

1 YEAR UPGRADE BUYER PROTECTION PLAN



Global Knowledge™
PROFESSIONAL REFERENCE



BUILDING **CISCO** REMOTE ACCESS NETWORKS



"BCRAN is about technological empowerment. This book will help you grow technically, expand your career opportunities, and enhance your experience of the Internet Revolution."

—Ralph Troupe, President and CEO
Callisma

Mark Edwards, CCNP, CCDP, MCSE, CNE

Ron Fuller, CCDP, CCNP, MCP, MCNE, CCIE

Andy McCullough, CCNA, CCDA

TECHNICAL EDITOR:

Wayne Lawson, CCIE, CCNA, CCDA, NNCSE,
CNX, MCSE, CNE, CBE

FREE Monthly
Technology Updates

One-year Vendor
Product Upgrade
Protection Plan

FREE Membership to
Access.Globalknowledge

SYN GRESS®

s o l u t i o n s @ s y n g r e s s . c o m

With over 1,500,000 copies of our MCSE, MCSD, CompTIA, and Cisco study guides in print, we have come to know many of you personally. By listening, we've learned what you like and dislike about typical computer books. The most requested item has been for a web-based service that keeps you current on the topic of the book and related technologies. In response, we have created solutions@syngress.com, a service that includes the following features:

- A one-year warranty against content obsolescence that occurs as the result of vendor product upgrades. We will provide regular web updates for affected chapters.
- Monthly mailings that respond to customer FAQs and provide detailed explanations of the most difficult topics, written by content experts exclusively for solutions@syngress.com.
- Regularly updated links to sites that our editors have determined offer valuable additional information on key topics.
- Access to "Ask the Author"™ customer query forms that allow readers to post questions to be addressed by our authors and editors.

Once you've purchased this book, browse to

www.syngress.com/solutions .

To register, you will need to have the book handy to verify your purchase.

Thank you for giving us the opportunity to serve you.



BUILDING CISCO REMOTE ACCESS NETWORKS

SYNGRESS®

Syngress Publishing, Inc., the author(s), and any person or firm involved in the writing, editing, or production (collectively "Makers") of this book ("the Work") do not guarantee or warrant the results to be obtained from the Work.

There is no guarantee of any kind, expressed or implied, regarding the Work or its contents. The Work is sold AS IS and WITHOUT WARRANTY. You may have other legal rights, which vary from state to state.

In no event will Makers be liable to you for damages, including any loss of profits, lost savings, or other incidental or consequential damages arising out from the Work or its contents. Because some states do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

You should always use reasonable care, including backup and other appropriate precautions, when working with computers, networks, data, and files.

Syngress Media® and Syngress® are registered trademarks of Syngress Media, Inc. "Career Advancement Through Skill Enhancement™," "Ask the Author™," "Ask the Author UPDATE™," and "Mission Critical™," and "Hack Proofing™" are trademarks of Syngress Publishing, Inc. Brands and product names mentioned in this book are trademarks or service marks of their respective companies.

KEY	SERIAL NUMBER
001	6LTM3ADSE2
002	XPS5PQB4C4
003	W3BM28FV7A
004	VBC8N4R52F
005	Z745QJJXBR
006	PF62RTSRR4
007	7TPLA5ZGG8
008	A2ZF743RTG
009	HN38M941DS
010	SM35MR55NT

PUBLISHED BY
Syngress Publishing, Inc.
800 Hingham Street
Rockland, MA 02370

Building Cisco Remote Access Networks

Copyright © 2000 by Syngress Publishing, Inc. All rights reserved. Printed in the United States of America. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but they may not be reproduced for publication.

Printed in the United States of America

1 2 3 4 5 6 7 8 9 0

ISBN: 1-928994-13-X

Copy edit by: Joeth Barlas and Judy Eby
Technical edit by: Wayne Lawson
Index by: Robert Saigh
Project Editor: Katharine Glennon

Proofreading by: Kate Bresnahan
Page Layout and Art by: Shannon Tozier
Co-Publisher: Richard Kristof

Distributed by Publishers Group West



Acknowledgments

We would like to acknowledge the following people for their kindness and support in making this book possible.

Richard Kristof, Duncan Anderson, Jennifer Gould, Robert Woodruff, Kevin Murray, Dale Leatherwood, Shelley Everett, Laurie Hedrick, Rhonda Harmon, Lisa Lavallee, and Robert Sanregret of Global Knowledge, for their generous access to the IT industry's best courses, instructors and training facilities.

Ralph Troupe and the team at Rt. 1 Solutions for their invaluable insight into the challenges of designing, deploying and supporting world-class enterprise networks.

Karen Cross, Kim Wylie, Harry Kirchner, John Hays, Bill Richter, Kevin Votel, Brittin Clark, Sarah Schaffer, Luke Kreinberg, Ellen Lafferty and Sarah MacLachlan of Publishers Group West for sharing their incredible marketing experience and expertise.

Peter Hoenigsberg, Mary Ging, Caroline Hird, Simon Beale, Julia Oldknow, Kelly Burrows, Jonathan Bunkell, Catherine Anderson, Peet Kruger, Pia Rasmussen, Denelise L'Ecluse, Rosanna Ramacciotti, Marek Lewinson, Marc Appels, Paul Chrystal, Femi Otesanya, and Tracey Alcock of Harcourt International for making certain that our vision remains worldwide in scope.

Special thanks to the professionals at Osborne with whom we are proud to publish the best-selling Global Knowledge Certification Press series.

From Global Knowledge

At Global Knowledge we strive to support the multiplicity of learning styles required by our students to achieve success as technical professionals. As the world's largest IT training company, Global Knowledge is uniquely positioned to offer these books. The expertise gained each year from providing instructor-led training to hundreds of thousands of students worldwide has been captured in book form to enhance your learning experience. We hope that the quality of these books demonstrates our commitment to your lifelong learning success. Whether you choose to learn through the written word, computer based training, Web delivery, or instructor-led training, Global Knowledge is committed to providing you with the very best in each of these categories. For those of you who know Global Knowledge, or those of you who have just found us for the first time, our goal is to be your lifelong competency partner.

Thank you for the opportunity to serve you. We look forward to serving your needs again in the future.

Warmest regards,

A handwritten signature in black ink, appearing to read "Duncan Anderson".

Duncan Anderson
President and Chief Executive Officer, Global Knowledge



Contributors

Tony Olzak (CCNP, MCSE) presently works as a consultant at Frontway in Toledo, OH. He specializes in the planning, design, and implementation of enterprise networks and is working towards the CCIE certification. In his free time, Tony likes to play guitar and write music.

Ron Fuller (CCIE, CCDP, CCNP-ATM, CCNP-Security, MCNE) has been in the internetworking industry for over six years. In that time he has worked as a consultant for clients looking for design, integration, and implementation expertise in Novell and Cisco environments.

Kevin Davis (MCP+I, MCSE, CCNA) is a consultant at Callisma in Dallas, TX. He has over ten years of WAN/LAN network design experience that includes some of the largest networks in the world using Cisco routers, WAN and LAN switches, Novell NetWare, and Microsoft Windows NT. Kevin graduated from the Dwight Look College of Engineering at Texas A&M University, College Station, TX with a degree in Computer Engineering. Recently he has consulted with some of the largest service providers in support of their Fortune 500 clients, and has authored several white papers on network security and anti-virus postures within a network.

Chris Larson (CNE, MCP+I, CCNP+Security) is a senior network engineer for PCT3, an international ASP. He has over 12 years of experience in network design and implementations.

Andy McCullough (CCNP, CCDA) has been in the network consulting industry for over five years. He is currently working at Lucent NPS as Chief Technical Architect. Andy has done design work for several global customers of Lucent Technologies including Level 3 Communications, Sprint, MCI/WorldCom, London Stock Exchange, and Birch Telecom. Prior to working for Lucent, Andy ran his own consulting company, Cisco reseller, and ISP. Andy is also an assistant professor teaching at a Cisco Network Academy in Lenexa, KS.

Venkata Ammu holds a master's degree in Computer Science, and is presently a manager at Callisma. Venkata has over 15 years of experience in the internetworking area, specifically in designing and implementing large networks. Venkata lives with his wife Syamala, son Kartik, and daughter Bhargavi in East Brunswick, NJ.

Mark Edwards (CCNP, CCDP, MCSE, CNE) is an IT consultant based in South Wales, UK. He qualified from the University of Glamorgan with a BSc (Hons) in Computer Science in 1994, and has been working in the network field ever since. He is currently working on achieving CCIE status and is set to take the lab in late 2000. Mark has worked for many large international organizations and has held a wide variety of roles in various major projects. These have included project management, infrastructure design and implementation, training, and testing. Mark is currently working as an infrastructure consultant for ACNielsen on their global intranet team. Mark lives in Cardiff, UK, and can be contacted at cetcrt@globalnet.co.uk.

Darrel Hinshaw (CCIE, CCNA, MCSE, MCP+I, MCNE) is a senior consultant at Callisma. He currently provides senior-level strategic and technical consulting to all Callisma clients in the south-central region of the US. His specialties include Cisco routers and LAN switches, Microsoft NT, Novell design and implementation, strategic network planning, network architecture and design, and network troubleshooting and optimization. Darrel's background includes positions as a senior engineer at Chancellor Media, and as a senior network engineer at Lucent Technologies in the Octel Messaging Division.

Richard Hamilton is a senior consultant at Callisma. He is currently responsible for leading engineering teams specializing in the design and deployment of ATM and WAN/LAN technologies. He is accountable for providing end-to-end solutions for diverse networking environments primarily in the service provider space. Richard has spent the past 13 years in both staff and consulting roles in the financial and service provider industries, for companies that include International Network Services Inc., and NatWest/Fleet Bank N.A.

Pankaj Chandhok is a senior network design consultant who has engineered, maintained, and managed worldwide LAN/WAN network infrastructures. He works at Callisma in Parsippany, NJ where he is accountable for leading a project team in the design and implementation of large-scale network projects. He has also taught formal training classes ranging from Microsoft Windows to Layer 3 Switching concepts. His formal education includes a M.S. and B.S. in Electrical Engineering from Rutgers University. He and his wife Poonam are expecting their first baby this year. He can be contacted at pankaj_chandhok@yahoo.com.

Cameron Brandon (MCSE, CNE, CNA, MCSE+Internet, A+, Network+) works as a network engineer/administrator in Portland, OR, and he specializes in Windows NT with BackOffice Integration. He helped in Intel's large-scale migration at its Oregon facility to Windows NT. Cameron completed all of his certifications in five months, demonstrating that determination and a strong sense of direction are the keys to success in one's career.

J.D. Wegner is a founder and director of The Empowerment Group, Inc. He has been working with computers for over 30 years, the last twelve of those involved with the design, installation, and support of data networks. As an instructor and course director for Global Knowledge, he has presented topics ranging from Internetworking with TCP/IP to Web Security to IP Address Management to thousands of IT professionals in the U.S. and abroad. His clients include many of the Fortune 500 as well as several government agencies. He lives in Hickory, NC with his wife, Laurie, and their two children, David and Sarah.

John Senkow (CCNA, CCDA, CCNP) is currently a consulting engineer at Callisma, in Philadelphia, PA. His key responsibilities include design, configuration, implementation, and analysis of LAN/WAN architectures. John has over five years of experience working with various network infrastructures. His background is primarily in Cisco routers and switches as well as in SNMP management.

Dave Capeci (MCSE, MCP+I, MCT) is the manager of professional services at Callisma. His professional experience includes positions as a senior network executive for a Fortune 1000 insurance company, and the director of technology for a regional healthcare system. He has been published in Windows NT Magazine and Windows 2000 Magazine. Dave lives in suburban Philadelphia, PA with his wife, Janine, and three children.

Brett M. Summerville (CCNA, MCP) is a network consultant at Callisma. He has over six years of LAN/WAN data communications experience providing internal and external clients with design, development, management, and operation of complex, multi-protocol, multi-platform internetworking environments.

Melissa Craft (CCNA, MCSE, Network+, CNE-5, CNE-3, CNE-4, CNE-GW, MCNE, Citrix CCA) designs business computing solutions using technology to automate processes. Her consulting experience has incorporated extensive project management, LAN and WAN design, deployment and operational turnover. Currently, Melissa is Director of e-Business Offering Development for MicroAge Technology Services, a global systems integrator. Melissa is a member of the IEEE, the Society of Women Engineers and American MENSA, Ltd. Melissa currently resides in Glendale, AZ with her family, and can be contacted at mmcrafc@compuserve.com.

Technical Editor

Wayne Lawson (CCIE #5244, CCNA, CCDA, Nortel Networks NNCSE, Certified Network Expert (CNX) Ethernet, Microsoft MCSE, Novell CNE, Banyan Systems CBE) is a systems engineer with Cisco Systems in Southfield, MI. Wayne has over nine years of experience in the IT industry. His core area of expertise is in the routed wide area network (WAN) arena, as well as the campus switching arena.

Contents

Foreword	xxiii
Chapter 1: Introduction to BCRAN and Cisco Remote Access Solutions	1
Introduction	2
WAN Connection Requirements	2
WAN Topology and Specifications	3
Connection Types	4
Dedicated Connections	4
Circuit-Switched Connections	6
Packet-Switched Connections	10
WAN Encapsulation Protocols	11
SDLC	11
HDLC	11
SLIP	12
PPP	12
X.25	12
Frame Relay	13
ATM	13
Selecting Cisco Access Servers and Routers	14
700 Series	14
800 Series	14
900 Series	15
1000 Series	15
1400 Series	15
1600 Series	15
1700 Series	16
2500 Series	16
2600 Series	16
3000 VPN Concentrators	16
3600 Series	16

AS5000 Series	17
7100, 7200, and 7500 Series	17
Considerations Before Installing a Remote Access Network	17
Network Planning and Design	18
Proper Analysis	18
Identifying Suitable Equipment for Each Site	21
Staging and Testing	23
Remote Access Network Implementation Considerations	24
Change Control Procedures	24
Backout Plans	24
Minimizing Network Interruption	25
Coordination of Resources	25
Verifying and Troubleshooting Network Installation	25
Summary	25
FAQs	26
Chapter 2: Configuring Asynchronous Remote Access Connections	29
Introduction	30
Modem Overview	30
Digital Modems	32
Modem Signaling and Cabling	32
Cisco Console and AUX Port Cabling	33
Modem Modulation Standards	34
Error Control and Data Compression Methods	35
Automatic Repeat Request (ARQ)	36
Microcom Networking Protocol (MNP)	36
Link Access Procedure for Modems (LAPM)	37
Data Compression Protocols	37
Configuring an Asynchronous Connection	38
Router Configuration	39
Modem Configuration	48
Manual Configuration	48
Automatic Configuration	51
Chat Scripts	55
Providing Asynchronous Dial-in	
Terminal Services	56
Terminal Services	57
The Autocommand Feature	66
Menus	67
EXEC Callback	69
Summary	73
FAQs	74

Chapter 3: Using PPP to Provide Remote Network Access	75
Introduction	76
PPP Overview	76
PPP Features	77
Multiple Protocols per Communication Line	77
Authentication	77
Link Configuration and Negotiation	77
Error Detection	77
Header Compression	78
Bonding of Communications Links	78
LCP	79
NCP	81
PPP vs. SLIP and ARAP	81
Relevant RFCs	82
Configuring PPP	83
Autoselect	84
PPP Addressing Methods	84
PPP Link Control Options	86
PAP and CHAP Authentication	86
Authentication Failures	91
PPP Callback	91
MSCB	93
PPP Compression	93
MPPC	93
Compression Effects	94
Multilink PPP	94
Multichassis Multilink PPP	96
Verifying and Troubleshooting PPP	99
PPP and Cisco Access Servers	99
PPP and ISDN Connections between Cisco Routers	99
Providing Remote Access Services for	
Microsoft Windows Clients	104
Microsoft Specific PPP Options	104
Windows 95 Clients	105
Windows 98 Clients	105
Windows NT4 Clients	107
Windows 2000 Clients	108
Troubleshooting Microsoft Windows Connections	110
Summary	111
FAQs	112

Chapter 4: Utilizing Virtual Private Network (VPN) Technology for Remote Access Connectivity	113
Introduction	114
VPN Technology	114
ISAKMP & IKE	114
IPSec	115
DES, Triple Pass DES & 3DES	116
VPN Operation	116
Cisco VPN Terminology	117
Site-to-Site VPN	119
An Intranet Solution	119
Configuring ISAKMP/IKE	120
Configuring IPSec	123
An Extranet Solution	126
Remote Access VPN	130
Configuring IPSec on the Network Access Server	131
Service Provider Solution	135
Configuring ISAKMP	136
Configuring IPSec	137
Configuring the VPN Client	138
Verifying and Debugging VPN Operation	140
Advantages and Disadvantages of VPN	143
Cisco's VPN Solutions	145
FW Solution (HW Accelerator)	145
3000 Series Product Line	145
Traditional Router with FW Feature Set	147
Policy Manager 2.x (VPN Configuration and Management)	147
Summary	148
FAQs	149
Chapter 5: Using ISDN and DDR to Enhance Remote Access Connectivity	151
Introduction	152
ISDN Overview	152
Basic Rate Interface (BRI)	154
BRI Call Setup	154
BRI Reference Points and Functional Groups	155
Primary Rate Interface (PRI)	156
PRI Reference Points and Functional Groups	157
ISDN Protocol Layers	157
U-plane	158
C-plane	159

ISDN Call Setup and Teardown	159
Dial-on-Demand Routing (DDR)	159
Interesting Traffic	161
Topologies	162
Point-to-Point Topology	162
Fully Meshed Topology	162
Hub-and-Spoke Topology	164
Dialer Interfaces	165
Dialer Profiles	166
Dialer Rotary Groups	166
Dialer Addressing	166
Dialer Mapping	166
Encapsulation	167
Supported Interfaces	167
Configuring ISDN and DDR	168
Caller ID Screening	179
Routing Issues with DDR	179
Static and Default Routes	180
Snapshot Routing	180
OSPF On-demand Circuits	181
Route Redistribution	182
Monitoring and Troubleshooting ISDN and DDR	182
Monitoring the ISDN Interface	182
Monitoring the Dialer	186
Monitoring PPP Multilink	188
Monitoring Snapshot Routing	189
Troubleshooting ISDN and DDR	190
Walkthrough	195
Summary	203
FAQs	205
Chapter 6: Enabling Dial-on-Demand Routing (DDR)	209
Introduction	210
Dialer Rotary Groups	210
Configuring Dialer Rotary Groups	210
Dialer Profiles	213
Physical Interface	214
Dialer List	214
Dialer Interface	214
Dialer Pool	214
Map Class	214
Configuring Dialer Profiles	215

Virtual Profiles	217
Case 1: Create a Virtual Profile Using the Virtual Template	218
Configure a Virtual Profile Using Virtual Templates	218
Case 2: Create a Virtual Profile Using the AAA Server	219
Configure a Virtual Profile Using the AAA Server	220
Case 3: Create a Virtual Profile Using Both the Virtual Template and AAA Server	221
Configure a Virtual Profile Using Both the Virtual Template and AAA Server	222
Fine Tuning Connections	223
Dialer Lists	223
Dialer Timers	225
Walkthrough	226
Summary	231
FAQs	232
Chapter 7: Configuring and Backing Up Permanent Connections	233
Introduction	234
Configuring Point-to-Point Connections	234
X.25 Connections	237
X.25 Overview	237
Data Terminal Equipment (DTE) and Data Circuit-Terminating Equipment (DCE)	238
Frames in X.25	238
X.25 Virtual Circuits	240
X.25 Call Setup and Disconnection	240
Configuring X.25	241
Verifying and Troubleshooting X.25 Connections	245
Frame Relay Connections	248
Frame Relay Overview	248
Frame Relay Topologies	253
Split Horizon and Poison Reverse	255
Subinterfaces	257
Configuring Frame Relay	259
Verifying and Troubleshooting Frame Relay	263
Loopback Tests	266
Local Loopback	266
Remote Loopback	267
Frame Relay Traffic Shaping (FRTS)	271
Enable Frame Relay Traffic Shaping (FRTS) on the Interface	272

Configuring Traffic Shaping	272
Verifying Traffic Shaping	280
ATM Connections	290
ATM Overview	290
ATM Packet Format	290
ATM Adaptation Layer (AAL)	291
ATM Virtual Circuits	292
PVC Mapping and Circuit Buildup	292
Configuring ATM	293
Verifying and Troubleshooting ATM Connections	297
The debug atm packet Command	300
The debug atm state Command	302
The debug atm ilmi Command	303
Backing up Permanent Connections	305
Backup Interface	305
The backup load Command	308
Floating Static Routes and Default Routes	309
Frame Relay Configuration with ISDN backup	310
Dialer Watch	315
Configuring a Dialer Profile	316
Verifying and Troubleshooting Backup Connections	317
Routing Issues	321
Redundant Hardware and Links/Design and Performance Issues	321
Load Balancing	322
Summary	323
FAQs	324
Chapter 8: Securing your Remote Access Network	325
Introduction	326
What is a Firewall?	326
Cisco IOS Firewall Feature Set	327
Firewall Feature Set Benefits and Features	327
Phase I	327
Phase I+	327
Phase II (Full Features)	327
Key Benefits	328
AAA Overview	328
AAA Servers	329
CiscoSecure	330
Authentication	331
Authorization	331

Accounting	332
Method-Lists	332
Security Protocols	333
Remote Authentication Dial-in User Service (RADIUS)	333
Terminal Access Controller Access Control System Plus (TACACS+)	333
Comparing TACACS+ and RADIUS	334
Using RADIUS and TACACS+ for AAA Services	336
Configuring AAA	336
Enabling AAA	336
Configuring the RADIUS or TACACS+ Parameters	336
Configuring TACACS+ Parameters	337
Configuring RADIUS Parameters	338
Configuring AAA Authentication	339
The aaa authentication login Command	339
The aaa authentication ppp Command	340
The aaa authentication enable default Command	341
Configuring AAA Authorization	342
Configuring AAA Accounting	344
Virtual Profiles and AAA	346
Scenario 1: Virtual Profiles Using Virtual Templates	347
Scenario 2: Virtual Profiles Using AAA Configuration	348
Scenario 3: Virtual Profiles Using Virtual Templates and AAA Configuration	349
Configuring Virtual Profiles	349
Configuring Virtual Profiles Using Virtual Templates	349
Configuring virtual Profiles Using AAA Configuration	352
Configuring Virtual Profiles Using Virtual Templates and AAA Configuration	352
Per-User Configuration Example	354
User 'Remote' RADIUS Configuration	354
Network Access Server Configuration (central)	355
Monitoring and Verifying AAA Access Control	358
AAA Debug And Show Commands	358
Walkthrough	362
Summary	368
FAQs	368
 Chapter 9: Optimizing Network Performance with Queuing and Compression	 371
Introduction	372
Network Performance	372
Queuing Overview	373

Queuing Methods and Configuration	373
First-in, First-out Queueing (FIFO)	374
Weighted Fair Queueing (WFQ)	375
Priority Queueing (PQ)	383
Custom Queueing (CQ)	387
Class-Based Weighted Fair Queueing (CBWFQ)	390
>Selecting a Cisco IOS Queueing Method	392
Verifying Queueing Operation	395
Weighted Random Early Detection (WRED) Overview	395
Tail Drop	396
Weighted Random Early Detection (WRED)	396
Flow-based WRED	396
Data Compression Overview	397
The Data Compression Mechanism	397
Header Compression	398
Link and Payload Compression	399
Per-Interface Compression (Link Compression)	401
Per-Virtual Circuit Compression (Payload Compression)	401
Hardware Compression	401
Selecting a Cisco IOS Compression Method	402
Verifying Compression Operation	403
Summary	403
FAQs	404
Chapter 10: Requirements for Network Address Translation in Remote Access Networks	407
Introduction	408
NAT Overview	408
Terminology	409
NAT Operation	411
Traffic Types Supported	412
NAT Commands	413
Translate Inside Source Addresses	414
Dynamic Translation	414
Configuring Dynamic NAT	416
Dynamic NAT Translation Screen Captures	418
Address Overloading	421
Configuring Address Overloading	423
Address Overloading Screen Captures	424
Static Translation	425
Configuring Static NAT Translations	427
Static NAT Translation Output	428
Dual Address Translation (Overlapping Networks)	430

Configuring Overlapping Networks	434
TCP Load Distribution	436
Configuring TCP Load Distribution	438
Output Showing TCP Load Distribution	440
Changing NAT Timeouts	443
NAT to an ISP	444
NAT to an ISP using Easy IP	445
Easy IP Operation	446
PAT to an ISP Using a Cisco 700 Series Router	449
Walkthrough	450
Summary	453
FAQs	454
Chapter 11: Private Addressing and Subnetting	
Large Networks	457
Introduction	458
Strategies to Conserve Addresses	458
Classless Inter-Domain Routing (CIDR)	459
Variable-Length Subnet Mask (VLSM)	459
Private Addresses	459
Addressing Economics	460
An Appeal	462
Public vs Private Address Spaces	463
Can I Pick My Own?	463
RFC 1918—Private Network Addresses	465
The Three Address Blocks	465
Considerations	466
Which to Use When	467
Strategy for Subnetting a Class A Private Network	468
The Network	469
The Strategy	470
Address Assignment	471
The Headquarters LANs	471
The WAN Links from Headquarters to the	
Distribution Centers	472
The Distribution Center LANs	472
The Store LANs	473
Results	474
BGP Requirements	475
IBGP and EBGP Requirements	479
Loopback Interfaces	481
Summary	482
FAQs	482

Appendix: Implementing the Windows 2000 Servers	485
Introduction	486
Installing Windows 2000	487
Overview of a Scripted Installation	488
Overview of Disk Duplication Methods	491
SYSPREP	491
RIPREP	492
Windows 2000 Setup Phases	495
WINNT Phase	496
Text Mode	496
GUI Mode	496
Installing the Active Directory	497
Which Domain First?	498
Which Server First?	499
DCPromo	500
Installing the Recovery Console	503
Populating a Domain with Organizational Units (OUs) and Objects	504
Creating an OU	505
Create an OU for Hidden Objects	505
Delegating Authority	506
Creating a User Account	508
Creating Groups	511
Publishing Printers	513
Publishing Folders	514
Applying a Group Policy	515
Setting Up Sites	516
Installing and Configuring Windows 2000 Components	519
Configuring DNS	519
Configuring the Distributed File System	521
Public Key Infrastructure	522
Internet Information Services	525
Asynchronous Transfer Mode	527
Terminal Services	527
Configuring Routing and Remote Access Services	534
DHCP	535
WINS	537
Case Studies	537
ABC Chemical Company	537
West Coast Accounting	539
Summary	540
FAQs	544
Index	547

Foreword

We are in the middle of a revolution! Never doubt that the Internet Revolution has changed history and that we're a part of this tremendous change and activity. Not unlike the Industrial Revolution of the eighteenth and nineteenth centuries, the Internet Revolution spans two centuries and the end is nowhere in sight. Revenue per employee increased by 19 percent from 1998 to 1999, as companies leveraged the Internet to increase operational efficiency. Leveraging the Internet means providing robust and reliable methods for remote access.

Building Cisco Remote Access Networks (BCRAN) is a book that covers the key technology area of remote access. Cisco is a dominant force in this Internet economy. BCRAN is more than a product line; it is a technology delivery platform of products. This book covers the key protocols involved, as well as technical connectivity considerations. It provides the reader with instruction on interconnecting central sites to branch offices, and supporting home office workers and telecommuters. BCRAN is about technological empowerment.

The Internet is the great enabler, in addition to being the great equalizer. Cisco remote access technology delivers on the promise of distance learning, e-learning and productive telecommuting. With Cisco remote access networks as a platform, both enterprises and service providers can reach a broader constituency and a bigger subscriber base, and empower remote workers. In this increasingly competitive labor market, the company that brings technology into the home will capture and retain more talent in the Internet economy. The Internet has brought e-learning right to our desktops, enabling lifelong learning. Web technologies and higher-speed access provide us with extreme productivity.

The Internet is moving fast. Only the fast will survive. We must do business at the speed of the Internet, absorbing change, anticipating change, and

executing change in a quick and fluid fashion. If you are reading this for your company, *Building Cisco Remote Access Networks* should be part of your strategy to recruit and retain, deliver greater productivity, and provide that technological enablement. If you are reading this as an individual, this book will help you grow technically, expand your career opportunities and enhance your experience of the Internet Revolution.

Sincerely,

Ralph Troupe
President and CEO
Callisma

Chapter 1

Introduction to BCRAN and Cisco Remote Access Solutions

Solutions in this chapter:

- WAN connection requirements
- WAN topology and specifications
- Network planning and design
- Considerations before installation
- Selecting Cisco access servers and routers
- Implementation considerations

Introduction

Wide area network (WAN) connections are used to connect geographically separate networks together. When a device on one network wants to communicate with a device on a different network or remote site, traffic has to traverse one or more WAN links. Unlike a local area network (LAN), a service provider typically provides the physical WAN connections. Studies have shown that these costs can comprise 80 percent of the annual network budget.

Remote connections link branch offices, telecommuters, and mobile users to a central office or to the Internet. Given the high cost of permanent WAN connections, if the traffic requirement between these sites is not for 24 hours per day connectivity, significant cost savings may be realized by using a dial-up connection over the Public Switched Telephone Network (PSTN) or the Integrated Services Digital Network (ISDN). These links connect only when traffic needs to be transferred.

In this chapter, we will start by looking at WAN connection requirements, topologies, and specifications. We will review the Cisco Access Server product line as well as the routers that are currently available. We will also review where the products fall within the Cisco product set. Additionally, we will look at some of the remote access options that are currently available.

In the second part of this chapter, we will look at what issues should be considered when planning the design, implementation, and installation of a Cisco remote access network, as well as identify suitable equipment for each site.

WAN Connection Requirements

WAN links connect various facilities—ranging in distance from two neighboring cities to different continents—for the exchange of information. These connections are usually rented from a service provider, and prices are based on distance, bandwidth, and the communication technologies chosen.

Connection requirements vary widely, depending on the function of the link; a small office/home office (SOHO) may only need a 56K modem to check e-mail. However, if files are transferred regularly, or most resources are at the central site, a faster ISDN link may be preferred. In a scenario where you have multiple departments transferring large files or documents, a dedicated solution such as Frame Relay, Point-to-Point Protocol (PPP), or a High-Level Data Link Control (HDLC) is usually a better choice.

Consider the future bandwidth requirements and networking technologies of the company when choosing a type of link and equipment. Will your

phone system use the network to deliver voice to remote locations? Do you have plans for video conferencing? Maybe creating a virtual private network (VPN) between sites using your Internet connections and some form of encryption (for example, IPSec) is more cost-effective for your organization. This is covered in detail in Chapter 4.

The network must balance the needs of the company with the total cost of ownership. The best way to accomplish this is to gain a good understanding of the types of WAN connections and product lines available.

WAN Topology and Specifications

The topology of a WAN can be broken down into four areas that divide the responsibility of the wiring and equipment between the customer and a service provider:

Customer Premises Equipment (CPE): Refers to all the equipment and wiring for which the customer is responsible. This includes any routers and channel service units/data service units (CSU/DSU) that are not rented from the service provider.

