obter resultados inesperados.

1. Para começar, abra o dispositivo `/dev/sdc` com `fdisk`, exclua (somente) a primeira partição e, então, adicione seis novas partições.

```
# fdisk -cu /dev/sdc
Command (m for help): d
Selected partition 1
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
P
Partition number (1-4): 1
First sector (2048-15667199, default 2048):
<Enter>
Using default value 2048
Last sector, +sectors or +size(K,M,G) (2048-15667199, default
15667199):+500M
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
P
Partition number (1-4): 2
First sector (1026048-15667199, default 1026048):
<Enter>
Using default value 1026048
Last sector, +sectors or +size
(K,M,G) (default 15667199):+500M
Command (m for help): n
Command action
   e   extended
   p   primary partition (1-4)
P
```

```

Partition number (1-4): 3
First sector (2050048-15667199, default 2050048):
<Enter>
Using default value 2050048
Last sector, +sectors or +size {K,M,G} (...default 15667199):+300M
Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
e Selected partition 4
First sector (2664448-15667199, default 2664448):
<Enter>
Using default value 2664448
Last sector,+sectors or + size{K,M,G}(... default 15667199):
<Enter>
Using default value 15667199
Command (m for help): n
First sector (2666496-15667199, default 2666496):
<Enter>
Using default value 2666496
Last sector, +sectors or +size{K,M,G} (...default 15667199): +350M
Command (m for help): n
First sector (...default 3385344):
<Enter>
Using default value 3385344
Last sector, +sectors or +size {K,M,G} (...default 15667199): +400M

```

2. Verifique o particionamento antes de salvar digitando **p**. Observe que há cinco partições utilizáveis (sdc1, sdc2, sdc3, sdc5 e sdc6) e que os setores entre o início e o fim de sdc4 estão sendo consumidos por sdc5 e sdc6.

```

Command (m for help): p
...

```

Device	Boot	Start	End	Blocks	Id	System
/dev/sdc1		2048	1026047	512000	83	Linux
/dev/sdc2		1026048	2050047	512000	83	Linux
/dev/sdc3		2050048	2664447	307200	83	Linux
/dev/sdc4		2664448	15667199	6501376	5	Extended
/dev/sdc5		2666496	3383295	358400	83	Linux
/dev/sdc6		3385344	4204543	409600	83	Linux

3. O tipo de partição padrão é Linux. Decidi que queria usar algumas das partições para espaço de troca (type 82), FAT32 (type x) e Linux LVM (type 8e). Para fazer isso, digitei **t** e indiquei qual tipo de partição usar. Digite **L** para ver uma lista de partições.

```

Command (m for help): t
Partition number (1-6): 2
Hex code (type L to list codes): 82
Changed system type of partition 2 to 82 (Linux swap / Solaris)
Command (m for help): t
Partition number (1-6): 5
Hex code (type L to list codes): c
Changed system type of partition 5 to c (W95 FAT32 (LBA))
Command (m for help): t
Partition number (1-6): 6
Hex code (type L to list codes): 8e
Changed system type of partition 6 to 8e (Linux LVM)

```

4. Verifico se a tabela de partições está do jeito que quero e depois gravo as alterações:

```

Command (m for help): p
...
  Device Boot      Start         End      Blocks   Id  System
/dev/sdc1          2048        1026047    512000    83  Linux
/dev/sdc2          1026048    2050047    512000    82  Linux swap /
Solaris
/dev/sdc3          2050048    2664447    307200    83  Linux
/dev/sdc4          2664448    15667199    6501376    5  Extended
/dev/sdc5          2666496    3383295    358400    c  W95 FAT32 (LBA)
/dev/sdc6          385344     4204543    409600    8e  Linux LVM
Command (m for help): w
The partition table has been altered!
Calling ioctl() to re-read partition table.
...

```

- Depois que a gravação estiver concluída, verifique se o kernel sabe sobre as alterações na tabela de partição. Para fazer isso, pesquise `sdc` em `/proc/partitions`. Se os novos dispositivos não estiverem lá, execute o comando `partprobe /dev/sdc` na unidade ou reinicie seu computador.

```

# grep sdc /proc/partitions
8      32      7833600 sdc
8      33      512000 sdc1
8      34      512000 sdc2
8      35      307200 sdc3
8      36          1 sdc4
8      37      358400 sdc5
8      38      409600 sdc6

```

- Embora as partições estejam agora definidas para diferentes tipos de conteúdo, outros comandos são necessários para estruturar as partições em sistemas de arquivos ou áreas de troca. Veja como fazer isso para as partições recém-criadas:

- **sdc1** — Para fazer isso em um sistema de arquivos Linux `ext4` normal, digite o seguinte:

```
# mkfs -t ext4 /dev/sdc1
```

- **sdc2** — Para formatar isso como uma área de troca, digite o seguinte:

```
# mkswap /dev/sdc2
```

- **sdc3** — Para fazer isso em um sistema de arquivos `ext2` (o padrão) digite o seguinte:

```
# mkfs /dev/sdc3
```

- **sdc5** — Para fazer isso em um sistema de arquivos VFAT (o padrão) digite o seguinte:

```
# mkfs -t vfat /dev/sdc5
```

- **sdc6** — Para fazer isso em um volume LVM físico digite o seguinte:

```
# pvcreate /dev/sdc6
```

Essas partições agora estão prontas para serem montadas, usadas como área de troca ou adicionadas a um grupo de volumes LVM. Consulte a próxima seção, “Usando partições LVM”, para ver como volumes LVM físicos são usados para, em última instância, criar volumes LVM lógicos a partir de grupos de volumes. Veja a seção “Montando sistemas de arquivos” para descrições de como montar sistemas de arquivos e habilitar áreas de troca.

Usando partições LVM

O particionamento de disco básico no Linux tem suas deficiências. O que acontece se você ficar sem

espaço em disco? Uma solução comum é copiar os dados para um disco maior, reiniciar o sistema com o disco novo e esperar que você não fique sem espaço novamente em breve. Esse processo significa tempo de inatividade e ineficiência.

O Logical Volume Manager (LVM) oferece muita flexibilidade e eficiência em lidar com a constante evolução das necessidades de armazenamento. Com o LVM, partições de discos físicas são adicionadas aos conjuntos de espaço chamados grupos de volumes. Volumes lógicos recebem espaço a partir de grupos de volumes conforme necessário. Isso lhe dá a capacidade de:

- Adicionar mais espaço a um volume lógico a partir do grupo de volumes enquanto o volume ainda está em uso.
- Adicionar mais volumes físicos a um grupo de volumes, se o grupo de volumes começar a ficar sem espaço. Os volumes físicos podem ser de discos.
- Mover dados de um volume físico para outro, de modo que você possa remover discos menores e substituí-los por outros maiores, enquanto os sistemas de arquivos ainda permanecem em uso — mais uma vez, sem tempo de inatividade.

Com o LVM também é mais fácil encolher sistemas de arquivos para recuperar espaço em disco, embora isso exija que você desmonte o volume lógico (mas não é necessário reiniciar). O LVM também suporta recursos avançados, como espelhamento e trabalhar em *clusters* (grupamentos).

Verificando uma LVM existente

Vamos começar examinando um exemplo LVM existente em um sistema Red Hat Enterprise Linux. O comando a seguir exibe as partições no meu primeiro disco rígido:

```
# fdisk -cul /dev/sda | grep /dev/sda
Disk /dev/sda: 160.0 GB, 160000000000 bytes
/dev/sda1 *      2048    1026047    512000    83  Linux
/dev/sda2 *     1026048  312498175 155736064    8e  Linux LVM
```

Nesse sistema RHEL, o disco rígido de 160GB está dividido em uma partição Linux de 500MB (*sda1*) e uma segunda partição (Linux LVM) que consome o resto do disco (*sda2*). Em seguida, uso o comando `pvdisplay` para ver se a partição está sendo usada em um grupo LVM:

```
# pvdisplay /dev/sda2
--- Physical volume ---
PV Name                /dev/sda2
VG Name                vg_abc
PV Size                148.52 GiB / not usable 2.00 MiB
Allocatable           yes (but full)
PE Size                4.00 MiB
Total PE               38021
Free PE                0
Allocated PE           38021
PV UUID                wlvuIv-UiI2-pNND-f39j-cH0X-9t0o-A0II7R
```

Você pode ver que o volume físico LVM representado por `/dev/sda2` tem 148,52GiB do espaço, que foi totalmente alocado para um grupo de volumes chamado `vg_abc`. A menor unidade de armazenamento que pode ser usada a partir desse volume físico é 4,0MiB, o que é referido como uma extensão física (PE).

ta

erve que as ferramentas LVM mostram o espaço em disco em MiB e GiB. Um MB é 1.000.000 bytes (10^6), enquanto um MiB é 1.048.576 bytes (2^{20}). Um MiB é uma maneira mais precisa de refletir como os dados são armazenados em um computador. Mas as pessoas de marketing tendem a usar MB porque faz parecer que os discos, CDs e DVDs que eles vendem têm mais capacidade do que realmente têm. Tenha em mente que a maioria das ferramentas no Linux exibe dados de armazenamento em MiB e GiB, embora algumas também possam mostrar em GB. Então, você quer ver informações sobre o grupo de volumes:

```
# vgsdisplay vg_abc
--- Volume group ---
VG Name                vg_abc
System ID
Format                 lvm2
Metadata Areas        1
Metadata Sequence No  4
VG Access              read/write
VG Status              resisable
MAX LV                 0
Cur LV               3
Open LV               3
Max PV                0
Cur PV               1
Act PV                1
VG Size                148.52 GiB
PE Size                4.00 MiB
Total PE              38021
Alloc PE / Size       38021 / 148.52 GiB
Free PE / Size        0 / 0
VG UUID                c25GHM-KU9H-wbXM-sgca-EtEr-UMaQ-Unn8Th
```

Você pode ver que todas as 38.021 PEs foram alocadas. Usando `lvdisplay` da maneira a seguir, você pode ver onde elas foram alocadas (eu cortei parte da saída):

```
# lvdisplay vg_abc
--- Logical volume ---
LV Name                /dev/vg_abc/lv_root
VG Name                vg_abc
LV UUID                33VeDc-jd01-h1Cc-RMuB-tkcw-QvFi-cKCEqa
LV Write Access        read/write
LV Status              available
# open                 1
LV Size                50.00 GiB
Current LE             12800
Segments               1
Allocation             inherit
Read ahead sectors     auto
- currently set to    256
Block device           253:0
--- Logical volume ---
LV Name                /dev/vg_abc/lv_home
VG Name                vg_abc
...
LV Size                92.64 GiB
--- Logical volume ---
LV Name                /dev/vg_abc/lv_swap
VG Name                vg_abc
...
LV Size                5.88 GiB
```

Há três volumes lógicos emprestando espaço de `vg_abc`. Cada volume lógico é associado a um nome de dispositivo que inclui o nome do grupo de volumes e o nome do volume lógico: `/dev/vg_abc/lv_root` (50GB), `/dev/vg_abc/lv_home` (92,64GB) e `/dev/vg_abc/lv_swap` (5,88GB). Outros dispositivos ligados a esses nomes estão localizados no diretório `/dev/mapper:vg_abc-lv_home,vg_abc-lv_root` e `vg_abc-lv_swap`. Qualquer conjunto de nomes pode ser usado para referenciar esses volumes lógicos.

Os volumes lógicos root e home são formatados como ext4, enquanto o volume de troca lógico é formatado como espaço de troca. Vamos dar uma olhada no arquivo `/etc/fstab` para ver como esses volumes lógicos são utilizados:

```
# grep vg_ /etc/fstab
/dev/mapper/vg_abc-lv_root /          ext4 defaults 1 1
/dev/mapper/vg_abc-lv_home /home  ext4 defaults 1 2
/dev/mapper/vg_abc-lv_swap swap   swap  defaults 0 0
```

A Figura 12.1 ilustra como as diferentes partições, grupos de volumes e volumes lógicos se relacionam com o sistema de arquivos Linux completo. O dispositivo `sda1` é formatado como um sistema de arquivos e montado no diretório `/boot`. O dispositivo `sda2` fornece espaço para o grupo de volumes `vg_abc`. Assim, volumes lógicos `lv-home` e `lv-root` são montados nos diretórios `/home` e `/`, respectivamente.

es lógicos LVM podem ser montados como partições regulares em um sistema de arquivos Linux.

Se ficar sem espaço em qualquer um dos volumes lógicos, você pode atribuir mais espaço a partir do grupo de volumes. Se o grupo de volumes ficar sem espaço, você pode adicionar outro disco rígido ou unidade de armazenamento de rede e adicionar espaço a partir dessa unidade ao grupo de volumes, disponibilizando, assim, mais espaço.

Agora que você sabe como funciona o LVM, a próxima seção mostra como criar volumes lógicos LVM a partir do zero.

Criando volumes lógicos LVM

Volumes lógicos LVM são usados de cima para baixo, mas são criados de baixo para cima. Como ilustrado na Figura 12.1, primeiro você cria um ou mais volumes físicos (pv), utiliza os volumes físicos para criar grupos de volumes (vg) e, então, cria volumes lógicos a partir dos grupos de volumes (lv).

Os comandos para trabalhar com cada componente LVM começam com as letras pv, vg e lv. Por exemplo, `pvdisplay` mostra volumes físicos, `vgdisplay` mostra grupos de volumes e `lvdisplay` mostra volumes lógicos.

O procedimento a seguir conduz você pelas etapas da criação de volumes LVM a partir do zero. Para fazer esse procedimento, você poderia usar o pen drive USB e as partições que descrevi no início deste capítulo.

1. Obtenha um disco com algum espaço livre e crie uma partição de disco do tipo LVM (8e). Então, utilize o comando `pvcreate` para identificar essa partição como um volume físico LVM. O processo para fazer isso é descrito na seção “Criando uma partição de múltiplos discos”, usando o `/dev/sdc6` device daquele exemplo.
2. Para adicionar esse volume físico a um novo grupo de volumes, use o comando `vgcreate`. O comando a seguir mostra como criar um grupo de volumes chamado `myvg0` usando o dispositivo `/dev/sdc6`:

```
# vgcreate myvg0 /dev/sdc6
Volume group "myvg0" successfully created
```

3. Para ver o novo grupo de volumes, digite o seguinte:

```
# vgdisplay myvg0
--- Volume group ---
VG Name                myvg0
...
VG Size                 396.00 MiB
PE Size                4.00 MiB
Total PE               99
Alloc PE / Size        0 / 0
Free PE / Size         99 / 396.00 MiB
```


- Da partição de 400MiB, 396 MiB de espaço podem ser usados em blocos de 4MiB. Veja como criar um volume lógico a partir de uma parte do espaço nesse grupo de volumes e então verifique se o dispositivo para esse volume lógico existe:

```
# lvcreate -n music -L 100M myvg0
Logical volume "music" created
# ls /dev/mapper/myvg0*
/dev/mapper/myvg0-music
```

- Como você pode ver, o procedimento criou um dispositivo chamado `/dev/mapper/myvg0-music`. Esse dispositivo já pode ser usado para colocar e montar um sistema de arquivos nele, assim como você fez com partições regulares na primeira parte deste capítulo. Por exemplo:

```
# mkfs -t ext4 /dev/mapper/myvg0-music
# mkdir /mnt/mymusic
# mount /dev/mapper/myvg0-music /mnt/mymusic
# df -h /mnt/mymusic
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/myvg0-music 97M  5.6M   87M   7% /mnt/mymusic
```

- Tal como acontece com as partições regulares, os volumes lógicos podem ser montados permanentemente adicionando uma entrada no arquivo `/etc/fstab`, por exemplo:

```
/dev/mapper/myvg0-music /mnt/mymusic ext4 defaults 1 2
```

Da próxima vez que você reiniciar, o volume lógico é automaticamente montado em `/mnt/mymusic`. (Certifique-se de desmontar o volume lógico e remover essa linha se quiser remover o pen drive USB de seu computador.)

Aumentando volumes lógicos LVM

Se ficar sem espaço em um volume lógico, você pode adicionar espaço a ele, mesmo sem desmontá-lo. Para fazer isso, você deve ter espaço disponível no grupo de volumes, aumentar o volume lógico e aumentar o sistema de arquivos para preenchê-lo. Com base no procedimento da seção anterior, eis como aumentar um volume lógico:

- Observe quanto espaço está atualmente no volume lógico e depois verifique se há espaço disponível no grupo de volumes do volume lógico:

```
# vgdisplay myvg0
...
VG Size          396.00 MiB
PE Size          4.00 MiB
Total PE         99
Alloc PE / Size  25 / 100.00 MiB
Free PE / Size   74 / 296.00 MiB
# df -h /mnt/mymusic/
Filesystem      Size  Used  Avail Use% Mounted on
/dev/mapper/myvg0-music 97M  5.6M   87M   7% /mnt/mymusic
```

- Expanda o volume lógico usando o comando `lvextend`:

```
# lvextend -L +100M /dev/mapper/myvg0-music
Extending logical volume music to 200.00 MiB
Logical volume music successfully resized
```

3. Redimensione o sistema de arquivos para ajustá-lo ao novo tamanho do volume lógico:
resize2fs -p /dev/mapper/myvg0-music
4. Verifique se o sistema de arquivos agora está redimensionado para incluir o espaço em disco adicional.

```
# df -h /mnt/mymusic/
Filesystem      Size  Used Avail Use% Mounted on
/dev/mapper/myvg0-music 194M  5.6M 179M   3% /mnt/mymusic
```

Você pode ver que o sistema de arquivos tem agora mais 100MB.

Montando sistemas de arquivos

Agora que você já teve a oportunidade de brincar com o particionamento de disco e os sistemas de arquivos, vou dar um passo para trás e falar sobre como os sistemas de arquivos são configurados para se conectarem permanentemente ao seu sistema Linux.

A maioria das partições de disco rígido criadas quando você instala o Linux são montadas automaticamente para você quando o sistema é inicializado. Quando instala o Fedora, o Ubuntu, o Red Hat Enterprise Linux e outros sistemas Linux, você tem a opção de deixar o instalador configurar automaticamente seu disco rígido ou criar partições você próprio e indicar os pontos de montagem para elas.

Quando você inicia o Linux, normalmente todas as partições Linux em seu disco rígido estão listadas no arquivo `/etc/fstab` e são montadas. Por essa razão, esta seção descreve o que você pode esperar encontrar no arquivo. Ela também descreve como você pode montar outras partições para que se tornem parte de seu sistema de arquivos Linux.

O comando `mount` é usado não apenas para montar os dispositivos, mas também para montar outros tipos de sistemas de arquivos em seu sistema Linux. Por exemplo, `mount` pode ser usado para montar diretórios (pastas) sobre a rede de servidores NFS ou Samba. Ele também pode ser usado para montar sistemas de arquivos a partir de um novo disco rígido ou pen drive USB que não está configurado para automontagem.

ta

a adição de recursos de montagem automática e mudanças na forma como a mídia removível é identificada com o kernel Linux 2.6 (usando recursos como Udev e camada de abstração de hardware), você não precisa mais montar manualmente mídia removível para muitos sistemas desktop Linux. A compreensão de como montar e desmontar manualmente sistemas de arquivos em um servidor Linux, porém, pode ser uma habilidade muito útil se você quiser montar sistemas de arquivos remotos ou temporariamente montar partições em determinados locais.

Sistemas de arquivos suportados

Para ver os tipos de sistemas de arquivos que estão atualmente carregados no kernel, digite **cat**

/proc/filesystems. A lista que se segue mostra um exemplo dos tipos de sistemas de arquivos que são suportados no Linux, embora eles possam não estar em uso no momento ou nem mesmo disponíveis na distribuição Linux que você está usando.

- **adfs** — Sistema de arquivos de disco Acorn, que é o sistema de arquivos padrão utilizado em sistemas operacionais RiscOS.
- **befs** — Sistema de arquivos usado pelo sistema operacional BeOS.
- **btrfs** — Um sistema de arquivos de “cópia na gravação” que implementa recursos avançados de sistema de arquivos. Ele oferece tolerância a falhas e fácil administração.
- **cifs** — Common Internet Filesystem (CIFS), o sistema virtual de arquivos usado para acessar os servidores que seguem a especificação SNIA CIFS. CIFS é uma tentativa de aperfeiçoar e padronizar o protocolo SMB usado pelo compartilhamento de arquivos Samba e Windows.
- **ext4** — Sucessor do popular sistema de arquivos ext3. Inclui muitas melhorias em relação ao ext3, como suporte a volumes de até 1 exbibyte e tamanhos de arquivo de até 16 tebibytes. (Esse sistema substituiu o ext3 como sistema de arquivos padrão usado no Fedora e RHEL.).
- **ext3** — sistemas de arquivos Ext são os mais comuns na maioria dos sistemas Linux. O sistema de arquivos ext3, também chamado de terceiro sistema de arquivos estendido, inclui recursos de *journaling* que, em comparação com o ext2, melhora a capacidade de um sistema de arquivos para se recuperar de falhas.
- **ext2** — O tipo de sistema de arquivos padrão para sistemas Linux antigos. Os recursos são os mesmos que o ext3, exceto que o ext2 não inclui recursos de diário.
- **ext** — Essa é a primeira versão do ext3. Ele não é mais utilizado muito frequentemente.
- **iso9660** — Evoluiu a partir do sistema de arquivos High Sierra (o padrão original para CD-ROMs). Extensões para o padrão High Sierra (chamadas extensões Rock Ridge) permitem que sistemas de arquivos iso9660 suportem nomes de arquivos longos e de informação no estilo UNIX (como permissões de arquivo, propriedade e vínculos). CD-ROMs de dados costumam usar esse tipo de sistema de arquivos.
- **kafs** — Sistema de arquivos de cliente AFS. Usado em ambientes de computação distribuída para compartilhar arquivos com clientes Linux, Windows e Macintosh.
- **minix** — Tipo de sistema de arquivos Minix, usado originalmente com a versão Minix do UNIX. Suporta nomes de arquivos com no máximo 30 caracteres.
- **msdos** — Um sistema de arquivos MS-DOS. Você pode utilizar esse tipo para montar disquetes que vêm de sistemas operacionais da Microsoft.
- **vfat** — Sistema de arquivos de FAT estendida (VFAT) da Microsoft.
- **umsdos** — Um sistema de arquivos do MS-DOS com extensões para permitir recursos que são semelhantes ao UNIX (incluindo nomes de arquivos longos).

- **proc** — Não um sistema de arquivo real, mas sim uma interface de sistema de arquivos para o kernel Linux. Você provavelmente não vai fazer nada de especial para configurar um sistema de arquivos proc. Mas o ponto de montagem `/proc` deve ser um sistema de arquivos proc. Muitas empresas usam `/proc` para ter acesso a informações do kernel do Linux.
- **reiserfs** — Sistema de arquivos ReiserFS com *journaling*. O ReiserFS já foi um tipo de sistema de arquivos padrão comum para várias distribuições Linux. Mas o `ext` é de longe o tipo mais comum de sistema de arquivos usado com o Linux hoje.
- **swap** — Usado para partições de troca. Áreas de troca são usadas para armazenar dados temporariamente quando a RAM está cheia. Os dados são transferidos para a área de troca e depois transferidos de volta para a memória RAM quando são necessários novamente.
- **squashfs** — Tipo de sistema de arquivos compactado, somente leitura. O Squashfs é popular em live CDs, onde há um espaço limitado e uma mídia somente leitura (por exemplo, um CD ou DVD).
- **nfs** — Tipo de sistema de arquivos Network Filesystem (NFS). O NFS é usado para montar sistemas de arquivos em Linux ou outros computadores UNIX.
- **hpfs** — Um sistema de arquivos usado para montagens somente leitura de um sistema de arquivos OS/2 HPFS.
- **ncpfs** — Um sistema de arquivos usado com o Novell NetWare. Sistemas de arquivos NetWare podem ser montados em uma rede.
- **ntfs** — Sistema de arquivos Windows NT. Dependendo da distribuição que você tem, ele pode ser suportado como um sistema de arquivos somente leitura (de modo que você pode montar e copiar arquivos a partir dele).
- **affs** — Sistema de arquivos usado em computadores Amiga.
- **ufs** — Sistema de arquivos popular em sistemas operacionais Sun Microsystems (isto é, Solaris e SunOS).
- **jfs** — Um sistema de arquivos de 64 bits da IBM com *journaling*, que é relativamente leve levando em conta os muitos recursos que ele tem.
- **xfs** — Um sistema de arquivos de alta performance desenvolvido originalmente pela Silicon Graphics, que funciona muito bem com arquivos grandes.
- **gfs2** — Um sistema de arquivos de disco compartilhado que permite que múltiplas máquinas usem o mesmo disco compartilhado, sem passar por uma camada de sistema de arquivos de rede, como CIFS, NFS etc.

Para ver a lista de sistemas de arquivos que vêm com o kernel que você está usando, digite `ls /lib/modules/versão_do_kernel/kernel/fs/`. Os módulos reais são armazenados em subdiretórios do diretório. A montagem de um sistema de arquivos de um tipo suportado faz com que o módulo de sistema de arquivos seja carregado, se ainda não estiver carregado.

Digite `man fs` para ver as descrições de sistemas de arquivos Linux.

Ativando áreas de troca

A área de troca é uma área do disco que está disponível para Linux se o sistema ficar sem memória (RAM). Se sua memória RAM estiver cheia e você tenta iniciar outro aplicativo sem uma área de troca, esse aplicativo falhará. Com uma área de troca, o Linux pode, temporariamente, transferir dados da RAM para a área de troca e depois recuperá-los quando necessário. Isso afeta negativamente o desempenho, mas é melhor do que deixar os processos falharem. Para realmente usar uma partição de troca que você criou, você precisa ativá-la. Para fazer isso temporariamente, você pode usar o comando `swapon`. Por exemplo:

```
# free -m
              total    used    free   shared  buffers   cached
Mem:           3629    1834    1795        0     226     701
-/+ buffers/cache:    905    2723
Swap:          4095    3995     100
# swapon /dev/sdc2
# free -m
              total    used    free   shared  buffers   cached
Mem:           3629    1834    1795        0     226     701
-/+ buffers/cache:    905    2723
Swap:          4595    4445     600
```

O comando `free` mostra a quantidade de troca antes e depois de ativar a área de troca com o comando `swapon`. Essa quantidade de troca está disponível imediata e temporariamente para seu sistema. Para tornar essa área de troca permanente, você precisa adicioná-la a seu arquivo `/etc/fstab`. Eis um exemplo:

```
/dev/sdc2 swap swap defaults 0 0
```

Essa entrada indica que a área de troca em `/dev/sdc2` deve ser ativada durante a inicialização. Como não há nenhum ponto de montagem para a área de troca, o segundo campo é ajustado apenas para isso, assim como o tipo de partição.

Desativando a área de troca

Se em algum momento você quiser desabilitar uma área de troca, pode fazer isso usando o comando `swapoff`. Você pode fazer isso, em particular, se a área de troca não for mais necessária e você quiser recuperar o espaço que está sendo consumido por um arquivo de troca ou remover um drive USB que está fornecendo uma partição troca.

Primeiro, certifique-se de que nenhum espaço está sendo usado no dispositivo de troca (usando o comando `free`) e, então, use `swapoff` para desligar a área de troca para que você possa reutilizar o espaço. Eis um exemplo:

```
# free -m
              total    used    free   shared  buffers   cached
Mem:           3629    2433    1195        0      99     580
-/+ buffers/cache:   1754    1874
Swap:          4095        0    4095
# swapoff /dev/sdc2
```

Utilizando o arquivo `fstab` para definir sistemas de arquivos montáveis

As partições do disco rígido de seu computador local e os sistemas de arquivos remotos que você usa todos os dias estão, provavelmente, configurados para montar automaticamente quando você inicia o Linux. O arquivo `/etc/fstab` contém definições para cada partição, juntamente com opções que descrevem como a partição está montada. Eis um exemplo de um arquivo `/etc/fstab`:

```
# /etc/fstab
/dev/mapper/vg_abc-lv_root      /          ext4 defaults 1 1
UUID=78bdae46-9389-438d-bfee-06dd934fae28 /boot ext4 defaults 1 2
/dev/mapper/vg_abc-lv_home    /home     ext4 defaults 1 2
/dev/mapper/vg_abc-lv_swap    swap      swap defaults 0 0
# Mount entries added later.
/dev/sdb1                     /win      vfat ro        1 2
192.168.0.27:/nfsstuff        /remote   nfs users,_netdev 0 0
//192.168.0.28/myshare        /share    cifs guest,_netdev 0 0
# special Linux filesystems
tmpfs                         /dev/shm tmpfs defaults 0 0
devpts                        /dev/pts devpts gid=5,mode=620 0 0
sysfs                         /sys      sysfs defaults 0 0
proc                          /proc     proc defaults 0 0
```

O arquivo `/etc/fstab` mostrado acima é de uma instalação de servidor Red Hat Enterprise Linux 6 padrão, com algumas linhas adicionadas.

Por enquanto, você pode ignorar as entradas `tmpfs`, `devpts`, `sysfs` e `proc`. Esses são dispositivos especiais associados com memória compartilhada, janelas de terminal, informações sobre o dispositivo e parâmetros do kernel, respectivamente.

Em geral, a primeira coluna do arquivo `/etc/fstab` mostra o dispositivo ou compartilhamento (o que estiver montado), enquanto a segunda coluna mostra o ponto de montagem (onde ele está montado). Isso é seguido pelo tipo de sistema de arquivos, as opções de montagem (ou padrões) e dois números (usados para dizer a comandos como `dump` e `fsck` o que fazer com o sistema de arquivos).

As três primeiras entradas representam as partições de disco atribuídas à raiz do sistema de arquivos (`/`), o diretório `/boot` e o diretório `/home`. Todas as três são sistemas de arquivo `ext4`. A quarta linha é um dispositivo de troca (usado para armazenar dados quando a RAM se esgota). Observe que os nomes de dispositivos para `/`, `/home` e `swap` começam, todos, com `/dev/mapper`. Isso ocorre porque eles são volumes lógicos LVM que recebem espaço de um conjunto de espaço chamado grupo LVM (mais sobre LVM na seção “Usando partições LVM”, mais adiante neste capítulo).

A partição `/boot` está em sua própria partição física, `/dev/sda1`. Em vez de usar `/dev/sda1`, porém, um identificador exclusivo (UUID) identifica o dispositivo. Por que usar um UUID em vez de `/dev/sda1` para identificar o dispositivo? Digamos que você conectou um outro disco em seu computador e inicializou. Provavelmente, não vai acontecer, mas é possível que o novo disco seja identificado como `/dev/sda`, fazendo com que o sistema procure o conteúdo de `/boot` na primeira partição desse disco.

Para ver todas as UUIDs atribuídas a dispositivos de armazenamento em seu sistema, digite o comando `blkid`, como segue:

blkid

```
/dev/sda1:  
  UUID="78bdae46-9389-438d-bfee-06dd934fae28"  TYPE="ext4"  
/dev/sda2:  
  UUID="wlvuIv-UiI2-pNND-f39j-oH0X-9too-AOII7R"  TYPE="LVM2_member"  
/dev/mapper/vg_abc-lv_root:  
  UUID="3e6f49a6-8fec-45e1-90a9-38431284b689"  TYPE="ext4"  
/dev/mapper/vg_abc-lv_swap:  
  UUID="77662950-2cc2-4bd9-a860-34669535619d"  TYPE="swap"  
/dev/mapper/vg_abc-lv_home:  
  UUID="7ffbcff3-36b9-4cbb-871d-091efb179790"  TYPE="ext4"  
/dev/sdb1:  
  SEC_TYPE="msdos"  UUID="75E0-96AA"  TYPE="vfat"
```

Qualquer um dos nomes de dispositivo pode ser substituído pela designação UUID na coluna esquerda de uma entrada `/etc/fstab`.

Eu adicionei as próximas três entradas em `/etc/fstab` para ilustrar alguns diferentes tipos de entradas. Conectei um disco rígido de um velho sistema Microsoft Windows e o montei no diretório `/win`. Adicionei a opção `ro` de modo a montá-lo como somente leitura.

As duas entradas seguintes representam sistemas de arquivos remotos. No diretório `/remote`, o diretório `/nfsstuff` é de leitura/gravação (`rw`), a partir do host no endereço 192.168.0.27 como um compartilhamento NFS. No diretório `/share`, a participação do Windows chamada `myshare` é montada a partir do host em 192.168.0.28. Em ambos os casos, adicionei a opção `_netdev`, que diz para o Linux esperar a rede aparecer antes de tentar montar os compartilhamentos. (Para mais informações sobre a montagem e compartilhamentos CIFS, NFS, consulte os capítulos 19, “Configurando um servidor de compartilhamento de arquivos Windows (Samba)”, e 20, “Configurando um servidor de arquivos NFS”, respectivamente.)

ndo do Windows

seção “Utilizando o arquivo `fstab` para definir sistemas de arquivos montáveis” mostra a montagem de uma partição do disco rígido a partir de um velho sistema de arquivos VFAT usado no Windows. A maioria dos sistemas Windows hoje usa o sistema de arquivos NTFS. O suporte para esse sistema, contudo, não é fornecido por todos os sistemas Linux. O suporte a NTFS é disponibilizado pelo Fedora no pacote `ntfs-3g`. Outro suporte a NTFS é disponibilizado pelo projeto Linux-NTFS (<http://www.linux-ntfs.org/>).

Para ajudar você a entender o conteúdo do arquivo `/etc/fstab`, eis o que há em cada campo desse

arquivo:

- **Campo 1** — O nome do dispositivo que representa o sistema de arquivos. Esse campo pode incluir a opção `LABEL` ou `UUID`, com a qual você pode indicar um rótulo de volume ou identificador universalmente único (UUID) em vez de um nome de dispositivo. A vantagem dessa abordagem é que como a partição é identificada pelo nome de volume, você pode mover um volume para um nome de dispositivo diferente e não ter de alterar o arquivo `fstab`. (Veja a descrição do comando `mkfs` mais adiante na seção “Usando o comando `mkfs` para criar um sistema de arquivos”, deste capítulo, para obter informações sobre como criar e utilizar rótulos.)
- **Campo 2** — O ponto de montagem no sistema de arquivos. O sistema de arquivos contém todos os dados desde o ponto de montagem para baixo na estrutura de árvore de diretórios, a menos que outro sistema de arquivos esteja montado em algum ponto abaixo dele.
- **Campo 3** — O tipo de sistema de arquivos. Tipos de sistemas de arquivos válidos são descritos na seção “Sistema de arquivos suportados”, anteriormente neste capítulo (embora você só possa usar os tipos de sistema de arquivos para os quais drivers de kernel estão inclusos).
- **Campo 4** — Use `defaults` ou uma lista de opções (sem espaços) separadas por vírgulas que você deseja usar quando a entrada for montada. Veja a página `man` do comando `mount` (sob a opção `-o` para obter informações sobre outras opções de suporte).

ca

nalmente, somente o usuário `root` tem permissão para montar um sistema de arquivos usando o comando `mount`. Mas para permitir que qualquer usuário monte um sistema de arquivos (como um sistema de arquivos em CD), você pode adicionar a opção `user` ao campo 4 do arquivo `/etc/fstab`.

- **Campo 5** — O número nesse campo indica se é necessário fazer um `dump` (uma cópia) do sistema de arquivos. `1` significa que é necessário fazer um `dump` do sistema de arquivos; e `0` que não é necessário. (Esse campo não é mais particularmente útil porque a maioria dos administradores Linux usa as opções de backup mais sofisticadas que o comando `dump`. Na maioria das vezes, um `0` é usado.)
- **Campo 6** — O número nesse campo indica se o sistema de arquivos indicado deve ser verificado com `fsck` quando chegar a hora de ele ser verificado: `1` significa que ele precisa ser verificado primeiro; `2` significa verificar depois que todos aqueles indicados por `1` foram verificados; e `0` significa não verificá-lo.

Se você quiser saber mais sobre as opções de montagem, bem como outros recursos do arquivo `/etc/fstab`, há páginas `man` que pode consultar, incluindo `man 5 nfs` e `man 8 mount`.

Utilizando o comando `mount` para montar sistemas de arquivos

Sistemas Linux executam automaticamente `mount -a` (montar todos os sistemas de arquivos) sempre

que você inicializa. Por essa razão, você geralmente usa o comando `mount` apenas para situações especiais. Em particular, o usuário médio ou o administrador utiliza `mount` de duas maneiras:

- Para exibir os discos, partições e sistemas de arquivos remotos atualmente montados
- Para montar temporariamente um sistema de arquivos

Qualquer usuário pode digitar o comando `mount` (sem opções) para ver quais sistemas de arquivos estão atualmente montados no sistema Linux local. Eis um exemplo do comando `mount`. Ele mostra uma única partição de disco rígido (`/dev/sda1`) contendo a raiz (`/`) do sistema de arquivos e os tipos de sistemas de arquivos `proc` e `devpts` montados em `/proc` e `/dev`, respectivamente.

```
$ mount
/dev/sda3 on / type ext4 (rw)
/dev/sda2 on /boot type ext4 (rw)
/dev/sda1 on /mnt/win type vfat (rw)
/dev/proc on /proc type proc (rw)
/dev/sys on /sys type sysfs (rw)
/dev/devpts on /dev/pts type devpts (rw,gid=5,mode=620)
/dev/shm on /dev/shm type tmpfs (rw)
none on /proc/sys/fs/binfmt_misc type binfmt_misc (rw)
/dev/cdrom on /media/MyOwnDVD type iso9660 (ro,nosuid,nodev)
```

Tradicionalmente, os dispositivos mais comuns para montar manualmente são mídias removíveis, como DVDs ou CDs. Mas dependendo do tipo de desktop que você está usando, CDs e DVDs podem ser montados automaticamente quando inseridos. (Em alguns casos, aplicativos específicos também são carregados quando a mídia é inserida. Por exemplo, um leitor de música de CD ou um editor de fotos pode ser iniciado quando a mídia inserida contém música ou imagens digitais.)

Ocasionalmente, porém, você pode achar que é útil montar um sistema de arquivos manualmente. Por exemplo, você quer examinar o conteúdo de um disco rígido antigo para instalá-lo como um segundo disco em seu computador. Se as partições do disco não forem automaticamente montadas, você pode montar partições a partir desse disco manualmente. Por exemplo, para montar no modo somente leitura a partir de uma partição de disco `sdb1` que tem um antigo sistema de arquivos `ext3`, você pode digitar o seguinte:

```
# mkdir /mnt/temp
# mount -t ext3 -o ro /dev/sdb1 /mnt/tmp
```

Outra razão para usar o comando `mount` é remontar uma partição para mudar suas opções de montagem. Digamos que você quer remontar `/dev/sdb1` como leitura/gravação, mas não quer desmontá-lo (talvez alguém o esteja usando). Você pode usar a seguinte opção do comando `remount`:

```
# mount -t ext3 -o remount,rw /dev/sdb1
```

Montando uma imagem de disco em loopback

Outra forma valiosa para usar o comando `mount` tem a ver com imagens de disco. Se você baixar uma imagem de CD ou disco flexível a partir da internet e quiser ver o que ela contém, pode fazer isso sem queimar em CD ou disquete. Com a imagem em seu disco rígido, crie um ponto de montagem e use a opção `-o loop` para montá-lo localmente. Eis um exemplo:

```
# mkdir /mnt/mycdimage
# mount -o loop whatever-i686-disc1.iso /mnt/mycdimage
```

Nesse exemplo, o diretório `/mnt/mycdimage` é criado e, então, o arquivo de imagem de disco (`whatever-i686-disc1.iso`) residente no diretório atual é montado nele. Agora, você pode usar o comando `cd` para mudar para esse diretório, ver o conteúdo do mesmo e copiar ou usar qualquer dos seus conteúdos. Isso é útil para imagens de CD baixadas a partir das quais você deseja instalar o software sem ter de gravar a imagem em CD. Você também pode compartilhar o ponto de montagem via NFS, assim pode instalar o software a partir de outro computador. Quando você terminar, basta digitar `umount /mnt/mycdimage` para desmontá-lo.

Outras opções de `mount` estão disponíveis apenas para tipos de sistemas de arquivos específicos. Consulte o manual de `mount` para essas e outras opções úteis.

Usando o comando `umount`

Quando você terminar de usar um sistema de arquivos temporário ou quiser desmontar um sistema de arquivos temporariamente, use o comando `umount`. Esse comando desconecta o sistema de arquivos a partir do ponto de montagem dele em seu sistema de arquivos Linux. Para usar `umount`, você pode atribuir-lhe um nome de diretório ou um nome de dispositivo. Por exemplo:

```
# umount /mnt/test
```

Isso desmonta o dispositivo a partir do ponto de montagem `/mnt/test`. Você também pode desmontá-lo utilizando a forma

```
# umount /dev/sdb1
```

Em geral, é melhor usar o nome do diretório (`/mnt/test`), porque o comando `umount` falhará se o dispositivo estiver montado em mais de um local. (Todos os nomes de dispositivo começam com `/dev`).

Se você receber a mensagem `device is busy`, a solicitação `umount` falhou ou porque um aplicativo tem um arquivo aberto no dispositivo ou você tem um shell aberto com um diretório no dispositivo como um diretório atual. Pare os processos ou mude para um diretório fora do dispositivo que você está tentando desmontar para a solicitação `umount` funcionar.

Uma alternativa para desmontar um dispositivo ocupado é a opção `-l`. Com `umount -l` (uma desmontagem lenta), a desmontagem ocorre logo que o dispositivo não mais estiver ocupado. Para

desmontar um sistema de arquivos NFS remoto que não está mais disponível (por exemplo, o servidor caiu), você pode usar a opção `umount -f` para forçar a desmontar o sistema de arquivos NFS.

ca

Uma ferramenta muito útil para descobrir o que está mantendo aberto um dispositivo que você deseja desmontar é o comando `lsof`. Digite `lsof` com o nome da partição que você deseja desmontar (como `lsof /mnt/test`). A saída mostra os comandos que estão segurando arquivos abertos nessa partição. O comando `fuser-v /mnt/test` pode ser usado da mesma maneira.

Usando o comando `mkfs` para criar um sistema de arquivos

Você pode criar um sistema de arquivos para qualquer tipo de sistema de arquivos suportado em um disco ou partição que você escolher. Você pode fazer isso com o comando `mkfs`. Embora isso seja mais útil para a criação de sistemas de arquivos em partições de disco rígido, você também pode criar sistemas de arquivos em pen drives USB, disquetes ou CDs regraváveis.

Antes de criar um novo sistema de arquivos, certifique-se de que:

- Você particionou o disco como você queria (usando o comando `fdisk`).
- Você sabe o nome correto do dispositivo ou pode apagar seu disco rígido por engano. Por exemplo, a primeira partição na segunda SCSI ou pen drive USB em seu sistema é `/dev/sdb1` e o terceiro disco é `/dev/sdc1`.
- Desmontar a partição, se ela estiver montada antes de criar o sistema de arquivos.

O seguinte é um exemplo do uso `mkfs` para criar um sistema de arquivos na primeira (e única) partição, em uma pen drive USB de 2GB localizado como terceiro disco SCSI (`/dev/sdc1`):

```
# mkfs -t ext3 /dev/sdc1
mke2fs 1.40.8 (13-Mar-2008)
Warning: 256-byte inodes not usable on older systems
Filesystem label=
OS type: Linux
Block size=4096 (log=2)
Fragment size=4096 (log=2)
122160 inodes, 487699 blocks
24384 blocks (5.00%) reserved for the super user
First data block=0
Maximum filesystem blocks=503316480
15 block groups
32768 blocks per group, 32768 fragments per group
8144 inodes per group
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912
Writing inode tables: done
```

```
Creating journal (8192 blocks): done
Writing superblocks and filesystem accounting information: done
This filesystem will be automatically checked every 39 mounts or
180 days, whichever comes first. Use tune2fs -c or -i to override.
```

Você pode ver as estatísticas enviadas para a saída com a formatação feita pelo comando `mkfs`. O número de inodes e blocos criados são enviados para a saída, assim como o número de blocos em cada grupo e os fragmentos por grupo. Um inode, que armazena metadados, como propriedade e registro de data/hora para cada arquivo, será consumido por todos os arquivos e diretórios no sistema de arquivos. Portanto, o número de inodes mostrados aqui limita o número total de arquivos que você pode criar no sistema de arquivos.

Agora você pode montar esse sistema de arquivos (`mkdir /mnt/myusb ; mount /dev/sdc1 /mnt/myusb`), mudar para `/mnt/usb` como seu diretório atual (`cd /mnt/myusb`) e criar arquivos nele como quiser.

Resumo

Gerenciar sistemas de arquivos é uma parte crítica da administração de um sistema Linux. Utilizando comandos como `fdisk`, você pode visualizar e alterar partições de disco. Sistemas de arquivos podem ser adicionados a partições usando o comando `mkfs`. Uma vez criados, os sistemas de arquivos podem ser montados e desmontados usando os comandos `mount` e `umount`, respectivamente.

Logical Volume Management (LVM) oferece uma maneira mais poderosa e flexível de gerenciar partições de disco. Com o LVM, você pode criar conjuntos de armazenamento, chamados volumes, que permitem aumentar e diminuir os volumes lógicos, bem como aumentar o tamanho de seus grupos de volumes, adicionando mais volumes físicos.

Com a maioria dos fundamentos necessários para se tornar um administrador de sistema abordados nesse ponto do livro, o Capítulo 13 introduz conceitos para estender essas habilidades a fim de gerenciar servidores de rede. Tópicos nesse capítulo incluem informações sobre como instalar, gerenciar e tornar seguros os servidores.

Exercícios

Use esses exercícios para testar seus conhecimentos sobre como criar partições de disco, gerenciar volumes lógicos e trabalhar com sistemas de arquivos. Você vai precisar de um pen drive USB que tenha pelo menos 1GB e que possa ser apagado para esses exercícios.

Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostrados no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. Execute um comando como root para ver o arquivo `/var/log/messages` e insira seu pen drive USB. Determine o nome do dispositivo do pen drive USB.

2. Execute um comando para listar a tabela de partições para o pen drive USB.
3. Apague todas as partições de seu pen drive USB, salve as alterações e certifique-se de que as alterações foram feitas tanto na tabela de partições do disco como no kernel do Linux.
4. Adicione três partições ao pen drive USB: uma partição Linux de 100MB, uma partição troca de 200MB e uma partição LVM de 500MB. Salve as alterações.
5. Coloque um sistema de arquivos `ext3` na partição Linux.
6. Crie um ponto de montagem chamado `/mnt/mypart` e monte a partição Linux nele.
7. Habilite a partição de troca e ative-a de modo que não haja espaço de troca adicional imediatamente disponível.
8. Crie um grupo de volumes chamado `abc` a partir da partição LVM, crie um volume lógico de 200MB a partir desse grupo chamado `data`, adicione uma partição VFAT e depois monte temporariamente o volume lógico em um novo diretório chamado `/mnt/test`. Verifique se ele foi montado com sucesso.
9. Aumente o volume lógico de 200MB para 300MB.
10. Faça o que é necessário para remover com segurança o pen drive USB do computador: desmonte a partição Linux, desative a partição troca, desmonte o volume lógico e exclua o grupo de volumes do pen drive USB.

Parte IV

Tornando-se um administrador de servidor Linux

NESTA PARTE

Capítulo 13

Entendendo administração de servidores

Capítulo 14

Administrando redes

Capítulo 15

Iniciando e parando serviços

Capítulo 16

Configurando um servidor de impressão

Capítulo 17

Configurando um servidor web

Capítulo 18

Configurando um servidor FTP

Capítulo 19

Configurando um servidor de compartilhamento de arquivos do Windows (Samba)

Capítulo 20

Configurando um servidor de arquivos NFS

Capítulo 21

Solução de problemas do Linux

Entendendo administração de servidores

Neste capítulo

Administrando servidores Linux

Comunicando-se com os servidores através de redes

Configurando o registro em log local e remotamente

Monitorando sistemas de servidores

Embara algumas tarefas de administração do sistema sejam necessárias mesmo em um sistema desktop (instalar software, configurar impressoras etc.) muitas novas tarefas aparecem quando você configura um sistema Linux para funcionar como um servidor. Isso é especialmente verdadeiro se o servidor que você configura é tornado público para qualquer pessoa na internet, caso em que você pode ficar sobrecarregado com pedidos de caras legais, enquanto precisa estar constantemente em guarda contra ataques dos caras maus.

Dezenas de diferentes tipos de servidores estão disponíveis para sistemas Linux. A maioria deles servem dados para clientes remotos, mas outros servem o sistema local (como aqueles que coletam mensagens de log ou disparam tarefas de manutenção em horários agendados utilizando o recurso `crontab`). Muitos servidores são representados por processos que são executados continuamente em segundo plano e respondem às solicitações que vêm a eles. Esses processos são referidos como processos *daemon*.

Como o nome indica, servidores existem para servir. Os dados que eles servem podem incluir páginas web, arquivos, informações de banco de dados, e-mail e muitos outros tipos de conteúdo. Como um administrador de servidor, alguns dos desafios adicionais para suas habilidades de administração de sistema incluem os seguintes:

- **Acesso remoto** — Para usar um sistema desktop, você costuma sentar-se à frente do seu console. Sistemas de servidor, por outro lado, tendem a ser alojados em racks em ambientes climatizados e fechados a sete chaves. Na maioria dos casos, depois que os computadores físicos estão no lugar, a maior parte da administração dessas máquinas é feita utilizando ferramentas de acesso remoto. Muitas vezes, não há uma interface gráfica disponível, de modo que você deve confiar em ferramentas de linha de comando para fazer coisas como login remoto, cópia remota e execução remota. As mais comuns dessas ferramentas são construídas com base no recurso Secure Shell (SSH).

- **Segurança diligente**— Para ser útil, um servidor precisa ser capaz de aceitar solicitações de conteúdo de usuários e sistemas remotos. Ao contrário dos sistemas desktop, que podem simplesmente fechar todas as portas de rede que permitem a entrada de solicitações de acesso, o servidor deve tornar-se vulnerável por permitir algum acesso a suas portas. É por isso que é importante, como um administrador de servidor, abrir portas para serviços que são necessários e bloquear as portas que não são necessárias. Você pode proteger serviços utilizando ferramentas como o `iptables` (firewall), TCP wrappers (para permitir e negar acesso a serviços) e Security Enhanced Linux (para limitar os recursos que um serviço pode acessar a partir do sistema local).
- **Monitoramento contínuo** — Embora você normalmente desligue um sistema laptop ou desktop quando não o está usando, servidores costumam ficar ligados dia e noite, 365 dias por ano. Como você não quer sentar-se à frente de cada servidor e continuamente monitorá-lo pessoalmente, você pode configurar as ferramentas para monitorar cada servidor, coletar mensagens de log e até encaminhar mensagens suspeitas para uma conta de e-mail de sua escolha. Você pode ativar repórteres de atividades do sistema para coletar dados continuamente sobre o uso de CPU, uso de memória, atividade de rede e acesso a disco.

Neste capítulo, explico algumas das ferramentas e técnicas básicas que você precisa conhecer para administrar servidores remotos Linux. Você aprenderá a usar ferramentas SSH para acessar seu servidor de forma segura, transferir dados de um sistema para outro e até mesmo carregar desktops remotos ou aplicativos gráficos e fazê-los aparecer em seu sistema local. Você aprenderá a usar o registro em log remoto e relatórios de atividade do sistema para monitorar continuamente as atividades do sistema.

Começando com administração do servidor

Quer você esteja instalando um servidor de arquivos, um servidor web ou qualquer um dos outros recursos de servidor disponíveis com sistemas Linux, muitos dos passos necessários para fazer o servidor funcionar são os mesmos. Onde a configuração do servidor diverge é nas áreas de configuração e ajuste. Nos próximos capítulos, descrevo servidores específicos e como eles diferem. Em cada um dos capítulos relacionados a servidor que se seguem a este capítulo, você vai percorrer os mesmos passos básicos para fazer o servidor iniciar e disponibilizá-lo para uso por seus clientes:

Passo 1: Instale o servidor

Embora a maior parte do software de servidor não venha pré-instalada no sistema Linux típico, qualquer sistema Linux de propósito geral vai oferecer os pacotes de software necessários para fornecer os principais tipos de servidores disponíveis.

Às vezes, vários pacotes de software associados com um determinado tipo de servidor são reunidos em Package Groups (às vezes chamados de Package Collections). Outras vezes, você só precisa instalar os pacotes do servidor que você quer individualmente. Eis algumas categorias de pacotes de servidores no Fedora e alguns dos pacotes disponíveis em cada categoria:

- **Servidor de registro em log do sistema** — O serviço `rsyslog` permite ao sistema local

recolher as mensagens de log entregues a partir de uma variedade de componentes no sistema. Ele também pode atuar como um servidor de log remoto, coletando mensagens de log enviadas de outros servidores de log. (O serviço `rsyslog` é descrito mais adiante, neste capítulo.)

- **Servidor de impressão** — O Common UNIX Printing Service (pacote `cups`) é usado mais frequentemente para fornecer recursos de servidor de impressão em sistemas Linux. Pacotes que fornecem administração gráfica do CUPS (`system-config-printer`) e drivers da impressora (`foomatic`, `hpijs` e outros) também estão disponíveis quando você instala o CUPS. (Veja o Capítulo 16, “Configurando um servidor de impressão”.)
- **Servidor Web** — O servidor web Apache (pacote `httpd`) é o software que é usado na maioria das vezes para servir páginas web (conteúdo HTTP). Pacotes relacionados incluem módulos para ajudar a servir determinados tipos de conteúdo (Perl, Python, PHP e conexões SSL). Da mesma maneira, há pacotes de documentação (`httpd-manual`), ferramentas para monitorar dados da web (`webalizer`) e ferramentas para o fornecimento de serviços web proxy (`squid`). (Veja o Capítulo 17, “Configurando um servidor web”.)
- **FTP Server** — The Very Secure FTP Daemon (pacote `vsftpd`) é o servidor FTP padrão usado no Fedora e no RHEL. Outros pacotes de servidor de FTP são o `proftpd` e o `pure-ftpd`. (Veja o Capítulo 18, “Configurando um servidor FTP”.)
- **Windows File Server** — Samba (pacote `samba`) permite que um sistema Linux funcione como um servidor de arquivos e impressão do Windows. (Veja o Capítulo 19, “Configurando um servidor de compartilhamento de arquivos Windows (Samba)”.)
- **NFS** — Network File System (NFS) é o recurso de Linux e UNIX padrão para fornecer diretórios compartilhados para outros sistemas em uma rede. O pacote `nfs-utils` fornece serviços NFS e comandos relacionados. (Veja o Capítulo 20, “Configurando um servidor de arquivos NFS”.)
- **Servidor de e-mail** — Esses tipos de pacotes permitem configurar servidores de e-mail, às vezes referidos como servidor Mail Transport Agent (MTA). Você tem várias opções de servidores de e-mail, incluindo `sendmail` (padrão no Fedora), `postfix` (padrão no RHEL) e `exim`. Pacotes relacionados, como `dovecot`, permitem que o servidor de e-mail entregue e-mail para clientes.
- **Servidor de diretório** — Pacotes nesta categoria fornecem serviços de autenticação remota e local. Esses incluem Kerberos (`krb5-server`), LDAP (`openldap-servers`) e NIS (`ypserv`).
- **Servidor DNS** — O serviço Berkeley Internet Name Domain (`bind`) fornece o software necessário para configurar um servidor para converter nomes de hosts em endereços IP.
- **Servidor NTP (Network Time Protocol)** — O pacote `ntpd` fornece um serviço que você pode ativar para sincronizar o relógio do sistema com relógios de servidores NTP públicos ou privados.
- **SQL Server** — O serviço PostgreSQL (`postgresql-server` e `PostgreSQL-servidor`) é um sistema objeto-relacional de gerenciamento de banco de dados. Pacotes relacionados

fornecem a documentação e ferramentas relacionadas do PostgreSQL. O serviço MySQL (pacotes `mysql` e `mysql-server`) é outro popular servidor de banco de dados SQL de código-fonte aberto.

Passo 2: Configure o servidor

A maioria dos pacotes de software de servidor é instalada com uma configuração padrão que tende mais para a segurança do que à plena utilização imediata. Eis algumas coisas a se pensar quando você começa a configurar um servidor.

Usando arquivos de configuração

A maioria dos servidores Linux é configurada por meio de arquivos de texto no diretório `/etc` (ou subdiretórios). Muitas vezes, há um arquivo de configuração principal e, às vezes, há um diretório de configuração relacionado em que os arquivos que terminam em `.conf` podem ser copiados para o arquivo de configuração principal.

O pacote `httpd` (Apache web server) é um exemplo de um pacote de servidor que tem um arquivo de configuração principal e um diretório onde outros arquivos de configuração podem ser colocados e incluídos no serviço. O principal arquivo de configuração no Fedora e no RHEL é o `/etc/httpd/conf/httpd.conf`. O diretório de configuração é `/etc/httpd/conf.d`.

Depois de instalar pacotes `httpd` e afins, você vai ver os arquivos no diretório `/etc/httpd/conf.d`, que foram colocados lá por pacotes diferentes: `mod_ssl`, `mod_perl` e assim por diante. Essa é uma maneira como pacotes suplementares de um serviço podem ter suas informações de configuração habilitadas no servidor `httpd`, sem que o pacote precise tentar executar um script para editar o arquivo `httpd.conf` principal.

A única desvantagem para arquivos de configuração em texto simples é que você não tem o tipo de verificação de erro imediato que tem quando usa ferramentas de administração gráfica. Você tem de executar um comando de teste (se o serviço incluir um) ou realmente tentar iniciar o serviço para ver se há algum problema com seu arquivo de configuração.

ca

vez de usar o `vi` para editar arquivos de configuração, utilize o `vim`. Usar o comando `vim` pode ajudá-lo a evitar erros ao editar o arquivo de configuração.

mando `vim` conhece os formatos de muitos arquivos de configuração (`passwd`, `httpd.conf`, `fstab` e `etc`). Se você cometer um erro e digitar um termo ou opção inválidos em um desses arquivos ou quebrar o formato alguma forma, a cor do texto será alterada. Por exemplo, em `/etc/fstab`, se você alterar a opção `defaults` para `default`, a cor da palavra muda de verde para preto.

Verificando a configuração padrão

A maioria dos pacotes de software de servidor no Fedora e no RHEL é instalada com uma configuração mínima e tende a privilegiar a segurança mais do que a plena funcionalidade. Algumas distribuições

Linux perguntam-lhe, ao instalar um pacote de software, coisas como o diretório em que você deseja instalá-lo ou a conta de usuário que você deseja para gerenciá-lo.

Como os pacotes RPM são projetados para serem instalados sem supervisão, a pessoa que instala o pacote não tem escolha de como ele é instalado. Os arquivos são instalados em locais fixos, contas de usuários específicas são habilitadas para gerenciá-los e quando você inicia o serviço, este poderia muito bem oferecer acessibilidade limitada. Espera-se que você configure o software após a instalação do pacote para tornar o servidor totalmente funcional.

Dois exemplos de servidores que são instalados com funcionalidades limitadas são os servidores de email (pacotes `sendmail` ou `postfix`) e os servidores DNS (pacotes `bind`). Ambos são instalados com configurações padrão e iniciam na reinicialização do sistema. Mas ambos também só atendem solicitações de seu `localhost`. Portanto, até você configurar esses servidores, as pessoas que não estão conectadas ao seu servidor local e não podem enviar e-mail para o servidor ou usar o computador como um servidor de DNS público.

Passo 3: Inicie o servidor

A maioria dos serviços que você instala no Linux é configurada para iniciar durante a inicialização do sistema e, então, executar continuamente, atendendo as solicitações pelo seu serviço, até que o sistema seja desligado. Há dois principais recursos para o gerenciamento de serviços: `systemd` (usado agora pelo Ubuntu e o Fedora) e scripts de inicialização System V (usados pelo Red Hat Enterprise Linux, ao menos até o RHEL 6.x).

Independentemente de qual mecanismo é usado no sistema Linux, é seu trabalho fazer coisas como definir se deseja que o serviço inicie na inicialização do sistema e iniciar, parar e recarregar o serviço conforme necessário (possivelmente para carregar novos arquivos de configuração ou interromper temporariamente o acesso ao serviço). Comandos para fazer essas tarefas são descritos no Capítulo 15, “Iniciando e parando serviços”.

A maioria dos serviços, mas não todos os, é implementada como processos `daemon`. Eis algumas coisas que você deve saber sobre esses processos:

- **Permissões de usuários e de grupo** — Processos `daemon` costumam rodar como usuários e outros grupos que não sejam `root`. Por exemplo, `httpd` é executado como `apache` e `ntpd` é executado como o usuário `ntp`. A razão disso é que, se alguém invadir esses servidores, ele não terá permissões para acessar arquivos além do que o serviço pode acessar.
- **Arquivos de configuração do `daemon`** — Muitas vezes, um serviço terá um arquivo de configuração para o `daemon` armazenado no diretório `/etc/sysconfig`. Esses arquivos são diferentes do arquivo de configuração, no sentido de que seu trabalho costuma ser apenas passar argumentos para o processo de servidor em si, em vez de configurar o serviço. Por exemplo, as opções definidas no arquivo `/etc/sysconfig/rsyslogd` são passadas para o `daemon rsyslogd` quando ele inicia. Então, você pode dizer para o `daemon`, por exemplo, enviar informações de depuração adicionais para a saída ou aceitar mensagens de log remotas. Veja a página `man` para o serviço (por exemplo, `man rsyslogd`) para ver quais opções são suportadas.

- **Números de porta** — Os pacotes de dados entram e saem do seu sistema através de interfaces de rede e portas para cada protocolo (UDP, TCP, ICMP etc.). A maioria dos serviços padrão tem números de porta específicos que os daemons ouvem e às quais os clientes se conectam. A menos que esteja tentando esconder a localização de um serviço, você normalmente não altera as portas que um processo daemon ouve. Ao configurar a segurança de um serviço, você deve certificar-se de que a porta para ele está aberta no firewall (ver Capítulo 25, “Protegendo o Linux em uma rede”, para obter informações sobre firewalls `iptables`). Além disso, se você mudar a porta que o serviço está ouvindo e o SELinux estiver no modo Enforcing, o SELinux pode impedir que o daemon ouça nessa porta (ver Capítulo 24, “Aumentando a segurança do Linux com o SELinux”, para mais informações sobre o SELinux).

ta

A razão para alterar os números de porta em um serviço é “segurança pela obscuridade”. Por exemplo, o serviço `xinetd` é um alvo bem conhecido para quem está tentando invadir um sistema adivinhando logins e senhas na porta 22.

Eu vi falar de pessoas que mudam seu serviço `sshd` voltado para internet para escutar em algum outro número de porta (talvez algum não utilizado ou um número de porta muito alto). Então, eles pedem para seus amigos ou colegas fazerem login em sua máquina a partir do `ssh`, apontando para essa outra porta. A ideia é que scanners de porta procurando invadir o sistema tendem a não verificar portas normalmente não utilizadas.

Nem todos os serviços são executados continuamente como processos daemon. Alguns serviços são executados sob demanda usando o superservidor `xinetd`. Outros serviços apenas são executados uma vez na inicialização e, então, fecham. Outros ainda são executados apenas um número definido de vezes, sendo carregados quando o daemon `crond` vê que o serviço foi configurado para ser executado na data e hora especificadas.

Dos serviços já mencionados, os sob demanda são a principal forma de executar serviços sempre disponíveis (se não sempre em execução). Serviços sob demanda não funcionam continuamente ouvindo solicitações. Em vez disso, seus serviços são registrados no daemon `xinetd`. Quando o daemon `xinetd` recebe solicitações de um serviço, ele carrega-o e serve-o ao cliente.

A vantagem do daemon `xinetd` é que você pode ter menos processos daemon executando e consumindo memória e slots de processamento. O superservidor `xinetd` (originalmente chamado `inetd`, quando foi criado nos primeiros dias do UNIX) surgiu em um momento em que a memória era muito cara e, portanto, liberar espaço carregando os serviços raramente utilizados apenas sob demanda fazia sentido. Como a quantidade de memória consumida por um processo daemon não é mais uma grande coisa, você pode observar que a maioria dos serviços `xinetd` são mais antigos (como `telnet` e `tftp`).

Para mais informações sobre a inicialização de serviços como serviços regulares ou sob demanda (`xinetd`), consulte o Capítulo 15, “Iniciando e parando serviços”.

Passo 4: Proteja o servidor

Abrir o sistema para permitir que usuários remotos o acessem através da rede não é uma decisão que você deve tomar sem as devidas considerações. Há crackers por todo o mundo rodando programas para procurar servidores vulneráveis que eles possam invadir para roubar dados ou capacidade de processamento. Felizmente, há medidas que você pode adotar em sistemas Linux para proteger seus servidores e serviços contra ataques e abusos.

Algumas técnicas de segurança comuns são descritas nas próximas seções. Esses e outros temas são abordados com mais profundidade na parte V, “Aprendendo técnicas de segurança do Linux”.

Proteção por senha

Boas senhas e políticas de senha são a primeira linha de defesa para proteger um sistema Linux. Se alguém pode fazer logon em seu servidor via ssh como o usuário root com a senha foobar, espere ser invadido. Uma boa técnica é não permitir login direto por root e exigir que cada usuário faça login como um usuário comum e depois usar `su` ou `sudo` para se tornar root.

Você também pode usar o recurso Pluggable Authentication Module (PAM) para ajustar o número de vezes que alguém pode errar nas tentativas de login antes de bloquear o acesso a essa pessoa. O PAM também inclui outros recursos para bloquear a autenticação em seu servidor Linux. Para uma descrição do PAM, consulte o Capítulo 23, “Noções básicas de segurança avançada em Linux”.

Firewalls

O serviço de firewall `iptables` pode acompanhar e responder a todos os pacotes indo e vindo das placas de rede em seu computador. Usando `iptables`, você pode descartar ou rejeitar todos os pacotes fazendo solicitações de serviços em seu sistema, exceto para aqueles poucos que você habilitou. Além disso, você pode dizer para o `iptables` permitir solicitações de serviço apenas de determinados endereços IP (os caras bons) ou não permitir solicitações de outros endereços (os caras maus).

Em cada um dos próximos capítulos sobre servidor, descrevo as portas que devem ser abertas para permitir o acesso aos serviços. Descrições de como funciona o `iptables` estão inclusas no Capítulo 25, “Protegendo o Linux em uma rede”.

TCP Wrappers

Usando os arquivos `/etc/hosts.allow` e `/etc/hosts.deny`, você pode permitir ou negar o acesso a esses serviços que têm os recursos de TCP Wrappers habilitados (`libwrap`). O acesso pode ser permitido ou negado com base no endereço IP ou nome do host. Descrições de TCP Wrappers estão contidas no Capítulo 25.

SELinux

Fedora, Red Hat Enterprise Linux e outras distribuições Linux vêm com o recurso Security Enhanced Linux (SELinux) incluso e em modo Enforcing. Embora o modo direcionado padrão não tenha muito impacto sobre a maioria dos aplicativos que você roda no Linux, ele tem um grande impacto sobre a maioria dos principais serviços.

Uma das principais funções do SELinux é proteger o conteúdo de seu sistema Linux contra processos em execução no sistema. Em outras palavras, o SELinux garante que um servidor web, FTP, Samba ou DNS possa acessar apenas um conjunto restrito de arquivos no sistema (conforme definido por contextos de

arquivos) e permitir apenas um conjunto restrito de recursos (conforme definido por opções booleanas).

Detalhes sobre como usar o SELinux estão contidos no Capítulo 24, “Aumentando a segurança do Linux com o SELinux”.

Configurações de segurança em arquivos de configuração

Dentro dos arquivos de configuração da maioria dos serviços estão valores que podem ser configurados para proteger ainda mais esses serviços. Por exemplo, para servidores de arquivos e servidores web, você pode restringir o acesso a determinados arquivos ou dados com base no nome de usuário, nome de host, endereço IP do cliente ou outros atributos.

Passo 5: Monitore o servidor

Como você não pode estar lá para monitorar cada serviço a cada minuto, você tem de colocar as ferramentas de monitoramento no lugar para monitorar seus servidores por você; assim se torna mais fácil descobrir quando algo precisa de atenção. Algumas das ferramentas que você pode usar para monitorar seus servidores estão descritas nas seções que se seguem.

Configure o registro em log

Usando o serviço `rsyslog` (daemon `rsyslogd`), você pode coletar informações cruciais e condições de erro em arquivos de log sobre muitos serviços diferentes. Por padrão, mensagens de log dos aplicativos são direcionadas para arquivos de log no diretório `/var/log`. Para maior segurança e comodidade, as mensagens de log também podem ser direcionadas para um servidor centralizado, oferecendo um local centralizado para visualizar e gerenciar o registro em log.

Vários pacotes de software diferentes estão disponíveis para trabalhar com `rsyslog` e gerenciar mensagens de log. O recurso `logwatch` varre seus arquivos de log a cada noite e envia informações críticas recolhidas a partir desses arquivos para uma conta de e-mail de sua escolha. O recurso `logrotate` faz o backup de arquivos de log em arquivos compactados depois que os logs atingem um determinado tamanho ou depois de um determinado período de tempo desde o backup anterior.

Os recursos para configurar e gerenciar o log do sistema são descritos na seção “Configurando o registro em log do sistema”, mais adiante neste capítulo.

Execute relatórios de atividade do sistema

O recurso `sar` (que é habilitado pelo pacote `sysstat`) pode ser configurado para observar atividades em seu sistema, como uso de memória, uso de CPU, latência de disco, atividades de rede e outros drenos de recursos. Por padrão, o recurso `sar` é carregado a cada poucos minutos, dia e noite, para coletar dados. Examinar esses dados mais tarde pode ajudar você a voltar e descobrir onde e quando a demanda tem picos em seu sistema. O recurso `sar` é descrito em “Verificando recursos de sistema com `sar`”, mais adiante neste capítulo.

Mantenha o software de sistema atualizado

À medida que brechas de segurança são descobertas e atualizadas, você deve certificar-se de que os pacotes de software atualizados contendo os patches (correções) estão instalados em seus servidores.

Novamente, com servidores de missão crítica, a maneira mais segura e mais eficiente é usar sistemas Red Hat Enterprise Linux por assinatura para seus servidores e, então, implantar atualizações de segurança relacionadas com o pacote em seu sistema, logo que eles são liberados e testados.

Para manter seu servidor pessoal e sistemas desktop atualizados, você pode usar a janela Add/Remove Software do PackageKit para verificar se há atualizações. Você também pode usar o comando `yum` para verificar e instalar todos os pacotes que estão disponíveis para sistemas RHEL ou Fedora (digite **yum update**).

Verifique sinais de invasão do sistema de arquivos

Para verificar sinais de invasão do sistema de arquivos, você pode executar comandos como `rpm -V`, que lhe mostrarão se quaisquer comandos, arquivos de documentos ou arquivos de configuração foram adulterados em seu sistema. Para mais informações sobre `rpm -V`, consulte a descrição desse comando no Capítulo 10, “Obtendo e gerenciando software”.

Agora que você tem uma visão geral de como a configuração do servidor Linux é feita, as próximas seções desse capítulo focalizam as ferramentas que você precisa para acessar, proteger e manter seus sistemas de servidores Linux.

Gerenciando o acesso remoto com o serviço Secure Shell

As ferramentas Secure Shell são um conjunto de aplicativos cliente-servidor que permite fazer comunicações básicas entre computadores clientes e servidores Linux. As ferramentas incluem `ssh`, `scp`, `sftp` e muitas outras. Como a comunicação é criptografada entre o servidor e os clientes, essas ferramentas são mais seguras do que as ferramentas similares antigas. Por exemplo, em vez de usar comandos de login remoto antigos, como `telnet` ou `rlogin`, você poderia usar `ssh`. O comando `ssh` também pode substituir comandos de execução remota antigos, como `rsh`. Comandos de cópia remotos, como `rcp`, podem ser substituídos por comandos seguros, como `scp` e `rsync`.

Com as ferramentas do Secure Shell, o processo de autenticação e todas as comunicações que se seguem são criptografados. Comunicações de `telnet` e os antigos comandos `r` expõem senhas e todos os dados para alguém fazendo *sniffing* (análise de pacotes) na rede. Hoje, `telnet` e comandos semelhantes devem ser usados apenas para testar o acesso a portas remotas ou fazer outras tarefas que não expõem seus dados privados.

ta

uma discussão mais profunda de técnicas de criptografia, consulte o Capítulo 23, “Noções básicas de segurança cada em Linux”.

A maioria dos sistemas Linux inclui clientes Secure Shell e muitos também incluem o servidor de shell seguro. Se você estiver usando uma distribuição do Fedora ou do RHEL, por exemplo, os pacotes de software cliente-servidor que contêm as ferramentas `ssh` são os pacotes `openssh`, `openssh-`

clients e openssh-server, como segue:

```
# yum list installed | grep ssh
...
openssh.i686                5.8p2-25.fc16 @updates
openssh-clients.i686        5.8p2-25.fc16 @updates
openssh-server.i686         5.8p2-25.fc16 @updates
```

No Ubuntu, apenas o pacote `openssh-clients` está instalado. Ele inclui a funcionalidade do pacote `openssh`. Se você precisar do servidor instalado, use o comando `sudo apt-get install openssh-server`.

```
$ sudo dpkg --get-selections | grep openssh
ii openssh-client 1:5.8p1-7ubuntu1
    secure shell (SSH) client, for secure
    access to remote machines
$ sudo apt-get install openssh-server
```

Iniciando o serviço openssh-server

Sistemas Linux que vêm com o pacote `openssh-server` já instalado, muitas vezes, não estão configurados para iniciar automaticamente. Serviços de gerenciamento do Linux (consulte o Capítulo 15, “Iniciando e parando serviços”) podem ser muito diferentes, dependendo das diferentes distribuições. A Tabela 13.1 mostra os comandos que devem ser usados a fim de garantir que o daemon de servidor `ssh`, o `sshd`, está instalado e funcionando em um sistema Linux.

TABELA 13.1 Comandos para Determinar o Status do sshd

tribuição	Comando para determinar o status do sshd
EL	<code>chkconfig --list sshd</code>
lora	<code>systemctl status sshd.service</code>
untu	<code>status ssh</code>

Se `sshd` não está sendo executado, você pode iniciá-lo emitindo um dos comandos listados na Tabela 13.2. Esses comandos precisam de privilégios de root para funcionar.

TABELA 13.2 Comandos para Iniciar o sshd

tribuição	Comando para iniciar o sshd
EL	<code>service sshd start</code>
lora	<code>systemctl start sshd.service</code>
untu	<code>service ssh start</code>

Os comandos na Tabela 13.2 só vão iniciar o serviço `ssh`. Eles não vão configurá-lo para iniciar automaticamente na inicialização. Para garantir que o serviço do servidor esteja configurado para iniciar automaticamente, você precisará usar um dos comandos na Tabela 13.3, com privilégios de `root`.

TABELA 13.3 Comandos para Iniciar o `sshd` na Inicialização

Distribuição	Comando para iniciar o <code>sshd</code> na inicialização
EL	<code>chkconfig sshd on</code>
Fedora	<code>systemctl enable sshd.service</code>
Ubuntu	<code>update-rc.d ssh defaults</code>

Quando você instala o `openssh-server` no Ubuntu, o daemon `sshd` está configurado para iniciar automaticamente na inicialização. Portanto, você pode não precisar executar o comando na Tabela 13.3 para seu servidor Ubuntu.

ca

Para gerenciar serviços em uma distribuição Fedora mais antiga, use o comando `chkconfig` tanto para iniciar o serviço como para garantir que ele vai iniciar na inicialização do sistema.

Modifique suas configurações de firewall em `netfilter/iptables` para permitir que o `openssh-client` acesse a porta 22 (firewalls são abordados no Capítulo 25, “Protegendo o Linux em uma rede”). Uma vez que o serviço está funcionando e o firewall está configurado corretamente, você deve ser capaz de usar comandos de cliente `ssh` para acessar o sistema via servidor `ssh`.

ta

Para melhor usar um TCP Wrapper com serviços `ssh`. TCP Wrappers são abordados no Capítulo 25, “Protegendo o Linux em uma rede”.

Mais configurações que o daemon `sshd` tem permissão para fazer são tratadas no arquivo `/etc/ssh/sshd_config/`. No mínimo, altere a configuração `PermitRootLogin` de `yes` para `no`. Isso impedirá todo mundo de fazer login remotamente como `root`.

```
# grep PermitRootLogin /etc/ssh/sshd_config
PermitRootLogin no
```

Depois de ter alterado o arquivo `sshd_config`, reinicie o serviço `sshd`. Após esse ponto, se você usar o `ssh` para fazer login no sistema a partir de um cliente remoto, você deve fazê-lo como um usuário comum e, então, usar `su` ou `sudo` para se tornar o usuário `root`.

Usando ferramentas de cliente SSH

Muitas ferramentas para acessar sistemas remotos Linux foram criadas para fazer uso do serviço SSH. A mais utilizada dessas ferramentas é o comando `ssh`, que pode ser usado para login remoto, execução remota e outras tarefas. Comandos como `scp` e `rsync` podem copiar um ou mais arquivos de cada vez entre sistemas SSH clientes e servidores. O comando `sftp` fornece uma interface do tipo FTP para navegar por um sistema de arquivos remoto e copiar e transferir arquivos entre os sistemas de forma interativa.

Por padrão, todas as ferramentas relacionadas com SSH autenticam usando o padrão de nomes de usuários e senhas do Linux, tudo feito através de conexões criptografadas. Mas o SSH também suporta autenticação baseada em chave, que pode ser usada para configurar a autenticação sem senha entre clientes e servidores SSH (como descrito na seção “Usando a autenticação baseada em chave (sem senha)”, mais adiante, neste capítulo).

Usando SSH para login remoto

Use o comando `ssh` a partir de outro computador Linux para testar se você é capaz de fazer login no sistema Linux rodando o serviço `sshd`. Você usará o comando `ssh` com frequência para acessar um shell nos servidores que você está configurando.

Tente fazer login em seu servidor Linux a partir de outro sistema Linux usando o comando `ssh`. (Se não tiver outro sistema Linux, você pode simular isso digitando `localhost` em vez do endereço IP e fazer login como um usuário local.) Eis um exemplo de login remoto com a conta de `johndoe` em `10.140.67.23`:

```
$ ssh johndoe@10.140.67.23
The authenticity of host '10.140.67.23 (10.140.67.23)' can't be established.
RSA key fingerprint is a4:28:03:85:89:6d:08:fa:99:15:ed:fb:b0:67:55:89.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.140.67.23' (RSA) to the list of known hosts.
johndoe@10.140.67.23's password: *****
```

Se esta é a primeira vez que você tenta fazer login no sistema remoto usando o comando `ssh`, o sistema pedirá que você confirme que deseja se conectar. Digite `yes` e pressione Enter. Então, quando solicitado, digite a senha do usuário.

Quando digita `yes` para continuar, você aceita a chave pública do host remoto. Nesse ponto, a chave pública do host remoto é baixada no arquivo `~/.ssh/known_hosts` do cliente. Agora, os dados trocados entre esses dois sistemas podem ser criptografados e descriptografados utilizando criptografia assimétrica RSA (consulte o Capítulo 23, “Noções básicas de segurança avançada em Linux”).

Depois de se conectar ao sistema remoto, você pode começar a digitar comandos de shell. A conexão funciona como um login normal, a única diferença é que os dados são criptografados ao trafegarem pela rede.

Quando você terminar, digite `exit` para fechar a conexão remota. A conexão será fechada e você será devolvido ao prompt de comando em seu sistema local. (Se o shell local não retornar após deixar o shell remoto, digitar `~` normalmente fecha a conexão.)

```
$ exit
logout
Connection to 10.140.67.23 closed.
$
```

Depois que você se conecta remotamente a um sistema, aparece um arquivo em um subdiretório do seu sistema local, `~/.ssh/known_hosts`. Esse arquivo contém a chave pública do host remoto junto com seu endereço IP. Chaves públicas e privadas de seu servidor são armazenadas no diretório `/etc/ssh`.

```
$ ls .ssh
known_hosts
$ cat .ssh/known_hosts
10.140.67.23 ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAQEAoyfJK1YwZhNmpHE4yLPZAZ9ZNEdRE
7I159f3IyGiH21IjfqS
NYFR10Z1BL1YyTQi06r/9019GwCaJ753InQ8FVHW+OOYOG5pQmghhn/
x0LD2uUb6egOu6zim1NEC
JwZf5DWkKdy4euCUEMSqADh/WYeuOSoZ0pp2IAVCdh6w/
PIHMF1HVR069cvdv+OTL4vD0X811Spw
0ozqRptz2UQgQBBbBjK1RakD7fY1TrWvNqhYG/
ugtGPaY4JDYeY6OBzcadpxZmf7EYUw0ucXGVQ1a
NP/erIDOQ9rA0YNzCRvy2LYCm2/9adpAxc+UYi5UsxTw4ewSBjmsXYq//Ahaw4mjw==
```

Qualquer tentativa posterior desse usuário de se conectar com o servidor em `10.140.67.23` será autenticada usando a chave armazenada. Se o servidor mudar a chave dele (o que acontece se o sistema operacional for reinstalado), tentativas de se conectar via `ssh` com esse sistema resultarão na recusa da conexão e terríveis advertências de que você pode estar sob ataque. Se a chave de fato mudou, para ser capaz de se conectar via `ssh` com o endereço novo, você precisa remover a chave do host (toda a linha) do arquivo `known_hosts` e você será capaz de baixar a nova chave.

Usando SSH para execução remota

Além de fazer login em um shell remoto, o comando `ssh` pode ser usado para executar um comando no sistema remoto e fazer o resultado ser retornado para o sistema local. Eis um exemplo:

```
$ ssh johndoe@10.140.67.23 hostname
johndoe@10.140.67.23's password: *****
jd.example.com
```

No exemplo mostrado, o comando `hostname` é executado como o usuário `johndoe` no sistema Linux localizado no endereço IP `10.140.67.23`. A saída do comando é o nome do servidor remoto (nesse caso, `jd.example.com`), que aparece na tela local.

Se você rodar um comando de execução remota com `ssh` que inclui opções ou argumentos, não se esqueça de colocar a linha de comando remoto inteira entre aspas. Tenha em mente que, se você referenciar arquivos ou diretórios em seus comandos remotos, caminhos relativos são interpretados em relação ao diretório inicial do usuário. Por exemplo:

```
$ ssh johndoe@10.140.67.23 "cat myfile"
johndoe@10.140.67.23's password: *****
This is the contents of the myfile file located in johndoe's home directory.
```

O comando `ssh` mostrado simplesmente vai para o host remoto localizado em 10.140.67.23 e executa o comando `cat myfile` como o usuário `johndoe`. Isso faz com que o conteúdo do arquivo `myfile` desse sistema seja exibido na tela local.

Outro tipo de execução remota que você pode fazer com `ssh` é o encaminhamento X11. Se o encaminhamento X11 estiver habilitado no servidor (`X11Forwarding yes` é definido no arquivo `/etc/sshd/sshd_config`), você pode executar aplicativos gráficos do servidor seguramente através da conexão SSH usando `ssh-X`. Para um administrador de servidor iniciante, isso significa que, se houver ferramentas gráficas de administração instaladas em um servidor, você pode executar essas ferramentas sem ter de sentar-se à frente do console. Por exemplo:

```
$ ssh -X johndoe@10.140.67.23 system-config-date
johndoe@10.140.67.23's password: *****
```

Depois de executar esse comando, você será solicitado a informar a senha de root. Depois disso, a janela Date/Time Properties aparece, pronta para você alterar a data e a hora atuais. Basta fechar a janela quando terminar e o prompt local retorna. Você pode fazer isso para qualquer ferramenta de administração gráfica ou apenas aplicativos X regulares (como o editor gráfico `gedit`, assim você não precisa usar o `vi`).

Se quiser executar vários comandos X e não quiser ter de se reconectar a cada vez, você também pode usar o encaminhamento X11 diretamente a partir de um shell remoto. Coloquem os comandos em segundo plano e você pode ter vários aplicativos X remotos em execução em seu desktop local ao mesmo tempo. Por exemplo:

```
$ ssh -X johndoe@10.140.67.23
johndoe@10.140.67.23's password: *****
$ system-config-network &
$ gedit &
$ exit
```

Depois de terminar de usar os aplicativos gráficos, feche-os como faria normalmente. Então, digite **exit**, como mostrado no código anterior, para deixar o shell remoto e retornar ao seu shell local.

Copiando arquivos entre sistemas com `scp` e `rsync`

O comando `scp` é semelhante ao antigo comando `rcp` do UNIX para copiar arquivos de e para sistemas Linux, exceto que todas as comunicações são encriptadas. Os arquivos podem ser copiados do sistema remoto para o sistema local ou do local para o remoto. Você também pode copiar arquivos recursivamente ao longo de uma estrutura de diretórios inteira, se escolher.

Eis um exemplo de como usar o comando `scp` para copiar um arquivo chamado `memo` do diretório inicial do usuário `chris` para o diretório `/tmp` em um computador remoto:

```
$ scp /home/chris/memo johndoe@10.140.67.23:/tmp
johndoe@10.140.67.23's password: *****
memo 100%|*****| 153 0:00
```

Você deve digitar a senha johndoe. Depois que a senha é aceita, o arquivo é copiado para o sistema remoto com sucesso.

Você pode fazer cópias recursivas com `scp` usando a opção `-r`. Em vez de um arquivo, passe um nome de diretório para o comando `scp` e todos os arquivos e diretórios abaixo desse ponto no sistema de arquivos serão copiados para o outro sistema.

```
$ scp johndoe@10.140.67.23:/usr/share/man/man1/ /tmp/
johndoe@10.140.67.23's password: *****
volname.1.gz          100% 543    0.5KB/s  00:00
mtools.1.gz          100% 6788   6.6KB/s  00:00
roqet.1.gz           100% 2496   2.4KB/s  00:00
...
```

Desde que o usuário johndoe tenha acesso aos arquivos e diretórios no sistema remoto e o usuário local seja capaz de gravar no diretório de destino (ambas são verdadeiras, neste caso), a estrutura de diretórios de `/usr/share/man/man1` abaixo é copiada para o diretório `/tmp` local.

O comando `scp` pode ser usado para fazer backup de arquivos e diretórios em uma rede. Mas se comparar `scp` ao comando `rsync`, você vê que `rsync` (que também funciona via conexões SSH) é a melhor ferramenta de backup. Tente executar o comando `scp` mostrado anteriormente para copiar o diretório `man1` (você pode simular o comando usando `localhost` em vez do endereço IP, se só tiver um sistema Linux acessível). Agora, digite o seguinte no sistema para o qual você copiou os arquivos:

```
$ ls -l /usr/share/man/man1/batch* /tmp/man1/batch*
-rw-r--r--.1 johndoe johndoe 2628 Apr 15 15:32 /tmp/man1/batch.1.gz lrwxrwxrwx.1 root root
7 Feb 14 17:49 /usr/share/man/man1/batch.1.gz -> at.1.gz
```

Então, execute o comando `scp` novamente e liste os arquivos mais uma vez:

```
$ scp johndoe@10.140.67.23:/usr/share/man/man1/ /tmp/
johndoe@10.140.67.23's password: *****
$ ls -l /usr/share/man/man1/batch* /tmp/man1/batch*
-rw-r--r--.1 johndoe johndoe 2628 Apr 15 15:40 /tmp/man1/batch.1.gz lrwxrwxrwx.1 root root
7 Feb 14 17:49 /usr/share/man/man1/batch.1.gz -> at.1.gz
```

A saída desses comandos informa algumas coisas sobre como o `scp` funciona:

- **Atributos perdidos** — Permissões ou atributos de registro de data/hora não são mantidos quando os arquivos são copiados. Se estiver usando `scp` como uma ferramenta de backup, você provavelmente vai querer manter permissões e registros de data/hora nos arquivos se precisar restaurá-los posteriormente.
- **Links simbólicos perdidos** — O arquivo `batch.1.gz` é, na verdade, um link simbólico para o arquivo `at.1.gz`. Em vez de copiar o link, `scp` segue-o e realmente copia o arquivo. Novamente, se você fosse restaurar esse diretório, `batch.1.gz` seria substituído pelo arquivo `at.1.gz` real, em vez de um link para ele.

- **Cópias repetidas desnecessariamente** — Se você examinou a segunda saída de `scp`, você deve ter notado que todos os arquivos foram copiados de novo, ainda que exatamente os mesmos arquivos já estivessem no destino. A data de modificação atualizada confirma isso. Em contraposição, `rsync` pode determinar que um arquivo já foi copiado e não copiá-lo novamente.

O comando `rsync` é uma ferramenta melhor de backup de rede, pois pode superar algumas das deficiências de `scp` que acabamos de citar. Tente executar um comando `rsync` para fazer a mesma ação que `scp` fez, mas com algumas opções adicionais:

```
$ rm -rf /tmp/man1/
$ rsync -avl johndoe@10.140.67.23:/usr/share/man/man1/ /tmp/
johndoe@10.140.67.23's password: *****
sending incremental file list
man1/
man1/HEAD.1.gz
man1/Mail.1.gz -> mailx.1.gz
...
$ rsync -avl johndoe@10.140.67.23:/usr/share/man/man1/ /tmp/
johndoe@10.140.67.23's password: *****
sending incremental file list
sent 42362 bytes received 13 bytes 9416.67 bytes/sec
total size is 7322223 speedup is 172.80
$ ls -l /usr/share/man/man1/batch* /tmp/man1/batch*
lrwxrwxrwx.1 johndoe johndoe 7 Feb 14 17:49 /tmp/man1/batch.1.gz ->
at.1.gz
lrwxrwxrwx.1 root root 7 Feb 14 17:49 /usr/share/man/man1/batch.1.gz ->
at.1.gz
```

Depois de remover o diretório `v`, você executa um comando `rsync` para copiar todos os arquivos para o diretório `v`, usando `-a` (arquivamento recursivo), `-v` (verboso), `-l` (copia links simbólicos). Então, execute imediatamente o comando de novo e observe que nada é copiado. O comando `rsync` sabe que todos os arquivos já estão lá e, por isso, não os copia novamente. Essa pode ser uma enorme economia de largura de banda de rede para diretórios com gigabytes de arquivos, nos quais apenas alguns megabytes mudam.

Além disso, observe a partir da saída de `ls -l` que os links simbólicos foram preservados no arquivo `batch.1.gz` e, por isso, têm o registro de data/hora no arquivo. Se precisar restaurar os arquivos mais tarde, você pode trazê-los de volta exatamente como eram.

Esse uso de `rsync` é bom para backups. Mas e se você quisesse espelhar dois diretórios, tornando o conteúdo de duas estruturas de diretórios exatamente o mesmo em duas máquinas? Os comandos a seguir ilustram como criar um espelho exato da estrutura de diretórios em ambas as máquinas, usando os diretórios mostrados com os comandos `rsync` anteriores.

Primeiro, no sistema remoto, copie um novo arquivo para o diretório a ser copiado:

```
# cp /etc/services /usr/share/man/man1
```

Então, no sistema local, execute `rsync` para copiar quaisquer novos arquivos (nesse caso, apenas o diretório e o novo arquivo, `services`):

```
$ rsync -avl johndoe@10.140.67.23:/usr/share/man/man1 /tmp
johndoe@10.140.67.23's password:
*****
sending incremental file list
man1/
man1/services
```

Depois disso, volte para o sistema remoto e remova o novo arquivo:

```
$ sudo rm /usr/share/man/man1/services
```

Agora, no sistema local, execute `rsync` novamente e perceba que nada acontece. Nesse ponto, os diretórios locais e remotos são diferentes, porque o sistema local tem o arquivo `services` e o remoto não. Esse é o comportamento correto para um diretório de backup (você quer ter os arquivos do backup no caso de algo ter sido removido por engano). Mas se quiser que os diretórios remotos e locais sejam espelhados, você teria de adicionar a opção `--delete`. O resultado é que o arquivo `services` é excluído do sistema local, fazendo com que as estruturas de diretórios local e remota sejam sincronizadas.

```
$ rsync -avl /usr/share/man/man1 localhost:/tmp
johndoe@10.140.67.23's password: *****
sending incremental file list
man1/
$ rsync -avl --delete johndoe@10.140.67.23:/usr/share/man/man1 /tmp
johndoe@10.140.67.23's password: *****
sending incremental file list
deleting man1/services
```

Cópia interativa com sftp

Se você não sabe exatamente o que deseja copiar para ou a partir de um sistema remoto, você pode usar o comando `sftp` para criar uma sessão no estilo FTP interativa sobre o serviço SSH. Usando `sftp`, você pode conectar-se a um sistema remoto via SSH, mudar de diretório, listar o conteúdo do diretório e, então, (dada a devida permissão) copiar arquivos do servidor (`get`) e transferir arquivos para o servidor (`put`).

O exemplo a seguir mostra o usuário `johndoe` se conectando com `jd.example.com`:

```
$ sftp johndoe@jd.example.com
Connecting to jd.example.com
johndoe@jd.example.com's password: *****
sftp>
```

Nesse ponto, você pode começar uma sessão interativa de FTP. Você pode usar os comandos `get` e `put` em arquivos como você faria com qualquer cliente de FTP, mas com o conforto de saber que está trabalhando em uma conexão criptografada e segura. Como o protocolo FTP passa nomes de usuário,

senhas e dados em texto claro, usando `sftp` sobre SSH, se possível, ele é uma alternativa muito melhor para permitir que seus usuários copiem interativamente arquivos do sistema.

Utilizando autenticação baseada em chave (sem senha)

Se estiver usando ferramentas SSH para conectar os mesmos sistemas ao longo do dia, você pode achar que é inconveniente digitar sua senha repetidamente. Em vez de usar autenticação baseada em senha, o SSH permite que você configure autenticação baseada em chave para utilizar em seu lugar. Veja como funciona:

- Você cria uma chave pública e uma chave privada.
- Você guarda a chave privada, mas copia a chave pública por meio da conta do usuário no host remoto para o qual você quer fazer autenticação baseada em chave.
- Com suas chaves copiadas para os locais apropriados, você pode usar todas as ferramentas SSH para se conectar à conta do usuário no host remoto, mas em vez de pedir uma senha, o serviço remoto SSH compara a chave pública e a chave privada e permite o acesso se as duas chaves coincidirem.

Quando você cria as chaves, você tem a opção de adicionar uma senha à sua chave privada. Se decidir adicionar uma senha, mesmo que você não precisa digitar uma senha para se autenticar no sistema remoto, você ainda precisará digitar sua senha para desbloquear sua chave privada. Se não adicionar uma senha, você pode se comunicar usando seu par de chaves pública/privada de uma maneira completamente sem senha. Mas se alguém se apossar de sua chave privada, essa pessoa poderia agir como você em qualquer comunicação que exigisse essa chave.

O procedimento a seguir demonstra como um usuário local chamado `chris` pode configurar autenticação baseada em chave para um usuário remoto chamado `johndoe` no endereço IP `10.140.67.23`. Se não tiver dois sistemas Linux, você pode simular isso usando duas contas de usuário no sistema local. Começo fazendo login como o usuário local chamado `chris` e digitando o seguinte para gerar meu par local de chaves pública/privada:

```
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/chris/.ssh/id_rsa): <ENTER>
Enter passphrase (empty for no passphrase): <ENTER>
Enter same passphrase again: <ENTER>
Your identification has been saved in /home/chris/.ssh/id_rsa.
Your public key has been saved in /home/chris/.ssh/id_rsa.pub.
The key fingerprint is:
bf:06:f8:12:7f:f4:c3:0a:3a:01:7f:df:25:71:ec:1d chris@chris.example.com
The key's randomart image is:
...
```

Aceitei o padrão de chave RSA (também são permitidas chaves DSA) e então pressionei Enter duas vezes para ter uma senha em branco associada com a chave. Como resultado, minha chave privada (`id_rsa`) e minha chave pública (`id_rsa.pub`) são copiadas para o diretório `.ssh`, no diretório inicial de `johndoe`. O próximo passo é copiar a chave para um usuário remoto, para que eu possa usar

autenticação baseada em chave cada vez que me conectar a essa conta do usuário com as ferramentas `ssh`:

```
$ ssh-copy-id -i ~/.ssh/id_rsa.pub johndoe@10.140.67.23
johndoe@10.140.67.23's password:
*****
Now try logging into the machine, with "ssh 'johndoe@10.140.67.23'",
and check in:
    .ssh/authorized_keys
to make sure we haven't added extra keys that you weren't expecting.
```

Quando solicitado, inseri a senha de `johndoe`. Com isso aceito, a chave pública pertencente a `chris` é copiada para o arquivo `authorized_keys` no diretório `.ssh` de `johndoe`. Agora, da próxima vez que `chris` tentar se conectar à conta de `johndoe`, a conexão SSH é autenticada usando essas chaves. Como nenhuma senha foi definida para a chave privada, nenhuma senha é necessária para desbloquear a chave quando ela é usada.

```
[chris]$ ssh johndoe@10.140.67.23
Last login: Sun Apr 15 10:12:22 2012 from 10.140.67.22
[johndoe]$
```

Com as chaves no lugar, `chris` poderia agora usar `ssh`, `scp`, `rsync` ou qualquer outro comando habilitado para SSH a fim de fazer autenticação baseada em chave. Usando essas chaves, por exemplo, um comando `rsync` poderia entrar em um script `cron` e fazer um backup automático do diretório inicial de `johndoe` toda noite.

Configurando o registro em log do sistema

Com o conhecimento de como acessar o servidor remoto usando ferramentas SSH, você pode fazer login no servidor e configurar alguns dos serviços necessários para se certificar de que estão funcionando perfeitamente. O registro em log do sistema é um dos serviços básicos configurados para Linux para monitorar o que está acontecendo no sistema.

O serviço `rsyslog` (daemon `rsyslogd`) fornece os recursos para coletar as mensagens de registro em log de aplicativos em execução no sistema Linux e direcionar essas mensagens para arquivos de log locais, dispositivos ou hosts de log remotos. A configuração de `rsyslog` é semelhante à configuração de seu predecessor, `syslog`. Mas `rsyslog` permite adicionar módulos para mais especificamente administrar e direcionar as mensagens de log.

Ativando o log do sistema com rsyslog

A maioria dos arquivos no diretório de `/var/log` é mantida pelo serviço `rsyslog`. O daemon `rsyslogd` é o de registro em log do sistema. Ele aceita mensagens de log de vários outros programas e grava essas mensagens nos arquivos de log apropriados. Isso é melhor do que ter todos os programas gravando diretamente em seu próprio arquivo de log, porque permite que você gerencie centralmente

como os arquivos de log são tratados.

É possível configurar `rsyslogd` para gravar vários níveis de detalhe nos arquivos de log. Pode-se instruí-lo a ignorar tudo, exceto as mensagens mais críticas ou ele pode registrar em log cada detalhe.

O daemon `rsyslogd` pode até aceitar mensagens de outros computadores em sua rede. Esse recurso de log remoto é particularmente útil porque permite centralizar o gerenciamento e revisão dos arquivos de log de muitos sistemas em sua rede. Há também um importante benefício de segurança nessa prática.

Com o registro em log remoto, se um sistema em sua rede for invadido, o cracker não pode excluir ou modificar os arquivos de log porque eles são armazenados em um computador separado. É importante lembrar, porém, que essas mensagens de log não são, por padrão, criptografadas. Qualquer pessoa que consiga espionar o tráfego de dados da sua rede local pode ler as mensagens que passam de uma máquina para outra. Além disso, embora o cracker possa não ser capaz de alterar as entradas de log antigas, ele pode afetar o sistema de tal forma que as novas mensagens de log podem não ser confiáveis.

Não é incomum executar um servidor de logs dedicado, um computador que não serve a nenhum propósito além de gravar mensagens de log de outros computadores na rede, porque, como esse sistema não executa nenhum outro serviço, é improvável que seja invadido. Isso torna quase impossível para os crackers apagar completamente seus rastros.

Entendendo o arquivo `rsyslog.conf`

O arquivo `/etc/rsyslog.conf` é o principal arquivo de configuração do serviço `rsyslog`. Se já usou o antigo recurso `syslog`, você vai notar que a seção de regras é a mesma em ambos os arquivos. Assim, a maneira de definir que tipo de mensagens são registradas e onde são registradas é exatamente o mesmo; mas os arquivos de configuração são diferentes quanto ao uso de módulos em `rsyslog.conf`.

No arquivo `/etc/rsyslog.conf`, uma seção de módulos permite incluir ou não recursos específicos em seu serviço `rsyslog`. Eis um exemplo da seção de módulos `/etc/rsyslog.conf` no Fedora:

```
$ModLoad imuxsock # provides support for local system logging (logger
command)
$ModLoad imklog # provides kernel logging support (previously done by
rklogd)
#$ModLoad immark # provides --MARK-- message capability
# Provides UDP syslog reception
#$ModLoad imudp
#$UDPServerRun 514
# Provides TCP syslog reception
#$ModLoad imtcp
#$InputTCPServerRun 514
```

As entradas começando com `$ModLoad` carregam os módulos que se seguem. Módulos que estão atualmente desabilitados são precedidos por um sinal de jogo da velha (`#`). O módulo `imuxsock` é necessário para aceitar as mensagens de sistema locais (cuja linha que não deve ser “comentada” — precedida por um sinal de jogo da velha — a menos que você tenha uma razão específica para isso). O módulo `imklog` registra mensagens do kernel.

Módulos não habilitados por padrão incluem o módulo `immark`, que permite que mensagens `--MARK--` sejam registradas em log (usado para indicar que um serviço está ativo). Os módulos `imudp` e `imtcp` e entradas relacionadas de número de porta são usados para permitir que o serviço `rsyslog` aceite mensagens de log remotas e são discutidos em mais detalhes na seção “Configurando e usando um

servidor de logs com rsyslogd”.

A maior parte do trabalho feito no arquivo de configuração `/etc/rsyslog.conf` envolve a modificação da seção `RULES`. Eis um exemplo de algumas das regras na seção `RULES` do arquivo `/etc/rsyslog.conf` (note que, no Ubuntu, você precisa procurar esse arquivo no diretório `/etc/rsyslog.d`):

```
#### RULES ####
# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none    /var/log/messages
# The authpriv file has restricted access.
authpriv.*                                  /var/log/secure
# Log all the mail messages in one place.
mail.*                                       -/var/log/maillog
# Log cron stuff
cron.*                                       /var/log/cron
```

Entradas de regras vêm em duas colunas. Na coluna esquerda, estão designações das mensagens que são correspondidas e a coluna direita mostra onde entram as mensagens correspondentes. Mensagens são correspondidas com base em recursos (`mail`, `cron`, `kern` etc.) e prioridade (começando em `debug`, `info`, `notice` e subindo até `crit`, `alert` e `emerg`), separados por um ponto (`.`). Portanto, `mail.info` corresponde a todas as mensagens do serviço de e-mail que são de nível `info` e acima.

Quanto ao destino das mensagens, a maioria é direcionada para arquivos no diretório `/var/log`. Mas você pode direcionar as mensagens para um dispositivo (como `/dev/console`) ou um host de log remoto `@loghost.example.com`). O sinal de arroba (`@`) indica que o nome que se segue é o nome do servidor de logs.

A primeira entrada no exemplo anterior mostra que as mensagens de nível informativo de todos os serviços (`*`) são correspondidas por essa regra, com exceção de mensagens dos serviços `mail`, `authpriv` e `cron` (que são excluídas com a palavra `none`). Todas essas mensagens correspondidas são direcionadas para o arquivo `/var/log/messages`.

Os serviços `mail`, `authpriv` (mensagens de autenticação) e `cron` (mensagens do recurso `cron`) têm seus próprios arquivos de log, como indicado nas colunas à direita deles. Para entender o formato desses e outros arquivos de log, o formato do arquivo `/var/log/messages` é descrito a seguir.

Entendendo o arquivo de log de mensagens

Por causa dos muitos programas e serviços que gravam informações no arquivo de log `messages`, entender o formato desse arquivo é importante. Você pode ser avisado a tempo sobre o desenvolvimento de problemas em seu sistema examinando esse arquivo. Cada linha no arquivo é uma única mensagem registrada por algum programa ou serviço. Eis um trecho de um arquivo de log `messages` real:

```
Feb 25 11:04:32 toys network: Bringing up loopback interface: succeeded
Feb 25 11:04:35 toys network: Bringing up interface eth0: succeeded
Feb 25 13:01:14 toys vsftpd(pam_unix)[10565]: authentication failure;
      logname= uid=0 euid=0 tty= ruser= rhost=10.0.0.5 user=chris
Feb 25 14:44:24 toys su(pam_unix)[11439]: session opened for
      user root by chris(uid=500)
```

As mensagens no arquivo `/var/log/messages` são divididas em cinco partes principais. Da esquerda para a direita, são elas:

- A data e a hora em que a mensagem foi registrada
- O nome do computador de onde a mensagem veio
- O nome do programa ou serviço a que a mensagem se refere
- O número do processo (entre colchetes) do programa que enviou a mensagem
- A mensagem de texto real

Examine novamente o trecho do arquivo de log anterior. Nas duas primeiras linhas, você pode ver que a rede foi reiniciada. A próxima linha mostra que o usuário chamado `chris` tentou e não conseguiu alcançar o servidor FTP no sistema a partir de um computador no endereço `10.0.0.5` (ele digitou a senha errada e a autenticação falhou). A última linha mostra `chris` usando o comando `su` para se tornar o usuário `root`.

Examinando ocasionalmente os arquivos `messages` e `secure`, é possível flagrar uma tentativa de invasão antes de ela ter sucesso. Se vir um número excessivo de tentativas de conexão para um serviço específico, especialmente se eles são provenientes de sistemas na internet, você pode estar sob ataque.

Configurando e usando um servidor de logs com rsyslogd

Para redirecionar os arquivos de log de seu computador para o `rsyslogd` de outro computador, você deve fazer alterações tanto no arquivo de configuração `rsyslog` local como no remoto, `/etc/rsyslog.conf`. Torne-se `root` usando o comando `su -` e então abra o arquivo `/etc/rsyslog.conf` em um editor de texto (como o `vi`).

No lado do cliente

Para enviar as mensagens para outro computador (o servidor de logs) em vez de um arquivo, comece substituindo o nome do arquivo de log pelo caractere `@` seguido do nome do servidor de logs. Por exemplo, para direcionar a saída de mensagens que estão sendo gravadas nos arquivos de log `messages`, `secure` e `maillog` para um servidor de logs também, adicione as linhas em negrito ao arquivo de mensagens:

```
# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;news.none;authpriv.none;cron.none /var/log/messages
*.info;mail.none;news.none;authpriv.none;cron.none @loghost
# The authpriv file has restricted access.
authpriv.* /var/log/secure
authpriv.* @loghost
# Log all the mail messages in one place.
mail.* -/var/log/maillog
mail.* @loghost
```

As mensagens serão agora enviadas para o `rsyslogd` em execução no computador, chamado `loghost` (servidor de logs). O nome “`loghost`” não foi uma escolha arbitrária. É costume criar um nome de máquina assim e torná-lo um alias para o sistema real agindo como o servidor de logs. Dessa forma, se precisar mudar os deveres do servidor de logs para uma máquina diferente, você só precisa alterar o alias do servidor de logs, não precisa reeditar o arquivo `syslog.conf` em cada computador.

No lado do servidor de logs (loghost)

O servidor de logs que está configurado para aceitar as mensagens deve ouvir essas mensagens em portas padrão (514 UDP, embora também possa ser configurado para aceitar mensagens na porta 514 TCP). Eis como você poderia configurar o servidor de logs Linux que também está executando o serviço `rsyslog`:

- Edite o arquivo `/etc/sysconfig/rsyslog` no sistema `loghost` e “descomente” (isto é, remova o caractere de comentário inicial) as linhas que permitem que o daemon `rsyslogd` ouça mensagens de log remotas. Descomente as duas primeiras linhas para permitir mensagens de log UDP recebidas na porta 514 (padrão); descomente as duas linhas depois disso para permitir mensagens que usam o protocolo TCP (também porta 514):

```
$ModLoad imudp.so
$UDPServerRun 514
$ModLoad imtcp.so
$InputTCPServerRun 514
```

- Abra seu firewall (`iptables`) para permitir que as novas mensagens sejam direcionadas para o servidor de logs. (Veja o Capítulo 25, “Protegendo o Linux em uma rede”, para ler uma descrição de como abrir portas específicas a fim de permitir o acesso ao seu sistema.)
- Reinicie o serviço `rsyslog` (`service rsyslog restart` ou `systemctl restart rsyslog.service`).
- Se o serviço estiver sendo executado, você deve ser capaz de ver que o serviço está ouvindo nas portas que você habilitou (portas 514 UDP e/ou TCP). Execute o comando `netstat` da seguinte forma para ver se o daemon `rsyslogd` está ouvindo serviços UDP e TCP nas portas 514 IPv4 e IPv6:

```
# netstat -tupln | grep 514
tcp    0    0 0.0.0.0:514 0.0.0.0:*    LISTEN  25341/rsyslogd
tcp    0    0 :::514      :::*        LISTEN  25341/rsyslogd
udp    0    0 0.0.0.0:514 0.0.0.0:*    25341/rsyslogd
udp    0    0 :::514      :::*        25341/rsyslogd
```

Observando logs com logwatch

O serviço `logwatch` roda na maioria dos sistemas Linux que fazem registro em log de sistema com `rsyslog`. Como logs em sistemas ocupados podem se tornar muito grandes ao longo do tempo, não demora muito para que haja muitas mensagens para um administrador de sistema observar cada mensagem em cada log.

O que `logwatch` faz é recolher uma vez por noite mensagens que parecem poder representar um problema, colocá-las em uma mensagem de e-mail e enviar para qualquer endereço de e-mail que administrador escolher. Para habilitar `logwatch` tudo que você tem a fazer é instalar o pacote `logwatch`.

O serviço `logwatch` é executado a partir de um trabalho `cron` (`01logwatch`) colocado em `/etc/cron.daily`. O arquivo `/etc/logwatch/conf/logwatch.conf` contém configurações locais. As opções padrão utilizadas para coletar mensagens de log são definidas no

arquivo `/usr/share/logwatch/default.conf/logwatch.conf`.

Algumas das configurações padrão definem a localização dos arquivos de log (`/var/log`), a localização do diretório temporário (`/var/cache/logwatch`) e o destinatário de e-mail de logwatch diário (o usuário `root` local). A menos que você espere fazer login no servidor para ler mensagens logwatch, você provavelmente vai querer alterar a configuração `MailTo` no arquivo `/etc/logwatch/conf/logwatch.conf`:

```
MailTo = chris@example.com
```

Examine `/usr/share/logwatch/default.conf/logwatch.conf` para ver outras configurações que podem ser modificadas (como o nível de detalhe ou o intervalo de tempo para cada relatório). Então, faça suas adições ao arquivo `/etc/logwatch/conf/logwatch.conf`, como mencionado.

Quando o serviço estiver ativado (o que é feito apenas instalando o pacote `logwatch`), você verá uma mensagem a cada noite na caixa postal do usuário `root`. Quando estiver conectado como `root`, você pode usar o antigo comando `mail` para ver a caixa postal do usuário `root`:

```
# mail
Heirloom Mail version 12.4 7/29/08. Type ? for help.
"/var/spool/mail/root": 2 messages 2 new
>N 1 logwatch@abc.ex Sun Apr 15 04:02 45/664 "Logwatch for abc
(Linux)"
2 logwatch@abc.ex Sun Apr 16 04:02 45/664 "Logwatch for abc
(Linux)"
& 1
& x
```

No e-mail, você deve ver as mensagens de email de `logwatch` executarem a cada dia (aqui às 4:02 a.m.). Digite o número da mensagem que você deseja visualizar e examine-as página a página, pressionando a barra de espaço, ou linha a linha, pressionando `Enter`. Digite `x` para sair quando terminar.

O tipo de informação que você vê inclui erros de kernel, pacotes instalados, falhas de autenticação e mau funcionamento de serviços. O uso do espaço em disco é relatado, assim você pode ver se seu armazenamento está se enchendo. Só de olhar essa mensagem de `logwatch`, você deve ter uma ideia se ataques sustentados estão em andamento ou se algumas falhas repetidas estão ocorrendo.

Verificando recursos do sistema com sar

O System Activity Reporter (`sar`) é uma dos mais antigos recursos de monitoramento do sistema criados para os primeiros sistemas UNIX — precedendo o Linux por algumas décadas. O comando `sar` sozinho pode exibir a atividade do sistema continuamente na tela, a intervalos definidos (a cada um ou dois segundo). Ele também pode exibir dados de atividade do sistema que foram coletados anteriormente.

O comando `sar` é parte do pacote `sysstat`. Ao instalar `sysstat` e habilitar o serviço `sysstat`, seu sistema imediatamente começa a coletar dados de atividades do sistema que podem ser analisados posteriormente utilizando certas opções do comando `sar`. A coleta de dados é feita por um arquivo de configuração `crontab (/etc/cron.d/sysstat)`, que é carregado em intervalos regulares. Dê uma olhada no que o arquivo contém:

```
# cat /etc/cron.d/sysstat
# Run system activity accounting tool every 10 minutes
*/10 * * * * root /usr/lib64/sa/sa1 -S DISK 1 1
# 0 * * * * root /usr/lib64/sa/sa1 -S DISK 600 6 &
# Generate a daily summary of process accounting at 23:53
53 23 * * * root /usr/lib64/sa/sa2 -A
```

A primeira linha descomentada roda o comando `sa1 -S DISK 1 1` a cada 10 minutos. Esse comando `sa1` coleta informações de atividade em disco e outras informações apenas uma vez (a cada segundo) e as copia para o arquivo `/var/log/sa/sa??`, em que `??` é substituído pelo dia atual. O comando `sa2 -A` reúne todos os dados coletados até esse ponto no dia (às 11:23 p.m.) e os coloca no arquivo `/var/log/sa/sar??`, em que `??` é substituído pelo dia atual.

Para ler os dados nos arquivos `sa??` e `sar??`, você pode usar alguns dos seguintes comandos `sar`:

```
# sar -u | less
Linux 2.6.32-220.7.1.el6.x86_64 (mybox) 04/17/2012 _x86_64_ (4 CPU)
12:00:01 AM CPU %user %nice %system %iowait %steal %idle
12:10:01 AM all 6.48 0.00 0.82 0.59 0.00 92.12
12:20:01 AM all 6.50 0.00 0.78 0.82 0.00 91.91
```

A opção `-u` mostra o uso da CPU. Por padrão, a saída começa à meia-noite do dia atual e, então, mostra quanto tempo de processamento está sendo consumido por diferentes partes do sistema. A saída continua a mostrar a atividade a cada 10 minutos, até que a hora atual seja alcançada.

Para ver a saída da atividade de disco, execute o comando `sar -d`. Novamente, a saída ocorre em intervalos de 10 minutos a partir de meia-noite.

```
# sar -d | less
Linux 2.6.32-220.7.1.el6.x86_64 (mybox) 04/17/2012 _x86_64_ (4 CPU)
12:00:01 AM DEV tps rd_sec/s wr_sec/s avgrq-sz avgqr-sz await svctm %util
12:10:01 AM dev8-0 1.39 0.24 18.49 13.44 0.04 27.01 16.77 2.34
12:10:01 AM dev253- 02.59 0.24 18.49 7.24 0.04 15.81 9.04 2.34
12:10:01 AM dev253-1 0.97 0.24 7.69 8.14 0.02 19.95 17.25 1.68
```

Se quiser executar relatórios de atividade “ao vivo” de `sar`, você pode fazer isso adicionando contagens e intervalos de tempo à linha de comando. Por exemplo:

```
# sar -n DEV 5 2
Linux 2.6.32-220.7.1.el6.x86_64 (cnegus.csb) 04/17/2012 _x86_64_ (4 CPU)
11:19:36 PM IFACE rxpck/s txpck/s rxkB/s txkB/s rxcmp/s txcmp/s rxcst/s
```

```

11:19:41 PM lo 5.42 5.42 1.06 1.06 0.00 0.00 0.00
11:19:41 PM eth0 0.00 0.00 0.00 0.00 0.00 0.00 0.00
11:19:41 PM wlan0 1.00 1.00 0.10 0.12 0.00 0.00 0.00
11:19:41 PM pan0 0.00 0.00 0.00 0.00 0.00 0.00 0.00
11:19:41 PM tun0 0.00 0.00 0.00 0.00 0.00 0.00 0.00
...
Average: IFACE rxpck/s txpck/s rxkB/s txkB/ rxcmp/s txcmp/s rxcst/s
Average: lo 7.21 7.21 1.42 1.42 0.00 0.00 0.00
Average: eth0 0.00 0.00 0.00 0.00 0.00 0.00 0.00
Average: wlan0 4.70 4.00 4.81 0.63 0.00 0.00 0.00
Average: pan0 0.00 0.00 0.00 0.00 0.00 0.00 0.00
Average: tun0 3.70 2.90 4.42 0.19 0.00 0.00 0.00

```

Com o exemplo de `-n Dev` que acabamos de mostrar, você pode ver quanta atividade ocorre nas diferentes interfaces de rede em seu sistema. Você pode ver quantos pacotes e quantos KB de dados foram transmitidos e recebidos. Nesse exemplo, amostras de dados foram realizadas em intervalos de 5 segundos e repetidas duas vezes.

Consulte as páginas man de `sar`, `sa1` e `sa2` para obter mais informações sobre como os dados de `sar` podem ser coletados e exibidos.

Verificando o espaço do sistema

Enquanto `logwatch` pode fornecer um instantâneo diário do consumo de espaço em discos de seu sistema, os comandos `df` e `du` podem ajudá-lo a imediatamente ver quanto espaço em disco está disponível. As seções a seguir mostram exemplos desses comandos.

Exibindo espaço em disco do sistema com df

Você pode exibir o espaço disponível em seu sistema de arquivos usando o comando `df`. Para ver a quantidade de espaço disponível em todos os sistemas de arquivos montados em seu computador Linux, digite `df` sem opções:

```

$ df
Filesystem 1k-blocks    Used Available Use% Mounted on
/dev/sda3   30645460 2958356 26130408  11% /
/dev/sda2    46668      8340    35919    19% /boot

```

Essa saída de exemplo mostra o espaço disponível na partição do disco rígido montado no diretório / (root) (`/dev/sda1`) e da partição `/boot` (`/dev/sda2`). O espaço em disco é mostrado em blocos de 1KB. Para produzir uma saída mais legível, use a opção `-h`:

```

$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda3       29G  2.9G   24G  11% /
/dev/sda2       46M  8.2M   25M  19% /boot

```

Com a opção `df -h`, a saída aparece em uma amigável listagem em megabytes ou gigabytes. Outras opções com `df` permitem fazer o seguinte:

- Imprimir apenas sistemas de arquivos de um tipo específico (`-t type`)
- Excluir sistemas de arquivos de um determinado tipo (`-x type`)

- Incluir sistemas de arquivos que não têm espaço, como `/proc` e `/dev/pts` (`-a`)
- Listar apenas inodes disponíveis e usados (`-i`)
- Mostrar espaço em disco em certos tamanhos de bloco (`--block-size=#`)

Verificando o uso do disco com du

Para saber quanto espaço está sendo consumido por um determinado diretório (e seus subdiretórios), use o comando `du`. Sem opções, `du` lista todos os diretórios abaixo do diretório atual, juntamente com o espaço consumido por cada diretório. No final, `du` exibe o espaço em disco total utilizado dentro dessa estrutura de diretórios.

O comando `du` é uma boa maneira de verificar quanto espaço está sendo usado por um determinado usuário (`du /home/user1`) ou em uma partição específica do sistema de arquivos (`du /var`). Por padrão, o espaço em disco é exibido em blocos de 1KB. Para tornar a saída mais amigável (em kilobytes, megabytes e gigabytes), use a opção `-h` como segue:

```
$ du -h /home/jake
114k      /home/jake/httpd/stuff
234k      /home/jake/httpd
137k      /home/jake/uucp/data
701k      /home/jake/uucp
1.0M      /home/jake
```

O resultado mostra o espaço em disco utilizado em cada diretório sob o diretório inicial do usuário chamado Jake (`/home/jake`). O espaço em disco consumido é mostrado em kilobytes (k) e megabytes (M). O espaço total consumido por `/home/jake` é mostrado na última linha. Adicione a opção `-s` para ver o espaço em disco total utilizado para um diretório e seus subdiretórios.

Descobrimo o consumo em disco com find

O comando `find` é uma ótima maneira de encontrar o consumo de arquivo de seu disco rígido, usando uma variedade de critérios. Você pode ter uma boa ideia do local onde o espaço em disco pode ser recuperado localizando arquivos acima de um certo tamanho ou que foram criados por uma pessoa particular.

ta

deve ser o usuário `root` para executar esse comando efetivamente, a menos que você esteja apenas verificando arquivos pessoais. Se você não for o usuário `root`, haverá muitos lugares no sistema de arquivos que você não tem permissão para verificar. Os usuários regulares normalmente podem verificar seus próprios diretórios iniciais, mas não os dos outros.

No exemplo a seguir, o comando `find` pesquisa o sistema de arquivos raiz (`/`) em busca de todos os arquivos de propriedade do usuário chamado Jake (`-user jake`) e imprime o nome dos arquivos. A saída do comando `find` é organizada em uma listagem por ordem de tamanho (`ls -ldS`). Por fim, essa saída é enviada para o arquivo `/tmp/jake`. Quando você visualizar o arquivo `/tmp/jake` (por exemplo, `less /tmp/jake`), você vai encontrar todos os arquivos que são de propriedade do usuário `jake` listados por ordem de tamanho. Eis a linha de comando:

```
# find / -xdev -user jake -print | xargs ls -ldS > /tmp/jake
```

ca

ção `-xdev` impede que outros sistemas de arquivos sejam pesquisados. Essa é uma boa maneira de cortar um monte de lixo que pode ser a saída do sistema de arquivos `/proc`. Isso também pode evitar que grandes sistemas de arquivos remotamente montados sejam pesquisados.

Eis outro exemplo, só que em vez de examinar arquivos de um usuário, estamos à procura de arquivos maiores que 100 kilobytes (`-size +100k`):

```
# find / -xdev -size +100k -print | xargs ls -ldS > /tmp/size
```

Você pode economizar uma grande quantidade de espaço em disco removendo apenas alguns arquivos maiores que não são mais necessários. Nesse exemplo, você pode ver que arquivos grandes são classificados por tamanho no arquivo `/tmp/size`.

Resumo

Embora muitos tipos diferentes de servidores estejam disponíveis em sistemas Linux, o procedimento básico para a instalação e configuração de um servidor é essencialmente o mesmo. O curso normal dos eventos é instalar, configurar, iniciar, proteger e monitorar seus servidores. Tarefas básicas que se aplicam a todos os servidores incluem o uso de ferramentas de rede (especialmente ferramentas SSH) para efetuar o login, copiar arquivos ou executar comandos remotos.

Como um administrador não pode ficar conectado para monitorar servidores o tempo todo, mais tarde, as ferramentas de coleta de dados e análise dos dados de registro em log são muito importantes na administração de servidores Linux. A instalação `rsyslog` pode ser utilizada para o registro em log local e remoto. O recurso `sar` reúne dados ao vivo ou reproduz dados coletados anteriormente em intervalos de 10 minutos. Para monitorar o espaço em disco, você pode executar os comandos `df` e `du`.

Embora seja fácil configurar a rede para alcançar seus servidores em casos simples e padrões, uma configuração de rede mais complexa requer um conhecimento dos arquivos de configuração de rede e ferramentas relacionados. O próximo capítulo descreve como configurar e administrar redes no Linux.

Exercícios

Os exercícios desta seção abordam algumas das ferramentas básicas para conectar-se e monitorar seus servidores Linux. Como de costume, muitas vezes há várias maneiras de realizar as tarefas aqui. Então, não se preocupe se você não acompanhar os exercícios da mesma maneira mostrada nas respostas, desde que você obtenha os mesmos resultados. Se você empacar, as soluções para as tarefas são mostrados no Apêndice B.

Alguns dos exercícios pressupõem que você tem um segundo sistema Linux disponível em que pode fazer login e experimentar diferentes comandos. Nesse segundo sistema, você precisa ter certeza de que o serviço `sshd` está em execução, que o firewall está aberto e que o `ssh` está habilitado para a conta de usuário que você está tentando acessar (root costuma ser bloqueado por `sshd`).

Se tiver apenas um sistema Linux, você pode criar uma conta de usuário adicional e simplesmente simular a comunicação com outro sistema, conectando-se ao nome `localhost` em vez de ao outro sistema. Por exemplo:

```
# useradd joe
# passwd joe
# ssh joe@localhost
```

1. Usando o comando `ssh`, faça login em outro computador (ou o computador local) usando qualquer conta a que você tem acesso. Digite a senha quando solicitado.
2. Usando execução remota com o comando `ssh`, exiba o conteúdo de um arquivo `/etc/system-release` remoto e exiba seu conteúdo no sistema local.
3. Use o comando `ssh` para usar o encaminhamento X11 a fim de exibir uma janela `gedit` em seu sistema local; então, salve um arquivo no diretório inicial do usuário remoto.
4. Copie recursivamente todos os arquivos do diretório `/usr/share/selinux` em um sistema remoto para o diretório `/tmp` em seu sistema local de tal maneira que todas as datas/horas de modificação nos arquivos sejam atualizados para a data/hora do sistema local quando eles forem copiados.
5. Copie recursivamente todos os arquivos do diretório `/usr/share/logwatch`, em um sistema remoto, para o diretório `/tmp`, em seu sistema local, de tal maneira que todas as datas/horas de modificação nos arquivos do sistema remoto sejam mantidas no sistema local.
6. Criar um par de chaves pública/privada para usar para comunicações SSH (sem senha na chave), copie o arquivo de chave pública para a conta de um usuário remoto com `ssh-copy-id` e use autenticação baseada em chave para fazer login nessa conta do usuário sem ter de digitar uma senha.
7. Crie uma entrada em `/etc/rsyslog.conf` que armazene todas as mensagens de autenticação (`authpriv`) de nível informativo e acima em um arquivo chamado `/var/log/myauth`. A partir de um terminal, monitore o arquivo à medida que dados

cheguem a ele e, em outro terminal, tente se conectar usando `ssh` em sua máquina local como qualquer usuário válido, com uma senha incorreta.

8. Use o comando `du` para determinar as maiores estruturas de diretórios sob `/usr/share`, classifique os diretórios do maior para o menor e liste os dez primeiros desses diretórios em termos de tamanho.
9. Use o comando `df` para mostrar o espaço que está em uso e o espaço disponível de todos os sistemas de arquivos atualmente conectados ao sistema local, mas exclua quaisquer sistemas de arquivos `tmpfs` ou `devtmpfs`.
10. Encontrar todos os arquivos no diretório `/usr` que têm mais de 10 MB.

Administrando redes

NESTE CAPÍTULO

Conectando-se automaticamente a uma rede Linux

Usando o NetworkManager para conectividade de rede simples

Configurando servidores

Trabalhando com arquivos de configuração de rede

Configurando roteamento, DHCP, DNS e recursos de infraestrutura de redes para a empresa

Conectar um único sistema desktop ou laptop a uma rede, especialmente uma que se conecta à internet, tornou-se tão fácil que achei que poderia adiar um capítulo inteiro sobre redes Linux para até este ponto no livro. Se você está tentando conectar um sistema desktop Fedora, RHEL, Ubuntu ou outro sistema desktop Linux à internet, eis o que você pode tentar, tendo disponível uma placa de rede com ou sem fio:

- **Rede com fio** — Se sua casa ou escritório tem uma porta Ethernet com fio que fornece um caminho para a internet e seu computador tem uma porta Ethernet, use um cabo Ethernet para conectar as duas portas. Depois de conectar o computador, inicialize o Linux e faça login. Clicar no ícone NetworkManager no desktop deve mostrar-lhe que você está conectado à internet.
- **Rede sem fio** — Para um computador sem fio executando Linux, faça o login e clique no ícone NetworkManager no desktop. A partir da lista de redes sem fio que aparecem, selecione a que você quer e, quando solicitado, digite a senha necessária. Sempre que você fizer login a partir desse computador e a partir do mesmo local, ele se conectará automaticamente à rede sem fio.

Se um desses tipos de conexões de rede funciona para você e você não tem curiosidade sobre como a rede funciona em Linux, isso pode ser tudo o que precisa saber. Mas e se seu sistema Linux não se conectar automaticamente à internet? E se você quiser configurar seu desktop para se comunicar com uma rede privada no trabalho (VPN)? E se você quiser bloquear as configurações de rede em seu servidor ou configurar o sistema Linux para funcionar como um roteador?

Neste capítulo, os temas relacionados com a rede são divididos em rede para desktops, servidores e computação corporativa. A abordagem geral para a configuração de rede nesses três tipos de sistemas Linux é como segue:

- **Rede desktop/laptop** — Em sistemas desktop, o NetworkManager é executado por padrão para gerenciar placas de rede. Com o NetworkManager, você pode aceitar automaticamente o endereço do servidor e as informações necessárias para se conectar à internet. Mas você também pode definir informações de endereço manualmente. Você pode configurar coisas como servidores proxy ou conexões de rede virtual privada para permitir que seu desktop funcione atrás de um firewall da organização ou se conecte por meio de um firewall, respectivamente.
- **Servidor de rede** — Embora o NetworkManager seja um excelente serviço para configuração de rede desktop e laptop, ele não funciona tão bem em servidores. Para configurar uma rede em servidores, descrevo como usar o serviço de rede básica no Red Hat Enterprise, incluindo a configuração de aliases e ligação de canal Ethernet, bem como a forma de usar os arquivos de configuração de rede subjacentes diretamente.
- **Rede corporativa** — A configuração de rede em uma grande empresa é um assunto tão vasto que pode ocupar vários livros por si só. Mas para dar-lhe uma boa introdução no uso do Linux em um ambiente corporativo, discuto tecnologias de rede básicas, tais como DHCP e DNS, que tornam possível que sistemas de desktop se conectem à internet automaticamente.

Configurando uma rede para desktops

Se você se conectar à internet a partir do Linux, Windows, um smartphone ou qualquer outro tipo de dispositivo habilitado para rede, há certas coisas que devem estar no lugar para que a conexão funcione. O computador deve ter uma placa de rede (com ou sem fio), um endereço IP, um servidor DNS atribuído e uma rota para a internet (identificada por um dispositivo do tipo gateway).

Antes de discutir como mudar sua configuração de rede em Linux, vamos examinar as atividades gerais que ocorrem quando o Linux é configurado para se conectar automaticamente à internet com o NetworkManager:

- **Ativação das placas de rede** — O NetworkManager procura ver quais placas de rede (com ou sem fio) estão configuradas para iniciar. Por padrão, as placas externas são configuradas para iniciar automaticamente usando DHCP.
- **Solicitação do serviço DHCP** — O sistema Linux atua como um cliente DHCP para enviar uma solicitação ao serviço DHCP em cada placa habilitada. Ele usa o endereço MAC da placa de rede para identificar-se na solicitação.
- **Obtenção da resposta do servidor DHCP** — Um servidor DHCP, possivelmente em execução no modem DSL, modem a cabo ou outro dispositivo fornecendo uma rota para a internet a partir de sua localização, responde à solicitação DHCP. Ele pode fornecer muitos diferentes tipos de informação para o cliente DHCP. Essas informações provavelmente contém, pelo menos, o seguinte:
 - **Endereço IP** — O servidor DHCP normalmente tem intervalo de endereços IP (internet Protocol) que ele pode entregar a qualquer sistema na rede que solicita um endereço. Em ambientes mais seguros ou um em que você quer garantir que determinadas máquinas obtenham endereços específicos, o servidor DHCP fornece um endereço IP especial para solicitações de

endereços MAC específicos. (Endereços MAC são projetados para serem únicos entre todas as interfaces de placas de rede e são atribuídos pelo fabricante de cada placa.)

- **Máscara de sub-rede** — Quando o cliente DHCP recebe um endereço IP, a máscara de sub-rede que o acompanha diz a esse cliente qual parte do endereço IP identifica a sub-rede e qual identifica o host. Por exemplo, um endereço IP de 192.168.0.100 e uma máscara de sub-rede de 255.255.255.0 diz ao cliente que a rede é 192.168.0 e a parte de host é 100.
- **Alocação de tempo** — Quando um endereço IP é alocado dinamicamente para o cliente DHCP (Linux), esse cliente recebe um tempo de alocação. O cliente não possui endereço, mas deve alocá-lo novamente quando o tempo expirar e solicitá-lo novamente ao reiniciar a placa de rede. Normalmente, o servidor DHCP vai se lembrar do cliente e atribuir o mesmo endereço quando o sistema for iniciado novamente ou solicitar para renovar a alocação. O tempo de alocação padrão é 86.400 segundos (24 horas).
- **Domain Name Server** — Como os computadores gostam de pensar em números (por exemplo, endereços IP como 192.168.0.100) e as pessoas tendem a pensar em nomes (como o hostname `http://www.example.com`), os computadores precisam de uma maneira de traduzir hostnames em endereços IP e, às vezes, o inverso também. O sistema de nomes de domínio (*domain name system*, DNS) foi projetado para lidar com esse problema, fornecendo uma hierarquia de servidores para fazer o mapeamento de nome para endereço na internet. A localização de um ou mais servidores DNS (geralmente dois ou três) é geralmente atribuída ao cliente DHCP do host DHCP.
- **Gateway padrão** — Embora a internet tenha um espaço de nomes único, na verdade ela é organizado como uma série de sub-redes interconectadas. Para uma solicitação de rede deixar sua rede local, ela deve saber o nó em que sua rede fornece uma rota para endereços fora de sua rede local. O servidor DHCP normalmente fornece o endereço IP do “gateway padrão”. Com placas de rede na sua sub-rede e na próxima rede no caminho até o destino final de sua comunicação, um gateway pode rotear seus pacotes para o destino.
- **Outras informações** — Um servidor DHCP pode ser configurado para fornecer todo tipo de informação para ajudar o cliente DHCP. Por exemplo, ela pode fornecer a localização de um servidor NTP (para sincronizar o tempo entre clientes), servidor de fontes (para obter fontes para seu monitor X), servidor de IRC (para bate-papos online) ou servidor de impressão (para designar as impressoras disponíveis).
- **Atualização das configurações de rede local** — Depois que as configurações são recebidas a partir do servidor DHCP, elas são implementadas conforme apropriado no sistema Linux local. Por exemplo, o endereço IP é definido na placa de rede, as entradas do servidor DNS são adicionadas ao arquivo local `/etc/resolv.conf` (pelo NetworkManager) e o tempo de alocação é armazenado pelo sistema local, assim ele sabe quando solicitar que a alocação seja renovada.

Todos os passos acima descritos tipicamente acontecem sem que você precise fazer nada além de conectar o sistema Linux e fazer login. Digamos que você quer ser capaz de verificar as placas de rede ou alterar alguma dessas configurações. Você pode fazer isso usando as ferramentas descritas nas próximas seções.

Verificando suas placas de rede

Há tanto ferramentas gráficas como ferramentas de linha de comando para visualizar informações sobre as placas de rede no Linux. A partir do desktop, as ferramentas NetworkManager são um bom lugar para começar.

Verificando sua rede a partir do NetworkManager

A maneira mais fácil de verificar a configuração básica de uma placa de rede iniciada pelo NetworkManager é abrir o ícone do NetworkManager em seu desktop. A Figura 14.1 mostra um exemplo do ícone NetworkManager no painel superior de um desktop GNOME 3 no Fedora, juntamente com a janela que aparece quando você abre o ícone.

cando as placas de rede com o NetworkManager

Como você pode ver na Figura 14.1, a conexão de rede com fio está ativa. A placa de rede possui um endereço MAC (Media Access Control) 00:1B21:0A:E8:5E. A placa recebeu o endereço IP 192.168.0.119 e a máscara de sub-rede 255.255.255.0.

Qualquer pacote não destinado à rede local, é enviado para o roteador localizado no endereço 192.168.0.1 (embora outras vias possam ser definidas conforme necessário). Três servidores DNS estão disponíveis (192.152.128.23, 192.168.1.254 e 192.168.0.1); portanto, se um não estiver disponível, as solicitações de serviço de DNS podem ser direcionadas para o próximo endereço do servidor DNS na lista.

Para ver mais sobre como o sistema Linux está configurado, clique no botão Configure na janela NetworkManager. A Figura 14.2 mostra um exemplo da janela que aparece.

lizando as configurações de rede com o NetworkManager

A figura 14.2 mostra a guia IPv4 Settings, porque essa é a guia que contém informações que mais provavelmente você vai querer modificar. A configuração Automatic (DHCP) é a que diz a essa placa para conectar ao DHCP na inicialização; portanto, talvez você queira mudar isso para definir manualmente as informações de IPv4. O guia IPv6 define como são feitas as conexões com redes IPv6 a partir dessa placa (também Automatic, por padrão). Mais tarde, neste capítulo, você aprenderá a configurar manualmente placas de rede IPv4.

A guia 802.1 Security permite configurar conexões seguras com sistemas remotos utilizando as especificações 802.1x do padrão IEEE. Por padrão, esse recurso está desativado, mas você pode ativá-lo e, então, identificar os hosts remotos a que você quer se conectar usando protocolos seguros. A última guia na janela é Wired or Wireless, ela permite que você altere o endereço MAC (normalmente não é uma boa ideia) e o MTU (que é definido como automático, mas pode ser alterado para modificar o tamanho máximo de pacote que a conexão pode enviar).

Verificando sua rede a partir da linha de comando

Para obter informações mais detalhadas sobre as placas de rede, tente executar alguns comandos. Há comandos que podem mostrar informações sobre as placas de rede, rotas, hosts e tráfego na rede.

Visualizando placas de rede

Para ver informações sobre cada placa de rede em seu sistema local Linux, digite o seguinte:

```
# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc
    pfifo_fast
        state DOWN qlen 1000
    link/ether f0:de:f1:28:46:d9 brd ff:ff:ff:ff:ff:ff
3: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state
    UP
    qlen 1000
    link/ether 00:24:d7:69:5b:2c brd ff:ff:ff:ff:ff:ff
    inet 192.168.0.105/24 brd 192.168.0.255 scope global wlan0
    inet6 fe80::224:d7ff:fe69:5b2c/64 scope link
        valid_lft forever preferred_lft forever
```

A saída de `ip addr show` exibe informações sobre as placas de rede, neste caso, de um laptop rodando RHEL. Os nomes das placas de rede são diferentes no Fedora (mais sobre isso depois), mas, fora isso, o resto deve ser semelhante. A entrada `lo` na primeira linha da saída mostra a placa de loopback (autorretorno), que é utilizada para permitir que os comandos de rede sejam executados no sistema local para se conectar ao sistema local. O endereço IP para localhost é 127.0.0.1/8 (o /8 é a notação CIDR, indicando que 127.0 é o número de rede e 0.1 é o número do host).

Nesse caso, a placa Ethernet (`eth0`) com fio está inativa (sem cabo), mas a placa sem fio está ativa (`wlan0`). O endereço MAC da placa wireless (`wlan0`) é 00:24:d7:69:5b:2c e o endereço internet (IPv4) é 192.168.0.105. Um endereço IPv6 também é ativado.

No Fedora, em vez de atribuir nomes de placa de rede, como `eth0` e `wlan0`, as placas são nomeadas por suas localizações no barramento do computador. Por exemplo, a primeira porta na placa de rede instalada no terceiro barramento PCI de um sistema Fedora é nomeada como `p3p1`. A primeira porta Ethernet interna seria `em1`. Às vezes, placas sem fio aparecem usando o nome da rede sem fio como o nome do dispositivo.

Outro comando popular para ver informações sobre placas de rede é o `ifconfig`. Por padrão, `ifconfig` mostra informações semelhantes às de `ip addr`, mas `ifconfig` também mostra o número de pacotes recebidos (RX) e transmitidos (TX), bem como a quantidade de dados e quaisquer erros ou perda de pacotes:

```
# ifconfig wlan0
wlan0 Link encap:Ethernet HWaddr 00:24:D7:69:5B:2C
    inet addr:192.168.0.105 Bcast:192.168.0.255 Mask:255.255.255.0
    inet6 addr: fe80::224:d7ff:fe69:5b2c/64 Scope:Link
    UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
    RX packets:22482 errors:0 dropped:0 overruns:0 frame:0
    TX packets:9699 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:1000
    RX bytes:9456897 (9.0 MiB) TX bytes:1232234 (1.1 MiB)
```

Verificando a conectividade com sistemas remotos

Para se certificar de que você pode chegar a sistemas que estão disponíveis na rede, você pode usar o comando `ping`. Desde que o computador responda a solicitações de `ping` (nem todos respondem), você pode usar esse comando para enviar pacotes para esse sistema de uma maneira que ele seja solicitado a responder. Eis um exemplo:

```
$ ping
pub1.kernel.org
PING pub1.kernel.org (149.20.20.133) 56(84) bytes of data.
64 bytes from pub1.kernel.org (149.20.20.133): icmp_seq=1 ttl=64
time=0.062 ms
64 bytes from pub1.kernel.org (149.20.20.133): icmp_seq=2 ttl=64
time=0.044 ms
^C
```

```
--- publ.kernel.org ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1822ms
rtt min/avg/max/mdev = 0.044/0.053/0.062/0.009 ms
```

O comando `ping` mostrado acima simplesmente “pinga” continuamente o host `publ.kernel.org`. Depois de alguns pings, pressione `Ctrl+C` para acabar com os pings e as últimas linhas mostram quantas solicitações de ping foram bem-sucedidas.