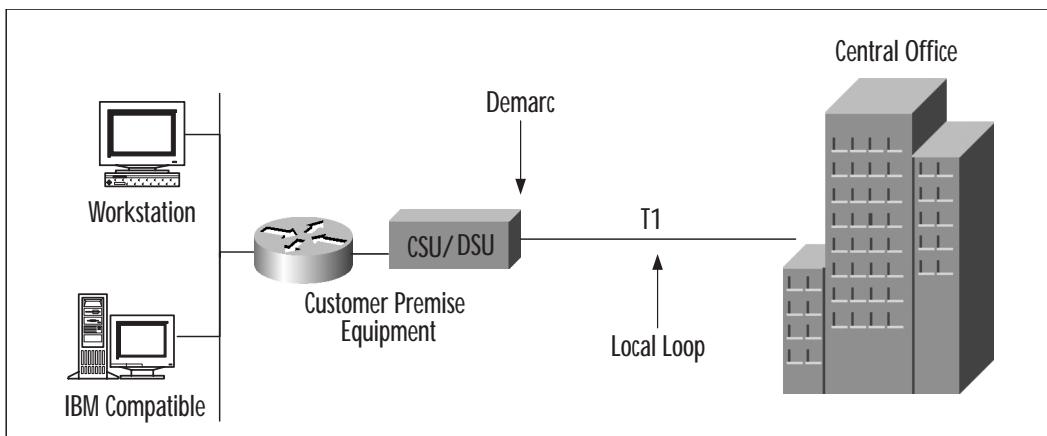
Demarc: Short for “demarcation point,” it marks the division between customer and service provider responsibility.

Local Loop: Wiring that runs from the demarc to the Central Office.

Central Office (CO): Often referred to as the “local POP,” or Point of Presence. This is where the local loop connects to the service provider’s backbone.

Refer to Figure 1.1 for an example of these four areas.

Figure 1.1 Customer premise equipment to the central office.



Connection Types

All current and emerging WAN technology can be grouped into three categories:

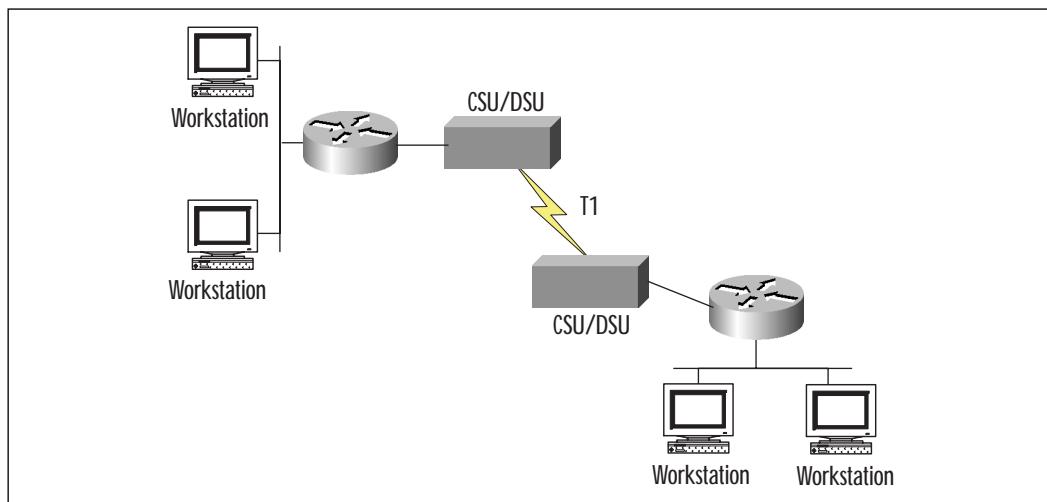
- Dedicated Connections
- Circuit-Switched
- Packet-Switched

Dedicated Connections

A dedicated link is a single point-to-point connection between two facilities (see Figure 1.2) that is leased from a service provider. A permanent connection is made through the carrier's network for the sole use of the customer. Since multiple connections cannot be made, a separate dedicated connection is needed for every facility to which the customer will connect. This can raise costs to inefficient levels if more than a few connections are needed. The pricing is based on speed (for example, Fractional T1 line, T1 line, or T3 line) and the distance between two sites.

The main benefit of a private line is the 24 hours per day, seven days per week availability of large amounts of bandwidth. Speeds up to 45 Mbps can be reached through a T3 line in North America and Japan, and up to 30 Mbps, in Europe with an E3 line. Because the link is not shared with other companies, the full bandwidth is always available to the customer.

Figure 1.2 Dedicated T1 line between two sites.



The flip side to not sharing the bandwidth with other customers is not sharing the price with them. In shared connections such as Frame Relay, the price is distributed among multiple companies. However, these prices vary by provider; dedicated links may still be cheaper if a small number of connections is needed.

Permanent connections are typically available in a range of speeds, including 56 Kbps, 64 Kbps, 1.5 Mbps, 2 Mbps, 30 Mbps, and 45 Mbps. Many providers also offer fractional connections to supply a portion of the speed available on a single line. These kinds of connections are usually employed with high-speed, dedicated Internet connections where a full T1 line may not be needed.

Dedicated lines normally connect through a CSU/DSU, which is available as a built-in or separate option. If the CSU/DSU is not integrated, another connection is made between the unit and the router's synchronous serial interface. A DSU converts the signal from the router's serial port to a WAN format that the CSU can use to connect with the interface of data circuit-terminating equipment (DCE), such as a switch. It also provides synchronization between the two devices, and can echo loopback signals from the phone company for line testing.

A DSU connects to the serial port using an industry standard format. Cisco routers support the following:

- X.21
- V.35
- EIA/TIA-232
- EIA/TIA-449
- EIA/TIA-530

Dedicated connections provide different advantages and disadvantages, which are displayed in Table 1.1.

Table 1.1 Advantages and Disadvantages of Dedicated Connections

Advantages	Disadvantages
Longer connection times (always up)	High cost
Maximum availability of bandwidth	Connection to only one site
High-speed capabilities	

For Managers**Wireless Options**

Other options for dedicated connections are becoming available through new breakthroughs in wireless technologies. Cisco's new Aironet series of wireless bridges can establish connections over 20 miles in distance and up to 11 Mbps in speed. This provides an opportunity not only for campus networks, but also for metropolitan area networks to purchase high-speed connections and eliminate local loop charges included in wired solutions. This also lowers the total cost of ownership and the strain dedicated links put on an IT budget.

More information on Cisco's wireless technologies can be found at:

www.cisco.com/warp/public/44/jump/wireless.shtml

Circuit-Switched Connections

In a circuit-switched connection, a dedicated path is established over a telephone company's network when a call is placed, and then terminated at the end of each session. People use circuit-switched connections whenever they place a call to another person. The link is brought up only when needed, and is used exclusively by the two connected parties. A new connection is created for every voice, fax, or data connection required.

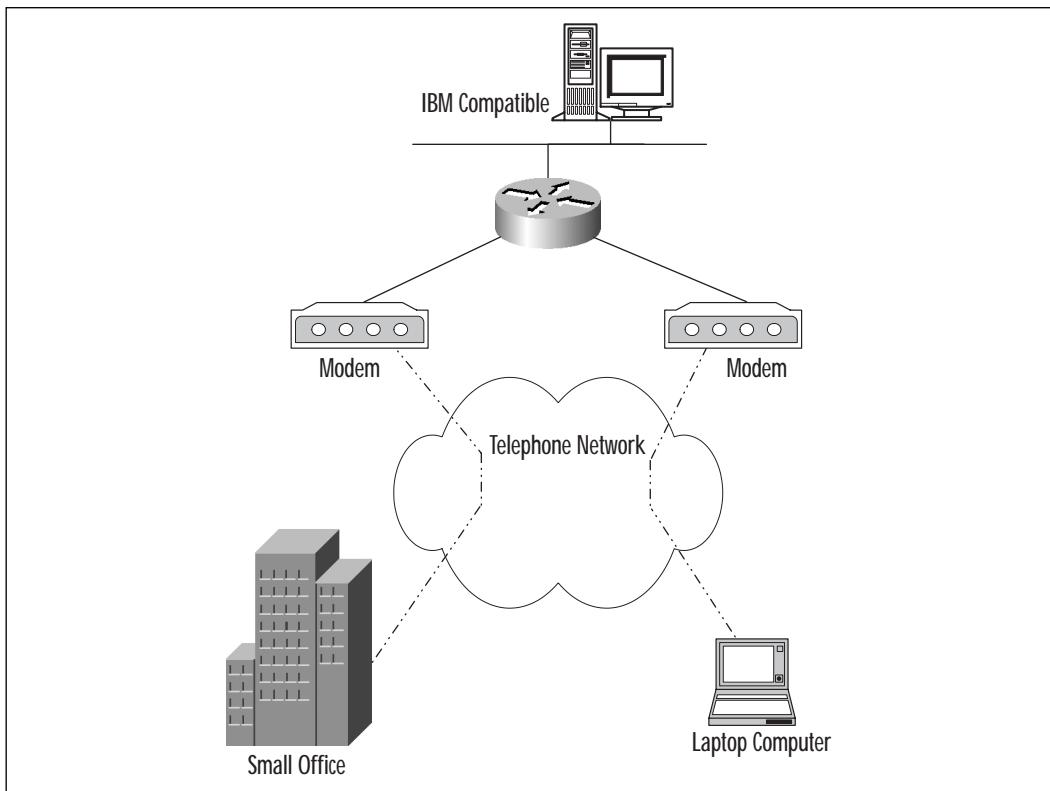
Analog Modem Connections

Analog modem connections are circuit-switched solutions (as illustrated in Figure 1.3) that use a modem and the asynchronous serial port of a router to create a dedicated connection on demand. They are typically used for low bandwidth activities such as a mobile user checking e-mail, as a backup for high-speed links, or when a remote area does not have any high-speed technologies available.

The modem is used to convert the digital signal from the router to the analog signal needed to traverse the network used by the telephone company. A path is created through the carrier's equipment and is received by the modem at the other end, where the signal is converted back to a digital format. Modem speeds range up to 56 Kbps (actually limited by the FCC at 53 Kbps).

Analog signals are an older technology and can be noisy and prone to error. They were originally designed for voice communications and are generally inefficient for use with sensitive data.

Figure 1.3 Asynchronous connections.



Dial-on-demand routing (DDR) is used to enable a router to make a connection whenever the exchange of data is needed. Access control lists (ACL) describe what is called *interesting traffic*; when data is present that meets the requirements of the ACL, a connection is established and terminated after a set period of inactivity. This keeps local traffic and routing updates from making unnecessary connections.

DDR requires that you either use static routes or an infrequent method of transferring updates, such as snapshot routing. This is explained in more detail in Chapter 5.

ISDN Connections

Integrated Services Digital Network (ISDN) is also a circuit-switched network that provides higher speeds (128 Kbps–1.5 Mbps) than asynchronous. ISDN is an all-digital format with the capability of carrying data, voice, and video. Without the need to convert to an analog signal, ISDN presents an efficient, reliable method of transport.

The two types of ISDN are:

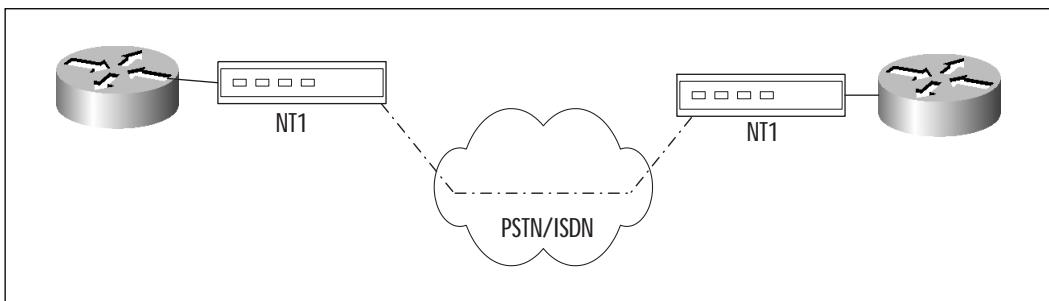
- Basic Rate Interface (BRI)
- Primary Rate Interface (PRI)

BRI (also known as 2B+D) uses three channels: two “B” channels operating at 64 Kbps and one “D” channel operating at 16 Kbps. The B channels are used for the transfer of data, voice, and video and can achieve a combined speed of 128 Kbps. The D channel is used for call setup and call teardown. It uses a data-link layer protocol called Link Access Procedure on the D channel (LAPD).

The requirements for a BRI connection are a BRI interface on the router and an ISDN terminal adapter. A terminal adapter is the equivalent of an analog modem for asynchronous serial ports. The NT1 line (adapter for BRI) is usually supplied by the customer in the United States, and by the service provider in Europe. Some routers come with an NT1 line integrated into the interface (called a “U” interface).

Placement of the equipment is shown in Figure 1.4.

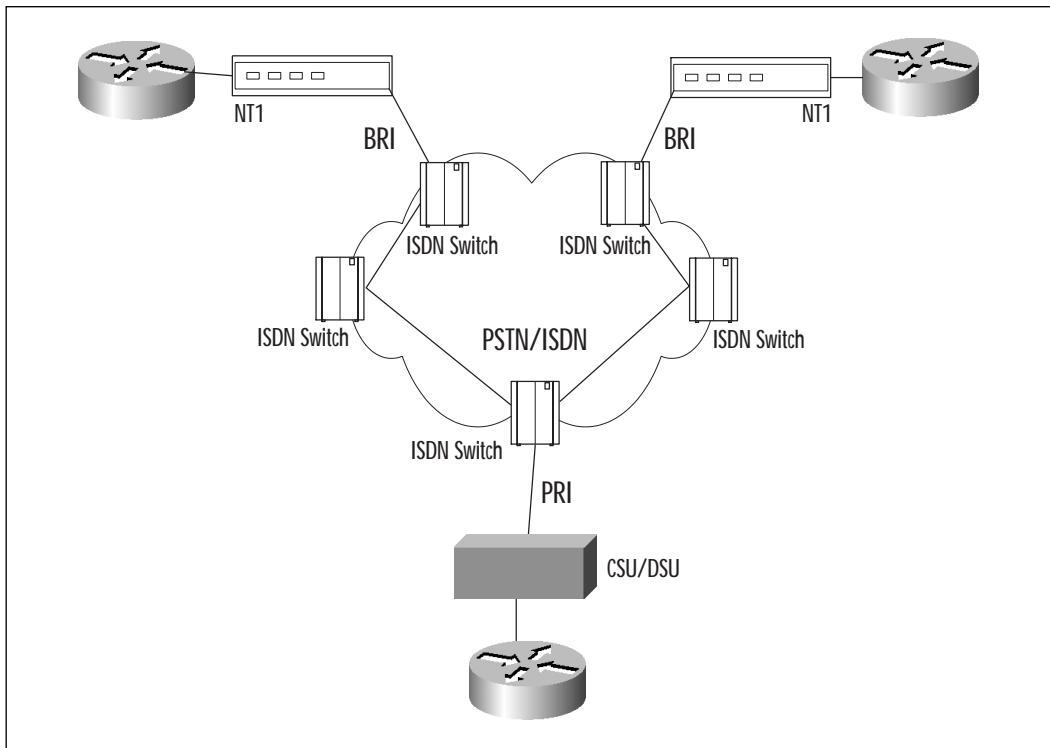
Figure 1.4 BRI network and placement of equipment.



PRI (also known as 23B+D) uses a total of 24 channels: 23 B channels operating at 64 Kbps and 1 D channel operating at 64 Kbps. PRI usually operates over T1 line technology and can achieve a maximum bandwidth of 1.544 Mbps. The D channel is used to set up the transfer of voice, video, and data over B channels. A PRI line uses a CSU/DSU (see Figure 1.5) and can handle multiple BRI calls.

Like asynchronous connections, ISDN can also use the functionality of DDR to control the likelihood of a connection.

One problem with circuit-switched networks is that every established connection dedicates the entire bandwidth, even when idle, to the customer who made the call. This is an inefficient use of the service provider's

Figure 1.5 ISDN network with PRI and BRI.

channels, which could be used to carry multiple streams of traffic (as in packet switching) from different customers all at the same time.

Table 1.2 lists the advantages and disadvantages of using a circuit-switched network.

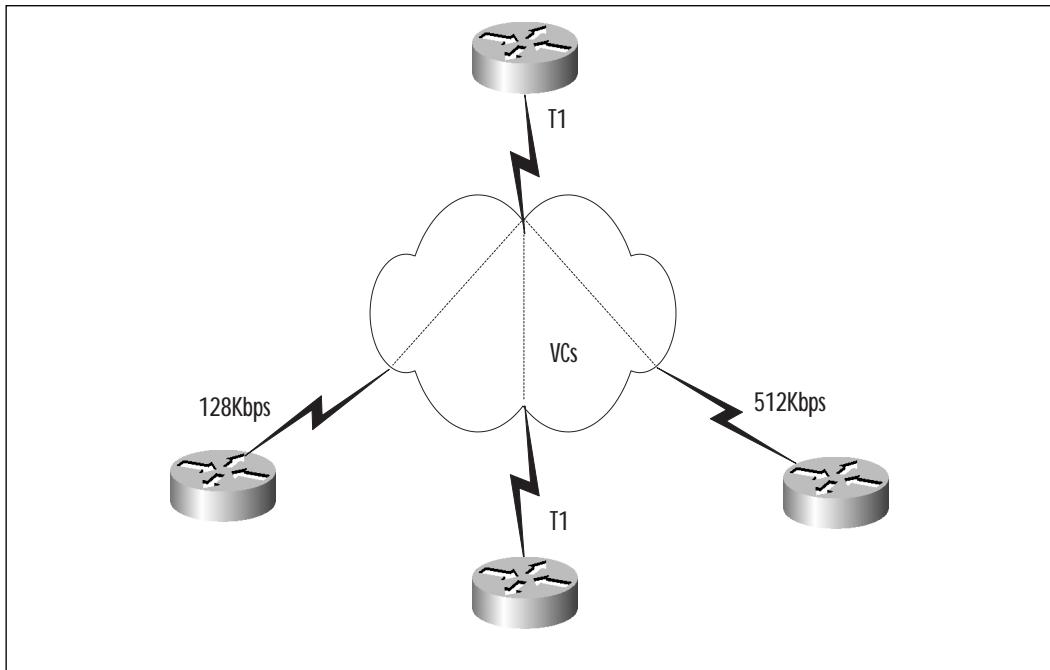
Table 1.2 Advantages and Disadvantages of a Circuit-Switched Network

Advantages	Disadvantages
Makes connections only when needed	Low speeds
Bandwidth is dedicated	Analog connections are noisy, prone to errors
Generally cheaper than dedicated lines	Inefficient use of provider's channels
Good backup for high-speed lines	

Packet-Switched Connections

Packet-switched networks, as shown in Figure 1.6, create point-to-point connections between sites using virtual circuits (VCs) to establish connectivity. These VCs can be permanent virtual circuits (PVCs), where a permanent path is configured to carry traffic to a destination, or switched virtual circuits (SVCs), which dynamically create a path when a connection is required.

Figure 1.6 Packet-switched network.



Benefits such as high speeds (for example, E3 line, T3 line) and the ability to transfer data, voice, and video, make packet switching an attractive option. The cost is also generally cheaper than dedicated lines, as multiple customers share the bandwidth.

Frame Relay is the most popular form of packet switching, and is best suited for WANs that require many connections with varying amounts of bandwidth. Characteristics of this type of network are listed in Table 1.3.

Table 1.3 Advantages and Disadvantages of a Packet-Switched Network

Advantages	Disadvantages
Generally cheaper than dedicated connections	Bandwidth is shared
High-speed connections	More complex than dedicated connections
Efficient use of carrier's bandwidth	
Carry voice, video, and data	

WAN Encapsulation Protocols

All routers must use some form of encapsulation when sending traffic across WAN connections. The type of encapsulation used depends on the modules or built-in interfaces on your router, the type of WAN technology chosen to transport the information, and the commands used to configure the interface.

Cisco routers support many encapsulation types, including:

- Synchronous Data Link Control (SDLC)
- High-Level Data Link Control (HDLC)
- Serial Line Internet Protocol (SLIP)
- Point-to-Point Protocol (PPP)
- X.25
- Frame Relay
- Asynchronous Transfer Mode (ATM)

SDLC

IBM originally developed SDLC in the mid-1970s for use with the Systems Network Architecture (SNA) protocol. A bit-oriented synchronous protocol that is the predecessor of HDLC, you generally will not find it in wide use.

HDLC

The HDLC protocol is the default encapsulation set on Cisco synchronous serial interfaces. It is used extensively for point-to-point and point-to-multipoint connections. HDLC comes from modifications done to SDLC by the International Standardization Organization (ISO).

The problem with an HDLC implementation is that the technology is proprietary to each vendor—Cisco’s version of HDLC will not communicate with another manufacturer’s version.

SLIP

SLIP is used for point-to-point Transmission Control Protocol/Internet Protocol (TCP/IP) connections. It is the predecessor to PPP and is no longer in wide use.

PPP

PPP is an industry standard protocol that can be used to make connections on various vendors’ equipment using multiple protocols. This is an improvement over SLIP, since it could only encapsulate TCP/IP, and HDLC, which could only communicate when the same brand of equipment was used on both ends. PPP can also be used over asynchronous or synchronous connections, and supports many features including:

- Encapsulation of multiple protocols
- Authentication using the Password Authentication Protocol (PAP) or Challenge Handshake Authentication Protocol (CHAP)
- Compression using Predictor or Stacker
- Multilink

The Network Control Protocol (NCP), a major component of PPP, is a family of protocols used to encapsulate the different Open System Interconnection (OSI) Layer 3 protocols supported by the Point-to-Point Protocol. IP, Internetwork Packet Exchange (IPX), AppleTalk, DECnet, and ISO Connectionless Network Service (CLNS) are the protocols that are currently supported.

Another component of PPP is the Link Control Protocol (LCP), which is used to set up and maintain connections. Authentication and compression are features of the LCP and are discussed in Chapter 3.

Multilink PPP allows multiple connections over the same interface in ISDN scenarios, or allows a group of dialup interfaces to operate as a single logical interface. This significantly increases the amount of bandwidth available. This concept is also discussed in Chapter 3.

X.25

X.25 is a packet-switching protocol designed for the exchange of data over a WAN. It is a predecessor of Frame Relay, containing support for error detection and correction, and was designed to transport packets over very

noisy, low-speed, analog lines. High overhead and more modern technology make this a poor choice for today's networks.

Frame Relay

Frame Relay is also an industry standard packet-switching solution for WAN connections. It is a Layer 2 encapsulation that relies on upper layers to provide error checking, increasing performance of the links. Low overhead and high speeds have made it a very popular style of encapsulation. See Chapter 7 for details on configuration.

ATM

ATM is a dedicated-connection switching technology that organizes digital data into 53-byte cells and transmits them over a physical medium using digital signal technology. Cells are actually 48 bytes and contain a 5-byte header. The fixed-length cells enable switching to occur at the hardware level, thereby increasing efficiency.

ATM networks can take advantage of high-speed technologies, including Synchronous Optical Network (SONET), and reach speeds of 10 Gbps (gigabits per second), making it an attractive option for demanding applications such as video conferencing and high-speed backbones.

The type of encapsulation used on an interface can be found by either viewing the running configuration or using the **show interface** command. The following is a sample output of this command:

```
PERO002#sh int s0/1
Serial0/1 is up, line protocol is up
  Hardware is PowerQUICC Serial
  Internet address is 206.57.5.6/30
  MTU 1500 bytes, BW 512 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation PPP, loopback not set
  Keepalive set (10 sec)
  LCP Open
  Listen: CDP/CP
  Open: IPCP
  Last input 00:00:00, output 00:00:00, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0 (size/max/drops); Total output drops: 500
  Queueing strategy: weighted fair
```

```
Output queue: 0/1000/64/500 (size/max total/threshold/drops)
  Conversations 0/22/256 (active/max active/max total)
  Reserved Conversations 0/0 (allocated/max allocated)
5 minute input rate 1000 bits/sec, 2 packets/sec
5 minute output rate 3000 bits/sec, 4 packets/sec
  10071029 packets input, 4064842154 bytes, 0 no buffer
  Received 1955569 broadcasts, 0 runts, 0 giants, 0 throttles
  3 input errors, 0 CRC, 3 frame, 0 overrun, 0 ignored, 0 abort
  11823665 packets output, 2032850506 bytes, 0 underruns
  0 output errors, 0 collisions, 7 interface resets
  0 output buffer failures, 0 output buffers swapped out
  0 carrier transitions
  DCD=up  DSR=up  DTR=up  RTS=up  CTS=up
```

Note how the encapsulation is listed as PPP in the seventh line.

Selecting Cisco Access Servers and Routers

So, now you know a little bit about WAN technologies. Now it's time to consider which Cisco products deliver the best solution for your needs. What types of interfaces do you need? How many interfaces do you need? The following is a breakdown of some of the Cisco line of routers and access servers. Up-to-date information can be found at:

www.cisco.com/public/products_prod.shtml

700 Series

The 700 series routers are used in a SOHO for ISDN connections, and come with a scaled-down version of the Cisco IOS. They are available in a variety of options that let you decide whether you want a built-in NT1 line, a standard BRI port, analog ports, one or more 10BaseT Ethernet ports, and support for up to thirty users.

800 Series

The 800 series is the lowest model that supports a full version of the IOS and is suitable for SOHO and telecommuters. It comes with support for ISDN, ISDN digital subscriber line (IDSL), asymmetric digital subscriber

line (ADSL), Smart Serial port for synchronous or asynchronous dialup, analog ports, and Ethernet.

900 Series

The 900 series is a Cisco cable modem/router line of products used for home and SOHO environments. Cable modems are a relatively new technology that use the cable provided by cable television companies for high-speed Internet connections. They are sometimes used in conjunction with a dial-up connection for upstream traffic.

The 900 series supports four Ethernet ports, but cable companies usually regulate how many connections are allowed.

1000 Series

This is a series of compact, fixed-configuration routers used to connect SOHO and remote office locations through ISDN and asynchronous or synchronous serial connections. They also support Ethernet connections and a Personal Computer Memory Card International Association (PCMCIA) slot for flash memory cards.

1400 Series

The 1400 series supports ADSL for high-speed, always-on Internet connections with downstream speeds up to 8 Mbps. Two models support either an ATM25 interface (needs external ADSL modem) or a built-in ADSL modem. The flash memory is stored on a flash PC card.

These ADSL routers also support VPN technologies that bring new options for connecting a corporate WAN.

1600 Series

The 1600 series provides the first look at modular routers. They provide several fixed configurations, including support for ISDN, ISDN phones, serial with integrated 56 Kbps DSU/CSU, Ethernet ports, and a WAN interface card slot.

The WAN interface cards (WIC) supported in the 1600 series are asynchronous and synchronous serial, T1 line/Fractional T1 line CSU/DSU, 56/64 Kbps four-wire CSU/DSU, ISDN BRI with S/T interface (dial and leased line), ISDN BRI with integrated NT1 line, U interface (dial and leased line), and ISDN BRI leased line (S/T interface). These cards are also interchangeable with the 1700, 2600, and 3600 series modular routers.

1700 Series

The 1700 series routers expand upon the 1600 series and continue with a modular design, allowing for two WAN interface cards shared with the 1600, 2600, and 3600 series routers, and a 10/100 Ethernet port.

2500 Series

The Cisco 2500 series routers provide a variety of models designed for branch office and remote site environments. They typically are a fixed configuration, although two models are modular, with at least two interfaces including Ethernet (AUI), Ethernet Hub, Token Ring, synchronous serial, asynchronous serial, and ISDN BRI.

The modular units do not share their cards with any other series of routers, making them less attractive than the newer 2600 series.

2600 Series

This series provides a more powerful and adaptable option for branch offices, featuring one or two fixed-LAN interfaces, a network module slot, and two WAN interface card slots. The 2600 series also supports voice modules inter-virtual LAN (VLAN) routing on the 10/100 Ethernet models. An internal Advanced Integration Module (AIM) is also included for such applications as hardware accelerated compression.

Most modules for this series can also be shared with the 1600, 1700, and 3600 series routers.

3000 VPN Concentrators

The Cisco VPN 3000 Concentrator series is a newer, remote-access VPN solution for enterprise networking. It includes a VPN client, scalable VPN tunnel termination devices, interfaces supporting up to full E3/T3 lines, and encryption throughput of up to 100 Mbps.

This series features support for a wide range of VPN client software implementations, including the Cisco VPN 3000 Client, the Microsoft Windows 2000 L2TP/IPSec Client, and the Microsoft Point-to-Point Tunneling Protocol (PPTP) for Windows 95/98, and Windows NT.

3600 Series

The 3600 series of high density, modular access servers/routers is good for branch office/central office implementations. With up to six interface slots, support for voice, and interface speeds up to OC-3 (155 Mbps), this series provides a cheaper alternative for smaller companies that do not require the power of a 7000 series router. Two internal AIM slots are also included,

as well as the option for one or two fixed Fast Ethernet ports that support inter-VLAN routing.

Again, most of the modules are shared with the 1600, 1700, and 2600 series routers.

AS5000 Series

This line of universal integrated access servers combines the functions of stand-alone CPUs, modems, communications servers, switches, and routers all in one chassis, suitable for implementation from a central office to ISP. The modular design supports three and 14 slots on the AS5300 and AS5800, respectively, with modules that support up to 12 T1/E1/PRI interfaces, two channelized T3 lines, and 144 modems per card.

7100, 7200, and 7500 Series

The Cisco 7100, 7200, and 7500 series routers are Cisco's premier high-end platform of data, voice, and video routers. These high-density, modular routers can take the heaviest amounts of traffic with a high throughput, making them perfect for core backbone environments. Reliability is also addressed with optional, redundant power supplies. These series also support any combination of ATM, channelized T3 line, Ethernet, Fast Ethernet, Fiber Distributed Data Interface (FDDI), IBM channel attachment, multichannel E1 line and T1 line, High-Speed Serial Interface (HSSI), synchronous serial, Token Ring, Packet OC-3, Gigabit Ethernet interfaces, and multiple routers within a single chassis.

Considerations Before Installing a Remote Access Network

Many issues must be resolved before a new network is set up or additional access is added to an existing network. Careful planning is needed to discover what the actual needs of the network are, what vendors to use, and how different configurations will affect your current design.

An IS staff cannot add new components to a network without first discovering the ramifications of adding any new equipment—that's assuming your own IS staff will be doing the installation.

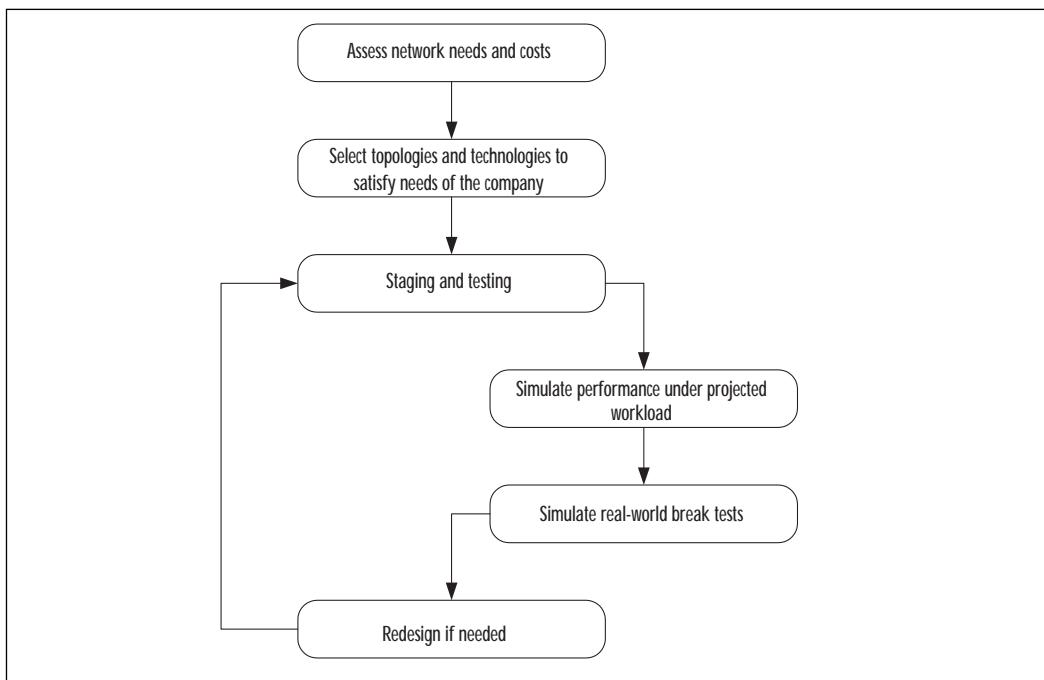
The following section describes basic network planning and a simple process for acquiring and implementing new equipment.