Você poderia ter usado o endereço IP (192.168.0.15, nesse caso) para ver a possibilidade de alcançar o sistema. Mas usar o nome lhe dá a vantagem adicional de saber que sua tradução de nome para endereço IP (que está sendo feita pelo seu servidor DNS ou o arquivo `hosts` local) está funcionando corretamente também.

Verificando as informações de roteamento

O roteamento é a próxima coisa que você pode verificar com relação a suas placas de rede. Eis como usar o comando `route` para fazer isso:

```
# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default 192.168.0.1 0.0.0.0 UG 0 0 0 p4p1
192.168.0.0 * 255.255.255.0 U 1 0 0 p4p1
```

A saída da tabela de roteamento do kernel é de um sistema Fedora com uma única placa de rede. A placa de rede está no slot PCI 4, porta 1 (p4p1). Todos os pacotes destinados à rede 192.168.0 utilizam a placa de rede p4p1. Pacotes destinados a qualquer outro local são encaminhados para o sistema de gateway em 192.168.0.1. Esse sistema representa meu roteador para a internet. Eis uma tabela de roteamento mais complexa:

```
# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
vpn-a.example. 192.168.0.1 255.255.255.255 UGH 0 0 0 wlan0
192.168.0.0 * 255.255.255.0 U 2 0 0 wlan0
10.99.8.0 * 255.255.255.0 U 0 0 0 tun0
172.1.0.0 * 255.255.0.0 U 0 0 0 tun0
10.0.0.0 * 255.0.0.0 U 0 0 0 tun0
192.168.99.0 192.168.0.2 255.255.255.0 UG 0 0 0 wlan0
default 192.168.0.1 0.0.0.0 UG 0 0 0 wlan0
```

No exemplo de roteamento mostrado acima, há uma placa sem fio (`wlan0`), bem como uma placa que representa um túnel de rede privada virtual (VPN). Uma VPN fornece uma maneira de ter comunicações privadas criptografadas entre um cliente e uma rede remota através de uma rede insegura (como a internet). Aqui, o túnel vai do sistema local pela placa `wlan0` para um host chamado `vpn-a.example.com` (parte do nome está truncada).

Toda a comunicação com a rede 192.168.0.0/24 ainda passa diretamente pela rede local sem fio. Mas os pacotes destinados às redes 10.99.8.0/24, 172.1.0.0/16 e 10.0.0.0/8 são encaminhados diretamente para `vpn-a.example.com` para comunicação com os hosts do outro lado da conexão VPN através da placa tunelada (`tun0`).

Uma rota especial para a rede 192.168.99.0 é acessível por meio do nó (presumivelmente um roteador) no endereço IP 192.168.0.2. Todos os outros pacotes vão para a rota padrão por meio do endereço

192.168.0.1. Como ocorre com os *flags* (sinalizadores) mostrados na saída, um U (*up*) informa que a rota está ativa, um G identifica a placa como um gateway e um H diz que o alvo é um host (como é o caso com a conexão VPN).

Até agora, só lhe mostrei as rotas para deixar o sistema local. Se quiser seguir toda a rota de um host do começo ao fim, você pode usar o comando `traceroute`. Por exemplo, para traçar a rota que um pacote leva de seu sistema local até o site `Google.com`, digite o seguinte comando `traceroute`:

```
# traceroute google.com
traceroute to google.com (74.125.235.136), 30 hops max, 60 byte
packets
...
 7 rrcs-70-62-95-197.midsouth.biz.rr.com (70.62.95.197) ...
 8 ge-2-1-0.rlghncpop-rtr1.southeast.rr.com (24.93.73.62) ...
 9 ae-3-0.cr0.dca10.tbone.rr.com (66.109.6.80) ...
10 107.14.19.133 (107.14.19.133) 13.662 ms ...
11 74.125.49.181 (74.125.49.181) 13.912 ms ...
12 209.85.252.80 (209.85.252.80) 61.265 ms ...
13 66.249.95.149 (66.249.95.149) 18.308 ms ...
14 66.249.94.22 (66.249.94.22) 18.344 ms ...
15 72.14.239.83 (72.14.239.83) 85.342 ms ...
16 64.233.174.177 (64.233.174.177) 167.827 ms ...
17 209.85.255.35 (209.85.255.35) 169.995 ms ...
18 209.85.241.129 (209.85.241.129) 170.322 ms ...
19 nrt19s11-in-f8.1e100.net (74.125.235.136) 169.360 ms ...
```

Eu trunquei parte da saída para descartar algumas das rotas iniciais e a quantidade de tempo (em milissegundos) que os pacotes estavam levando para percorrer cada rota. Usando `traceroute`, você pode ver quais são os gargalos ao longo do caminho, se sua comunicação de rede estiver ficando lenta.

Visualizando os nomes de host e de domínio

Para ver o hostname atribuído ao sistema local, digite `hostname`. Para ver apenas a parte do domínio desse nome, use o comando `dnsdomainname`.

```
# hostname
spike.example.com
# dnsdomainname
example.com
```

Configurar interfaces de rede

Se não quiser ter suas placas de rede atribuídas automaticamente a partir de um servidor DHCP (ou se não houver nenhum servidor DHCP), você pode atribuir endereços IP e informações relacionadas usando endereços IP estáticos e outras informações necessárias para seu computador funcionar corretamente na

rede. A maioria dessas informações básicas podem ser configuradas usando o NetworkManager.

Para alterar a configuração de rede por meio do NetworkManager, faça o seguinte:

1. Selecione o ícone do NetworkManager no painel superior do desktop e escolha Network Settings.
2. Selecione a placa que pretende alterar (com ou sem fio) e clique no botão Configure.
3. Escolha IPv4 Settings e altere o método de Automatic (DHCP) para Manual.
4. Selecione Add e preencha as seguintes informações:
 - **Address** — O endereço IP que você deseja atribuir à sua placa de rede local. Por exemplo, 192.168.0.40.
 - **Netmask** — A máscara de sub-rede que define qual parte do endereço IP representa a rede e qual representa o hospedeiro. Por exemplo, uma máscara de rede de 255.255.255.0 identificaria a parte rede do endereço anterior como 192.168.0 e a parte host como 40.
 - **Gateway** — O endereço IP do computador ou dispositivo na rede que funciona como a rota padrão. A rota padrão roteará pacotes da rede local para qualquer endereço que não está disponível na rede local ou via outra rota personalizada.
 - **DNS servers** — Preencha os endereços IP para os sistemas que fornecem serviço de DNS para seu computador.
 - **Search domains** — Preencha todos os nomes de domínio dos domínios que você gostaria que o sistema pesquisasse se você inserir apenas um hostname ao executar um comando. Por exemplo, com um domínio de pesquisa de `example.com`, se você digitasse `ping spike`, o sistema tentaria emitir um `ping` para o sistema `spike.example.com`.
 - **Save** — Clique no botão Save. As novas informações são salvas e a rede é reiniciada com elas. A Figura 14.3 mostra um exemplo dessas configurações de rede.

ando configurações de rede com o NetworkManager

Você pode olhar na janela NetworkManager ou executar um comando `ifconfig` ou `ip addr` descrito em “Visualizando as placas de rede”, anteriormente, neste capítulo, para testar se as novas configurações de rede tiveram efeito.

Configurando uma conexão de rede proxy

Se seu sistema desktop está executando atrás de um firewall corporativo, talvez você não tenha acesso direto à internet. Em vez disso, você pode ter de acessar a internet por meio de um servidor proxy. Em vez de permitir-lhe acesso total à internet, um servidor proxy permite que você faça solicitações apenas para determinados serviços fora da rede local. O servidor proxy passa as solicitações para a internet ou outra rede.

Os servidores proxy normalmente fornecem acesso a servidores web (`http://` e `https://`) e servidores de FTP (`ftp://`). Mas um servidor proxy que suporta SOCKS pode fornecer um serviço de proxy para diferentes protocolos fora da rede local. (SOCKS é um protocolo de rede feito para permitir que computadores clientes acessem a internet por meio de um firewall.) Você pode identificar um servidor proxy no NetworkManager e fazer as comunicações com protocolos selecionados passarem por esse servidor (selecione Network proxy na janela Network Settings).

Em vez de identificar um servidor proxy para suas placas de rede (via NetworkManager), você pode configurar seu navegador para usar um servidor proxy diretamente alterando as preferências do Firefox. Veja como definir um servidor proxy a partir da janela do Firefox:

1. No Firefox, selecione Edit Preferences. A janela Firefox Preferences aparece.
2. A partir da janela Firefox Preferences, clique no botão Advanced.
3. Escolha a guia Network e escolha o botão Settings sob o título Connection. A janela Configure Proxies aparece.
4. Você pode tentar detectar automaticamente as configurações de proxy ou, se configurar o proxy no NetworkManager, pode optar por usar as configurações de proxy do sistema. Você também pode selecionar Manual Proxy Configuration, preencher as seguintes informações e clicar em OK.
 - **HTTP Proxy** — O endereço de IP do computador que oferece o serviço de proxy. Isso faz com que todas as solicitações de páginas web (protocolo `http://`) sejam encaminhadas para o servidor proxy.
 - **Port** — A porta associada ao serviço de proxy. Por padrão, o número da porta é 3128, mas pode variar.
 - **Use this proxy server for all protocols** — Marque essa caixa para usar o mesmo servidor

proxy e a mesma porta associada com o HTTP proxy para todas as solicitações de outros serviços. Isso faz com que as outras configurações de proxy fiquem desabilitadas. (Em vez de selecionar essa caixa, você pode definir os serviços de proxy separadamente.)

- **No Proxy for** — Deixando localhost e o endereço IP local (127.0.0.1) nessa caixa, todas as solicitações para o sistema local, que de outra forma seriam direcionadas para o servidor proxy, vão diretamente para o sistema local.

A Figura 14.4 ilustra um exemplo da janela Configure Proxies preenchida para configurar uma conexão com um servidor proxy localizado no endereço IP 10.0.100.254 para todos os protocolos. Após clicar em OK, todas as solicitações do navegador Firefox para locais fora do sistema local são direcionadas para o servidor proxy, que as encaminha para o servidor apropriado.

gurando o Firefox para usar um servidor proxy

Configurando redes para servidores

Embora o NetworkManager faça um ótimo trabalho autodetectando redes com fio ou apresentando-lhe listas de redes sem fio para seu laptop se conectar, ele é menos adequado para configurar redes em servidores. Assim, nesta seção, você irá desativar o NetworkManager, ativar o serviço de rede e fazer os procedimentos desta seção, trabalhando diretamente com comandos e arquivos de configuração de rede.

Em particular, nesta seção, você vai fazer o seguinte:

- **Configuração básica** — Veja como usar `system-config-network` para configurar uma rede básica, com uma interface baseada em menus.
- **Arquivos de configuração** — Entenda arquivos de configuração associados com redes Linux e como configurá-los diretamente.
- **Alias** — Configure vários endereços nas mesmas placas, de modo que uma placa de rede possa ter vários endereços atribuídos a ela (aliases).
- **Agregação de canal Ethernet** — Configure uma agregação de canal Ethernet (várias placas de rede ouvindo o mesmo endereço IP).
- **Rotas personalizadas** — Defina rotas personalizadas.

Como mencionado anteriormente, eu recomendo que você desative o NetworkManager na maioria dos casos, ao configurar a rede em um servidor. Para fazer isso no RHEL, Fedora ou um sistema mais antigo, digite o seguinte como root (faça isso em um console, já que esse procedimento vai derrubar sua rede se você estiver conectado):

```
# service NetworkManager stop
# service network restart
# chkconfig NetworkManager off
# chkconfig network on
```

Para sistemas mais recentes do Fedora, que utilizam o comando `systemctl` para iniciar, você pode digitar o seguinte para ativar e desativar serviços:

```
# systemctl stop NetworkManager.service
# systemctl disable NetworkManager.service
# service network restart
```

```
# chkconfig network on
```

Utilizando system-config-network

Muitos servidores não têm interfaces gráficas disponíveis. Então, se você quiser configurar a rede, deve ser capaz de fazê-lo a partir do shell. Uma maneira de fazer isso é editar arquivos de configuração de rede diretamente. Outra é usar `system-config-network`.

Antes de o NetworkManager existir, `system-config-network` carregava uma interface gráfica que era a principal forma de configurar a rede no Fedora e no Red Hat Enterprise Linux. Nas últimas versões desses sistemas Linux, `system-config-network` fornece uma placa baseada em menus que roda no shell. Como root, digite `system-config-network` para ver uma tela semelhante à mostrada na Figura 14.5.

gurando uma rede com system-config-network

Use as setas e a tecla TAB para navegar pela interface. Com o item que deseja selecionar destacado, pressione Enter para selecioná-lo. A interface limita-se a alterar dois tipos de informação: configuração de dispositivo (placas de rede) e configuração de DNS (conversão de nomes).

ta

É possível que suas placas de rede não apareçam quando você selecionar Device configuration. Há várias razões possíveis para isso. O mais provável é que se trata de uma placa de rede sem fios que necessita de um controlador que não está disponível para o Fedora. Execute o comando `dmesg` logo após a inicialização do sistema e procure o erro do modelo de uma placa Ethernet para a qual o Linux não conseguiu encontrar um driver. Então, pesquise na Internet esse modelo e a palavra Linux. Outra possibilidade é que, em alguns laptops, haja um minúsculo interruptor que pode desativar o laptop. Certifique-se de que esse interruptor está na posição ligado e reinicie a rede. A maioria das placas Ethernet simplesmente funciona.

Escolhendo a configuração do dispositivo

Com configuração do dispositivo destacada, pressione Enter. Uma lista de dispositivos de rede (geralmente placas Ethernet com fio ou sem fio) é exibida. Destaque um dos dispositivos de rede e pressione Enter. A janela Network Configuration que aparece permite que você altere as informações relativas ao dispositivo de rede selecionado. A Figura 14.6 mostra um exemplo.

DHCP ou defina endereços IP estáticos.

Você pode deixar os campos Name e Device como estão. Por padrão, um asterisco aparece no campo Use DHCP. Isso é o que permite que a rede apareça automaticamente na rede se um serviço de DHCP estiver disponível. Para inserir o endereço e outras informações você mesmo, use a tecla Tab para destacar o campo Use DHCP e, então, pressione a barra de espaço para desativar o DHCP. A figura mostra a tela depois que o asterisco foi removido.

Agora, preencha as informações de endereço (endereço IP e máscara de rede). Digite o endereço IP do computador ou roteador que está fornecendo o caminho para a internet. Então, você pode digitar os endereços IP dos servidores de um ou dois DNS, para dizer ao sistema aonde ir para traduzir nomes de máquinas que você requisita em endereços IP. Pressione Tab até alcançar o botão OK e pressione a barra de espaço. Então, clique em Save e, então, em Save & Quit.

Escolhendo a configuração do DNS

Com a configuração de DNS destacada, pressione Enter. Na tela de configuração de DNS, você pode definir o hostname do servidor, os servidores DNS e o caminho de pesquisa DNS. A Figura 14.7 mostra um exemplo dessa tela.

gurando o hostname e as informações de DNS

O hostname pode ser qualquer nome que você quiser ou o hostname completamente qualificado do sistema. Você pode definir os endereços IP de até três servidores DNS. Quando você digita um hostname (em um navegador web ou outro aplicativo), o primeiro servidor DNS é consultado quanto ao endereço IP do host solicitado. Se esse servidor DNS estiver inativo, o servidor DNS secundário é verificado (então, o DNS terciário é verificado se o secundário não for encontrado).

O último campo contém o caminho de pesquisa DNS. O caminho de pesquisa DNS é usado quando você solicita um host a partir de um aplicativo sem usar um nome de domínio completamente qualificado. Por exemplo, se você digitar `ping host1` com um caminho de pesquisa `example.com`, o comando tenta enviar pacotes ping para `host1.example.com`.

Clique em OK para salvar as informações. Então, clique em Save & Quit. As informações são gravadas em vários arquivos de configuração diferentes. Você precisa reiniciar o serviço de rede para que as novas configurações tenham efeito. Se você estiver interessado em quais arquivos de configuração são atualizados, consulte a próxima seção.

Entendendo os arquivos de configuração de rede

Se você mudar sua configuração de rede usando o NetworkManager ou `system-config-network`, a maioria dos mesmos arquivos de configuração é atualizada. No Fedora e no RHEL, placas de rede e rotas personalizadas são configuradas em arquivos no diretório `/etc/sysconfig/network-scripts`. Outras configurações de rede são armazenadas em outros arquivos no diretório `/etc`.

Em vez de usar algumas das ferramentas descritas acima, você também pode configurar a rede no Linux editando diretamente os arquivos de configuração. As seções a seguir descrevem os arquivos de configuração de rede.

Arquivos de configuração de placas de rede

Os arquivos de configuração para cada placa de rede, seja com fio, sem fio, ISDN, dial-up ou outro tipo, são representados por arquivos no diretório `/etc/sysconfig/network-scripts` que começam com `ifcfg-interface`. A palavra *interface* é substituída pelo nome da placa de rede.

No Red Hat Enterprise Linux e sistemas Fedora mais antigos, as placas de rede têm nomes como `eth0`, `eth1`, `eth2` (para cada placa de rede com fio), `wlan0` (para a primeira placa sem fio) e `ppp0` (para a primeira placa dial-up ponto a ponto). No RHEL, o arquivo de configuração para a primeira placa Ethernet seria `ifcfg-eth0`. Eis um exemplo de um arquivo `ifcfg-eth0` para uma conexão DHCP com fio:

```
DEVICE=eth0
```

```
HWADDR=F0:DE:F1:28:46:D9
TYPE=Ethernet
BOOTPROTO=dhcp
ONBOOT=yes
NM_CONTROLLED=no
USERCTL=no
```

Nesse exemplo `ifcfg-eth0`, as três primeiras linhas definem o nome do dispositivo, o endereço MAC e o tipo de placa para Ethernet. A variável `BOOTPROTO` está configurada como `dhcp`, o que faz com que ela solicite informações de endereço a um servidor DHCP. Com `ONBOOT=yes`, a placa inicia automaticamente no momento da inicialização do sistema. Configurar `NM_CONTROLLED=no` garante que o NetworkManager não controle a placa. Outra configuração no exemplo `ifcfg-eth0` impede que usuários regulares sejam capazes de iniciar e parar a placa (`USERCTL=no`).

Eis o que um arquivo `ifcfg-eth1` pode parecer para uma placa Ethernet com fio que usa endereços IP estáticos:

```
DEVICE=eth1
HWADDR=00:1B:21:0A:E8:5E
TYPE=Ethernet
BOOTPROTO=none
ONBOOT=yes
NM_CONTROLLED=no
USERCTL=no
IPADDR=192.168.0.140
NETMASK=255.255.255.0
GATEWAY=192.168.0.1
```

Nesse exemplo de `ifcfg-eth1`, como isso está configurando o endereço e outras informações estaticamente, `BOOTPROTO` é configurado como `none`. Outras diferenças são necessárias para definir as informações de endereço, que são normalmente coletadas a partir de um servidor DHCP. Nesse caso, o endereço IP está configurado como `192.168.0.140`, com uma máscara de rede de `255.255.255.0`. O `GATEWAY=192.168.0.1` identifica o endereço do roteador para a internet.

Se você estiver interessado em outras configurações que pode usar em arquivos `ifcfg`, verifique o arquivo `sysconfig.txt` no diretório `/usr/share/doc/initscripts-*`. Eis algumas outras configurações que podem lhe interessar:

- **PEERDNS** — Configurar `PEERDNS=no` impede que DHCP substitua o arquivo `/etc/resolv.conf`. Isso permite que você defina quais servidores DNS seu sistema usa sem medo de que as informações sejam apagadas por dados que são fornecidos pelo servidor DHCP.
- **DNS?** — Se um arquivo `ifcfg` estiver sendo gerenciado pelo NetworkManager, ele define o endereço dos servidores DNS utilizando entradas “DNS?”. Por exemplo, `DNS1=192.168.0.2` faz com que o endereço IP seja gravado no arquivo `/etc/resolv.conf` como o primeiro servidor de DNS a ser utilizado no sistema. Você pode ter várias entradas “DNS?” (`DNS2=`,

DNS3= etc.).

Depois que um arquivo `ifcfg-*` é criado, você pode ativar e desativar as placas individualmente usando os comandos `ifup` e `ifdown`, em vez de ativar e desativar todas as placas juntas. Por exemplo, você pode ativar e desativar a placa `lo` (`ifcfg-lo`), usando os seguintes comandos:

```
# ifdown lo
# ifconfig
# ifup lo
# ifconfig
```

Os comandos mostrados acima desativam a placa de rede de loopback (`ifdown lo`) e então deixam você ver que ela não está ativa (`ifconfig`). Depois disso, você a ativa novamente (`ifup lo`) e verifica novamente se ela está ativa (`ifconfig`).

Além de configurar as placas de rede primárias, você também pode criar arquivos no diretório `/etc/sysconfig/network-scripts` que podem ser usados para configurar aliases (vários endereços IP para a mesma placa), placas agregadas (várias placas de rede ouvindo o mesmo endereço) e rotas personalizadas. Isso está descrito mais adiante nesta seção.

Outros arquivos de rede

Além dos arquivos de placa de rede, há outros arquivos de configuração de rede que você pode editar diretamente para configurar a rede Linux. Eis alguns desses arquivos.

file/etc/sysconfig/network

Configurações globais de sistema associadas à sua rede local podem ser incluídas no arquivo `/etc/sysconfig/network`. O `hostname` do sistema é mais comumente configurado nesse arquivo, mas outras configurações também podem ser adicionadas a ele. Eis um exemplo do conteúdo de um arquivo `/etc/sysconfig/network`:

```
NETWORKING=yes
HOSTNAME=abc.example.com
GATEWAY=192.168.0.1
```

O exemplo anterior mostra que a rede é ativada por padrão. O `hostname` do sistema local está configurado como `abc.example.com`. O valor `HOSTNAME` é o lugar correto para colocar o `hostname` do sistema porque ele é lido toda vez que o sistema é inicializado. Observe que você também pode adicionar o `GATEWAY` padrão (aqui `192.168.0.1`). Placas diferentes podem usar endereços de `GATEWAY` diferentes. Para outras configurações que podem aparecer nos arquivos `network`, verifique o arquivo `sysconfig.txt` no diretório `/usr/share/doc/initscripts-*`.

file/etc/host

Antes de o DNS ter sido criado, a tradução de nomes de máquinas para endereços IP era feita passando um único arquivo `hosts`. Embora houvesse apenas algumas dezenas, e depois algumas centenas, de `hosts`

na internet, essa abordagem funcionava muito bem. Mas, com o crescimento da internet, o uso de um único arquivo `hosts` tornou-se inviável e o DNS foi inventado.

O arquivo `/etc/hosts/` ainda existe em sistemas Linux. Ele ainda pode ser usado para mapear endereços IP para nomes de host. O arquivo `/etc/hosts` é uma maneira de criar nomes e endereços para uma pequena rede local ou apenas criar aliases, para facilitar o acesso aos sistemas que você usa o tempo todo.

Eis um exemplo de um arquivo `/etc/hosts`:

```
127.0.0.1    localhost.localdomain  localhost
::1         mycomputer    chris localhost6.localdomain6  localhost6
192.168.0.201  node1.example.com node1 joe
192.168.0.202  node2.example.com node2 sally
```

As duas primeiras linhas (`127.0.0.1` e `::1`) configuram endereços para o sistema local. O endereço IPv4 para o host local é `127.0.0.1`, o endereço IPv6 para o host local é `::1`. Há também entradas para dois endereços de IP. Você poderia alcançar o primeiro endereço IP (`192.168.0.201`) pelos nomes `node1.example.com`, `node1` ou `joe`. Por exemplo, digitar `ping joe` resulta no envio de pacotes para `192.168.0.201`.

file/etc/resolv.conf

Servidores DNS e domínios de pesquisa são configurados no arquivo `/etc/resolv.conf`. Se o NetworkManager estiver habilitado e funcionando, você não deve editar esse arquivo diretamente. Se você usar entradas DNS?= em arquivos `ifcfg-*`, o NetworkManager irá sobrescrever o arquivo `/etc/resolv.conf` e você perderá todas as entradas que tiver adicionado ao arquivo. Eis um exemplo do arquivo `/etc/resolv.conf` que foi modificado pelo NetworkManager.

```
# Generated by NetworkManager
nameserver 192.168.0.2
nameserver 192.168.0.3
```

Cada entrada `nameserver` identifica o endereço IP de um servidor DNS. A ordem é a m que os servidores DNS são verificados. É normal ter duas ou três entradas `nameserver`, no caso de a primeira não estar disponível. Mais do que isso pode levar muito tempo para um hostname não traduzível ser verificado para cada servidor.

Outro tipo de entrada que você pode adicionar a esse arquivo é uma entrada de pesquisa. A entrada de pesquisa permite indicar domínios a serem pesquisados quando uma máquina é requisitada por seu nome de base em vez de seu nome de domínio completamente qualificado. Você pode ter várias entradas de pesquisa identificando um ou mais nomes de domínio depois da palavra-chave de pesquisa. Por exemplo:

```
search example.com example.org example.net
```

As opções de pesquisa são separadas por espaços ou tabulações.

/etc/nsswitch.conf

Configurações no arquivo `/etc/nsswitch.conf` determinam que a conversão de hostname é feita

pela primeira pesquisa no arquivo `/etc/hosts local` e então nos servidores DNS listados no arquivo `/etc/resolv.conf`. É assim que a entrada `hosts` no arquivo `/etc/resolv.conf` aparece no Red Hat Enterprise Linux:

```
hosts: files dns
```

Você pode adicionar outros locais, como os bancos de dados do Network Information Service (`nis` ou `nisplus`), para consultar `hostname` para a conversão de endereços IP. Você também pode alterar a ordem em que os diferentes serviços são consultados. Você pode verificar se a conversão de `host` para endereço IP está funcionando corretamente usando diferentes comandos.

Se quiser verificar se seus servidores DNS estão sendo consultados corretamente, você pode usar os comandos `host` ou `dig`. Por exemplo:

```
$ host redhat.com
redhat.com has address 209.132.183.181
redhat.com mail is handled by 5 mx1.redhat.com.
redhat.com mail is handled by 10 mx2.redhat.com.
$ dig redhat.com
; <<>> DiG 9.7.3-P3-RedHat-9.7.3-8.P3.el6_2.2 <<>> redhat.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54399
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0,
ADDITIONAL: 0
;; QUESTION SECTION:
;redhat.com.                IN      A
;; ANSWER SECTION:
redhat.com.                60      IN      A      209.132.183.181
;; Query time: 105 msec
;; SERVER: 8.8.128.23#53(8.8.128.23)
;; WHEN: Sun Apr 29 08:32:32 2012
;; MSG SIZE rcvd: 44
```

Por padrão, o comando `host` produz uma saída mais simples para consultas DNS. Ele mostra o endereço IP para `Redhat.com` e os nomes dos servidores de email (registros MX) que servem `Redhat.com`. O comando `dns` mostra informações semelhantes às que aparecem nos arquivos que armazenam registros de DNS. A seção `QUESTION` da saída mostra que a seção de endereço requisitou o endereço de `Redhat.com` e a seção `ANSWER` mostrou a resposta (209.132.183.181). Você também pode ver o endereço do servidor de DNS que foi consultado.

Os comandos `host` e `dig` são utilizados apenas para consultar servidores DNS. Eles não verificam o arquivo `nsswitch.conf` para encontrar outros lugares para consulta, como o arquivo `hosts local`. Para isso, você teria de usar o comando `getent`. Por exemplo:

```
# getent hosts node1
192.168.0.201 node1
```

Este exemplo de `getent` encontra um `host` chamado `node1`, que foi digitado no meu arquivo `/etc/hosts local`. (O comando `getent` pode ser usado para consultar qualquer configuração de informações no arquivo `nsswitch.conf`. Por exemplo, digitar `getent passwd root` mostra a entrada para a conta de usuário `root` no arquivo local, mas também pode consultar um banco de dados LDAP remoto para obter informações de usuários se você tiver configurado esse recurso, conforme descrito no

Configurando aliases de placas de rede

Há ocasiões em que você pode querer que sua placa de rede ouça em vários endereços IP. Por exemplo, se você fosse configurar um servidor web que estava servindo conteúdo seguro (`https`) para vários domínios (`example.com`, `example.org` etc.), cada domínio exigiria um endereço IP separado (associado a um certificado separado). Nesse caso, em vez de adicionar várias placas de rede ao computador, você pode simplesmente criar vários aliases em uma única placa de rede.

Para criar um alias de placa de rede, basta criar outro arquivo `ifcfg-`. Seguindo o exemplo de uma placa `eth0` em um sistema RHEL, você pode criar uma placa `eth0:0` associada com a mesma placa de rede. Para tanto, crie um arquivo no diretório `/etc/sysconfig/networkscripts` chamado `ifcfg-eth0:0`, que contém informações como as seguintes:

```
DEVICE=eth0:0
ONPARENT=yes
IPADDR=192.168.0.141
NETMASK=255.255.255.0
```

O código de exemplo cria um alias para a placa de rede `eth0` chamado `eth0:0`. Em vez de `ONBOOT`, a entrada `ONPARENT` diz para ativar essa placa se o pai (`eth0`) tiver iniciado e ouvir no endereço `192.168.0.141`. Você pode ativar essa placa digitando `ifup eth0:0`. Você pode, então, verificar se a placa foi ativada usando o comando `ip`:

```
$ ip addr show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc
  pfifo_fast state UP qlen 1000
  link/ether f0:de:f1:28:46:d9 brd ff:ff:ff:ff:ff:ff
  inet 192.168.0.140/24 brd 192.168.0.255 scope global eth0
  inet 192.168.0.141/24 brd 192.168.0.255 scope global
secondary eth0:0
  inet6 fe80::f2de:f1ff:fe28:46d9/64 scope link
  valid_lft forever preferred_lft forever
```

Você pode ver que a placa de rede representada por `eth0` está ouvindo em dois endereços: `192.168.0.140` (`eth0`) e `192.168.0.141` (`eth0:0`). Assim, esse sistema responderá a pacotes destinados a qualquer um desses dois endereços. Você pode adicionar mais endereços IP para a placa, criando mais arquivos `ifcfg-eth0:?` (`ifcfg-eth0:1`, `ifcfg-eth0:2` etc.).

Configurando agregação de canais Ethernet

A agregação de canais Ethernet permite que você tenha mais de uma placa de rede em um computador

associado a um único endereço IP. Há várias razões pelas quais você pode querer fazer isso:

- **Alta disponibilidade** — Várias placas de rede no mesmo endereço IP podem garantir que, se uma sub-rede cair ou uma placa de rede quebrar, o endereço ainda pode ser alcançado em uma placa de rede agregada a outra sub-rede.
- **Desempenho** — Se houver muito tráfego de rede para ser tratado por uma placa de rede, você pode distribuir esse tráfego por várias placas de rede.

No Red Hat Enterprise Linux e no Fedora em um computador com várias placas de rede, você pode configurar a agregação de canal Ethernet, criando alguns arquivos `ifcfg` e carregando o módulo necessário. Você pode começar com um arquivo de agregação (por exemplo, `ifcfg-bond0`) e, então, apontar múltiplos arquivos `ifcfg-eth?` para essa placa agregada. Depois, você pode carregar o módulo de agregação.

Dependendo do tipo de conexão que você quer fazer, pode configurar sua placa agregada para diferentes modos. Utilizando a variável `BONDING_OPTS`, você define o modo de agregação e outras opções (todas são passadas para o módulo de agregação). Você pode ler sobre o módulo de agregação digitando `modinfo bonding` ou instalando o pacote `kernel-docs` e lendo o arquivo `bonding.txt` do diretório `/usr/share/doc/kernel-doc*/ Documentation/networking/bonding.txt`.

Eis um exemplo do arquivo que define uma placa agregada. O arquivo nesse exemplo é `/etc/sysconfig/network-scripts/ifcfg-bond0`:

```
DEVICE=bond0
ONBOOT=yes
IPADDR=192.168.0.50
NETMASK=255.255.255.0
BOOTPROTO=none
BONDING_OPTS="mode=active-backup"
```

A placa `bond0` nesse exemplo usa o endereço IP `192.168.0.50`. Ela inicia na inicialização do sistema. A linha `BONDING_OPTS` configura o modo de agregação como `active-backup`. Isso significa que apenas uma placa de rede por vez está ativa e a placa de rede seguinte só assume quando a anterior falha (failover). Nenhuma placa de rede está associada com a placa `bond0` ainda. Para isso, você deve criar opções de arquivo `ifcfg` separadas. Por exemplo, crie um `/etc/sysconfig/network-scripts/ifcfg-eth0` parecido com o seguinte (depois de criar `eth1`, `eth2`, `eth3` etc. para cada placa de rede que você quer usar no agregado de placas):

```
DEVICE=eth0
MASTER=bond0
SLAVE=yes
BOOTPROTO=none
ONBOOT=yes
```

Com a placa `eth0` utilizada como parte da placa `bond0`, não há um endereço de IP atribuído. Isso

porque a placa `eth0` usa o endereço IP da placa `bond0`, definindo-se como uma escrava (`SLAVE=yes`) para `bond0` (`MASTER=bond0`).

A última coisa que você quer fazer é certificar-se de que a placa `bond0` está configurada para usar o módulo `bonding`. Para fazer isso, crie um arquivo `/etc/modprobe.d/bonding.conf` que contém a seguinte entrada:

```
alias bond0 bonding
```

Como todas as placas estão definidas para `ONBOOT=yes`, a placa `bond0` começa e todas as placas `eth?` estão disponíveis à medida que são necessárias.

Definindo rotas personalizadas

Em uma configuração de rede simples, as comunicações que são destinadas para a rede local são direcionadas para a placa apropriada na rede local, enquanto as comunicações com hosts fora de sua rede local vão para um gateway padrão a fim de serem enviadas para hosts remotos. Como alternativa, você pode definir rotas personalizadas para fornecer caminhos alternativos para redes específicas.

Para definir uma rota personalizada no Fedora e no RHEL, você cria um arquivo de configuração no diretório `/etc/sysconfig/network-scripts`. Nesse percurso, você define:

- `GATEWAY?` — O endereço IP do nó na rede local que fornece a rota para a sub-rede representada pela rota estática.
- `ADDRESS?` — O endereço de IP que representa a rede que pode ser alcançada pela rota estática.
- `NETMASK?` — A máscara de rede que determina qual parte de `ADDRESS?` representa a rede e qual representa os hosts que podem ser alcançados na rede.

O nome de cada arquivo de rota personalizada é `route-placa`. Assim, por exemplo, uma rota personalizada que pode ser alcançada por meio da placa `eth0` seria chamada `route-eth0`. Você pode ter múltiplas rotas personalizadas nesse arquivo, com cada entrada de rota substituindo o `?` pelo número de placa. Por exemplo:

```
ADDRESS0=192.168.99.0
NETMASK0=255.255.255.0
GATEWAY0=192.168.0.5
```

Nesse exemplo, qualquer pacote destinado a um host na rede `192.168.99` seria enviado por meio da placa local `eth0` e direcionado ao nó gateway em `192.168.0.5`. Presumivelmente, esse nó seria uma rota para outra rede contendo hosts na faixa de endereço `192.168.99`. Essa rota terá efeito quando a placa de rede `eth0` for reiniciada.

Para verificar se a rota está funcionando depois de reiniciar a placa de rede, você pode digitar o seguinte:

```
# route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
default 192.168.0.1 0.0.0.0 UG 0 0 0 eth0
192.168.0.0 * 255.255.255.0 U 1 0 0 eth0
192.168.99.0 192.168.0.5 255.255.255.0 UG 0 0 0 eth0
```

A saída do comando `route -n` mostra que a rota padrão (qualquer coisa não destinada à rede local 192.168.0 ou à rede 192.168.99) é via o endereço 192.168.0.1. Todos os pacotes destinados à rede 192.168.99 são direcionados através do endereço 192.168.0.5.

Se quiser adicionar mais rotas personalizadas, você pode adicioná-las a esse mesmo arquivo `route-eth0`. O próximo conjunto de informações seria nomeado `ADDRESS1`, `NETMASK1`, `GATEWAY1` assim por diante.

Configurando redes na empresa

Até agora, a configuração de rede descrita neste capítulo concentrou-se na criação de sistemas simples para se conectar a uma rede. Recursos disponíveis em Linux podem ir muito além disso, fornecendo software que suporta a infraestrutura de rede real necessária para computadores host se comunicarem.

Esta seção apresenta alguns tipos de serviço de infraestrutura de rede disponíveis no Linux. A implementação completa desses recursos está além do escopo deste livro, mas saiba que se você precisar gerenciar recursos de infraestrutura de rede, esta seção lhe dará uma ideia de como esses recursos são implementados em Linux.

Configurando o Linux como um roteador

Se tiver mais de uma placa de rede em um computador (normalmente, duas ou mais placas de rede), você pode configurar o Linux como um roteador. Para que isso aconteça, tudo o que é necessário é uma mudança em um parâmetro do kernel que habilita o roteamento de pacotes. Para ativar o parâmetro `ip_forward` imediata e temporariamente, digite o seguinte como root:

```
# cat /proc/sys/net/ipv4/ip_forward
0
# echo 1 > /proc/sys/net/ipv4/ip_forward
# cat /proc/sys/net/ipv4/ip_forward
1
```

O encaminhamento (roteamento) de pacotes está desativado por padrão, com o valor de `ip_forward` configurado como 0. Configurando-o como 1, o roteamento de pacotes é imediatamente ativado. Para tornar essa alteração permanente, você deve adicionar esse valor ao arquivo `/etc/sysctl.conf`, por isso ele aparece como segue:

```
net.ipv4.ip_forward = 1
```

Com esse arquivo modificado, como mostrado, cada vez que o sistema for reinicializado, o valor para

`ip_forward` é redefinido como 1. (Observe que `net.ipv4.ip_forward` reflete a localização real do arquivo `ip_forward`, menos o `/proc/sys` e com pontos substituindo barras. Você pode mudar os parâmetros do kernel estabelecidos na estrutura de diretórios `/proc/sys` dessa maneira.)

Quando um sistema Linux é usado como um roteador, ele também costuma ser usado como um firewall entre uma rede privada e uma rede pública, como a internet. Se esse for o caso, você também pode querer usar esse mesmo sistema como um firewall que faz tradução de endereços de rede (NAT) e oferece o serviço DHCP, para que os sistemas na rede privada possam encaminhar por meio do sistema Linux usando endereços IP privados. (Veja o Capítulo 25, “Protegendo o Linux em uma rede”, para obter informações sobre como trabalhar com regras de firewall Linux utilizando o recurso `iptables`.)

Configurando o Linux como um servidor DHCP

Além de poder usar um servidor DHCP para obter seu endereço IP e outras informações, um sistema Linux também pode ser configurado para funcionar, ele próprio, como um servidor DHCP. Na sua forma mais básica, um servidor DHCP pode distribuir endereços IP a partir de um pool de endereços para qualquer sistema que o solicite. Normalmente, porém, o servidor DHCP também vai distribuir os locais dos servidores DNS e o gateway padrão.

Configurar um servidor DHCP não é algo que deve ser feito sem alguma consideração. Não adicione um servidor DHCP em uma rede que não está sob seu controle e que já tem um servidor DHCP funcionando. Muitos clientes estão configurados para obter informações de endereço a partir de qualquer servidor DHCP que irá passá-las adiante.

O serviço DHCP é fornecido pelo pacote `dhcp` no Fedora e no RHEL. O serviço é chamado `dhcpd` no RHEL 6 e `dhcpd.service` na última versão do Fedora. O arquivo de configuração principal é o `/etc/dhcp/dhcpd.conf` para redes IPv4 (há um arquivo `dhcpd6.conf` no mesmo diretório para fornecer o serviço DHCP para redes IPv6). Por padrão, o daemon `dhcpd` escuta na porta UDP 67, portanto, lembre-se de manter essa porta aberta no firewall.

Para configurar um servidor DHCP, você pode copiar o arquivo `dhcpd.conf.sample` do diretório `/usr/share/doc/dhcp-4*` e substituir o arquivo `/etc/dhcp/dhcpd.conf`. Então, modifique-o como quiser. Antes de usar esse arquivo, altere as opções de nomes de domínio para refletir seu domínio e intervalos de endereços IP para atender àqueles que você está usando. Os comentários no arquivo irão ajudá-lo a fazer isso.

Configurando o Linux como um servidor de DNS

No Linux, a maioria dos servidores Domain Name System (DNS) profissionais são implementados usando o serviço Berkeley internet Name Domain (BIND). Isso é implementado no Fedora e no RHEL, instalando os pacotes `bind`, `bind-utils` e `bind-libs`. Para maior segurança, algumas pessoas instalam o pacote `bind-chroot`.

Por padrão, `bind` é configurado editando o arquivo `/etc/named.conf`. O mapeamento de endereço IP para hostname é feito em arquivos de zonas localizados no diretório `/var/named`. Se você instalar o pacote `bind-chroot`, os arquivos de configuração de `bind` são movidos para baixo do diretório

`/var/named/chroot`, que tenta replicar os arquivos de `/etc` e `/var` que são necessários para configurar `bind`, de modo que o daemon identificado (que fornece o serviço) é confinado à estrutura de diretórios `/etc/named/chroot`.

Se você está interessado em experimentar `bind`, recomendo que primeiro experimente-o configurando DNS para uma pequena rede doméstica atrás de um firewall como uma forma de tornar mais fácil para as pessoas em sua casa se comunicarem entre si. Você pode bloquear os endereços IP das máquinas em sua casa anexando os endereços MAC da placa de rede de cada computador a endereços IP específicos em um servidor DHCP e depois mapeando esses nomes para os endereços em um servidor DNS.

Atenção

Antes de criar um servidor DNS público, tenha em mente que é muito importante proteger seu servidor DNS adequadamente. Um servidor DNS público invadido pode ser usado para redirecionar o tráfego para qualquer servidor que os invasores quiserem. Então, se você estiver usando esse servidor, você está em perigo de ser apresentado a sites que não são aqueles que você acha que eles são.

Configurando o Linux como um servidor proxy

Um servidor proxy fornece um meio de restringir o tráfego de rede de uma rede privada para uma rede pública, como a internet. Esses servidores oferecem uma excelente maneira de bloquear um laboratório de informática em uma escola ou restringir os sites que funcionários podem visitar a partir do seu trabalho.

Configurando fisicamente o Linux como um roteador, mas configurando-se como um servidor proxy, todos os sistemas em sua rede doméstica ou corporativa podem ser configurados para acessar a internet usando apenas certos protocolos e somente depois de filtrar o tráfego.

Usando o Squid Proxy Server, que vem com a maioria dos sistemas Linux (pacote `squid` no Fedora e no RHEL), você pode ativar o sistema para aceitar solicitações para servidores web (HTTP e HTTPS), servidores de arquivos (FTP) e outros protocolos. Você pode restringir quais sistemas podem usar o servidor proxy (por nome ou endereço IP) e até mesmo limitar os sites que eles podem visitar (por endereço específico, faixa de endereços, hostname ou nomes de domínio).

Configurar um servidor proxy squid pode ser tão simples como instalar o pacote `squid`, editar o arquivo `/etc/squid/squid.conf` e iniciar o serviço `squid`. O arquivo vem com uma configuração recomendada mínima. Mas você pode querer definir os hosts (com base no endereço IP ou nome) que você deseja permitir para usar o serviço. Há listas negras disponíveis com o `squid` que permitem negar o acesso a todo um conjunto de sites que podem ser inadequados para crianças visitarem.

Configurando VLANs no Linux

A virtualização é um recurso que permite que um sistema operacional, como o Linux, tenha outros sistemas operacionais rodando nele. O Linux atua como um host virtual no qual vários sistemas operacionais Linux, Windows, BSD ou outros podem ser executados ao mesmo tempo. Como cada

sistema virtualizado acredita estar sendo executado diretamente no hardware do computador, cabe ao host virtual fornecer placas que se parecem com hardware. Para a rede, os hosts fornecem redes locais virtuais (ou VLANs).

Como seu nome indica, uma VLAN é uma rede local virtual que roda dentro do computador host Linux. Para os sistemas virtuais em execução no host, a VLAN parece um comutador de rede física. Como o comutador físico, a VLAN pode ser configurada para fazer uma ponte ou roteamento. Em outras palavras, os sistemas virtuais podem ser executados em sua própria rede, com pacotes roteados fora dessa rede para placas de rede do host (roteamento) ou simplesmente passados através de placas de rede do computador como se os sistemas virtuais existissem na rede física (ponte).

A configuração de VLANs está além do escopo deste livro. Mas você deve ficar ciente de que, se estiver executando o Linux como um host virtual, redes locais virtuais podem ser automaticamente configuradas e iniciadas para você. Em sistemas Fedora e RHEL recentes, a virtualização é fornecida pelo recurso Kernel Virtualization Module (KVM). Outros sistemas Linux oferecem o Xen como sua tecnologia de virtualização.

Se você instalou a virtualização em seu sistema (`yum groupinstall Virtualization`), você pode ficar confuso com o aparecimento de novas placas de rede nele depois de reiniciá-lo (ou iniciar manualmente o serviço `libvirtd`). Por exemplo, você verá uma nova placa `vibr0`. O seguinte é a saída do comando `ifconfig` em um sistema depois de instalar o `Virtualization package group` e reiniciar o RHEL:

```
# ifconfig vibr0
virbr0 Link encap:Ethernet HWaddr 52:54:00:2F:54:85
        inet addr:192.168.122.1 Bcast:192.168.122.255
Mask:255.255.255.0

        UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
        RX packets:0 errors:0 dropped:0 overruns:0 frame:0
        TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:0
        RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```

O resultado mostra uma VLAN configurada para fazer uma ponte sobre a rede 192.168.122. Essa VLAN fornece um meio de sistemas virtuais instalados no host serem capazes de alcançar um ao outro e também o mundo exterior usando as placas de rede fornecidas pelo host. Para obter informações sobre instalação e configuração de hosts de virtualização KVM e máquinas virtuais, consulte o Red Hat Virtualization Host Configuration and Guest Installation Guide (http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Virtualization_Host_Configuration_and_Guest_Installation_Guide/index.h

Resumo

A maioria das conexões de rede a partir de um desktop ou sistema portátil Linux pode ser feita com

pouca ou nenhuma intervenção do usuário. Se você usar o NetworkManager sobre uma conexão Ethernet com ou sem fio, as informações de endereço e servidor necessárias para iniciar podem ser obtidas automaticamente a partir de um servidor DHCP.

Com a interface gráfica NetworkManager, você pode fazer algumas configurações de rede, se quiser. Você pode configurar endereços IP estáticos e selecionar qual servidor DNS e gateway usar. Para fazer uma configuração de rede mais manual e complexa, considere desativar o NetworkManager e trabalhar diretamente com os arquivos de configuração de rede.

Arquivos de configuração de rede em Linux podem ser usados para configurar recursos mais avançados, tais como aliases de rede, agregação de canal Ethernet e rotas personalizadas. Para utilizar esses serviços mais avançados, você pode habilitar o serviço de rede.

Além da conectividade básica de rede no Linux, há recursos disponíveis que permitem fornecer serviços de infraestrutura de rede. Este capítulo apresentou serviços e funcionalidades como roteamento, DHCP, DNS e VLANs, que você precisa saber ao trabalhar com recursos mais avançados de rede em Linux.

Com sua rede configurada, agora você pode começar a configurar serviços para executar em suas redes. O Capítulo 15 descreve as ferramentas que você precisa para ativar, desativar, iniciar, parar e verificar o status dos serviços que estão configurados para seu sistema Linux.

Exercícios

Os exercícios desta seção ajudam você a examinar e alterar as placas de rede em seu sistema Linux, bem como a entender como configurar recursos de rede mais avançados. Inicie estes exercícios em um sistema Linux que tenha uma conexão de rede ativa, mas *não* esteja no meio de alguma atividade de rede crítica.

Recomendo que você faça esses exercícios diretamente no console do computador (em outras palavras, não use ssh no computador para fazê-las). Alguns dos comandos que você executa podem interromper sua conectividade de rede e algumas das configurações que você faz, se você cometer um erro, podem tornar seu computador temporariamente indisponível na rede.

Muitas vezes há várias maneiras de completar as tarefas descritas nesses exercícios. Se você empacar, consulte as soluções das tarefas que são mostradas no Apêndice B.

1. Use o desktop para verificar se o NetworkManager iniciou com sucesso sua placa de rede (com ou sem fio) na rede. Se não iniciou, tente, então, iniciar sua conexão de rede.
2. Execute um comando para verificar as placas de rede ativas disponíveis em seu computador.
3. Tente acessar `google.com` a partir da linha de comando de uma maneira que assegure que o DNS está funcionando corretamente.
4. Execute um comando para verificar as rotas sendo usadas para se comunicar fora da rede local.
5. Trace a rota a ser tomada para se conectar com `google.com`.
6. Desative e desabilite o NetworkManager e inicie o serviço de rede.
7. Crie uma entrada de host que permite que você se comunique com o sistema host local usando o nome `myownhost`.

8. Adicione o servidor DNS Google público (endereço IP 8.8.8.8) como o último na sua lista de servidores DNS.
9. Crie uma rota personalizada que direciona o tráfego destinado à rede 192.168.99.0/255.255.255.0 para algum endereço IP em sua rede local, como 192.168.0.5 (primeiro garantindo que a rede 10.0.99 não está sendo usada em seu local).
10. Verifique se seu sistema foi configurado para permitir o roteamento de pacotes IPv4 entre as placas de rede de seu sistema.

Iniciando e parando serviços

NESTE CAPÍTULO

Entendendo os vários daemons de inicialização do Linux

Auditando os serviços controlados pelo daemon `init` do Linux

Parando e iniciando serviços

Alterando o nível de execução do servidor Linux padrão

Removendo serviços

Um *serviço*, também chamado de *daemon*, é um programa ou um processo em execução que fornece uma função específica. O principal trabalho de seu servidor Linux é oferecer serviços. Como um administrador de servidor Linux, uma de suas principais tarefas é gerenciar esses serviços.

O daemon `cupsd` é um exemplo de um serviço que seu servidor Linux pode oferecer. Esse serviço é responsável pela impressão e é coberto no Capítulo 16, “Configurando um servidor de impressão”.

Como você inicia o daemon `cupsd`? Como você o para? Como você faz para ele ser carregado na inicialização do sistema? E se você tiver um programa especial que você pretende iniciar em seu servidor? Como você faz para ele carregar na inicialização? Todas essas questões e muitas outras serão respondidas neste capítulo. Logo, você terá um excelente entendimento sobre o gerenciamento desses serviços.

Entendendo o daemon `init` do Linux

A fim de entender o gerenciamento do serviço, você precisa entender o daemon `init`. O daemon `init` pode ser pensado como a “mãe de todos os processos”. Esse daemon é o primeiro processo a ser iniciado pelo kernel no servidor Linux. O kernel do Linux possui um ID de processo (PID) de 0. Assim, o daemon `init` tem um ID de processo pai (PPID) de 0 e um PID de 1. Uma vez iniciado, `init` é responsável por carregar processos configurados para serem iniciados na inicialização do sistema do servidor, como o shell de login (processo `getty` ou `mingetty`). Ele também é responsável pelo

gerenciamento de serviços. Isso é muita responsabilidade para um daemon!

O daemon `init` do Linux foi baseado no daemon `init` do Unix System V. Por isso, ele é chamado de daemon `SysVinit`. Mas ele não foi o único daemon `init` clássico. O daemon `init` não é parte do kernel Linux. Portanto, ele pode vir em diferentes versões, e as distribuições do Linux podem escolher qual versão usar. Outro clássico daemon `init` baseia-se no Unix Berkeley, também chamado de BSD. Portanto, os dois daemons `init` originais do Linux eram `BSD init` e `SysVinit`.

Os daemons `init` clássicos funcionaram sem problemas por muitos anos. Mas esses servidores foram criados para trabalhar dentro de um ambiente estático. À medida que surgiram novos hardwares, como dispositivos USB, os daemons `init` clássicos tinham dificuldade para lidar com esses e outros dispositivos *hot-plug* (que podem ser conectados com o computador ligado). O hardware do computador tinha mudado de estático para baseado em eventos. Novos daemons `init` tornaram-se necessárias para lidar com esses ambientes fluidos.

Além disso, à medida que surgiam novos serviços, os daemons `init` clássicos tinham de lidar com a inicialização de cada vez mais serviços. Assim, o processo de inicialização do sistema inteiro era menos eficiente e, em última instância, mais lento.

Os daemons `init` modernos têm tentado resolver os problemas de inicialização ineficiente do sistema e ambientes não estáticos. Dois desses daemons `init` são `Upstart` e `systemd`. Muitas distribuições Linux têm feito a transição para os novos daemons `init` enquanto mantém retrocompatibilidade com os daemons `SysVinit` e `BSD init` clássicos.

`Upstart`, disponível em <http://upstart.ubuntu.com>, foi originalmente desenvolvido pela Canonical, o pai da distribuição Ubuntu. Mas muitas outras distribuições o adotaram, incluindo:

- RHEL, versão 6
- Fedora, versões 9 a 14
- Ubuntu, versões-6–10 e superiores
- openSUSE, versões 11.3 e superiores

Um novo daemon, `systemd`, disponível em <http://fedoraproject.org/wiki/Systemd>, foi escrito principalmente por Lennart Poettering, um desenvolvedor da Red Hat. Atualmente, é usado pelo Fedora versão 15 e superiores.

A fim de gerenciar adequadamente seus serviços, você precisa saber qual daemon `init` seu servidor tem. Descobrir isso pode ser um pouco complicado. Examine o seguinte para ajudar a determinar o daemon `init` do seu servidor Linux.

- Sua distribuição Linux e versão aparecem na lista anterior de adotantes de `Upstart`? Então, seu daemon `init` Linux é o daemon `Upstart init`.
- Sua distribuição Linux Fedora é versão 15 ou superior? Então, seu daemon `init` Linux é o daemon `systemd init`.
- Tente pesquisar o daemon `init` da sua distribuição Linux para obter pistas, utilizando os

comandos `strings` e `grep`. O exemplo de código a seguir mostra o daemon `init` em uma distribuição Linux Mint sendo pesquisado pelas referências de daemon `systemd` e `Upstart` `init`. A pesquisa por `systemd` não retorna nada. Mas a pesquisa por `Upstart` produz resultados. Assim, você pode ver que essa distribuição Linux Mint usa o daemon `Upstart` `init`.

```
$ sudo strings /sbin/init | grep -i systemd
$
$ sudo strings /sbin/init | grep -i upstart
upstart-devel@lists.ubuntu.com
UPSTART_CONFDIR
UPSTART_NO_SESSIONS
...
```

Em um servidor Fedora, a pesquisa por `Upstart` não retorna nada. Mas você pode ver que a pesquisa por `systemd` mostra a existência do daemon `systemd`.

```
# strings /sbin/init | grep -i upstart
#
# strings /sbin/init | grep -i systemd
systemd.unit=
systemd.log_target=
systemd.log_level=
...
```

ca
Se não tiver o comando `strings` em seu sistema Linux, você pode instalá-lo por meio do pacote `binutils`. No Fedora, use o comando `yum install binutils`. No Ubuntu, use o comando `sudo apt-get install binutils`

- Se você ainda não pode determinar o daemon `init` que seu servidor tem, tente olhar a página `init` da Wikipedia sobre “Replacements for `init`?” (<http://wikipedia.org/wiki/Init>).
- Você não conseguiu encontrar nenhuma informação usando as sugestões anteriores? Então, muito provavelmente, sua distribuição ainda está usando o daemon `SysVinit` ou `BSD init` clássico.

Tenha em mente que algumas distribuições Linux não migraram para os daemons mais recentes. A maioria das que migraram ainda mantêm retrocompatibilidade com os daemons `SysVinit` e `BSD init`.

Entendendo os daemons de inicialização clássicos

Os daemons `init`, `SysVinit` e `BSD init` clássicos merecem uma explicação, ainda que o servidor

Linux tenha um daemon `init` diferente. Não é apenas pela retrocompatibilidade dos novos daemons `init` com os clássicos, mas pelo fato de muitos dos novos se basearem nos clássicos. Entender os daemons `init` clássicos ajudará você a entender os daemons `init` modernos.

Os daemons `SysVinit` e `BSD init` clássicos funcionam de forma muito semelhante. Embora, no início, eles possam ter sido um pouco diferentes, com o tempo, bem poucas diferenças significativas permaneceram. Por exemplo, o antigo daemon `BSD init` obtinha informações de configuração do arquivo `/etc/ttytab`. Agora, assim como o daemon `SysVinit`, as informações de configuração do daemon `BSD init` são obtidas na inicialização, a partir do arquivo `/etc/inittab`. Eis um arquivo `SysVinit /etc/inittab` clássico:

```
# cat /etc/inittab
#
# inittab      This file describes how the INIT process should set up
#             the system in a certain run-level.
#
# Author:     Miquel van Smoorenburg, <miquels@drinkel.nl.mugnet.org>
#             Modified for RHS Linux by Marc Ewing and Donnie Barnes
#
# Default runlevel. The runlevels used by RHS are:
#  0 - halt (Do NOT set initdefault to this)
#  1 - Single user mode
#  2 - Multiuser, without NFS (The same as 3, if you do not have networking)
#  3 - Full multiuser mode
#  4 - unused
#  5 - X11
#  6 - reboot (Do NOT set initdefault to this)
#
id:5:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit

10:0:wait:/etc/rc.d/rc 0
11:1:wait:/etc/rc.d/rc 1
12:2:wait:/etc/rc.d/rc 2
13:3:wait:/etc/rc.d/rc 3
14:4:wait:/etc/rc.d/rc 4
15:5:wait:/etc/rc.d/rc 5
16:6:wait:/etc/rc.d/rc 6

# Trap CTRL-ALT-DELETE
ca::ctrlaltdel:/sbin/shutdown -t3 -r now

# When our UPS tells us power has failed, assume we have a few minutes
# of power left. Schedule a shutdown for 2 minutes from now.
# This does, of course, assume you have powerd installed and your
# UPS connected and working correctly.
pf::powerfail:/sbin/shutdown -f -h +2 "Power Failure; System Shutting Down"

# If power was restored before the shutdown kicked in, cancel it.
pr:12345:powerokwait:/sbin/shutdown -c "Power Restored; Shutdown Cancelled"
# Run gettys in standard runlevels
1:2345:respawn:/sbin/mingetty tty1
2:2345:respawn:/sbin/mingetty tty2
3:2345:respawn:/sbin/mingetty tty3
4:2345:respawn:/sbin/mingetty tty4
5:2345:respawn:/sbin/mingetty tty5

6:2345:respawn:/sbin/mingetty tty6

\# Run xdm in runlevel 5
x:5:respawn:/etc/X11/prefdm -nodaemon
```

O arquivo `/etc/inittab` diz ao daemon `init` que o `runlevel` é o padrão. Um `runlevel` é um número de categorização que determina quais serviços são iniciados e quais serviços são parados. No exemplo anterior, um `runlevel` padrão de 5 é configurado com o código `id:5:initdefault:.` A Tabela 15.1 mostra os sete níveis de execução padrão do Linux.

TABELA 15.1 Níveis de Execução Padrão do Linux

do nível de execução	Nome	Descrição
	Halt (Suspensão)	Todos os serviços estão desligados e o servidor está parado.
u S	Single User Mode (Modo monousuário)	A conta <code>root</code> é automaticamente logada no servidor. Outros usuários não podem fazer login no servidor. Apenas a interface de linha de comando está disponível. Serviços de rede não são iniciados.
	Multiuser Mode (Modo multiusuário)	Os usuários podem fazer login no servidor, mas apenas a interface de linha de comando está disponível. Serviços de rede não são iniciados.
	Extended Multiuser Mode (Modo multiusuário estendido)	Os usuários podem fazer login no servidor, mas apenas a interface de linha de comando está disponível. Os serviços de rede são iniciados.
	User Defined (Definido pelo usuário)	Os usuários podem personalizar esse nível de execução.
	Graphical Mode (Modo gráfico)	Os usuários podem fazer login no servidor. Linha de comando e interfaces gráficas estão disponíveis. Os serviços de rede são iniciados.
	Reboot (Reiniciado)	O servidor foi reiniciado.

Distribuições Linux podem diferir um pouco sobre a definição de cada `runlevel`, bem como quais `runlevels` são oferecidos. A distribuição Ubuntu, por exemplo, oferece níveis de execução 0–6, mas `runlevels` 2–5 iniciam os mesmos serviços que o `runlevel` 5 padrão listado na Tabela 15.1.

Atenção

Alguns níveis de execução que devem ser usados no arquivo `/etc/inittab` são 2 a 5. Os outros níveis de execução podem causar problemas. Por exemplo, se você configurasse o `runlevel` 6 no arquivo `/etc/inittab` como o padrão, quando o servidor fosse reinicializado, ele entraria em um loop e continuaria a reiniciar repetidamente.

Observe que não faz sentido incluir algumas das categorias de `runlevel` descritas na Tabela 15.1 no arquivo `/etc/inittab`. Não há nenhuma razão pela qual você poderia querer que seu `runlevel` padrão fosse 0, o que suspenderia o servidor sempre que ele simplesmente reiniciasse.

Os níveis de execução não são utilizados apenas como um `runlevel` padrão no arquivo `/etc/inittab`. Eles também podem ser chamados diretamente usando o daemon `init`. Assim, se quiser parar imediatamente seu servidor, você digita `init 0` na linha de comando:

```
# init 0
...
System going down for system halt NOW!
```

O comando `init` aceitará qualquer um dos números de `runlevel` da Tabela 15.1, o que lhe permite mudar rapidamente seu servidor de uma categoria de `runlevel` para outra. Por exemplo, se você precisar solucionar problemas, o que requer que a interface gráfica seja desativada, você pode digitar `init 3` na linha de comando:

```
# init 3
INIT: Sending processes the TERM signal
starting irqbalance:          [ OK ]
Starting setroubleshootd:
Starting fuse: Fuse filesystem already available.
...
Starting console mouse services: [ OK ]
```

Para ver o `runlevel` atual de seu servidor Linux, basta digitar o comando **`runlevel`**. O primeiro item exibido será o `runlevel` anterior do servidor, que no exemplo a seguir é 5. O segundo item exibido mostra o `runlevel` atual do servidor, que neste exemplo é 3.

```
$ runlevel
5 3
```

Além do comando `init`, você também pode usar o comando `telinit`, que é funcionalmente o mesmo. No exemplo a seguir, o comando `telinit` é usado para reiniciar o servidor, levando-o para `runlevel` 6:


```

# telinit 6
INIT: Sending processes the TERM signal
Shutting down smartd:          [ OK ]
Shutting down Avahi daemon:    [ OK ]
Stopping dhcdbd:               [ OK ]
Stopping HAL daemon:          [ OK ]
...
Starting killall:
Sending all processes the TERM signal... [ OK ]
Sending all processes the KILL signal... [ OK ]

...
Unmounting filesystems        [ OK ]
Please stand by while rebooting the system
...

```

Em um servidor Linux recém-iniciado, o número do `runlevel` atual deve ser o mesmo que o número do `runlevel` padrão no arquivo `/etc/inittab`. Mas note que o `runlevel` anterior no exemplo a seguir é N. O N significa “não existente” e indica que o servidor foi recém-iniciado.

```

$ runlevel
N 5

```

Como é que o servidor sabe quais serviços parar e quais iniciar quando um `runlevel` especial é escolhido? Quando um `runlevel` é escolhido, os scripts localizados no diretório `/etc/rc.d/rc#.d` (onde # é o `runlevel` escolhido) são executados. Esses scripts são executados se o `runlevel` for escolhido via uma inicialização do servidor e a configuração `/etc/inittab` `runlevel` ou quando o comando `init` ou `telinit` é usado. Por exemplo, se `runlevel 5` for escolhido, todos os scripts no diretório `/etc/rc.d/rc5.d` serão executados.

```
# ls /etc/rc.d/rc5.d
```

```

K01smolt                K88wpa_supplicant    S22messagebus
K02avahi-dnsconfd       K89dund              S25bluetooth
K02NetworkManager      K89netplugd         S25fuse
K02NetworkManagerDispatcher K89pand             S25netfs
K05saslauthd           K89rdisc             S25pcscd
K10dc_server           K91capi              S26hidd
K10psacct              S00microcode_ctl    S26udev-post
K12dc_client           S04readahead_early  S28autofs
K15gpm                 S05kudzu             S50hplip
K15httpd               S06cpuspeed          S55cups
K20nfs                 S08ip6tables         S55sshd
K24irda                S08iptables         S80sendmail
K25squid               S09isdn              S90ConsoleKit
K30spamassassin        S10network           S90crond
K35vncserver           s11auditd            s90xfs

```

K50netconsole	s12restorecond	s95anacron
K50tux	s12syslog	s95atd
K69rpcsvcgssd	s13irqbalance	s96readahead_later
K73winbind	s13mcstrans	s97dhcdbd
K73ypbind	s13rpcbind	s97yum-updatesd
K74nscd	s13setroubleshoot	s98avahi-daemon
K74ntpd	s14nfslock	s98haldaemon
K84btseed	s15mdmonitor	s99firstboot
K84bttrack	s18rpcidmapd	s99local
K87multipathd	s19rpcgssd	s99smartd

Note que alguns dos scripts dentro do diretório `/etc/rc.d/rc5.d` iniciam com um K e alguns com um S. O K refere-se a um script que irá eliminar (*kill*, ou parar) um processo. O S refere-se a um script que irá iniciar (*start*) um processo. Além disso, cada script K e S tem um número antes do nome do serviço ou daemon que eles controlam. Isso permite que os serviços sejam parados ou iniciados em uma ordem controlada específica. Você não iria gostar que os serviços de seu servidor de rede Linux fossem iniciados antes que a rede em si fosse iniciada.

Um diretório `/etc/rc.d/rc#.d` existe para todos os níveis de execução padrão do Linux. Cada um contém scripts para iniciar e parar serviços para seu `runlevel` específico.

```
# ls -d /etc/rc.d/rc?.d
/etc/rc.d/rc0.d /etc/rc.d/rc2.d /etc/rc.d/rc4.d /etc/rc.d/rc6.d
/etc/rc.d/rc1.d /etc/rc.d/rc3.d /etc/rc.d/rc5.d
```

Os arquivos no diretório `/etc/rc.d/rc#.d` não são realmente scripts, mas sim vínculos simbólicos para os scripts no diretório `/etc/rc.d/init.d`. Portanto, não é necessário ter múltiplas cópias de scripts específicos.

```
# ls -l /etc/rc.d/rc5.d/K15httpd
lrwxrwxrwx 1 root root 15 2011-10-27 19:59
/etc/rc.d/rc5.d/K15httpd -> ../init.d/httpd
#
# ls /etc/rc.d/init.d
```

anacron	functions	multipathd	rpcidmapd
atd	fuse	netconsole	rpcsvcgssd
auditd	gpm	netfs	saslauthd
autofs	haldaemon	netplugd	sendmail
avahi-daemon	halt	network	setroubleshoot
avahi-dnsmasq	hidd	NetworkManager	single

bluetooth	hplip	NetworkManagerDispatcher	smartd
btseed	hsqldb	nfs	smolt
bttrack	httpd	nfslock	spamassassin
capi	ip6tables	nscd	squid
ConsoleKit	iptables	ntpd	sshd
cpuspeed	irda	pand	syslog
crond	irqbalance	pcscd	tux
cups	isdn	psacct	udev-post
cups-config-daemon	killall	rdisc	vncserver
dc_client	kudzu	readahead_early	winbind
dc_server	mcstrans	readahead_later	wpa_supplicant
dhcdbd	mdmonitor	restorecond	xfstools
dund	messagebus	rpcbind	ypbind
firstboot	microcode_ctl	rpcgssd	yum-updatesd

Observe que cada serviço tem um único script em `/etc/rc.d/init.d`. Não há scripts separados para interromper e iniciar um serviço. Esses scripts param ou iniciam um serviço dependendo do parâmetro que lhes é passado pelo daemon `init`.

Nota

Dependendo de sua distribuição e seu daemon `init`, os scripts de parar e iniciar um serviço podem ser armazenados em um local diferente do diretório `/etc/rc.d/init.d`. As três localizações possíveis são:

- `/etc/rc.d/init.d` – Um diretório usado pelo daemon `SysVinit`.
- `/etc/rc.d` – O diretório utilizado pelo daemon `BSD init`
- `/etc/init.d` – Um diretório também utilizado pelo daemon `SysVinit`, normalmente em distribuições baseadas no `Debian`, como o `Ubuntu`

Cada script em `/etc/rc.d/init.d` cuida de tudo o que é necessário para iniciar ou parar um serviço específico no servidor. Eis um exemplo parcial do script `httpd` em um sistema `Linux` que usa o daemon `SysVinit`. Ele contém uma instrução `case` para lidar com o parâmetro (`$1`) que foi passado a ele, como `start`, `stop`, `status` etc.

```

# cat /etc/rc.d/init.d/httpd
#!/bin/bash
#
# httpd          Startup script for the Apache HTTP Server
#
# chkconfig: - 85 15
# description: Apache is a World Wide Web server.
#              It is used to serve \
#              HTML files and CGI.
# processname: httpd
# config: /etc/httpd/conf/httpd.conf
# config: /etc/sysconfig/httpd
# pidfile: /var/run/httpd.pid

# Source function library.
. /etc/rc.d/init.d/functions
...
# See how we were called.
case "$1" in
  start)
    start
    ;;
  stop)
    stop
    ;;
  status)
    status $httpd
    RETVAL=$?
    ;;
...
esac

exit $RETVAL

```

Depois que os scripts `runlevel` desejados no diretório `/etc/init.d` são executados, o carregamento do processo do daemon `SysVinit` está completo. Esse curso inteiro é uma abordagem passo a passo bem organizada para iniciar um servidor Linux com os serviços apropriados em execução. A Figura 15.1 revisa o papel do daemon `SysVinit` no carregamento dos processos corretos durante a inicialização do sistema.

Curso de ação clássico do daemon SysVinit na inicialização do servidor.

O curso é muito semelhante ao mudar de uma categoria de nível de execução para outra. A Figura 15.2 revisa os papéis do daemon SysVinit de parar e iniciar os processos corretos durante uma mudança de nível de execução.

modo de ação clássico do daemon SysVinit na mudança de nível de execução.

Agora que você já tem uma ideia dos daemons `init` clássicos, é hora de passar para o daemon `Upstart init` mais moderno.

Entendendo o daemon Upstart init

Como mencionado anteriormente, muitas distribuições Linux mudaram de daemons `init` clássicos para o daemon `Upstart init` mais moderno. Incluídos nessa lista de distribuição estão as distribuições RHEL e Ubuntu.

Aprendendo noções básicas sobre o daemon Upstart init

A principal diferença entre os daemons clássicos e o `Upstart` é a capacidade de parar e iniciar serviços. O daemon `SysVinit` foi criado para operar em um ambiente estático. O daemon `Upstart init` foi criado para operar em um ambiente flexível e em constante mudança. Com `SysVinit`, os serviços são parados e iniciados com base em níveis de execução. O daemon `Upstart init` não está preocupado com níveis de execução, mas com eventos do sistema. Os eventos são o que determinam quando os serviços são interrompidos e/ou iniciados. Um *evento* é uma ocorrência de um servidor Linux que desencadeia uma mudança de estado necessária do sistema, a qual é comunicada ao daemon `Upstart init`. Eis alguns exemplos de eventos de sistema:

- O servidor é inicializado.
- O comando `init` é utilizado.
- Um dispositivo USB é conectado ao servidor.

Enquanto os daemons `init` clássicos poderiam lidar com os dois primeiros exemplos de eventos, eles não poderiam lidar bem com o terceiro.

`Upstart` trata serviços por meio trabalhos definidos. Um *trabalho* `Upstart` pode ser uma tarefa ou um serviço. Uma *tarefa* executa um serviço limitado, completa seu trabalho e depois retorna a um estado de espera. Um *serviço*, por outro lado, é um programa de longa duração que nunca termina seu trabalho ou autotermina, mas em vez disso permanece em estado de execução. Um daemon é um exemplo de um trabalho do serviço `Upstart`.

O exemplo a seguir mostra diversos trabalhos `Upstart` que incluem tanto trabalhos de tarefa como trabalhos de serviço. Os trabalhos de tarefa (*task jobs*) estão em estado de `stop/waiting`, como a tarefa `rc`. Os trabalhos de serviço (*service jobs*) estão em um estado `start/running`, como o daemon `cups`.

```
$ initctl list
```

```
avahi-daemon start/running, process 456
```

```
mountall-net stop/waiting rc stop/waiting
rsyslog start/running, process 411
...
ssh start/running, process 405
udev-fallback-graphics stop/waiting
control-alt-delete stop/waiting
hwclock stop/waiting
mounted-proc stop/waiting
network-manager start/running, process 458
...
rc-sysinit stop/waiting cups start/running, process 1066
...
tty6 start/running, process 833
ureadahead stop/waiting
```

Esses vários trabalhos são definidos por meio de um arquivo de definição de trabalho. Todos os arquivos de definição de trabalho estão localizados no diretório `/etc/init`, como mostrado aqui:

```
$ ls /etc/init
```

```
acpid.conf                networking.conf
alsa-restore.conf         network-interface.conf
alsa-store.conf           network-interface-security.conf
anacron.conf              network-manager.conf
control-alt-delete.conf   procps.conf
cron.conf                 rc.conf
cups.conf                 rcS.conf
dbus.conf rc-sysinit.conf  dmesg.conf rsyslog.conf
failsafe.conf             setvtrgb.conf
friendly-recovery.conf    ssh.conf
hostname.conf             tty1.conf
hwclock.conf              tty2.conf
hwclock-save.conf         tty3.conf
irqbalance.conf          tty4.conf
lightdm.conf              tty5.conf
...
```

O daemon `Upstart init` depende de eventos para acionar determinados serviços para iniciar, parar, reiniciar etc. Eventos são comunicados ao daemon `Upstart init` ou são criados pelo daemon `Upstart`. Isso é chamado de um **evento emitido**. As ações tomadas quando um evento é emitido dependem das configurações em um arquivo de configuração de trabalho. Considere o

seguinte arquivo de configuração do daemon Network Manager:

```
⌘ cat /etc/init/network-manager.conf
# network-manager - network connection manager
#
# The Network Manager daemon manages the system's network connections,
# automatically switching between the best available.

description    "network connection manager"

start on (local-filesystems and started dbus)
stop on stopping dbus

expect fork
respawn

exec NetworkManager
⌘
```

A partir do exemplo, você pode ver que há dois eventos que devem ocorrer para disparar o daemon `Upstart init` a fim de iniciar o daemon Network Manager:

- O evento `local-filesystems` — O daemon `local-filesystems` emitirá esse evento quando todos os sistemas de arquivos locais no arquivo de configuração `Upstart init` estiverem montados.
- O evento `dbus daemon started` — O daemon `Upstart init` emite esse evento `started` quando o daemon `dbus` atingir o estado `start/running`.

Assim, quando esses dois eventos ocorrem, o daemon `Upstart init` é informado e, então, inicia o daemon `NetworkManager`.

Como o daemon `Upstart init` pode lidar com esses eventos e acompanha o status (estado) dos processos, ele é muitas vezes referido como uma “máquina de estado”. O daemon `Upstart init` também é referido como um “mecanismo de evento”, porque ele próprio emite eventos.

O daemon `Upstart init` é uma abordagem flexível, organizada e inteligente de lidar com eventos do servidor. A Figura 15.3 mostra o curso de ação do daemon `Upstart init` carregando os processos corretos na inicialização do sistema.

urso de ação de `Upstart` `init` na inicialização do servidor.

Aprendendo a retrocompatibilidade de `Upstart` com `SysVinit`

`Upstart` fornece retrocompatibilidade com o daemon `SysVinit`. Isso deu tempo para as distribuições Linux migrarem lentamente para o `Upstart`.

O arquivo `/etc/inittab` ainda está em algumas distribuições. Distribuições RHEL e Fedora ainda usando `Upstart` usam `/etc/inittab` para inicializar o padrão `runlevel` listado. A distribuição Ubuntu não tem mais o arquivo. O exemplo do arquivo `/etc/inittab` que se segue vem de um servidor que executa uma versão do Fedora, a qual usa o daemon `Upstart` `init`.

```
$ cat /etc/inittab
# inittab is only used by upstart for the default runlevel.
#
# ADDING OTHER CONFIGURATION HERE WILL HAVE NO EFFECT ON YOUR
SYSTEM.
#
...
#
id:5:initdefault:
```

Como você pode ver a partir das linhas de comentário no arquivo `/etc/inittab`, a única coisa para que esse arquivo é usado nas distribuições Linux que o mantém é mudar o `runlevel` padrão durante a inicialização do servidor.

Dica

Para alterar o nível de execução padrão em uma distribuição Ubuntu que usa o `Upstart`, edite `/etc/init/rc-sysinit.conf` e altere a linha `DEFAULT_RUNLEVEL=#`, onde `#` é um número de 2 a 5. Mas lembre-se que os níveis de execução 2–5 no Ubuntu são equivalentes a `SysVinit` `runlevel` 5. Portanto, essa atividade não tem sentido.

A compatibilidade do sistema de inicialização com `SysVinit` é mantida em algumas distribuições, como o Ubuntu, por meio do arquivo de configuração `/etc/init/rc-sysinit.conf`. Trata-se de um dos arquivos de configuração utilizados para a inicialização do sistema, como mostrado anteriormente na Figura 15.3. No exemplo a seguir, você pode ver que o `Upstart` verifica se existe um arquivo `/etc/inittab` e também executa os scripts que ainda podem estar no diretório `/etc/init.d/rcS`:

```
$ cat /etc/init/rc-sysinit.conf
# rc-sysinit - System V initialisation compatibility
```

```

#
# This task runs the old System V-style system initialisation
scripts,
# and enters the default runlevel when finished.
...
start on (filesystem and static-network-up) or failsafe-boot
stop on runlevel
# Default runlevel, this may be overridden on the kernel command-
line
# or by faking an old /etc/inittab entry
env DEFAULT_RUNLEVEL=2

emits runlevel
...
task

script
    # Check for default runlevel in /etc/inittab
    if [ -r /etc/inittab ]
    then
        eval "$(sed -nre 's/^[^#][^:]*:([0-6sS]):initdefault:
.*\/DEFAULT_RUNLEVEL="\1";/p' /etc/inittab || true)"
    fi

# Check kernel command-line for typical arguments
for ARG in $(cat /proc/cmdline)
do
case "${ARG}" in
-b|emergency)
# Emergency shell
[ -n "${FROM_SINGLE_USER_MODE}" ] || sulogin
;;
[0123456sS])
# Override runlevel
DEFAULT_RUNLEVEL="${ARG}"
;;
-s|single)
# Single user mode
[ -n "${FROM_SINGLE_USER_MODE}" ] || DEFAULT_RUNLEVEL=S
;;
esac
done

# Run the system initialisation scripts
[ -n "${FROM_SINGLE_USER_MODE}" ] || /etc/init.d/rcS

# Switch into the default runlevel

```

```
telinit "${DEFAULT_RUNLEVEL}"  
end script
```

Como você pode ver no exemplo anterior, o conceito de runlevel é mantido no daemon Upstart init. Na verdade, não há sequer um sinal de runlevel que Upstart pode emitir.

man -k “event signal”

```
control-alt-delete (7) - event signalling console press of Control  
Alt-Delete  
keyboard-request (7) - event signalling console press of Alt-  
UpArrow  
power-status- (7) - event signalling change of power status  
changed  
runlevel (7) - event signalling change of system runleve  
started (7) - event signalling that a job is running  
starting (7) - event signalling that a job is starting  
startup (7) - event signalling system startup  
stopped (7) - event signalling that a job has stopped  
stopping (7) - event signalling that a job is stopping
```

Mudar para um runlevel ainda é permitido por meio dos comandos init ou telinit. Qualquer evento de runlevel é tratado pela tarefa rc.

```
$ initctl status rc  
rc stop/waiting
```

O arquivo de configuração do trabalho de tarefa rc é mostrado a seguir. Quando um evento de runlevel é emitido, o arquivo de configuração rc chama o script /etc/rc.d/rc. Quando chamado, o script /etc/rc.d/rc executa os scripts localizados no diretório /etc/rc.d/rc#.d, onde # é o runlevel escolhido. Isso fornece retrocompatibilidade do runlevel com o SysVinit.

```
$ cat /etc/init/rc.conf  
# rc - System V runlevel compatibility  
#  
# This task runs the old sysv-rc runlevel scripts. It  
# is usually started by the telinit compatibility wrapper.  
start on runlevel [0123456]  
stop on runlevel [!$RUNLEVEL]  
task
```

```
export RUNLEVEL
console output
exec /etc/rc.d/rc $RUNLEVEL
```

Se voltar e examinar o arquivo `/etc/inittab` na seção do daemon SysVinit clássico, você vai notar que `/etc/inittab` também tratou do carregamento dos processos `getty` ou `mingetty`. O daemon `Upstart init` trata dessa tarefa via `start-ttys`.

```
# initctl status start-ttys
start-ttys stop/waiting
```

O arquivo de configuração do trabalho de tarefa `start-ttys` é mostrado a seguir. Quando um evento de `runlevel` é emitido, o arquivo de inicialização `start-ttys` carrega o processo `getty` ou `mingetty`.

```
$ cat /etc/init/start-ttys.conf
#
# This service starts the configured number of gettys.
start on stopped rc RUNLEVEL=[2345]
env ACTIVE_CONSOLES=/dev/tty[1-6]
env X_TTY=/dev/tty1
task
script
  . /etc/sysconfig/init
  for tty in $(echo $ACTIVE_CONSOLES) ; do
    [ "$RUNLEVEL" = "5" -a "$tty" = "$X_TTY" ] && continue
    initctl start tty TTY=$tty
  done
end script
```

Embora o daemon `Upstart init` forneça retrocompatibilidade com o daemon SysVinit clássico, e seja uma máquina de estado que pode lidar com eventos em constante mudança em um servidor, ele não é o único daemon `init` moderno disponível para o servidor Linux. Outro daemon `init` moderno é `systemd`.

Entendendo `systemd init`

O daemon `systemd init` também é chamado de daemon `systemd` ou `daemon system`. `systemd` pode substituir os daemons SysVinit e `Upstart init`. Esse daemon `init` moderno atualmente funciona no Fedora 15 e versões superiores e é compatível tanto com SysVinit como com `Upstart`. O tempo de inicialização do sistema é reduzido por `systemd` porque ele inicia menos serviços e de forma paralela. Além disso, `systemd` pode lidar com um ambiente fluido

porque supervisiona todos os processos em todo o servidor Linux.

Aprendendo noções básicas sobre systemd

Com o daemon `SysVinit`, os serviços são parados e iniciados com base em níveis de execução. `systemd` também está interessado nos níveis de execução, mas eles são chamados de `target units`. Unidades são o foco de `systemd`. Uma *unidade* é um grupo constituído por um nome, um tipo e um arquivo de configuração, e se concentra em um determinado serviço ou ação. Os oito tipos de `systemd units` são:

- `automount`
- `device`
- `mount`
- `path`
- `serviço`
- `snapshot`
- `socket`
- `target`

As duas principais unidades `systemd` com que você precisa se preocupar para lidar com serviços são `service units` e `target units`. Uma *unidade de serviço* (`service unit`) serve para gerenciar daemons em seu servidor Linux. Uma *unidade alvo* (`target unit`) é simplesmente um grupo de outras unidades.

O exemplo a seguir mostra várias `systemd service units` e `target units`. As `service units` têm nomes de daemon familiares, como `cups` and `sshd`. Note que o nome de cada `service unit` termina com `.service`. As `target units` mostradas aqui têm nomes como `sysinit`. (`sysinit` é usado para iniciar os serviços na inicialização do sistema.) O nome das `target units` termina com `.target`.

```
# systemctl list-units | grep .service
...
cups.service          loaded active running
                        CUPS Printing Service
dbus.service          loaded active running
                        D-Bus System Message
                        Bus
...
NetworkManager.service loaded active running
                        Network Manager
```

```

prefdm.service          loaded active running
                        Display Manager

remount-rootfs.service loaded active exited
                        Remount Root FS

rsyslog.service        loaded active running
                        System Logging
                        Service

...

sshd.service         loaded active running
                        OpenSSH server daemon

system-
s...yboard.service     loaded active running
                        System Setup Keyboard

systemd-logind.service loaded active running
                        Login Service

...

#
# systemctl list-units | grep .target
basic.target           loaded active active
                        Basic System

cryptsetup.target     loaded active active
                        Encrypted Volumes

getty.target          loaded active active
                        Login Prompts

graphical.target      loaded active active
                        Graphical Interface

local-fs-pre.target   loaded active active local-   loaded active
                        Local File Systems fs.target active Local
                        (Pre)                               File Systems

multi-user.target     loaded active active
                        Multi-User

network.target        loaded active active
                        Network

remote-fs.target      loaded active active
                        Remote File Systems

sockets.target        loaded active active
                        Sockets

sound.target          loaded active active
                        Sound Card

swap.target           loaded active active

```

	Swap
sysinit.target	loaded active active System Initialization
syslog.target	loaded active active Syslog

Os arquivos de configuração das unidades de sistema Linux estão localizados nos diretórios `/lib/systemd/system` e `/etc/systemd/system`. Você pode utilizar o comando `ls` para examinar esses diretórios, mas o método preferido é usar uma opção no comando `systemctl`, como segue:

```
# systemctl list-unit-files --type=service
UNIT FILE                                STATE
...
cups.service                             enabled
...
dbus.service static
...
NetworkManager.service                  enabled
...
poweroff.service                         static
...
sshd.service                             enabled
sssd.service                             disabled
...
134 unit files listed.
```

Os arquivos de configuração de unidade mostrados no código anterior estão associados a uma `service unit`. Os arquivos de configuração de `Target units` podem ser visualizados pelo seguinte método.

```
# systemctl list-unit-files --type=target
UNIT FILE                                STATE
anaconda.target                          static
basic.target                              static
bluetooth.target                          static
cryptsetup.target                         static
ctrl-alt-del.target                       disabled
```

```

default.target                               enabled
...
shutdown.target                             static
sigpwr.target                               static
smartcard.target                           static
sockets.target                              static
sound.target                                static
swap.target                                 static
sysinit.target                              static
syslog.target                               static
time-sync.target                            static
umount.target                               static
43 unit files listed.

```

Observe que ambos os exemplos de arquivo de configuração de unidade mostrados exibem as unidades com um status de `static`, `enabled` ou `disabled`. O status `enabled` significa que a unidade está habilitada. O status `disabled` significa que a unidade está desabilitada. O próximo status, `static`, é um pouco confuso. Significa “estaticamente habilitada” e isso significa que a unidade está habilitada por padrão e não pode ser desabilitada, nem por `root`.

Os arquivos de configuração `service unit` contêm uma grande quantidade de informações, tais como quais outros serviços devem ser iniciados, quando podem ser iniciados, qual arquivo ambiental usar etc. O exemplo a seguir mostra o arquivo de configuração de unidade do `sshd`:

```

# cat /lib/systemd/system/sshd.service
[Unit]
Description=OpenSSH server daemon
After=syslog.target network.target auditd.service

[Service]
EnvironmentFile=/etc/sysconfig/sshd
ExecStart=/usr/sbin/sshd -D $OPTIONS
ExecReload=/bin/kill -HUP $MAINPID

[Install]
WantedBy=multi-user.target
#

```

Esse arquivo de configuração de `service unit` básico tem as seguintes opções:

- `Description` — Essa é uma descrição livre (linha de comentário) do serviço.
- `After` — Essa definição configura a ordenação. Em outras palavras, ela lista as unidades que

devem ser ativadas antes de esse serviço ser iniciado.

- `Environment File` — O arquivo de configuração do serviço.
- `ExecStart` — O comando usado para iniciar esse serviço.
- `ExecReload` — O comando usado para recarregar esse serviço.
- `WantedBy` — Isso identifica a `target unit` a que esse serviço pertence.

Note que a `target unit`, `multi-user.target`, é utilizada no arquivo de configuração `sshd service unit`. A `sshd service unit` é procurada pela `multi-user.target`. Em outras palavras, quando a unidade `multi-user.target` é ativada, a `sshd service unit` é iniciada.

Você pode ver as várias unidades que uma `target unit` ativará usando o seguinte comando:

```
# systemctl show --property "Wants" multi-user.target  
Wants=mutipathd.service avahi-daemon.service sshd-keygen.se  
(END) q
```

Infelizmente, o comando `systemctl` não formata bem a saída disso. O texto literalmente rola para fora da tela e não é possível ver os resultados completos. E você tem que digitar um `q` para voltar ao prompt de comando. Para corrigir esse problema, redirecione a saída por meio de alguns comandos de formatação para produzir uma listagem elegante, ordenada alfabeticamente, como mostra o exemplo a seguir.

```
# systemctl show --property "Wants" multi-user.target |  
> fmt -10 | sed 's/Wants=//g' | sort  
abrt-ccpp.service  
abrt-d.service  
abrt-oops.service  
abrt-vmcore.service  
atd.service  
auditd.service  
avahi-daemon.service  
cron.d.service  
cups.path  
dbus.service  
fcoe.service  
getty.target  
irqbalance.service  
iscsid.service  
iscsi.service  
livesys-late.service  
livesys.service  
lldpad.service
```

```
mcelog.service
mdmonitor.service
multipathd.service
netfs.service
NetworkManager.service
plymouth-quit.service
plymouth-quit-wait.service
remote-fs.target
rsyslog.service
sendmail.service
sm-client.service
sshd-keygen.service
sshd.service
systemd-ask-password-wall.path
systemd-logind.service
systemd-update-utmp-runlevel.service
systemd-user-sessions.service
#
```

Essa tela mostra todos os serviços e várias outras unidades que serão ativadas (iniciadas), incluindo o `sshd`, quando a unidade `multi-user.target` for ativada. Lembre-se de que uma `target unit` é simplesmente um agrupamento de outras unidades, como mostrado no exemplo anterior. Além disso, observe que as unidades desse grupo não são todas `service units`. Há `path units` e também outra `target unit`.

A `target unit` tem tanto `Wants` como requisitos, chamados `Requires`. Um `Wants` significa que todas as unidades listadas são acionadas para ativar (iniciar). Se elas falharem ou não puderem ser iniciadas, sem problema — a `target unit` continua alegremente. O exemplo anterior é uma exibição de `Wants` somente.

Um `Requires` é muito mais rigoroso e potencialmente catastrófico do que um `Wants`. Um `Requires` significa que todas as unidades indicadas são acionadas para ativar (iniciar). Se elas falharem ou não puderem ser iniciadas, toda a unidade (grupo de unidades) é desativada.

Você pode ver as várias unidades de uma `target unit` `Requires` (deve ativar ou a unidade falhará) usando o comando no exemplo a seguir. Observe que a saída de `Requires` é muito mais curta do que a de `Wants` para o `multi-user target`. Assim, nenhuma formatação especial da saída é necessária.

```
# systemctl show --property "Requires" multi-user.target
Requires=basic.target
```

A `target units` também têm arquivos de configuração, assim como as unidades de serviço. O exemplo a seguir mostra o conteúdo do arquivo de configuração `multi-user.target`.

```
# cat /lib/systemd/system/multi-user.target
```

```
# This file is part of systemd.  
#  
...  
[Unit]  
Description=Multi-User  
Requires=basic.target  
Conflicts=rescue.service rescue.target  
After=basic.target rescue.service rescue.target  
AllowIsolate=yes  
  
[Install]  
Alias=default.target
```

Esse arquivo de configuração de `target unit` básico tem as seguintes opções:

- `Description` — Isso é apenas uma descrição livre do alvo.
- `Requires` — Se este `multi-user.target` for ativado, a `target unit` listada também será ativada. Se a `target unit` listada estiver desativada ou falhar, então `multi-user.target` será desativado. Se não houver opções `After` e `Before`, então, tanto o `multi-user.target` como a `target unit` listada serão ativados simultaneamente.
- `Conflicts` — Essa configuração evita conflitos de serviços. Iniciar `multi-user.target` para os alvos e serviços listados e vice-versa.
- `After` — Essa definição configura a ordenação. Em outras palavras, essa opção determina que as unidades devem ser ativadas antes de iniciar esse serviço.
- `AllowIsolate` — Essa opção é uma booleana do tipo `yes` ou `no`. Se configurada como `yes`, essa `target unit`, `multi-user.target`, é ativada junto com suas dependências e todas as outras são desativadas.
- `ExecStart` — Esse comando inicia o serviço.
- `ExecReload` — Esse comando recarrega o serviço.
- `Alias` — Com esse comando, `systemd` vai criar um vínculo simbólico a partir dos nomes de `target unit` listados com essa unidade, `multi-user.target`.

Para obter mais informações sobre esses arquivos de configuração e suas opções, digite `man systemd.service`, `man systemd.target` e `man systemd.unit` na linha de comando.

Para o servidor Linux usando `systemd init`, o processo de inicialização será mais fácil de seguir, agora que você entende `systemd target units`. Na inicialização, `systemd` ativa a unidade `default.target`. Essa unidade é transformada em alias para `multi-user.target` ou `graphical.target`. Assim, dependendo do alias configurado, os serviços visados pela `target unit` são iniciados. A Figura 15.4 mostra o curso de ação de `systemd init` em

carregar os processos corretos na inicialização do sistema. Se você está acompanhando os exemplos utilizados nesta seção, o `default.target` é transformado em alias para `multiuser.target`, como mostrado na Figura 15.4.

Se precisar de mais ajuda para entender `systemd init`, há uma excelente documentação feita pelo próprio autor em <http://0pointer.de/blog/projects/systemd-docs.html>. Além disso, você pode digitar **man -k systemd** na linha de comando para obter uma listagem das várias utilidades de `systemd` nas páginas `man`.

ro de ação de `systemd` `init` na inicialização do servidor.

Aprendendo a retrocompatibilidade de `systemd` com `SysVinit`

O daemon `systemd` `init` manteve retrocompatibilidade com o daemon `SysVinit`. Isso permite às distribuições Linux lentamente migrarem para `systemd`.

Embora os níveis de execução não sejam realmente parte de `systemd`, a infraestrutura de `systemd` foi criada para oferecer compatibilidade com o conceito de níveis de execução. Há sete arquivos de configuração de `target` `units` criados especificamente para retrocompatibilidade com `SysVinit`:

- `runlevel0.target`
- `runlevel1.target`
- `runlevel2.target`
- `runlevel3.target`
- `runlevel4.target`
- `runlevel5.target`
- `runlevel6.target`

Como você provavelmente já descobriu, há um arquivo de configuração de `target` `unit` para cada um dos sete níveis de execução clássicos de `SysVinit`. Esses arquivos de configuração de `target` `unit` são simbolicamente vinculados aos arquivos de configuração de `target` `unit` que mais se aproximam da ideia do `runlevel` original. No exemplo a seguir, são mostrados vínculos simbólicos para `runlevel` `target` `units`. Note que as `runlevel` `target` `units` para `runlevel` 2, 3 e 4 são simbolicamente vinculadas a `multi-user.target`. A unidade `multi-user.target` é semelhante ao legado do Extended Multi-user Mode.

```
# ls -l /lib/systemd/system/runlevel*.target
lrwxrwxrwx. 1 root root 15 Mar 27 15:39
/lib/systemd/system/runlevel0.target -> poweroff.target
lrwxrwxrwx. 1 root root 13 Mar 27 15:39
/lib/systemd/system/runlevel1.target -> rescue.target
lrwxrwxrwx. 1 root root 17 Mar 27 15:39
/lib/systemd/system/runlevel2.target -> multi-user.target
lrwxrwxrwx. 1 root root 17 Mar 27 15:39
/lib/systemd/system/runlevel3.target -> multi-user.target
lrwxrwxrwx. 1 root root 17 Mar 27 15:39
/lib/systemd/system/runlevel4.target -> multi-user.target
```

```
lrwxrwxrwx. 1 root root 16 Mar 27 15:39
/lib/systemd/system/runlevel5.target -> graphical.target
lrwxrwxrwx. 1 root root 13 Mar 27 15:39
/lib/systemd/system/runlevel6.target -> reboot.target

#
```

O arquivo `/etc/inittab` ainda existe, mas só contém comentários dizendo que esse arquivo de configuração não é usado e dá algumas informações básicas sobre `systemd`. O arquivo `/etc/inittab` não tem mais nenhum verdadeiro uso funcional. O exemplo a seguir é de um arquivo `/etc/inittab` em um servidor Linux que usa `systemd`.

```
# cat /etc/inittab
# inittab is no longer used when using systemd.
#
# ADDING CONFIGURATION HERE WILL HAVE NO EFFECT ON YOUR SYSTEM.
#
# Ctrl-Alt-Delete is handled by
/etc/systemd/system/ctrl-alt-del.target
#
# systemd uses 'targets' instead of runlevels.
By default, there are two main targets:
#
# multi-user.target: analogous to runlevel 3
# graphical.target: analogous to runlevel 5
#
# To set a default target, run:
#
# ln -s /lib/systemd/system/<target name>.target
/etc/systemd/system/default.target
```

O arquivo `/etc/inittab` explica que, se você quer algo semelhante a um clássico `runlevel 3` ou `5` como seu `runlevel` padrão, terá de criar um vínculo simbólico a partir da unidade `default.target` para o `runlevel target` unit de sua escolha. Para verificar a que `default.target` está simbolicamente vinculado (ou, em termos de legado, verificar o `runlevel` padrão), use o comando mostrado aqui. Você pode ver que no servidor Linux, o padrão é iniciar no `runlevel 3` legado.

```
# ls -l /etc/systemd/system/default.target
lrwxrwxrwx. 1 root root 36 Mar 13 17:27
/etc/systemd/system/default.target ->
/lib/systemd/system/runlevel3.target
```

A capacidade de mudar níveis de execução usando o comando `init` ou `telinit` ainda está disponível. Quando emitido, qualquer um dos comandos será traduzido em uma solicitação de

ativação de `systemd target unit`. Portanto, digitar `init 3` na linha de comando na verdade emite o comando `systemctl isolate multi-user.target`. Além disso, você ainda pode usar o comando `runlevel` para determinar o `runlevel` legado atual, mas isso é fortemente desencorajado.

O `SysVinit /etc/inittab` clássico cuidava de carregar os processos `getty` ou `mingetty`. `systemd init` lida com isso via unidade `getty.target`. `getty.target` é ativado pela unidade `multi-user.target`. Você pode ver como essas duas `target units` estão vinculadas com o seguinte comando:

```
# systemctl show --property "WantedBy" getty.target
WantedBy=multi-user.target
```

A partir do Fedora 16, a conversão de serviços básicos de `SysVinit` para `systemd` está completa. Mas para saber o estado atual de um determinado serviço, visite a página web de compatibilidade de `systemd` em

<http://fedoraproject.org/wiki/User:Johannbg/QA/Systemd/compatibility>

Agora que você tem uma compreensão básica dos `daemons init` clássicos e modernos, é hora de fazer algumas ações práticas de administração de servidor que envolvem o `daemon init`.

Auditando serviços

Como um administrador de Linux, você precisa auditar os serviços que estão sendo oferecidos em seu servidor com relação à documentação, segurança e solução de problemas. Normalmente, você auditará serviços para fins de documentação se tiver “herdado” um servidor, embora alguns sites façam isso regularmente. Por razões de segurança, será necessário desabilitar e remover qualquer sistema de serviços não utilizados descobertos por meio do processo de auditoria. O mais importante para a solução de problemas é que você precisa ser capaz de rapidamente saber o que deve e o que não deve ser executado em seu servidor Linux.

Naturalmente, saber qual `daemon init` está sendo usado pelo seu servidor Linux é a primeira informação a obter. Como determinar isso foi visto na seção “Entendendo o `daemon init` do Linux” deste capítulo. O resto desta seção está organizado em subseções sobre os vários `daemons init`.

Auditando o `daemon SysVinit` clássico

Para ver todos os serviços que estão sendo oferecidos por um servidor Linux usando o `daemon SysVinit` clássico, use o comando `chkconfig`. O exemplo a seguir mostra os serviços disponíveis em um clássico servidor Linux `SysVinit`. Note que cada `runlevel` (0–6) é mostrado para cada serviço com um status de `on` ou `off`. O status indica se um determinado serviço está iniciado (`on`) ou não (`off`) para esse `runlevel`.

```

# chkconfig --list
ConsoleKit      0:off 1:off 2:off 3:on          4:on  5:on  6:of
NetworkManager 0:off 1:off 2:off 3:off          4:off 5:off 6:of
...
crond           0:off 1:off 2:on  3:on          4:on  5:on  6:of
cups          0:off 1:off 2:on  3:on          4:on  5:on  6:of
...
sshd            0:off 1:off 2:on  3:on          4:on  5:on  6:of
syslog          0:off 1:off 2:on  3:on          4:on  5:on  6:of
tux             0:off 1:off 2:off 3:off 4:off 5:off 6:off
udev-post       0:off 1:off 2:off 3:on          4:on  5:on  6:of
vncserver    0:off 1:off 2:off 3:off          4:off 5:off 6:of
winbind         0:off 1:off 2:off 3:off          4:off 5:off 6:of
wpa_supplicant 0:off 1:off 2:off 3:off          4:off 5:off 6:of
xfs             0:off 1:off 2:on  3:on          4:on  5:on  6:of
ypbind          0:off 1:off 2:off 3:off          4:off 5:off 6:of
yum-updatesd   0:off 1:off 2:off 3:on          4:on  5:on  6:of

```

Alguns serviços no exemplo, nunca são iniciados, como `vncserver`. Outros serviços, como o `daemon cups`, são iniciados nos níveis de execução 2 a 5.

Usando o comando `chkconfig`, você não pode dizer se o serviço está sendo executado. Para fazer isso, você vai precisar usar o comando `service`. Para ajudar a isolar apenas os serviços que estão sendo executados, o comando `service` é redirecionado para o comando `grep` e ordenado, como segue.

```

# service --status-all | grep running... | sort
anacron (pid 2162) is running...
atd (pid 2172) is running...
auditd (pid 1653) is running...
automount (pid 1952) is running...
console-kit-daemon (pid 2046) is running...
crond (pid 2118) is running...
cupsd (pid 1988) is running...
...
sshd (pid 2002) is running...
syslogd (pid 1681) is running...
xfs (pid 2151) is running...
yum-updatesd (pid 2205) is running...

```


Você também pode usar os comandos `chkconfig` e `service` para ver as configurações de um serviço individual. Usando os dois comandos no exemplo a seguir, você pode visualizar as configurações do daemon `cups`.

```
# chkconfig --list cups
cups          0:off  1:off  2:on   3:on   4:on   5:on   6:off
#
# service cups status
cupsd (pid 1988) is running...
```

Você pode ver que `cupsd` está configurado para iniciar em cada `runlevel`, exceto 0, 1 e 6, e, a partir do comando `service`, você pode ver que ele está sendo executado. Além disso, o número de ID de processo (PID) é fornecido para o daemon.

Auditando o daemon Upstart `init`

Para ver todos os serviços executados em um servidor Linux usando o daemon Upstart `init`, use o seguinte comando:

```
# initctl list | grep start/running
tty (/dev/tty3) start/running, process 1163
...
system-setup-keyboard start/running, process 656
prefdm start/running, process 1154
```

Tenha em mente que muitos serviços podem ainda não ter sido portados para o daemon Upstart `init`. Portanto, você também vai precisar usar o comando SysVinit clássico, `service`, para verificar quaisquer serviços SysVinit restantes. Note que em algumas distribuições, você pode ver alguns serviços na saída de *ambos* os comandos, `initctl` e `service`.

```
# service --status-all | grep running
abrtd (pid 1118) is running...
acpid (pid 996) is running...
atd (pid 1146) is running...
...
rsyslogd (pid 752) is running...
sendmail (pid 1099) is running...
...
```

Atenção

Ó porque um serviço não está em estado `running` (funcionando) não significa que ele não está disponível. O serviço pode estar em um estado `stopped/wait` (parado/espera), esperando um evento no sistema. Para ver todos os serviços, não importa seu estado, remova a parte `grep` dos comandos `initctl list` e `service --status-all` anteriores.

Para mostrar o status de um serviço individual, utilize `initctl` se o serviço tiver sido portado para o Upstart; e o comando `service`, se ele ainda não tiver sido portado. O exemplo a seguir mostra dois estados — um serviço que foi portado para o Upstart e um que não foi.

```
# initctl status vpnc-cleanup
vpnc-cleanup stop/waiting
#
# service ssh status
sshd (pid 970) is running...
```

O daemon `ssh` ainda não foi portado, no servidor Linux, para o Upstart. Portanto, `ssh` precisa do comando `service` com a opção de status para ser utilizado para auditoria. O serviço `vpnc-cleanup` é um serviço Upstart, portanto, ele precisa que o comando `initctl status` seja utilizado. Em algumas distribuições, como o Ubuntu, você também pode usar o comando `initctl status` para serviços que ainda não foram migrados para o Upstart.

Auditando init systemd

Para ver todos os serviços que estão sendo oferecidos por um servidor Linux usando `systemd`, use o seguinte comando:

```
# systemctl list-unit-files --type=service | grep -v disabled
```

UNIT	FILE	STATE
	<code>abrt-ccpp.service</code>	enabled
	<code>abrt-oops.service</code>	enabled
	<code>abrt-vmcore.service</code>	enabled
	<code>abrt.service</code>	enabled
	<code>alsa-restore.service</code>	static
	<code>alsa-store.service</code>	static
	<code>anaconda-shell@.service</code>	static
	<code>arp-ethers.service</code>	enabled
	<code>atd.service</code>	enabled
	<code>auditd.service</code>	enabled
	<code>avahi-daemon.service</code>	enabled
	<code>bluetooth.service</code>	enabled
	<code>console-kit-log-system-restart.service</code>	static
	<code>console-kit-log-system-start.service</code>	static
	<code>console-kit-log-system-stop.service</code>	static
	<code>crond.service</code>	enabled

```
cups.service                                enabled
...
sshd-keygen.service                         enabled
sshd.service                                enabled
system-setup-keyboard.service              enabled
...
134 unit files listed.
```

Lembre-se de que as três possibilidades de status para um serviço `systemd` são `enabled`, `disabled` ou `static`. Não é necessário incluir `disabled` nessa auditoria, o que é efetivamente feito usando a opção `-v` no comando `grep`, como mostrado no exemplo anterior. O estado `static` é essencialmente `enabled`, e, portanto, deve ser incluído.

Nota
maioria dos serviços de sistema foi portada para o `systemd`. Mas se quiser confirmar, você pode ver o status a migração de um serviço particular em <http://fedoraproject.org/wiki/User:Johannbg/QA/Systemd/compatibility>.

Para ver se um determinado serviço está sendo executado, use o seguinte comando:

```
# systemctl status cups.service
cups.service - CUPS Printing Service
  Loaded: loaded (/lib/systemd/system/cups.service; enabled)
  Active: active (running) since Mon, 30 Apr 2015 12:36:31 -0400;
  13h ago
  Main PID: 1315 (cupsd)
  CGroup: name=systemd:/system/cups.service
          1315 /usr/sbin/cupsd -f
```

O comando `systemctl` pode ser usado para mostrar o status de um serviço individual. No exemplo anterior, o serviço de impressão foi escolhido. Observe que o nome do serviço é `cups.service`. Uma grande quantidade de informações úteis sobre ele é fornecida aqui, como o fato de que está habilitado e ativo, sua hora de início e seu ID de processo (PID) também.

Agora que você pode auditar os serviços e determinar algumas informações sobre eles, você precisa saber como iniciar, parar e recarregar os serviços em seu servidor Linux.

Parando e iniciando serviços

As tarefas de iniciar, parar e reiniciar serviços normalmente estão relacionadas com necessidades

imediatas — em outras palavras, gerenciar os serviços sem uma reinicialização do servidor. Por exemplo, se você quiser parar temporariamente um serviço, então você está na seção certa. Mas se quiser parar um serviço e não permitir que ele seja reiniciado na reinicialização do servidor, então você precisa realmente desativar o serviço, o que é coberto na seção “Configurando serviços persistentes”, mais adiante, neste capítulo.

Parando e iniciando o daemon SysVinit clássico

O comando principal para parar e iniciar serviços SysVinit é o comando `service`. Com o comando `service`, o nome do serviço que você quer controlar vem em segundo lugar na linha de comando. A última opção é o que você quer fazer com o serviço, `stop`, `start`, `restart` etc. O exemplo a seguir mostra como parar o daemon `cups`. Observe que um `OK` é dado, o que permite que você saiba que `cupsd` foi parado com êxito.

```
# service cups status
cupsd (pid 5857) is running...
#
# service cups stop
Stopping cups:          [ OK ]
#
# service cups status
cupsd is stopped
```

Para iniciar um serviço, basta adicionar uma opção `start` em vez de uma opção `stop` no fim do comando `service`, como segue.

```
# service cups start
Starting cups:          [ OK ]
#
# service cups status
cupsd (pid 6860) is running...
```

Para reiniciar um serviço SysVinit, a opção `restart` é usada. Essa opção para o serviço e, então, imediatamente, o inicia de novo.

```
# service cups restart
Stopping cups:          [ OK ]
Starting cups:          [ OK ]
#
# service cups status
cupsd (pid 7955) is running...
```

Quando um serviço já está parado, um `restart` gera um status `FAILED` na tentativa de pará-lo. Mas como mostrado no exemplo a seguir, o serviço será iniciado com êxito quando uma reinicialização for feita.

```
# service cups stop
Stopping cups:          [ OK ]
#
# service cups restart
Stopping cups:          [FAILED]
Starting cups:          [ OK ]
#
# service cups status
cupsd (pid 8236) is running...
```

Recarregar um serviço é diferente de reiniciar um serviço. Quando você recarrega (`reload`) um serviço, ele em si não está parado, somente os arquivos de configuração dele são carregados novamente. O exemplo a seguir mostra como recarregar o daemon `cups`.

```
# service cups status
cupsd (pid 8236) is running...
#
# service cups reload
Reloading cups:          [ OK ]
#
# service cups status
cupsd (pid 8236) is running...
```

Se um serviço `SysVinit` for interrompido quando você tentar recarregá-lo, você terá um status `FAILED`. Isso é mostrado no exemplo a seguir:

```
# service cups status
cupsd is stopped
#
# service cups reload
Reloading cups:          [FAILED]
```

Parando e iniciando o daemon Upstart init

O comando principal para parar e iniciar serviços Upstart `init` é o comando `initctl`. As opções são muito semelhantes ao comando `service` de `SysVinit`:

- Parando um serviço com Upstart `init` — No exemplo a seguir, o status do daemon `cups` é verificado e, então, parado usando o comando `initctl stop cups.service`.

```
# initctl status cups
cups start/running, process 2390
#
# initctl stop cups
cups stop/waiting
#
# initctl status cups
cups stop/waiting
```

- Iniciando um serviço com Upstart `init` — No exemplo a seguir, o daemon `cups` é parado usando o comando `cups.service`.

```
# initctl start cups
cups start/running, process 2408
#
# initctl status cups
cups start/running, process 2408
```

- Reiniciando um serviço com Upstart `init` — Reiniciar um serviço com Upstart `init` vai parar e, então, iniciar o serviço. Mas o arquivo de configuração não será recarregado.

```
# initctl restart cups
```

```
cups start/running, process 2490
#
# initctl status cups
cups start/running, process 2490
#
```

- Recarregar um serviço com Upstart init — Atualizar *não* irá parar e iniciar o serviço. Isso só carrega o arquivo de configuração novamente. Essa é a opção a usar quando você tiver feito alterações no arquivo de configuração. O exemplo abaixo ilustra como recarregar o daemon cups com `initctl`. Note que o ID do processo (PID) ainda é 2490, que é o mesmo que era no exemplo para reiniciar o daemon cups porque o processo não foi parado e iniciado no processo de recarregamento.

```
#initctl reload cups
#
# initctl status cups
cups start/running, process 2490
```

Nota

Você precisa de privilégios de root para parar e iniciar os serviços, mas não precisa para verificar o status de um serviço.

Parando e iniciando o daemon systemd

Para o daemon `systemd`, o comando `systemctl` funcionará para parar, iniciar, recarregar e reiniciar. As opções para o comando `systemctl` devem parecer familiares.

Parando um serviço com systemd

No exemplo a seguir, o status do daemon cups é verificado e, então, parado usando o comando `systemctl stop cups.service`:

```
# systemctl status cups.service
cups.service - CUPS Printing Service
  Loaded: loaded (/lib/systemd/system/cups.service; enabled)
  Active: active (running) since Mon, 30 Apr 2015 12:36:3...
  Main PID: 1315 (cupsd)
  CGroup: name=systemd:/system/cups.service
          1315 /usr/sbin/cupsd -f

#
# systemctl stop cups.service
#
# systemctl status cups.service
cups.service - CUPS Printing Service
```

```
Loaded: loaded (/lib/systemd/system/cups.service; enabled)
Active: inactive (dead) since Tue, 01 May 2015 04:43:4...
Process: 1315 ExecStart=/usr/sbin/cupsd -f
(code=exited, status=0/SUCCESS)
CGroup: name=systemd:/system/cups.service
```

Observe que quando o estado é obtido, depois de parar o daemon cups, o serviço está **inactive (dead)**, mas ainda considerado **enabled**. Isso significa que o daemon cups ainda será iniciado após a inicialização do servidor.

Iniciando um serviço com systemd

Iniciar o daemon cups é tão fácil quanto pará-lo. O exemplo a seguir demonstra essa facilidade.

```
# systemctl start cups.service
#
# systemctl status cups.service
cups.service - CUPS Printing Service
Loaded: loaded (/lib/systemd/system/cups.service; enabled)
Active: active (running) since Tue, 01 May 2015 04:43:5...
Main PID: 17003 (cupsd)
CGroup: name=systemd:/system/cups.service
17003 /usr/sbin/cupsd -f
```

Depois que o daemon cups é iniciado, usar `systemctl` com a opção de status mostra que o serviço está **active (running)**. Além disso, seu número de ID (PID) de processo, 17003, é mostrado.

Reiniciando um serviço com systemd

Reiniciar um serviço significa que um serviço foi parado e depois reiniciado. Se o serviço não estiver atualmente em execução, reiniciá-lo simplesmente inicia o serviço.

```
# systemctl restart cups.service
#
# systemctl status cups.service
cups.service - CUPS Printing Service
Loaded: loaded (/lib/systemd/system/cups.service; enabled)
Active: active (running) since Tue, 01 May 2015 04:45:2...
Main PID: 17015 (cupsd)
CGroup: name=systemd:/system/cups.service
17015 /usr/sbin/cupsd -f
```

Você também pode executar uma reinicialização condicional de um serviço usando `systemctl`. Um reinício condicional apenas reinicia um serviço se ele estiver sendo executado. Qualquer serviço em um estado inativo não será iniciado.

```
# systemctl status cups.service
cups.service - CUPS Printing Service
  Loaded: loaded (/lib/systemd/system/cups.service; enabled)
  Active: inactive (dead) since Tue, 01 May 2015 06:03:32...
  Process: 17108 ExecStart=/usr/sbin/cupsd -f
  (code=exited, status=0/SUCCESS)
  CGroup: name=systemd:/system/cups.service
```

```
#
# systemctl condrestart cups.service
#
# systemctl status cups.service
cups.service - CUPS Printing Service
  Loaded: loaded (/lib/systemd/system/cups.service; enabled)
  Active: inactive (dead) since Tue, 01 May 2015 06:03:32...
  Process: 17108 ExecStart=/usr/sbin/cupsd -f
  (code=exited, status=0/SUCCESS)
  CGroup: name=systemd:/system/cups.service
```

Observe no exemplo que o daemon cups estava em um estado inativo. Quando o reinício condicional foi emitido, nenhuma mensagem de erro foi gerada! O daemon cups não foi iniciado porque reinícios condicionais afetarão os serviços ativos. Assim, é sempre uma boa prática verificar o status de um serviço depois de parar, iniciar, condicionalmente reiniciar etc.

Recarregando um serviço com systemd

Recarregar um serviço é diferente de reiniciar um serviço. Quando você recarrega (`reload`) um serviço, o serviço em si não está parado. Somente os arquivos de configuração do serviço são carregados novamente.

```
# systemctl reload cups.service
Failed to issue method call: Job type reload is
  not applicable for unit cups.service.
```

```
#
# systemctl reload sshd.service
#
# systemctl status sshd.service
sshd.service - OpenSSH server daemon
  Loaded: loaded (/lib/systemd/system/sshd.service; enabled)
  Active: active (running) since Mon, 30 Apr 2015 12:35:2...
  Process: 17009 ExecReload=/bin/kill -HUP $MAINPID
  (code=exited, status=0/SUCCESS)
  Main PID: 786 (sshd)
  CGroup: name=systemd:/system/sshd.service
  786 /usr/sbin/sshd -D
```


Fazer um `reload` de um serviço, em vez de um `restart`, impedirá que quaisquer operações de serviços pendentes sejam abortadas. O `reload` é um método melhor para um servidor ocupado Linux.

Agora que você sabe como parar e iniciar os serviços para fins de solução de problemas e para emergências, você pode aprender como habilitar e desabilitar serviços.

Configurando serviços persistentes

Você usa `stop` e `start` para necessidades imediatas, não para serviços que precisam ser persistentes. Um serviço *persistente* é aquele que é iniciado na inicialização do servidor. Em geral, serviços que precisam ser definidos como persistentes são novos serviços que o servidor Linux oferece.

Configurando serviços persistentes do daemon SysVinit clássico

Um dos recursos interessantes do daemon `SysVinit` clássico é que tornar persistente um determinado serviço ou remover sua persistência é muito fácil de fazer. Considere os seguintes exemplos:

```
# chkconfig --list cups
cups          0:off 1:off 2:off 3:off 4:off 5:off 6:off
```

Nesse servidor Linux, o daemon `cups` não está iniciado em nenhum `runlevel`, como mostrado com o comando `chkconfig`. Você também pode verificar se há qualquer vínculo simbólico `start (S)` configurado em cada um dos sete diretórios de `runlevel`, `/etc/rc.d/rc?.d`. Lembre-se de que `SysVinit` mantém vínculos simbólicos aqui para iniciar e parar vários serviços em certos níveis de execução. Cada diretório representa um `runlevel` específico, por exemplo, `rc5.d` é para `runlevel 5`. Observe que apenas os arquivos que começam com um `K` são listados e, por isso, há vínculos para eliminar o daemon `cups`. Nenhum é listado com `S`, o que é consistente com `chkconfig` no sentido de que o daemon `cups` não começa em qualquer `runlevel` nesse servidor.

```
# ls /etc/rc.d/rc?.d/*cups
/etc/rc.d/rc0.d/K10cups          /etc/rc.d/rc3.d/K10cups
/etc/rc.d/rc1.d/K10cups          /etc/rc.d/rc4.d/K10cups
/etc/rc.d/rc2.d/K10cups          /etc/rc.d/rc5.d/K10cups
/etc/rc.d/rc6.d/K10cups
```

Para tornar persistente um serviço em um `runlevel` específico, o comando `chkconfig` é usado novamente. Em vez da opção `--list`, a opção `--level` é utilizada, como mostra o seguinte código:

```
# chkconfig --level 3 cups on
#
# chkconfig --list cups
cups          0:off 1:off 2:off 3:on 4:off 5:off 6:off
#
# ls /etc/rc.d/rc3.d/S*cups
/etc/rc.d/rc3.d/S56cups
```

A persistência do serviço em `runlevel 3` é verificada tanto usando o comando `chkconfig --list` como procurando no diretório `rc3.d` todos os arquivos que começam com a letra `S`.

Para tornar persistente um serviço em mais de um `runlevel`, você pode fazer o seguinte:

```
# chkconfig --level 2345 cups on
#
# chkconfig --list cups
cups          0:off 1:off 2:on 3:on 4:on 5:on 6:off
#
# ls /etc/rc.d/rc?.d/S*cups
/etc/rc.d/rc2.d/S56cups  /etc/rc.d/rc4.d/S56cups
/etc/rc.d/rc3.d/S56cups  /etc/rc.d/rc5.d/S56cups
```

Desativar um serviço é tão fácil quanto habilitar um com `SysVinit`. Você só precisa mudar o `on` no comando `chkconfig` para `off`. O exemplo a seguir demonstra o uso do comando `chkconfig` para desativar o serviço `cups` no `runlevel 5`.

```
# chkconfig --level 5 cups off
#
# chkconfig --list cups
cups          0:off 1:off 2:on 3:on 4:on 5:off 6:off
#
# ls /etc/rc.d/rc5.d/S*cups
ls: cannot access /etc/rc.d/rc5.d/S*cups: No such file or directory
```

Como esperado, não há agora nenhuma vinculação simbólica começando com a letra `S` para o daemon `cups` no diretório `/etc/rc.d/rc5.d`.

Configurando serviços persistentes do daemon Upstart init

O daemon `Upstart init` emite o sinal de inicialização que aciona os trabalhos de serviço para iniciar. Na inicialização do servidor, vários trabalhos podem emitir sinais por si próprios. Esses sinais emitidos, então, fazem outros trabalhos iniciarem. Assim, a chave para tornar persistente um serviço é assegurar que o arquivo de definição do serviço seja acionado por um dos sinais emitidos quando o servidor é inicializado.

Lembre-se de que os arquivos de definição de trabalho do daemon `Upstart init` estão localizados em `/etc/init`. Considere o seguinte arquivo de definição de trabalho para o daemon `ssh`:

```
# cat /etc/init/ssh.conf
# ssh - OpenBSD Secure Shell server
#
# The OpenSSH server provides secure shell access to the system.
description "OpenSSH server"
start on filesystem or runlevel [2345]
```

```
stop on runlevel [!2345]
```

```
respawn
```

Para determinar qual ou quais eventos acionaram um serviço, procure `start on` no arquivo de configuração. O daemon `ssh` é acionado por vários possíveis eventos emitidos, `filesystem`, `runlevel 2`, `runlevel 3`, `runlevel 4` ou `runlevel 5`. Basicamente, o daemon `ssh` inicia na inicialização do servidor e é configurado como persistente. A sintaxe para os eventos de `runlevel`, `runlevel [2345]`, é utilizada em muitos dos arquivos de trabalho e indica que o nome “runlevel” pode acabar em 2, 3, 4 ou 5.

Para tornar um trabalho persistente (iniciar na inicialização do sistema), você terá de modificar a linha `start on` em seu arquivo de configuração, de modo que ele inicie em determinados eventos emitidos na inicialização do servidor. Para desativar um trabalho na inicialização, basta colocar um caractere de comentário (`#`) no início da linha `start on`. Veja a discussão sobre `Upstart` na seção “Adicionando serviços novos ou personalizados” para obter uma explicação mais aprofundada desses arquivos de configuração.

Configurando serviços persistentes de `systemd` init

Para o daemon `systemd`, novamente o comando `systemctl` é usado. Com ele, você pode desativar e ativar serviços no servidor Linux.

Ativando um serviço com `systemd`

Usar a opção `enable` no comando `systemctl` definirá para que um serviço inicie sempre na inicialização (ser persistente). Eis exatamente como fazer isso:

```
# systemctl status cups.service
cups.service - CUPS Printing Service
  Loaded: loaded (/lib/systemd/system/cups.service; disabled)
  Active: inactive (dead) since Tue, 01 May 2015 06:42:38 ...
  Main PID: 17172 (code=exited, status=0/SUCCESS)
```

```
CGroup: name=systemd:/system/cups.service
```

```
#
```

```
# systemctl enable cups.service
```

```
ln -s '/lib/systemd/system/cups.service'
'/etc/systemd/system/printer.target.wants/cups.service
```

```
ln -s '/lib/systemd/system/cups.socket'
'/etc/systemd/system/sockets.target.wants/cups.socket'
```

```
ln -s '/lib/systemd/system/cups.path' `
/etc/systemd/system/multi-user.target.wants/cups.path'
```

```
#  
# systemctl status cups.service  
cups.service - CUPS Printing Service  
  Loaded: loaded (/lib/systemd/system/cups.service; enabled)  
  Active: inactive (dead) since Tue, 01 May 2015 06:42:38...  
  Main PID: 17172 (code=exited, status=0/SUCCESS)  
  CGroup: name=systemd:/system/cups.service
```

Observe que o status de `cups.service` muda de `disabled` para `enabled` depois de usar a opção `enable` em `systemctl`. Além disso, observe que a opção `enable` simplesmente cria alguns vínculos simbólicos. Você pode ser tentado a criar esses vínculos por conta própria, mas o método preferido é utilizar o comando `systemctl` para conseguir isso.

Desativando (removendo) um serviço com systemd

Você pode usar a opção `disable` no comando `systemctl` para impedir que um serviço inicie na inicialização do sistema. Mas isso não para imediatamente o serviço. Você precisa usar a opção `stop` discutida na seção “Parando um serviço com systemd”. O exemplo a seguir mostra como desativar (`disable`) um serviço atualmente ativado (`enabled`).

```
# systemctl disable cups.service  
rm '/etc/systemd/system/printer.target.wants/cups.service'  
rm '/etc/systemd/system/sockets.target.wants/cups.socket'  
rm '/etc/systemd/system/multi-user.target.wants/cups.path'  
#  
# systemctl status cups.service  
cups.service - CUPS Printing Service  
  Loaded: loaded (/lib/systemd/system/cups.service; disabled)  
  Active: active (running) since Tue, 01 May 2015 06:06:41...  
  Main PID: 17172 (cupsd)  
  CGroup: name=systemd:/system/cups.service  
          17172 /usr/sbin/cupsd -f
```

A opção `disable` simplesmente elimina alguns arquivos por meio do método preferido do comando `systemctl`. Repare também no exemplo anterior que, embora o serviço `cups` agora esteja desativado, o daemon `cups` ainda está `active (running)`.

A `systemd` `init` tem alguns serviços que não podem ser desativados. Esses serviços são `static`. Considere o seguinte serviço, `dbus.service`:

```
# systemctl status dbus.service  
dbus.service - D-Bus System Message Bus  
  Loaded: loaded (/lib/systemd/system/dbus.service; static)  
  Active: active (running) since Mon, 30 Apr 2015 12:35:...  
  Main PID: 707 (dbus-daemon)
```

```
...
#
# systemctl disable dbus.service
#
# systemctl status dbus.service
dbus.service - D-Bus System Message Bus
  Loaded: loaded (/lib/systemd/system/dbus.service; static)
  Active: active (running) since Mon, 30 Apr 2015 12:35:...
  Main PID: 707 (dbus-daemon)
...
```

Quando o comando `systemctl disable` é emitido em `dbus.service`, ele é simplesmente ignorado. Lembre-se de que `static` significa “estaticamente habilitado” e isso quer dizer que o serviço é ativado por padrão e não pode ser desativado, nem por `root`. Quaisquer serviços que não devem ser ativados em seu servidor Linux são configurados como `static`.

Agora que você entende como ativar (ou desativar) serviços individuais para serem persistentes, você precisa olhar para os grupos de serviços como um todo. A próxima seção aborda como iniciar grupos de serviços na inicialização do sistema.

Configurando um runlevel ou uma target unit padrão

Enquanto um serviço persistente é aquele que é iniciado na inicialização do servidor, um `runlevel` ou uma `target unit` (padrão) persistente é um grupo de serviços que são iniciados na inicialização do sistema. Os dois daemons clássicos, `SysVinit` e `Upstart`, definem esses grupos de serviços como `runlevels`, enquanto `systemd` chama-os de `target units`.

Configurando o nível de execução padrão do SysVdaemon init clássico

Você configura o `runlevel` persistente para um servidor Linux usando o daemon `SysVinit` no arquivo `/etc/inittab`. Uma parte desse arquivo é mostrada aqui:

```
# cat /etc/inittab
#
# inittab          This file describes how the INIT process should
#                  set up the system in a certain run-level.
...
id:5:initdefault:

# System initialization.
si::sysinit:/etc/rc.d/rc.sysinit
...
```

A linha em negrito no exemplo mostra que o `runlevel` padrão atual é o `runlevel 5`. Para mudar isso, basta editar o arquivo `/etc/inittab` usando seu editor favorito e trocar o `5` por um dos seguintes níveis de execução: `2`, `3` ou `4`. Não use os níveis de execução `0` ou `6` nesse arquivo! Isso faria

o servidor ficar suspenso ou reiniciar quando ligado.

Configurando o nível de execução padrão do daemon Upstart init

Algumas distribuições ainda usam o arquivo `/etc/inittab` para definir o `runlevel` padrão, enquanto outras usam o arquivo `/etc/init/rc-sysinit.conf`.

O daemon `Upstart init` do Fedora e do RHEL ainda usam o arquivo `/etc/inittab`. Portanto, basta alterar o `runlevel` padrão como você faria em um sistema `SysVinit`.

O daemon `Upstart init` do Ubuntu usa o arquivo `/etc/init/rc-sysinit.conf` para definir o `runlevel` padrão, uma parte do qual é mostrada no código que se segue. A linha de código a alterar é `env DEFAULT_RUNLEVEL=`. Basta editar o arquivo e mudar esse número para o `runlevel` que você deseja. Mas lembre-se de que o `runlevel 2` do Ubuntu é equivalente aos níveis de execução 3, 4 e 5.

```
$ cat /etc/init/rc-sysinit.conf
# rc-sysinit - System V initialisation compatibility
...
# Default runlevel, this may be overridden on the kernel command-
line
# or by faking an old /etc/inittab entry
env DEFAULT_RUNLEVEL=2
```

Configurando a target unit padrão de systemd init

`systemd init` usa o termo “`target units`” para os grupos de serviços a serem iniciados. A seguir são apresentadas as várias `target units` que você pode configurar para serem persistentes e suas `target units` equivalentes compatíveis, específicas do nível de execução.

- `multi-user.target =`
 - `runlevel2.target`
 - `runlevel3.target`
 - `runlevel4.target`
- `runlevel5.target = graphical.target`

A `target unit` persistente é configurada via um vínculo simbólico para o arquivo `default.target`. Considere o seguinte:

```
# ls -l /etc/systemd/system/default.target
lrwxrwxrwx. 1 root root 36 Mar 13 17:27
/etc/systemd/system/default.target ->
/lib/systemd/system/runlevel5.target
```

```
#  
# ls -l /lib/systemd/system/runlevel5.target  
lrwxrwxrwx. 1 root root 16 Mar 27 15:39  
/lib/systemd/system/runlevel5.target ->  
graphical.target
```

O exemplo mostra que a `target` unit persistente atual no servidor é `runlevel5.target` porque `default.target` é um vínculo simbólico para o arquivo de unidade `runlevel5.target`. Mas observe que `runlevel5.target` também é um vínculo simbólico que aponta para `graphical.target`. Assim, a `target` unit persistente atual desse servidor é `graphical.target`.

Para configurar uma `target` unit diferente para ser persistente, você só precisa mudar o vínculo simbólico para `default.target`. Para ser coerente, atenha-se às `runlevel target` units se elas forem utilizadas em seu servidor.

O exemplo a seguir muda a `target` unit persistente do servidor de `graphical.target` para `multi-user.target` mudando o vínculo simbólico `default.target` de `runlevel3.target` para `runlevel5.target`. A opção `f` é utilizada no comando `ls -s` para forçar que qualquer vínculo simbólico atual seja quebrado e que o novo vínculo simbólico designado seja imposto.

```
# ls -l /lib/systemd/system/runlevel3.target  
lrwxrwxrwx. 1 root root 17 Mar 27 15:39  
/lib/systemd/system/runlevel3.target ->  
multi-user.target  
#  
# ln -sf /lib/systemd/system/runlevel3.target  
/etc/systemd/system/default.target  
#  
# ls -l /etc/systemd/system/default.target  
lrwxrwxrwx. 1 root root 36 May 1 10:06  
/etc/systemd/system/default.target ->  
/lib/systemd/system/runlevel3.target
```

Quando o servidor for reinicializado, o `multi-user.target` será a `target` unit persistente. Quaisquer serviços na unidade `multi-user.target` serão iniciados (ativados) nesse momento.

Adicionando serviços novos ou personalizados

Ocasionalmente, você tem de adicionar um novo serviço ao seu servidor Linux. Além disso, você pode precisar personalizar um determinado serviço. Quando essas necessidades surgem, você deve seguir passos específicos para o daemon `init` de seu servidor Linux assumir o gerenciamento do serviço ou reconhecer a personalização do mesmo.

Adicionando novos serviços ao SysVdaemon init clássico

Ao adicionar um novo serviço ou personalizar um servidor SysVinit Linux, você precisa completar três etapas, a fim de ter o serviço gerenciado pelo SysVinit.

1. **Crie um arquivo de script de serviço novo ou personalizado.**
2. **Mova o script de serviço novo ou personalizado para o local adequado para ser gerenciado por SysVinit.**
3. **Adicione o serviço a um ou mais `runlevel(s)` específico(s).**

Passo 1: Crie um arquivo de script de serviço novo ou personalizado

Se você estiver personalizando um script de serviço, basta fazer uma cópia do arquivo da unidade original a partir do arquivo `/etc/rc.d/init.d` e adicionar as personalizações desejadas.

Se estiver criando um novo roteiro, você precisa ter certeza de lidar com todas as opções que você quer que o comando `service` aceite para seu serviço, como `start`, `stop`, `restart` etc.

Para um novo script, especialmente se você nunca criou um script de serviço antes, seria prudente fazer uma cópia de um script de serviço atual a partir do arquivo `/etc/rc.d/init.d` e modificá-lo para atender as necessidades de seu novo serviço. Considere o seguinte exemplo parcial do script do serviço `cupsd`:

```
# cat /etc/rc.d/init.d/cups
#!/bin/sh
#
...

start () {
    echo -n $"Starting $prog: "
    # start daemon
    daemon $DAEMON
    RETVAL=$?
    echo
    [ $RETVAL = 0 ] && touch /var/lock/subsys/cups
    return $RETVAL
}

stop () {
    # stop daemon
    echo -n $"Stopping $prog: "
    killproc $DAEMON
    RETVAL=$?
    echo
    [ $RETVAL = 0 ] && rm -f /var/lock/subsys/cups
```



```

}
  restart () {

stop
start

}

case $1 in

...

```

O script de serviço do daemon `cups` começa criando funções para cada uma das opções `start`, `stop` e `restart`. Se você se sentir desconfortável com a codificação de script de shell, consulte o Capítulo 7, “Escrevendo scripts de shell simples” ou consulte uma obra de referência, como *Linux Command Line and Shell Scripting Bible* (Wiley, 2011), para melhorar suas habilidades.

Passo 2: Mova o script de serviço

Uma vez que você tenha modificado ou criado e testado o arquivo de script de seu serviço, você pode movê-lo para o local adequado, `/etc/rc.d/init.d`:

```

# cp My_New_Service /etc/rc.d/init.d
#
# ls /etc/rc.d/init.d/My_New_Service
/etc/rc.d/init.d/My_New_Service

```

Passo 3: Adicione o serviço a runlevels

Este passo final é necessário apenas se você quiser que o serviço seja persistente em certos níveis de execução. Você deve criar um vínculo simbólico para cada `runlevel` em que deseja que o serviço seja persistente. Abaixo, estão as ações necessárias para realizar esta etapa final.

1. **Verifique cada diretório `runlevel` em que você quer que o serviço inicie e determine qual deve ser o número `S` apropriado para o seu serviço. Por exemplo, o `My_New_Service` deve ser iniciado no `runlevel 3` e depois de todos os serviços de `runlevel 3` serem iniciados. Portanto, `S100` é um número `S` apropriado para o nome de arquivo simbólico, como mostrado aqui:**

```

# ls /etc/rc.d/rc3.d/S*
/etc/rc.d/rc3.d/S00microcode_ctl
/etc/rc.d/rc3.d/S04readahead_early
/etc/rc.d/rc3.d/S05kudzu
...
/etc/rc.d/rc3.d/S98haldaemon
/etc/rc.d/rc3.d/S99firstboot
/etc/rc.d/rc3.d/S99local
/etc/rc.d/rc3.d/S99smartd
# ln -s /etc/rc.d/init.d/My_New_Service

```

```
/etc/rc.d/rc3.d/S100My_New_Service
#
# ls -l /etc/rc.d/rc3.d/S100My_New_Service
lrwxrwxrwx 1 root root 31 2015-05-07 11:01
/etc/rc.d/rc3.d/S100My_New_Service ->
/etc/rc.d/init.d/My_New_Service
```

2. Depois de ter criado o(s) vínculo(s) simbólico(s), teste se seu serviço novo ou modificado funcionará como esperado antes de reinicializar o servidor.

```
# service My_New_Service start
Starting My_New_Service:      [ OK ]
#
# service My_New_Service stop
Stopping My_New_Service:     [ OK ]
```

Depois que tudo estiver no lugar, o serviço novo ou modificado vai iniciar em cada `runlevel` que você selecionou em seu sistema. Além disso, você pode iniciá-lo ou pará-lo manualmente utilizando o comando `service`.

Acrescentando novos serviços ao daemon Upstart init

Você precisa completar apenas um passo para adicionar um novo serviço ou personalizar um serviço já existente com o daemon `Upstart init`. Basta adicionar um novo arquivo de configuração de trabalho ou modificar um existente. Mas esse passo pode ser bastante complicado.

Todos os arquivos de configuração de trabalho do serviço `Upstart` estão localizados no diretório `/etc/init`. Esses são arquivos de texto simples que usam uma sintaxe especial para instruir o daemon `Upstart init` sobre como lidar com um determinado serviço. O seguinte exemplo de um arquivo de configuração tem uma sintaxe muito simples:

```
# cat ck-log-system-restart.conf
# Upstart event
# ck-log-system-restart - write system restart to log
#
start on runlevel 6
task
exec /usr/sbin/ck-log-system-restart
```

Qualquer sinal de jogo da velha (`#`) indica uma linha de comentário e é ignorado por `Upstart`. As outras linhas são chamadas de “estrofes” (*stanzas*), ou sub-rotinas, e têm sintaxe especial para controlar trabalhos `Upstart`. As sub-rotinas do arquivo acima são as seguintes:

- `start on` — Essa sub-rotina define o evento emitido que iniciará o serviço ou tarefa. Nesse caso em particular, quando o evento `runlevel 6` é emitido, o `ck-log-system-restart` inicia.
- `task` — A sub-rotina aqui define que esse trabalho em particular é um trabalho de `task` em

oposição a um de `service`.

- `exec` — Essa sub-rotina define o programa que será executado para iniciar a tarefa. Em vez da sub-rotina `exec`, você pode inserir um script de linha de comando real para executar aqui usando a sub-rotina do script antes do código atual e `end script` depois.

Um arquivo de configuração de trabalho um pouco mais complicado é mostrado a seguir — para o daemon `cron`. Há algumas sub-rotinas adicionais que não estavam no exemplo anterior. Observe que a sub-rotina `task` está faltando no arquivo. Isso indica que esse trabalho específico é um trabalho de `service`, em vez de um trabalho de `task`.