Network Planning and Design

Although this is not a network design book, basic design knowledge is needed when deciding which solutions are best for a company. What kind of applications will you be running? What routing protocols are you considering and will they run on all of your equipment? Are there any budget constraints, and how will this benefit the company? What are the current trends in networking?

These questions and others can all be addressed using a basic network design process, as shown in Figure 1.7:

Figure 1.7 Network design process.



Proper Analysis

A network designer needs to provide a full analysis to assist in providing information and business justification to get a project approved. This includes forecasting potential impacts on the existing network.

Network Needs

One of the most challenging tasks of a design analysis is discovering the actual needs of the network for whatever goal the new design is to accom-

plish. In order to properly assess the current and future needs of the network, a few questions must be asked:

- What applications are used now and which are being considered for the future?
- What protocols are running on the network?
- What equipment from which manufacturer(s) is in use?
- How will the new equipment impact the current network?
- How important is redundancy?
- What kind of latency is acceptable?
- How many users do you need to support?
- What kind of user growth is projected in the next year? Three years? Five years?

The existing network must always be considered; the last thing an IS staff needs is a new configuration causing havoc to mission-critical applications.

Conducting interviews, discussion groups, and surveys are also good ways of compiling the needs of the users themselves, although you will usually find that the greatest needs are response time, throughput, and reliability.

Time Frame

Most companies have a time frame for new projects. Imagine that for the last five years you've been connecting to your biggest customer through a modem—dialing into their internal system to verify shipments and payments, and to plan the future quantities of your product. One day this customer, without whom your business would fail, decides they are going to build a private, high-speed network with encryption, with a new web-based program. Then you are notified that you have six months to make the necessary modifications to your network in order to do business.

The implications of your time frame will help decide what kind of labor you will need, whether it's internal and/or outsourced, and how much time can be devoted to planning. It will also determine whether you can schedule all updates around normal user hours or if time must be set aside for network interruptions.

Cost

Costs must be kept as low as possible without compromising network performance. Narrow down the choices of the types of connections that will accomplish your goals based on what you gathered in discovering your

network needs, and then contact available service providers to select the best price-per-performance solution. Remember, leased lines are generally more expensive, but per-minute services like ISDN can add up if long connect times are required.

Costs also include any outsourced labor or consulting, training for internal staff, or the addition of staff to help maintain large installations.

Resources

Your resources include labor (both contracted and internal employees), the resellers from whom you purchase the equipment, and colleagues. Don't be afraid to ask professional colleagues for information on their experiences with different products, consulting firms, and service providers. Most will give an unbiased viewpoint, unlike the companies with which you are dealing; they tend to have partnerships with specific vendors that get in the way.

You can also gain a wealth of information from consultants that recommend a variety of vendors and solutions. They have usually seen more products in action than have professionals who work in a closed environment.

Training

Training becomes an issue when maintenance and configuration of the new equipment will not be outsourced. The price of training and lost time must also be figured into your costs (training is usually done on work time).

Installation Plan

All installations, whether large or small, need an installation plan. Every step of the installation is documented and fit it into a time line with delegation of all duties. This provides a view of all that needs to be done, preventing the problems that come from lack of planning; many times conclusions derived from previous steps in the design process are changed after creating an installation plan.

Business Justification

Management and financial controllers must approve any new purchases and costs incurred by new implementations, a process that can be difficult and time-consuming. Sometimes, the whole time frame is thrown off by the length of time it takes to get a project approved. That is why it is important to use every resource available to justify the cost, and to get approval in a timely manner.

Ask yourself the following questions:

- Will this improve the efficiency of the workforce?
- Will this improve the operations of the business?

- Will this improve network downtime?
- How will this benefit your customers?
- What is the total cost of ownership?
- Do you have competitive bids?
- What other companies are doing this?

Addressing all of these questions gives you the resources to write effective proposals and gain the approval of the necessary parties. They may also lead you to reassess the vendors and service providers originally chosen for the project.

Identifying Suitable Equipment for Each Site

The choice of topologies and equipment is determined by the location and where it stands in the company hierarchy. A few industry-standard terms are used to describe site structure and the relative demands of each location. Figure 1.8 provides a theoretical example of company sites and the connections used for each.

Central Site

The central site is the main location to which most other offices connect for retrieving information and data. This may be the corporate headquarters or another location where enterprise servers and resources are located, and it must be able to scale to the demands of a growing WAN and multiple types of connections.

New trends in networking to central sites include utilizing high-speed Internet technologies and VPNs to create cheaper, secure connections.

Branch Office

Often referred to as a remote office, the branch office is a regional office employing more than a few users. Branch offices generally require high-speed connections due to their large size and their support of regional SOHOs and mobile users.

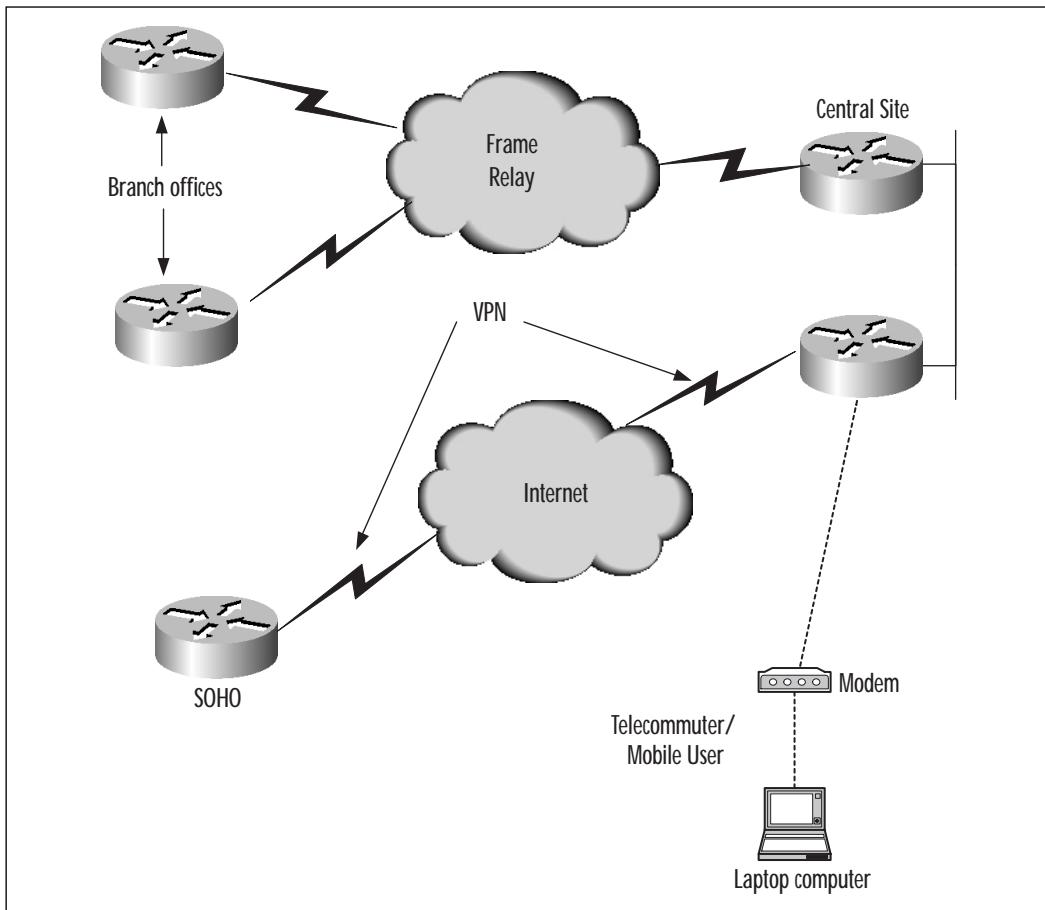
Small Office/Home Office (SOHO)

A SOHO refers to an office consisting of one to a few users or an employee who does a large percentage of work from home. Slower connections are suitable for these locations. A VPN, using a local Internet connection, is a solution that is gaining popularity with SOHOs in locations that are not within the local toll-free area.

Telecommuter/Mobile User

Mobile users are usually employees in sales departments who travel and need access only to small amounts of information such as e-mail.

Figure 1.8 Example of company sites.



Telecommuters may also be single employees in remote locations (for example, field engineers) who may need faster access from home.

Table 1.4 describes the router platforms discussed previously in this chapter, and their typical implementation regarding the type of site in which they are best suited.

Table 1.4 Router Platforms and Configurations

Router Series	Configurations	Best Implementation
700	ISDN BRI, analog telephone ports, scaled-down IOS	Telecommuter, SOHO
800	XDSL, ISDN BRI, Smart Serial, analog telephone ports	Telecommuter, SOHO
900	Cable modem	Internet Solutions
1000	ISDN BRI, serial	SOHO
1400	ADSL	Internet Solutions
1600	ISDN BRI, 1 WIC slot	Branch Office
1700	2 WIC slots	Branch Office
2500	Various fixed configurations—ISDN BRI, Async and Sync serial, Ethernet, Token Ring, WAN modules	Branch Office
2600	1 Network Module Slot, 2 WIC slots, various fixed LAN ports, voice support	Branch Office/Central Site
3000VPN	Up to T3/E3, Various VPN clients	Enterprise VPN Solution
3600	Up to 6 module slots, various fixed LAN ports, voice support	Branch Office/Central Site
AS5000	Access servers with up to 14 slots	Central Site
7100-7500	High density routers with a wide variety of interfaces	Central Site

Staging and Testing

Building a test lab provides the benefits of addressing configuration, performance, and conflicts before the project goes live. Building a similar environment with actual users to test the implementation is invaluable in making a smooth transition to the new equipment, and often uncovers issues that are far better resolved before the equipment is in use.

It is important to use a sampling of real users in your tests. They usually provide good questions, concerns, and procedures that are often overlooked by an IS department. Their input on items like acceptable latency also helps in planning future projects.

Make sure that anything found in the staging and testing phase of the design process is documented for future use. This prevents valuable time and resources from being used fixing reoccurring problems.

Sometimes recreating a close environment is too difficult or expensive. Cisco has tools that can help in this kind of situation. The NetSys program simulates network configurations and their effects in an environment. This allows you to test an implementation before it goes live.

Remote Access Network Implementation Considerations

Once you have carefully executed your design and planning procedures, make sure your implementation process is also planned carefully. The benefits involved include fully documented changes, a backout plan, minimal user disruptions, efficient coordination of resources, and smooth troubleshooting.

Change Control Procedures

Change control is a mechanism for tracking all changes, reasons for changes, and the obtaining of authority for changes. It provides accountability and the information necessary for reversing any changes—often called a “backout plan.” This is done through documentation of proposals and their approvals, installation plans and procedures, and the tracking activities of your labor force.

Accountability becomes a factor when a problem occurs, but not for putting the blame on another employee. It simply eases the task of tracking down what changes were made by whom. Problems are much easier to solve when you know what recent changes have been made.

Backout Plans

All installations require backout plans in case anything goes wrong in the implementation. More than one network administrator has escaped a lashing from coworkers with a few simple practices. When replacing old hardware or connections, never discard them until the new equipment has been working properly for a reasonable amount of time. Use them as backup links in case the new connection goes down or needs to be taken down for changes.

Strict documentation can be a tiresome activity, but is invaluable when making changes to configurations. The ability to trace all changes to the router makes backing out configuration lines a breeze. Be sure to observe the effects of each configuration change before proceeding to other changes. Adding additional variables just makes troubleshooting a nightmare.

Minimizing Network Interruption

It is extremely important that any new installations minimize interruptions to normal daily operation of the network. Plan on spending nights and weekends, or at least off-peak times (for example, lunch), implementing the project. Any planned outages or interruptions should be advertised well in advance in order to prevent user problems and disruptions to the normal operation of the business.

Coordination of Resources

Use the established time frame and installation plan to help coordinate the activities of external consultants, telephone companies, and resellers.

Make sure the equipment you are purchasing is not back-ordered and will arrive by a set date. This allows easier scheduling of service providers, who can often take extended periods of time before assisting in any new project.

Consulting firms are generally easier to schedule around the time frame created by the telephone company and arrival of new equipment.

Verifying and Troubleshooting Network Installation

The final steps to any project involve making sure everything is operating the way it's supposed to. Use **ping**, **traceroute**, and **show interface** commands to verify connectivity to remote sites. Check routing tables, neighbor commands, and configurations to assist in tracking down problems.

Another item that is often overlooked is simply checking the LEDs on routers and modules. These are always a quick, sure way of narrowing down connection problems. This gives you an overview of all the equipment and which ports are not active or are having some kind of problem. Look for activity LEDs and connection indicators that signify whether a link is up and is receiving any information.

In-depth technical troubleshooting will be covered in the upcoming chapters.

Summary

In this chapter you have formed a solid foundation on WAN technologies that will facilitate your understanding of advanced topics found later in this book. You've learned what types of topologies are best for different kinds of sites and how to take a new project from start to finish. You have also established a familiarity with the Cisco line of products, which is a good starting point for choosing the proper equipment.

WAN links connect facilities over large geographical distances and are usually leased from a service provider. The types of lines available vary by region and carriers.

Present and future bandwidth requirements should be considered when planning the type of technologies used to connect sites. The types of WAN links are:

- Dedicated connections
- Circuit-switched
- Packet-switched

All links require a type of adapter, which is either built into the router interface or purchased separately. Sometimes the service provider will supply the adapter or lease the equipment.

Keep the type of site you have in mind when choosing routing equipment. Choose models that satisfy current requirements and can scale to future demands. Provide a cost/benefit analysis with competitive quotes to speed approval.

Discover all of the needs of the network before creating an installation plan. The installation plan should detail every procedure and coincide with a time line. Record all configurations and changes you have made to ease troubleshooting.

FAQs

Q: Who can I contact about more details on Cisco equipment?

A: Contact your local or regional Cisco partner or reseller. You can locate them on the Cisco Web site under the “How to Buy” section, or you can use the Web site itself to gather more information on product lines.

Q: How do I know which service providers to use?

A: Get competitive bids from each provider and references from their customers. Contact the references to see how their experience has been. Talk to colleagues or friends in the field. Do not always take the cheapest provider, as they may not offer the best overall service.

Q: Where can I get more information on VPNs?

A: Chapter 4 explains this technology in more depth. Also check with your regional Cisco representative for seminars.

Q: In the first section, you mentioned using my network to transport voice traffic. Where can I get more information on this technology?

A: Try *Configuring Cisco Voice Over IP* by Syngress Media. A description and sample from the book can be found at www.syngress.com/marketing/cisco.htm

Q: What are the reasons for buying modules with, or without, a built-in CSU/DSU?

A: Your provider may supply you with this equipment. If this is the case, it is unnecessary to purchase a module with a built-in CSU/DSU.

Q: Can the service provider supply the whole unit, not just the CSU/DSU?

A: Some may include the router in the deal, or allow you to lease it in addition to the line.

Configuring Asynchronous Remote Access Connections

Solutions in this chapter:

- Modem overview
- Configuring asynchronous connections with modems
- Providing asynchronous dial-in terminal services

Introduction

Having identified your communications requirements and selected the equipment, let's now look at how to establish connections from a home user, telecommuter, or dial-up client to a central site using asynchronous communications.

First let's review modem technologies and then look at how to configure modems attached to access servers to permit asynchronous connectivity. You will learn how to use reverse Telnet to connect to the modem for manual configuration, and will also learn how automatic configuration and modem discovery work.

The final section of this chapter will explain how to provide terminal services on the access server to permit access to legacy equipment. Although more and more access requirements are for PPP network connections, there are still times when the provision of asynchronous terminal services can be of value. The next chapter will show how you can use the same interface to provide both terminal services and PPP access by the use of the **autoselect** command.

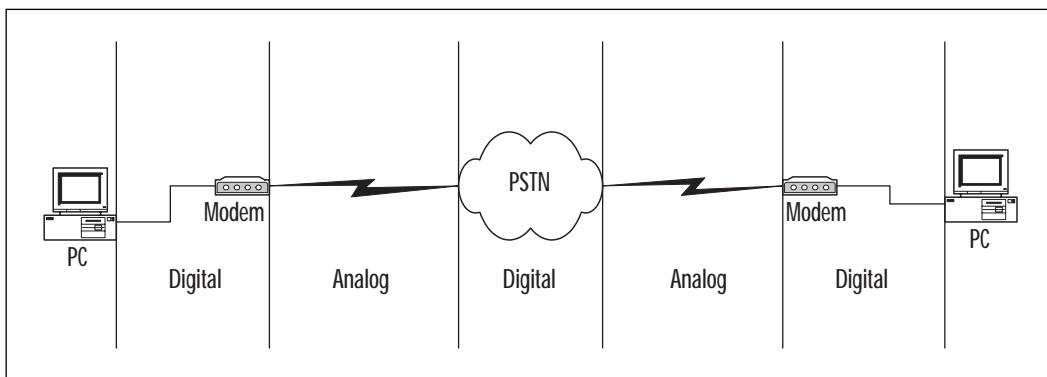
Modem Overview

A modem is a common communications device that almost everyone with a PC has used. You might use a modem to dial up from your home computer to the Internet, or into the office for remote networking services. But what does a modem really do? *Modem* is an abbreviation for modulator-demodulator, and it refers to a device that allows digital signals to be carried over an analog network. So when you dial into the Internet, your PC is sending digital signals that the modem translates into analog signals that are carried across the Public Switched Telephone Network (PSTN). The modem on the other end of the call demodulates the analog signals and converts them back to digital signals.

From this example, it would appear that the communication through the PSTN is purely analog, but that is not the case. The PSTN was originally designed to provide end-to-end analog communications to carry the human voice. However, as the popularity of the telephone grew, the number of lines required to support its widespread use became cumbersome. In the 1950s, AT&T started looking toward digital communications to streamline the PSTN. This streamlining allowed for faster connections and better voice quality, and offered a whole new range of services. Today, the majority of the PSTN is based on digital communications, although the local loop is still predominately analog.

To convert the analog signals coming from your home telephone to a digital format that is transportable over the PSTN, a technology called Pulse Code Modulation (PCM) was created. PCM is the method by which the human voice, or any analog signal for that matter, is digitized. To properly digitize the voice, it is sampled 8000 times per second. This number is based on Harry Nyquist's Sampling Theorem, which shows that to be able to accurately reproduce an analog signal from a series of samples, sampling must occur at twice the highest frequency of the signal. The maximum frequency a local loop will carry is 4MHz and requires a sample rate of 8000 times per second, or a sample interval of 125 microseconds. Each sample is converted into a digital bit stream through PCM (see Figure 2.1).

Figure 2.1 PCM Diagram.

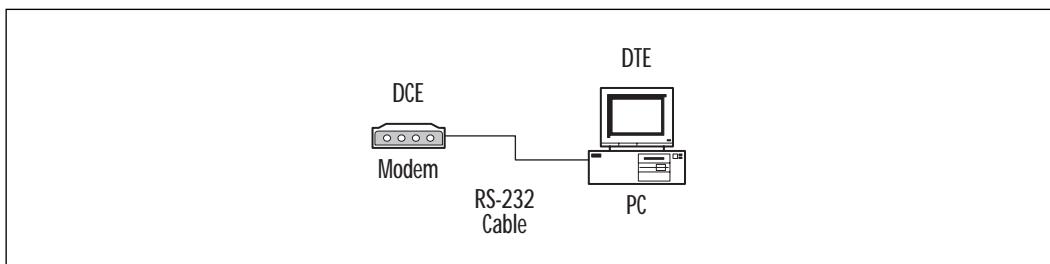


There are many types of interfaces available when working with modems: EIA/TIA-232, EIA/TIA-449, V.35, High-Speed Serial Interface (HSSI), X.21, and others. These specifications define the physical layer of communication used on the cable. In the Open System Interconnection (OSI) model, Layer 1 (the physical layer) is responsible for the electronic and mechanical characteristics of the connection. The application using the modem, as well as the speed of the modem, will dictate the interface required. For example, you wouldn't use a V.35 cable to connect your new modem to your PC for Internet dial-up access. Most PCs do not have an interface built into them that allows for V.35 communications; however most PCs do have EIA/TIA-232 interfaces.

Devices communicating through serial communications can be divided into two categories: Data Communications Equipment (DCE) and Data Terminal Equipment (DTE). DCE refers to equipment such as the modem and channel service unit/data service unit (CSU/DSU) that interface with the PSTN. DTE refers to the device that connects to the DCE. In a simple

example, a PC with a modem connected to an EIA/TIA-232 port can be broken down into the two categories. The PC is DTE and the modem is DCE, as illustrated in Figure 2.2.

Figure 2.2 DCE and DTE.



Digital Modems

Digital modems are similar in configuration and functionality to the standard analog modems; they differ in that digital modems use digital lines, not analog phone lines. Typically, digital modems are connected to Integrated Services Digital Network (ISDN) circuits such as Basic Rate Interface (BRI) and Primary Rate Interface (PRI). Since digital modems do not connect to analog lines, they are not required to do the analog-to-digital conversion that a standard modem does. This absence of signal conversion—as well as the generally higher quality of digital lines—allows for higher connection speeds.

The analog-to-digital conversion process reduces the signal quality slightly. This reduction in signal quality explains why you cannot purchase two 56K modems and place 56K calls between them. To obtain the maximum connect speed, one end of the call must be made or answered on a digital line.

Modem Signaling and Cabling

To gain further understanding of modems and remote connectivity, focus first on the lowest layer of the OSI system model: Layer 1, the physical layer. To connect a modem to a device such as a PC, router, or system of some other kind, you must establish physical connectivity. We've already discussed the various types of physical connections; now let's look deeper into the underlying communications that occur on a modem cable.

There are five primary signals that are required for modem communication on the physical layer: Data Set Ready (DSR), Data Terminal Ready

(DTR), Carrier Detect (CD), Ready to Send (RTS), and Clear to Send (CTS). These signals are used between the DCE and the DTE to determine when communications can occur, and when a call can be placed. Other signals such as Transmit (TX), Receive (RX), Ring Indicator (RI), and signal ground are used as well, but the first five are the basic building blocks for modem signaling.

There are two types of flow control in asynchronous communications: hardware and software. Software flow control is typically referred to as X-ON/X-OFF. Software flow control places the start and stop signals in the data stream, incurring a 2-byte per packet overhead.

Hardware flow control is typically referred to as CTS/RTS. Hardware flow control uses pin signaling to determine the flow of traffic in an asynchronous environment.

Cisco Console and AUX Port Cabling

To connect a modem to a Cisco router, you must use a cable. Most Cisco routers include two ports capable of having modems connected to them, the Console and Auxiliary (or AUX) port. These two ports have different cabling, pin, and speed requirements. You need to know the differences between the Console and AUX port to obtain performance from your router and the applications being used.

We'll start with the console port. Most of us have used the console port on a Cisco router to connect to the router for initial setup, configuration, and troubleshooting. The console port on most Cisco routers only support speeds up to 9600 bps—not a very desirable speed if you want to do dial-on-demand routing (DDR) or dial backup. Console cables are rolled cables, where pins 1 through 8 on one end are rolled in the cable and correspond to pins 8 through 1 at the other end. Figure 2.3 illustrates a rolled cable.

Figure 2.3 Rolled Cable.

Pin 1	_____	Pin 8
Pin 2	_____	Pin 7
Pin 3	_____	Pin 6
Pin 4	_____	Pin 5
Pin 5	_____	Pin 4
Pin 6	_____	Pin 3
Pin 7	_____	Pin 2
Pin 8	_____	Pin 1

The AUX port, in contrast to the Console port, has been designed to have modems connected to it. The AUX port on most routers can support speeds up to 38,400 bps, and the newer series of routers, 2600 and 3600 specifically, support speeds up to 115,200 bps. A rolled cable with a modem adapter (typically RJ-45 to DB-25) will suffice.

Modem Modulation Standards

The International Telecommunication Union Telecommunication Standardization Sector (ITU-T), formerly known as the International Telegraph and Telephone Consultative Committee (CCITT), is responsible for creating the standards for access to public telecommunications networks. Some of the more common standards created by the ITU-T are:

- E-series Telephone network and ISDN
- G-series International telephone connections and circuits
- I-series ISDN
- Q-series Telephone switching and signaling networks
- V-series Digital communications over the telephone network
- X-series Public data communications networks

The standards that apply to this chapter of the book come from the V-series. Some of the common standards and their respective speeds are as follows:

- V.22 Provides 1200 bits per second at 600 baud.
- V.22bis Provides 2400 bits per second at 600 baud.
- V.32 Provides 4800 and 9600 bits per second at 2400 baud.
- V.32bis Provides 14,400 bits per second or fallback to 12,000, 9600, 7200 and 4800 bits per second.
- V.32ter Provides 19,200 bits per second or fallback to 12,000, 9600, 7200 and 4800 bits per second. V.32ter was not an ITU-T standard and can operate at higher data rates with compression.
- V.34 Provides 28,800 bits per second or fallback to 24,000 and 19,200 bits per second and backwards compatibility with V.32 and V.32bis.

- V.32bis Provides up to 33,600 bits per second or fallback to 31,200 or V.34 transfer rates.
- V.35 The trunk interface between a network access device and a packet network at data rates greater than 19,200 bits per second. V.35 may use the bandwidth of several telephone circuits as a group.
- V.42 Provides the same transfer rates as V.32 and V.32bis, but with enhanced error-correction it is more reliable.
- V.42bis Provides the same error-correction as V.42, but with the addition of data compression.
- V.90 Provides up to 56,000 bits per second downstream (although usually somewhat less, based on line conditions and other factors).

There are proprietary standards as well as the ITU-T standards. These standards mostly came about due to the long delays it took the ITU-T to ratify new standards. In the past, new standards were reviewed and ratified by the ITU-T every four years. As technology accelerated, the four-year time span was too long to wait for ratification. Modem vendors were able to develop and deploy new technologies to the market faster than the ITU-T could keep up. This led to the creation of proprietary standards such as US Robotics, now 3Com, High Speed Transfer (HST) and X2 protocols, Telebit's Packetized Ensemble Protocol (PEP) and the K-Flex56 standard. These protocols are typically not found in the field anymore, as they have been replaced by internationally supported standards.

Error Control and Data Compression Methods

Given the speed limitations of modem communications, as well as the susceptibility of line noise and other outside influences on an analog circuit, it didn't take long for error-control and data-compression standards to be created. Let's first look at error control. Error control comes in many different modes, and although these modes use different methods for maintaining error control, they all serve the same function. Error control can be implemented in either hardware or software. The predominant form of error control on a modem connection is hardware-based (it is actually in the firmware of the modem). It is important to note that both modems must support the same error-control protocol.

Error control can be divided into two sub-categories: error checking and error correction. It is important to understand the differences between the two. Error control looks for errors in transmitted data; if errors are detected, it requests that the data be re-sent. The data is re-sent until it is transmitted error-free or until a timeout is reached and the connection is dropped.

The error-correction processes work by examining the header transmitted with the received block of data. If an error is found, the error correction protocol attempts to correct the block of data. If the block cannot be repaired, a retransmission of the block in question is requested.

Automatic Repeat Request (ARQ)

Automatic repeat request (ARQ) is a generic name for any error-correction scheme that mimics the way some binary file transfer protocols work, including Microcom Networking Protocol (MNP) and Link Access Procedure for Modems (LAPM).

Microcom Networking Protocol (MNP)

MNP is perhaps the most popular error-checking protocol. MNP is a proprietary system of error-correction and file-compression protocols developed by Microcom. MNP has nine classes, or levels: Class 1 through Class 10 (there is no Class 8). MNP is typically programmed into a modem's ROM or firmware. MNP Levels 4 and 5 are the most common and beneficial for asynchronous communications. The following is a listing of the main features of the various classes.

MNP 1 Asynchronous communications, in one direction (half duplex), whose main purpose is error checking. This error checking slows down communications by approximately 30 percent.

MNP 2 Asynchronous communications, in two directions simultaneously (full duplex). The error checking slows down communication by approximately 16 percent.

MNP 3 Synchronous communications, in two directions simultaneously. In addition to performing error checking, MNP 3 strips out the start and stop bits that were added to each byte before the data was transmitted, as start and stop bits are not required in synchronous communication. MNP then puts the data into packets. Removing the start and stop bits means that only 8 bits, rather than 10, are sent for each byte, gaining as much as a 20 percent increase in data transfer. Keep in mind that the time required for error checking, for the modem to strip the start and stop bits before transmission, and add them again on the receiving end before sending bytes on to the computer's serial port, results in an overall increase in speed of approximately 8 percent.

MNP 4 This class of MNP works with either synchronous or asynchronous communications with data placed into packets to reduce errors. The packetization also increases transmission speeds. The packet size is variable as the modem monitors the line conditions. A smaller packet is used on noisy lines and a larger packet can be used on a clean line. MNP 4 also streamlines some information in packet headers and increases data transmission overall by approximately 22 percent. MNP 4 also provides automatic error correction.

MNP 5 MNP 5 uses the same type of error correction and packetizing as MNP 4, but with a different twist. MNP 5 can alter data to reduce its size. This compression encodes data so that repeating or redundant data is eliminated and therefore is represented by fewer bits. The receiving modem decodes the data before transmitting it to the host's serial port. The effective throughput can be almost twice as much as a modem that's not using MNP 5. Keep in mind that if the data is already compressed, such as into a ZIP or TAR file, it might actually take longer for the data to be transmitted with MNP 5. This increase in time is caused by the modem examining the data for compressibility.

MNP 6, 7, 9, and 10 MNP levels 6, 7, and 9 feature enhancements in data compression and error correction. MNP Level 10 is used by a cellular modem developed by Microcom. No MNP level 8 exists.

Link Access Procedure for Modems (LAPM)

Link Access Procedure for Modems (LAPM) is a protocol that provides error control. LAPM is part of the V.42 specification. When a V.42 modem establishes a connection with another V.42 modem, it tries to establish LAPM as the error-correction protocol. If LAPM is not negotiated, MNP is tried. In the event that MNP is not available or not negotiated, a “normal” connection with no error correction or control is established. In a “normal” connection, error correction is typically implemented in software or the computer’s serial ports, in the program making the connection.

Data Compression Protocols

Data compression makes it possible to transfer more data quickly over a low bandwidth connection, such as a modem line. The suffix *bis* appended to a modem standard indicates data-compression capability. The ITU-T V.42bis standard, for example, specifies V.42bis as the data-compression scheme. A modem uses V.42bis only when LAPM is the error-correction protocol in use. MNP5 is the backup for the V.42bis with some V.42bis modems. A modem uses MNP 5 only when MNP is the hardware error-correction protocol in use. These data-compression and error-correction

techniques can increase data throughput dramatically. Let's examine the difference between speed and throughput.

Modem speed is a measure of the actual number of bits transmitted each second (bps). The number of bits transmitted by each baud, or change in signal state, is multiplied by the number of bauds per second. Throughput is a measure of the amount of useful data bytes transmitted. This measure is not always the same as the number bits transmitted per second. With the use of data compression, redundant or repeated bytes are stripped. Start and stop bits may also be removed, depending on the error-checking technology in use—in MNP 3, for example. As data is organized into packets to be transmitted by the modem, some data is tokenized, which means that characters are removed and replaced by fewer characters to represent the removed characters during transmission. The receiving modem must reconstruct the original characters before sending it to the PC, and can do this because it is using the same data compression technology.

For example, if a 9600 bps modem uses a data-compression technique that transmits only 2048 bytes for a 4096 byte file, the effective useful data transfer rate—or throughput—is twice what would be achieved using a normal 9600 bps connection. In essence, a 19,200 bps throughput rate is achieved. The modems do not actually transmit data any faster than 9600 bps, but the file is transmitted faster because the modems use fewer characters to represent the data in the file.

Configuring an Asynchronous Connection

There are two main types of asynchronous connections: inbound and outbound. Inbound, as the name implies, is a connection into the modem. For example, dialing into the office is an inbound connection to the receiving modem. Outbound, on the other hand, is a connection out of the modem. For example, when you dial into the office, the modem at the calling end is making an outbound connection. Sounds pretty easy, right? Well, if you add reverse Telnet to the mix, you can be making an inbound connection to the modem from a router and then establishing an outbound connection from your previously inbound connection. Reverse Telnet will be described in more detail in the Manual Configuration section of this chapter.

When connecting a modem to a router, it's important to know how you access the modem. Asynchronous connections on a router are also called TTY lines. TTY lines are similar to the virtual type terminal (VTY) ports on a router that allow Telnet access to the unit. Lines are addressed differ-

ently on each model of router based on the following information: The AUX port is line 1 on a standard router; the last TTY line +1 on access servers such as the 2509, 2510, 2511, AS5200, and AS5300; line 65 on the 2600s and 3620s; and line 129 on the 3640. So the line number for the AUX port on a 2501 is 1, while the AUX port on a 2620 is 65.

Router Configuration

So now that you have your modem cabled into the AUX port of your router, you are ready to start using it, right? Not quite. You still must configure the router with the appropriate parameters to communicate with the modem. You need to tell the router what line you are using, the speed, flow control, and direction in which you will be using the modem, and the application in use.

Let's start first with configuring the line, because you need to tell the router where the modem is located. This is done by going into configuration mode on the router and issuing the following command:

```
Central(config)#line 129  
Central(config-line)#[/pre>
```

As you can see from the information displayed on the screen, you are now in line configuration mode. By using the context-sensitive help you can see all of the commands that apply to line configuration:

Line configuration commands:

absolute-timeout	Set absolute timeout for line disconnection
access-class	Filter connections based on an IP access list
activation-character	Define the activation character
arap	Appletalk Remote Access Protocol
autobaud	Set line to autobaud
autocommand	Automatically execute an EXEC command
autocommand-options	Autocommand options
autohangup	Automatically hangup when last connection closes
autoselect	Set line to autoselect
callback	Callback settings
data-character-bits	Size of characters being handled

databits	Set number of data bits per character
default	Set a command to its defaults
disconnect-character	Define the disconnect character
dispatch-character	Define the dispatch character
dispatch-machine	Reference a TCP dispatch state machine
dispatch-timeout	Set the dispatch timer
domain-lookup	Enable domain lookups in show commands
editing	Enable command line editing
escape-character	Change the current line's escape character
exec	Start an EXEC process
exec-banner	Enable the display of the EXEC banner
exec-character-bits	Size of characters to the command exec
exec-timeout	Set the EXEC timeout
exit	Exit from line configuration mode
flowcontrol	Set the flow control
flush-at-activation	Clear input stream at activation
full-help	Provide help to unprivileged user
help	Description of the interactive help system
history	Enable and control the command history function
hold-character	Define the hold character
insecure	Mark line as 'insecure' for LAT
international	Enable international 8-bit character support
ip	IP options
keymap-type	Specify a keymap entry to use
lat	DEC Local Area Transport (LAT) protocol-specific configuration
length	Set number of lines on a screen
location	Enter terminal location description
lockable	Allow users to lock a line

logging	Modify message logging facilities
login	Enable password checking
logout-warning	Set Warning countdown for absolute timeout of line
modem	Configure the Modem Control Lines
monitor	Copy debug output to the current terminal line
motd-banner	Enable the display of the MOTD banner
no	Negate a command or set its defaults
notify	Inform users of output from concurrent sessions
ntp	Configure NTP
padding	Set padding for a specified output character
parity	Set terminal parity
password	Set a password
private	Configuration options that user can set will remain in effect between terminal sessions
privilege	Change privilege level for line
refuse-message	Define a refuse banner
rotary	Add line to a rotary group
rxspeed	Set the receive speed
script	specify event related chat scripts to run on the line
session-disconnect-warning	Set warning countdown for session-timeout
session-limit	Set maximum number of sessions
session-timeout	Set interval for closing connection when there is no input traffic
special-character-bits	Size of the escape (and other special) characters
speed	Set the transmit and receive speeds
start-character	Define the start character
stop-character	Define the stop character

stopbits	Set async line stop bits
telnet	Telnet protocol-specific configuration
terminal-type	Set the terminal type
timeout	Timeouts for the line
transport	Define transport protocols for line
txspeed	Set the transmit speeds
vacant-message	Define a vacant banner
width	Set width of the display terminal
x25	X25 protocol-specific configuration

Next you'll set the speed, as it will dictate to the modem the bit rate of the data flowing between the modem and the router. First, let's look at the line before we make any changes:

```
Central#show line 129
      Tty Typ Tx/Rx A Modem RotaY AccO AccI   Uses   Noise   Overruns   Int
  129 AUX  9600/9600 -  -  -  -    0       1     0/0          -
Line 129, Location: "", Type: ""
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 9600/9600, no parity, 2 stopbits, 8 databits
Status: Ready
Capabilities: none
Modem state: Ready
Group codes:      0
Modem hardware state: CTS* noDSR  DTR RTS
TTY NUMBER 129
Parity Error = 0 Framing Error = 0 Receive Error = 0 Overrun = 0
Outcount = 0 totalout = 39 incount = 0 totalin = 39

Special Chars: Escape Hold Stop Start Disconnect Activation
               ^^x    none  -  -    none
Timeouts: Idle EXEC  Idle Session  Modem Answer  Session Dispatch
          00:10:00      never           none    not set
```

```
Idle Session Disconnect Warning
    never
Login-sequence User Response
    00:00:30
Autoselect Initial Wait
    not set

Modem type is unknown.
Session limit is not set.
Time since activation: never
Editing is enabled.
History is enabled, history size is 10.
DNS resolution in show commands is enabled
Full user help is disabled
Allowed transports are lat pad v120 lapb-ta mop telnet rlogin nasi.
Preferred is lat.
No output characters are padded
No special data dispatching characters
Central#
```

Now let's implement the speed change, then exit configuration mode to see the speed we set for the line. Let's also change the default stop bits for the line from 2 to 1 to reduce the asynchronous framing overhead, and set the flow control to hardware (CTS/RTS):

```
Central(config)#line 129
Central(config-line)#speed 115200
Central(config-line)#stopbits 1
Central(config-line)#flowcontrol hardware
Central(config-line)#end

Central#sh line 129
Tty Typ Tx/Rx A Modem Roty AccO AccI Uses Noise Overruns Int
  129 AUX 115200/115200- - - - - 0 1
  0/0      -
```

```
Line 129, Location: "", Type: ""

Length: 24 lines, Width: 80 columns

Baud rate (TX/RX) is 115200/115200, no parity, 1 stopbits, 8 databits

Status: Ready

Capabilities: Hardware Flowcontrol In, Hardware Flowcontrol Out

Modem state: Ready

Group codes: 0

Modem hardware state: CTS* noDSR DTR RTS

TTY NUMBER 129

Parity Error = 0 Framing Error = 0 Receive Error = 0 Overrun = 0

Outcount = 0 totalout = 39 incount = 0 totalin = 39

Special Chars: Escape Hold Stop Start Disconnect Activation

      ^^x    none   -   -    none

Timeouts: Idle EXEC Idle Session Modem Answer Session Dispatch

      00:10:00    never        none  not set

                           Idle Session Disconnect Warning

                           never

                           Login-sequence User Response

                           00:00:30

                           Autoselect Initial Wait

                           not set

Modem type is unknown.

Session limit is not set.

Time since activation: never

Editing is enabled.

History is enabled, history size is 10.

DNS resolution in show commands is enabled

Full user help is disabled

Allowed transports are lat pad v120 lapb-ta mop telnet rlogin nasi.

Preferred i

s lat.
```

```
No output characters are padded  
No special data dispatching characters  
Central#
```

You can see that the speed of the line has been set to the maximum for this platform, a Cisco 3640; you can also see the change made to the stop bits and the flow control. The router now has the parameters it is to use when communicating with the modem. A modem on a router can be configured as dial-in only, dial-out only, or both. let's look first at dial-in mode.

If you go into line configuration mode on the router and look at the context-sensitive help, you'll see that there are two commands that would configure the modem for dial-in. There are significant differences between the two commands that need to be understood before configuring your modem. Below is a list of the commands you can apply to the modem.

```
Central(config)#line 129  
Central(config-line)#modem ?  
CTS-Alarm      Alarm device which only uses CTS for call control  
DTR-active     Leave DTR low unless line has an active incoming  
               connection  
               or EXEC  
Dialin        Configure line for a modern dial-in modem  
Host          Devices that expect an incoming modem call  
InOut         Configure line for incoming AND outgoing use of modem  
Printer       Devices that require DSR/CD active  
answer-timeout Set interval between the time the server raises DTR in  
               response to RING and the modem responds to CTS  
autoconfigure  Automatically configure modem on line  
busyout       Block calls to and from the modem
```

Let's focus on the **modem inout** and **modem dialin** commands. The **modem dialin** uses the DSR signal and supports the use of hardware flow control between the router and the modem. This configures the line for dial-in access only. An older command, **modem callin**, is not listed in the context-sensitive help, but can be used as long as the **flowcontrol hardware** command is not used. The **modem callin** command is designed for use with older modems that do not support auto-answer. The **modem callin** command uses CTS; when a ring is detected on the line, the router raises the DTR signal, which indicates the modem should answer the call.

Below is the output of a show line after the **modem dialin** command has been given. You can see that the router now can use the modem for dial-in and that the modem RI is Carrier Detect using DSR:

```
Central#show line 129
      Tty Typ Tx/Rx A Modem Roty AccO AccI Uses   Noise  Overruns  Int
      129 AUX 115200/115200- DialIn - - - 0       1       0/0      -
Line 129, Location: "", Type: ""
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 115200/115200, no parity, 1 stopbits, 8 databits
Status: No Exit Banner
Capabilities: Hardware Flowcontrol In, Hardware Flowcontrol Out
Modem RI is CD
Modem state: Idle
Group codes: 0
Modem hardware state: CTS* noDSR DTR RTS
TTY NUMBER 129
Parity Error = 0 Framing Error = 0 Receive Error = 0 Overrun = 0
Outcount = 0 totalout = 39 incount = 0 totalin = 39

Special Chars: Escape Hold Stop Start Disconnect Activation
              ^^x    none   -     -      none
Timeouts:     Idle EXEC  Idle Session Modem Answer Session Dispatch
              00:10:00      never        none      not set
                           Idle Session Disconnect Warning
                           never
                           Login-sequence User Response
                           00:00:30
                           Autoselect Initial Wait
                           not set

Modem type is unknown.
Session limit is not set.
Time since activation: never
```

```
Editing is enabled.  
History is enabled, history size is 10.  
DNS resolution in show commands is enabled  
Full user help is disabled  
Allowed transports are lat pad v120 lapb-ta mop telnet rlogin nasi.  
Preferred i  
s lat.  
No output characters are padded  
No special data dispatching characters  
Central#
```

The **modem inout** command is used to allow both incoming and outgoing connections to modems. When the **modem inout** command is issued, the router uses the RING and DTR signals for carrier detection. Note that Cisco has a specific Windows utility that will allow client PCs to use the outbound capabilities of a modem. This utility is downloadable from www.cisco.com. The following example is the output of a show line after the **modem inout** command has been configured. You see that the router now can use the modem for dial-in and dial-out and that the modem RI is Carrier Detect using DSR.

```
Central#sh line 129  
Tty Typ Tx/Rx A Modem Roty Acc0 AccI Uses Noise Overruns Int  
129 AUX 115200/115200- inout - - - 0 1 0/0 -  
  
Line 129, Location: "", Type: ""  
Length: 24 lines, Width: 80 columns  
Baud rate (TX/RX) is 115200/115200, no parity, 1 stopbits, 8 databits  
Status: No Exit Banner  
Capabilities: Hardware Flowcontrol In, Hardware Flowcontrol Out  
Modem Callout, Modem RI is CD  
Modem state: Idle  
Group codes: 0  
Modem hardware state: CTS* noDSR DTR RTS  
TTY NUMBER 129  
Parity Error = 0 Framing Error = 0 Receive Error = 0 Overrun = 0  
Outcount = 0 totalout = 39 incount = 0 totalin = 39
```

```
Special Chars: Escape Hold Stop Start Disconnect Activation
              ^^x    none   -      -      none

Timeouts:    Idle EXEC    Idle Session    Modem Answer    Session    Dispatch
             00:10:00    never           none        not set

                                         Idle Session Disconnect Warning
                                         never

                                         Login-sequence User Response
                                         00:00:30

                                         Autoselect Initial Wait
                                         not set

Modem type is unknown.
Session limit is not set.
Time since activation: never
Editing is enabled.
History is enabled, history size is 10.
DNS resolution in show commands is enabled
Full user help is disabled
Allowed transports are lat pad v120 lapb-ta mop telnet rlogin nasi.
Preferred i
s lat.
No output characters are padded
No special data dispatching characters
Central#
```

Modem Configuration

Now that the modem is connected to the router and configured for dial-in/dial-out, it's time to configure the modem. This includes setting modem and vendor specific strings to the modem, as well as any other requirements, such as the number of rings to answer on. There are two ways to configure the modem from the router: manual configuration and automatic configuration.

Manual Configuration

Manual configuration of the modem is accomplished by using reverse Telnet. Reverse Telnet establishes a terminal session to modems connected

to an access server. This can be useful for modem configuration, troubleshooting, or even as part of an application. A reverse Telnet session is initiated from the router to the modem rather than the “normal” forward connection from the modem to the router. Reverse Telnet sessions are established by using an active up/up interface on the router’s IP address and port 2000 + n, where n is the number of the line the modem is connected to. For example, to connect to a modem on line 129, the AUX port on a Cisco 3640, you would use the following command:

```
Router#telnet 1.1.1.1 2129  
Trying 1.1.1.1, 2129 ... Open
```

In networks where there is more than one path to the router, the use of a loopback interface for the reverse Telnet session may be desirable. Loopback interfaces are virtual interfaces on a router that are always up as long as the router is running. This means that the loopback will always be reachable in a fault tolerant or redundant network, thus the modem is reachable as well. If you were to use the IP address of the Ethernet interface of the router and that interface goes down for any reason, the modem is unreachable for reverse Telnet. Loopback interfaces have many uses on a network and reverse Telnet is just one example.

A way to simplify the reverse Telnet process, especially when you have many modems on an access server, is to create an IP host entry for each modem. This allows you to type in the name of the modem and reverse Telnet to it. So, for example, you could create an IP host entry for modem 1 2129 1.1.1.1 and type in **modem1** from the router to connect to the modem.

```
Central(config)#ip host modem1 2129 1.1.1.1  
Central(config)#exit  
Central#modem1  
Translating "modem1"  
Trying modem1 (1.1.1.1, 2129)... Open
```

Disconnecting from the reverse Telnet session requires two steps. The first step is to suspend the connection. This is done by using the Ctrl-Shift-6 X keyboard command (press Ctrl-Shift-6 at the same time, then release the keys and press the letter X. This will suspend the session).

```
at  
OK  
(Ctrl+Shift+6 x was performed)
```

```
Central#
```

Now we can disconnect the session by using the disconnect command.

```
Central#disconnect  
Closing connection to modem1 [confirm]  
Central#
```

Once connected to the modem, you can enter any command that the modem can accept from a PC directly connected to the modem using a terminal emulation program. AT commands that alter the modem's default configuration or display the modem's setting can be used. Additionally, you can use initialization strings that are required for the modem to work the way you intend it to work—for example, if you wanted to set up the modem so that it answers calls on the fifth ring, you can reverse Telnet to the modem and enter in the required string. The following example shows the modem's default configuration that is stored in nonvolatile RAM (NVRAM):

```
Central#modem1  
Translating "modem1"  
Trying modem1 (1.1.1.1, 2129)... Open  
at  
OK  
ati5  
USRobotics Courier V.Everything NVRAM Settings...
```

```
DIAL=PULSE B0 F1 M1 X1
```

```
BAUD=115200 PARITY=N WORDLEN=8
```

```
&A1 &B1 &G0 &H0 &I0 &K1 &L0 &M4 &N0  
&P0 &R1 &S0 &T5 &X0 &Y1 %N6 #CID=0
```

```
S00=001 S02=043 S03=013 S04=010 S05=008 S06=002 S07=060  
S08=002  
S09=006 S10=007 S11=070 S12=050 S13=000 S15=000 S19=000  
S21=010  
S22=017 S23=019 S24=150 S25=005 S26=001 S27=000 S28=008  
S29=020
```

```
S31=000  S32=009  S33=000  S34=000  S35=000  S36=000  S37=000  
S38=000  
S39=000  S40=000  S41=000  S42=126   S43=200   S44=015   S51=000  
S53=000  
S54=064  S55=000  S56=000  S57=000  S69=000  S70=000  
STORED PHONE NUMBERS
```

OK

Now you change the appropriate S register to make the modem answer on the fifth ring, and save the change to NVRAM using the following command, then you display your changes to verify they were accepted:

```
ats0=5&w  
OK  
atis5  
USRobotics Courier V.Everything NVRAM Settings...
```

```
DIAL=PULSE  B0  F1  M1  X1  
BAUD=115200  PARITY=N  WORDLEN=8  
  
&A1  &B1  &G0  &H0  &I0  &K1  &L0  &M4  &N0  
&P0  &R1  &S0  &T5  &X0  &Y1  %N6  #CID=0  
  
S00=005  S02=043  S03=013  S04=010  S05=008  S06=002  S07=060  
S08=002  
S09=006  S10=007  S11=070  S12=050  S13=000  S15=000  S19=000  
S21=010  
S22=017  S23=019  S24=150  S25=005  S26=001  S27=000  S28=008  
S29=020  
S31=000  S32=009  S33=000  S34=000  S35=000  S36=000  S37=000  
S38=000  
S39=000  S40=000  S41=000  S42=126   S43=200   S44=015   S51=000  
S53=000  
S54=064  S55=000  S56=000  S57=000  S69=000  S70=000
```

OK

Automatic Configuration

Now that we have covered the manual configuration of a modem for an access server, let's look at how you can automate the modem configuration

process. Cisco has included initialization strings for 14 of the more common modems in a *modemcap* database built into their IOS. The default modem initialization strings in the *modemcap* database are for the following modems:

- Codex 3620
- US Robotics Courier
- US Robotics Sportster
- Hayes Optima
- Global Village
- Viva
- Telebit T3000
- Microcom HDMS
- Microcom Server
- NEC V34
- NEC V110
- NEC PIAFS
- Cisco V110
- MICA

The initialization strings for each modem type can be viewed by typing **show modemcap name** with name being the entry of the model in the *modemcap* database. For example, to see the *modemcap* database entry for a US Robotics Courier modem, the command would be **show modemcap usr_courier**. The following are the results of the output from the command:

```
Central#show modemcap usr_courier
Modemcap values for usr_courier
Factory Defaults (FD):  &F
Autoanswer (AA):  S0=1
Carrier detect (CD):  &C1
Drop with DTR (DTR):  &D2
Hardware Flowcontrol (HFL):  &H1&R2
Lock DTE speed (SPD):  &B1
DTE locking speed (DTE):  [not set]
```

```
Best Error Control (BER): &M4
Best Compression (BCP): &K1
No Error Control (NER): &M0
No Compression (NCP): &K0
No Echo (NEC): E0
No Result Codes (NRS): Q1
Software Flowcontrol (SFL): [not set]
Caller ID (CID): [not set]
On-hook (ONH): H0
Off-hook (OFH): H1
Miscellaneous (MSC): [not set]
Template entry (TPL): default
Modem entry is built-in.
```

With the modemcap database in the IOS you can instruct the router to use a specific initialization string for each line. This is done using the **modem autoconfigure modem_type** command. In the line configuration you can issue the **modem autoconfigure usr_courier** command and the router will then use the settings in the modemcap database for the US Robotics Courier modem.

In the event that you are unsure as to which modemcap entry to use for your modem, you can use the **modem autodiscovery** command. This command, when applied to the line of a router, makes the router go through the modemcap database to find the correct initialization sting for your modem. In the event that the autodiscovery process is not successful in identifying your modem, manual configuration is required. The next example illustrates the use of the **modem autodiscovery** command on the access server.

```
Central#config t
Enter configuration commands, one per line. End with CNTL/Z.
Central(config)#line 129
Central(config-line)#modem autoconfigure discovery
Central(config-line)#end
Central#
14:51:43: TTY129: autoconfigure probe started
```

Now look at the line and see that the modem type has been detected and configured by IOS.

```
Central#sh line 129

  Tty Typ Tx/Rx A Modem   Roty AccO AccI   Uses   Noise   Overruns   Int
  129 AUX 115200/115200- inout - - -      5       1       0/0          -
Idle

Line 129, Location: "", Type: ""
Length: 24 lines, Width: 80 columns
Baud rate (TX/RX) is 115200/115200, no parity, 2 stopbits, 8 databits
Status: No Exit Banner, Modem Detected
Capabilities: Hardware Flowcontrol In, Hardware Flowcontrol Out
               Modem Callout, Modem RI is CD, Modem Discovery
Modem state: Idle
Group codes:    0
Modem hardware state: CTS* noDSR  DTR RTS
TTY NUMBER 129
Parity Error = 0 Framing Error = 0 Receive Error = 0 Overrun = 0
Outcount = 0 totalout = 464 incount = 0 totalin = 13156
, Modem Configured
Special Chars: Escape Hold Stop Start Disconnect Activation
               ^^x   none   -   -   none
Timeouts: Idle EXEC   Idle Session   Modem Answer   Session   Dispatch
          00:10:00   never   none   not set
                           Idle Session Disconnect Warning
                           never
                           Login-sequence User Response
                           00:00:30
                           Autoselect Initial Wait
                           not set

Modem type is usr_courier.
Session limit is not set.
```

```
Time since activation: never
Editing is enabled.
History is enabled, history size is 10.
DNS resolution in show commands is enabled
Full user help is disabled
Allowed transports are lat pad v120 lapb-ta mop telnet rlogin nasi.
Preferred is lat.
No output characters are padded
No special data dispatching characters
Central#
```

Chat Scripts

Chat scripts are useful tools when working with asynchronous communications. These scripts help automate the processes involved with dial-in connectivity and can save the administrator of a dial-in service quite a bit of time. Chat scripts are strings of text used to send commands for modem dialing, logging on to remote systems, and initializing asynchronous devices connected to asynchronous lines. Chat scripts can be configured to run automatically when a specific event occurs on a line such as a reset, line activation, incoming connection initiation, asynchronous dial-on-demand routing, and line startup. Chat scripts can also be run manually from the privileged EXEC mode.

Creating a chat script is a two-step process. The first step is to define the chat script in the router's global configuration. Chat scripts can be named anything you would like—however, Cisco's recommendation for chat script naming for modem scripts uses the modem vendor, modem type and modulation (a Practical Peripheral PM14000FX V.34 modem would have a chat script name of pp-pm1400fx-v34). It is important to note that chat scripts are case-sensitive.

The second step is to apply the chat script to a line. The chat script can be automatically executed based on the five specific events mentioned earlier, using the `script` command. The following is a list of the `script` command options and when the script will be run:

- *script activation regexp* Start a chat script on a line whenever a command EXEC is started on the line.
- *script connection regexp* Start a chat script whenever a network connection is made to the line.

- *script dialer regexp* Specify a modem script for dial-on-demand routing on a line.
- *script reset regexp* Start a chat script whenever a line is reset.
- *script startup regexp* Start a chat script whenever the router starts up.

Note that *regexp* stands for *regular expression*. A regular expression is a pattern to match against an input string—when creating a regular expression, you specify a pattern that a string must match. Regular expressions are used for many different functions in Cisco IOS, but in this context they refer to the name of a chat script created in the global configuration of the router.

To create a chat script that would redial a number until a connection has been established, you could use the following script.

```
Central(config)#chat-script redial ABORT ERROR ABORT BUSY ABORT "NO  
ANSWER" "" "ATH" OK "ATDT\T"  
TIMEOUT 30 CONNECT
```

This chat script instructs the modem to abort the dialing process and start again if the router receives an *error*, *busy*, or *no answer* result from the modem. The router then sends the **ATH** command to hang up the modem, waits for an **OK** from the modem, then issues an **ATDT\T** command. This command forces the modem to re-dial the number with a timeout of 30 seconds (the default timeout is 5), until the modem returns a *connect* result. The two quotes with nothing between them tell the router to expect a null string from the modem.

This particular chat script would be best used in a dial-on-demand routing scenario where it is imperative that the modem establishes a connection to the called site. This script automates and controls the dialing process so that no administrator or user intervention is required.

Providing Asynchronous Dial-in Terminal Services

The flexibility of the Cisco access server platform is remarkable. The same access server can provide a multitude of dial-in, dial-out services and service a wide variety of network clients ranging from UNIX clients, to DEC LAT and IBM mainframe 3270 clients. We'll cover the abilities of the access servers—focusing on Telnet, rlogin, LAT, and TN3270 in this section.

Terminal Services

As networks evolve, most applications are being re-written for Layer 3 protocols such as Transmission Control Protocol/Internet Protocol (TCP/IP). However, there is still a large installed base of legacy systems that require network connectivity. The Cisco access server platform can provide the required connectivity to many of these systems.

Telnet and rlogin are protocols that enable TCP/IP login to a host. Telnet is a virtual terminal protocol that is part of the TCP/IP suite. Telnet is a widely used protocol currently supported on most platforms. Rlogin is a remote login service that was developed for the BSD UNIX environment. Rlogin provides better control and output suppression than Telnet, but can only be used when the host supports rlogin. Rlogin can be configured in the UNIX environment to support a “trusted host” model (that is, a user can rlogin to another UNIX system that is trusted with no username or password prompting). Cisco’s implementation of rlogin does not support the “trusted host” model.

Cisco’s implementation of Telnet works in most environments “out of the box,” with no additional configuration required. However, in some instances the Telnet configuration may require some modification to meet your needs.

The Telnet command is issued from the router’s EXEC prompt and requires at least one command-line argument, the destination host. This can be either the IP address of the destination host or the DNS name. For DNS resolution to work, the router must be configured with the IP addresses of your DNS server(s).

```
Central>telnet 1.1.1.1  
Trying 1.1.1.1 ... Open
```

User Access Verification

Password:

The IP address or name of the destination host is not the only argument Telnet supports. Telnet defaults to establish a connection on TCP port 23. This can be overridden by specifying an alternative port number after the IP address. The next example illustrates how you would Telnet to TCP port 25, SMTP, on a test AS/400 to verify connectivity.

```
Central>telnet 1.1.1.2 25
```

```
Trying 1.1.1.2, 25 ... Open
220 TEST400 running IBM AS/400 SMTP V04R03M00 on Thu, 27 Jul 2000
07:30:
08 -0400.
quit
221 TEST400 running IBM AS/400 SMTP V04R03M00. Connection closing.
```

Below is a list of the options available when using Telnet from a Cisco router:

```
Central>telnet 1.1.1.1 ?
/diag          Enable telnet debugging mode
/line          Enable telnet line mode
/noecho        Disable local echo
/route:        Enable telnet source route mode
/source-interface Specify source interface
/stream         Enable stream processing
<0-65535>      Port number
bgp             Border Gateway Protocol (179)
chargen         Character generator (19)
cmd              Remote commands (rcmd, 514)
daytime         Daytime (13)
discard         Discard (9)
domain          Domain Name Service (53)
echo             Echo (7)
exec             Exec (rsh, 512)
finger           Finger (79)
ftp               File Transfer Protocol (21)
ftp-data         FTP data connections (used infrequently, 20)
gopher           Gopher (70)
hostname         NIC hostname server (101)
ident             Ident Protocol (113)
irc               Internet Relay Chat (194)
klogin            Kerberos login (543)
kshell            Kerberos shell (544)
login             Login (rlogin, 513)
```

```
lpd                  Printer service (515)
nntp                Network News Transport Protocol (119)
pim-auto-rp         PIM Auto-RP (496)
pop2                Post Office Protocol v2 (109)
pop3                Post Office Protocol v3 (110)
smtp                Simple Mail Transport Protocol (25)
sunrpc              Sun Remote Procedure Call (111)
syslog              Syslog (514)
tacacs              TAC Access Control System (49)
talk                Talk (517)
telnet              Telnet (23)
time                Time (37)
uucp                Unix-to-Unix Copy Program (540)
whois               Nicname (43)
www                 World Wide Web (HTTP, 80)
<cr>
```

These optional commands can change the operation of Telnet dramatically. You can force the Telnet packets to take a different route than they would normally take, based on the router's routing table by using the **/route:** option. In the following example, you force the router to take a path that goes from your router Central to another router with an IP address of 1.1.1.10, then go to the router with an IP address of 2.2.2.2. This can be useful when troubleshooting path-related issues or unknown access lists on the "normal" route the packet would take.

```
Central>telnet 1.1.1.1 /route: 1.1.1.10 2.2.2.2
```

Rlogin does not have as many available options for the command line as Telnet. The following options can be used with the **rlogin** command.

```
Central#rlogin 1.1.1.1 ?
-l      Specify remote username
/user   Specify remote username
debug   Enable rlogin debugging output
<cr>
```

You can see that there are two options that have the same function, the specification of a remote username. The first option, **-l**, is supported by the standard BSD UNIX rlogin program. The second option, **/user**, allows

remote users to login without the **-l** option. It is important to note that the **/user** option is not compatible with the UNIX **-l** option.

An example of an **rlogin** command that would log in to a remote system with an IP address of 1.1.1.1 and a username of joeuser would look like this.

```
Central#rlogin 1.1.1.1 -l joeuser
```

Cisco routers can also support local-area transport (LAT) terminal services. LAT is a proprietary protocol developed by Digital Equipment Corporation (DEC). LAT is the most commonly-used protocol for connectivity to DEC VMS hosts. LAT is similar to Telnet in that it allows remote users to establish terminal sessions and pass keystrokes between the systems. However, LAT was designed for use in the local area network (LAN) and cannot be routed as it has no network layer. Cisco allows the translation of LAT into X.25 or Telnet packets that can then be routed across an internetwork.

Let's cover some basic LAT functionality. LAT is an asymmetrical protocol, meaning that it has a master-and-slave functionality. A LAT master initiates a LAT session to a LAT slave by sending a LAT circuit start message. The LAT slave responds with a circuit start message of its own. The circuit setup between the master and the slave can support anywhere from 1 to 255 sessions. When using a Cisco router as a LAT terminal server, the router is the master and the destination VMS host is the slave. Cisco IOS software supports the LAT 5.2 specification.

Devices on a LAT network such as modems, printers, hosts and application software are referred to as *services*. LAT supports service advertisement through Ethernet multicast messages, or *service announcements*. LAT devices listen to these announcements and build a table of services referred to as *learned services*. The Cisco IOS supports both advertised and learned services and can therefore participate fully in a LAT network.

Services in a LAT network can have ratings. Ratings are parameters that allow devices in a LAT network make intelligent decisions as to which service to connect. A LAT cluster will have different service ratings for its various nodes. The LAT node can intelligently connect to the LAT service with the highest rating, as it has the lowest load.

On a LAT network, the potential exists for any user to connect to any service. To restrict access to devices on a LAT network, LAT group codes were developed. Devices in different LAT groups can only see and communicate with devices or services in their same group. By default the LAT group codes allow all devices on a LAT network to see and communicate with each other. Group codes can be implemented to allow controlled access to the network. Group codes typically are broken down into logical

breaks in an organization such as department or application. It is important to note that a LAT node's services cannot be filtered on a service-by-service basis. Access to a LAT node is either all or none.