```
# cat cron.conf
# cron - regular background program processing daemon
#
# cron is a standard UNIX program that runs user-specified
# programs at periodic scheduled times
description "regular background program processing daemon"
start on runlevel [2345]
stop on runlevel [!2345]
expect fork
respawn
exec cron
```

As sub-rotinas adicionais nesse exemplo são as seguintes:

- `description` — Essa sub-rotina é opcional e simplesmente descreve o serviço.
- `start on` — Apesar de a parte `start on` dessa sub-rotina ter sido coberta anteriormente, a sintaxe `[2345]` não foi. Usar colchetes significa que a entrada é válida para qualquer desses números. Desse modo, o serviço será iniciado no `runlevel 2, 3, 4 ou 5`.
- `stop on` — A sub-rotina aqui define quais eventos emitidos o serviço vai parar. O `[!2345]` nesta sub-rotina significa *não* `runlevel 2 ou 3 ou 4 ou 5`. Em outras palavras, ele só vai parar `runlevel 0, runlevel 1 ou runlevel 6`.
- `expect` — Essa sub-rotina particular é bastante importante e um pouco complicada. A sintaxe `expect fork` permitirá que `Upstart` monitore esse daemon e qualquer um dos seus processos filhos (bifurcações).
- `respawn` — A sub-rotina aqui diz para `Upstart` reiniciar esse serviço caso venha a ser terminado através de um meio fora do seu `stop on normal`.

como um evento não padrão. Em outras palavras, você pode criar seu próprio nome de evento. Por exemplo, use o nome do evento `MyTest`. Para testar o novo arquivo de configuração, você digita `initctl emit MyTest` na linha de comando. Se o arquivo de configuração funcionar corretamente, modifique a sub-rotina `start on` para o formato `Upstart` correto.

Cada arquivo de configuração de trabalho deve seguir pelo menos três regras. O arquivo de configuração de trabalho deve:

- Não estar vazio
- Ser sintaticamente correto
- Conter pelo menos uma sub-rotina válida

Apesar de existirem apenas três regras, criar ou modificar um arquivo de configuração de trabalho de serviço corretamente pode ser uma tarefa bastante difícil. Veja <http://upstart.ubuntu.com/getting-started.html> ou <http://upstart.ubuntu.com/cookbook> para ajuda sobre a sintaxe necessária para esses arquivos. Além disso, você pode descobrir mais sobre os eventos que o daemon `Ubuntu init` emite digitando `man upstart-events` na linha de comando.

Acrescentando novos serviços a systemd init

Ao adicionar um serviço novo ou personalizado a um servidor `systemd` Linux, você tem de completar três etapas, a fim de ter o serviço gerenciado por `systemd`:

1. **Criar um arquivo de configuração de unidade de serviço novo ou personalizado para o serviço novo ou personalizado.**
2. **Mover o arquivo de configuração de unidade de serviço novo ou personalizado para o local adequado para o gerenciamento por `systemd`.**
3. **Adicionar o serviço ao `Wants` de uma `target unit` específica se quiser que o serviço novo ou personalizado inicie automaticamente com outros serviços.**

Passo 1: Criar um arquivo de configuração de unidade de serviço novo ou personalizado

Se você estiver personalizando um arquivo de configuração de unidade de serviço, basta fazer uma cópia do arquivo de unidade original a partir de `/lib/systemd/system` e adicionar as personalizações desejadas.

Para novos arquivos, obviamente, você estará criando um arquivo de configuração de `service unit` a partir do zero. Considere o seguinte modelo de arquivo básico de `service unit`. No mínimo, você precisa definir as opções `Description` e `ExecStart` para um arquivo de configuração de `service unit`.

```
# cat My_New_Service.service
```

```
[Unit]
```

```
Description=My New Service
```

```
[Service]
```

```
ExecStart=/usr/bin/My_New_Service
```

Para obter ajuda adicional sobre a personalização ou a criação de um novo arquivo de configuração de unidade e as várias opções necessárias, você pode usar as páginas `man`. Na linha de comando, digite `man systemd.service` para saber mais sobre as diversas opções do arquivo de `service unit`.

Passo 2: Mover o arquivo de configuração de unidade de serviço

Antes de mover o arquivo de configuração de unidade de serviço novo ou personalizado, você precisa estar ciente de que há dois possíveis locais para armazenar arquivos de configuração de unidade de serviço. O que você escolher vai determinar se ou não as personalizações têm efeito e se elas permanecem persistentes depois de atualizações de software.

Você pode colocar seu arquivo de configuração de unidade de serviço de sistema em um dos seguintes locais:

- `/etc/systemd/system`
 - Essa localização é usada para armazenar arquivos de configuração de unidade de serviço personalizados locais.
 - Arquivos nessa localização não são substituídos por instalações ou upgrades de software.
 - Arquivos aqui são usados pelo sistema, *mesmo* se houver um arquivo com o mesmo nome no diretório `/lib/systemd/system`.
- `/lib/systemd/system`
 - Essa localização é usada para armazenar arquivos de configuração de unidade de serviço.
 - Arquivos nessa localização são substituídos por instalações e atualizações de software.
 - Arquivos aqui são usados pelo sistema *somente* se *não* houver um arquivo com o mesmo nome no diretório `/etc/systemd/system`.

Assim, o melhor lugar para armazenar seu arquivo de configuração de unidade de serviço novo ou personalizado é em `/etc/systemd/system`.

ca

Quando você cria um serviço novo ou personalizado, para que a alteração tenha efeito sem uma reinicialização do sistema, é necessário emitir um comando especial. Na linha de comando, digite `systemctl daemon-reload`.

Passo 3: Adicionar o serviço ao diretório Wants

Este último passo é opcional. Ele precisa ser feito somente se você quiser que seu novo serviço inicie

com uma `systemd target unit` específica. Para um serviço ser ativado (iniciado) por uma `target unit` específica, ele deve estar no diretório `Wants` da `target unit`.

Primeiro, adicione a linha **`WantedBy= alvo.desejado`** ao final de seu arquivo de configuração de unidade de serviço. O exemplo a seguir mostra que a `target unit` desejada para esse novo serviço é `multi-user.target`.

```
# cat /etc/systemd/system/My_New_Service.service
[Unit]
Description=My New Fake Service

[Service]
ExecStart=/usr/bin/My_New_Service

[Install]
WantedBy=multi-user.target
```

Para adicionar uma nova unidade de serviço a uma unidade alvo, você precisa criar um vínculo simbólico. O exemplo a seguir mostra os arquivos localizados no diretório `Wants` da unidade `multiuser.target`. Anteriormente, na seção “Entendendo `systemd` init”, o comando `systemctl` foi usado para listar `Wants` e esse ainda é o método preferido. Observe que, nesse diretório, os arquivos são vínculos simbólicos que apontam para arquivos de configuração de `service unit` no diretório `/lib/systemd/system`.

```
# ls /etc/systemd/system/multi-user.target.wants
abrt-ccpp.service      cups.path              remote-fs.target
abrttd.service         fcoe.service          rsyslog.service
abrt-oops.service     irqbalance.service    sendmail.service
abrt-vmcore.service   lldpad.service        sm-client.service
atd.service           mcelog.service        sshd-keygen.service
auditd.service        mdmonitor.service     sshd.service
...
#
# ls -l /etc/systemd/system/multi-user.target.wants
total 0
lrwxrwxrwx. 1 root root 37 Nov 2 22:29
abrt-ccpp.service ->
/lib/systemd/system/abrt-ccpp.service
lrwxrwxrwx. 1 root root 33 Nov 2 22:29
abrttd.service ->
/lib/systemd/system/abrttd.service
...
lrwxrwxrwx. 1 root root 32 Apr 26 20:05
sshd.service ->
/lib/systemd/system/sshd.service
```

O exemplo a seguir ilustra o processo de adicionar um arquivo de vínculo simbólico a `My_New_Service`:

```
# ln -s /etc/systemd/system/My_New_Service.service
/etc/systemd/system/multi-user.target.wants/My_New_Service.service
```

Um vínculo simbólico foi criado no diretório `multi-user.target.wants`. Agora, o novo serviço, `My_New_Service`, será ativado (iniciado) quando a unidade `multi-user.target` for ativada.

Para mudar a `systemd target unit` de um serviço, você tem de alterar o vínculo simbólico de modo que ele aponte para a nova localização do diretório alvo `Wants`. Use o comando `ls -sf` para forçar qualquer vínculo simbólico atual a ser quebrado e o novo vínculo simbólico designado ser imposto.

Juntas, as três etapas adicionarão seu novo serviço personalizado a um servidor Linux `systemd`. Lembre-se de que, neste ponto, um novo serviço não será executado até que o servidor seja reinicializado. Para iniciar o novo serviço antes de uma reinicialização, revise os comandos da seção “Parando e iniciando serviços”.

Resumo

A maneira como você inicia e para serviços depende de como o daemon `init` é utilizado pelo seu servidor Linux. Antes de fazer qualquer gestão de serviços, certifique-se de usar os exemplos neste capítulo para ajudar a determinar o daemon `init` do seu servidor Linux.

Os conceitos de iniciar e parar serviços seguem outros conceitos de gerenciamento de serviços, tais como tornar um serviço persistente, iniciar certos serviços na inicialização do servidor, recarregar e reiniciar um serviço. Todos esses conceitos serão muito úteis quando você aprender sobre como configurar e gerenciar um servidor de impressão Linux no próximo capítulo.

Exercícios

Consulte o material neste capítulo para completar as tarefas a seguir. Se você empacar, soluções para as tarefas são mostrados no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa). Tente resolver cada um dos exercícios antes de consultar as respostas. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux).

1. Determine qual daemon `init` seu servidor está usando atualmente.
2. O daemon `init` está usando `sshd` em seu servidor Linux?
3. Determine o `runlevel` anterior e o atual de seu servidor.
4. Como você pode mudar o `runlevel` padrão ou a `target unit` padrão em seu servidor Linux?
5. Para cada daemon `init`, que comando(s) lista(m) os serviços em execução (ou ativos) em seu servidor?
6. Liste os serviços que estão sendo executados (ou estão ativos) em seu servidor Linux.
7. Para cada daemon `init`, que comando(s) mostra(m) o status atual de um serviço

específico?

8. Exiba o status do daemon `cups` em seu servidor Linux.
9. Tente reiniciar o daemon `cups` em seu servidor Linux.
10. Tente recarregar o daemon `cups` em seu servidor Linux.

Configurando um servidor de impressão

NESTE CAPÍTULO

Entendendo impressão em Linux

Configurando impressoras

Usando comandos de impressão

Gerenciando a impressão de documentos

Compartilhando impressoras

Você pode configurar o sistema Linux para usar impressoras conectadas diretamente a ele (por meio de uma porta USB ou paralela) ou que estão disponíveis para impressão em rede. Da mesma maneira, qualquer impressora que você configura em seu sistema local pode ser compartilhada com os usuários em outros sistemas Linux, Windows ou Mac, abrindo a impressora como um servidor de impressão.

Você pode configurar uma impressora como uma impressora Linux nativa no Fedora, RHEL, Ubuntu e outros sistemas Linux com o Common UNIX Printing System (CUPS). Para configurar uma impressora para funcionar como um servidor de impressão no estilo do Microsoft Windows, você pode usar o serviço Samba no Linux.

Este capítulo concentra-se no CUPS. Em particular, ele mostra a interface gráfica para o CUPS, chamada de janela Printer Configuration, que vem com o Fedora, o Red Hat Enterprise Linux e outras distribuições Linux. Usando a janela Printer Configuration, você também pode configurar impressoras como servidores de impressão de modo que as pessoas possam imprimir em sua impressora a partir do computador delas.

Se você não tiver um desktop ou quiser imprimir a partir de um script de shell, este capítulo mostra como usar comandos de impressão. A partir da linha de comando, comandos de impressão como `lpr` estão disponíveis para fazer a impressão. Também há comandos para consultar (`lpq`), manipular (`lpc`) e remover filas de impressão (`lprm`).

Sistema comum de impressão UNIX

O CUPS se tornou o padrão para impressão a partir do Linux e outros sistemas operacionais tipo Unix.

Ele foi projetado para atender às necessidades de hoje no que se refere a definições padronizadas e compartilhamento de impressoras em redes baseadas no Internet Protocol (como a maioria das redes de computadores é hoje). Quase todas as distribuições Linux hoje vêm com o CUPS como seu serviço de impressão. Eis alguns dos recursos do serviço:

- **IPP** — O CUPS é baseado no Internet Printing Protocol (<http://www.pwg.org/ipp>), um padrão que foi criado para simplificar a forma como as impressoras podem ser compartilhadas em redes IP. No modelo IPP, servidores de impressão e clientes que querem imprimir podem trocar informações sobre o modelo e os recursos de uma impressora por meio do protocolo HTTP (isto é, o conteúdo web). Um servidor também pode transmitir a disponibilidade de uma impressora para um cliente de impressão poder facilmente encontrar uma lista de impressoras disponíveis localmente, sem configuração.
- **Drivers** — O CUPS também padronizou a maneira como os drivers de impressora são criados. A ideia era ter um formato comum que poderia ser usado pelos fabricantes de impressoras de modo que um driver pudesse funcionar em todos os diferentes tipos de sistema UNIX. Dessa forma, o fabricante precisaria criar o driver apenas uma vez para funcionar em Linux, Mac OS X e uma variedade de derivados do UNIX.
- **Classes de impressoras** — Você pode usar classes de impressoras para criar entradas de múltiplos servidores de impressão que apontam para a mesma impressora ou uma entrada de servidor de impressão que aponta para múltiplas impressoras. No primeiro caso, cada uma das múltiplas entradas pode permitir diferentes opções (tais como apontar para uma bandeja de papel específica ou imprimir com certos tamanhos de caractere ou margens). No segundo caso, você pode ter um pool de impressoras para que a impressão seja distribuída. Nesse exemplo, uma impressora avariada ou uma impressora que está lidando com documentos muito grandes não vai suspender todos os trabalhos de impressão. O CUPS também suporta **classes implícitas**, que são classes de impressão que se formam pela associação de impressoras de rede idênticas automaticamente.
- **Procura de impressora** — Com a procura de impressora, computadores clientes podem ver todas as impressoras CUPS em sua rede local com a procura ativada. Como resultado, os clientes podem simplesmente selecionar as impressoras que desejam usar a partir dos nomes de impressora transmitidos na rede, sem a necessidade de saber de antemão como as impressoras são nomeadas e onde elas estão conectadas. Você pode desativar o recurso para evitar que outras pessoas na rede local vejam uma impressora.
- **Comandos de impressão UNIX** — Para se integrar ao Linux e outros ambientes UNIX, o CUPS oferece versões de comandos padrão para impressão e gerenciamento de impressoras que têm sido tradicionalmente oferecidas com sistemas UNIX.

Em vez de usar a janela Printer Configuration, há outras maneiras como você também pode configurar impressão CUPS:

- **Configurando o CUPS a partir de um navegador** — O projeto CUPS oferece uma interface baseada na web para adicionar e gerenciar impressoras. Com o serviço `cupsd` em execução, digite `localhost:631` a partir de um navegador no computador que executa o serviço CUPS para

gerenciar impressão. (Consulte a seção “Usando administração do CUPS baseada na web” mais adiante, neste capítulo.)

- **Configurando o CUPS manualmente** — Você também pode configurar o CUPS manualmente (ou seja, editar os arquivos de configuração e iniciar o daemon `cupsd` a partir da linha de comando). Os arquivos de configuração para o CUPS estão contidos no diretório `/etc/cups`. Em particular, você poderia estar interessado no arquivo `cupsd.conf`, que identifica permissões, autenticações e outras informações para o servidor de impressão, e `printers.conf`, que identifica os endereços e as opções para impressoras configuradas. Use o arquivo `classes.conf` para definir classes de impressoras locais.

ando do Windows

ocê também pode imprimir no CUPS a partir de sistemas não UNIX. Por exemplo, você pode ar um driver de impressora PostScript para imprimir diretamente a partir do Windows XP em u servidor CUPS. Você pode usar o CUPS, sem modificação, configurando o computador XP m um driver PostScript que usa `http://printservername:631/printers/targetPrinter` como porta de impressão.

ocê também pode ser capaz de usar os drivers de impressora nativos do Windows para a pressora em vez do driver PostScript. Se o driver nativo do Windows não funcionar como m de fábrica na fila de impressão do CUPS, você pode criar uma Raw Print Queue sob o JPS e utilizá-la no lugar. A Raw Print Queue passará os dados do driver de impressão nativo Windows diretamente para a impressora.

Para usar o CUPS, você precisa ter instalado o pacote de `cups` no Fedora ou no RHEL. A maioria das distribuições Linux desktop incluem o CUPS durante a instalação inicial do sistema. Se ele não for instalado em uma instalação do Fedora ou do RHEL, instale-o digitando o seguinte:

```
# yum install cups
```

Configurando impressoras

Embora utilizar as ferramentas de administração de impressoras construídas especificamente para sua distribuição seja geralmente melhor, muitos sistemas Linux simplesmente confiam nas ferramentas que vêm com o pacote de software do CUPS.

Esta seção explora como usar as ferramentas de administração baseada na web do CUPS que vêm com cada distribuição Linux e, então, examina a ferramenta de configuração de impressora `system-config-printer`, que vem com os sistemas Fedora e o Red Hat Enterprise Linux, para permitir

configurar impressoras. Em alguns casos, nenhuma configuração é necessária, porque as impressoras conectadas podem ser automaticamente detectadas e configuradas.

Adicionando uma impressora automaticamente

Impressoras CUPS podem ser configuradas para transmitir automaticamente sua disponibilidade na rede para um sistema cliente poder detectá-las e usá-las sem configuração. Conecte uma impressora USB ao seu computador e ela pode ser automaticamente detectada e disponibilizada. Na verdade, se você anexar uma impressora local no Fedora e o driver de impressão ainda não estiver instalado, você será solicitado a instalar os pacotes de software necessários para usar a impressora.

Na primeira vez que você tentar imprimir um documento ou ver sua ferramenta de configuração de impressora, as impressoras estarão lá prontas para uso. Configuração adicional pode ser feita usando a ferramenta de administração baseada na web do CUPS ou a janela Printer Configuration.

Usando a administração baseada na web do CUPS

O CUPS oferece sua própria ferramenta administrativa baseada na web para adicionar, excluir e modificar configurações de impressora em seu computador. O serviço de impressão do CUPS (usando o daemon `cupsd`) ouve na porta 631 para fornecer acesso à interface administrativa baseada na web do CUPS e compartilhar impressoras.

Se o CUPS já estiver em execução em seu computador, você pode usar imediatamente a administração baseada na web do CUPS a partir de seu navegador. Para ver se o CUPS está em execução e para começar a configurar as impressoras, abra o navegador web no computador local e digite o seguinte em sua caixa de localização: `http://localhost:631/`.

Um prompt pedindo um nome de login e uma senha válidos pode aparecer quando você solicitar funções que necessitam dele. Se esse for o caso, digite o nome de login e a senha do usuário root e clique em OK. Uma tela semelhante à mostrada na Figura 16.1 aparece.

PS fornece uma ferramenta de administração baseada na web.

Por padrão, a administração baseada na web do CUPS está disponível apenas a partir do host local. Para acessar administração baseada na web do CUPS a partir de outro computador, na página principal do CUPS, selecione a guia Administration, marque a caixa de seleção ao lado de Allow remote administration e selecione o botão Change Settings. Então, a partir de um navegador remoto, você pode acessar a página de administração do CUPS indo para a porta 631 no servidor CUPS (por exemplo, `http://host.example.com:631`).

Quando uma impressora não é detectada automaticamente, você pode adicioná-la a partir da tela de administração, da seguinte maneira:

1. **Clique no botão Add Printer.** A tela Add New Printer aparece.
2. **Selecione o dispositivo ao qual a impressora está conectada.** A impressora pode estar conectada localmente a uma porta paralela, SCSI, serial ou USB diretamente no computador. Alternativamente, você pode selecionar um tipo de conexão de rede para impressoras Apple (AppSocket/HP JetDirect), Internet Printing Protocol (`http` ou `ipp`) ou uma impressora Windows (usando o Samba ou SMB).
3. **Se for solicitado a fornecer mais informações, talvez você precise descrever a conexão com a impressora.** Por exemplo, você pode precisar inserir a taxa de transmissão e a paridade de uma porta serial ou você pode ser solicitado a informar o endereço de rede para uma impressora IPP ou Samba.
4. Digite um nome, localização e descrição para a impressora e selecione se você deseja compartilhá-la. Então, clique em Continue.
5. **Selecione o fabricante do driver de impressão.** Se você não vir o fabricante de sua impressora na lista, escolha PostScript para uma impressora PostScript, ou HP para uma impressora PCL. Para o fabricante que você escolher, você será capaz de selecionar um modelo específico.
6. **Configure opções.** Se for solicitado a configurar opções para sua impressora, você pode fazê-lo selecionando Set Printer Options para continuar.
7. **A impressora deve estar disponível.** Se a impressora for adicionada com sucesso, clique no nome dela para fazer a página da nova impressora aparecer. A partir da página da impressora, você pode selecionar Maintenance ou Administration para imprimir uma página de teste ou modificar a configuração da impressora.

Com a configuração básica da impressora feita, agora você pode trabalhar ainda mais com suas impressoras. Eis alguns exemplos do que você pode fazer:

- **Listar trabalhos de impressão.** Clique em Show All Jobs para ver quais trabalhos de impressão estão atualmente ativos a partir de qualquer das impressoras configuradas para esse servidor. Clique em Show Completed Jobs para ver informações sobre os trabalhos que já estão impressos.

- **Criar uma classe de impressora.** Clique na guia Administration e escolha Add Class e identifique um nome, descrição e localização para uma classe de impressoras. Na lista Printers (Members) configurada em seu servidor, selecione as que vão para essa classe.
- **Cancelar um trabalho de impressão.** Se você imprimir um trabalho de 100 páginas por engano, ou se a impressora estiver imprimindo lixo, o recurso Cancel pode ser muito útil. Na guia Administration, clique em Manage Jobs, então, clique em Show Active Jobs para ver os trabalhos de impressão que estão na fila para a impressora. Selecione o botão Cancel Job ao lado do trabalho de impressão que você deseja cancelar.
- **Visualizar impressoras.** Você pode clicar na guia Printers a partir do topo de qualquer uma das páginas de administração baseadas na web do CUPS para exibir as impressoras que você configurou. Para cada impressora que aparece, você pode selecionar tarefas de manutenção (Maintenance) ou tarefas administrativas (Administrative). Em Maintenance, clique em Pause Printer (para fazer a impressora parar de imprimir, mas ainda aceitar trabalhos de impressão na fila), Reject Jobs (para não aceitar mais nenhum trabalho de impressão naquele momento) ou Print Test Page (para imprimir uma página). A Figura 16.2 mostra as informações na guia Printers.

pode fazer as tarefas de administração na guia Printers.

Usando a janela Printer Configuration

Se estiver usando o Fedora, o RHEL ou outros sistemas baseados no Red Hat, você pode usar a janela Printer Configuration para configurar as impressoras. Na verdade, recomendo que você use-a no lugar da administração web do CUPS, pois os arquivos de configuração de impressora resultantes são adaptados para funcionar com a maneira como o serviço CUPS é iniciado nesses sistemas.

Para instalar uma impressora a partir de seu desktop GNOME, inicie a janela Printer Configuration, selecionando System ⇒ Administration ⇒ Printing, ou como usuário root digitando `system-config-printer`. Essa ferramenta permite adicionar e excluir impressoras e editar as propriedades delas. Ela também permite que você envie páginas de teste para as impressoras para se certificar de que estão funcionando corretamente.

A chave aqui é que você está configurando as impressoras que são gerenciadas por seu daemon de impressão (`cupsd` para o serviço CUPS). Depois que uma impressora está configurada, os usuários em seu sistema local podem usá-la. Você pode consultar a seção “Configurando servidores de impressão” para aprender a tornar o servidor disponível para usuários de outros computadores em sua rede.

As impressoras configuradas podem ser conectadas diretamente ao computador (como em uma porta paralela) ou a outro computador na rede (por exemplo, a partir de outro sistema UNIX ou Windows).

Configurando impressoras locais com a janela Printer Configuration

Adicione uma impressora local (em outras palavras, uma impressora conectada diretamente ao computador) com a janela Printer Configuration utilizando o procedimento que se segue.

Como adicionar uma impressora local

Para adicionar uma impressora local a partir de um desktop GNOME no Fedora 16, siga estes passos:

1. Selecione Applications ⇒ Other ⇒ Printing ou digite o seguinte como usuário root em uma janela Terminal:

```
# system-config-printer &
```

A janela Printing será exibida.

2. Clique em “Add”. (Se solicitado, selecione o botão Adjust Firewall para permitir o acesso à porta de impressora 631.) Uma janela New Printer aparece.
3. Se a impressora que você deseja configurar for detectada, basta selecioná-la e clicar em Forward. Se não for detectada, escolha o dispositivo ao qual a impressora está conectada (LPT #1 e Serial Port #1 são as primeiras portas paralela e serial, respectivamente) e clique em Forward. (Digite `/usr/sbin/lpinfo -v | less` em um shell para ver os tipos de

conexão de impressora.) Você é solicitado a identificar o driver da impressora.

4. Para usar um driver instalado para a impressora, escolha Select Printer From Database e depois escolha o fabricante de sua impressora. Como alternativa, você pode selecionar Provide PPD File e fornecer seu próprio arquivo PPD (por exemplo, se você tiver uma impressora que não é suportada no Linux e tiver um driver que foi fornecido com a impressora). PPD significa PostScript Printer Description. Selecione Forward para ver uma lista de modelos de impressora a partir da qual você pode escolher.

ca

Se a impressora não aparecer na lista, mas suportar PCL (Printer Control Language da HP), tente selecionar uma das impressoras HP (como HP LaserJet). Se a impressora suportar PostScript, selecione impressora PostScript na lista. Selecionar Raw Print Queue permite que você envie documentos que já estão formatados de acordo com um tipo particular de impressora para uma impressora específica.

5. Com seu modelo de impressora selecionado, clique no driver que você deseja usar com essa impressora. Clique em Forward para continuar.
6. Adicione as seguintes informações e clique em Forward:
 - **Printer Name** — Adicione o nome que você quer dar para identificar a impressora. O nome deve começar com uma letra, mas, depois da letra inicial, ele pode conter uma combinação de letras, números, traços (-) e sublinhados (_). Por exemplo, uma impressora HP em um computador chamado maple poderia ser chamado de hp-maple.
 - **Description** — Adicione algumas palavras que descrevem a impressora, tais como seus recursos (por exemplo, uma HP LaserJet 2100M com suporte PCL e PS).
 - **Localização** — Adicione algumas palavras que descrevem a localização da impressora (por exemplo, “Na Sala 205, ao lado da cafeteira”).
7. Quando a impressora é adicionada, você pode ser solicitado a imprimir uma página de teste (clique em No ou Yes). A entrada da nova impressora é exibida na janela Printing, conforme mostrado na Figura 16.3.

ela Printer Properties depois de adicionar uma impressora

8. Se você quiser que a impressora seja a padrão, clique com o botão direito na impressora e selecione Set As Default. À medida que adiciona outras impressoras, você pode alterar a impressora padrão, selecionando aquela que você quer e clicando em Set As Default novamente.
9. A impressão deve estar funcionando nesse momento. Para se certificar, abra uma janela Terminal e use o comando `lpr` para imprimir um arquivo (como `lpr/etc/hosts`). (Se quiser compartilhar a impressora com outros computadores em sua rede, consulte a seção “Configurando servidores de impressão”, mais adiante, neste capítulo.)

Editando uma impressora local

Após clicar duas vezes na impressora que deseja configurar, escolha entre as seguintes opções de menu para alterar sua configuração:

- **Settings** — Os campos Description, Location, Device URI e Make and Model que você criou anteriormente são exibidos nessa caixa de diálogo.
- **Policies** — Clique em Policies para configurar os seguintes itens:
 - **State** — Selecione as caixas de seleção para indicar se a impressora imprimirá trabalhos que estão na fila (Enabled), aceitar novos trabalhos para a impressão (Accepting Jobs) ou estar disponível para ser compartilhada com outros computadores que podem se comunicar com o computador (Shared). Você também deve selecionar Server Settings e clicar na caixa de seleção Share Published printers connected to this system (“Compartilhar impressoras publicadas conectadas a este sistema”) antes que a impressora aceite trabalhos de impressão a partir de outros computadores.
 - **Policies** — Em caso de erro, a seleção stop-printer faz com que toda a impressão nessa impressora pare. Você também pode optar por fazer com que o trabalho seja descartado (abort-job) ou que uma nova tentativa de imprimi-lo seja feita (retry-job) no caso de ocorrer um erro.
 - **Banner** — Por padrão, não há páginas de identificação de início ou fim para a impressora. Escolha páginas de identificação de início ou fim que incluem texto, tal como Classified, Confidential, Secret etc.
- **Access Control** — Se a impressora é compartilhada, você pode selecionar essa janela para criar uma lista de usuários que podem acessar ou parar a impressora (e negar acesso a todos os outros) ou uma lista de usuários que não podem acessar a impressora (e permitir acesso a todos os outros).
- **Printer Options** — Clique em Printer Options para definir padrões para opções relacionadas com o driver da impressora. As opções disponíveis são diferentes para diferentes impressoras. Muitas dessas opções podem ser substituídas quando alguém imprime um documento. Eis

algumas das opções que você pode (ou não pode) ter disponíveis:

- **Watermark** — Várias configurações estão disponíveis para que você possa adicionar e alterar marcas d'água em suas páginas impressas. Por padrão, Watermark e Overlay estão desativados (None). Ao selecionar Watermark (atrás do texto) ou Overlay (por cima do texto), você pode definir as configurações de marca d'água, para determinar como marcas d'água e sobreposições são feitas. Marcas d'água podem ser impressas em cada página (All) ou somente na primeira página (First Only).

Selecione Watermark Text para escolher as palavras que são usadas para a marca d'água ou sobreposição (Draft, Copy, Confidential, Final etc.). Você pode, então, selecionar o tipo de letra, tamanho, estilo e intensidade da marca d'água ou da sobreposição.

- **Resolution Enhancement** — Você pode usar as configurações atuais da impressora ou optar por ativar ou desativar o aprimoramento de resolução.
- **Page Size** — O padrão é o tamanho carta dos EUA (U.S. letter), mas você também pode fazer a impressora imprimir tamanho ofício (legal), envelopes, padrão ISO A4 ou vários outros tamanhos de página.
- **Media Source** — Escolha a bandeja em que imprimir. Selecione Tray 1 para inserir páginas manualmente.
- **Levels of Gray** — Escolha utilizar os níveis de cinza atuais da impressora ou ativar níveis de cinza melhorados ou padrões.
- **Resolution** — Selecione a resolução de impressão padrão (como 300, 600 ou 1.200 pontos por polegada). Resoluções mais altas resultam em melhor qualidade, mas demoram mais para imprimir.
- **EconoMode** — Use a configuração atual da impressora, ou escolha um modo em que você economiza tóner, ou um em que você tem a melhor qualidade possível.
- **Job Options** — Clique em Job Options para configurar as opções padrão comuns que serão utilizadas para essa impressora se o aplicativo que está imprimindo o trabalho já não as configurou. Essas incluem opções comuns (número de cópias, orientação, redimensionar para caber e páginas por lado), Image Options (escala, saturação, matiz e gama) e Text Options (caracteres/polegada, linhas/polegada e configurações de margem) .
- **Ink/Toner Levels** — Clique em Ink/Toner Levels para ver informações sobre a quantidade de tinta ou tóner que resta na impressora. (Nem todas as impressoras informam esses valores.)

Clique em Apply quando estiver satisfeito com as alterações feitas na impressora local.

Configurando impressoras remotas

Para usar uma impressora que está disponível em sua rede, você deve identificá-la para seu sistema Linux. Conexões de impressora remota suportadas incluem impressoras Networked CUPS (IPP), Networked UNIX (LPD), Networked Windows (Samba) e JetDirect. (Obviamente, os servidores de impressão CUPS e UNIX podem ser executados a partir de sistemas Linux, bem como a partir de outros sistemas UNIX.)

Em cada caso, você precisa de uma conexão de rede de seu sistema Linux com os servidores para que as impressoras sejam conectadas. Usar uma impressora remota exige que alguém configure essa impressora no computador servidor remoto. Consulte a seção “Configurando servidores de impressão”, mais adiante, neste capítulo, para obter informações sobre como fazer isso em seu servidor Linux.

Use a janela Printing para configurar cada um dos tipos de impressora remota. Esta é a maneira como isso é feito no Fedora 16:

1. A partir do painel Desktop, selecione Applications ⇒ Other ⇒ Printing.
2. Clique em “Add”. A janela New Printer aparece.
3. Dependendo do tipo de porta que você tem em seu computador, selecione uma das seguintes opções:
 - **LPT #1** — Para uma impressora conectada à porta paralela.
 - **Serial Port # 1** — Para uma impressora conectada à porta serial.
 - **AppleSocket/HP JetDirect** — Para uma impressora JetDirect.
 - **Internet Printing Protocol (IPP)** — Para uma impressora CUPS ou outra impressora IPP. A maioria das impressoras Linux e Mac OS X se enquadram nessa categoria.
 - **Internet Printing Protocol (HTTPS)** — Para uma impressora CUPS ou outra impressora IPP sendo compartilhada através de uma conexão segura (certificados válidos necessários).
 - **LPD/LPR Host ou Impressora** — Para uma impressora UNIX.
 - **Windows Printer via SAMBA** — Para uma impressora do sistema Windows.

Continue com essas etapas em qualquer das próximas seções nas quais que forem apropriadas.

Adicionando uma impressora CUPS remota

Se você optar por adicionar uma impressora de CUPS (IPP) que é acessível através de sua rede local a partir da janela Printer Configuration, você deve adicionar a seguinte informação à janela que aparece:

- **Host** — O hostname do computador ao qual a impressora está conectada (ou de outra forma acessível). Isso pode ser um endereço IP ou um hostname TCP/IP do computador. O nome TCP/IP é acessível a partir de seu arquivo `/etc/hosts` ou por meio de um servidor de nomes DNS.
- **Queue** — Nome da impressora no servidor de impressão remota CUPS. O CUPS suporta instâncias da impressora, o que permite que cada impressora tenha vários conjuntos de opções. Se a impressora CUPS remota for configurada dessa maneira, você é capaz de escolher um caminho específico para uma impressora, como `hp/300dpi` ou `hp/1200dpi`. Uma barra separa o nome da fila de impressão, de um lado, e a instância da impressora, de outro.

Complete o resto do processo como você faria para uma impressora local (consulte a seção “Adicionando uma impressora local”, anteriormente, neste capítulo).

Adicionando uma impressora remota UNIX (LDP/LPR)

Se optar por adicionar uma impressora UNIX (LPD/LPR) a partir da janela Printer Configuration, você deve adicionar a seguinte informação à janela que aparece:

- **Host** — O hostname do computador ao qual a impressora está conectada (ou de outra forma acessível). Esse é o endereço IP ou o nome do computador (o nome da máquina é acessível a partir de seu arquivo `/etc/hosts` ou por meio de um servidor DNS). Selecione o botão Probe para procurar o host.
- **Queue** — Nome da impressora no computador UNIX remoto.

Complete o resto do processo como você faria para uma impressora local (consulte a seção “Adicionando uma impressora local”, anteriormente, neste capítulo).

ca

trabalho de impressão que você enviar para testar a impressora for rejeitado, o servidor de impressão pode não permitir o acesso à impressora. Peça para o administrador do computador remoto seu hostname ao arquivo `/etc/lpd.perms`. (Digite `lpq -P impressora` para ver o status do trabalho de impressão.)

Adicionando uma impressora ao Windows (SMB)

Permitir que seu computador acesse uma impressora SMB (o serviço de impressão do Windows) envolve a adição de uma entrada para a impressora na janela Select Connection.

Quando você escolher adicionar uma impressora Windows à janela Printer Configuration (Windows Printer via Samba), selecione Browse para ver uma lista de computadores em sua rede que foram detectados como oferecendo serviços SMB (serviço de arquivo e/ou impressão). Você pode configurar a impressora a partir dessa janela da seguinte forma:

1. Digite o URI da impressora excluindo o `smb://` inicial. Por exemplo, `/host1/myprinter` ou `/mygroup/host1/myprinter`.
2. Selecione Prompt user if authentication is required (Exibir prompt para o usuário se for necessária autenticação) ou Set authentication details now (Definir detalhes de autenticação agora).
3. Se você escolheu a segunda opção, preencha o nome de usuário e senha necessários para acessar a impressora SMB. Clique em Verify para verificar se você pode se autenticar no servidor.
4. Clique em Forward para continuar.

Alternativamente, você pode identificar um servidor que não aparece na lista de servidores. Digite as informações necessárias para criar um URI SMB que contém as seguintes informações:

- **Workgroup** — O nome do grupo de trabalho designado para o servidor SMB. Usar o nome do grupo de trabalho não é necessário em todos os casos.
- **Server** — O nome NetBIOS ou o endereço IP do computador, o qual pode ou não ser o mesmo que seu nome TCP/IP. Para traduzir esse nome para o endereço necessário para chegar ao host SMB, o Samba verifica vários lugares nos quais o nome pode ser atribuído a um endereço IP. O Samba

verifica o seguinte (na ordem mostrada) até que localize uma correspondência: o arquivo `/etc/hosts` local, o arquivo `/etc/lmhosts` local, um servidor WINS na rede e respostas às transmissões em cada interface de rede local para resolver o nome.

- **Share** — Nome sob o qual a impressora é compartilhada com o computador remoto. Ele pode ser diferente do nome pelo qual os usuários locais da impressora SMB conhecem a impressora.
- **User** — O nome de usuário é exigido pelo sistema do servidor SMB para dar-lhe acesso à impressora SMB. Um nome de usuário não é necessário se você estiver autenticando a impressora com base no nível de compartilhamento em vez de usar controle de acesso no nível de usuário. Com o nível acesso de compartilhamento, você pode adicionar uma senha para cada impressora compartilhada ou sistema de arquivos.
- **Password** — A senha associada ao nome de usuário SMB ou ao recurso compartilhado, dependendo do tipo de controle de acesso a ser utilizado.

Atenção

Quando você insere um usuário e uma senha para o SMB, essa informação é armazenada criptografada no arquivo `/etc/cups/printers.conf`. Certifique-se de que o arquivo permaneça legível apenas por root.

Eis um exemplo do URI SMB que você poderia adicionar à caixa SMB://box:

```
jjones:my9passwd@FSTREET/NS1/hp
```

O URI mostrado aqui identifica o nome de usuário (`jjones`), a senha do usuário (`my9passwd`), o grupo de trabalho (`FSTREET`), o servidor (`NS1`) e o nome da fila da impressora (`hp`).

Complete o resto do processo como você faria para uma impressora local (consulte a seção “Adicionando uma impressora local”, anteriormente, neste capítulo).

Se tudo estiver configurado corretamente, você pode usar o comando `lpr` padrão para imprimir o arquivo na impressora. Usando esse exemplo, empregue a seguinte sintaxe para imprimir:

```
$ cat file1.ps | lpr -P NS1-PS
```

Ca

Quando você estiver recebendo mensagens de falha, certifique-se de que o computador no qual você está imprimindo está disponível. Para o exemplo Printer NS1 hp, você pode digitar `smbclient -L NS1 -U jjones`. Então, digite sua senha (`my9passwd`, neste caso). A opção `-L` pede informações sobre o servidor, a opção `-U jjones` diz para fazer login como o usuário `jjones`. Se receber uma resposta de consulta de nome positiva depois de digitar uma senha, você deve ver uma lista de impressoras e arquivos compartilhados do servidor. Confira os nomes e tente imprimir de novo.

Trabalhando com impressão CUPS

Ferramentas como a administração baseada na web do CUPS e a janela Printer Configuration efetivamente escondem a instalação do CUPS subjacente. Podem haver momentos, porém, em que você quer trabalhar diretamente com as ferramentas e arquivos de configuração que vêm com o CUPS. As próximas seções descrevem como usar alguns recursos especiais do CUPS.

Configurando o servidor CUPS (cupsd.conf)

O processo daemon `cupsd` ouve solicitações para o servidor de impressão CUPS e responde a esses pedidos com base nas configurações no arquivo `/etc/cups/cupsd.conf`. As variáveis de configuração no arquivo `cupsd.conf` são na mesma forma que aquelas no arquivo de configuração do Apache (`httpd.conf` ou `apache2.conf`). Digite **man cupsd.conf** para ver detalhes sobre qualquer uma das configurações.

A janela Printer Configuration acrescenta informações de acesso ao arquivo `cupsd.conf`. Para outros sistemas Linux ou se você não tem um desktop em seu servidor, você pode precisar configurar o arquivo `cupsd.conf` manualmente. É possível percorrer o arquivo `cupsd.conf` para ajustar detalhes do servidor CUPS. A maioria das configurações é opcional ou pode simplesmente ser deixada como padrão. Vamos dar uma olhada em algumas das configurações no arquivo `cupsd.conf`.

Nenhuma classificação é configurada por padrão. Com a classificação configurada para `topsecret`, você pode ter Top Secret exibido em todas as páginas que passam pelo servidor de impressão:

```
Classification topsecret
```

Outras classificações que podem substituir `topsecret` incluem `classified`, `confidential`, `secret` e `unclassified`.

As linhas `ServerCertificate` e `ServerKey` (“comentada” por padrão) podem ser configuradas para indicar onde o certificado e a chave são armazenados, respectivamente:

```
ServerCertificate /etc/cups/ssl/server.crt
ServerKey /etc/cups/ssl/server.key
```

Ative essas duas linhas se quiser fazer conexões criptografadas. Então, adicione seu certificado e sua chave aos arquivos anotados. O uso de certificados permite que você compartilhe sua impressora como uma impressora HTTPS IPP.

O termo *browsing* (procurar, navegar) se refere ao ato de transmitir informações sobre a impressora em sua rede local e ouvir informações de outros servidores de impressão. O `browsing` está ativado por padrão somente para o host local (`@LOCAL`). Você pode permitir informações de `browsing` do CUPS (`BrowseAllow`) para endereços adicionais selecionados. Informações de `browsing` são transmitidas, por padrão, no endereço `255.255.255.255`. Esses padrões aparecem no arquivo `cupsd.conf` assim:

```
Browsing On
BrowseProtocols cups
BrowseOrder Deny,Allow
BrowseAllow from @LOCAL
BrowseAddress 255.255.255.255
Listen *:631
```

Para ativar a administração baseada na web do CUPS e compartilhar impressoras com outras pessoas na rede, o daemon `cupsd` pode ser configurado para ouvir na porta 631 para todas as placas de rede para seu computador com base nesta entrada: `Listen *:631`. Por padrão, ele ouve na placa local somente (`Listen localhost:631`).

Ao ativar `BrowseRelay` (está desativado por padrão), você pode permitir que o CUPS procure informação a ser passada entre duas ou mais redes. O `sourceaddress` e o `destination-address` podem ser endereços IP individuais ou podem representar números de rede:

```
BrowseRelay source-address destination-address
```

Essa é uma boa maneira de permitir que usuários em várias redes locais conectadas descubram e usem impressoras em outras redes locais próximas.

Você pode permitir ou negar acesso a diferentes recursos do servidor CUPS. Uma definição de acesso para uma impressora CUPS (criada a partir da janela `Printer Configuration`) pode aparecer assim:

```
<Location /printers/ns1-hp1>
Order Deny,Allow
Deny From All
Allow From 127.0.0.1
AuthType None
</Location>
```

Aqui, a impressão na impressora `ns1-hp1` é permitida somente para usuários na máquina local (`127.0.0.1`). Nenhuma senha é necessária (`AuthType None`). Para permitir acesso à ferramenta de administração, o CUPS deve ser configurado para solicitar uma senha (`AuthType Basic`).

Iniciando o servidor CUPS

Para sistemas Linux que usam scripts de inicialização no estilo SystemV (tal como muitas versões do Fedora e do RHEL), iniciar e desligar o serviço de impressão CUPS é muito fácil. Use o comando `chkconfig` para ativar o CUPS de modo que ele inicie em cada reinicialização. Execute o script de inicialização `cups` para fazer o serviço CUPS iniciar imediatamente. No RHEL 6.x ou versão anterior, digite o seguinte como o usuário `root`:

```
# chkconfig cups on
# service cups start
```

Se o serviço CUPS já estiver executando, você deve usar `restart` em vez de `start`. Usar a opção `restart` também é uma boa maneira de reler todas as opções de configuração que você pode ter mudado no arquivo `cupsd.conf` (embora, se o CUPS já estiver em execução, `service cups reload` releia os arquivos de configuração sem reiniciar).

No Fedora, você usa o comando `systemctl` em vez de `service` para iniciar e parar serviços:

```
# systemctl status cups.service
```

```
cups.service - CUPS Printing Service
```

```
Loaded: loaded (/lib/systemd/system/cups.service; disabled)
```

```
Active: active (running) since Wed, 15 Feb 2012 23:04:01 -0500;
1min
```

```
54s ago
```

```
Main PID: 6492 (cupsd)
```

```
CGroup: name=systemd:/system/cups.service~CA
```

```
6492 /usr/sbin/cupsd -f
```

Você pode determinar se o serviço CUPS está funcionando porque o status apresenta o daemon `cupsd` ativo com PID 6492. Se esse serviço não estivesse funcionando, você poderia iniciar o serviço CUPS da seguinte maneira:

```
# systemctl start cups.service
```

Veja o Capítulo 15, “Iniciando e parando serviços”, para obter mais informações sobre os comandos `systemctl` e `service` a fim de trabalhar com serviços.

Configurando opções de impressora CUPS manualmente

Se sua distribuição Linux não tiver uma maneira gráfica de configurar o CUPS, você pode editar os arquivos de configuração diretamente. Por exemplo, quando uma nova impressora é criada a partir da janela `Printer Configuration`, ela é configurada no arquivo `/etc/cups/printers.conf`. Eis um exemplo de uma entrada de impressora:

```
<DefaultPrinter printer>
```

```
Info HP LaserJet 2100M
```

```
Location HP LaserJet 2100M in hall closet
```

```
DeviceURI parallel:/dev/lp0
```

```
State Idle
```

```
Accepting Yes
```

```
Shared No
```

```
JobSheets none none
```

```
QuotaPeriod 0
```

```
PageLimit 0
```

```
KLimit 0
```


</Printer>

Esse é um exemplo de uma impressora local que serve como a impressora padrão para o sistema local. O valor `Shared` não está configurado porque a impressora está atualmente disponível somente no sistema local. A informação mais interessante relaciona-se com `DeviceURI`, o qual mostra que a impressora está conectada à porta paralela `/dev/lp0`. O estado é `Idle` (pronto para aceitar trabalhos de impressão) e o valor de `Accepting` é `Yes` (a impressora aceita trabalhos de impressão por padrão).

`DeviceURI` tem várias maneiras de identificar o nome do dispositivo de uma impressora, refletindo onde a impressora está conectada. Eis alguns exemplos listados no arquivo `printers.conf`:

```
DeviceURI parallel:/dev/lp0
DeviceURI serial:/dev/ttyd1?baud=38400+size=8+parity=none+flow=soft
DeviceURI scsi:/dev/scsi/scld610
DeviceURI socket://hostname:port
DeviceURI tftp://hostname/path
DeviceURI ftp://hostname/path
DeviceURI http://hostname[:port]/path
DeviceURI ipp://hostname/path
DeviceURI smb://hostname/printer
```

Os três primeiros exemplos mostram a forma para impressoras locais (paralela, serial e SCSI). Os outros exemplos são para servidores remotos. Em cada caso, *hostname* pode ser o nome do host ou endereço IP. Números de porta ou caminhos identificam os locais de cada impressora no host.

ca

Se você não é capaz de imprimir porque um particular driver de impressora não é suportado no CUPS, você pode configurar a impressora para aceitar trabalhos no modo bruto. Isso pode funcionar bem se você estiver imprimindo a partir de clientes Windows que têm os drivers de impressão corretos instalados. Para ativar a impressão em modo bruto no CUPS, tire o caractere inicial de comentário da seguinte linha no arquivo `/etc/cups/mime.types`, no Linux:

```
application/octet-stream
```

Comente a seguinte linha no arquivo `/etc/cups/mime.convs`:

```
application/octet-stream application/vnd.cups-raw 0 -
```

Depois disso, você pode imprimir arquivos como dados brutos em suas impressoras sem usar a opção `-oraw` para o comando `print`.

Usando comandos de impressão

Para permanecer compatível com os antigos recursos de impressão UNIX e Linux, o CUPS suporta muitos dos comandos antigos para trabalhar com impressão. A maioria das tarefas de impressão com o CUPS pode ser realizada com o comando `lpr`. Aplicações de processamento de texto, como o StarOffice, OpenOffice e AbiWord, são configuradas para usar esse recurso para impressão.

Você pode usar a janela Printer Configuration para configurar os filtros necessários para cada impressora de modo que o texto possa ser formatado corretamente. Opções para o comando `lpr` podem adicionar filtros para processar adequadamente o texto. Outros comandos para o gerenciamento de documentos impressos incluem `lpq` (para visualizar o conteúdo de filas de impressão), `lprm` (para remover trabalhos de impressão da fila) e `lpc` (para controlar impressoras).

Imprimindo com lpr

Você pode usar o comando `lpr` para imprimir documentos em impressoras locais e remotas (desde que as impressoras sejam configuradas localmente). Arquivos de documentos podem ser adicionados ao fim da linha do comando `lpr` ou direcionados para o comando `lpr` usando uma barra vertical (`|`). Eis um exemplo de um comando `lpr` simples:

```
$ lpr doc1.ps
```

Quando você especifica somente um arquivo de documento com `lpr`, a saída é direcionada para a impressora padrão. Como um usuário individual, você pode alterar a impressora padrão, definindo o valor da variável `PRINTER`. Normalmente, você adiciona a variável `PRINTER` a um dos seus arquivos de inicialização, como `$HOME/.bashrc`. Adicionando a seguinte linha ao seu arquivo `bashrc`, por exemplo, configura sua impressora padrão como `lp3`:

```
export PRINTER=lp3
```

Para substituir a impressora padrão, especifique uma impressora em particular na linha de comando `lpr`. O exemplo a seguir usa a opção `-P` de selecionar uma impressora diferente:

```
$ lpr -P canyonps doc1.ps
```

O comando `lpr` tem uma variedade de opções que permitem que ele interprete e formate diversos tipos de documentos. Esses incluem `-# num`, em que `num` é substituído pelo número de cópias (de 1 a 100) e `-l` (o que faz com que um documento seja enviado no modo bruto, presumindo-se que o documento já foi formatado). Para saber mais opções para `lpr`, digite `man lpr`.

Listando o status com lpc

Use o comando `lpc` para listar o status de suas impressoras. Eis um exemplo:

```
$ /usr/sbin/lpc status
printer is on device 'parallel' speed -1
```

```
hp:                queuing is enabled
                   printing is disabled
                   no entries
                   daemon present

deskjet_5550:      printer is on device '/dev/null' speed -1
                   queuing is enabled
                   printing is disabled
                   no entries
                   daemon present
```

Essa saída mostra duas impressoras ativas. A primeira (hp) está conectada à porta paralela. A segunda (deskjet_5550) é uma impressora de rede (mostrada como /dev/null). A impressora hp está desativada (offline), embora a fila esteja habilitada e as pessoas possam continuar a enviar trabalhos para a impressora.

Removendo trabalhos de impressão com lprm

Os usuários podem retirar seus próprios trabalhos de impressão da fila com o comando `lprm`. Usado sozinho na linha de comando, `lprm` remove da impressora padrão todos os trabalhos do usuário. Para remover trabalhos de uma impressora específica, utilize a opção `-P`, como segue:

```
$ lprm -P lp0
```

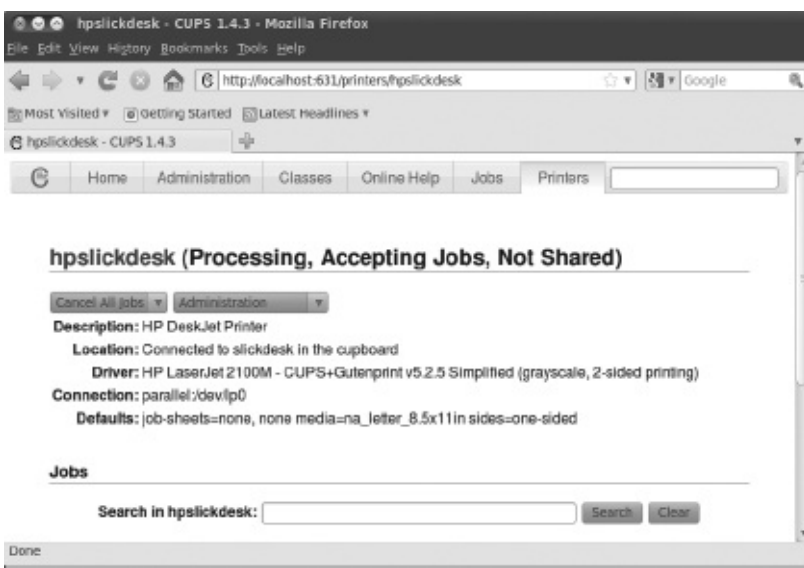
Para remover todos os trabalhos de impressão do usuário atual, digite o seguinte:

```
$ lprm -
```

O usuário root pode remover todos os trabalhos de impressão de um usuário específico indicando esse usuário na linha de comando `lprm`. Por exemplo, para remover todos os trabalhos de impressão do usuário mike, o usuário root digita o seguinte:

```
$ lprm -U mike
```

Para remover uma tarefa de impressão individual da fila, indique seu número de trabalho na linha de comando `lprm`. Para encontrar o número da tarefa, digite o comando `lpq`. Eis como pode ser uma saída desse comando:



A saída mostra dois trabalhos imprimíveis esperando na fila. (A impressora está pronta e imprimindo o trabalho listado como ativo.) Na coluna Job, você pode ver o número do trabalho associado a cada documento. Para remover o primeiro trabalho de impressão, digite o seguinte:

```
# lprm 133
```

Configurando servidores de impressão

Você configurou uma impressora para que você e os outros usuários em seu computador possam imprimir nela. Agora você quer compartilhar essa impressora com outras pessoas em sua casa, escola ou escritório. Basicamente, isso significa configurar a impressora como um servidor de impressão.

As impressoras configuradas em um sistema Linux podem ser compartilhadas de diferentes maneiras com outros computadores na rede. Não só seu computador pode agir como um servidor de impressão Linux (configurando CUPS), como também pode aparecer como um servidor de impressão SMB (Windows) para computadores clientes. Depois que uma impressora local está conectada ao seu sistema Linux e seu computador está conectado à sua rede local, você pode usar os procedimentos desta seção para compartilhar a impressora com computadores clientes usando uma interface Linux (UNIX) ou SMB.

Configurando uma impressora CUPS compartilhada

Disponibilizar a impressora local adicionada ao seu computador Linux para outros computadores em sua rede é bastante fácil. Se existe uma conexão de rede TCP/IP entre os computadores que estão compartilhando a impressora, você simplesmente concede permissão a todos os hosts, máquinas individuais ou usuários de hosts remotos para acessar o serviço de impressão de seu computador.

Para configurar manualmente uma entrada de impressora no arquivo `/etc/cups/printers.conf` a aceitar trabalhos de impressão de todos os outros computadores, certifique-se de que a linha `Shared Yes` está configurada. O exemplo a seguir de uma entrada de `printers.conf` mostrada anteriormente neste capítulo demonstra como a nova entrada ficaria:

```
<DefaultPrinter printer>
```

```
Info HP LaserJet 2100M
Location HP LaserJet 2100M in hall closet
DeviceURI parallel:/dev/lp0
State Idle
Accepting Yes
Shared Yes
JobSheets none none
QuotaPeriod 0
PageLimit 0
KLimit 0
</Printer>
```

Em sistemas Linux que usam a janela Printer Configuration descrita no início deste capítulo, é melhor configurar a impressora compartilhada usando a janela. Veja como, usando o Fedora 16:

1. No painel de Desktop no GNOME no Fedora, selecione Applications ⇒ Other ⇒ Printing. A janela Printer Configuration aparece.
2. Para permitir que todas as suas impressoras sejam compartilhadas, selecione Server ⇒ Settings. Se você não for o usuário root, será solicitado a inserir a senha de root. O pop-up Basic Server Settings aparece.
3. Selecione a caixa de seleção ao lado de Publish shared printers connected to this system (“Publicar impressoras compartilhadas conectadas a esse sistema”) e clique em OK. Você pode ser solicitado a modificar seu firewall para abrir as portas necessárias para sistemas remotos acessarem suas impressoras.
4. Para continuar a permitir ou restringir a impressão em uma impressora específica, clique duas vezes no nome da impressora que você deseja compartilhar. (Se a ela ainda não estiver configurada, consulte a seção “Configurando impressoras”, anteriormente, neste capítulo.)
5. Escolha o título Policies e selecione Shared.
6. Se você quiser restringir o acesso à impressora para usuários selecionados, selecione o título Access Control e escolha uma das seguintes opções:
 - **Allow Printing for Everyone Except These Users** — Com essa opção selecionada, todos os usuários têm permissão de acesso à impressora. Ao digitar nomes de usuário na caixa Users e clicar em Add, você exclui os usuários selecionados.
 - **Deny Printing for Everyone Except These Users** — Com isso selecionado, todos os usuários são excluídos do uso da impressora. Digite nomes de usuário na caixa de Users e clique em Add para permitir o acesso à impressora para somente os nomes inseridos.

Agora você pode configurar outros computadores para usar a impressora, como descrito na seção “Configurando impressoras” deste capítulo. Se você tentar imprimir a partir de outro computador e ele não funcionar, aqui estão algumas dicas de solução de problemas:

- **Abra seu firewall.** Se você tiver um firewall restritivo, ele pode não permitir a impressão. Você deve habilitar o acesso à porta 631 para permitir o acesso à impressão em seu computador. A

janela Printing pode solicitar que você em algum momento abra essa porta. (Verifique o arquivo `/etc/sysconfig/iptables` para ver se há uma regra de firewall configurada que aceita a impressão de TCP e UDP 631.)

- **Ative o estilo LPD de impressão.** Certos aplicativos podem exigir um serviço de impressão no estilo LPD mais antigo para imprimir em sua impressora compartilhada. Para ativar estilo LPD de impressão em seu servidor CUPS, você deve ativar o serviço `cups-lpd`. A maioria das distribuições Linux que incluem CUPS também devem ter `cups-lpd` disponível. No Fedora e sistemas RHEL, digite **yum install cups-lpd** como usuário root para instalá-lo. Então, ative o serviço `cups-lpd`:

```
# chkconfig cups-lpd on
```

- **Confira os nomes e endereços.** Certifique-se de que você digitou o nome do computador e a fila de impressão corretamente quando o configurou em outro computador. Tente usar o endereço IP em vez do hostname. (Se isso funcionar, é sinal de que há um problema de conversão de nome DNS.) Executar uma ferramenta como o `etherreal` permite ver onde a transação falha.
- **Verifique qual endereço cupsd está ouvindo.** O daemon `cupsd` deve estar ouvindo fora do localhost para sistemas remotos que querem imprimir nele. Para verificar isso, use o comando `netstat` (como usuário root) da seguinte maneira. O primeiro exemplo mostra `cupsd` só ouvindo no host local (127.0.0.1:631); o segundo mostra `cupsd` ouvindo em todas as placas de rede (0 0.0.0.0:631):

```
# netstat -tupln | grep 631
tcp        0      0 127.0.0.1:631      0.0.0.0:*          LISTEN    6492/cupsd
# netstat -tupln | grep 631
tcp        0      0 0.0.0.0:631       0.0.0.0:*          LISTEN    6492/cupsd
```

Mudanças de acesso à sua impressora compartilhada são feitas nos arquivos `cupsd.conf` e `printers.conf` em seu diretório `/etc/cups`.

Configurando uma impressora compartilhada na rede Samba

Suas impressoras Linux podem ser configuradas como impressoras SMB compartilhadas a fim de parecerem estar disponíveis a partir de sistemas Windows. Para compartilhar sua impressora como se fosse uma impressora Samba (SMB), basta definir as configurações básicas do servidor Samba, como descrito no Capítulo 19, “Configurando um servidor de compartilhamento de arquivos Windows (Samba)”. Todas as suas impressoras devem estar compartilhadas em sua rede local por padrão. A próxima seção mostra como são as configurações resultantes e como você pode querer mudá-las.

Entendendo `smb.conf` para impressão

Quando você configura o Samba, o arquivo `/etc/samba/smb.conf` é criado para permitir que todas as suas impressoras configuradas sejam compartilhadas. Eis algumas linhas do arquivo `smb.conf` que se relacionam com o compartilhamento de impressoras:

```
[global]
```

```
...
```

```

load printers = yes
cups options = raw

; printcap name = /etc/printcap
; printing = cups
...

[printers]
    comment = All Printers
    path = /var/spool/samba
    browseable = yes
    writeable = no
    printable = yes

```

Essas definições de exemplo são o resultado de configurar o Samba a partir do Samba Server Configuration (`system-config-samba`), no Fedora. Você pode ler as linhas de comentário e saber mais sobre o conteúdo do arquivo. Linhas que começam com um ponto e vírgula (;) indicam a configuração padrão para a opção em uma linha de comentário. Remova o ponto e vírgula para mudar a configuração.

As linhas selecionadas mostram que as impressoras a partir de `/etc/printcap` foram carregadas e que o serviço CUPS está sendo usado. Com `cups options` configurado como `raw`, o Samba assume que os arquivos de impressão já foram formatados no momento em que eles alcançam seu servidor de impressão. Isso permite que os clientes Linux ou Windows forneçam seus próprios drivers de impressão. As últimas linhas são a definição real da impressora. Ao alterar a opção de `browseable` de `no` para `yes`, os usuários podem imprimir em todas as impressoras (`printable = yes`).

Também é possível armazenar os drivers de impressão nativos do Windows em seu servidor Samba. Quando um cliente Windows utiliza a impressora, o driver automaticamente se torna disponível. Você não precisa usar um driver de CD ou baixar um driver do site do fornecedor. Para ativar o compartilhamento de driver de impressora, adicione um compartilhamento Samba chamado `print$` que se parece com o seguinte:

```

[print$]
comment = Printer Drivers
path = /var/lib/samba/drivers
browseable = yes
guest ok = no
read only = yes
write list = chris, dduffey

```

Depois que o compartilhamento está disponível, você pode começar a copiar os drivers de impressão do Windows para o compartilhamento, como descrito no Samba HOWTO:

<http://www.samba.org/samba/docs/man/Samba-HOWTO-Collection/classicalprinting.html#id2626941>

Configurando clientes SMB

Há boas chances de que, se você estiver configurando uma impressora Samba em seu computador Linux, irá querer compartilhá-la com os clientes Windows também. Se o Samba estiver corretamente configurado em seu computador e os computadores clientes podem alcançar você através da rede, os usuários não devem ter problemas para encontrar e utilizar sua impressora.

Para o Windows 7, clique em Iniciar ⇒ Dispositivos e Impressoras e selecione a impressora na lista que aparece para configurá-la. Em alguns sistemas Windows mais antigos, procure sua impressora compartilhada Samba em Ambiente de Rede (ou Meus Locais de Rede). Na área de trabalho do Windows 9x, dê um clique duplo no ícone Network Neighborhood. (A partir do Windows 2000 ou XP, clique duas vezes no ícone Meus Locais de Rede.)

Com o Windows Vista, abra o ícone Rede. O nome de seu computador host (o nome NetBIOS, o qual provavelmente também é o nome TCP/IP) é exibido na tela ou dentro de uma pasta de grupo de trabalho na tela. Abra o ícone que representa seu computador. A janela que se abre mostra suas impressoras e pastas compartilhadas.

ca

Se o ícone de seu computador não aparecer no Ambiente de Rede ou Meus Locais de Rede, tente usar a janela de pesquisa. A partir do Windows XP, selecione Iniciar ⇒ Procurar ⇒ Computador ou Pessoas ⇒ Computador. Escreva o nome de seu computador na caixa Nome do Computador e clique em Pesquisar. Clique duas vezes em seu computador no painel de resultados da janela Pesquisar. Uma janela que exibe as impressoras e as pastas compartilhadas do seu computador aparece.

Depois de sua impressora compartilhada aparecer na janela, configure um ponteiro para ela abrindo (dando um clique duplo) o ícone da impressora. A mensagem diz que você deve configurar a impressora para poder usá-la. Clique em Yes para prosseguir e configurar a impressora para utilização local. O Add Printer Wizard aparece. Responda às perguntas sobre como você pretende usar a impressora e adicione os drivers apropriados. Quando você terminar, a impressora aparecerá na janela da impressora.

Outra forma de configurar uma impressora SMB a partir de um sistema operacional Windows XP é ir para Iniciar ⇒ Impressoras e Faxes. Na janela Impressoras e Faxes que aparece, clique no ícone Adicionar Impressora na parte superior esquerda da janela e selecione Impressora de Rede na primeira janela. A partir daí você pode navegar e/ou configurar a impressora SMB.

Resumo

Prestação de serviços de impressão em rede é essencial na rede de negócios de hoje. Com o uso de alguns dispositivos conectados à rede, você pode concentrar sua impressão investindo em alguns poucos dispositivos de alta qualidade que vários usuários podem compartilhar em vez de numerosos dispositivos de menor custo. Além disso, uma impressora centralmente localizada pode facilitar a manutenção, enquanto ainda permite que todos os trabalhos de impressão sejam concluídos.

O serviço de impressão padrão em quase todas as principais distribuições Linux hoje é o Common UNIX

Printing System (CUPS). Qualquer sistema Linux que inclui CUPS oferece a interface administrativa CUPS baseada na web para a configuração de impressão CUPS. Ele também oferece arquivos de configuração no diretório `/etc/cups` para configurar impressoras e o serviço CUPS (daemon `cupsd`).

No RHEL, Fedora e outros sistemas Linux, você pode configurar a impressora com as janelas de configuração de impressão disponíveis nos desktops KDE e GNOME. Uma variedade de drivers torna possível imprimir em diferentes tipos de impressoras, bem como em impressoras que estão conectadas aos computadores na rede.

Você pode configurar seu computador como servidor de impressão Linux e também pode fazê-lo emular um servidor de impressão SMB (Windows). Depois que sua rede está configurada corretamente e uma impressora local está instalada, compartilhar essa impressora através da rede como um servidor de impressão UNIX ou SMB não é muito complicado.

Exercícios

Use esses exercícios para testar seus conhecimentos de configuração de impressoras no Linux. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostradas no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. Use a janela Printer Configuration para adicionar uma nova impressora chamada `myprinter` ao seu sistema (a impressora não precisa estar conectada para configurar uma fila de impressão para a impressora nova). Torne-a uma impressora PostScript genérica conectada a uma porta local serial, LPT ou outra porta.
2. Use o comando `lpc` para ver o status de todas suas impressoras.
3. Use o comando `lpr` para imprimir o arquivo `/etc/hosts` nessa impressora.
4. Verifique a fila de impressão da impressora para ver o trabalho de impressão que está lá.
5. Remova o trabalho de impressão da fila (cancele-o).
6. Usando a janela Printing, configure a configuração básica do servidor que publica suas impressoras de modo que outros sistemas em sua rede local possam imprimir em suas impressoras.
7. Habilite a administração remota de seu sistema a partir de um navegador.
8. Demonstre que você pode fazer a administração remota de seu sistema, abrindo um navegador web na porta 631 de outro sistema para o sistema Linux rodando o servidor de impressão.
9. Use o comando `netstat` para ver qual endereço o daemon `cupsd` está ouvindo (a porta de impressão é 631).
10. Exclua a entrada da impressora `myprinter` de seu sistema.

Configurando um servidor web

NESTE CAPÍTULO

Instalando um servidor web Apache

Configurando o Apache

Tomando o Apache seguro com iptables e SELinux

Criando hosts virtuais

Construindo um site seguro (HTTPS)

Verificando erros no Apache

Os servidores web são responsáveis por servir o conteúdo que você vê na internet todos os dias. De longe, o servidor web mais popular é o servidor web Apache (HTTPD), que é patrocinado pela Apache Software Foundation (<http://apache.org>). Como o Apache é um projeto de código aberto, ele está disponível em todas as principais distribuições Linux, incluindo o Fedora, RHEL e Ubuntu.

Você pode configurar um servidor web básico para rodar em Linux em apenas alguns minutos. Mas há um número enorme de maneiras pelas quais você pode configurar o servidor web Apache. Ele pode ser configurado para servir conteúdo para vários domínios (hospedagem virtual), fornecer comunicações criptografadas (HTTPS) e proteger parte de ou todo um site usando diferentes tipos de autenticação.

Este capítulo conduz você ao longo dos passos para instalar e configurar um servidor web Apache. Essas etapas incluem procedimentos para proteger seu servidor, bem como a utilização de uma variedade de módulos, de modo que você pode usar diferentes métodos de autenticação e linguagens de script em seu servidor web. Em seguida, descrevo como gerar certificados para criar um site HTTPS Secure Sockets Layer (SSL).

Entendendo o servidor web Apache

O Apache HTTPD (também conhecido como o Apache HTTPD Server) fornece o serviço com o qual os navegadores web clientes se comunicam. O processo daemon (`httpd`) executa em segundo plano em seu servidor e aguarda solicitações de clientes web. Navegadores da web fornecem essas conexões ao

daemon HTTP e enviam solicitações, que o daemon interpreta enviando os dados apropriados (como uma página web ou outro conteúdo).

O Apache HTTPD inclui uma interface que permite que os módulos sejam amarrados ao processo para lidar com partes específicas de uma solicitação. Entre outras coisas, há módulos disponíveis para lidar com o processamento de linguagens de script, como Perl ou PHP, dentro de documentos, web e para adicionar criptografia a conexões entre os clientes e o servidor.

O Apache começou como uma coleção de correções e melhorias do servidor HTTP feitas pelo Centro Nacional de Aplicações de Supercomputação (NCSA), da Universidade de Illinois, em Urbana-Champaign. O daemon NCSA HTTP foi o mais popular servidor HTTP da época, mas começou a ficar defasado depois que seu autor, Rob McCool, deixou o NCSA em meados de 1994.

ta

o projeto que veio do NCSA é o Mosaic. A origem da maioria dos navegadores modernos pode ser remontada até Mosaic.

No início de 1995, um grupo de desenvolvedores formou o Apache Group e começou a fazer modificações importantes na base de código do NCSA HTTPD.

O Apache logo substituiu o NCSA HTTPD como o servidor web mais popular, um título que mantém até hoje. O Grupo Apache mais tarde formou a Apache Software Foundation (ASF), para promover o desenvolvimento do Apache e outros softwares livres. Com o início de novos projetos na ASF, o servidor Apache ficou conhecido como Apache HTTPD, embora os dois termos sejam usados como sinônimos. Atualmente, a ASF tem mais de 100 projetos de alto nível, incluindo o Tomcat (que inclui as tecnologias de código-fonte aberto Java Servlet e JavaServer Pages), o Hadoop (um projeto que fornece computação distribuída de alta disponibilidade) e o SpamAssassin (um programa de filtragem de e-mail).

Obtendo e instalando o servidor web

Embora o Apache seja disponibilizado em todas as importantes distribuições Linux, muitas vezes ele é empacotado de maneiras diferentes. Na maioria dos casos, tudo que você precisa para iniciar um servidor web Apache simples é o pacote que contém o daemon Apache (`/usr/sbin/httpd`) e seus arquivos relacionados. No Fedora, RHEL e outros, o servidor web Apache vem no pacote `httpd`.

Entendendo o pacote httpd

Para examinar o pacote `httpd` no Fedora ou no RHEL antes de instalá-lo, baixe-o usando o comando `yumdownloader` e execute alguns poucos comandos `rpm` para visualizar seu conteúdo:

```
# yumdownloader httpd
# rpm -qpi httpd-*rpm
```

```

Name      : httpd                Relocations: (not
relocatable)
Version   : 2.2.22                Vendor: Fedora Project
Release   : 2.fc16                Build Date: Mon 05 Mar 2012
06:54:02 AM EST
Install Date: (not installed)    Build Host: x86-09.phx2.
fedoraproject.org
Group     : System Environment/Daemons Source RPM: httpd-2.2.22-2.
fc16.src.rpm
Size      : 2921066 License: ASL 2.0
Signature : RSA/8, Tue 06 Mar 2012 12:41:45 AM EST, Key ID
067f00b6a82ba4b7
Packager  : Fedora Project
URL       : http://httpd.apache.org/
Summary   : Apache HTTP Server
Description:
The Apache HTTP Server is a powerful, efficient, and extensible
web server.

```

O comando `yumdownloader` baixa a versão mais recente do pacote `httpd` para o diretório atual. O comando `rpm -qpi` consulta o pacote RPM `httpd` que você acabou de baixar para obter informações. Você pode ver que o pacote foi criado pelo Projeto Fedora, que é assinado e que é de fato o pacote Apache HTTP Server. Então, olhe dentro do pacote para ver os arquivos de configuração:

```

# rpm -qpc httpd-*rpm
/etc/httpd/conf/httpd.conf
/etc/httpd/conf.d/welcome.conf
/etc/httpd/conf/magic
/etc/sysconfig/httpd
/etc/logrotate.d/httpd
/etc/tmpfiles.d/httpd.conf
/var/www/error/HTTP_BAD_GATEWAY.html.var
/var/www/error/HTTP_BAD_REQUEST.html.var
/var/www/error/HTTP_FORBIDDEN.html.var
...

```

O principal arquivo de configuração é o `/etc/httpd/conf/httpd.conf` do Apache. O arquivo `welcome.conf` define a página inicial padrão para seu site, até que você adicione algum conteúdo. O arquivo de `magic` define regras que o servidor pode usar para descobrir um tipo de arquivo quando o servidor tentar abri-lo. Opções de linha de comando que são usadas com o daemon `httpd` são definidas no arquivo `/etc/sysconfig/httpd`.

O arquivo `/etc/logrotate.d/httpd` define como os arquivos de log produzidos pelo Apache são rodados. O arquivo `/etc/tmpfiles.d/httpd.conf` define um diretório que contém arquivos executáveis temporários (não há necessidade de mudar esse arquivo). As últimas entradas do arquivo de configuração estão no diretório `/var/www/error`. Arquivos no diretório definem as respostas que um usuário vê quando um erro é encontrado, como uma mensagem de arquivo não encontrado ou permissão de acesso negada.

Um local não mostrado na lista de arquivos de configuração `httpd` é o diretório `/etc/httpd/conf.d`. Qualquer arquivo no diretório que termina em `.conf` é puxado para o arquivo `httpd.conf` principal e é usado para configurar o Apache. A maioria dos pacotes de módulos que vêm com arquivos de configuração coloca-os no diretório `/etc/httpd/conf.d`. Por exemplo,

os módulos `mod_ssl` (para servidores web seguros) e `mod_python` (para interpretar código python) têm arquivos de configuração relacionados no diretório `/etc/httpd/conf.d`, chamados `ssl.conf` e `python.conf`, respectivamente.

Você pode simplesmente instalar o pacote `httpd` para iniciar a configuração de seu servidor web, mas pode preferir adicionar alguns outros pacotes que são frequentemente associados com o pacote `httpd`. Uma maneira de fazer isso é instalar o grupo Web Server inteiro como segue:

```
# yum groupinstall "Web Server"
```

Eis os pacotes, junto com o pacote `httpd`, no grupo Web Server no Fedora, que você recebe por padrão:

- **httpd-manual** — Preencha o diretório `/var/www/manual` com os manuais de documentação do Apache. Depois de iniciar o serviço `httpd`, você pode acessar esse conjunto de manuais a partir de um navegador para a máquina local digitando `http://localhost/manual` na caixa de localização. Externamente, em vez de `localhost`, você poderia usar o nome de domínio totalmente qualificado ou endereço IP do sistema. A tela Apache Documentation aparece como mostrado na Figura 17.1.

se a documentação do Apache diretamente a partir do servidor Apache local.

- **mod_ssl** — Contém o módulo e o arquivo de configuração necessários para o servidor web fornecer conexões seguras para os clientes usando Secure Sockets Layer (SSL) e Transport Layer Security (TLS). Esses recursos são necessários se você precisar de comunicações criptografadas para compras online ou outros dados que você deseja manter privados. O arquivo de configuração está localizado em: `/etc/httpd/conf.d/ssl.conf`.
- **crypto-utils** — Contém comandos para a geração de chaves e certificados necessários para fazer a segurança das comunicações com o servidor web Apache.
- **mod_perl** — Contém o módulo Perl (`mod_perl`), o arquivo de configuração e os arquivos associados necessários para permitir que o servidor web Apache execute diretamente qualquer código Perl.
- **mod_python** — Contém o módulo Python, o arquivo de configuração e a documentação associados com a execução de código Python diretamente no Apache.
- **php** — Contém o módulo PHP e o arquivo de configuração necessários para rodar scripts PHP diretamente no Apache. Pacotes relacionados incluem `php-ldap` (para a execução de código PHP que precisa acessar bancos de dados LDAP) e `php-mysql` (para adicionar suporte de banco de dados ao servidor Apache).
- **squid** — Como mencionado no Capítulo 14 (“Administrando redes”), o servidor proxy Squid fornece serviços de proxy para os protocolos específicos (como HTTP). Apesar de não fornecer conteúdo HTTP, um servidor proxy Squid normalmente encaminha solicitações de clientes de proxy para a internet ou outra rede que fornece conteúdo web. Isso proporciona um meio de controlar ou filtrar conteúdo que clientes podem alcançar de casa, da escola ou do local de trabalho.
- **distcache** — Usado com o pacote `mod_ssl` para cache de dados de sessão SSL/TLS.
- **webalizer** — Contém ferramentas para analisar dados do servidor web.

Pacotes opcionais do grupo Web Server também são exibidos a partir do comando `yum groupinfo`. Alguns desses pacotes oferecem maneiras especiais de fornecer conteúdo, tais como wikis (`moin`), sistemas de gerenciamento de conteúdo (`drupal7`) e blogs (`wordpress`). Outros incluem ferramentas para plotar gráficos de estatísticas web (`awstats`) ou oferecem alternativas web “leves” a um servidor Apache (`lighttpd` e `cherokee`).

Instalando o Apache

Embora você só precise de `httpd` para começar com um servidor web Apache, se estiver começando a aprender sobre o Apache, você deve também instalar os manuais (`httpd-manual`). Se estiver pensando em criar um site seguro (SSL) e, possivelmente, gerar algumas estatísticas sobre seu site, você pode simplesmente instalar todo o grupo:

```
# yum groupinstall "Web Server"
```

Supondo que você tenha uma conexão de internet com o repositório Fedora (ou o repositório RHEL, se você estiver usando RHEL), todos os pacotes obrigatórios e padrão desse grupo são instalados. Você tem todo o software de que precisa para fazer os procedimentos e exercícios descritos neste capítulo.

Iniciando o Apache

Para fazer o servidor web Apache funcionar, é recomendável ativar o serviço para iniciar a cada reinicialização e iniciá-lo imediatamente. No Red Hat Enterprise Linux (RHEL até 6) e em distribuições mais antigas do Fedora, você pode digitar o seguinte como root:

```
# chkconfig httpd on
# service httpd start
Starting httpd: [ OK ]
```

Em sistemas Fedora recentes, você habilita e inicia `httpd` usando o comando `systemctl`:

```
# systemctl enable httpd.service
# systemctl start httpd.service
# systemctl status httpd.service
httpd.service - The Apache HTTP Server (prefork MPM)
  Loaded: loaded (/lib/systemd/system/httpd.service; enabled)
  Active: active (running) since Tue, 01 May 2012 20:18:44
         -0400; 8s ago
  Process: 26918 ExecStart=/usr/sbin/httpd $OPTIONS -k start
           (code=exited, status=0/SUCCESS)
  Main PID: 26919 (httpd)
  CGroup: name=systemd:/system/httpd.service
          └─ 26919 /usr/sbin/httpd -k start
             └─ 26921 /usr/sbin/httpd -k start
  ...
```

Quando o serviço `httpd` inicia, oito processos `daemon httpd` são carregados por padrão, para responder às solicitações ao servidor web. Você pode configurar mais ou menos `daemons httpd` para serem iniciados com base em configurações no arquivo `httpd.conf` (descritas na seção “Entendendo os arquivos de configuração do Apache”). Para alterar o comportamento do `daemon httpd`, você pode adicionar opções à variável `OPTIONS=` no arquivo `/etc/sysconfig/httpd`.

Como há diferentes versões do `httpd` por aí, verifique a página `man (man httpd)` para ver quais opções podem ser passadas para o `daemon httpd`. Por exemplo, definir `OPTIONS="-e debug"` aumenta o nível de registro em log de modo que o número máximo de mensagens do Apache é enviado em arquivos de log. Reinicie o serviço `httpd` para que as alterações tenham efeito. Digite o comando `ps` para verificar se as opções fizeram efeito:

```
$ ps -ef | grep httpd
root      27126      1 0 16:56 ?        00:00:00 /usr/sbin/httpd -e debug
apache    27128 27126 0 16:56 ?        00:00:00 /usr/sbin/httpd -e debug
```

Se você adicionou uma opção de depuração, lembre-se de remover essa opção de `/etc/sysconfig/httpd` quando terminar de depurar o Apache e reiniciar o serviço. Se deixar a depuração ativa, irá rapidamente encher seus arquivos de log.

Tornando o Apache seguro

Para tornar o Apache seguro, você precisa estar ciente dos recursos padrão de segurança (permissões do Linux, posse, firewalls e segurança reforçada Linux), bem como recursos de segurança que são específicos do Apache. As seções a seguir descrevem os recursos de segurança que se relacionam com o Apache.

Permissões e posse de arquivos no Apache

O processo daemon `httpd` é executado sob o usuário `apache` e o grupo `apache`. Por padrão, o conteúdo HTML é armazenado no diretório `/var/www/html` (como determinado pelo valor de `DocumentRoot` no arquivo `httpd.conf`).

Para o daemon `httpd` ser capaz de acessar esse conteúdo, as permissões padrão do Linux se aplicam: se a permissão de leitura não estiver ativada para “outros” usuários, ela deve estar ativada para o usuário ou o grupo `apache` de modo que os arquivos possam ser lidos e servidos aos clientes. Da mesma maneira, qualquer diretório que o daemon `httpd` precise percorrer para chegar ao conteúdo deve ter permissão de execução para o usuário `apache`, o grupo `apache` ou outro usuário.

Embora você não possa logar como o usuário `apache` (`/sbin/nologin` é o shell padrão), você pode criar conteúdo como `root` e alterar sua posse (comando `chown`) ou permissão (comando `chmod`). Com frequência, porém, contas separadas de usuário ou grupo são adicionadas para criar conteúdo que pode ser lido por todos (outro), mas apenas gravável por esse usuário ou grupo especial.

Apache e iptables

Se tiver bloqueado seu firewall `iptables` no Linux, você precisa abrir várias portas para que os clientes possam falar com o Apache através do firewall. Os serviços web padrão (HTTP) são acessíveis pela porta TCP 80 e serviços web seguros (HTTPS) são acessíveis pela porta TCP 443. Para verificar quais portas estão sendo usadas pelo servidor `httpd`, use o comando `netstat`:

```

# netstat -tupln | grep httpd
tcp    0      0 :::80      :::*        LISTEN    29169/httpd
tcp    0      0 :::443     :::*        LISTEN    29169/httpd
```

A saída mostra que o daemon `httpd` (ID de processo 29169) está ouvindo todos os endereços na porta 80 (`:::80`) e na porta 443 (`:::443`). Ambas as portas estão associadas ao o protocolo TCP (`tcp`). Para abrir essas portas no Fedora ou no Red Hat Enterprise Linux, você pode adicionar regras ao arquivo `/etc/sysconfig/iptables` (em algum lugar antes de um `DROP` ou `REJECT` final), como a seguir:

```

-A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT
```

Reinicie o serviço de `iptables` para as novas regras entrarem em vigor.

Apache e SELinux

Se Security Enhanced Linux (SELinux) estiver configurado como Enforcing (como está por padrão no Fedora e no Red Hat Enterprise Linux), o SELinux acrescenta outra camada de segurança por cima de seu serviço `httpd`. Basicamente, o SELinux serve para proteger o sistema contra danos causados pelo daemon `httpd`. Isso é feito criando políticas que:

- Negam acesso a arquivos que não são definidos para os contextos de arquivos corretos. Para `httpd` no SELinux, há diferentes contextos de arquivos de conteúdo, de configuração, de log, scripts e outros relacionados com `httpd`. Qualquer arquivo que não esteja configurado para o contexto apropriado não será acessível para o daemon `httpd`.
- Impedem que recursos inseguros sejam usados, como o upload de arquivos e autenticação em texto simples, definindo booleanos para tais recursos na posição desativado. Você pode ativar booleanos seletivamente conforme eles forem necessários, caso atendam aos requisitos de segurança.
- Impedem que o daemon `httpd` acesse recursos não padrão, como uma porta fora das portas padrão que o serviço esperaria para usar.

Uma descrição completa do SELinux está contida no Capítulo 24, “Aprimorando a segurança do Linux com o SELinux”. Mas aqui estão algumas coisas específicas que você deve saber sobre o uso do SELinux com o serviço `httpd` Apache:

- **Desative o SELinux.** Você não precisa usar o SELinux. Você pode configurar o SELinux para funcionar no modo Permissive, caso ache que é muito difícil e desnecessário criar as políticas dele exigidas para que o servidor web trabalhe com ele no modo Enforcing. Você pode mudar o modo para Permissive editando o arquivo `/etc/sysconfig/selinux` de modo que o valor de `SELINUX` seja definido da forma mostrada abaixo. Com isso configurado, da próxima vez que você reiniciar o sistema, ele estará no modo Permissive. Isso significa que se você quebrar políticas do SELinux, esse evento é registrado, mas não impedido (como seria no modo Enforcing).

```
SELINUX=permissive
```

- **Leia a página `man httpd_selinux`.** Digite `man httpd_selinux` no shell. Essa página `man` vai mostrar os contextos de arquivos apropriados e os booleanos disponíveis.
- **Use localizações padrão para arquivos.** Quando você cria novos arquivos, eles herdam seus contextos dos diretórios em que estão. Como `/etc/httpd` está configurado para o contexto certo para arquivos de configuração, `/var/www/html` está certo para arquivos de conteúdo etc., simplesmente copiar os arquivos para essas localizações ou criar novos arquivos nelas faz com que os contextos de arquivos sejam definidos corretamente.
- **Modifique o SELinux para permitir recursos não padrão.** Você pode querer servir conteúdo web a partir do diretório `/mystuff` ou colocar os arquivos de configuração no diretório `/etc/whatever`. Da mesma maneira, você pode querer permitir que os usuários de seu servidor façam upload de arquivos, executem scripts ou ativem outros recursos que estão

desabilitados no SELinux por padrão. Nesses casos, você pode usar comandos do SELinux para definir os contextos de arquivos e booleanos de que você precisa para fazê-lo trabalhar do jeito que você quer.

Não deixe de ler o Capítulo 24 para saber mais sobre o SELinux.

Entendendo os arquivos de configuração do Apache

Os arquivos de configuração para Apache HTTPD são incrivelmente flexíveis, o que significa que você pode configurar o servidor para se comportar de quase todas as maneiras que quiser. Essa flexibilidade vem com o custo de uma maior complexidade na forma de um grande número de opções de configuração (chamadas de *diretivas*). Mas, na prática, você precisa estar familiarizado com algumas diretivas.

ta

```
Consulte http://httpd.apache.org/docs/ para ver uma lista completa das diretivas suportadas pelo Apache. Se tiver o httpd-manual instalado, você pode ler as descrições dessas diretivas e outros recursos do Apache abrindo o manual a partir do servidor executando o Apache que você tem: http://localhost/manual/.
```

No Fedora e no RHEL, o principal arquivo de configuração básica do servidor Apache está em `/etc/httpd/conf/httpd.conf`. Além desse arquivo, qualquer outro que termina com `.conf` no diretório `/etc/httpd/conf.d` também é utilizado para configuração do Apache (com base em uma linha `Include` no arquivo `httpd.conf`). No Ubuntu, a configuração do Apache é armazenada em arquivos de texto lidos pelo servidor Apache, começando com `/etc/apache2/apache2.conf`. A configuração é lida do início ao fim, com a maioria das diretivas sendo processadas na ordem em que são lidas.

Usando diretivas

O escopo de muitas diretivas de configuração pode ser alterado com base no contexto. Em outras palavras, alguns parâmetros podem ser definidos em um nível global e depois mudados para um arquivo, diretório ou máquina virtual específicos. Outras diretivas são sempre de natureza global, como aquelas especificando quais endereços IP o servidor ouve. Outras ainda só são válidas quando aplicadas a uma localização específica.

Localizações são configuradas na forma de uma tag inicial, contendo o tipo de localização e uma localização de recursos, seguidos pelas opções de configuração para essa localização e terminando com uma tag final. Essa forma é muitas vezes chamada de *bloco de configuração* e é muito parecida com código HTML. Um tipo especial de bloco de configuração, conhecido como *bloco de localização*, é usado para limitar o escopo das diretivas a arquivos ou diretórios específicos. Esses blocos têm a seguinte forma:

```
<especificador da tag de localização>
```

```
(opções específicas para objetos correspondentes ao especificador
```

```
entram neste bloco)
</tag de localização >
```

Há diferentes tipos de tags de localização e eles são selecionados com base no tipo de localização de recurso que está sendo especificado. O especificador que é incluído na tag inicial é tratado com base no tipo da tag de localização. As tags de localização que geralmente você usa e encontra são `Directory`, `Files` e `Location`, o que limita o escopo das diretivas a determinados diretórios, arquivos ou localizações, respectivamente.

- Tags `Directory` são usadas para especificar um caminho com base na localização no sistema de arquivos. Por exemplo, `<Directory />` se refere ao diretório-raiz no computador. Diretórios herdam configurações dos diretórios acima deles, com o bloco `Directory` mais específico substituindo os menos específicos, independentemente da ordem em que aparecem nos arquivos de configuração.
- Tags `Files` são usadas para especificar os arquivos pelo nome. Tags `Files` podem estar contidas dentro de blocos `Directory` para limitá-los a arquivos sob esse diretório. Configurações dentro de um bloco `Files` substituem os blocos `Directory`.
- Tags `Location` são usadas para especificar o URI usado para acessar um arquivo ou diretório. Essas tags são diferentes das tags `Directory` no sentido de que se relacionam com o endereço contido na solicitação e não com a localização real do arquivo no disco. Tags `Location` são processadas por último e substituem as configurações em blocos `Directory` e `Files`.

Versões de correspondência de texto dessas tags — `DirectoryMatch`, `FilesMatch` e `LocationMatch` — têm a mesma função, mas podem conter expressões regulares na especificação de recursos. Blocos `FilesMatch` e `LocationMatch` são processados ao mesmo tempo em que `Files` e `Location`, respectivamente. Blocos `DirectoryMatch` são processados depois dos blocos `Directory`.

O Apache também pode ser configurado para processar opções de configuração contidas em arquivos com o nome especificado na diretiva `AccessFileName` (que é geralmente `.htaccess`). Diretivas em arquivos de configuração de acesso são aplicadas a todos os objetos sob o diretório que elas estão contidas, incluindo subdiretórios e seus conteúdos. Os arquivos de configuração de acesso são processados ao mesmo tempo, como blocos `Directory`, utilizando uma ordem similar de “correspondência de texto mais específica”.

ta

ivos de controle de acesso são úteis para permitir que os usuários alterem as configurações específicas sem ter so aos arquivos de configuração do servidor. As diretivas de configuração permitidas dentro de um arquivo de guração de acesso são determinadas pela configuração `AllowOverride`, no diretório em que estão idas. Algumas diretivas não fazem sentido nesse nível e geralmente resultam em uma mensagem “server internal” ao tentar acessar a URI. A opção `AllowOverride` é coberta em detalhes em <http://httpd.apache.org/docs/mod/core.html#allowoverride>.

Três diretivas comumente encontradas em blocos de localização e arquivos de controle de acesso são `DirectoryIndex`, `Options` e `ErrorDocument`:

- `DirectoryIndex` diz ao Apache qual arquivo carregar quando o URI contém um diretório, mas não um nome de arquivo. Essa diretiva não funciona em blocos `Files`.
- `Options` é usada para ajustar a forma como o Apache lida com arquivos dentro de um diretório. A opção `ExecCGI` diz ao Apache que arquivos no diretório podem ser executados como scripts CGI e a opção `Includes` diz ao Apache que inclusões do lado do servidor (SSI) são permitidas. Outra opção comum é a opção `Indexes`, que instrui o Apache a gerar uma lista de arquivos se um dos nomes encontrados na configuração `DirectoryIndex` estiver faltando. Uma lista de opções absolutas pode ser especificada ou a lista de opções pode ser modificada pela adição de um sinal de + ou - na frente de um nome de opção. Veja <http://httpd.apache.org/docs/mod/core.html#options> para obter mais informações.
- Diretivas `ErrorDocument` podem ser usadas para especificar um arquivo contendo mensagens para enviar aos clientes da web quando um determinado erro ocorre. A localização do arquivo é relativa ao diretório `/var/www`. A diretiva deve especificar um código de erro e o URI completo para o documento de erro. Códigos de erro possíveis incluem 403 (acesso negado), 404 (arquivo não encontrado) e 500 (erro interno do servidor). Você pode encontrar mais informações sobre a diretiva `ErrorDocument` em <http://httpd.apache.org/docs/mod/core.html#errordocument>. Como exemplo, quando um cliente solicita uma URL do servidor que não foi encontrado, a seguinte linha `ErrorDocument` faz com que o código de erro 404 seja enviado ao cliente com uma mensagem de erro que está listada no arquivo `/var/www/error/HTTP_NOT_FOUND.html.var`.

```
ErrorDocument 404 /error/HTTP_NOT_FOUND.html.var
```

Outro uso comum dos blocos de localização e arquivos de controle de acesso é limitar o acesso a um recurso. A diretiva `Allow` pode ser usada para permitir o acesso a hosts correspondentes, e a diretiva `Deny` pode ser usada para proibir. Ambas as opções podem ocorrer mais de uma vez dentro de um bloco e são tratadas com base na configuração `Order`. Configurar `Order` como `Deny, Allow` permite o acesso a qualquer máquina que não esteja listada em uma diretiva `Deny`. Uma configuração de `Allow, Deny` nega acesso a qualquer host não permitido em uma diretiva `Allow`.

Como a maioria das outras opções, a opção `Allow` ou `Deny` mais específica para um host é usada, o que significa que você pode negar (`Deny`) acesso a um intervalo e permitir (`Allow`) acesso a subconjuntos desse intervalo. Ao adicionar a opção `Satisfy` e alguns parâmetros adicionais, você pode adicionar a autenticação por senha. Para mais informações sobre controle de acesso, consulte http://httpd.apache.org/docs/mod/mod_access.html.

Entendendo as configurações padrão

A razão por que você pode começar a usar o servidor web Apache logo que o instala é que o arquivo `httpd.conf` inclui configurações padrão que informam ao servidor coisas como onde encontrar

conteúdo da web, scripts, arquivos de log e outros itens que o servidor precisa para operar. Esse arquivo também inclui configurações que informam ao servidor quantos processos dele executar por vez e como o conteúdo do diretório é exibido.

Se quiser hospedar um único site (como para o domínio `example.com`), você pode simplesmente adicionar conteúdo ao diretório `/var/www/html` e adicionar o endereço de seu site a um servidor de DNS para que outros possam procurá-lo. Você pode, então, alterar diretivas, tais como aquelas descritas na seção anterior, conforme você precisar.

Para ajudar você a entender as configurações que vêm no arquivo `httpd.conf` padrão, mostro a seguir algumas dessas configurações com descrições abaixo. Removi comentários e reorganizei algumas das configurações para maior clareza.

As definições a seguir mostram localizações em que o servidor `httpd` aceita solicitações para baixar (*get*) e subir (*put*) conteúdo por padrão:

```
ServerRoot "/etc/httpd"
PidFile run/httpd.pid
Include conf.d/*.conf
ErrorLog logs/error_log
CustomLog logs/access_log combined
DocumentRoot "/var/www/html"
ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"
```

A diretiva `ServerRoot` identifica `/etc/httpd` como a localização onde arquivos de configuração, arquivos de log e arquivos PID são armazenados. O caminho definido pela configuração `PidFile` (`run/httpd.pid`) indica onde o processo de identificação do processo `httpd` é armazenado. Ele é anexado ao caminho definido pela configuração `ServerRoot`, o que resulta na gravação do PID em `/etc/httpd/run/httpd.pid/`.

No ponto dentro do arquivo em que a linha `Include` aparece, todos os arquivos que terminam em `.conf` no diretório `/etc/httpd/conf.d` são incluídos no arquivo `httpd.conf`. Os arquivos de configuração são frequentemente associados com módulos Apache ou com blocos de host virtuais, que você mesmo pode adicionar à configuração de host virtual em arquivos separados (ver a seção “Adicionando um host virtual ao Apache”).

À medida que são encontrados erros e conteúdo é servido, mensagens sobre essas atividades são colocadas em arquivos indicados pelas entradas `ErrorLog` e `CustomLog`. A partir das entradas mostradas aqui, esses registros são armazenados nos diretórios `/etc/httpd/logs/error_log` e `/etc/httpd/logs/access_log`, respectivamente. Esses logs também estão vinculados ao diretório `/var/log/httpd`, assim você pode acessar o mesmo arquivo a partir dessa localização também.

As diretivas `DocumentRoot` e `ScriptAlias` determinam onde o conteúdo que é servido pelo servidor `httpd` é armazenado. Normalmente, você coloca um arquivo `index.html` no diretório `DocumentRoot` (`/var/www/html`, por padrão) como página inicial e depois adiciona outros conteúdos, conforme necessário. A diretiva `ScriptAlias` diz ao daemon `httpd` que todos os scripts

solicitados a partir do diretório `cgi-bin` devem ser encontrados no diretório `/var/www/cgi-bin`. Por exemplo, um cliente pode acessar um script localizado em `/var/www/cgi-bin/script.cgi` digitando uma URL, como `http://example.com/cgi-bin/script.cgi`.

Além de localizações de arquivo, você pode encontrar outras informações no arquivo `httpd.conf`. Eis alguns exemplos:

```
Listen 80
User apache
Group apache
ServerAdmin root@localhost
DirectoryIndex index.html index.html.var
AccessFileName .htaccess
```

A diretiva `Listen 80` diz para o `httpd` ouvir solicitações de entrada na porta 80 (a porta padrão para o protocolo do servidor HTTP na internet). As diretivas `User` e `Group` dizem para o `httpd` rodar como `apache` tanto para o usuário como para o grupo. O valor de `ServerAdmin` (`root@localhost`, por padrão) é publicado em algumas páginas web para informar aos usuários para onde enviar um e-mail se houver problemas com o servidor.

O `DirectoryIndex` lista os arquivos que `httpd` servirá se um diretório for solicitado. Por exemplo, se um navegador solicitasse `http://host/whatever/`, `httpd` veria se `/var/www/html/whatever/index.html` existe e o serviria em caso afirmativo. Caso contrário, `httpd` procuraria `index.html.var`. Se esse arquivo não pudesse ser encontrado, o conteúdo do diretório seria exibido.

A diretiva `AccessFileName` diz para `httpd` usar o conteúdo do arquivo `.htaccess` se ele existir em um diretório para ler configurações que se aplicam ao acesso a esse diretório. Por exemplo, o arquivo pode ser utilizado para solicitar proteção por senha para o diretório ou para indicar que o conteúdo dele deve ser exibido de determinada maneira. Para esse arquivo funcionar, porém, um contêiner `Directory` (descrito a seguir) teria de ter `AllowOverride` aberto.

Os seguintes contêineres `Directory` definem o comportamento quando o diretório-raiz (`/`) e o diretório `/var/www/html` são acessados:

```
<Directory />
    Options FollowSymLinks
    AllowOverride None
</Directory>
<Directory "/var/www/html">
    Options Indexes FollowSymLinks
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>
```

O primeiro contêiner `Directory (/)` indica que se `httpd` tentar acessar todos os arquivos no sistema de arquivos Linux, `httpd` terá permissão para seguir arquivos simbolicamente vinculados a outros arquivos (`Options FollowSymLinks`). A diretiva `AllowOverride None` impede que arquivos `.htaccess` sobrescrevam configurações para esse diretório. Essas configurações se aplicam a todas as subpastas que não são definidas em outros contêineres `Directory`.

O contêiner `/var/www/html` `Directory` também segue vínculos simbólicos e não permite substituições. Ele tem diretivas `Order` e `Allow` (descritas anteriormente) que não impedem qualquer acesso ao servidor, mas podem ser modificadas para indicar que somente hosts selecionados podem acessar o diretório `/var/www/html` e seu conteúdo.

Se todas as configurações que acabamos de descrever funcionarem para você, você pode começar a adicionar o conteúdo que quiser aos diretórios `var/www/html` e `var/www/cgi-bin`. Uma razão pela qual você pode não estar satisfeito com a configuração padrão é que você pode querer servir conteúdo para vários domínios (como `example.com`, `example.org` e `example.net`). Para isso, é preciso configurar hosts virtuais. As máquinas virtuais, que são descritas em maior detalhe na próxima seção, são uma ferramenta conveniente (e quase essencial) para alterar o comportamento do servidor com base no endereço dele ou no nome para o qual uma solicitação é dirigida. Opções de configuração mais globais são aplicadas a máquinas virtuais, mas podem ser substituídas por diretivas dentro do bloco `VirtualHost`.

Adicionando um host virtual ao Apache

O Apache suporta a criação de sites separados dentro de um mesmo servidor para manter o conteúdo separado. Sites individuais são configurados no mesmo servidor naquilo que é chamado de hosts virtuais.

Hosts virtuais são, na realidade, apenas uma maneira de ter o conteúdo de vários nomes de domínio disponíveis a partir do mesmo servidor Apache. Em vez de precisar ter um sistema físico para servir conteúdo para cada domínio, você pode servir conteúdo para vários domínios a partir do mesmo sistema operacional.

Um servidor Apache que está fazendo hospedagem virtual pode ter vários nomes de domínio que são convertidos para o endereço IP do servidor. O conteúdo que é servido para um cliente web é baseado no nome usado para acessar o servidor.

Por exemplo, se um cliente solicitar ao servidor o nome `www.example.com`, ele será direcionado a um contêiner de host virtual que tem seu `ServerName` configurado para responder a `www.example.com`. O contêiner forneceria a localização do conteúdo e possivelmente diferentes logs de erros ou diretivas `Directory` das configurações globais. Dessa forma, cada máquina virtual pode ser gerenciada como se ela estivesse em uma máquina separada.

Para usar hospedagem virtual baseada em nome, ative a diretiva `NameVirtualHost`. Então, adicione quantos contêineres `VirtualHost` você quiser. Veja como configurar um host virtual:

is de ativar `NameVirtualHost`, seu `DocumentRoot` padrão (`/var/www/html`) não é mais usado se em acessa o servidor por endereço IP ou algum nome que não está definido em um contêiner `VirtualHost`. vez disso, o contêiner `VirtualHost` primeiro será usado como a localização padrão para o servidor.

1. No Fedora ou no RHEL, crie um arquivo chamado `/etc/httpd/conf.d/example.org.conf` usando este modelo:

```
NameVirtualHost *:80
<VirtualHost *:80>
    ServerAdmin    webmaster@example.org
    ServerName     www.example.org
    ServerAlias    web.example.org
    DocumentRoot  /var/www/html/example.org/
DirectoryIndex  index.php index.html index.htm
</VirtualHost>
```

Este exemplo inclui as seguintes definições:

- A linha `NameVirtualHost` diz ao Apache para determinar a partir de qual host virtual servir documentos com base no hostname fornecido pelo cliente HTTP. O `*:80` significa que as solicitações para a porta 80 em qualquer endereço IP serão tratadas dessa maneira.
 - Da mesma maneira, a especificação `*:80` no bloco `VirtualHost` indica o endereço e a porta a que esse host virtual se aplica. Com vários endereços IP associados com seu sistema Linux, o `*` pode ser substituído por um endereço IP específico. A porta é opcional para ambas as especificações, `NameVirtualHost` e `VirtualHost`, mas sempre deve ser usada para evitar a interferência em hosts virtuais SSL.
 - As linhas `ServerName` e `ServerAlias` informam ao Apache os nomes sob os quais esse host virtual deve ser reconhecido; portanto, substitua-os pelos nomes apropriados para seu site. Você pode deixar a linha `ServerAlias` se não tiver nenhum nome alternativo para o servidor e pode especificar mais de um nome por linha `ServerAlias` ou ter várias linhas `ServerAlias` se você tiver vários nomes alternativos.
 - O `DocumentRoot` especifica onde os documentos web (conteúdo servido por esse site) são armazenados. Embora apresentado como um subdiretório que você cria sob o `DocumentRoot` padrão (`/var/www/html`), muitas vezes sites são anexados aos diretórios de usuários específicos (como `/home/chris/public_html`), de modo que cada site possa ser gerenciado por um usuário diferente.
2. Com o host habilitado, use `apachectl` para verificar a configuração e depois fazer uma reinicialização `graceful`:

```
# apachectl configtest
Syntax OK
# apachectl graceful
```

Desde que você tenha registrado o sistema com um servidor DNS, um navegador deve ser capaz de acessar esse site usando `www.example.org` ou `web.example.org`. Se isso funcionar, você pode começar a adicionar outros hosts virtuais ao sistema igualmente.

Outra forma de ampliar o uso de seu site é permitir que múltiplos usuários compartilhem conteúdo produzido por eles no servidor que você administra. Você pode permitir que usuários adicionem o conteúdo que eles quiserem compartilhar por meio de seu servidor web em um subdiretório dos diretórios deles, como descrito na próxima seção.

ta

ter hosts virtuais individuais em arquivos separados é uma maneira conveniente de gerenciar hosts virtuais. Mas deve ter o cuidado de manter seu host primário virtual em um arquivo que será lido antes dos outros, porque o primeiro host virtual recebe solicitações de nomes de sites que não correspondem a nenhum em sua configuração. Em um ambiente de hospedagem web comercial, é comum criar um host virtual padrão especial que contém uma mensagem de erro indicando que nenhum site com esse nome foi configurado.

Permitindo que os usuários publiquem seu próprio conteúdo web

Nas situações em que você não tem a capacidade de criar uma máquina virtual para cada usuário ao qual deseja fornecer espaço na web, você pode facilmente fazer uso do módulo `mod_userdir` no Apache. Com esse módulo habilitado (que, por padrão, não está), o diretório `public_html` no diretório inicial de cada usuário está disponível para a web em `http://servername/~username/`.

Por exemplo, um usuário chamado `wtucker` em `www.example.org` armazena conteúdo web em `/home/wtucker/public_html`. Esse conteúdo estará disponível em `http://www.example.org/~wtucker`.

Faça essas alterações no arquivo `/etc/httpd/conf/httpd.conf` para permitir que os usuários publiquem conteúdo da web a partir de seus próprios diretórios iniciais:

1. Modifique o bloco `<IfModule mod_userdir.c>` para desativar com um caractere de comentário as linhas `UserDir disabled` e `UserDir public_html`, de modo que ele apareça da seguinte maneira (eu removi as linhas de comentário adicionais para maior clareza):

```
<IfModule mod_userdir.c>
  # UserDir disabled
  UserDir public_html
</IfModule>
```

2. Desative com um caractere de comentário o bloco de diretiva `<Directory /home/*/public_html>` e altere as configurações que você quiser. Essa é a forma como o bloco ficará:

```
<Directory /home/*/public_html>
  AllowOverride FileInfo AuthConfig Limit
  Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
  <Limit GET POST OPTIONS>
```

```
    Order allow,deny
    Allow from all
</Limit>
<LimitExcept GET POST OPTIONS>
    Order deny,allow
    Deny from all
</LimitExcept>
</Directory>
```

3. Faça seus usuários criarem seus próprios diretórios `public_html` em seus próprios diretórios iniciais.

```
$ mkdir $HOME/public_html
```

4. Configure a permissão de execução para permitir que o daemon `httpd` acesse o diretório inicial:

```
# chmod +x /home /home/*
```

5. Configure o contexto de arquivo do SELinux corretamente, de modo que o SELinux permita que o daemon `httpd` acesse o conteúdo:

```
# chcon -R --reference=/var/www/html/ /home/*/public_html
```

6. Defina o booleano SELinux para permitir que os usuários compartilhem conteúdo HTML a partir de seus diretórios iniciais:

```
# setsebool -P httpd_enable_homedirs true
```

7. Reinicie ou recarregue o serviço `httpd`.

Nesse ponto, apontando um navegador web para `http://hostname/~user`, você deve ser capaz de acessar o conteúdo colocado no diretório `public_html` de um usuário.

Protegendo seu tráfego na web com SSL/TLS

Todos os dados que você compartilha a partir de seu site usando o protocolo HTTP padrão são enviados em texto claro. Isso significa que qualquer um que pode ver o tráfego em uma rede entre o servidor e seu cliente pode ver seus dados desprotegidos. Para garantir essas informações, você pode adicionar certificados a seu site (assim um cliente pode validar quem você é) e criptografar seus dados (assim ninguém pode espionar sua rede e ver seus dados).

Aplicações de comércio eletrônico, como compras online e operações bancárias, são geralmente criptografadas usando as especificações Secure Sockets Layer (SSL) ou Transport Layer Security (TLS). A TLS é baseada na versão 3.0 das especificações SSL e, portanto, são de naturezas muito semelhantes. Devido a essa semelhança — e como o SSL é mais antigo — a sigla SSL costuma ser usada para se referir a ambas as variedades. Para conexões de web, a conexão SSL é estabelecida primeiro e, então, a comunicação HTTP normal é “tunelada” através dela.

ta

o a negociação SSL ocorre antes de qualquer comunicação HTTP, a hospedagem virtual baseada em nome (que re na camada HTTP) não funciona facilmente com SSL. Como consequência, cada host SSL virtual que você figura deve ter um único endereço IP. (Consulte o site do Apache para mais informações: pd.apache.org/docs/vhosts/name-based.html.)

Enquanto você está estabelecendo uma conexão entre um cliente SSL e um servidor SSL, a criptografia assimétrica (chave pública) é usada para verificar as identidades e estabelecer os parâmetros e a chave de sessão. Um algoritmo de criptografia simétrica, como DES ou RC4, é então usado com a chave negociada para criptografar os dados que são transmitidos durante a sessão. O uso de criptografia assimétrica durante a fase de *handshaking* permite a comunicação segura, sem o uso de uma chave pré-compartilhada, e a criptografia simétrica é mais rápida e mais prática para uso sobre os dados da sessão.

Para o cliente verificar a identidade do servidor, este deve ter uma chave privada previamente gerada, bem como um certificado que contém a chave pública e informações sobre o servidor. Esse certificado deve ser verificável usando uma chave pública, que é conhecida pelo cliente.

Os certificados são geralmente assinados digitalmente por uma autoridade de certificação (CA) independente que verificou a identidade do solicitante e a validade da solicitação para ter o certificado assinado. Na maioria dos casos, a CA é uma empresa que fez um acordo com o fornecedor do navegador web para ter seu próprio certificado instalado e considerado confiável por instalações de cliente padrão. A CA então cobre o operador do servidor pelos seus serviços.

Autoridades de certificação comerciais variam em preço, recursos e suporte ao navegador, mas lembre-se de que o preço não é sempre um indicador de qualidade. Algumas CAs populares são a InstantSSL (<http://www.instantssl.com>), a Thawte (<http://www.thawte.com>) e a VeriSign (<http://www.verisign.com>).

Você também tem a opção de criar certificados autoassinados, embora estes devam ser usados apenas para teste ou quando um número muito pequeno de pessoas estará acessando seu servidor e você não planeja ter certificados em várias máquinas. Instruções para gerar um certificado autoassinado estão inclusas na seção “Gerando uma chave SSL e um certificado autoassinado”.

A última opção é executar sua própria autoridade de certificação. Isso provavelmente só é prático se você tiver um pequeno número de usuários esperados e os meios para distribuir seu certificado CA para eles (incluindo ajudá-los a instalar em seus navegadores). O processo para a criação de uma CA é muito complexo para ser abordado neste livro, mas é uma alternativa interessante à geração de certificados autoassinados.

As próximas seções descrevem como comunicações HTTPS são configuradas por padrão no Fedora e no RHEL quando você instala o pacote `mod_ssl`. Depois disso, descrevo como melhor configurar comunicações SSL gerando suas próprias chaves e certificados SSL para usar com o servidor web (rodando em um sistema Fedora ou RHEL) configurado neste capítulo.

Entendendo como o SSL é configurado

Se você tiver instalado o pacote `mod_ssl` no Fedora ou no RHEL (o que é feito por padrão se você

instalou o grupo Web Server), um certificado autoassinado e uma chave privada são criados. Isso permite que você imediatamente use o protocolo HTTPS para se comunicar com o servidor web.

Embora a configuração padrão do `mod_ssl` permita que você tenha comunicações criptografadas entre seu servidor web e seus clientes, pois o certificado é autoassinado, um cliente acessando seu site vai ser avisado de que o certificado não é confiável. Para começar a explorar a configuração de SSL para o servidor web Apache, verifique se o pacote `mod_ssl` está instalado no servidor que executa o serviço Apache (`httpd`):

```
# yum install mod_ssl
```

O pacote inclui o módulo `mod_ssl` necessário para implementar SSL em seu servidor web (`mod_ssl.so`) e um arquivo de configuração para seus hosts SSL:

`/etc/httpd/conf.d/ssl.conf`. Há muitos comentários e configurações nesse arquivo, incluindo o seguinte:

```
LoadModule ssl_module modules/mod_ssl.so
Listen 443
<VirtualHost _default_:443>
ErrorLog logs/ssl_error_log
TransferLog logs/ssl_access_log
LogLevel warn
SSLEngine on
SSLCertificateFile /etc/pki/tls/certs/localhost.crt
SSLCertificateKeyFile /etc/pki/tls/private/localhost.key
</VirtualHost>
```

Das configurações globais no arquivo `ssl.conf`, o módulo `mod_ssl.so` é carregado para fornecer recursos de SSL. O serviço SSL também é configurado para ouvir na porta SSL padrão 443.

É criado um bloco `VirtualHost` que faz com que mensagens de erro e mensagens de acesso sejam registradas em arquivos de log que são separados dos logs padrão usados pelo servidor (`ssl_error_log` e `ssl_access_log` no diretório `/var/log/httpd/`). O nível de mensagens de log é configurado como `warn` e o `SSLEngine` é ativado.

No código de exemplo anterior, as duas últimas linhas do bloco `VirtualHost` contêm a chave e as informações do certificado. Como mencionado anteriormente, uma chave é gerada quando `mod_ssl` é instalado e colocado no arquivo `/etc/pki/tls/certs/localhost.crt`. Um certificado autoassinado, `/etc/pki/tls/certs/localhost.crt`, é criado usando essa chave. Quando você cria sua própria chave e certificado mais tarde, você terá de substituir os valores de `SSLCertificateFile` e `SSLCertificateKeyFile` nesse arquivo.

Depois de instalar o pacote `mod_ssl` e recarregar o arquivo de configuração, você pode testar se o certificado padrão está funcionando, seguindo estes passos:

1. Abra uma conexão com o site de um navegador web usando o protocolo HTTPS. Por

exemplo, se você estiver usando o Firefox no sistema onde o servidor web está sendo executado, digite `https://localhost` na caixa de localização e pressione Enter. A Figura 17.2 mostra um exemplo da página que aparece.

ando um site SSL com um certificado padrão

2. A página mostrada na Figura 17.2 avisa que não há nenhuma maneira de verificar quem criou o certificado que você está aceitando. Como você está acessando o site por meio de um navegador no host local, você pode clicar em Add Exception para permitir conexões a esse site. Você é avisado de que está instruindo o Firefox a aceitar esse certificado.
3. Selecione View para ver o certificado que foi gerado. Ele incluirá seu hostname, informações sobre quando foi emitido e quando expira o certificado e muitas outras informações sobre a organização.
4. Feche a janela e selecione Confirm Security Exception para aceitar a conexão. Agora você deve ver sua página web padrão usando o protocolo HTTPS. A partir de agora, seu navegador irá aceitar conexões HTTPS para o servidor web usando esse certificado e criptografar todas as comunicações entre o servidor e o navegador.

Como você não quer que seu site assuste os usuários, a melhor coisa a fazer é obter um certificado válido para usar com ele. A próxima melhor coisa a fazer é criar um certificado autoassinado que pelo menos inclua melhores informações sobre seu site e sua organização. A próxima seção descreve como fazer isso.

Gerando uma chave SSL e um certificado autoassinado

Para começar a configurar o SSL, use o comando `openssl`, que é parte do pacote `openssl`, para gerar seu par de chaves pública e privada. Depois disso, você pode gerar seu próprio certificado autoassinado para testar o site ou para usar internamente.

1. Se o pacote `openssl` não estiver instalado, instale-o da seguinte forma:

```
# yum install openssl
```

2. Gere uma chave privada RSA de 1024 bits e salve-a em um arquivo:

```
# cd /etc/pki/tls/private
# openssl genrsa -out server.key 1024
# chmod 600 server.key
```

ta

pode usar um nome de arquivo diferente de `server.key` e deve fazer isso se pretende ter mais de um host em sua máquina (o que exige mais de um endereço IP). Apenas certifique-se de especificar o nome correto na configuração do Apache depois.

Em ambientes de maior segurança, é uma boa ideia criptografar a chave adicionando o

argumento `-des3` depois do argumento `genrsa` na linha de comando `openssl`. Quando solicitado a informar uma senha, pressione Enter:

```
# openssl genrsa -des3 -out server.key 1024
```

3. Se você não pretende ter seu certificado assinado ou se quiser testar sua configuração, gere um certificado autoassinado e salve-o em um arquivo chamado `server.crt` no arquivo `/etc/pki/tls/certs`:

```
# cd /etc/pki/tls/certs
# openssl req -new -x509 -nodes -sha1 -days 365 \
-key /etc/pki/tls/private/server.key \
-out server.crt
```

```
Country Name (2 letter code) [AU]: US
State or Province Name (full name) [Some-State]: NJ
Locality Name (eg, city) []: Princeton
Organization Name (eg, company) [Internet Widgits Pty
Ltd]:TEST USE ONLY
Organizational Unit Name (eg, section) []:TEST USE ONLY
Common Name (eg, YOUR name) []:secure.example.org
Email Address []:dom@example.org
```

4. Edite o arquivo `/etc/httpd/conf.d/ssl.conf` a fim de alterar a localização da chave e do certificado para usar os que você acabou de criar. Por exemplo:

```
SSLCertificateFile /etc/pki/tls/certs/server.crt
SSLCertificateKeyFile /etc/pki/tls/private/server.key
```

5. Reinicie ou recarregue o servidor `httpd`.
6. Abra `https://localhost` a partir de um navegador local novamente e repita o procedimento para examinar e aceitar o novo certificado.

Para uso interno ou de teste, um certificado autoassinado pode funcionar para você. Mas para sites públicos, você deve usar um certificado que é validado por uma autoridade certificadora (AC). O procedimento para fazer isso é mostrado a seguir.

Gerando uma solicitação de assinatura de certificado (Certificate Signing Request - CSR)

Se você pretende ter seu certificado assinado por uma CA (incluindo uma que você executa por conta própria), você pode usar sua chave privada para gerar uma solicitação de assinatura de certificado (CSR):

1. Crie um diretório para armazenar o CSR.

```
# mkdir /etc/pki/tls/ssl.csr
# cd /etc/pki/tls/ssl.csr/
```

2. Use o comando `openssl` para gerar o CSR. O resultado é um arquivo CSR no diretório atual chamado `server.csr`. Quando você inserir a informação, o nome comum deve

corresponder ao nome que os clientes usarão para acessar o servidor. Certifique-se de configurar os outros detalhes corretamente para poder ser validado por uma CA independente. Além disso, se tiver inserido uma senha para sua chave, você será solicitado a inseri-la aqui para usar essa chave.

```
# openssl req -new -key ../private/server.key -out  
server.csr
```

```
Country Name (2 letter code) [AU]:US  
State or Province Name (full name) [Some-State]:Washington  
Locality Name (eg, city) []:Bellingham  
Organization Name (eg, company) [Internet Widgits Pty  
Ltd]:Example Company, LTD.  
Organizational Unit Name (eg, section) []:Network  
Operations  
Common Name (eg, YOUR name) []:secure.example.org  
Email Address []:dom@example.org
```

```
Please enter the following 'extra' attributes  
to be sent with your certificate request  
A challenge password []:  
An optional company name []:
```

3. Visite o site da autoridade assinante do certificado que você escolher e solicite um certificado assinado. Em algum momento, o site da CA provavelmente pedirá que você copie e cole o conteúdo de seu CSR em um formulário necessário para fazer a solicitação.
4. A CA irá enviar-lhe o certificado, provavelmente via e-mail. Salve-o no diretório `/etc/pki/tls/certs/` usando um nome baseado no site que você está hospedando — por exemplo, `example.org.crt`.
5. Altere o valor de `SSLCertificateFile` no arquivo `/etc/httpd/conf.d/ssl.conf` para apontar para o novo arquivo CRT. Ou, se você tiver vários hosts SSL, pode querer criar uma entrada separada (possivelmente em um arquivo `.conf` separado) parecida com o seguinte:

```
Listen 192.168.0.56:443  
<VirtualHost *:443>  
    ServerName      secure.example.org  
    ServerAlias     web.example.org  
    DocumentRoot    /home/username/public_html/  
    DirectoryIndex  index.php index.html index.htm  
    SSLEngine       On  
    SSLCertificateKeyFile /etc/pki/tls/private/server.  
key  
    SSLCertificateFile /etc/pki/tls/certs/example.org.  
crt  
</VirtualHost>
```

O endereço IP mostrado na diretiva `Listen` deve ser substituído pelo endereço IP público que representa o host SSL que você está servindo. Lembre-se de que cada host SSL deve ter seu próprio endereço IP.

Solucionado problemas do servidor web

Em qualquer ambiente complexo, você ocasionalmente tem problemas. Esta seção inclui dicas para isolar e resolver os erros mais comuns que você pode encontrar.

Verificando erros de configuração

Ocasionalmente, você pode se deparar com erros de configuração ou problemas de script que impedem o Apache de iniciar ou impedem que arquivos específicos sejam acessíveis. A maioria desses problemas pode ser isolada e resolvida por meio de duas ferramentas fornecidas pelo Apache: o programa `apachectl` e o log de erro do sistema.

Ao encontrar um problema, primeiro utilize o programa `apachectl` com o parâmetro `configtest` para testar a configuração. De fato, é uma boa ideia desenvolver o hábito de executar esse programa sempre que você fizer uma mudança de configuração:

```
# apachectl configtest
Syntax OK
# apachectl graceful
/usr/sbin/apachectl graceful: httpd gracefully restarted
```

No caso de um erro de sintaxe, `apachectl` indica onde o erro ocorre e também faz o melhor que ele pode para dar uma pista sobre a natureza do problema. Você pode, então, usar a opção de reinicialização `graceful` (`apachectl graceful`) para instruir o Apache a recarregar sua configuração sem desconectar todos os clientes ativos.

ta
ção de reinicialização `graceful` no `apachectl` testa automaticamente a configuração antes de enviar o de recarregar para o `apache`, mas ainda é uma boa ideia adquirir o hábito de executar o teste de configuração usual depois de fazer qualquer mudança de configuração.

Alguns problemas de configuração passam pelos testes de sintaxe realizados por `apachectl` mas fazem com que o servidor HTTP se feche imediatamente depois de recarregar sua configuração. Se isso acontecer, use o comando `tail` para verificar o log de erro do Apache a fim de obter informações úteis. Em sistemas Fedora e RHEL, o log de erro está em `/var/log/httpd/error.log`. Em outros sistemas, você pode encontrar a localização olhando para a diretiva `ErrorLog` na configuração do Apache.

Você pode encontrar uma mensagem de erro parecida com esta:

```
[crit] (98)Address already in use: make_sock: could not bind to port 80
```

Esse erro geralmente indica que algo mais está ligado à porta 80 (o que não é muito comum, a menos que você tenha tentado instalar outro servidor web), que outro processo Apache já está em execução

(`apachectl` geralmente flagra isso) ou que você instruiu o Apache a ligar a mesma combinação de endereço IP e porta em mais de um lugar. Meu amigo Richard disse que viu estudantes que instalaram o Skype no Linux de uma maneira que fez com que o Skype usasse a porta TCP 80, quando executado em segundo plano.

Você pode usar o comando `netstat` para ver a lista de programas (incluindo o Apache) com as portas TCP no estado `LISTEN`:

```
# netstat -nltp
Active Internet connections (only servers)
Proto Local Address Foreign Address State PID/Program name
tcp 0.0.0.0:80 0.0.0.0:* LISTEN 2105/httpd
```

A saída de `netstat` (que foi reduzida para caber aqui) indica que uma instância do processo `httpd` com um ID de processo de 2105 está ouvindo (como indicado pelo estado `LISTEN`) conexões para qualquer endereço IP local (indicado por `0.0.0.0`) na porta 80 (a porta HTTP padrão). Se um programa diferente estiver ouvindo a porta 80, ele é mostrado aí. Você pode usar o comando `kill` para encerrar o processo, mas se for algo diferente de `httpd`, você também deve descobrir por que ele está sendo executado.

Se você não vir nenhum outro processo ouvindo na porta 80, pode ser que você, acidentalmente, instruiu o Apache a ouvir a mesma combinação de endereço IP e porta em mais de um lugar. Três diretivas de configuração podem ser usadas para isso: `BindAddress`, `Port` e `Listen`:

- `BindAddress` — Permite que você especifique um único endereço IP para ouvir ou você pode especificar todos os endereços IP usando um curinga `*`. Você nunca deve ter mais de uma instrução `BindAddress` em seu arquivo de configuração.
- `Port` — Especifica qual porta TCP ouvir, mas não permite que você especifique o endereço IP. `Port` geralmente não é usado mais de uma vez na configuração.
- `Listen` — Permite especificar um endereço IP e uma porta à qual ligá-lo. O endereço IP pode ser na forma de um curinga e você pode ter várias instruções `Listen` em seu arquivo de configuração.

Para evitar confusão, geralmente é uma boa ideia usar apenas um desses tipos de diretiva. Dos três, `Listen` é o mais flexível, por isso é provavelmente o que você mais vai querer usar. Um erro comum quando se utiliza `Listen` é especificar uma porta em todos os endereços IP (`*:80`), bem como a mesma porta de um endereço IP específico (`1.2.3.4:80`), o que resulta no erro a partir de `make_sock`.

Erros de configuração relacionados com SSL comumente resultam na inicialização incorreta do Apache. Certifique-se de que todos os arquivos de chave e certificado existem e de que eles estão no formato adequado (use `openssl` para examiná-los).

Para outras mensagens de erro, tente fazer uma pesquisa na web para ver se alguém encontrou o problema. Na maioria dos casos, você pode encontrar uma solução dentro das primeiras correspondências.

Se não estiver recebendo informações suficientes no `ErrorLog`, você pode configurá-lo para registrar

em log mais informações usando a diretiva `LogLevel`. As opções disponíveis para essa diretiva, em ordem crescente de detalhamento, são `emerg`, `alert`, `crit`, `error`, `warn`, `notice`, `info` e `debug`. Selecione apenas uma delas. Qualquer mensagem que é pelo menos tão importante quanto o `LogLevel` selecionado será armazenada no `ErrorLog`. Em um servidor típico, `LogLevel` está configurado como `warn`. Você não deve configurá-lo para qualquer valor menor do que `crit` e deve evitar deixá-lo configurado como `debug`, porque isso pode tornar o servidor lento e resultar em um `ErrorLog` muito grande.

Como último recurso, você também pode tentar executar `httpd -X -e debug` manualmente para verificar se há falhas ou outras mensagens de erro. O `-X` executa o `httpd` em primeiro plano, enquanto `-e debug` exibe mensagens de nível de depuração e superiores na tela.

Erros de acesso proibido e erros internos do servidor

Os dois tipos mais comuns de erros que você pode encontrar ao tentar exibir páginas específicas em seu servidor são erros de permissão e erros internos do servidor. Ambos os tipos de erros geralmente podem ser isolados usando as informações do log de erros. Depois de fazer qualquer uma das alterações descritas na lista a seguir para tentar resolver um desses problemas, tente a solicitação novamente e verifique o log de erro para ver se a mensagem foi alterada (por exemplo, para mostrar que a operação foi concluída com sucesso).

Erros do tipo “File not found” (“Arquivo não encontrado”) podem ser verificados da mesma maneira que “access denied” (“acesso proibido”) e “server internal errors” (“erros internos do servidor”). Às vezes você pode achar que o `apache` não está olhando para onde você pensa que ele está, no caso de um arquivo específico. Geralmente, o caminho inteiro do arquivo aparece no log de erros. Certifique-se de que você está acessando o host virtual correto e que qualquer configuração de `Alias` que possa estar direcionando sua localização para um lugar que você não esperava.

- **Permissões de arquivo** — O erro “File permissions prevent access” (“Permissões de arquivo impedem o acesso”) indica que o processo `apache` está executando como um usuário que não é capaz de abrir o arquivo solicitado. Por padrão, `httpd` é executado pelo usuário `apache` e pelo grupo `apache`. Certifique-se de que a conta tem permissões de execução no diretório e todos os diretórios acima dele, bem como permissões de leitura sobre os próprios arquivos. Permissões de leitura em um diretório também são necessárias se você quiser que o Apache gere um índice de arquivos. Veja a página `man` para `chmod` para mais informações sobre como visualizar e alterar as permissões.

Permissões de leitura não são necessárias para binários compilados, tais como aqueles escritos em C ou C++, mas devem ser adicionadas com segurança a não ser que haja a necessidade de manter o conteúdo do programa em

- **Acesso negado** — Um erro “Client denied by server configuration” (“Cliente negado pela configuração do servidor”) indica que o Apache foi configurado para negar o acesso ao objeto. Verifique nos arquivos de configuração as seções `Location` e `Directory` que podem afetar o arquivo que você está tentando acessar. Lembre-se de que as configurações aplicadas a um caminho também são aplicadas a todos os caminhos abaixo dele. Você pode substituir isso alterando as permissões apenas para o caminho mais específico para o qual você deseja permitir acesso.
- **Índice não encontrado** — O erro “Directory index forbidden by rule” (“Índice de diretório proibido por regra”) indica que o Apache não pôde encontrar um arquivo de índice com um nome especificado na diretiva `DirectoryIndex` e foi configurado para não criar um índice contendo uma lista de arquivos em um diretório. Certifique-se de que sua página de índice, se você tiver uma, tem um dos nomes especificados na diretiva `DirectoryIndex` relevante ou adicione uma linha `Options Indexes` à seção `Directory` ou `Location` apropriada para esse objeto.
- **Falha de script** — Erros do tipo “Premature end of script headers” (“Fim prematuro de cabeçalhos de script”) podem indicar que um script está falhando antes de terminar. Ocasionalmente, erros que causam isso também aparecem no log de erros. Ao utilizar `suexec` ou `suPHP`, esse erro também pode ser causado por um erro de posse ou permissões de arquivo. Esses erros aparecem nos arquivos de log em `/var/log/httpd`.
- **Erros do SELinux** — Se as permissões de arquivo estiverem abertas, mas as mensagens negando permissão aparecem em arquivos de log, o SELinux poderia estar causando o problema. Desative o SELinux temporariamente (`setenforce 0`) e tente acessar o arquivo novamente. Se agora o arquivo estiver acessível, reative o SELinux (`setenforce 1`) e verifique contextos de arquivos e valores booleanos. Contextos de arquivos devem estar corretos para o `httpd` para ser capaz de acessar um arquivo. Um booleano pode impedir que um arquivo seja servido a partir de um diretório montado remotamente ou impedir que uma página envie um e-mail ou faça upload de um arquivo. Digite `man httpd_selinux` para obter detalhes sobre as definições de configuração do SELinux relacionadas com os serviços `httpd`.

Resumo

O projeto de código-fonte aberto Apache é o servidor web mais popular. Embora o Apache ofereça uma enorme flexibilidade, segurança e complexidade, um servidor web Apache básico pode ser configurado em poucos minutos no Fedora, no RHEL e na maioria das outras distribuições Linux.

Este capítulo descreve os passos para instalar, configurar, proteger e solucionar problemas de um servidor web Apache básico. Você aprendeu a configurar hospedagem virtual e proteger hosts SSL. Você também aprendeu a configurar o Apache para permitir que qualquer conta de usuário no sistema publique conteúdo a partir de seu próprio diretório `public_html`.

Continuando o tema da configuração do servidor, no Capítulo 18, você aprenderá a configurar um servidor de FTP no Linux. Os exemplos ilustram como configurar um servidor de FTP usando o pacote `vsftpd`.

Exercícios

Os exercícios nesta seção abordam temas relacionados à instalação e configuração de um servidor web Apache. Como de costume, recomendo que você use um sistema Fedora ou o Red Hat Enterprise Linux de reserva para fazer os exercícios. Não faça esses exercícios em uma máquina de produção, porque eles modificam os arquivos de configuração e serviços do Apache e podem danificar serviços que estão atualmente configurados. Tente encontrar um computador em que a interrupção dos serviços no sistema não causará nenhum dano.

Esses exercícios supõem que você está começando com uma instalação do Fedora ou do RHEL em que o servidor Apache (pacote `httpd`) ainda não está instalado.

Se você empacar, as soluções para as tarefas são mostrados no Apêndice B. Essas soluções mostram uma abordagem para cada tarefa, embora no Linux, muitas vezes, haja várias maneiras de completar uma tarefa.

1. A partir de um sistema Fedora, instale todos os pacotes associados com o grupo Web Server.
2. Crie um arquivo chamado `index.html` no diretório designado para `DocumentRoot` no arquivo principal de configuração do Apache. O arquivo deve ter as palavras “My Own Web Server” (“Meu Próprio Servidor Web”).
3. Inicie o servidor web Apache e configure-o para iniciar automaticamente no momento da inicialização. Verifique se ele está disponível a partir de um navegador web no host local. (Você deve ver as palavras “My Own Web Server” exibidas se ele estiver funcionando corretamente.)
4. Use o comando `netstat` para ver as portas que o servidor `httpd` está ouvindo.
5. Tente se conectar ao seu servidor web Apache a partir de um navegador que está fora do sistema local. Se ele falhar, corrija quaisquer problemas que encontrar investigando o firewall, o SELinux e outros recursos de segurança.
6. Usando o comando `openssl` ou similar, crie sua própria chave RSA privada e seu próprio certificado SSL autoassinado.
7. Configure o servidor web Apache para usar sua chave e certificado autoassinado para servir conteúdo seguro (HTTPS).
8. Use um navegador web para criar uma conexão HTTPS com o servidor web e ver o conteúdo do certificado que você criou.
9. Crie um arquivo chamado `/etc/httpd/conf.d/example.org.conf` que se transforma em hospedagem virtual baseada em nome e cria uma máquina virtual que:

- Ouve a porta 80 em todas as placas de rede
- Tem um administrador de servidor chamado `joe.example.org`
- Tem um nome de servidor de `joe.example.org`
- Tem um `DocumentRoot` de `/var/www/html/example.org`
- Tem um `DirectoryIndex` que inclui pelo menos `index.html`

Crie um arquivo `index.html` em `DocumentRoot` que contém as palavras “Welcome to the House of Joe” (“Bem-Vindo à Casa do Joe”).

10. Adicione o texto **`joe.example.org`** ao final da entrada `localhost` no arquivo `/etc/hosts` na máquina que está executando o servidor web. Então, digite **`http://joe.example.org`** na caixa de localização de seu navegador. Você deverá ver “Welcome to the House of Joe” quando a página for exibida.

Configurando um servidor FTP

NESTE CAPÍTULO

Aprendendo como o FTP funciona

Instalando um servidor vsftpd

Escolhendo as configurações de segurança para o vsftpd

Criando arquivos de configuração do vsftpd

Executando clientes FTP

O File Transfer Protocol (FTP) é um dos mais antigos protocolos existentes para compartilhar arquivos através de redes. Embora existam protocolos mais seguros para o compartilhamento de arquivos de rede, o FTP ainda costuma ser usado para disponibilizar arquivos gratuitamente na internet.

Vários servidores FTP estão disponíveis no Linux hoje em dia. Mas o usado por padrão com o Fedora, o Red Hat Enterprise Linux e outras distribuições Linux é o Very Secure FTP Daemon (pacote `vsftpd`). Este capítulo descreve como instalar, configurar, usar e proteger um servidor FTP usando o pacote `vsftpd`.

Entendendo o FTP

O FTP opera em um modelo cliente/servidor. Um daemon de servidor FTP ouve solicitações de entrada (na porta TCP 21) de clientes FTP. O cliente apresenta um login e uma senha. Se o servidor aceitar a informação de login, o cliente pode percorrer o sistema de arquivos de forma interativa, listar arquivos e diretórios e depois baixar (e, por vezes, subir) arquivos.

O que torna o FTP inseguro é que tudo enviado entre o cliente e ele é feito em texto claro. O protocolo FTP foi criado em uma época em que a maioria das comunicações entre computadores era feita por linhas dedicadas ou por ligações discadas (dial-up), nas quais a criptografia não era considerada algo crítico. Se você usar o FTP por meio de uma rede pública, alguém espionando a linha em qualquer lugar entre o cliente e o servidor seria capaz de ver não apenas os dados que estão sendo transferidos, mas também o

processo de autenticação (login e senha).

Assim, o FTP não era bom para compartilhar arquivos privados (use comandos SSH, como `sftp`, `scp` ou `rsync`, se você precisar fazer transferências de arquivos privados e criptografados). Mas se você estiver compartilhando documentos públicos, repositórios de software de código-fonte aberto ou outros dados abertamente disponíveis, o FTP é uma boa escolha. Independentemente do sistema operacional que as pessoas usam, elas certamente terão um aplicativo de transferência de arquivos FTP disponível para obter os arquivos que você oferece a partir do servidor FTP.

Quando os usuários autenticam um servidor FTP em Linux, seus nomes de usuário e senhas são autenticados com as contas padrão do Linux. Também há uma conta especial não autenticada utilizada pelo servidor FTP chamada `anonymous`. A conta `anonymous` pode ser acessada por qualquer pessoa, pois não requer uma senha válida. Na verdade, o termo servidor FTP anônimo é frequentemente usado para descrever um servidor FTP público que não requer (ou nem mesmo permite) autenticação de uma conta de usuário legítimo.

ta

ora a capacidade de fazer login no servidor `vsftpd` usando uma conta de usuário regular Linux esteja ativada padrão no Fedora e no Red Hat Enterprise Linux, se o SELinux estiver configurado no modo Enforcing, ele impedirá uploads e transferências de arquivos. Se quiser manter o SELinux no modo Enforcing, mas ainda permitir que faça logins de usuários regulares, há um booleano que você pode mudar (ver o “Configurando a seção SELinux para seu servidor FTP”) para permitir logins de usuários regulares.

Após a fase de autenticação (na porta de controle, a porta TCP 21), uma segunda conexão é feita entre o cliente e o servidor. O FTP suporta os tipos de conexão ativa e passiva. Com uma conexão de FTP ativa, o servidor envia dados de sua porta TCP 20 para uma porta aleatória que o servidor escolhe acima da porta 1023 no cliente. Com FTP passivo, é o cliente quem solicita a conexão passiva e solicita uma porta aleatória a partir do servidor.

Muitos navegadores suportam o modo FTP passivo, de modo que, se houver um firewall no cliente, ele não bloqueará a porta de dados que o servidor FTP poderia usar no modo ativo. Apoiar modo passivo requer algum trabalho extra no firewall do servidor para permitir conexões aleatórias em portas acima de 1023 no servidor. A seção “Abrindo seu firewall para FTP” mais adiante, neste capítulo, descreve o que você precisa fazer para seu firewall Linux fazer tanto conexões FTP passivas como ativas funcionarem.

Uma vez estabelecida a conexão entre o cliente e o servidor, o diretório atual do cliente também é estabelecido. Para o usuário anônimo, o diretório `/var/ftp` é o diretório inicial do usuário. O usuário anônimo não pode sair da estrutura de diretórios `/var/ftp`. Se um usuário regular, digamos `joe`, conectou-se ao servidor FTP, `/home/joe` seria o diretório atual de `joe`, mas `joe` seria capaz de ir para qualquer parte do sistema de arquivos para o qual ele tivesse permissão.

Clientes FTP que usam linha de comando (como os comandos `lftp` e `ftp`) entram em um modo interativo depois de se conectarem com o servidor. A partir do prompt que você vê, é possível executar muitos comandos que são semelhantes àqueles que você usaria a partir do shell. Você pode usar `pwd` para ver seu atual diretório, `ls` para listar o conteúdo do diretório e `cd` para mudar de diretório. Depois

de encontrar o arquivo desejado, você pode usar os comandos `get` e `put` para baixar ou subir arquivos para o servidor, respectivamente.

Com ferramentas gráficas para acessar servidores FTP (como um navegador), você digita a URL do site que deseja visitar (como `ftp://docs.example.com`) na caixa de localização do navegador. Se você não adicionar nenhum nome de usuário ou senha, uma conexão anônima é feita e o conteúdo do diretório inicial do site é exibido. Clique nos links para diretórios a fim de mudar para esses diretórios. Clique nos links para arquivos a fim de exibir ou baixar os arquivos para o sistema local.

Munido de algum conhecimento de como o FTP funciona, agora você está pronto para instalar um servidor FTP (pacote `vsftpd`) em seu sistema Linux.

Instalando o servidor FTP vsftpd

Configurar o servidor Very Secure FTP requer apenas um pacote no Fedora, RHEL e outras distribuições Linux: `vsftpd`. Supondo que você tenha uma conexão com o repositório de software, para instalar `vsftpd` basta digitar o seguinte como root:

```
# yum install vsftpd
```

Depois que o pacote `vsftpd` está instalado, eis alguns comandos que você pode executar para se familiarizar com o conteúdo do pacote. Execute esse comando para obter algumas informações gerais sobre o pacote:

```
# rpm -qi vsftpd
...
Packager           : Fedora Project
Vendor            : Fedora Project
URL               : https://security.appspot.com/vsftpd.html
Summary           : Very Secure Ftp Daemon
Description       : vsftpd is a Very Secure FTP daemon. It was
                  : written
                  : completely from scratch.
```

Se quiser obter mais informações sobre `vsftpd`, siga a URL listada para o site relacionado (<https://security.appspot.com/vsftpd.html>). Você pode obter a documentação e informações adicionais sobre as últimas revisões do `vsftpd`.

Você pode ver todo o conteúdo do pacote `vsftpd` (`rpm -ql vsftpd`), ou apenas a documentação (`-qd`), ou apenas os arquivos de configuração (`-qc`). Para ver os arquivos de documentação do pacote `vsftpd`, use o seguinte:

```
# rpm -qd vsftpd
```

```
/usr/share/doc/vsftpd-2.3.4/EXAMPLE/INTERNET_SITE/README
...
/usr/share/doc/vsftpd-2.3.4/EXAMPLE/PER_IP_CONFIG/README
...
/usr/share/doc/vsftpd-2.3.4/EXAMPLE/VIRTUAL_HOSTS/README
/usr/share/doc/vsftpd-2.3.4/EXAMPLE/VIRTUAL_USERS/README
...
/usr/share/doc/vsftpd-2.3.4/FAQ
...
/usr/share/doc/vsftpd-2.3.4/vsftpd.xinetd
/usr/share/man/man5/vsftpd.conf.5.gz
/usr/share/man/man8/vsftpd.8.gz
```

Na estrutura de diretórios `/usr/share/doc/vsftpd-*/EXAMPLE` há arquivos de configuração inclusos para ajudar a configurar o `vsftpd` de maneiras que são apropriadas para um site da internet, sites com múltiplos endereços IP e hosts virtuais. O diretório `/usr/share/doc/vsftpd-*` principal contém um FAQ (perguntas mais frequentes), dicas de instalação e informações sobre a versão.

As páginas `man` podem ter informações mais úteis quando você começar a configurar o servidor `vsftpd`. Digite **`man vsftpd.conf`** para ler sobre o arquivo de configuração e **`man vsftpd`** para ler sobre o processo `daemon`.

Para listar os arquivos de configuração, digite o seguinte:

```
# rpm -qc vsftpd
/etc/logrotate.d/vsftpd
/etc/pam.d/vsftpd
/etc/vsftpd/ftpusers
/etc/vsftpd/user_list
/etc/vsftpd/vsftpd.conf
```

O principal arquivo de configuração é `/etc/vsftpd/vsftpd.conf`. Os arquivos `ftpusers` e `user_list` no mesmo diretório armazenam informações sobre contas de usuários que têm restrições de acesso ao servidor. O arquivo `/etc/pam.d/vsftpd` define como a autenticação é feita no servidor FTP. O arquivo `/etc/logrotate.d/vsftpd` configura como os arquivos de log são reciclados ao longo do tempo.

Agora você instalou o `vsftpd` e deu uma rápida olhada em seu conteúdo. O próximo passo é iniciar e testar o serviço `vsftpd`.

Iniciando o serviço vsftpd

Nenhuma configuração é necessária para carregar o serviço `vsftpd` se você quer apenas usar as configurações padrão. Se você iniciar o `vsftpd` como ele é entregue com o Fedora, isso é o que você

tem:

- O serviço `vsftpd` inicia o daemon `vsftpd`, que roda em segundo plano.
- A porta padrão em que o servidor `vsftpd` ouve é a porta TCP 21. Por padrão, os dados são transferidos para o usuário, depois que a conexão é feita, na porta TCP 20. A porta TCP 21 deve ser aberta no firewall para permitir que novas conexões acessem o serviço. (Consulte a seção “Protegendo seu servidor FTP” para obter detalhes sobre abertura de portas, sobre como possibilitar o acompanhamento de conexões necessárias para FTP passivo e sobre definição de outras regras de firewall relacionadas ao FTP.)
- O daemon `vsftpd` lê `vsftpd.conf` para determinar os recursos que o serviço permite.
- Contas de usuário Linux (excluindo usuários administrativos) e a conta de usuário anônimo (não requer senha) podem acessar o servidor FTP. (Se o SELinux estiver no modo Enforcing, você precisa definir um valor booleano para permitir aos usuários regulares fazer login no servidor FTP. Consulte a seção “Protegendo seu servidor FTP” para obter detalhes.)
- O usuário anônimo tem acesso apenas ao diretório `/var/ftp` e seus subdiretórios. Um usuário comum começa com seu diretório inicial como o diretório atual, mas pode acessar qualquer diretório a que ele tenha permissão de acesso por meio de uma sessão de login normal ou SSH. Listas de usuários nos arquivos `/etc/vsftpd/user_list` e `/etc/vsftpd/ftpusers` definem alguns usuários administrativos e especiais que não têm acesso ao servidor FTP (`root`, `bin`, `daemon` e outros).
- O usuário anônimo pode baixar os arquivos do servidor, mas não enviá-los. Um usuário comum pode fazer upload ou download de arquivos, com base em permissões regulares do Linux.
- Mensagens de log detalhando o upload ou download de arquivos são gravadas no arquivo `/var/log/xferlogs`. Essas mensagens de log são armazenadas em um formato `xferlog` padrão.

Se você estiver pronto para iniciar o servidor usando os padrões acima descritos, os exemplos a seguir mostram como fazer isso. Se primeiro você quiser alterar algumas configurações, vá para a seção “Configurando o servidor FTP,” finalize suas configurações e depois volte aqui para instruções sobre como ativar e iniciar seu servidor.

Antes de iniciar o serviço `vsftpd`, você pode verificar se ele já está ou não funcionando. No Fedora, você faria o seguinte:

```
# systemctl status vsftpd.service
vsftpd.service - Vsftpd ftp daemon
   Loaded: loaded (/lib/systemd/system/vsftpd.service; disabled)
   Active: inactive (dead)
   CGroup: name=systemd:/system/vsftpd.service
```

No Red Hat Enterprise Linux 6, você precisa de dois comandos para ver a mesma informação:

```
# service vsftpd status
vsftpd is stopped
# chkconfig --list vsftpd
vsftpd          0:off  1:off  2:off  3:off  4:off  5:off  6:off
```

Em ambos os exemplos acima, do Fedora e do RHEL, os comandos `service` `chkconfig` mostram o estado `systemctl` como parado. Você também pode ver que ele está desabilitado no Fedora e desativado em cada nível de execução para o RHEL. Desativado (`off`) significa que o serviço não será automaticamente iniciado quando seu sistema inicializar.

Para iniciar e ativar o `vsftpd` no Fedora (e então verificar o status), digite o seguinte:

```
# systemctl start vsftpd.service
# systemctl enable vsftpd.service
ln -s '/lib/systemd/system/vsftpd.service'
  '/etc/systemd/system/multi-user.target.wants/vsftpd.service'
# systemctl status vsftpd.service
vsftpd.service - Vsftpd ftp daemon
Loaded: loaded (/lib/systemd/system/vsftpd.service; enabled)
Active: active (running) since Thu, 10 May 2012 23:22:08 -0400;
 11s ago
Main PID: 29787 (vsftpd)
CGroup: name=systemd:/system/vsftpd.service
        29787 /usr/sbin/vsftpd /etc/vsftpd/vsftpd.conf
```

No Red Hat Linux, inicie e ative (habilite) o `vsftpd` (e então verifique o status), como segue:

```
# service vsftpd start
Starting vsftpd for vsftpd:                [ OK ]
# chkconfig vsftpd on ; chkconfig --list vsftpd
vsftpd          0:off  1:off  2:on   3:on   4:on   5:on   6:off
```

Agora, em qualquer sistema, você pode verificar se o serviço está sendo executado com o comando `netstat`:

```
# netstat -tupln | grep vsftpd
tcp        0      0 0.0.0.0:21          0.0.0.0:*        LISTEN    29787/vsftpd
```

A partir da saída de `netstat`, você pode ver que o processo `vsftpd` (ID de processo 29787) está ouvindo (`LISTEN`) todos os endereços IP para conexões de entrada na porta 21 (`0.0.0.0:21`) do protocolo TCP (`tcp`). Uma maneira rápida de verificar se `vsftpd` está funcionando é colocar um arquivo no diretório `/var/ftp` e tentar abri-lo a partir de seu navegador na máquina local:

```
# echo "Hello From Your New FTP Server" > /var/ftp/hello.txt
```

A partir de um navegador no sistema local, digite o seguinte na caixa de localização do Firefox ou outro navegador:

```
ftp://localhost/hello.txt
```

Se o texto `Hello From Your New FTP Server` (que significa: “Olá do Seu Novo servidor

FTP”) aparecer no navegador, o servidor `vsftpd` está funcionando. Então, tente novamente, substituindo `localhost` pelo endereço IP de seu host ou o nome totalmente qualificado do host, a partir de um navegador web em outro sistema. Se isso funcionar, o servidor `vsftpd` está acessível ao público. Caso contrário, o que é muito improvável, consulte a próxima seção, “Protegendo seu servidor FTP”. Essa seção explica como abrir firewalls e modificar outros recursos de segurança para permitir acesso e também forma proteger seu servidor FTP.

Protegendo seu servidor FTP

Embora seja fácil fazer um servidor FTP `vsftpd` funcionar, isso não significa que ele é plenamente acessível de imediato. Se você tem um firewall em seu sistema Linux, provavelmente ele está bloqueando o acesso a todos os serviços em seu sistema, exceto aqueles que você explicitamente permitiu.

Se decidir que a configuração padrão do `vsftpd` funciona para você, como descrito na seção anterior, você pode fazê-lo funcionar permitindo acesso adequado e fornecendo segurança para seu serviço `vsftpd`. Para proteger seu servidor `vsftpd`, as próximas seções descrevem como configurar o firewall (`iptables`), TCP wrappers (`hosts.allow` e `hosts.deny`) e SELinux (booleanos e contextos de arquivos).

Abrindo seu firewall para FTP

Se tiver um firewall implementado em seu sistema, você precisa adicionar regras de firewall que permitem solicitações de entrada para seu site FTP e permitir que os pacotes retornem ao seu sistema em conexões estabelecidas. No Fedora e no Red Hat Enterprise Linux, regras de firewall são armazenadas no arquivo `/etc/sysconfig/iptables` e o serviço se chama `iptables` (RHEL) ou `iptables.service` (Fedora). Os módulos são carregados no firewall a partir do arquivo `/etc/sysconfig/iptables-config`.

ta

o melhor trabalhar em seu firewall diretamente a partir de um console do sistema, se possível, em vez de a partir de um login remoto (como `ssh`), pois um pequeno erro pode bloquear você imediatamente, impedindo o acesso ao servidor. Depois disso, você deve ir para o console para voltar ao servidor e corrigir o problema.

Há algumas coisas que você precisa adicionar a seu firewall para permitir o acesso a seu servidor FTP sem abrir o acesso a outros serviços. Primeiro, você precisa permitir que o sistema aceite solicitações na porta TCP 21; depois, precisa ter certeza de que o módulo de rastreamento de conexão está carregado.

Se você estiver usando um firewall padrão, no começo as regras abrem o acesso a solicitações de todos os serviços provenientes do host local e permitem a entrada de pacotes associados ou relacionados com conexões estabelecidas. No meio, há regras que abrem portas para solicitações de serviços já permitidos, tais como o serviço Secure Shell (`sshd` na porta TCP 22). No final das regras, há geralmente uma regra final que descarta (DROP) ou rejeita (REJECT) qualquer solicitação que não foi

explicitamente permitida.

Para permitir o acesso público a alguém solicitando seu servidor FTP, você deve permitir novas solicitações na porta TCP 21. Você geralmente adiciona a regra em algum lugar antes da regra DROP ou REJECT. A seguinte saída mostra um conteúdo parcial do arquivo `/etc/sysconfig/iptables` com a regra que permite o acesso ao seu servidor FTP em negrito:

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 21 -j ACCEPT
...
-A INPUT -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

Esse exemplo mostra que, para a tabela filter, o firewall aceita pacotes de conexões estabelecidas, conexões de hosts locais e quaisquer novas solicitações na porta TCP 22 (serviço SSH). A linha que acabamos de adicionar (`--dport 21`) permite que qualquer pacotes em novas conexões com a porta TCP 21 sejam aceitos.

ta

importante ter a linha `ESTABLISHED, RELATED` nas regras de firewall de `iptables`. Sem essa linha, os usuários seriam capazes de se conectar a seus serviços de SSH (porta 22) e FTP (porta 21), mas não seriam capazes de comunicar depois disso. Assim, um usuário pode ser autenticado, mas não ser capaz de transferir dados.

A próxima coisa que você tem a fazer é instalar o módulo de rastreamento de conexão FTP para ser carregado cada vez que o firewall inicia. Edite essa linha no início do arquivo `/etc/sysconfig/iptables-config` para fazê-la parecer com o seguinte:

```
IPTABLES_MODULES="nf_conntrack_ftp"
```

Nesse ponto, você pode reiniciar seu firewall (tendo em mente que um erro pode bloqueá-lo se você estiver conectado remotamente). Use um dos seguintes comandos para reiniciar o `iptables`:

```
# service iptables restart
ou
# systemctl restart iptables.service
```

Tente novamente acessar o servidor FTP a partir de um sistema remoto (usando um navegador web ou

algum outro cliente de FTP).

Permitindo acesso FTP no TCP wrappers

O recurso TCP wrappers no Linux permite que você adicione informações aos arquivos `/etc/hosts.allow` e `/etc/hosts.deny` para indicar quem pode ou quem não pode acessar os serviços selecionados. Nem todos os serviços implementam TCP wrappers, mas `vsftpd` faz isso.

Por padrão, os arquivos `hosts.allow` e `hosts.deny` estão vazios, o que não impõe restrições sobre quem pode acessar serviços protegidos por TCP wrappers. Mas se você estiver bloqueando o acesso ao arquivo `hosts.deny` a todos os serviços que não tenham sido explicitamente permitidos (adicionando a linha `ALL:ALL` a `hosts.deny`), acrescentar uma linha como a seguinte ao início do arquivo `/etc/hosts.allow` permitirá acesso ao servidor `vsftpd`:

```
vsftpd:                                ALL :                                ALLOW
```

Para mais informações sobre como usar o TCP wrappers, consulte o Capítulo 25, “Protegendo o Linux em uma rede” ou a página `man hosts.allow` (digite `man hosts.allow`).

Configurando o SELinux para seu servidor FTP

Se o SELinux estiver configurado como `Permissive` ou `Disabled`, ele não irá bloquear o acesso ao serviço `vsftpd` de qualquer forma. Mas se o SELinux estiver no modo `Enforcing`, há algumas questões com o SELinux que poderiam fazer com que seu servidor `vsftpd` não se comportasse como você gostaria. Use o seguinte código para verificar o estado do SELinux em seu sistema:

```
# getenforce
Enforcing
# grep ^SELINUX= /etc/sysconfig/selinux
SELINUX=enforcing
```

O comando `getenforce` mostra como o SELinux está atualmente configurado (aqui, está no modo `Enforcing`). A variável `SELINUX=` em `/etc/sysconfig/selinux` mostra como o SELinux está configurado quando o sistema inicializa. Se ele estiver no modo `Enforcing`, como é o caso aqui, verifique a página `man ftpd_selinux` para obter informações sobre as configurações do SELinux que podem afetar o funcionamento de seu serviço `vsftpd`. Eis alguns exemplos de contextos de arquivo que devem ser configurados para o SELinux permitir que arquivos e diretórios sejam acessados por `vsftpd`:

- Para compartilhar conteúdo de modo que ele possa ser baixado para clientes FTP, esse conteúdo deve ser marcado com um contexto de arquivo `public_content_t`. Os arquivos criados no diretório `/var/ftp` ou seus subdiretórios herdam o contexto de arquivo `public_content_t` automaticamente. (Certifique-se de criar um novo conteúdo ou copiar conteúdo existente para os diretórios `/var/ftp`. Mover os arquivos para esses diretórios pode não mudar o contexto do arquivo corretamente.)

- Para permitir o upload de arquivos por usuários anônimos, o contexto de arquivo no diretório de destino deve ser configurado como `public_content_rw_t`. (Outras permissões, booleanos SELinux e configurações `vsftpd.conf` devem ser definidos para que isso funcione bem.)

Se tiver arquivos na estrutura de diretórios `/var/ftp` que têm os contextos de arquivo errados (o que pode acontecer se você mover arquivos para lá a partir de outros diretórios em vez de copiá-los), você pode alterar ou restaurar o contexto de arquivo sobre esses arquivos para que eles possam ser compartilhados. Por exemplo, para recursivamente alterar o contexto de arquivo do diretório `/var/ftp/pub/stuff` de modo que o conteúdo possa ser lido a partir do servidor FTP através do SELinux, digite o seguinte:

```
# semanage fcontext -a -t public_content_t
"/var/ftp/pub/stuff(/.*)?"
# restorecon -F -R -v /var/ftp/pub/stuff
```

Se quiser permitir que os usuários também gravem em um diretório, assim como o leiam, você precisa atribuir o contexto de arquivo `public_content_rw_t` ao diretório no qual você deseja permitir uploads. Esse exemplo instrui o SELinux a permitir o upload de arquivos no diretório `/var/ftp/pub/uploads`:

```
# semanage fcontext -a -t public_content_rw_t "/var/ftp/pub/
uploads(/.*)?"
# restorecon -F -R -v /var/ftp/pub/uploads
```

Recursos de servidor FTP que são considerados inseguros pelo SELinux têm booleanos que permitem que você habilite ou não esses recursos. Eis alguns exemplos:

- Para permitir que os usuários regulares sejam capazes de autenticar e ler e gravar arquivos e diretórios através do servidor FTP, o booleano `ftp_home_dir` deve estar ativo. Esse é um dos booleanos de FTP mais comuns para ativar (ele está desativado por padrão). Para ativá-lo permanentemente, digite o seguinte:

```
# setsebool -P ftp_home_dir on
```
- Para o SELinux permitir que usuários anônimos leiam e gravem arquivos e diretórios, você precisa ativar o booleano `allow_ftpd_anon_write`:

```
# setsebool -P allow_ftpd_anon_write on
```
- Para ser capaz de montar sistemas de arquivos NFS ou CIFS (Windows) compartilhados remotamente e compartilhá-los a partir de seu servidor `vsftpd`, você precisa ativar os dois seguintes valores booleanos, respectivamente:

```
# setsebool -P allow_ftpd_use_nfs on
# setsebool -P allow_ftpd_use_cifs on
```

Se alguma vez você achar que não pode acessar arquivos ou diretórios a partir de seu servidor FTP que você acredita que deveriam ser acessíveis, tente desativar o SELinux temporariamente:


```
# setenforce 0
```

Se você puder acessar os arquivos ou diretórios com o SELinux agora no modo Permissive, coloque o sistema de volta no modo Enforcing (`setenforce 1`). Agora, você sabe que tem de voltar às suas configurações do SELinux e descobrir o que está impedindo o acesso. (Veja o Capítulo 24, “Aumentando a Segurança Linux com SELinux” para mais informações sobre o SELinux).

Relacionando as permissões de arquivos Linux com o vsftpd

O servidor `vsftpd` depende de permissões de arquivos padrão do Linux para permitir ou negar acesso a arquivos e diretórios. Como seria de se esperar, para um usuário anônimo ver ou baixar um arquivo, pelo menos a permissão de leitura deve estar aberta para `other` (`-----r--`). Para acessar um diretório, pelo menos a permissão de execução deve estar ativada para `other` (`-----x`).

Para contas de usuários regulares, a regra geral é que, se um usuário pode acessar um arquivo a partir do shell, esse usuário pode acessar o mesmo arquivo de um servidor FTP. Assim, normalmente, os usuários regulares devem pelo menos ser capazes de baixar (`get`) e subir (`put`) arquivos de/para seus próprios diretórios iniciais, respectivamente.

Depois que as permissões e outras provisões de segurança estão definidas para seu servidor FTP, há outras configurações que você deve querer considerar para seu servidor FTP.

Configurando seu servidor FTP

A maioria das configurações do serviço `vsftpd` é feita no arquivo `/etc/vsftpd/vsftpd.conf`. Exemplos de `vsftpd.conf` para diferentes tipos de sites estão inclusos no diretório `/usr/share/doc/vsftpd-*`. Dependendo de como você quer usar seu site FTP, as próximas seções discutem algumas maneiras de configurar o servidor FTP.

Lembre-se de reiniciar o serviço `vsftpd` depois de alterar qualquer configuração.

Configurando o acesso do usuário

O servidor `vsftpd` vem com o usuário anônimo e todos os usuários do Linux (os locais listados no arquivo `/etc/passwd`) configurados para acessar o servidor. Isso se baseia nas seguintes configurações de `vsftpd.conf`:

```
anonymous_enable=YES
local_enable=YES
```

Como observado anteriormente, apesar da configuração `local_enable`, na verdade o SELinux vai impedir que usuários do `vsftpd` se autentiquem e transfiram dados. Tirar o SELinux do modo Enforcing ou definir o booleano correto permite que contas locais façam login e transfiram dados.

Há empresas de servidor web que permitem aos usuários usar FTP para carregar conteúdo que é usado nos próprios servidores dos usuários web. Em alguns casos, os usuários têm contas somente FTP, o que significa que eles não podem fazer login em um shell, mas podem fazer login via FTP para gerenciar seu conteúdo. Criar uma conta de usuário que não tem shell padrão (na verdade, `/sbin/nologin`) é a maneira como você pode impedir que um usuário faça login em um shell, mas ainda permitir acesso FTP. Por exemplo, a entrada em `/etc/passwd` para a conta somente FTP do usuário `bill` pode ser algo como o seguinte:

```
bill:x:1000:1000:Bill Jones:/home/chris:/sbin/nologin
```

Com a conta de usuário configurada com `/sbin/nologin` como o shell padrão, quaisquer tentativas de login a partir de um console ou via `ssh` como o usuário `bill` será negada. Mas enquanto `bill` tiver uma senha, e uma conta de acesso local ao servidor FTP estiver habilitada, `bill` deve ser capaz de logar-se no servidor FTP por meio de um cliente FTP.

Nem todo usuário com uma conta no sistema Linux tem acesso ao servidor FTP. A configuração `userlist_enable=YES` no arquivo `vsftpd.conf` diz para negar acesso ao servidor FTP a todas as contas listadas no arquivo `/etc/vsftpd/user_list`. Essa lista inclui os usuários administrativos `root`, `bin`, `daemon`, `adm`, `lp` e outros. Você pode adicionar outros usuários a essa lista aos quais você gostaria de negar acesso.

Se você alterar `userlist_enable` para `NO`, o arquivo `user_list` torna-se uma lista com apenas os usuários que têm acesso ao servidor. Em outras palavras, configurar `userlist_enable=NO`, remover todos os nomes de usuário do arquivo `user_list` e acrescentar os nomes de usuário `chris`, `joe` e `mary` a esse arquivo faz com que o servidor permita que somente esses três usuários façam login no servidor.

Independentemente de como o valor de `userlist_enable` está configurado, o arquivo `/etc/vsftpd/ftpusers` sempre inclui usuários que não têm acesso ao servidor. Assim como o arquivo `userlist_enable`, o arquivo `ftpusers` inclui uma lista de usuários administrativos. Você pode adicionar mais usuários a esse arquivo se quiser que o acesso FTP lhes seja negado.

Uma forma de limitar o acesso de usuários com contas regulares em seu sistema é usar as configurações `chroot`. Eis alguns exemplos de configurações `chroot`:

```
chroot_local_user=YES
chroot_list_enable=YES
chroot_list_file=/etc/vsftpd/chroot_list
```

Ativando as configurações mostradas, você pode criar uma lista de usuários locais e adicioná-los ao arquivo `/etc/vsftpd/chroot_list`. Depois que um desses usuários faz login, ele deve ser impedido de ir a lugares no sistema que estejam fora da estrutura do diretório inicial dele.

Se uploads para o servidor FTP forem permitidos, os diretórios em que um usuário tentar subir arquivos devem ser graváveis por esse usuário. Mas os uploads podem ser armazenados com um nome que não o do usuário que enviou o arquivo. Esse é um dos recursos discutidos a seguir, na seção “Permitindo upload”.

Permitindo upload

Para permitir qualquer forma de gravação no servidor `vsftpd`, você deve ter `write_enable=YES` configurado no arquivo `vsftpd.conf` (o que está, por padrão). Por isso, se contas locais estiverem habilitadas, os usuários podem fazer login e começar imediatamente a fazer upload de arquivos para seus próprios diretórios iniciais. Mas usuários anônimos não têm a capacidade de fazer upload de arquivos por padrão.

Para permitir uploads anônimos com `vsftpd`, você deve ter a primeira opção no exemplo de código a seguir e pode querer a segunda linha de código também (ambas podem ser ativadas removendo o caractere de comentário do arquivo `vsftpd.conf`). O primeiro permite que usuários anônimos façam upload de arquivos, o segundo lhes permite criar diretórios:

```
anon_upload_enable=YES
anon_mkdir_write_enable=YES
```

O próximo passo é criar um diretório onde os usuários anônimos podem gravar. Qualquer diretório sob o diretório `/var/ftp`, que tem permissões de gravação para o usuário `ftp`, o grupo `ftp` ou `other` pode ser gravado por um usuário anônimo. Uma coisa comum é criar um diretório de uploads com permissão aberta para gravação. Eis alguns exemplos de comandos para executar no servidor:

```
# mkdir /var/ftp/uploads
# chown ftp:ftp /var/ftp/uploads
# chmod 775 /var/ftp/uploads
```

Desde que o firewall esteja aberto e booleanos SELinux estejam configurados corretamente, um usuário anônimo pode usar `cd` para mudar para o diretório `uploads` e transferir um arquivo do sistema local do usuário para o diretório `uploads`. No servidor, o arquivo pertenceria ao usuário `ftp` e ao grupo `ftp`. As permissões configuradas no diretório (775) permitiriam a você ver os arquivos que foram enviados, mas não alterá-los ou substituí-los.

Uma razão para permitir FTP anônimo e, então, habilitá-lo para uploads anônimos, é permitir que as pessoas que você não conhece enviem arquivos para a pasta de uploads deles. Como qualquer um que pode encontrar o servidor pode gravar nesse diretório, alguma forma de segurança precisa existir. Você quer evitar que um usuário anônimo veja arquivos enviados por outros usuários, pegue arquivos ou exclua arquivos enviados por outros usuários anônimos de FTP. Uma forma de segurança é o recurso `chown` do FTP.

Ao definir os dois seguintes valores, você pode permitir uploads anônimos. O resultado dessas definições é que quando um usuário anônimo faz upload de um arquivo, a posse desse arquivo é imediatamente atribuída a um usuário diferente. O seguinte é um exemplo de algumas configurações `chown` que você pode colocar em seu arquivo `vsftpd.conf` para usar com o diretório de upload anônimo:

```
chown_uploads=YES
chown_username=joe
```

Se um usuário anônimo fizesse upload de um arquivo depois de `vsftpd` ter sido reiniciado com essas configurações, a posse sobre o arquivo enviado seria do usuário `joe` e do grupo `ftp`. As permissões seriam de leitura/gravação para o proprietário e nada os outros (`rw-----`).

Até agora, você viu as opções de configuração para recursos individuais em seu servidor `vsftpd`. Há alguns conjuntos de variáveis `vsftp.conf` que podem funcionar juntos de maneiras que seriam adequadas para certos tipos de sites FTP. A próxima seção contém um desses exemplos, representados por um arquivo de configuração `vsftpd.conf` de exemplo que vem com o pacote `vsftpd`. Esse arquivo pode ser copiado de um diretório de arquivos de exemplo para o arquivo `/etc/vsftpd/vsftpd.conf`, para uso com um servidor FTP que está disponível na Internet.

Configurando vsftpd para a internet

Para compartilhar com segurança arquivos de seu servidor FTP na internet, você pode bloquear seu servidor, limitando-o a somente permitir downloads e apenas de usuários anônimos. Para iniciar com uma configuração que é projetada para compartilhar arquivos com segurança `vsftpd` pela internet, faça um backup do seu arquivo `/etc/vsftpd/vsftpd.conf` atual e copie esse arquivo para substituir seu `vsftpd.conf`:

```
/usr/share/doc/vsftpd-*/EXAMPLE/INTERNET_SITE/vsftpd.conf
```

Os parágrafos a seguir descrevem o conteúdo desse `vsftpd.conf`. As configurações na primeira seção definem os direitos de acesso ao servidor:

```
# Access rights
anonymous_enable=YES
local_enable=NO
write_enable=NO
anon_upload_enable=NO
anon_mkdir_write_enable=NO
anon_other_write_enable=NO
```

Ativar `anonymous_enable` (YES) e desativar `local_enable` (NO) assegura que ninguém pode fazer logon no servidor FTP usando uma conta de usuário Linux regular. Todos devem vir pela conta anônima. Ninguém pode fazer upload de arquivos (`write_enable=NO`). Portanto, o usuário `anonymous` não pode fazer upload de arquivos (`anon_upload_enable=NO`), criar diretórios (`anon_mkdir_write_enable=NO`) nem gravar no servidor (`anon_other_write_enable=NO`). Eis as configurações de segurança:

```
# Security
anon_world_readable_only=YES
connect_from_port_20=YES
hide_ids=YES
pasv_min_port=50000
```

```
pasv_max_port=60000
```

Como o daemon `vsftpd` pode ler arquivos atribuídos ao usuário e grupo `ftp`, configurar `anon_world_readable_only=YES` garante que usuários anônimos serão capazes de ver arquivos em que o bit de permissão de leitura está ativado para `other` (`-----r--`), mas não grava arquivos. A configuração `connect_from_port_20=YES` fornece ao daemon `vsftpd` um pouco mais de permissão para enviar dados da maneira como um cliente pode solicitar, permitindo comunicações de dados no estilo `PORT`. Usar `hide_ids=YES` oculta as permissões reais configuradas em arquivos de tal modo que, para o usuário que acessar o site `FTP`, tudo parecerá pertencer ao usuário `ftp`. As duas configurações `pasv` restringem o intervalo de portas que podem ser utilizadas com `FTP` passivo (no qual o servidor pega uma porta de maior número para enviar dados) a um número entre 50000 e 60000.

A próxima seção contém recursos do servidor `vsftpd`:

```
# Features
xferlog_enable=YES
ls_recurse_enable=NO
ascii_download_enable=NO
async_abor_enable=YES
```

Com `xferlog_enable=YES`, todas as transferências de arquivos de e para o servidor são registradas no arquivo `/var/log/xferlog`. Configurar `ls_recurse_enable=NO` impede que os usuários listem recursivamente o conteúdo de um diretório `FTP` (em outras palavras, impede o tipo de lista que você pode obter com o comando `ls -R`), porque, em um site de grande porte, isso poderia drenar recursos. Desabilitar downloads de `ASCII` obriga todos os downloads a estarem no modo binário (impedindo que os arquivos sejam traduzidos em `ASCII`, o que é inadequado para arquivos binários).

A configuração `async_abor_enable=YES` garante que alguns clientes `FTP` que poderiam travar ao abortar uma transferência não travem.

As definições a seguir têm um impacto sobre o desempenho:

```
# Performance
one_process_model=YES
idle_session_timeout=120
data_connection_timeout=300
accept_timeout=60
connect_timeout=60
anon_max_rate=50000
```

Com `one_process_model=YES` configurado, o desempenho pode melhorar, porque `vsftpd` carregará um processo por conexão. Reduzir o `idle_session_timeout` do padrão 300 segundos para 120 segundos faz com que clientes `FTP` que estão ociosos há mais de 2 minutos sejam desconectados. Então, menos tempo é gasto no gerenciamento de sessões `FTP` que não estão mais em uso. Se a transferência de dados parar por mais de `data_connection_timeout` segundos (300 segundos aqui), a conexão com o cliente é derrubada.

A configuração `accept_timeout` de 60 segundos permite um minuto para uma conexão PASV ser aceita pelo cliente remoto. O `connect_timeout` configura quanto tempo um cliente remoto tem para responder a uma solicitação para estabelecer uma conexão de dados no estilo PORT. Limitar a taxa de transferência de 50000 (bytes por segundo) com `anon_max_rate` pode melhorar o desempenho geral do servidor, limitando a quantidade de largura de banda que cada cliente pode consumir.

Usando clientes FTP para se conectar ao servidor

Há muitos programas clientes que vêm com Linux que você pode usar para se conectar ao servidor FTP. Se você simplesmente quer fazer um download anônimo de alguns arquivos de um servidor FTP, seu navegador Firefox oferece uma interface fácil para fazer isso. Para interações mais complexas entre seu cliente e servidor FTP, há clientes de linha de comando FTP que você pode usar. A seção a seguir descreve algumas dessas ferramentas.

Acessando um servidor FTP a partir do Firefox

O navegador Firefox oferece uma maneira rápida e fácil de testar o acesso ao seu servidor FTP ou para acessar qualquer servidor FTP público. Em seu próprio sistema, escreva **ftp://localhost** na caixa de localização. Se o servidor estiver acessível, você deve ver algo semelhante ao exemplo mostrado na Figura 18.1.

ando um servidor FTP a partir do Firefox

Para efetuar login em um servidor FTP como um usuário específico a partir do Firefox, você pode preceder o nome do host com a notação `username:password@`. Por exemplo:

```
ftp://chris:MypassWd5@localhost
```

Se fornecer o nome de usuário e senha corretamente, você deve ver imediatamente o conteúdo de seu diretório inicial. Clique nas pastas para abri-las. Clique em um arquivo para baixá-lo ou visualizá-lo.

Acessando um servidor FTP com o comando `lftp`

Para testar seu servidor FTP a partir da linha de comando, você pode usar o comando `lftp`. Para instalar o comando `lftp` no Fedora ou no RHEL, digite o seguinte na linha de comando:

```
# yum install lftp
```

Se você usar o comando `lftp` com apenas o nome do servidor FTP que você está tentando acessar, o comando tenta se conectar ao servidor FTP como usuário anônimo. Adicionando `-u nome_de_usuario`, você pode digitar a senha do usuário quando solicitado e ter acesso ao servidor FTP como o usuário que você conectado.

Depois de adicionar suas informações de usuário e senha, você recebe um prompt `lftp` pronto para você começar a digitar comandos. A conexão com o servidor é feita quando você digita seu primeiro comando. Você pode usar os comandos para mover o servidor FTP e então usar os comandos de `get` e `put` para download e upload de arquivos.

O exemplo a seguir mostra como usar os comandos que acabamos de descrever. Ele assume que o servidor FTP (e as medidas de segurança associadas) foi configurado para permitir aos usuários locais se conectarem, lerem e gravarem arquivos:

```

# lftp -u chris localhost
Password:
*****
lftp chris@localhost:~> pwd
ftp://chris@localhost/%2Fhome/chris
lftp chris@localhost:~> cd stuff/state/
lftp chris@localhost:~/stuff/state> ls
-rw-r--r-- 1 13597 13597          1394 Oct 23 2009 enrolled-20091012
-rw-r--r-- 1 13597 13597          514 Oct 23 2009 enrolled-20091013
lftp chris@localhost:~/stuff/state> !pwd
/root
lftp chris@localhost:~/stuff/state> get survey-20091023.txt
3108 bytes transferred
lftp chris@localhost:~/stuff/state> put /etc/hosts
201 bytes transferred
lftp chris@localhost:~/stuff/state> ls
-rw-r--r-- 1 13597 13597          1394 Oct 23 2009 enrolled-20091012
-rw-r--r-- 1 13597 13597          514 Oct 23 2009 enrolled-20091013
-rw-r--r-- 1 0 0                201 May 03 20:22 hosts
lftp chris@localhost:~/stuff/state> !ls
anaconda-ks.cfg bin          install.log
dog Pictures          sent
Downloads Public     survey-20091023.txt
lftp chris@localhost:~/stuff/state> quit

```

Depois de fornecer o nome do usuário (-u chris), lftp solicitará a senha do usuário chris. Digitar pwd mostra que Chris está conectado ao host local e que /home/chris é o diretório atual. Assim como você faria a partir de um shell de linha de comando do Linux, você pode usar cd para mudar para outro diretório e ls para listar o conteúdo desse diretório.

Para ter os comandos que você executa interpretados pelo sistema cliente, você pode simplesmente colocar um ponto de exclamação (!) na frente de um comando. Por exemplo, !pwd mostra que o diretório atual no sistema que iniciou o lftp é /root. É bom saber isso, porque se você receber um arquivo do servidor sem especificar seu destino, ele vai para o diretório atual do cliente (neste caso, /root). Outros comandos que você pode executar para que sejam interpretados pelo sistema cliente são !cd (para mudar de diretório) e !ls (para listar os arquivos).

Supondo que você tenha permissão de leitura sobre um arquivo no servidor e permissão de gravação a partir do diretório atual no sistema iniciador, você pode usar o comando get para baixar um arquivo do servidor (get survey-20091023.txt). Se tiver permissão de gravação e upload no diretório atual no servidor, você pode usar put para copiar um arquivo para o servidor (put /etc/hosts).

Executar um comando ls mostra que o /etc/hosts foi copiado para o servidor. Executar o comando !ls permite que você veja que o arquivo survey-20091023.txt foi transferido do servidor para o sistema iniciador.

Usando o cliente gFTP

Muitos outros clientes FTP também estão disponíveis no Linux. Outro cliente de FTP que você pode tentar é gFTP. O cliente gFTP fornece uma interface que permite ver ambos os lados, local e remoto, de sua sessão de FTP. Para instalar gFTP no Fedora e no Red Hat Enterprise Linux, execute o seguinte comando para instalar o pacote gftp:

```
# yum install gftp
```


Para iniciar gFTP, carregue-o a partir do menu de aplicativos ou execute `gftp` & a partir do shell. Para usá-lo, digite a URL do servidor FTP a que você deseja se conectar, digite o nome de usuário que você deseja usar (como `anonymous`) e pressione Enter. A Figura 18.2 mostra um exemplo de gFTP sendo usado para se conectar a um diretório de documentação no site `ftp://kernel.org`.

ente FTP do gFTP permite que você veja os dois lados de uma sessão FTP.

Para percorrer o site FTP a partir de gFTP, basta clicar duas vezes em pastas (tal como faria a partir de uma janela do gerenciador de arquivos). Os caminhos completos para o diretório local (à esquerda) e o diretório remoto (à direita) são mostrados acima das listagens de arquivos e pastas abaixo.

Para transferir um arquivo do lado remoto para o lado local, selecione o arquivo que você quer à direita e, então, clique na seta no meio da tela apontando para a esquerda. Observe o progresso da transferência de arquivos a partir das mensagens na parte inferior da tela. Quando a transferência for concluída, você verá o arquivo aparecer no painel esquerdo.

Você pode gravar o endereço para se conectar a um site FTP na forma de um bookmark. Esse endereço é adicionado a um conjunto de bookmarks que já estão armazenados no menu Bookmarks. Você pode selecionar sites da lista para experimentar o gFTP. A maioria dos sites é para distribuições Linux e outros sites de software de código-fonte aberto.

Resumo

Configurar um servidor FTP é uma maneira fácil de compartilhar arquivos em uma rede TCP. The Very Secure FTP Daemon (pacote `vsftpd`) está disponível para o Fedora, o Red Hat Enterprise Linux e outros sistemas Linux.

Um servidor `vsftpd` padrão permite a usuários anônimos baixar arquivos do servidor e a usuários regulares do Linux fazer o upload ou download de arquivos (desde que algumas configurações de segurança sejam alteradas). Navegar por um servidor FTP é semelhante a navegar por um sistema de arquivos Linux. Você se move para cima e para baixo na estrutura de diretório para encontrar o conteúdo desejado.

Há clientes FTP gráficos e clientes baseados em texto. Um cliente baseado em texto popular para Linux é o `lftp`. Quanto a clientes FTP gráficos, você pode usar um navegador web comum, como o Firefox, ou clientes FTP dedicados, como gFTP.

Servidores FTP não são a única maneira de compartilhar arquivos em uma rede de Linux. O serviço Samba fornece uma maneira de compartilhar arquivos em uma rede de modo que o diretório do Linux compartilhado parece um diretório compartilhado a partir de um sistema Windows. O Capítulo 19 descreve como usar o Samba para oferecer o compartilhamento de arquivos no estilo Windows.

Exercícios

Os exercícios desta seção descrevem tarefas relacionadas com configurar um servidor FTP no Linux e se conectar a esse servidor usando um cliente FTP. Se você empacar, as soluções para as tarefas são

mostrados no Apêndice B. Tenha em mente que as soluções mostradas no Apêndice B são geralmente apenas uma das várias maneiras de completar uma tarefa.

Não faça esses exercícios em um sistema Linux rodando um servidor FTP público porque eles quase certamente irão interferir nesse servidor.

1. Determine qual pacote fornece o serviço Very Secure FTP Daemon.
2. Instale o pacote Very Secure FTP Daemon em seu sistema e procure os arquivos de configuração nesse pacote.
3. Inicie o serviço Very Secure FTP Daemon e configure-o para iniciar quando o sistema é inicializado.
4. No sistema que executa o servidor FTP, crie um arquivo chamado `test` no diretório FTP anônimo que contém as palavras “Welcome to your vsftpd server” (“Bem-Vindo ao seu Servidor vsftpd”).
5. A partir de um navegador no sistema executando o servidor FTP, abra o arquivo `test` a partir do diretório inicial do FTP anônimo. Tenha certeza de que você pode ver o conteúdo desse arquivo.
6. A partir de um navegador fora do sistema que está executando o servidor FTP, tente acessar o arquivo `test` no diretório inicial do FTP anônimo. Se você não puder acessar o arquivo, verifique se o firewall (`iptables`), o SELinux e TCP wrappers estão configurados para permitir acesso a esse arquivo.
7. Configure o servidor `vsftpd` para permitir envio de arquivos por usuários anônimos para um diretório chamado `in`.
8. Instale o `lftp` cliente de FTP (se você não tem um segundo sistema Linux, instale `lftp` no mesmo host que executa o servidor FTP). Se você não conseguir fazer upload de arquivos para o diretório `in`, verifique se o firewall (`iptables`), o SELinux e os TCP wrappers estão configurados para permitir o acesso a esse arquivo.
9. Usando qualquer cliente de FTP que você escolher, visite o diretório `/pub/linux/docs/man-pages`, no site `ftp://kernel.org`, e liste o conteúdo desse diretório.
10. Usando qualquer cliente de FTP que você quiser, baixe o arquivo `man-pages-3.41.tar.gz`, a partir do diretório `kernel.org`, que você acabou de visitar, para o diretório `/tmp` em seu sistema local.

Configurando um servidor de compartilhamento de arquivos do Windows (Samba)

NESTE CAPÍTULO

Obtendo e instalando o Samba

Usando recursos de segurança do Samba

Editando o arquivo de configuração `smb.conf`

Acessando o Samba a partir de clientes Linux e Windows

Usando o Samba na empresa

O Samba é um projeto que implementa versões de código aberto dos protocolos usados para compartilhar arquivos e impressoras, bem como para autenticar usuários e restringir hosts, entre sistemas Windows. O Samba oferece uma série de maneiras de compartilhar arquivos entre sistemas Windows, Linux e OS/X Mac que são bem conhecidas e facilmente disponíveis para os usuários desses sistemas.

Este capítulo orienta você ao longo do processo de instalação e configuração de um servidor Samba. Ele descreve os recursos de segurança que você precisa saber para compartilhar recursos de arquivos e impressoras e descreve como acessar esses recursos a partir de sistemas Linux e Windows.

Entendendo o Samba

O Samba (www.samba.org) é um conjunto de programas que permite que Linux, UNIX e outros sistemas interoperem com protocolos de compartilhamento de arquivos e impressoras do Microsoft Windows. Windows, DOS, OS/2, Mac OS/X e outros sistemas clientes podem acessar servidores Samba para compartilhar arquivos e impressoras da mesma maneira como fariam a partir de servidores de arquivo e impressão Windows.

Com o Samba, você pode usar o padrão TCP/IP para se comunicar com os clientes. Para o serviço de nomes, o Samba suporta regulares hostnames TCP/IP, assim como nomes NetBIOS. Por esse motivo, o Samba não requer o protocolo NetBEUI (frame Microsoft Raw NetBIOS). O compartilhamento de arquivos é feito usando o Common Internet File System (CIFS), que é uma implementação aberta do

protocolo Server Message Block (SMB).

O projeto Samba tem feito grandes esforços para tornar seu software seguro e robusto. Na verdade, muitas pessoas preferem usar servidores Samba em vez dos servidores de arquivos do Windows por causa da maior segurança que é inerente à execução de serviços no estilo Windows de compartilhamento de arquivos no Linux ou outros sistemas operacionais tipo Unix.

Para além de toda a discussão técnica, porém, o resultado final é que o Samba torna fácil compartilhar arquivos e impressoras entre servidores Linux e sistemas desktop Windows. Para o servidor, apenas alguns arquivos de configuração e ferramentas são necessários para gerenciar o Samba. Para os clientes, os recursos compartilhados apenas aparecem sob a seleção Network no gerenciador de janelas ou no Ambiente de Rede em sistemas Windows mais antigos.

Interfaces para gerenciar o Samba em um servidor Linux incluem a janela Samba Server Configuration (`system-config-samba`) e interface baseada na web SWAT Samba (Samba Web Administration Tool). Como alternativa, você pode editar diretamente os arquivos de configuração do Samba (particularmente `smb.conf`) e executar alguns comandos para configurá-lo.

ta

ora não seja abordada neste capítulo, muitas pessoas preferem a interface SWAT do Samba para editar diretamente os arquivos de configuração. Em algumas distribuições Linux, o Samba SWAT pode ser a única interface disponível.

Um guia rápido para fazer o Samba SWAT funcionar em seu sistema Fedora ou RHEL:

1. Instale os pacotes `samba-swat` e `xinetd`.
2. Edite o arquivo `/etc/xinetd.d/swat` e altere a linha `disable` para o seguinte:
`disable = no`.
3. Reinicie o serviço `xinetd`: `xinetd start ; chkconfig xinetd on`.
4. Abra seu navegador web local neste endereço: `http://localhost:901`.

A Samba Web Administration Tool que se abre no navegador fornece uma interface diferente para configurar o servidor Samba, diretórios compartilhados e impressoras que são descritos no restante deste capítulo. Explorando a interface, você deve ser capaz de encontrar a maneira de inserir essas informações na interface SWAT.

Para começar a usar o Samba em seu sistema Linux, você precisa instalar alguns poucos pacotes de software, conforme descrito na próxima seção.

Instalando o Samba

No Red Hat Enterprise Linux e no Fedora, para configurar um servidor de arquivo e impressão Samba, os únicos pacotes necessários são `samba` e `samba-common`. Entre outros componentes, o pacote `samba` inclui o daemon do serviço Samba (`/usr/sbin/smbd`) e o daemon do servidor de nomes NetBIOS

(/user/sbin/nmbd). O pacote `samba-common` contém arquivos de configuração do servidor (`smb.conf`, `lmhosts` e outros) e comandos para adicionar senhas e testar os arquivos de configuração, juntamente com outros recursos do Samba.

Recursos de outros pacotes são referenciados neste capítulo, portanto vou descrever como instalar esses pacotes também. Esses pacotes incluem:

- **Pacote `samba-client`** — Contém ferramentas de linha de comando, como `smbclient` (para conexão com compartilhamentos Samba ou Windows), `nmblookup` (para procurar endereços de host) e `findsmb` (para localizar máquinas SMB na rede).
- **Pacote `samba-doc`** — Contém a documentação do Samba nos formatos HTML e PDF. A documentação está no diretório `/usr/share/doc/samba-doc-*`.
- **Pacote `samba-swat`** — Contém uma interface baseada na web para a configuração de um servidor Samba.
- **Pacote `samba-winbind`** — Inclui componentes que permitem que seu servidor Samba no Linux se torne um membro completo de um domínio Windows, incluindo o uso de contas de usuário e grupo Windows no Linux.

Para instalar todos os pacotes mencionados acima (`samba-common` será instalado como uma dependência do `samba`, por isso não precisa ser especificamente observado), digite o seguinte, como `root`, na linha de comando no Fedora:

```
# yum install samba samba-client samba-doc samba-swat samba-winbind
...
Dependencies Resolved
=====
Package      Arch  Version           Repository Size
=====
Installing:
samba        i686  1:3.6.5-85.fc16  updates  4.7 M
samba-client i686  1:3.6.5-85.fc16  updates  10 M
samba-doc    i686  1:3.6.5-85.fc16  updates  7.5 M
samba-swat   i686  1:3.6.5-85.fc16  updates  2.8 M
samba-winbind i686  1:3.6.5-85.fc16  updates  3.2 M
Installing for dependencies:
samba-common i686  1:3.6.5-85.fc16  updates  9.0 M
Transaction Summary
=====
Install 5 Packages (+1 Dependent package)
Total download size: 37 M
Installed size: 134 M
Is this ok [y/N]: y
```

Depois de ter instalado os pacotes Samba, dê uma olhada nos arquivos de configuração nos pacotes `samba` e `samba-common`:

```
# rpm -qc samba samba-common
/etc/logrotate.d/samba
/etc/pam.d/samba
/etc/samba/smbusers
/etc/samba/lmhosts
/etc/samba/smb.conf
/etc/sysconfig/samba
```

Os arquivos `/etc/logrotate.d/samba`, `/etc/pam.d/samba` e `/etc/sysconfig/samba` geralmente não são modificados. O primeiro configura a maneira como os arquivos em `/var/log/samba` são reciclados (copiados para outros arquivos e removidos) ao longo do tempo.

O segundo contém definições de autenticação do Samba (que essencialmente usam o arquivo `password-auth` para incluir os mesmos métodos de autenticação básicos baseados em senha e outros que `login`, `sshd` e outras ferramentas usam para fazer autenticação de senha no sistema). O terceiro é um arquivo em que você pode colocar opções que são passadas para os daemons `smbd`, `nmbd` ou `winbindd`, as quais permitem desativar recursos como depuração.

A maioria dos arquivos de configuração do Samba está no diretório `/etc/samba`. O arquivo `smb.conf` é o principal arquivo de configuração, onde você coloca as configurações globais para o servidor Samba, bem como informações sobre compartilhamento de arquivos e impressoras individuais (mais sobre isso mais tarde). O arquivo `smbusers` permite mapear usuários do Linux para nomes de usuários do Samba. O arquivo `lmhosts` permite que o hostname NetBIOS do Samba seja mapeado para endereços IP.

Uma grande quantidade de documentação é incluída nos pacotes Samba. À medida que configura o servidor Samba, você vai querer consultar a página `man` do arquivo `smb.conf` (`man smb.conf`). Há também páginas `man` para comandos do Samba, como `smbpasswd` (para alterar senhas), `smbclient` (para conectar a um servidor Samba) e `nmblookup` (para procurar informações NetBIOS).

Depois de ter instalado os pacotes Samba e completado um rápido levantamento do que eles contêm, tente iniciar o serviço Samba e ver o que você tem em uma configuração padrão.

Iniciando e parando o Samba

Com `samba` e `samba-common` instalados, você pode iniciar o servidor e investigar como ele é executado na configuração padrão. Os dois principais serviços estão associados a um servidor Samba, e cada qual tem seu próprio serviço `daemon`. Esses serviços incluem:

- **smb** — Esse serviço controla o processo `daemon` `smbd`, que fornece os serviços de compartilhamento de arquivo e impressão que podem ser acessados por clientes Windows.
- **NMB** — Esse serviço controla o `daemon` `nmbd`. Ao fornecer serviço de mapeamento de nome NetBIOS para endereço, o `nmbd` pode mapear solicitações de clientes Windows para nomes NetBIOS de modo que eles possam ser convertidos em endereços IP.

Para compartilhar arquivos e impressoras com outros sistemas Linux com Samba, apenas o serviço `smb` é necessário. A próxima seção descreve como iniciar e ativar o serviço `smb`.

Iniciando o serviço Samba (smb)

O serviço `smb` é o que inicia o servidor `smbd` e torna arquivos e impressoras disponíveis a partir de seu sistema local para outros computadores na rede. Como de costume, os serviços são ativados e começam

de forma diferente em diferentes sistemas Linux. Para diferentes sistemas Linux, você precisa encontrar o nome do serviço e a ferramenta correta para iniciar o daemon `smbd`.

No Fedora, para habilitar o Samba a iniciar na inicialização do sistema e ser carregado imediatamente, digite o seguinte na linha de comando como root:

```
# systemctl enable smb.service
# systemctl start smb.service
# systemctl status smb.service
smb.service - Samba SMB Daemon
  Loaded: loaded (/lib/systemd/system/smb.service; enabled)
  Active: active (running) since Thu, 17 May 2012 06:14:08 -0400;
        2 days ago
  Main PID: 4838 (smbd)
  CGroup: name=systemd:/system/smb.service
          └─ 4838 /usr/sbin/smbd
          └─ 4840 /usr/sbin/smbd
```

O primeiro comando `systemctl` habilita o serviço, o segundo inicia-o imediatamente e o terceiro mostra o status. Para investigar mais, note que o serviço de arquivo está localizado em `/lib/systemd/system/smb.service`. Dê uma olhada no conteúdo desse arquivo:

```
# cat /lib/systemd/system/smb.service
[Unit]
Description=Samba SMB Daemon
After=syslog.target network.target nmb.service winbind.service
[Service]
Type=forking
PIDFile=/run/smbd.pid
LimitNOFILE=16384
EnvironmentFile=-/etc/sysconfig/samba
ExecStart=/usr/sbin/smbd $SMBDOPTIONS
```

O processo de daemon do Samba (`smbd`) inicia depois dos serviços `syslog`, `network`, `nmb` e `winbind`. O arquivo `/etc/sysconfig/samba` contém variáveis que são passadas como argumentos para o daemon `/usr/sbin/smbd` quando ele inicia. Nenhuma opção está configurada por padrão para o daemon `smbd` (nenhuma está configurada para os daemons `nmbd` ou `winbindd` também, os quais também podem ter opções inseridas nesse arquivo).

No RHEL 6 e versões mais recentes, você pode iniciar o serviço Samba da seguinte forma:

```
# service smb start
Starting SMB services:          [ OK ]
# chkconfig smb on
# service smb status
smbd (pid 28056) is running...
# chkconfig --list smb
smb          0:off 1:off 2:on 3:on 4:on 5:on 6:off
```


Quer você esteja executando o servidor Samba no RHEL, Fedora, Linux ou outro sistema, você pode verificar o acesso ao servidor Samba usando o comando `smbclient` (do pacote `samba-client`). Você pode obter informações básicas de um servidor Samba usando o seguinte comando:

```
# smbclient -L 192.168.0.119
Enter root's password: <ENTER>
Anonymous login successful
Domain=[MYGROUP] OS=[Unix] Server=[Samba 3.6.5-85.fc16]
  Sharename      Type            Comment
  -----
  IPC$           IPC            IPC Service
                  (Samba Server Version 3.6.5-85.fc16)
  DeskJet       Printer        DeskJet
  Jeeves        Printer        HP Deskjet 3050 J610 series
  deskjet-5550  Printer        hp deskjet 5550
Anonymous login successful
Domain=[MYGROUP] OS=[Unix] Server=[Samba 3.6.5-85.fc16]
  Server          Comment
  -----
  Workgroup       Master
  -----
```

A saída `smbclient` permite ver que serviços estão disponíveis a partir do servidor. Eu dei o endereço IP do servidor como um argumento para identificar a localização dele. Por padrão, o login anônimo é permitido ao consultar o servidor (assim, simplesmente pressionei Enter quando fui solicitado a informar uma senha).

Você pode distinguir uma série de coisas sobre a configuração do servidor Samba padrão a partir dessa saída:

- O domínio padrão (esse é um domínio do Windows e não um domínio DNS) está configurado como MYGROUP.
- O nome do servidor padrão é configurado como a versão atual do Samba, neste caso, Samba 3.6.5-85.fc16, o que representa build 85 da release 3.6.5 do Samba no Fedora 16.
- Todas as impressoras que são compartilhadas por meio do servidor CUPS em seu sistema Linux também estão, por padrão, disponíveis a partir do servidor Samba em execução no mesmo sistema.
- Nenhum diretório ainda está compartilhado no servidor.
- Não há serviço de nomes NetBIOS ainda em execução no servidor Samba.

Então, você pode decidir se quer executar o serviço de nomes NetBIOS no servidor Samba.

Iniciando o servidor de nomes NetBIOS (nmbd)

Se não houver nenhum servidor de domínio do Windows rodando na rede, como é o caso aqui, você pode iniciar o serviço `nmb` no host Samba para prestar esse serviço. Para iniciar o serviço `nmb` (daemon `nmbd`) no Fedora, digite o seguinte:

```
# systemctl enable nmb.service
# systemctl start nmb.service
# systemctl status nmb.service
```

No RHEL 6 e versões anteriores, você deve digitar o seguinte para iniciar o serviço nmb:

```
# service nmb start
# service nmb status
# chkconfig nmb on
# chkconfig --list nmb
```

Independentemente de como o serviço NetBIOS foi iniciado, o daemon `nmbd` agora deve estar em execução e pronto para servir mapeamento de nome para endereço NetBIOS. Execute o comando `smbclient -L` novamente, seguido pelo endereço IP do servidor. Desta vez, as últimas linhas da saída devem exibir a informação obtida a partir do servidor NetBIOS agora em execução no servidor. Neste caso, as últimas linhas estavam assim:

```
# smbclient -L 192.168.0.119
...
Server           Comment
-----
SPIKE            Samba Server Version 3.6.5-85.fc16
Workgroup        Master
-----
MYGROUP          SPIKE
```

Você pode ver que o novo nome NetBIOS do servidor é `SPIKE` e que é o servidor principal para o grupo de trabalho. Para consultar o servidor `nmbd` para o endereço IP de `SPIKE`, você deve digitar o seguinte:

```
# nmblookup -U 192.168.0.119 SPIKE
querying SPIKE on 192.168.0.119
192.168.0.119 SPIKE<00>
```

Você deve ser capaz de ver seu servidor Samba executando a partir do sistema local agora. O hostname atribuído ao sistema (neste caso, `SPIKE`) vem do valor hostname que você pode configurar no arquivo `/etc/sysconfig/network` ou no servidor DHCP. Se você não tiver configurado o hostname (digite **hostname** para ver se ele está configurado), o nome do servidor NetBIOS apareceria como `LOCALHOST`.

Mas se você tiver um firewall configurado ou o SELinux habilitado, você pode ainda não ser capaz de acessar plenamente o servidor Samba a partir de um sistema remoto. A próxima seção deve ajudar a abrir o Samba para sistemas fora do sistema local, bem como habilitar alguns recursos do Samba, que podem ser desativados pelo SELinux.

Parando os serviços do Samba (SMB) e do NetBIOS (BNM)

A fim de parar os serviços `smb` e `nmb` no Fedora, você pode usar o mesmo comando `systemctl` utilizado para iniciá-los. Você pode usar o mesmo comando para desabilitar os serviços, assim eles não iniciam de novo quando o sistema é inicializado. Eis alguns exemplos de como parar imediatamente os serviços `smb` e `nmb`:

```
# systemctl stop smb.service
```

```
# systemctl stop nmb.service
```

No RHEL 6 e versões anteriores, você deve digitar o seguinte a fim de parar os serviços smb e nmb:

```
# service smb stop
# service nmb stop
```

Para impedir que os serviços smb e nmb iniciem da próxima vez que o sistema for reinicializado, digite os seguintes comandos no Fedora:

```
# systemctl disable smb.service
# systemctl disable nmb.service
```

No Red Hat Enterprise Linux, digite os seguintes comandos para desabilitar os serviços smb e nmb:

```
# chkconfig smb off
# chkconfig nmb off
# chkconfig --list nmb
```

Naturalmente, você só quer parar ou desabilitar os serviços smb e nmb se não quiser mais utilizar o serviço Samba. Se estiver pronto para continuar a configurar seu serviço Samba, você pode continuar e começar a configurar seus recursos de segurança do Linux para permitir que o serviço do Samba se torne disponível para os outros em sua rede.

Protegendo o Samba

Se não é capaz de acessar o servidor Samba imediatamente após iniciá-lo, você provavelmente tem algum trabalho de segurança a fazer. Como muitas das instalações padrão do Linux impedem, em vez de permitir, o acesso ao sistema, lidar com a segurança de um serviço como o Samba geralmente tem mais a ver com torná-lo disponível do que torná-lo seguro.

Eis os recursos de segurança dos quais você deve estar ciente ao configurar seu sistema Samba:

- **Firewalls** — O firewall padrão para o Fedora, RHEL e outros sistemas Linux impede qualquer acesso a serviços locais a partir de sistemas externos. Assim, para permitir que usuários de outros computadores acessem seu serviço Samba, você deve criar regras de firewall que abrem uma ou mais portas para os protocolos TCP selecionados (em particular).
- **SELinux** — Muitos recursos do Samba são considerados potencialmente inseguros pelo SELinux. Como os booleanos (opções que ativam/desativam determinados recursos) padrão do SELinux são configurados para oferecer o acesso mínimo exigido, você precisa ativar os booleanos para recursos como permitir que os usuários acessem seus próprios diretórios iniciais com o Samba. Em outras palavras, você pode configurar o Samba para compartilhar o diretório inicial do usuário, mas o SELinux proibirá que alguém tente usar esse recurso a menos que você configure o SELinux para permitir esse recurso.

- **Restrições de host e usuário**— Dentro dos arquivos de configuração do Samba, você pode indicar as máquinas e usuários que podem ter acesso ao servidor Samba como um todo ou a determinados diretórios compartilhados.

As próximas seções descrevem como configurar os recursos de segurança que acabamos de mencionar para o Samba.

Configurando firewalls para o Samba

Se um firewall `iptables` estiver configurado para seu sistema quando você instalar o Samba, o firewall geralmente permitirá quaisquer solicitações de serviço por parte de usuários locais, mas nenhuma por parte de usuários externos. É por isso que, no final da seção de instalação deste capítulo, você deve ter sido capaz de testar que o Samba estava funcionando usando o comando `smbclient` do sistema local. Mas se a solicitação tivesse origem em outro sistema, ela teria sido rejeitada.

Configurar regras de firewall para o Samba consiste principalmente em abrir as portas entrantes que os daemons `smbd` e `nmbd` estão ouvindo. Essas são as portas que você deve abrir para fazer um serviço Samba funcionar em seu sistema Linux:

- **Porta TCP 445** — Essa é a porta principal que o daemon `smbd` do Samba ouve. Seu firewall deve dar suporte às solicitações de pacotes entrantes nessa porta para o Samba para funcionar.
- **Porta TCP 139** — O daemon `smbd` também ouve na porta TCP 139 para lidar com sessões associadas a nomes de host NetBIOS. É possível utilizar o Samba sobre TCP sem abrir essa porta, mas isso não é recomendado.
- **Portas UDP 137 e 138** — O daemon `nmbd` usa essas duas portas para receber solicitações NetBIOS. Se você estiver usando o daemon `nmbd`, essas duas portas devem ser abertas para solicitações de novos pacotes para a conversão de nomes NetBIOS.

Considere um firewall padrão do Fedora que permite pacotes entrantes a partir do host local, de conexões estabelecidas, e relacionados com as conexões estabelecidas, mas nega todos os outros pacotes entrantes. O exemplo a seguir representa um conjunto de regras de firewall no arquivo `/etc/sysconfig/iptables`, com quatro novas regras (destacadas no código que se segue) adicionadas para abrir as portas para o Samba:

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-I INPUT -m state --state NEW -m udp -p udp --dport 137 -j ACCEPT
-I INPUT -m state --state NEW -m udp -p udp --dport 138 -j ACCEPT
-I INPUT -m state --state NEW -m tcp -p tcp --dport 139 -j ACCEPT
```

```
-I INPUT -m state --state NEW -m tcp -p tcp --dport 445 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

Seu firewall pode incluir regras adicionais para permitir solicitações de pacotes entrantes para outros serviços, como serviços Secure Shell (sshd) ou web (httpd). Você pode deixar essas como estão. O ponto principal é ter suas regras do Samba colocadas em algum lugar antes das regras REJECT finais.

Se seu firewall estiver ativado, você pode reiniciá-lo para que as novas regras entrem em vigor. Para isso, digite **systemctl restart iptables.service** (no Fedora) ou **service restart iptables** (no RHEL). Tente conectar-se ao serviço Samba usando o comando `smbclient` novamente ou usando outras técnicas descritas na seção “Acessando compartilhamentos do Samba”, mais adiante, neste capítulo.

Veja o Capítulo 25, “Protegendo o Linux em uma rede”, para mais informações sobre o uso de `iptables`.

Configurando o SELinux para o Samba

Há tanto considerações sobre contextos de arquivo como sobre booleanos relacionados ao uso do Samba com o SELinux no modo Enforcing. Contextos de arquivos devem ser devidamente configurados em um diretório que é compartilhado pelo Samba. Booleanos permitem substituir a abordagem “segura por padrão” para determinados recursos do Samba.

Você pode encontrar informações sobre como o SELinux confina o Samba na página `man samba_selinux` (`man samba_selinux`). Para uma compreensão mais profunda do SELinux, consulte o Capítulo 24, “Aprimorando a segurança do Linux com o SELinux”.

Configurando os booleanos do SELinux para o Samba

Há várias maneiras de ver os booleanos associados com o Samba. No Fedora e no RHEL, você pode instalar o pacote `policycore-utils-gui` e executar o comando `system-config-selinux`. Na janela SELinux Administration que aparece, selecione Booleans na coluna esquerda e examine os booleanos listados com o módulo `samba`.

Outra forma para listar booleanos para o Samba é usar o comando `semanage`, como segue:

```
# semanage boolean -l | grep -i samba
```

O que se segue é uma lista de booleanos do SELinux que se aplicam ao Samba e suas descrições. O booleano primeiro permite que você deixe o SELinux no modo Enforcing, mas impede que o SELinux restrinja scripts que o servidor Samba pode executar. Todos os outros configuram quais arquivos e diretórios do servidor Samba podem ser lidos e gravados por usuários do Samba:

- **samba_run_unconfined** — Permite que o Samba execute scripts não confinados.
- **allow_smbd_anon_write** — Permite que o Samba deixe usuários anônimos modificarem arquivos

públicos utilizados para serviços públicos de transferência de arquivos. Arquivos e diretórios devem ser rotulados como `public_content_rw_t`.

- **samba_enable_home_dirs** — Permite que o Samba compartilhe o diretório inicial dos usuários.
- **samba_export_all_ro** — Permite que o Samba compartilhe qualquer arquivo e diretório somente leitura.
- **use_samba_home_dirs** — Permite que um servidor Samba remoto acesse diretórios na máquina local.
- **samba_create_home_dirs** — Permite que o Samba crie novos diretórios iniciais (por exemplo, via PAM).
- **samba_export_all_rw** — Permite que o Samba compartilhe qualquer arquivo ou diretório de leitura/gravação.

Os seguintes booleanos afetam a capacidade do Samba para compartilhar diretórios que são eles próprios montados a partir de outros serviços remotos (tal como o NFS) ou atuem como um controlador de domínio do Windows:

- **samba_share_fusefs** — Permite que o Samba exporte volumes `ntfs/fusefs`.
- **samba_share_nfs** — Permite que o Samba exporte volumes NFS.
- **samba_domain_controller** — Permite que o Samba atue como controlador de domínio, adicione usuários e grupos e altere senhas.

O comando `setsebool` é usado para transformar booleanos do SELinux ativados ou desativados. Usado com a opção `-P`, `setsebool` configura permanentemente o booleano que você indicar. Por exemplo, para permitir que o Samba compartilhe qualquer arquivo ou diretório com permissão somente leitura a partir do servidor, você pode digitar o seguinte em um shell, como usuário `root`:

```
# setsebool -P samba_export_all_ro on
# getsebool samba_export_all_ro
samba_export_all_ro --> on
```

O comando `setsebool` configura o booleano, neste caso, como `on`. `getsebool` permite que você veja o valor do booleano.

Definindo contextos de arquivo do SELinux para o Samba

O SELinux confina os arquivos que o serviço Samba pode acessar. Em vez de permitir que qualquer arquivo com permissões apropriadas de leitura e gravação sejam compartilhados pelo servidor Samba, o SELinux (quando no modo Enforcing) exige que os arquivos e diretórios tenham os contextos de arquivo corretos configurados sobre eles antes de o serviço Samba poder até mesmo ver que os arquivos existem.

Para o serviço Samba funcionar com o SELinux imediatamente, alguns arquivos e diretórios já são fornecidos com os contextos de arquivo apropriados. Por exemplo, os arquivos de configuração (`/etc/samba/*`), os arquivos de log (`/var/log/samba/*`) e as bibliotecas

(`/var/lib/samba/*`) do Samba têm regras atribuídas para assegurar que o serviço obtenha os contextos de arquivo apropriados. Para localizar arquivos e diretórios associados com o serviço do Samba e o daemon `smbd` que possuam contextos de arquivo pré-configurados, execute o seguinte:

```
# semanage fcontext -l | grep -i samba
# semanage fcontext -l | grep -i smb
```

A parte contexto de arquivo em que você está interessado termina com `_t`: por exemplo, `samba_etc_t`, `samba_log_t` e `smbd_var_t`, para os diretórios `/etc/samba`, `/var/log/samba` e `/var/lib/samba`, respectivamente.

Você pode achar que precisa mudar contextos de arquivos — por exemplo, quando coloca arquivos em locais não padrão (como mover o arquivo `smb.conf` para `/root/smb.conf`) ou quando quer compartilhar um diretório (que não os diretórios iniciais, que podem ser ativados configurando um booleano). Ao contrário dos servidores `vsftpd` (FTP) e `httpd` (web) que vêm com o Linux, o Samba não tem diretórios padrão de conteúdo compartilhado (aqueles que acabamos de mencionar usavam `/var/ftp` e `/var/www/html`).

Você pode alterar um contexto de arquivo permanentemente criando uma nova regra de contexto de arquivo e, então, aplicando essa regra ao arquivo ou diretório para o qual ele se destina. Você pode fazer isso com o comando `semanage` (para criar a regra) e o comando `restorecon` (para aplicar a regra). Por exemplo, se quiser compartilhar um diretório, `/mystuff`, você deve criá-lo com as permissões adequadas e executar o seguinte comando para torná-lo disponível para acesso de leitura/gravação a partir do Samba:

```
# semanage fcontext -a -t samba_share_t "/mystuff(/.*)?"
# restorecon -v /mystuff
```

Depois que esses comandos são executados, o diretório `/mystuff`, juntamente com todos os arquivos e diretórios abaixo desse ponto, tem o contexto de arquivo `samba_share_t`. Então, cabe a você atribuir a posse e as permissões de arquivo corretas para permitir acesso aos usuários que você escolher. A seção “Configurando o Samba” fornece um exemplo de como criar uma ação e mostra como adicionar permissões e posse a um diretório compartilhado usando comandos padrão do Linux.

Configurando permissões de host/usuário do Samba

Dentro do arquivo `smb.conf`, você pode permitir ou restringir acesso ao servidor Samba inteiro ou a compartilhamentos específicos com base nos hosts ou usuários que tentam obter acesso. Você também pode restringir o acesso ao servidor Samba fornecendo o serviço apenas em placas de redes específicas.

Por exemplo, se você tem uma placa de rede conectada à internet e outra conectada à rede local, você pode dizer para o Samba atender as solicitações somente na placa de rede local. A próxima seção descreve como configurar o Samba, incluindo a forma de identificar quais hosts, usuários ou placas de rede podem acessar seu servidor Samba.

Configurando o Samba

A maior parte da configuração do Samba é feita no arquivo `/etc/samba/smb.conf`. Conforme você configura o servidor Samba, também pode adicionar informações sobre hosts e sobre usuários aos arquivos `lmhosts` e `smbusers`, respectivamente.

Para configurar o `smb.conf`, você pode usar um editor de texto simples. Mas o Samba vem com uma ferramenta baseada na web que você pode obter instalando o pacote `sambaswat`. E o Fedora tem uma interface gráfica de configuração do Samba (`system-config-samba`). As próximas seções ensinam passo a passo como usar as interfaces gráficas para configurar o Samba. Depois disso, examine o arquivo `smb.conf` resultante.

Utilizando `system-config-samba`

Nos sistemas Fedora, você pode instalar o pacote `system-config-samba` para usar a janela Samba Server Configuration para configurar o Samba. A partir dessa janela, você pode configurar as definições básicas do servidor, adicionar compartilhamentos e criar contas de usuário para que as pessoas possam acessar esses compartilhamentos. Para instalar `system-config-samba`, digite o seguinte como root na linha de comando:

```
# yum install system-config-samba
```

Com `system-config-samba` instalado, você pode iniciá-lo no Fedora, a partir da página Applications, clicando duas vezes no ícone Samba. Ou você pode simplesmente digitar o seguinte no shell:

```
# system-config-samba &
```

A janela Samba Server Configuration deve aparecer em seu desktop. Agora, você pode configurar o servidor Samba básico e começar a adicionar diretórios compartilhados.

Escolhendo as configurações do servidor Samba

A partir da janela de configuração do servidor Samba, você deve primeiro definir as configurações básicas e o tipo de autenticação que está usando. Comece selecionando Preferences ⇒ Server Settings. A janela pop-up Server Settings aparece, como mostrado na Figura 19.1.

ha Basic Samba server settings.

Nesse exemplo, o nome Workgroup é alterado para datagroup e Sales Data é configurado como a Description. Então, selecione a guia Security. O modo de autenticação padrão é User, o que significa que os nomes de usuário Samba (Windows) são mapeados para contas de usuários reais do sistema Linux. Nenhum sistema externo é contatado para autenticar usuários.

Senhas criptografadas são usadas por padrão. Também por padrão, não há nenhuma conta de convidado. Se você quiser ter uma conta de convidado não autenticado que possa acessar seu servidor Samba, clique na caixa que diz No guest account e escolha qual usuário do Linux é atribuído ao usuário convidado. Por exemplo, se você atribuiu joe à conta de convidado, a conta do usuário joe precisa ter acesso a um diretório ou arquivo compartilhado para um usuário convidado acessá-lo. A Figura 19.2 mostra um exemplo das configurações de segurança.

ha as configurações de segurança do servidor Samba.

Configurando contas de usuários do Samba

Se estiver usando o tipo de autenticação de usuário padrão, como estamos aqui, você pode configurar contas de usuário para acessar o sistema Fedora selecionando Preferences ⇒ Samba Users. A princípio, não há usuários do Samba listados. Para adicionar um usuário do Samba, selecione Add User a partir na janela Samba Users.

Na tela Create New Samba User que aparece, selecione a conta de usuário local Linux (nome de usuário UNIX) a partir da caixa drop-down. Você pode usar o mesmo nome ou um diferente para Windows Username. Então, preencha a senha do Samba para o usuário e selecione OK. A Figura 19.3 mostra como criar um novo usuário do Samba chamado `chris` que tem a mesma permissão que o usuário Linux `chris` tem para acessar o sistema.

one contas de usuário ao seu servidor Samba.

Você pode configurar quantos usuários quiser a partir de suas contas de usuário Linux para conceder-lhes o acesso a arquivos e diretórios por meio de seu servidor Samba.

Com as configurações básicas do sistema e do usuário no lugar, você pode começar a criar diretórios compartilhados.

Criando uma pasta compartilhada Samba

Antes que você possa criar uma pasta compartilhada, a pasta (diretório) deve existir e ter o conjunto de permissões apropriado. Neste exemplo, o diretório `/var/salesdata` está compartilhado. Queremos que os dados sejam graváveis pelo usuário chamado `chris`, mas visíveis para qualquer pessoa na nossa rede. Para criar esse diretório e configurar as permissões e contextos SELinux adequados, digite o seguinte, como usuário `root`:

```
# mkdir /var/salesdata
# chmod 775 /var/salesdata
# chown chris:chris /var/salesdata
# semanage fcontext -a -t samba_share_t /var/salesdata
# restorecon -v /var/salesdata
```

Então, a partir da janela de configuração do servidor Samba, selecione `File ⇒ Add Share`. A janela `Create Samba Share` aparece. Comece preenchendo o nome da pasta (selecione `Choose` para procurá-la), um nome para representar a ação e uma descrição. Então, você deve escolher se quer ou não deixar que alguém possa gravar nela e se quer ou não que a existência do compartilhamento seja visível para qualquer pessoa. A Figura 19.4 mostra um exemplo da janela.

ha a pasta e o nome do novo compartilhamento.

Depois de preencher as informações básicas, selecione a guia Access. Qualquer usuário do Samba que você criou até este ponto aparecerá na guia Access. Coloque uma marca de seleção ao lado de qualquer usuário ao qual você quer dar acesso a esse diretório compartilhado. Como eu atribuí a posse a `chris` e deixei abertas as permissões de leitura/gravação para ele, o usuário `chris` será capaz de ler e gravar no diretório por meio do Samba. A Figura 19.5 mostra um exemplo das configurações de acesso para o compartilhamento.

gure o acesso ao novo compartilhamento.

Selecione OK quando terminar de configurar o compartilhamento. Em seguida, você deve testar se o compartilhamento está disponível.

Verificando o compartilhamento Samba

Para que as alterações na configuração do Samba tenham efeito, é necessário reiniciar o serviço `smb`. Uma vez feito isso, verifique se o compartilhamento Samba que você criou está disponível e se qualquer usuário que você atribuiu ao compartilhamento pode acessá-lo. Para fazer essas coisas, digite o seguinte, como usuário `root`, a partir de um shell no servidor Samba:

```
# systemctl restart smb.service
# smbclient -L localhost
Enter root's password: <ENTER>
Anonymous login successful
Domain=[DATAGROUP] OS=[Unix] Server=[Samba 3.6.5-85.fc16]
  Sharename      Type            Comment
  -----
  salesdata      Disk            Sales data for current year
  IPC$           IPC             IPC Service (Sales Data)
Anonymous login successful
Domain=[DATAGROUP] OS=[Unix] Server=[Samba 3.6.5-85.fc16]
...
```

Aqui você pode ver o nome do compartilhamento (`salesdata`), o domínio definido para o nome de grupo de trabalho `DATAGROUP` e a descrição inserida anteriormente (`Sales Data`). Em seguida, uma maneira rápida de testar o acesso ao compartilhamento é usar o comando `smbclient`. Você pode usar o nome ou endereço IP com `smbclient` para acessar o compartilhamento. Como estamos no sistema local, nesse exemplo, é só usar o nome `localhost` e o usuário que adicionamos (`chris`):

```
# smbclient -U chris //localhost/salesdata
Enter chris's password: *****
Domain=[DATAGROUP] OS=[Unix] Server=[Samba 3.6.5-85.fc16]
smb: \> lcd /etc

smb: \> put hosts
putting file hosts as \hosts (43.5 kb/s) (average 43.5 kb/s)
smb: \> ls
.                D          0   Sat May 26 09:52:51 2012
..               D          0   Sat May 26 09:11:50 2012
hosts            A          89   Sat May 26 09:52:51 2012
39941 blocks of size 524288. 28197 blocks available
smb: \> quit
```

Um compartilhamento do Samba aparece na forma `//host/share` ou `\\host\share`. Mas quando você identifica um compartilhamento do Samba a partir de um shell do Linux no último caso, as barras invertidas precisam ser “escapadas”. Então, como um argumento, o primeiro exemplo do compartilhamento teria de aparecer como `\\\\localhost\\salesdata`. Logo se vê, portanto, que a primeira forma é mais fácil de usar.

ta

“escapar” um caractere que você digita a partir do shell, você coloca uma barra invertida (\) na frente do caractere. Isso instrui o shell a usar o caractere após a barra invertida literalmente, em vez de dar ao caractere um significado especial para o shell. (Os caracteres * e ? são exemplos de caracteres com significado especial.) Como a barra invertida tem um significado especial para o shell, se você quiser usar uma barra invertida literalmente, precisa escapá-la com uma barra invertida. É por isso que, quando você quer digitar um endereço do Samba que inclui duas barras invertidas, você precisa, na verdade, colocar quatro barras invertidas.

Quando solicitado, digite a senha do Samba para esse usuário (que pode ser diferente da senha do usuário do Linux). Você vê o prompt `smb: \>` depois disso.

Nesse ponto, você tem uma sessão aberta para o host Samba que é semelhante a uma sessão de `lftp` para navegar por um servidor FTP. O comando `lcd /etc` torna `/etc` o diretório atual no sistema local. O comando `put hosts` copia o arquivo `hosts` do sistema local para o diretório compartilhado. Digitar `ls` mostra que o arquivo existe no servidor. `quit` termina a sessão.

Em vez de configurar o servidor, os usuários e os compartilhamentos com a janela `system-config-samba`, você poderia usar a interface `samba-swat` baseada na web ou editar o arquivo `smb.conf` diretamente. A próxima seção mostra como usar o arquivo `smb.conf` diretamente para configurar o Samba.

Configurando o Samba no arquivo `smb.conf`

Dentro de `/etc/samba/smb.conf` estão ajustes para configurar o servidor Samba, definir impressoras compartilhadas, configurar como a autenticação é feita e criar diretórios compartilhados. O arquivo consiste nas seguintes seções pré-definidas:

- **[global]** — Configurações que se aplicam ao servidor Samba como um todo são colocadas nesta seção. Aqui é onde você configura a descrição do servidor, seu grupo de trabalho (domínio), a localização dos arquivos de log, o tipo padrão de segurança e outras configurações.
- **[homes]** — Esta seção determina se ou não os usuários com contas no servidor Samba podem ver seus diretórios iniciais (`browseable`) ou gravar neles (`writable`).
- **[printers]** — Nesta seção, as configurações dizem ao Samba se ele deve ou não tornar disponíveis as impressoras que estão configuradas para impressão pelo Linux (CUPS).

Dentro do arquivo `smb.conf` também há várias seções cujas linhas estão desativadas com o caractere de comentário ponto e vírgula (;). Remover os pontos e vírgulas permite configurar rapidamente diferentes tipos de informações compartilhadas. A outra informação ilustrada no arquivo `smb.conf` nas próximas seções reflete a mesma configuração do Samba que adicionamos durante o processo de `system-config-samba`.

Tenha em mente que não é necessário usar qualquer interface gráfica ao configurar o Samba. Você pode simplesmente editar o arquivo `smb.conf` diretamente, usando as técnicas descritas nesta seção. Editar

`smb.conf` diretamente é a forma mais comum de configurar o Samba em um servidor Red Hat Enterprise Linux que não tem interface de desktop gráfica instalada.

Configurando a seção [global]

Se examinar a seção [global] do arquivo `smb.conf`, você pode ver que algumas configurações foram modificadas e outras estão desativadas com caracteres de comentário, prontas para você modificá-las como quiser:

```
[global]
    workgroup = datagroup
    server string = Sales Data
;    netbios name = MYSERVER
;    interfaces = lo eth0 192.168.12.2/24 192.168.13.2/24
;    hosts allow = 127. 192.168.12. 192.168.13.
    log file = /var/log/samba/log.%m
    max log size = 50
    security = user
    cups options = raw
```

`workgroup` (também usado como o nome de domínio) está configurado como `datagroup` nesse exemplo. Quando um cliente se comunica com o servidor Samba, esse nome informa ao cliente o grupo de trabalho em que o servidor Samba está. Qualquer valor configurado como `server string` é utilizado mais tarde para preencher valores de comentário apresentadas ao lado das linhas de IPC e na caixa de comentário da impressora quando essa informação é apresentada para aplicativos clientes.

Por padrão, o `hostname` do seu servidor DNS (digite **hostname** para ver o que ele é) também é usado como o nome NetBIOS de seu servidor Samba. Você pode substituir isso e configurar um nome NetBIOS separado simplesmente removendo o caractere de comentário da linha `netbios name` e adicionando o nome do servidor que você quiser. Por exemplo: `netbios name = myownhost.localhost` é usado como o nome NetBIOS se ele não for configurado de outra forma.

Como o valor de `log file` está configurado, logs associados à atividade do Samba são gravados em `/var/log/samba/log.%m`, onde `%m` representa o nome ou endereço IP do sistema que está contatando o servidor Samba. Isso torna mais fácil depurar problemas relacionados com solicitações de um determinado sistema, pois cada cliente tem seu próprio arquivo de log atribuído.

O `max log size` limita o tamanho de cada arquivo de log a um determinado número de kilobytes (50KB por padrão). Uma vez que o tamanho tenha sido excedido, o Samba copia o arquivo para um arquivo de log de mesmo nome com `.old` anexado a ele. O tipo de `security` padrão é configurado como `cups options` (nomes de usuário e senhas do Samba) e `cups options` deixa você passar quaisquer opções que quiser para as impressoras CUPS servidas pelo servidor Samba. Por padrão, apenas `raw` está configurado, o que permite que clientes do Windows usem seus próprios drivers de

impressão. Impressoras em seu servidor Samba imprimem as páginas em que são apresentadas na forma genérica (raw).

Há várias opções desativadas com um caractere de comentário que você pode considerar configurar. Configurar `netbios name` faz com que o nome configurado seja usado como o nome do host para o serviço Samba. Se isso não for configurado, a parte host do nome do sistema de domínio totalmente qualificado é usada como o hostname.

Se quiser restringir o acesso ao servidor Samba para que ele só responda em certas placas de rede, você pode remover o comentário da linha `interfaces` e adicionar o endereço IP ou nome (`lo`, `eth0`, `eth1` etc.) das placas de rede que você quer.

Você também pode restringir o acesso ao servidor Samba para hosts específicos. Ative a linha `hosts allow` (removendo o ponto e vírgula) e insira os endereços IP dos hosts que você quer permitir. Para inserir um intervalo de endereços, simplesmente termine a parte sub-rede do endereço, seguida por um ponto. Por exemplo, `127.` está associado a endereços IP que apontam para o host local. A entrada `192.168.12.` corresponde a todos os endereços IP de `192.168.12.1` a `192.168.12.254`.

Configurando a seção [homes]

A seção `[homes]` está configurada, por padrão, para permitir que qualquer conta de usuário Samba possa acessar seu próprio diretório inicial através do servidor Samba. A configuração `browseable = no` impede que o servidor Samba exiba a disponibilidade dos diretórios compartilhados. Usuários que podem fornecer seu próprio nome de usuário e senha do Samba serão capazes de ler e gravar em seu próprio diretório inicial (`writable = yes`). Eis como se parece a entrada `homes` padrão:

```
[homes]
        comment = Home Directories
        browseable = no
        writable = yes
;       valid users = %S
;       valid users = MYDOMAIN\%S
```

Observe que dois exemplos das entradas `valid users` estão desativados com caracteres de comentário. Com esse valor não configurado (como está por padrão), qualquer usuário válido pode fazer login no Samba. Configurá-lo como `%S` substitui o nome do serviço atual, o que permite qualquer usuário válido do serviço. Você também pode limitar o acesso do usuário, indicando que apenas um determinado grupo de trabalho (domínio) pode ser usado para identificar usuários que solicitam esse serviço.

Se, depois de iniciar o serviço `smb`, você não for capaz de fazer login usando uma conta de usuário válida, você pode precisar mudar alguns aspectos da segurança em seu sistema. Em sistemas Fedora e RHEL, em particular, recursos do SELinux precisam ser alterados para permitir que os usuários acessem seus diretórios iniciais, se você estiver no modo Enforcing do SELinux.

Por exemplo, se você tentasse usar o `smbclient` para fazer login em seu diretório inicial, o login teria sucesso, mas quando você tentasse listar o conteúdo do diretório, poderia ver a seguinte mensagem:


```
NT_STATUS_ACCESS_DENIED listing \*
```

A fim de dizer para o SELinux permitir que usuários do Samba acessem seus diretórios iniciais como compartilhamentos Samba, você precisa ativar os booleanos `samba_enable_home_dirs` digitando o seguinte, como root, a partir de um shell:

```
# setsebool -P samba_enable_home_dirs on
```

O comando `setsebool` ativa a capacidade do Samba de compartilhar diretórios (o que está desativado por padrão). A forma para usar o comando `smbclient` a fim de verificar o acesso ao diretório inicial do usuário, novamente para o usuário `chris`, seria a seguinte (substituindo o endereço IP pelo nome ou o endereço de seu servidor Samba):

```
$ smbclient -U chris //192.168.0.119/chris
Enter chris's password:
Domain=[DATAGROUP] OS=[Unix] Server=[Samba 3.6.5-85.fc16]
smb: \> ls file.txt
file.txt                               149946368 Sun Apr 8 09:28:53 2012
39941 blocks of size 524288. 28191 blocks available
```

O principal ponto a lembrar é que, mesmo que o compartilhamento não seja navegável, você pode solicitá-lo fornecendo o nome ou endereço IP do servidor Samba, seguido do nome do usuário (aqui, `chris`), para acessar o diretório inicial do usuário.

Configurando a seção [printers]

Qualquer impressora que você configure para impressão CUPS em seu sistema Linux é automaticamente compartilhada com outras no Samba, com base na seção `[printers]`, que é adicionada por padrão. A configuração global `cups options = raw` torna todas as impressoras genéricas (o que significa que o cliente Windows precisa fornecer o driver de impressora para cada impressora compartilhada).

Eis como se parece a seção `printers` padrão no arquivo `smb.conf`:

```
[printers]
                                comment = All Printers
                                path = /var/spool/samba
                                browseable = no
;                                guest ok = no
;                                writable = No
                                printable = yes
```

`printable = yes` faz com que todas as suas impressoras CUPS no sistema local sejam compartilhadas pelo Samba. Impressoras são graváveis e permitem a impressão de convidados por padrão. Você pode remover o caractere de comentário da linha `guest ok = no` e da linha `writable = No`, respectivamente, para alterar essas configurações.

Para ver as impressoras que estão disponíveis, você pode executar o comando `smbclient -L` a partir de um sistema Linux, como mostrado anteriormente. Em um sistema Windows, você pode selecionar Rede (Network) na janela do gerenciador de arquivos do Windows Explorer e selecionar o ícone que representa seu servidor Samba. Todas as impressoras e pastas compartilhadas aparecem nessa janela. (Consulte a seção “Acessando compartilhamentos Samba” neste capítulo para obter detalhes sobre como visualizar e usar impressoras compartilhadas.)

Criando diretórios compartilhados personalizados

Com a configuração básica de seu servidor Samba no lugar, você pode começar a criar seções personalizadas para compartilhar impressoras e pastas específicas e protegê-las como preferir. Para o primeiro exemplo, eis como se parece o compartilhamento criado na demonstração de `system-config-samba` no início deste capítulo (chamado `salesdata`) no arquivo `smb.conf`:

```
[salesdata]
comment = Sales data for current year
path = /var/salesdata
read only = no
; browseable = yes
valid users = chris
```

Antes de criar esse compartilhamento, o diretório `/var/salesdata` foi criado, com `chris` atribuído como o usuário e grupo, e o diretório foi configurado para ser legível e gravável por `chris`. (O contexto de arquivo do SELinux também deverá ser configurado se o SELinux estiver no modo Enforcing.) O nome de usuário Samba `chris` deve ser apresentado junto com a respectiva senha para acessar o compartilhamento. Uma vez que `chris` está conectado ao compartilhamento, `chris` tem acesso de leitura e gravação nele (`read only = no`).

Agora que você já viu as configurações padrão para o Samba e um exemplo de um diretório simples compartilhado (pasta), leia as próximas seções para ver como configurar ainda mais os compartilhamentos. Em particular, os exemplos demonstram como disponibilizar compartilhamentos para determinados usuários, hosts e placas de rede.

Restringindo o acesso ao Samba por placa de rede

Para restringir o acesso a todos os seus compartilhamentos, você pode definir a configuração global `interfaces` no arquivo `smb.conf`. O Samba é mais projetado para o compartilhamento de arquivos locais do que para o compartilhamento sobre redes remotas. Se seu computador tem uma placa de rede conectada a uma rede local e uma conexão com a internet, considere permitir o acesso apenas à rede local.

Para configurar as interfaces que Samba ouve, ative a linha `interfaces` na seção `[global]` do arquivo `smb.conf` removendo seu caractere de comentário. Então, adicione os nomes de placa de rede ou intervalos de endereços IP dos computadores aos quais você quer permitir acesso a seu computador. Eis um exemplo:

```
interfaces = lo 192.168.22.15/24
```

Essa entrada `interfaces` permite acesso ao serviço Samba para todos os usuários no sistema local (`lo`). Ela também permite acesso a todos os sistemas na rede `192.168.22`. Veja a descrição da página `man de smb.conf` sobre as diferentes formas de identificar hosts e placas de rede.

Restringindo o acesso ao Samba por host

O acesso de host ao servidor Samba pode ser configurado para todo o serviço ou por compartilhamentos individuais. A sintaxe usada é semelhante à de `hosts.allow` e `hosts.deny` no recurso TCP wrappers. Aqui, porém, as entradas `hosts allow` e `hosts deny` são adicionadas diretamente ao arquivo `smb.conf`.

Eis alguns exemplos das entradas `hosts allow` e `hosts deny`:

```
hosts allow = 192.168.22. EXCEPT 192.168.22.99
hosts allow = 192.168.5.0/255.255.255.0
hosts allow = .example.com market.example.net
hosts deny = evil.example.org 192.168.99.
```

Essas entradas podem ser colocadas na seção `[global]` ou em qualquer seção de diretório compartilhado. O primeiro exemplo permite acesso a qualquer máquina na rede `192.168.22`, exceto `192.168.22.99`, a qual é negado. Note que um ponto é necessário no fim do número da rede. O exemplo `192.168.5.0/255.255.255.0` usa a notação de máscara de rede para identificar `192.168.5` como o conjunto de endereços que são permitidos.

Na terceira linha do código de exemplo, qualquer host da rede `.example.com` é permitido, assim como é o host `market.example.net` individual. O exemplo `hosts deny` mostra que você pode usar a mesma forma para identificar nomes e endereços IP a fim de impedir o acesso de certos hosts.

Restringindo o acesso ao Samba por usuário

Usuários e grupos particulares do Samba podem ter acesso a compartilhamentos específicos do Samba identificando esses usuários e grupos dentro de um compartilhamento no arquivo `smb.conf`. Além de usuários convidados, os quais você pode ou não permitir, a autenticação de usuário padrão para o Samba requer que você adicione uma conta de usuário do Samba (Windows) que mapeia para uma conta de usuário local no Linux.

Para permitir que um usuário acesse o servidor Samba, você precisa criar uma senha para o usuário. Eis um exemplo de como adicionar uma senha do Samba ao usuário `jim`:

```
# smbpasswd -a jim
New SMB password: ****
Retype new SMB password: ****
```

Depois de executar o comando `smbpasswd`, `jim` pode usar esse nome de usuário e senha para acessar o servidor Samba. O arquivo `/var/lib/samba/private/passdb.tdb` armazena a senha

inserida para `jim`. Depois disso, o usuário `jim` pode alterar a senha simplesmente digitando `smbpasswd` quando ele estiver conectado. O usuário `root` pode alterar a senha executando novamente o comando mostrado no exemplo, mas descartando a opção `-a`.

Se quiser dar acesso a um compartilhamento de `jim`, você pode adicionar uma linha de `valid users` a esse bloco compartilhado no arquivo `smb.conf`. Por exemplo, para fornecer tanto a `chris` como a `jim` acesso a um compartilhamento, você pode adicionar a seguinte linha:

```
valid users = jim, chris
```

Se a opção `read only` estiver configurada como `no` para o compartilhamento, os usuários poderiam gravar arquivos no compartilhamento (dependendo das permissões de arquivo). Se `read only` estiver configurado como `yes`, você ainda pode permitir acesso a `jim` e `chris` para gravar arquivos, adicionando uma linha `write list` da seguinte maneira:

```
write list = jim, chris
```

A lista de gravação pode conter grupos (isto é, grupos do Linux contidos no arquivo `/etc/group`) para dar permissão de gravação a qualquer usuário Linux que pertença a um determinado grupo Linux. Você pode adicionar permissão de gravação a um grupo, colocando um sinal de mais (+) na frente de um nome. Por exemplo, a linha seguir adiciona acesso de gravação ao grupo `market` para o compartilhamento a que esta linha está associada:

```
write list = jim, chris, +market
```

Há muitas maneiras de mudar e estender os recursos de compartilhamento do Samba. Para mais informações sobre a configuração do Samba, não deixe de examinar o arquivo `smb.conf` (que inclui muitos comentários úteis) e a página `man` do `smb.conf`.

Acessando compartilhamentos do Samba

Depois de ter criado alguns diretórios compartilhados no Samba, há muitas ferramentas de cliente disponíveis no Linux e no Windows para acessar esses compartilhamentos. Ferramentas de linha de comando no Linux incluem o comando `smbclient`, demonstrado anteriormente neste capítulo. Para meios gráficos de acessar compartilhamentos, você pode usar os gerenciadores de arquivos disponíveis no Windows (Windows Explorer) e Linux (Nautilus, com o desktop GNOME).

Acessando compartilhamentos do Samba no Linux

Abrir um gerenciador de arquivos no Linux pode fornecer-lhe acesso aos diretórios compartilhados do Linux (Samba) e do Windows (SMB). Você acessa o gerenciador de arquivos de forma diferente em diferentes desktops Linux. No GNOME 3, você pode clicar no ícone Files. Em outros desktops, abra a pasta Home.

Com o gerenciador de janelas Nautilus exibido, procure uma seleção Browse Network no painel esquerdo ou selecione Go ⇒ Network no menu. Os servidores de rede disponíveis na rede local devem aparecer, como mostrado na Figura 19.6.

Com o servidor Samba exibido (chamado `MYSERVER` aqui), dê um duplo clique no ícone desse servidor. Quaisquer compartilhamentos navegáveis (`browseable`) devem aparecer na janela. Dê um clique duplo em um compartilhamento. Se ele for acessível a `guest`, o compartilhamento será aberto. Se a autenticação do usuário for necessária, adicione o nome de usuário e senha, conforme solicitado.

use compartilhamentos do Samba a partir do Nautilus.

Se o usuário e a senha forem aceitos, você deve ver o conteúdo do diretório remoto. Se tiver acesso de gravação ao compartilhamento, você pode abrir outra janela do Nautilus e arrastar e soltar arquivos entre os dois sistemas.

Se uma parte não for navegável, você pode digitar o endereço do compartilhamento na janela do Nautilus. Escolha **File** ⇒ **Connect to Server** e selecione **Windows Share** como o **Type**. Preencha as informações necessárias para se conectar ao compartilhamento. Na Figura 19.7, o usuário `chris` tenta se conectar ao diretório inicial no servidor Samba `MYSERVER`.

do compartilhamentos do Samba a partir de Connect to Server no Nautilus.

Nesse ponto, o diretório inicial de `chris` deve aparecer no Nautilus do cliente Linux. Dependendo das permissões de acesso, você deve ser capaz de arrastar e soltar arquivos entre o sistema local e o servidor Samba.

Uma vez que um diretório compartilhado do Samba pode ser visto como um sistema de arquivos remoto, você pode usar as ferramentas do Linux para se conectar a um compartilhamento do Samba (temporária ou permanentemente) para seu sistema Linux. Usando o comando `mount` padrão, você pode montar um compartilhamento Samba remoto como um sistema de arquivos CIFS em Linux. Esse exemplo monta o compartilhamento `salesdata` do host no endereço IP `192.168.0.119` sobre o diretório local `/mnt/sales`:

```
# mkdir /mnt/sales
# mount -t cifs -o user=chris //192.168.0.119/salesdata /mnt/sales
Password: *****
# ls /mnt/sales
hosts services
```

Quando solicitado, digite a senha do Samba para `chris`. Dado que o usuário `chris` nesse exemplo tem permissão de leitura-gravação sobre o diretório compartilhado, usuários em seu sistema devem ser capazes de ler e gravar no diretório montado. Independentemente de quem salva os arquivos no diretório compartilhado, no servidor esses arquivos pertencerão ao usuário `chris`. Essa montagem dura até que o sistema seja reiniciado ou que você execute o comando `umount` no diretório. Se quiser que o compartilhamento seja montado permanentemente (ou seja, cada vez que o sistema inicializa) no mesmo local, você pode fazer uma configuração adicional. Primeiro, abra o arquivo `/etc/fstab` e adicione uma entrada semelhante à seguinte:

```
//192.168.0.119/salesdata /mnt/sales cifs credentials=/root/cif.txt
0 0
```

Então, crie um arquivo de credenciais (neste exemplo, `/root/cif.txt`). Nesse arquivo, coloque o nome do usuário e a senha que você quer apresentar quando o sistema tenta montar o sistema de arquivos. Eis um exemplo do conteúdo desse arquivo:

```
user=chris
pass=a
```

Antes de reiniciar para verificar se a entrada está correta, tente montá-la a partir da linha de comando. O comando `mount -a` tenta montar qualquer sistema de arquivos listados no arquivo `/etc/fstab` que

ainda não está montado. O comando `df` mostra informações sobre espaço em disco para o diretório montado. Por exemplo:

```
# mount -a
# df -h /mnt/sales
Filesystem                Size  Used Avail  Ues% Mounted on
//192.168.0.119/salesdata 20G  5.7G   14G   30% /mnt/sales
```

Agora você deve ser capaz de usar o diretório Samba compartilhado como você usa qualquer diretório no sistema local.

Acessando compartilhamentos do Samba no Windows

Tal como acontece com o Linux, você pode acessar compartilhamentos do Samba a partir da janela do gerenciador de arquivos, neste caso, o Windows Explorer. Para fazer isso, abra qualquer pasta no Windows e selecione Rede no painel esquerdo. Um ícone que representa o servidor Samba deve aparecer na tela. Clique nesse ícone e digite uma senha, se solicitado a inserir uma. Você deverá ver todas as impressoras e pastas compartilhadas do servidor (ver Figura 19.8).

usando compartilhamentos do Samba a partir do Windows.

Na Figura 19.8, você pode ver que há duas pastas compartilhadas (diretórios): `chris` e `salesdata`. Também há várias impressoras compartilhadas. Para usar as pastas, clique duas vezes nelas e insira as informações de autenticação necessárias. Como as impressoras são configuradas para usar os drivers genéricos (raw drivers) por padrão, você precisará obter os drivers do Windows para usar qualquer uma das impressoras do Samba.

Usando o Samba na empresa

Embora fora do escopo deste livro, o compartilhamento de arquivos e impressoras Windows através de servidores Samba é uma aplicação muito popular em grandes empresas. Apesar de o Linux ter crescido enormemente no mercado de servidores corporativos, os sistemas Microsoft Windows ainda são os sistemas desktop predominantes utilizados no mundo corporativo.

Os principais recursos necessários para integrar servidores Samba em uma grande empresa com muitos desktops Microsoft Windows estão relacionados com a autenticação. A maioria das grandes empresas usa servidores Microsoft Active Directory (ADS) para autenticação. No lado do Linux, isso significa configurar Kerberos sobre o sistema Linux e usar o ADS (em vez de usuário) para o tipo de segurança no arquivo `smb.conf`.

A vantagem da autenticação central é que os usuários têm de se lembrar de apenas um conjunto de credenciais por toda a empresa e os administradores de sistemas precisam gerenciar menos contas de usuário e senhas. Se você estiver interessado em investigar mais o tema, recomendo que leia a página Samba & Active Directory no wiki em Samba.org:

http://wiki.samba.org/index.php/Samba_&_Active_Directory

Resumo

Por causa da popularidade de desktops Windows, servidores Samba tornaram-se populares para compartilhamento de arquivos e impressoras entre sistemas Windows e Linux. O Samba fornece um modo de interagir com sistemas Windows por meio do aplicativo Server Message Block (SMB) ou o protocolo Common Internet File (CIFS) para o compartilhamento de recursos através de uma rede.

Este capítulo percorreu o processo de instalar, inicializar, proteger, configurar e acessar servidores Samba em um sistema Linux. Tanto as ferramentas gráficas como as de linha de comando podem ser usadas para configurar um servidor Samba e chegar a ele a partir de sistemas Linux e Windows.

O próximo capítulo descreve o recurso Network File System (NFS). O NFS é um recurso nativo do Linux

Exercícios

Os exercícios desta seção descrevem tarefas relacionadas a configurar um servidor Samba no Linux e acessar o servidor usando um cliente Samba. Como de costume, muitas vezes há várias maneiras de realizar algumas das tarefas aqui. Então, não se preocupe se você não resolver os exercícios exatamente da mesma maneira como se faz nas respostas, desde que você obtenha os mesmos resultados. Consulte o Apêndice B para obter as soluções sugeridas.

Não faça esses exercícios em um sistema Linux rodando um servidor Samba, porque eles quase certamente irão interferir nesse servidor. Esses exercícios foram testados em um sistema Fedora. Alguns dos passos podem ser ligeiramente diferentes em outro sistema Linux.

1. Instale os pacotes `samba`, `samba-client` e `samba-doc`.
2. Inicie e habilite os serviços `smb` e `nmb`.
3. Configure o grupo de trabalho do servidor Samba como `TESTGROUP`, o `netbios name` como `MYTEST` e a string do servidor como `Samba Test System`.
4. Adicione um usuário Linux chamado `phil` ao seu sistema e adicione uma senha Linux e uma senha do Samba a `phil`.
5. Defina a seção `[homes]` de modo que os diretórios iniciais sejam navegáveis (`browseable=yes`) e graváveis (`writable=yes`) e `phil` seja o único usuário válido.
6. Defina qualquer booleano do SELinux que seja necessário para fazer `phil` acessar o diretório inicial dele por meio de um cliente Samba.
7. A partir do sistema local, use o comando `smbclient` para listar que o compartilhamento `homes` está disponível.
8. A partir de uma janela do Nautilus (gerenciador de arquivos) no sistema local, conecte-se ao compartilhamento `homes` para o usuário `phil` no servidor Samba local de uma forma que permita que você arraste e solte arquivos para essa pasta.
9. Abra o firewall de modo que qualquer pessoa que tenha acesso ao servidor possa acessar o serviço Samba (daemons `smbd` e `nmbd`).
10. A partir de outro sistema em sua rede (Windows ou Linux) tente abrir o compartilhamento `homes` novamente como o usuário `phil` e novamente se certifique de que você pode arrastar e soltar arquivos para ele.

Configurando um servidor de arquivos NFS

NESTE CAPÍTULO

Obtendo software de servidor NFS

Ativando e iniciando o NFS

Exportando diretórios NFS

Configurando recursos de segurança para o NFS

Montando diretórios NFS compartilhados remotos

Em vez de representar os dispositivos de armazenamento como letras de unidade (A, B, C etc.), como ocorre em sistemas operacionais Microsoft, o Linux conecta-se transparentemente a sistemas de arquivos a partir de vários discos rígidos, disquetes, CD-ROMs e outros dispositivos locais para formar um sistema de arquivos Linux único. O recurso Network File System (NFS) permite estender seu sistema de arquivos Linux para conectar sistemas de arquivos em outros computadores na sua estrutura de diretórios local.

Um servidor de arquivos NFS fornece uma maneira fácil de compartilhar grandes quantidades de dados entre usuários e computadores em uma organização. Um administrador de um sistema Linux que é configurado para compartilhar seus sistemas de arquivos usando NFS tem de realizar as seguintes tarefas, para configurar NFS:

1. **Configurar a rede.** O NFS é normalmente usado em redes locais privadas, em oposição às redes públicas, tais como a internet.
2. **Iniciar o serviço NFS.** Vários daemons de serviço precisam ser inicializados e executados para ter um pleno funcionamento do serviço NFS. No Fedora, você pode iniciar o serviço `nfs-server`; no Red Hat Enterprise Linux, você pode iniciar o serviço `nfs`.
3. **Escolher o que compartilhar a partir do servidor.** Decida quais sistemas de arquivos em seu servidor NFS Linux você quer disponibilizar para outros computadores. Você pode escolher qualquer ponto do sistema de arquivos e tornar todos os arquivos e diretórios abaixo desse ponto acessíveis para outros computadores.
4. **Configurar a segurança no servidor.** Você pode usar vários recursos de segurança diferentes para aplicar o nível de segurança com o qual você se sente confortável. A

segurança no nível de montagem permite que você restrinja os computadores que podem montar um recurso e, para aqueles autorizados a montá-lo, permite que você especifique se ele pode ser montado como leitura/gravação ou somente leitura. No NFS, a segurança no nível de usuário é implementada mapeando usuários de sistemas cliente para usuários no servidor NFS (com base no UID e não no nome de usuário), de modo que eles possam confiar nas permissões de leitura/gravação/execução padrão do Linux, posse de arquivos e permissões de grupo para acessar e proteger arquivos.

5. **Montar o sistema no cliente.** Cada computador cliente que tem permissão para acessar o servidor NFS do sistema de arquivos compartilhado pode montá-lo em qualquer lugar que o cliente escolher. Por exemplo, você pode montar um sistema de arquivos de um computador chamado maple no diretório `/mnt/maple` em seu sistema de arquivos local. Depois que ele está montado, você pode visualizar o conteúdo desse diretório digitando `ls /mnt/maple`. Então, você pode usar o comando `cd`, abaixo do ponto de montagem `/mnt/maple`, para ver os arquivos e diretórios que ele contém.

A Figura 20.1 ilustra um servidor de arquivos Linux usando o NFS para compartilhar (export) um sistema de arquivos e um computador cliente montando o sistema de arquivos para disponibilizá-lo para seus usuários locais.

S pode disponibilizar sistemas de arquivos selecionados para outros computadores.

Nesse exemplo, um computador chamado oak disponibiliza seu diretório `/apps/bin` para os clientes da rede (pine, maple e spruce), adicionando uma entrada ao arquivo `/etc/exports`. O computador cliente (pine) vê que o recurso está disponível e monta-o em seu sistema de arquivos local no ponto de montagem `/oak/apps`, depois que todos os arquivos, diretórios ou subdiretórios de `/apps/bin` em oak estão disponíveis para os usuários em pine (dadas as permissões adequadas).

Embora seja muitas vezes usado como um servidor de arquivos (ou outro tipo de servidor), o Linux é um sistema operacional de propósito geral, de modo que qualquer sistema Linux pode compartilhar sistemas de arquivos (export) como um servidor ou usar sistemas de arquivos de outro computador (mount) como um cliente.

ta

Um sistema de arquivos é geralmente uma estrutura de arquivos e diretórios que há em um dispositivo individual (como uma partição do disco rígido ou CD-ROM). O termo *sistema de arquivos Linux* refere-se à estrutura de arquivos e diretórios inteira (o que pode incluir sistemas de arquivos de vários discos, NFS ou uma variedade de recursos de armazenamento), começando na raiz (`/`) de um computador individual. Um diretório compartilhado em NFS pode representar o sistema de arquivos inteiro ou parte do sistema de arquivos de um computador, o qual pode ser anexado (a partir do diretório compartilhado para baixo na árvore de diretórios) ao sistema de arquivos de outro computador.

Instalando um servidor NFS

Para executar um servidor NFS, você precisa de um conjunto de módulos do kernel (que são entregues com o próprio kernel) mais algumas ferramentas de nível de usuário para configurar o serviço, executar processos `daemon` e consultar o serviço de várias maneiras. Todos os componentes necessários que ainda não estão no kernel podem ser adicionados instalando o pacote `nfs-utils`:

```
# yum install nfs-utils
```

Além de alguns documentos no diretório `/usr/share/doc/nfs-utils*`, a maior parte da documentação no pacote `nfs-utils` inclui páginas `man` para seus vários componentes. Para ver uma listagem da documentação, digite o seguinte:

```
# rpm -qd nfs-utils | less
```

Há ferramentas e páginas `man` tanto para o lado servidor NFS (para compartilhar um diretório com os outros) como para o lado cliente (para montar um diretório NFS remoto localmente). Para configurar um

servidor, você pode consultar a página man (a fim de configurar o arquivo `/etc/exports` para compartilhar seus diretórios). A página man do comando `exportfs` descreve como compartilhar e ver uma listagem de diretórios que você compartilha a partir do arquivo `/etc/exports`. A página man de `nfsd` descreve as opções que você pode passar para o daemon de servidor `rpc.nfsd`, que permite que você faça coisas como executar o servidor em modo de depuração.

Páginas man no lado do cliente incluem a página man `mount.nfs` (para ver quais opções de montagem você pode usar ao montar diretórios NFS remotos em seu sistema local). Também há uma página man `nfsmount.conf` que descreve como usar o arquivo `/etc/nfsmount.conf` para configurar como o sistema se comporta quando você monta recursos remotos localmente. A página man de `showmount` descreve como usar o comando `showmount` para ver quais diretórios compartilhados estão disponíveis a partir de servidores NFS.

Para saber mais sobre o pacote `nfs-utils`, você pode executar os comandos a seguir a fim de ver as informações sobre o pacote, os arquivos de configuração e os comandos, respectivamente:

```
# rpm -qi nfs-utils
# rpm -qc nfs-utils
# rpm -ql nfs-utils | grep bin
```

Iniciando o serviço NFS

Iniciar o servidor NFS envolve carregar diversos daemons de serviço. O serviço é iniciado de forma diferente para diferentes distribuições Linux. O serviço NFS básico no Fedora chama-se `nfs-server`. Para iniciar esse serviço, habilite-o (de modo que ele inicie sempre que seu sistema inicializar) e verifique o status executando os três seguintes comandos:

```
# systemctl start nfs-server.service
# systemctl enable nfs-server.service
# systemctl status nfs-server.service
systemctl status nfs-server.service
nfs-server.service - NFS Server
   Loaded: loaded (/lib/systemd/system/nfs-server.service; enabled)
   Active: active (running) since Sat, 2 Jun 2012 08:40:25 -0400;
           1h 28min ago
   Main PID: 7767 (rpc.mountd)
   CGroup: name=systemd:/system/nfs-server.service
           └─7767 /usr/sbin/rpc.mountd
```

Você pode ver a partir do status que o serviço `nfs-server` está habilitado e ativo. O serviço NFS também exige que o serviço RPC esteja em execução (`rpcbind`). O serviço `nfs-server` iniciará automaticamente o serviço `rpcbind`, se ele não estiver em execução.

No Red Hat Enterprise Linux 6, você precisa dos comandos `service` e `chkconfig` para verificar, iniciar e ativar o serviço NFS (`nfs`). Os comandos a seguir mostram que o serviço `nfs` não está sendo executado e está desativado atualmente:

```
# service nfs status
rpc.svcgssd is stopped
rpc.mountd is stopped
nfsd is stopped
```

```
# chkconfig --list nfs
nfs 0:off 1:off 2:off 3:off 4:off 5:off 6:off
```

Como mencionado anteriormente, o serviço `rpcbind` deve estar em execução para o NFS funcionar. Assim, você poderá usar os seguintes comandos para iniciar e permanentemente habilitar os serviços `rpcbind` e `nfs`.

```
# service rpcbind start
Starting rpcbind: [ OK ]
# service nfs start
Starting NFS services: [ OK ]
Starting NFS quotas: [ OK ]
Starting NFS daemon: [ OK ]
Starting NFS mountd: [ OK ]
# chkconfig nfs on
```

Uma vez que o serviço está sendo executado, os comandos (`mount`, `exportfs` etc.) e arquivos (`/etc/exports` `/etc/fstab`, etc.) para realmente configurar o NFS são basicamente os mesmos em todos os sistemas Linux. Então, depois que você tem o NFS instalado e funcionando, basta seguir as instruções neste capítulo para começar a utilizá-lo.

Compartilhando sistemas de arquivos NFS

Para compartilhar um sistema de arquivos NFS a partir de seu sistema Linux, você precisa exportá-lo a partir do sistema do servidor. A exportação é realizada no Linux adicionando entradas ao arquivo `/etc/exports`. Cada entrada identifica um diretório em seu sistema de arquivos local que você quer compartilhar com outros computadores. A entrada também identifica os outros computadores que podem compartilhar o recurso (ou abri-lo para todos os computadores) e inclui outras opções que refletem permissões associadas com o diretório.

Lembre-se de que quando você compartilha um diretório, também está compartilhando todos arquivos e subdiretórios abaixo desse diretório (por padrão). Então, você precisa certificar-se de que quer compartilhar tudo nessa estrutura de diretórios. Há ainda maneiras de restringir o acesso dentro dessa estrutura de diretórios; isso é discutido mais adiante, neste capítulo.

ta

edora, há uma janela NFS Server Configuration que você pode instalar digitando `yum install system-config-nfs` (o comando para abrir a janela é o mesmo nome do pacote). Essa janela pode ajudar a configurar es no arquivo `/etc/exports`, e também pode ajudá-lo a fazer coisas mais complicadas, como bloquear um ço em portas específicas e configurar o acesso dos usuários.

ora eu descreva como configurar o NFS editando diretamente o arquivo de configuração, é bastante razoável usar ferramenta gráfica. A Figura 20.2 mostra um exemplo da janela NFS Server Configuration.

ela NFS Server Configuration (`system-config-nfs`) fornece uma maneira gráfica de configurar os NFS.

Configurando o arquivo `etc/exports`

Para disponibilizar um diretório de seu sistema Linux para outros sistemas, você precisa exportar o diretório. A exportação é realizada em caráter permanente, adicionando informações sobre um diretório exportado ao arquivo `/etc/exports`.

O formato do arquivo `/etc/exports` é

```
Diretório Host(Opções...) Host(Opções...) # Comentários
```

onde *diretório* é o nome do diretório que você quer compartilhar e *Host* indica o computador do cliente para o qual o compartilhamento desse diretório é restrito. *Opções* podem incluir uma variedade de opções para definir as medidas de segurança associadas ao diretório compartilhado do host. (Você pode repetir pares *Host/Option*.) *Comentários* são quaisquer comentários opcionais que você queira adicionar (após o sinal #).

A página `man exports` (`man exports`) contém detalhes sobre a sintaxe do arquivo `/etc/exports`. Em particular, você pode ver as opções que pode usar para limitar o acesso e proteger cada diretório compartilhado.

Como usuário `root`, você pode usar qualquer editor de texto para configurar o arquivo `/etc/exports` a fim de modificar as entradas de diretório compartilhadas ou adicionar novas. Eis um exemplo de um arquivo `/etc/exports`:

```
/cal    *.linuxtoys.net(rw)                # Company events
/pub    *(ro,insecure,all_squash)         # Public dir
/home   maple(rw,root_squash) spruce(rw,root_squash)
```

A entrada `/cal` representa um diretório que contém informações sobre os eventos relacionados com a empresa. Qualquer computador no domínio da empresa (`*.linuxtoys.net`) pode montar esse compartilhamento NFS. Os usuários podem gravar arquivos no diretório, bem como lê-los (indicado pela opção `rw`). O comentário (`# Company events`) simplesmente serve para lembrá-lo do que o diretório contém.

A entrada `/pub` representa um diretório público. Ela permite que qualquer computador e usuário leia arquivos do diretório (indicado pela opção `ro`), mas não gravar arquivos). A opção `insecure` permite que qualquer computador, mesmo que não use uma porta NFS segura, acesse o diretório. A opção

`all_squash` faz com que todos os usuários (UIDs) e grupos (GIDs) sejam mapeados para o usuário ID 65534 (que é o usuário `nfsnobody` no Fedora e o usuário `nobody` no Ubuntu), dando-lhes permissão mínima sobre arquivos e diretórios.

A entrada `/home` permite que um conjunto de usuários tenha o mesmo diretório `/home` em computadores diferentes. Digamos, por exemplo, que você está compartilhando `/home` a partir de um computador chamado `oak`. Os computadores chamados `maple` e `spruce` poderiam montar esse diretório em seus próprios diretórios `/home`. Se der a todos os usuários o mesmo nome de usuário/UID em todas as máquinas, você pode ter o mesmo diretório `/home/user` disponível para cada usuário, independentemente do computador a que ele está conectado. `root_squash` é usado para impedir que o usuário `root` de outro computador tenha privilégios de `root` sobre o diretório compartilhado.

Esses são apenas exemplos, você pode compartilhar todos os diretórios que escolher, incluindo o sistema de arquivos inteiro (`/`). Obviamente, há implicações de segurança em compartilhar o sistema de arquivos inteiro ou partes sensíveis dele (como `/etc`). As opções de segurança que você pode adicionar ao seu arquivo `/etc/exports` são descritas nas seções que seguem.

Hostnames em `/etc/exports`

Você pode indicar no arquivo `/etc/exports` os computadores `host` que podem ter acesso ao seu diretório compartilhado. Se você quiser associar vários nomes de `host` ou endereços IP com um determinado diretório compartilhado, certifique-se de ter um espaço antes de cada `hostname`. Mas não inclua espaços entre um `hostname` e suas opções. Por exemplo:

```
/usr/local maple(rw) spruce(ro,root_squash)
```

Note que há um espaço depois de `(rw)`, mas nenhum depois de `maple`. Você pode identificar `hosts` de várias maneiras:

- **Host individual** — Digite um ou mais `hostnames` TCP/IP ou endereços IP. Se o `host` estiver em seu domínio local, você pode simplesmente indicar o nome do `host`. Caso contrário, use o formato completo `host.domain`. Essas são maneiras válidas para indicar os computadores `host` individuais:

```
maple
maple.handsonhistory.com
10.0.0.11
```

- **Rede IP** — Permite o acesso a todos os `hosts` de um endereço de rede específico indicando um número de rede e sua máscara, separados por uma barra (`/`). Aqui estão formas válidas de designar números de rede:

```
10.0.0.0/255.0.0.0 172.16.0.0/255.255.0.0
192.168.18.0/255.255.255.0
192.168.18.0/24
```

- **TCP/IP de domínio** — Usando curingas, você pode incluir todos ou alguns computadores `host` de um nível a partir de domínio específico. Eis alguns usos válidos dos curingas asterisco e ponto de

interrogação:

```
*.handsonhistory.com  
*craft.handsonhistory.com  
???.handsonhistory.com
```

O primeiro exemplo identifica todos os hosts no domínio `handsonhistory.com`. O segundo exemplo identifica `woodcraft`, `basketcraft` ou quaisquer outros hostnames terminando em `craft` no domínio `handsonhistory.com`. O último exemplo identifica qualquer hostname de três letras no domínio.

- **Grupos NIS** — Você pode permitir acesso a hosts contidos em um grupo NIS. Para indicar um grupo NIS, preceda o nome do grupo com um sinal de arroba (`@`) (por exemplo, `@group`).

Opções de acesso em `/etc/exports`

Você não tem que apenas dar seus arquivos e diretórios quando exporta um diretório com o NFS. Na parte opções de cada entrada no arquivo `/etc/exports`, você pode adicionar opções que permitem ou limitam o acesso por meio da criação de leitura/gravação. Essas opções, que são passadas para o NFS, são como segue:

- `ro` — O cliente pode montar esse sistema de arquivos somente leitura exportado. O padrão é montar o sistema de leitura/gravação.
- `rw` — Solicita explicitamente que um diretório compartilhado seja compartilhado com permissões de leitura/gravação. (Se o cliente optar, ele pode ainda montar o diretório como somente leitura.)

Opções de mapeamento de usuário em `/etc/exports`

Além de opções que definem como as permissões são tratadas de maneira geral, você pode usar as opções para configurar as permissões que usuários específicos têm sobre sistemas de arquivos NFS compartilhados.

Um método que simplifica esse processo é fazer com que cada usuário com várias contas de usuário tenham o mesmo nome de usuário e UID em cada máquina.

Isso torna mais fácil mapear os usuários para que eles tenham as mesmas permissões sobre um sistema de arquivos montado que eles têm sobre arquivos armazenados em seus discos rígidos locais. Se esse método não for conveniente, IDs de usuário podem ser mapeados de muitas outras maneiras. Eis alguns métodos de definição de permissões de usuário e a opção `/etc/exports` que você usa para cada método:

- **usuário root** — O usuário `root` do cliente é mapeado por padrão para o nome de usuário `nfsnobody` (UID 65534). Isso impede que o usuário `root` de um computador cliente seja capaz de alterar todos os arquivos e diretórios no sistema de arquivos compartilhados. Se você quiser que o usuário `root` do cliente tenha permissão de `root` no servidor, use a opção `no_root_squash`.

em mente que mesmo que root seja restringido, o usuário root do cliente ainda pode se transformar em qualquer outra conta de usuário e acessar arquivos dessas contas no servidor. Portanto, fique ciente de que você está dando a root todos os seus dados de usuário antes de compartilhá-lo com permissões de leitura/gravação com o cliente.

- **nfsnobody ou usuário/grupo nobody** — Ao usar o ID de usuário e o ID grupo 65534, essencialmente você cria uma permissão de usuário/grupo com permissões que não concedem o acesso a arquivos que pertencem a todos os usuários reais no servidor, a menos que esses usuários abram permissões para todos. Mas os arquivos criados pelo usuário ou grupo 65534 estão disponíveis para qualquer pessoa que tenha sido atribuída ao usuário ou grupo 65534. Para configurar todos os usuários remotos para o usuário/grupo 65534, use a opção `all_squash`.

Os UIDs e GIDs 65534 são usados para impedir que o ID execute para um usuário ou grupo válido ID. Usando as opções `anonuid` ou `anongid`, você pode alterar o usuário ou grupo 65534, respectivamente. Por exemplo, `anonuid=175` configura todos os usuários `anonymous` com o UID 175 e `anongid=300` configura o GID como 300. (Apenas o número é exibido quando você lista a permissão de arquivo, a menos que você adicione entradas com nomes a `/etc/passwd` e `/etc/group` para os novos UIDs e GIDs.)

- **Mapeamento de usuário** — Se um usuário tiver contas de login para um conjunto de computadores (e tem o mesmo ID), o NFS, por padrão, mapeia esse ID. Isso significa que se o usuário chamado `mike` (UID 110) no computador `maple` tiver uma conta no computador `pine` (`mike`, UID 110), ele pode usar seus próprios arquivos remotamente montados em qualquer computador a partir de qualquer computador.

Se um usuário cliente que não está configurado no servidor criar um arquivo no diretório NFS montado, o arquivo receberá o UID e o GID do cliente remoto. (O comando `ls -l` no servidor mostra o UID do proprietário.) Use a opção `map_static` para identificar um arquivo que contém mapeamentos de usuário.

Exportando os sistemas de arquivos compartilhados

Depois de ter adicionado as entradas a seu arquivo `/etc/exports`, execute o comando `exportfs` para ter os diretórios exportados (disponível para outros computadores na rede). Reinicie o computador ou reinicie o serviço NFS e o comando `exportfs` é executado automaticamente para exportar seus diretórios. Se você quiser exportá-los imediatamente, execute `exportfs` partir da linha de comando (como root).

Executar o comando `exportfs` depois de alterar o arquivo `exports` é uma boa ideia. Se houver algum erro no

Eis um exemplo do comando `exportfs`:

```
# /usr/sbin/exportfs -a -r -v
exporting maple:/pub
exporting spruce:/pub
exporting maple:/home
exporting spruce:/home
exporting */:/mnt/win
```

A opção `-a` indica que todos os diretórios listados em `/etc/exports` devem ser exportados. `-r` resincroniza todos os exports com o arquivo `/etc/exports` atual (desativando as exportações não listadas no arquivo). A opção `-v` imprime uma saída detalhada. Nesse exemplo, os diretórios `/pub` e `/home` do servidor local são imediatamente disponibilizados para montagem pelos computadores clientes que são nomeados (`maple` e `spruce`). O diretório `/mnt/win` está disponível para todos os computadores clientes.

Protegendo seu servidor NFS

A instalação NFS foi criada em uma época em que criptografia e outras medidas de segurança não eram rotineiramente integradas em serviços de rede (como login remoto, compartilhamento de arquivos e execução remota). Portanto, o NFS (mesmo na versão 3) sofre com alguns problemas de segurança gritantes.

Questões de segurança tornaram o NFS um recurso inadequado para usar em redes públicas e até dificultaram sua utilização segura dentro de uma organização. Algumas dessas questões foram:

- **Usuários root remotos** — Mesmo com o padrão `root_squash` (que impede que os usuários `root` tenham acesso de `root` a partes remotas), o usuário `root` em qualquer máquina com que você compartilha diretórios NFS pode ter acesso a qualquer outra conta de usuário. Portanto, se você estiver fazendo algo como compartilhar diretórios com permissão de leitura/gravação, o usuário `root` em qualquer diretório que você está compartilhando terá acesso completo ao conteúdo desses diretórios.
- **Comunicações não criptografadas** — Como o tráfego NFS não é criptografado, alguém espionando sua rede será capaz de ver os dados que estão sendo transferidos.
- **Mapeamento de usuário** — Permissões padrão para o compartilhamento NFS são mapeadas por ID de usuário. Assim, por exemplo, um usuário com UID 500 em um cliente NFS terá acesso a arquivos de propriedade UID 500 no servidor NFS. Isso é independente dos nomes de usuário utilizados.
- **Estrutura do sistema de arquivos exposta** — Até o NFSv3, se você compartilhasse um diretório via NFS, você expunha a localização desse diretório no sistema de arquivos do servidor. (Em

outras palavras, se você compartilhasse o diretório `/var/stuff`, os clientes saberiam que `/var/stuff` era sua localização exata em seu servidor).

Essa é a má notícia. A boa notícia é que a maioria dessas questões é abordada no NFSv4 mas requer alguma configuração extra. Ao integrar suporte Kerberos, o NFSv4 permite configurar o acesso do usuário com base em cada usuário que recebe uma permissão Kerberos. Para você, o trabalho extra é configurar um servidor Kerberos. Quanto à exposição dos locais NFS compartilhados, com o NFSv4 você pode ligar diretórios compartilhados a um diretório `/exports`, de tal modo que quando eles são compartilhados, a localização exata desses diretórios não é exposta.

Visite <https://help.ubuntu.com/community/NFSv4Howto> para obter detalhes sobre recursos NFSv4 no Ubuntu.

Recursos padrão de segurança do Linux associados com o NFS, firewalls `iptables`, TCP wrappers e SELinux podem desempenhar um papel na obtenção e fornecimento de acesso ao seu servidor NFS a partir de clientes remotos. Em particular, fazer os recursos de firewall `iptables` funcionarem com NFS pode ser particularmente difícil. Esses recursos de segurança são descritos nas seções que se seguem.

Abrindo seu firewall para NFS

O serviço NFS depende de vários daemons de serviços diferentes para operação normal, com a maioria desses daemons ouvindo em diferentes portas de acesso. Para o NFSv4 padrão usado no Fedora, as portas TCP e UDP 2049 (`nfs`) e 111 (`rpcbind`) devem ser abertas para um servidor NFS executar corretamente. O servidor também deve abrir portas TCP e UDP 20048 para o comando `showmount` ser capaz de consultar diretórios NFS compartilhados disponíveis a partir do servidor.

Para abrir as portas no firewall do servidor NFS, certifique-se de que `iptables` está ativado e inicializado com regras de firewall semelhantes às seguintes adicionadas ao arquivo `/etc/sysconfig/iptables`:

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 111 -j ACCEPT
-A INPUT -m state --state NEW -m udp -p udp --dport 111 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 2049 -j ACCEPT
-A INPUT -m state --state NEW -m udp -p udp --dport 2049 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 20048 -j ACCEPT
-A INPUT -m state --state NEW -m udp -p udp --dport 20048 -j ACCEPT
```

No Red Hat Enterprise Linux 6.x e versões anteriores, a questão do firewall é um pouco mais complexa. O problema, no que se relaciona com firewalls, é que há vários serviços diferentes associados com o NFS que ouvem em portas diferentes e essas portas são atribuídas aleatoriamente. Para contornar esse problema, você precisa bloquear os números de porta que esses serviços usam e abrir o firewall para essas portas serem acessíveis.

Para tornar o processo de bloquear as portas do servidor NFS mais fácil, entradas no arquivo `/etc/sysconfig/nfs` podem ser adicionadas para atribuir números de porta específicos aos

serviços. Eis alguns exemplos de opções no arquivo `/etc/sysconfig/nfs` com números de porta estáticos configurados:

```
RQUOTAD_PORT=49001
LOCKD_TCPPORT=49002
LOCKD_UDPPORT=49003
MOUNTD_PORT=49004
STATD_PORT=49005
STATD_OUTGOING_PORT=49006
```

Com essas portas configuradas, reiniciei o serviço `nfs` (`service nfs restart`). Usando o comando `netstat`, você pode ver os processos resultantes que estão ouvindo nessas portas atribuídas:

```
tcp  0  0  0.0.0.0:49001  0.0.0.0:*  LISTEN  4682/rpc.rquotad
tcp  0  0  0.0.0.0:49002  0.0.0.0:*  LISTEN  -
tcp  0  0  0.0.0.0:49004  0.0.0.0:*  LISTEN  4698/rpc.mountd
tcp  0  0  :::49002      :::*      LISTEN  -
tcp  0  0  :::49004      :::*      LISTEN  4698/rpc.mountd
udp  0  0  0.0.0.0:49001  0.0.0.0:*  -       4682/rpc.rquotad
udp  0  0  0.0.0.0:49003  0.0.0.0:*  -       -
udp  0  0  0.0.0.0:49004  0.0.0.0:*  -       4698/rpc.mountd
udp  0  0  :::49003      :::*      -       -
udp  0  0  :::49004      :::*      -       4698/rpc.mountd
```

Com esses números de porta configurados e sendo utilizados por diversos serviços, agora você pode adicionar regras de `iptables`, como você fez com as portas 2049 e 111 para o serviço NFS básico.

Permitindo o acesso NFS em TCP wrappers

Para serviços como `vsftpd` e `sshd`, os TCP wrappers no Linux permitem adicionar informações aos arquivos `/etc/hosts.allow` e `/etc/hosts.deny` para indicar quais hosts podem ou não podem acessar o serviço. Embora o daemon servidor `nfsd` em si não esteja habilitado para TCP wrappers, o serviço `rpcbind` está.

Para versões NFSv3 e anteriores, simplesmente adicionar uma linha como a seguinte no arquivo `/etc/hosts.deny` seria negar acesso ao serviço `rpcbind`, mas também negar acesso ao seu serviço NFS:

```
rpcbind: ALL
```

Para os servidores que executam NFSv4 por padrão, porém, a linha `rpcbind: ALL` recém-mostrada impede que hosts externos obtenham informações sobre serviços RPC (como o NFS), usando comandos como `showmount`. Mas isso não vai impedir que você monte um diretório NFS compartilhado.

Configurando o SELinux para seu servidor NFS

Com o SELinux configurado como `Permissive` ou `Disabled`, ele não bloqueará o acesso ao serviço NFS. No modo `Enforcing`, porém, há poucos booleanos do SELinux que você precisa conhecer. Para verificar o estado do SELinux em seu sistema, digite o seguinte:

```
# getenforce  
Enforcing  
# grep ^SELINUX= /etc/sysconfig/selinux  
SELINUX=enforcing
```

Se seu sistema estiver no modo Enforcing, como está aqui, verifique a página man do `nfs_selinux` para obter informações sobre as configurações do SELinux que podem afetar o funcionamento de seu serviço `vsftpd`. A seguir, estão alguns contextos do SELinux associados com o NFS que você pode precisar conhecer:

- `nfs_export_all_ro` — Com esse conjunto booleano configurado, o SELinux permitirá que você compartilhe arquivos com permissão de somente leitura usando NFS. O compartilhamento de arquivos NFS somente leitura é habilitado com essa opção ativada, independentemente do contexto do arquivo SELinux configurado nos arquivos compartilhados e diretórios.
- `nfs_export_all_rw` — Com esse booleano configurado, o SELinux permitirá que você compartilhe arquivos com permissão de leitura/gravação usando o NFS. Como com o booleano anterior, isso vai funcionar independentemente do contexto de arquivo configurado sobre os arquivos e diretórios compartilhados.
- `use_nfs_home_dirs` — Para permitir que o servidor NFS compartilhe seus diretórios iniciais através do NFS, ative esse valor booleano.

Dos booleanos descritos, os dois primeiros estão ativados por padrão. O booleano `use_nfs_home_dirs` está desativado. Para ativar o diretório `use_nfs_home_dirs`, você pode digitar o seguinte:

```
# setsebool -P use_nfs_home_dirs on
```

Você pode ignorar todos os booleanos relacionados com compartilhamento de arquivos NFS; mas, mudando os contextos de arquivos sobre os arquivos e diretórios, você vai querer compartilhar via NFS. Os contextos de arquivos `public_content_t` e `public_content_rw_t` podem ser configurados em qualquer diretório que você quer compartilhar via NFS (ou protocolos de compartilhamento de arquivo, como HTTP, FTP e outros, nesse aspecto). Por exemplo, para configurar uma regra que permite que o diretório `/whatever` e seus subdiretórios sejam compartilhados como leitura/gravação via NFS e depois aplicar essa regra, digite o seguinte:

```
# semanage fcontext -a -t public_content_rw_t "/whatever(/.*)?"  
# restorecon -F -R -v /whatever
```

Se quiser permitir que os usuários sejam capazes apenas de ler arquivos de um diretório, mas não gravar nele, você pode atribuir o contexto de arquivo `public_content_t` ao diretório.

Usando sistemas de arquivos NFS

Depois que um servidor exporta um diretório através da rede utilizando NFS, um computador cliente

conecta esse diretório ao seu próprio sistema de arquivos usando o comando `mount`. Esse é o mesmo comando usado para montar sistemas de arquivos de discos rígidos locais, CDs e pen drives, mas com opções ligeiramente diferentes.

O comando `mount` permite que um cliente automaticamente monte diretórios NFS adicionados ao arquivo `/etc/fstab`, exatamente como ele faz com os discos locais. Diretórios NFS também podem ser adicionados ao arquivo `/etc/fstab` de tal maneira que eles não são montados automaticamente (assim, você pode montá-los manualmente quando escolher). Com uma opção `noauto`, um diretório NFS listado em `/etc/fstab` é inativo até que o comando `mount` é usado, depois que o sistema está instalado e funcionando, para montar o sistema de arquivos.

Além do arquivo `/etc/fstab`, você também pode configurar opções de montagem usando o arquivo `/etc/nfsmount.conf`. Dentro desse arquivo, você pode configurar as opções de montagem que se aplicam a qualquer diretório NFS que você montar ou apenas àqueles associados com específicos pontos de montagem ou servidores NFS.

Antes de começar a montar diretórios NFS compartilhados, porém, você provavelmente vai querer verificar quais diretórios compartilhados estão disponíveis via NFS usando o comando `showmount`.

Visualizando compartilhamentos NFS

A partir de um sistema cliente Linux, você pode usar o comando `showmount` para ver quais diretórios compartilhados estão disponíveis a partir de um computador selecionado. Por exemplo:

```
$ /usr/sbin/showmount -e server.example.com
/export/myshare client.example.com
/mnt/public *
```

A saída `showmount` mostra que o diretório compartilhado chamado `/export/myshare` está disponível apenas para o host `client.example.com`. O diretório `/mnt/public` compartilhado, porém, está disponível para qualquer um.

Montando manualmente um sistema de arquivos NFS

Uma vez que você sabe que o diretório a partir de um computador em sua rede foi exportado (isto é, foi disponibilizado para montagem), você pode montar o diretório manualmente usando o comando `mount`. Essa é uma boa maneira de certificar-se de que está disponível e funcionando antes de você configurá-lo para montar permanentemente. O seguinte é um exemplo de como montar o diretório `/stuff` de um computador chamado `maple` em seu computador local:

```
# mkdir /mnt/maple
# mount maple:/stuff /mnt/maple
```

O primeiro comando (`mkdir`) cria o diretório do ponto de montagem. (`/mnt` é um lugar comum para colocar discos e sistemas de arquivos NFS montados temporariamente.) O comando `mount` identifica o

computador remoto e o sistema de arquivos compartilhado, separados por dois-pontos (`maple:/stuff`), e o diretório do ponto de montagem local (`/mnt/maple`) vem em seguida.

ta

montagem falhar, certifique-se de que o serviço NFS está em execução no servidor e que as regras de firewall do servidor não negam acesso ao serviço. A partir do servidor, digite `ps ax | grep nfsd` para ver uma lista de processos do servidor `nfsd`. Se você não vir a lista, tente iniciar seu servidor NFS como descrito anteriormente, neste título. Para ver as regras do firewall, digite `iptables -vnl`. Por padrão, o daemon `nfsd` ouve solicitações na porta de número 2049. Seu firewall deve aceitar solicitações `udp` nas portas 2049 (`nfs`) e 111 (`rpc`). No Hat Enterprise Linux 6 e versões anteriores do Fedora, você pode precisar configurar portas estáticas para serviços relacionados e depois abrir as portas para esses serviços no firewall. Consulte a seção “Protegendo seu servidor NFS” anteriormente, neste capítulo, para avaliar como superar esses problemas de segurança.

Para garantir que a montagem do NFS ocorreu, digite `mount -t nfs`. Esse comando lista todos os sistemas de arquivos NFS montados. Eis um exemplo do comando `mount` e sua saída (com sistemas de arquivos não pertinentes a essa discussão removidos):

```
# mount -t nfs
maple:/stuff on /mnt/maple type nfs (rw,relatime,vers=3,rsize=65536,
 wsize=65536,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,
 mountaddr=192.168.0.122,mountvers=3,mountport=892,mountproto=udp,
 local_lock=none,addr=192.168.0.122)
```

A saída do comando `mount -t nfs` mostra apenas os sistemas de arquivos montados a partir de servidores de arquivos NFS. O sistema de arquivos NFS recém-montado é o diretório `/stuff` no computador `maple` (`maple:/stuff`). Ele é montado em `/mnt/maple` e seu tipo de montagem é `nfs`. O sistema de arquivos foi montado para leitura/gravação (`rw`) e o endereço IP de `maple` é `192.168.0.122` (`addr=192.168.0.122`). Muitas outras configurações relacionadas com montagem também são mostradas, como tamanhos de pacote de leitura e gravação, e número de versão do NFS.