The basics of enabling LAT on an access server is as simple as one command, **lat enable**, on an interface connected to a LAT network, such as Ethernet. However, Cisco's IOS allows us to configure LAT in a number of different ways and gives us very granular control of LAT on the access server. The following is an example of a minimal configuration for a LAT enabled access server.

```
hostname Central
...
interface Ethernet0
    ip address 192.168.1.2 255.255.255.0
    no ip directed-broadcast
    lat enabled
...
lat service CENTRAL enabled
...
```

This configuration enables LAT on the Ethernet interface and advertises the access server, named Central, as a LAT service. The following is an example of the output you would get from a LAT-enabled access server that is on the same LAT network as a VMS host called LATHOST. In this example, the VMS host LATHOST is actually another Cisco router.

```
Central#sh lat services
Service Name      Rating   Interface  Node (Address)
CENTRAL           5        Local
LATHOST           5        Ethernet0  LATHOST (00b0.6416.be80)
Central#
```

With this configuration you can use LAT to connect to the LATHOST by using the **lat lathost** command, where the lathost is the name of the LAT service you want to connect to.

```
Central#lat lathost
Trying LATHOST...Open
```

User Access Verification

```
Password:
```

```
R3>
```

This works the same way when using a VMS host or a Cisco router. Either way, you are using LAT as your transport. This can be verified by issuing a **show lat sessions** command from the router and viewing the session you just created.

```
R3>sh lat sessions
```

```
tty130, virtual tty from host CENTRAL
```

Session data:

```
Name LATHOST, Remote Id 1, Local Id 1  
Remote credits 1, Local credits 1, Advertised Credits 3  
Flags: DataA, Send Credits  
Max Data Slot 255, Max Attn Slot 255, Stop Reason 0
```

Remote Node data:

```
Node "CENTRAL", usage 1, Interface FastEthernet0/0, Address  
0010.7b38.663f  
Timer 109, sequence 1, changes 159, flags 0x0, protocol 5.2  
Facility 0, Product code 234, Product version 48  
Recv 128/91/204, Xmit 129/82/1684, 0 Dups, 0 ReXmit  
Bad messages: 0, Bad slots: 0, Solicits accepted: 0  
Solicits rejected: 0, Multiple nodes: 0  
Groups: 0  
Service classes: 1
```

```
R3>
```

When defining a LAT service on a router, a number of options can be specified. The following is a list of the options that are available to you when configuring a LAT service.

```
Central(config)#lat service Central ?  
autocommand      Associate a command with a service  
enabled          Enable inbound connections
```

```
identification      Set LAT service identification for specified service
password          Set up a LAT password for the service
rating            Set the static service rating for specified service
rotary             Associate a rotary group with a service
<cr>
```

These options allow LAT to be configured to automatically run a command, add a descriptive string to differentiate services, set the rating, configure a password, or associate the service with a rotary group. The following is an example configuration for a LAT service named Central that provides a password, identification, and an autocommand:

```
lat service CENTRAL ident Central Router
lat service CENTRAL autocommand show ip route
lat service CENTRAL password LAT
lat service CENTRAL enabled
```

Here is the output you would see from this LAT service once you have connected to it.

```
R3>sh lat services
Service Name      Rating   Interface  Node (Address)
CENTRAL           5        FastEthernet0/0  CENTRAL (0010.7b38.663f)

Ident: Central Router
LATHOST           5        Local
```

```
R3>lat central
Trying CENTRAL...Password required
```

```
Password: Trying CENTRAL...Open
```

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
```

```
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate default
U - per-user static route, o - ODR
T - traffic engineered route
```

Gateway of last resort is not set

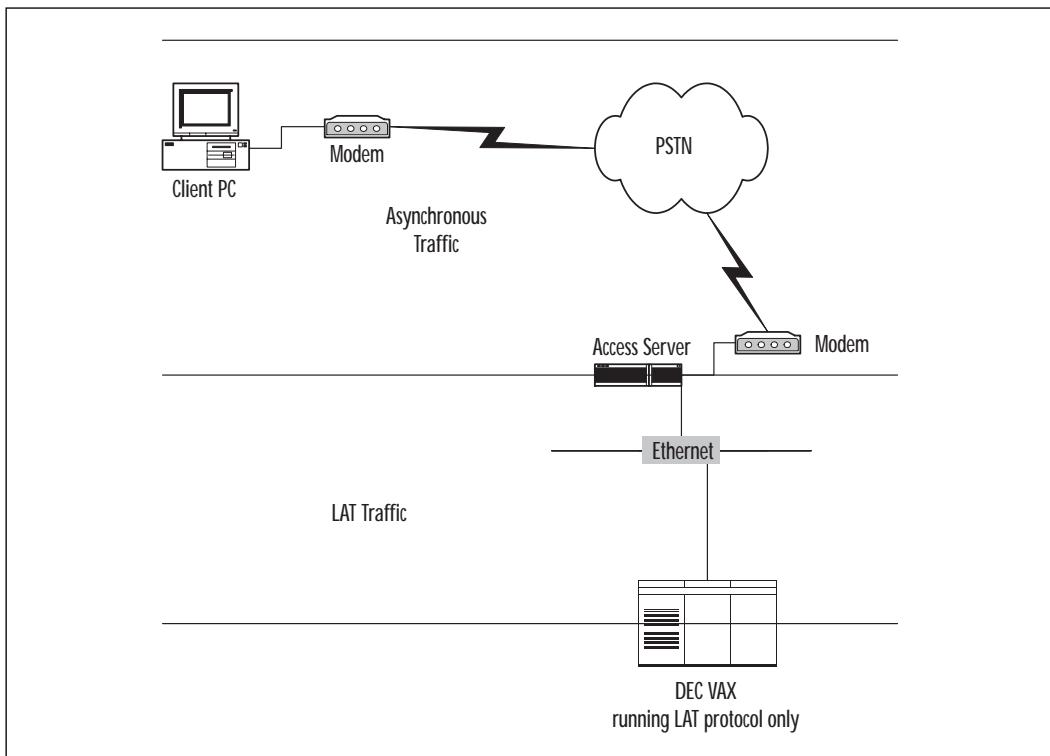
```
C      192.168.1.0/24 is directly connected, Ethernet0
```

```
[Connection to central closed by foreign host]
```

```
R3>
```

Cisco routers can also support IBM TN3270 services. TN3270 allows any terminal to emulate an IBM 3270 terminal. IBM 3270 terminals allow connectivity to IBM mainframes. A Cisco access server can be used to provide TN3270 emulation services to non-TN3270 users. The following is a listing of the IBM 3270 terminal types supported by Cisco IOS:

Figure 2.4 LAT Terminal Services from a Cisco Router.



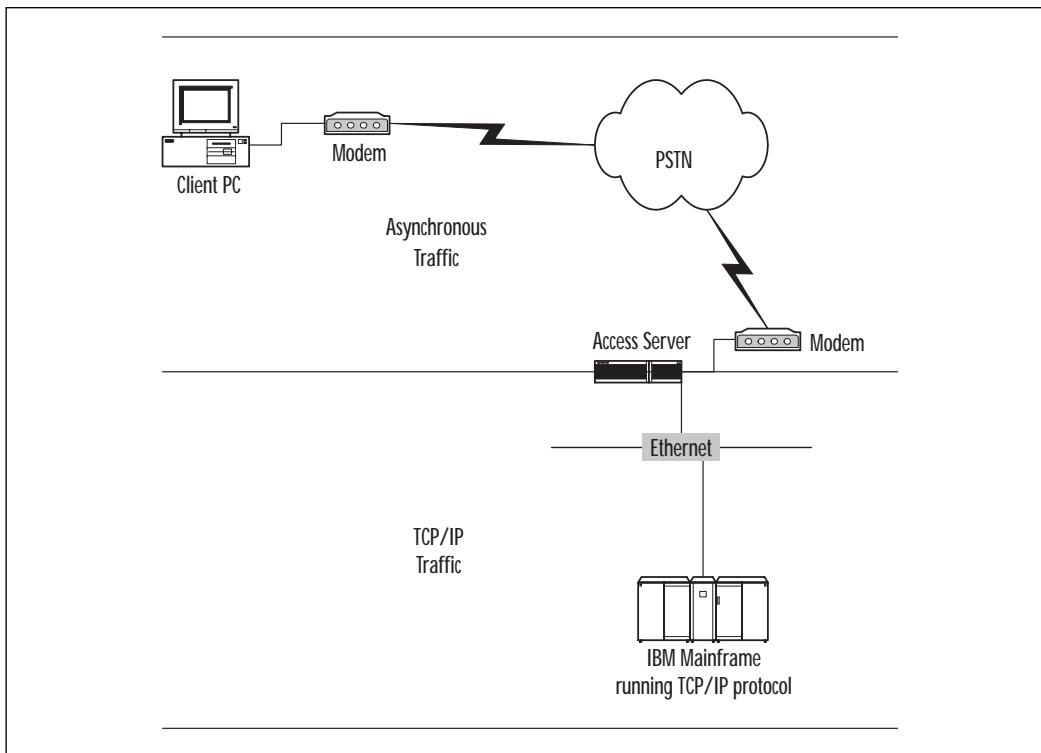
- IBM 3278-2 terminal with an 80-by-24 display
- IBM 3278-2 terminal with a 24-by-80 display
- IBM 3278-3 terminal with a 32-by-80 display
- IBM 3278-4 terminal with a 48-by-80 display
- IBM 3278-5 terminal with a 27-by-132 display

IBM terminals use a character format referred to as extended binary-coded decimal interchange code, or EBCDIC. EBCDIC consists of eight-bit coded characters and was developed by IBM. TN3270 emulation is made possible by the use of a protocol called *termcap*. Termcap functions translate the keyboard and terminal characteristics of a standard ASCII terminal into those functions required by IBM hosts. Termcap is a two-part terminal-handling mechanism. The first part of termcap consists of a database of terminals. This database outlines the capabilities of each supported terminal. The second part of termcap consists of a subroutine library. This library allows programs to query the database and make use of the values it contains. Cisco IOS has a default termcap database for the Digital VT100 terminal emulation. Additional entries can be made into the termcap database as well. This is done through the use of the **keymap** and **TTYcap** commands.

The **keymap** and **TTYcap** commands create entries that translate non-IBM terminal commands to functions to IBM commands or functions. With keymapping, terminals send a key sequence for every key used to send packets to an IBM host. The keymapping function in the Cisco IOS identifies special sequences and converts them to directives to the IBM host. A minimal level of keymapping is supported by default and it is important to note that several keys can convert to the same IBM directives.

With **TTYcap**, the IBM host sends commands to the terminal, including cursor position, clear screen, and so forth. The TTYcap functionality in the Cisco IOS software changes IBM directives into the terminal language. By default, protocol translation on access servers and routers conforms to the ANSI terminal standard, which is VTxxx terminal compatible.

Figure 2.5 TN3270 Service from a Cisco Router.



The Autocommand Feature

Cisco routers support the automation of tasks that are associated with terminal lines. This is done through the use of the *autocommand* option. Autocommand allows the execution of any EXEC mode command when a connection is established to a terminal line. This is convenient when you want to control the operating characteristics of a dial-in modem.

For example, if you want to have users dial in to an access server and connect to a UNIX host, user intervention can be averted and the session will automatically be initiated to the UNIX host. In the following example, the use of the autocommand feature will establish a session to a UNIX host with an IP address of 192.168.1.1

```
line vty 129
autocommand connect 192.168.1.1
```

The same principle can be applied using the protocol translation feature discussed earlier. Remember, the autocommand feature can issue any

EXEC command, not just Telnet sessions. So you could configure the line to establish a connection to a LAT host named Central using the following example:

```
line vty 129  
autocommand lat central
```

You could also configure the autocommand feature for remote support for technical staff. If you wanted the staff to be able to dial in and view the TCP/IP routing table, you could use the autocommand feature to automate this process as well, as illustrated in the following example:

```
line vty 129  
autocommand show ip route
```

Menus

Menus can be configured within Cisco IOS to provide users connecting to a router with an easy-to-use interface. This is helpful because users do not need to learn the underlying command syntax to accomplish basic tasks. The following is an example of a basic menu that users can utilize to access network services.

```
Welcome to the Corporate Network
```

```
Type a number to select an option;
```

```
Type 9 to exit the menu.
```

- | | |
|---|---------------------------------------|
| 1 | Connect to VMS (LAT) |
| 2 | Connect to the IBM Mainframe (TN3270) |
| 3 | Read E-Mail |
| 4 | Start PPP |

```
Exit the Menu
```

When users connect to this router, this is the menu they will see. The following is the command structure for the menu you just created:

```
menu Basic title ^C
Welcome to the Corporate Network

Type a number to select an option;
Type 9 to exit the menu.^C
menu Basic text 1 Connect to VMS (LAT)
menu Basic command 1 LAT CENTRAL
menu Basic text 2 Connect to the IBM Mainframe (TN3270)
menu Basic command 2 tn3270 mainframe
menu Basic text 3 Read E-Mail
menu Basic command 3 telnet mail.corp.com
menu Basic text 4 Start PPP
menu Basic command 4 ppp
menu Basic text 9 Exit the Menu
menu Basic command 9 exit
menu Basic clear-screen
menu Basic default 3
```

Menus have a basic command structure that we will now examine. Menus can have a title that is displayed when the menu starts. The syntax to create a title for a menu is similar to the syntax used to create a login banner. The base command is **menu name title delimiter**. The delimiter is important in that it is the ASCII character the router will use to signify the end of the character string used for the title. Typically you would not want to use a standard letter, because that letter may appear in the text you enter. Using a rarely-used character such as a tilde (~) can save you quite a bit of frustration.

To create the entries the users will see when the menu is executed, you use the **menu name text item text** format. The item is the number that you want to appear next to the text. This number is the number that the users will use to invoke that particular selection. It is important to note that menus can only have 18 entries, but Cisco has built in the ability to create sub-menus. We'll cover sub-menus later in this section.

Now that you have your entries created, you need to configure the commands that will be executed when a user picks a menu option. To do this, you use the format of **menu name command item text**. The item is the number of the command we want to use, while the text is the actual command executed. It is important to note that the value placed in the text

portion matches exactly to the command a user would enter if they were connected to the router with no menu system.

You also have some additional controls over the way a menu is displayed and operates. Commands such as **menu title clear-screen** make the router insert 24 new lines, which effectively clears the screen. It is important to note that the menu system default is a standard “dumb” terminal that only displays text in a 24-line-by-80-column format.

You saw earlier that menus can contain sub-menus because any given menu can contain only 18 entries. With the use of sub-menus, a very complex and feature-rich menu system can be created. The following example builds on the previous one; the menu now has an option for support personnel and calls an additional menu with support functions.

You are now in the Support Menu.

Use is restricted to authorized personnel only.

- 1 Show IP Routing Table
- 2 Show CDP Neighbors
- 3 Show Users Logged In
- 4 Show Frame Relay Maps
- 9 Exit Menu

As you can see, you now have a new menu named Support. The Support menu has a different title and menu options than the previous menu. It is important to note that *all* menus should have an exit menu option—otherwise, you can get stuck in a menu loop with no way to exit.

EXEC Callback

Cisco has incorporated a function within IOS called EXEC callback. EXEC callback can be an important part of a dial-in terminal service configuration. The idea behind EXEC callback is that a user can initiate a call to an access server, enter the phone number from which they are calling, and have the access server call them back. This can be useful for a number of reasons—including security, toll avoidance, and centralized billing.

There are two main types of EXEC callback: EXEC callback with verification and EXEC callback with no verification. EXEC callback with verifi-

cation is typically implemented for mobile users such as a roving sales force, or any user who calls into the network from different locations. EXEC callback without verification is typically used for home users dialing into the network from the same phone number every time.

The following is a list of the steps used when using EXEC callback with no verification:

1. A user at a remote site calls into the access server and authenticates.
2. The user is disconnected.
3. The access server calls the user back at the pre-configured number and starts an EXEC session.

The following is a list of the steps used when using EXEC callback with verification:

1. A user at a remote site calls into the access server and authenticates.
2. The user enters a telephone number to receive the call back.
3. The user is disconnected.
4. The access server calls the user back at the pre-configured number and starts an EXEC session.

The only additional step between the two configurations is the user entering the remote callback number.

The following is a sample configuration of EXEC callback with both forms of verification:

Current configuration:

```
!
version 12.0
service exec-callback
!
hostname Central
!
!
username homer nocallback-verify callback-dialstring 5551212 password 0
cisco
username mobile callback-dialstring "" password 0 cisco
!
```

```
!  
chat-script offhook " " "ATH1" OK  
chat-script dial ABORT ERROR ABORT BUSY " " "ATZ" OK "ATDT \T" TIMEOUT  
30 CONNEC  
T \c  
!  
!  
line aux 0  
script modem-off-hook offhook  
script callback dial  
login local  
modem InOut  
transport input all  
callback forced-wait 5  
speed 115200  
flowcontrol hardware  
!  
!
```

You have configured two users, home and mobile. The home-based user always calls in from the same location. You have the callback string configured as 555-1212 and you can see the password the home user must enter to authenticate to the access server. The mobile user must enter the phone number they are calling from when they dial in to the access server; this indicates where to call them back.

You also have a chat script configured to initialize the modem, pick up the phone line, and call the phone number. This number will vary from user to user, depending on the particular user's configuration. You configured the access server to wait five seconds before calling the user back as well using the **callback forced-wait 5** command on the line configuration.

Here is a view of what users would see when they dial into the access server without callback verification.

```
ats0=1  
!-- AT command to set modem to autoanswer mode.  
!  
OK  
!-- Modem accepts command.
```

```
!
atdt 5551111
!-- AT command to pass dial string to the modem.
!
CONNECT

username: homer
password:

Callback initiated - line is disconnected
```

NO CARRIER

RING

CONNECT

Central>

Here is a view of what users would see when they dial into the access server with verification of the callback number.

```
ats0=1
!-- AT command to set modem to autoanswer mode.
!
OK
!-- Modem accepts command.
!
atdt 5551111
!-- AT command to pass dial string to the modem.
!
CONNECT
```

Username: mobile
password:

```
Callback Dialstring: 5554444  
Callback initiated - line is disconnected
```

```
NO CARRIER
```

```
RING
```

```
CONNECT
```

```
Username: mobile  
password:  
Central>
```

Summary

This chapter illustrates many of the functions that Cisco access servers can provide. Cisco access servers can become valuable tools for providing remote access terminal services to your user community. Cisco access servers are capable of supporting both dial-in and dial-out services.

Access servers have many features to automate the configuration of the access server, such as the autoconfigure and autodiscovery features. Autoconfigure uses the internal modemcap database to automatically configure a modem with the most commonly used initialization strings for a modem. Autodiscovery dynamically determines the model of modem connected to the access server and uses the appropriate modemcap database entry to configure the modem. The modemcap database can be modified to add your particular brand of modem if it is not listed by default. Chat scripts also ease the administration of an access server, by allowing a pre-defined series of actions to be taken when the appropriate prerequisites are met. Tools like autoconfigure, autodiscovery, and chat scripts are some of the features that make Cisco access servers such a valuable addition to any network.

Access servers can also provide legacy terminal services to systems on your network. Services such as Telnet and rlogin for UNIX-based systems provide an comprehensive system for remote access to UNIX systems. Access servers also support Digital Equipment Corporation's (DEC) LAT services. An access server can allow TCP/IP-based systems access to non-routable LAT services, easing network administration and minimizing LAT traffic on your network. Access servers can also provide TN3270 terminal

emulation to IBM mainframes for non-TN3270 devices. This is a great asset for any company that wants to provide remote TN3270 access without the need for additional software on the remote systems.

Cisco must have had a network administrator's job in mind when they created the strong menu system for access servers. This powerful system can automate the commands that users accessing the network need to learn to do their job. Users can dial into the network and navigate the easy-to-use menu system to access systems to perform their job. The auto-command feature can also be used to ease user training for remote network access.

Finally, to address security, Cisco access servers can be configured to use the EXEC callback feature to provide secure remote access. This feature can be configured for fixed, secure dial-back numbers or allow enhanced security for a mobile work force.

FAQs

Q: How many modems will an access server support?

A: The answer to this question depends on the model of access server implemented. The base access servers, such as the 2509 and 2510, support eight asynchronous interfaces, plus the AUX port. Access servers such as the 2511 can support 16 asynchronous interfaces, while the 2600, 3600, AS 5200, and AS 5300 can support a larger quantity of modems. The decision of what access server to use really depends on the needs of your network.

Q: What can I do if my modem is not listed in the modemcap database?

A: In this scenario, the modemcap database can be expanded to include the required parameters for your particular model of modem.

Q: Why would I want to use protocol translation?

A: The implementation of protocol translation is beneficial when you do not want to support a number of different terminal emulation programs on your remote clients. By using protocol translation, you can avoid having non-routable protocols such as LAT and SNA bridged across your internetwork.

Using PPP to Provide Remote Network Access

Solutions in this chapter:

- Point-to-Point Protocol (PPP) overview
- Configuring PPP
- Password Authentication Protocol (PAP) and Challenge Handshake Authentication Protocol (CHAP)
- Multilink PPP (MP)
- Multichassis Multilink PPP (MMP)
- Microsoft Windows Access

Introduction

Providing remote access as part of an organization's network infrastructure is becoming a common requirement today. Traveling salesmen, telecommuters, and remote offices all need to gain access to corporate network services, so they must be able to connect onto a network.

In the previous chapter, we looked at how to provide asynchronous connections to the central site. In this chapter, we will look at how to use that dial-up connection to connect to the actual network using Point-to-Point Protocol (PPP) encapsulation.

PPP encapsulates network layer protocol information (including, but not limited to, Internet Protocol, or IP) over point-to-point links. We will look at how this protocol works, and we will also look at the Link Control Protocol (LCP) mechanisms for establishing, configuring, and testing the data-link connection. We will also focus on how to control access to the network by using the authentication methods used by PPP—the Password Authentication Protocol (PAP), and the Challenge Handshake Authentication Protocol (CHAP).

In the final section of this chapter, we will look at how to configure Microsoft Windows clients to access the central site. This will include an overview of the dial-up networking implementation in various Microsoft Windows clients.

PPP Overview

PPP is one of the most popular and cost-effective methods of giving users remote access to corporate intranets and/or the Internet. Businesses and Internet service providers (ISPs) prefer giving their users dial-in or dedicated line access using PPP because of several key factors that will be covered in this chapter, including scalability, operability, and reliability.

PPP is an OSI Layer 2 protocol standard that allows two computing devices to communicate with each other using point-to-point connections such as an analog phone line, an integrated services digital network (ISDN) line, or a serial link. These point-to-point connections can be client-to-network or router-to-router.

The physical media that can be used to transport PPP includes unshielded twisted-pair (UTP), fiber optic, and wave transmissions such as satellite systems. PPP is a full-duplex protocol unconcerned with transmission rates, which can be used with either synchronous or asynchronous communication lines.

PPP can be used to encapsulate popular network protocols such as IP (the Internet standard) and Internetwork Packet Exchange, (IPX, Novell's

native standard). This encapsulation is done by placing the Open System Interconnection (OSI) Layer 3 IP packet inside the PPP OSI Layer 2 frame and sending it down the transmission media to the other side where the PPP encapsulation frame is stripped away. The Layer 3 IP packet is then passed up to the next layer of the protocol stack.

There are four ways PPP can be used as a data-link layer protocol on a Cisco router to provide access to computing resources:

- To provide dial-in access to remote users
- To provide backup services over an asynchronous or synchronous connection in case a circuit fails between two routers
- To provide encapsulation between two routers over a leased line
- To provide dial-on-demand routing (DDR) services between two routers

PPP Features

PPP offers several features that add the benefits of efficiency, security, and reliability to communications links.

Multiple Protocols per Communication Line

PPP allows multiple network protocols (such as IP, IPX, DECnet, Vines, or AppleTalk) to run over the same communications link. Each network protocol is transported by use of an additional associated Network Control Protocol (NCP). For example, IP uses the IP Control Protocol (IPCP) and IPX uses the Internet Packet Exchange Control Protocol (IPXCP) as their respective NCPs.

Authentication

Security can be implemented over the link by the use of an authentication protocol such as PAP, CHAP, or Microsoft's MS-CHAP. These protocols will be explained later in this chapter.

Link Configuration and Negotiation

Link layer parameters (such as the use of special escape characters and a maximum frame size) add flexibility and reliability to the communications link.

Error Detection

Transmission errors can be detected through the use of Frame Check Sequence (FCS) fields in the PPP frame.

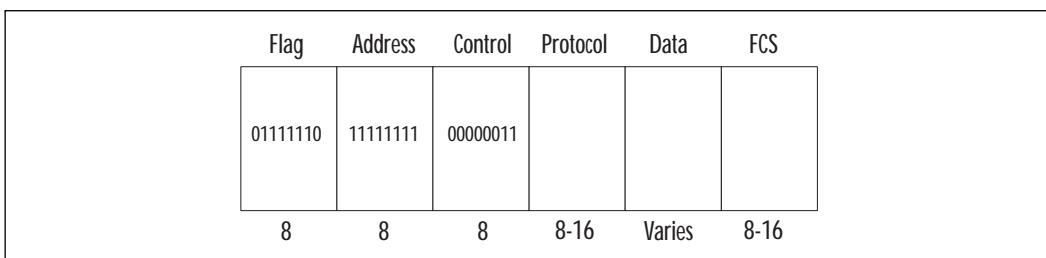
Header Compression

PPP allows for the compression of packet headers to more efficiently utilize link bandwidth by reducing transmission overhead.

Bonding of Communications Links

PPP allows multiple communications links and/or remote access servers to be “bonded,” to increase the amount of bandwidth between end devices. This “bonding” action allows two physical communications lines to appear as a single virtual link for remote access services.

Figure 3.1 PPP frame format.



The PPP frame consists of the following six fields, as illustrated in Figure 3.1:

- **Flag**—(8 bits) start of frame consisting of the value 01111110
- **Address**—(8 bits) broadcast address consisting of the value 11111111
- **Control**—(8 bits) transmission control field consisting of the value 00000011
- **Protocol**—(8-16 bits) identifies network protocol encapsulated within frame
- **Data**—(Variable length) frame payload (maximum size is 1500 bytes)
- **FCS**—(8-16 bits) frame check sequence for error detection. By prior agreement, consenting PPP applications can use 4 bytes for greater error detection

There are several components that make up the point-to-point protocol. Each of these component sublayers executes specific tasks that enable PPP to exhibit its many capabilities while remaining a stable and robust link-layer protocol.

LCP

The LCP sits on top of the physical layer and establishes, authenticates, and tests the functionality of the data-link connection through a four-phase process:

- **Phase 1** LCP sets up a data-link connection and negotiates configuration parameters
- **Phase 2** LCP determines sufficiency of link quality (this phase is optional)
- **Phase 3** LCP sets up a network layer connection and configuration
- **Phase 4** LCP tears down the connection and notifies network layer of the status

There are three types of LCP frames that correspond with each mandatory phase of the LCP process:

- **Link configuration** to set up a data-link connection
- **Link management** to maintain and debug a connection
- **Link termination** to tear down a connection

When two LCP peers initiate the negotiation process, they use their unique LCP parameters to either accept or reject each other's unique LCP option values. LCP peers do this by sending any of the following responses to an initial configuration request:

- **Configure-NACK** due to unacceptable values
- **Configure-Reject** because some or all values are unknown
- **Configure-ACK** because all of the values are within accepted parameters

Where LCP configuration options are not included in the configuration request packet, the default value for those options are used.

When a Configure-NACK or Configure-Reject is received as a configuration response, the values are modified until they are within acceptable limits. At that time, a Configure-ACK is returned to the requestor.

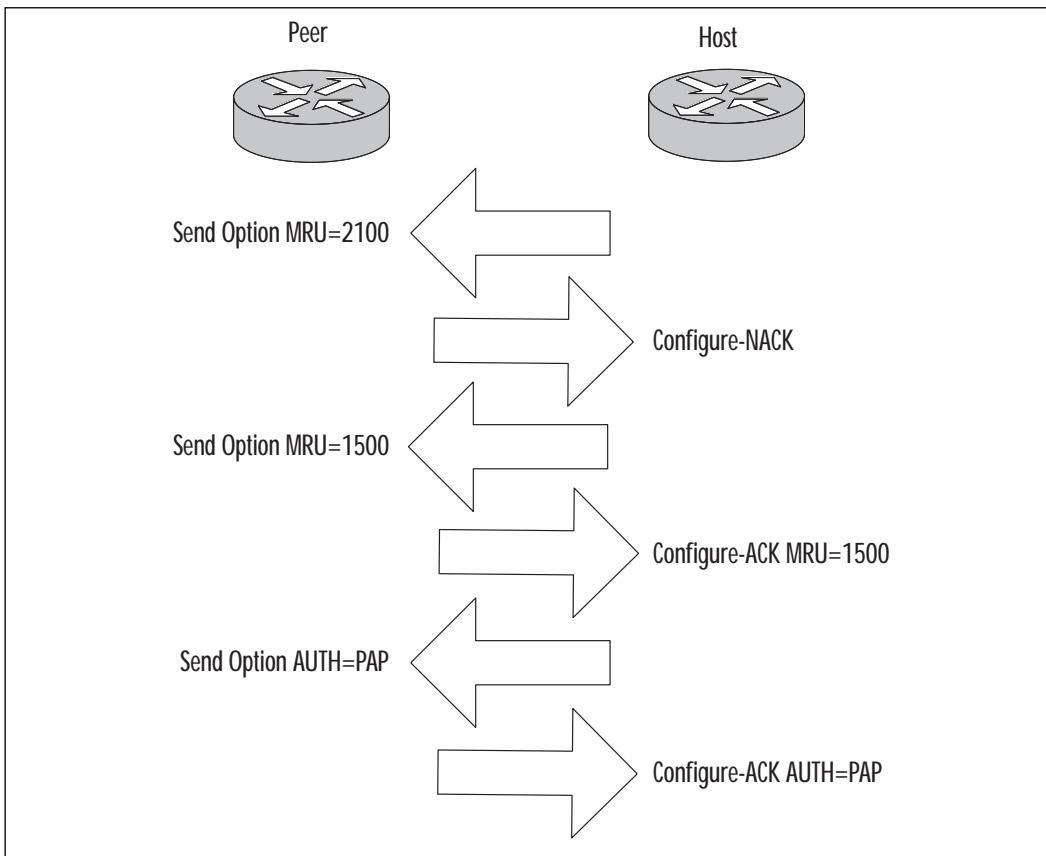
Two of the most important parts of the LCP process are the negotiation of the Maximum Receive Unit (MRU) parameter and the authentication of peers (see Figure 3.2).

The MRU parameter limits the size of packets and determines the overall bandwidth of the communications link. The MRU can be different sizes in either direction, or the same size in both directions. This process is

completed by the configuration request responses mentioned in the previous list of LCP acknowledgements.

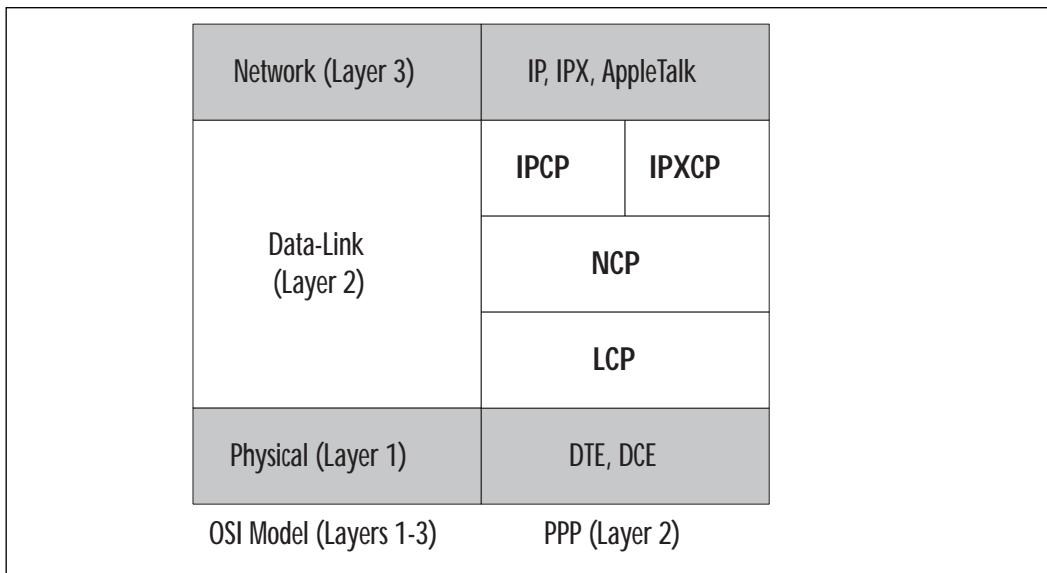
LCP authenticates point-to-point peers by using either PAP or CHAP. Which authentication protocol that LCP uses is configurable by the user. MS-CHAP is an authentication protocol proprietary to Microsoft that is also supported by Cisco. These three authentication protocols will be discussed later in this chapter.

Figure 3.2 LCP negotiation of MRU and authentication values.



Once LCP has established the data-link layer for the connection, the responsibility for setting up the network layer is passed up to the NCP.

Figure 3.3 Layers of PPP.



NCP

The NCP resides on top of the LCP, and is responsible for establishing and configuring network layer protocols such as IP, IPX, and AppleTalk (see Figure 3.3). NCP can also signal LCP to terminate the communications link when necessary.

NCP uses the IPCP to manage the use of IP over the communications link. IPCP allows the Dynamic Host Configuration Protocol (DHCP) to be used for IP address assignment to the remote peer (RFC 1332). NCP uses IPXCP. This permits negotiation of the routing protocol and compressed IPX (RFC 1552, RFC 1553).

PPP vs. SLIP and ARAP

When connection to the Internet using Windows- or Macintosh-based computer systems first became popular, the two choices that users had were Serial Line Internet Protocol (SLIP) and AppleTalk Remote Access Protocol (ARAP). These two protocols allowed users to exchange IP packets of data with remote computing systems, and represented an alternative to the straight ASCII text characters that were exchanged between remote terminals and mainframe computing systems.

The ability to send IP packets instead of character text allowed remote users to run a number of applications concurrently, or to have several “virtual” connections due to the various transport layer (OSI Layer 4) ports that could be used.

While SLIP and ARAP advanced remote connectivity, they had many shortcomings that needed to be addressed in order to support robust applications between distant endpoints. Enter the PPP protocol.

PPP provides the ability to sustain several virtual connections over a single line, and provides a number of other benefits lacking in SLIP and/or ARAP:

- PPP provides error checking, whereas SLIP does not.
- SLIP supports only the IP protocol (it lacks a protocol identifier field); ARAP supports only the AppleTalk protocol, whereas PPP supports several others including IP, IPX, AppleTalk, and NetBIOS.
- PPP can share a communications line with other devices; SLIP and ARAP allow only a single remote machine to connect over a single communications line.
- ARAP does not support routing as do PPP and SLIP.
- PPP is simple to configure on either end device.

Because of these differences, and because PPP offers superior scalability, operability, and reliability, PPP has become the de facto standard protocol for remote access networks.

Relevant RFCs

- RFC 1661–Point-to-Point Protocol (PPP)
- RFC 1332–PPP Internet Protocol Control Protocol (IPCP)
- RFC 1333–PPP Link Quality Monitoring
- RFC 1334–PPP Authentication Protocols (PAP)
- RFC 1378–PPP AppleTalk Control Protocol (ATCP)
- RFC 1552–PPP Internet Protocol Exchange (IPXCP)
- RFC 1553–PPP Compressed IPX (CIPX)
- RFC 1570–PPP LCP Extensions
- RFC 1990–PPP Multilink Protocol (MP)
- RFC 1994–PPP Challenge Handshake Protocol (CHAP)

You can read the relevant RFCs by using the search engine located at:

www.rfc-editor.org/rfcsearch.html

Configuring PPP

Configuring PPP on a Cisco router involves the following steps:

1. Configuring Cisco parameters necessary to communicate with a third-party device such as an ISDN switch.
2. Entering global configuration commands to identify the Cisco device and to implement routing over the established link.
3. Entering interface configuration commands to define the router's interface, determine the encapsulation type, and select the kind of authentication performed over the line.
4. Saving the configuration changes to nonvolatile RAM (NVRAM).

TIP

When working on a Cisco router for the first time, always use the **show version** command to verify the Cisco IOS version number, and check the Cisco website (www.cisco.com) for any known bugs in that particular version of IOS. “Interesting traffic,” as referenced in the configuration example below, is defined by access lists as traffic that you want to initiate/maintain or transport across an ISDN or other DDR link.

To configure IP over PPP on an ISDN interface on a Cisco router, follow these steps:

1. Enter the enable mode so that the configuration of the router can be changed. [**enable**]
2. Enter the global configuration mode. [**config terminal**]
3. Select the ISDN switch type of your ISDN provider.
[**isdn switch-type switch-type**]
4. Enter the remote router host name and password.
[**username remote password pwd**]
5. Configure a dialer list of interesting traffic.
[**dialer-list number protocol ip permit**]
6. Enter a static route to host end router.
[**ip route subnet mask next-hop-address**]
7. Enter the interface configuration mode. [**interface bri number**]

8. Assign an IP address. [**ip address address mask**]
9. Enable PPP. [**encapsulation ppp**]
10. Assign a dialer list to the interface. [**dialer-group number**]
11. Enable CHAP or PAP. [**ppp authentication type**]
12. Map the next hop address. [**dialer map protocol next-hop-address name hostname class classname dialstring**]
13. Return to global configuration mode. [**exit**]
14. Save changes. [**copy running-config startup-config**]

Autoselect

Cisco access routers can automatically allow PPP, ARAP, and SLIP sessions to start when they are requested. This allows the user to be prompted for his username without having to press the “return” key. This can help alleviate any confusion as to the status of the PPP connection to the user during initialization and logon.

To configure a Cisco access server to automatically start PPP sessions when requested, follow these steps:

1. Enter the enable mode. [**enable**]
2. Enter the global configuration mode. [**config t**]
3. Enter the line configuration mode. [**line line-number**]
4. Enable autoselect. [**autoselect ppp during-login**]

PPP Addressing Methods

The local interface of the Cisco access router can be assigned a network address for the IP protocol in one of two ways:

- Manual assignment—Enter an IP address on the router interface. [**ip address address mask**]
- Use an address from the Ethernet interface to conserve an IP address. [**ip unnumbered interface-type number**]

The local interface can also assign a network address for the IPX protocol in one of two ways:

- Manual assignment—Enter an IPX network number on the router interface. [**ipx network network-number**]

- Associate an asynchronous interface with a loopback address (also involves using IP unnumbered on the interface with the [`ipx ppp-client loopback number`] command). This technique is used to conserve IP address space as the asynchronous interface uses the IP address of the loopback interface. Using unnumbered interfaces is a convenient way to simplify router configuration while saving valuable IP address space for other uses.

NOTE

When “ip unnumbered” is used, the IP address of the loopback interface *does not* have to be on the same subnet as the remote host router being called.

Cisco supports a couple of methods for the assignment of network addresses to remote end-user client computers that dial into Cisco routers and Access Servers:

- **Asynchronous dynamic address** Allows clients to enter in their network address after they enter in the PPP EXEC command. To select this option, use the `async dynamic address` command in interface configuration mode.
- **DHCP** Allows a third-party DHCP server to assign IP addresses to remote clients. To select this option, use the `ip dhcp-server address` command in global configuration mode.

Using the DHCP option seamlessly integrates the user into the IP addressing scheme of the dial-in network and requires no intervention by the user. The `async` option may be necessary when applications are hard-coded to work only with certain IP addresses, or when static addressing is necessary for administrative or security purposes.

Following is an example configuration for a local IP address pool and DNS Service to be assigned to dial-in clients:

To assign the address pool consisting of 253 IP addresses in the range of 10.10.11.2-10.10.11.254, enter the following configuration command:

```
ip local pool pool_name 10.10.11.2 10.10.11.254
```

To assign a primary DNS Service with IP address 10.10.13.254 and a secondary DNS Service with IP address 10.10.13.253, enter the following command:

```
async-bootp dns-server 10.10.13.254 10.10.13.253
```

PPP Link Control Options

As discussed earlier, LCP is responsible for establishing and negotiating the data-link connection. The two most commonly set options are the MRU and the setting that maps the character escape sequences—the Asynchronous Control Character Map (ACCM).

The MRU instructs the PPP peer as to how many High-Level Data Link Control (HDLC) frames to send across the wire (a peer interface must be able to receive frames of up to 1500 bytes in length). Setting the MRU to lower values may aid the performance of interactive applications over the WAN links. Lower MRU values allow for a “quicker send” of smaller packets that are common to interactive applications.

Escape sequences are used to replace special control characters that may appear naturally in the data stream, causing interruption of communication. An example is the XOFF character. Such control characters are replaced with a two-character representation that is unlikely to appear within the data stream. The use of escape sequences prevents the user data being sent from inadvertently interrupting the data flow by appearing as control signals to the computing devices or the protocol in use.

PAP and CHAP Authentication

A common method hackers use to attack computing systems is using software called “war dialers.” A “war dialer” is a software program that continuously dials telephone numbers until a modem picks up at the other end. Once it detects a modem at the other end, it will launch one of a number of attacks attempting to gain access to the computer system. In order to protect remote access networks from these types of attacks, some means of security needs to be provided that can perform authentication before access is given to the network.

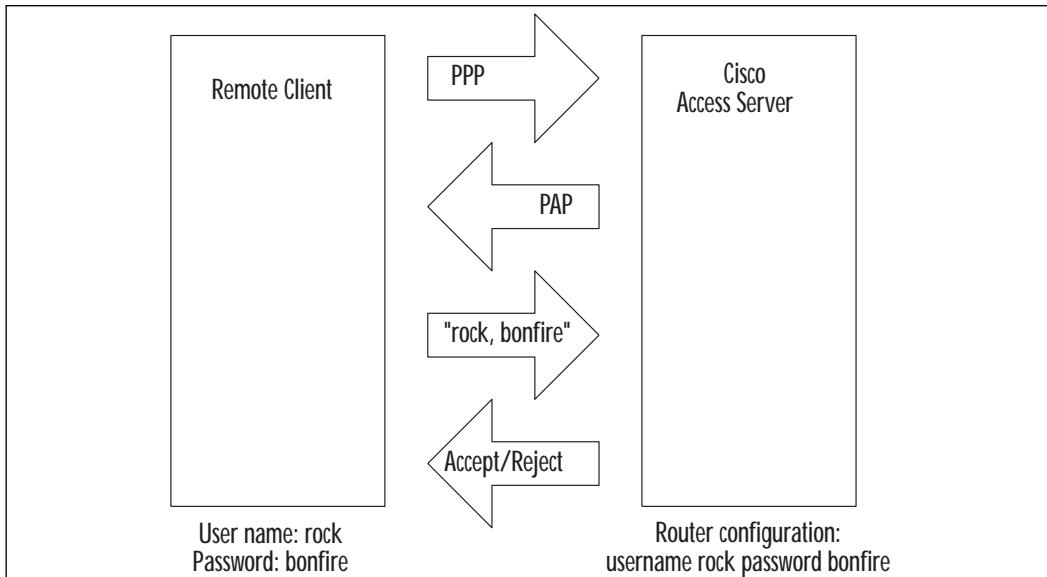
PPP provides several types of authentication methods to enhance the security of providing remote access over publicly accessible communication lines. These authentication protocols need to work at a layer lower than the network layer, to avoid the passing out of IP addresses to unknown systems that may attempt a connection to the network.

PAP and CHAP work at the LCP layer of PPP. CHAP is the more secure of the two-link-layer authentication protocols, and is quickly becoming standard.

PAP

Both the peer (the client requesting access) and the authenticator (the access server) must be configured for PAP authentication, and a matching

Figure 3.4 PAP authentication.



set of ID/passwords must be entered in both the peer and the authenticator's configuration.

First, the link establishment phase is completed. The peer and authenticator send LCP packets to each other until framing is agreed upon and the link is established.

Once the PPP link has been established, the authentication phase begins, in which the peer repeatedly sends its ID/password in clear text to the authenticator until the authentication is validated or the connection is terminated.

The authenticator validates the ID/password by checking for a match of the ID/password in its authentication list. See Figure 3.4 for an illustration of the authentication process.

Because PAP sends the password across the link in plain text and is vulnerable to “playback” and repeated heuristic hacking attempts, it is considered a low measure of security.

Figure 3.5 illustrates relevant PAP configuration commands of two routers that are configured for PAP authentication using PPP.

CHAP

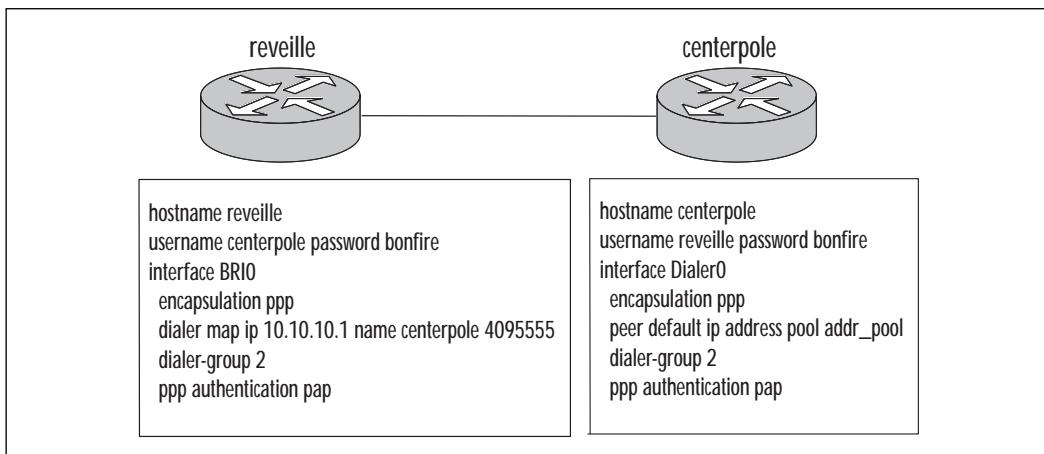
CHAP works without having to send the authentication password over the communications link. As with PAP, the link establishment phase is completed before the authentication phase begins.

For IT Professionals**What's in a name?**

Make sure that when you configure the “username” command line in each router, you use the host name of the opposite router as the username. This is a common mistake made by even the most seasoned Cisco professionals. The passwords must be identical. The format should be as follows:

```
username other-router-host password same4both
```

Figure 3.5 Example PAP configurations.



The authenticator instructs the other end to use CHAP for authentication. The calling peer then requests a challenge.

The authenticator issues the CHAP verification “challenge” to the peer in the form of a random selection (like a number) that is encrypted using its ID/password. The peer in turn uses its password to encrypt the challenge using a “one-way hash,” and sends the encrypted result back to the authenticator.

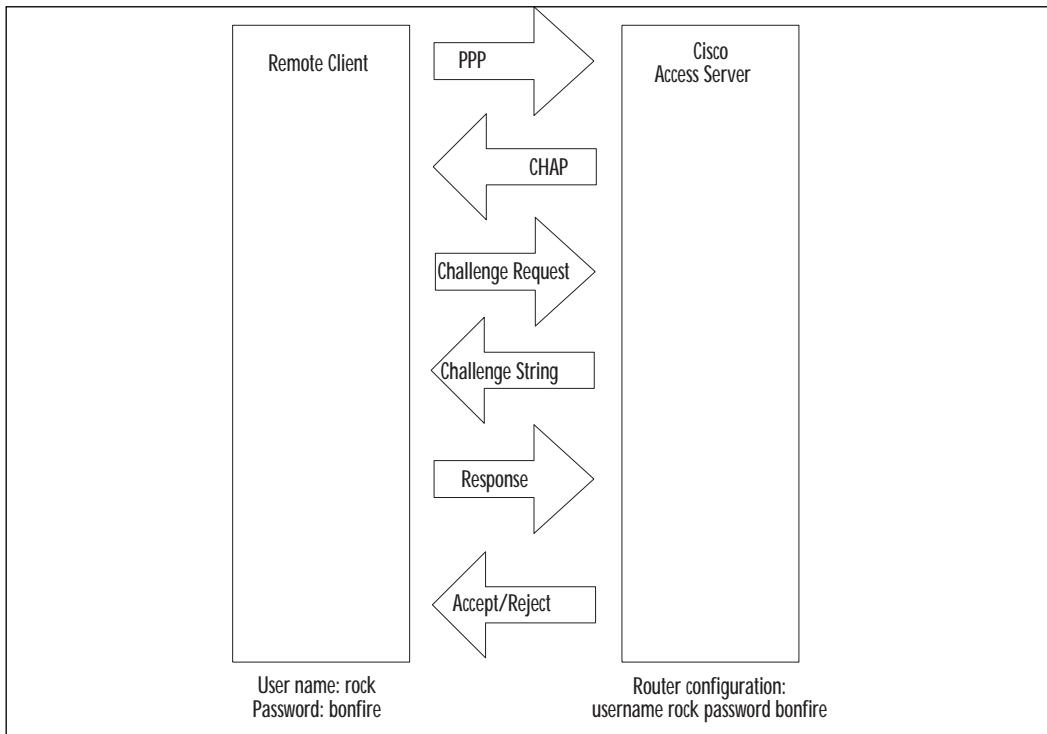
The authenticator authenticates the received response and establishes the authenticated connection if the challenge was validated. If the challenge fails, the connection is rejected.

Because a failed challenge has its connection terminated, CHAP is not vulnerable to “brute force” attacks like PAP is.

Both the calling peer and the called peer must be configured to use either CHAP or PAP, or the connection will be rejected. A peer configured to use PAP cannot authenticate to an authenticator that is configured only to use CHAP.

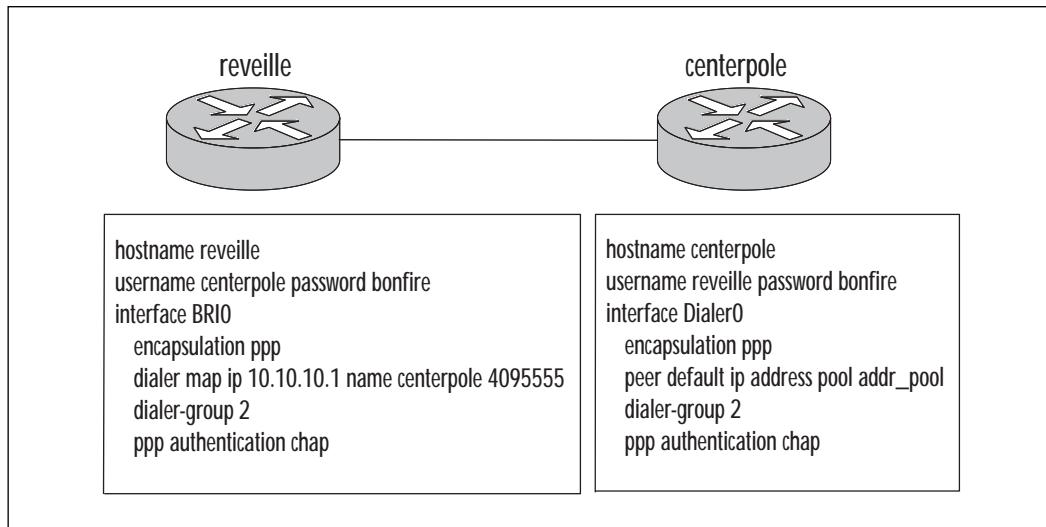
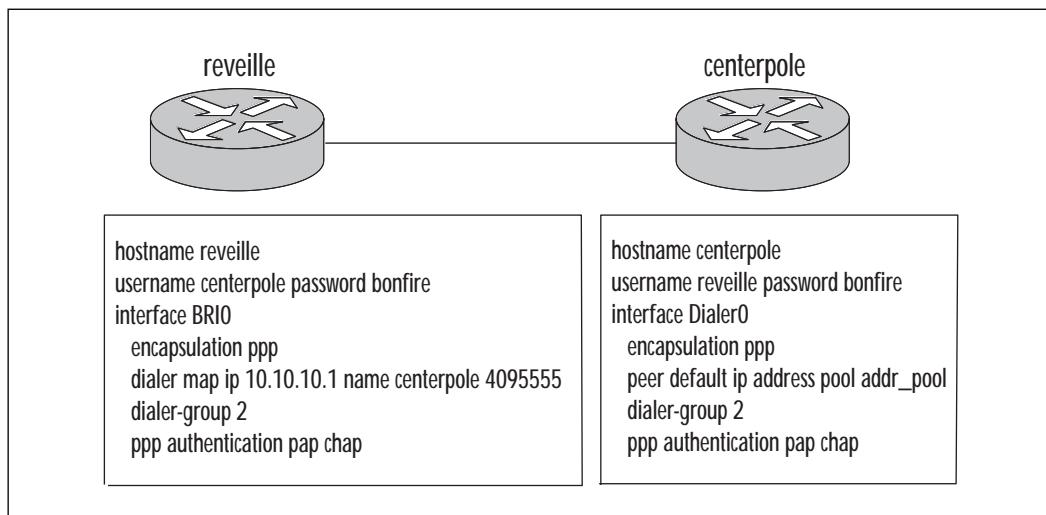
Figures 3.6, 3.7, and 3.8 show relevant CHAP configuration commands of two routers that are configured for CHAP authentication using PPP.

Figure 3.6 CHAP authentication.



TIP

This example tries PAP authentication first; if that fails, it will next try CHAP. To configure MS-CHAP, use `ppp authentication ms-chap`.

Figure 3.7 Example of CHAP configurations.**Figure 3.8** Example using both PAP and CHAP.

Authentication Failures

Most PAP and CHAP authentication failures using Cisco equipment are due to either the appropriate authentication protocol not being configured on both ends of the PPP link, or the wrong ID/password being configured on the “username” line.

The Cisco username configuration line has the format of:

```
username other_end_hostname password same_password_4both
```

When troubleshooting PPP authentication failures use either the debug ppp pap or debug ppp chap command to aid in determining the configuration error. These commands are covered later in this chapter.

PPP Callback

PPP Callback is used to enhance the security of a remote access network by verifying the phone number of the initiating client through returning the phone call. It can also be used to reverse phone charges so that billing can be managed from a single hub site.

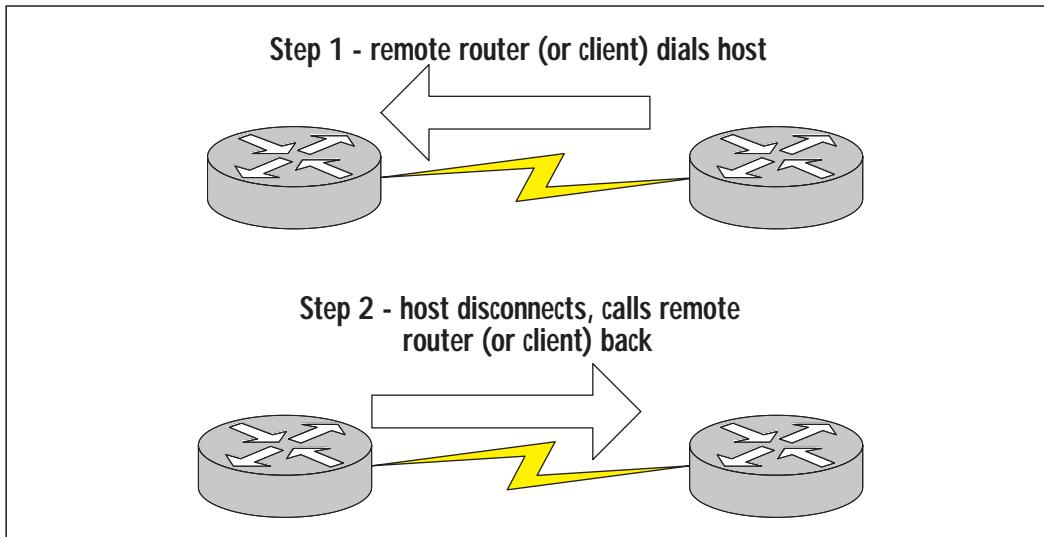
With PPP Callback, the initiating client dials into the host router and passes authentication information to it (such as the host name and dialer string). The host router returns the call if the information is authenticated (Figure 3.9).

PPP Callback must be configured on both the initiating client and the host router, with the client being configured to make PPP callback requests and the host router being configured to accept and return authenticated callback requests.

Microsoft operating systems (Windows NT, Windows 2000) use their own version of callback based on the proprietary Microsoft Callback Control Protocol called MS Callback (MSCB). MSCB has the following restrictions:

- Only supports IP
- Can be used only on Public Switched Telephone Network (PSTN) or ISDN lines
- Both ends must use PAP or CHAP authentication

PPP Callback can also be configured between a Cisco access router and a personal computer running Microsoft Windows utilizing MSCB. MSCB is enabled by default in Cisco IOS 11.3(2)T and later when PPP Callback is enabled. If a participating router is not configured for callback, the connection will not be successful.

Figure 3.9 PPP callback process.

Configuring PPP between two Cisco routers is straightforward. To configure the host router as the call back server, do the following:

1. Enter the enable mode. [**enable**]
2. Enter the global configuration mode. [**config terminal**]
3. Enter the interface configuration mode.
[**interface type number**]
4. Enable DDR. [**dialer in-band**]
5. Enable PPP. [**encapsulation ppp**]
6. Enable CHAP or PAP. [**ppp authentication type**]
7. Map the next hop address.
[**dialer map protocol next-hop-address name hostname class classname dialstring**]
8. Set interface to accept callback. [**ppp callback accept**]
9. Return to global configuration mode. [**exit**]
10. Configure PPP dialer map class. [**map-class dialer classname**]
11. Configure dialer map class as callback.
[**dialer callback-server username**]
12. Save changes to memory.
[**copy running-config startup-config**]

To configure a remote router as the callback client, do the following:

1. Enter the enable mode. [**enable**]
2. Enter the global configuration mode. [**config terminal**]
3. Enter the interface configuration mode.
[**interface type number**]
4. Enable dial-on-demand routing. [**dialer in-band**]
5. Enable PPP as the link-layer encapsulation. [**encapsulation ppp**]
6. Enable CHAP or PAP authentication. [**ppp authentication type**]
7. Map the next hop address.
[**dialer map protocol next-hop-address name hostname class classname dialstring**]
8. Set interface to request callback. [**ppp callback request**]
9. Save changes to memory. [**copy running-config startup-config**]

MSCB

The MSCB function provides callback services for Microsoft Windows' client computers using Microsoft's proprietary protocol, MSCB. If you configure a Cisco router running IOS version 11.3(2)T or later, MSCB is enabled by default and no additional configuration is necessary.

PPP Compression

PPP Compression is used to minimize the utilized bandwidth across the link. Payload data within a PPP packet can be compressed by two methods supported by Cisco:

- **Stacker** Compresses each data type once and then determines where each occurs.
- **Predictor** Examines the data to see if it has previously been compressed, to avoid attempting to compress data that is already compressed.

MPPC

Microsoft Point-to-Point Compression (MPPC) compresses PPP packets between Cisco access servers and Microsoft clients such as Windows 9x, Windows NT, and Windows 2000. Such compression optimizes bandwidth between the two end devices.

Compression Effects

Be sure to check the effects of enabling compression on your equipment, as compression can be central processor unit (CPU) and memory intensive. Typically, compression will result in about a 2:1 reduction in payload size.

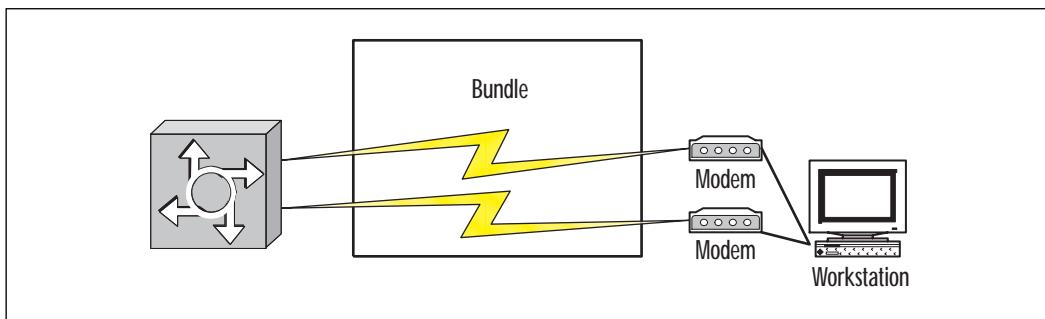
For more information on PPP compression, please see Chapter 9, “Optimizing Network Performance with Queuing and Compression.”

Multilink PPP

Multilink PPP (MP) allows multiple communications lines to be bound together in a “bundle” between one to two remote peers (Figure 3.10). For example: two 56 Kbps links can be bound together to form a single logical link with a bandwidth of 112 Kbps.

Packets are fragmented at the origination end and sent over the multiple links at the same time to the remote end. When they arrive at the remote end, the packets are reassembled, resequenced, and sent on to their destination. (See RFC 1717 for more information.)

Figure 3.10 Multilink PPP: two links are bundled together to form one logical connection.



The bandwidth of the logical link has an upper bound of the aggregate bandwidth of each individual physical connection (though the actual aggregation will not be realized as pure data throughput due to link negotiation and protocol overhead).

The individual communication channels do not have to be the same type in order to be bundled. Asynchronous and synchronous lines can be mixed together. For example, four channels can be bound together, with two channels consisting of 56 Kbps modem lines and two channels consisting of two B channels of a Basic Rate Interface (BRI) ISDN line.

In order to implement this feature, both end devices must support MP and have the necessary facilities to build out the bundle. For example, a

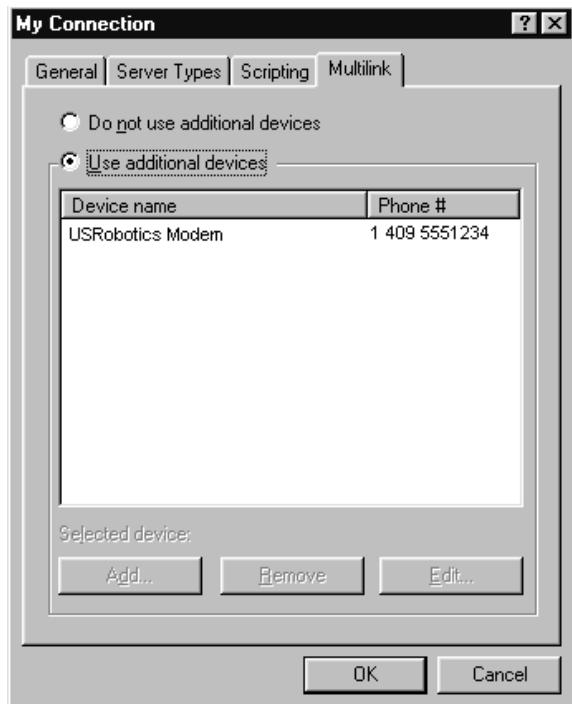
remote user using analog phone lines must have at least two available phone lines and two modems connected to a computer that is configured to support MP (such as Microsoft Windows 9x, Windows NT, or Windows 2000). The other end must also have at least two lines and two ports available and be configured to support MP. Microsoft refers to MP as “bonding” or “MLPPP.”

MP uses the Bandwidth Allocation Control Protocol (BACP) to bind several physical connections into a single logical link. It is initiated when a system sends the Maximum Received Reconstruction Unit (MRRU) option during the first stages of LCP option negotiation. The MRRU LCP option defines the bandwidth of the connection.

MP works by splitting the Layer 2 datagrams on one end, ordering them in a sequence, and sending the datagrams across the several different physical connections of the bundle. When received on the other end, the datagrams are recombined and resequenced before being passed up to the Layer 3 network protocol.

To configure MP using Microsoft Windows 9x Dial-Up Networking (DUN) you must have at least two modems installed and configured on your computer, and do the following (see Figure 3.11 for Windows 98):

Figure 3.11 Configuring MP on Windows 98.



1. Double-click the “My Computer” icon on your desktop.
2. Double-click “Dial-up Networking.”
3. Select the connection you wish to make multilink by right-clicking it.
4. Select “Properties.”
5. In the Properties dialog box, select the “Multilink” tab.
6. Select the “Use additional devices” check box.
7. Highlight the device and click “Add.”

To configure MP on an ISDN BRI using the IP protocol, perform the following configuration tasks in the “enabled” mode:

1. Select the BRI interface. [**interface bri interface_number**]
2. Assign an IP address. [**ip address ip_address mask**]
3. Enable PPP. [**encapsulation ppp**]
4. Specify the dialer load threshold.
[**dialer load-threshold load**]
5. Set up interface to make outbound calls. [**dialer map ip next_hop_address name hostname broadcast**]
6. Select Access-list to control access to the interface.
[**dialer-group group_number**]
7. Select an authentication type. [**ppp authentication type**]
8. Enable Multilink PPP. [**ppp multilink**]

Multichassis Multilink PPP

Multichassis Multilink PPP (MMP) is an extension of MP, in that it allows for a bundle to be split and reconstructed across several different communications lines spanning several different Cisco Access Servers (Figure 3.12). These access servers are combined into a single rotary group that can be accessed via a single phone number. The fact that the different access servers are grouped together is completely transparent to the end user.

This allows corporations and ISPs to publish a single dial-in phone number to automatically distribute user access across all of their bound access servers. Otherwise, users might have to dial a sequence of dial-in numbers until they found an available port—a process that could be time consuming and frustrating.

When multiple Cisco access servers are configured using MMP, the grouping is referred to as a “stack group.” Supported interfaces for MMP are PRI, BRI, Serial, and Asynchronous.

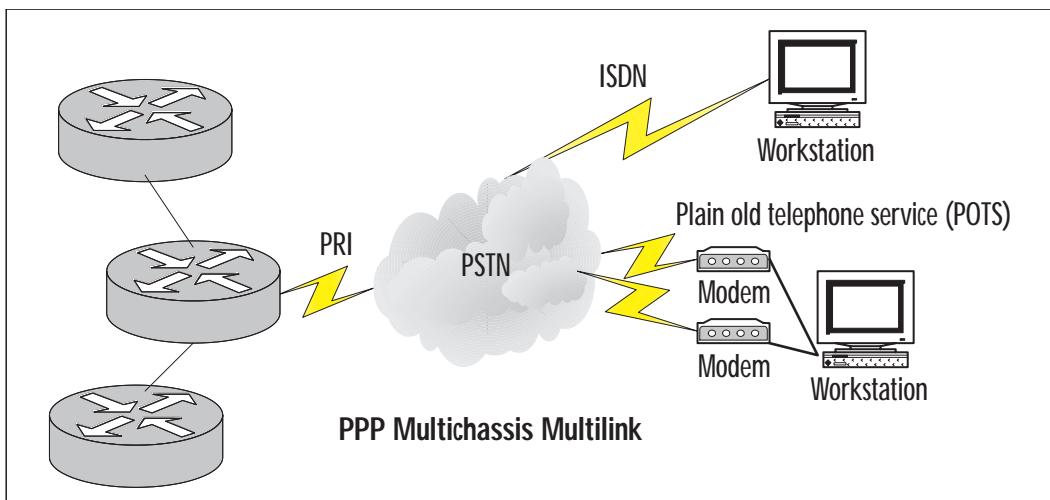
MMP requires that each associated router be configured with the following parameters:

- PPP
- Stack Group Bidding Protocol (SGBP)—A protocol for arbitrating the location of bundles within a stack group to the “highest bidder” (normally the stack group member that locates the initial bundle for the first link in a multilink connection)
- MP
- Virtual template for interface cloning

Simple stack groups are composed of member peer routers and do not need to have a permanent “lead” router. Any stack group member who answers an incoming call becomes the “owner” of the call, if it is the first call in a new session with the particular remote-end device.

When a second call comes in from this same remote-end device to the stack group, the answering router will forward the call to the stack group where the member routers will “bid” for the call. Since the first router “owns” the session by answering the first call, it will win the bid and the answering router will forward the call to it.

Figure 3.12 MMP configuration using routers.



The second router accomplishes this by establishing a tunnel to the “owner” router and forwarding all packets to the owner. The owner router is responsible for reassembling and resequencing the packets. The owner router then forwards these packets on to the local network.

There are two basic steps to configuring MMP on Cisco routers and access servers:

Step 1 Configure the stack group and make member assignments.

1. Create the stack group on the first router to be configured, where “name” is the hostname of that router.