A operação de montagem recém-mostrada monta temporariamente um sistema de arquivos NFS no sistema local. A próxima seção descreve como tornar a montagem mais permanente (usando o arquivo `/etc/fstab`) e como selecionar várias opções para montagens NFS.

Montagem de um sistema de arquivos NFS no momento da inicialização

Para configurar um sistema de arquivos NFS para montar automaticamente em um ponto de montagem especificado cada vez que você iniciar seu sistema Linux, você precisa adicionar uma entrada a esse sistema de arquivos NFS no arquivo `/etc/fstab`. Esse arquivo contém informações sobre todos os tipos de sistemas de arquivos montados (e disponíveis para serem montados) em seu sistema.

Eis o formato para adicionar um sistema de arquivos NFS ao sistema local:

O primeiro item (*host: diretório*) identifica o computador do servidor NFS e o diretório compartilhado. *ponto de montagem* é o ponto de montagem local em que o diretório NFS está montado. Ele é seguido pelo tipo de sistema de arquivos (*nfs*). Quaisquer opções relacionadas à montagem aparecem próximas em uma lista separada por vírgulas. (Os dois últimos zeros configuram o sistema para não copiar o conteúdo do sistema de arquivos e para não executar *fsck* no sistema de arquivos.)

Eis alguns exemplos de entradas NFS em `/etc/fstab`:

```
maple:/stuff      /mnt/maple      nfs bg, rsize=8192, wsize=8192 0 0
oak:/apps         /oak/apps       nfs noauto, ro 0 0
```

No primeiro exemplo, o diretório remoto `/stuff` do computador chamado `maple` (`maple:/stuff`) é montado no diretório local `/mnt/maple` (o diretório local deve existir). Se a montagem falhar, porque a ação está indisponível, `bg` faz com que a tentativa de montagem entre em segundo plano e tente novamente mais tarde.

O tipo de sistema de arquivos é `nfs` e os tamanhos de buffer de leitura (`rsize`) e gravação (`wsize`) (discutidos na seção “Usando opções de montagem”, mais adiante, neste capítulo) estão fixados em 8192 para acelerar a transferência de dados associada com essa conexão. No segundo exemplo, o diretório remoto é `/apps` no computador chamado `oak`. Ele é configurado como um sistema de arquivos NFS (`nfs`), que pode ser montado no diretório `/oak/apps` localmente. Esse sistema de arquivos não é montado automaticamente (`noauto`), mas pode ser montado como somente leitura (`ro`) usando o comando `mount` depois que o sistema já está em execução.

ca

padrão é montar um sistema de arquivos NFS como leitura/gravação. Mas o padrão para a exportação de um sistema de arquivos é somente leitura. Se você não conseguir gravar em um sistema de arquivos NFS, verifique se ele é exportado como leitura/gravação a partir do servidor.

Montando sistemas de arquivos noauto

Seu arquivo `/etc/fstab` também pode conter dispositivos para outros sistemas de arquivos que não são montados automaticamente. Por exemplo, você pode ter múltiplas partições de disco em seu disco rígido ou um sistema de arquivos NFS compartilhado que queira montar apenas ocasionalmente. Um sistema de arquivos `noauto` pode ser montado manualmente. A vantagem é que quando você digita o comando `mount`, pode gravar menos informações e ter o resto preenchido com o conteúdo do arquivo `/etc/fstab`. Então, por exemplo, você poderia digitar:

```
# mount /oak/apps
```

Com esse comando, `mount` sabe verificar o arquivo `/etc/fstab` para obter o sistema de arquivos a

montar (`oak:/apps`), o tipo de sistema de arquivos (`nfs`) e as opções para usar com a montagem (nesse caso `ro`, de read-only [somente leitura]). Em vez de digitar o ponto de montagem local (`/oak/apps`), você poderia ter digitado o nome do sistema de arquivos remoto (`oak:/apps`) e ter as outras informações preenchidas.

ca

Quando nomear pontos de montagem, a inclusão do nome do servidor remoto de NFS nesse nome pode ajudá-lo a lembrar onde os arquivos realmente estão sendo armazenados. Isso pode não ser possível se você estiver compartilhando diretórios iniciais (`/home`) ou diretórios de email (`/var/spool/mail`). Por exemplo, você pode montar um sistema de arquivos a partir de uma máquina chamada `duck` no diretório `/mnt/duck`.

Usando as opções de montagem

Você pode adicionar várias opções de `mount` ao arquivo `/etc/fstab` (ou a uma linha de comando `mount` sozinha) para influenciar a forma como o sistema de arquivos é montado. Quando você adiciona opções a `/etc/fstab`, elas devem ser separadas por vírgulas. Por exemplo, aqui, as opções `noauto`, `ro` e `hard` são usadas quando `oak:/apps` está montado:

```
oak:/apps          /oak/apps          nfs    noauto,ro,hard    0 0
```

A seguir, estão algumas opções que são valiosas para a montagem de sistemas de arquivos NFS. Você pode ler sobre essas e outras opções de montagem NFS que podem ser colocadas no arquivo `/etc/fstab` na página `man` do comando `nfs` (`man 5 nfs`):

- `hard` — Se essa opção for usada e o servidor NFS desconectar ou cair enquanto um processo está esperando para acessá-lo, o processo esperará até o servidor voltar. Isso é útil se for essencial que os dados com que você está trabalhando permaneçam em sincronia com os programas que os estão acessando. (Esse é o comportamento padrão).
- `soft` — Se o servidor NFS desconectar ou cair, um processo tentando acessar os dados do servidor expirará após um determinado período de tempo, quando essa opção estiver ativada. Um erro de entrada/saída é enviado para o processo tentar acessar o servidor NFS.
- `rsize` — O tamanho dos blocos de dados (em bytes) que o cliente NFS solicitará para ser utilizado quando ele estiver lendo dados a partir de um servidor NFS. O padrão é 1024. Usando um número maior (como 8192), você obterá melhor desempenho em uma rede que é rápida (como uma rede local) e é relativamente livre de erros (isto é, uma que não tem muito ruído ou colisões).
- `wsize` — O tamanho dos blocos de dados (em bytes) que o cliente NFS solicitará para ser utilizado quando gravar dados em um servidor NFS. O padrão é 1024. Problemas de desempenho são os mesmos que com a opção `rsize`.
- `timeo=#` — Define o tempo que uma segunda transmissão é feita depois de ocorrer um timeout de RPC, onde `#` representa um número em décimos de segundo. O valor padrão é sete décimos de segundo. Cada tempo limite sucessivo faz com que o valor de tempo limite seja dobrado (até o

máximo de 60 segundos). Aumente esse valor se você acredita que o tempo limite está ocorrendo por causa da resposta lenta do servidor ou uma rede lenta.

- `retrans=#` — Define o número de timeouts e retransmissões menores que precisam acontecer antes do timeout maior ocorrer.
- `retry=#` — Define por quantos minutos continuar a repetir as solicitações de falha de montagem, onde `#` é substituído pelo número de minutos para tentar novamente. O padrão é 10.000 minutos (que é aproximadamente uma semana).
- `bg` — Se a primeira tentativa de montagem falhar, todas as tentativas de montagem subsequentes ocorrerão em segundo plano. Essa opção é muito útil se você estiver montando um sistema de arquivos NFS lento ou esporadicamente disponível. Ao colocar solicitações de montagem no segundo plano, o sistema pode continuar a montar outros sistemas de arquivos, em vez de esperar que o atual se complete.

ta

Um ponto de montagem aninhado estiver faltando, ocorre um tempo limite para permitir que o ponto de montagem necessário seja adicionado. Por exemplo, se você montar `/usr/trip` e `/usr/trip/extra` como sistemas de arquivos NFS e `/usr/trip` ainda não estiver montado quando `/usr/trip/extra` tentar montar, `/usr/trip/extra` expirará. Se você tiver sorte, `/usr/trip` sobe e `/usr/trip/extra` é montado na primeira tentativa.

- `fg` — Se a primeira tentativa de montagem expirar, as montagens subsequentes ocorrerão em primeiro plano. Esse é o comportamento padrão. Use essa opção se for imperativo que a montagem seja bem-sucedida antes de continuar (por exemplo, se você estivesse montando `/usr`).

Nem todas as opções de montagem NFS precisam ir para o arquivo `/etc/fstab`. No lado do cliente, o arquivo `/etc/nfsmount.conf` pode ser configurado para as seções `Mount`, `Server` e `Global`. Na seção `Mount`, você pode indicar quais opções de montagem são utilizadas quando um sistema de arquivos NFS é montado em um ponto de montagem específico. A seção `Server` permite adicionar opções a qualquer sistema de arquivos NFS montado a partir de um servidor NFS específico. Opções globais se aplicam a todos os NFS montados a partir desse cliente.

A seguinte entrada no arquivo `/etc/nfsmount.conf` configura um tamanho do bloco de leitura e gravação de 32KB para todos os diretórios NFS montados a partir do sistema chamado `thunder.example.com`:

```
[ Server "thunder.example.com" ]
  rsize=32k
  wsize=32k
```

Para configurar opções padrão para todos os NFS montados em seus sistemas, você pode ativar o bloco `NFSMount_Global_Options`, removendo o caractere de comentário. Nesse bloco, você pode

configurar coisas como protocolos e versões de NFS, bem como as taxas de transmissão e as configurações de novas tentativas. Eis um exemplo de um bloco `NFSMount_Global_Options`:

```
[ NFSMount_Global_Options ]
# Isso configura a versão padrão como NFS 4
Defaultvers=4
# Configura o número de vezes que uma solicitação será tentada novamente
# gerando um timeout
Retrans=2
# Configura como 2 minutos o tempo antes de tentar novamente
# uma montagem que falhou
# Retry=2
```

No exemplo mostrado, a versão padrão do NFS é 4. Os dados são retransmitidos duas vezes (2) antes de gerar um timeout. O tempo de espera é de 2 minutos antes de tentar novamente uma transmissão que falhou. Você pode substituir qualquer um desses valores padrão adicionando opções de montagem à `/etc/fstab` ou à linha de comando de montagem quando o diretório NFS é montado.

Usando o `autofs` para montar sistemas de arquivos NFS sob demanda

As recentes melhorias para autodetecção e montagem de dispositivos tornaram possível simplesmente inserir ou conectar esses dispositivos para detectá-los, montá-los e exibi-los. Mas para tornar o processo de detecção e montagem de sistemas de arquivos remotos NFS mais automático, você ainda precisa usar uma instalação como `autofs` (abreviação de *automatically mounted filesystems*, isto é, sistemas de arquivos montados automaticamente).

O recurso `autofs` monta sistemas de arquivos de rede sob demanda quando alguém tenta usar o sistema de arquivos. Com o recurso `autofs` configurado e ativado, você pode fazer com que quaisquer diretórios NFS compartilhados disponíveis sejam montados sob demanda. Para usar o recurso de `autofs`, você precisa ter o pacote `autofs` instalado. (Para o Fedora e o RHEL, você pode digitar **`yum install autofs`** ou, para o Ubuntu ou o Debian, **`apt-get install autofs`** para instalar o pacote a partir da rede.)

Automontando o diretório `/net`

Com `autofs` habilitado, se você sabe o `hostname` e o diretório que está sendo compartilhado por outro computador `host`, basta mudar (`cd`) para o diretório de montagem `autofs` (`/net` ou `/var/autofs` por padrão). Isso faz com que o recurso compartilhado seja automaticamente montado e disponibilizado para você.

Os passos a seguir explicam como ativar o recurso `autofs` no Fedora:

1. No Fedora, como usuário `root` de uma janela do Terminal, abra o arquivo `/etc/auto.master` e procure a seguinte linha:

```
/net -hosts
```

Isso faz com que o diretório `/net` atue como o ponto de montagem para os diretórios NFS

compartilhados que você quer acessar na rede. (Se houver um caractere de comentário no início da linha, remova-o.)

2. Para iniciar o serviço `autofs` no Fedora, digite o seguinte, como usuário `root`:

```
# systemctl start autofs.service
```

3. Em um sistema Fedora, configure o serviço `autofs` para reiniciar a cada vez que você iniciar o sistema:

```
# chkconfig autofs on
```

Acredite ou não, isso é tudo que você tem de fazer. Se você tiver uma conexão de rede para os servidores NFS a partir da qual você quer compartilhar diretórios, tente acessar um diretório NFS compartilhado. Por exemplo, se você sabe que o diretório `/usr/local/share` está sendo compartilhado a partir do computador em sua rede chamada `shuttle`, você pode fazer o seguinte:

```
$ cd /net/shuttle
```

Se esse computador tiver qualquer diretório compartilhado disponível para você, você pode mudar para esse diretório.

Você também pode digitar o seguinte:

```
$ ls  
usr
```

Você deve ser capaz de ver que o diretório `usr` é parte do caminho para um diretório compartilhado. Se houvesse diretórios compartilhados a partir de outros diretórios de nível superior (tais como `/var` ou `/tmp`), você os veria também. É claro que, ver qualquer um desses diretórios depende de como a segurança está configurada no servidor.

Tente ir direto para o diretório compartilhado também. Por exemplo:

```
$ cd /net/shuttle/usr/local/share  
$ ls  
info man music television
```

Nesse ponto, o comando `ls` deve revelar o conteúdo do diretório `/usr/local/share` no computador chamado `shuttle`. O que você pode fazer com esse conteúdo depende de como ele foi configurado para compartilhamento pelo servidor.

Isso pode ser um pouco desconcertante, porque você não vai ver os arquivos ou diretórios até realmente tentar usá-los, como para mudar para um diretório montado pela rede. O comando `ls`, por exemplo, não vai mostrar nada sob um diretório montado pela rede até que o diretório seja montado, o que pode levar a uma impressão de “às vezes ele aparece, às vezes não”. Basta mudar para um diretório montado pela rede ou acessar um arquivo em tal diretório e `autofs` vai cuidar do resto.

No exemplo mostrado, o hostname `shuttle` é usado. Mas você pode usar qualquer nome ou endereço IP

que identifica a localização do computador do servidor NFS. Por exemplo, em vez de `shuttle`, você poderia ter usado `shuttle.example.com` ou um endereço IP como `192.168.0.122`.

Automontando diretórios iniciais

Em vez de simplesmente montar um sistema de arquivos NFS sob o diretório `/net`, você pode querer configurar `autofs` para montar um diretório NFS específico em um local específico. Por exemplo, você pode configurar o diretório inicial do usuário a partir de um servidor centralizado que pode ser montado automaticamente a partir de uma máquina diferente quando um usuário faz login. Da mesma maneira, você pode usar um mecanismo de autenticação central, como o LDAP (conforme descrito no Capítulo 11, “Gerenciando contas de usuário”), para oferecer contas de usuário centralizadas também.

O procedimento a seguir ilustra como configurar uma conta de usuário em um servidor NFS e compartilhar o diretório inicial de um usuário chamado `joe` desse servidor para que ele possa ser montado automaticamente quando `joe` faz login em um computador diferente. Nesse exemplo, em vez de usar um servidor central de autenticação, contas correspondentes são criadas em cada sistema.

1. No servidor NFS (`my nfs.example.com`) que fornece um diretório inicial centralizado para o usuário chamado `joe`, crie uma conta de usuário para `joe` com um diretório inicial `/home/shared/joe` como seu nome. Também localize o número de ID do usuário Joe a partir do arquivo `/etc/passwd` (terceiro campo), para poder correspondê-lo ao configurar uma conta de usuário para `joe` em outro sistema.

```
# mkdir /home/shared
# useradd -c "Joe Smith" -d /home/shared/joe joe
# grep joe /etc/passwd
joe:x:507:507:Joe Smith:/home/shared/joe:/bin/bash
```

2. No servidor NFS, exporte o diretório `/home/shared/` para qualquer sistema em sua rede local (eu uso `192.168.0.*` aqui), assim, você pode compartilhar o diretório inicial de `joe`, e quaisquer outros usuários que criar, adicionando esta linha ao arquivo `/etc/exports`:

```
# arquivo /etc/exports para compartilhar diretórios sob
/home/shared
# somente para outros sistemas na rede 192.168.0.0/24:
/home/shared 192.168.0.*(rw,insecure)
```

ta

Exemplo de arquivo `exports` acima, a opção `insecure` permite que clientes inseguros utilizem portas acima da porta 1024 para fazer solicitações de montagem. Alguns clientes NFS exigem isso, porque não têm acesso a portas reservadas para NFS.

3. No servidor NFS, reinicie o serviço `nfs-server` ou, se ele já estiver funcionando, você pode simplesmente exportar o diretório compartilhado, como segue:


```
# exportfs -a -r -v
```

4. No servidor NFS, certifique-se de que as portas apropriadas estão abertas no firewall. Consulte a seção “Protegendo seu servidor NFS” para mais detalhes.
5. No sistema NFS cliente, adicione uma entrada ao arquivo `/etc/auto.master` que identifique o ponto de montagem onde você quer que o diretório NFS remoto seja montado e um arquivo (de sua escolha) onde você identificará a localização do diretório NFS remoto. Eu adicionei essa entrada ao arquivo `auto.master`:

```
/home/remote /etc/auto.joe
```

6. No sistema NFS cliente, adicione uma entrada ao arquivo que você acabou de anotar (`/etc/auto.joe` é o que usamos) que contenha uma entrada como a seguinte:

```
joe -rw mynfs.example.com:/home/shared/joe
```

7. No sistema NFS cliente, reinicie o serviço `autofs`:

```
# systemctl restart autofs.service
```

8. No sistema NFS cliente, crie um usuário chamado `joe` usando o comando `useradd`. Para essa linha de comando, você precisa obter o UID para `joe` no servidor (507, neste exemplo), de modo que `joe` no sistema cliente seja o proprietário dos arquivos a partir do diretório inicial de Joe NFS. Quando você executa o seguinte comando, a conta de usuário `joe` é criada, mas você verá uma mensagem de erro informando que o diretório já existe (o que é correto):

```
# useradd -u 507 -c "Joe Smith" -d /home/remote/joe joe
```

```
# passwd joe
```

```
Changing password for user joe.
```

```
New password: *****
```

```
Retype new password: *****
```

9. No sistema NFS cliente, faça login como `joe`. Se tudo estiver funcionando corretamente, quando Joe fizer login e tentar acessar seu diretório inicial (`/home/remote/joe`), o diretório `/home/share/joe` deve ser montado a partir do servidor `mynfs.example.com`. O diretório NFS foi compartilhado e montado como leitura/gravação com posse configurada como UID 507 (`joe` em ambos os sistemas), de modo que o usuário `joe` no sistema local deve ser capaz de adicionar, excluir, alterar e visualizar arquivos no diretório.

Depois de Joe ter feito logoff (na verdade, quando ele para de acessar o diretório) por um período de tempo (10 minutos, por padrão), o diretório é desmontado.

Desmontando sistemas de arquivos NFS

Depois que um sistema de arquivos NFS é montado, desmontar é simples. Você pode usar o comando

umount com um ponto de montagem local ou o nome do sistema de arquivos remoto. Por exemplo, eis duas maneiras como você pode desmontar `maple:/stuff` a partir do diretório local `/mnt/maple`:

```
# umount maple:/stuff
# umount /mnt/maple
```

Ambas as formas funcionam. Se `maple:/stuff` for montado automaticamente (a partir de uma listagem no arquivo `/etc/fstab`), o diretório será remontado na próxima vez que você inicializar o Linux. Se fosse uma montagem temporária (ou listada como `noauto` em `/etc/fstab`), o diretório não seria remontado na inicialização.

o comando é `umount`, não `unmount`. Isso é fácil de errar.

Se você receber a mensagem `device is busy` ao tentar desmontar um sistema de arquivos, isso significa que o desmonte falhou porque o sistema de arquivos está sendo acessado. Muito provavelmente, um dos diretórios no sistema de arquivos NFS é o diretório atual do seu shell (ou o shell de outra pessoa em seu sistema). A outra possibilidade é que um comando está segurando um arquivo aberto no sistema de arquivos NFS (como um editor de texto). Verifique suas janelas Terminal e outros shells e depois use `cd` para sair do diretório, se você estiver nele, ou simplesmente feche as janelas Terminal.

Se um sistema de arquivos NFS não desmontar, você pode forçá-lo (`umount -f /mnt/maple`) ou desmontar e limpar mais tarde (`umount -l /mnt/maple`). A opção `-l` é geralmente a melhor escolha, porque uma desmontagem forçada pode atrapalhar uma modificação de arquivo que está em andamento. Outra alternativa é executar `-v pontoDeMontagem` para ver quais usuários estão segurando seu compartilhamento NFS montado aberto e, depois, `fuser -k pontoDeMontagem` para eliminar todos esses processos.

Resumo

O Network File System (NFS) é um dos mais antigos produtos de compartilhamento de arquivos existentes hoje. Ele ainda é o mais popular para compartilhar diretórios de arquivos entre sistemas UNIX e Linux. O NFS permite que os servidores designem diretórios específicos para serem disponibilizados a hosts designados e então permite que sistemas clientes se conectem a esses diretórios montando-os localmente.

O NFS pode ser protegido usando regras de firewall (`iptables`), TCP wrappers (para permitir e negar acesso ao host) e o SELinux (para limitar a forma como protocolos de compartilhamento de arquivos podem compartilhar recursos do NFS). Embora o NFS fosse inerentemente inseguro quando foi criado (os dados são compartilhados sem criptografia e o acesso do usuário é bastante aberto), novos recursos do NFS versão 4 têm ajudado a melhorar a segurança geral do NFS.

Este capítulo sobre NFS é o último dos capítulos do livro sobre servidor. O Capítulo 21 abrange uma

ampla gama de temas sobre desktop e servidor à medida que ajuda a compreender as técnicas para a solução de problemas de seu sistema Linux.

Exercícios

Os exercícios desta seção vão guiá-lo ao longo das tarefas relacionadas com a configuração e utilização de um servidor NFS em Linux. Se possível, tenha dois sistemas Linux disponíveis que estejam conectados em uma rede local. Um desses sistemas Linux agirá como um servidor NFS, enquanto o outro será um cliente NFS.

Para obter o máximo desses exercícios, recomendo que você não use um servidor Linux que já tenha NFS já instalado e funcionando. Você não pode fazer todos os exercícios aqui sem interromper um serviço NFS que já está em execução e compartilhando recursos.

Consulte o Apêndice B para obter as soluções sugeridas.

1. No sistema Linux que você quer usar como um servidor NFS, instale os pacotes necessários para configurar um serviço NFS.
2. No servidor NFS, liste os arquivos de documentação que vêm no pacote que fornece o software do servidor NFS.
3. No servidor NFS, determine o nome do serviço NFS e inicie-o.
4. No servidor NFS, verifique o status do serviço NFS que você acabou de iniciar.
5. No servidor NFS, crie o diretório `/var/mystuff` e compartilhe-o a partir de seu servidor NFS com os seguintes atributos: disponível para todos, somente leitura e o usuário root no cliente tem acesso de root ao compartilhamento.
6. No servidor NFS, certifique-se de que o compartilhamento que você criou é acessível a todos os hosts abrindo TCP wrappers, iptables e SELinux.
7. Em um segundo sistema Linux (cliente NFS), visualize as ações disponíveis a partir do servidor NFS. (Se não tiver um segundo sistema, você pode fazer isso a partir do mesmo sistema.) Se você vir o diretório NFS compartilhado, volte à pergunta anterior e tente novamente.
8. No cliente NFS, crie um diretório chamado `/var/remote` e temporariamente monte o diretório `/var/mystuff` a partir do servidor NFS nesse ponto de montagem.
9. No cliente NFS, desmonte `/var/remote`, adicione uma entrada para que essa mesma montagem seja feita automaticamente quando você reiniciar (com uma opção de montagem `bg`) e teste se essa entrada que você criou está funcionando corretamente.
10. A partir do servidor NFS, copie alguns arquivos para o diretório `/var/mystuff/`. A partir do cliente NFS, certifique-se de que você pode ver os arquivos recém-adicionados a esse diretório e tenha certeza de que você não possa gravar arquivos no diretório do cliente.

Solução de problemas do Linux

NESTE CAPÍTULO

Solucionando problemas com gerenciadores de inicialização

Solucionando problemas de inicialização do sistema

Corrigindo problemas de empacotamento de software

Verificando questões de placa de rede

Lidando com problemas de memória

Usando o modo de recuperação

Em qualquer sistema operacional complexo, há muitas coisas que podem dar errado. Talvez você não consiga salvar um arquivo por falta de espaço em disco. Um aplicativo pode falhar porque o sistema está sem memória. O sistema pode falhar em inicializar corretamente por, bem, uma série de razões diferentes.

No Linux, a dedicação à abertura e o foco em fazer o software funcionar com a máxima eficiência levou a um número surpreendente de ferramentas que você pode usar para solucionar todos os problemas imagináveis. Na verdade, se o software não estiver funcionando como gostaria, você ainda tem como última alternativa reescrever o código (apesar de não abordarmos a forma de fazer isso aqui).

Este capítulo aborda alguns dos problemas mais comuns com que você pode se deparar em um sistema Linux e descreve as ferramentas e procedimentos que você pode usar para superar esses problemas. Os tópicos são divididos por áreas de solução de problemas, tais como o processo de inicialização, pacotes de software, rede, problemas de memória e o modo de recuperação.

Solucionando problemas de inicialização

Antes de você propriamente começar a solucionar problemas de um sistema rodando Linux, esse sistema precisa inicializar. Para um sistema Linux inicializar, uma série de coisas tem de acontecer. Um sistema Linux instalado diretamente em um computador de arquitetura PC passa pelos seguintes passos para

inicializar:

- Ligar a energia
- Iniciar o hardware da BIOS
- Encontrar a localização do carregador de inicialização (*boot loader*) e iniciá-lo
- Escolher um sistema operacional a partir do carregador de inicialização
- Iniciar o kernel e o disco de RAM inicial para o sistema operacional selecionado
- Iniciar o processo `init`
- Iniciar todos os scripts de inicialização e serviços associados com o nível selecionado de atividade (`runlevel` ou `default target`)

As atividades exatas que ocorrem em cada um desses pontos estão passando por uma transformação. Carregadores de inicialização estão mudando para acomodar novos tipos de hardware. O processo de inicialização está mudando para permitir que os sistemas façam o ajuste fino da ordem em que os serviços iniciam e param.

Para ajudar a entender os passos básicos que ocorrem no processo de inicialização, as próximas seções acompanham o processo de inicialização de um sistema Red Hat Enterprise Linux 6. Embora os detalhes desse processo sejam diferentes para os sistemas Fedora e Ubuntu mais recentes, você vai seguir os mesmos passos básicos para a solução de problemas do processo de inicialização.

O sistema operacional Red Hat Enterprise Linux 6 usa muitos componentes para inicializar o sistema, os quais todos existem há bastante tempo. O gerenciador de boot GRUB para o RHEL 6 ainda não migrou para a nova interface GRUB 2. Embora você possa esperar que uma versão mais recente do daemon `init`, que dirige a iniciação e parada de serviços, migre para o novo estilo de inicialização do `systemd` que está agora no Fedora, o processo `init` do RHEL 6 (que utiliza o novo processo `init`) ainda suporta os scripts de inicialização mais antigos do System V (usado para iniciar serviços do sistema).

A solução de problemas do processo de inicialização no RHEL 6 começa quando você liga o computador e termina quando todos os serviços estão funcionando. Nesse ponto, há tipicamente um login gráfico ou um prompt baseado em texto disponível a partir do console, pronto para você fazer login. Percorra as seguintes seções ordenadas para entender o que acontece em cada fase do processo de inicialização e onde você pode precisar solucionar problemas.

Começando pela BIOS

BIOS, que significa Basic Input Output System, é o primeiro código a ser executado quando você liga seu PC. Sua principal tarefa é inicializar o hardware e então passar o controle do processo de inicialização para um carregador de inicialização. Uma vez que o sistema operacional está instalado, normalmente você deve apenas deixar a BIOS fazer seu trabalho e não interrompê-la.

Há, porém, ocasiões em que você quer interromper a BIOS. Logo depois de ligar a energia, você deve ver uma tela da BIOS que normalmente inclui algumas palavras mostrando como entrar no modo de

configuração e alterar a ordem de inicialização. Se pressionar a tecla de função mostrada (muitas vezes F1, F2 ou F12) para escolher um desses dois itens, eis o que você pode fazer:

- **Setup utility** — O utilitário de configuração permite alterar as configurações da BIOS. Essas configurações podem ser usadas para ativar ou desativar determinados componentes de hardware ou ligar ou desligar as funções de hardware selecionadas.
- **Boot order** — Os computadores são capazes de iniciar um sistema operacional ou, mais especificamente, um carregador de inicialização que pode iniciar um sistema operacional a partir de vários dispositivos diferentes conectados ao computador. Esses dispositivos podem incluir uma unidade de CD, DVD, disco rígido, controlador USB ou placa de rede. A ordem de inicialização configura a ordem em que esses dispositivos são verificados. Ao modificar a ordem de inicialização, você pode dizer ao computador para ignorar temporariamente a ordem padrão e tentar inicializar a partir do dispositivo que você selecionar.

Para a minha estação de trabalho Dell, depois de ver a tela da BIOS, eu imediatamente pressiono a tecla de função F2 a fim de ir para a tela Setup ou F12 para alterar temporariamente a ordem de inicialização. As próximas seções exploram os problemas que você pode solucionar a partir das telas Setup e Boot Order.

Solucionando problemas de configuração da BIOS

Como já observado, geralmente você pode deixar que a BIOS inicie sem interrupção e que o sistema inicialize para o dispositivo de inicialização padrão (provavelmente o disco rígido). Mas aqui estão alguns exemplos que podem lhe fazer querer entrar no modo Setup e mudar alguma coisa na BIOS.

- **Ter uma visão geral de seu hardware** — Se seu problema estiver relacionado com o hardware, a configuração da BIOS é um ótimo lugar para começar a examinar seu sistema. A tela de instalação informará o tipo de sistema, sua versão de BIOS, seus processadores, slots e tipos de memória, se ele é de 32 ou 64 bits, quais dispositivos estão em cada slot e muitos detalhes sobre os tipos de dispositivos conectados ao sistema.

Se você não conseguir fazer um sistema operacional inicializar, a tela BIOS Setup pode ser a única maneira de determinar o modelo do sistema, o tipo de processador e outras informações que você precisa procurar para obter ajuda ou ligar para o suporte.

- **Ativar/desativar um dispositivo** — A maioria dos dispositivos conectados ao computador estão habilitados e disponíveis para uso pelo sistema operacional. Para solucionar um problema, pode ser necessário desativar um dispositivo.

Por exemplo, digamos que seu computador tem duas placas de rede. Você quer usar a segunda placa de rede para instalar o Linux em uma rede, mas o instalador continua tentando usando a primeira para se conectar à rede. Você pode desabilitar a primeira placa de rede para que o instalador nem mesmo a veja ao tentar se conectar à rede.

Talvez você tenha uma placa de áudio e queira desativar o áudio integrado na placa-mãe. Isso pode ser feito na BIOS também.

Por outro lado, pode haver momentos em que você quer ativar um dispositivo que foi desativado. Talvez você tenha recebido um computador com um dispositivo desativado na BIOS. A partir do

sistema operacional, por exemplo, pode parecer que você não tem uma porta paralela (LPT) ou uma unidade de CD. Examinando a BIOS, você pode determinar se esses dispositivos não estão disponíveis simplesmente porque foram desativados na BIOS.

- **Alterar a configuração do dispositivo** — Às vezes, as configurações padrão que vêm em sua BIOS não funcionam para sua situação. Você pode querer alterar as seguintes configurações da BIOS:
- **Configurações PXE de inicialização da placa de rede** — As placas de rede mais modernas são capazes de inicializar a partir de servidores encontrados na rede. Se precisar fazer isso e achar que a placa de rede não vem como um dispositivo inicializável na tela Boot Order, você pode ter de habilitar esse recurso na BIOS.
- **Configurações de virtualização** — Se você quiser executar um sistema RHEL 6 como uma máquina virtual, a CPU do computador deve incluir suporte à Intel Virtual Technology ou à AMD Secure Virtual Machine (SVM). É possível, porém, que, mesmo que sua CPU tenha esse suporte, este não esteja habilitado na BIOS. Para ativá-lo, vá para a tela BIOS Setup e procure por uma seleção de virtualização (possivelmente sob a categoria Performance). Verifique se ela está configurada como On.

Solucionando problemas de ordem de inicialização

Dependendo do hardware conectado ao computador, uma ordem de inicialização típica pode carregar um CD/DVD em primeiro lugar, depois, uma unidade de disquete, o disco rígido, um dispositivo USB, e, por fim, a placa de rede. A BIOS iria a cada dispositivo à procura de um carregador de inicialização. Se a BIOS encontrar um carregador de inicialização, ela o inicia, se não, a BIOS passa para o próximo dispositivo, até que todos sejam tentados. Se nenhum carregador de inicialização for encontrado, o computador não inicializa.

Um problema que pode ocorrer com a ordem de inicialização é que o dispositivo que você quer inicializar pode simplesmente não aparecer durante ela. Nesse caso, ir para a tela Setup, como descrito na seção anterior, para ativar o dispositivo ou alterar uma configuração a fim de torná-lo inicializável, pode ser a coisa a fazer.

Se o dispositivo a partir do qual você quer inicializar aparecer na ordem de inicialização, normalmente você só tem de mover a seta para destacar o dispositivo que você quer e pressionar Enter. O que se segue são razões para escolher seu próprio dispositivo para inicializar:

- **Modo de recuperação** — Se o Linux não inicializar a partir do disco rígido, selecionar a unidade de CD ou uma unidade USB permite que você inicialize em um modo de recuperação (descrito mais adiante, neste capítulo), que pode ajudá-lo a reparar o disco rígido em um sistema que não inicializa. Consulte a seção “Solucionando problemas no modo de recuperação”, mais adiante, neste capítulo, para obter mais informações.
- **Nova instalação** — Às vezes, a ordem de inicialização tem o disco rígido listado primeiro. Se decidir que precisa fazer uma nova instalação do sistema operacional, você terá de selecionar o dispositivo de inicialização que contém sua mídia de instalação (CD, DVD, unidade USB ou placa de rede).

Supondo que você já tenha verificado todos os problemas que podem ter a ver com a BIOS, o próximo passo é a BIOS iniciar o carregador de inicialização.

Solucionando problemas do carregador de inicialização GRUB

Normalmente, a BIOS encontra o registro de inicialização mestre (*master boot record*, MBR) do primeiro disco rígido e começa a carregar esse carregador em etapas. O Capítulo 9, “Instalando o Linux”, descreve o GRUB usado nos mais modernos sistemas Linux, incluindo RHEL, Fedora e Ubuntu. O carregador de inicialização GRUB no RHEL, descrito aqui, é uma versão mais antiga do que o carregador de inicialização GRUB 2 incluído no Fedora e no Ubuntu.

Nesta discussão, estou interessado no carregador de inicialização a partir da perspectiva do que fazer se ele falhar ou de que maneira você pode querer interrompê-lo para alterar o comportamento do processo de inicialização.

Eis algumas maneiras como o carregador de inicialização pode falhar no RHEL 6 e algumas maneiras de você superar essas falhas:

- **Não foi possível localizar a partição ativa** — Quando um carregador de inicialização é instalado em uma mídia de armazenamento, a partição é normalmente marcada como inicializável. Se você vir essa mensagem, isso significa que a partição de inicialização foi encontrada. Se você achar que o carregador de inicialização está no disco, tente usar o comando `fdisk` (provavelmente, a partir de uma mídia de recuperação) para criar a partição de inicialização e tente novamente. Veja o “Particionando os discos rígidos”, do Capítulo 12, “Gerenciando discos e sistemas de arquivos”, para mais informações sobre o comando `fdisk`.
- **Dispositivo de inicialização selecionado não disponível** — Uma mensagem como essa pode aparecer quando o MBR é apagado do disco rígido. Ou pode ser apenas que o conteúdo do disco rígido espera ser carregado a partir de outro carregador de inicialização, como um CD de inicialização. Primeiro, tente ver se o sistema inicializa a partir de outras mídias. Se descobrir que o MBR foi apagado, você pode tentar inicializar a mídia de recuperação para tentar recuperar o conteúdo do disco. Mas se o MBR tiver sido perdido, é possível que outros dados no disco também tenham sido apagados ou exijam uma perícia técnica para serem localizados. Se o MBR simplesmente tiver sido sobrescrito (o que poderia acontecer se você tiver instalado outro sistema operacional em uma partição de disco diferente) talvez você possa reinstalá-lo a partir do modo de recuperação (descrito na seção “Solucionando problemas no modo de recuperação”, mais adiante, neste capítulo).
- **Prompt baseado em texto do GRUB aparece** — É possível que a BIOS inicie o GRUB e vá direto para um prompt do GRUB, sem dar nenhuma opção de selecionar um sistema operacional disponível. Isso provavelmente significa que o MBR do GRUB não foi encontrado, mas quando o GRUB examinou o disco rígido para encontrar a próxima fase do processo de inicialização e um menu de sistemas operacionais para carregar, ele não conseguiu encontrá-los. Às vezes, isso acontece quando a BIOS detecta os discos na ordem errada e olha para o arquivo `grub.conf` na partição errada.

Uma solução para esse problema, assumindo que `grub.conf` está na primeira partição do

primeiro disco, é listar o conteúdo desse arquivo e inserir as linhas `root`, `kernel` e `initrd` manualmente. Para listar o arquivo, digite `cat (hd0,0)/grub/grub.conf`. Se isso não funcionar, tente `hd0,1` para acessar a próxima partição no disco (e assim sucessivamente) ou `hd1,0` para tentar a primeira partição do disco seguinte (e assim sucessivamente). Depois de encontrar as linhas que representam o arquivo `grub.conf`, digite manualmente as linhas `root`, `kernel` e `initrd` para a entrada que você quer (substituindo a localização do disco rígido que você encontrou na linha raiz). Então, digite `boot`. O sistema deve iniciar e você pode ir e corrigir manualmente os arquivos do carregador de inicialização. Consulte o Capítulo 9 para mais informações sobre o carregador de inicialização GRUB.

Se a BIOS encontrar o carregador de inicialização no MBR do disco e esse carregador encontrar os arquivos de configuração do GRUB no disco, ele inicia uma contagem regressiva de cerca de três a cinco segundos. Durante essa contagem, você pode interromper o carregador de inicialização (antes de ele iniciar o sistema operacional padrão), pressionando qualquer tecla.

Quando interromper o carregador de inicialização, você deverá ver um menu de entradas disponíveis para inicializar. Essas entradas podem representar diferentes kernels disponíveis para inicializar. Mas eles também podem representar sistemas operacionais totalmente diferentes (como o Windows, BSD ou Ubuntu).

Eis algumas razões para interromper o processo de inicialização a partir do menu de inicialização para solucionar problemas do Linux:

- **Iniciar em um nível de execução diferente** — Sistemas RHEL 6 tipicamente iniciam no runlevel 3 (inicializa para um prompt de texto) ou 5 (inicializa para uma interface gráfica). Você pode substituir o nível de execução padrão, colocando um número diferente de nível de execução no final da linha do kernel no menu de inicialização. Para fazer isso, selecione a entrada do sistema operacional que você quer e digite **e**, destaque o kernel e digite **e**, e adicione o novo nível de execução ao final da linha (por exemplo, adicione um espaço e o número 1 para ir para o modo monousuário). Então, pressione Enter e digite **b** para inicializar a nova entrada.

Por que você iria inicializar para diferentes níveis de execução para solução de problemas? Runlevel 1 ignora autenticação, assim, você pode inicializar diretamente para um prompt de root. Isso é bom se você esqueceu a senha de root e precisa mudar (digite `passwd` para fazer isso). Runlevel 3 pula o carregamento de sua interface de desktop. Vá para o runlevel 3, se você estiver tendo problemas com o driver de vídeo e quiser tentar depurá-lo sem ele tentar iniciar automaticamente a interface gráfica.

- **Selecionar um kernel diferente** — Quando o RHEL instala um novo kernel via `yum`, ele sempre mantém pelo menos um kernel antigo. Se o novo kernel falhar, você sempre pode iniciar o kernel anterior, presumivelmente funcional. Para inicializar um kernel diferente a partir do menu do GRUB, basta usar a seta para destacar o que você quer.
- **Selecionar um sistema operacional diferente** — Se tiver outro sistema operacional instalado em seu disco rígido, você pode escolher inicializá-lo em vez do RHEL. Por exemplo, se você tiver o Fedora e o RHEL no mesmo computador e o RHEL não estiver funcionando, você pode inicializar o Fedora, montar os sistemas de arquivos RHEL de que você precisa e tentar corrigir o problema.

- **Alterar as opções de inicialização** — Na linha do kernel, você vai notar que há uma série de opções que são passadas para o kernel. No mínimo, essas opções devem conter o nome do kernel (como `vmlinuz-2.6.32.el6.x86_64`) e a partição que contém o sistema de arquivos raiz (como `/dev/mapper/abc-root`). Se quiser, você pode adicionar outras opções à linha do kernel. Você pode querer adicionar opções de kernel para adicionar recursos ao kernel ou desativar temporariamente o suporte a hardware para um determinado componente. Por exemplo, adicionar `init=/bin/bash` faz o sistema impedir o processo de inicialização e ir direto para um shell (semelhante à execução de `init 1`). Adicionar `nousb` desativa temporariamente as portas USB (presumivelmente para garantir qualquer coisa conectada a essas portas também seria desativada).

Supondo que você tenha selecionado o kernel que você quer, o carregador de inicialização tenta executá-lo, incluindo o conteúdo do disco de RAM inicial (que contém os drivers e outros softwares necessários para inicializar seu hardware específico).

Iniciando o kernel

Depois que o kernel inicia, não há muito a fazer senão observar potenciais problemas. No RHEL, você verá uma tela Red Hat Enterprise Linux com um ícone girando lentamente. Se você quiser observar as mensagens detalhando o processo de inicialização rolando na tela, pressione a tecla Esc.

Nesse ponto, o kernel tenta carregar os drivers e módulos necessários para usar o hardware no computador e as principais coisas a procurar (embora elas possam rolar rapidamente) são falhas de hardware que podem impedir algum recurso de funcionar adequadamente. Embora muito mais raro do que costumava ser, pode não haver driver disponível para um hardware ou a unidade errada pode estar sendo carregada e causando erros.

Além de rolar na tela, as mensagens produzidas quando o kernel inicia são copiadas para o *buffer do kernel*. Como o próprio nome indica, o buffer do kernel armazena as mensagens do kernel em um buffer, descartando mensagens mais antigas quando o buffer está cheio. Depois que o computador inicializa completamente, você pode entrar no sistema e digitar o seguinte comando para capturar essas mensagens do kernel em um arquivo (e, então, examiná-las com o comando `less`):

```
# dmesg > /tmp/kernel_msg.txt
# less /tmp/kernel_msg.txt
```

Eu gosto de direcionar as mensagens do kernel para um arquivo (escolha o nome que quiser) para que as mensagens possam ser examinadas mais tarde ou enviadas para alguém que pode ajudar a depurar problemas. As mensagens aparecem como componentes que são detectados, tais como CPU, memória, placas de rede, discos rígidos etc.

O que você quer examinar são mensagens de falha de carregamento de driver ou de falha de ativação de certos recursos do hardware. Por exemplo, certa vez eu tinha uma placa sintonizadora de TV (para assistir televisão na tela do meu computador) que estava configurando o tipo errado de sintonizador para a placa que foi detectada. Usando as informações sobre o modelo da placa de TV e o tipo de falha, descobri que passando uma opção para o driver da placa me permitia experimentar diferentes

configurações até eu encontrar a que correspondia com minha placa de TV.

Ao descrever como visualizar as mensagens de inicialização do kernel, adiantei-me um pouco. Antes que você possa entrar e ver as mensagens do kernel, ele precisa terminar de carregar o sistema. Assim que o kernel termina a detecção e o carregamento iniciais dos drivers de hardware, ele passa o controle de tudo o mais que precisa ser feito para inicializar o sistema ao processo `init`.

Solucionando problemas do processo `init`

O processo `init` do Red Hat Enterprise Linux versão 6 está atualmente migrando do conhecido estilo System V de iniciar os serviços para uma versão mais recente do processo `init`, chamada Upstart. Nesta seção, descreverei como o antigo processo `init` funciona e, então, mencionarei como isso é diferente em versões mais recentes do Fedora e provavelmente versões posteriores do RHEL. (O Capítulo 15, “Iniciando e parando serviços”, contém mais detalhes sobre o processo de `init` e scripts de inicialização.)

No RHEL, quando o kernel passa o controle do processo de inicialização para o processo `init`, este verifica o arquivo `/etc/inittab` para obter instruções sobre como carregar o sistema. O arquivo `inittab` informa ao processo `init` qual é o nível de execução padrão e, então, aponta para arquivos no diretório `/etc/init` a fim de fazer coisas como remapear algumas teclas (como Ctrl+Alt+Delete para reiniciar o sistema), iniciar consoles virtuais e identificar o local do script para inicializar serviços básicos no sistema: `/etc/rc.sysinit`.

Quando você estiver solucionando problemas no Linux que ocorrem depois que o processo `init` assume, dois prováveis culpados são o processamento do arquivo `rc.sysinit` e os scripts de `runlevel`.

Solucionando problemas do script `rc.sysinit`

Como o nome indica, o script `/etc/rc.sysinit` inicializa muitos recursos básicos do sistema. Quando o arquivo é executado por `init`, `rc.sysinit` configura o nome do host do sistema, configura os sistemas de arquivos `/proc` e `/sys`, configura o SELinux, configura os parâmetros do kernel e realiza dezenas de outras ações.

Uma das funções mais importantes de `rc.sysinit` é configurar o armazenamento no sistema. De fato, se o processo de inicialização falhar durante o processamento de `rc.sysinit`, muito provavelmente o script não foi capaz de encontrar, montar ou decodificar os dispositivos de armazenamento local ou remoto necessários para o sistema funcionar. A seguir está uma lista de algumas falhas comuns que podem ocorrer em decorrência de tarefas executadas a partir do arquivo `rc.sysinit` e maneiras de lidar com essas falhas.

- **Montagens locais falham** — Se ocorrer uma falha de montagem de uma entrada no arquivo `/etc/fstab`, o processo de inicialização termina antes que os serviços de `runlevel` iniciem. Isso normalmente acontece quando você adiciona uma entrada com erro ao arquivo `/etc/fstab`, mas esquece de testá-la antes de reiniciar. Quando o arquivo `fstab` falha, você é levado para um shell do usuário `root` com o sistema de arquivos raiz montado somente para

leitura. Para corrigir o problema, é preciso remontar o sistema de arquivos raiz, corrigir o arquivo `fstab`, montar a entrada do sistema de arquivos para ter certeza de que agora ela funciona e reiniciar. Eis como se parece a sequência de comandos:

```
# mount -o remount,rw /
# vim /etc/fstab
# mount -a
# reboot
```

ta

O comando `vim` é usado particularmente ao editar o arquivo `/etc/fstab` porque ele conhece o formato do arquivo. Quando você usa `vim`, as colunas aparecem em cores e uma verificação de erros é feita. Por exemplo, as palavras no campo Mount Options tornam-se verdes quando são válidas e pretas quando não.

- **Hostname não definido** — Se sua máquina não estiver configurada corretamente, você pode verificar o processamento de `rc.sysinit` para ver o que pode ter dado errado. Para configurar o hostname do sistema, `rc.sysinit` usa o valor de `HOSTNAME=` no arquivo `/etc/sysconfig/network`. Se isso não for configurado, o nome `localhost` é usado. O valor de hostname também pode ser adquirido a partir do servidor DHCP.
- **Não é possível descriptografar o sistema de arquivos** — O script `rc.sysinit` procura no arquivo `/etc/crypttab` informações necessárias para descriptografar sistemas de arquivos criptografados. Se esse arquivo estiver corrompido, você pode precisar encontrar um backup dele para ser capaz de descriptografar o sistema de arquivos. Se for solicitado a fornecer uma senha e não souber qual é, você pode estar sem sorte.

Outros recursos também são configurados pelo arquivo `rc.sysinit`. O script `rc.sysinit` configura o modo do SELinux e carrega módulos de hardware. O roteiro constrói arrays RAID de software e configura grupos de volumes e volumes LVM (Logical Volume Management). Problemas que ocorrem em qualquer uma dessas áreas são refletidos em mensagens de erro que aparecem na tela após a inicialização do kernel e antes que processos de runlevel iniciem.

Solucionando problemas de processos de runlevel

No Red Hat Enterprise Linux 6.x e versões anteriores, quando o sistema inicializa pela primeira vez, os serviços são iniciados com base no nível de execução padrão. Há sete diferentes níveis de execução, de 0 a 6. O nível de execução padrão é tipicamente 3 (para um servidor) ou 5 (para um desktop). Eis as descrições dos níveis de execução em sistemas Linux até o RHEL 6:

- **0** — Nível de execução desligado. Todos os processos estão parados e o computador está desligado.
- **1** — Nível de execução monousuário. Apenas os processos que são necessários para inicializar o computador (incluindo montagem de todos os sistemas de arquivos) e ter o sistema disponível a partir do console são executados. Redes e serviços de rede não são iniciados. Esse nível de

execução ignora a autenticação normal e inicializa até um prompt do usuário root (chamado `sulogin`). Se inicializar nesse modo, você pode usá-lo para se tornar imediatamente o usuário root para alterar a senha de root perdida. (Você também pode usar a palavra `single` em vez de `1` para chegar a um nível de execução monousuário. A diferença entre `single` e `1` é que `single` não inicia scripts no diretório `/etc/rc1.d`)

- **2** — Nível de execução multiusuário. Esse nível de execução é raramente utilizado hoje e seu significado original foi perdido. Os primeiros sistemas UNIX utilizavam-no para iniciar processos `tty` para sistemas em que havia vários terminais burros conectados para as pessoas usarem. Isso permitia que muitas pessoas acessassem um sistema simultaneamente a partir de terminais baseados em caractere (várias pessoas trabalhando a partir de um shell sem interface gráfica). As placas de rede não eram iniciadas, geralmente porque as placas de rede sempre ativas não eram comuns. Hoje em dia, o nível de execução 2 normalmente iniciará as placas de rede, embora nem todos os serviços de rede sejam iniciados.
- **3** — Nível de execução multiusuário mais rede. Esse nível de execução é normalmente usado em servidores Linux que não inicializam para uma interface gráfica, mas sim apenas um prompt de texto simples no console. A rede é iniciada, assim como o são todos os serviços de rede. Um ambiente gráfico pode ou não ser instalado (geralmente não) em máquinas que inicializam no nível de execução 3, mas os ambientes gráficos devem ser iniciados após a inicialização para serem utilizados.
- **4** — Não definido. Esse nível de execução tende a iniciar os mesmos serviços que o nível de execução 3. Ele pode ser usado se você quiser ter diferentes serviços disponíveis a partir dos níveis de execução 3 e 4. Esse nível de execução não é normalmente usado. Em vez disso, o nível de execução 3 ou 5 é usado para inicializar, com um administrador simplesmente ativando ou desativando serviços conforme a necessidade do sistema em funcionamento.
- **5** — Nível de execução multiusuário, rede, mais interface gráfica. Esse é o nível de execução tipicamente usado com sistemas desktop Linux. Ele normalmente inicia a rede e todos os serviços dela, além disso, carrega um prompt de login gráfico no console. Ao fazer login, os usuários veem um ambiente de desktop gráfico.
- **6** — Nível de execução de reinicialização. Esse é como o nível de execução 0 no sentido de que derruba todos os serviços e interrompe todos os processos. Mas o nível de execução 6, então, inicia o sistema novamente.

Níveis de execução destinam-se a configurar o nível de atividade em um sistema Linux. Um nível de execução padrão é configurado no arquivo `/etc/inittab`, mas você pode alterá-lo sempre que quiser usar o comando `init`. Por exemplo, como root, você pode digitar **`init 0`** para desligar, **`init 3`** se quiser eliminar a interface gráfica (a partir do runlevel 5), mas deixar todos os outros serviços ativos, ou **`init 6`** para reiniciar.

Os níveis de execução padrões normais (em outras palavras, o nível de execução para o qual você inicializou) são 3 (para um servidor) e 5 (para um desktop). Muitas vezes, os servidores não têm desktops instalados e inicializam no nível de execução 3 porque não querem incorrer em sobrecarga de processamento ou nos riscos adicionais de segurança de ter um desktop rodando em seus servidores web

ou servidores de arquivos.

Você pode ir para cima ou para baixo nos níveis de execução. Por exemplo, um administrador fazendo manutenção em um sistema pode inicializar para um nível de execução e digitar `init 3` a fim de inicializar para os serviços completos necessários em um servidor. Alguém depurando um desktop pode inicializar para o nível de execução 5 e depois descer para o nível de execução 3 para tentar corrigir o desktop (como instalar um novo driver ou mudar a resolução de tela) antes de digitar `init 5` para voltar ao desktop.

O nível dos serviços em cada nível de execução é determinado pelos scripts de runlevel que estão configurados para iniciar. Há diretórios `rc` para cada nível de execução: `/etc/rc0.d/`, `/etc/rc1.d/`, `/etc/rc2.d/`, `/etc/rc3.d/` etc. Quando um aplicativo tem um script de inicialização associado a ele, esse script é colocado no diretório `/etc/init.d/` e, então, simbolicamente vinculado a um arquivo em cada diretório `/etc/rc?.d/`.

Scripts vinculados a cada diretório `/etc/rc?.d` começam com a letra `K` ou `S`, seguidas de dois números e o nome do serviço. Um script começando com `K` indica que o serviço deve ser interrompido, enquanto um começando com um `S` indica que ele deve ser iniciado. Os dois números que se seguem indicam a ordem em que o serviço é iniciado. Eis alguns arquivos que você pode encontrar no diretório `/etc/rc3.d/`, que são configurados para iniciar (com uma descrição de cada um à direita):

- `S01sysstat` — Inicia a coleta de estatísticas do sistema.
- `S08iptables` — Inicia o firewall `iptables`.
- `S10network` — Inicia as placas de rede.
- `S12rsyslog` — Inicia o log do sistema.
- `S28autofs` — Inicia o montador.
- `S50bluetooth` — Inicia o serviço Bluetooth.
- `S55sshd` — Inicia o serviço secure shell.
- `S58ntpd` — Inicia o serviço de sincronização de hora NTP.
- `S85httpd` — Inicia o serviço web Apache.
- `S90crond` — Inicia o serviço `crond`.
- `S91smb` — Inicia o serviço samba.
- `S97rhnsd` — Inicia o serviço do Red Hat Network.
- `S99local` — Inicia comandos locais definidos pelo usuário.

Esse exemplo de alguns serviços iniciados a partir do diretório `/etc/rc3.d` deve dar uma noção da ordem em que os processos inicializam quando você entra no nível de execução 3. Observe que o serviço `sysstat` (que coleta as estatísticas do sistema) e o serviço `iptables` (que cria o sistema de firewall)

são ambos iniciados antes das placas de rede. Estes são seguidos por `rsyslog` (serviço de log do sistema) e, depois, pelos vários serviços de rede.

No momento em que os scripts de nível de execução começam, você já deve ter um sistema basicamente funcional. Ao contrário de alguns outros sistemas Linux que iniciam todos os scripts de nível de execução 1, depois 2, depois 3 etc., o RHEL vai direto para o diretório que representa o nível de execução, primeiro parando todos os serviços que começam com `K` e iniciando todos aqueles que começam com `S` no diretório.

À medida que cada script `S` é executado, você verá uma mensagem dizendo se o serviço foi iniciado. Eis algumas coisas que podem dar errado durante essa fase de inicialização do sistema:

- **Um serviço pode falhar.** Um serviço pode exigir o acesso a placas de rede para iniciar corretamente ou o acesso a uma partição de disco que não está montada. A maioria dos serviços irá expirar, falhar e permitir que o próximo script seja executado. Depois que você for capaz de efetuar o login, poderá depurar o serviço. Algumas técnicas de depuração de serviços incluem adicionar uma opção de depuração ao processo `daemon` para ele fornecer mais dados em um arquivo de log ou executar o processo `daemon` manualmente, de modo que as mensagens de erro venham diretamente para sua tela. Veja o Capítulo 15 para obter mais informações sobre como iniciar serviços manualmente.
- **Um serviço pode travar.** Alguns serviços que não recebem o que precisam para iniciar podem travar indefinidamente, impedindo o registro em log para depurar o problema. Alguns processos levam mais tempo para subir pela primeira vez após uma nova instalação, então, você pode querer esperar por alguns minutos para ver se o script ainda está funcionando e não apenas girando em loop infinito.

Se não puder passar por um serviço travado, você pode reiniciar em um *modo de inicialização interativo*, no qual você é perguntado se quer iniciar cada serviço. Para entrar no modo de inicialização interativo no RHEL, reinicialize e interrompa o carregador de inicialização (pressione qualquer tecla quando vir a contagem regressiva de 5 segundos). Destaque a entrada que você quer iniciar e digite **e**. Destaque a linha do kernel e digite **e**. Então, adicione a palavra `confirm` ao final da linha do kernel, pressione Enter e digite **b** para iniciar o novo kernel.

A Figura 21.1 mostra um exemplo das mensagens que aparecem quando o RHEL inicia no modo de inicialização interativo.

rme cada serviço no modo de inicialização interativo do RHEL.

A maioria das mensagens mostradas na Figura 21.1 é gerada a partir de `rc.sysinit`.

Após a mensagem de boas-vindas, `udev` inicia (para cuidar de qualquer novo hardware que seja conectado ao sistema e carregar drivers quando necessário). O `hostname` é configurado, os volumes LVM (Logical Volume Management) são ativados, todos os sistemas de arquivos são verificados (com os volumes LVM adicionados), quaisquer sistemas de arquivos ainda não montados são montados, o sistema de arquivos raiz é remontado para leitura e gravação e qualquer LVM de swap é habilitado. Consulte o Capítulo 12 para mais informações sobre LVM e outros tipos de partição e sistema de arquivos.

A última mensagem “Entering interactive startup” (“Entrando na inicialização interativa”) informa que `rc.sysinit` concluiu e os serviços para o nível de execução selecionado estão prontos para iniciar. Como o sistema está no modo interativo, aparece uma mensagem perguntando se você quer iniciar o primeiro serviço (`sysstat`). Digite **Y** para iniciar o serviço e vá para o próximo.

Depois de ver o serviço problemático pedindo para iniciar, digite **N** para impedir que ele seja iniciado. Se, em algum momento, você achar que é seguro iniciar os demais serviços, digite **C** para continuar a carregá-los. Depois que seu sistema subiu, com os serviços problemáticos não iniciados, você pode voltar e tentar depurar esses serviços individualmente.

Um último comentário sobre scripts de inicialização: o arquivo `/etc/rc.local` é um dos últimos serviços a executar em cada nível de execução. Como um exemplo, no nível de execução 5, ele está vinculado a `/etc/rc5.d/S99local`. Qualquer comando que você quiser executar cada vez que seu sistema iniciar pode ser colocado no arquivo `rc.local`. Você poderia usar `rc.local` para enviar uma mensagem de e-mail ou executar uma regra de firewall iptables rápida quando o sistema é iniciado. Em geral, é melhor usar um script de inicialização existente ou criar um novo por conta própria (para que você possa gerenciar o comando ou comandos como um serviço). Saiba que o arquivo `rc.local` é uma maneira rápida e fácil de fazer alguns comandos serem executados cada vez que o sistema é inicializado.

ta

dora não usa mais níveis de execução ou scripts `init` do System V para indicar qual serviço iniciar quando o sistema é inicializado e desligado. Em vez disso, ele utiliza o que se costuma chamar de unidades de destino (*targets*). Há unidades de destino `multi-user.target` (semelhantes ao nível de execução 3) e `graphical.target` (semelhantes ao nível de execução 5), que são configuradas para determinar quais serviços iniciar quando o sistema é inicializado. O novo método substitui o antigo serviço `init` pelo serviço `systemd`. (O comando `/sbin/init` ainda funciona, mas é um vínculo simbólico para `/bin/systemd`.)

O método `systemd` de iniciar serviços tem várias vantagens sobre o método anterior. Enquanto os scripts `init` do System V passam por cada serviço, um após o outro, até que todos tenham sido iniciados, `systemd` permite definir dependências mais complexas entre serviços. A vantagem é que, enquanto alguns serviços estão à espera

Depois que o último serviço inicia, aparece uma tela de login. Seu sistema deve estar totalmente operacional. Comece a usar o sistema ou continue o processo de solução de problemas, conforme necessário. A próxima seção descreve como solucionar problemas que podem surgir com seus pacotes de software.

Solucionando problemas de pacotes de software

Facilidades de empacotamento de software (como `yum` para o RPM e `apt-get` para pacotes DEB) são projetadas para que você gerencie o software do sistema mais facilmente. (Consulte o Capítulo 10, “Obtendo e gerenciando software”, para os princípios básicos sobre como gerenciar pacotes de software.) Apesar dos esforços para fazer todo o trabalho, às vezes o empacotamento de software pode falhar.

As próximas seções descrevem alguns problemas comuns que você pode encontrar com pacotes RPM em um sistema RHEL ou Fedora e como você pode superá-los.

Você tenta instalar ou atualizar um pacote usando o comando `yum` e mensagens de erro dizem que os pacotes dependentes que você precisa para fazer a instalação não estão disponíveis. Isso pode acontecer em pequena escala (ao tentar instalar um pacote) ou em grande escala (ao tentar atualizar ou fazer um upgrade do sistema inteiro).

Por causa dos curtos ciclos de release e o grande tamanho dos repositórios do Fedora e do Ubuntu, inconsistências nas dependências de pacotes são mais prováveis de ocorrer do que nos repositórios menores e mais estáveis (como os oferecidos pelo Red Hat Enterprise Linux). Para evitar falhas de dependência, eis algumas práticas recomendadas que você pode seguir:

- **Use repositórios recentes e bem-testados.** Há milhares de pacotes de software no Fedora. Se você usar os repositórios principais para instalar o software da versão atual, é raro ter problemas de dependência.

Quando os pacotes são adicionados ao repositório, desde que o mantenedor do repositório execute os comandos corretos para configurá-lo (e você não use repositórios externos), raramente ocorrerão problemas de dependência. Mas quando você começa a usar repositórios de terceiros, eles podem depender de repositórios que não podem controlar. Por exemplo, se um repositório cria uma nova versão de seu próprio software que requer versões posteriores do software básico (como bibliotecas), que são entregues com o repositório Fedora, as versões de que eles precisam podem não estar disponíveis no repositório Fedora.

- **Atualize consistentemente seu sistema.** Executar `yum update` todas as noites torna a ocorrência de grandes problemas de dependência menos provável do que se você atualizar seu sistema apenas a cada alguns meses. No Fedora, há um pacote `yum-updatesd` que permite fazer verificações noturnas de atualizações e, então, enviar e-mails para um usuário de sua escolha se houver atualizações disponíveis. No RHEL, você poderia criar um trabalho `cron` para verificar se há atualizações ou executar atualizações noturnas. Veja o quadro “Usando cron

para atualizações de software”, para detalhes sobre como fazer isso.

- **Ocasionalmente, faça um upgrade de seu sistema.** O Fedora e o Ubuntu lançam novas releases a cada seis meses. O Fedora interrompe o fornecimento de pacotes atualizados para cada versão 13 meses depois que ela é lançada. Assim, embora não precise fazer um upgrade para cada nova release a cada seis meses, você deve atualizar uma vez por ano ou enfrentar possíveis problemas de dependência e de segurança quando o Fedora parar de fornecer atualizações. Se você estiver procurando por um sistema estável, o Red Hat Enterprise Linux é a melhor aposta, porque ele fornece atualizações para cada versão principal (*major release*) por sete anos.

ta

Se você usa o comando `apt-get` no Ubuntu para atualizar seus pacotes, tenha em mente que há diferentes significados para as opções `update` e `upgrade` no Ubuntu com o comando `apt-get` do que há com o comando `yum` (no Fedora e no RHEL).

No Ubuntu, o comando `apt-get update` faz com que os metadados (nome, número de versão do pacote etc.) mais recentes dos pacotes sejam baixados para o sistema local. Executar `apt-get upgrade` faz com que o sistema atualize os pacotes instalados que possuem novas versões disponíveis, com base nos últimos metadados baixados.

Em contraposição, cada vez que você executa um comando `yum` no Fedora ou no RHEL, os últimos metadados sobre os pacotes são baixados. Ao executar `yum update`, você tem os pacotes mais recentes disponíveis para a versão atual do Fedora ou do RHEL. Quando você executar `yum upgrade`, o sistema realmente tenta fazer um upgrade para uma versão totalmente nova dessas distribuições (como do Fedora 16 para o Fedora 17).

Se você encontrar um problema de dependência, há algumas coisas que pode fazer para tentar resolvê-lo:

- **Use repositórios estáveis.** Para releases recentes de distribuições conhecidas (RHEL, Fedora ou Ubuntu, por exemplo), problemas de dependência são raros e costumam ser corrigidos rapidamente. Mas se você estiver confiando em repositórios para releases mais antigas ou repositórios voltados para desenvolvimento (como repositório rawhide do Fedora), espere encontrar mais problemas de dependência. Reinstalar ou fazer upgrade pode, muitas vezes, corrigir problemas de dependência.
- **Utilize aplicativos de terceiros e repositórios somente quando necessário.** Quanto mais longe você ficar do núcleo de uma distribuição Linux, mais provavelmente terá problemas de dependência um dia. Sempre procure sua distribuição nos principais repositórios antes de procurar um pacote em outro lugar ou tentar construir um você mesmo.
Mesmo que ele funcione quando você o instalar, um pacote que alguém simplesmente deu para você pode não ser passível de upgrade. Um pacote de um repositório de terceiros pode falhar se os criadores não fornecerem uma nova versão quando os pacotes dependentes mudarem.
- **Resolva dependências relacionadas com o kernel.** Se você pegar os pacotes RPM de terceiros para coisas como placas de vídeo ou placas de rede sem fio que contêm os drivers do kernel e instalar um kernel mais tarde, esses drivers não funcionarão mais. O resultado pode ser que a tela

de login gráfica não será iniciada quando o sistema for inicializado ou sua placa de rede não será carregada, deixando você sem rede sem fio.

Como a maioria dos sistemas Linux mantém os dois núcleos mais recentes, você pode reiniciar, interromper o GRUB e selecionar o kernel anterior (ainda funcional) para inicializar. Isso coloca seu sistema de pé, com o kernel e os drivers antigos funcionando, enquanto você procura uma correção mais permanente.

A solução de longo prazo é obter um novo driver que foi recompilado para seu kernel atual. Há sites, como o `rpmfusion.org`, que compilam pacotes de drivers não abertos de fornecedores independentes e atualizam os drivers quando um novo kernel está disponível. Com o repositório `rpmfusion.org` ativado, o sistema deve buscar os novos drivers quando o kernel for adicionado.

Como uma alternativa para sites como o `rpmfusion.org`, você pode ir direto para o site do fabricante e tentar baixar e compilar o driver você mesmo (a Nvidia oferece drivers Linux para suas placas de vídeo) ou, caso o código-fonte esteja disponível para o driver, você mesmo pode tentar compilar.

- **Exclua alguns pacotes da atualização.** Se estiver atualizando um grande número de pacotes de uma só vez, você pode excluir os pacotes que falham em fazer os outros funcionarem à medida que você tenta resolver os problemáticos. Veja como atualizar todos os pacotes que necessitam de atualização, com exceção do pacote chamado *somepackage* (substitua *somepackage* pelo nome do pacote que você quer excluir):

```
# yum -y --exclude=somepackage update
```

- **Tente fazer um pré-upgrade para os upgrades.** Se você estiver fazendo um upgrade do Fedora de uma versão para outra (por exemplo, do Fedora 16 para o Fedora 17), há uma ferramenta chamada `preupgrade` que você pode usar em vez de apenas digitar `yum upgrade`. A vantagem de executar `preupgrade` primeiro é que esse comando verifica as dependências e baixa todos os pacotes necessários antes de comprometer seu sistema com o upgrade, de modo que você não acabe com um meio upgrade do sistema. Você também pode continuar usando seu sistema durante esse processo.

Se tiver problemas de dependência durante o `preupgrade`, você pode trabalhar com eles antes de a atualização realmente ocorrer. Remover pacotes que têm problemas de dependência é uma forma de lidar com o problema (`yum remove somepackage`). Uma vez que o upgrade está concluído, muitas vezes você pode adicionar o pacote de volta, usando uma versão dele que pode ser mais consistente com os novos repositórios que a que você está usando.

Usando cron para atualizações de software

O recurso `cron` oferece uma maneira de executar comandos em momentos e intervalos pré-

terminados. Você pode configurar o exato minuto, hora, dia ou mês que um comando é executado. Você pode configurar um comando para executar a cada cinco minutos, a cada três horas, ou em uma hora específica na tarde de sexta-feira, por exemplo.

Se quiser usar `crontab` para configurar atualizações de software noturnas, você pode fazer isso como usuário `root` executando o comando `crontab -e`. Isso abre um arquivo usando seu editor padrão (o comando `vi` por padrão), que você pode configurar como um arquivo `crontab`. Abaixo, veja um exemplo da aparência que pode ter o arquivo `crontab` que você cria:

Um arquivo `crontab` consiste em cinco campos, designando dia e hora, e um sexto campo, indicando a linha de comando a ser executada. Eu adicionei a linha de comentário para indicar os campos. Aqui, o comando de `yum -y update` é executado, com sua saída enviada para o usuário `root@localhost`. O comando é executado às 23:59. Os asteriscos (*) são obrigatórios como espaços reservados, instruindo `cron` a executar o comando a cada dia do mês, a cada mês e a cada dia da semana.

Quando você cria uma entrada no `crontab`, certifique-se de que quer direcionar a saída para um arquivo ou um comando que pode lidar com a saída. Se você não fizer isso, qualquer saída é enviada para o usuário que executou o comando `crontab -e` (neste caso, `root`).

Em um arquivo `crontab`, você pode ter uma série de números, uma lista de números ou ignorar os números. Por exemplo, 1, 5 ou 17 no primeiro campo fazem com que o comando seja executado 1, 5 e 17 minutos após a hora. Um `*/3` no segundo campo faz com que o comando seja executado a cada três horas (meia-noite, 3:00, 6:00 etc.). Um `1-3` no quarto campo instrui `cron` a executar o comando em janeiro, fevereiro e março. Dias da semana e meses podem ser introduzidos como números ou palavras.

Para mais informações sobre o formato de um arquivo `crontab`, digite `man 5 crontab`. Para ler sobre o comando `crontab`, digite `man 1 crontab`.

Corrigindo bancos de dados e cache RPM

Informações sobre todos os pacotes RPM em seu sistema são armazenadas em seu banco de dados RPM local. Embora isso aconteça com muito menos frequência do que nas versões anteriores do Fedora e RHEL, é possível que o banco de dados RPM torne-se corrompido. Isso impede você de instalar, remover ou listar pacotes RPM.

Se seus comandos `rpm` e `yum` estiverem travando ou falhando e retornando uma mensagem `rpmdb open fails`, você pode tentar reconstruir o banco de dados RPM. Para verificar se não é um problema em seu banco de dados RPM, você pode executar o comando de `yum check`. Eis um exemplo da saída do comando com um banco de dados corrompido:

```
# yum check
error: db4 error(11) from dbenv->open: Resource temporarily
```

```
unavailable
error: cannot open Packages index using db4 - Resource temporarily
unavailable (11)

error cannot open Packages database in /var/lib/rpm
CRITICAL:yum.main:
Error: rpmdb open fails
```

O banco de dados RPM e outras informações sobre os pacotes RPM instalados estão armazenados no diretório `/var/lib/rpm`. Você pode remover os arquivos de banco de dados que começam com `__db*` e reconstruí-los a partir dos metadados armazenados em outros arquivos no diretório.

Antes de começar, é uma boa ideia fazer o backup de `/var/lib/rpm`. Depois, você precisa remover os antigos arquivos `__db*` e reconstruí-los. Digite os seguintes comandos para fazer isso:

```
# cp -r /var/lib/rpm /tmp
# cd /var/lib/rpm
# rm __db*
# rpm --rebuilddb
```

Novos arquivos `__db*` devem aparecer depois de alguns segundos no diretório. Tente um comando `rpm` ou `yum` simples para garantir que os bancos de dados agora estejam em ordem.

Assim como RPM tem bancos de dados de pacotes instalados localmente, o recurso Yum armazena informações associadas com repositórios Yum no diretório `/var/cache/yum` local. Os dados em cache incluem metadados, cabeçalhos, pacotes e dados de plug-in do `yum`.

Se houver algum problema com os dados armazenados em cache pelo `yum`, você pode limpá-lo. Na próxima vez que você executar um comando `yum`, os dados necessários são baixados novamente. Eis algumas razões para limpar o cache do `yum`:

- **Os metadados estão obsoletos.** Na primeira vez que você se conecta a um repositório Yum (baixando um pacote ou consultando o repositório), os metadados são baixados para seu sistema. Os metadados consistem em informações sobre todos os pacotes disponíveis a partir do repositório.

À medida que pacotes são adicionados e removidos do repositório, os metadados têm de ser atualizados ou seu sistema estará trabalhando a partir de informações dos pacotes antigos. Por padrão, se você executar um comando `yum`, este procura novos metadados se os velhos tiverem mais de 90 minutos de idade (ou quantos minutos estiverem configurados na opção `metadata_expire=` do arquivo `/etc/yum.conf`).

Se suspeitar que os metadados estejam obsoletos, mas o tempo de expiração não foi alcançado, você pode executar `yum clean metadata` para remover todos os metadados, forçando novos metadados a serem carregados no próximo upload. Alternativamente, você pode executar `yum makecache` para atualizar os metadados de todos os repositórios.

- **Você está sem espaço em disco.** Normalmente, o `yum` pode armazenar em cache algumas centenas

de megabytes de dados nos diretórios `/var/cache/yum`. Mas, dependendo das configurações de seu arquivo `/etc/yum.conf` (como `keepcache=1`, que mantém todos os RPMs baixados, mesmo depois que eles foram instalados), os diretórios de cache podem conter vários gigabytes de dados.

Para limpar todos os pacotes, metadados, cabeçalhos e outros dados armazenados no diretório `/var/cache/yum`, digite o seguinte:

```
# yum clean all
```

Neste ponto, o sistema vai começar a baixar informações atualizadas de repositórios da próxima vez que um comando `yum` for executado.

A próxima seção aborda informações sobre solução de problemas de rede.

Solucionando problemas rede

Com cada vez mais informações, imagens, vídeos e outros conteúdos que usamos todo dia agora disponíveis fora dos nossos computadores locais, uma conexão de rede é necessária em quase todos os sistemas de computador. Então, se sua conexão de rede cair ou não puder alcançar os sistemas com os quais você quer se comunicar, é bom saber que há muitas ferramentas no Linux para ajudar a resolver o problema.

Para computadores clientes (laptops, desktops e dispositivos portáteis), a conexão de rede é necessária para alcançar outros sistemas de computador. Em um servidor, você quer que seus clientes sejam capazes de alcançar você. As próximas seções descrevem diferentes ferramentas para solução de problemas de conectividade de rede para sistemas cliente e servidor Linux.

Solucionando problemas de conexões de saída

Você abre seu navegador, mas não consegue entrar em nenhum site. Você suspeita de que não está conectado à rede. Talvez o problema seja a conversão de nomes, mas pode ser a conexão fora de sua rede local.

Para verificar se suas conexões de rede de saída estão funcionando, você pode usar muitos dos comandos descritos no Capítulo 14, “Administrando redes”. Você pode testar a conectividade por meio de um comando `ping` simples. Para ver se a conversão de nome para endereço está funcionando, use `host` e `dig`.

As próximas seções abordam problemas que podem ocorrer com a conectividade de rede para conexões de saída e quais ferramentas usar para descobrir os problemas.

Visualize as placas de rede

Para ver o estado de suas placas de rede, use o comando `ip`. O resultado a seguir mostra que a interface de loopback (`lo`) está ativa (de modo que você pode executar comandos de rede em seu sistema local), mas `eth0` (sua primeira placa de rede com fio) é baixo (`state DOWN`). Se a placa estivesse ativa, não

haveria uma linha `inet` mostrando o endereço IP da placa. Aqui, apenas a interface de loopback tem um endereço `inet` (127.0.0.1).

```
# ip addr show
```

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 state DOWN
    qlen 1000
    link/ether f0:de:f1:28:46:d9 brd ff:ff:ff:ff:ff:ff
```

Verifique as conexões físicas

Para uma conexão com fio, verifique se seu computador está conectado à porta do switch da rede. Se você tiver várias placas de rede, verifique se o cabo está conectado à porta correta. Se você sabe o nome de uma placa de rede (`eth0`, `p4p1` ou outra), para descobrir qual placa de rede está associada à interface, digite `ethtool -p eth0` a partir da linha de comando e olhe atrás de seu computador para ver qual placa de rede estão piscando (Ctrl+C para de piscar). Conecte o cabo na porta correta.

Se, em vez de mostrar uma placa de rede desativada, o comando `ip` simplesmente não mostrar nenhuma placa de rede, verifique se o hardware não está desabilitado. No caso de uma placa de rede com fio, ela pode não estar bem encaixada em seu slot ou pode ter sido desabilitada na BIOS.

Em uma conexão sem fio, você pode clicar no ícone do NetworkManager e não ver uma placa sem fio disponível. Mais uma vez, ela pode estar desabilitada na BIOS. Mas, em um laptop, verifique se não há um minúsculo interruptor que desativa a placa de rede. Já vi várias pessoas retalharem suas configurações de rede apenas para descobrir que esse minúsculo interruptor na parte da frente ou do lado de seus laptops foi mudado para a posição desligado.

Verifique as rotas

Se sua placa de rede estiver ativa, mas você ainda não puder alcançar o host desejado, tente verificar a rota para esse host. Comece verificando sua rota padrão. Então, tente alcançar o dispositivo de gateway da rede local para a rede seguinte. Por fim, tente “pingar” um sistema em algum lugar na internet:

```
# route
route
Kernel IP routing table
Destination Gateway Genmask Flags Metric Ref Use Iface
192.168.0.0 * 255.255.255.0 U 2 0 0 eth0
default 192.168.0.1 0.0.0.0 UG 0 0 0 eth0
```

A linha padrão mostra que o gateway padrão (UG) está no endereço 192.168.0.1 e que o endereço pode ser alcançado pela placa `eth0`. Como há apenas a placa `eth0` aqui e apenas uma rota para a rede 192.168.0.0 é mostrada, toda a comunicação não dirigida a um host na rede 192.168.0.0/24 é enviada através do gateway padrão (192.168.0.1). O gateway padrão é mais propriamente conhecido como um roteador.

Para se certificar de que você pode alcançar seu roteador, tente emitir um `ping` para ele. Por exemplo:

```
# ping -c 2 192.168.0.1
```

```
PING 192.168.0.1 (192.168.0.1) 56(84) bytes of data.
```

```
From 192.168.0.105 icmp_seq=1 Destination Host Unreachable
```

```
From 192.168.0.105 icmp_seq=2 Destination Host Unreachable
```

```
--- 192.168.0.1 ping statistics ---
```

```
2 packets transmitted, 0 received, +2 errors, 100% packet loss
```

A mensagem “Destination Host Unreachable” (“Host de Destino Inacessível”) informa que o roteador está desligado ou não fisicamente conectado a você (talvez o roteador não esteja conectado ao switch que você compartilha). Se o ping for bem-sucedido e você puder alcançar o roteador, o próximo passo é tentar um endereço fora de seu roteador.

Tente emitir um ping para um endereço IP amplamente acessível. Por exemplo, o endereço IP para o servidor DNS público do Google é 8.8.8.8. Tente pingar esse endereço (`ping -c2 8.8.8.8`). Se esse ping for bem-sucedido, sua rede provavelmente está muito bem e é mais provável que sua conversão de nomes para endereços não esteja funcionando corretamente.

Se puder alcançar um sistema remoto, mas a conexão é muito lenta, você pode usar o comando `traceroute` para seguir a rota até o host remoto. Por exemplo, este comando mostra cada *hop* (salto entre nós de uma rede) na rota para `http://www.google.com`:

```
# traceroute www.google.com
```

O resultado mostrará o tempo necessário para cada hop ao longo do caminho para o site do Google. Em vez de `traceroute`, você pode usar o comando `mtr` (`yum install mtr`) para ver a rota seguida até um host. Com `mtr`, a rota é consultada continuamente, o que permite ver o desempenho de cada etapa da viagem ao longo do tempo.

Verifique a conversão de hostname

Se você não pode alcançar hosts remotos pelo nome, mas pode alcançá-los pingando endereços IP, o sistema está tendo um problema com conversão de nomes. Sistemas conectados à internet fazem conversão de nome para endereço consultando um servidor DNS (*domain name system*) que pode fornecer-lhes endereços IP dos hosts solicitados.

O servidor DNS que seu sistema usa pode ser inserido manualmente ou recuperado automaticamente de um servidor DHCP quando você inicia suas placas de rede. De uma maneira ou de outra, os nomes e endereços IP de um ou mais servidores DNS acabarão em seu arquivo `/etc/resolv.conf`. Eis um exemplo desse arquivo:

```
search example.com
nameserver 192.168.0.254
nameserver 192.168.0.253
```

Quando você pede para conectar-se a um hostname no Fedora ou Red Hat Enterprise Linux, o arquivo `/etc/hosts` é procurado; em seguida, o servidor DNS da primeira entrada `nameserver` em `resolv.conf` é consultado; e, então, cada subsequente servidor DNS é consultado. Se um host que

você solicitar não for encontrado, todos os locais são verificados antes de você obter algum tipo de mensagem “Host não encontrado”. Eis algumas maneiras de depurar a conversão de nome para endereço:

- **Verifique se o servidor DNS pode ser alcançado.** Conhecendo os endereços de servidor DNS, você pode tentar pingar o endereço IP de cada servidor DNS para ver se ele está acessível. Por exemplo: `ping -c 2 192.168.0.254`. Se o endereço IP puder ser alcançado, então talvez você tivesse atribuído o endereço errado ao servidor DNS ou ele está atualmente fora do ar.
- **Verifique se o servidor DNS está funcionando.** Você tenta usar especificamente cada servidor DNS com o comando `host` ou `dig`. Por exemplo, qualquer um desses dois comandos pode ser usado para ver se o servidor DNS em `192.168.0.254` pode converter o hostname `www.google.com` em um endereço IP. Repita esse procedimento para cada endereço IP do servidor DNS até você descobrir quais funcionam:

```
# host www.google.com 192.168.0.254
# dig @192.168.0.254 www.google.com
```

- **Corrija seus servidores DNS.** Se você determinar que tem o endereço IP errado configurado para seus servidores DNS, modificá-los pode ser um pouco complicado. Pesquise em `/var/log/messages` o endereço IP dos seus servidores DNS. Se o NetworkManager for usado para iniciar sua rede e se conectar a um servidor DHCP, você deve ver as linhas `nameserver` com os endereços IP que estão sendo atribuídos. Se os endereços estiverem errados, você pode substituí-los.

Com NetworkManager habilitado, você não pode simplesmente adicionar entradas de servidor DNS ao arquivo `/etc/resolv.conf` porque o NetworkManager sobrescreverá esse arquivo com suas próprias entradas de servidor DNS. Em vez disso, adicione uma linha `PEERDNS=no` ao arquivo `ifcfg` para a placa de rede (por exemplo, `ifcfg-eth0` no diretório `/etc/sysconfig/network-scripts`). Então, configure `DNS1=192.168.0.254` (ou qualquer que seja o endereço IP do seu servidor DNS). O novo endereço será usado na próxima vez que você reiniciar sua rede.

Se estiver usando o serviço de rede, em vez de NetworkManager, você ainda pode usar `PEERDNS=no` para impedir que o servidor DHCP sobrescreva seus endereços de DNS. Mas, nesse caso, você pode editar o arquivo `resolv.conf` diretamente para configurar seus endereços de servidor DNS.

O procedimento aqui descrito para verificar sua conectividade de rede de saída aplica-se a qualquer tipo de sistema, seja ele um laptop, um desktop ou um servidor. Em geral, as conexões recebidas não são um problema com laptops ou desktops, porque a maioria das solicitações é simplesmente negada. Mas para os servidores, a próxima seção descreve maneiras de tornar seu servidor acessível se os clientes estiverem tendo problemas para alcançar os serviços que você fornece a partir desse servidor.

Solucionando problemas de conexões de entrada

Se você estiver solucionando problemas de placas de rede em um servidor, há considerações diferentes das de um sistema desktop. Como a maioria dos sistemas Linux é configurada como servidor, você deve

saber como solucionar os problemas encontrados por aqueles que estão tentando alcançar seus servidores Linux.

Vou começar com a ideia de ter um servidor web Apache (`httpd`) rodando em seu sistema Linux, mas com nenhum cliente web capaz de alcançá-lo. As próximas seções descrevem as coisas que você pode tentar para ver onde está o problema.

Verifique se realmente o cliente pode acessar seu sistema

Para ser um servidor público, o `hostname` deve ser convertido de modo que qualquer cliente na internet possa alcançá-lo. Isso significa associar seu sistema a um determinado endereço IP público e registrar esse endereço em um servidor DNS público. Você pode usar uma empresa de registro de domínio (como `http://www.networksolutions.com`) para fazer isso.

Quando os clientes não podem alcançar seu site pelo nome a partir dos navegadores deles, se o cliente for um sistema Linux, você pode usar `ping`, `host`, `traceroute` e outros comandos descritos na seção anterior para rastrear o problema de conectividade. Sistemas Windows têm sua própria versão de `ping` que você pode usar a partir desses sistemas.

Se a conversão de nome para endereço estiver funcionando para alcançar seu sistema e você for capaz de emitir `ping` para um servidor a partir de fora da rede interna, a próxima coisa a tentar é a disponibilidade do serviço.

Verifique se o serviço está disponível para o cliente

A partir de um cliente Linux, você pode verificar se o serviço que você está procurando (nesse caso `httpd`) está disponível a partir do servidor. Uma maneira de fazer isso é usando o comando `nmap`.

O comando `nmap` é a ferramenta favorita dos administradores de sistema para verificar vários tipos de informações em redes. Mas ele também é a ferramenta favoritas dos crackers, porque permite varrer servidores, procurando potenciais vulnerabilidades. Portanto, é bom usar `nmap` a fim de varrer seus próprios sistemas para ver se há problemas. Mas saiba que usar `nmap` em outro sistema é como verificar as portas e janelas na casa de alguém para ver se você pode entrar. Você vai parecer um intruso.

Verificar seu próprio sistema para ver quais portas em seu servidor estão abertas ao mundo exterior (essencialmente, verificar quais serviços estão em execução) é perfeitamente legítimo e fácil de fazer. Depois que `nmap` está instalado (`yum install nmap`), use o `hostname` ou o endereço IP de seu sistema para fazer `nmap` varrer seu sistema a fim de ver o que está rodando em portas comuns:

```
# nmap 192.168.0.119
Starting Nmap 5.21 ( http://nmap.org ) at 2012-06-16 08:27 EDT
Nmap scan report for spike (192.168.0.119)
Host is up (0.0037s latency).
Not shown: 995 filtered ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
443/tcp   open  https
631/tcp   open  ipp
MAC Address: 00:1B:21:0A:E8:5E (Intel Corporate)
Nmap done: 1 IP address (1 host up) scanned in 4.77 seconds
```

A saída anterior mostra que as portas TCP estão abertas para os serviços web regulares (`http`) e

seguros (<https>). Quando você vir que o estado está `open` (aberto), isso indica que há um serviço ouvindo na porta também. Se alcançar esse ponto, isso significa que sua conexão de rede está bem e que você deve dirigir seus esforços de solução de problemas para a forma como o serviço em si está configurado (por exemplo, você pode examinar o arquivo `/etc/httpd/conf/httpd.conf` para ver se há `hosts` específicos com acesso permitido ou negado).

Se as portas TCP 80 e/ou 443 não forem mostradas, isso significa que estão sendo filtradas. Você precisa verificar se seu firewall está *bloqueando* (não aceitando pacotes para) essas portas. Se a porta não é filtrada, mas o estado é fechado, isso significa que o serviço `httpd` não está funcionando ou não está ouvindo nessas portas. O próximo passo é entrar no servidor e verificar essas questões.

Verifique o firewall no servidor

A partir de seu servidor, você pode usar o comando `iptables` para listar as regras da tabela de filtro que estão em vigor. Eis um exemplo:

```
# iptables -vnL
Chain INPUT (policy ACCEPT 0 packets, 0 bytes)
pkts bytes target prot opt in out source destination
...
 0      0 ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 state NEW tcp dpt:80
 0      0 ACCEPT tcp -- * * 0.0.0.0/0 0.0.0.0/0 state NEW tcp dpt:443
...
```

Deve haver regras de firewall, como as duas mostradas no código anterior, no meio das suas outras regras. Se não houver, adicione as regras ao arquivo `/etc/sysconfig/iptables`. Eis alguns exemplos de como essas regras podem ser:

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
-A INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT
```

Com as regras adicionadas ao arquivo, limpe todas as regras de firewall (`systemctl stop iptables.service` ou `service iptables stop`) e, então, inicie-as de novo (`systemctl start iptables.service` ou `service iptables start`).

Se o firewall ainda estiver bloqueando o acesso do cliente às portas do servidor web, eis algumas coisas para verificar em seu firewall:

- **Verifique a ordem das regras.** Olhe para as regras em `/etc/sysconfig/iptables` e veja se há regras `DROP` ou `REJECT` que apareçam antes das regras de abertura das portas 80 e/ou 443. Mover as regras para abrir as portas antes de quaisquer linhas `DROP` ou `REJECT` finais pode resolver o problema.
- **Olhe para os hosts negados.** Verifique se há regras que descartam ou rejeitam pacotes a partir de determinados hosts ou redes. Procure regras que incluam `-s` ou `--source`, seguidos por um endereço IP ou intervalo de endereços e, depois, `-j DROP` ou `ACCEPT`. Modifique ou adicione uma regra antes de suas regras para criar uma exceção para o host ao qual você quer permitir acesso ao seu serviço.

Se a porta estiver aberta, mas o serviço em si estiver fechado, verifique se o serviço está realmente rodando e ouvindo as placas de rede apropriadas.

Verifique o serviço no servidor

Se parecer não haver nada bloqueando o acesso do cliente ao servidor através das portas reais que fornecem o serviço que você quer compartilhar, é hora de verificar o próprio serviço. Assumindo que o serviço está sendo executado (digite **service httpd status** para verificar), a próxima coisa a verificar é se ele está ouvindo nas portas e placas de rede adequadas.

O comando `netstat` é uma ferramenta de uso geral, ótimo para verificar os serviços de rede. O seguinte comando lista os nomes e IDs de processo (`p`) para todos os processos que estão ouvindo (*listening*, `l`) serviços TCP (`t`) e UDP (`u`), juntamente com o número da porta (`n`) em que eles estão ouvindo. A linha de comando a seguir filtra todas as linhas, exceto aquelas associadas com o processo `httpd`:

```
# netstat -tupln | grep httpd
tcp                0          0 :::80           :::*           LISTEN
                   2567/httpd
tcp                0          0 :::443          :::*           LISTEN
                   2567/httpd
```

O exemplo anterior mostra que o processo `httpd` está ouvindo na porta 80 e 443 de todas as interfaces de rede. É possível que o processo `httpd` esteja ouvindo em interfaces de rede selecionadas. Por exemplo, se o processo `httpd` estiver ouvindo apenas na interface local (127.0.0.1) para solicitações HTTP (porta 80), a entrada ficaria da seguinte forma:

```
tcp                0          0 127.0.0.1:80   :::*           LISTEN 2567/httpd
```

Para `httpd`, bem como para outros serviços de rede que ouvirem solicitações de placas de rede, é possível editar o arquivo de configuração do serviço principal (nesse caso, `/etc/httpd/conf/httpd.conf`) para dizer-lhe para ouvir na porta 80 todos os endereços (`Listen 80`) ou um endereço específico (`Listen 192.168.0.100:80`).

Solucionando problemas de memória

Solucionar problemas de desempenho em seu computador é uma das mais importantes tarefas, embora muitas vezes arduas, que você precisa completar. Talvez você tenha um sistema que estava funcionando bem, mas começa a desacelerar até um ponto em que é praticamente inutilizável. Talvez os aplicativos simplesmente comecem a falhar sem motivo aparente. Encontrar e corrigir o problema pode demandar um trabalho de detetive.

O Linux vem com muitas ferramentas para observar atividades em seu sistema e descobrir o que está acontecendo. Usando uma variedade de utilitários Linux, você pode fazer coisas como descobrir quais processos estão consumindo grandes quantidades de memória ou impondo altas demandas sobre seus processadores, discos ou largura de banda de rede. As soluções podem incluir:

- **Adição de capacidade** — Seu computador pode estar tentando fazer o que você pediu a ele, mas falhas podem ocorrer porque você não tem memória suficiente, poder de processamento, espaço em disco ou capacidade de rede para obter um desempenho razoável. Mesmo o simples fato de se aproximar dos limites do esgotamento de um recurso pode causar problemas de desempenho. Melhorar a capacidade de hardware do computador é muitas vezes a maneira mais fácil de resolver problemas de desempenho.
- **Ajustando o sistema** — O Linux vem com configurações padrão que definem como ele salva internamente os dados, como ele move os dados pelo sistema e como ele protege os dados. Parâmetros ajustáveis do sistema podem ser alterados se as configurações padrão não funcionarem bem para os tipos de aplicativos que você tem em seu sistema.
- **Descobrimo aplicativos ou usuários problemáticos** — Às vezes, um sistema executa mal porque um usuário ou um aplicativo está fazendo algo errado. Aplicações mal configuradas ou quebradas podem travar ou devorar todos os recursos que podem obter. Um usuário inexperiente pode equivocadamente iniciar várias instâncias de um programa que drenam os recursos do sistema. Como um administrador de sistema, você quer saber como encontrar e corrigir esses problemas.

Para solucionar problemas de desempenho no Linux, você pode usar algumas das ferramentas básicas para observar e manipular processos em execução em seu sistema. Consulte o Capítulo 6, “Gerenciando processos em execução”, se você precisa de mais detalhes sobre comandos como `ps`, `top`, `kill` e `killall`. Nesta seção, vamos examinar comandos como `memstat`, para nos aprofundarmos um pouco mais no que os processos estão fazendo e onde as coisas estão indo mal.

A área mais complexa de solução de problemas no Linux é o gerenciamento de memória virtual. As próximas seções descrevem como visualizar e gerenciar a memória virtual.

Descobrimo problemas de memória

Computadores armazenam dados de forma permanente (discos rígidos) e temporária (*Random Access Memory*, ou *RAM*, e *espaço de troca*). Pense em você como uma CPU, em uma mesa, tentando fazer seu trabalho. Você poderia colocar os dados que quer manter permanentemente em um armário do outro lado da sala (que é como o armazenamento em disco rígido) e colocaria as informações em que você está trabalhando atualmente em sua mesa (que é como a memória RAM em um computador).

O espaço de troca é uma forma de estender a RAM. É realmente apenas um lugar para colocar os dados temporários que não cabem na memória RAM, mas que a CPU precisará em algum momento mais tarde. Embora o espaço de troca esteja no disco rígido, não é um sistema de arquivos Linux regular no qual os dados são armazenados de forma permanente. Pense no espaço de troca como uma gaveta do armário em que a informação é mantida em uma caixa misturada, onde ela pode ser classificada dentro de armários permanentes ou trazida de volta para ser usada sobre a mesa.

Comparado com armazenamento em disco, a memória de acesso aleatório tem os seguintes atributos:

- **Mais perto do processador** — Assim como a mesa está perto de você enquanto você trabalha, a memória está fisicamente perto da CPU na placa-mãe do computador. Assim, quaisquer dados que a CPU necessite podem ser imediatamente obtidos se estiverem na memória RAM.

- **Mais rápido** — Sua proximidade da CPU e a maneira como a RAM é acessada (estado sólido contra óptico) tornam muito mais rápido para a CPU obter informações da memória RAM do que de um disco rígido. É mais rápido olhar para um pedaço de papel em sua mesa (um espaço pequeno e íntimo) do que ir até a uma fila de armários de arquivo e começar a procurar o que quer.
- **Menos capacidade** — Um novo computador pode ter um disco rígido de 360GB, mas 2GB ou 4GB de memória RAM. Embora colocar na RAM cada arquivo e cada fragmento de dados que o processador precisasse pudesse fazer o computador funcionar mais rápido, na maioria dos casos simplesmente não haveria espaço. Além disso, tanto os slots de memória física do computador como o sistema de computador em si (computadores de 64 bits podem endereçar mais memória RAM do que computadores de 32 bits) podem limitar a quantidade de memória disponível que um computador é capaz de ter.
- **Mais cara** — Embora a RAM seja tremendamente mais acessível do que uma ou duas décadas atrás, ainda é muito mais cara (por GB) do que os discos rígidos.
- **Temporária** — A RAM armazena dados e metadados que a CPU está usando agora para o trabalho que está fazendo (além de algum conteúdo que o kernel do Linux mantém por perto porque suspeita que um processo precise dele antes do tempo). Quando você desliga o computador, porém, tudo na RAM é apagado. Quando a CPU termina de usar os dados, eles são descartados se não forem mais necessários, deixados na RAM para uso posterior possível ou marcados para serem gravados em disco para armazenamento permanente se precisarem ser salvos.

É importante entender a diferença entre armazenamento temporário (RAM) e permanente (disco rígido), mas isso não conta toda a história. Se a demanda por memória exceder a oferta de RAM, o kernel pode temporariamente mover dados de memória RAM para uma área chamada espaço de troca ou *swap*.

Se voltarmos à analogia da mesa de trabalho, isso seria como dizer: “Não há espaço na minha mesa, mas tenho de colocar mais documentos em cima dela para continuar a trabalhar nos projetos. Em vez de armazenar papéis que vou precisar em breve em um armário de arquivo permanente, vou ter um armário de arquivo especial (como uma gaveta) para manter os papéis com que ainda estou trabalhando, mas que não estou pronto para armazenar de forma permanente ou jogar fora.”

Consulte o Capítulo 12, “Gerenciando discos e sistemas de arquivos”, para mais informações sobre arquivos e partições de troca e como criá-los. Por enquanto, porém, há algumas coisas que você deve saber sobre esses tipos de área de troca e quando eles são usados:

- Quando os dados são movidos da RAM para uma área de troca, você tem um impacto no desempenho. Lembre-se, gravar em disco é muito mais lento do que gravar em RAM.
- Quando os dados são movidos da área de troca de volta para a RAM porque são necessários novamente, você tem outra queda no desempenho.
- Quando o Linux fica sem espaço na memória RAM, a troca é como ser ferido em uma batalha. Não é algo que você deseja, mas é melhor do que ser morto. Em outras palavras, todos os seus processos permanecem ativos e não perdem quaisquer dados, mas o desempenho do sistema pode cair bastante.

- Se tanto a RAM como o espaço de troca ficarem cheios e os dados não puderem ser descartados ou gravados em disco, o sistema pode alcançar uma condição de falta de memória (*out-of-memory*, OOM). Quando isso acontece, o eliminador de OOM do kernel entra em ação e começa a eliminar processos, um a um, para recuperar o máximo de memória possível para o kernel voltar a funcionar corretamente.

A regra de ouro sempre foi a de que a troca é ruim e deve ser evitada. Mas há alguns que argumentam que, em certos casos, uma troca mais agressiva pode realmente melhorar o desempenho.

Pense no caso em que você abre um documento em um editor de texto e então o minimiza em seu desktop por vários dias enquanto trabalha em diferentes tarefas. Se os dados desse documento forem movidos para o disco, mais memória RAM estaria disponível para aplicativos mais ativos que poderiam empregar melhor esse espaço. A queda no desempenho viria na próxima vez em que você precisasse acessar os dados do documento editado e eles fossem movidos do disco para a RAM. As configurações que se relacionam com a agressividade com que um sistema fará a troca de dados são referidas como *swappiness*.

Tanto quanto possível, o Linux quer disponibilizar tudo o que um aplicativo aberto necessita imediatamente. Assim, usando a analogia da mesa de trabalho, se eu estou trabalhando em nove projetos ativos e não há espaço na mesa para armazenar as informações que eu preciso para todos os nove projetos, por que não deixá-las todas dentro do alcance sobre a mesa? Seguindo essa mesma linha de raciocínio, o kernel às vezes mantém bibliotecas e outros conteúdos na RAM que ele pensa que você pode, eventualmente, precisar, mesmo que um processo não esteja procurando por ele imediatamente.

O fato de o kernel estar inclinado a armazenar informações na RAM que ele espera que possam ser necessárias em breve (ainda que não o sejam agora) pode fazer com que um administrador do sistema inexperiente pense que o sistema está quase sem RAM e que os processos estão prestes a começar a falhar. É por isso que é importante conhecer os diferentes tipos de informação que são mantidos na memória — para que você possa determinar quando situações reais de falta de memória podem ocorrer. O problema não é apenas esgotar a RAM, é esgotar a RAM quando restam apenas dados não trocáveis.

Mantenha essa visão geral de *memória virtual* (RAM e espaço de troca) em mente, à medida que a próxima seção descreve maneiras de solucionar problemas relacionados com a memória virtual.

Verificando problemas de memória

Digamos que você está conectado a um desktop Linux, com muitos aplicativos em execução e tudo começa a ficar mais lento. Para descobrir se os problemas de desempenho ocorreram porque você esgotou a memória, você pode tentar comandos como `top` e `ps` para começar a avaliar o consumo de memória no sistema.

Para executar o comando `top` a fim de observar o consumo de memória, digite **top** e digite um **M** maiúsculo. Eis um exemplo:

```

# top
top - 22:48:24 up 3:59, 2 users, load average: 1.51, 1.37, 1.15
Tasks: 281 total, 2 running, 279 sleeping, 0 stopped, 0 zombie
Cpu(s): 16.6%us, 3.0%sy, 0.0%ni, 80.3%id, 0.0%wa, 0.0%hi, 0.2%si, 0.0%st
Mem: 3716196k total, 2684924k used, 1031272k free, 146172k buffers
Swap: 4194296k total, 0k used, 4194296k free, 784176k cached
  PID USER      PR  NI  VIRT  RES  SHR  S %CPU %MEM    TIME+  COMMAND
 6679 cnegus    20   0 1665m 937m 32m  S  7.0 25.8   1:07.95  firefox
 6794 cnegus    20   0  743m 181m 30m  R 64.8  5.0   1:22.82  npviewer.bin
 3327 cnegus    20   0 1145m 116m 66m  S  0.0  3.2   0:39.25  soffice.bin
 6939 cnegus    20   0  145m  71m 23m  S  0.0  2.0   0:00.97  acroread
 2440 root      20   0  183m  37m 26m  S  1.3  1.0   1:04.81  Xorg
 2795 cnegus    20   0 1056m 22m 14m  S  0.0  0.6   0:01.55  nautilus

```

Há duas linhas (Mem e Swap) e quatro colunas de informação (VIRT, RES, SHR e %MEM) relativas à memória na saída de `top`. Nesse exemplo, você pode ver que a RAM não se esgota na linha de Mem (apenas 268492k de 3716196k são usados) e que nada está sendo movido para disco na linha de Swap (0k usado).

Mas, somando apenas as primeiras seis linhas da saída na coluna VIRT, você veria que 4937MB de memória foram alocados para esses aplicativos, o que excede os 3629MB de RAM total (3716196k) que estão disponíveis. Isso porque a coluna VIRT mostra apenas a quantidade de memória que foi prometida ao aplicativo. A linha RES mostra a quantidade de memória não trocável que está realmente sendo usada, o que totaliza apenas 1364MB.

Observe que, quando você pede para ordenar por uso de memória digitando um **M** maiúsculo, `top` sabe ordenar na coluna RES. A coluna SHR mostra a memória que poderia ser compartilhada por outros aplicativos (como bibliotecas) e %MEM mostra a porcentagem do total de memória consumida por cada aplicativo.

Se você acha que o sistema está para atingir um estado de falta de memória, eis algumas coisas para procurar:

- O espaço livre mostrado na linha Mem estaria em ou perto de zero.
- O espaço usado mostrado na linha de Swap seria diferente de zero e continuaria a crescer. Isso deve ser acompanhado por uma queda no desempenho do sistema.
- Como a tela de `top` é atualizada a cada poucos segundos, se não houver um processo com um vazamento de memória (continuamente pedindo e usando mais memória, mas não devolvendo nada dela), a quantidade de memória VIRT cresce, mas, sobretudo, a memória RES continuará a crescer para esse processo.
- Se o espaço de Swap realmente acabar, o kernel vai começar a eliminar processos para lidar com essa condição de falta de memória.

Lidando com problemas de memória

A curto prazo, há várias coisas que você pode fazer para lidar com essa condição de falta de memória:

- Elimine um processo. Se o problema de memória for devido a um processo errante, você pode simplesmente eliminá-lo. Assumindo que você está conectado como root ou como o usuário que possui o processo errante, digite **k** na janela superior e, então, digite o PID do processo que você quer eliminar e escolha 15 ou 9 como o sinal a enviar.

- **Descarte caches de página.** Se quiser apenas liberar alguma memória imediatamente, deixando para resolver o problema mais tarde, você poderia instruir o sistema a descartar caches de páginas inativas. Quando você faz isso, algumas páginas de memória são gravadas em disco e outras são simplesmente descartadas (porque estão armazenados permanentemente e podem ser obtidas a partir do disco de novo quando forem necessárias).

Essa ação é o equivalente a limpar sua mesa e colocar todas as informações, exceto as mais críticas, no lixo ou em um armário de arquivo. Você pode precisar recuperar informações de um armário de arquivo novamente em breve, mas quase certamente não vai precisar de tudo isso imediatamente. Mantenha `top` executado em uma janela Terminal para ver a linha `Mem` mudar quando você digitar o seguinte (como `root`) em outra janela Terminal:

```
# echo 3 > /proc/sys/vm/drop_caches
```

- **Elimine um processo sem memória.** Há momentos em que o esgotamento de memória torna o sistema tão inutilizável que você pode não ser capaz de obter uma resposta de um shell ou GUI. Nesses casos, você pode ser capaz de usar as teclas `Alt+SysRq` para eliminar um processo sem memória. A razão por que você pode usar teclas `Alt+SysRq` em um sistema que não responde de outra forma é que o kernel processará as solicitações das teclas `Alt+SysRq` antes de outras solicitações.

Para ativar as teclas `Alt+SysRq`, o sistema já deve ter configurado `/proc/sys/kernel/sysrq` como 1. Uma maneira fácil de fazer isso é adicionar `kernel.sysrq = 1` no arquivo `/etc/sysctl.conf`. Além disso, você deve pressionar as teclas `Alt+SysRq` a partir de uma interface baseada em texto (como o console virtual que você vê quando pressiona `Ctrl+Alt+F2`).

Com `kernel.sysrq` configurado como 1, você pode eliminar o processo em seu sistema com a maior pontuação OOM pressionando `Alt+SysRq+f` a partir de uma interface baseada em texto. Uma listagem de todos os processos em execução no sistema aparece na tela, com o nome do processo que foi eliminado listado no final. Você pode repetir as teclas até eliminar processos suficientes para ser capaz de acessar o sistema normalmente a partir do shell de novo.

ta

Muitas outras teclas `Alt+SysRq` que você pode usar para lidar com um sistema que não responde. Por exemplo, `Alt+SysRq+e` termina todos os processos, exceto `init`. `Alt+SysRq+t` exibe uma lista de todas as funções e informações sobre essas tarefas no console. Para reiniciar o sistema, pressione `Alt+SysRq+b`. Veja o arquivo `sysrq.txt` no diretório `/usr/share/doc/kernel-doc*/Documentation` para obter mais informações sobre as teclas `Alt+SysRq`.

Solucionando problemas no modo de recuperação

Se seu sistema Linux torna-se não inicializável, sua melhor opção para corrigir o problema é provavelmente entrar no *modo de recuperação*. Para entrar no modo de recuperação, você ignora o

sistema Linux instalado em seu disco rígido e inicia uma mídia de recuperação (como um pen drive USB inicializável ou CD de inicialização). Depois que a mídia de recuperação inicializa, ela tenta montar algum sistema de arquivos que pode encontrar a partir de seu sistema Linux para que você possa reparar os problemas.

Para muitas distribuições Linux, o CD ou DVD de instalação pode servir como mídia de inicialização para entrar no modo de recuperação. Eis um exemplo de como usar um DVD de instalação do Fedora para entrar no modo de recuperação a fim de corrigir um sistema Linux corrompido:

1. Obtenha a imagem do CD ou DVD de instalação que você quer usar e grave-a na mídia apropriada (CD ou DVD). Veja o Apêndice A, “Mídia”, para obter informações sobre gravação de CDs e DVDs. (Para meu exemplo, usei um DVD de instalação do Fedora 16.)
2. Insira o CD ou DVD na unidade no computador que tem o sistema corrompido Linux instalado e reinicie.
3. Quando vir a tela da BIOS, pressione a tecla de função indicada nessa tela para selecionar o dispositivo de inicialização (possivelmente a tecla de função F12 ou F2).
4. Escolha a unidade (CD ou DVD) a partir da lista de dispositivos de inicialização e pressione Enter.
5. Quando o menu de inicialização do Fedora aparecer, use as teclas de seta para destacar a palavra *Troubleshooting* (*Solucionando problemas*) e pressione Enter. Em outras mídias de inicialização do Linux, a seleção poderia dizer *Rescue Mode* (*Modo de recuperação*) ou algo parecido. Na próxima tela que aparece, selecione Rescue a Fedora System e pressione Enter.
6. Depois de alguns instantes, o sistema Linux na mídia de recuperação inicializa. Quando solicitado, selecione seu idioma e teclado. Você será perguntado se quer iniciar as placas de rede no sistema.
7. Se achar que precisa obter algo de outro sistema em sua rede (como pacotes RPM ou ferramentas de depuração), selecione Yes e tente configurar suas placas de rede. Depois, você é perguntado se quer tentar montar os sistemas de arquivos de seu sistema Linux instalado sob `/mnt/sysimage`.
8. Selecione Continue para ter seu sistema de arquivos montado (se possível), sob o diretório `/mnt/sysimage`. Se isso for bem-sucedido, uma mensagem de recuperação aparece dizendo que seu sistema de arquivos foi montado sob `/mnt/sysimage`.
9. Selecione OK para continuar. Você deverá ver uma janela de comandos para o usuário root (#). Você está pronto para começar o processo de solução de problemas no modo de recuperação.

Uma vez que você está no modo de recuperação, a parte de seu sistema de arquivos que não está corrompida será montada sob o diretório `/mnt/sysimage`. Mude para esse diretório (`cd /mnt/sysimage`) e digite **ls** para verificar se os arquivos e diretórios do disco rígido estão lá.

Agora, a raiz do sistema de arquivos (/) inicia a partir do sistema de arquivos que vem na mídia de recuperação. Para solucionar problemas do seu sistema Linux instalado, porém, você pode digitar o seguinte comando:

```
# chroot /mnt/sysimage
```

Agora, o diretório `/mnt/sysimage` torna-se a raiz de seu sistema de arquivos (`/`) e se parece com o sistema de arquivos instalado em seu disco rígido. Eis algumas coisas que você pode fazer para reparar seu sistema enquanto está no modo de recuperação:

- **Corrija `/etc/fstab`.** Se seu sistema de arquivos não puder ser montado devido a um erro em seu arquivo `/etc/fstab`, você pode tentar corrigir todas as entradas que possam ter problemas (como nomes de dispositivos errados ou um diretório de ponto de montagem que não existe). Digite `mount -a` para se certificar de que todos os sistemas de arquivo podem ser montados.
- **Reinstale os componentes que faltam.** Pode ser que os sistemas de arquivos estejam bons, mas o sistema não conseguiu inicializar porque algum comando ou arquivo de configuração essencial está faltando. Você pode ser capaz de resolver o problema reinstalando o pacote com os componentes que faltam. Por exemplo, se alguém tivesse apagado `/bin/mount` por engano, o sistema não teria nenhum comando para montar sistemas de arquivos. A reinstalação do pacote `util-linux` iria repor o comando `mount` ausente.
- **Verifique os sistemas de arquivos.** Se seus problemas derivam de sistemas de arquivos de inicialização corrompidos, você pode tentar executar o comando `fsck` (filesystem check) para ver se há qualquer tipo de corrupção na partição do disco. Se houver, `fsck` tentará corrigir os problemas encontrados.

Quando você terminar de corrigir seu sistema, digite `exit` para sair do ambiente `chroot` e retornar ao layout do sistema de arquivos que a mídia de recuperação vê. Quando terminar tudo, digite `reboot` para reiniciar o sistema. Certifique-se de sair da mídia antes de o sistema reiniciar.

Resumo

A solução de problemas no Linux pode começar a partir do momento em que você liga o computador. Problemas podem ocorrer com a BIOS do computador, o carregador de inicialização ou em outras partes do processo de inicialização, e você pode corrigi-los interceptando-os em diferentes estágios do processo de inicialização.

Depois que o sistema foi iniciado, você pode solucionar problemas com pacotes de software, placas de rede ou falta de memória. O Linux vem com muitas ferramentas para encontrar e corrigir qualquer parte do sistema Linux que possa ser danificada e precise ser corrigida.

O próximo capítulo aborda o tema da segurança Linux. Usando as ferramentas descritas neste capítulo, você pode fornecer acesso àqueles serviços que você e seus usuários necessitam, enquanto bloqueia o acesso a recursos do sistema que você quer proteger.

Exercícios

Os exercícios nesta seção permitem experimentar técnicas úteis de solução de problemas no Linux. Como algumas das técnicas descritas aqui podem danificar seu sistema, recomendo que você não use um sistema de produção que não puder correr o risco de ser danificado. Consulte o Apêndice B para obter as soluções sugeridas.

Esses exercícios se relacionam com tópicos de solução de problemas em Linux. Para fazer esses exercícios, você precisa ser capaz de reiniciar seu computador e interromper qualquer trabalho que possa estar sendo feito.

1. Inicialize o computador e, assim que você vir a tela da BIOS, entre no modo Setup conforme as instruções na tela da BIOS.
2. Na tela BIOS Setup, determine se seu computador é de 32 bits ou 64 bits, se inclui suporte a virtualização e se sua placa de rede é capaz de inicialização PXE (Preboot eXecution Environment).
3. Reinicie o computador e, logo após a tela da BIOS desaparecer, quando você vir a contagem regressiva para a inicialização do sistema Linux, pressione qualquer tecla a fim de ir para o carregador de inicialização GRUB.
4. A partir do carregador de inicialização GRUB, adicione uma opção para inicializar com um nível de execução 1, de modo que você possa fazer manutenção do sistema.
5. Reinicie o computador com o Red Hat Enterprise Linux instalado e, a partir do carregador de inicialização GRUB, adicione uma opção que faz com que os serviços do sistema solicitem para você confirmar a inicialização de cada serviço.
6. Depois que o sistema inicializar, procure mensagens que foram produzidas no buffer do kernel mostrando a atividade do kernel durante a inicialização.
7. No Fedora ou no RHEL, execute um `yum update` e exclua qualquer pacote de kernel que esteja disponível.
8. Verifique os processos que estão ouvindo conexões de entrada em seu sistema.
9. Verifique quais portas estão abertas em sua placa de rede externa.
10. Execute o comando `top` em uma janela Terminal. Abra uma segunda janela Terminal, limpe o cache de página e observe na tela de `top` se mais memória RES está agora disponível.

Parte V

Aprendendo técnicas de segurança do Linux

NESTA PARTE

Capítulo 22

Entendendo a segurança básica do Linux

Capítulo 23

Entendendo a segurança avançada do Linux

Capítulo 24

Aprimorando a segurança do Linux com o SELinux

Capítulo 25

Protegendo o Linux na rede

Entendendo a segurança básica do Linux

NESTE CAPÍTULO

Entendendo o Ciclo de Vida do Processo de Segurança

Planejando a segurança

Implementando a segurança

Monitorando a segurança

Auditando e revisando a segurança

Entender a segurança é uma parte crucial da administração do sistema Linux. Um nome de usuário e uma senha simples não são mais suficientes para proteger seu servidor. O número e a variedade de ataques a computadores aumentam todos os dias e a necessidade de melhorar a segurança do computador continua a crescer com eles.

Alguns dos problemas de segurança que você pode enfrentar como um administrador incluem ataques de negação de serviço (DoS), rootkits, worms, vírus, bombas lógicas, ataques *man-in-the-middle*, cavalos de Troia etc. Os ataques não vêm apenas de fora da organização. Eles também podem vir de dentro dela. Proteger os ativos informacionais valiosos da sua organização pode ser uma árdua tarefa.

Seu primeiro passo é conhecer os procedimentos e princípios básicos de segurança. Com essas informações, você pode começar o processo de bloquear e proteger seus servidores Linux. Além disso, você pode aprender a se manter informado sobre as novas ameaças diárias e as novas formas de continuar a proteger os ativos informacionais valiosos da sua organização.

Introdução ao Ciclo de Vida do Processo de Segurança

Como o desenvolvimento de softwares, a proteção de um sistema de computador tem um ciclo de vida de processo. Esse Ciclo de Vida do Processo de Segurança tem quatro fases principais, como mostrado na Figura 22.1. Cada fase tem atividades importantes que ajudam a criar um sistema fortalecido contra ataques de segurança.

quatro fases principais do Ciclo de Vida do Processo de Segurança.

Observe que não há nenhum ponto de partida identificável no Ciclo de Vida do Processo de Segurança. Sob circunstâncias ideais, você começaria o processo na fase de planejamento, então passaria para a implementação, monitoramento, auditoria/revisão e então retornaria à fase de planejamento. Mas na maioria dos sistemas, o ponto de partida estaria em algum lugar no meio. Embora o ciclo de vida real possa avançar de forma ordenada, ainda é importante entender o ciclo de vida e suas fases. A segurança de seu sistema irá melhorar se você aprender o ciclo de vida e entender quais atividades precisam ocorrer em cada fase.

O Ciclo de Vida do Processo de Segurança tem vários termos e acrônimos próprios. Alguns desses termos e acrônimos entram em conflito com a terminologia consagrada da informática. Por exemplo, ao trabalhar com redes de computadores, um endereço MAC é um endereço Media Access Control (Controle de Acesso de Mídia). Na segurança de computadores, um MAC é uma forma de controle de acesso chamado Mandatory Access Control (Controle de Acesso Mandatório). Você pode notar a confusão que logo pode ocorrer.

Eis algumas definições básicas de segurança para ajudá-lo no mundo da segurança e neste capítulo:

- **Sujeito** — Um usuário ou um processo realizando um trabalho em nome de um usuário.
- **Objeto** — Um recurso particular em um servidor de computador, como um banco de dados ou um dispositivo de hardware.
- **Autenticação** — Determinar se um sujeito é quem ele diz que é. Isso às vezes é chamado de “identificação e autenticação”.
- **Controle de acesso** — Controle da capacidade de um sujeito de usar um determinado objeto.
- **Lista de controle de acesso (ACL)** — Uma lista de sujeitos que podem acessar um determinado objeto.

Em um sistema de computador, sujeitos acessam objetos. Mas antes de poderem fazer isso, eles devem ser autenticados. Os sujeitos então só recebem níveis de acesso a objetos por meio dos controles de acesso pré-estabelecidos. Tanto os procedimentos de autenticação como controle de acesso para sujeitos e objetos devem ser determinados durante a fase de planejamento do Ciclo de Vida do Processo de Segurança.

Examinando a fase de planejamento

Embora cada fase do Ciclo de Vida do Processo de Segurança tenha atividades importantes, a fase de planejamento pode ser a mais crítica. Na fase de planejamento, uma organização determina itens como:

- Quais sujeitos precisam ter acesso a quais objetos
- Quais objetos precisam de proteção contra quais sujeitos
- A segurança de cada objeto ou nível de classificação de confidencialidade
- Nível autorização de segurança ou confidencialidade de cada indivíduo
- Políticas e procedimentos para a concessão de acesso a objetos para os sujeitos
- Políticas e procedimentos para revogar o acesso a objetos para os sujeitos
- Quem deve aprovar e permitir o acesso a objetos

Escolhendo um modelo de controle de acesso

As respostas às perguntas listadas acima formam a base dos controles de acesso de segurança. Os controles de acesso podem ser usados em vários modelos para implementar e manter um ambiente de computador seguro. Seria sensato determinar qual modelo de controle de acesso usar na sua organização no início da fase de planejamento.

Há três modelos de controle primário de acesso:

1. Controle de Acesso Discricionário (*Discretionary Access Control*, DAC)
2. Controle de Acesso Mandatório (*Mandatory Access Control*, MAC)
3. Controle de Acesso Baseado em Papéis (*Role Based Access Control*, RBAC)

Todos os três modelos são usados atualmente nos sistemas Linux. Aquele que você escolhe usar depende das necessidades de segurança da organização. As seções a seguir discutem cada modelo em detalhes e o ajudarão a escolher o modelo certo para sua organização.

Controle de acesso discricionário

O *Controle de Acesso Discricionário* (DAC) é o controle de acesso tradicional de segurança do Linux. O acesso a um objeto baseia-se na identidade de um sujeito e dos membros de um grupo. No Capítulo 4, discutimos as permissões de arquivos, como leitura, gravação e execução. Além disso, veremos posse de arquivo e grupos. Isso é o DAC básico para um sistema Linux. Acesso a um determinado objeto ou arquivo baseia-se nas permissões definidas para o proprietário desse objeto, para os grupos e para os outros.

No exemplo a seguir, o arquivo, `DAC_file` tem permissões de leitura e gravação dadas ao proprietário do arquivo, `christine`. A permissão de leitura é dada ao grupo, `writer`, assim como aos demais usuários do sistema.

```
$ ls -l DAC_File
-rw-r--r-- 1 christine writer 41094 2012-02-02 13:24 DAC_File
```

A principal fraqueza do DAC é que o proprietário de um objeto tem o controle das configurações de segurança desse objeto. Em vez de usar uma política estabelecida pela organização, o proprietário do

objeto pode fazer alterações arbitrárias nos controles de segurança. Usando o arquivo e seu proprietário do exemplo anterior, o sujeito `christine` pode modificar as permissões do objeto `DAC_File`:

```
$ chmod 777 DAC_File
$ ls -l DAC_File
-rwxrwxrwx 1 christine christine 41094 2012-02-02 13:24 DAC_File
```

Outra grande fraqueza do DAC está na diferenciação entre os sujeitos. O DAC só pode realizar transações com os proprietários do objeto e outros usuários. Ele não pode definir direitos de acesso para os aplicativos do sistema. Em outras palavras, o DAC não sabe quem é um humano e quem é um programa. Assim, quando um sujeito executa um programa, esse programa herda os direitos de acesso ao objeto do humano.

Controle de acesso mandatório

O *Controle de Acesso Mandatório* (MAC) supera a fragilidade do DAC e é um modelo popular de controle de acesso. Com o MAC, você tem um modelo muito mais refinado:

- Os objetos são classificados de acordo com a necessidade de integridade e privacidade de dados.
- Os sujeitos são classificados pelo nível de autorização sobre os dados ou o nível de autorização de segurança.
- O acesso de um sujeito a um objeto baseia-se no nível de autorização do sujeito.

O proprietário de um objeto não pode mais alterar as configurações de segurança desse objeto, como no modelo DAC. Todas as configurações e modificações de segurança são tratadas por uma pessoa ou equipe designada do processo de planejamento de segurança. Além disso, o MAC não permite que um programa herde o direito de acesso de um usuário a um objeto quando o programa é executado por esse usuário. Isso supera o segundo ponto fraco no modelo DAC.

No geral, os objetos recebem proteção contra ameaças maliciosas confinando os sujeitos aos respectivos níveis de autorização. Há um refinamento adicional no MAC que fornece níveis ainda mais altos de segurança, o modelo Controle de Acesso Baseado em Papéis (RBAC).

Controle de Acesso Baseado em Papéis

Controle de Acesso Baseado em Papéis (RBAC) foi desenvolvido no National Institute of Standards and Technology (NIST). Ele é um pequeno, mas importante, refinamento do modelo MAC. O acesso a um objeto baseia-se na identidade do sujeito e seu papel associado. Cada papel, não cada sujeito, é classificado de acordo com seu nível de autorização sobre os dados ou o nível de autorização de segurança. Assim, o acesso a objetos baseia-se em um nível de autorização do papel e não em um nível específico de autorização do sujeito.

O modelo RBAC tem várias vantagens:

- Os pontos fortes do modelo MAC são mantidos.
- Os direitos de acesso do papel são mais fáceis de classificar em níveis de autorização sobre dados ou níveis de autorização de segurança do que os direitos de acesso de um sujeito específico.

- Os direitos de acesso do papel raramente mudam.
- Mover os sujeitos individuais entre os vários papéis elimina a necessidade de reclassificação dos direitos de acesso de cada sujeito.

Portanto, com o RBAC, você tem um modelo de acesso mais simples e mais fácil de gerenciar que é tão forte quanto o modelo MAC. Esse casamento entre simplicidade e força tornou o Controle de Acesso Baseado em Papéis popular entre os produtos de segurança, incluindo o Security Enhanced Linux (SELinux), que é discutido no Capítulo 24.

Usando listas de verificação de segurança

Listas de verificação de segurança são uma parte necessária da fase de planejamento. Essas listas também serão úteis durante a fase de implementação do seu plano de segurança.

Matriz de controle de acesso

Uma lista de verificação útil é uma *Matriz de Controle de Acesso*. Essa matriz é simplesmente uma tabela que você mesmo cria. Ela enumera sujeitos ou papéis em relação a objetos e seus níveis de classificação.

atenção

tenha em mente que sua Matriz de Controle de Acesso é considerada sensível a dados; ela é um dos objetos dos dados e deve ser classificada de acordo com as necessidades de integridade e privacidade de dados. Apenas as pessoas com níveis apropriados de autorização devem ser autorizadas a ver e/ou modificá-la.

A Tabela 22.1 é uma matriz de controle de acesso simplificada para uma organização, utilizando um modelo de acesso MAC. Neste exemplo, os sujeitos (usuários e programas) são listados na primeira coluna e os objetos (programas e dados) são listados na primeira linha.

TABELA 22.1 Matriz de Controle de Acesso Baseada em um Modelo MAC

Sujeito/objeto	Programa ABC (Baixo)	Dados XYZ (Alto)
Smith, Christine	Sim	Sim
Jones, Tim	Sim	Não
ABC Program	N/D	Não

Na Tabela 22.1, você pode ver que Tim Jones tem acesso ao programa ABC, mas não aos dados XYZ. Além disso, o programa ABC não tem acesso aos dados XYZ. Essas matrizes permitem que uma equipe de planejamento de segurança revise facilmente os direitos de acesso e que uma equipe de implementação de segurança configure precisamente os direitos de acesso corretos.

Listas de verificação de segurança da indústria

As vulnerabilidades de segurança mudam constantemente. Toda vez que você se protege contra uma

potencial invasão ou um programa malicioso, outro ganha vida.

Por causa disso, listas de verificação de segurança são continuamente atualizadas por organizações governamentais e pela indústria. Essas listas são uma ótima ajuda para planejar como proteger seu sistema Linux. Eis alguns bons sites onde verificar listas de segurança novas e melhoradas. Para uma lista bem básica de segurança do Linux, visite www.sans.org/score/linuxchecklist.php. Se você precisar de uma lista de verificação de segurança mais detalhada, consulte as informações nestes sites:

- National Checklist Program's (NCP) National Vulnerability Database:
<http://web.nvd.nist.gov/view/ncp/repository>
- Normas de gerenciamento para executar um sistema de computador seguro da International Standards Organization (ISO): <http://iso27001security.com>
- Recomendações gerais da International Standards Organization para proteger sistemas de computador por meio de controles:
<http://iso27001security.com/html/27002.html>

Agora, com as classificações dos sujeitos e objetos, matriz de controle de acesso e listas de verificação de segurança, você pode passar para a próxima fase da segurança. A fase de implementação é onde você realmente começará a proteger o sistema de computador Linux.

Entrando na fase de execução

É irrealista pensar que você iniciaria a fase de implementação do Ciclo de Vida do Processo de Segurança apenas depois de concluir a fase de planejamento. A realidade é que muitas vezes as fases do Ciclo de Vida do Processo de Segurança se sobrepõem. Frequentemente, os sistemas de computador têm de estar prontos e instalados e já dentro do processo de garantia de segurança antes mesmo de uma fase de planejamento ser iniciada.

Por isso, é importante entender algumas implementações básicas de segurança. Esses princípios básicos incluem segurança física, recuperação após desastre, gerenciamento de conta de usuário, senhas de contas e outros temas relacionados.

Implementação da segurança física

A fechadura da porta da sala do servidor é uma primeira linha de defesa. Apesar de um conceito muito simples, isso é frequentemente ignorado. O acesso ao servidor físico significa acesso a todos os dados que ele contém. Nenhum software de segurança pode proteger seus sistemas se alguém com intenções maliciosas tiver acesso físico ao servidor Linux.

Segurança básica da sala física do servidor inclui itens como:

- Uma fechadura ou um alarme de segurança na porta da sala do servidor
- Controles de acesso que só permitem acesso autorizado e identificação de quem acessou a sala e

quando o acesso ocorreu, como um sistema de entrada com cartão

- Um sinal indicando “Proibida a entrada de pessoal não autorizado” na porta
- Políticas sobre quem pode acessar a sala e quando o acesso pode ocorrer, por grupos como a equipe de limpeza, administradores de servidor e outros

A segurança física inclui controles ambientais. Devem ser implementados sistemas apropriados de combate a incêndio e ventilação adequada para a sala dos servidores.

Além da segurança física básica de uma sala de servidores, deve ser dada atenção àquilo que está fisicamente na mesa de cada funcionário. Desktops e laptops talvez precisem ser mantidos bloqueados. Impressões digitais muitas vezes permanecem nos tablets, o que pode revelar PINs e senhas. Portanto, uma política de limpeza de tela dos tablets talvez precise ser implementada.

Uma política de mesa limpa (*Clean Desk Policy*, CDP) exige que ou apenas os documentos em que as pessoas estão atualmente trabalhando permaneçam em uma mesa ou que todos os documentos sejam trancados no final do dia. Uma CDP protege as informações sigilosas contra coleta por pessoal não autorizado ou bisbilhoteiros. E, por fim, uma política “nenhuma senha escrita” é obrigatória.

Implementando recuperação após desastre

Desastres acontecem e eles podem expor os dados de sua organização a situações inseguras. Portanto, parte da segurança do computador inclui a preparação para um desastre.

Semelhante ao ciclo de vida completo do processo de segurança, a recuperação após desastre inclui a criação de planos de recuperação após desastre, teste e revisão dos planos. Os planos devem ser testados e atualizados para manter sua confiabilidade em situações reais de desastre.

Os planos de recuperação após desastre devem incluir:

- Quais dados devem ser incluídos nos backups
- Onde os backups devem ser armazenados
- Por quanto tempo os backups são mantidos
- Como a mídia de backup é reciclada no armazenamento

Dados, mídia e software de backup devem ser incluídos na lista de verificação da matriz de controle de acesso.

ênção

importante determinar quantas cópias de backup de cada objeto devem ser mantidas. Embora você possa precisar apenas três cópias de segurança de um determinado objeto, outro objeto poderia ser importante o bastante para manter mais cópias.

Utilitários de backup em um sistema Linux incluem:

- amanda (Advanced Maryland Automatic Network Disk Archiver)
- cpio
- dump/restore
- tar

Os utilitários `cpio`, `dump/restore` e `tar` normalmente são pré-instalados em uma distribuição Linux. Apenas `amanda` não é instalado por padrão. Mas `amanda` é extremamente popular porque é bastante flexível e pode até fazer o backup de um sistema Windows. Se precisar de mais informações sobre o utilitário de backup `amanda`, consulte www.amanda.org. O utilitário que você em última instância escolhe deve atender às necessidades de segurança específicas da organização para backup.

Com sorte, os desastres só ocorrerão raramente. Mas, todos os dias, os usuários efetuam o login no seu sistema Linux. Contas e senhas de usuário têm configurações básicas de segurança que devem ser revisadas e implementadas conforme a necessidade.

Protegendo contas de usuário

As contas de usuário são parte do processo de autenticação que permite aos usuários se conectar ao sistema Linux. O gerenciamento adequado das contas de usuário aumenta a segurança de um sistema. A configuração de contas de usuário foi abordada no Capítulo 11. Mas algumas regras adicionais são necessárias para aumentar a segurança por meio do gerenciamento de contas de usuário. Elas incluem:

- Um usuário por conta de usuário.
- Nenhum login para a conta `root`.
- Definir datas de expiração para as contas temporárias.
- Remover contas de usuário não utilizadas.

Um usuário por conta de usuário

As contas devem reforçar a responsabilização. Assim, várias pessoas não devem efetuar o login em uma conta. Quando várias pessoas compartilham uma conta, não há como provar que um determinado indivíduo realizou uma ação particular. As ações deles são negáveis, o que é chamado *repúdio* no mundo da segurança. As contas devem ser configuradas como *não repúdio*. Em outras palavras, deve haver uma pessoa por conta de usuário de modo que as ações não possam ser negadas.

ca

sempre vai querer configurar a segurança do seu computador para não repúdio. Mas esse termo às vezes pode ser confuso. Para ajudá-lo a lembrar dos termos, pense desta maneira:

repúdio — O usuário pode negar ações ou recusar a responsabilidade.

não repúdio — O usuário não pode negar as ações ou recusar a responsabilidade.

Nenhum login na conta root

Se várias pessoas puderem fazer login na conta root, você terá outra situação de repúdio. Você não pode monitorar o uso individual da conta root e para permitir o monitoramento do uso dela por indivíduos, deve ser instituída uma política para usar `sudo` (ver Capítulo 8) em vez de login na conta root.

ta

Em Ubuntu, não há acesso concedido para efetuar login na conta root. Você deve usar `sudo` para todas as operações de administração.

Usar `sudo` fornece os seguintes benefícios de segurança:

- A senha da conta root não tem de ser distribuída.
- Você pode refinar o acesso a comandos.
- Todo o uso de `sudo` (quem, o que, quando) é gravado em `/var/log/secure`.
- Todas as tentativas de acesso malsucedido a `sudo` são registradas em `log`.

Embora o uso de `sudo` possa ser imposto, uma coisa que você precisa saber é que os usuários com acesso a `sudo` podem emitir o comando `sudo su -` para efetuar o login na conta root. No código a seguir, o usuário chamado `joe`, que teve seus privilégios de `sudo` configurados no Capítulo 8, agora utiliza o comando `sudo` para fazer login na conta root.

```
[joe]$ sudo su -
```

```
We trust you have received the usual lecture  
from the local System Administrator. It usually  
boils down to these two things:
```

```
#1) Respect the privacy of others.
```

```
#2) Think before you type.
```

```
Password: *****
```

```
[joe]# whoami
```

```
root
```

Isso essencialmente evita os benefícios de segurança `sudo`. Para manter seu sistema Linux seguro, você terá de desativar esse acesso.

Para desativar `sudo su -` para usuários `sudo` designados, você deve voltar e editar o arquivo de configuração `/etc/sudoers`. Mais uma vez, usando o exemplo do Capítulo 8, o usuário chamado `joe` atualmente tem o seguinte registro no arquivo `/etc/sudoers`:

```
joe ALL=(ALL) NOPASSWD: ALL
```

Para desativar a capacidade de `joe` de usar `sudo su -`, você precisará alterar o registro dele para que se pareça com o seguinte:

```
joe ALL=(ALL) NOPASSWD: !/bin/su, /bin/, /sbin/, /usr/bin/, /usr/sbin/
```

Joe ainda será capaz de executar todos os comandos no sistema como `root`, exceto `sudo su -`.

Adicionar o ponto de exclamação (!) na frente de `/bin/su` do registro de `joe` instrui `sudo` a não permitir que `joe` execute esse comando específico. Mas agora os comandos que `joe` tem permissão para executar devem estar listados no registro dele. `/bin/`, `/sbin/`, `/usr/bin/` e `/usr/sbin/` instruem `sudo` a permitir que `joe` execute todos os comandos localizados nesses diretórios.

ca

o sistema Linux talvez mantenha o comando `su` em outro local, em vez de `/bin`. Para determinar onde o comando `su` está localizado, digite o comando `which su` na linha de comando do Linux.

Se `joe` tentar usar `sudo su -`, ele receberá a seguinte mensagem:

```
[joe]$ sudo su -  
Sorry, user joe is not allowed to execute  
`/bin/su -' as root on host.domain
```

Agora, o uso do `sudo` no seu sistema Linux será monitorado adequadamente e o acesso a `sudo` terá a capacidade desejada de não repúdio. A segurança da sua organização precisa determinar a quantidade de detalhes que você deve adicionar ao arquivo de configuração `sudo`.

ta

figure grupos administrativos no sistema Linux, como `wheel` ou `admin`, no arquivo `/etc/sudoers` como não dio. Se um registro em log de usuário for configurado como nenhum acesso ao comando `su` no arquivo `/etc/sudoers file`, mas se esse usuário for um membro de um grupo que tem acesso a ele, o usuário ainda será emitir `sudo su -` no sistema. Portanto, também é necessário configurar `!/bin/su` para esse grupo administrativo.

Configurando as datas de expiração nas contas temporárias

Se houver consultores, estagiários ou empregados temporários que precisam de acesso aos seus sistemas Linux, será importante configurar as contas de usuário com datas de expiração. A data de expiração é uma salvaguarda, caso você se esqueça de remover as contas deles quando eles não mais precisarem de acesso aos sistemas da sua organização.

Para configurar uma conta de usuário com uma data de expiração, use o comando `usermod`. O formato é `usermod -e aaaa-mm-dd nome_do_usuario`. No código a seguir, a conta de `Tim` foi definida para expirar em 1º de janeiro de 2015.

```
# usermod -e 2015-01-01 Tim
```

Para verificar se a conta foi devidamente configurada para expirar, confirme você mesmo usando o comando `chage`. O comando `chage` é principalmente utilizado para visualizar e alterar as informações de envelhecimento de senha de uma conta de usuário. Mas esse comando também contém informações sobre expiração da conta. A opção `-l` permite classificar as várias informações às quais `chage` tem acesso. Para simplificar, redirecione a saída do comando `chage` para `grep` e procure a palavra “Account”. Isso produzirá somente a data de expiração da conta do usuário.

```
# chage -l Tim | grep Account
Account expires                : Jan 01, 2015
```

Como você pode ver, a data de expiração da conta de `Tim` foi alterada com sucesso para 1º de janeiro de 2015.

ca

ocê não usar o arquivo `/etc/shadow` para armazenar as senhas das suas contas, o utilitário `chage` não funcionará. A configuração do sistema para usar o arquivo `/etc/shadow` é discutida mais adiante, neste capítulo.

Configure as datas de expiração das contas de todos os funcionários temporários. Além disso, considere revisar a data de expiração das contas de usuários como parte das atividades de monitoramento de segurança. Essas atividades ajudam a eliminar quaisquer potenciais brechas em seu sistema Linux.

Removendo contas de usuários não utilizadas

Manter disponíveis contas antigas expiradas é sinônimo de problemas. Depois que um usuário saiu de uma organização, é melhor seguir uma série de passos para remover a conta dele juntamente com os dados:

1. Expirar ou desativar a conta.
2. Encontre arquivos no sistema pertencentes à conta usando o comando `find /-user nome_de_usuario`.
3. Faça backup dos arquivos.
4. Remova os arquivos ou reatribua-os a um novo proprietário.
5. Exclua a conta do sistema.

Os problemas ocorrem quando o Passo 5 é esquecido e as contas expiradas ou desativadas ainda estão no sistema. Um usuário mal-intencionado que ganha acesso ao seu sistema poderia renovar a conta e então se passar por um usuário legítimo.

Para encontrar essas contas, pesquise o arquivo `/etc/shadow`. A data de expiração da conta está no oitavo campo de cada registro em log. Seria conveniente se um formato de data fosse utilizado. Em vez disso, esse campo mostra a data de expiração da conta como o número de dias desde 1º de janeiro de 1970.

Você pode usar um processo de dois passos para encontrar automaticamente as contas expiradas no arquivo `/etc/shadow`. Primeiro, configure uma variável de shell (ver Capítulo 7) com data de hoje no formato “dias desde 1º de janeiro de 1970”. Então, usando o comando `gawk`, você pode obter e formatar as informações necessárias a partir do arquivo `/etc/shadow`.

Configurar uma variável de shell com a data atual convertida no número de dias desde 1º de janeiro de 1970 não é particularmente difícil. O comando `date` pode produzir o número de segundos desde 1º de janeiro de 1970. Para conseguir o que você precisa, divida o resultado do comando `date` pelo número de segundos em um dia: 86.400. Eis uma demonstração de como configurar a variável de shell `TODAY`.

```
# TODAY=$(echo $(($(date --utc --date "$1" +%s)/86400)))
# echo $TODAY
15373
```

Em seguida, as contas e as respectivas datas de expiração serão selecionadas do arquivo `/etc/shadow` usando `gawk`. O comando `gawk` é a versão GNU do programa `awk` usado no UNIX. A saída do comando é mostrada no código a seguir. Como seria de se esperar, muitas das contas não têm uma data de expiração. Mas duas contas, `Consultant` e `Intern`, mostram uma data de expiração no formato “dias desde 1º de janeiro de 1970”. Note que você pode pular essa etapa. Ela é apenas para fins de demonstração.

```
# gawk -F: '{print $1,$8}' /etc/shadow
...
chrony
tcpdump
johndoe
Consultant 13819
Intern 13911
```

O `$1` e `$8` no comando `gawk` representam os campos nome de usuário e data de expiração nos registros do arquivo `/etc/shadow`. Para verificar as datas de expiração dessas contas e ver se elas expiraram, é necessária uma versão mais refinada do comando `gawk`.

```
# gawk -F: '{if (($8 > 0) && ($TODAY > $8)) print $1}' /etc/shadow
Consultant
Intern
```

Somente contas com uma data de expiração são coletadas pela parte (`$8 > 0`) do comando `gawk`. Para certificar-se de que essas datas de expiração estão depois da data atual, a variável `TODAY` é comparada com o campo data de expiração, `$8`. Se `TODAY` for maior que a data de expiração da conta, a conta será listada. Como você pode ver no exemplo anterior, há duas contas vencidas que ainda estão no sistema e precisam ser removidas.

Isso é tudo o que você precisa fazer. Configure a variável `TODAY` e então execute o comando `gawk`. Todas as contas expiradas no arquivo `/etc/shadow` serão listadas para você. Para remover essas

contas, use o comando `userdel`.

Contas de usuário são apenas uma parte do processo de autenticação, permitindo que os usuários acessem o sistema Linux. Senhas de conta do usuário também desempenham um papel importante no processo.

Protegendo senhas

As senhas são a ferramenta de segurança mais básica de qualquer sistema operacional moderno e, conseqüentemente, o recurso de segurança mais comumente atacado. É natural que os usuários queiram escolher uma senha que seja fácil de lembrar, mas muitas vezes isso significa que eles selecionam uma senha que também é fácil de adivinhar.

Métodos de força bruta são normalmente utilizados para ganhar acesso a um sistema de computador. Tentar senhas populares frequentemente produz resultados. Algumas das senhas mais comuns são:

- 123456
- Senha
- princess
- rockyou
- abc123

Basta usar seu mecanismo de busca favorito na internet e procurar “senhas comuns”. Se você conseguir encontrar essas listas, então atacantes maliciosos também conseguem. Obviamente, escolher boas senhas é crucial para ter um sistema seguro.

Escolhendo boas senhas

Em geral, a senha não deve ser fácil de adivinhar, ser comum ou popular ou estar vinculada a você de qualquer forma. Eis algumas regras a seguir ao escolher uma senha:

- Não utilize qualquer variação de seu nome de login nem seu nome completo.
- Não use uma palavra de dicionário.
- Não utilize nomes próprios de qualquer tipo.
- Não utilize seu número de telefone, endereço, sobrenomes, nem nomes de animais de estimação.
- Não use nomes de sites.
- Não use nenhuma linha contígua de letras ou números no teclado (como “qwerty” ou “asdfg”).
- Não use nenhuma das recomendações acima com números ou pontuações adicionados ao início ou final, ou digitadas de trás para frente.

Bem, agora, você sabe o que não fazer — dê uma olhada nos dois itens principais que compõem uma senha forte:

1. Uma senha deve ter pelo menos 15 a 25 caracteres de comprimento.
2. Uma senha deve conter todos os seguintes:
 - Letras minúsculas
 - Letras maiúsculas
 - Números
 - Caracteres especiais, como: ! \$ % * () - + = , < > : : “ ‘

Vinte e cinco caracteres é uma senha longa. Mas quanto maior a senha, mais segura ela é. O que sua organização escolhe como o comprimento mínimo de senha depende das suas necessidades de segurança.

ca

Johnson Research Center tem materiais excelentes sobre senhas fortes, incluindo um artigo chamado “De que tamanho é seu monte de feno... e quão bem sua agulha está escondida?” em www.jrc.com/haystack.htm.

Escolher uma boa senha pode ser difícil. Ela tem de ser difícil o bastante para não ser adivinhada e fácil o bastante para que você possa lembrar. Uma boa maneira de escolher uma senha forte é selecionar a primeira letra de cada palavra de uma frase fácil de lembrar. Certifique-se de adicionar números, caracteres especiais e letras maiúsculas e minúsculas. A frase que você selecionar deve fazer sentido apenas para você, e não deve estar publicamente disponível. A Tabela 22.2 lista exemplos de senhas fortes e os truques usados para lembrar-se delas.

TABELA 22.2 Ideias para Boas Senhas

Senha	Como lembrar
ci7yo!	My rusty car is 7 years old!
mBp1ib	2 elephants make BAD pets, 1 is better
Mc?Gib	Is that MY coat? Give it back

As senhas parecem não fazer sentido, mas, na verdade, são relativamente fáceis de lembrar. É claro, não se esqueça de não usar as senhas listadas aqui. Agora que são públicas, elas serão adicionadas aos dicionários dos invasores maliciosos.

Configurando e alterando senhas

Você define sua própria senha usando o comando `passwd`. Digite o comando `passwd` e ele permitirá que você mude sua senha. Primeiro, ele pede que você digite sua senha antiga. Para evitar que alguém espiando por trás de você possa vê-la, a senha não será exibida enquanto você digita.

Supondo que você digite sua senha antiga corretamente, o comando `passwd` pedirá a nova senha. Ao digitar sua nova senha, ela é verificada usando um utilitário chamado `cracklib` para determinar se ela

é uma senha boa ou ruim. Usuários não root deverão tentar uma senha diferente se aquela que eles escolheram não for uma boa senha.

O usuário root é o único usuário que tem permissão de atribuir senhas ruins. Depois que a senha foi aceita por `cracklib`, o comando `passwd` pede para você digitar a nova senha uma segunda vez para garantir que não há erros de digitação (que são difíceis de detectar quando você não pode ver aquilo que está digitando).

Ao executar como root, é possível alterar a senha de um usuário fornecendo o nome de login desse usuário como um parâmetro para o comando `passwd`. Por exemplo:

```
# passwd joe
Changing password for user joe.
New UNIX password: ****
Retype new UNIX password: ****
passwd: all authentication tokens updated successfully.
```

Aqui, o comando `passwd` pede duas vezes para digitar uma nova senha para `joe`. Nesse caso, ele não solicita a senha antiga dele.

Impondo melhores práticas de senha

Agora, você sabe como seria uma boa senha e como mudá-la, mas como a impõe no seu sistema Linux? Uma boa maneira de começar a impor as melhores práticas de senha é educar os usuários do sistema. Usuários educados são usuários melhores. Algumas ideias para a educação incluem:

- Adicionar um artigo sobre as melhores práticas de senha ao boletim mensal da sua organização.
- Postar folhas de dicas nas salas de descanso, como “As dez piores senhas”.
- Enviar regularmente e-mails sobre segurança de computador aos funcionários contendo dicas de senha.
- Fornecer treinamento sobre senhas aos novos funcionários.

Funcionários que entendem a segurança de senha muitas vezes se esforçam para criar boas senhas no trabalho, bem como em casa. Um dos “ganchos” para ganhar a atenção do usuário é deixar que os funcionários entendam que essas senhas também funcionam bem ao criar senhas pessoais, como para suas contas bancárias online.

Ainda assim, sempre haverá alguns usuários que se recusam a implementar boas práticas de senha. Além disso, as políticas de segurança da empresa muitas vezes exigem que a senha seja trocada a cada certo número de dias. Pode se tornar cansativo pensar em novas senhas fortes a cada 30 dias! É por isso que frequentemente é necessário impor algumas técnicas.

ca

Os usuários tiverem dificuldades para criar senhas seguras e únicas, considere instalar o utilitário `pwgen` no sistema Linux. Esse utilitário de geração de senhas de código-fonte aberto cria senhas que são feitas para serem

unciáveis e memoráveis. Você pode usar essas palavras geradas como um ponto de partida para criar senhas de as.

Valores padrão no arquivo `/etc/login.defs` para novas contas foram abordados no Capítulo 11. No arquivo `login.defs` estão algumas configurações que afetam o envelhecimento e a duração das senhas:

<code>PASS_MAX_DAYS</code>	30
<code>PASS_MIN_DAYS</code>	5
<code>PASS_MIN_LEN</code>	16
<code>PASS_WARN_AGE</code>	7

Nesse exemplo, o número máximo de dias, `PASS_MAX_DAYS`, até que a senha precise ser alterada é 30. O número que você define aqui depende da configuração específica da sua conta. Para as organizações que praticam a diretriz de “uma pessoa para uma conta”, esse número pode ser muito maior que 30. Se houver contas compartilhadas ou várias pessoas que conhecem a senha de root, é imperativo alterar a senha com frequência. Essa prática atualiza efetivamente a lista das pessoas que conhecem a senha.

Para evitar que os usuários alterem as senhas para uma nova senha e então imediatamente mude-as novamente, você terá de definir `PASS_MIN_DAYS` como um número maior que 0. No exemplo anterior, o período de tempo mais curto em que um usuário pode alterar novamente a senha é 5 dias.

A configuração `PASS_WARN_AGE` é o número de dias que um usuário será avisado antes de ser forçado a mudar sua senha. As pessoas tendem a precisar de muitas advertências e estímulos, portanto, o exemplo anterior define o tempo de aviso como 7 dias.

No início deste capítulo, mencionei que uma senha forte tem de 15 a 25 caracteres. Com a definição `PASS_MIN_LEN`, você pode forçar os usuários a usar um determinado número mínimo de caracteres nas senhas. A configuração escolhida deve basear-se nos planos do ciclo de vida de segurança da organização.

ta

untu não tem a configuração `PASS_MIN_LEN` no arquivo `login.defs`. Em vez disso, essa configuração é da pelo utilitário PAM. O PAM é abordado no Capítulo 23.

Para as contas que já foram criadas, você precisará controlar o envelhecimento de senha por meio do comando `chage`. As opções necessárias para controlar o envelhecimento de senha com `chage` estão listadas na Tabela 22.3. Observe que não há uma definição de duração de senha no utilitário `chage`.

TABELA 22.3 Opções de `chage`

Opção	Descrição
<code>-M</code>	Configura o número máximo de dias antes de ser necessário alterar uma senha.

Equivalente a `PASS_MAX_DAYS` em `/etc/login.defs`

Configura o número mínimo de dias antes de uma senha poder ser alterada novamente.
Equivalente a `PASS_MIN_DAYS` em `/etc/login.defs`

Configura o número de dias que um usuário será avisado antes de ser forçado a mudar a senha da conta. Equivalente a `PASS_WARN_AGE` em `/etc/login.defs`

O exemplo a seguir usa o comando `chage` para configurar os parâmetros de envelhecimento de senha para a conta `Tim`. Todas as três opções são usadas ao mesmo tempo.

```
# chage -l Tim | grep days
Minimum number of days between password change      : 0
Maximum number of days between password change      : 99999
Number of days of warning before password expires   : 7
#
# chage -M 30 -m 5 -W 7 Tim
#
# chage -l Tim | grep days
Minimum number of days between password change      : 5
Maximum number of days between password change      : 30
Number of days of warning before password expires   : 7
```

Você também pode usar o comando `chage` como outro método de vencimento de conta, que se baseia na expiração de senha da conta. Anteriormente, o utilitário `usermod` foi usado para expiração de conta. Use o comando `chage` com as opções `-M` e `-I` para bloquear a conta. No código a seguir, a conta de `Tim` é visualizada usando `chage -l`. Apenas as informações para as senhas da conta `Tim` são extraídas.

```
# chage -l Tim | grep Password
Password expires          : never
Password inactive        : never
```

Você pode ver que não há definições para a expiração de senha (`Password expires`) ou inatividade de senha (`Password inactive`). No código a seguir, a conta está configurada para ser bloqueada 5 dias depois que a senha de `Tim` expirar, usando apenas a opção `-I`.

```
# chage -I 5 Tim
#
# chage -l Tim | grep Password
Password expires          : never
```

```
Password inactive
```

```
: never
```

Observe que nenhuma configuração mudou! Sem uma configuração de expiração de senha, a opção `-I` não tem nenhum efeito. Assim, usando a opção `-M`, o número máximo de dias será definido antes de a senha expirar e a configuração para o tempo de inatividade de senha será utilizada.

```
# chage -M 30 -I 5 Tim
```

```
#
```

```
# chage -l Tim | grep Password
```

```
Password expires           : Mar 03, 2014
```

```
Password inactive         : Mar 08, 2014
```

Agora, a conta de `Tim` será bloqueada 5 dias após sua senha expirar. Isso é útil em situações em que um funcionário saiu da empresa, mas sua conta de usuário ainda não foi removida. Dependendo das necessidades de segurança da organização, considere configurar todas as contas para que sejam bloqueadas alguns dias após a senha ter expirado.

Entendendo os arquivos de senha e hashes de senha

Os primeiros sistemas Linux armazenavam as senhas no arquivo `/etc/passwd`. As senhas eram *hasheadas*. Uma senha hasheada é criada usando um processo matemático unidirecional. Depois de criar o hash, você não pode recriar os caracteres originais a partir do hash. Eis como isso funciona: quando um usuário digita a senha da conta, o sistema Linux define um novo hash para a senha e, então, compara o resultado do hash com o hash original em `/etc/passwd`. Se eles corresponderem, o usuário é autenticado e autorizado a acessar o sistema.

O problema com o armazenamento desses hashes de senha no arquivo `/etc/passwd` tem a ver com as configurações de segurança do sistema de arquivos (veja Capítulo 4). As configurações de segurança do sistema de arquivos para o arquivo `/etc/passwd` estão listadas aqui:

```
# ls -l /etc/passwd
```

```
-rw-r--r--. 1 root root 1644 Feb 2 02:30 /etc/passwd
```

Como você pode ver, todo mundo pode ler o arquivo de senha. Você poderia achar que isso não é um problema porque todas as senhas estão hasheadas. Mas indivíduos com intenções maliciosas criaram arquivos chamados *tabelas arco-íris*. Uma tabela arco-íris é simplesmente um dicionário de potenciais senhas que foram hasheadas. Por exemplo, a tabela arco-íris deve conter o hash para a popular senha “Password”, que é:

```
$6$dhN5ZMUj$CNghjYIteau5x18yX.f6PTOpendJwTOcXj1TDQUQZhhy  
V8hKzQ6Hxx6Egj8P3VsHJ8Qrkv.VSR5dxcK3QhyMc.
```

Por causa do acesso fácil aos hashes de senha no arquivo `/etc/passwd`, é apenas uma questão de tempo antes que uma senha hasheada seja correspondida em uma tabela arco-íris e a senha de texto simples seja descoberta.

ta

especialistas em segurança dirão que as senhas não estão apenas hasheadas, mas elas também estão salgadas. Adicionar um hash significa que um valor gerado aleatoriamente é adicionado à senha original antes de ela ser hasheada. Isso torna ainda mais difícil que a senha hasheada seja correspondida com a senha original. Mas, no Linux, o sal de hash também é armazenado com as senhas hasheadas. Assim, acesso de leitura ao arquivo `/etc/passwd` significa que você tem o valor de hash e seu sal.

Portanto, as senhas hasheadas foram transferidas para um novo arquivo de configuração, `/etc/shadow`. Esse arquivo tem as seguintes configurações de segurança:

```
# ls -l /etc/shadow
-r----- . 1 root root 1049 Feb 2 09:45 /etc/shadow
```

Somente root pode visualizar esse arquivo. Assim, as senhas hasheadas permanecem protegidas. Eis o final de um arquivo `/etc/shadow`. Você pode ver que há palavras longas sem sentido no registro de cada usuário. Essas palavras são as senhas hasheadas.

```
# tail -2 /etc/shadow
johndoe:$6$jJjdRN9/qELmb8xWM1LgOYGhEIxc/:15364:0:99999:7:::
Tim:$6$z760AJ42$QXdhFyndpbVPVM5oVtNHs4B/:15372:5:30:7:16436::
```

enção

Se você poderia herdar um sistema Linux que ainda utiliza o método antigo de manter as senhas hasheadas no arquivo `/etc/passwd`. É fácil corrigir isso. Basta usar o comando `pwconv` e o arquivo `/etc/shadow` será criado e as senhas hasheadas transferidas para ele.

As seguintes informações também são armazenadas no arquivo `/etc/shadow`, além do nome da conta e a senha hasheada:

- Número de dias (desde 1º de janeiro de 1970) desde que a senha foi alterada
- Número de dias antes de a senha poder ser alterada
- Número de dias antes de uma senha precisar ser alterada
- Número de dias para avisar um usuário antes de uma senha precisar ser alterada
- Número de dias em que conta é desabilitada após a senha expirar
- Número de dias (desde 1º de janeiro de 1970) que uma conta foi desativada

Isso deve parecer familiar, uma vez que são as definições para o envelhecimento de senha abordadas anteriormente no capítulo. Lembre-se de que o comando `chage` não funcionará se você não tiver um arquivo `/etc/shadow` configurado, nem o arquivo `/etc/login.defs` estará disponível.

Obviamente, as configurações de segurança do sistema de arquivos são muito importantes para manter seu sistema Linux seguro. Isso é especialmente verdadeiro para todos os arquivos de configuração de sistemas Linux e outros sistemas.

Protegendo o sistema de arquivos

Outra parte importante da fase de implementação é configurar adequadamente a segurança do sistema de arquivos. Os princípios básicos para as configurações de segurança foram abordados no Capítulo 4 e Listas de Controle de Acesso (ACL) no Capítulo 11. Mas há algumas outras questões que precisam ser adicionados à sua base de conhecimento.

A matriz de controle de acesso criada na fase de planejamento será uma grande ajuda aqui. A configuração dos proprietários e grupos de um arquivo ou diretório, bem como suas permissões, deve ser guiada pela matriz. Será um processo longo e entediante, mas vale a pena o tempo e esforço. Além de seguir a matriz de controle de acesso da sua organização, há algumas permissões e configurações adicionais de arquivo às quais você precisa prestar atenção.

Gerenciando permissões perigosas do sistema de arquivos

Se você concedeu acesso `rwXrwXrwx` (777) total a cada arquivo no sistema Linux, você poderá imaginar o caos que viria a seguir. De muitas maneiras, caos semelhante pode ocorrer se você não gerenciar de perto as permissões SetUID (SUID) e SetGID (SGID) (consulte os Capítulos 4 e 11).

Arquivos com a permissão SUID na categoria `Owner` e permissão de execução na categoria `Other` permitirão que qualquer pessoa torne-se temporariamente o proprietário do arquivo enquanto ele está sendo executado na memória. O pior caso seria se o proprietário do arquivo fosse `root`.

Da mesma maneira, arquivos com a permissão SGID na categoria `Owner` e permissão de execução na categoria `Other` permitirão que qualquer pessoa torne-se temporariamente um membro do grupo do arquivo enquanto ele está sendo executado na memória. SGID também pode ser definido nos diretórios. Isso define o ID de grupo de todos os arquivos criados no diretório para o ID do grupo do diretório.

Arquivos executáveis com SUID ou SGID são os favoritos de usuários maliciosos. Assim, o melhor é utilizá-los moderadamente. Mas alguns arquivos precisam manter essas configurações. Dois exemplos são os comandos `passwd` e `sudo` a seguir. Cada um desses arquivos deve manter suas permissões SUID.

```
$ ls -l /usr/bin/passwd
-rwsr-xr-x. 1 root root 28804 Feb 8 2011 /usr/bin/passwd
$
$ ls -l /usr/bin/sudo
---s--x--x. 2 root root 77364 Nov 10 04:50 /usr/bin/sudo
```

Mas alguns sistemas Linux vêm com o comando `ping` definido com a permissão SUID. Isso poderia permitir que um usuário iniciasse um ataque de negação de serviço (*denial-of-service*, DoS), chamado tempestade de ping. Para remover a permissão, use o comando `chmod` desta maneira:

```
$ chmod u-s /bin/ping
$ ls -l /bin/ping
-rwxr-xr-x. 1 root root 39344 Nov 10 04:32 /bin/ping
```

Remoção e concessão de permissões SUID e GUID a objetos devem estar na matriz de controle de acesso e serem determinadas durante a fase do processo de ciclo de vida do planejamento de segurança.

Protegendo os arquivos de senha

O arquivo `/etc/passwd` é o arquivo que o sistema Linux usa para verificar as informações da conta do usuário e foi discutido anteriormente neste capítulo. O arquivo `/etc/passwd` deve ter as seguintes configurações de permissão:

- Proprietário: root
- Grupo: root
- Permissões: (644) Proprietário: rw- Grupo: r-- Outros: r--

O exemplo a seguir mostra que o arquivo `/etc/passwd` tem as configurações adequadas.

```
# ls -l /etc/passwd
-rw-r--r--. 1 root root 1644 Feb 2 02:30 /etc/passwd
```

Essas configurações são necessárias para que os usuários possam fazer login no sistema e ver os nomes de usuário associados com os números de ID de usuário e ID de grupo. Mas os usuários não devem ser capazes de modificar o arquivo `/etc/passwd` diretamente. Por exemplo, um usuário malicioso poderia adicionar uma nova conta ao arquivo se o acesso de gravação fosse concedido a `Other`.

O próximo arquivo é o `/etc/shadow`. Obviamente, ele está estreitamente relacionado com o arquivo `/etc/passwd`, porque também é utilizado durante o processo de autenticação de login. Esse arquivo `/etc/shadow` deve ter as seguintes configurações de permissão:

- Proprietário: root
- Grupo: root
- Permissões: (400) Proprietário: r -- Grupo: --- Outros: ---

O código a seguir mostra que o arquivo `/etc/shadow` tem as configurações adequadas.

```
# ls -l /etc/shadow
-r-----. 1 root root 1049 Feb 2 09:45 /etc/shadow
```

O arquivo `/etc/passwd` tem acesso de leitura para proprietário, grupo e outro. Observe como o arquivo `/etc/shadow` é muito mais restrito que o arquivo `/etc/passwd`. Para o arquivo `/etc/shadow`, apenas o proprietário recebe acesso de leitura. Isso é assim para que apenas root possa visualizar esse arquivo, como mencionado anteriormente. Assim, se apenas root puder visualizar esse

arquivo, como um usuário conseguiria mudar sua senha uma vez que ela está armazenada no arquivo `/etc/shadow`? O utilitário `passwd`, `/usr/bin/passwd`, usa o SUID de permissão especial. Essa configuração de permissão é mostrada aqui:

```
# ls -l /usr/bin/passwd
-rwsr-xr-x. 1 root root 28804 Feb 8 2013 /usr/bin/passwd
```

Portanto, o usuário que executa o comando `passwd` torna-se temporariamente `root` enquanto o comando é executado na memória, e pode, então, gravar no arquivo `/etc/shadow`.

ta
não tem acesso de gravação às permissões `/etc/shadow`, então como `root` grava no arquivo `/etc/shadow`? O usuário `root` é o todo-poderoso e tem acesso completo a todos os arquivos, quer as permissões sejam ou não listadas.

O arquivo `/etc/group` (ver Capítulo 11) contém todos os grupos no sistema Linux. Suas permissões de arquivo devem ser definidas exatamente como o arquivo `/etc/passwd`:

- Proprietário: `root`
- Grupo: `root`
- Permissões: (644) Proprietário: `rw-` Grupo: `r--` Outros: `r--`

Além disso, o arquivo de senha do grupo, `/etc/gshadow`, precisa estar adequadamente protegido. Como seria de se esperar, a permissão do arquivo deve ser definida exatamente como o arquivo `/etc/shadow`:

- Proprietário: `root`
- Grupo: `root`
- Permissões: (400) Proprietário: `r--` Grupo: `---` Outros: `---`

Bloqueando o sistema de arquivos

A tabela do sistema de arquivos (ver Capítulo 12), `/etc/fstab`, também precisa de atenção especial. O arquivo `/etc/fstab` é usado na inicialização para montar dispositivos. Ele também é usado pelos comandos `mount`, `dump` e `fsck`. O arquivo `/etc/fstab` deve ter as seguintes configurações de permissão:

- Proprietário: `root`
- Grupo: `root`
- Permissões: (644) Proprietário: `rw-` Grupo: `r--` Outros: `r--`

Como o arquivo `/etc/fstab` é legível por todos no sistema, nenhum campo de comentário descrevendo os dispositivos montados deve ser incluído. Não há necessidade de revelar informações adicionais sobre o sistema de arquivos que possam ser utilizadas por um usuário mal-intencionado.

Dentro da tabela do sistema de arquivos, há algumas configurações de segurança importantes que precisam ser revisadas. Além das partições `root`, `boot` e `swap`, considere o seguinte:

- Armazene o subdiretório `/home`, onde os diretórios de usuário estão localizados, em uma partição própria e:
 - Configure a opção `nosuid` para evitar que programas executáveis com permissões `SUID` e `SGID` ativadas sejam executados aí. Programas que precisam de permissões `SUID` e `SGID` não devem ser armazenados em `/home` e provavelmente são os mais maliciosos.
 - Configure a opção `nodev` de modo que nenhum arquivo de dispositivo localizado aí seja reconhecido. Os arquivos de dispositivo devem ser armazenados em `/dev`, não em `/home`.
 - Configure a opção `noexec` de modo que nenhum programa executável armazenado em `/home` possa ser executado. Programas executáveis não devem ser armazenados em `/home` e provavelmente são os mais maliciosos.
- Coloque o subdiretório `/tmp`, onde os arquivos temporários estão localizados, em uma partição própria e use as mesmas configurações de opções que `/home`:
 - `nosuid`
 - `nodev`
 - `noexec`
- Armazene o subdiretório `/usr`, onde os programas e dados do usuário estão localizados, em uma partição própria e configure a opção `nodev` para que nenhum arquivo de dispositivo localizado aí seja reconhecido.
- Coloque o subdiretório `/var`, onde os arquivos importantes de log de segurança estão localizados, em uma partição própria e use as mesmas configurações de opções que `/home`:
 - `nosuid`
 - `nodev`
 - `noexec`

Colocar o item anterior no arquivo `/etc/fstab` irá se parecer com isto:

```
/dev/sdb1    /home        ext4          nodev,noexec,nosuid1 2
/dev/sdc1    /tmp         ext4          nodev,noexec,nosuid1 1
/dev/sdb2    /usr         ext4          nodev          1 2
/dev/sdb3    /var         ext4          nodev,noexec,nosuid1 2
```

Essas opções de `mount` ajudarão a bloquear ainda mais o sistema de arquivos e adicionarão outra camada de proteção contra pessoas com intenções maliciosas.

ta

Passado, era comum que os usuários mantivessem os programas nos diretórios iniciais `$HOME/bin`. Isso pode ser uma prática muito perigosa do ponto de vista da segurança. Mas são as necessidades e normas de segurança da organização que irão determinar se essa prática deve continuar.

Mais uma vez, o gerenciamento das várias permissões de arquivo e opções de `fstab` devem estar na matriz de controle de acesso e ser determinado durante a fase de Planejamento do Ciclo de Vida do Processo de Segurança. Os itens que você escolhe implementar devem ser determinados pelas necessidades de segurança da organização.

Gerenciando softwares e serviços

Muitas vezes, o foco do administrador é garantir os softwares e serviços necessários em um sistema Linux. Do ponto de vista da segurança, você precisa adotar a perspectiva oposta e assegurar que softwares e serviços desnecessários não estejam em um sistema Linux.

Removendo softwares e serviços não utilizados

Durante a fase de planejamento do Ciclo de Vida do Processo de Segurança, um dos seus resultados deve ser uma “lista de verificação de softwares e serviços necessários”. Esse será seu ponto de partida para determinar o que remover do sistema. Com a lista de verificação em mãos, você pode começar o processo de revisão e remoção de softwares e serviços desnecessários:

1. Revise os serviços atuais no sistema.
2. Observe quais serviços não utilizados devem ser removidos.
3. Desabilite e remova os serviços não utilizados.
4. Revise os softwares instalados no sistema.
5. Observe quais softwares não utilizados devem ser removidos.
6. Remova pacotes de software não utilizados.

Cada remoção de serviços (ver Capítulo 15) e pacotes de software (ver Capítulo 10) deve ser inicialmente realizada em um ambiente de teste de algum tipo. Isso é especialmente verdadeiro se você não tiver certeza do efeito que isso poderia ter sobre um sistema de produção.

Atualizando pacotes de software

Além de remover os serviços e softwares desnecessários, manter os softwares atualizados é fundamental para a segurança. As correções de bugs e patches de segurança mais recentes são obtidas via atualizações de software. As atualizações de pacotes de software foram discutidas nos Capítulos 9 e 10.

As atualizações de software precisam ser feitas regularmente. Quantas vezes e quando você faz isso, naturalmente, depende das necessidades de segurança da organização.

Você pode automatizar facilmente as atualizações de software, mas como na remoção de serviços e softwares, seria prudente primeiro testar as atualizações em um ambiente de teste. Depois que os softwares atualizados não mostram problemas, você pode atualizá-los nos sistemas de produção Linux.

Implementação avançada

Há vários outros tópicos importantes na fase de implementação do Ciclo de Vida do Processo de Segurança. Eles incluem criptografia, Pluggable Authentication Modules (PAM) e SELinux. Esses tópicos avançados e detalhados foram colocados em capítulos separados, Capítulo 23 e Capítulo 24.

Trabalhando na fase de monitoramento

Se você fizer um bom trabalho de planejamento e implementação de segurança do seu sistema, a maioria dos ataques maliciosos será detida. Mas se um ataque ocorrer, você precisa ser capaz de reconhecê-lo. O monitoramento é uma atividade que precisa acontecer continuamente, independentemente de onde sua organização está atualmente no Ciclo de Vida do Processo de Segurança.

Nessa fase, você monitora os arquivos de log, contas de usuários e o próprio sistema de arquivos. Além disso, você precisará de algumas ferramentas para ajudá-lo a detectar invasões e outros tipos de malware.

Monitorando arquivos de log

Entender os arquivos de log em que o Linux grava eventos importantes é fundamental para a fase de monitoramento. Esses logs devem ser revistos regularmente.

Os arquivos de log do sistema Linux estão principalmente localizados no diretório `/var/log`. A maioria dos arquivos no diretório `/var/log` é mantida pelo serviço `rsyslogd` (ver Capítulo 13). A Tabela 22.4 contém uma lista dos arquivos `/var/log` e uma breve descrição de cada um.

TABELA 22.4 Arquivos de log no Diretório `/var/log`

Nome do log de sistema	Nome de arquivo	Descrição
Apache Access Log	<code>/var/log/httpd/access_log</code>	Armazena em log solicitações de informações do seu servidor Apache Web.
Apache Error Log	<code>/var/log/httpd/error_log</code>	Registra em log erros de clientes tentando acessar dados no servidor Apache Web.
Failed Logins Log	<code>btmp</code>	Registra tentativas de login malsucedidas.

ot Log	boot.log	Contém mensagens indicando quais serviços de sistema foram inicializados e desativados com sucesso e quais (se houver algum) não conseguiram ser inicializados ou parados. As mensagens de inicialização mais recentes estão no fim do arquivo.
ot Log	dmesg	Registra mensagens impressas pelo kernel durante a inicialização do sistema.
n Log	cron	Contém mensagens de status do daemon crond.
g Log	dpkg.log	Contém informações sobre os pacotes Debian instalados.
P Log	vsftpd.log	Contém mensagens relacionadas a transferências feitas utilizando o daemon vsFTPd (servidor FTP).
P Transfer Log	xferlog	Contém informações sobre os arquivos transferidos usando o serviço FTP.
HOME Display Manager Log	/var/log/gdm/:0.log	Guarda mensagens relacionadas à tela de login (gerenciador de vídeo do GNOME). Sim, há realmente um caractere de dois-pontos no nome de arquivo.
stLog	lastlog	Registra a última vez que uma conta efetua login no sistema.
gin/out Log	wtmp	Contém um histórico dos logins e logouts no sistema.
il Log	maillog	Contém informações sobre endereços para os quais e dos quais foi enviado correio eletrônico. Útil para detectar spamming.
SQL Server Log	mysqld.log	Inclui informações relacionadas a atividades do servidor do banco de dados MySQL (mysqld).
ws Log	spooler	Fornece um diretório contendo os logs das mensagens a partir do servidor Usenet News se você estiver executando-o.
M Packages	rpm_pkgs	Contém uma listagem de pacotes de RPM que estão instalados em seu sistema. Não está mais

		disponível por padrão. Você pode instalar o pacote ou simplesmente usar <code>rpm-qa</code> para visualizar uma lista dos pacotes RPM instalados.
amba Log	<code>/var/log/samba/log.smbd</code>	Mostra as mensagens do serviço de daemon de arquivo Samba SMB.
curity Log	<code>secure</code>	Registra a data, a hora e a duração de tentativas de login e das sessões.
idmail Log	<code>sendmail</code>	Mostra mensagens de erro registradas pelo daemon <code>sendmail</code> .
id Log	<code>/var/log/squid/access.log</code>	Contém mensagens relacionadas ao servidor de proxy/cache <code>squid</code> .
stem Log	<code>messages</code>	Fornecer um arquivo de log de uso geral em que muitos programas registram mensagens.
ICP Log	<code>uucp</code>	Mostra mensagens de status do UNIX para o daemon UNIX Copy Protocol.
IM Log	<code>yum.log</code>	Mostra mensagens do Yellow Dog Update Manager.
Org X11 Log	<code>Xorg.0.log</code>	Inclui mensagens produzidas pelo servidor X.Org X.

Os arquivos de log que estão no diretório `/var/log` do sistema dependerão dos serviços que estão em execução. Além disso, alguns arquivos de log são dependentes da distribuição. Por exemplo, se você usar o Fedora Linux, você não terá o arquivo de log `dpkg`.

A maioria dos arquivos de log é exibida usando os comandos `cat`, `head`, `tail`, `more` ou `less`. Mas alguns deles têm comandos especiais para visualização (ver Tabela 22.5).

TABELA 22.5 Visualizando Arquivos de log que Precisam de Comandos Especiais

Nome de arquivo	Comando de visualização
<code>utmp</code>	<code>dump-utmp btmp</code>
<code>esg</code>	<code>dmesg</code>
<code>log</code>	<code>lastlog</code>
<code>wtmp</code>	<code>dump-utmp wtmp</code>

Para um exemplo prático de como você pode usar os arquivos de log para monitoramento, veja a seguir o final do arquivo `/var/log/messages`. Observe que o comando `yum` foi usado para instalar o pacote `sysstat`.

```
# tail -5 /var/log/messages
Feb 3 11:51:10 localhost dbus-daemon[720]: **
```



```
Message: D-Bus service launched with name:
net.reactivated.Fprint
Feb 3 11:51:10 localhost dbus-daemon[720]: **
Message: entering main loop
Feb 3 11:51:40 localhost dbus-daemon[720]: **
Message: No devices in use, exit
Feb 3 12:08:25 localhost yum[5698]: Installed:
lm_sensors-libs-3.3.1-1.fc16.i686
Feb 3 12:08:25 localhost systemd[1]: Reloading.
Feb 3 12:08:25 localhost yum[5698]: Installed:
sysstat-10.0.2-2.fc16.i686
```

Para verificar a instalação, você pode visualizar o `/var/log/yum.log`, como mostrado no exemplo a seguir. Você pode ver que o `yum.log` verifica se a instalação ocorreu. Se, por alguma razão, você ou alguém na equipe de administrador não instalou esse pacote, uma investigação mais aprofundada deve ser feita.

```
# tail -3 /var/log/yum.log
Feb 03 11:08:51 Installed: 2:nmap-5.51-1.fc16.i686
Feb 03 12:08:25 Installed: lm_sensors-libs-3.3.1-1.fc16.i686
Feb 03 12:08:25 Installed: sysstat-10.0.2-2.fc16.i686
```

Você deve ter em mente as seguintes importantes questões de segurança relativas ao gerenciamento dos arquivos de log:

Reveja o arquivo de configuração do daemon `rsyslog`, `/etc/rsyslog.conf` para garantir que as informações desejadas são monitoradas (ver Capítulo 13).

- Monitore quaisquer modificações em `/etc/rsyslog.conf`.
- Revise `/etc/rsyslog.conf` para garantir que as tentativas de login são enviadas para `/var/log/secure` pela configuração `authpriv* /var/log/secure`. Para o Ubuntu, as tentativas de login são enviadas para `/var/log/auth.log` pela configuração `auth,authpriv.* /var/log/auth.log`.
- Crie e acompanhe os cronogramas de revisão de log para “olhos humanos”. Por exemplo, pode ser importante visualizar diariamente quem usa o comando `sudo`, que é registrado no arquivo `/var/log/secure`.
- Crie alertas de log. Por exemplo, um aumento repentino no tamanho de um arquivo de log, como `/var/log/btmp`, é uma indicação de um problema potencialmente malicioso.
- Mantenha a integridade do arquivo log.
 - Revise as permissões de arquivo de log.
 - Revise o cronograma de rotação de log no arquivo de configuração `logrotate`, `/etc/logrotate.conf`.
 - Inclua os arquivos de log nos planos de backup e retenção de dados.

- Implemente as melhores práticas, como limpar arquivos de log adequadamente para não perder as configurações de permissão de arquivo. Para limpar um arquivo de log sem perder suas configurações de permissão use o símbolo de redirecionamento da seguinte forma:
> `nomeDoArquivoDeLog`.

Arquivos de log são obviamente importantes na fase de monitoramento do Ciclo de Vida do Processo de Segurança. Outro item importante a monitorar é contas de usuário.

Monitorando contas de usuário

Contas de usuário são frequentemente utilizadas em ataques maliciosos em um sistema por meio de acesso não autorizado a uma conta atual, criação de novas contas falsas ou deixando para trás uma conta para acessar mais tarde. Para evitar esses problemas de segurança, acompanhar as contas de usuário precisa fazer parte da fase de monitoramento.

Detectando novas contas e privilégios falsificados

Contas criadas que não passam pela autorização apropriada devem ser consideradas falsas. Além disso, modificar uma conta de qualquer forma que fornece um diferente número de User Identification (UID) não autorizado ou adiciona associações de grupo não autorizadas é uma maneira de escalada de direitos. Preste atenção aos arquivos `/etc/passwd` e `/etc/group` para monitorar essas possíveis violações.

Para ajudá-lo a monitorar os arquivos `/etc/passwd` e `/etc/group`, use o daemon de auditoria. O daemon de auditoria é uma ferramenta de auditoria extremamente poderosa que permite selecionar os eventos do sistema para monitorar e gravá-los, e fornece capacidades de criação de relatórios.

Para começar a auditar os arquivos `/etc/passwd` e `/etc/group`, você precisa usar o comando `auditctl`. No mínimo são necessárias duas opções para iniciar esse processo:

- **`-w nome_do_arquivo`** — Ativa a monitoração do arquivo *nome_do_arquivo*. O daemon de auditoria irá monitorar o arquivo por seu número de inode. Um *número inode* é uma estrutura de dados que contém informações sobre um arquivo, incluindo sua localização.
- **`-p gatilho(s)`** — Se um desses tipos de acesso ocorrer (*r=read*, *w=write*, *x=execute*, *a=attribute change*) para *nome_do_arquivo*, essa opção dispara um registro de auditoria.

No código a seguir, a monitoração do arquivo `/etc/passwd` foi ativada usando o comando `auditctl`. O daemon de auditoria irá monitorar o acesso, o que consiste em qualquer leitura, gravação ou alteração no atributo de arquivo:

```
# auditctl -w /etc/passwd -p rwa
```

Nota

Depois de iniciar uma auditoria de arquivo, talvez você queira desativá-lo em algum momento. Para desativar uma auditoria, use o comando, `auditctl -W nomeDoArquivo -p gatilho(s)`.

Para ver uma lista dos arquivos atuais auditados e suas configurações de monitoramento, digite `auditctl -l` na linha de comando.

Para analisar os logs de auditoria, use o comando `ausearch` do daemon de auditoria. A única opção necessária aqui é a `-f`, que especifica os registros que você quer ver no log de auditoria. Abaixo, está um exemplo das informações de auditoria de `/etc/passwd`:

```
# ausearch -f /etc/passwd
time->Fri Feb 3 04:27:01 2014
type=PATH msg=audit(1328261221.365:572):
item=0 name="/etc/passwd" inode=170549
dev=fd:01 mode=0100644 ouid=0 ogid=0
rdev=00:00 obj=system_u:object_r:etc_t:s0
type=CWD msg=audit(1328261221.365:572): cwd="/"
...
time->Fri Feb 3 04:27:14 2014
type=PATH msg=audit(1328261234.558:574):
item=0 name="/etc/passwd" inode=170549
dev=fd:01 mode=0100644 ouid=0 ogid=0
rdev=00:00 obj=system_u:object_r:etc_t:s0
type=CWD msg=audit(1328261234.558:574):
cwd="/home/johndoe"
type=SYSCALL msg=audit(1328261234.558:574):
arch=40000003 syscall=5 success=yes exit=3
a0=3b22d9 a1=80000 a2=1b6 a3=0 items=1 ppid=3891
pid=21696 auid=1000 uid=1000 gid=1000 euid=1000
suid=1000 fsuid=1000 egid=1000 sgid=1000 fsgid=1000
tty=pts1 ses=2 comm="vi" exe="/bin/vi"
  subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023"
```

Há muitas informações para revisar. Alguns itens irão ajudá-lo a ver qual evento de auditoria aconteceu para acionar o registro na parte inferior.

- `time` — A data/hora da atividade
- `name` — O nome do arquivo, `/etc/passwd`, sendo monitorado
- `inode` — O número de inode de `/etc/passwd` nesse sistema de arquivos
- `uid` — O ID de usuário, 100, do usuário executando o programa
- `exe` — O programa, `/bin/vi`, usado no arquivo `/etc/passwd`

Para determinar a conta de usuário à qual é atribuído o UID de 100, examine o arquivo `/etc/passwd`. Neste caso, a UID de 100 pertence ao usuário `johndoe`. Assim, a partir do registro do evento de auditoria apresentado acima, você pode determinar que a conta `johndoe` tentou

usar o editor `vi` no arquivo `/etc/passwd`. É duvidoso que isso tenha sido uma ação inocente e exige mais investigação.

ta

mando `ausearch` não retornará nada se nenhum evento de monitoração em um arquivo foi disparado.

O daemon `audit` e suas ferramentas associadas são extremamente ricos. Para entender mais sobre isso, examine nas páginas `man` os seguintes utilitários de daemons de auditoria e arquivos de configuração:

- **auditd** — O daemon de auditoria
- **auditd.conf** — Arquivo de configuração do daemon de auditoria
- **auditctl** — Controla o sistema de auditoria
- **audit.rules** — Regras de configuração carregadas na inicialização
- **ausearch** — Pesquisa os itens especificados nos logs de auditoria
- **aureport** — Criador de relatório para os logs de auditoria
- **auditpd** — Envia informações de auditoria para outros programas

O daemon de auditoria é uma maneira de monitorar os arquivos importantes. Você deve revisar os arquivos de conta e grupo regularmente também com um “olho humano” para ver se algo parece irregular.

Arquivos importantes, como `/etc/passwd`, precisam ser monitorados no que diz respeito à criação de contas não autorizadas. Mas tão ruim quanto uma nova conta de usuário não autorizada é uma conta de usuário autorizada com uma senha ruim.

Detectando senhas ruins de contas

Mesmo com todos os seus bons esforços, senhas ruins serão utilizadas. Portanto, você precisa monitorar as senhas das contas dos usuários para garantir que elas são fortes o suficiente para resistir a um ataque.

Uma ferramenta sólida de monitoramento de senha que você pode usar é a mesma que usuários mal-intencionados usam para quebrar senhas de contas, John the Ripper. John the Ripper é uma ferramenta livremente disponível e de código-fonte aberto que pode ser usada na linha de comando do Linux. Ela não é instalada por padrão. Para uma distribuição do Fedora, você precisará emitir o comando `yum install john` para instalá-la.

ca

instalar John the Ripper no Ubuntu, use o comando `sudo apt-get install john`.

Para poder usar John the Ripper a fim de testar senhas de usuário, você deve primeiro extrair os nomes e senhas das contas usando o comando `unshadow`. Essas informações precisam ser redirecionadas para

um arquivo para ser usado por `john`, como mostrado aqui:

```
# unshadow /etc/passwd /etc/shadow > password.file
```

Agora, edite o `password.file` usando seu editor de texto favorito para remover quaisquer contas sem senhas. Como é prudente limitar John the Ripper a testar apenas algumas poucas contas de cada vez, remova as contas que você não quer testar no momento.

Atenção

Os usuários `john` usam intensamente a CPU. Ele não define seu valor `nice` como 19 a fim de diminuir sua prioridade. Mas seria prudente executá-lo em um sistema que seja de produção ou fora do horário de pico e apenas algumas contas de cada vez.

Agora, use o comando `john` para tentar quebrar as senhas. Para executar `john` contra o arquivo de senha criado, emita o comando `john nomeDoArquivo`. No trecho de código a seguir, você pode ver a saída da execução de `john` contra o `password.file` de exemplo. Para fins de demonstração, apenas uma conta foi deixada no arquivo de exemplo. E a conta, Samantha, recebeu a senha ruim `password`. Você pode ver como foi rápido para John the Ripper quebrar a senha.

```
# john password.file
Loaded 1 password hash (generic crypt(3) [?/32])
password      (Samantha)
guesses: 1 time: 0:00:00:44 100% (2) c/s: 20.87
trying: 12345 - missy
Use the "--show" option to display all of the
cracked passwords reliably
```

Para demonstrar como senhas fortes são vitais, considere o que acontece quando a senha da conta de Samantha é alterada de `password` para `Password1234`. Embora `Password1234` ainda seja uma senha fraca, leva mais de 7 dias de tempo de CPU para quebrá-la. No código a seguir, `john` foi finalmente abortado para acabar com a tentativa de quebra de senha.

```
# passwd Samantha
Changing password for user Samantha.
...
# john password.file
Loaded 1 password hash (generic crypt(3) [?/32])
...
time: 0:07:21:55 (3) c/s: 119 trying: tth675 - tth787
Session aborted
```

Assim que as tentativas de quebra de senhas forem concluídas, o `password.file` deve ser removido do sistema. Para saber mais sobre John the Ripper, visite www.openwall.com/john.

Monitorando o sistema de arquivos

Programas mal-intencionados muitas vezes modificam os arquivos. Eles também podem tentar cobrir seus rastros fingindo ser arquivos e programas comuns. Mas há maneiras de descobri-los por meio de várias táticas de monitoramento discutidas nesta seção.

Verificando pacotes de software

Normalmente, se você instalar um pacote de software a partir de um repositório padrão ou baixar um pacote de um site respeitável, você não terá nenhum problema. Mas sempre é bom verificar seus pacotes de software instalados para ver se eles foram comprometidos. O comando para fazer isso é `rpm -V nome_do_pacote`.

Ao verificar o software, as informações dos arquivos dos pacotes instalados são comparadas com os metadados do pacote (ver Capítulo 10) no banco de dados `rpm`. Se nenhum problema for encontrado, o comando `rpm -V` não retornará nada. Mas se houver discrepâncias, você terá uma lista codificada. A Tabela 22.6 mostra os códigos utilizados e uma descrição da discrepância.

TABELA 22.6 Discrepâncias na Verificação de Pacotes

ligo	Discrepância
	Tamanho do arquivo
	Tipos e permissões de arquivo
	Soma de verificação MD5
	Números primários e secundários do arquivo do dispositivo
	Links simbólicos
	Propriedade de usuário
	Propriedade de grupo
	Número de vezes que os arquivos foram modificados (mtime)
	Outros pacotes instalados dos quais esse pacote depende (conhecidos com capacidades)

Na lista parcial a seguir, todos os pacotes instalados são verificados. Você pode ver que os códigos 5, S e T retornaram, indicando alguns potenciais problemas.

```
# rpm -qaV
5S.T..... c /etc/hba.conf
...
...T..... /lib/modules/3.2.1-3.fc16.i686/modules.devname
...T..... /lib/modules/3.2.1-3.fc16.i686/modules.softdep
```

Você não tem de verificar todos os pacotes de uma só vez, pode verificar apenas um pacote de cada vez. Por exemplo, se você quiser verificar o pacote `nmap`, simplesmente digite `rpm -V nmap`.

verificar os pacotes no Ubuntu, você precisa do utilitário `debsums`. Ele não é instalado por padrão. Para instalar `debsums`, use o comando `sudo apt-get install debsums`. Para verificar todos os pacotes instalados, utilize o comando `debsums -a`. Para verificar um pacote individual, digite `debsums nome_do_pacote`.

Verificando o sistema de arquivos

A menos que você tenha atualizado recentemente seu sistema, os arquivos binários não devem ter sido modificados por nenhuma razão. Para verificar modificações em arquivos binários, use a data/hora de modificação dos arquivos, ou `mtime`. O arquivo `mtime` é a data/hora em que o conteúdo de um arquivo foi modificado pela última vez. Além disso, você pode monitorar a data/hora de criação ou alteração do arquivo ou `ctime`.

Se suspeitar de atividades maliciosas, você pode rapidamente verificar o sistema de arquivos para ver se quaisquer arquivos binários foram modificados ou alterados hoje (ou ontem, dependendo de quando você acha que a invasão ocorreu). Para fazer essa verificação, use o comando `find`.

No exemplo a seguir, uma verificação é feita no diretório `/sbin`. Para ver se quaisquer arquivos binários foram modificados há menos de 24 horas, o comando `find /sbin -mtime -1` é usado. No exemplo, há vários arquivos exibidos, mostrando que eles foram modificados recentemente. Isso indica que uma atividade maliciosa está ocorrendo no sistema. Para investigar ainda mais, revise a data/hora da última modificação de cada arquivo individual usando o comando `stat nome_do_arquivo`, como mostrado aqui:

```
# find /sbin -mtime -1
/sbin
/sbin/init
/sbin/reboot
/sbin/halt
#
# stat /sbin/init
  File: '/sbin/init' -> '../bin/systemd'
  Size: 14 Blocks: 0 IO Block: 4096 symbolic link
 Device: fd01h/64769d   Inode: 9551       Links: 1
Access: (0777/lrwxrwxrwx)
Uid: (  0/   root)   Gid: (  0/   root)
Context: system_u:object_r:bin_t:s0
Access: 2014-02-03 03:34:57.276589176 -0500
Modify: 2014-02-02 23:40:39.139872288 -0500
Change: 2014-02-02 23:40:39.140872415 -0500
 Birth: -
```

Você pode criar um banco de dados de todos `mtimes` e `ctimes` originais do arquivo binário e então executar um script para encontrar os `mtimes` e `ctimes` atuais, compará-los com o banco de dados e observar quaisquer discrepâncias. No entanto, esse tipo de programa já foi criado e funciona bem. Ele é chamado Sistema de Detecção de Intrusos (Intrusion Detection System) e é discutido mais adiante, neste capítulo.

Há várias verificações do sistema de arquivos que você precisa executar regularmente. Os arquivos favoritos ou as configurações de arquivo favoritas dos invasores maliciosos estão listados na Tabela 22.7. A tabela também lista os comandos para realizar as verificações e por que o arquivo ou a configuração dele são potencialmente problemáticos.

TABELA 22.7 Verificações Adicionais do Sistema de Arquivos

Uivo ou configuração	Comando de varredura	Problema com arquivo ou configuração
missão SetUID	<code>find / -perm -4000</code>	Permite que qualquer pessoa torne-se temporariamente o proprietário do arquivo enquanto ele está em execução na memória.
missão SetGID	<code>find / -perm -2000</code>	Permite que qualquer pessoa torne-se temporariamente um membro do grupo do arquivo enquanto ele está em execução na memória.
quívios rhost	<code>find /home -name .rhosts</code>	Permite que um sistema confie totalmente em outro sistema. Não deve estar em diretórios /home.
quívios sem proprietário	<code>find / -nouser</code>	Indica um arquivo comprometido.
quívios sem grupo	<code>find / -nogroup</code>	Indica um arquivo comprometido.

Essas verificações do sistema de arquivos ajudarão a monitorar o que acontece no sistema e ajudar a detectar ataques maliciosos. Mas há outros tipos de ataques que podem ocorrer nos seus arquivos, incluindo vírus e rootkits.

Detectando vírus e rootkits

Duas ferramentas populares de ataque malicioso são vírus e rootkits, porque eles permanecem ocultos durante a execução das atividades maliciosas. Ambas as ferramentas precisam ser monitoradas nos sistemas Linux.

Monitorando vírus

Um *vírus de computador* é um software malicioso que pode se anexar aos softwares já instalados no sistema e tem a capacidade de se espalhar através de mídias ou redes. É um equívoco a ideia de que não há vírus em Linux. Os criadores mal-intencionados de vírus costumam focalizar os sistemas operacionais desktop mais populares, como o Windows. Mas isso não significa que não sejam criados vírus para sistemas Linux.

Ainda mais importante, os sistemas Linux são muitas vezes utilizados para lidar com serviços, como servidores de email, para sistemas desktop Windows. Portanto, os sistemas Linux utilizados para esses propósitos também precisam ser verificados quanto a vírus para Windows.

O software antivírus verifica os arquivos usando assinaturas de vírus. Uma *assinatura de vírus* é um hash criado a partir do código binário de um vírus. O hash identificará positivamente esse vírus. Os programas antivírus têm um banco de dados de assinaturas de vírus que é usado para comparar com os arquivos para ver se há uma correspondência de assinatura. Dependendo do número de novas ameaças, um banco de dados de assinaturas de vírus pode ser atualizado frequentemente para fornecer proteção contra essas novas ameaças.

Um bom software antivírus para seu sistema Linux, que é de código-fonte aberto e livre, é o ClamAV. Para instalar o ClamAV em um sistema Fedora ou RHEL, digite o comando `yum install clamav`. Você pode descobrir mais sobre o ClamAV em www.clamav.net, onde há documentação sobre como configurar e executar o software antivírus.

ca

pode revisar os pacotes disponíveis para instalação do Ubuntu digitando o comando `apt-cache search clamav`. Há dois pacotes diferentes disponíveis para o Ubuntu; portanto, informe-se no site do ClamAV antes de escolher um.

Monitorando rootkits

Um *rootkit* é um pouco mais insidioso do que um vírus. *Ele* é um programa malicioso que:

- Permanece oculto, muitas vezes substituindo comandos ou programas do sistema
- Mantém acesso de alto nível a um sistema
- É capaz de enganar os softwares criados para localizá-lo

O propósito de um rootkit é obter e manter acesso de nível de root a um sistema. O termo foi criado juntando “root”, que significa que ele precisa ter acesso de administrador, e “kit”, que significa que normalmente há vários programas que operam em conjunto.

Um detector de rootkit que pode ser usado em um sistema Linux é `chkrootkit`. Para instalar `chkrootkit` em um sistema Fedora ou RHEL, emita o comando `yum install chkrootkit`. Para instalar `chkrootkit` em um sistema Ubuntu, use o comando `sudo apt-get install chkrootkit`.

ca

o melhor é usar um Live CD ou pen drive para executar `chkrootkit` de modo que os resultados não sejam mascarados por um rootkit. O Fedora Security Spin tem o `chkrootkit` no Live CD. Você pode obter essa distribuição em <http://spins.fedoraproject.org/security>.

Encontrar um rootkit com `chkrootkit` é simples. Depois de instalar o pacote ou inicializar o Live CD, digite `chkrootkit` na linha de comando. Ele pesquisará toda a estrutura de arquivos indicando quaisquer arquivos infectados.

O código a seguir mostra uma execução de `chkrootkit` em um sistema infectado. O comando `grep` foi usado para pesquisar a palavra-chave `INFECTED`. Observe que muitos dos arquivos listados como “infectados” são arquivos de comando do shell `bash`. Isso é típico de um rootkit.

```
# chkrootkit | grep INFECTED
Checking `du'... INFECTED
Checking `find'... INFECTED
Checking `ls'... INFECTED
Checking `lsof'... INFECTED
Checking `pstree'... INFECTED
Searching for Suckit rootkit... Warning: /sbin/init INFECTED
```

Na última linha do código anterior de `chkrootkit` está uma indicação de que o sistema foi infectado com o rootkit Suckit. Mas, na verdade, ele não está infectado com esse rootkit. Ao executar utilitários, como programas antivírus e software de detecção de rootkit, muitas vezes você vai obter um número de falsos positivos. Um *falso positivo* é uma indicação de um vírus, rootkit ou outra atividade maliciosa que realmente não existe. Nesse caso em particular, esse falso positivo é causado por um erro conhecido.

O utilitário `chkrootkit` deve ser agendado para execuções regulares e, obviamente, deve ser executado sempre que houver suspeitas de uma infecção por rootkit. Para encontrar mais informações sobre `chkrootkit`, acesse <http://chkrootkit.org>.

Detectando uma invasão

Durante a fase de Monitoramento do Ciclo de Vida do Processo de Segurança, há várias coisas a monitorar. Um *Sistema de Detecção de Intrusos (Intrusion Detection System, IDS)* — um pacote de software que monitora atividades de um sistema (ou sua rede) quanto a potenciais atividades maliciosas e informa essas atividades — pode ajudá-lo nesse trabalho. O sistema de detecção de intrusos está intimamente relacionado com um pacote de software que impedirá uma invasão, chamado software de *Prevenção de Intrusos*. Alguns desses pacotes são distribuídos juntos para fornecer *Detecção e Prevenção de Intrusos*.

Vários pacotes de software do tipo sistema de detecção de intrusos estão disponíveis para sistemas Linux. Alguns dos utilitários mais populares estão listados na Tabela 22.8. Você deve saber que o `tripwire` não mais é de código-fonte aberto. Mas o código original do `tripwire` ainda está disponível. Consulte mais detalhes no site do `tripwire` listado na Tabela 22.8.

TABELA 22.8 Sistemas Populares de Detecção de Intrusos do Linux

Nome do sistema	Instalação	Site Web
aide	<code>yum install aide</code> <code>apt-get install aide</code>	http://aide.sourceforge.net
snort	Pacotes rpm ou tarball do site	http://snort.org
tripwire	<code>yum install tripwire</code> <code>apt-get install tripwire</code>	http://tripwire.org

O IDS Advanced Intrusion Detection Environment (`aide`), isto é, Ambiente Avançado de Detecção de Invasão, usa um método de comparação para detectar invasões. Quando você era criança, talvez tenha brincado com o jogo de comparar duas imagens e encontrar aquilo que era diferente entre elas. O utilitário `aide` usa um método semelhante. Um banco de dados da “primeira imagem” é criado. Em algum momento, mais tarde, outro banco de dados da “segunda imagem” é criado e o `aide` compara os dois bancos de dados e informa qual é diferente.

Para começar, você precisa tirar a “primeira foto”. O melhor momento para criar essa imagem é quando o sistema acabou de ser instalado. O comando para criar o banco de dados inicial é `aide -i` e a execução dele é demorada. Parte da sua saída está a seguir. Observe que `aide` informa onde ele cria o

banco de dados inicial da “primeira imagem”.

```
# aide -i
```

```
AIDE, version 0.15.1
```

```
### AIDE database at /var/lib/aide/aide.db.new.gz initialized.
```

O próximo passo é mover o banco de dados inicial da “primeira imagem” para um novo local. Isso evita que o banco de dados original seja sobrescrito. Além disso, a comparação só funcionará se o banco de dados for movido. O comando para movê-lo para a nova localização e lhe dar um novo nome é o seguinte:

```
# cp /var/lib/aide/aide.db.new.gz /var/lib/aide/aide.db.gz
```

Agora, você precisa criar um novo banco de dados, a “segunda imagem”, e compará-lo com o banco de dados original, a “primeira imagem”. A opção de verificação no comando `aide`, `-c`, criará um novo banco de dados e executará uma comparação com o banco de dados antigo. A saída mostrada a seguir ilustra essa comparação sendo feito e o comando `aide` informando alguns problemas.

```
# aide -C
```

```
...
```

```
-----  
Detailed information about changes:  
-----
```

```
File: /bin/find
```

```
Size : 189736 , 4620
```

```
Ctime : 2012-02-10 13:00:44 , 2012-02-11 03:05:52
```

```
MD5 : <NONE> , rUJj8NtNalv4nmV5zfoOjg==
```

```
RMD160 : <NONE> , 0CwkiYhqNnfwPUPM12HdKuUSFUE=
```

```
SHA256 : <NONE> , jg60Soawj4S/UZXm5h4aEGJ+xZgGwCmN
```

```
File: /bin/ls
```

```
Size : 112704 , 6122
```

```
Ctime : 2012-02-10 13:04:57 , 2012-02-11 03:05:52
```

```
MD5 : POeOop46MvRx9qfEoYTXOQ== , IShMBpbSOY8axhwlKj8Wdw==
```

```
RMD160 : N3V3Joe5Vo+cOSSnedf9PCDXyKI= ,
```

```
e0ZneB7CrWHV42hAEgT21wrVfP4=
```

```
SHA256 : vu0Fe6FUgoAyNgIxYghOo6+SxR/zxS1s ,
```

```
Z6nEMMBQyYm8486yFSIbKBuMUi/+jrUi
```

```
...
```

```
File: /bin/ps
```

```
Size : 76684 , 4828
```

```
Ctime : 2012-02-10 13:05:45 , 2012-02-11 03:05:52
```

```
MD5 : 1pCVAWbpeXINiBQWSUEJfQ== , 4ElJhyWkyMtm24vNLya6CA==
```

```
RMD160 : xwICWNtQH242jHsH2E8rV5kgSkU= ,
```

```
AZ1I2QN1KrWH45i3/V54H+1QQZk=
```

```
SHA256 : ffUDesbfx3YsLDhD0bLTW0c6nykc3m0 ,
```

```
w1qXvGWPFzFfir5yxN+n6t3eOWw1TtNC/
```

```
...
```

```
File: /usr/bin/du
```

```
Size : 104224 , 4619
```

```
Ctime : 2012-02-10 13:04:58 , 2012-02-11 03:05:53
```

```
MD5 : 5DUMKWj6LodWj4C0xfPBIw== , nzn7vrwfBawAeL8nkayICg==
```

```
RMD160 : Z1bm0f/bUWRLgilB5nVjhanuX9Q= ,
```

```
2e5S001BWqLq4Tnac4b6QIXRCwY=
```

```
SHA256 : P/jVAKr/SO0epBBxvGP900nLXrRY9tnw ,
```

HhTqWgDyIkUDxA1X232ijmQ/OMA/kRgl

```
File: /usr/bin/pstree
Size : 20296 , 7030
Ctime : 2012-02-10 13:02:18 , 2012-02-11 03:05:53
MD5 : <NONE> , ry/MUZ7XvU4L2QfWJ4GXxg==
RMD160 : <NONE> , tFZer6As9EoOi58K7/LgmeiExjU=
SHA256 : <NONE> , iAsMkqNShagD4qe7dL/EwcgKTRzvKRSe
...
```

Os arquivos listados pela verificação *aide* nesse exemplo estão infectados. Mas *aide* também pode exibir muitos falsos positivos.

O local onde os bancos de dados *aide* são criados, as comparações que são feitas e várias outras configurações são tratadas no arquivo `/etc/aide.conf`. Eis uma exibição parcial do arquivo. Você pode ver os nomes do arquivo de banco de dados e os diretórios dos arquivos de log configurados aqui:

```
# cat /etc/aide.conf
# Example configuration file for AIDE.

@@define DBDIR /var/lib/aide
@@define LOGDIR /var/log/aide

# The location of the database to be read.
database=file:@@{DBDIR}/aide.db.gz

# The location of the database to be written.
#database_out=sql:host:port:database:login_name:passwd:table
#database_out=file:aide.db.new
database_out=file:@@{DBDIR}/aide.db.new.gz
```

Um IDS pode ser uma grande ajuda para monitorar o sistema. Pelo menos uma das ferramentas IDS deve fazer parte da fase de Monitoramento do Ciclo de Vida do Processo de Segurança.

Trabalhando na fase de Auditoria/Revisão

A fase final do Ciclo de Vida do Processo de Segurança é a fase de Auditoria/Revisão. Não só é importante que a segurança que está sendo implementada siga políticas e procedimentos, mas que as políticas e procedimentos estejam corretos.

Há dois termos importantes para serem reconhecidos nessa fase. Uma *revisão de conformidade* é uma auditoria do ambiente geral do sistema de computador para garantir que as políticas e procedimentos determinados na fase de planejamento do Ciclo de Vida do Processo de Segurança estão sendo realizados corretamente. Os resultados da auditoria dessa revisão serviriam como feedback na fase de implementação. Uma *revisão de segurança* é uma auditoria das atuais políticas e procedimentos para assegurar que seguem as melhores práticas de segurança aceitas. Os resultados da auditoria dessa revisão serviriam como feedback para a fase de planejamento.

Realizando revisões de conformidade

Semelhantes às auditorias em outros campos, como contabilidade, as auditorias podem ser realizadas internamente ou por pessoal externo. Essas revisões podem ser tão simples como alguém sentando e comparando a segurança implementada com a matriz de controle de acesso e as políticas estabelecidas. Contudo, mais popular é realizar auditorias utilizando testes de penetração.

Teste de penetração é um método de avaliação utilizado para testar a segurança de um sistema de computador simulando ataques maliciosos. Ele também é chamado *pen testing* e *hacking ético*. Você não mais precisa reunir ferramentas e hackers da vizinhança local para ajudar a realizar esses testes. A seguir, estão as distribuições do Linux que você pode usar para realizar testes de penetração bem completos:

- BackTrack (www.backtrack-linux.org)
 - Distribuição do Linux criada especificamente para testes de penetração
 - Pode ser usado a partir de um Live DVD ou um pen drive
 - Treinamento sobre o uso do BackTrack é oferecido por www.offensive-security.com
- Fedora Security Spin (<http://spins.fedoraproject.org/security>)
 - Também chamado “Fedora Security Lab”
- Parte da distribuição do Fedora Linux
- Fornece um ambiente de teste para trabalhar a auditoria de segurança
- Pode ser usado a partir de um pen drive

Embora o teste de penetração seja divertido, para uma revisão completa de conformidade, um pouco mais é necessário. Você também deve usar listas de verificação dos sites do setor de segurança e a matriz de controle de acesso da fase de Planejamento do Ciclo de Vida do Processo de Segurança para auditar seus servidores.

Realizando revisões de segurança

Realizar uma revisão de segurança exige que você conheça as melhores práticas de segurança atuais. Há várias maneiras de se manter informado sobre as melhores práticas de segurança. A seguir, é fornecida uma breve lista das organizações que podem ajudá-lo.

- United States Computer Emergency Readiness Team (CERT)
 - URL: www.us-cert.gov
 - Oferece o National Cyber Alert System
 - Oferece feeds RSS sobre as ameaças de segurança mais recentes
- The SANS Institute

- URL: www.sans.org/security-resources
- Oferece boletins da Computer Security Research
- Oferece feeds RSS sobre as ameaças de segurança mais recentes
- Gibson Research Corporation
 - URL: www.grc.com
 - Oferece o netcast de segurança Security Now!

As informações fornecidas por esses sites irão ajudá-lo a criar políticas e procedimentos mais fortes. Dada a rapidez com que as melhores práticas de segurança mudam, é prudente realizar frequentes revisões de segurança, dependendo das necessidades de segurança da organização.

Agora, você entende muito mais sobre a segurança básica no Linux. A parte difícil é realmente colocar todos esses conceitos em prática.

Resumo

Entender o Ciclo de Vida do Processo de Segurança irá ajudá-lo bastante a proteger os valiosos ativos informacionais da sua organização. Cada área do ciclo de vida, planejamento, implementação, monitoramento e auditoria/revisão, tem informações e atividades importantes. Embora o ciclo de vida talvez não avance de forma ordenada na sua empresa, ainda é importante implementar o Ciclo de Vida do Processo de Segurança e suas fases.

Na fase de Planejamento, escolhe-se o modelo de controle de acesso. As três opções são DAC, MAC e RBAC. DAC é o modelo padrão em todas as distribuições Linux. O modelo escolhido aqui influencia significativamente as demais fases do ciclo de vida. A matriz de controle de acesso é criada nessa fase. Essa matriz é usada em todo o processo, como uma lista de verificação de segurança.

Na fase de implementação, na verdade você implementa o modelo de controle de acesso escolhido usando as várias ferramentas disponíveis no Linux. Essa é a fase em que a maioria das empresas tipicamente começa o Ciclo de Vida do Processo de Segurança. Atividades como gerenciamento de contas de usuários, segurança de senhas e gerenciamento de software e serviços são incluídas nessa fase.

A fase de monitoramento pode ser mais demorada. Nela, você observa os arquivos de log do sistema, verifica intrusões maliciosas, monitora o sistema de arquivos etc. Há muitas ferramentas disponíveis no Linux para ajudá-lo nessa fase.

Por fim, a fase de auditoria/revisão completa o ciclo de vida. Nessa fase, revisões de conformidade e segurança são conduzidas. Essas auditorias ajudam a garantir que seu sistema Linux está seguro e que as políticas e práticas adequadas de segurança estão corretas.

Você completou a primeira etapa da coleta de conhecimento dos princípios e procedimentos básicos de segurança. Não é suficiente somente conhecer os princípios básicos, você precisa adicionar ferramentas avançadas de segurança em Linux à sua caixa de ferramentas de segurança. No próximo capítulo, discutiremos os temas avançados de segurança dos módulos de criptografia e autenticação.

Exercícios

Consulte o material neste capítulo para completar as tarefas a seguir. Se você empacar, soluções para as tarefas são mostradas no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa). Tente resolver cada um dos exercícios antes de consultar as respostas. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux).

1. A partir de um arquivo de log, crie uma lista dos serviços que foram iniciados no seu sistema durante inicialização do sistema.
2. Liste as permissões no arquivo de senhas do sistema e determine se elas são adequadas.
3. Determine o envelhecimento da senha da sua conta e se ela vai expirar usando um único comando.
4. Inicie a auditoria de gravações no arquivo `/etc/shadow` com o daemon `auditd` e então verifique as configurações de auditoria
5. Crie um relatório a partir do daemon `auditd` no arquivo `/etc/shadow` e desative a auditoria nesse arquivo.
6. Verifique um pacote de software instalado no seu sistema comparando-o com os metadados do pacote.
7. Você suspeita que hoje houve um ataque malicioso ao seu sistema e os arquivos binários importantes foram modificados. Que comando você deve usar para encontrar esses arquivos modificados?
8. Instale e execute `chkrootkit` para ver se o ataque malicioso do exercício acima instalou um rootkit.
9. Localize arquivos com a permissão SetUID configurada.
10. Localize arquivos com a permissão SetGID configurada.

Entendendo a segurança avançada do Linux

NESTE CAPÍTULO

Entendendo hashing e criptografia

Verificando a integridade do arquivo

Criptografando arquivos, diretórios e sistemas de arquivos

Entendendo os módulos de autenticação conectáveis

Gerenciando a segurança do Linux com PAM

Devido às crescentes ameaças e sua constante mudança, implementar apenas a segurança básica de computador não mais é suficiente. À medida que usuários maliciosos ganham acesso e conhecimento sobre ferramentas avançadas, um administrador de sistemas Linux também deve ganhar. Entender os tópicos avançados e as ferramentas de segurança de computadores precisa ser parte da sua preparação.

Neste capítulo, você aprenderá os conceitos básicos de criptografia, como cifras e encriptação. Você também aprenderá como o utilitário módulo de autenticação pode simplificar as funções administrativas, mesmo sendo um tema avançado de segurança.

Implementando a segurança do Linux com criptografia

Usar criptografia aumenta a segurança do sistema Linux e as comunicações em rede. *Criptografia* é a ciência de ocultar informações. Ela tem uma longa e rica história que remonta há muito tempo antes de os computadores começarem a ser utilizados. Por causa do uso pesado de algoritmos matemáticos, a criptografia migrou facilmente para o ambiente de computador. O Linux vem com muitas ferramentas de criptografia prontas para uso.

Para entender os conceitos criptográficos e as várias ferramentas do Linux, é importante conhecer alguns termos da criptografia.

- Texto simples — Texto que um humano ou uma máquina pode ler e compreender

- Texto cifrado — Texto que um humano ou uma máquina não podem ler e compreender
- Criptografia — O processo de converter texto simples em texto cifrado usando um algoritmo (também conhecida como encriptação)
- Descritografia — O processo de converter texto cifrado em texto simples utilizando um algoritmo (também conhecida como deciptação)
- Cifra — O algoritmo usado para criptografar texto simples em texto cifrado e descritografar texto cifrado em texto simples
- Cifra de bloco — Uma cifra que divide os dados em blocos antes da criptografia
- **Cifra de fluxo** — Uma cifra que criptografa os dados sem dividi-los
- **Chave** — Os dados exigidos pela cifra para criptografar ou descritografar com sucesso os dados

Os pais muitas vezes usam uma forma de criptografia. Eles soletram as palavras em vez de falá-las. Um pai poderia pegar a palavra em texto simples “doce” e transformá-la em texto cifrado dizendo à mãe “D-O-C-E”. A mãe decifra a palavra utilizando a mesma cifra soletrada, e reconhece que a palavra é “doce”. Felizmente, não demora muito para que as crianças aprendam a decifrar por meio da cifra soletrada.

Você deve ter observado que o hash não foi incluído na lista de definição de criptografia anterior. O hashing precisa de atenção especial, porque muitas vezes é confundido com criptografia.

Entendendo o hashing

Hashing não é criptografia, mas uma forma de criptografia. Lembre-se do Capítulo 22 que *hashing* é um processo matemático unidirecional usado para criar um texto cifrado. Mas, ao contrário da criptografia, depois que um hash é criado, você não pode transformá-lo de volta ao texto simples original.

Para que um algoritmo de hashing seja utilizado na segurança de computadores, ele precisa estar *livre de colisão*, o que significa que o algoritmo de hashing não produz o mesmo hash para duas entradas totalmente diferentes. Cada entrada deve ter uma única saída hasheada. Assim, o *hashing criptográfico* é um processo matemático unidirecional que é livre de colisão.

Por padrão, a criptografia já está em uso em um sistema Linux. Por exemplo, o arquivo `/etc/shadow` contém as senhas hasheadas. O hashing é usado em sistemas Linux para:

- Senhas (Capítulo 22)
- Verificar arquivos
- Assinaturas digitais
- Assinaturas de vírus (Capítulo 22)

Um hash também é chamado de resumo da mensagem, soma de verificação, impressão digital ou assinatura. Um utilitário do Linux para produzir resumos de mensagem é o `md5sum`. No Capítulo 10, “Obtendo e gerenciando softwares”, você aprendeu a obter softwares para seu sistema Linux. Ao baixar um arquivo de software, você pode certificar-se de que o arquivo não foi corrompido no download. A

Figura 23.1 mostra o site para baixar o software de distribuição Linux Mint (também chamado ISO). A página web contém um número Message Digest 5 (MD5) que você pode usar para garantir que a imagem ISO que você baixou não foi corrompida durante o download.

A página Web Linux Mint ISO download fornece um número MD5.

Um hash é produzido a partir de um arquivo de software na localização original, usando o algoritmo de hash MD5. Os resultados do hash podem ser postados publicamente, como foi feito na Figura 23.1. Para garantir a integridade do arquivo de software baixado, você cria um hash MD5 do arquivo de software na sua localização e então compara os resultados do hash com os resultados postados do hash. Se eles corresponderem, o arquivo de software não foi corrompido durante o download.

Para criar o hash, execute o algoritmo de hashing no seu sistema usando `md5sum`. Os resultados do hash `md5sum` para o arquivo de software baixado são mostrados no código a seguir.

```
$ md5sum linuxmint-12-gnome-cd-nocodecs-32bit.iso
40562d26447207cb5111f94b93957a58
linuxmint-12-gnome-cd-nocodecs-32bit.iso
```

Você pode ver que o hash resultante *não* corresponde com aquele postado no site web na Figura 23.1. Isso significa que o arquivo ISO baixado não foi corrompido e está pronto para uso.

Atenção

Embora o hashing MD5 seja suficiente para garantir que um arquivo de software baixado não foi corrompido, o algoritmo não é livre de colisão. Portanto, ele não é mais considerado um verdadeiro hash criptográfico. Para obter um hash criptográfico real, você vai precisar usar um dos hashes criptográficos — SHA-2 ou SHA-3 — discutidos mais adiante, neste capítulo.

Você pode implementar ainda mais criptografia além de hashing no sistema Linux. Os utilitários do Linux para fazer isso são muito fáceis de usar. Mas primeiro você precisa entender alguns outros conceitos criptográficos subjacentes.

Entendendo criptografia/decriptografia

O principal uso da criptografia em um sistema Linux é codificar os dados para ocultá-los (criptografia) de olhos não autorizados e, então, decodificar os dados (decriptografia) para olhos autorizados. Em um sistema Linux, você pode criptografar:

- Arquivos individuais
- Partições e volumes
- Conexões a páginas web

- Conexões de rede
- Backups
- Arquivos Zip

Esses processos de criptografia/decriptografia usam algoritmos matemáticos especiais para realizar as respectivas tarefas. Os algoritmos são chamados cifras criptográficas.

Entendendo as cifras criptográficas

Uma das cifras originais, chamada cifra de César, foi criada e usada por Júlio César. Mas ela era muito fácil de quebrar. Hoje, há cifras muito mais seguras disponíveis. Entender como cada cifra funciona é importante, porque a força da cifra que você escolhe deve se relacionar diretamente com as necessidades de segurança dos seus dados. A Tabela 23.1 lista algumas cifras modernas.

TABELA 23.1 Cifras de Criptografia

Method	Descrição
AES (Advanced Encryption Standard), também chamado Rijndael	Criptografia simétrica Cifra de bloco, criptografa os dados em blocos de 128, 192 ou 256 bits usando uma chave de 128, 192 ou 256 bits para criptografar/descriptografar.
Twofish	Criptografia simétrica Cifra de bloco, criptografia de dados em blocos de 64 bits usando as mesmas chaves de 32 a 448 bits para criptografia/descriptografia.
CAST5	Criptografia simétrica Cifra de bloco, criptografa os dados em blocos de 64 bits usando a mesma chave de 128 bits para criptografar/descriptografar.
IDEA (Data Encryption Standard)	Não mais considerado seguro Criptografia simétrica. Cifra de bloco, criptografa os dados em blocos de 64 bits usando a mesma chave de 56 bits para criptografia/descriptografia.
DES	Cifra DES aprimorada Criptografia simétrica. Os dados são codificados até 48 vezes com 3 diferentes chaves de 56 bits antes de o processo de criptografia estar concluído.
Serpent	Criptografia simétrica Utiliza duas chaves derivadas de um algoritmo de logaritmo.
Point Curve Cryptosystems	Criptografia assimétrica Utiliza duas chaves derivadas de um algoritmo que contém dois pontos escolhidos aleatoriamente em uma curva elíptica.
Blowfish	Criptografia simétrica

Cifra de bloco, criptografa os dados em blocos de 64 bits usando a mesma chave de 128 bits para criptografia/descriptografia.

4 também chamado ArcFour ou ARC4	Cifra de fluxo, criptografa os dados em blocos de 64 bits usando um tamanho variável de chave para criptografia/descriptografia.
5	Criptografia simétrica Cifra de bloco, criptografa os dados em blocos de 32, 64 ou 128 bits usando as mesmas chaves de até 2048 bits para criptografia/descriptografia.
6	Criptografia simétrica O mesmo que RC5, mas um pouco mais rápido.
Advanced Encryption Standard (também chamado AES)	Criptografia simétrica Cifra de bloco, criptografia de dados em blocos de 128, 192 ou 256 bits usando uma chave de 128, 192 ou 256 bits para criptografar/descriptografar.
A	Criptografia assimétrica mais popular Utiliza duas chaves derivadas de um algoritmo que contém um múltiplo de dois números primos gerados aleatoriamente.

Entendendo chaves de cifra criptográfica

Cifras criptográficas exigem um dado, chamado chave, para completar o processo matemático de criptografia/descriptografia. A chave pode ser uma única chave ou um par de chaves.

Observe os diferentes tamanhos de chave de cifra listados na Tabela 23.1. O tamanho da chave está diretamente relacionado com a facilidade com que a cifra é quebrada. Quanto maior o tamanho da chave, menor a chance de quebrar a cifra. Por exemplo, o DES não mais é considerado seguro devido ao seu pequeno tamanho de chave de 56 bits. Mas uma cifra com um tamanho de chave de 256 bits é considerada segura porque levaria trilhões de anos para quebrar na força bruta uma cifra tão codificada.

Criptografia de chave simétrica

A *criptografia simétrica*, também chamada de chave secreta ou criptografia de chave privada, criptografa texto simples usando uma cifra codificada uma única vez. A mesma chave é necessária a fim de descriptografar os dados. A vantagem da criptografia de chave simétrica é a velocidade. A desvantagem é a necessidade de compartilhar a chave única quando os dados criptografados precisarem ser descriptografados por outra pessoa.

Um exemplo de criptografia de chave simétrica em um sistema Linux é o uso do utilitário OpenPGP, GNU Privacy Guard, `gpg`. O exemplo a seguir mostra o utilitário `gpg` usado para criptografar o arquivo `Secret.File`. Com a opção `-c`, o `gpg` criptografa o arquivo com uma chave simétrica. O arquivo original é mantido e um novo arquivo criptografado, `Secret.File.gpg`, é criado.

```
$ cat Secret.File
This is my secret message.
$
$ gpg -c Secret.File
Enter passphrase:
Repeat passphrase:
$
```

```
$ ls Secret.File.gpg
Secret.File.gpg
$
$ ls Secret.File
Secret.File
```

A chave única usada para criptografar o arquivo é protegida por uma senha. Essa *passphrase* é simplesmente uma senha ou frase escolhida pelo usuário no momento da criptografia.

Para descriptografar o arquivo, o utilitário `gpg` é usado novamente. O usuário deve usar a opção `-d` e fornecer a senha para a chave secreta.

```
$ gpg -d Secret.File.gpg
gpg: CAST5 encrypted data
gpg: encrypted with 1 passphrase
This is my secret message.
...
```

A criptografia de chave simétrica é bastante simples e fácil de entender. A criptografia assimétrica é muito mais complicada e muitas vezes é um ponto de confusão em criptografia.

Criptografia de chave assimétrica

A *criptografia assimétrica*, também chamada de criptografia de chave privada/pública, utiliza duas chaves, chamadas *par de chaves*. Um par de chaves consiste em uma chave pública e uma chave privada. A chave pública é exatamente isso, pública. Não há necessidade de mantê-la em segredo. Já a chave privada precisa ser mantida em segredo.

A ideia geral da criptografia de chave assimétrica é mostrada na Figura 23.2. Um arquivo de texto simples é criptografado utilizando uma chave pública de um par de chaves. O arquivo criptografado então pode ser transmitido com segurança a outra pessoa. Para descriptografar o arquivo, a chave privada é utilizada. Essa chave privada deve vir do par de chaves pública/privada. Portanto, os dados que foram criptografados com a chave pública só podem ser descriptografados com a chave privada. A vantagem da criptografia assimétrica é a maior segurança. A desvantagem é a velocidade e o gerenciamento de chaves.

criptografia básica de chave assimétrica.

Você pode executar a criptografia assimétrica no sistema Linux usando `gpg`. Ele é um utilitário de criptografia muito versátil. Antes que possa criptografar um arquivo, você deve primeiro criar o par de chaves e um “chaveiro”. No exemplo a seguir, o comando `gpg --gen-key` foi usado. Esse comando cria um par de chaves pública/privada para o usuário `johndoe`, de acordo com as especificações desejadas. Ele também vai gerar um anel de chaves para armazená-las.

```
$ gpg --gen-key
```

```
gpg (GnuPG) 1.4.12; Copyright (C)  
2012 Free Software Foundation, Inc.
```

```

...
gpg: directory '/home/johndoe/.gnupg' created
gpg: new configuration file
'/home/johndoe/.gnupg/gpg.conf' created
...
gpg: keyring '/home/johndoe/.gnupg/secring.gpg' created
gpg: keyring '/home/johndoe/.gnupg/pubring.gpg' created
Please select what kind of key you want:
  (1) RSA and RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
Your selection? 1
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (2048)
Requested keysize is 2048 bits
Please specify how long the key should be valid.
  0 = key does not expire
  <n> = key expires in n days
  <n>w = key expires in n weeks
  <n>m = key expires in n months
  <n>y = key expires in n years
Key is valid for? (0) 7
Key expires at Mon 05 Mar 2015 03:55:29 AM EST
Is this correct? (y/N) y

You need a user ID to identify your key;
the software constructs the user ID
from the Real Name, Comment and Email Address in this form:
  "Heinrich Heine (Der Dichter) <heinrichh@duesseldorf.de>"

Real name: John Doe
Email address: jdoe@gmail.com
Comment: The User
You selected this USER-ID:
  "John Doe (The User) <jdoe@gmail.com>"

Change (N)ame, (C)omment, (E)mail or (O)kay/(Q)uit? O
You need a Passphrase to protect your secret key.
Enter passphrase:
Repeat passphrase:
...
gpg: /home/johndoe/.gnupg/trustdb.gpg: trustdb created
gpg: key 3B2E46D5 marked as ultimately trusted
upublic and secret key created and signed.
...
pub   2048R/3B2E46D5 2015-02-27 [expires: 2015-03-05]
       Key fingerprint = E202 8E43 3784 69EF 118B
       275C BA45 7DBF 3B2E 46D5
uid   John Doe (The User) <jdoe@gmail.com>
       sub 2048R/0F0E0672 2015-02-27 [expires: 2015-03-05]

```

No exemplo anterior, o utilitário `gpg` solicita diversas especificações para gerar o par desejado de chaves pública/privada:

- **Cifra de criptografia** — O RSA foi escolhido no exemplo.
- **Tamanho em bits** — Um tamanho maior de chave fornece maior segurança.
- **Período de validade** — Sete dias foram escolhidos no exemplo.
- **User ID** — Identifica a parte pública do par de chaves públicas/privadas.
- **Senha** — Usada para identificar e proteger a parte privada do par de chaves pública/privada.

enção

ícil, se não impossível matematicamente, derivar a chave privada a partir da chave pública. Mas uma possível
erabilidade foi descoberta recentemente. Um par de chaves é gerado utilizando dois números primos aleatórios. A
é que não há dois pares de chaves idênticos. Pesquisadores de segurança descobriram que os números
tórios gerados não são tão aleatórios. Assim, há a possibilidade de haver o mesmo par de chaves que outra
soa na internet já possui. Se compartilhar o mesmo par de chaves, você terá a capacidade de descriptografar as
sagens criptografadas deles com a chave pública usando sua chave privada. Portanto, você deve, no mínimo,
ar o tamanho de chave de 2048 bits para reduzir a possibilidade dessa potencial situação.

O usuário johndoe pode verificar o chaveiro usando o comando `gpg --list-keys`, como
mostrado no código a seguir. Observe que o User ID (UID) da chave pública é mostrado da maneira
como foi criado, contendo o nome verdadeiro, um comentário e o e-mail johndoe.

```
$ gpg --list-keys
/home/johndoe/.gnupg/pubring.gpg
-----
pub   2048R/3B2E46D5 2015-02-27 [expires: 2015-03-05]
uid           John Doe (The User) <jdoe@gmail.com>
sub   2048R/0F0E0672 2015-02-27 [expires: 2015-03-05]
```

Depois que o par de chaves e o chaveiro são gerados, os arquivos podem ser criptografados e
descriptografados. Primeiro, a chave pública deve ser extraída do chaveiro para poder ser compartilhada.
No exemplo a seguir, o utilitário `gpg` é usado para extrair a chave pública do chaveiro de johndoe. A
chave extraída é colocada em um arquivo para ser compartilhada. O nome do arquivo pode ser qual você
quiser. Nesse caso, o usuário johndoe escolheu o nome de arquivo `JohnDoe.pub`.

```
$ gpg --export John Doe > JohnDoe.pub
$ ls *.pub
JohnDoe.pub
```

O arquivo contendo a chave pública pode ser compartilhado de várias maneiras. Ele pode ser enviado
como um anexo via e-mail ou até mesmo publicado em uma página web. A chave pública é considerada
pública, por isso, não há necessidade de ocultá-la. No exemplo a seguir, johndoe deu o arquivo que
contém sua chave pública para a usuária christineb. Ela adiciona a chave pública de johndoe ao
seu chaveiro, usando o comando `gpg --import`. A usuária christineb verifica se a chave pública
de johndoe foi adicionada usando o comando `gpg --list-keys` para ver seu chaveiro.

```
$ ls *.pub
JohnDoe.pub
```

```

$
$ gpg --import JohnDoe.pub
gpg: directory '/home/christineb/.gnupg' created
...
gpg: key 3B2E46D5:
  public key "John Doe (The User) <jdoe@gmail.com>" imported
gpg: Total number processed: 1
gpg: imported: 1 (RSA: 1)
$
$ gpg --list-keys
/home/christineb/.gnupg/pubring.gpg
-----
pub   2048R/3B2E46D5 2015-02-27 [expires: 2015-03-05]
uid           John Doe (The User) <jdoe@gmail.com>
sub   2048R/0F0E0672 2015-02-27 [expires: 2015-03-05]

```

Depois que a chave é adicionada ao chaveiro, essa chave pública pode ser usada para criptografar os dados para o proprietário original dela. No código de exemplo a seguir, observe o seguinte:

- `christineb` criou um arquivo de texto `MessageForJohn.txt` para o usuário `johndoe`.
- Ela criptografa o arquivo usando a chave pública *dele*.
- O arquivo criptografado, `MessageForJohn`, é criado pela opção `--out`.
- A opção `--recipient` identifica a chave pública de `johndoe` usando apenas a parte do nome real do UID da sua chave pública entre aspas, "John Doe".

```

$ gpg --out MessageForJohn --recipient "John Doe"
--encrypt MessageForJohn.txt
...
$ ls
JohnDoe.pub MessageForJohn MessageForJohn.txt

```

O arquivo criptografado da mensagem, `MessageForJohn`, criado a partir do arquivo de texto simples, `MessageForJohn.txt`, pode ser enviado seguramente para o usuário `johndoe`. Para decriptografar essa mensagem, `johndoe` usará a chave privada *dele*, identificada e protegida pela senha secreta usada originalmente para criar a chave. Depois que `johndoe` fornece a senha apropriada, `gpg` decriptografa o arquivo da mensagem e o coloca no arquivo `ChristinesMessage`, designado pela opção `--out`. Depois de decriptografada, ele poderá ler a mensagem de texto simples.

```

$ ls MessageForJohn
MessageForJohn
$
$ gpg --out ChristinesMessage --decrypt MessageForJohn
You need a passphrase to unlock the secret key for
user: "John Doe (The User) <jdoe@gmail.com>"
2048-bit RSA key, ID 0F0E0672, created 2015-02-27
  (main key ID 3B2E46D5)
gpg: encrypted with 2048-bit RSA key, ID 0F0E0672,
created 2015-02-27
  "John Doe (The User) <jdoe@gmail.com>"

```

```
$  
$ cat ChristinesMessage  
I know you are not the real John Doe.
```

Para revisar, os passos necessários para criptografar/decriptografar os arquivos usando chaves assimétricas são os seguintes:

1. **Gerar o par de chaves e o chaveiro.**
2. **Exportar uma cópia da sua chave pública para um arquivo.**
3. **Compartilhar o arquivo de chave pública.**
4. **Indivíduos que quiserem enviar-lhe arquivos criptografados adicionam *sua* chave pública (a que você criou para você mesmo) ao chaveiro deles.**
5. **Um arquivo é criptografado usando *sua* chave pública.**
6. **O arquivo criptografado é enviado para você.**
7. **Você decriptografa o arquivo com *sua* chave privada.**

Você pode ver por que as chaves assimétricas podem causar confusão! Lembre-se de que na criptografia assimétrica cada par de chave pública/chave privada é um conjunto que funciona integralmente.

Entendendo assinaturas digitais

Uma *assinatura digital* é um comprovante eletrônico utilizado para verificação e autenticação de dados. Uma assinatura digital não é uma verificação da sua assinatura física. Em vez disso, ela é um token criptográfico enviado com um arquivo para que o receptor do arquivo possa ter certeza de que o arquivo veio de você e não foi modificado de nenhuma maneira.

Ao criar uma assinatura digital, ocorrem as seguintes etapas:

1. **Você cria um arquivo ou uma mensagem.**
2. **Usando o utilitário `gpg`, você cria um hash ou um resumo de mensagem do arquivo.**
3. **O utilitário `gpg` então criptografa o hash e o arquivo usando uma cifra de chave assimétrica. Para a criptografia, a chave privada do par de chaves pública/privada é utilizada. Isso agora é um arquivo criptografado digitalmente assinado.**
4. **Você envia o hash criptografado (conhecido como assinatura digital) e o arquivo para o receptor.**
5. **O receptor recria o hash ou resumo da mensagem do arquivo criptografado recebido.**
6. **Com o utilitário `gpg`, o receptor decriptografa a assinatura digital recebida usando a chave pública para obter o hash original ou resumo de mensagem.**
7. **O utilitário `gpg` compara o hash original com o hash recriado para ver se eles correspondem. Se corresponderem, o receptor é informado de que a assinatura digital é boa.**
8. **O receptor agora pode ler o arquivo decriptografado.**

Observe no Passo 3 que a chave privada é utilizada em primeiro lugar. Na descrição da criptografia de chave assimétrica, a chave pública foi utilizada em primeiro lugar. A criptografia de chave assimétrica é flexível o suficiente para permitir que você use sua própria chave privada para criptografar e o receptor use a chave pública que você criou para decifrar.

ta

Assinaturas digitais têm suas próprias cifras especiais. Embora várias cifras possam lidar com a criptografia e criação de assinaturas, há algumas cuja única função é criar assinaturas digitais. As cifras criptográficas mais populares para criar assinaturas são RSA e Digital Signature Algorithm (DSA). O algoritmo RSA pode ser utilizado para criptografar e criar assinaturas, enquanto o DSA só pode ser utilizado para criar Digital Signatures.

Como você pode ver, uma assinatura digital contém o hash criptográfico e a criptografia de chave assimétrica. Esse processo complicado é frequentemente tratado por um aplicativo que foi configurado para fazer isso, em vez de tratado diretamente pelos usuários do sistema Linux. Mas você pode adicionar manualmente suas próprias assinaturas digitais a documentos.

Digamos que o usuário `john DOE` enviará uma mensagem para a usuária `christineb`, juntamente com sua assinatura digital. Ele criou um arquivo de texto simples contendo a mensagem a enviar. Ele usa o utilitário `gpg` para criar o arquivo de assinatura e criptografar o arquivo de mensagem. A opção `--sign` informa o utilitário `gpg` que `MessageForChristine.txt` é o arquivo que deve ser criptografado e usado para criar a assinatura digital. Em resposta, o utilitário `gpg`:

- Cria um resumo de mensagem (conhecido como hash) do arquivo de mensagem
- Criptografa o resumo da mensagem, o que cria a assinatura digital
- Criptografa o arquivo de mensagem
- Armazena o conteúdo criptografado no arquivo especificado pela opção `--output`, `JohnDoe.DS`

O arquivo `JohnDoe.DS` agora contém uma mensagem criptografada e assinada digitalmente. O código a seguir demonstra esse processo:

```
$ gpg --output JohnDoe.DS --sign MessageForChristine.txt
```

```
You need a passphrase to unlock the secret key for
user: "John Doe (The User) <jdoe@gmail.com>"
2048-bit RSA key, ID 3B2E46D5, created 2015-02-27
```

Depois que a usuária `christineb` recebe o arquivo assinado e criptografado, ela pode usar o utilitário `gpg` para verificar a assinatura digital e decifrar o arquivo em uma única etapa. No código a seguir, a opção `--decrypt` é usada juntamente com o nome do arquivo assinado digitalmente, `JohnDoe.DS`. A mensagem do arquivo é decifrada e mostrada. A assinatura digital do arquivo é verificada e é válida.

```
$ gpg --decrypt JohnDoe.DS
```

```
I am the real John Doe!
```

```
gpg: Signature made Mon 27 Feb 2015 09:42:36 AM EST
```

```
using RSA key ID 3B2E46D5
```

```
gpg: Good signature from "John Doe (The User) <jdoe@gmail.com>"
```

```
...
```

Sem a chave pública de johndoe no chaveiro dela, christineb não seria capaz de decriptografar essa mensagem e verificar a assinatura digital.

ca

Exemplo anterior de assinatura digital de um documento permite que qualquer pessoa com a chave pública consiga decriptografar o documento. A fim de mantê-lo realmente privado, use a chave pública do destinatário para assinar e criptografar com as opções de `gpg: --sign` e `--encrypt`. O destinatário pode decriptografar com a chave pública dele.

Entender alguns princípios básicos de criptografia irá ajudá-lo a começar a proteger seu sistema Linux com criptografia. Tenha em mente que só discutimos os princípios básicos neste capítulo. Há vários outros aspectos da criptografia, como certificados digitais e infraestrutura de chave pública, que são importantes entender.

Implementando a criptografia no Linux

Muitas ferramentas de criptografia estão disponíveis no sistema Linux. Aquelas que você opta por usar dependem dos requisitos de segurança da organização. A seguir, fazemos uma resenha de algumas das ferramentas de criptografia disponíveis no Linux.

Garantindo a integridade dos arquivos

No início deste capítulo, a integridade de um arquivo ISO foi verificada com o utilitário de resumo de mensagem `md5sum`. A outra ferramenta mais popular de resumo de mensagem é uma que usa o hash SHA-1, `sha1sum`. Ele funciona de forma idêntica ao utilitário de linha de comando `md5sum`, como mostrado no código a seguir. Se um arquivo ISO tiver um hash SHA-1 listado, em vez de uma soma de verificação MD5, você pode usar o seguinte para verificar o hash.

```
$ sha1sum Fedora-16-i686-Live-Desktop.iso
```

```
791f2677c7ac03ccacde3e26df50a36a277d3bf0
```

```
Fedora-16-i686-Live-Desktop.iso
```

Infelizmente, desde 2005, o padrão de hash SHA-1 não é mais considerado um hash criptográfico devido a algumas “fraquezas matemáticas”. Mas, como ocorre com o MD5, isso não diminuiu sua popularidade para verificação de integridade de arquivos.

Se sua organização específica exigir um utilitário de hash criptográfico real, você precisará usar uma das ferramentas de hash criptográfico SHA-2. No Linux, estas incluem:

- `sha224sum`
- `sha256sum`
- `sha384sum`
- `sha512sum`

Essas ferramentas funcionam como o comando `sha1sum`, exceto, naturalmente, que usam o padrão de hash criptográfico SHA-2. A única diferença entre as várias ferramentas SHA-2 é o tamanho de chave que elas usam. O comando `sha224sum` utiliza um tamanho de chave de 224 bits, o comando `sha256sum` utiliza um tamanho de chave de 256 bits etc. Lembre-se de que quanto maior o tamanho da chave, menor a chance de quebrar a cifra.

O padrão de hash criptográfico SHA-2 foi criado pela National Security Agency (NSA). A NSA já tem outro padrão de hash criptográfico que será lançado em breve, o SHA-3.

Criptografando um sistema de arquivos Linux

Talvez você precise criptografar todo um sistema de arquivos no servidor Linux. Isso pode ser feito de várias maneiras diferentes, incluindo o uso de uma ferramenta FOSS (Free and Open Source Software), como o TrueCrypt, www.truecrypt.org.

Uma das suas opções no Linux é criptografar sua partição raiz na instalação (ver Capítulo 9, “Instalando o Linux”). Muitas distribuições Linux incluem uma opção de criptografia durante o processo de instalação. A Figura 23.3 mostra a opção de criptografia durante uma instalação do Fedora.

ção de criptografia na instalação do Linux Fedora.

Depois de selecionar essa opção durante a instalação, você será solicitado a fornecer uma senha. Essa é a criptografia de chave simétrica com uma senha protegendo a chave única. A Figura 23.3 mostra a instalação solicitando a senha da chave. A senha deve ter pelo menos oito caracteres.

ha de criptografia de chave simétrica do Linux Fedora.

Se selecionar essa opção de criptografia, sempre que iniciar o sistema, você será solicitado a fornecer a senha da chave simétrica. A Figura 23.5 mostra como ela aparece. Isso protege a partição raiz, caso o disco em que ela reside for roubado.

itando a senha da chave de criptografia simétrica na inicialização.

Se você herdar um sistema com um disco criptografado, usando privilégios de root, poderá usar os comandos `lvs` e `cryptsetup` e o arquivo `/etc/crypttab` para ajudá-lo. A seguir, o comando `lvs` mostra todos os volumes lógicos atuais no sistema e seus nomes de dispositivo subjacentes. Veja no Capítulo 12, “Gerenciando discos e sistemas de arquivos”, uma revisão do comando `lvs`.

```
# lvs -o devices
Devices
/dev/mapper/luks-b099fbbe-0e56-425f-91a6-44f129db9f4b (56)
/dev/mapper/luks-b099fbbe-0e56-425f-91a6-44f129db9f4b (0)
```

Nesse sistema, observe que os nomes de dispositivo subjacentes começam com `luks`. Isso indica que o padrão Linux Unified Key Setup (LUKS) para criptografia de disco rígido foi usado. Para obter mais informações sobre o LUKS, consulte a página inicial do LUKS em <http://code.google.com/p/cryptsetup>.

ta

untu não vem com o comando `lvs` instalado por padrão. Para instalá-lo, digite `sudo apt-get install lvm2` na linha de comando.

Os volumes lógicos criptografados são montados na inicialização usando as informações do arquivo `/etc/crypttab`, como mostrado no código a seguir. Observe que os nomes `luks` são os mesmos que os listados pelo comando `lvs` no exemplo anterior.

```
# cat /etc/crypttab
luks-b099fbbe-0e56-425f-91a6-44f129db9f4b
    UUID=b099fbbe-0e56-425f-91a6-44f129db9f4b none
```

Você também pode usar o comando `cryptsetup` para ajudá-lo a descobrir mais informações sobre os volumes criptografados do seu sistema Linux. No exemplo a seguir, a opção `status` é usada junto com o nome do dispositivo `luks` para determinar informações adicionais.

```
# cryptsetup status luks-b099fbbe-0e56-425f-91a6-44f129db9f4b
/dev/mapper/luks-b099fbbe-0e56-425f-91a6-44f129db9f4b
is active and is in use.
type:      LUKS1
cipher:    aes-xts-plain64
keysize:   512 bits
device:    /dev/sda3
offset:    4096 sectors
size:      15742976 sectors
mode:      read/write
```

Criptografando um diretório do Linux

Você também pode usar o utilitário `ecryptfs` para criptografar em um sistema Linux. O utilitário `ecryptfs` não é um tipo de sistema de arquivos, como o nome indica. Em vez disso, ele é um utilitário compatível com POSIX que permite criar uma camada de criptografia sobre qualquer sistema de arquivos.

O utilitário `ecryptfs` não é instalado por padrão em um sistema Linux Fedora ou RHEL. Você precisa usar o comando `yum install ecryptfs-utils`. Se ele não estiver instalado em um sistema Debian, use o comando `sudo apt-get install ecrypt-utils`.

ca

O utilitário `ecryptfs` é usado para a criptografia, um erro comum é colocar a letra `n` depois da letra `e` na opção `ecryptfs`. Se você receber um erro ao usar os utilitários `ecryptfs`, certifique-se de que não utilizou a opção `encryptfs` por engano.

No exemplo a seguir, o usuário `johndoe` terá um subdiretório criptografado usando o utilitário `ecryptfs`. Primeiro, não deve haver atualmente arquivos no diretório antes de ele ser criptografado. Se houver arquivos localizados aí, mova-os para um lugar seguro até que a criptografia tenha sido concluída. Se não movê-los, você não será capaz de acessá-los enquanto o diretório estiver sendo criptografado.

Agora, para criptografar o diretório `/home/johndoe/Secret`, use o comando `mount`. Examine o comando `mount` usado no exemplo a seguir. Ele é um pouco parecido com o comando `mount` normal, exceto que o tipo de partição usado seja `ecryptfs`. O item a montar e seu ponto de montagem estão no mesmo diretório! Você literalmente criptografa o diretório e monta-o no próprio diretório. O outro item incomum sobre esse comando `mount` é que ele inicializa o utilitário `ecryptfs`, que faz algumas perguntas interativas.

```
# mount -t ecryptfs /home/johndoe/Secret /home/johndoe/Secret
```

```
Select key type to use for newly created files:
```

- 1) tspi
- 2) passphrase
- 3) pkcs11-helper
- 4) openssl

```
Selection: 2
```

```
Passphrase:
```

```
Select cipher:
```

- 1) aes: blocksize = 16;
min keysize = 16; max keysize = 32 (loaded)
- 2) blowfish: blocksize = 16;
min keysize = 16; max keysize = 56 (not loaded)
- 3) des3_ede: blocksize = 8;
min keysize = 24; max keysize = 24 (not loaded)

```
4) twofish: blocksize = 16;
min keysize = 16; max keysize = 32 (not loaded)
5) cast6: blocksize = 16;
min keysize = 16; max keysize = 32 (not loaded)
6) cast5: blocksize = 8;
min keysize = 5; max keysize = 16 (not loaded)
```

Selection [aes]: 1

Select key bytes:

- 1) 16
- 2) 32
- 3) 24

Selection [16]:

Enable plaintext passthrough (y/n) [n]:

Enable filename encryption (y/n) [n]:

Attempting to mount with the following options:

```
ecryptfs_unlink_sigs
ecryptfs_key_bytes=16
ecryptfs_cipher=aes
ecryptfs_sig=70993b8d49610e67
```

WARNING: Based on the contents of [/root/.ecryptfs/sig-cache.txt] it looks like you have never mounted with this key before. This could mean that you have typed your passphrase wrong.

Would you like to proceed with the mount (yes/no)? : yes

Would you like to append sig [70993b8d49610e67] to
[/root/.ecryptfs/sig-cache.txt]

in order to avoid this warning in the future (yes/no)? : yes

Successfully appended new sig to user sig cache file

Mounted eCryptfs

O utilitário `ecryptfs` permite que você escolha o seguinte:

- Tipo de chave
- Senha
- Cifra
- Tamanho da chave (em bytes)
- Ativar ou desativar o texto simples a enviar
- Ativar ou desativar a criptografia do nome de arquivo

Ele também avisa quando você monta esse diretório criptografado pela primeira vez porque a chave não foi usada antes. O utilitário permite aplicar uma assinatura digital ao diretório montado de modo que, se

you use `mount` again, it will only mount the directory and not prompt for a password.

ca

It would be a good idea to note the selections you make when you initially mount a `ecryptfs` partition. You will need the exact same selections the next time you remount the partition.

To verify if the encrypted directory is now mounted, use the `mount` command again. In the example that follows, the `mount` command is used and then redirected to `grep` to search for the directory `/home/johndoe/Secret`. As you can see, the directory is mounted with a type of `ecryptfs`.

```
# mount | grep /home/johndoe/Secret
/home/johndoe/Secret on /home/johndoe/Secret type encryptfs
(rw,relatime,encryptfs_sig=70993b8d49610e67,encryptfs_cipher=aes,
encryptfs_key_bytes=16,encryptfs_unlink_sigs)
```

Up to now, you have not seen the effects of this mounted and encrypted directory. In the text that follows, the file `my_secret_file` is copied to the encrypted partition. The user `johndoe` can still use the `cat` command to display the file in plain text. The file is decrypted automatically by the `ecryptfs` layer.

```
$ cp my_secret_file Secret
$ cat /home/johndoe/Secret/my_secret_file
Shh... It's a secret.
```

The user `root` can also use the `cat` command to display the file in plain text.

```
# cat /home/johndoe/Secret/my_secret_file
Shh... It's a secret.
```

But after the encrypted directory is unmounted using the `umount` command, the files are no longer automatically decrypted. The file `my_secret_file` is now not useful and cannot be read, even by the user `root`.

```
# umount /home/johndoe/Secret
#
# cat /home/johndoe/Secret/my_secret_file
#d-6 X1 O#### #####"3DUfw' #Qng y+ 4 #W #_CONSOLEp ; Iag `L
...
```

Thus, the utility `ecryptfs` will allow you to create a local file system in the system to encrypt and decrypt files quickly. But after the directory is no longer mounted as a type of

encryptfs, os arquivos estão seguros e não podem ser descriptografados.

Criptografando um arquivo do Linux

A ferramenta mais popular para criptografar arquivos em um sistema Linux é um utilitário OpenPGP, o GNU Privacy Guard, ou `gpg`. Sua flexibilidade e variedade de opções, juntamente com o fato de ser instalado por padrão na maioria das distribuições do Linux, contribuem para sua popularidade.

Atenção

Se sua organização usar uma empresa terceirizada de armazenamento em nuvem, você precisará entender que nem todas essas empresas, como a DropBox, só criptografam os arquivos depois que eles são recebidos. Isso significa que a empresa tem as chaves necessárias para descriptografar os arquivos e pode deixar os dados da sua organização vulneráveis. Criptografar os arquivos no seu sistema Linux antes de eles serem enviados para o armazenamento em nuvem adicionará uma camada extra de proteção necessária.

Mas há várias outras ferramentas de criptografia que você pode usar em um sistema Linux para criptografar os arquivos. E, assim como o `gpg`, muitas dessas ferramentas permitem fazer muito mais do que apenas criptografar arquivos. A seguir estão algumas das ferramentas populares de criptografia do Linux que você pode usar para criptografar arquivos:

- **aescript** Usa o software de cifra de chave simétrica Rijndael, também chamado AES. Essa ferramenta FOSS de terceiros está disponível para download em www.aescript.com.
- **bcrypt** Essa ferramenta usa a cifra de chave simétrica blowfish. Ele não é instalado por padrão. Depois que `bcrypt` está instalado, as páginas man tornam-se disponíveis.
 - Para o Fedora e RHEL: `yum install bcrypt`.
 - Para o Ubuntu: `sudo apt-get install bcrypt`.
- **ccrypt** Essa ferramenta usa o software de cifra de chave simétrica Rijndael, também chamado AES. Ela foi criada para substituir o utilitário `crypt` padrão do Unix e não é instalada por padrão. Depois que `ccrypt` está instalado, as páginas man tornam-se disponíveis.
 - Para o Fedora e RHEL: `yum install ccrypt`.
 - Para o Ubuntu: `sudo apt-get install ccrypt`.
- **gpg** Esse utilitário pode usar pares de chaves assimétricas ou uma chave simétrica. Ele é instalado por padrão e é a ferramenta de criptografia preferida para servidores Linux. A cifra padrão a usar é definida no arquivo `gpg.conf`. Há páginas man disponíveis, bem como `info gnupg`.
- **gpg2** Semelhante ao `gpg`, mas direcionada para o ambiente de desktop, a ferramenta `gpg2` não é instalada por padrão. Depois que `gpg2` está instalada, as páginas man tornam-se disponíveis.
 - Para o Fedora e RHEL: `yum install ccrypt`.
 - Não disponível no Ubuntu.

Tenha em mente que essa lista só abrange as ferramentas mais populares. Além disso, lembre-se de que muitas dessas ferramentas de criptografia de arquivo podem ser usadas para mais do que apenas criptografia de arquivos.

Criptografando várias coisas no Linux

Você pode aplicar criptologia a quase tudo no Linux. Além de sistemas de arquivos, diretórios e arquivos, você também pode criptografar backups, arquivos zip, conexões de rede e muito mais.

A Tabela 23.2 lista algumas das diversas ferramentas de criptologia do Linux e o que elas fazem. Se quiser ver a lista completa das ferramentas de criptografia instaladas na sua distribuição atual do Linux, digite **man -k crypt** na linha de comando.

TABELA 23.2 Diversas Ferramentas de Criptografia do Linux

Ferramenta	Descrição
duplicit	Criptografa backups. Instalada por padrão no Fedora e no RHEL. Para instalar no Ubuntu, digite sudo apt-get install duplicity na linha de comando.
gpg	Utiliza o GNU Privacy Guard para criptografar ou assinar arquivos em um repositório de arquivos. Instalada por padrão.
openssl	Um conjunto de ferramentas que implementa os protocolos Secure Socket Layer (SSL) e Transport Layer Security (TLS). Esses protocolos exigem criptografia. Instalada por padrão.
ssh	Um gerenciador de chave de criptografia GNU Privacy Guard. Instalado por padrão no Ubuntu. Para instalar no Fedora e no RHEL, digite yum install seahorse na linha de comando.
sshpass	Criptografa acesso remoto através de uma rede. Instalada por padrão.
zipcloak	Criptografa as entradas em um arquivo Zip. Instalada por padrão.

Como vários outros itens em um sistema Linux, as ferramentas de criptografia disponíveis são ricas e abundantes. Isso lhe dá a flexibilidade e variedade de que você precisa para implementar os padrões de criptologia que sua organização específica exige.

Outra ferramenta de segurança extremamente poderosa disponível no Linux é o PAM. A próxima seção neste capítulo aborda os conceitos básicos do PAM e como você pode usar essa ferramenta para aprimorar ainda mais a segurança do seu sistema Linux.

Implementando a segurança do Linux com PAM

O *Pluggable Authentication Modules (PAM)*, isto é, *Módulos de Autenticação Plugáveis*, foi inventado pela Sun Microsystems e inicialmente implementado no sistema operacional Solaris. O projeto Linux-PAM começou em 1997. Hoje, a maioria das distribuições Linux usa o PAM.

PAM simplifica o processo de gerenciamento de autenticação. Lembre-se de que a autenticação (ver Capítulo 22, “Entendendo a segurança básica do Linux”) é o processo de determinar que um sujeito (conhecido como usuário ou processo) é quem ele diz ser. Esse processo é às vezes chamado de “identificação e autenticação”. O PAM é um método centralizado para fornecer autenticação para

aplicativos e sistema Linux.

Os aplicativos podem ser escritos para usar o PAM e são chamados “compatíveis com PAM”. Um aplicativo compatível com PAM não tem de ser reescrito e recompilado para que as configurações de autenticação possam ser alteradas. As alterações necessárias são feitas dentro de um arquivo de configuração PAM para os aplicativos compatíveis com PAM. Assim, o gerenciamento de autenticação para esses aplicativos é centralizado e simplificado.

Você pode ver se um determinado aplicativo ou utilitário Linux é compatível com PAM. Verifique se ele foi compilado com a biblioteca PAM, `libpam.so`. No exemplo a seguir, o aplicativo `crontab` é verificado em termos da compatibilidade com PAM. O comando `ldd` verificará as dependências de um arquivo de biblioteca compartilhado. Para mantê-lo simples, `grep` é usado para pesquisar a biblioteca PAM. Como você pode ver, `crontab` nesse sistema Linux particular é compatível com PAM.

```
# ldd /usr/bin/crontab | grep pam
libpam.so.0 => /lib/libpam.so.0 (0x44d12000)
```

Os benefícios da utilização do PAM em um sistema Linux incluem os seguintes:

- Gerenciamento simplificado e centralizado de autenticação do ponto de vista do administrador.
- Desenvolvimento simplificado de aplicativos, porque os desenvolvedores podem escrever aplicativos usando a biblioteca PAM documentada, em vez de escrever suas próprias rotinas de autenticação.
- Flexibilidade na autenticação:
 - Permitir ou negar acesso a recursos com base em critérios tradicionais, como a identificação.
- Permitir ou negar acesso com base em outros critérios, como o horário das restrições do dia.
- Configurar limites aos sujeitos, como o uso de recursos.

Embora os benefícios do PAM simplifiquem o gerenciamento de autenticação, a forma como o PAM funciona não é tão simples.

Entendendo o processo de autenticação PAM

Quando um sujeito (usuário ou processo) solicita acesso a um aplicativo ou utilitário compatível com PAM, dois componentes principais são usados para completar o processo de autenticação do sujeito:

- Arquivo de configuração do aplicativo compatível com PAM
- Os módulos PAM que o arquivo de configuração usa

O arquivo de configuração de cada aplicativo compatível com PAM está no centro do processo. Os arquivos de configuração do PAM chamam certos módulos PAM para realizar a autenticação necessária.

Módulos PAM autenticam os sujeitos a partir dos dados de autorização do sistema, como uma conta de usuário centralizada utilizando LDAP (ver Capítulo 11, “Gerenciando contas de usuário”).

O Linux vem com muitos aplicativos que são compatíveis com PAM, seus arquivos de configuração necessários e módulos PAM já instalados. Se tiver quaisquer necessidades especiais de autenticação, é mais provável que você encontre um módulo PAM que já foi escrito para essa necessidade. Mas antes de começar a ajustar o PAM, você precisa entender mais sobre como ele funciona.

Uma série de passos é seguida pelo PAM utilizando os módulos e os arquivos de configuração para garantir que a autenticação apropriada de aplicativo ocorra:

1. **Um sujeito (usuário ou processo) solicita acesso a um aplicativo.**
2. **O arquivo de configuração PAM do aplicativo, que contém uma política de acesso, é aberto e lido.**

A política de acesso é definida por meio de uma lista de todos os módulos PAM a serem usados no processo de autenticação. Essa lista de módulo(s) PAM é chamada pilha.

3. **Cada módulo PAM na pilha é invocado na ordem em que está listado.**
4. **Cada módulo PAM retorna um status de sucesso ou falha.**
5. **A pilha continua a ser lida na ordem e não necessariamente é parada por um simples status de falha retornado.**
6. **Os resultados do status de todos os módulos PAM são combinados em um único resultado geral de sucesso ou falha da autenticação.**

Normalmente, se um único módulo PAM retornar um status de falha, o acesso ao aplicativo é negado. Mas isso depende das especificações das configurações de arquivo. A Figura 23.6 mostra um exemplo de uma solicitação de autenticação sendo negada pelo PAM.

Todos os arquivos de configuração do PAM estão localizados em `/etc/pam.d`. O formato geral de um arquivo de configuração do PAM é: `context control flag PAM module module options`

so de autenticação do PAM.

Eis o arquivo de configuração do PAM para o comando `poweroff`. Observe que o arquivo começa com uma linha de comentário. Toda linha que começa com um caractere `#` é ignorada.

```
$ cat /etc/pam.d/poweroff
#%PAM-1.0
auth        sufficient    pam_rootok.so
auth        required      pam_console.so
#auth       include       system-auth
account     required        pam_permit.so
```

Lembre-se de que todos os módulos PAM listados na pilha do arquivo de configuração serão chamados, na ordem, e solicitados a retornar um status. No arquivo de configuração anterior do PAM, há três status retornados para determinar se o sujeito pode acessar o comando `poweroff`. Para entender melhor como esses arquivos de configuração são usados, você precisa analisar cada pedaço do formato geral.

ca

Em Ubuntu, os arquivos de configuração do PAM podem incluir outros arquivos de configuração para autenticação. O arquivo de configuração a ser incluído é listado com um `@` na frente do nome.

Entendendo contextos PAM

Módulos PAM têm funções padrão que fornecem diferentes serviços de autenticação. Essas funções padrão dentro de um módulo PAM podem ser divididas em dois tipos de funções, chamadas *contextos*. Contextos também podem ser chamados interfaces ou tipos de módulo. Na Tabela 23.3, os diferentes contextos PAM estão listados juntamente com o tipo de serviço de autenticação que eles fornecem.

Tabela 23.3 Contextos PAM

Contexto	Descrição do serviço
auth	Fornecer serviços de gerenciamento de autenticação, como verificação de senhas de contas
account	Fornecer serviços de validação de conta, como restrições de acesso em certos horários
password	Gerenciar senhas de conta, tais como restrições de tamanho de senha
session	Gerenciar o início e o fim de uma sessão autenticada, como o envio de informações para logs de segurança

No arquivo de configuração `poweroff`, mostrado novamente aqui, apenas dois contextos PAM são usados `auth` e `account`. Assim, não há necessidade de esse aplicativo ter serviços de gerenciamento

password ou session do PAM. Observe que um dos contextos auth será ignorado, porque sua linha começa com um caractere #:

```
$ cat /etc/pam.d/poweroff
#%PAM-1.0
auth      sufficient    pam_rootok.so
auth      required      pam_console.so
#auth     include        system-auth
account   required      pam_permit.so
```

O contexto auth é listado duas vezes no arquivo de configuração mostrado no código anterior. Mas para cada contexto auth, um flag de controle um e módulo PAM diferentes são usados. Cada um dos flags de controle tem seu próprio significado e função especiais.

Entendendo flags de controle PAM

Em um arquivo de configuração do PAM, os flags de controle são utilizados para determinar o status geral, o qual retornará ao aplicativo. Um flag de controle é um dos seguintes:

- Palavra-chave simples**- A única preocupação aqui é se o módulo PAM correspondente retorna uma resposta de “falha” ou “sucesso”. Consulte a Tabela 23.4 para saber como esses status são tratados.
- Série de ações**- O status do módulo retornado é tratado por meio de uma série de ações listadas no arquivo.

A Tabela 23.4 mostra os vários flags de controle de palavras-chave e suas respostas ao status do módulo retornado. Observe que alguns dos flags de controle precisam ser cuidadosamente colocados na pilha do arquivo de configuração. Alguns flags de controle farão com que o processo de autenticação pare imediatamente e os demais módulos PAM não sejam chamados. Os flags de controle simplesmente controlam como os resultados do status do módulo PAM são combinados em um único resultado geral. A Tabela 23.4 demonstra como os resultados do status são combinados.

Tabela 23.4 Flags de controle de configuração PAM e como a resposta é tratada

Flag de controle	Descrição do tratamento da resposta
required	Se falhar, retorna um status de falha ao aplicativo, depois que todos os demais contextos foram executados na pilha.
required_once	Se falhar, retorna imediatamente um status de falha ao aplicativo, sem executar o restante da pilha. (Tenha cuidado sobre onde você coloca esse controle na pilha.)
sufficient	Em caso de falha, o status do módulo é basicamente ignorado. Em caso de sucesso, então, um status de sucesso é imediatamente devolvido para o aplicativo, sem executar o resto da pilha. (Tenha cuidado sobre onde você coloca esse controle na pilha.)
optional	Esse flag de controle só é importante para o status de retorno geral final de sucesso ou falha. Pense nele como um desempate. Quando os outros módulos na pilha do arquivo de configuração retornam status que não são status claros de falha ou sucesso, o status desse módulo opcional é usado para determinar o status final ou o desempate. Nos casos em que os outros módulos na pilha retornam um caminho claro de falha ou sucesso, esse status é ignorado.

lude	Obtém todos os status de retorno da pilha desse arquivo específico de configuração PAM para incluir no status de retorno geral dessa pilha. É como se a pilha inteira proveniente do arquivo de configuração nomeado estivesse agora nesse arquivo de configuração.
stack	Similar ao flag de controle incluído, exceto por como certos erros e avaliações afetam a pilha principal. Isso força a pilha do arquivo de configuração incluída a funcionar como uma subpilha para a pilha principal. Assim, alguns erros e avaliações só afetam a subpilha, não a pilha principal.

Você deve entender que os módulos PAM retornam muitos outros códigos de resultado de status do que apenas “sucesso” ou “falha”. Por exemplo, um módulo poderia retornar o código de status de `PAM_ACCT_EXPIRED`, o que significa que a conta do usuário expirou. Isso seria considerado uma “falha”.

Entendendo módulos PAM

Um módulo PAM é, na verdade, um conjunto compartilhado de módulos de biblioteca (arquivos DLL) armazenados em `/lib/security`. Você pode ver uma lista dos vários módulos PAM instalados no sistema digitando `ls /lib/security/pam*.so` na linha de comando.

ta
buntu, para encontrar os módulos PAM, digite o comando `sudo find / -name pam*.so` na linha de comando.

O sistema Linux vem com muitos dos módulos PAM necessários já instalados. Não há necessidade de você mesmo escrever quaisquer módulos PAM. Se você precisar de um módulo ainda não instalado, é muito provável que outra pessoa já o escreveu. Confira fontes como:

- <http://www.openwall.com/pam/>
- <http://puszcza.gnu.org.ua/software/pam-modules/download.html>

Compreendendo os arquivos de configuração de evento de sistema PAM

Até agora, o foco foi aplicativos compatíveis com PAM e seus arquivos de configuração. Mas outros eventos de sistema, como fazer o login no sistema Linux, também usam PAM. Assim, esses eventos também têm arquivos de configuração.

O seguinte é uma listagem parcial de diretórios do diretório do arquivo de configuração PAM. Observe que há arquivos de configuração de aplicativo compatíveis com PAM, como `cond`, e arquivos de configuração de evento de sistema, como `postlogin-ac`.

```
# ls -l /etc/pam.d
total 204
-rw-r--r--. 1 root root 272 Nov 15 10:06 atd
...
-rw-r--r--. 1 root root 232 Jan 31 12:35 config-util
-rw-r--r--. 1 root root 293 Oct 26 23:10 crond
```

```
...
-rw-r--r--. 1 root root 109 Feb 28 01:33 postlogin-ac
-rw-r--r--. 1 root root 147 Oct 3 11:51 poweroff
...
-rw-r--r--. 1 root root 981 Feb 28 01:33 system-auth-ac
...
```

Você pode modificar esses arquivos de configuração de evento de sistema para implementar as necessidades específicas de segurança da sua organização. Por exemplo, o arquivo `system-auth-ac` pode ser modificado para forçar certas restrições de senha.

Atenção

Modificar ou excluir esses arquivos incorretamente pode bloquear você e impedir que acesse seu próprio sistema. Fique-se de testar as alterações em um ambiente virtual ou de teste antes de modificar os servidores de distribuição Linux.

Esses arquivos de configuração de evento de sistema PAM funcionam exatamente da mesma maneira que os arquivos de configuração de aplicativo compatíveis com PAM. Eles têm o mesmo formato, usam a mesma sintaxe e chamam os módulos PAM. Mas muitos desses arquivos estão simbolicamente vinculados (ver Capítulo 4, “Movendo-se pelo sistema de arquivos”). Portanto, esses arquivos de configuração exigem alguns passos extras quando são feitas alterações neles. Os procedimentos “como fazer” são discutidos mais adiante, neste capítulo.

Ca

Alguns dos arquivos de configuração PAM têm uma página `man` associada a eles. Por exemplo, para encontrar informações adicionais sobre o módulo `pam_unix`, digite `man pam_unix` na linha de comando da distribuição Fedora e RHEL.

Embora o Linux venha com muitos aplicativos compatíveis com PAM, diversos arquivos de configuração e módulos PAM já instalados, você não pode simplesmente esperar que o PAM vá cuidar de si mesmo. Certas etapas administrativas são necessárias para gerenciar o PAM.

Administrando o PAM no sistema Linux

A tarefa de administrar o PAM no sistema Linux é relativamente mínima. Você precisará verificar se o PAM está adequadamente implementado e fazer ajustes para atender às necessidades de segurança da organização específica.

Além disso, o PAM vai um pouco além dos simples passos de autenticação de aplicativo descritos anteriormente. O PAM também pode limitar recursos, restringir os tempos de acesso, impor uma boa seleção de senha etc.

Gerenciando arquivos de configuração de aplicativos compatíveis com PAM

Você deve revisar os arquivos de configuração PAM para seus aplicativos e utilitários compatíveis com PAM para garantir que o processo de autenticação corresponde ao processo de autenticação desejado da sua organização. Sua matriz de controle de acesso (ver Capítulo 22, “Entendendo a segurança básica do Linux”) e as informações sobre o PAM fornecidas neste capítulo devem ajudá-lo a realizar uma auditoria dos arquivos de configuração PAM.

Cada aplicativo compatível com PAM deve ter um arquivo próprio de configuração PAM. Cada arquivo de configuração define quais módulos PAM específicos serão utilizados para esse aplicativo. Se não existir nenhum arquivo de configuração, uma falha de segurança poderia ser criada para esse aplicativo. Essa falha poderia ser usada para intenções maliciosas. Como uma medida de segurança, o PAM vem com o arquivo de configuração “other”. Se um aplicativo compatível com PAM não tiver um arquivo de configuração PAM, ele usa, por padrão, o arquivo de configuração PAM “other”.

Você pode verificar se o sistema Linux tem o arquivo de configuração `/etc/pam.d/other` usando o comando `ls`. O exemplo a seguir mostra que o arquivo de configuração PAM `/etc/pam.d/other` existe nesse sistema.

```
$ ls /etc/pam.d/other
/etc/pam.d/other
```

O arquivo de configuração PAM `/etc/pam.d/other` deve negar todos os acessos, o que em termos de segurança é chamado Implicit Deny. No controle de acesso de segurança do computador, *Implicit Deny* significa que, se determinados critérios não forem claramente satisfeitos, o acesso deverá ser negado. Nesse caso, se nenhum arquivo de configuração existir para um aplicativo PAM, todo o acesso a ele será negado. A seguir, você pode ver o conteúdo de um arquivo `/etc/pam.d/other`:

```
$ cat /etc/pam.d/other
#%PAM-1.0
auth      required      pam_deny.so
account   required      pam_deny.so
password  required      pam_deny.so
session   required      pam_deny.so
```

Observe que todos os quatro contextos PAM — `auth`, `account`, `password` e `session` — são listados. Cada contexto usa o flag de controle `required` e o módulo `pam_deny.so`. O módulo PAM `pam_deny.so` é usado para negar o acesso.

Mesmo com o arquivo de configuração “other” correto, se um arquivo de configuração PAM para um aplicativo compatível com PAM não estiver aí, ele precisará ser criado. Adicione esse item à sua lista de verificação de auditoria PAM. Você também deve analisar o arquivo de configuração PAM “other” no sistema Linux para garantir que impõe Implicit Deny.

Gerenciando os arquivos de configuração de eventos de sistema PAM

Similar aos arquivos de configuração de aplicativo e utilitário compatíveis com PAM, seus arquivos de configuração de eventos de sistema PAM terão de ser auditados com a matriz de controle de acesso da sua organização. Mas para quaisquer modificações necessárias nesses arquivos, há passos adicionais que precisam ser seguidos.

No material a seguir, você aprenderá a configurar os requisitos especiais de segurança via PAM no sistema Linux, como restrições de tempo de login de conta. Muitos dos requisitos especiais exigirão que você faça uma alteração nos arquivos de configuração de eventos de sistema PAM, como `/etc/pam.d/system-auth-ac`.

O problema ao fazer mudanças em alguns desses arquivos de configuração de eventos de sistema PAM é que o utilitário `authconfig` pode reescrever esses arquivos e remover quaisquer alterações feitas localmente. Felizmente, cada arquivo de configuração PAM que corre esse risco está documentado em uma linha de comentário. Usando `grep`, você pode rapidamente encontrar os arquivos de configuração PAM que têm esse potencial problema.

```
# grep "authconfig" /etc/pam.d/*
/etc/pam.d/fingerprint-auth:# User changes will be destroyed
the next time authconfig is run.
/etc/pam.d/fingerprint-auth-ac:# User changes will be destroyed
the next time authconfig is run.
...
/etc/pam.d/system-auth:# User changes will be destroyed
the next time authconfig is run.
/etc/pam.d/system-auth-ac:# User changes will be destroyed
the next time authconfig is run.
...
```

Esses arquivos de configuração de eventos de sistema PAM usam links simbólicos (ver Capítulo 4, “Movendo-se pelo sistema de arquivos”). Por exemplo, você pode ver que o arquivo `system-auth`, na verdade, é um vínculo simbólico apontando para o arquivo `system-auth-ac`. O primeiro caractere na segurança do arquivo é um `l`. Isso indica que o arquivo está vinculado. O símbolo `->` mostra que o arquivo está vinculado simbolicamente.

```
# ls -l system-auth
lrwxrwxrwx. 1 root root 14 Feb 28 01:36
system-auth -> system-auth-ac
```

ta

todas as distribuições do Linux, como o Ubuntu, têm o utilitário `authconfig`, que sobrescreverá os arquivos de configuração PAM. Para verificar se sua distribuição tem o utilitário `authconfig`, digite `which authconfig` na linha de comando. Se nada retornar, sua distribuição do Linux não tem esse utilitário.

Em algumas distribuições do Linux, o utilitário `pam-auth-config` é semelhante ao utilitário `authconfig` devido à sua capacidade de sobrescrever os arquivos de configuração. Isso pode acontecer se o comando `pam-auth-config --force` for digitado na linha de comando. Leia a página `man pam-auth-config` para entender mais sobre esse utilitário se ele estiver instalado no seu sistema.

O utilitário `authconfig` não usa os links simbólicos, nem irá modificá-los. Assim, você pode criar um

novo arquivo de configuração local de eventos de sistema PAM e apontar o vínculo simbólico para ele. Isso permitirá que seu sistema tenha as modificações necessárias de segurança implementadas e evitará que os arquivos de configuração sejam sobrescritos pelo utilitário `authconfig`. Os passos básicos são os seguintes, incluindo um exemplo de como executar esses passos para o arquivo `system-auth-ac`:

1. **Copie o atual arquivo de configuração de eventos de sistema PAM para um novo arquivo adicionando um novo final de nome de arquivo, como “local”.**

```
# cp system-auth-ac system-auth-local
```
2. **Faça as alterações necessárias no novo arquivo de configuração.**

```
# vi system-auth-local
```
3. **Remova o antigo arquivo de vínculo simbólico.**

```
# ls -l system-auth
lrwxrwxrwx. 1 root root 14 Feb 28 01:36
system-auth -> system-auth-ac
#
# rm -i system-auth
rm: remove symbolic link 'system-auth'? y
```
4. **Crie um novo vínculo simbólico apontando-o para o novo arquivo de configuração.**

```
# ln -s system-auth-local system-auth
#
# ls -l system-auth
lrwxrwxrwx. 1 root root 17 Feb 28 01:37
system-auth -> system-auth-local
```

Depois que essas alterações foram feitas, você pode fazer as alterações necessárias nos arquivos de configuração “locais” de eventos de sistema PAM sem se preocupar com o fato de que o utilitário `authconfig` sobrescreverá os arquivos.

Implementando limites de recursos com PAM

O gerenciamento de recursos não é apenas uma tarefa administrativa de sistema. Também é uma tarefa administrativa de segurança. Definir limites de recursos ajuda a evitar muitos problemas adversos no sistema Linux. Problemas como bombas de fork podem ser evitados limitando o número de processos que um único usuário pode criar. Uma *bomba de fork* ocorre quando um processo gera um processo após o outro de uma maneira recursiva até que os recursos do sistema sejam consumidos. Bombas de fork podem ser maliciosas ou apenas acidentais — como aquelas criadas por desenvolvimento ruim de código de programa.

O módulo PAM `pam-limits` usa um arquivo de configuração especial para definir esses limites de recursos, `/etc/security/limits.conf`. Por padrão, esse arquivo não tem limites de recursos definidos dentro dele. Portanto, você terá de revisar o arquivo e definir limites de recurso para atender as necessidades de segurança da organização.

ivos de configuração PAM estão no diretório `/etc/pam.d` e no diretório `/etc/security`.

O trecho de código a seguir mostra o arquivo `/etc/security/limits.conf`. Como você pode ver, o arquivo está bem documentado, incluindo descrições completas de formato e exemplos dos limites que podem ser configurados.

```
$ cat /etc/security/limits.conf
# /etc/security/limits.conf
# #
Each line describes a
limit for a
user in the form:
#
#<domain>      <type> <item> <value>
#
#Where:
#<domain> can be:
#   - an user name
#   - a group name, with @group syntax
#   - the wildcard *, for default entry
#   - the wildcard %, can be also used with %group syntax,
#       for maxlogin limit
#
#<type> can have the two values:
#   - "soft" for enforcing the soft limits
#   - "hard" for enforcing hard limits
#
```



```

#<item> can be one of the following:
# - core - limits the core file size (KB)
# - data - max data size (KB)
# - fsize - maximum filesize (KB)
# - memlock - max locked-in-memory address space (KB)
# - nofile - max number of open files
# - rss - max resident set size (KB)
# - stack - max stack size (KB)
# - cpu - max CPU time (MIN)
# - nproc - max number of processes
# - as - address space limit (KB)
# - maxlogins - max number of logins for this user
# - maxsyslogins - max number of logins on the system
# - priority - the priority to run user process with
# - locks - max number of file locks the user can hold
# - sigpending - max number of pending signals
# - msgqueue - max memory used by POSIX
message queues (bytes)
# - nice - max nice priority allowed to
raise to values: [-20, 19]
# - rtprio - max realtime priority
#
#<domain> <type> <item> <value>
#
#* soft core 0
#* hard rss 10000
#@student hard nproc 20
#@faculty soft nproc 20
#@faculty hard nproc 50
#ftp hard nproc 0
#@student - maxlogins 4

# End of file

```

Os itens de formato *domain* e *type* precisam ser explicados um pouco mais além do que está documentado no arquivo de configuração:

- **domain** O limite se aplica ao usuário ou grupo listado. Se o domínio for “*” ele será aplicado a todos os usuários.
- **type** Um limite `hard` não pode ser excedido. O limite `soft` pode ser excedido, mas apenas temporariamente.

Analise o exemplo do arquivo de configuração `limits.conf` a seguir. O grupo `faculty` está listado, mas o que você deve observar é `nproc`. O limite `nproc` define o número máximo de processos que um usuário pode iniciar. Essa configuração é o que impede uma bomba de `fork`. Observe que o `type` selecionado é `hard`; assim, o limite de 50 processos não pode ser excedido. Obviamente, esse limite não é imposto porque a linha está desativada com o caractere de comentário `#`.

```

#@faculty hard nproc 50

```

As definições dos limites são configuradas por login e existem apenas para a duração da sessão de login. Um usuário malicioso poderia fazer o login várias vezes para criar uma bomba de `fork`. Assim, definir o número máximo de logins para essas contas de usuário também é uma boa ideia.

Limitar o número máximo de logins talvez precise ser feito individualmente por usuário. Por exemplo, `johndoe` precisa fazer login no sistema Linux apenas uma vez. Para evitar que outras pessoas usem a conta de `johndoe`, configure o `maxlogins` dessa conta como 1.

O passo final para limitar esse recurso é assegurar que o módulo PAM utilizando `limits.conf` está incluído em um dos arquivos de configuração PAM de eventos de sistema. O módulo PAM utilizando `limits.conf` é `pam_limits`. Na listagem parcial a seguir, `grep` é utilizado para verificar se o módulo PAM é usado dentro dos arquivos de configuração de eventos de sistema.

```
# grep "pam_limits" /etc/pam.d/*
...
system-auth:session      required      pam_limits.so
system-auth-ac:session   required     pam_limits.so
system-auth-local:session required     pam_limits.so
```

Limites de tempo para o acesso a serviços e contas não são tratados pelo arquivo de configuração `/etc/security/limits.conf` PAM. Em vez disso, ele é tratado pelo arquivo `time.conf`.

Implementando restrições de tempo com PAM

O PAM pode fazer com que todo o sistema Linux opere em “tempo PAM”. Restrições de tempo, como acesso a determinados aplicativos durante horários específicos do dia ou permissões de login apenas durante os dias especificados da semana, são tratadas pelo PAM.

O arquivo de configuração PAM que lida com essas restrições está localizado no diretório `/etc/security`. O código a seguir mostra o arquivo de configuração PAM `/etc/security/time.conf` bem documentado.

```
$ cat /etc/security/time.conf
# this is an example configuration file for the pam_time module
...
# the syntax of the lines is as follows:
#
#      services;ttys;users;times
...
# services
# is a logic list of PAM service names that the rule applies to.
#
# ttys
# is a logic list of terminal names that this rule applies to.
#
```

```

# users
# is a logic list of users or a netgroup of users to whom this
# rule applies.
##
NB. For these items the simple wildcard '*' may be used
only once.
#
# times
# the format here is a logic list of day/time-range
# entries the days are specified by a sequence of two character
# entries, MoTuSa for example is Monday Tuesday and
Saturday. Note
# that repeated days are unset MoMo = no day, and
MoWk = all weekdays
# bar Monday. The two character combinations accepted are
#
#      Mo Tu We Th Fr Sa Su Wk Wd Al
#
# the last two being week-end days and all 7 days of the week
# respectively. As a final example, AlFr means all
days except Friday.
#
# each day/time-range can be prefixed with a '!' to
indicate "anything
# but"
#
# The time-range part is two 24-hour times HHMM separated
by a hyphen
# indicating the start and finish time (if the finish time
is smaller
# than the start time it is deemed to apply on the following
day).
#
# for a rule to be active, ALL of service+ttys+users must be
satisfied
# by the applying process.
...
# End of example file.

```

Observe que o formato para cada entrada válida vem depois da sintaxe a seguir: *serviços; ttys; usuários; horários*. Cada campo é separado por um ponto e vírgula. Os valores válidos de campo são documentados no arquivo de configuração `time.conf`.