```
[sgbp group group_name]
```

2. Add additional stack group members.

```
[sgbp member router2_hostname router2_ip_address]
```

```
[sgbp member router3_hostname router3_ip_address]
```

```
<add additional sgbp member lines for each additional member router>
```

Step 2 Configure a virtual template and Virtual Template Interface.

1. Create a virtual template for the stack group.

```
[multilink virtual-template template_number]
```

2. Create IP address pool (a local pool is used in this example).

```
[ip local pool default ip_address]
```

3. Create a Virtual Template Interface (not required for ISDN interfaces or if physical interfaces are using dialers).

```
[interface virtual-template template_number]
```

4. Use unnumbered IP addressing.

```
[ip unnumbered ethernet 0]
```

5. Configure PPP.

```
[encapsulation ppp]
```

6. Enable Multilink PPP.

```
[ppp multilink]
```

7. Enable PPP authentication.

```
[ppp authentication type]
```

Verifying and Troubleshooting PPP

Sometimes problems arise when configuring PPP for remote access servers. Cisco provides a very powerful and robust set of commands to aid in isolating problems and solving communication problems. These commands exist in two different command sets: *show commands* and *debug commands*.

Show commands are used to determine the current status of an interface or protocol, whereas debug commands are used to show the processes an interface or protocol executes in order to establish continuity or communication.

Basic troubleshooting involves ensuring that the hardware is functioning correctly, then checking to see that configurations are correct and communication processes are proceeding normally over the wire. You should start at the physical layer and work your way up the OSI model to determine where the problem(s) are in establishing the connection.

PPP and Cisco Access Servers

Below are some basic steps that you can use to troubleshoot remote connections to a Cisco access server.

1. Does the user's modem connect? If No, use these commands to determine the status of the modem: **show modem log**, **debug modem**.
2. Does the LCP negotiation succeed? If No, use these commands to determine the point of failure: **debug PPP negotiation**, **debug PPP error**.
3. Does the authentication succeed? If No, use this command to determine the cause of failure: **debug PPP authentication**.
4. Does the network layer succeed? If No, use this command to determine the point of failure: **debug PPP negotiation**.
5. If all of the above is successful, use this command to inspect the user's session: **show caller {line, user, ip, interface}**.

PPP and ISDN Connections between Cisco Routers

Following is a typical scenario to determine the problem(s) that occur when an BRI interface fails to establish a remote connection using PPP over an ISDN line:

First, we need to check the status of the physical layer:

```
Cisco command: show isdn stat  
The current ISDN Switchtype = basic-nil  
ISDN BRI0 interface  
Layer 1 Status:  
    DEACTIVATED  
Layer 2 Status:  
    Layer 2 NOT Activated  
Layer 3 Status:  
    No Active Layer 3 Call(s)  
Activated ds1 0 CCBs = 0  
Total Allocated ISDN CCBs = 0
```

The output above indicates that there is a problem with the physical layer. The layer 1 status being “DEACTIVATED” indicates this. This could be caused by a bad cable, a bad NT-1 device (or no power to an external NT-1 device), or a bad demarc.

In this instance, we had a bad cable between the NT-1 device and the BRI interface of the Cisco router. We replaced our cable and executed the command again:

```
The current ISDN Switchtype = basic-nil  
ISDN BRI0 interface  
Layer 1 Status:  
    ACTIVE  
Layer 2 Status:  
    Layer 2 NOT Activated  
Layer 3 Status:  
    No Active Layer 3 Call(s)  
Activated ds1 0 CCBs = 0  
Total Allocated ISDN CCBs = 0
```

The output above indicates that the physical layer is functioning properly as evidenced by the Layer 2 status being “ACTIVE.” Now we turn our attention to Layer 2 to determine where the problem is within that layer. If Layer 2 were functioning correctly, the router would receive TEIs (Terminal Endpoint Identifiers) from the ISDN switch.

To determine whether there are any Layer 2 problems, turn on terminal monitoring (`term mon`), execute the following command, and then PING the IP address of the BRI0 interface:

Cisco command: `debug isdn q921`

ISDN Q921 packets is on

(after ping):

Type escape sequence to abort.

```
Sending 5, 100 byte ICMP Echos to 10.1.20.2, timeout is 2 seconds:
12:20:01: TX -> IDREQ ri = 18543 ai = 127 dsl = 0
12:20:03: TX -> IDREQ ri = 1546 ai = 127 dsl = 0
12:20:05: TX -> IDREQ ri = 1834 ai = 127 dsl = 0
12:20:07: TX -> IDREQ ri = 17456 ai = 127 dsl = 0
.....
12:21:03: TX -> IDREQ ri = 1654 ai = 127 dsl = 0
```

The output above indicates a malfunctioning NT-1 device, an incorrectly provisioned circuit, or an incorrect ISDN switch type configured on the router. After speaking with the local exchange carrier (LEC), it was determined that the circuit was not correctly provisioned.

Here is what a good Layer 2 output looks like for this debug command:

Type escape sequence to abort

```
Sending 5, 1000 byte ICMP Echos to 10.1.20.2, timeout is 2 seconds:
12:45:17: BRI0: TX -> RRp sapi = 0 tei = 102 nr = 1
12:45:17: BRI0: RX <- RRF sapi = 0 tei = 102 nr = 1
12:45:19: BRI0: TX -> RRp sapi = 0 tei = 101 nr = 3
12:45:19: BRI0: TX <- RRF sapi = 0 tei = 101 nr = 3
12:45:19: BRI0: TX -> INFOc sapi = 0 tei = 101 ns = 1 nr = 2
I = 0x04E120406283703C14033348C4001233
12:45:21: BRI0: TX <- RRr sapi = 0 tei = 101 nr = 2
....
12:45:25: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0: B-
Channel 1, changed state to up. !!!
Success rate is 60 percent (3/5), round-trip min/avg/max = 100/110/120 ms
```

Please note the reception of TEIs from the ISDN switch. Each time you shut down the BRI0 interface and bring it back up, you should receive new TEIs from the ISDN switch.

Now, if you execute the **show isdn status** command, you will receive the following:

```
Cisco command: show isdn status
The current ISDN Switchtype = basic-nil
ISDN BRI0 interface
  Layer 1 Status:
    ACTIVE
  Layer 2 Status:
    TEI = 102, State = MULTIPLE_FRAME_ESTABLISHED
    TEI = 101, State = MULTIPLE_FRAME_ESTABLISHED
  Layer 3 Status:
    1 Active Layer 3 Call(s)
    Activated ds1 0 CCBs = 1
    CCB:called=800C, sapi=0, ces=1, B-chan=1
```

If Layer 3 does not activate, use the **debug isdn q931** command to troubleshoot the Layer 3 problems. Below is an example of output from a router whose Layer 3 is functioning properly (be sure to turn on terminal monitoring, execute the command, then ping the IP address of the router's BRI0 interface):

```
Cisco command: debug isdn q931
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.20.2, timeout is 2 seconds:
12:51:11: %SEC-6-IPACCESSLOGDP: list 100 permitted icmp 10.1.20.2 ->
10.1.20.2 (0/0), 1 packet
12:51:11: BRI0: TX -> SETUP pd = 8 callref =0x08
12:51:11: BRI0:     Bearer Capability I = 0x8890
12:51:11: BRI0:     Channel ID I = 0x62
12:51:13: BRI0:     Called Party Number I = 0x70, `4097004509'
12:51:13: BRI0: RX <- CALL_PROC pd = 8 callref = 0x82
12:51:13: BRI0:     Channel ID I = 0x89
12:51:15: BRI0: ISDN Event: incoming ces value = 1
.....
12:51:17: %LINK-3-UPDOWN: Interface BRI0: B-Channel 1, changed state to
up
12:51:17: BRI0: TX -> CONNECT_ACK pd = 8 callref = 0x08
```

```
12:51:17: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0: B-  
Channel 1, changed state to up!  
Success rate is 60 percent (3/5), round-trip min/avg/max = 110/130/150  
ms
```

(If the line in bold contains “HOST_TERM_REGISTER_NACK – invalid EID/SPID, or TEI not assigned Cause I = 0x8082 – No route to specified network,” check to see that your service profile identifiers (SPIDs) are valid and that your ISDN switch-type is correct.) The most common Layer 3 problems are incorrect IP addressing, incorrect SPIDs, or erroneous access lists assigned to the interface.

Many communication problems with remote access systems are due to an authentication failure.

Below is an example of debugging CHAP:

Cisco command: debug ppp chap (make sure your router is in terminal monitor mode and then ping the IP address of the BRI0 interface)

```
12:53:11: %LINK-3-UPDOWN: Interface BRI0: B-Channel 1, changed state to  
up  
12:53:11: PPP BRI0: B-Channel 1: CHAP challenge from ciscotr2  
12:53:11: PPP BRI0: B-Channel 1: CHAP response received from ciscotr2  
12:53:11: PPP BRI0: B-Channel 1: remote passed CHAP authentication.  
12:53:11: PPP BRI0: B-Channel 1: Passed CHAP authentication with remote
```

If the output from the command states, “PPP BRI0: B-Channel 1: failed CHAP authentication with remote,” please check your username and password for correctness—passwords and usernames are case sensitive.

Other useful Cisco debug commands:

```
debug ppp ?  
debug ppp chap  
debug ppp pap  
debug ppp multilink  
debug isdn events  
debug ppp negotiation  
debug dialer
```

To debug MSCB:

```
debug ppp cbcp
```

Providing Remote Access Services for Microsoft Windows Clients

Microsoft Windows clients using either the native DUN that comes with the Windows operating system, or a third-party dialing program provided by an ISP or corporate IT department, can access Remote Access Services (RAS).

There are two basic steps for configuring an RAS client on a Windows workstation:

1. Install a modem to be used for dial up (Microsoft Windows 9x and Windows 2000 should automatically recognize and configure most modems when booted for the first time after the device has been physically installed), and connect it to an operational communications line.
2. Configure the software to be used as the dial-up program. Configuration issues include the number to be dialed, the link-layer and network protocols to be used, the manner in which the network address is assigned, and so on.

The Microsoft DUN client supports TCP/IP, Internetwork Packet Exchange/Sequenced Packet Exchange (IPX/SPX), and NetBEUI by default, as well as support for multilink when two modems are installed within the same computer.

By default the “Log on to network” check box is selected under “Advanced options” of the “Server Types” tab of the “Properties” dialog box. This check box should be deselected when dialing into a Cisco access server. If this box is not deselected, the client will attempt to use your Windows user ID and password for logon, and you will be disconnected from the Cisco access server.

Microsoft Specific PPP Options

There are several PPP options that may be configured to provide remote access to Microsoft Windows clients using Microsoft’s proprietary protocols such as MS-CHAP and MSCB.

MSCB is enabled by default when PPP callback is configured on Cisco routers running IOS version 11.3(2)T or later.

MS-CHAP may be configured by using the keyword “ms-chap” on the PPP authentication command line under the interface configuration mode. For example:

```
username rudder password elephantwalk  
interface Dialer1
```

```
ip address 10.10.10.1 255.255.255.0
encapsulation ppp
dialer in-band
dialer group 1
ppp authentication ms-chap
```

Windows 95 Clients

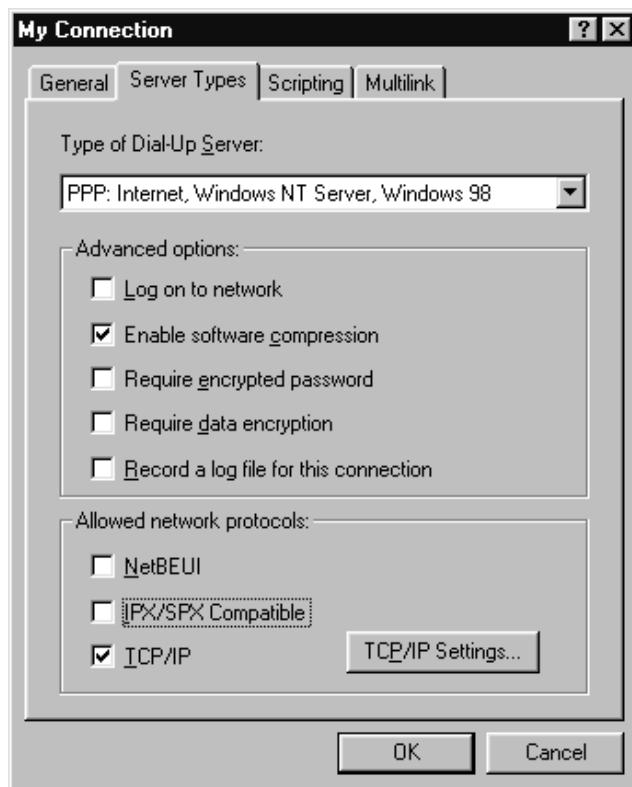
Windows 95 clients default to the PPP dial-up server when using Microsoft's DUN software. To confirm this setting, or to change a manually configured dial-up connection to PPP, do the following:

1. Double-click the "My Computer" icon on your desktop.
2. Double-click "Dial-up Networking."
3. Right-click the dial-up connection of interest and select "Properties."
4. Select the "Server Types" tab.
5. Under "Type of dial-up server," select "PPP: Windows 95, Windows NT 3.5, Internet."
6. Deselect the "Log on to network" radio button (unless dialing into a Windows server).
7. Select the check boxes of the network protocols you will be using.
8. If your IP address is to be dynamically assigned by your ISP or the corporate intranet, select "TCP/IP Settings."
9. Next, select the "Server assigned IP address" radio button; the "Server assigned name server addresses" should also be selected.
10. Leave all other defaults as they are.
11. Click "OK" to save your changes and return to the DUN window.

Windows 98 Clients

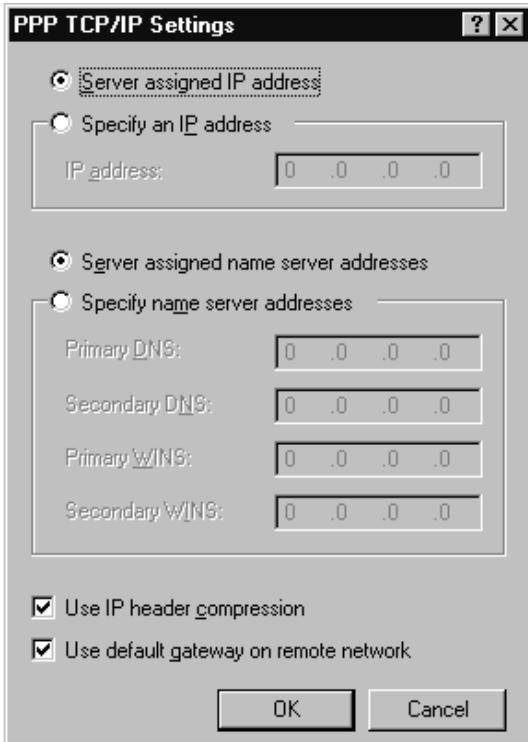
Windows 98 clients default to a PPP dial-up server when using Microsoft's DUN software. To confirm this setting, or to change a manually configured dial-up connection to PPP, do the following (Figures 3.13 and 3.14):

1. Double-click the "My Computer" icon on your desktop.
2. Double-click "Dial-up Networking."

Figure 3.13 Selecting PPP in MS dial-up networking.

3. Right-click the dial-up connection of interest and select “Properties.”
4. Select the “Server Types” tab.
5. Under “Type of Dial-Up Server,” select “PPP: Internet, Windows NT Server, Windows 98.”
6. Uncheck the “Log on to network” check box (unless dialing into a Windows server).
7. Select the check boxes of the network protocols you will be using.
8. If your IP address is to be dynamically assigned by your ISP or the corporate intranet, select the “TCP/IP Settings” radio button. Next, select the “Server assigned IP address” radio button. (“Server assigned name server addresses” should also be selected.)
9. Leave all other defaults as they are.
10. Click “OK” to save your changes and return to the DUN window.

Figure 3.14 Selecting DHCP IP address assignment on Windows 98.



Windows NT4 Clients

Windows 95 clients default to a PPP dial-up server when using Microsoft's DUN software. To confirm this setting, or to change a manually configured dial-up connection to PPP, do the following:

1. Double-click the "My Computer" icon on your desktop.
2. Double-click "Dial-up Networking."
3. Right-click the dial-up connection of interest and select "Properties."
4. Select the "Server Types" tab.
5. Under "Type of Dial-Up Server," select "PPP: Windows NT, Windows 95 Plus, Internet."
6. Uncheck the "Log on to network" check box (unless dialing into a Windows server).
7. Select the check boxes of the network protocols you will be using, such as "TCP/IP."

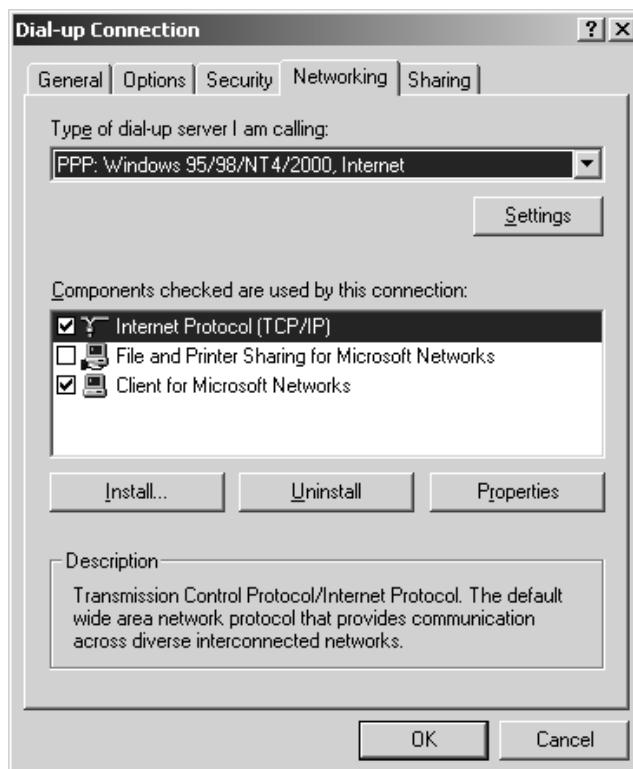
8. Select whether to have DHCP assign your IP address, or assign a static IP configuration (IP address, mask, default gateway, and so on).
9. If you need to configure MSCB in NT, select “User Preferences,” select the “Callback” tab, and select “Yes, call me back at the number(s) below” and enter your phone number.

Windows 2000 Clients

Windows 2000 clients also default to a PPP dial-up service when using Microsoft’s DUN software. To confirm this setting, or to change a manually configured dial-up connection to PPP, do the following (Figures 3.15, 3.16, and 3.17):

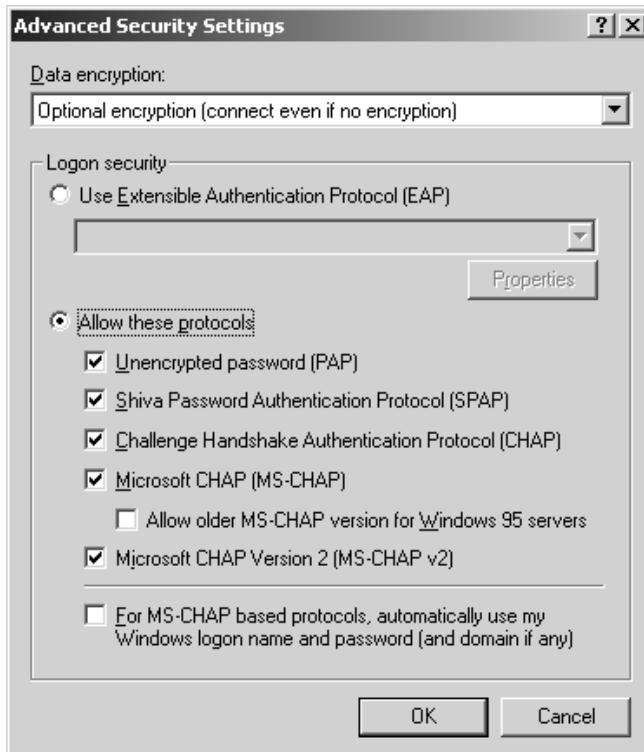
1. Double-click the “My Computer” icon on your Windows 2000 desktop.

Figure 3.15 Windows 2000 dial-up connection properties.



2. Double-click “Network and Dial-up Connections.”
3. Right-click the dial-up connection of interest and select “Properties.”
4. Select the “Networking” tab.
5. Under “Type of dial-up server I am calling,” select “PPP: Windows 95/98/NT 4/2000, Internet.”
6. To select whether to have DHCP assign your IP address, or to assign a static IP address, highlight “Internet Protocol (TCP/IP)” and select the “Properties” button. To use DHCP services, select the “Obtain an IP address automatically” radio button. To use a manually assigned IP address, select the “Use the following IP address” radio button and enter the IP address.
7. To select authentication protocol (such as PAP, CHAP, or MS-CHAP), select the “Security” tab, and then press the “Advanced Security Settings” button and check all applicable authentication protocols.

Figure 3.16 Windows 2000 advanced security settings dialog box.



Windows 2000 clients use an installation wizard to guide users through the installation of new dial-up connections. To install a new dial-up connection, do the following:

1. Double-click the “My Computer” icon.
2. Select “Network and Dial-up Connections.”
3. Select “Make New Connection.”
4. Follow the wizard prompts.

Figure 3.17 Windows 2000 dial-up configuration wizard.



Troubleshooting Microsoft Windows Connections

To troubleshoot MS Windows connections from the client end, do the following general steps:

1. Make sure that the dial-in line the modem is connected to has a dial tone.

2. Go to Windows' "Control Panel" (and/or "Device Manager" in the "System Panel" for Win95/98) and make sure your modem driver is installed, your modem is operational, and that it has no conflicts with other hardware.
3. Check in the "Network" panel and make sure that the proper network protocols are configured (such as TCP/IP) for the dial-up adapter, and that "Client for Microsoft Windows" or another client has been installed.

Summary

From our thorough examination of PPP, we can see the reason for its popularity as the de facto standard for remote access networks. It is a reliable, versatile, secure, and scalable protocol for connecting two point-to-point devices.

PPP's LCP and NCP sublayers handle the creation, configuration, and maintenance of the point-to-point connection. Through LCP frames, the status of the link is monitored and maintained.

Configuration and negotiation parameters support the use of multiple network protocols (such as TCP/IP, IPX, and AppleTalk) over the same communications link. Neither SLIP nor ARAP support more than one native network protocol.

Another very important part of PPP's popularity is the authentication of end-to-end peers using PAP, CHAP, and the technique of PPP Callback. These authentication methods enhance network security to help ease the concerns of network administrators and other IT professionals.

Through the use of MP, several communications lines can be bound together to form a single logical connection between two point-to-point peers that is transparent to the end user. By using MMP, such "bonds" can be distributed across several Cisco access servers to distribute dial-in usage and simplify user access by using only a single telephone number for all dial-in access. Such usage allows IT departments and ISPs to fully utilize their dial-in access servers while providing higher bandwidths to "power users" using current access technologies such as analog dial-in lines and ISDN services.

All of these benefits are achieved through a protocol that is simple for network engineers and end users alike to implement, maintain, and use.

FAQs

Q: Can PPP be used over an ISDN line?

A: Yes. PPP can be used over ISDN and most asynchronous and synchronous communications links.

Q: Does PPP support TCP/IP, IPX, NetBEUI, and AppleTalk?

A: Yes. SLIP supports only TCP/IP, and ARAP supports only AppleTalk.

Q: Can I use PPP over a Frame Relay network?

A: No. Frame Relay is the Layer 2 protocol used on Frame Relay networks.

Q: If I have 10 users dial into my Cisco access router, do they all appear as different networks for each connection?

A: Yes. PPP treats each connection as a different network, and an associated entry will be placed into the Cisco access router's routing table.

Q: Can multiple Cisco access servers be grouped together in a single rotary group so that all incoming calls go to a single dial-in number?

A: Yes, this grouping of servers is known as MMP. MMP is completely transparent to the end user.

Q: What version of the Cisco IOS must be used to support MMP?

A: The enterprise j-image of the Cisco IOS. See www.cisco.com/warp/public/131/6.html

Utilizing Virtual Private Network (VPN) Technology for Remote Access Connectivity

Solutions in this chapter:

- Site-to-site VPN technology
- Remote access VPN technology
- Advantages of VPN technology
- Disadvantages of VPN technology
- Security
- Cisco's VPN solutions

Introduction

The term VPN (virtual private network) is a hot term that often pops up when discussing today's networking infrastructure technologies. A VPN is another term for a secure, private network over a public infrastructure like the Internet. With many companies utilizing a shared office or being faced with providing network access to traveling users, it is becoming increasingly popular for corporations to provide a VPN solution. It's as easy as installing a secure client on employees' computers, providing them with public Internet access, and allowing them to dial in to the Internet and access the same private data that they would if they were locally connected to their company's local area network (LAN). There are many cost advantages that make it clear why VPNs are now being implemented over traditional infrastructures like Frame Relay or Integrated Services Digital Network (ISDN), but there are also some disadvantages that need to be reviewed. This chapter walks you through the different types of VPN solutions and describes the important factors to consider when determining whether a VPN solution is right for your environment.

VPN Technology

VPN technology allows private secure networking over public network infrastructures. This is done through technology that allows VPN devices to authenticate their identity, verify the integrity of the data being sent and received, and optionally, provide for confidentiality of data through encryption. Today's VPNs are based on the Internet Security Association and Key Management Protocol (ISAKMP) and Internet Protocol Security (IPSec) standards.

ISAKMP & IKE

ISAKMP is a framework for exchanging keys and establishing security associations. ISAKMP does not negotiate keys, but simply provides for rules to follow.

Internet Key Exchange (IKE) provides added features, flexibility, and ease of configuration for the IPSec standard. IKE uses part Skeme and part Oakley protocols, which follow the ISAKMP framework. IKE is used to authenticate peers, set up IPSec keys, and negotiate security associations. A security association is created when two VPN devices decide on what algorithms and keys to use for key exchange, authenticating, and encrypting data. Generally, when speaking about ISAKMP and IPSec together, there are two initial security associations that take place—the authentication of the devices and IPSec operations.

For IT Professionals

Skeme and Oakley Protocols

The Oakley protocol describes a series of key exchanges, called *modes*, and details the services provided by each (for example, perfect forward secrecy for keys, identity protection, and authentication). The Skeme protocol describes a versatile key exchange technique that provides anonymity, reputability, and quick key refreshment. Their relationship to ISAKMP is fairly straightforward: where Oakley defines *modes* of exchange, ISAKMP defines phases of when each is applied.

IPSec

IPSec is a set of protocols used at the network layer to secure data. IPSec consists of two protocols, Authentication Header (AH) and Encapsulating Security Payload (ESP).

AH provides protection by placing itself in the header data. The authentication header is used to validate the integrity of the packet, as well as to validate the origin of the packet. AH can also prevent replay attacks, where a captured session of data is replayed against a host service. The AH protocol uses a hash algorithm to provide this data integrity. Using AH, the receiving peer can be assured that the header information is valid and originated from the source without intervention. AH can be used alone to provide authenticated traffic or in combination with ESP to provide encrypted data.

ESP is the other protocol in the IPSec suite. ESP is used to encrypt the payload or data in an IP datagram to provide data confidentiality. It encapsulates the datagram, whereas AH embeds itself into the datagram. ESP is also used to validate authenticity of origination and integrity of the datagram. ESP provides for data confidentiality through the encryption of the packet payload; confidentiality can be used with or without the optional authenticity and integrity parameters. Confidentiality used without authenticating or validating integrity can allow for certain other forms of attack, so validation and integrity are recommended in using ESP or AH. ESP can also be used to prevent replay attacks and to thwart traffic flow analysis.

DES, Triple Pass DES & 3DES

The Data Encryption Standard (DES) is a very mature cryptographic system. The DES algorithm is a complex symmetric algorithm that specifies that data be encrypted in 64 bit blocks. A 64-bit block of clear text goes into the algorithm along with a 56-bit key; the result is a 64-bit block of cipher text. Since the key size is fixed at 56-bits, the number of keys available (the key space) is 256 (about 72,000,000,000,000,000 keys).

Triple pass DES is a cryptographic system that uses multiple passes of the DES algorithm to increase the effective key space available to the system. In triple pass DES, the clear text data is first encrypted with a 56-bit key. The resulting cipher text is then decrypted with a different key. Of course, decrypting cipher text with the wrong key will result in garbage. Finally, the garbage is encrypted again with the first key. This implementation of triple pass DES is known as EDE (for Encrypt, Decrypt, Encrypt), and the technique increases the effective key length from 56 bits to 112 bits. Ninety-bit keys should protect encrypted data for about 20 years.

3DES is a cryptographic system that uses multiple passes of the DES algorithm to increase the effective key space available to the system even further than triple pass DES. The same EDE technique employed in triple pass DES is used, except that three different keys are used. This increases the effective key length from 56 bits for simple DES to 168 bits for 3DES.

The benefit of using 3DES over DES is obvious. The very strong encryption and security of the key make it the best solution when the highest security is needed. The drawback to 3DES is its effect on processing. It takes a lot more processing power to compute such a complex algorithm; for this reason, vendors have begun selling add-on cards that separate crypto processing functions from the processor of the VPN device so the processor can do its normal functions and the add-on card takes the crypto load off the processor.

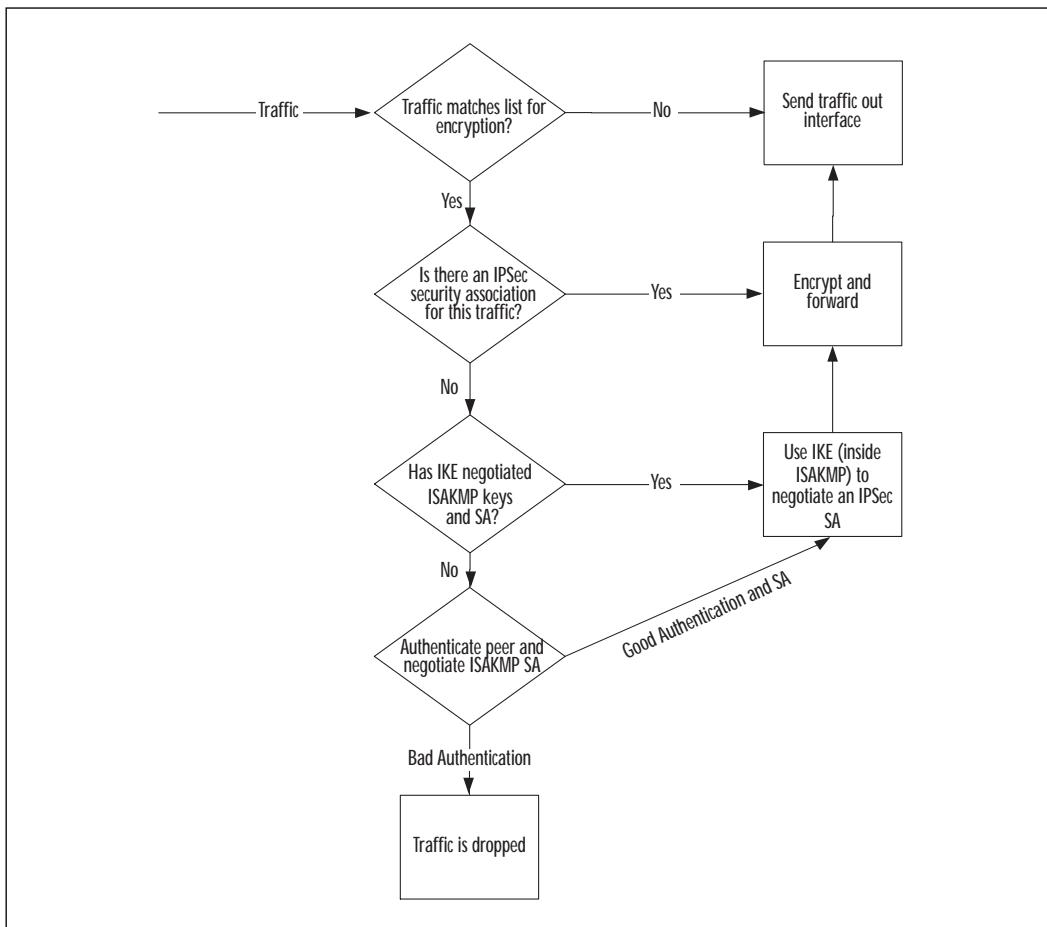
VPN Operation

There is often confusion over how IPSec, IKE, and ISAKMP work together to create a VPN. To sort this out, let's take a look at the flowchart in Figure 4.1 to see how they operate together to form a VPN tunnel.

As traffic enters the router to be forwarded, it is checked against an access list associated with the crypto map applied to that particular interface. If the traffic matches the list, the router checks to see if there is an IPSec security association (IPSec SA) with the peer for this traffic. If there is, the traffic is encrypted and sent out the interface. If there is no IPSec SA, the router will check to see if it has an ISAKMP security association (ISAKMP SA). If it does, then IKE will negotiate IPSec keys and SAs,

encrypt the traffic using IPSec and forward the traffic. If there is no ISAKMP SA, then IKE will attempt to authenticate the peer and create an ISAKMP SA; upon successful completion of an ISAKMP SA, IKE will negotiate an IPSec SA, encrypt the data, and forward the traffic. IKE uses the Skeme and Oakley protocols inside the ISAKMP framework, so that when we are using IKE to negotiate keys and security associations, it is operating within ISAKMP.

Figure 4.1 The interaction among IPSec, IKE, and ISAKMP.



Cisco VPN Terminology

Here are some of the terms used in the world of Cisco VPN technology. Make sure you know what they mean before reading on.

Peer The “other side,” or the other router that will be doing encryption. It takes at least two encryption devices to make a VPN, and each one is the peer of the other.

Transform-Set Used to define the IPSec protocols you want to use for authentication and/or encryption.

Crypto Map Used to tie together configurations such as the transform set, the peer, and the data to be encrypted.

Dynamic Crypto Map A crypto map before some of the information is provided by the remote peer.

ISAKMP (Internet Security Association and Key Management Protocol) Framework providing a means for policy negotiations and key management.

IKE (Internet Key Exchange) Uses parts of the ISAKMP framework to authenticate peers and negotiate IPSec keys and security associations.

ESP (Encapsulating Security Payload) Used as the method to encrypt the packet payload and/or authentication packets.

DES (Data Encryption Standard) Uses a 56-bit encrypting algorithm to encrypt data.

3DES (Triple Data Encryption Standard) Uses a 168-bit encrypting algorithm to encrypt data.

MD5 (Message Digest 5) A hash algorithm used to hash keys and pass the hash instead of passing the key or password.

SHA (Secure Hash Algorithm) Another hash algorithm used to hash keys and pass the hash instead of passing the key or password.

NOTE

Hashing is the process of running a password or shared key through an algorithm to come up with a string of numbers representing the key or password. This is then sent to the peer, as opposed to sending the key or password itself. The other side then de-hashes the key or password and checks it against its own database entry for the password or key. If the de-hashed string matches what the router has in its configuration, it is a good match. MD5 uses a 128-bit hash and SHA uses a 168-bit hash. Parallel processing on an MD5 hashed key is not possible.

VPNs can take different forms; a VPN can be created between two computers, a computer and a network, or a network and a network. VPNs

between a single computer and a network sometimes use client software installed on the machine to create a VPN tunnel between the computer and the device that connects to the network, such as a router—or in the case of an extranet, a firewall. In most enterprise scenarios the VPN tunnel is not actually created from the end computer to the remote end computer, but rather between two intermediary devices that sit between the computers or networks (such as routers, VPN concentrators, or firewalls). The IPSec standards have allowed various devices and software to interoperate when forming VPNs.

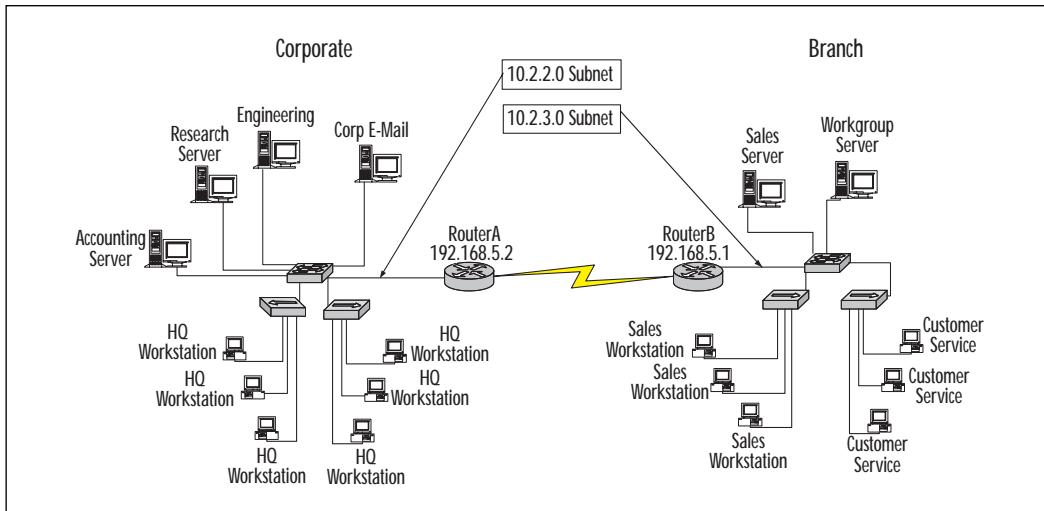
Site-to-Site VPN

Here we will begin exploring the various types of VPN scenarios. As stated earlier, a VPN in the enterprise is usually not created between two end host systems but rather the intermediary devices that connect the network. We will look at the various intermediary devices such as the Cisco router and the PIX Firewall, and how they are configured to form VPN tunnels. Later in the chapter we will also look at how to create VPN tunnels from client to intermediary device using software installed on the client system.

An Intranet Solution

In this section we will walk through several different scenarios in securing communication between a branch office and the corporate network. Let's begin by exploring the networks in Figure 4.2. First, look at the corporate network. On the corporate LAN are the accounting, research, engineering, and e-mail servers, which service both the corporate users and the branch office. The corporate network in this example is a 10.2.2.0 subnet, and is connected to the branch office through the 192.168.5.2 interface on the Central router. The branch office is subnet 10.2.3.0, which consists of a small sales force and customer services department, connected to Corporate through the Branch router on the 192.168.5.1 interface.

By utilizing VPN technology, we can secure communications between all of the corporate networks and all branch office networks, or a single host and the networks. In this scenario we will secure all communications between the networks by terminating VPN tunnels on the outside interfaces of both Branch and Corporate routers, and defining that all traffic between them gets encrypted. This is done in access lists based on source addresses, or networks and destination addresses, or networks. Let's begin by taking a look at how we configure ISAKMP and IKE to facilitate key management and exchange.

Figure 4.2 Corporate to branch office VPN.

Configuring ISAKMP/IKE

The first thing we will want to look at is how we configure ISAKMP policy to define security parameters to be used in Internet Key Exchange negotiation. It is possible to have several ISAKMP policies facilitate communications between peers requiring different encryption and hashing schemes; therefore, we assign a policy number to each of our ISAKMP policies. A peer must match one of the configured policies to begin negotiating the security association (SA). If there is no policy match, no SA is created and hence no VPN tunnel. Let's start by looking at the configuration of the Central router.

We need to define an ISAKMP policy. We use a policy number to assign commands specific to this configuration to an ISAKMP policy. If we had multiple peers and needed a different policy for each peer, we would simply add additional policies with different policy numbers. The lowest policy number takes precedence. For our config, we only need the single policy.