Embora `time.conf` esteja bem documentado, um exemplo sempre é útil. Por exemplo, você decidiu que os usuários normais devem ter autorização para fazer login nos terminais apenas durante a semana (segunda a sexta). Eles podem fazer login entre 7 e 19 horas nesses dias da semana. A lista a seguir descreve os elementos que precisam ser definidos:

- *serviços* — Login
- *ttys* — * (Designando que todos os terminais devem ser incluídos)
- *usuários* — Todo mundo, exceto root (!root)
- *horários* — Permitido nos dias de semana (Wd) entre 7:00 (0700) e 19:00 (1900)

A entrada em `time.conf` se pareceria com o seguinte:

```
login; * ; !root ; Wd0700-1900
```

O último passo para implementar essa restrição de tempo de exemplo é assegurar que o módulo PAM

utilizando `time.conf` está incluído em um dos arquivos de configuração de eventos de sistema PAM. O módulo PAM utilizando `time.conf` é `pam_time`. Na lista parcial a seguir, `grep` mostra o módulo PAM; `pam_time` não é usado em nenhum dos arquivos de configuração de eventos de sistema.

```
# grep "pam_time" /etc/pam.d/*
config-util:auth          sufficient    pam_timestamp.so
config-util:session      optional    pam_timestamp.so
selinux-polgengui:auth   sufficient  pam_timestamp.so
selinux-polgengui:session optional    pam_timestamp.so
system-config-selinux:auth sufficient  pam_timestamp.so
system-config-selinux:session optional    pam_timestamp.so
```

Como `pam_time` não está listado acima, você deve modificar o arquivo `/etc/pam.d/system-auth` para que o PAM imponha as restrições de tempo. O arquivo de configuração PAM `system-auth` é usado pelo PAM no login do sistema e durante modificações de senha. Esse arquivo de configuração verifica vários itens, como restrições de tempo.

Adicione o seguinte perto da parte superior da seção “account” do arquivo de configuração. Agora o módulo `pam_time` verificará as restrições de login definidas dentro do arquivo `/etc/security/time.conf`.

```
account required pam_time.so
```

ta

Ubuntu, você precisará modificar o arquivo `/etc/pam.d/common-auth` em vez do arquivo de configuração `system-auth`.

Lembre-se de que `system-auth` é um arquivo vinculado simbolicamente. Se modificar esse arquivo, você precisará seguir passos adicionais para preservar as modificações a partir do utilitário `authconfig`. Revise a seção, “Gerenciando os arquivos de configuração de eventos de sistema PAM”, anteriormente, neste capítulo.

Há módulos e arquivos de configuração PAM adicionais que você pode empregar para definir ainda mais restrições nos sujeitos. Um módulo de segurança importante é `pam_cracklib`.

Impondo boas senhas com PAM

Quando uma senha é modificada, o módulo PAM `pam_cracklib` é envolvido no processo. O módulo solicita ao usuário uma senha e verifica sua força contra um dicionário de sistema e um conjunto de regras para identificar escolhas ruins.

ta

O módulo `pam_cracklib` é instalado por padrão no Fedora e no RHEL. Para sistemas Ubuntu Linux, ele não é instalado por padrão. Portanto, para ter acesso ao módulo `pam_cracklib` no Ubuntu, emita o comando `sudo apt-get install libpam-cracklib`.

Usando `pam_cracklib`, você pode verificar uma senha recém-escolhida considerando o seguinte:

- É uma palavra de dicionário?
- É um palíndromo?
- É a senha antiga com letras maiúsculas/minúsculas modificadas?
- É muito parecida com a senha antiga?
- É muito curta?
- É uma versão reciclada da senha antiga?
- Ela usa os mesmos caracteres consecutivos?
- Ela contém o nome de usuário de alguma forma?

Você pode mudar as regras que `pam_cracklib` usa para verificar novas senhas fazendo modificações no arquivo `/etc/pam.d/system-auth`. Você poderia achar que as mudanças devem ser feitas no arquivo de configuração `passwd` compatível com PAM. Mas o arquivo `/etc/pam.d/passwd` inclui o arquivo `systemauth` na pilha.

```
# cat /etc/pam.d/passwd
#%PAM-1.0
auth      include      system-auth
account   include      system-auth
password  substack        system-auth
-password optional     pam_gnome_keyring.so use_authtok
password  substack        postlogin
```

ta

Ubuntu, você precisará modificar o arquivo `/etc/pam.d/common-password`, em vez do arquivo de configuração `system-auth`.

As configurações atuais do arquivo `system-auth` são mostradas aqui. Você pode ver que atualmente não há uma entrada que chama o módulo PAM `pam_cracklib`.

```
# cat /etc/pam.d/system-auth
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth      required     pam_env.so
auth      sufficient   pam_fprintd.so
```

```

auth      sufficient  pam_unix.so nullok try_first_pass
auth      requisite  pam_succeed_if.so uid >= 1000 quiet
auth      required   pam_deny.so

account   required   pam_unix.so
account   sufficient pam_localuser.so
account   sufficient pam_succeed_if.so uid < 1000 quiet
account   required   pam_permit.so

password requisite  pam_cracklib.so try_first_pass retry=3
...

```

A entrada `pam_cracklib` na listagem anterior usa a palavra-chave `retry`. Abaixo, há uma lista das várias palavras-chave disponíveis para `cracklib`.

- **retry=N**

- Default = 1
- Solicita o usuário no máximo *N* vezes antes de retornar com um erro.

- **difok=N**

- Default = 5
- O número de caracteres na nova senha que não devem estar presentes na senha antiga.
- Exceção 1: Se metade dos caracteres na nova senha for diferente, então a nova senha será aceita.
- Exceção 2: Ver `difignore`.

- **difignore=N**

- Default = 23
- O número de caracteres que a senha tem antes que a configuração `difok` seja ignorada.

- **minlen=N**

- Default = 9
- O tamanho mínimo aceitável para a nova senha.
- Ver em `dcredit`, `ucredit`, `lcredit` e `ocredit` como essas configurações afetam `minlen`.

- **dcredit=N**

- Default = 1
- Se ($N \geq 0$): O número máximo de dígitos na nova senha. Se houver menos de ou *N* dígitos, cada dígito será contado +1 para satisfazer o valor `minlen` atual.
- Se ($N < 0$): O número mínimo de dígitos que uma nova senha deve ter.

- **ucredit=N**

- Default = 1
- Se ($N \geq 0$): O número máximo de letras maiúsculas da nova senha. Se houver menos de ou *N*

letras maiúsculas, cada letra será contada +1 para satisfazer o valor `minlen` atual.

- Se ($N < 0$): O número mínimo de letras maiúsculas que uma nova senha deve ter.

■ **lcredit=N**

- Default = 1

- Se ($N \geq 0$): O número máximo de letras minúsculas da nova senha. Se houver menos que ou N letras minúsculas, cada letra será contada +1 para satisfazer o valor `minlen` atual.

- Se ($N < 0$): O número mínimo de letras minúsculas de uma nova senha.

■ **ocredit=N**

- Default = 1

- Se ($N \geq 0$): O número máximo de outros caracteres da nova senha. Se houver menos do que ou N outros caracteres, cada caractere será contado +1 para satisfazer o valor `minlen` atual.

- Se ($N < 0$): O número mínimo de outros caracteres de uma nova senha.

■ **minclass=N**

- Default = 0

- N classes de quatro classes de caracteres são necessárias para a nova senha. As quatro classes são números, letras maiúsculas, letras minúsculas e outros caracteres.

■ **maxrepeat=N**

- Default = 0

- Rejeita senhas que contêm mais que N caracteres iguais consecutivos.

■ **reject_username**

Verifica se o nome do usuário está contido na nova senha (na ordem direta ou invertida). Se ele for encontrado, a nova senha será rejeitada.

■ **try_first_pass**

Tenta obter a senha a partir de um módulo PAM anterior. Se isso não funcionar, solicitará ao usuário a senha.

■ **use_authok**

Esse argumento é utilizado para forçar (*force*) o módulo a não solicitar ao usuário uma nova senha. Em vez disso, a nova senha é fornecida pelo módulo *password* previamente empilhado.

■ **dictpath=/caminho**

O caminho para os dicionários `cracklib`.

Por exemplo, se sua organização exigir que as senhas tenham 10 caracteres de tamanho que contenham dois dígitos, você adicionaria uma linha semelhante ao seguinte ao arquivo `/etc/pam.d/system-`

auth:

```
password required pam_cracklib.so minlen=10 dcredit=-2
```

As palavras-chave utilizadas neste exemplo com `pam_cracklib` são:

- `minlen=10` A nova senha deve ter pelo menos 10 caracteres.
- `dcredit=-2` A nova senha deve conter dois números.

Restrições `pam_cracklib` não se aplicam ao usuário `root`.

Incentivando o uso de `sudo` com PAM

Para permitir o monitoramento do uso da conta `root` pelas pessoas e evitar uma situação de repúdio (ver Capítulo 22, “Entendendo segurança básica no Linux”), é melhor restringir o uso do comando `su` e incentivar o uso de `sudo`. Se sua organização tiver uma política assim, você pode alcançar isso com PAM em apenas alguns passos.

O comando `su` é compatível com PAM, o que simplifica muito as coisas. Ele usa o módulo PAM `pam_wheel` para verificar se há usuários no grupo `wheel`. O arquivo de configuração `/etc/pam.d/su` é mostrado aqui:

```
# cat /etc/pam.d/su
##PAM-1.0
auth sufficient pam_rootok.so
# Uncomment the following line to implicitly trust users
# in the "wheel" group.
#auth sufficient pam_wheel.so trust use_uid
# Uncomment the following line to require a user to be
# in the "wheel" group.
#auth required pam_wheel.so use_uid
auth include system-auth
auth include postlogin
account sufficient pam_succeed_if.so uid = 0 use_uid quiet
account include system-auth
password include system-auth
session include system-auth
session include postlogin
session optional pam_xauth.so
```

Primeiro, para restringir o uso de `su`, se você estiver usando o grupo `wheel` como o grupo administrativo, você precisará reatribuir o grupo administrativo para um novo grupo (ver Capítulo 11, “Gerenciando contas de usuário”). Se você não estiver usando o grupo `wheel`, apenas certifique-se de não atribuir futuramente nenhuma pessoa a esse grupo.

Em seguida, você precisa editar o arquivo de configuração `/etc/pam.d/su`. Remova a marca de comentário, `#`, da seguinte linha:

```
#auth required pam_wheel.so use_uid
```


Com essas modificações, o PAM desativará o uso do comando `su`. Usuários administrativos agora terão de usar `sudo`, que o sistema monitora e fornece um ambiente desejado de não repúdio (ver Capítulo 22, “Entendendo segurança básica no Linux”).

Bloqueando contas com PAM

Os requisitos específicos de segurança da sua organização poderiam exigir o bloqueio de uma conta de usuário após um determinado número de tentativas de login. O padrão típico é bloquear uma conta depois de três tentativas malsucedidas. Isso acontece para evitar um ataque de senha de força bruta contra uma conta.

O módulo PAM usado para gerenciar tentativas de login é `pam_tally2`. O arquivo de configuração PAM a editar é `/etc/pam.d/system-auth`.

ta
distribuições mais antigas do Linux talvez usem o módulo PAM `pam_tally` em vez de `pam_tally2`.

Mais uma vez, você deve fazer essas alterações no arquivo local `system-auth-local`, em vez de `system-auth-ac`, porque `authconfig` sobrescreverá as modificações na próxima vez que for executado. As duas linhas que você precisa adicionar a `system-auth-local` estão destacadas no código a seguir. Sua colocação nesse arquivo é extremamente importante.

```
# cat system-auth-local
#%PAM-1.0
# Local system-auth file.
# Changes will not be destroyed by authconfig
auth      required      pam_tally2.so deny=3 quiet
auth      required      pam_env.so
auth      sufficient    pam_fprintd.so
auth      sufficient    pam_unix.so nullok try_first_pass
auth      requisite     pam_succeed_if.so uid >= 1000 quiet
auth      required      pam_deny.so

account   required      pam_tally2.so
account   required      pam_unix.so
account   sufficient    pam_localuser.so
account   sufficient    pam_succeed_if.so uid < 1000 quiet
account   required      pam_permit.so
...
```

enção
uma cópia de backup de `system-auth-local` e teste as alterações em um ambiente de teste antes de fazer alterações no sistema de produção Linux. Uma modificação incorreta poderia bloquear todo mundo do seu sistema, incluindo o usuário `root`.

Na primeira linha de contexto `auth` envolvendo `pam_tally.so` no código anterior, observe que duas opções foram adicionadas, `deny=3` e `quiet`. A opção `deny=3` permitirá a um usuário apenas três tentativas de login malsucedidas antes de a conta ser bloqueada.

Se a conta for bloqueada, a opção `quiet` não informará isso ao usuário ela simplesmente continuará fornecendo mensagens “senha incorreta”. Não deixar o usuário saber o que aconteceu com uma conta é útil se você estiver sendo atacado de forma maliciosa. O atacante malicioso não saberá que a conta foi bloqueada e achará que ele acabou de inserir outra senha incorreta. Isso pode lhe dar tempo para monitorar o que está acontecendo.

Usar a opção `quiet` pode, porém, causar muitos problemas para os usuários. Por exemplo, um usuário talvez não perceba que ele inseriu a senha errada um número suficiente de vezes para bloquear a conta. Isso pode causar um atraso na procura de ajuda. Você pode remover a opção `quiet` do arquivo de configuração de modo que, quando um usuário fizer muitas tentativas de login malsucedidas, ele receberá uma mensagem como “Conta bloqueada devido a 4 logins malsucedidos”.

ta

Ubuntu, em vez do arquivo de configuração `system-auth`, você precisará adicionar as informações de contexto `auth` ao arquivo `/etc/pam.d/commonauth` e adicionar informações de contexto `account` ao arquivo `/etc/pam.d/common-account`.

O módulo `pam_tally2` também inclui uma interface de linha de comando que você pode usar para monitorar as tentativas de login malsucedidas. Se o módulo `pam_tally2` estiver incluído em um dos seus arquivos de configuração de eventos de sistema PAM, ele manterá um registro em log de quantas tentativas de login malsucedidas ocorreram no sistema. Para ver essas falhas, digite o comando `pam_tally2`, como mostrado no seguinte código:

```
# pam_tally2
Login          Failures Latest failure    From
Samantha      2        03/10/15 06:24:01    pts/1
```

O nome de usuário, número de falhas e a última tentativa são listados juntamente com o terminal em que a falha mais recente ocorreu. Você também pode usar o comando `pam_tally2` para desbloquear uma conta de usuário depois que ela foi bloqueada pelo módulo `pam_tally2` PAM.

Quando uma conta é bloqueada pelo PAM, ela não é listada como bloqueada no arquivo `/etc/shadow` e você não pode desbloqueá-la usando o comando `usermod -U nome_do_usuario`. Para desbloqueá-la, você precisa usar o comando `pam_tally2`.

No exemplo a seguir, a conta da usuária `Samantha` teve muitas tentativas de login malsucedidas. Mas a conta não é listada como bloqueada no arquivo `/etc/shadow`, mostrado pelo comando `passwd`. Bloquear a conta usando o comando `usermod -L` fará com que ela seja bloqueada via o arquivo `/etc/shadow`, não via o PAM.

```
# pam_tally2
Login          Failures Latest failure    From
Samantha      5      03/10/15 06:32:24 pts/1
# passwd -S Samantha
Samantha PS 2015-03-09 0 99999 7 -1 (Password set, SHA512 crypt.)
#
# usermod -L Samantha
# passwd -S Samantha
Samantha LK 2015-03-09 0 99999 7 -1 (Password locked.)
#
# usermod -U Samantha
# passwd -S Samantha
Samantha PS 2015-03-09 0 99999 7 -1 (Password set, SHA512 crypt.)
```

No código a seguir, o comando `pam_tally2 -r -u Samantha` é emitido para desbloquear a conta da usuária Samantha. Observe que o comando `pam_tally2` lista novamente o número de tentativas de login malsucedidas uma vez que ele elimina o “bloqueio”. Quando o comando `pam_tally2` é emitido novamente, os registros das tentativas malsucedidas da usuária Samantha foram removidos, porque o bloqueio foi removido.

```
# pam_tally2 -r -u Samantha
Login          Failures Latest failure    From
Samantha      5      03/10/15 06:34:09 pts/1
# pam_tally2
#
```

Há muitas outras opções que você pode usar com `pam_tally2`. Para explorar esse módulo PAM ainda mais, emita o comando `pam_tally2 man` na linha de comando.

Obtendo mais informações sobre o PAM

PAM é outra ferramenta de segurança rica e versátil disponível para você no sistema Linux. Nas páginas `man` do sistema Linux, você pode ler sobre como gerenciar os arquivos de configuração PAM e sobre os módulos no diretório `/lib/security`.

- Para obter mais informações sobre os arquivos de configuração PAM, use o comando **`man pam.conf`**.
- Você pode ver todos os módulos PAM disponíveis no seu sistema digitando **`ls /lib/security/pam*.so`** na linha de comando. Para obter mais informações sobre cada módulo PAM, digite **`man pam_nome_do_módulo_pam`**. Certifique-se de remover a extensão de arquivo “so” para o `nome_do_módulo_pam`. Por exemplo, digite **`man pam_lastlog`** para aprender mais sobre o módulo `pam_lastlog.so`.

Há também vários sites que podem fornecer informações adicionais sobre o PAM:

- O site web Linux-PAM Oficial: <http://linux-pam.org>
- O Guia do Administrador do Sistema PAM do Linux: http://linux-pam.org/Linux-PAM-html/Linux-PAM_SAG.html
- Referência do módulo PAM:

Resumo

As ferramentas de criptografia e o PAM devem ser tratados com cuidado à medida que você aprende sobre o Linux. Certifique-se de testar todas as modificações que você faz em um sistema Linux de teste ou em um sistema Linux virtualizado antes de implementá-las em uma máquina de produção.

Antes de adicionar essas ferramentas avançadas de criptografia e o PAM à fase de implementação do ciclo de vida do processo de segurança da organização, revise as várias opções que elas oferecem. Depois de determinar de quais recursos dessas ferramentas avançadas sua organização precisa, adicione-os à fase de implementação.

O próximo capítulo discute o SELinux. Embora criptografia e PAM sejam ferramentas que você pode usar no sistema Linux, o SELinux é uma camada inteira de aprimoramento de segurança.

Exercícios

Use esses exercícios para testar seu conhecimento sobre o uso das ferramentas de criptografia e o PAM. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostrados no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. Criptografe um arquivo usando o utilitário `gpg` e uma chave simétrica.
2. Gere um chaveiro usando o utilitário `gpg`.
3. Liste o chaveiro que você gerou.
4. Criptografe um arquivo e adicione sua assinatura digital usando o utilitário `gpg`.
5. Acesse a página inicial Linux Mint em `www.linuxmint.com`. A partir da página Download, selecione uma das distribuições do Linux Mint para download. Quando o download estiver completo, use o utilitário apropriado de resumo de mensagem para garantir que o arquivo baixado não foi corrompido.
6. Usando o comando `which su`, determine o nome completo de arquivo do comando `su`. Então, determine se o comando `su` no seu sistema Linux é compatível com PAM.
7. O comando `su` tem um arquivo de configuração PAM? Se tiver, exiba-o na tela e liste quais contextos PAM ele utiliza.
8. Liste na sua tela os vários módulos PAM no seu sistema.
9. Encontre o arquivo de configuração PAM “other” no seu sistema. Ele existe? Ele impõe um Implicit Deny?
10. Encontre o arquivo de configuração de limites do PAM. Ele tem uma definição para evitar

que uma bomba de fork ocorra no seu sistema?

Aprimorando a segurança do Linux com o SELinux

NESTE CAPÍTULO

Entendendo os benefícios do SELinux

Aprendendo como o SELinux funciona

Configurando o SELinux

Corrigindo problemas com o SELinux

Obtendo informações adicionais sobre o SELinux

O Security Enhanced Linux (SELinux), isto é, Linux de Segurança Reforçada, foi desenvolvido pela National Security Agency (NSA) juntamente com outras organizações de pesquisa de segurança, como a Secure Computing Corporation (SCC). O SELinux foi lançado para a comunidade de código aberto em 2000 e tornou-se popular quando o Red Hat começou a implementar o SELinux como um pacote padrão. Agora, o SELinux é usado por muitas organizações e está amplamente disponível.

Entendendo os benefícios do SELinux

O SELinux é um módulo de aprimoramento de segurança implantado sobre o Linux. Ele fornece medidas de segurança adicionais e é incluído por padrão no RHEL e no Fedora.

O SELinux oferece melhor segurança para o sistema Linux via controle de acesso baseado em papéis (RBAC), em sujeitos e objetos (conhecidos como processos e recursos). A segurança “tradicional” do Linux usa controles de acesso discricionário (*Discretionary Access Controls, DAC*).

ta

de parte de documentação do SELinux afirma que o SELinux usa o modelo de controle de acesso mandatório (*Mandatory Access Control, MAC*). Lembre-se de que o RBAC é uma versão refinada do MAC (ver Capítulo 22, “Entendendo a segurança básica do Linux”). Assim, a documentação está correta quando diz que o SELinux usa MAC.

O SELinux não é um substituto para o DAC. Em vez disso, ele é uma camada de segurança adicional.

- Regras DAC ainda são utilizadas ao usar o SELinux.
- Regras DAC são verificadas primeiro e, se o acesso for permitido, então as políticas do SELinux são verificadas.
- Se as regras DAC negarem o acesso, as políticas do SELinux não são revisadas.

Se um usuário tentar executar um arquivo ao qual ele não tem acesso de execução ($rxw-$), os controles DAC “tradicionais” do Linux negarão o acesso. Portanto, as políticas do SELinux nem mesmo serão verificadas.

ta

SELinux é o aprimoramento de segurança padrão das distribuições Red Hat e o AppArmor é o aprimoramento de segurança padrão para o Ubuntu. Você ainda pode instalar o SELinux no Ubuntu usando o comando `sudo apt-get install selinux` e depois reiniciar. Se quiser saber mais sobre o AppArmor, acesse <https://help.ubuntu.com/community/AppArmor>.

Embora controles “tradicionais” de segurança do Linux ainda funcionem, há vários benefícios para o uso do SELinux, como:

- **Implementar o modelo de controle de acesso RBAC** — Esse é considerado o modelo mais forte de controle de acesso.
- **Usar o acesso de menor privilégio para sujeitos (por exemplo, usuários e processos)** — O termo *menor privilégio* significa que cada sujeito recebe um conjunto limitado de privilégios que são apenas suficientes para permitir que o sujeito seja funcional nas suas tarefas. Com o menor privilégio implementado, os danos acidentais (ou propositais) que um usuário ou processo pode causar nos objetos são limitados.
- **Permite colocar o processo em uma caixa de areia** — O termo *caixa de areia* significa que cada processo é executado em uma área própria (a caixa de areia). Ele não pode acessar outros processos ou seus arquivos, a menos que permissões especiais sejam concedidas. Essas áreas onde os processos são executados são chamadas “domínios”.
- **Permite que seja feito um teste da sua funcionalidade antes da implementação** — O SELinux tem um modo permissivo, que permite que você veja o efeito da aplicação do SELinux no sistema.

Outra maneira de analisar os benefícios do SELinux é examinar o que pode acontecer se ele não estiver em execução no sistema Linux. Por exemplo, o daemon do servidor web (`httpd`) ouve algo acontecer em uma porta. Uma simples solicitação de um navegador web chega para visualizar a página inicial. Passando por sua rotina normal, o daemon `httpd` ouve a solicitação e apenas a segurança “tradicional” do Linux é aplicada. Sendo irrestrito pelo SELinux, `httpd` é capaz de:

- Acessar *qualquer* arquivo ou diretório com base em permissões de leitura/gravação/execução para o proprietário e grupo associados.

- Realizar atividades potencialmente inseguras, como permitir carregar um arquivo ou alterar os limites do sistema.
- Ouvir solicitações de entrada em qualquer porta que ele quiser.

Em um sistema restringido pelo SELinux, o daemon `httpd` é muito mais bem controlado. Usando o exemplo anterior, `httpd` só pode ouvir na porta em que o SELinux permite que ele ouça. Em essência, o SELinux limita severamente códigos e atividades maliciosos no sistema Linux.

Entendendo como o SELinux funciona

O SELinux pode ser comparado a um guarda em uma porta: Nessa comparação, o sujeito (o usuário) quer acessar o objeto (o arquivo) dentro da sala. Para ter acesso a esse objeto:

1. O sujeito deve apresentar um crachá de identificação para o guarda.
2. O guarda analisa o crachá de identificação e as regras de acesso mantidas em um grande manual.
 - Se as regras de acesso permitirem que esse crachá de identificação especial abra a porta, o sujeito poderá entrar na sala para acessar o objeto.
 - Se as regras de acesso não permitirem que esse crachá de identificação especial acesse o objeto, então o guarda não o deixará entrar.

O SELinux fornece uma combinação de controle de acesso baseada em papéis (*Role Based Access Control*, RBAC) e *Type Enforcement (TE)* ou *Multi-Level Security (MLS)*. No controle de acesso baseado em papéis (ver Capítulo 22), o acesso a um objeto baseia-se no papel atribuído a um sujeito na organização. Portanto, ele não se baseia no nome de usuário do sujeito ou ID de processo. Cada papel recebe direitos de acesso.

Entendendo o Type Enforcement

O *Type Enforcement (TE)* é necessário para implementar o modelo RBAC e *ele* protege um sistema:

- Rotulando objetos como certos tipos de segurança
- Atribuindo aos sujeitos domínios e papéis específicos
- Fornecendo regras que permitem que certos domínios e papéis acessem certos tipos de objetos

O exemplo a seguir usa o comando `ls -l` para mostrar os controles DAC no arquivo `my_stuff`. O arquivo tem proprietários e grupo listados, bem como suas atribuições de leitura, gravação e execução. Se você precisar revisar as permissões de arquivo, consulte o Capítulo 4, “Movendo-se pelo sistema de arquivos”.

```
$ ls -l my_stuff
-rw-rw-r--. 1 johndoe johndoe 0 Feb 12 06:57 my_stuff
```


O exemplo a seguir inclui `ls -Z` e o mesmo arquivo, `my_stuff`, mas, em vez de apenas os controles DAC, a opção `-Z` também exibe os controles de segurança RBAC do SELinux.

```
$ ls -Z my_stuff
-rw-rw-r--. johndoe johndoe
unconfined_u:object_r:user_home_t:s0 my_stuff
```

O exemplo `ls -Z` exibe quatro itens a mais do que o comando `ls -l`:

- Um usuário (`unconfined_u`)
- Um papel (`object_r`)
- Um tipo (`user_home_t`)
- Um nível (`s0`)

Esses quatro itens RBAC (usuário, papel, tipo e nível) são usados no controle de acesso do SELinux para determinar os níveis apropriados de acesso. Juntos, os itens são chamados *contexto de segurança* do SELinux. Um contexto de segurança (crachá de identificação) é às vezes chamado “rótulo de segurança”.

Essas atribuições de contexto de segurança são dadas aos sujeitos (processos e usuários). Cada contexto de segurança tem um nome específico. O nome dado depende a qual objeto ou sujeito ele foi atribuído: Os arquivos têm um contexto de arquivo, os usuários têm um contexto de usuário e os processos têm um contexto de processo, também chamado de “domínio”.

As regras que permitem o acesso são chamadas “regras de permissão” ou “regras de política”. A *regra de política* é o processo que o SELinux segue para conceder ou negar acesso a um tipo particular de segurança do sistema. Voltando à comparação do SELinux com o guarda, o SELinux funciona como o guarda que tem de ver o contexto de segurança do sujeito (o crachá de identificação) e revisar as regras de política (manual das regras de acesso) antes de permitir ou negar o acesso a um objeto. Assim, o Type Enforcement garante que apenas alguns “tipos” dos sujeitos possam acessar certos “tipos” de objetos.

Entendendo a Multi-Level Security

Com o SELinux, você pode escolher Multi-Level Security, que usa o Type Enforcement, juntamente com o recurso adicional de autorizações de segurança. Ele também oferece Multi-Category Security, que fornece os níveis de classificação para objetos.

ca

omes da Multi-Level Security (MLS) podem causar confusão. Multi-Category Security (MCS) é às vezes chamada Multi-Clearance Security. Como a MLS oferece MCS, às vezes ele é chamado de MLS/MCS.

A Multi-Level Security impõe o modelo de segurança Bell-LaPadula Mandatory Access. O modelo Bell-LaPadula foi desenvolvido pelo governo dos EUA para impor a confidencialidade das informações. A aplicação desse modelo é alcançada concedendo acesso a objetos com base na autorização de segurança

do papel e nível de classificação do objeto. *Autorização de segurança* é um atributo concedido aos papéis permitindo acesso aos objetos classificados. *Nível de classificação* é um atributo concedido a um objeto, fornecendo proteção contra sujeitos que têm um atributo de autorização de segurança que é muito baixo. Você provavelmente já ouviu falar do nível de classificação “Top Secret” (“ultrassecreto”). O personagem de livros e filmes de ficção James Bond tinha uma autorização de segurança ultrassecreta, que lhe concedia acesso a informações classificadas como ultrassecretas. Isso é um uso clássico do modelo Bell-LaPadula.

A combinação de RBAC juntamente com Type Enforcement (TE) ou Multi-Level Security (MLS) permite ao SELinux fornecer uma melhoria de segurança muito forte. O SELinux também oferece diferentes modos operacionais para sua utilização.

Implementando modelos de segurança do SELinux

O modelo controle de acesso baseado em papéis e os modelos Type Enforcement, Multi-Level Security e Bell-LaPadula são temas interessantes. O SELinux implementa esses modelos por meio de uma combinação de quatro partes principais do SELinux:

- Modos operacionais
- Contextos de segurança
- Tipos de política
- Regra de pacotes de políticas

Embora alguns desses elementos de design já tenham sido discutidos brevemente, a discussão a seguir lhe dará um entendimento detalhado deles. Esse entendimento é necessário antes de você começar a configurar o SELinux no sistema.

Entendendo os modos operacionais do SELinux

O SELinux vem com três modos operacionais: Disabled, Permissive e Enforcing. Cada um desses modos oferece diversos benefícios para a segurança do sistema Linux.

Usando o modo Disabled

No modo Disabled, o SELinux é desativado. O método padrão de controle de acesso, Discretionary Access Control (DAC), é usado em seu lugar. Esse modo é útil para situações em que a segurança reforçada não é necessária.

ca

Quando com que você importa é desativar o SELinux, você encontrou a resposta. Basta editar o arquivo de configuração `/etc/selinux/config` e alterar o texto `SELINUX=` para o seguinte: `SELINUX=disabled`. O SELinux será desativado após a reinicialização do sistema. Agora você pode pular o restante deste capítulo.

Usando o modo Permissive

No modo permissivo, o SELinux é ativado, mas as regras de política de segurança não são aplicadas. Quando uma regra de política de segurança deve negar a admissão, o acesso ainda é permitido. Mas uma mensagem é enviada para um arquivo de log indicando que o acesso deve ser negado.

O modo Permissive do SELinux é usado para o seguinte:

- Auditar as regras atuais de política do SELinux
- Testar novos aplicativos para ver quais efeitos as regras de política do SELinux terão sobre eles
- Testar novas regras de política do SELinux para ver o efeito que as novas regras terão sobre os serviços e aplicativos atuais
- Determinar por que um determinado serviço ou aplicativo não mais funciona corretamente sob o SELinux

Utilizando o modo Enforcing

O nome praticamente diz tudo. No modo Enforcing, o SELinux é ativado e todas as regras de política de segurança são impostas (*enforced*).

Compreendendo contextos de segurança do SELinux

Como mencionado anteriormente, um contexto de segurança do SELinux é o método utilizado para classificar objetos (como arquivos) e sujeitos (como usuários e programas). O contexto de segurança definido permite ao SELinux impor regras de política para os sujeitos que acessam os objetos. Um contexto de segurança consiste em quatro atributos: `user`, `role`, `type` e `level`.

- `user` — O atributo `user` é um mapeamento de um nome de usuário do Linux para um nome do SELinux. Isso não é a mesma coisa que o nome de login de um usuário, e é chamado especificamente de usuário SELinux. O nome de usuário do SELinux termina com um **u**, tornando mais fácil identificá-lo na saída.
- `role` — Um papel designado na empresa é mapeado para o nome de um papel do SELinux. O atributo `role` é então atribuído a vários sujeitos e objetos. A cada papel, é concedido acesso a outros sujeitos e objetos com base na autorização de segurança do papel e no nível de classificação do objeto. Mais especificamente, para o SELinux, um papel é atribuído a cada usuário e os papéis são autorizados para determinados tipos ou domínios. O uso de papéis pode forçar contas, como `root`, a uma posição menos privilegiada. O nome do papel do SELinux tem um “r” no final.
- `type` — Esse atributo define um tipo de domínio para processos, um tipo de usuário para usuários e um tipo de arquivo para arquivos. Esse atributo também é chamado de “tipo de segurança”. Atribuir um tipo é um método de agrupar itens com base nas suas semelhanças a partir de um ponto de vista de segurança. Regras de política permitem que certos domínios e papéis acessem certos tipos de objetos. Isso é um elemento chave no Type Enforcement. Por exemplo, cada processo é executado em um domínio. Esse tipo de domínio atribuído determina diretamente o acesso de cada processo aos vários tipos de arquivo, bem como o acesso a outros tipos de

domínios dos processos. A maioria das regras de política está preocupada com quais tipos de segurança têm acesso a quais outros tipos de segurança. O nome do tipo no SELinux termina com um **t**.

- `level` — O `level` é um atributo do Multi-Level Security (MLS) e impõe o modelo Bell-LaPadula. Ele é opcional no TE, mas é necessário se você estiver usando MLS. O nível MLS é uma combinação dos valores de sensibilidade e categoria, que juntos formam o nível de segurança. Um nível é escrito como `sensitivity : category`.
- `sensitivity`
 - Representa o nível de segurança ou sensibilidade de um objeto como confidencial ou ultrassecreto.
 - É hierárquico com `s0` (não classificado) normalmente sendo o mais baixo.
 - É listado como um par de níveis de sensibilidade (*nívelBaixo-nívelAlto*) se os níveis forem diferentes.
 - É listado como um único nível de sensibilidade (`s0`) se não houver níveis baixos e altos. Mas em alguns casos, mesmo que não haja níveis baixos e altos, o intervalo ainda é mostrado (`s0-s0`).
- `category`
 - Representa a categoria de um objeto, como No Clearance, Top Clearance etc.
 - Tradicionalmente, os valores estão entre `c0` e `c255`.
 - É listado como um par de níveis de categoria (*nívelBaixo-nívelAlto*) se os níveis forem diferentes.
 - É listado como um único nível de categoria (*nível*) se não houver níveis baixos e altos.

Os usuários têm contextos de segurança

Para ver o contexto de usuário do SELinux, digite o comando `id` no shell de comando. Eis um exemplo do contexto de segurança para o usuário `johndoe`:

```
$ id
uid=1000(johndoe) gid=1000(johndoe) groups=1000(johndoe)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

A lista do contexto de segurança de usuário mostra o seguinte:

- `user` — O usuário Linux, `johndoe`, é mapeado para o usuário `unconfined_u` do SELinux.
- `role` — O usuário SELinux, `unconfined_u`, é mapeado para o papel do `unconfined_r`.
- `type` — O usuário recebeu o tipo de `unconfined_t`.
- `level` —

- `sensitivity` — O usuário tem apenas um único nível de sensibilidade e ele é o nível mais baixo de `s0`.
- `categories` — O usuário tem acesso a `c0.c1023`, que são todas as categorias (`c0` até `c1023`).

Os arquivos têm contextos de segurança

Um arquivo também tem um contexto de segurança. Para ver o contexto de um arquivo individual, use a opção `-Z` no comando `ls`. O seguinte é um contexto de segurança para o arquivo `my_stuff`:

```
$ ls -Z my_stuff
-rw-rw-r--. johndoe johndoe
unconfined_u:object_r:user_home_t:s0 my_stuff
```

A lista do contexto de arquivo mostra o seguinte:

- `user` — O arquivo é mapeado para o usuário `unconfined_u` do SELinux.
- `role` — O arquivo é mapeado para a função de `object_r`.
- `type` — O arquivo é considerado parte do domínio `user_home_t`.
- `level` —
 - `sensitivity` — O usuário tem apenas um único nível de sensibilidade e ele é o nível mais baixo de `s0`.
 - `categories` — MCS não é configurado como esse arquivo.

Os processos têm contextos de segurança

O contexto de segurança de um processo tem os mesmos quatro atributos que o contexto de um usuário e um arquivo. Para ver informações do processo em um sistema Linux, você normalmente usa uma variante do comando `ps`. No código a seguir, o comando `ps -e1` foi usado.

```
# ps -e1 | grep bash
0 S 1000 1589 1582 0 80 0 - 1653 n_tty_ pts/0 00:00:00 bash
0 S 1000 5289 1582 0 80 0 - 1653 wait pts/1 00:00:00 bash
4 S 0 5350 5342 0 80 0 - 1684 wait pts/1 00:00:00 bash
```

Para ver o contexto de segurança de um processo, use a opção `-Z` no comando `ps`. No exemplo a seguir, o comando `ps -eZ` foi usado e então redirecionado para `grep` a fim de pesquisar somente os processos executando o bash shell.

```
# ps -eZ | grep bash
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 1589 pts/0 00:00:00 bash
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 5289 pts/1 00:00:00 bash
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023 5350 pts/1 00:00:00 bash
```

A lista de contexto do processo mostra o seguinte:

- `user` — O processo é mapeado para o usuário `unconfined_u` do SELinux.

- `role` — O processo está em execução para o papel `unconfined_r`.
- `type` — O processo está em execução no domínio `unconfined_t`.
- `level` —
 - `sensitivity` — O processo tem apenas o nível `s0`.
 - `categories` — O processo tem acesso a `c0.c1023`, o que significa todas as categorias (`c0` até `c1023`).

Esses contextos de segurança podem ser alterados para atender às necessidades de segurança específicas da organização. Mas antes de você aprender a alterar as configurações desses contextos de segurança, você entender outra peça do quebra-cabeça do SELinux, os tipos de política do SELinux.

Entendendo os tipos de política do SELinux

O tipo de política escolhido determina diretamente quais conjuntos de regras de política são utilizados para determinar o que um objeto pode acessar. O tipo de política também determina quais atributos específicos do contexto de segurança são necessários. É aqui que você começa a ver o nível refinado do controle de acesso que pode ser implementado por meio do SELinux.

ta

pos de política disponíveis na sua distribuição talvez não correspondam com aqueles listados aqui. Por exemplo, distribuições mais antigas do Linux, a política rigorosa ainda está disponível. Nas distribuições mais recentes, a política rigorosa foi mesclada com a política Targeted.

O SELinux tem diferentes políticas que você pode escolher:

- Targeted
- MLS
- Minimum

Cada política implementa diferentes controles de acesso para atender às necessidades da organização. É fundamental entender esses tipos de política a fim de selecionar a correta para suas necessidades de segurança particulares.

Política Targeted

O objetivo principal da política Targeted é restringir daemons “direcionados”. Mas ela pode também restringir outros processos e usuários. Daemons direcionados são colocados na caixa de areia. Uma *caixa de areia* é um ambiente em que os programas podem ser executados, mas o acesso a outros objetos é rigidamente controlado. Diz-se que um processo em execução nesse ambiente está na “caixa de areia”. Assim, um daemon direcionado é restrito de modo que nenhum ataque lançado por meio dele pode afetar outros serviços ou o sistema Linux como um todo.

Todos os sujeitos e objetos não direcionados são executados no domínio `unconfined_t`. O domínio

`unconfined_t` não tem restrições de política do SELinux e, portanto, só utiliza a segurança “tradicional” do Linux.

O SELinux vem com o conjunto de políticas específicas padrão. Assim, por padrão, o SELinux visa apenas alguns daemons.

A política MLS (Multi-Level Security)

Objetivo principal da política MLS é aplicar o modelo Bell-LaPadula. Ele concede acesso a outros sujeitos e objetos com base na *autorização de segurança* de um papel e *nível de classificação* do objeto.

Na política MLS, o atributo MLS de um contexto de segurança é crucial. Do contrário, as regras de política não saberão como impor restrições de acesso.

Política Minimum

Essa política é exatamente isso, mínima. Ela foi originalmente criada para máquinas com pouca memória ou dispositivos como smart phones.

A política Minimum é essencialmente a mesma que a política Target, mas apenas o pacote básico da política de regras é usado. Essa política “essencial” pode ser usada para testar os efeitos do SELinux em um único daemon designado. Para dispositivos com pouca memória, a política Minimum permite que o SELinux seja executado sem consumir uma grande quantidade de recursos.

Entendendo os pacotes de regras do SELinux

Regras de política, também chamadas *regras de permissão*, são as regras usadas pelo SELinux para determinar se um sujeito tem acesso a um objeto. As regras de política são instaladas com o SELinux e são agrupadas em pacotes, também chamados de *módulos*. Cada arquivo específico do pacote de políticas termina com um `*.pp`.

O diretório `/etc/selinux/tipo_de_politica/modules/active/modules` contém alguns arquivos de pacotes de política (`*.pp`). O exemplo a seguir mostra os pacotes de regras de políticas para um sistema Linux com a política Targeted implementada:

```
# ls /etc/selinux/targeted/modules/active/modules/*.pp
/etc/selinux/targeted/modules/active/modules/abrt.pp
/etc/selinux/targeted/modules/active/modules/accountsd.pp
/etc/selinux/targeted/modules/active/modules/acct.pp
/etc/selinux/targeted/modules/active/modules/ada.pp
/etc/selinux/targeted/modules/active/modules/afs.pp
...
/etc/selinux/targeted/modules/active/modules/xserver.pp
/etc/selinux/targeted/modules/active/modules/zabbix.pp
/etc/selinux/targeted/modules/active/modules/zarafa.pp
/etc/selinux/targeted/modules/active/modules/zebra.pp
/etc/selinux/targeted/modules/active/modules/zosremote.pp
```

No sistema Linux, há documentação sobre esses vários módulos de política, sob a forma de arquivos HTML. Para consultar essa documentação no Fedora ou no RHEL, abra o navegador do sistema e digite o seguinte URL: `file:///usr/share/doc/politica_selinux-nº_da_versao_do_selinux/html/index.html`. A Figura 24.1 mostra o índice da documentação do módulo de política. Para o Ubuntu, o URL é

file:///usr/share/doc/selinux-policy-doc/html/index.html. Se não tiver a documentação da política no seu sistema, você pode instalá-la em um sistema Fedora ou RHEL digitando **yum install selinux-policy-doc** na linha de comando. No Ubuntu, digite **sudo apt-get install selinuxpolicy-doc** na linha de comando.

mentação do módulo de política do SELinux.

Você pode revisar essa documentação da política para ver como as regras de política são criadas e empacotadas.

Os pacotes das regras de política, juntamente com o modo de operação do SELinux, tipo Policy e vários contextos de segurança, funcionam juntos para proteger o sistema Linux via o SELinux. A seção a seguir discutirá como começar a configurar o SELinux para atender às necessidades de segurança específicas da sua organização.

Configurando o SELinux

O SELinux vem pré-configurado. Você pode usar os recursos do SELinux sem nenhum trabalho de configuração. Mas raramente as definições pré-configuradas atendem a todas as necessidades de segurança do sistema Linux.

As configurações do SELinux só podem ser definidas e modificadas pelo usuário root. Os arquivos de configuração e política estão localizados no diretório `/etc/selinux`.

O arquivo de configuração principal é o `/etc/selinux/config` e ele se parece com isto:

```
# cat /etc/selinux/config
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - SELinux is fully disabled.
SELINUX=enforcing
# SELINUXTYPE= type of policy in use. Possible values are:
#   targeted - Only targeted network daemons are protected.
#   strict - Full SELinux protection.
SELINUXTYPE=targeted
```

Esse arquivo de configuração principal do SELinux permite definir o modo operacional e o tipo de política.

Definindo o modo operacional do SELinux

Para ver o modo operacional atual do SELinux no seu sistema, use o comando `getenforce`. Para ver o modo operacional atual e o modo definido no arquivo de configuração, use o comando `sestatus`. Ambos os comandos são mostrados no código a seguir:

```
# getenforce
Enforcing
#
# sestatus
SELinux status:          enabled
SELinuxfs mount:        /sys/fs/selinux
Current mode:            enforcing
Mode from config file:   enforcing
Policy version:          26
Policy from config file: targeted
```

Policy from config file: targeted

Para alterar a configuração do modo operacional, use o `setenforce` **novaconfiguração**, onde **novaconfiguração** é:

- `enforcing` ou `1`
- `permissive` ou `0`

Observe que você não pode usar o comando `setenforce` para mudar o SELinux para o modo `disabled`.

O exemplo a seguir mostra o modo do SELinux sendo alterado para o modo `permissive` por meio do comando `setenforce`. O comando `sestatus` mostra o modo operacional atual e o modo no arquivo de configuração, que não foi modificado. Quando o sistema é reinicializado, ele determina o modo operacional do SELinux a partir do arquivo de configuração. Assim, o modo `permissive` configurado no exemplo a seguir é temporário porque o modo `enforcing` será configurado por meio do arquivo de configuração quando o sistema for reiniciado.

```
# setenforce 0
#
# getenforce
Permissive
#
# sestatus
SELinux status:          enabled
SELinuxfs mount:        /sys/fs/selinux
Current mode:            permissive
Mode from config file:   enforcing
...
```

atenção

Não tente mudar do `disabled` para o modo `enforcing` modificando o arquivo de configuração e reiniciando. Tentar mudar do `disabled` para `enforcing` por meio do comando `setenforce` pode travar o sistema como resultado de rótulos incorretos de arquivo.

Para desativar o SELinux, você precisa editar o arquivo de configuração dele. Para evitar que as configurações atuais do modo operacional mudem, você também deve editar o arquivo de configuração. O método preferido para mudar o modo operacional do SELinux é modificar o arquivo de configuração e então reiniciar o sistema.

Ao mudar de `disabled` para o modo `enforcing` ou `permissive`, o SELinux irá atribuir automaticamente um novo nome ao sistema de arquivos depois de uma reinicialização. Isso significa que o SELinux irá verificar e alterar os contextos de segurança de todos os arquivos com contextos de

segurança incorretos (por exemplo, arquivos erroneamente rotulados) que podem causar problemas no novo modo operacional. Além disso, quaisquer arquivos não rotulados são rotulados com contextos. Esse processo de renomeação pode demorar bastante porque o contexto de cada arquivo é verificado. A seguir está a mensagem que você recebe quando um sistema passa por um processo de rerotulação após a reinicialização:

```
*** Warning -- SELinux targeted policy relabel is required.
*** Relabeling could take a very long time, depending on file
*** system size and speed of hard drives.
```

Para modificar o modo operacional no arquivo `/etc/selinux/config`, altere a linha `SELINUX=` para um dos seguintes:

- `disabled`
- `enforcing`
- `permissive`

O exemplo do arquivo de configuração do SELinux a seguir mostra que o modo foi configurado como `permissive`. Agora, quando ocorrer uma reinicialização do sistema, o modo será alterado.

```
# setenforce 0
#
# getenforce
Permissive
#
# sestatus
SELinux status:          enabled
SELinuxfs mount:        /sys/fs/selinux
Current mode:            permissive
Mode from config file:   enforcing
...
```

O arquivo de configuração principal do SELinux não contém apenas a configuração do modo operacional, ele também especifica o tipo de política que será aplicado.

Definindo o tipo de política do SELinux

O tipo de política que você escolhe determinará se o SELinux aplica TE, MLS ou um pacote básico. Essa configuração de tipo determina diretamente os conjuntos das regras de política utilizados para determinar o que um objeto pode acessar.

Por padrão, o tipo de política é definido como `targeted`. Para alterar o tipo de política padrão, edite o arquivo `/etc/selinux/config`. Altere a linha `SELINUXTYPE=` para um dos seguintes:

- `targeted`
- `mls`
- `minimum`

Se definir o tipo do SELinux como `mls` ou `minimum`, você precisará certificar-se de que o pacote de políticas foi instalado. Verifique isso digitando o seguinte comando: `yum list selinux-policy-mls` ou `yum list selinux-policy-minimum`.

ta

Para verificar os pacotes de política do SELinux no Ubuntu, use o comando `sudo apt-cache policy sepol`.

O exemplo do arquivo de configuração do SELinux a seguir mostra que o tipo foi configurado como `mls`. Agora, quando ocorrer uma reinicialização do sistema, o tipo de política será alterado.

```
# cat /etc/selinux/config
# This file controls the state of SELinux on the system.
...
# SELINUXTYPE= type of policy in use. Possible values are:
#   targeted - Only targeted network daemons are protected.
#   strict - Full SELinux protection.
SELINUXTYPE=mls
```

enção

Cuidado para não se deixar enganar pelos comentários desatualizados no arquivo de configuração do SELinux. Você não pode definir `SELINUXTYPE` como `strict` nas novas distribuições do Linux. Se fizer isso, o sistema travará na próxima reinicialização e você precisará usar os comandos `grub` para corrigir o problema. O tipo de política `strict` agora é parte do tipo de política `targeted`.

Gerenciando os contextos de segurança do SELinux

Contextos de segurança do SELinux permitem que ele imponha regras de política aos sujeitos que acessam os objetos. O sistema Linux vem com contextos de segurança já atribuídos.

Para visualizar os atuais contextos de segurança de processo e arquivo do SELinux, use o comando `secon`. A Tabela 24.1 lista as opções disponíveis no comando `secon`.

TABELA 24.1 Opções do Comando `secon`

Opção	Descrição
<code>-u</code>	Use essa opção para mostrar o usuário do contexto de segurança.
<code>-r</code>	Use essa opção para mostrar o papel do contexto de segurança.
<code>-t</code>	Use essa opção para mostrar o tipo de contexto de segurança.
<code>-s</code>	Use essa opção para mostrar o nível de sensibilidade do contexto de segurança.

Use essa opção para mostrar o nível de autorização do contexto de segurança.

Use essa opção para mostrar o nível de sensibilidade e autorização do contexto de segurança como um intervalo MLS.

Se você usar o comando `secon` sem uma designação, ele mostrará o contexto de segurança do processo atual. Para ver outro contexto de segurança do processo, use a opção `-p`. O exemplo a seguir mostra como usar `secon` para ver o contexto segurança atual do processo `init`.

```
# secon -urt
user: unconfined_u
role: unconfined_r
type: unconfined_t
#
# secon -urt -p 1
user: system_u
role: system_r
type: init_t
```

Para visualizar o contexto de segurança de um arquivo, use a opção `-f`, como mostrado aqui:

```
# secon -urt -f /etc/passwd
user: system_u
role: object_r
type: etc_t
```

O contexto de segurança de um usuário não é visualizado usando o comando `secon`. Para ver o contexto de segurança de um usuário, use o comando `id`. Para ver o contexto de segurança de um usuário além do seu próprio, a sintaxe do comando é `id -Z nome_do_usuario`.

Gerenciando o contexto de segurança do usuário

Lembre-se de que cada ID de login de usuário do sistema é mapeado para um determinado ID de usuário do SELinux. Para ver uma lista de mapeamento no seu sistema, digite o comando **`semanage login -l`**. O comando `semanage` e sua saída são mostrados no código a seguir. Se um ID de login de usuário não estiver listado, então o mapeamento de login “padrão” é utilizado, que é o Login Name de `_default_`. Observe que as configurações MLS/MCS associadas para cada usuário do SELinux também são mostradas.

```
# semanage login -l
Login Name      SELinux User      MLS/MCS Range
__default__     unconfined_u      s0-s0:c0.c1023
root            unconfined_u      s0-s0:c0.c1023
system_u        system_u           s0-s0:c0.c1023
```

Para ver uma exibição atual dos usuários do SELinux e seus papéis associados, use o comando

`semanage user -l`. A exibição parcial a seguir mostra os papéis mapeados para os nomes de usuário do SELinux:

```
# semanage user -l
      Labeling MLS/      MLS/
SELinux User Prefix  MCS Level  MCS      Range SELinux Roles

git_shell_u  user      s0        s0        git_shell_r
guest_u      user      s0        s0        guest_r
...
user_u       user      s0        s0        user_r
xguest_u     user      s0        s0        xguest_r
```

Se você precisar adicionar um novo nome de usuário do SELinux, o utilitário `semanage` é usado novamente. Dessa vez, o comando é `semanage user -a nome_de_usuario_no_selinux`. Para mapear um ID de login para o usuário SELinux recém-adicionado, o comando é `semanage login -a -s nome_de_usuario_no_selinux loginID`. O utilitário `semanage` é uma ferramenta poderosa para gerenciar a configuração do SELinux. Para mais informações sobre o utilitário `semanage`, consulte as pages `man`.

Gerenciando o contexto de segurança de arquivo

Rotular arquivos é fundamental para manter um controle de acesso adequado aos dados de cada arquivo. O SELinux não define rótulos de segurança de arquivo na instalação e na reinicialização do sistema quando o modo de operação `disabled` do SELinux é alterado para outro modo. Para ver o rótulo atual de um arquivo (conhecido como contexto de segurança), use o comando `ls -Z`, como mostrado aqui:

```
# ls -Z /etc/passwd
-rw-r--r--. root root
system_u:object_r:etc_t:s0      /etc/passwd
```

Há vários comandos que você pode usar para gerenciar rótulos de contexto de segurança de arquivo, como mostrado na Tabela 24.2.

TABELA 24.2 Comandos do Gerenciamento de Rótulos do Contexto de Segurança de Arquivo

Utilitário	Descrição
<code>cat</code>	Usado para mudar a categoria do rótulo de um contexto de segurança de arquivo.
<code>con</code>	Usado para alterar o rótulo do contexto de segurança de um arquivo.
<code>xfiles</code>	Chama o utilitário <code>restorecon/setfiles</code> .
<code>storecon</code>	Faz exatamente a mesma coisa que o utilitário <code>setfiles</code> , mas tem uma interface diferente da de <code>setfiles</code> .
<code>tfiles</code>	Utilizado para verificar e/ou corrigir rótulos de contexto de segurança. Ele pode ser executado para verificação de rótulos de arquivos e/ou renomeação de arquivos ao adicionar um novo módulo de política ao sistema. Faz exatamente a mesma coisa que o utilitário

Os comandos `chcat` e `chcon`, mostrados na Tabela 24.2, permitem alterar o contexto de segurança de um arquivo. No exemplo a seguir, o comando `chcon` é usado para mudar o usuário SELinux associado a `file.txt` de `undefined_u` para `system_u`.

```
# ls -Z file.txt
-rw-rw-r--. johndoe johndoe
  undefined_u: object_r:user_home_t:s0 file.txt

#
# chcon -u system_u file.txt
#
# ls -Z file.txt
-rw-rw-r--. johndoe johndoe
  system_u:object_r:user_home_t:s0 file.txt
```

Observe na Tabela 24.2 que `fixfiles`, `restorecon` e `setfiles` são essencialmente o mesmo utilitário. Mas `restorecon` é a escolha popular para corrigir rótulos de arquivo. O comando `restorecon -R nome_do_arquivo` restaura o contexto de segurança padrão de um arquivo.

Gerenciando o contexto de segurança do processo

A definição de um processo é um programa em execução. Ao executar programas ou serviços em um sistema Linux, cada um recebe um ID de processo (ver Capítulo 6). Em um sistema com o SELinux, um processo também recebe um contexto de segurança.

O modo como um processo recebe seu contexto de segurança depende de qual processo o iniciou. Lembre-se de que `init` é a “mãe” de todos os processos (ver Capítulo 15). Assim, muitos daemons e processos são iniciados por `init`. As partidas dos processos `init` recebem novos contextos de segurança. Por exemplo, quando o daemon `apache` é iniciado pelo `init`, ele recebe o tipo (conhecido como domínio) `httpd_t`. O contexto atribuído é tratado pela política do SELinux escrita especificamente para esse daemon. Se não houver nenhuma política para um processo, então ele recebe um tipo padrão, `unconfined_t`.

Para um programa ou aplicativo executado por um usuário (processo pai), o novo processo (processo filho) herda o contexto de segurança do usuário. É claro, isso só ocorre se o usuário tiver permissão para executar o programa. Um processo também pode executar um programa. O processo filho, nesse caso, também herda seu contexto de segurança do processo pai. Portanto, o processo filho é executado no mesmo domínio.

Assim, um contexto de segurança de um processo é definido antes de o programa ser executado e depende de quem o iniciou. Há alguns comandos que você pode usar para mudar os contextos de segurança sob os quais um programa é executado:

- `runcon` — Executa o programa usando opções para determinar o usuário, o papel e o tipo (conhecido como domínio).

- `sandbox` — Executa o programa dentro de um domínio rigidamente controlado (conhecido como caixa de areia).

Você pode causar vários problemas usando `runcon`, portanto, utilize-o com cautela. Mas `sandbox` oferece muita proteção. Ele permite flexibilidade ao testar novos programas no sistema Linux.

Gerenciando pacotes de regras de política do SELinux

Regras de política são as regras usadas pelo SELinux para determinar se um sujeito tem acesso a um objeto. Elas são agrupadas em pacotes, também chamados módulos, e são instaladas com o SELinux. Uma maneira fácil de visualizar os módulos no seu sistema é usar o comando `semodule -l`. Ele listará todos os módulos de política junto com o número da versão atual. Um exemplo do comando `semodule -l` é mostrado aqui:

```
# semodule -l
abrt          1.2.0
accountsd    1.0.0
acct         1.5.0
...
xserver      3.5.6
zabbix       1.3.1
zarafa       1.0.0
zebra        1.12.0
zosremote    1.1.0
```

Várias ferramentas irão ajudá-lo a gerenciar e até mesmo criar seus próprios módulos de política. A Tabela 24.3 mostra as várias ferramentas de pacotes de regras de política disponíveis em um sistema Fedora.

TABELA 24.3 Ferramentas do Pacote de Políticas do SELinux

Ferramenta de política	Descrição
<code>dit2allow</code>	Gera regras de políticas <code>allow/dontaudit</code> a partir dos logs das operações negadas
<code>dit2why</code>	Gera uma descrição do motivo por que o acesso foi negado a partir dos logs das operações negadas
<code>eckmodule</code>	Compila os módulos de política
<code>eckpolicy</code>	Compila políticas do SELinux
<code>ad_policy</code>	Carrega novas políticas no kernel
<code>module</code>	Gerencia módulos de política

<code>module_deps</code>	Lista as dependências entre os pacotes de políticas
<code>module_expand</code>	Expande um pacote de módulo de política
<code>module_link</code>	Agrupa pacotes de módulo de política
<code>module_package</code>	Cria um pacote de módulo de política

O seguinte é um exemplo da política normalmente usada como um framework para criar regras de política locais. A política de exemplo é bastante longa, assim, apenas uma parte é mostrada.

```
# cat /usr/share/selinux/devel/example.te
policy_module(myapp,1.0.0)

#####
#
# Declarations
#

type myapp_t;
type myapp_exec_t;
domain_type(myapp_t)
domain_entry_file(myapp_t, myapp_exec_t)

type myapp_log_t;
logging_log_file(myapp_log_t)

type myapp_tmp_t;
files_tmp_file(myapp_tmp_t)

...
allow myapp_t myapp_tmp_t:file manage_file_perms;
files_tmp_filetrans(myapp_t,myapp_tmp_t,file)
#
```

Você pode ver a partir do código de exemplo anterior que há uma sintaxe especial usada no código de política. Para criar e modificar as regras de política, você terá de entender a sintaxe da linguagem delas; aprender a usar os compiladores de política do SELinux; aprender a agrupar arquivos de regras de política para formar módulos; e, provavelmente, precisará ter duas aulas de um dia de duração sobre o assunto. Você poderia ser tentado a desistir do SELinux nesse momento. Mas é muito mais fácil utilizar booleanos para modificar as políticas.

Gerenciando o SELinux via booleanos

Escrever uma regra de política e criar um módulo do SELinux são atividades bastante complicadas e demoradas. Criar regras de política incorretas poderia comprometer a segurança do sistema Linux. Felizmente, o SELinux fornece booleanos.

Um booleano é uma chave que alterna uma configuração entre ativada ou desativada. Uma chave booleana permite alterar partes das regras de política do SELinux sem que você precise entender como

escrever a política. Essas mudanças na política também podem ser feitas sem reinicializar o sistema.

Para ver uma lista de todos os booleanos atuais usados no SELinux, use o comando `getsebool -a`. Eis um exemplo das regras de política do SELinux com booleanos em um sistema Linux Fedora:

```
# getsebool -a
abrt_anon_write --> off
abrt_handle_event --> off
allow_console_login --> off
...
xserver_object_manager --> off
zabbix_can_network --> off
```

Para ver uma política específica que pode ser modificada por um booleano, use mais uma vez o comando `getsebool`. Desta vez, o nome da política é passado para ele, como mostrado no exemplo a seguir:

```
# getsebool httpd_can_connect_ftp
httpd_can_connect_ftp --> off
```

Para alternar uma política, você pode usar o comando `setsebool` ou o comando `togglebool`. Ambos os comandos alteram a regra da política temporariamente. Quando o sistema é reinicializado, o booleano retornará à sua configuração original. Se for necessário tornar essa configuração, use apenas o `setsebool` com a opção `-P`.

O comando `togglebool` apenas alterna a configuração booleana atual da política que você especifica entre ativada e desativada. Por exemplo, se você emitisse o comando `togglebool httpd_can_connect_ftp`, mudaria o status da configuração de política a partir da definição anterior de “desativado” para “ativado”. O comando `setsebool` tem seis definições: três para ativar uma política (`on`, `1` ou `true`) e três para desativar uma política (`off`, `0` ou `false`).

Para um exemplo usando `setsebool`, lembre-se do Capítulo 22, que não é uma boa prática de segurança permitir que os usuários executem programas a partir do diretório `/home`. Normalmente, os programas executados pelos usuários nesse local são malwares. Para evitar que isso aconteça, a regra de política `allow_user_exec_content` precisa ser desativada. O exemplo a seguir mostra o comando `setsebool` sendo usado para fazer exatamente isso. Observe que a opção `-P` é usada para tornar essa configuração permanente.

```
# setsebool -P allow_user_exec_content off
```

O comando `getsebool` verificará se a configuração booleana foi criada corretamente:

```
# getsebool allow_user_exec_content
allow_user_exec_content --> off
```

Booleanos tornam a modificação das atuais regras de política do SELinux muito mais fácil. No geral, os utilitários de configuração de linha de comando do SELinux, como `getsebool`, são fáceis de utilizar.

Mas se você quiser uma ferramenta de configuração GUI, o SELinux tem uma. Ela é instalada por meio do comando `yum install policycoreutils-gui`. No Ubuntu, use o comando `sudo apt-get install policycoreutils`. Para utilizar essa ferramenta de configuração, basta digitar o comando `system-config-selinux` e uma interface gráfica aparece.

Monitoramento e solução de problemas no SELinux

O SELinux é outra ferramenta para monitorar o sistema. Ele registra todas as proibições de acesso, o que pode ajudar a determinar se um ataque está sendo tentado. Esses mesmos arquivos de log do SELinux também são úteis para solucionar problemas no SELinux.

Entendendo o registro em log do SELinux

O SELinux usa um cache chamado Access Vector Cache (AVC) ao revisar as regras de política para contextos específicos de segurança. Quando o acesso é negado, o que é chamado de negação AVC, uma mensagem de negação é armazenada em um arquivo de log.

Essas mensagens de negação registradas em log podem ajudar a diagnosticar e tratar violações rotineiras da política do SELinux. O arquivo em que essas mensagens de negação são registradas depende do estado dos daemons `auditd` e `rsyslogd`:

- Se o daemon `auditd` estiver em execução, as mensagens de negação são registradas em `/var/log/audit/audit.log`.
- Se `auditd` não estiver em execução, mas o daemon `rsyslogd` estiver em execução, as mensagens de negação são registradas em `/var/log/messages`.

ta
Se o daemon `auditd` e `rsyslogd` estiverem em execução e você tiver o daemon `setroubleshootd` no seu sistema, as mensagens de negação são enviadas para os arquivos de log `audit.log` e `messages`. Mas informações de negação no arquivo de log `messages` são armazenadas em um formato mais compreensível pelo daemon `setroubleshootd`.

Revisando mensagens SELinux no log de auditoria

Se o daemon `auditd` estiver em execução, você poderá ver rapidamente se as negações AVC foram registradas usando o comando `aureport`. O exemplo a seguir mostra o uso de `aureport` e `grep` para procurar negações AVC. Pelo menos uma negação foi registrada em `/var/log/audit/audit.log`:

```
# aureport | grep AVC
Number of AVC's: 1
```

Depois de descobrir que uma negação AVC foi registrada em `audit.log`, você pode usar `ausearch` para revisar a(s) mensagem(ns) de negação. O exemplo a seguir mostra o comando `ausearch` sendo usado para revisar a mensagem de negação AVC registrada em log.

```
# ausearch -m avc
time->Sat Feb 25 09:18:07 2015
type=SYSCALL msg=audit(1330179487.213:250):
 arch=40000003 syscall=226 success=no exit=-22
 a0=8c1d8d8 a1=4c6db1cb a2=8c1ec90 a3=1f items=0
 ppid=2582 pid=3053 auid=1000 uid=0 gid=0 euid=0
 suid=0 fsuid=0 egid=0 sgid=0 fsgid=0 tty=pts0
 ses=2 comm="chcon" exe="/usr/bin/chcon"
 subj=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
 key=(null)

type=AVC msg=audit(1330179487.213:250): avc: denied
 { mac_admin | for pid=3053 comm="chcon" capability=33
 scontext=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
 tcontext=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
 tclass=capability2
```

A tela fornecerá informações sobre quem tentou o acesso, juntamente com o contexto de segurança ao tentar isso. Palavras-chave a procurar em uma mensagem de negação AVC são:

- `type=AVC`
- `AVC: negado`
- `pid=`
- `exe=`
- `subj=`

Isso pode lhe dar dados suficientes para começar a corrigir um problema ou rastrear atividades maliciosas.

Revisando mensagens SELinux no log de mensagens

Se o `rsyslogd` estiver em execução, você poderá encontrar mensagens de negação AVC pesquisando o arquivo `/var/log/messages` usando `grep`. O exemplo a seguir mostra a pesquisa de mensagens de negação do SELinux.

```
# grep "SELinux is preventing" /var/log/messages
Feb 25 09:18:08 localhost setroubleshoot: SELinux is preventing
 /usr/bin/chcon from mac_admin access on the None . For complete
 SELinux messages.
run sealert -l b1386ca2-6e83-4c29-b0db-a5470fa34f75
```

A partir do exemplo, você pode ver que um usuário Linux tentou executar o comando `chcon`. Além disso, observe que a mensagem de negação AVC informa que você pode executar o comando `sealert` para obter mais informações. A mensagem inclui essa informação porque, nesse sistema Linux em particular, o daemon `setroubleshootd` está em execução.

O utilitário `sealert` permite obter mais informações sobre uma determinada mensagem de negação AVC. O formato de informação que `sealert` oferece ajudará a diagnosticar seus problemas. O exemplo a seguir mostra as informações que `sealert` fornece em relação à negação AVC, que foi mostrada no exemplo anterior. Observe que o comando usado é exatamente o mesmo comando sugerido a partir do arquivo de mensagem de log anterior. O número longo usado no comando `sealert` é o número de ID da mensagem de negação AVC.

```
# sealert -l b1386ca2-6e83-4c29-b0db-a5470fa34f75
SELinux is preventing /usr/bin/chcon from mac_admin access on the
None .

***** Plugin catchall (100. confidence) suggests *****

If you believe that chcon should be allowed mac_admin access on the
<Unknown> by default.
Then you should report this as a bug.
You can generate a local policy module to allow this access.
Do
allow this access for now by executing:
# grep chcon /var/log/audit/audit.log | audit2allow -M mypol
# semodule -i mypol.pp
```

Additional Information:

```
Source Context          unconfined_u:unconfined_r:unconfined_t:s(
s0:c0.

c1

023

Target Context         unconfined_u:unconfined_r:unconfined_t:s(
s0:c0.

c1

023

Target Objects         [ None ]
Source                  chcon
Source Path             /usr/bin/chcon
Port                    <Unknown>
Host                    localhost.localdomain
Source RPM Packages     coreutils-8.12-6.fc16.i686
```

```
Target RPM Packages
Policy RPM                selinux-policy-3.10.0-75.fc16.noarch
Selinux Enabled           True
Policy Type               targeted
Enforcing Mode            Enforcing
Host Name                 localhost.localdomain
Platform                 Linux localhost.localdomain 3.2.7-
                          1.fc16.i686 #1
                          SMP Tue Feb 21 01:38:57 UTC 2012 i686 i686
Alert Count               1
First Seen                Sat 25 Feb 2015 09:18:07 AM EST
Last Seen                 Sat 25 Feb 2015 09:18:07 AM EST
Local ID                  b1386ca2-6e83-4c29-b0db-a5470fa34f75
```

Raw Audit Messages

```
type=AVC msg=audit(1330179487.213:250): avc: denied
...
```

Você pode ver que há uma grande quantidade de informações úteis na saída de `sealert`. Se o SELinux foi aplicado ao seu sistema, seria prudente também executar o daemon `setroubleshootd`.

Solucionado problemas no registro em log do SELinux

Obviamente, os arquivos de log são extremamente importantes para diagnosticar e resolver as violações da política do SELinux. Os arquivos de log são o primeiro passo na solução de problemas no SELinux. Assim, antes de tudo, é importante certificar-se de que seu sistema Linux registra em log as mensagens.

Uma maneira rápida de determinar se o registro em log está ocorrendo é verificar se os daemons apropriados estão em execução: `auditd`, `rsyslogd` e/ou `setroubleshootd`. Use um comando apropriado, como `systemctl status auditd.service`. É claro, o comando que você usa depende da distribuição e versão do Linux. Consulte no Capítulo 15 mais detalhes. Se o daemon não estiver em execução, inicie-o de modo que o registro em log possa começar a ocorrer.

Atenção

Às vezes negações AVC não registradas em log por causa das regras de política de `dontaudit`. Embora as regras `dontaudit` ajudem a reduzir os falsos positivos nos logs, elas podem interferir negativamente na solução de problemas. Para corrigir isso, desative temporariamente todas as regras de política `dontaudit` usando o comando `module -DB`.

Agora você pode forçar uma negação AVC falsa emitindo um comando que você sabe que vai gerar uma negação, como `chcon -u fake_u nome_do_arquivo` a partir de uma conta que não seja de

administrador. Emitir um comando `chcon` a partir de uma conta de usuário não administrador deve gerar a mensagem de teste desejada. Agora, revise o(s) arquivo(s) de log apropriado(s) para ver se uma mensagem de negação AVC foi gerada.

Solucionando problemas comuns do SELinux

Ao começar a trabalhar com o SELinux, é fácil ignorar o óbvio. Qualquer que seja o acesso negado, você primeiro deve verificar as permissões DAC “tradicionais” do Linux. Por exemplo, use o comando `ls -l` e verifique duas vezes se o proprietário de um arquivo, grupo e atribuições de leitura, gravação e execução estão corretos.

Com o SELinux, vários itens comuns podem causar problemas:

- Usar um diretório não padrão para um serviço
- Usar uma porta não padrão para um serviço
- Mover arquivos que resultam na perda dos seus rótulos do contexto de segurança
- Configurar booleanos incorretamente

Cada um desses problemas pode ser resolvido bem rapidamente.

Usar um diretório não padrão para um serviço

Por várias razões, talvez você decida armazenar os arquivos de um serviço em um diretório não padrão. Ao fazer isso, o SELinux precisa saber que esse comportamento não padrão ocorreu. Do contrário, ele negará o acesso a solicitações legítimas de acesso a serviços.

Por exemplo, você decidiu manter seus arquivos HTML em um local diferente do `/var/www/html` padrão. Você armazena os arquivos em `/srv/www/html`. O SELinux precisa saber que você quer que o serviço `http` seja capaz de acessar os arquivos dentro de `/srv/www/html`. Os comandos para fazer isso são `semanage` e `restorecon`. A seguir, os comandos são usados para adicionar o tipo apropriado de contexto de segurança ao diretório `/srv/www/html` e tudo que ele contém:

```
# semanage fcontext -a -t httpd_sys_content_t "/srv/www/html(/.*)?"
```

Para realmente definir o novo tipo do contexto de segurança para os arquivos dentro do diretório, você precisará usar o comando `restorecon -R`. Isso é feito desta maneira:

```
# restorecon -R -v /srv/www/html
```

Agora, o daemon `httpd` tem permissão para acessar seus arquivos HTML em um diretório local não padrão.

Usar uma porta não padrão para um serviço

Semelhante ao problema recém-descrito, talvez você decida que um serviço ouça em uma porta não padrão. Ao fazer essa alteração de porta, o serviço frequentemente não inicializa.

Por exemplo, para fins de segurança, você decide mover `sshd` da porta 22 para uma porta não padrão, 47347. O SELinux não conhece essa porta e o serviço não será iniciado. Para corrigir esse problema, você deve primeiro encontrar o tipo de contexto de segurança para `sshd`. Isso é feito usando o código a seguir emitindo o comando `semanage port -l` e redirecionando os resultados para `grep` a fim de procurar `ssh`.

```
# semanage port -l | grep ssh
```

```
ssh_port_t                                tcp                                22
```

No exemplo anterior, você pode ver que o tipo de contexto necessário é `ssh_port_t`. Agora, usando o comando `semanage` novamente, você adiciona esse tipo à porta 47347, como mostrado aqui:

```
# semanage port -a -t ssh_port_t -p tcp 47347
```

Nesse ponto, você deve ser capaz de iniciar o `sshd` na porta não padrão 47347.

Movendo arquivos e perdendo rótulos de contexto de segurança

Você usou o comando `cp` para mover alguns arquivos temporariamente para um novo diretório. Então, você usou o comando `mv` para colocá-los de volta. Agora, os arquivos têm o contexto de segurança do diretório temporário em vez do contexto original de segurança e seu sistema recebe mensagens de negação AVC.

Isso é fácil de corrigir, graças ao comando `restorecon -R`. Basta digitar **`restorecon -R diretório_original`** e os arquivos com o contexto original de segurança serão restaurados.

Booleanos definidos incorretamente

Outro problema comum é simplesmente configurar um booleano incorretamente. Isso pode lhe dar várias negações AVC.

Por exemplo, se os scripts do sistema não mais são capazes de se conectar à rede e você está recebendo negações AVC nos registros em log, você precisará verificar os booleanos `httpd`. Use o comando `getsebool -a` e direcione-o para `grep` a fim de procurar quaisquer booleanos que afetam `httpd`. O exemplo aqui mostra que esses comandos estão em uso:

```
# getsebool -a | grep http
...
httpd_can_network_connect --> off
...
```

O comando `getsebool` mostra que o booleano `httpd_can_network_connect` está desativado. Para alterar esse booleano, use o seguinte comando: `setsebool -P httpd_can_network_connect on`. Observe que a opção `-P` foi usada para tornar a configuração permanente. Agora, os scripts devem ser capazes de se conectar à rede.

À medida que você encontra vários problemas com o SELinux, suas habilidades de solução de problemas vão melhorar. Enquanto isso, eis um excelente recurso para ajudá-lo a solucionar problemas, <http://docs.redhat.com>. O documento do Red Hat “Red Hat Enterprise Linux” tem um capítulo inteiro (Capítulo 8) dedicado à solução de problemas do SELinux.

Juntando tudo

Obviamente, o SELinux é uma ferramenta bem rica e complicada. Você agora tem uma boa e sólida base sobre os conceitos básicos do SELinux. Eis algumas recomendações para você começar a implementar o SELinux no seu sistema.

- Comece com a fase de Planejamento do Ciclo de Vida do Processo de Segurança (ver Capítulo 22). Isso não é uma ferramenta de segurança que você pode usar sem planejamento. Você precisa determinar os papéis organizacionais, quem estará nesses papéis e atribuir níveis de autorização; determinar um nível de classificação de um objeto individual etc.
- Use uma matriz de controle de acesso (ver Capítulo 22).
Uma matriz de controle de acesso irá ajudá-lo a implementar os papéis determinados, níveis de autorização e os tipos; atribuir os diversos contextos de segurança do SELinux e modificar as regras de acesso.
- Comece com o modo operacional permissivo.
Execute seu sistema atual por um período de tempo significativo no modo Permissive. Revise os logs e veja quais problemas poderiam ocorrer com as configurações padrão do SELinux. Depois de revisar os problemas, ative o modo Enforcing.
- Altere uma coisa de cada vez.
De maneira geral, implemente as mudanças de configuração do SELinux uma de cada vez, em um ambiente de teste ou usando o modo Permissive. Veja que tipo de efeito cada alteração de configuração tem antes de passar para a próxima.

Obtendo informações adicionais sobre o SELinux

Há várias fontes de informações adicionais para ajudá-lo com o SELinux no seu sistema Linux:

- **Páginas man do seu sistema** — Emita o comando `man -k selinux` para encontrar todas as várias páginas man que você pode revisar para os utilitários SELinux atualmente instalados no seu sistema.
- **Documentação do módulo das políticas do SELinux do seu sistema** — Para visualizar essa documentação, abra o navegador do sistema e digite o seguinte URL:
<http:///usr/share/docs/selinux-policy-selinuxversion#/index.html>.

- **Manuais do Red Hat Enterprise Linux** — Localizado em <http://docs.redhat.com>, esse site contém um manual inteiro sobre o SELinux.
- **The Fedora Project SELinux Guide** — Localizado em <http://docs.fedoraproject.org>, esse site tem um Security-Enhanced Linux Guide. Mas o guia não está atualizado para cada versão do Fedora, assim, talvez você precise examinar as versões mais antigas para encontrá-lo. Além disso, o SELinux Guide não está localizado no manual Security, mas o manual Security também é um bom manual para revisar.
- **SELinux no Ubuntu** — Como há diferenças sutis entre o SELinux no RHEL/Fedora e no Ubuntu, o site <https://wiki.ubuntu.com/SELinux> fornece ajuda adicional de que você precisa.
- **SELinux Project Wiki** — Essa é a página oficial do projeto SELinux. Há vários recursos nesse site, que está localizado em <http://selinuxproject.org>.
- **SELinux News** — Assim como parece, há notícias atuais sobre o SELinux em <http://selinuxnews.org>.

Resumo

Proteger seu servidor Linux é crítico e o SELinux pode ajudar. O SELinux fornece uma melhoria de segurança para o Linux e é instalado por padrão em muitas distribuições do Linux. Neste capítulo, você aprendeu os benefícios do SELinux; como ele funciona; como configurá-lo; como corrigir os vários problemas com o SELinux; e como obter informações adicionais sobre essa importante melhoria de segurança.

O SELinux fornece a capacidade de implementar os modelos de controle aprimorados — MAC e RBAC — além do DAC Linux padrão. Ele oferece melhor segurança usando recursos de segurança importantes, como o acesso menos privilegiado e caixa de areia de processo. Esses recursos sozinhos tornam o SELinux um claro vencedor no aprimoramento de segurança.

À primeira vista, o SELinux parece bastante complicado. Mas uma vez dividido nos seus vários componentes, modos operacionais, contextos de segurança, tipos de política e pacotes de política, você pode ver como as várias partes funcionam em conjunto. Cada componente desempenha um papel importante para aplicar e testar os requisitos de segurança escolhidos para sua organização.

Você aprendeu as várias etapas para configurar o SELinux. Embora o SELinux venha pré-configurado, talvez você precise fazer algumas modificações para atender às necessidades de segurança da sua organização. Cada componente tem seus próprios passos de configuração e definições a escolher. Embora a criação de regras de política não tenha sido discutida, você aprendeu a modificar as políticas fornecidas por meio de booleanos.

O SELinux fornece outra ferramenta para monitorar a segurança do sistema Linux. Como o SELinux registra em log todas as negações de acesso, ele pode ajudá-lo a determinar se um ataque foi ou está sendo tentado. Até mesmo os melhores planos podem dar errado. Portanto, neste capítulo, você aprendeu a corrigir problemas comuns de configuração do SELinux.

Uma vez que aprender a configurar, usar e manter o SELinux pode ser bem trabalhoso, é importante saber

onde você pode obter informações adicionais sobre o SELinux. Este capítulo, juntamente com as fontes adicionais úteis incluídas, deve fazer com que você comece a configurar e usar praticamente o Security Enhanced Linux para melhorar a segurança do seu sistema Linux.

No próximo capítulo, veremos como proteger seu sistema Linux em uma rede. Discutiremos como controlar o acesso, gerenciar firewalls e proteger o acesso remoto.

Exercícios

Use estes exercícios para testar seus conhecimentos de como usar o SELinux. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux). Se você empacar, soluções para as tarefas são mostradas no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa).

1. Sem fazer alterações no arquivo primário de configuração do SELinux, anote o comando para configurar o sistema no modo Permissive Operating para o SELinux.
2. Sem fazer alterações no arquivo primário de configuração do SELinux, anote o comando para configurar o sistema no modo de operação Enforcing para o SELinux. (Atenção: É melhor só executar esse comando no seu sistema para um exercício depois que você estiver pronto para impor o SELinux.)
3. Qual é o tipo atual de política do SELinux definido no seu sistema e como você o encontrou?
4. Liste o contexto de segurança de um arquivo e identifique os diferentes atributos de contexto de segurança.
5. Qual comando alteraria o atributo `type` de um arquivo? (Atenção: Só emita o comando no seu sistema se você quiser alterar o tipo do arquivo.)
6. Liste o contexto de segurança do processo atual e identifique os diferentes atributos do contexto de segurança.
7. Que comando restauraria o contexto do arquivo padrão do SELinux de um arquivo? (Atenção: Só execute esse comando no seu sistema se você conhecer seus efeitos.)
8. Liste os booleanos atuais usados no seu sistema. Anote um comando para modificar um deles.
9. Qual comando listaria todos os módulos de política do SELinux no seu sistema, juntamente com os respectivos números de versão?
10. Crie uma mensagem de negação AVC e então revise-a no(s) log(s) usando as ferramentas apropriadas.

Protegendo o Linux em uma rede

NESTE CAPÍTULO

Gerenciando serviços de rede

Controlando o acesso aos serviços de rede

Implementando firewalls

Configurar o sistema Linux em uma rede, especialmente em uma rede pública, cria todo um novo conjunto de desafios quando se trata de segurança. A melhor maneira de proteger seu sistema Linux é mantê-lo fora de todas as redes. Mas isso raramente é uma opção viável.

Livros inteiros estão cheios de informações sobre como proteger um sistema de computador em uma rede. Muitas organizações contratam em tempo integral administradores de segurança de computadores para monitorar os sistemas Linux conectados a uma rede. Portanto, pense neste capítulo como uma introdução breve para proteger o Linux em uma rede.

Um ponto de partida para aprender sobre segurança de rede é entender o modelo OSI. Cada rede funciona em uma série de camadas, chamadas de modelo de referência Open Systems Interconnection (OSI) (ver Figura 25.1). O modelo consiste em sete camadas, cada uma representando o processo de transferir pacotes de dados entre um remetente e um receptor.

O modelo OSI é uma representação conceitual. Muitos protocolos reais de rede operam em várias camadas em vez de se moverem em uma linha reta entre uma fase e outra. Mas esse modelo é muito útil para fins de segurança. Você pode obter uma visão geral das várias fases das comunicações em rede que devem ser mantidas seguras para o servidor Linux.

delo de referência OSI.

Auditando serviços de rede

A principal função do sistema Linux é oferecer serviços. Um *serviço de rede* é qualquer tarefa que o computador executa exigindo que ele envie e receba informações pela rede utilizando um conjunto pré-definido de regras. Encaminhar e-mails é um serviço de rede, assim como entregar páginas web também é.

Um servidor Linux tem o potencial de fornecer milhares de serviços. Muitos deles estão listados no arquivo `/etc/services`. Considere as seguintes seções do arquivo `/etc/services`:

```
$ cat /etc/services
# /etc/services:
# $Id: services,v 1.53 2011/06/13 15:00:06 ovasik Exp $
#
# Network services, Internet style
# IANA services version: last updated 2011-06-10
#
# Note that it is presently the policy of IANA to assign
  a single well-known
# port number for both TCP and UDP; hence, most entries
here have two entries
# even if the protocol doesn't support UDP operations.
# Updated from RFC 1700, "Assigned Numbers" (October 1994).
  Not all ports
# are included, only the more common ones.
#
# The latest IANA port assignments can be gotten from
# http://www.iana.org/assignments/port-numbers
# The Well Known Ports are those from 0 through 1023.
# The Registered Ports are those from 1024 through 49151
# The Dynamic and/or Private Ports are those from 49152
  through 65535
#
# Each line describes one service, and is of the form:
#
```

```

# service-name port/protocol [aliases ...] [# comment]
...

echo                7/tcp
echo                7/udp
discard            9/tcp                sink null
discard            9/udp                sink null
sysstat            11/tcp                users
sysstat            11/udp                users
daytime            13/tcp
daytime            13/udp
qotd                17/tcp                quote
qotd                17/udp                quote
...

chargen            19/tcp                ttytst source
chargen            19/udp                ttytst source
ftp-data           20/tcp
ftp-data           20/udp
# 21 is registered to ftp, but also used by fsp
ftp                21/tcp
...

http                80/tcp                www www-http        #
WorldWideWeb
HTTP
http                80/udp                www www-http        # HyperText
Transfer
Protocol
http                80/sctp                # HyperText
Transfer
Protocol
kerberos            88/tcp                kerberos5 krb5      # Kerberos
v5
kerberos            88/udp                kerberos5 krb5      # Kerberos
v5
...

blp5                48129/udp            # Bloomberg
locator
com-bardac-dw       48556/tcp            # com-

```

com-bardac-dw	48556/udp	bardac-dw # com- bardac-dw
iqobject	48619/tcp	# iqobject
iqobject	48619/udp	# iqobject

Após as linhas de comentário, você observará três colunas de informações. A coluna esquerda contém o nome de cada serviço. A coluna do meio define o número da porta e o tipo de protocolo usado para esse serviço. A coluna da direita contém um alias opcional ou lista de aliases para o serviço.

Muitas distribuições do Linux vêm com serviços de rede desnecessários em execução. Um serviço desnecessário expõe o sistema Linux a ataques maliciosos. Por exemplo, se o servidor Linux for um servidor de impressão, então ele só deve oferecer serviços de impressão. Ele não deve também oferecer serviços Apache Web. Isso só exporia desnecessariamente o servidor de impressão a ataques maliciosos que se aproveitam das vulnerabilidades do serviço web.

Avaliando o acesso aos serviços de rede

Um dos resultados da fase de Planejamento do Ciclo de Vida da Segurança é uma “lista de verificação dos softwares e serviços necessários” (consulte o Capítulo 22, “Entendendo segurança básica no Linux”, para ver uma descrição). Usando essa lista de verificação, você precisa rever e remover serviços do ponto de vista do host (ver Capítulo 15, “Iniciando e parando serviços”) e do ponto de vista da rede.

Além disso, você deve revisar a forma como os serviços de rede necessários são “anunciados”. Se o servidor Linux precisa oferecer um serviço de rede como Secure Shell, apenas aqueles autorizados pela organização precisam saber que ele está lá.

Usando nmap para criar uma lista de serviços de rede

Uma ferramenta maravilhosa para ajudá-lo a revisar os serviços de rede a partir de um ponto de vista de rede é o scanner de segurança nmap. O utilitário nmap está disponível na maioria dos repositórios de distribuição do Linux e tem uma página web repleta de informações em <http://nmap.org>.

Para instalar o nmap em uma distribuição do Fedora ou do RHEL, use o comando yum (usando privilégios root), como mostrado no exemplo a seguir.

```
# yum install nmap
Loaded plugins: langpacks, presto, refresh-packagekit
...
Setting up Install Process
Resolving Dependencies
--> Running transaction check
---> Package nmap.i686 2:5.51-1.fc16 will be installed
--> Finished Dependency Resolution

Dependencies Resolved
```

```
...
Install
```

```
1 Package
```

```
...
Installing : 2:nmap-5.51-1.fc16.i686 1/1
Installed:
  nmap.i686 2:5.51-1.fc16
Complete!
```

Para instalar o utilitário `nmap` em uma distribuição do Ubuntu, digite **`sudo apt-get install nmap`** na linha de comando.

O nome completo do utilitário `nmap` é “Network Mapper”. Ele tem uma variedade de usos para auditorias de segurança e exploração de rede. As diferentes varreduras de portas do `nmap` permitem ver quais serviços estão em execução em todos os servidores na rede local e se eles estão anunciando suas disponibilidades.

ta

o que é uma porta? Uma porta ou, mais corretamente, uma *porta de rede*, é um valor numérico utilizado pelo protocolo de rede TCP/IP para identificar os serviços que podem estar disponíveis em um sistema. Por exemplo, a porta 80 é a porta de rede padrão para o serviço web Apache. Pense em uma porta de rede como uma porta para seu servidor Linux. Cada porta é numerada e atrás de cada uma há um serviço específico esperando ajudar quem quer bater nessa porta. Por razões de segurança, se o servidor não deve oferecer esse serviço, você quer que quem tentar encontrar uma parede de tijolos atrás da porta desse serviço.

Para auditar as portas do seu servidor, o utilitário `nmap` oferece vários tipos de verificação úteis. O site `nmap` tem um manual completo sobre todas as técnicas de varredura de porta que você pode usar em <http://nmap.org/book/man-port-scanning-techniques.html>. Eis duas varreduras básicas de porta para você começar a auditar um serviço:

- **Varredura de porta de conexão TCP** — Para essa varredura, o `nmap` tenta um Transmission Control Protocol (TCP) para conectar-se às portas no servidor. Se uma porta estiver ouvindo, a tentativa de conexão será bem-sucedida.

O TCP é um protocolo de rede usado no conjunto de protocolos TCP/IP. Seu objetivo principal é negociar uma conexão usando o que é chamado de “handshake de três vias”. O TCP envia um pacote de sincronização (SYN) para um servidor remoto, determinando um número específico de porta no pacote. O servidor remoto recebe o SYN e responde com um pacote de confirmação (SYN-ACK) para o computador de origem. O servidor original então reconhece (ACK) a resposta e uma conexão TCP é estabelecida oficialmente. Esse handshake de três vias é muitas vezes chamado SYN-SYN-ACK ou SYN, SYN-ACK, ACK.

Se você selecionar uma varredura de porta TCP Connect, o utilitário `nmap` usa esse handshake de três vias para fazer um pouco de atividade investigativa em um servidor remoto. Quaisquer

serviços que usam o protocolo TCP responderão à varredura.

- **Varredura de porta UDP** — Para essa varredura, o `nmap` envia um pacote UDP para cada porta no sistema que está sendo varrido. O UDP é outro protocolo popular no conjunto de protocolos TCP/IP. Se a porta estiver ouvindo e tiver um serviço que usa o protocolo UDP, ela responderá à varredura.

ca

tenha em mente que utilitários Free e Open Source Software (FOSS) também estão disponíveis para aqueles com intenções maliciosas. Ao fazer essas varreduras de `nmap`, entenda que os resultados da varredura remota que você vê para o servidor Linux são os resultados da mesma varredura que outras pessoas verão. Isso vai ajudá-lo a avaliar as configurações de segurança do seu sistema em termos da quantidade de informações disponibilizadas para varreduras de portas.

Ao executar o utilitário `nmap`, ele fornecerá um pequeno relatório útil com informações sobre o sistema que está varrendo e as portas que ele vê. As portas recebem um status de “estado”. O `nmap` informa seis possíveis estados de porta:

- `open` — Esse é o estado mais perigoso que uma varredura de `nmap` pode relatar para uma porta. Uma porta `open` indica que um servidor tem um serviço que trata de solicitações de serviço nessa porta. Pense nisso como um sinal na porta, “Entre! Estamos aqui para ajudá-lo”.
- `closed` — Esse é um estado mais apropriado do que `open`. A porta `closed` é acessível, mas não há nenhum serviço esperando no outro lado dela. No entanto, o status da varredura ainda indica que há um servidor Linux nesse endereço IP particular.
- `filtered` — Esse é o melhor estado para uma porta. Ele não pode determinar se a porta está `open` ou até mesmo se um servidor Linux está no endereço IP varrido. Um scanner malicioso ganharia poucas informações. Normalmente é necessário um firewall (discutido na seção “Trabalhando com firewalls” mais adiante, neste capítulo) para obter esse status que o `nmap` relatou em uma porta.
- `unfiltered` — A varredura de `nmap` vê a porta, mas não pode determinar se a porta está `open` ou `closed`.
- `open|filtered` — A varredura de `nmap` vê a porta, mas não pode determinar se a porta está `open` ou `filtered`.
- `closed|filtered` — A varredura de `nmap` vê a porta, mas não pode determinar se a porta está `closed` ou `filtered`.

Para ajudá-lo a entender melhor como usar o utilitário `nmap`, revise o exemplo a seguir. Para propósitos de construção de uma lista de serviços, as varreduras de exemplo de `nmap` são realizadas em um sistema Fedora. A primeira varredura é uma TCP Connect a partir da linha de comando, usando o endereço de loopback, 127.0.0.1.

```
# nmap -sT 127.0.0.1
```

```
Starting Nmap 5.51 ( http://nmap.org ) at 2015-03-22 10:33 EDT
```

```
Nmap scan report for localhost.localdomain (127.0.0.1)
```

```
Host is up (0.016s latency).
```

```
Not shown: 998 closed ports
```

PORT	STATE	SERVICE
25/tcp	open	smtp
631/tcp	open	ipp

```
Nmap done: 1 IP address (1 host up) scanned in 1.34 seconds
```

A varredura TCP Connect nmap informa que duas portas TCP estão abertas e têm serviços ouvindo na porta:

- O Simple Mail Transfer Protocol (SMTP) ouve na porta 25.
- O Internet Printing Protocol (IPP) ouve na porta 631.

A próxima varredura de nmap é uma UDP no endereço loopback do sistema Fedora.

```
# nmap -sU 127.0.0.1
```

```
Starting Nmap 5.51 ( http://nmap.org ) at 2015-03-22 10:36 EDT
```

```
Nmap scan report for localhost.localdomain (127.0.0.1)
```

```
Host is up (0.00048s latency).
```

```
Not shown: 997 closed ports
```

PORT	STATE	SERVICE
68/udp	open filtered	dhcpc
631/udp	open filtered	ipp

```
Nmap done: 1 IP address (1 host up) scanned in 2.24 seconds
```

A varredura UDP nmap relata que duas portas UDP estão abertas e têm serviços ouvindo na porta:

- O cliente Dynamic Host Control Protocol (dhcpc) ouve na porta 68.
- O Internet Printing Protocol (ipp) ouve na porta 631.

Observe que o IPP da porta 631 é listado tanto na varredura TCP Connect do nmap como na varredura UDP, porque o protocolo IPP usou o protocolo TCP e o protocolo UDP e, portanto, é listado nas duas varreduras.

Usando essas duas varreduras de nmap simples, TCP Connect e UDP, no endereço de loopback, você pode construir uma lista dos serviços de rede oferecidos pelo servidor Linux. Depois de construir a lista,

compare-a com a “lista de verificação de softwares e serviços necessários” que você criou no Capítulo 22. Quaisquer serviços de rede que não estão na “lista de verificação de software e serviços necessários” devem ser desativados no servidor Linux (ver Capítulo 15).

Usando nmap para auditar anúncios dos serviços de rede

Para alguns serviços de rede do Linux, você quer chamar muita atenção. Para outros serviços de rede, você quer chamar a atenção de um grupo seletivo e autorizado. Você precisa saber como seus serviços de rede são anunciados. Em outras palavras, você quer saber se scanners maliciosos podem ver as portas de rede do seu servidor Linux e os serviços que elas oferecem.

A ideia aqui é comparar o que o servidor Linux se parece internamente com o que ele se parece externamente. Se determinar que um número excessivo de serviços de rede vulneráveis é anunciado, você pode tomar medidas para que eles não sejam vistos.

ca

poderia ser tentado a pular as varreduras da rede interna dentro da organização. Não faça isso. Atividades maliciosas frequentemente ocorrem a partir de funcionários da própria empresa ou por alguém que já penetrou as portas externas.

Mais uma vez, o utilitário `nmap` será uma grande ajuda aqui. Para ter uma visão adequada de como as portas do seu servidor Linux são vistas, você precisará realizar varreduras a partir de vários locais. Por exemplo, uma auditoria simples faria com que as varreduras ocorressem:

- no próprio servidor Linux.
- a partir de outro servidor na mesma rede da organização.
- a partir de fora da rede da organização.

Nos exemplos a seguir, parte de uma auditoria simples é realizada. O utilitário `nmap` é executado em um sistema Fedora, designado como “Host-A”. O Host-A é o servidor Linux cujos serviços de rede devem ser protegidos. O Host-B é um servidor Linux, utilizando a distribuição Linux Mint, e está na mesma rede que o Host-A.

ca

Configurações de segurança em vários componentes de rede, como o firewall do servidor e os roteadores da rede, devem ser consideradas ao realizar varreduras de auditoria.

Para esse exemplo de auditoria, uma varredura é executada a partir do Host-A, usando não o endereço de loopback, mas o endereço IP real. Primeiro, o endereço IP para o Host-A é determinado usando o comando `ifconfig`. O endereço IP é 10.140.67.23.

```
# ifconfig
```

```
lo Link encap:Local Loopback
  inet addr:127.0.0.1 Mask:255.0.0.0
  inet6 addr: ::1/128 Scope:Host
  UP LOOPBACK RUNNING MTU:16436 Metric:1
  RX packets:4 errors:0 dropped:0 overruns:0 frame:0
  TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
  collisions:0 txqueuelen:0
  RX bytes:240 (240.0 b) TX bytes:240 (240.0 b)
```

```
p2p1 Link encap:Ethernet HWaddr 08:00:27:E5:89:5A
  inet addr:10.140.67.23
  Bcast:10.140.67.255 Mask:255.255.255.0
```

```
inet6 addr: fe80::a00:27ff:fee5:895a/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:81 errors:0 dropped:0 overruns:0 frame:0
TX packets:102 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:50015 (48.8 KiB) TX bytes:14721 (14.3 KiB)
```

Agora, usando o endereço IP do Host-A, uma varredura TCP Connect nmap é emitida a partir do Host-A. A varredura de nmap vai até a rede para realizar a varredura. Todas as portas são reportadas como tendo um status de closed.

```
# nmap -sT 10.140.67.23
```

```
Starting Nmap 5.51 ( http://nmap.org ) at 2015-03-22 10:33 EDT
Nmap scan report for 10.140.67.23
```

```
Host is up (0.010s latency).
```

```
All 1000 scanned ports on 10.140.67.23 are closed
```

```
Nmap done: 1 IP address (1 host up) scanned in 1.48 seconds
```

A varredura de nmap é movida da origem no Host-A para a origem em um Host-B. Agora, a varredura TCP Connect é tentada nas portas do Host-A a partir da linha de comando do Host-B.

```
$ nmap -sT 10.140.67.23
```

```
Starting Nmap 5.21 ( http://nmap.org ) at 2015-03-22 05:34 HADT
```

```
Note: Host seems down. If it is really up,  
but blocking our ping probes, try -PN
```

```
Nmap done: 1 IP address (0 hosts up) scanned in 0.11 seconds
```

Aqui, nmap dá uma dica útil. O Host-A parece estar parado, ou ele simplesmente poderia estar bloqueando as sondagens. Então, outra varredura de nmap é tentada a partir do Host-B, usando conselho de nmap para desativar as sondagens de ping da varredura por meio da opção -PN.

```
$ nmap -sT -PN 10.140.67.23
```

```
Starting Nmap 5.21 ( http://nmap.org ) at 2015-03-22 05:55 HADT
```

```
Nmap scan report for 10.140.67.23
```

```
Host is up (0.0015s latency).
```

```
All 1000 scanned ports on 10.140.67.23 are filtered
```

```
Nmap done: 1 IP address (1 host up) scanned in 5.54 seconds
```

Você pode ver que o Host-A (10.140.67.23) está funcionando e executando e todas as suas portas têm um status de `filtered`. Isso significa que há um firewall no Host-A. Essas varreduras do Host-B fornecem uma ideia melhor do que um scanner malicioso poderia ver ao varrer seu servidor Linux. Nesse exemplo, o scanner malicioso não vê muita coisa.

ta

Se você estiver familiarizado com o `nmap`, sabe que a varredura TCP SYN é a padrão que o `nmap` usa. A varredura SYN faz um excelente trabalho de sondar um sistema remoto de forma furtiva. Como você está sondando seu próprio sistema para fins de auditoria de segurança, faz sentido usar as varreduras mais “pesadas” do utilitário `nmap`. Se você ainda quiser usar a varredura TCP SYN, o comando é `nmap -sS ip_address`.

Os serviços atualmente em execução no Host-A não são tão “interessantes”. Assim, no exemplo a seguir, outro serviço, `ssh`, é iniciado no Host-A usando o comando `systemctl` (ver Capítulo 15). Isso deve dar ao utilitário `nmap` um alvo mais interessante a procurar.

```
# systemctl start sshd.service
#
# systemctl status sshd.service
sshd.service - OpenSSH server daemon
  Loaded: loaded (/lib/systemd/system/sshd.service; disabled)
  Active: active (running) since
  Thu, 22 Mar 2015 10:57:24 -0400; 12s ago

Main PID: 1750 (sshd)
  CGroup: name=systemd:/system/sshd.service
          1750 /usr/sbin/sshd -D

#
```

Além disso, como o firewall do Host-A está bloqueando as varreduras de `nmap` do Host-B, seria interessante ver o que uma varredura de `nmap` pode relatar quando o firewall está desativado. O exemplo a seguir mostra o firewall sendo desativado no Host-A:

```
*****
```

```
# systemctl stop iptables.service
```

```
#
```

```
# systemctl status iptables.service
```

```
iptables.service - IPv4 firewall with iptables
```

```
Loaded: loaded (/lib/systemd/system/iptables.service; enabled)
Active: inactive (dead) since
Thu, 22 Mar 2015 11:21:28 -0400; 8s ago
Process: 1806 ExecStop=/usr/libexec/iptables.init stop
(code=exited, status=0/SUCCESS)
Process: 656 ExecStart=/usr/libexec/iptables.init start
(code=exited, status=0/SUCCESS)
CGroup: name=systemd:/system/iptables.service
```

Com um novo serviço em execução e um nível baixo de proteção no firewall do Host-A, as varreduras de nmap devem encontrar algo. Na parte a seguir, as varreduras de nmap são executadas novamente a partir do Host-B. Dessa vez, o utilitário nmap mostra o serviço ssh em execução na porta aberta 22. Observe que com o firewall desativado no Host-A, ambas as varreduras de nmap capturam muito mais informações. Isso realmente demonstra a importância do firewall do seu servidor Linux.

```
$ nmap -sT 10.140.67.23
```

```
Starting Nmap 5.21 ( http://nmap.org ) at 2012-03-22 06:22 HADT
Nmap scan report for 10.140.67.23
Host is up (0.016s latency).
Not shown: 999 closed ports

PORT STATE SERVICE
```

```
22/tcp open ssh
```

```
Nmap done: 1 IP address (1 host up) scanned in 0.40 seconds
$
```

```
$ sudo nmap -sU 10.140.67.23
```

```
[sudo] password for johndoe: *****
Starting Nmap 5.21 ( http://nmap.org ) at 2012-03-22 06:22 HADT
Nmap scan report for 10.140.67.23
Host is up (0.00072s latency).
Not shown: 997 closed ports
```

```
PORT STATE SERVICE
68/udp open|filtered dhcpc
631/udp open|filtered ipp
```

```
...
Nmap done: 1 IP address (1 host up) scanned in 1081.83 seconds
```

Para realizar uma auditoria completa, não se esqueça de incluir a varredura UDP. Além disso, há varreduras de nmap adicionais que podem ser benéficas para sua organização. Dê uma olhada no site do

utilitário `nmap` para outras sugestões.

atenção

Você tiver seguido o processo e reduziu o nível de proteção no firewall do seu servidor para realizar essas tarefas de `nmap`, certifique-se de aumentá-lo novamente!

Você ainda precisa implementar os controles para os serviços que seu servidor Linux deve oferecer. Uma maneira de conseguir isso é por meio de um *wrapper* (“empacotador”).

Controlando o acesso aos serviços de rede

Sem problemas desativar completamente um serviço não utilizado, mas, para os serviços de rede necessários, você deve configurar o controle de acesso. Esse controle de acesso necessário é alcançado por meio dos arquivos `/etc/hosts.allow` e `/etc/hosts.deny`, para sistemas Linux que incorporam suporte TCP Wrapper.

Se ele tiver suporte TCP Wrapper, um serviço de rede utilizará `libwrap`. Para verificar `libwrap`, você pode executar o comando `ldd` no daemon do serviço de rede. O exemplo a seguir mostra que o daemon `ssh` utiliza TCP Wrappers. Se seu sistema Linux `sshd` não usa TCP Wrappers, então nenhuma saída será mostrada.

```
$ ldd /usr/sbin/sshd | grep libwrap
libwrap.so.0 => /lib/libwrap.so.0 (0x0012f000)
```

Quando um serviço de rede que incorpora o suporte a TCP Wrapper é solicitado, os arquivos `hosts.allow` e `hosts.deny` são varridos e verificados quanto a uma entrada que corresponde ao endereço do sistema remoto que faz a solicitação. As etapas a seguir ocorrem quando um sistema remoto solicita acesso a um serviço TCP Wrapper suportado:

1. O arquivo `hosts.allow` é verificado.
 - Se o endereço do sistema remoto estiver listado:
 - O acesso é permitido.
 - Nenhuma verificação TCP Wrapper adicional é feita.
 - Se o endereço do sistema remoto **não** estiver listado, o processo de verificação TCP Wrapper continua no arquivo `hosts.deny`.
2. O arquivo `hosts.deny` é verificado.
 - Se o endereço do sistema remoto estiver na lista, o acesso é negado.
 - Se o endereço do sistema remoto **não** estiver na lista, o acesso é permitido.

A ordem em que os arquivos `hosts` são avaliados é importante. Por exemplo, você não pode negar

acesso a um host no arquivo `hosts.deny` que já recebeu acesso no arquivo `hosts.allow`.

Não é necessário listar cada endereço único que poderia tentar se conectar ao seu computador. Os arquivos `hosts.allow` e `hosts.deny` permitem especificar sub-redes inteiras e grupos de endereços. Você pode até usar a palavra-chave `ALL` para especificar todos os possíveis endereços IP. Além disso, você pode restringir entradas específicas nesses arquivos para que elas só se apliquem aos serviços de rede. Considere o seguinte exemplo de um par típico de arquivos `hosts.allow` e `hosts.deny`.

```
# cat /etc/hosts.allow
# hosts.allow This file describes the names of the hosts that
are
#         allowed to use the local INET services, as decided
#         by the '/usr/sbin/tcpd' server.
#
sshd: 199.170.177.
vsftpd: ALL
#
# cat /etc/hosts.deny
# hosts.deny This file describes names of the hosts which are
#         *not* allowed to use the local INET services, as
#         decided by the '/usr/sbin/tcpd' server.
#
ALL: ALL
```

Nos arquivos `hosts`, as linhas que começam com um caractere `#` são comentários e são ignoradas. Outras linhas consistem em uma lista separada por vírgulas dos nomes dos serviços seguida por um caractere de dois pontos (`:`) e, então, uma lista separada por vírgulas dos endereços dos clientes a verificar. Nesse contexto, um cliente é qualquer computador que tenta acessar um serviço de rede em seu sistema.

O exemplo anterior é de uma configuração bastante restritiva. A linha:

```
sshd: 199.170.177.
```

no arquivo `hosts.allow`, permite conexões aos serviços `sshd` a partir de certos sistemas remotos. A linha:

```
vsftpd: ALL
```

permite que todos os sistemas remotos se conectem ao serviço FTP, `vsftpd`. Mas se o sistema remoto não estiver solicitando uma conexão com `sshd` ou `vsftpd`, o acesso a ele será negado pela linha no arquivo `hosts.deny`:

```
ALL: ALL
```

A entrada `client` pode ser um endereço IP numérico (como `199.170.177.25`) ou um hostname (como `jukebox.linuxtoys.net`). Muitas vezes, usa-se um curinga que especifica todo um intervalo de endereços. Observe no arquivo `host.allow` de exemplo que o endereço IP permitido da entrada do `sshd` é `199.170.177`. Isso corresponderá a qualquer endereço IP que inicia com essa string, como `199.170.177.25`. O curinga `ALL` usado no arquivo `hosts.deny` especifica que *todos* os sistemas

remotos entrando em contato com esse arquivo, solicitando *quaisquer* serviços de rede TCP Wrapper suportados, serão negados.

ta

ringa `ALL` também foi usado no arquivo `hosts.allow` de exemplo para o serviço `vsftpd`, instruindo o TCP Wrapper permitir absolutamente qualquer host a se conectar ao serviço FTP no sistema Linux. Isso é apropriado para executar servidor de FTP anônimo que qualquer pessoa na internet pode acessar. Se não estiver executando um site FTP anônimo, talvez você não queira usar o flag `ALL` aqui.

Uma boa regra geral é tornar seu arquivo `hosts.deny` o mais restritivo possível e então permitir explicitamente somente os serviços de que você realmente precisa. Conceda acesso apenas aos sistemas que realmente precisam de acesso de acordo com a matriz de controle de acesso da sua organização (ver Capítulo 22).

Junto com os TCP Wrappers, os firewalls também controlam o acesso às portas do seu sistema Linux. Na verdade, os firewalls fazem muito mais do que apenas proteger os serviços de rede.

Trabalhando com firewalls

Um firewall em um edifício é uma parede à prova de fogo que impede a propagação do fogo pelo edifício. Um *firewall* de um computador não impede a propagação do fogo, mas em vez disso bloqueia a transmissão de dados maliciosos ou indesejados dentro e fora de um sistema de computador ou rede. Por exemplo, um firewall pode bloquear varreduras maliciosas nas portas do servidor Linux. Um firewall também pode permitir que atualizações desejadas de software sejam feitas no mesmo servidor.

Entendendo firewalls

Embora você possa tender a pensar em um firewall como uma barreira completa, um firewall na verdade é apenas um filtro que verifica cada solicitação de pacote de rede ou aplicativo que chega ou sai de um sistema de computador ou rede.

ta

que é um pacote de rede? Um *pacote de rede* são dados que foram divididos em blocos transmissíveis. Os pacotes ou pacotes contêm dados adicionais enquanto eles passam pelo modelo OSI. Um dos propósitos desses dados adicionais é assegurar que um pacote chegue seguro e intacto ao seu destino. Os dados adicionais são removidos do pacote ao passar pelo modelo OSI no seu destino.

Firewalls podem ser divididos em diferentes categorias, dependendo da sua função. Cada categoria tem um lugar importante na proteção do servidor e da rede.

- **Um firewall é baseado em rede ou baseado em host.** Um firewall baseado em rede é um que

protege toda a rede ou sub-rede. Um exemplo de um firewall de rede está no seu local de trabalho, onde a rede deve ser protegida pelo firewall do roteador de filtragem.

Um firewall baseado em host é aquele que está em execução e protegendo um host ou servidor individual. É mais provável que você tenha um firewall no PC em casa. Isso é um firewall baseado em host. Outro exemplo de um firewall baseado em host é o aplicativo `firestarter`. Você pode aprender mais sobre o `firestarter` em <http://www.fs-security.com>.

- **Um firewall é um firewall de hardware ou software.** Firewalls podem estar localizados em dispositivos de rede, como roteadores. Seus filtros são configurados no firmware do roteador. Na sua casa, o provedor de serviços de internet (ISP) pode fornecer um serviço DSL ou modem a cabo para ter acesso à internet. O modem contém o firmware do firewall e é considerado um firewall de hardware.

Firewalls podem estar localizados em um sistema de computador como um aplicativo. O aplicativo permite definir que regras de filtragem sejam ajustadas, as quais filtram o tráfego de entrada. Isso é um exemplo de um firewall de software. Um firewall de software também é chamado firewall baseado em regras.

- **Um firewall é um filtro de camada de rede ou de aplicativo.** Um firewall que examina pacotes individuais de rede também é chamado *filtro de pacote*. Um firewall de camada de rede só permite que alguns pacotes entrem e saiam do sistema. Ele opera nas camadas inferiores do modelo de referência OSI.

Um firewall de camada de aplicativo filtra nas camadas superiores do modelo OSI. Esse firewall só permitirá a certos aplicativos o acesso ao/a partir do sistema.

Você pode ver como essas categorias de firewall se sobrepõem. A melhor configuração de firewall é uma combinação de todas as categorias. Como acontece com muitas práticas de segurança, quanto mais camadas houver, mais difícil é para atividades maliciosas penetrar.

Implementando firewalls

Em um sistema Linux, o firewall é baseado em host, que está na camada de rede e se baseia em software gerenciado pelo utilitário `iptables`. Com `iptables`, você pode criar uma série de regras para cada pacote de rede que passa pelo servidor Linux. Você pode ajustar as regras para permitir o tráfego de rede a partir de um local, mas não a partir de outro. Essas regras essencialmente compõem uma lista de controle de acesso de rede para o servidor Linux.

ca

utilitário `iptables` gerencia o firewall do Linux, chamado `netfilter`. Assim, você frequentemente verá o firewall do Linux sendo chamado de `netfilter/iptables`.

Entendendo o utilitário `iptables`

Antes de começar a mudar as regras do firewall por meio do utilitário `iptables`, você precisa

entender os princípios básicos do `netfilter/iptables`, que incluem o seguinte:

- Tabelas
- Cadeias
- Políticas
- Regras

Cada um desses princípios é fundamental para configurar e gerenciar o firewall do servidor Linux corretamente.

As tabelas `netfilter/iptables`

O firewall `iptables` tem a capacidade de fazer mais do que apenas filtragem de baixo nível de pacotes. Ele define o tipo de funcionalidade do firewall que ocorre. Há quatro tabelas no utilitário `iptables`, com uma tabela adicional acrescentada pelo SELinux. As tabelas oferecem as seguintes funcionalidades:

- `filter` — A tabela `filter` é o recurso de filtragem de pacotes do firewall. Nessa tabela, as decisões do controle de acesso são tomadas para pacotes entrando, saindo e passando pelo sistema Linux.
- `nat` — A tabela `nat` é usada para o Network Address Translation (NAT). Firewalls podem ser configurados para NAT, que é um recurso de segurança diferente da filtragem de pacotes.
- `mangle` — Como você poderia suspeitar, os pacotes são modificados de acordo com as regras da tabela `mangle`. A modificação de pacotes é usada no Network Address Translation.
- `raw` — A tabela `raw` é usada para excluir determinados pacotes de rede de algo chamado “rastreamento de conexão”. Esse recurso é importante ao usar o Network Address Translation e Virtualization no servidor Linux.
- `security` — Essa tabela só está disponível nas distribuições do Linux com o SELinux (ver Capítulo 24, “Aprimorando a segurança do Linux com o SELinux”). A tabela `security` é utilizada para filtrar pacotes de rede usando regras MAC (ver Capítulo 22). Essa tabela é utilizada com a tabela `filter`. As regras da tabela `security` só são aplicadas depois que as regras na tabela `filter` são aplicadas. Dessa forma, as regras MAC só são aplicadas depois que as regras DAC (ver Capítulo 22) são aplicadas, o que é consistente com a implementação do SELinux.

Entre todas as tabelas listadas, três focalizam o Network Address Translation. Portanto, a tabela `filter` é a tabela principal para focalizar a filtragem básica de pacotes do firewall.

Cadeias `Netfilter/iptables`

O firewall `netfilter/iptables` categoriza os pacotes de rede em categorias, chamadas cadeias.

Há cinco cadeias (categorias) para designar um pacote de rede:

- INPUT — Pacotes de rede chegando *ao* servidor Linux.
- FORWARD — Pacotes de rede chegando ao servidor Linux que devem ser *roteados* para outro lugar.
- OUTPUT — Pacotes de rede *saindo* do servidor Linux.
- PREROUTING — Usado pelo NAT, para modificar os pacotes de rede quando eles chegam no servidor Linux.
- POSTROUTING — Usado pelo NAT, para modificar os pacotes de rede antes de sair do servidor Linux.

A tabela `netfilter/iptables` com que você opta por trabalhar determinará as cadeias que estão disponíveis para categorizar os pacotes de rede. A Tabela 25.1 mostra quais cadeias estão disponíveis para cada tabela.

TABELA 25.1 Cadeias Disponíveis para cada Tabela `netfilter/iptables`

Tabela	Cadeias disponíveis
filter	INPUT, FORWARD, OUTPUT
raw	PREROUTING, OUTPUT, POSTROUTING
nftables	INPUT, FORWARD, PREROUTING, OUTPUT, POSTROUTING
nat	PREROUTING, OUTPUT
security	INPUT, FORWARD, OUTPUT

Depois que um pacote de rede é categorizado em uma cadeia específica, `iptables` pode determinar quais políticas ou regras se aplicam a esse pacote particular.

Regras, políticas e alvos `netfilter/iptables`,

Para cada pacote de rede, uma regra pode ser configurada definindo o que fazer com esse pacote individual. Pacotes de rede podem ser identificados de muitas maneiras pelo firewall `netfilter/iptables`. Alguns dessas maneiras incluem:

- Endereço IP da origem
- Endereço IP do destino
- Protocolo de rede
- Porta de entrada
- Porta de saída

■ Estado da rede

Se não existir nenhuma regra para um determinado pacote, então a política global é usada. Cada categoria ou cadeia de pacote tem uma política padrão.

Quando um pacote de rede corresponde a uma regra particular ou entra na política padrão, então a ação no pacote pode ocorrer. A ação tomada depende do alvo `iptables` que está configurado. Algumas das ações (alvos) que podem ser tomadas são:

- `ACCEPT` — Pacotes de rede são aceitos no servidor.
- `REJECT` — Pacotes de rede são descartados e não são permitidos no servidor. Uma mensagem de rejeição é enviada.
- `DROP` — Pacotes de rede são descartados e não são permitidos no servidor. Nenhuma mensagem de rejeição é enviada.

Embora `REJECT` dê uma mensagem de rejeição, `DROP` é mudo. Você poderia considerar o uso de `REJECT` para funcionários internos, que devem ser informados que você está rejeitando o tráfego de rede de saída deles e por quê. Considere o uso de `DROP` para o tráfego de entrada de modo que qualquer funcionário mal-intencionado não saiba que o tráfego está sendo bloqueado.

ca

alguns novos alvos adicionais mais sofisticados para `iptables`, como `QUEUE`. Você pode descobrir mais sobre esses alvos por meio do comando `man iptables`.

Na Figura 25.2, um pacote de rede de entrada é seguido pela tabela `filter` do firewall `netfilter/iptables`. Nesse exemplo, nenhuma regra específica foi definida para esse pacote de entrada, então o alvo da política, `ACCEPT`, é usado.

Exemplo de tabela de filtragem iptables

O utilitário `iptables` implementa um firewall de software utilizando a tabela `filter` por meio de políticas e regras. Agora que você tem uma compreensão geral da implementação do firewall de software, você pode começar a se aprofundar nos comandos específicos para implementar o firewall por meio do utilitário `iptables`.

Usando o utilitário iptables

Seu servidor Linux deve vir com o firewall instalado e funcionando. Mas uma boa ideia é verificar se ele está realmente ativado. Antes de verificar, primeiro você tem de entender que os serviços do firewall `netfilter/iptables` são ligeiramente diferentes, dependendo da distribuição do Linux:

- **Firewall netfilter/iptables do RHEL** — O serviço de interface de firewall em execução nessa distribuição é `iptables`. Para ver se esse serviço de firewall está em execução, digite **`service iptables status`** na linha de comando.
 - Para ativar o firewall, digite **`service iptables start`** na linha de comando.
 - Para desativar o firewall, digite **`service iptables stop`** na linha de comando.
- **Firewall netfilter/iptables do Fedora** — O serviço de interface de firewall em execução nessa distribuição é `iptables`. Para ver se esse serviço de firewall está em execução, digite **`systemctl status iptables.service`** na linha de comando.
 - Para ativar o firewall, digite **`systemctl start iptables.service`** na linha de comando.
 - Para desativar o firewall, digite **`systemctl stop iptables.service`** na linha de comando.
- **Firewall netfilter/iptables do Ubuntu** — O serviço de interface de firewall em execução nessa distribuição é `ufw`. Para ver se o serviço de firewall está em execução, digite **`sudo ufw status`** na linha de comando. O serviço `ufw` é uma interface para o utilitário `iptables` que não é executado como um serviço no Ubuntu. Você pode usar os comandos `ufw` para manipular as regras de firewall. Mas todos os comandos do utilitário `iptables` ainda são válidos para o Ubuntu:
 - Para ativar o firewall, digite **`sudo ufw enable`** na linha de comando.
 - Para desativar o firewall, digite **`sudo ufw disable`** na linha de comando.

Felizmente, depois de verificar o status e ativar ou desativar o firewall `netfilter/iptables`, terminam as diferenças entre as distribuições.

Para ver quais políticas e as regras estão atualmente em vigor para a tabela `filter`, digite **`iptables -t`**

filter -L na linha de comando. No exemplo a seguir, esse comando é inserido em um sistema Linux Mint. Observe que o comando `sudo` é necessário porque o utilitário `iptables` requer privilégios de root.

```
$ sudo iptables -t filter -L
[sudo] password for johndoe: *****

Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Há várias coisas a observar no exemplo anterior. Todas as cadeias disponíveis para a tabela `filter` estão listadas: `INPUT`, `FORWARD` e `OUTPUT`. Para cada cadeia, nenhuma regra é definida. Portanto, as políticas padrão da cadeia serão aplicadas a todos os pacotes de rede. Atualmente, todas as políticas padrão são configuradas como `ACCEPT`. Todos os pacotes de rede têm permissão para entrar, passar e sair. Um firewall nesse estado está essencialmente desativado.

ca

servidor Linux lidar com pacotes de rede IP v6, você poderá usar o utilitário `ip6tables` para gerenciar o firewall para endereços IPv6. O utilitário `ip6tables` é quase idêntico ao utilitário `iptables`. Para mais informações, digite `man ip6tables` na linha de comando.

Modificando políticas e regras iptables

Antes de começar a modificar o firewall `netfilter/iptables`, é útil entender algumas opções de comando. Abaixo estão algumas opções para modificar o firewall:

- **-t *tabela***

O comando `iptables` listado juntamente com essa chave é aplicado à *tabela*. Por padrão, a tabela `filter` é usada. Exemplo:

```
iptables -t filter -P OUTPUT DROP
```

- **-P *cadeia alvo***

Define a política geral de uma *cadeia* particular. São verificadas correspondências nas regras na *cadeia*. Se não houver nenhuma correspondência, o `target` listado da *cadeia* é usado. Exemplo:

```
iptables -P INPUT ACCEPT
```

- **-A *cadeia***

Define uma regra, chamada “regra anexada”, que será uma exceção à política geral para a *cadeia* designada. Exemplo:

```
iptables -A OUTPUT -d 10.140.67.25 -j REJECT
```

- **-I *n°_da_regra cadeia***

Insere uma regra anexada em um local específico, designado pelo *n°_da_regra*, na lista de regras

anexadas para a *cadeia* designada. Exemplo:

```
iptables -I 5 INPUT -s 10.140.67.23 -j DROP
```

■ **-D *cadeia n°_da_regra***

Exclui uma regra específica, designada pela *n°_da_regra*, da *cadeia* designada. Exemplo:

```
iptables -D INPUT 5
```

■ **-j *alvo***

Se os critérios na regra forem atendidos, o firewall deve saltar para esse *alvo* designado para processamento. Exemplo:

```
iptables -A INPUT -s 10.140.67.25 -j DROP
```

■ **-d *endereço_IP***

Atribui a regra listada para aplicar ao *endereço_IP* de destino designado. Exemplo:

```
iptables -A OUTPUT -d 10.140.67.25 -j REJECT
```

■ **-s *endereço_IP***

Atribui a regra listada para aplicar ao *endereço_IP* de origem designado. Exemplo:

```
iptables -A INPUT -s 10.140.67.24 -j ACCEPT
```

■ **-p *protocolo***

Atribui a regra listada para aplicar ao *protocolo* designado. Exemplo:

```
iptables -A INPUT -p icmp -j DROP
```

■ **--dport *n°_da_porta***

Atribui a regra listada para aplicar a certos pacotes de protocolo que entram no *n°_da_porta* designado. Exemplo:

```
iptables -A INPUT -p tcp --dport 22 -j DROP
```

■ **--sport *n°_da_porta***

Atribui a regra listada para aplicar a certos pacotes de protocolo saindo do *n°_da_porta* designada. Exemplo:

```
iptables -A OUTPUT -p tcp --sport 22 -j ACCEPT
```

■ **-m state --state *estado_da_rede***

Atribui a regra listada para aplicar ao(s) *estado(s)_da_rede* designado(s). Exemplo:

```
iptables -A INPUT -m state --state RELATED,ESTABLISH -j ACCEPT
```

Para ver como as opções `iptables` funcionam, considere o seguinte exemplo. Você tem um servidor Linux (Host-A) no endereço IP 10.140.67.23. Há dois outros servidores Linux na sua rede. Um deles é o Host-B no endereço IP 10.140.67.22 e o outro é o Host-C no endereço IP 10.140.67.25. Seu objetivo é:

■ Permitir ao Host-C acesso total ao Host-A

■ Bloquear conexões de login remoto usando `ssh` do Host-B para o Host-A

Definindo uma política de Drop. O código a seguir mostra o estado atual do firewall do Host-A. O firewall está completamente aberto, sem restrições implementadas. Nenhuma regra está definida e todas as políticas estão configuradas como ACCEPT.

```
# iptables -L
```

```
Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Primeiro, o que aconteceria se a política de INPUT fosse alterada de ACCEPT para DROP? Isso alcançaria o objetivo? Dê uma olhada no que acontece quando tentamos isso. Lembre-se de que se nenhuma regra estiver listada para um pacote de entrada, então a `policy` de `chain` será seguida. Essa mudança é feita para o firewall do Host-A no exemplo a seguir.

```
# iptables -P INPUT DROP
#
# iptables -L

Chain INPUT (policy DROP)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

ca

políticas, você não pode definir o alvo como REJECT. Isso falhará e você receberá a mensagem, “iptables: Bad policy name.” Use DROP como sua política em vez disso.

O Host-B tentará fazer um ping no Host-A e então tentará uma conexão ssh como mostrado no exemplo a seguir. Como você pode ver, ambas as tentativas falham. Como os comandos ping estão bloqueados, isso não alcança o objetivo de bloquear apenas conexões de login remoto usando ssh a partir do Host-B.

```
$ ping -c 2 10.140.67.23
PING 10.140.67.23 (10.140.67.23) 56(84) bytes of data.
--- 10.140.67.23 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1007ms
$
$ ssh root@10.140.67.23

ssh: connect to host 10.140.67.23 port 22: Connection timed out
```

Quando o Host C tenta fazer um ping no Host-A e criar uma conexão ssh, ambas as tentativas falham. Assim, confirma-se que a configuração de firewall, política INPUT é igual a DROP, não é aquilo que é necessário para alcançar o objetivo.

```
$ ping -c 2 10.140.67.23
PING 10.140.67.23 (10.140.67.23) 56(84) bytes of data.

--- 10.140.67.23 ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1008ms
$
$ ssh root@10.140.67.23

ssh: connect to host 10.140.67.23 port 22: Connection timed out
```

Bloqueando um endereço IP de origem. E se, em vez disso, apenas o endereço IP do Host-B fosse bloqueado? Isso permitiria que o Host-C entrasse em contato com o Host-A. Essa configuração alcançaria o objetivo desejado?

No exemplo a seguir, primeiro a política DROP deve ser alterada para ALLOW no iptables do Host-A. Depois disso, uma regra específica deve ser anexada para bloquear somente pacotes de rede provenientes do endereço IP do Host-B, 10.140.67.22.

```
# iptables -P INPUT ACCEPT
#
# iptables -A INPUT -s 10.140.67.22 -j DROP
#
# iptables -L

Chain INPUT (policy ACCEPT)
target     prot opt source                destination
DROP      all  --  10.140.67.22          anywhere

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

O Host-C agora pode com sucesso fazer ping e ssh no Host-A, satisfazendo um dos objetivos estabelecidos.

```
$ ping -c 2 10.140.67.23
PING 10.140.67.23 (10.140.67.23) 56(84) bytes of data.
64 bytes from 10.140.67.23: icmp_req=1 ttl=64 time=11.7 ms
64 bytes from 10.140.67.23: icmp_req=2 ttl=64 time=0.000 ms

--- 10.140.67.23 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1008ms
rtt min/avg/max/mdev = 0.000/5.824/11.648/5.824 ms
$ ssh root@10.140.67.23
root@10.140.67.23's password:
```

Mas o Host-B não pode fazer ping nem ssh no Host-A. Assim, a regra anexada não é exatamente o que

é necessário para alcançar todo o objetivo.

```
$ ping -c 2 10.140.67.23
```

```
PING 10.140.67.23 (10.140.67.23) 56(84) bytes of data.
```

```
--- 10.140.67.23 ping statistics ---
```

```
2 packets transmitted, 0 received, 100% packet loss, time 1007ms
```

```
$ ssh root@10.140.67.23
```

```
ssh: connect to host 10.140.67.23 port 22: Connection timed out
```

Bloqueando um protocolo e uma porta. E se, em vez de bloquear totalmente o endereço IP do Host-B, apenas as conexões para a porta `ssh` (porta 22) do endereço IP do Host-B fossem bloqueadas? Isso alcançaria o objetivo de permitir ao Host C acesso total ao Host-A, e só bloquear conexões `ssh` do Host-B?

No exemplo a seguir, as regras `iptables` para o Host-A são modificadas para tentar bloquear o endereço IP do Host-B na porta 22. Observe que a opção `--dport` deve acompanhar um determinado protocolo, por `-p tcp`. Antes de a nova regra ser adicionada, a regra do exemplo anterior deve ser excluída com a opção `-D`. Caso contrário, a regra do exemplo anterior seria usada pelo firewall `netfilter/iptables` para os pacotes de 10.140.67.22 (Host-B).

```
# iptables -D INPUT 1
#
# iptables -A INPUT -s 10.140.67.22 -p tcp --dport 22 -j DROP
#
# iptables -L

Chain INPUT (policy ACCEPT)
target     prot opt source                destination
DROP      tcp  --  10.140.67.22          anywhere          tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Primeiro, a nova regra `iptables` é testada a partir do Host-C para garantir que tanto as tentativas de `ping` como as conexões `ssh` permaneçam inalteradas. Funciona.

```
$ ping -c 2 10.140.67.23
```

```
PING 10.140.67.23 (10.140.67.23) 56(84) bytes of data.
```

```
64 bytes from 10.140.67.23: icmp_req=1 ttl=64 time=1.04 ms
```

```
64 bytes from 10.140.67.23: icmp_req=2 ttl=64 time=0.740 ms
```

```
--- 10.140.67.23 ping statistics ---
```

```
2 packets transmitted, 2 received, 0% packet loss, time 1000ms
```

```
rtt min/avg/max/mdev = 0.740/0.892/1.045/0.155 ms
```

```
$ ssh root@10.140.67.23
```

```
root@10.140.67.23's password:
```

Em seguida, a nova regra `iptables` é testada a partir do Host-B para garantir que `ping` funciona e as conexões `ssh` são bloqueadas. Também funciona!

```
$ ping -c 2 10.140.67.23
```

```
PING 10.140.67.23 (10.140.67.23) 56(84) bytes of data.  
64 bytes from 10.140.67.23: icmp_req=1 ttl=64 time=1.10 ms  
64 bytes from 10.140.67.23: icmp_req=2 ttl=64 time=0.781 ms
```

```
--- 10.140.67.23 ping statistics ---
```

```
2 packets transmitted, 2 received, 0% packet loss, time 1001ms  
rtt min/avg/max/mdev = 0.781/0.942/1.104/0.164 ms
```

```
$ ssh root@10.140.67.23
```

```
ssh: connect to host 10.140.67.23 port 22: Connection timed out
```

Mais uma vez, a matriz de controle de acesso da sua organização (ver Capítulo 22) vai ajudá-lo a criar as regras necessárias para o firewall `netfilter/iptables` no seu servidor Linux. E cada modificação deve ser testada em um ambiente virtual ou de teste antes de implementá-la no firewall dos sistemas de produção Linux.

Salvando uma configuração iptables

Depois de concluir o trabalho duro de criar as políticas e regras de configuração de firewall do seu servidor Linux, é recomendável salvá-las. Todas as modificações devem ser salvas no arquivo de configuração do `iptables`, `/etc/sysconfig/iptables`, porque esse é o arquivo usado na inicialização do sistema para carregar o firewall.

No exemplo a seguir, as modificações feitas anteriormente ainda estão no firewall. Antes de elas serem salvas no arquivo de configuração, uma cópia de backup do arquivo original é criada. Isso sempre é uma boa ideia. As modificações são, então, salvas com o comando `iptables-save`. Observe que a saída é direcionada para o arquivo `/etc/sysconfig/iptables` usando um símbolo de redirecionamento, `>` (ver a última linha do código no exemplo).

```
# iptables -L  
  
Chain INPUT (policy ACCEPT)  
target      prot opt source          destination  
DROP        tcp  --  10.140.67.22    anywhere    tcp dpt:ssh  
  
Chain FORWARD (policy ACCEPT)  
target      prot opt source          destination  
  
Chain OUTPUT (policy ACCEPT)  
target      prot opt source          destination  
  
# cp /etc/sysconfig/iptables/etc/sysconfig/iptables.bck  
#  
# iptables-save > /etc/sysconfig/iptables  
#
```

Você também pode remover todas as modificações no firewall `netfilter/iptables` atual usando a opção de limpeza, `iptables -F`. Depois de concluir isso, todas as regras (mas não as políticas) são

removidas, como mostrado no código a seguir. Isso é útil para testar políticas e regras individuais.

```
# iptables -F
#
# iptables -L

Chain INPUT (policy ACCEPT)
target     prot opt source                destination

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

Uma limpeza das regras não afeta o arquivo de configuração iptables. Para restaurar o firewall à sua condição original, use o comando `iptables-restore`. No exemplo a seguir, o arquivo de configuração iptables é redirecionado para o comando `restore` e a regra DROP original para `10.140.67.22` é restaurada.

```
# iptables-restore < /etc/sysconfig/iptables
#
# iptables -L

Chain INPUT (policy ACCEPT)
target     prot opt source                destination
DROP      tcp  --  10.140.67.22          anywhere           tcp dpt:ssh

Chain FORWARD (policy ACCEPT)
target     prot opt source                destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source                destination
```

ta

em um sistema Ubuntu, salvar e restaurar as modificações `netfilter/iptables` são muito semelhantes. Você ainda pode usar o comando `iptables-save` para criar um arquivo de configuração iptables a partir da configuração iptables atual e usar `iptables-restore` para restaurá-lo. Mas fazer uma configuração iptables salva ser carregada na inicialização é um pouco mais complicado. Não há um arquivo `/etc/sysconfig/iptables`. Há várias opções para carregar um arquivo de configuração na inicialização do sistema. Consulte no site da comunidade do Ubuntu em <https://help.ubuntu.com/community/IptablesHowTo> as várias opções.

Você também pode salvar suas regras de firewall `netfilter/iptables` para criar um relatório de auditoria. Revisar essas regras periodicamente deve ser parte da fase de Auditoria/Revisão do Ciclo de Vida do Sistema da sua organização.

Resumo

Proteger seu servidor Linux é fundamental em uma rede. Inerentemente, a maioria dos ataques maliciosos irá se originar de uma rede, especialmente da internet. Este capítulo cobriu alguns dos princípios básicos, como o modelo OSI, que você precisa a fim de começar esse processo.

Proteger seus serviços de rede pode ser simplificado depois que você determina e remove quaisquer serviços de rede desnecessários. O utilitário `nmap` irá ajudá-lo aqui. Além disso, você pode usar `nmap` para auditar os anúncios dos serviços de rede do servidor Linux. Essas auditorias ajudarão a determinar quais modificações no firewall são necessárias.

Para os serviços de rede necessários, um controle de acesso deve ser implementado. TCP Wrappers podem auxiliar nessa atividade. Configurando as permissões individualmente por serviço, o acesso pode ser permitido ou negado, fazendo o ajuste fino do acesso para cada serviço de rede.

Além disso, o firewall em um servidor Linux fornece controle de acesso. O firewall `netfilter/iptables` é um firewall de software baseado em host que pertence à camada de rede. Ele é gerenciado pelos utilitários `iptables` e `ip6tables`. Com esses utilitários, uma série de políticas e regras pode ser criada para cada pacote de rede que chega ao servidor Linux. Essas políticas e regras essencialmente compõem uma lista de controle de acesso para a rede de servidores Linux.

Obrigado por escolher a Bíblia do Linux para aprender mais sobre o sistema operacional Linux. Mas não pare aqui. Continue a aprender sobre o Linux e considere também retribuir. Há várias comunidades online dedicadas a software Linux livre e de código-fonte aberto, nas quais você pode compartilhar seu conhecimento recém-descoberto e permanecer em contato com os avanços no Linux. Aproveite o resto da sua jornada, e bem-vindo ao Linux!

Exercícios

Consulte o material neste capítulo para completar as tarefas a seguir. Se você empacar, soluções para as tarefas são mostrados no Apêndice B (embora no Linux costume haver várias maneiras de fazer uma tarefa). Tente resolver cada um dos exercícios antes de consultar as respostas. Essas tarefas supõem que você está executando um Fedora ou um Red Hat Enterprise Linux (embora algumas tarefas também funcionem em outros sistemas Linux).

1. Instale o utilitário Network Mapper no seu sistema Linux local.
2. Execute uma varredura TCP Connect no seu endereço de loopback local. Que portas têm um serviço em execução nelas?
3. Execute uma varredura UDP Connect no seu sistema Linux a partir de um sistema remoto.
4. Verifique se o daemon `ssh` no seu sistema Linux utiliza o suporte TCP Wrapper.
5. Usando os arquivos TCP Wrapper, permita acesso às ferramentas `ssh` no sistema Linux a partir de um sistema remoto designado. Negue qualquer outro acesso.
6. Determine as políticas e as regras atuais do firewall `netfilter/iptables` do seu sistema Linux.
7. Limpe as regras atuais do firewall do seu sistema Linux e então as restaure.
8. Para o firewall do seu sistema Linux, defina uma política de tabela de filtros para que a cadeia de entrada a rejeite.
9. Mude a política da tabela de filtros do firewall do sistema Linux de volta a `accept` para a

cadeia de entrada e então adicione uma regra para descartar todos os pacotes de rede a partir do endereço IP 10.140.67.23.

10. Sem descartar ou restaurar as regras do firewall do sistema Linux, remova a regra que você adicionou anteriormente.

Parte VI

Apêndices

NESTA PARTE

Apêndice A

Mídia

Apêndice B

Respostas dos exercícios

Mídia

NESTE APÊNDICE

Obtendo as distribuições do Linux

Criando um CD ou DVD

A menos que você tenha comprado um computador com Linux pré-instalado ou alguém o instalou para você, é necessário encontrar uma maneira de obter uma distribuição Linux e, então, instalá-la ou executá-la a partir de um Live CD em seu computador. Felizmente, as distribuições Linux estão amplamente disponíveis e vêm em uma variedade de formas.

Neste Apêndice, você aprenderá a:

- Obter algumas diferentes distribuições do Linux
- Criar um disco de inicialização para instalar sua distribuição
- Inicializar o Linux a partir de um pen drive USB

Para usar este livro eficientemente, você deve ter uma distribuição Linux à sua frente para trabalhar. É importante ser capaz de experimentar o Linux à medida que você lê. Então, experimente os exemplos e faça os exercícios.

Distribuições Linux estão mais comumente disponíveis nos sites das organizações que as produzem. As seções a seguir descrevem sites associados a distribuições Linux que oferecem imagens ISO que você pode baixar.

ta

ISO é uma imagem de disco que está formatada no formato de sistema de arquivos ISO 9660, um formato que é comumente usado com imagens de CD e DVD. Como esse é um formato bem conhecido, ele é lido por sistemas Windows, Mac e Linux.

Uma imagem ISO pode ser gravada em uma mídia de CD ou DVD, dependendo do tamanho da imagem, e, se estiver no seu sistema de arquivos, pode ser montada em um Linux no modo de loopback, o que permite ver ou copiar seu conteúdo.

Quando uma imagem ISO contém um Linux Live CD ou uma imagem da instalação, as imagens são inicializáveis. Isso significa que, em vez de iniciar um sistema operacional, como o Windows ou Linux, a partir do disco rígido do

computador, você pode dizer para seu computador inicializar a partir do CD ou DVD. Isso permite que você execute um sistema operacional totalmente diferente do que está instalado em seu disco rígido sem alterar ou danificar os dados no disco.

Obtendo o Fedora

Uma verdadeira página “Obtenha o Fedora” está disponível no site do Fedora (<http://fedoraproject.org/get-fedora>). Essa página contém links para download de ISOs, informações sobre o tipo de computador que você precisa e uma visão geral de como gravar uma imagem ISO.

ta

Quando quiser baixar o Fedora Desktop Live CD para trabalhar junto com este livro, porque a maior parte do livro vai trabalhar com essa distribuição. Você pode executá-lo como um Live CD sem arriscar afetar o disco rígido de seu computador até você se sentir confortável o suficiente para instalá-lo permanentemente.

Para testar os exemplos neste livro, usei o Fedora 16, Live CD desktop de 32 bits (GNOME). Se você tiver uma máquina de 64 bits, a ISO de 64 bits seria melhor ainda. Versões mais recentes do Fedora que vêm com um desktop GNOME devem funcionar igualmente bem. Eis um link para a ISO exata utilizada: <http://download.fedora.devel.redhat.com/pub/fedora/linux/releases/16/ISO-NetInst/16-i686-Live-Desktop.iso>

Você pode baixar as ISOs de que precisa a partir de qualquer computador que tenha um gravador de CD/DVD. Veja as descrições das ferramentas de gravação de CD/DVD disponíveis para Windows, Mac OS X e Linux mais adiante, neste apêndice.

A Figura A.1 mostra um exemplo da página Get Fedora.

imagens ISO do Fedora a partir da página Obtenha o Fedora.

O download padrão é uma imagem CD ISO de um Live CD Fedora Desktop (GNOME) para PC 32 bits. Você será capaz de inicializar uma imagem em seu computador e, se assim quiser, pode instalar permanentemente no disco rígido de seu computador. Para baixar essa imagem, faça o seguinte:

1. Selecione o botão Download Now. Um pop-up deve aparecer, perguntando o que você quer fazer com a ISO.
2. Selecione fazer download da ISO. Dependendo de suas configurações, você será perguntado se quer fazer download ou o arquivo simplesmente vai começar a ser baixado para uma pasta padrão.
3. Se você for perguntado sobre onde colocar a ISO, selecione uma pasta que tenha espaço suficiente para contê-la. Lembre-se de onde essa pasta está porque você vai precisar localizar a ISO quando for gravar o CD mais tarde.

Se precisar de mais informações sobre o que fazer com a imagem baixada, há links para ajudá-lo na página do Fedora que aparece. Na época em que escrevíamos este texto, havia links para “O que devo fazer com esse arquivo ISO?”, “Preciso de ajuda para executar ou instalar o Fedora” e “Quero saber mais sobre o Fedora”.

Você tem outras opções para baixar ISOs do Fedora. A partir da página Get Fedora, selecione o link More download options. Eis algumas alternativas ao Fedora Desktop Live CD de 32 bits:

- **Fedora 64 bits**— Se você sabe que tem um computador de 64 bits, baixe essa ISO, em vez da versão de 32 bits. Se você estiver em dúvida, a versão de 32 bits vai funcionar em qualquer tipo de computador, mas não vai tirar o máximo proveito dos ganhos de desempenho do sistema de 64 bits.
- **Spins** — Há versões especiais do Fedora que são referidas como **spins**. Exemplos de spins especiais do Fedora incluem aqueles que oferecem interfaces desktop diferentes daquelas que você obtém com o GNOME (KDE, LXDE ou Xfce). Se você estiver notando um mau desempenho do desktop GNOME, o LXDE e o XFCE são boas alternativas “leves”.
- **Mídia de instalação** — Selecione o link Download Methods para ver alguns métodos alternativos de download e imagens ISO diferentes. Em particular, a partir dessa página estão disponíveis imagens de DVD ISO que permitem fazer instalações do Fedora completas a partir do zero (semelhante à instalação Red Hat Enterprise Linux descrita no Capítulo 9, “Instalando o Linux”). Um Network Install CD, que é um pequeno CD que carrega o processo de instalação, pode ser usado para instalar através da rede ou outras mídias.

Depois de ter a imagem ISO que você deseja, vá para a descrição mais adiante neste capítulo de como gravar a imagem em CD ou DVD.

Obtendo o Red Hat Enterprise Linux

Muitas grandes empresas, agências governamentais e universidades utilizam o Red Hat Enterprise Linux para rodar seus aplicativos de missão crítica. Embora a maioria dos procedimentos deste livro execute bem no Fedora, há muitas referências a como as coisas são feitas diferentemente no Red Hat Enterprise Linux, porque, quando você for conseguir um emprego como administrador de sistemas Linux, na maioria dos casos, irá trabalhar com sistemas Red Hat Enterprise Linux.

Enquanto o código-fonte para o Red Hat Enterprise Linux está disponível gratuitamente, as ISOs contendo os pacotes que você instala (muitas vezes referidas como binários) só estão disponíveis para aqueles que têm contas de cliente no portal da Red Hat (<https://access.redhat.com>).

Se não tiver uma conta, você pode inscrever-se para um teste de 30 dias. Se você ou sua empresa tiver uma conta com a Red Hat, você pode baixar as ISOs de que você precisa. Vá para o site abaixo e siga as instruções para baixar uma Red Hat Enterprise Linux Server ISO ou inscreva-se para obter uma cópia de avaliação:

<https://access.redhat.com/downloads>

A Red Hat não oferece versões Live CD do Red Hat Enterprise Linux. Em vez disso, você pode baixar DVDs de instalação que podem ser instalados como descrito no Capítulo 9 deste livro.

ta

Se não conseguir obter um DVD de instalação do Red Hat Enterprise Linux, você pode obter uma experiência semelhante usando o DVD de instalação do CentOS. O CentOS não é exatamente o mesmo que o RHEL. Mas se você usar o DVD de instalação do CentOS 6.2 de um site espelho CentOS, o procedimento de instalação será semelhante ao descrito para o Red Hat Enterprise Linux no Capítulo 9.

Obtendo o Ubuntu

Muitas pessoas iniciantes em Linux começam baixando e instalando o Ubuntu. O Ubuntu tem uma enorme base de fãs e muitos colaboradores ativos. Se você tiver problemas com o Ubuntu, há grandes fóruns ativos, onde muitas pessoas estão dispostas a ajudá-lo a superar os problemas.

Se já tiver um sistema Ubuntu instalado, você pode acompanhar a maior parte da primeira metade deste livro. Você pode obter o Ubuntu com um desktop GNOME e sua interface shell padrão é semelhante ao bash (ou você pode alternar para o bash no Ubuntu para acompanhar os exemplos de shell neste livro).

À medida que o livro avança sobre tópicos relacionados com a administração do sistema, porém, os procedimentos podem não corresponder exatamente a como você faz no Ubuntu e, portanto, o Fedora e o Red Hat Enterprise Linux servem como melhores opções para aprender tópicos como administração do sistema, administração de servidores e segurança abordados em capítulos posteriores.

Para obter o Ubuntu, você pode baixar um Live CD ou uma mídia de instalação a partir da página Download Ubuntu: <http://www.ubuntu.com/download/ubuntu/download>

A Figura A.2 mostra um exemplo da página de download do Ubuntu.

as imagens ISO do Ubuntu Live CD ou escolha um download alternativo.

Como ocorre com o Fedora, a maneira mais fácil de baixar o Ubuntu é selecionar o Ubuntu Live CD de 32 bits, baixá-lo e gravá-lo. Veja como fazer isso a partir da página Ubuntu Download:

1. Clique no botão Start Download. Por padrão, isso baixa a mais recente imagem ISO do Ubuntu desktop GNOME Live CD para computadores de 32 bits.
2. Você será perguntado se quer fazer o download da imagem ISO ou simplesmente o arquivo começará a ser baixado para uma pasta padrão.
3. Se lhe for perguntado onde colocar a ISO, selecione uma pasta que tenha espaço suficiente para armazená-la. Lembre-se de onde essa pasta está porque você vai precisar localizar a ISO quando for gravar o CD mais tarde.

Após a conclusão do download, grave a imagem ISO em um CD usando os procedimentos descritos mais adiante na seção “Criando CDs e DVDs Linux”.

Outros tipos de mídia de instalação do Ubuntu também estão disponíveis. Eis alguns exemplos:

- **Ubuntu Windows Installer** — Para rodar o Ubuntu como um aplicativo dentro de um sistema Windows, você pode baixar essa versão especial do Ubuntu e seguir o link para informações na página sobre como instalá-lo:

<http://www.ubuntu.com/download/ubuntu/windows-installer>

- **Downloads alternativos** — Você pode selecionar diferentes versões do Ubuntu para download na página Alternative Downloads. Há versões específicas para servidores, desktops e netbooks. Também há imagens de DVD que oferecem mais pacotes do que você obtém com os Live CDs. A página Alternative Downloads está localizada em:

<http://www.ubuntu.com/download/ubuntu/alternative-download>

Criando CDs e DVDs Linux

Depois de baixar uma imagem de CD ou DVD do Linux, você pode usar várias ferramentas para criar CDs ou DVDs inicializáveis para instalar ou apenas executar o Linux “Live” a partir dessas mídias. Antes de começar, você deve ter o seguinte:

- **Imagens ISO de DVD ou CD** — Baixe as imagens ISO para seu computador que representam o DVD ou CD físicos que você acabará gravando.
- **DVDs/CDs virgens** — Você precisa de DVDs ou CDs virgens para gravar as imagens. CDs armazenam até cerca de 700MB; DVDs armazenam até cerca de 4,7GB (camada única).

- **Gravador de CD** — Você precisa de uma unidade que seja capaz de gravar CDs ou DVDs, dependendo do que você está gravando. Nem todas as unidades de CD/DVD podem gravar CDs (especialmente as mais antigas). Assim, você pode precisar encontrar um computador com uma unidade que tem essa capacidade.

As próximas seções descrevem como gravar CDs e DVDs inicializáveis a partir dos sistemas Windows, Mac OS X e Linux.

Gravando CDs/DVDs no Windows

Se tiver baixado a imagem ISO do Linux para um sistema Windows, você pode gravar a imagem em CD ou DVD de maneiras diferentes, dependendo de qual versão do Windows você está usando. Eis alguns exemplos:

- **Windows 7** — No Windows 7, a função de imagens ISO para gravação de CD ou DVD está integrada ao sistema operacional. Uma vez que uma imagem ISO é baixada, basta inserir o CD apropriado ou DVD na unidade de seu computador (assumindo que o disco é gravável), clicar com o botão direito do mouse no ícone de imagem ISO a partir da pasta em que você a baixou e selecionar Gravar imagem de disco. Quando a janela Gravador de imagem de disco do Windows aparecer, selecione Gravar para gravar a imagem.
- **Roxio Creator** — Esse aplicativo Windows da Roxio contém muitos recursos para ripar e gravar CDs e DVDs. Você pode ler sobre o produto aqui:

<http://www.roxio.com/enu/products/creator/>

- **Nero CD/DVD-ROM** — Nero é outro software popular de gravação de CD/DVD para sistemas Windows. Você pode descobrir mais sobre o Nero aqui:

<http://www.nero.com>

Gravando CDs/DVDs em um sistema Mac OS X

Assim como o Linux, o Mac OS X tem software de gravação de CD/DVD integrado ao sistema operacional. Para gravar uma imagem ISO para disco em um sistema Mac OS X, siga estes passos:

1. Baixe a imagem ISO que você quer em seu sistema Mac OS X. Um ícone representando a ISO deve aparecer em seu desktop.
2. Insira um CD ou DVD virgem em seu gravador de CD/DVD, conforme for apropriado para o tamanho da imagem.
3. Clique com o botão direito do mouse no ícone que representa a ISO Linux que você acabou de baixar e selecione Burn “Linux” to Disk. Uma janela pop-up aparece, perguntando se você tem certeza de que quer gravar a imagem.
4. Preencha o nome que você quer dar à ISO e a velocidade de gravação. Então, selecione Burn. A imagem começa a gravar em disco.

5. Após a imagem ter sido queimada, ejetar o disco e você está pronto para iniciar o CD ou DVD em um computador apropriado.

Gravando CDs/DVDs no Linux

O Linux tem ferramentas gráficas e também linha de comando para gravar imagens de CD e DVD em mídia física. Os exemplos desta seção mostram como usar o K3b a partir do desktop ou o `cdrecord` (ou o `wodim`) para gravar imagens ISO em CD ou DVD.

Gravando CDs a partir de um desktop Linux

Veja como criar CDs inicializáveis do Linux a partir de um sistema Linux em execução (como o Fedora) usando o K3b. O K3b vem com o desktop KDE, mas também roda no desktop GNOME.

1. Faça o download das imagens ISO que você quer para o disco rígido de seu computador. (A imagem do CD tem cerca de 700MB de tamanho. Imagens de DVD de camada única estão abaixo de 4,7GB).
2. Abra um aplicativo de gravação de CD/DVD. Para esse procedimento, recomendo o K3b CD and DVD Kreator (<http://www.k3b.org>). No Fedora, selecione o menu Applications e escolha Sound & Video ⇒ K3b (ou digite **k3b** em uma janela Terminal). A janela K3b – CD e DVD Kreator aparece.

ta

K3b não estiver instalado em seu sistema Linux, você pode instalá-lo na maioria das distribuições Linux. Para o Fedora, instale o K3b digitando `yum install k3b` como usuário root em uma janela de Terminal.

3. A partir da janela do K3b, selecione Tools ⇒ Burn CD ISO Image para gravar uma imagem de CD, ou Tools ⇒ Burn DVD ISO Image para gravar uma imagem de DVD. Você é solicitado a escolher um arquivo de imagem.
4. Procure a imagem que você acabou de baixar ou copiar para o disco rígido e selecione-a. Depois de selecionar a imagem desejada, a janela Burn CD Image aparece, assim como uma soma de verificação da imagem. (Muitas vezes, você pode comparar o número da soma de verificação que aparece com o número em um arquivo md5 do diretório de download, onde você obteve o Live CD, para ter certeza de que a imagem do CD não foi corrompida.) A Figura A.3 mostra a janela Burn CD Image pronta para gravar uma imagem do Fedora.

K3b para gravar seus CDs ou DVDs Linux.

5. Insira um CD ou DVD virgem na unidade de CD/DVD, a qual pode ser uma combinação de unidade de CD e DVD. (Se uma janela CD/DVD Creator aparecer, você pode simplesmente fechá-la.)
6. Verifique as configurações na janela Burn CD Image (muitas vezes, as padrões são boas configurações, mas você pode querer diminuir a velocidade se tiver problemas de gravação). Você também pode selecionar a caixa de seleção Simulate para testar a gravação antes de realmente “queimar” o CD/DVD. Clique em Start para continuar.
7. Quando a gravação terminar, ejete o CD (ou ele pode ser automaticamente ejetado) e o rotule de forma adequada (informações como o nome da distribuição, número de versão, data e nome da imagem ISO).

Agora você está pronto para começar a instalar (ou iniciar) a distribuição Linux que acabou de gravar.

Gravando CDs a partir de uma linha de comando do Linux

Se não tiver nenhuma GUI ou não se importar de trabalhar a partir do shell, você pode usar o comando `cdrecord` para gravar as ISOs. Com um CD virgem inserido e a imagem ISO que você quer gravar no diretório atual, você pode usar a simples linha de comando a seguir para gravar uma imagem de CD no CD virgem usando `cdrecord`:

```
# cdrecord -v nomeDaImagem.iso
```

Consulte a página `man` de `cdrecord` (`man cdrecord`) para outras opções disponíveis com o comando `cdrecord`.

Iniciando o Linux a partir de um pen drive USB

Em vez de gravar imagens ISO em um CD ou DVD, você pode colocar seu sistema Linux em um pen drive USB. Pen drives USB oferecem a vantagem de ser regraváveis, bem como de fácil leitura, de modo que você pode salvar seu conteúdo entre as sessões. A maioria dos computadores modernos pode iniciar a partir de um pen drive USB, embora você possa ter de interromper o processo de inicialização para instruir a BIOS a inicializar a partir da unidade USB em vez do disco rígido ou CD/DVD.

Você pode encontrar os procedimentos para colocar o Fedora e Ubuntu em um pen drive USB nos seguintes locais:

- **Fedora em um pen drive** — Usando uma ferramenta chamada Live USB Creator (<https://fedorahosted.org/liveusb-creator/>), você pode instalar uma imagem

ISO do Fedora em um pen drive USB. Para executar o Fedora a partir dessa unidade, conecte o pen drive em uma porta USB de seu computador, reinicie o computador, interrompa a BIOS quando ela estiver inicializando (possivelmente com F12) e selecione inicializar a partir de um pen drive USB. O procedimento para usar Live USB Creator encontra-se em:

http://docs.fedoraproject.org/en-US/Fedora/16/html/Installation_Guide/Making_USB_Media.html

- **Ubuntu em um pen drive USB** — O Ubuntu tem procedimentos para a criação de uma unidade USB inicializável que funciona em Windows, Mac OS X e Linux. Para saber como fazer isso, vá até a página de download do Ubuntu e selecione o tipo de mídia (USB) e o sistema operacional (Windows, Mac ou Linux) que quer usar para criar o pen drive USB inicializável:

<http://www.ubuntu.com/download/ubuntu/download>

site o site *Linux Bible*

Para encontrar links para várias distribuições Linux, dicas sobre como obter certificação Linux e correções deste livro à medida que elas se tornam disponíveis, visite <http://www.wiley.com/WileyCDA/WileyTitle/productCd-111821854X.html>.

Respostas dos Exercícios

Este apêndice fornece respostas para cada um dos exercícios dos capítulos. Há muitas maneiras de completar tarefas no Linux. O que é fornecido aqui são sugestões.

Alguns dos exercícios exigem que você modifique arquivos de sistema, o que poderia mudar o funcionamento básico do sistema ou até mesmo torná-lo não inicializável. Portanto, recomendamos que você faça os exercícios em um sistema Linux que possa modificar e apagar se algo der errado.

Capítulo 2: Criando o desktop Linux perfeito

A seção a seguir detalha algumas maneiras de fazer essas tarefas nos desktops GNOME 2 e 3.

1. Para começar, você precisa de um sistema Linux à sua frente para fazer os procedimentos neste livro. Um sistema instalado é preferível porque você não perde as alterações ao reiniciar. Para começar, você pode usar um Fedora Live CD (ou sistema instalado), um sistema Ubuntu instalado ou um sistema Red Hat Enterprise Linux instalado. Eis suas opções:
 - **Fedora Live CD (GNOME 3)** — Obtenha um Fedora Live CD como descrito no Apêndice A. Execute-o como um Live CD, como descrito na seção “Iniciando com o GNOME Fedora Live CD”, do Capítulo 2, ou instale-o e execute-o a partir do disco rígido, como descrito no Capítulo 9, “Instalando o Linux”.
 - **Ubuntu (GNOME 3)** — Instale o Ubuntu e o GNOME Shell, como descrito no início do Capítulo 2.
 - **Red Hat Enterprise Linux (GNOME 2)** — Instale o Red Hat Enterprise Linux, como descrito no Capítulo 9.
2. Para iniciar o navegador web Firefox e ir para a página inicial do GNOME (<http://gnome.org>), há alguns passos fáceis de seguir. Se sua rede não estiver funcionando, consulte o Capítulo 14, “Administrando redes”, a fim de obter ajuda para se conectar a redes com e sem fio.
 - **GNOME 3**

No GNOME 3, você pode pressionar a tecla Windows para alcançar a tela Overview. Então, digite Firefox para destacar apenas o ícone do navegador Firefox.

Pressione Enter para iniciá-lo. Digite **http://gnome.org** na caixa de localização e pressione Enter.

■ GNOME 2

No GNOME 2, selecione o ícone do Firefox na barra de menu superior. Digite **http://gnome.org** na caixa de localização e pressione Enter.

3. Para escolher um fundo de tela de que você gostou no site de arte GNOME (<http://art.gnome.org/backgrounds>), baixá-lo para a pasta Pictures e selecioná-lo como seu fundo atual no GNOME 2 e 3, faça o seguinte:
 - Digite **http://art.gnome.org/backgrounds** na caixa de localização do Firefox e pressione Enter.
 - Encontre um fundo de que você goste e clique em GO para exibi-lo.
 - Clique com o botão direito do mouse na imagem e selecione Set as Desktop Background.
 - No menu pop-up que aparece, selecione a posição e a cor da imagem de fundo.
 - Selecione o botão Set Desktop Background. A imagem é usada como fundo do desktop e copiada para o arquivo `Firefox_wallpaper.png` em seu diretório inicial.
4. Para iniciar uma janela do Nautilus File Manager e movê-la para o segundo espaço de trabalho em seu desktop, faça o seguinte:
 - Para o GNOME 3
 - Pressione a tecla Windows.
 - Clique no ícone de arquivos no Dash (lado esquerdo) e arraste-o para um espaço de trabalho não utilizado no lado direito. Uma nova instância do Nautilus inicia nesse espaço de trabalho.
 - Para o GNOME 2
 - Abra a pasta Home a partir do desktop GNOME 2 (clique duplo).
 - Clique com o botão direito do mouse na barra de título do Nautilus que aparece e selecione Move to Workspace Right ou Move to Another Workspace (você pode selecionar qual espaço de trabalho você quer na lista).
5. Para encontrar a imagem que você baixou para usar como fundo do desktop e abri-la em qualquer visualizador de imagem, primeiro vá para sua pasta Home.

A imagem deve aparecer nessa pasta quando você abrir o Nautilus. Basta dar um clique duplo no ícone `Firefox_wallpaper.png` para abrir a imagem no visualizador de imagens padrão. Se você tiver múltiplos visualizadores de imagens em seu sistema, clique com botão direito no ícone e selecione o aplicativo que você quer usar para abri-lo.
6. Para alternar entre o espaço de trabalho com o Firefox e o espaço de trabalho com o

gerenciador de arquivos Nautilus é bem simples.

Se você fez os exercícios anteriores corretamente, o Nautilus e o Firefox devem estar em diferentes espaços de trabalho. Veja como você pode se mover entre esses espaços de trabalho no GNOME 2 e 3:

- GNOME 3

Pressione a tecla Windows e clique duas vezes no espaço de trabalho que você quer na coluna direita. Como alternativa, você pode ir diretamente para o aplicativo desejado pressionando Alt+Tab e pressionando Tab novamente para destacar o aplicativo que quer abrir.

- GNOME 2

Selecione o espaço de trabalho que você quer com o mouse clicando na pequena representação do espaço de trabalho no lado direito da parte inferior do painel. Se você habilitou Desktop Effects (System ⇒ Preferences Desktop Effects ⇒ Compiz), tente pressionar Ctrl+Alt+seta para a direita (ou seta para a esquerda) para girar para o espaço de trabalho seguinte.

7. Para abrir uma lista de aplicativos instalados em seu sistema e selecionar um visualizador de imagens para abrir a partir dessa lista usando o mínimo possível de cliques ou teclas, faça o seguinte:

- No GNOME 3

Mova o mouse para o canto superior esquerdo da tela para obter a tela de Overview. Selecione Applications, escolha Graphics na coluna da direita e, então, clique em Image Viewer.

- No GNOME 2

Selecione Applications ⇒ Graphics ⇒ Image Viewer para abrir uma janela do visualizador de imagens no desktop.

8. Para alterar a exibição das janelas em seu espaço de trabalho atual para visualizações menores dessas janelas, faça o seguinte:

- No GNOME 3

Com várias janelas abertas em vários espaços de trabalho, pressione e segure as teclas Alt+Tab. Com a tecla Alt pressionada, pressione Tab até destacar o aplicativo que você quer. Solte a tecla Alt para selecioná-lo. (Observe que os aplicativos que não estão na área de trabalho atual estão à direita de uma linha dividindo os ícones.)

- No GNOME 2

Com várias janelas abertas em vários espaços de trabalho, pressione e segure as teclas Ctrl+Alt+TAB. Com as teclas Ctrl+Alt Tab pressionadas, pressione Tab até destacar o aplicativo desejado. Solte as teclas Ctrl e Alt para selecioná-lo.

9. Para carregar um leitor de música a partir de seu desktop usando apenas o teclado,

faça o seguinte:

- No GNOME 3

- Pressione a tecla Windows para ir para a tela Overview.
- Digite **Rhyth** (até que o ícone apareça e seja destacado) e pressione Enter. (No Ubuntu, se você não tem o Rhythmbox instalado, digite **Bansh** para abrir o Banshee Media Player).

- No GNOME 2

Pressione Alt+F2. A partir da caixa Run Application que aparece, digite **rhythmbox** e pressione Enter.

10. Para tirar uma foto de seu desktop usando apenas o teclado, pressione a tecla Print Screen para fazer uma captura de tela de seu desktop inteiro, tanto no GNOME 3 como GNOME 2. Pressione Ctrl+Print Screen para tirar uma captura de tela apenas da janela atual.

Capítulo 3: Utilizando o shell

1. Para alternar entre os consoles virtuais e voltar para o desktop:

- Segure Ctrl+Alt e pressione F2 (Ctrl+Alt+F2). Um console baseado em texto deve aparecer.
- Digite seu nome de usuário (pressione Enter) e senha (pressione Enter).
- Digite alguns comandos, como **id**, **pwd** e **ls**.
- Digite **exit** para sair do shell e voltar ao prompt de login.
- Pressione Ctrl+Alt+F1 para voltar ao console virtual que contém seu desktop. (Em diferentes sistemas Linux, o desktop pode estar em diferentes consoles virtuais. Ctrl+Alt+F7 é outro jeito comum de encontrá-lo.)

2. Para sua janela Terminal, torne a fonte vermelha e o fundo amarelo.

- No desktop GNOME, selecione Applications ⇒ System Tools ⇒ Terminal para abrir uma janela Terminal.
- Na janela Terminal, selecione Edit ⇒ Profiles.
- Com Default destacado na janela Profiles, selecione Edit.
- Selecione a guia Colors e desmarque a caixa Use colors from system theme.
- Selecione a caixa ao lado Text Color, clique na cor vermelha que você quiser na roda de cores e clique em OK.
- Selecione a caixa ao lado de Background Color, clique na cor amarela que você quer na roda de cores e clique em OK.

- Clique em Close em cada janela para voltar para a janela Terminal com as novas cores.
 - Volte e selecione novamente a caixa Use colors from system theme para restaurar as cores padrão Terminal.
3. Encontre o comando `mount` e a página `man tracepath`.
- Execute `type mount` para ver que a localização do comando de `mount` é `/bin/mount`.
 - Execute `locate tracepath` para ver que a página `tracepath` está em `man /usr/share/man/man8/tracepath.8.gz`.
4. Execute, recupere e altere esses comandos como descrito:
- ```
$ cat /etc/passwd
$ ls $HOME
$ date
```
- Pressione a seta para cima até ver o comando `cat /etc/passwd`. Se o cursor não estiver no final da linha, pressione `Ctrl+E` para chegar lá. Pressione `backspace` sobre a palavra `passwd`, digite a palavra **group** e pressione `Enter`.
  - Digite **man ls** e localize a opção para listar pelo tempo (`-t`). Pressione a seta para cima até ver o comando `ls $HOME`. Use a seta para a esquerda ou `Alt+B` para posicionar o cursor à esquerda de `$HOME`. Digite `-t`, para que a linha apareça como `ls -t $HOME`. Pressione `Enter` para executar o comando.
  - Digite **man date** para ver a página `man date`. Use a seta para cima para recuperar o comando `date` e adicionar o indicador de formato que você encontrou. Um indicador de formato `%D` fornecerá os resultados de que você precisa:
- ```
$ date +%D
12/08/11
```
5. Use a conclusão de comando (tecla `Tab`) para digitar **basename /usr/share/doc/**. Digite **basen<Tab>/u<Tab>sh<Tab>do<Tab>** para obter o nome de base `/usr/share/doc/`.
6. Redirecione `/etc/services` para o comando `less`:
- ```
$ cat /etc/services | less
```
7. Faça a saída do comando `date` aparecer no seguinte formato: Hoje é quinta-feira, 8 de dezembro, 2011.
- ```
$ echo "Today is $(date +%A, %B %d, %Y)"
```
8. Veja as variáveis para encontrar seu atual nome de usuário, `hostname`, `shell` e diretórios.
- ```
$ echo $HOSTNAME
```

```
$ echo $USERNAME
$ echo $SHELL
$ echo $HOME
```

9. Adicione um alias `mypass` permanente que exibe o conteúdo do arquivo `/etc/passwd`.

- Digite **nano** `$HOME/.bashrc`.

- Mova o cursor para uma linha em branco na parte inferior da página (pressione Enter para criar uma nova linha se necessário).

- Em sua própria linha, digite **alias** `m="cat /etc/passwd"`.

- Digite Ctrl+S para salvar e Ctrl+X para sair do arquivo.

- Digite **source** `$HOME/.bashrc`.

- Digite **alias** `m` para se certificar de que o apelido foi criado corretamente:

```
ias m='cat /etc/passwd'.
```

- Digite `m` (o arquivo `/etc/passwd` é exibido na tela).

10. Para exibir a página `man` para a chamada de sistema de montagem, use o comando `man -k` para encontrar páginas `man` que incluem a palavra `mount` (usar `^` assegura que apenas os comandos que começam com a palavra `mount` sejam exibidos). Então, use o comando `mount` com o número da seção correta (2) para obter a página `man` `mount` apropriada:

```
$ man -k ^mount
mount (2) - mount file system
mount (8) - mount a filesystem
mountpoint (1) - see if a directory is a mountpoint
mountstats (8) - Displays NFS client per-mount
statistics
$ man 2 mount
MOUNT(2) Linux Programmer's Manual MOUNT(2)
NAME
 mount - mount file system
SYNOPSIS
 #include <sys/mount.h>
.
.
.
```

## Capítulo 4: Movendo-se pelo sistema de arquivos

---

1. Crie o diretório de projetos, crie nove arquivos vazios (`house1` para `house9`) e liste apenas esses arquivos.

```
$ mkdir $HOME/projects/
$ touch $HOME/projects/house{1..9}
$ ls $HOME/projects/house{1..9}
```

2. Crie o caminho do diretório `$HOME/projects/houses/doors/` e crie alguns arquivos vazios nesse caminho.

- ```

$ cd
$ mkdir projects/houses
$ touch /home/joe/houses/bungalow.txt
$ mkdir $HOME/projects/houses/doors/
$ touch $HOME/projects/houses/doors/bifold.txt
$ mkdir -p $HOME/projects/outdoors/vegetation/
$ touch projects/outdoors/vegetation/landscape.txt

```
3. Copie os arquivos `house1` e `house5` para o diretório `$HOME/projects/houses/..`.

```

$ cp $HOME/projects/house[15] $HOME/projects/houses

```
 4. Copie recursivamente o diretório `/usr/share/doc/initscripts*` para o diretório `$HOME/projects/`.

```

$ cp -ra /usr/share/doc/initscripts*/ ~ /projects/

```
 5. Liste recursivamente o conteúdo do diretório `$HOME/projects/`. Redirecione a saída para o comando `less` para que você possa percorrer a saída página por página.

```

$ ls -lR $HOME/projects/ | less

```
 6. Remova os arquivos `house6`, `house7` sem ser perguntado se realmente deseja excluí-los.

```

$ rm -f $HOME/projects/house[678]

```
 7. Mova `house3` e `house4` para o diretório `$HOME/projects/houses/doors`.

```

$ mv projects/house{3,4} projects/houses/doors/

```
 8. Exclua o diretório `$HOME/projects/houses/doors` e seu conteúdo.

```

$ rm -rf projects/houses/doors/

```
 9. Altere as permissões sobre o arquivo `$HOME/projects/house2` de modo que ele possa ser lido e gravado pelo usuário que possui o arquivo, apenas lido pelo grupo e que os outros não tenham permissões sobre ele.

```

$ chmod 640 $HOME/projects/house2

```
 10. Altere recursivamente as permissões do diretório `$HOME/projects/` para que ninguém tenha permissão de gravação para qualquer arquivo ou diretório abaixo desse ponto no sistema de arquivos.

```

$ chmod -R a-w $HOME/projects/

```

Capítulo 5: Trabalhando com arquivos de texto

1. Siga estes passos para criar o arquivo de serviços `/tmp/` e, então, edite-o de forma que “WorldWideWeb” apareça como “World Wide Web”.


```
$ cp /etc/services /tmp
$ vi /tmp/services
/WorldWideWeb<Enter>
cwWorld Wide Web<Esc>
```

As duas linhas a seguir mostram o antes e depois.

```
http 80/tcp www www-http # WorldWideWeb HTTP
http 80/tcp www www-http # World Wide Web HTTP
```

2. Uma maneira de mover o parágrafo em seu arquivo `/tmp/services` é pesquisar a primeira linha do parágrafo, excluir cinco linhas (`5dd`), ir para o final do arquivo (`G`) e colocar o texto (`p`):

```
$ vi /tmp/services
/Note that it is<Enter>
5dd
G
p
```

3. Para usar o modo `ex` a fim de procurar todas as ocorrências do termo `tcp` (com distinção entre maiúsculas e minúsculas) em seu arquivo `/tmp/services` e alterá-las para `WHATEVER`, você pode digitar o seguinte:

```
$ vi /tmp/services
:g/tcp/s//WHATEVER/g<Enter>
```

4. Para pesquisar no diretório `/etc` cada arquivo chamado `passwd` e redirecionar os erros de sua pesquisa para `/dev/null`, você pode digitar o seguinte:

```
$ find /etc -name passwd 2> /dev/null
```

5. Crie um diretório no seu diretório inicial chamado `TEST`. Crie três arquivos nesse diretório com os nomes `one`, `two` e `three`, e atribua permissões plenas de leitura/gravação/execução sobre esses arquivos para todo mundo (usuário, grupo e outros). Construa um comando `find` que localize esses arquivos e outros arquivos que têm permissões de gravação abertas para os “outros” a partir do seu diretório inicial e abaixo.

```
$ mkdir $HOME/TEST
$ touch $HOME/TEST/{one,two,three}
$ chmod 777 $HOME/TEST/{one,two,three}
$ find $HOME -perm -002 -type f -ls
148120 0 -rwxrwxrwx 1 chris chris 0 Jan 1 08:56
/home/chris/TEST/two
148918 0 -rwxrwxrwx 1 chris chris 0 Jan 1 08:56
home/chris/TEST/three
147306 0 -rwxrwxrwx 1 chris chris 0 Jan 1 08:56
```

```
/home/chris/TEST/one
```

6. Localize arquivos no diretório `/usr/share/doc` que não tenham sido modificados há mais de 300 dias.

```
$ find /usr/share/doc -mtime +300
```

7. Crie um diretório `/tmp/FILES`. Localize todos os arquivos no diretório `/usr/share` que tenham mais que 5MB e menos de 10MB e copie-os para o diretório `/tmp/FILES`.

```
$ mkdir /tmp/FILES
```

```
$ find /usr/share -size +5M -size -10M -exec cp {}  
/tmp/FILES \;
```

```
$ du -sh /tmp/FILES/*
```

```
7.0M    /tmp/FILES/cangjie5.db
```

```
5.4M    /tmp/FILES/cangjie-big.db
```

```
8.3M    /tmp/FILES/icon-theme.cache
```

8. Localize todos os arquivos no diretório `/tmp/FILES` e faça uma cópia de backup de cada arquivo no mesmo diretório. Use o nome de cada arquivo existente e apenas anexe `.mybackup` para criar cada arquivo de backup.

```
$ find /tmp/FILES/ -type f -exec cp {} {}.mybackup \;
```

9. Instale o pacote `kernel-doc` no Fedora ou no Red Hat Enterprise Linux. Usando a pesquisa `grep`, pesquise dentro dos arquivos contidos no diretório `/usr/share/doc/kernel-doc*` pelo termo `e1000` (sem distinção entre maiúsculas e minúsculas) e liste os nomes dos arquivos que contenham esse termo.

```
# yum install kernel-doc
```

```
$ cd /usr/share/doc/kernel-doc*
```

```
$ grep -rli e1000.
```

```
./Documentation/powerpc/booting-without-of.txt
```

```
/usr/share/doc/kernel-doc-
```

```
2.6.32/Documentation/networking/e100.txt
```

10. Pesquise pelo termo `e1000` novamente no mesmo local, mas desta vez liste cada linha que contém o termo e destaque o termo com cores.

```
$ cd /usr/share/doc/kernel-doc-*
```

```
$ grep -ri --color e1000 .
```

Capítulo 6: Gerenciando processos em execução

1. Para listar todos os processos em execução em seu sistema com um conjunto completo

de colunas, enquanto redireciona a saída para `less`, digite o seguinte:

```
$ ps -ef | less
```

2. Para listar todos os processos em execução no sistema e ordená-los pelo nome do usuário que executa cada processo, digite o seguinte:

```
$ ps -ef --sort=user | less
```

3. Para listar todos os processos em execução no sistema com os nomes de coluna ID do processo, nome de usuário, nome do grupo, valor de `nice`, tamanho da memória virtual, o tamanho da memória residente e de comando, digite o seguinte:

```
$ ps -eo 'pid,user,group,nice,vsz,rss,comm' | less
  PID USER      GROUP      NI   VSZ   RSS  COMMAND
   1 root         root        0 19324 1236  init
   2 root         root        0     0     0  kthreadd
   3 root         root        -     0     0  migration/0
   4 root         root        0     0     0  ksoftirqd/0
```

4. Para executar o comando `top` e, então, alternar entre a ordem por uso de CPU e a ordem por consumo de memória, digite o seguinte:

```
$ top
```

```
P
```

```
M
```

```
P
```

```
M
```

5. Para iniciar o processo `gedit` a partir de seu desktop e usar a janela System Monitor para eliminar esse processo, faça o seguinte:

```
$ gedit &
```

Então, selecione Applications ⇒ System Tools ⇒ System Monitor. Localize o processo `gedit` na guia Processes (você pode classificar por ordem alfabética, para facilitar a localização, clicando no cabeçalho Process Name). Clique com o botão direito do mouse no comando `gedit` e, então, selecione End Process ou Kill Process e a janela `gedit` em sua tela deve desaparecer.

6. Para executar o processo `gedit` e usar o comando `kill` a fim de enviar um sinal para pausar (parar) esse processo, digite o seguinte:

```
$ gedit &
```

```
[1] 21532
```

```
$ kill -SIGSTOP 21578
```

7. Para usar o comando `killall` a fim de instruir o comando `gedit` (pausado no exercício anterior) a continuar a trabalhar, faça o seguinte:

```
$ killall -SIGCONT gedit
```

Verifique se o texto digitado depois que `gedit` foi pausado agora aparece na janela.

8. Para instalar o comando `xeyes`, executá-lo cerca de 20 vezes em segundo plano e executar `killall` para eliminar todos os 20 processos `xeyes` de uma vez, digite o

seguinte:

```
# yum install xorg-x11-apps
$ xeyes &
$ xeyes &
...
$ killall xeyes &
```

Lembre-se, você precisa ser o usuário `root` para instalar o pacote. Depois disso, lembre-se de repetir o comando `xeyes` 20 vezes. Distribua as janelas pela tela e mova o mouse apenas por diversão observando o movimento olhos. Todas as janelas `xeyes` devem desaparecer de uma vez quando você digitar **`killall xeyes`**.

9. Como um usuário regular, execute o comando `gedit` para que ele inicie com um valor de `nice` de 5.

```
# nice -n 5 gedit &
[1] 21578
```

10. Para utilizar o comando `renice` a fim de alterar o valor `nice` do comando `gedit` que você acabou de iniciar como 7, digite o seguinte:

```
# renice -n 7 21578
21578: old priority 0, new priority 7
```

Utilize qualquer comando que você quiser para verificar se o valor atual `nice` do comando `gedit` já está configurado como 7. Por exemplo, você poderia digitar o seguinte:

```
# ps -eo 'pid,user,nice,comm' | grep gedit
21578 chris 7 gedit
```

Capítulo 7: Escrevendo scripts de shell simples

1. Eis um exemplo de como criar um script em seu diretório `$HOME/bin` chamado `myownscript`. Quando o script é executado, ele deve apresentar informações que se parecem com o seguinte:

```
Today is Sat Dec 10 15:45:04 EST 2011.
You are in /home/joe and your host is abc.example.com.
```

Os passos a seguir mostram uma maneira de criar o script chamado `myownscript`:

- Se ele não existir, crie um diretório `bin`:

```
$ mkdir $HOME/bin
```

- Usando qualquer editor de texto, crie um script chamado `$HOME/bin/myownscript` que contém o seguinte:

```
#!/bin/bash
```

```
# myownscript
# List some information about your current system
echo "Today is $(date)."
```

- Torne o script executável:

```
$ chmod 755 $HOME/bin/myownscript
```

2. Para criar um script que lê três parâmetros posicionais na linha de comando, atribui os parâmetros às variáveis nomeadas ONE, TWO e THREE, respectivamente, e, então, gera essa informação no formato especificado, faça o seguinte:

Substitua X pelo número de parâmetros e Y por todos os parâmetros inseridos. Então, substitua A pelo conteúdo da variável ONE, B pela variável TWO e C pela variável THREE.

- Eis um exemplo do que esse script pode conter:

```
#!/bin/bash
# myposition
ONE=$1
TWO=$2
THREE=$3
echo "There are $# parameters that include: $@"
echo "The first is $ONE, the second is $TWO, the third is $THREE."
```

- Para tornar o script executável, digite o seguinte:

```
$ chmod 755 $HOME/bin/myposition
```

- Para testá-lo, execute-o com alguns argumentos de linha de comando, como a seguir:

```
$ myposition Where Is My Hat Buddy?
```

```
There are 5 parameters that include: Where Is My Hat Buddy?
```

```
The first is Where, the second is Is, the third is My.
```

3. Para criar o script descrito, faça o seguinte: torne o script executável:

```
$ chmod 755 $HOME/bin/myposition
```

- Eis como script mytown pode ficar:

```
#!/bin/bash
# myhome
read -p "What street did you grow up on? " mystreet
read -p "What town did you grow up in? " mytown
echo "The street I grew up on was $mystreet and the town was $mytown."
```

- Execute o script para verificar se funciona. O exemplo a seguir mostra como

poderiam ser a entrada e a saída do script:

```
$ myhome
```

```
What street did you grow up on? Harrison
```

```
What town did you grow up in? Princeton
```

```
The street I grew up on was Harrison and the town was  
Princeton.
```

4. Para criar o script necessário, faça o seguinte: Usando qualquer editor de texto, crie um script chamado `$HOME/bin/myos` e torne o script executável:

```
$ chmod 755 $HOME/bin/myos
```

- O script pode conter o seguinte:

```
#!/bin/bash  
# myos  
read -p "What is your favorite operating system, Mac,  
Windows or  
Linux? "  
opsys  
if [ $opsys = Mac ] ; then  
echo "Mac is nice, but not tough enough for me."  
elif [ $opsys = Windows ] ; then  
echo "I used Windows once. What is that blue screen for?"  
elif [ $opsys = Linux ] ; then  
echo "Great Choice!"  
else  
echo "Is $opsys an operating system?"  
fi
```

5. Para criar um script chamado `$HOME/bin/animals` que lê as palavras moose, cow, goose e sow por meio de um loop e coloca cada uma dessas palavras ao final da linha "I have a...", faça o seguinte:

- Torne o script executável:

```
$ chmod 755 $HOME/bin/animals
```

- O script pode conter o seguinte:

```
#!/bin/bash  
# animals  
for ANIMALS in moose cow goose sow ; do  
echo "I have a $ANIMALS"  
done
```

- Quando você executar o script, a saída deve ser a seguinte:

```
$ animals
I have a moose
I have a cow
I have a goose
I have a sow
```

Capítulo 8: Aprendendo administração de sistema

1. Você pode abrir a janela Date & Time a partir de um desktop GNOME no RHEL ou no Fedora, seguindo uma destas instruções:
 - Abra uma janela Terminal e digite **system-config-date**. Se fizer isso como um usuário comum, você será solicitado a informar a senha de root.
 - Em um desktop GNOME 2. X, selecione System Administration Date & Time.
 - Em um desktop GNOME 3, selecione Activities ⇒ Applications ⇒ System-Config-Date.

Quando a janela Date & Time se abrir, selecione a guia Time Zone para verificar seu fuso horário.

2. Para executar um comando `ps` a fim de ordenar todos os processos em execução no sistema pelo nome de usuário, digite o seguinte:

```
$ ps -ef --sort=user | less
chris  3774  3202  0 21:08 pts/0    00:00:00 less

dbus   1869      1  0 20:42 ?        00:00:00 dbus-daemon
--system

gdm    2616      1  0 20:44 ?        00:00:00 /usr/bin/dbus-
launch
68     2010      1  0 20:43 ?        00:00:00 hald

lp     1971  1970  0 20:43 ?        00:00:00 cups-pollD example
root      1      0  0 20:40 ?        00:00:01 /sbin/init
```

3. Para encontrar todos os arquivos no diretório `/var/spool` que pertencem a outros usuários que não root e criar uma longa lista deles, digite o seguinte (recomendo se tornar root para encontrar arquivos que podem estar ocultos de outros usuários):

```
$ su -
Password: *****
# find /var/spool -not -user root -ls | less
```

4. Para tornar-se usuário root e criar um arquivo vazio ou um arquivo de texto simples chamado `/mnt/test.txt`, digite o seguinte:

```
$ su -
Password: *****
# touch /mnt/test.txt
# ls -l /mnt/test.txt
```

```
-rw-r--r--. 1 root root 0 Jan 9 21:51 /mnt/test.txt
```

5. Para se tornar root e editar o arquivo `/etc/sudoers` a fim de permitir que sua conta de usuário regular (por exemplo, `bill`) tenha privilégios de root completos por meio do comando `sudo`, faça o seguinte:

```
$ su -  
Password: *****  
# visudo  
o bill ALL=(ALL) ALL  
Esc ZZ
```

Como `visudo` abre o arquivo `/etc/sudoers` no `vi`, o exemplo digita `o` para abrir uma linha e depois digita a linha para permitir que `bill` tenha privilégios de root completos. Depois que a linha é digitada, pressione `ESC` para voltar ao modo de comando e digite `ZZ` para gravar e sair.

6. Para usar o comando `sudo` a fim de criar um arquivo chamado `/mnt/test2.txt` e verificar se o arquivo está lá e pertence ao usuário `root`, digite o seguinte:

```
[bill]$ sudo touch /mnt/test2.txt  
We trust you have received the usual lecture from the  
local System  
Administrator. It usually boils down to these three  
things:  
  
#1) Respect the privacy of others.  
#2) Think before you type.  
#3) With great power comes great responsibility.  
  
[sudo] password for bill:  
*****  
[bill]$ ls -l /mnt/text2.txt  
-rw-r--r--. 1 root root 0 Jan 9 23:37 /mnt/text2.txt
```

7. Para observar as mensagens que são gravadas no arquivo `/var/log/messages` quando você conecta um pen drive USB (que será montado automaticamente) e depois desmontar o dispositivo e removê-lo, faça o seguinte:

- Torne-se o usuário `root` e veja as mensagens à medida que elas entram em `/var/log/messages`, fazendo o seguinte:

```
$ su -  
Password: *****  
# tail -f /var/log/messages  
Jan 9 23:44:14 chris kernel: usb 1-1.1: new high speed  
USB device  
  
using ehci_hcd and address 6  
  
Jan 9 23:44:15 chris kernel: usb 1-1.1: New USB device
```



```
found,  
idVendor=090c, idProduct=1000
```

```
...
```

```
Jan 9 23:44:15 chris kernel: Initializing USB Mass  
Storage driver...
```

```
Jan 9 23:44:15 chris kernel: scsi6: SCSI emulation for  
USB Mass Storage devices
```

```
...
```

```
Jan 9 23:44:21 chris kernel: sd 6:0:0:0: [sdb] 8343552  
512-byte logical  
blocks:
```

```
(4.27 GB/3.97 GiB)
```

```
Jan 9 23:44:21 chris kernel: sd 6:0:0:0: [sdb] Write  
Protect is off
```

```
Jan 9 23:44:21 chris kernel: sd 6:0:0:0: [sdb] Assuming  
drive cache:
```

```
write through
```

```
Jan 9 23:44:21 chris kernel: sd 6:0:0:0: [sdb] Assuming  
drive cache:
```

```
write through
```

```
Jan 9 23:44:21 chris kernel: sdb: sdb1
```

- Desmonte o dispositivo clicando no ícone do dispositivo no desktop e selecionando Safely Remove Drive ou, como root, digite `umount /media/?????`, onde `?????` é substituído pelo nome criado quando o dispositivo foi montado.
 - Remova o dispositivo, mas continue a observar o arquivo de mensagens. Então, pressione `Ctrl+C` a fim de parar a anexação de mensagens ao final do arquivo.
8. Para ver quais dispositivos USB estão conectados ao computador, digite o seguinte:

```
$ lsusb
```
 9. Para carregar o módulo `bttv`, listar os módulos que foram carregados e descarregá-lo, digite o seguinte:

```
# modprobe -a bttv  
# lsmod | grep bttv
```

```
bttv                124516    0  
v4l2_common         10572     1bttv  
videobuf_dma_sg     9814      1bttv
```

```

videobuf_core          20076      2bttv,videobuf_dma_sg
btcx_risc              4416      1bttv
rc_core                19686     7ir_lirc_codec,ir_sony_decoder,
ir_jvc_decoder,ir_rc6_decoder
tveeprom              14042     1bttv
videodev               76244     3bttv,v4l2_common,uvccvideo
i2c_algo_bit          5728      2bttv,i915
i2c_core               31274     9bttv,v4l2_common,tveeprom,v:
i2c_i801,i915,drm_kms_helper

```

Observe que outros módulos (`v4l2_common`, `videodev` e outros) foram carregados quando você carregou `bttv` com `modprobe -a`.

10. Digite o seguinte para remover o módulo `bttv`, juntamente com todos os outros módulos que foram carregados com ele. Repare que todos eles desaparecem depois que você executa `modprobe -r`.

```

# modprobe -r bttv
# lsmod | grep bttv

```

Capítulo 9: Instalando o Linux

1. Para instalar um sistema Fedora a partir de um Live CD Fedora, siga as instruções na seção “Instalando o Fedora a partir de um Live CD”. Em geral, os passos incluem:
 - Inicializar o Live CD.
 - Clicar no ícone Install to Hard Drive no desktop.
 - Adicionar informações como solicitado sobre seu teclado, armazenamento, fuso horário, hostname, senha de root e outras informações necessárias para configurar inicialmente seu sistema.
 - Reinicializar o computador, remover o Live CD e, então, inicializar o novo sistema instalado a partir do disco rígido.
2. Para atualizar os pacotes, depois que a instalação do Fedora Live CD está completa, faça o seguinte:
 - Reinicie o computador e responda às perguntas da primeira inicialização quando solicitado.
 - Usando uma conexão com ou sem fio, certifique-se de ter uma conexão com a internet. Consulte o Capítulo 14, “Administrando redes”, caso tenha problemas em fazer sua conexão de rede funcionar corretamente. Abra um shell como o usuário

root e digite **yum update**.

- Quando solicitado, digite **y** para aceitar a lista de pacotes exibidos. O sistema começa a baixar e instalar os pacotes.
3. Para executar a instalação RHEL no modo texto, faça o seguinte:
- Inicialize o DVD do RHEL.
 - Quando vir o carregador de inicialização começar a contagem regressiva, pressione a tecla para interromper o processo de inicialização.
 - Com o título do sistema RHEL que você quer inicializar em destaque, pressione **e**. Mover o cursor para a direita até o fim da linha do kernel e digite a opção literal **text** no final da linha. Pressione Enter e então **b** para inicializar.
 - Teste o resto da instalação no modo texto.
4. Para definir o particionamento de disco, tal como descrito na questão 4, para um DVD de instalação Linux RedHat Enterprise, faça o seguinte:

Nota

Esse procedimento acabará apagando todo o conteúdo do disco rígido. Se quiser apenas usar este exercício para a prática de particionamento, você pode reiniciar seu computador antes de clicar em Next no final desse procedimento, sem apagar seu disco rígido. Mas se quiser prosseguir e particionar seu disco, lembre-se de que todos os dados serão excluídos.

- Em um computador, você pode apagar com pelo menos 11GB de espaço em disco, insira um DVD de instalação do RHEL, reinicie e comece a seguir as telas de instalação.
- Quando chegar à tela que pergunta que tipo de instalação você deseja, selecione Create Custom Layout e clique em Next.
- A partir da tela de particionamento de disco, selecione o dispositivo a ser usado para a instalação (provavelmente `sda` se você tiver um único disco rígido que pode apagar completamente).
- Para criar uma partição de 400MB/boot, selecione a linha Free e selecione Create. No pop-up Create Storage, selecione a partição padrão e clique em Create. No pop-up Add Partition, insira **/boot** como Mount Point, **ext4** como sistema de arquivos e **400** como o tamanho e, então, clique em OK.
- Para criar uma partição LVM de 10GB, selecione a linha Free e clique em Create. No pop-up Create Storage, selecione LVM Physical Volume e clique em Create. No pop-up Add Partition, digite **10000** como o tamanho, selecione Fixed size e clique em OK.
- Para criar um grupo de volumes chamado `tracker` a partir volume físico LVM que você acabou de criar, clique em Create. No pop-up Create Storage de Volume,

selecione LVM Volume Group e clique em Create. No pop-up Make LVM Volume Group, digite **tracker** como o Volume Group Name. Verifique se há uma marca de seleção ao lado do dispositivo LVM que você acabou de criar na caixa Volumes to Use e clique em OK.

- Para criar uma partição `/` de 3GB a partir do grupo de volumes `tracker`, selecione o grupo de volumes `tracker` que você acabou de criar sob LVM Volume Groups e selecione Edit. No pop-up Edit LVM Volume Group, selecione Add. No pop-up Make Logical Volume digite `/` para Mount Point, digite **3000** para Size e selecione OK. Selecione OK para voltar à tela principal.
- Para criar uma partição `/var` de 2 GB a partir do grupo de volumes `tracker`, selecione o grupo de volumes `tracker` que você acabou de criar sob LVM Volume Groups e selecione Edit. No pop-up Edit LVM Volume Group selecione Add. No pop-up Make Logical Volume digite `/var` para Mount Point, digite **2000** para Size e selecione OK. Selecione OK para voltar à tela principal.
- Para criar uma partição `/home` de 3GB a partir do grupo de volumes `tracker`, selecione o grupo de volumes `tracker` que você acabou de criar sob LVM Volume Groups e selecione Edit. No pop-up Edit LVM Volume Group selecione Add. No pop-up Make Logical Volume digite `/home` para Mount Point, digite **3000** para Size e selecione OK. Selecione OK para voltar à tela principal.

Nesse ponto, você deve ver `tracker` sob LVM Volume Groups e três partições (`/`, `/var` e `/home`) listadas sob o grupo `tracker`. Sob Hard Drives, você deve ver o nome do seu disco (provavelmente `sda`) com a partição `/boot` e o volume físico `tracker` listados sob isso. Agora você tem a opção de continuar com a instalação clicando em Next ou, se não quiser concluir a instalação, pode simplesmente reiniciar o computador e remover o disco sem ter feito nenhum dano ao seu sistema instalado atualmente.

Capítulo 10: Obtendo e gerenciando software

1. Para pesquisar o pacote que fornece o comando `mogrify` no repositório YUM, digite o seguinte:

```
# yum provides mogrify
```

2. Para exibir informações sobre o pacote que fornece o comando `mogrify` e determinar qual é página inicial (URL) desse pacote, digite o seguinte:

```
# yum info ImageMagick
```

Você verá que o URL da página inicial do ImageMagick é <http://www.imagemagick.org>.

3. Para instalar o pacote contendo o comando `mogrify`, digite o seguinte:

```
# yum install ImageMagick
```

4. Para listar todos os arquivos de documentação contidos no pacote que fornece o comando `mogrify`, digite o seguinte:

```
# rpm -qd ImageMagick
...
/usr/share/doc/ImageMagick-6.7.0.10/README.txt
...
/usr/share/man/man1/identify.1.gz
/usr/share/man/man1/import.1.gz
/usr/share/man/man1/mogrify.1.gz
```

5. Para examinar o changelog do pacote que fornece o comando `mogrify`, digite o seguinte:

```
# rpm -q --changelog ImageMagick | less
```

6. Para apagar o comando `mogrify` de seu sistema e verificar seu pacote no banco de dados RPM a fim de ver que o comando está realmente faltando, digite o seguinte:

```
# type mogrify
mogrify is /usr/bin/mogrify
# rm /usr/bin/mogrify
rm remove regular file '/usr/bin/mogrify'?
# rpm -V ImageMagick
missing /usr/bin/mogrify
```

7. Para reinstalar o pacote que fornece o comando `mogrify` e certificar-se de que todo o pacote está intacto novamente, digite o seguinte:

```
# yum reinstall ImageMagick
# rpm -V ImageMagick
```

8. Para baixar o pacote que fornece o comando `mogrify` para o diretório atual, digite o seguinte:

```
# yumdownloader ImageMagick
ImageMagick-6.7.0.10-4.fc16.i686.rpm
```

9. Para exibir informações gerais sobre o pacote que você acabou de baixar, consultando arquivo do pacote RPM no diretório atual, digite o seguinte:

```
# rpm -qip ImageMagick-6.7.0.10-4.fc16.i686.rpm
```

```
Name           : ImageMagick
Version        : 6.7.0.10
Release        : 4.fc16
Architecture   : i686
...
```

10. Para remover o pacote contendo o comando `mogrify` de seu sistema, digite o

seguinte:

```
# yum remove ImageMagick
```

Capítulo 11: Gerenciando contas de usuário

Para as questões que envolvem a adição e remoção de contas de usuário, você pode usar a janela User Manager ou ferramentas de linha de comando como `useradd` e `usermod`. A ideia aqui é garantir que você obtenha os resultados corretos mostrados nas respostas que se seguem, não necessariamente fazê-las exatamente da mesma maneira como eu fiz. Há várias maneiras como você pode conseguir os mesmos resultados. As respostas aqui mostram como completar os exercícios a partir da linha de comando. (Torne-se o usuário `root` quando você vir um prompt `#`.)

1. Para adicionar uma conta de usuário local a seu sistema Linux que tem um nome de usuário `jbaxter` e um nome completo John Baxter, que usa `/bin/sh` como seu shell padrão e que é o próximo UID disponível (o seu pode ser diferente do mostrado aqui), digite o seguinte. Você pode usar o comando `grep` para verificar a nova conta de usuário. Então, defina a senha para `jbaxter` como: **My1N1te0ut!**

```
# useradd -c "John Baxter" -s /bin/sh jbaxter
# grep jbaxter /etc/passwd
jbaxter:x:1001:1001:John Baxter:/home/jbaxter:/bin/sh
# passwd jbaxter
Changing password for user jbaxter
New password: My1N1te0ut!
Retype new password: My1N1te0ut!
passwd: all authentication tokens updated successfully
```

2. Para criar uma conta de grupo com o nome `testing` que usa o ID de grupo 315, digite o seguinte:

```
# groupadd -g 315 testing
# grep testing /etc/group
testing:x:315:
```

3. Para adicionar `jbaxter` ao grupo `testing` e ao grupo `bin`, digite o seguinte:

```
# usermod -aG testing,bin jbaxter
# grep jbaxter /etc/group
bin:x:1:bin,daemon,jbaxter
jbaxter:x:1001:
testing:x:315:jbaxter
```

4. Para se tornar `jbaxter` e, temporariamente, fazer o grupo `testing` ser grupo padrão de `jbaxter`, execute `touch /home/jbaxter/file.txt` — assim o grupo `testing` será definido como grupo do arquivo — e faça o seguinte:

```
$ su - jbaxter
```

```
Password: My1N1te0ut!  
-sh-4.2$ newgrp testing  
sh-4.2$ touch /home/jbaxter/file.txt  
sh-4.2$ ls -l /home/baxter/file.txt  
-rw-rw-r--. 1 jbaxter testing 0 Jan 25 06:42  
/home/jbaxter/file.txt sh-4.2$ exit ; exit
```

5. Observe que ID de usuário foi atribuído a jbaxter e então exclua a conta do usuário sem excluir o diretório inicial atribuído a jbaxter.

```
$ userdel jbaxter
```

6. Use o seguinte comando para localizar arquivos no diretório /home (e seus subdiretórios) que são atribuídos ao ID de usuário que recentemente pertencia ao usuário chamado jbaxter (quando fiz isso, ambos, UID e GID, eram 1001; o seu pode diferir). Observe que o nome de usuário jbaxter já não está atribuído no sistema, de modo que todos os arquivos que o usuário criou são listados como pertencentes ao UID 1001 e ao GID 1001, exceto por dois arquivos que foram atribuídos ao grupo testing, por causa do comando newgrp executado anteriormente:

```
# find /home -uid 1001 -ls  
262184 4 drwx----- 4 1001 1001 4096 Jan 25 08:00  
/home/jbaxter  
262193 4 -rw-r--r-- 1 1001 1001 176 Jan 27 2011  
/home/jbaxter/.bash_profile  
262196 4 -rw----- 1 13602 testing 93 Jan 25 08:00  
/home/jbaxter/.bash_history  
262194 0 -rw-rw-r-- 1 13602 testing 0 Jan 25 07:59  
/home/jbaxter/file.txt  
...
```

7. Execute esses comandos para copiar o arquivo /etc/services para o diretório /etc/skel/; então adicione ao sistema um novo usuário chamado mjones, com o nome completo Mary Jones e um diretório inicial /home/maryjones. Liste seu diretório inicial para garantir que o arquivo services está lá.

```
# cp /etc/services /etc/skel/  
# useradd -d /home/maryjones -c "Mary Jones" mjones  
# ls -l /home/maryjones  
total 628  
-rw-r--r--. 1 mjones mjones 640999 Jan 25 06:27 services
```

8. Execute o seguinte comando para localizar todos os arquivos no diretório /home que pertencem a mjones. Se você fez os exercícios na ordem, note que excluiu o usuário com o ID de usuário e o ID de grupo mais altos, esses números foram designados a mjones. Como resultado, todos os arquivos existentes no sistema que pertenciam a jbaxter agora pertencem a mjones. (Por essa razão, você deve remover ou alterar a posse de arquivos deixados para trás ao excluir um usuário.)

```

# find /home -user mjones -ls
262184 4 drwx----- 4 mjones mjones 4096 Jan 25 08:00
/home/jbaxter
262193 4 -rw-r--r-- 1 mjones mjones 176 Jan 27 2011
/home/jbaxter/.bash_profile
262189 4 -rw-r--r-- 1 mjones mjones 18 Jan 27 2011
/home/jbaxter/.bash_logout
262194 0 -rw-rw-r-- 1 mjones testing 0 Jan 25 07:59
/home/jbaxter/file.txt
262188 4 -rw-r--r-- 1 mjones mjones 124 Jan 27 2011
/home/jbaxter/.bashrc
262197 4 drwx----- 4 mjones mjones 4096 Jan 25 08:27
/home/maryjones
262207 4 -rw-r--r-- 1 mjones mjones 176 Jan 27 2011
/home/maryjones/.bash_profile
262202 4 -rw-r--r-- 1 mjones mjones 18 Jan 27 2011
/home/maryjones/.bash_logout
262206 628 -rw-r--r-- 1 mjones mjones 640999 Jan 25 08:27
/home/maryjones/services
262201 4 -rw-r--r-- 1 mjones mjones 124 Jan 27 2011
/home/maryjones/.bashrc

```

9. Como o usuário mjones, você pode usar o seguinte para criar um arquivo chamado /tmp/maryfile.txt e usar ACLs para atribuir ao usuário bin e ao grupo lp permissão de leitura/gravação sobre esse arquivo.

```

[mjones]$ touch /tmp/maryfile.txt
[mjones]$ setfacl -m u:bin:rw /tmp/maryfile.txt
[mjones]$ setfacl -m g:lp:rw /tmp/maryfile.txt
[mjones]$ getfacl /tmp/maryfile.txt
# file: tmp/maryfile.txt
# owner: mjones
# group: mjones
user::rw-
user:bin:rw-
group::rw-
group:lp:rw-
mask::rw-
other::r -

```

10. Execute esse conjunto de comandos (como mjones) para criar um diretório chamado /tmp/mydir e usar ACLs para atribuir permissões padrão sobre ele de tal modo que o usuário adm tenha permissão de leitura/gravação/execução sobre esse diretório e todos os arquivos ou diretórios criados dentro dele. Teste se isso funcionou criando o diretório /tmp/mydir/testing/ e /tmp/mydir/newfile.txt.

```

[mary]$ mkdir /tmp/mydir

```



```

[mary]$ setfacl -m d:u:adm:rwx /tmp/mydir
[mjones]$ getfacl /tmp/mydir
# file: tmp/mydir
# owner: mjones
# group: mjones
user::rwx
group::rwx
other::r-x
default:user::rwx
default:user:adm:rwx
default:group::rwx
default:mask::rwx
default:other::r-x
[mjones]$ mkdir /tmp/mydir/testing
[mjones]$ touch /tmp/mydir/newfile.txt
[mjones]$ getfacl /tmp/mydir/testing/
# file: tmp/mydir/testing/
# owner: mjones
# group: mjones
user::rwx
user:adm:rwx
group::rwx
mask::rwx
other::r-x
default:user::rwx
default:user:adm:rwx
default:group::rwx
default:mask::rwx
default:other::r-x
[mjones]$ getfacl /tmp/mydir/newfile.txt
# file: tmp/mydir/newfile.txt
# owner: mjones
# group: mjones

user::rw-
user:adm:rwx                #effective:rw-
group::rwx                  #effective:rw-
mask::rw-
other::r--

```

Note que o usuário `adm` efetivamente tem apenas permissão `rw-`. Para remediar isso, você precisa expandir as permissões da máscara. Uma maneira de fazer isso é com o comando `chmod`, como segue:

```
[mjones]$ chmod 775 /tmp/mydir/newfile.txt
[mjones]$ getfacl /tmp/mydir/newfile.txt
# file: tmp/mydir/newfile.txt
# owner: mjones
# group: mjones
user::rwx
user:adm:rwx
group::rwx
mask::rwx
other::r-x
```

Capítulo 12: Gerenciando discos e sistemas de arquivos

1. Para determinar o nome do dispositivo de um pen drive que você quer inserir em seu computador, digite o seguinte e insira o pen drive.

```
# tail -f /var/log/messages
kernel: [sdb] 15667200 512-byte logical blocks:
(8.02 GB/7.47 GiB)
Feb 11 21:55:59 cnegus kernel: sd 7:0:0:0:
[sdb] Write Protect is off
Feb 11 21:55:59 cnegus kernel: [sdb] Assuming
drive cache: write through
Feb 11 21:55:59 cnegus kernel: [sdb] Assuming
drive cache: write through
```

2. Para listar as partições no pen drive, digite o seguinte:

```
# fdisk -c -u -l /dev/sdb
```

3. Para apagar partições no pen drive, assumindo o dispositivo `/dev/sdb`, faça o seguinte:

```
# fdisk -cu /dev/sdb
Command (m for help): d
Partition number (1-6): 6
Command (m for help): d
Partition number (1-5): 5
Command (m for help): d
Partition number (1-5): 4
Command (m for help): d
```

```
Partition number (1-4): 3
Command (m for help): d
Partition number (1-4): 2
Command (m for help): d
Selected partition 1
Command (m for help): w
# partprobe /dev/sdb
```

4. Para adicionar uma partição Linux de 100MB, partição de troca de 200MB e partição LVM de 500MB ao pen drive, digite o seguinte:

```
# fdisk -cu /dev/sdb
```

```
Command (m for help): n
Command action
```

```
e extended
```

```
p primary partition (1-4)
```

```
p
```

```
Partition number (1-4): 1
```

```
First sector (2048-15667199, default 2048): <ENTER>
```

```
Last sector, +sectors or +size{K,M,G} (default 15667199):  
+100M
```

```
Command (m for help): n
Command action
```

```
e extended
```

```
p primary partition (1-4)
```

```
p
```

```
Partition number (1-4): 2
```

```
First sector (616448-8342527, default 616448): <ENTER>
```

```
Last sector, +sectors or +size{K,M,G} (default 15667199):  
+200M
```

```
Command (m for help): n
Command action
```

```
e extended
```

```
p primary partition (1-4)
```

```
p
```

```
Partition number (1-4): 3
```

```
First sector (616448-15667199, default 616448): <ENTER>
```

```
Using default value 616448
```

```
Last sector, +sectors or +size{K,M,G} (default 15667199):  
+500M
```

```
Command (m for help): t
```

```
Partition number (1-4): 2
```

```
Hex code (type L to list codes): 82
Changed system type of partition 2 to 82 (Linux swap /
Solaris)
Command (m for help): t
Partition number (1-4): 3
Hex code (type L to list codes): 8e
Changed system type of partition 3 to 8e (Linux LVM)
Command (m for help): w
# partprobe /dev/sdb
# grep sdb /proc/partitions

8    16          7833600 sdb
8    17          102400 sdb1
8    18          204800 sdb2
8    19          512000 sdb3
```

5. Para colocar um sistema de arquivos `ext3` na partição Linux, digite o seguinte:

```
# mkfs -t ext3 /dev/sdb1
```

6. Para criar um ponto de montagem chamado `/mnt/mypart` e montar a partição Linux nele temporariamente, faça o seguinte:

```
# mkdir /mnt/mypart
# mount -t ext3 /dev/sdb1 /mnt/mypart
```

7. Para criar a partição de troca e ativá-la de modo que não haja espaço de troca adicional imediatamente disponível, digite o seguinte:

```
# mkswap /dev/sdb2
# swapon /dev/sdb2
```

8. Para criar um grupo de volumes chamado `abc` a partir da partição LVM, criar um volume lógico de 200MB a partir desse grupo chamado `data`, criar um sistema de arquivos VFAT nele, montar temporariamente o volume lógico em um novo diretório chamado `/mnt/test` e, então, verificar se ele foi montado com sucesso, digite o seguinte:

```
# pvcreate /dev/sdb3
# vgcreate abc /dev/sdb3
# lvcreate -n data -L 200M abc
# mkfs -t vfat /dev/mapper/abc-data
# mkdir /mnt/test
# mount /dev/mapper/abc-data /mnt/test
```

9. Para ampliar o volume lógico de 200MB para 300MB, digite o seguinte:

```
# lvextend -L +100M /dev/mapper/abc-data
# resize2fs -p /dev/mapper/abc-data
```

10. Para remover com segurança o pen drive do computador, faça o seguinte:

```
# umount /dev/sdb1
# swapoff /dev/sdb2
# umount /mnt/test
# lvremove /dev/mapper/abc-data
# vgremove abc
# pvremove /dev/sdb3
```

Agora você pode remover com segurança o dispositivo USB do computador.

Capítulo 13: Entendendo administração de servidores

1. Para efetuar login em qualquer conta em outro computador usando o comando `ssh`, digite o seguinte e, então, digite a senha quando solicitado:

```
$ ssh joe@localhost
joe@localhost's password:
*****
[joe]$
```

2. Para exibir o conteúdo de um arquivo remoto `/etc/system-release` e ter seus conteúdos apresentados no sistema local usando execução remota com o comando `ssh`, faça o seguinte:

```
$ ssh joe@localhost "cat /etc/system-release"
Fedora release 16 (Verne)
```

3. Para usar o encaminhamento X11 para exibir uma janela `gedit` em seu sistema local e, então, salvar um arquivo no diretório inicial remoto, faça o seguinte:

```
$ ssh -X joe@localhost "gedit newfile"
joe@localhost's password:
$ ssh joe@localhost "cat newfile"
joe@localhost's password:
This is text from the file I saved in joe's remote home
directory
```

4. Para copiar recursivamente todos os arquivos do diretório `/usr/share/selinux` em um sistema remoto para o diretório `/tmp` em seu sistema local de tal maneira que todos os horários de modificação nos arquivos sejam atualizados com a hora do sistema local quando eles forem copiados, faça o seguinte:

```
$ scp -r joe@localhost:/usr/share/selinux /tmp
joe@localhost's password:
irc.pp.bz2          100% 9673      9.5KB/s   00:00
dcc.pp.bz2          100% 15KB     15.2KB/s  00:01
$ ls -l /tmp/selinux | head
total 20
drwxr-xr-x. 3 root root 4096 Apr 18 05:52 devel
drwxr-xr-x. 2 root root 4096 Apr 18 05:52 packages
drwxr-xr-x. 2 root root 12288 Apr 18 05:52 targeted
```

5. Para copiar recursivamente todos os arquivos do diretório /usr/share/logwatch em um sistema remoto para o diretório /tmp em seu sistema local de tal maneira que todos os horários de modificação nos arquivos do sistema remoto sejam mantidos no sistema local, tente isto:

```
$ rsync -av joe@localhost:/usr/share/logwatch /tmp
joe@localhost's password:
*****
receiving incremental file list
logwatch/
logwatch/default.conf/
logwatch/default.conf/logwatch.conf
$ ls -l /tmp/logwatch | head
total 16
drwxr-xr-x. 5 root root 4096 Apr 19 2011 default.conf
drwxr-xr-x. 4 root root 4096 Feb 28 2011 dist.conf
drwxr-xr-x. 2 root root 4096 Apr 19 2011 lib
```

6. Para criar um par de chaves pública/privada a fim de usar comunicações SSH (sem senha na chave), copiar o arquivo de chave pública para a conta de um usuário remoto com ssh-copy-id, e usar autenticação baseada em chave para efetuar login nessa conta de usuário sem ter de digitar uma senha, utilize o seguinte código:

```
$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key
(/home/joe/.ssh/id_rsa): <ENTER>
/home/joe/.ssh/id_rsa already exists.
Enter passphrase (empty for no passphrase): <ENTER>
Enter same passphrase again: <ENTER>
Your identification has been saved in
/home/joe/.ssh/id_rsa.
Your public key has been saved in
/home/joe/.ssh/id_rsa.pub.
The key fingerprint is:
58:ab:c1:95:b6:10:7a:aa:7c:c5:ab:bd:f3:4f:89:1e
joe@cnegus.csb
The key's randomart image is:
$ ssh-copy-id -i ~/.ssh/id_rsa.pub joe@localhost
joe@localhost's password: *****
Now try logging into the machine, with "ssh
'joe@localhost'",
and check in:
.ssh/authorized_keys
to make sure we haven't added extra keys that you weren't
expecting.
```

```

$ ssh joe@localhost
$ cat .ssh/authorized_keys
ssh-rsa
AAAAB3NzaC1yc2EAAAABIWAAAQEAYN2Psp5/LRUC9E8BDCx53yPUa0qoO.
v6H4sF3vmn04V6E7D1iXpzwPzdo4rpvmR1ZiinHR2xGAEr2uZag7feKgL:
iR7lZrOhQjV+SGb/a1dxrIeZqKMq1Tk07G4EvboIrq//9J47vI417iNu0:
DdCTbpG6J3uSJm1BKzdUtwb413x35W2bRgMI75aIdeBsDgQBBiOdu+zuTI
XeJ7gIwRvBaMQdOSvSdlkX353tmIjmJheWdgCccM/1jKdoELpaevg9anC
tTo4I+qTfzAQD5+66oqW0LgMkWVvfZI7dUz3WUPmcMw==
chris@abc.example.com

```

- Para criar uma entrada em `/etc/rsyslog.conf` que armazena todas as mensagens de autenticação no nível de informação e superior em um arquivo chamado `/var/log/myauth`, faça o seguinte. A partir de um terminal, observe os dados chegando.

```

# vim /etc/rsyslog.conf
authpriv.info                                /var/log/myauth
# service rsyslog restart
or
# systemctl restart rsyslog.service
<Terminal 1>                                <Terminal 2>
# tail -f /var/log/myauth                    $ ssh joe@localhost
Apr 18 06:19:34 abc unix_chkpwd[30631]       joe@localhost's
password:
Apr 18 06:19:34 abc sshd[30631]              Permission denied, try
again
: pam_unix(sshd:auth):
authentication failure; logname= uid=501

```

```

euid=501 tty=ssh ruser= rhost=localhost
user=joe

```

```

Apr 18 06:19:34 abc sshd[30631]:

```

```

Failed password for joe from
127.0.0.1 port 5564 ssh2

```

- Para determinar as maiores estruturas de diretórios sob `/usr/share`, ordená-las da maior para a menor e listar os 10 maiores desses diretórios em termos de tamanho usando o comando `du`, digite o seguinte:

```

$ du -s /usr/share/* | sort -rn | head

```

```

458320          /usr/share/locale
129400          /usr/share/doc
124116          /usr/share/icons
80524           /usr/share/gnome
...

```

- Para mostrar o espaço usado e o espaço disponível de todos os sistemas de arquivos atualmente conectados ao sistema local, mas excluir quaisquer sistemas de arquivos `tmpfs` ou `devtmpfs` usando o comando `df`, digite o seguinte:

```
$ df -h -x tmpfs -x devtmpfs
```

Filesystem	Size	Used	Avail	Use%	Mounted on
/deev/sda4	20G	4.2G		16G 22%	/

10. Para encontrar todos os arquivos no diretório `/usr` com mais de 10MB de tamanho, faça o seguinte:

```
$ find /usr -size +10M
```

```
/usr/lib/jvm/java-1.6.0-openjdk-1.6.0.0/jre/lib/rt.jar  
/usr/lib/jvm/java-1.7.0-openjdk-1.7.0.3/jre/lib/rt.jar  
/usr/lib/llvm/libLLVM-2.9.so  
/usr/lib/flash-plugin/libflashplayer.so
```

Capítulo 14: Administrando redes

1. Para usar o desktop a fim de verificar se o NetworkManager iniciou com sucesso sua placa de rede (com ou sem fio), faça o seguinte:

Clique no ícone do NetworkManager em seu painel superior. Qualquer conexão de rede ativa com ou sem fio deve ser destacada em negrito.

Se não houver uma conexão ativa, selecione uma rede na lista de redes com ou sem fio disponíveis e digite o nome de usuário e a senha, se solicitado, para iniciar uma conexão ativa.

2. Para executar um comando a fim de verificar as placas de rede ativas disponíveis em seu computador, digite:

```
$ ifconfig
```

ou

```
$ ip addr show
```

3. Tente entrar em contato com `google.com` a partir da linha de comando de uma maneira que assegure que o DNS está funcionando corretamente:

```
$ ping google.com
```

Ctrl-C

4. Para executar um comando a fim de verificar as rotas sendo usadas para se comunicar com o mundo externo à sua rede local, digite:

```
$ route
```

5. Para traçar a rota sendo tomada para se conectar a `google.com`, use o comando `traceroute`:

```
$ traceroute google.com
```


6. Para desligar e desativar o NetworkManager e iniciar o serviço de rede, faça o seguinte:

A partir de um sistema RHEL 6, digite:

```
# service NetworkManager stop
# service network restart
# chkconfig NetworkManager off
# chkconfig network on
```

Para sistemas Fedora mais recentes, digite:

```
# systemctl stop NetworkManager.service
# systemctl disable NetworkManager.service
# service network restart
# chkconfig network on
```

7. Para criar uma entrada de host que permita que você se comunique com o sistema host local usando o nome myownhost, faça o seguinte:

Edite o arquivo `/etc/hosts` (`vi /etc/hosts`) e adicione myownhost ao final da entrada localhost de modo que ela apareça como a seguir (depois pingue myownhost para ver se funcionou):

```
127.0.0.1      localhost.localdomain localhost myownhost
# ping myownhost
Ctrl+C
```

8. Para adicionar o servidor DNS público do Google (endereço IP 8.8.8.8) como o último de sua lista de servidores DNS, siga estes procedimentos:

Faça uma cópia de seu arquivo `resolv.conf` antes de prosseguir (e, então, copie-o de volta depois de concluir o procedimento):

```
# cp /etc/resolv.conf $HOME
```

Se você estiver usando o serviço NetworkManager, à esquerda, clique no ícone do NetworkManager e selecione Network Settings. Selecione IPv4 Settings. Então, marque a caixa Method e escolha Automatic (DHCP) addresses only e preencha 8.8.8.8 na caixa DNS servers (juntamente com quaisquer outros servidores DNS de que você precisar). Se isso não funcionar, tente um dos servidores DNS listados no arquivo `resolv.conf` que você acabou de copiar para seu diretório inicial.

Ou, se você estiver usando o serviço de rede, edite o arquivo `/etc/resolv.conf` diretamente de modo que o arquivo inclua pelo menos a seguinte linha:

```
nameserver 8.8.8.8
```

Em ambos os casos, use o comando `dig` para verificar se o servidor DNS foi capaz de converter um endereço:

```
# dig google.com
...
google.com.      91941      IN         NS         ns3.google.com.
;; Query time: 0 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Mon Apr 30 13:57:44 2012
;; MSG SIZE rcvd: 276
```

9. Para criar uma rota personalizada que direciona o tráfego destinado à rede 192.168.99.0/255.255.255.0 para algum endereço IP em sua rede local, como 192.168.0.5 (primeiro garantindo que a rede 10.0.99 não está sendo usada em seu local), faça o seguinte:

Determine o nome da placa de rede. Para o RHEL, sua primeira placa de rede é provavelmente `eth0`. Nesse caso, como `root` execute os seguintes comandos:

```
# cd /etc/sysconfig/network-scripts
# vi route-eth0
```

Adicione as seguintes linhas ao arquivo:

```
ADDRESS0=192.168.99.0
NETMASK0=255.255.255.0
GATEWAY0=192.168.0.5
```

Reinicie a rede e execute `route` para ver se a rota está ativa:

```
# service network restart
```

```
# route
```

Destination	Gateway	Genmask	Flags	Metric	Ref	Use
default	192.168.0.1	0.0.0.0	UG	0	0	0
192.168.0.0	*	255.255.255.0	U	1	0	0
192.168.99.0	192.168.0.5	255.255.255.0	UG	0	0	0

10. Para verificar se seu sistema foi configurado para permitir que pacotes IPv4 sejam roteados entre as placas de rede em seu sistema, digite o seguinte:

```
# cat /proc/sys/net/ipv4/ip_forward
0
```

0 significa que o encaminhamento de pacotes IPv4 está desativado; um 1 quer dizer que está habilitado.

Capítulo 15: Iniciando e parando serviços

1. Para determinar qual daemon `init` seu servidor está usando atualmente, considere o seguinte:

- Você tem o daemon `Upstart` `init` se:

Seu servidor Linux executa uma das seguintes distribuições: RHEL versão 6, Fedora versões 9 a 14, Ubuntu versões 6-10 ou superior, ou openSUSE versões 11.3 ou superior. E o comando `strings` mostra `Upstart` `init` em uso, como demonstrado no exemplo abaixo:

```
$ strings /sbin/init | grep -i upstart
upstart-devel@lists.ubuntu.com
UPSTART_CONFDIR
UPSTART_NO_SESSIONS
...
```

- Você tem o daemon `systemd` se:

Seu servidor Linux roda o Fedora, versão 15 ou superior. E o comando `strings` mostra `systemd` em uso, como demonstrado no exemplo abaixo:

```
# strings /sbin/init | grep -i systemd
systemd.unit=
systemd.log_target=
systemd.log_level=
...
```

- Muito provavelmente, você tem o daemon `SysVinit` ou `BSD init` se seu daemon `init` não é o daemon `Upstart` `init` ou `systemd`. Mas confirme em <http://wikipedia.org/wiki/Init>.
2. Para determinar o daemon `init` que `sshd` está usando em seu servidor Linux, esteja ciente de que o daemon `init` que `sshd` utiliza não depende somente do daemon `init` que servidor está usando atualmente. Vários serviços podem ainda não ter sido portados para novos daemons `init`. Portanto, experimente tanto o novo daemon `init` como os comandos `SysVinit` clássicos.

- Para o daemon `Upstart` `init`, um resultado positivo, mostrado aqui, significa que o `sshd` foi convertido para `Upstart`:

```
# initctl status ssh
ssh start/running, process 2390
```

- Para `systemd`, um resultado positivo, mostrado aqui, significa que o `sshd` foi convertido para `systemd`:

```
# systemctl status sshd.service
sshd.service - OpenSSH server daemon
Loaded: loaded (/lib/systemd/system/sshd.service;
enabled)
Active: active (running) since Mon, 30 Apr 2015
12:35:20...
```

- Se você não vir resultados positivos para os testes anteriores, experimente o seguinte comando para o daemon `SysVinit` `init`. Um resultado positivo aqui, junto com resultados negativos para os testes anteriores, significa que `sshd` ainda está usando o daemon `SysVinit`.

```
# service ssh status
sshd (pid 2390) is running...
```

3. Para determinar o `runlevel` anterior e atual de seu servidor, use o comando `runlevel`. Ele ainda funciona em todos os daemons `init`:

```
$ runlevel
N 3
```

4. Para alterar o nível de execução padrão ou unidade de destino no servidor Linux, você pode fazer uma das seguintes opções (dependendo do daemon `init` de seu servidor):

- Para o daemon `SysVinit`, edite o arquivo `/etc/inittab` e altere o `#` na linha `id:::initdefault:` para 2, 3, 4 ou 5.

- Para o daemon `Upstart init`, edite o arquivo `/etc/inittab` e altere o `#` na linha `id:::initdefault:` para 2, 3, 4 ou 5.

- Para `systemd`, altere o link simbólico `default.target` para o `runlevel#.target` desejado, onde `#` é 2, 3, 4, ou 5. O código a seguir mostra como mudar o vínculo simbólico de `target unit` para `runlevel3.target`.

```
# ln -sf /lib/systemd/system/runlevel3.target
/etc/systemd/system/default.target
/lib/systemd/system/runlevel3.target
```

5. Para listar os serviços em execução (ou ativos) em seu servidor, você vai precisar usar comandos diferentes, dependendo do daemon `init` que você está usando.

- Para o daemon `SysVinit`, use o comando `service`, como mostrado no exemplo a seguir:

```
# service --status-all | grep running... | sort
anacron (pid 2162) is running...
atd (pid 2172) is running...
...
```

- Para o daemon `Upstart init`, use o comando `initctl`. Mas também não se esqueça de usar o comando `service`, porque nem todos os serviços podem ter sido portados para `Upstart`:

```
# initctl list | grep start/running
tty (/dev/tty3) start/running, process 1163
...
# service --status-all | grep running
abrttd (pid 1118) is running...
...
```

Para `systemd`, use o comando `systemctl`, como aqui:

```
# systemctl list-unit-files --type=service | grep -v disa
UNIT FILE STATE
```

```
abrt-ccpp.service          enabled
abrt-oops.service         enabled
...
```

6. Para listar os serviços atuais (ou ativos) em seu servidor Linux, use o(s) comando(s) apropriado(s) determinado(s) na Resposta 5 para o daemon `init` que seu servidor está usando.
7. Para cada daemon `init`, os comandos a seguir mostrarão o status atual de um serviço específico:

Para o daemon `SysVinit`, o comando `service status nome_do_serviço` é usado.

Para o daemon `Upstart init`, o comando `initctl status nome_do_serviço` é usado.

Para `systemd`, o comando `systemctl status nome_do_serviço` é usado.

8. Para mostrar o status do daemon `cups` em seu servidor Linux, use o seguinte:

- Para o daemon `SysVinit`:

```
# service cups status
  cupsd (pid 8236) is running...
```

- Para o daemon `Upstart init`:

```
# initctl status cups
  cups start/running, process 2390
```

- Lembre-se de que se um serviço ainda não foi portado para o `Upstart`, você vai precisar usar o comando `service` em vez de `initctl`.

- Para `systemd`:

```
# systemctl status cups.service
  cups.service - CUPS Printing Service
  Loaded: loaded (/lib/systemd/system/cups.service;
  enabled)
  Active: active (running) since Tue, 01 May 2015
  04:43:5...
```

```
Main PID: 17003 (cupsd)
  CGroup: name=systemd:/system/cups.service
  17003 /usr/sbin/cupsd -f
```

9. Para tentar reiniciar o daemon `cups` em seu servidor Linux, use o seguinte:

Para o daemon `SysVinit`:

```
# service cups restart
  Stopping cups:          [ OK ]
  Starting cups:         [ OK ]
```

- Para o daemon `Upstart` `init`:

```
# initctl restart cups  
cups start/running, process 2490
```

Lembre-se de que se um serviço ainda não foi portado para o `Upstart`, você vai precisar usar o comando `service` em vez de `initctl`.

- Para `systemd`:

```
# systemctl restart cups.service
```

10. Para tentar recarregar o daemon `cups` em seu servidor Linux, use o seguinte: para o daemon `SysVinit`:

```
# service cups reload
```

```
Reloading cups: [ OK ]
```

Para o daemon `Upstart` `init`:

```
# initctl reload cups
```

Lembre-se de que se um serviço ainda não foi portado para o `Upstart`, você vai precisar usar o comando `service` em vez de `initctl`.

- Para `systemd`, essa é uma pergunta capciosa. Você não pode recarregar o daemon `cups` em um servidor Linux `systemd`!

```
# systemctl reload cups.service
```

```
Failed to issue method call: Job type reload is not  
applicable for unit cups.service.
```

Capítulo 16: Configurando um servidor de impressão

Para as questões que envolvem trabalho com impressoras, você pode usar qualquer ferramenta gráfica ou de linha de comando na maioria dos casos. A intenção é garantir que você obtenha os resultados corretos, mostrados nas respostas que se seguem. As respostas aqui incluem uma mistura de maneiras gráficas e de linha de comando para resolver os exercícios. (Torne-se o usuário `root` quando você vir um prompt `#`.)

1. Para usar a janela de configuração da impressora para adicionar uma nova impressora chamada `myprinter` a seu sistema (impressora PostScript genérica, conectada a uma porta), faça o seguinte, no Fedora 16:
 - a. No desktop GNOME 3, selecione `Applications` ⇒ `Other` ⇒ `Printing`.
 - b. Selecione o botão `Add` (digite a senha de `root`, se for solicitado).
 - c. Selecione uma porta `Serial`, `LPT`, ou outra porta como o dispositivo e clique em

Forward.

- d. Para o driver, escolha Generic e clique em Forward; então, escolha PostScript Printer e clique em Forward.
 - e. Clique em Forward para pular todas as opções instaláveis.
 - f. Para o nome da impressora, use `myprinter`, dê qualquer descrição e localização que quiser e clique em Apply.
 - g. Clique em No, para não imprimir uma página de teste. A impressora deve aparecer na janela the Printer Configuration.
2. Para usar o comando `lpc` a fim de ver o status de todas as suas impressoras, digite o seguinte:

```
# lpc status
```

```
myprinter:
```

```
queuing is enabled
```

```
printing is enabled
```

```
no entries
```

```
daemon present
```

3. Para utilizar o comando `lpr` para imprimir o arquivo `/etc/hosts`, digite o seguinte:

```
$ lpr /etc/hosts -P myprinter
```

4. Para verificar a fila de impressão para a impressora, digite o seguinte:

```
# lpq -P myprinter
```

```
myprinter is not ready
```

Rank	Owner	Job	File(s)	Total Size
1st	root	655	hosts	1024 bytes

5. Para remover o trabalho de impressão da fila (cancelar), digite o seguinte.

```
# lprm -P myprinter
```

6. Para usar a janela de impressão a fim de definir a configuração básica do servidor que publica suas impressoras de modo que outros sistemas em sua rede local possam imprimir nelas, faça o seguinte:
 - a. A partir do desktop GNOME, selecione System ⇒ Administration ⇒ Printing.
 - b. Selecione Server ⇒ Settings.
 - c. Clique para ativar a caixa de seleção ao lado de “Publish shared printers connected to this session” e clique em OK.
7. Para permitir a administração remota de seu sistema a partir de um navegador web, siga estes passos:
 - a. A partir do desktop GNOME, selecione System ⇒ Administration ⇒ Printing.

- b. Selecione Server ⇒ Settings.
 - c. Clique para ativar a caixa de seleção ao lado de Allow remote administration e clique em OK.
8. Para demonstrar que você pode fazer a administração remota de seu sistema a partir de um navegador em outro sistema, faça o seguinte:
 - a. Na caixa de localização da janela de um navegador de outro computador em sua rede, digite **http://hostname:631**.
 - b. Substitua **hostname** pelo nome ou o endereço IP do sistema que está executando o serviço de impressão. A página inicial do CUPS deve aparecer a partir desse sistema.
9. Para usar o comando `netstat` a fim de ver o endereço que o daemon `cupsd` está ouvindo, digite o seguinte:

```
# netstat -tupln | grep 631
tcp    0  0  0.0.0.0:631  0.0.0.0:*    LISTEN  6492/cupsd
```
10. Para excluir a entrada da impressora `myprinter` de seu sistema, faça o seguinte:
 - a. Na janela de configuração da impressora, clique com o botão direito do mouse no ícone do `myprinter` e selecione Delete.
 - b. Quando solicitado, selecione Delete novamente.

Capítulo 17: Configurando um servidor web

1. Para instalar todos os pacotes associados ao grupo Web Server em um sistema Fedora, faça o seguinte:

```
# yum groupinstall "Web Server"
```
2. Para criar um arquivo chamado `index.html` no diretório designado para `DocumentRoot` no arquivo de configuração principal do Apache (com as palavras `My Own Web Server` dentro dele), faça o seguinte:
 - Determine a localização de `DocumentRoot`:

```
#grep ^DocumentRoot /etc/httpd/conf/httpd.conf
DocumentRoot "/var/www/html"
```
 - Ecoe as palavras “meu My Own Web Server” no arquivo `index.html` localizado em `DocumentRoot`:

```
# echo "My Own Web Server" > /var/www/html/index.html
```
3. Para iniciar o servidor web Apache e configurá-lo para iniciar automaticamente na inicialização do sistema e, então, verificar se ele está disponível a partir de um navegador na máquina local, faça o seguinte (você deve ver as palavras “My Own Web Server” exibidas se ele estiver funcionando corretamente):

O serviço `httpd` é iniciado e ativado de maneira diferente no Fedora e no RHEL. No Fedora, digite o seguinte:

```
# systemctl start httpd.service  
# systemctl enable httpd.service
```

No RHEL 6 ou anterior, digite:

```
# service httpd start  
# chkconfig httpd on
```

4. Para usar o comando `netstat` para ver quais portas o servidor `httpd` está ouvindo, digite o seguinte:

```
# netstat -tupln | grep httpd  
  
tcp    0 0  :::80          :::*           LISTEN        2496/httpd  
tcp    0 0  :::443         :::*           LISTEN        2496/httpd
```

5. Tente se conectar ao seu servidor web Apache a partir de um navegador que está fora do sistema local. Se ele falhar, corrija quaisquer problemas que você encontrar investigando o firewall, o SELinux e outros recursos de segurança.

Se você não tiver DNS configurado, porém, use o endereço IP do servidor para ver seu servidor Apache a partir de um navegador web remoto, como `http://192.168.0.1`. Se você não for capaz de se conectar, tente se conectar ao servidor a partir de seu navegador após a realização de cada uma das seguintes etapas no sistema executando o servidor Apache:

```
# iptables -F  
# setenforce 0  
# chmod 644 /var/www/html/index.html
```

O comando `iptables -F` limpa as regras de firewall. Se a conexão com o servidor web for bem-sucedida depois disso, você precisa adicionar novas regras de firewall para abrir as portas `tcp 80` e `443` no servidor. Adicionar uma regra antes da regra `DROP` ou `REJECT`, que faz o seguinte, deve fazer o truque.

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT  
-A INPUT -m state --state NEW -m tcp -p tcp --dport 443 -j ACCEPT
```

O comando `setenforce 0` coloca seu firewall no modo permissivo. Se a conexão com o servidor web for bem-sucedida depois disso, você precisa corrigir o contexto de arquivo do SELinux e/ou questões booleanas (provavelmente contexto de arquivo, nesse caso). O seguinte deve funcionar:

```
# chcon --reference=/var/www/html  
/var/www/html/index.html
```

Se o comando `chmod` funcionar, isso significa que o usuário `apache` e o grupo `apache` não tinham permissão de leitura sobre o arquivo. Você deve ser capaz de deixar as novas permissões como elas estão.

6. Para usar o comando `openssl` ou semelhante a fim de criar sua própria chave RSA privada e seu próprio certificado SSL autoassinado, faça o seguinte:

```
# yum install openssl
# cd /etc/pki/tls/private
# openssl genrsa -out server.key 1024
# chmod 600 server.key
# cd /etc/pki/tls/certs
# openssl req -new -x509 -nodes -sha1 -days 365 \
-key /etc/pki/tls/private/server.key \
-out server.crt
```

```
Country Name (2 letter code) [AU]: US
State or Province Name (full name) [Some-State]: NJ
Locality Name (eg, city) []: Princeton
Organization Name (eg, company) [Internet Widgits Pty Ltd]: TEST USE ONLY
Organizational Unit Name (eg, section) []: TEST USE ONLY
Common Name (eg, YOUR name) []: secure.example.org
Email Address []: dom@example.org
```

Você agora deve ter um arquivo de chave

`/etc/pki/tls/private/server.key` e um arquivo de certificado

`/etc/pki/tls/certs/server.crt`.

7. Para configurar o servidor web Apache para usar sua chave e seu certificado autoassinado a fim de servir conteúdo seguro (HTTPS), faça o seguinte: edite o arquivo `/etc/httpd/conf.d/ssl.conf` para alterar as localizações da chave e do certificado a fim de usar os que você acabou de criar:

```
SSLCertificateFile /etc/pki/tls/certs/server.crt
SSLCertificateKeyFile /etc/pki/tls/private/server.key
```

8. Para utilizar um navegador web para criar uma conexão HTTPS com o servidor web e ver o conteúdo do certificado que você criou, faça o seguinte: a partir do sistema executando o servidor Apache, digite **`https://localhost`** na caixa do navegador local. Você deverá ver uma mensagem que diz: “Essa conexão não é confiável”. Para completar a conexão, faça o seguinte:

- Clique em **I Understand the Risks**.
- Clique em **Add Exception**.
- Clique em **Get Certificate**.

- Clique em Confirm Security Exception.
9. Para criar um arquivo chamado `/etc/httpd/conf.d/example.org.conf`, que ativa a hospedagem virtual baseada em nome e cria uma máquina virtual que 1) ouve a porta 80 em todas as interfaces, 2) tem um administrador de servidor chamado `joe@example.org`, 3) tem um servidor chamado `joe.example.org`, 4) seu `DocumentRoot` é `/var/www/html/joe.example.org`, 5) tem uma `DirectoryIndex` que inclui pelo menos `index.html`, e cria um arquivo `index.html`, em `DocumentRoot` que contém as palavras “Welcome to the House of Joe”, faça o seguinte:

Crie um arquivo `example.org.conf` que se parece com o seguinte:

```
NameVirtualHost *:80
<VirtualHost *:80>
```

```
    ServerAdmin      joe@example.org
    ServerName       joe.example.org
    ServerAlias      web.example.org
    DocumentRoot     /var/www/html/joe.example.org/
```

```
    DirectoryIndex  index.html
</VirtualHost>
```

Eis como você pode criar o texto a fim de ir para o arquivo `index.html`:

```
# echo "Welcome to the House of Joe" >
/var/www/html/joe.example.org/index.html
```

10. Para adicionar o texto `joe.example.org` ao final da entrada `localhost` do arquivo `/etc/hosts` na máquina que está executando o servidor web e verificá-lo digitando `http://joe.example.org` na caixa de localização de seu navegador a fim de ver “Welcome to the House of Joe” quando a página é exibida, faça o seguinte:

- Atualize o arquivo `httpd.conf` modificado no exercício anterior:

```
# apachectl graceful
```

- Edite o arquivo `/etc/hosts` com qualquer editor de texto de modo que a linha `local host` apareça assim:

```
127.0.0.1 localhost.localdomain localhost joe.example.org
```

- A partir de um navegador no sistema local onde `httpd` está em execução, você deve ser capaz de digitar `http://joe.example.org` na caixa de localização para acessar o servidor web Apache usando a autenticação baseada em nome.

Capítulo 18: Configurando um servidor FTP

Preparação

faça as tarefas descritas aqui em um servidor FTP público funcional porque elas vão interferir nas suas operações no servidor. (Você pode, porém, usar essas tarefas para configurar um novo servidor FTP.)

1. Para determinar qual pacote fornece o serviço Very Secure FTP Daemon, digite o seguinte como root:

```
# yum search "Very Secure FTP"
...
===== N/S Matched: Very Secure FTP =====
vsftpd.i686 : Very Secure Ftp Daemon
```

A pesquisa encontrou o pacote vsftpd.

2. Para instalar o pacote Very Secure FTP Daemon em seu sistema e procurar os arquivos de configuração nesse pacote, digite o seguinte:

```
# yum install vsftpd
# rpm -qc vsftpd | less
```

3. Para iniciar o serviço Very Secure FTP Daemon e configurá-lo para começar quando o sistema é inicializado, digite o seguinte em um sistema Fedora:

```
# systemctl start vsftpd.service
# systemctl enable vsftpd.service
```

Em um sistema Red Hat Enterprise Linux, digite o seguinte:

```
# service vsftpd start
# chkconfig vsftpd on
```

4. No sistema que executa o servidor FTP, digite o seguinte para criar um arquivo chamado `test` no diretório de FTP anônimo que contém as palavras “Welcome to your vsftpd server”:

```
# echo "Welcome to your vsftpd server" > /var/ftp/test
```

5. Para abrir o arquivo `test` a partir do diretório inicial do FTP anônimo, usando um navegador no sistema executando o servidor FTP, faça o seguinte: inicie o navegador Firefox, digite o seguinte na caixa de localização e pressione Enter:

```
ftp://localhost/test\
```

O texto “Welcome to your Very Secure FTP Daemon server” deve aparecer na janela do Firefox.

6. Para acessar o arquivo `test` no diretório inicial do FTP anônimo, faça o seguinte. (Se você não puder acessar o arquivo, verifique se o firewall [iptables], o

SELinux e os TCP wrappers estão configurados para permitir acesso a esse arquivo, conforme descrito aqui.)

- a. Digite o seguinte na caixa de localização de um navegador em um sistema em sua rede que pode alcançar o servidor FTP (substitua *host* pelo nome totalmente qualificado ou o endereço IP do seu sistema):

```
ftp://host/test
```

Se você não puder ver a mensagem de boas vindas na janela de seu navegador, verifique o que pode estar impedindo o acesso. Para desativar temporariamente seu firewall (limpar suas regras *iptables*), digite o seguinte comando como usuário *root* de um shell em seu sistema de servidor FTP e tente acessar o site novamente:

```
# iptables -F
```

- b. Para desativar temporariamente o SELinux, digite o seguinte e tente acessar o site novamente:

```
# setenforce 0
```

- c. Para desativar temporariamente os TCP wrappers, adicione o seguinte ao início do arquivo */etc/hosts.allow* (certifique-se de remover essa linha novamente quando o teste terminar):

```
ALL: ALL
```

Depois de ter determinado o que está fazendo com que o arquivo em seu servidor FTP esteja indisponível, volte à seção “Protegendo seu servidor FTP” e siga os passos para determinar o que pode estar bloqueando o acesso ao seu arquivo. Prováveis causas são:

- Para *iptables*, certifique-se de que há uma regra abrindo a porta TCP 21 no servidor.
 - Para SELinux, certifique-se de que o contexto de arquivo está configurado como *public_content_t*.
 - Para TCP wrappers, certifique-se de que há uma linha *vsftpd: ALL* ou similar no arquivo */etc/hosts.allow*. Uma entrada como essa só é necessária se houver uma linha no arquivo */etc/hosts.deny* que nega o acesso a serviços que não são explicitamente permitidos.
7. Para configurar o servidor Very Secure FTP Daemon para permitir o upload de arquivos por usuários anônimos em um diretório *in*, faça o seguinte, como *root*, em seu servidor FTP:

- a. Crie o diretório *in* da seguinte maneira:

```
# mkdir /var/ftp/in  
# chown ftp:ftp /var/ftp/in  
# chmod 770 /var/ftp/in
```

b. Dentro do arquivo `/etc/vsftpd/vsftpd.conf`, certifique-se de que as seguintes variáveis estão configuradas:

```
anonymous_enable=YES  
write_enable=YES  
anon_upload_enable=YES
```

c. Configure o firewall `iptables` para permitir novas solicitações na porta TCP 21, adicionando a seguinte regra em algum momento antes da regra `DROP` ou `REJECT` final em seu arquivo `/etc/sysconfig/iptables`:

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 21 -j  
ACCEPT
```

d. Configure o firewall `iptables` para fazer o rastreamento de conexão carregando o módulo apropriado no arquivo `/etc/sysconfig/iptables-config`:

```
IPTABLES_MODULES="nf_conntrack_ftp"
```

e. Para o SELinux permitir o upload no diretório, primeiro configure os contextos de arquivo corretamente:

```
# semanage fcontext -a -t public_content_rw_t  
"/var/ftp/in(/.*)?"  
# restorecon -F -R -v /var/ftp/in
```

f. Então, configure o booleano SELinux para permitir o upload:

```
# setsebool -P allow_ftp_anon_write on
```

g. Reinicie o serviço `vsftpd` (`service vsftpd restart` ou `systemctl restart vsftpd.service`).

8. Para instalar o cliente FTP `lftp` (se você não tem um segundo sistema Linux, instale `lftp` no mesmo host que está executando o servidor FTP) e tentar fazer o upload do arquivo `/etc/hosts` no diretório `incoming` do servidor, execute os seguintes comandos, como usuário `root`:

```
# yum install lftp  
# lftp localhost  
lftp localhost: /> cd in  
lftp localhost: /in> put /etc/hosts  
89 bytes transferred  
lftp localhost: /in>  
quit
```

Você não será capaz de ver que copiou o arquivo `hosts` para o diretório de entrada. Mas digite o seguinte em um shell no host que está executando o servidor FTP para certificar-se de que o arquivo `hosts` está lá:

```
# ls /var/ftp/in  
Hosts
```

Se você não conseguir fazer o upload do arquivo, inicie os procedimentos de solução de problemas conforme descrito no Exercício 7, verifique as configurações de `vsftpd.conf` e revise a posse e as permissões sobre o diretório `/var/ftp/in`.

9. Usando qualquer cliente de FTP que você escolher, visite o diretório `/pub/linux/docs/man-pages` no site `ftp://kernel.org` e liste o conteúdo dele. Veja como fazer isso com o cliente `lftp`:

```
# lftp ftp://kernel.org/pub/linux/docs/man-pages
cd ok, cwd=/pub/linux/docs/man-pages
lftp kernel.org:/pub/linux/docs/man-pages> ls
drwxrwsr-x 2 536 536 24576 May 10 20:29 Archive
-rw-rw-r-- 1 536 536 1135808 Feb 09 23:23 man-pages-
3.34.tar.bz2
-rw-rw-r-- 1 536 536 1674738 Feb 09 23:23 man-pages-
3.34.tar.gz
-rw-rw-r-- 1 536 536 543 Feb 09 23:23 man-pages-
3.34.tar.sign
...
```

10. Usando qualquer cliente de FTP que você quiser, baixe o arquivo `man-pages-3.41.tar.gz` a partir do diretório `kernel.org` que você acabou de visitar para o diretório `/tmp` em seu sistema local.

```
# lftp ftp://kernel.org/pub/linux/docs/man-pages
cd ok, cwd=/pub/linux/docs/man-pages
lftp kernel.org:man-pages> get man-pages-3.41.tar.gz
1739208 bytes transferred in 4 seconds (481.0K/s)
lftp kernel.org:man-pages> quit
```

Capítulo 19: Configurando um servidor de compartilhamento de arquivos do Windows (Samba)

1. Para instalar os pacotes `samba`, `samba-client` e `samba-doc`, digite o seguinte, como `root`, a partir de um shell no sistema local:

```
# yum install samba samba-client samba-doc
```

2. Para iniciar e habilitar os serviços `smb` e `nmb`, digite o seguinte, como `root`, a partir de um shell no sistema local:

```
# systemctl enable smb.service
# systemctl start smb.service
# systemctl enable nmb.service
# systemctl start nmb.service
```

ou

```
# chkconfig smb on  
# service smb start  
# chkconfig nmb on  
# service nmb start
```

3. Para configurar o grupo de trabalho do servidor Samba como TESTGROUP, o netbios name como MYTEST e a string de servidor como Samba Test System, como usuário root em um editor de texto, abra o arquivo /etc/samba/smb.conf e altere três linhas de modo que elas fiquem assim:

```
workgroup = TESTGROUP  
netbios name = MYTEST  
server string = Samba Test System
```

4. Para adicionar um usuário Linux chamado phil ao seu sistema e adicionar uma senha Linux e uma senha Samba para phil, digite o seguinte, como usuário root de um shell (certifique-se de lembrar as senhas que você configurou):

```
# useradd phil  
# passwd phil  
New password: *****  
Retype new password: *****  
# smbpasswd -a phil  
New SMB password: *****  
Retype new SMB password: *****  
Added user phil.
```

5. Para configurar a seção [homes] de modo que os diretórios sejam navegáveis (browseable=yes) e graváveis (writable=yes) e que phil seja o único usuário válido, abra o arquivo /etc/samba/smb.conf, como root, e mude a seção [homes], assim:

```
[homes]  
comment = Home Directories  
browseable = yes  
writable = yes  
valid users = phil
```

6. Para configurar booleanos do SELinux de modo que phil possa acessar seu diretório inicial com um cliente Samba, digite o seguinte, como root, a partir de um shell:

```
# setsebool -P samba_enable_home_dirs on
```

7. A partir do sistema local, use o comando smbclient para listar que o compartilhamento homes está disponível.

```
# smbclient -L localhost  
Enter root's password: <ENTER>  
Anonymous login successful  
Domain=[DATAGROUP] OS=[Unix] Server=Samba 3.6.5-85.fc16
```


Sharename	Type	Comment
-----	----	-----
homes	Disk	Home Directories
...		

8. Para conectar-se ao compartilhamento `homes` a partir de uma janela do Nautilus (gerenciador de arquivos) no sistema local do servidor Samba para o usuário `phil` de uma maneira que permita que você arraste e solte arquivos para essa pasta, faça o seguinte:
 - a. Abra a janela do Nautilus (selecione o ícone de arquivos).
 - b. Sob o título Network no painel esquerdo, selecione Browse Network.
 - c. Abra o servidor Samba (ícone MYTEST).
 - d. Abra a compartilhamento `homes`.
 - e. Quando solicitado, digite **phil** como nome de usuário e digite a senha de phil.
 - f. Abra outra janela do Nautilus arraste e solte um arquivo para o diretório inicial de `phil`.
9. Para abrir o firewall de modo que qualquer pessoa que tenha acesso ao servidor possa acessar o serviço Samba (daemons `smbd` e `nmbd`), altere o arquivo `/etc/sysconfig/iptables` de modo que o firewall fique parecido com o seguinte (sendo as regras que você adiciona destacadas em negrito):

```
*filter
:INPUT ACCEPT [0:0]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [0:0]
-A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
-A INPUT -p icmp -j ACCEPT
-A INPUT -i lo -j ACCEPT
-I INPUT -m state --state NEW -m udp -p udp --dport 137 -
j ACCEPT
-I INPUT -m state --state NEW -m udp -p udp --dport 138 -
j ACCEPT
-I INPUT -m state --state NEW -m tcp -p tcp --dport 139 -
j ACCEPT
-I INPUT -m state --state NEW -m tcp -p tcp --dport 445 -
j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
-A FORWARD -j REJECT --reject-with icmp-host-prohibited
COMMIT
```

Então, digite o seguinte para as regras de firewall serem recarregadas:

```
# service iptables restart
```

10. Para abrir o compartilhamento `homes` novamente como o usuário `phil` de outro sistema em sua rede (Windows ou Linux) e certificar-se de que você pode arrastar e soltar arquivos para ele, faça o seguinte:

Essa etapa está apenas repetindo o exemplo Nautilus descrito anteriormente ou acessando uma janela do Windows Explorer e abrindo o compartilhamento (selecionando `Network` e, então, o servidor Samba). O truque é ter certeza de que o serviço foi disponibilizado através dos recursos de segurança do servidor Linux.

Se você não puder acessar o compartilhamento Samba, tente desativar o firewall e, então, desabilitar o SELinux. Se o compartilhamento estiver acessível quando você desligar um desses serviços, volte e depure os problemas com o serviço que não está funcionando:

```
# setenforce 0
# service iptables stop
```

Quando você tiver corrigido o problema, configure o SELinux de volta ao modo Enforcing e reinicie `iptables`:

```
# setenforce 1
# service iptables start
```

Capítulo 20: Configurando um servidor de arquivos NFS

1. Para instalar os pacotes necessários a fim de configurar o serviço NFS no sistema Linux que você escolheu, digite o seguinte, como usuário `root`, em um shell (Fedora ou RHEL):

```
# yum install nfs-utils
```

2. Para listar os arquivos de documentação que vêm no pacote que fornece o software do servidor NFS, digite o seguinte:

```
# rpm -qd nfs-utils
/usr/share/doc/nfs-utils-1.2.5/ChangeLog
...
/usr/share/man/man5/exports.5.gz
/usr/share/man/man5/nfs.5.gz
/usr/share/man/man5/nfsmount.conf.5.gz
/usr/share/man/man7/nfsd.7.gz
/usr/share/man/man8/blkmapd.8.gz
/usr/share/man/man8/exportfs.8.gz
...
```

3. Para determinar o nome do serviço NFS, iniciá-lo e ativá-lo, digite o seguinte, como usuário `root`, no servidor NFS:

```
# systemctl start nfs-server.service  
# systemctl enable nfs-server.service
```

4. Para verificar o status do serviço NFS que você acabou de iniciar no servidor NFS, digite o seguinte, como usuário root:

```
# systemctl status nfs-server.service
```

5. Para compartilhar um diretório `/var/mystuff` de seu servidor NFS como disponível para todos, somente leitura e com o usuário root cliente tendo acesso de root ao compartilhamento, primeiro crie o diretório de montagem assim:

```
# mkdir /var/mystuff
```

Então, crie uma entrada no arquivo `/etc/exports` semelhante a esta:

```
/var/mystuff *(ro,no_root_squash,insecure)
```

Para tornar o compartilhamento disponível, digite o seguinte:

```
# exportfs -v -a  
exporting */var/mystuff
```

6. Para garantir que o compartilhamento que você criou seja acessível a todos os hosts, primeiro verifique se `rpcbind` não está bloqueada pelos TCP wrappers, adicionando a seguinte entrada ao início do arquivo `/etc/hosts.allow`:

```
rpcbind: ALL
```

Para abrir as portas necessárias que permitem aos clientes alcançarem o NFS por meio do firewall `iptables`, você precisa abrir pelo menos TCP e UDP 111 (`rpcbind`), 20.048 (`mountd`) e 2049 (`nfs`), acrescentando as seguintes regras ao arquivo `/etc/sysconfig/iptables` e iniciar o serviço `iptables`:

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 111 -  
j ACCEPT  
-A INPUT -m state --state NEW -m udp -p udp --dport 111 -  
j ACCEPT  
-A INPUT -m state --state NEW -m tcp -p tcp --dport 2049  
-j ACCEPT  
-A INPUT -m state --state NEW -m udp -p udp --dport 2049  
-j ACCEPT  
-A INPUT -m state --state NEW -m tcp -p tcp --dport 20048  
-j ACCEPT  
-A INPUT -m state --state NEW -m udp -p udp --dport 20048  
-j ACCEPT
```

O SELinux deve ser capaz de compartilhar sistemas de arquivos NFS enquanto no modo Enforcing sem nenhuma alteração em contextos de arquivo ou opções booleanas. Para garantir que o compartilhamento que você criou possa ser compartilhado para leitura, execute o seguinte comando como usuário root no servidor NFS:

```
# setsebool -P nfs_export_all_ro on
```

7. Para ver as ações disponíveis a partir do servidor NFS, assumindo que o servidor NFS é nomeado `nfsserver`, digite o seguinte, a partir do cliente NFS:

```
# showmount -e nfsserver
Export list for nfsserver:
/var/mystuff *
```

8. Para criar um diretório chamado `/var/remote` e temporariamente montar o diretório `/var/mystuff` a partir do servidor NFS (nomeado `nfsserver` neste exemplo) nesse ponto de montagem, digite o seguinte, como usuário `root` do cliente NFS:

```
# mkdir /var/remote
# mount -t nfs nfsserver:/var/mystuff /var/remote
```

9. Para adicionar uma entrada de modo que a mesma montagem seja feita automaticamente quando você reiniciar, primeiro desmonte `/var/remote` como segue:

```
# umount /var/remote
```

Então, adicione uma entrada como a seguinte ao arquivo `/etc/fstab` no sistema cliente:

```
/var/remote nfsserver:/var/mystuff nfs bg,ro 0 0
```

Para testar se o compartilhamento está configurado corretamente, digite o seguinte no cliente NFS, como usuário `root`:

```
# mount -a
# mount | grep /var/remote
nfsserver:/var/mystuff on /var/remote type nfs4
(ro,relatime,vers=4,rsize=524288...
```

10. Para copiar alguns arquivos para o diretório `/var/mystuff`, digite o seguinte no servidor NFS:

```
# cp /etc/hosts /etc/services /var/mystuff
```

A partir do cliente NFS, para se certificar de que você pode ver os arquivos recém-adicionados a esse diretório e para se certificar de que você não pode gravar arquivos nele a partir do cliente, digite o seguinte:

```
# ls /var/remote
hosts services
# touch /var/remote/file1
touch: cannot touch '/var/remote/file1': Read-only file
system
```

Capítulo 21: Solução de problemas do Linux

1. Para entrar no modo Setup a partir da tela da BIOS em seu computador, faça o seguinte:
 - Reinicialize seu computador.
 - Dentro de alguns segundos, você deve ver a tela da BIOS, com uma indicação da tecla de função a pressionar para entrar no modo Setup. (Na minha workstation Dell, é a tecla de função F2.)
 - A tela azul da BIOS deve aparecer. (Se o sistema começar a inicializar o Linux, você não pressionou a tecla de função rápido o suficiente.)
2. Na tela de configuração da BIOS, faça o seguinte para determinar se seu computador é de 32 ou 64 bits, se inclui suporte à virtualização e se sua placa de rede é capaz de inicializar PXE (Preboot eXecution Environment).

Sua experiência pode ser um pouco diferente da minha, dependendo de seu computador e do sistema Linux. A tela de configuração da BIOS é diferente para diferentes computadores. Em geral, porém, você pode usar as teclas de seta e a tecla Tab para se deslocar entre diferentes colunas e pressionar Enter para selecionar uma entrada.

- Na minha estação de trabalho Dell, sob o título System, destaco Processor Info para ver que o meu é um computador com tecnologia de 64 bits. Examine a seção Processor Info, ou semelhante, em seu computador, para ver o tipo de processador que você tem.
 - Na minha estação de trabalho Dell, sob o título Onboard Devices, destaco Integrated NIC e pressiono Enter. A tela Integrated NIC que aparece à direita permite-me escolher para habilitar ou desabilitar a placa de rede (On ou Off) ou habilitar com PXE ou RPL (se quiser iniciar o computador pela rede).
3. Para interromper o processo de inicialização e impedi-lo de chegar ao carregador de inicialização GRUB, faça o seguinte:
 - Reinicialize o computador.
 - Logo após a tela da BIOS desaparecer, quando você vir a contagem regressiva para a inicialização do sistema Linux, pressione qualquer tecla (talvez a barra de espaço).
 - O GRUB carregador de inicialização deve aparecer pronto para permitir que você selecione qual kernel do sistema operacional inicializar.
 4. Para inicializar o computador no nível de execução 1 a fim de fazer manutenção do sistema, alcance a tela de inicialização do GRUB (como descrito no exercício anterior) e, então, faça o seguinte:
 - Use as teclas de seta para destacar o sistema operacional e o kernel que você quer inicializar.

- Digite **e** para ver as entradas necessárias para inicializar o sistema operacional.
- Mova o cursor para a linha que inclui o kernel. (Ela deve incluir a palavra `vmlinux` em algum lugar da linha.)
- Mova o cursor para o fim da linha, adicione um espaço e digite o número **1**.
- Siga as instruções para iniciar a nova entrada. Você provavelmente pressiona `Ctrl+X` ou `Enter`; então, quando vir a próxima tela, digite **b**.

Se funcionar, o sistema deve ignorar o prompt de login e inicializar diretamente para um shell de usuário `root`, onde você pode fazer tarefas administrativas sem fornecer uma senha.

5. Para iniciar o Red Hat Enterprise Linux (até o RHEL 6.x) de modo que possa confirmar cada serviço depois que ele é iniciado, faça o seguinte:
 - Siga os dois últimos exercícios, mas em vez de colocar um `1` no final de uma linha kernel, coloque a palavra `confirm`.
 - Quando o processo de inicialização alcançar o ponto em que ele inicia os serviços de runlevel, você será solicitado a confirmar (Y) ou negar (N) cada serviço, ou continuar (C) e simplesmente iniciar todos os demais serviços.

Note que essa opção não está disponível com as releases mais recentes do Fedora e do Ubuntu.

6. Para ver as mensagens que foram produzidas no buffer do kernel (que mostra a atividade do kernel durante a inicialização), digite o seguinte, a partir do shell, depois que o sistema terminar de inicializar:

```
# dmesg | less
```

7. Para executar `yum update` a partir do Fedora ou do RHEL e excluir qualquer pacote do kernel que esteja disponível, digite o seguinte (quando solicitado, digite **N** para não realmente fazer a atualização, se houver atualizações disponíveis):

```
# yum update --exclude='kernel*'
```

8. Para verificar os processos que estão aguardando conexões de entrada em seu sistema, digite o seguinte:

```
# netstat -tupln | less
```

9. Para verificar quais portas estão abertas em sua placa de rede externa, faça o seguinte:
Se possível, execute o comando `nmap` a partir de outro sistema Linux em sua rede, substituindo `yourhost` pelo nome ou endereço IP de seu sistema:

```
# nmap yourhost
```

10. Para limpar o cache de página do sistema e ver o efeito que isso tem sobre o uso da memória, faça o seguinte:

- Selecione Terminal a partir do menu de um aplicativo em seu desktop (ele está

localizado em menus diferentes para sistemas diferentes).

- Execute o comando `top` (para ver os processos que estão rodando em seu sistema) e digite um **M** maiúsculo para ordenar os processos por aqueles que consomem mais memória.
- Na janela Terminal, selecione File e Open Terminal para abrir uma segunda janela Terminal.
- Na janela do segundo terminal, torne-se usuário root (`su -`).
- Enquanto observa a linha Mem (coluna used) na primeira janela Terminal, digite o seguinte na segunda janela Terminal:

```
# echo 3 > /proc/sys/vm/drop_caches
```
- A memória RES usada deve cair significativamente na linha Mem. Os números na coluna RES para cada processo devem cair também.

Capítulo 22: Entendendo a segurança básica do Linux

1. Para criar uma lista a partir de um arquivo de log dos serviços que foram iniciados em seu sistema durante a inicialização, faça o seguinte:
 - a. Na linha de comando, digite `cat /var/log/boot.log`.
 - b. Examine o arquivo de log para encontrar os daemons iniciados.
2. Para listar as permissões sobre o arquivo de senhas de seu sistema e determinar se elas são apropriadas, você pode digitar `ls -l /etc/shadow` na linha de comando. (Se não existir nenhum arquivo de sombra, você precisa executar `pwconv`.) Eis as configurações apropriadas:

```
# ls -l /etc/shadow  
-r----- . 1 root root 1049 Feb 2 09:45 /etc/shadow
```
3. Para determinar a data de validade da senha da sua conta e se esta vai expirar usando um único comando, digite `chage -l nome_do_usuario` ou `cat /etc/shadow | grep nome_do_usuario`.
4. Para iniciar a auditoria de gravações no arquivo `/etc/shadow` com o daemon `auditd`, digite o seguinte na linha de comando:

```
# auditctl -w /etc/shadow -p w
```

Para verificar as configurações de auditoria, digite `auditctl -l` na linha de comando.
5. Para criar um relatório a partir do daemon `auditd` no arquivo `/etc/shadow`, digite `ausearch -f /etc/shadow` na linha de comando. Para desativar a auditoria sobre esse arquivo, digite `auditctl -W /etc/shadow -p w` na

- linha de comando.
6. Para comparar um pacote de software instalado em seu sistema com os metadados do pacote do Fedora ou do RHEL, digite **rpm -V nome_do_pacote** na linha de comando. Para o Ubuntu, digite **debsums nome_do_pacote** na linha de comando.
 7. Se suspeitar que sofreu um ataque malicioso em seu sistema hoje e que arquivos binários importantes foram modificados, você pode encontrar esses arquivos modificados digitando o seguinte na linha de comando: **find diretório -mtime -1** para os diretórios `/bin`, `/sbin`, `/usr/bin`, e `/usr/sbin`.
 8. Para instalar e executar `chkrootkit` a fim de ver se o ataque malicioso do exercício acima instalou um rootkit, escolha sua distribuição e faça o seguinte:
 - a. Para instalar em uma distribuição Fedora ou RHEL, digite **yum install chkrootkit** na linha de comando.
 - b. Para instalar em uma distribuição Ubuntu ou baseada no Debian, digite **sudo apt-get install chkrootkit** na linha de comando.
 - c. Para executar a verificação, digite **chkrootkit** na linha de comando e analise os resultados.
 9. Para encontrar arquivos com o conjunto de permissões SetUID, digite **find / -perm -4000** na linha de comando.
 10. Para encontrar arquivos com a permissão SetGID configurada, digite **find / -perm -2000** na linha de comando.

Capítulo 23: Entendendo a segurança avançada do Linux

1. Para criptografar um arquivo usando o utilitário `gpg` e uma chave simétrica, digite **gpg -c nome_do_arquivo** na linha de comando. O utilitário `gpg` vai pedir uma senha para proteger a chave simétrica.
2. Para gerar um chaveiro usando o utilitário `gpg`, digite **gpg --gen-key** na linha de comando. Você terá de fornecer as seguintes informações:
 - a. Que tipo de chave assimétrica você quer:
 - RSA e RSA (padrão)
 - DSA e ElGamal
 - DSA (apenas assinatura)
 - RSA (apenas assinatura)
 - b. O tamanho da chave (em número de bits) que você quer.
 - c. Por quantos dias, semanas, meses, anos, a chave deve ser válida. Você também pode

- solicitar que a chave seja válida permanentemente.
- d. Seu verdadeiro nome, e-mail, endereço e um comentário para criar o User ID para a chave pública.
 - e. A senha para a chave privada.
3. Para listar o chaveiro que você gerou, digite **gpg --list-keys** na linha de comando.
 4. Para criptografar um arquivo e adicionar sua assinatura digital usando o utilitário **gpg**, faça o seguinte:
 - a. Você deve ter gerado pela primeira vez um chaveiro (Exercício 2).
 - b. Depois de gerar o chaveiro, digite **gpg --output Encrypted&SignedFile --sign FiletoEncrypt&Sign** na linha de comando.
 5. Para usar o utilitário apropriado de resumo de mensagem a fim de assegurar que o arquivo baixado não está corrompido, você precisa fazer o seguinte. (Lembre-se de que um resumo de mensagem ou *message digest* também é chamado de soma de verificação ou *checksum*.)
 - a. Consulte o site de download para obter o arquivo ou número MD5 ou SHA-1.
 - Se for um número de soma de verificação, você precisa ir para a próxima etapa.
 - Se for um arquivo de soma de verificação, você também vai precisar baixar o arquivo e, então, usar o comando `cat` para exibir seu conteúdo na tela.
 - b. Se for um MD5, então digite **md5sum PrimeiroArquivoBaixado** na linha de comando e compare os números com o arquivo ou o número soma de verificação MD5 no site.
 - c. Se for um hash SHA-1, então digite **shasum PrimeiroArquivoBaixado** na linha de comando e compare os números com o arquivo ou número de soma de verificação SHA-1 no site.
 6. Para determinar se o comando `su` em seu sistema Linux é ciente do PAM, digite **ldd suDirectory/ su | grep pam** na linha de comando. *diretorio_su* é a localização do comando `su` que você encontrou com o comando `where su`. Se o comando `su` em seu sistema Linux for ciente do PAM, você vai ver um nome de biblioteca PAM listado ao emitir o comando `ldd`.
 7. Para determinar se o comando `su` tem um arquivo de configuração do PAM, digite **ls /etc/pam.d/su** na linha de comando. Se existir um arquivo de configuração do PAM, a lista de arquivos será exibida. Se ele não existir, então digite **cat /etc/pam.d/su** na linha de comando para exibir seu conteúdo. Os contextos PAM usados serão um dos seguintes: `auth`, `account`, `password`, `session`.
 8. Para listar os vários módulos PAM em seu sistema Fedora ou RHEL, digite **ls**

`/lib/security/pam*.so` na linha de comando. Para listar os vários módulos do PAM em seu sistema Linux Ubuntu, digite `sudo find / -name pam*.so` na linha de comando.

9. Para encontrar o “outro” arquivo de configuração do PAM em seu sistema (o “other”), digite `ls /etc/pam.d/other` na linha de comando. Um arquivo de configuração “other” que impõe Implicit Deny deve ser semelhante ao seguinte código:

```
$ cat /etc/pam.d/other
```

```
##PAM-1.0
```

auth	required	pam_deny.so
account	required	pam_deny.so
password	required	pam_deny.so
session	required	pam_deny.so

10. Para encontrar o arquivo de configuração de limites do PAM, digite `ls /etc/security/limits.conf` na linha de comando. Exiba o conteúdo do arquivo digitando `cat /etc/security/limits.conf`. Configurações nesse arquivo para evitar uma bomba fork serão parecidas com as seguintes:

@staff	hard	nproc
@staff	hard	maxlogins

Capítulo 24: Aprimorando a segurança do Linux com o SELinux

1. Para configurar seu sistema para o modo de operação permissivo no SELinux, digite `setenforce permissive` na linha de comando. Também seria aceitável digitar `setenforce 0` na linha de comando.
2. Para configurar seu sistema para o modo de funcionamento impositivo no SELinux sem alterar o arquivo de configuração do SELinux primário, tenha cuidado. É melhor não executar esse comando em seu sistema para um exercício até que você esteja pronto para usar o modo impositivo do SELinux. Use o seguinte comando: `setenforce enforcing` na linha de comando. Também seria aceitável digitar `setenforce 1` na linha de comando.
3. Para localizar e visualizar o tipo de política do SELinux atual, vá para o arquivo de configuração principal do SELinux, `/etc/selinux/config`. Para visualizá-lo, digite `cat /etc/selinux/config` na linha de comando.
4. Para listar contexto de segurança de um arquivo e identificar os diferentes atributos de

- contexto de segurança, digite **ls -Z nome_do_arquivo** na linha de comando.
- O contexto de usuário do arquivo terminará com um **u**.
 - O papel do arquivo terminará com uma **r**.
 - O tipo do arquivo terminará com um **t**.
 - Nível de sensibilidade do arquivo começa com um **s** e termina com um número. Pode ser listado em um intervalo de números, como **s0-s3**.
 - Nível de categoria do arquivo começa com um **c** e termina com um número. Pode ser listado em um intervalo de números, como **c0-c102**.
5. O comando que mudaria o atributo `type` de um arquivo é **chcon -t novoTipo _t nomeDoArquivo**. (Atenção: Só emita o comando no seu sistema se você quiser alterar o tipo do arquivo.)
6. Para listar contexto de segurança do processo atual e identificar os diferentes atributos de contexto de segurança, digite **ps -Z pid** na linha de comando.
- O contexto de usuário do processo terminará com um **u**.
 - O papel do processo terminará com uma **r**.
 - O tipo ou o domínio do processo terminará com um **t**.
 - O nível de sensibilidade do processo começa com um **s** e termina com um número. Pode ser listado em um intervalo de números, como **s0.s3**.
 - Nível de categoria do processo começa com um **c** e termina com um número. Pode ser listado em um intervalo de números, como **c0.c102**.
7. O comando que restauraria o contexto de um arquivo SELinux padrão é **restorecon -R nomeDoArquivo**. (Atenção: Só execute esse comando no seu sistema se você conhecer seus efeitos.)
8. Para obter uma lista das opções booleanas atuais utilizadas em seu sistema, digite **getsebool -a** na linha de comando. Você pode usar qualquer um dos seguintes comandos para modificar um dos valores booleanos: **setsebool nome_booleano off**, **setsebool nome_booleano on** ou **togglebool nome_booleano**.
9. O comando que listaria todos os módulos de política do SELinux em seu sistema, juntamente com seus números de versão, é **semodule -l**.
- Nota: Se você escreveu **ls *.pp** em sua resposta, tudo bem, mas esse comando não lhe dá os números de versão dos módulos de política. Apenas **semodule -l** fornecerá os números de versão.
10. Para criar uma mensagem de negação AVC e, então, analisar o(s) log(s) da mensagem, faça o seguinte:

- a. Como um usuário não administrador, digite **chcon -u fake_u nomeDoArquivo** na linha de comando.
- b. Se você tiver apenas o daemon `auditd` rodando em seu sistema, digite o seguinte na linha de comando:

```
aureport | grep AVC  
ausearch -m avc
```

- c. Se você tiver apenas o daemon `rsyslogd` em execução no sistema, digite na linha de comando:

```
grep "SELinux is preventing" /var/log/messages
```

- d. Se você tiver o daemon `rsyslogd` e o daemon `setroubleshootd` em execução no sistema, digite a linha de comando:

```
grep "SELinux is preventing" /var/log/messages  
sealert -l AVC_denial_message_id_number
```

Capítulo 25: Protegendo o Linux em uma rede

1. Para instalar o utilitário Network Mapper (`nmap`) em seu sistema local Linux:
 - a. No Fedora ou o no RHEL, digite **yum install nmap** na linha de comando.
 - b. No Ubuntu, `nmap` pode vir pré-instalado. Se não, digite **sudo apt-get install nmap** na linha de comando.
2. Para executar uma varredura de conexão TCP em seu endereço de loopback local, digite **nmap -sT 127.0.0.1** na linha de comando. As portas que você tem em execução em seu servidor Linux variarão. Mas podem parecer semelhantes ao seguinte:

```
# nmap -sT 127.0.0.1
```

```
...
```

PORT	STATE	SERVICE
25/tcp	open	smtp
631/tcp	open	ipp

3. Para executar uma varredura de conexão UDP em seu sistema Linux a partir de um sistema remoto:
 - a. Determine o endereço IP do servidor Linux, digitando **ifconfig** na linha de comando. O resultado será semelhante ao seguinte e o endereço IP do sistema vem depois de "inet addr:", na saída do comando `ifconfig`.

```
# ifconfig
```

...

```
p2p1 Link encap:Ethernet HWaddr 08:00:27:E5:89:5A
      inet addr:10.140.67.23
```

b. A partir de um sistema Linux remoto, digite o comando **nmap -sU endereçoIP** na linha de comando, usando o *endereçoIP* que você obteve a partir do comando anterior.

4. Para verificar se o daemon `ssh` em seu sistema Linux utiliza suporte a TCP Wrapper, digite **ldd /usr/sbin/sshd | grep libwrap** na linha de comando. O resultado será semelhante ao seguinte, se ele usar suporte a TCP Wrapper. Se não usar, não haverá nenhuma saída.

```
$ ldd /usr/sbin/sshd | grep libwrap
libwrap.so.0 => /lib/libwrap.so.0 (0x0012f000)
```

5. Para permitir o acesso às ferramentas de `ssh` em seu sistema Linux a partir de um sistema remoto designado e negar qualquer outro acesso usando TCP Wrappers, você terá de modificar tanto o arquivo `/etc/hosts.allow` como o arquivo `/etc/hosts.deny`. As modificações serão semelhantes ao seguinte:

```
# cat /etc/hosts.allow
```

```
...
```

```
sshd: 10.140.67.32
```

```
#
```

```
# cat /etc/hosts.deny
```

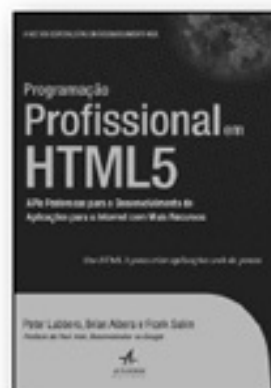
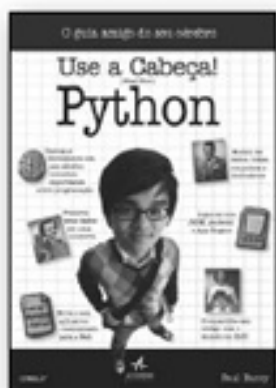
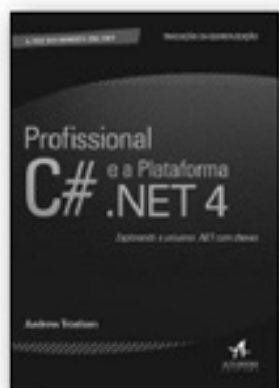
```
#...
```

```
ALL: ALL
```

6. Para determinar as atuais políticas e regras de firewall de `netfilter/iptables` de seu sistema Linux, digite **iptables -L** na linha de comando.
7. Para limpar as atuais regras de firewall de seu sistema Linux, digite **iptables -F** na linha de comando. Para restaurar as regras de firewall em um sistema Fedora ou RHEL, digite **iptables-restore < /etc/sysconfig/iptables**.
8. Essa é uma pergunta difícil! Você não pode configurar a política de firewall de um sistema Linux para rejeitar. Você pode configurá-lo para descartar, mas não rejeitar. Para configurar a tabela de filtro de firewall do seu sistema Linux para a cadeia de entrada como uma política de DROP, digite **iptables -P INPUT DROP** na linha de comando.
9. Para mudar a política da tabela de filtro do firewall do sistema Linux de volta para `accept` para a cadeia de entrada, digite **iptables -P INPUT ACCEPT** na linha de comando. Para adicionar uma regra a fim de descartar todos os pacotes de rede provenientes do endereço IP, 10.140.67.23, digite **iptables -A INPUT -s 10.140.67.23 -j DROP** na linha de comando.

10. Para remover a regra que você adicionou acima, sem limpar ou restaurar as regras de firewall do seu sistema Linux, digite **iptables -D INPUT 1** na linha de comando. Isso pressupõe que a regra que você adicionou acima foi a regra 1. Se não, mude o 1 para o número de regra apropriado em seu comando `iptables`.

Conheça alguns de nossos outros livros sobre informática_



Todas as imagens são meramente ilustrativas



ALTA BOOKS
EDITORA

- Idiomas
- Culinária
- Informática
- Negócios
- Guias de Viagem
- Interesse Geral



Visite também nosso site para conhecer lançamentos e futuras publicações!

www.altabooks.com.br



/alta_books



/altabooks



Seja autor da Alta Books

Todo o custo de produção fica por conta da editora e você ainda recebe direitos autorais pela venda no período de contrato.*

Envie a sua proposta para autoria@altabooks.com.br ou encaminhe o seu texto** para:
Rua Viúva Cláudio 291 - CEP: 20970-031 Rio de Janeiro

*Caso o projeto seja aprovado pelo Conselho Editorial.

**Qualquer material encaminhado à editora não será devolvido.



Seu guia definitivo para se tornar um especialista em Linux

Como um renomado autor em Linux e instrutor em tempo integral de Red Hat, Christopher Negus tem ajudado milhares de usuários iniciantes e experientes em Linux a se tornarem profissionais certificados. Nesta edição inteiramente atualizada do popular *Linux, A Bíblia* Negus e a colaboradora Christine Bresnahan dão a você um completo tutorial Linux, incluindo úteis exercícios no final de cada capítulo.

Este livro é uma ferramenta prática e uma excelente referência que fará você se converter de um iniciante em um usuário avançado.

Se você quer...

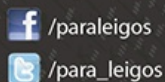
- Aprender Linux, mas nunca o usou antes
- Adquirir uma base para se tornar um profissional certificado em Linux
- Iniciar em uma carreira que vai durar décadas
- Dominar habilidades que você pode usar em todas as distribuições Linux

...este é o livro para você.

Saiba como:

- Instalar, configurar e usar poderosos sistemas Linux para desktops e servidores
- Configurar o sistema desktop Linux perfeito
- Executar tarefas de administração de sistema críticas
- Configurar seus próprios servidores de impressão, arquivo e web
- Obter um sistema estável e seguro usando ferramentas de segurança do Linux
- Dedicar-se à computação no nível corporativo

Chris Negus é instrutor da Red Hat, Inc. e autor de dezenas de livros sobre Linux e UNIX, incluindo todas as edições do *Linux, A Bíblia*, *CentOS Bible*, *Fedora Bible*, *Ubuntu Linux Toolbox*, *Linux Troubleshooting Bible*, *Linux Toys* e *Linux Toys II*. Christine Bresnahan tem mais de 25 anos de experiência como administradora de sistemas. É professora adjunta do Ivy Tech Community College, onde leciona administração de sistema Linux, segurança em Linux e segurança em Windows. Ela é coautora de *Linux Command Line and Shell Scripting Bible*, 2nd Edition.



Inicie com qualquer sistema Linux e avance para a computação corporativa utilizando Linux

- Utilize sua distribuição Linux favorita para aprender e testar suas habilidades com as ferramentas de linha de comando do Linux
- Aprenda tarefas de administração de sistema profissional usando o Fedora, o Red Hat Enterprise Linux ou outros sistemas corporativos

Categoria:
Computadores / Sistemas operacionais / Linux

Nível:
Iniciante / Avançado

Imagem da capa:
Aleksandar Valasevic / iStockPhotos

ISBN: 978-85-7608-774-8



9 788576 087748 >