```
Central(config)# crypto isakmp policy 100
```

Next we need to decide what type of encryption we want to use for data confidentiality. We will use 56-bit data encryption standard (DES). Notice that the router prompt has changed. All configuration commands for ISAKMP from here on are part of policy 100.

```
Central(config-isakmp)# encryption des
```

Define which hash algorithm to use. This could be MD5 or SHA.

```
Central(config-isakmp)# hash md5
```

Now we define the method the two routers will use to authenticate each other. This can be done with pre-shared keys or using digital certificates. In our configuration we will use pre-shared keys.

```
Central(config-isakmp)# authentication pre-share
```

Specify the Diffie-Hellman 768-bit group identifier.

```
Central(config-isakmp)# group 1
```

When using pre-shared keys it is also necessary to define the identity of each peer. The identity can be the hostname or its IP address. The default is to use IP addresses for peer identity. We will specify that we want to use the ip address to identify our peer.

```
Central(config)# crypto isakmp identity address
```

Specify the pre-shared key and the identity (the IP address) of our encryption peer. The key will need to be the same on both ends.

```
Central(config-isakmp)# crypto isakmp key secretkey address 192.168.5.1
```

Verify the ISAKMP configuration.

```
Central router# show crypto isakmp policy
```

Issuing the **show crypto isakmp policy** command allows you to verify that the router is using the information that you entered for its configuration, and to quickly check the parameters of ISAKMP without having to read through the whole configuration of the device.

```
Protection suite of priority 100
  encryption algorithm: DES - Data Encryption Standard (56 bit keys).
  hash algorithm: Message Digest 5
  Authentication method: Pre-Shared Key
  Diffie-Hellman group: #1 (768 bit)
  Lifetime: 86400 seconds, no volume limit
Default protection suite
  encryption algorithm: DES - Data Encryption Standard (56 bit keys).
  hash algorithm: Secure Hash Standard
  authentication method: Rivest-Shamir-Adleman Signature
  Diffie-Hellman group: #1 (768 bit)
  lifetime: 86400 seconds, no volume limit
```

Now that we have configured the Central router on the corporate network with an Internet Key Exchange policy, let's configure the Branch router at the branch office. The ISAKMP policy config for the Branch router will be very similar to that of the Central router. After we finish the ISAKMP parameters on both routers, we will move on to configuring IPSec.

Define ISAKMP policy 100.

```
Branch(config)# crypto isakmp policy 100
```

Specify that DES will be used for encryption, as that is what we are using on the peer.

```
Branch(config-isakmp)# encryption des
```

Define which hash algorithm to use. We need to use MD5 because that is what we are using on the Central router.

```
Branch(config-isakmp)# hash md5
```

Specify the method of authentication. Again, we will use pre-share because that is what we are using on the Central router.

```
Branch(config-isakmp)# authentication pre-share
```

Specify the Diffie-Hellman 768-bit group identifier.

```
Branch(config-isakmp)# group 1
```

Specify that we will identify our peer by its IP address.

```
Central(config)# crypto isakmp identity address
```

Specify the pre-shared key and the identity (the IP address) of our encryption peer (Central router). The key will need to be the same on both ends.

```
Branch(config-isakmp)#crypto isakmp key secretkey address 192.168.5.2
```

Verify the ISAKMP configuration.

```
Branch router# show crypto isakmp policy
```

NOTE

You can use the same key for multiple peers—however, in the interest of security, it is advisable that you assign each peer a different key.

Again we issue the **show crypto isakmp policy** command to verify that the router has accepted all our commands and that the policy is accurate.

```
Protection suite of priority 100
  encryption algorithm: DES - Data Encryption Standard (56 bit keys).
  hash algorithm: Message Digest 5
  authentication method: Pre-Shared Key
  Diffie-Hellman group: #1 (768 bit)
  lifetime: 86400 seconds, no volume limit

Default protection suite
  encryption algorithm: DES - Data Encryption Standard (56 bit keys).
  hash algorithm: Secure Hash Standard
  authentication method: Rivest-Shamir-Adleman Signature
  Diffie-Hellman group: #1 (768 bit)
  lifetime: 86400 seconds, no volume limit
```

Configuring IPSec

We have defined items necessary for IKE operation, peer authentication, and methods for encrypting and hash. Now we can now move on to defining IPSec policy. Again we will start with the Central router. The first step in defining IPSec is to determine which IP traffic will or will not be protected by encryption. This is done through the use of access lists. These access lists are not like regular access lists, in that they are not used to define which traffic is blocked or permitted—these access lists are used to define what traffic is encrypted/decrypted and what traffic is not. The access list is not applied to an interface, nor is it specific to IPSec. Rather, it is the crypto map entry that ties the access list to IPSec, and the crypto map that is applied to the interface.

The first step in configuring IPSec will be to configure an access list defining the traffic that needs to be encrypted. You will configure a “mirror” access list on the remote peer:

```
Central(config)# access-list 120 permit ip 10.2.2.0 0.0.0.255 10.2.3.0
0.0.0.255
```

Now we must define a *transform set*. A transform set defines the type of authentication and encryption or data confidentiality you will use for IPSec. The first argument (esp-md5-hmac) defines the message hash for authentication; the second argument (esp-des) defines that the encryption will be 56-bit DES.

```
Central(config)# crypto ipsec transform-set MYSET esp-md5-hmac esp-des
```

Now that we have defined the transform set and the access list, defining what will be encrypted, we are ready to build the crypto map. For IPSec to successfully operate, the crypto map must contain compatible configurations between peers. Crypto map configurations are compatible if:

- Crypto map entries have “mirror” image access lists, or in the case of a dynamic crypto map, the local crypto must be permitted by the remote dynamic map.
- Crypto map entries properly identify the peer(s).
- Crypto map entries have at least one transform set in common between peers.

We will start by defining our crypto map name and the crypto map policy number, and by telling the router that the key negotiation and security association will be done using ISAKMP:

```
Central(config)# crypto map MYMAP 2 ipsec-isakmp
```

Next we need to tell the crypto map what gets encrypted (we actually defined this in the access list previously). We are now going to associate the access list with the crypto map:

```
Central(config-crypto-map)# match address 120
```

We need to define the peer that we will be doing IPSec with:

```
Central(config-crypto-map)# set peer 192.168.5.1
```

And finally, we associate the transform set we want to use with the crypto map:

```
Central(config-crypto-map)# set transform-set MYSET
```

Now all we need to do is to apply the crypto map to the appropriate interface on the router.

```
Central(config)# interface serial0/1
```

```
Central(config-if)#crypto map MYMAP
```

```
Central(config-if)#exit
```

Now we can move on to configuring the Branch office router. The Branch router configuration will be very similar to the Central router, because the crypto maps must be compatible, and we will use a mirror image access list on the Branch router. The list and peer will really be the only difference between the two configurations.

Again, we start by defining what should be encrypted. This should be a mirror image of the access list created on the Central router.

```
Branch(config)# access-list 120 permit ip 10.2.3.0 0.0.0.255 10.2.2.0  
0.0.0.255
```

Define the transform set.

```
Branch(config)# crypto ipsec transform-set MYSET esp-md5-hmac esp-des
```

Define the crypto map policy number and configure the router to use ISAKMP to exchange key information and create the security associations.

```
Branch(config)# crypto map MYMAP 2 ipsec-isakmp
```

Associate the mirror image access list with the crypto map.

```
Branch(config-crypto-map)# match address 120
```

Define the peer.

```
Branch(config-crypto-map)# set peer 192.168.5.2
```

Associate the transform set with the crypto map.

```
Branch(config-crypto-map)# set transform-set MYSET
```

And finally, apply the crypto map to the interface.

```
Branch(config)# interface serial0/1
```

```
Branch(config-if)#crypto map MYMAP
```

```
Branch(config-if)#exit
```

To see your crypto map configuration on the Central router, issue the **show crypto map** command.

```
Central#sh crypto map  
Crypto Map "MYMAP" 2 ipsec-isakmp  
Peer = 192.168.5.1  
Extended IP access list 120  
access-list 120 permit ip 10.2.2.0 0.0.0.255 10.2.3.0 0.0.0.255  
Current peer: 192.168.5.1  
Security association lifetime: 4608000 kilobytes/3600 seconds  
PFS (Y/N): N  
Transform sets={ MYSET, }
```

Now look at the Branch router crypto map.

```
Central#sh crypto map
Crypto Map "MYMAP" 2 ipsec-isakmp
  Peer = 192.168.5.2
  Extended IP access list 120
    access-list 120 permit ip 10.2.3.0 0.0.0.255 10.2.2.0 0.0.0.255
  Current peer: 192.168.5.2
  Security association lifetime: 4608000 kilobytes/3600 seconds
  PFS (Y/N): N
  Transform sets={ MYSET, }
```

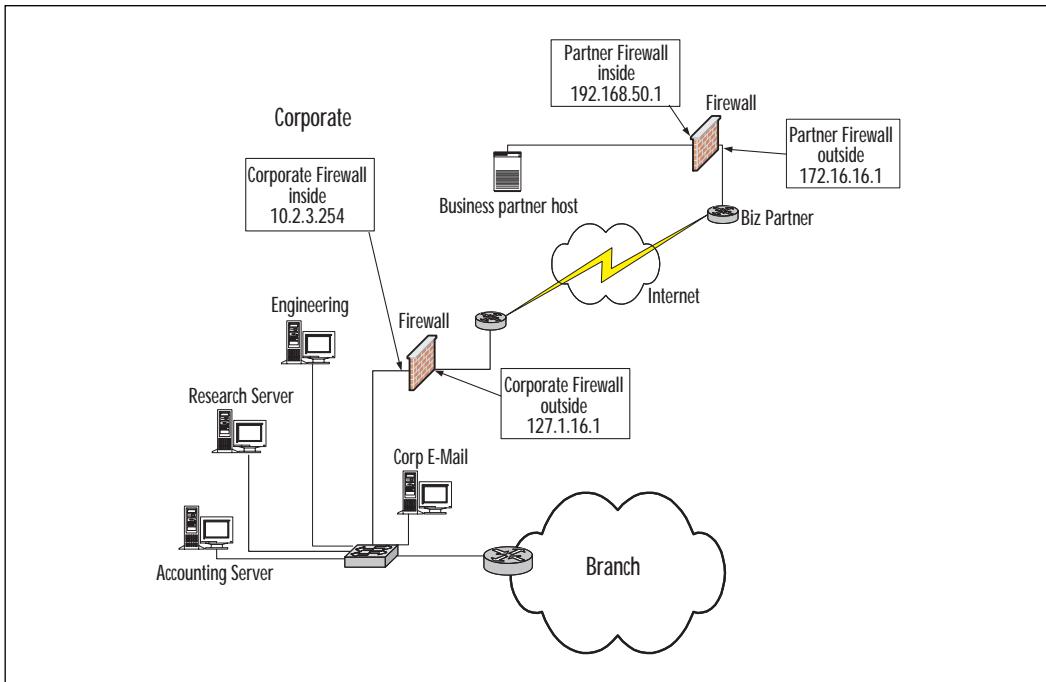
If you make changes to a crypto map, transform set, or any other item relating to your VPN, it may be necessary to issue the **clear crypto sa** command. This will clear the existing IPSec SAs so that renegotiation takes place and the changes are implemented immediately.

An Extranet Solution

We have taken care of our remote office, so let's take a look at adding a business partner communicating through the Internet. This will be very similar to the previous scenario. Most companies would do this on the firewall or a special VPN concentrator (we will discuss this later) for security reasons—that being the case, in this scenario we will look at configuring PIX to PIX Firewall VPN (see Figure 4.3). You can do this on the router and would follow the same principles as in the previous scenario. You could use the same pre-shared key with different ISAKMP and IPSec policies if you wished; however, it is advisable *not* to use the same key for different peers for security reasons.

Configuring the PIX Firewall for VPN can be done in many different ways. You can configure a VPN to use the Network Address Translation (NAT) address of the inside or “demilitarized zone” (DMZ) hosts, or you can configure the PIX to allow your peer to use the actual IP of the inside or DMZ hosts. The latter is the simpler of the two and is what we will be configuring here. Just keep in mind that you can use NAT when configuring a firewall VPN if needed. Let's start with the corporate firewall.

Figure 4.3 PIX to PIX VPN.



NOTE

If you are explicitly blocking traffic on your perimeter router, it may be necessary to build an access list allowing IPSec protocols through to the firewall. This can be done by permitting the ahp and esp protocol types and udp isakmp port. For example:

```
access-list 100 permit ahp host 172.1.16.1 host 192.168.52.1  
access-list 100 permit esp host 172.1.16.1 host 192.168.52.1  
access-list 100 permit udp host 172.1.16.1 host 192.168.52.1 eq  
isakmp
```

First, you need to configure the firewall to allow IPSec connections. If you don't explicitly allow IPSec connections, then you must use the conduit command to allow IPSec traffic to flow to the destination. For this configuration you can implicitly allow IPSec connections with the following command:

```
Sysopt connection permit-ipsec
```

Define a list specifying what needs to be encrypted. In this case you will encrypt all communications between networks. If you wanted to only allow and encrypt data between a single host on Corporate and a single host on the Business partner network, you would define that here in this access list.

```
Access-list 100 permit ip 10.2.3.0 0.0.0.255 192.168.50.0 0.0.0.255
```

This states that anything passing the list should not have to use NAT. This command does not get applied to any interface, but is associated with the crypto map so that only traffic that is already encrypted uses this feature.

```
Nat (inside) 0 access-list 100
```

Like the router-based VPN, you must define a transform set to tell the firewall what type of algorithm to use for encryption and authentication.

```
Crypto ipsec transform-set myset esp-des esp-md5-hmac
```

Now, define your crypto map to allow IPSec keys and security association negotiation to be done using ISAKMP.

```
Crypto map mymap 5 ipsec-isakmp
```

The following tells the firewall that traffic matching access list 100 should use this crypto map:

```
Crypto map mymap 5 match address 100
```

Set the address of your peer encrypting device.

```
Crypto map mymap 5 set peer 172.16.16.1
```

Configure the crypto map to use the transform set you created earlier.

```
Crypto map mymap 5 set transform-set myset
```

Configure the firewall to use the crypto map on traffic passing the outside interface.

```
Crypto map mymap interface outside
```

To use ISAKMP for SA negotiation, you must enable ISAKMP on the particular interface where it will be used:

```
Isakmp enable outside
```

Define the pre-shared key to be used and the peer that you will be negotiating with. The peer or your firewall must have a compatible policy.

```
Isakmp key partnetsecret address 172.16.16.1 netmask 255.255.255.255
```

Configure the firewall to use the IP address to identify its peer or peers.

```
Isakmp identity address
```

Configure the ISAKMP policy to use the pre-shared key for authentication.

```
Isakmp policy 10 authentication pre-share
```

Configure your ISAKMP policy to use 56-bit des for encryption.

```
Isakmp policy 10 encryption des
```

Configure ISAKMP to use MD5 as the hash algorithm for passing the key and SA info.

```
Isakmp policy 10 hash md5
```

Configure ISAKMP to use Diffie-Hellman 1.

```
Isakmp policy 10 group 1
```

The next configuration command tells the firewall the lifetime of the SA. When this expires, the firewall will renegotiate the SA.

```
Isakmp policy 10 lifetime 86400
```

The business partner must have a similar configuration on its firewall. Configure the list defining what traffic will get encrypted.

```
Access-list 100 permit ip 192.168.50.0 0.0.0.255 10.2.3.0 0.0.0.255
```

Use the **nat 0** command so that traffic passing the list can use the real IP address of the destination, as opposed to a NAT or static address.

```
Nat (inside) 0 access-list 110
```

Define the algorithms you will be using in your transform set.

```
Crypto ipsec transform-set myset esp-des esp-md5-hmac
```

Begin defining your crypto map to tell the router that you will want to use ISAKMP to negotiate SAs.

```
Crypto map mymap 5 ipsec-isakmp
```

Associate the list you created earlier with your crypto map.

```
Crypto map mymap 5 match address 110
```

Define your peer encrypting devices address.

```
Crypto map mymap 5 set peer
```

Associate the transform set to the crypto map.

```
Crypto map mymap 5 set transform-set myset
```

Configure the crypto map to the outside interface.

```
Crypto map mymap interface outside
```

Enable ISAKMP on the outside interface.

```
Isakmp enable outside
```

Configure the pre-shared key and the peer with whom you will be authenticating.

```
Isakmp key partnetsecret address 10.0.0.0 netmask 255.255.255.255
```

Configure the device so that ISAKMP identities use IP addresses.

```
Isakmp identity address
```

Configure ISAKMP to use the pre-shared key.

```
Isakmp policy 10 authentication pre-share
```

Configure ISAKMP to use 56-bit DES encryption for key exchange and SAs.

```
Isakmp policy 10 encryption des
```

Configure ISAKMP to use the MD5 hash.

```
Isakmp policy 10 hash md5
```

Use Diffie-Hellman 1.

```
Isakmp policy 10 group 1
```

Configure the security association lifetime for 86400 seconds.

```
Isakmp policy 10 lifetime 86400
```

Remote Access VPN

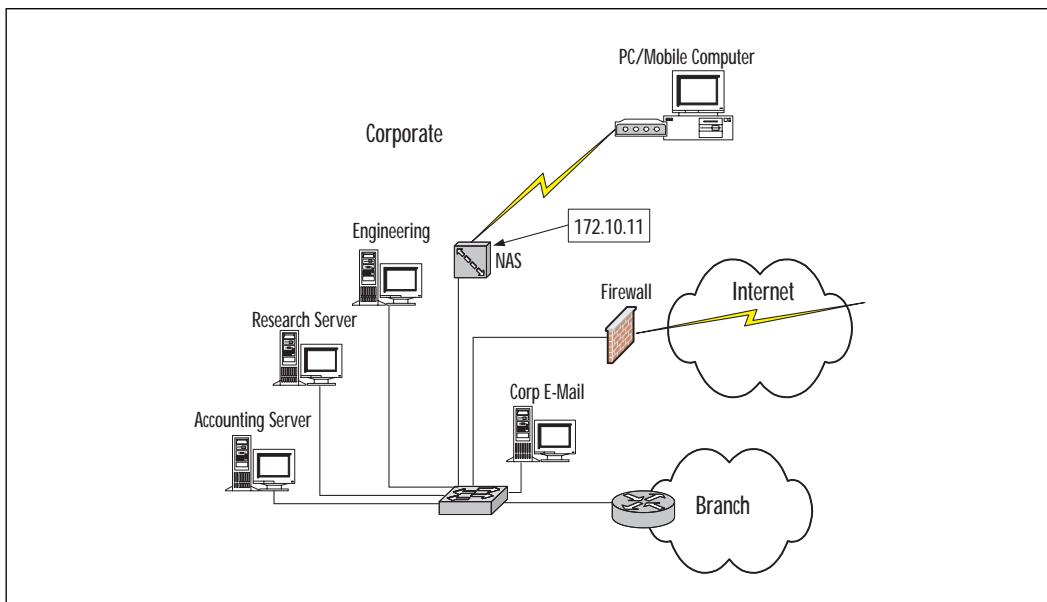
If you look at Figure 4.4, you can see we have added a network access server (NAS) to our corporate network. This is used to allow the employees, and possibly business partners, to connect to the internal network using a dial-in connection. It is depicted here as a generic symbol, but in the real world could be an AS5300 or a 3600 or even 2600 series Cisco router with modems and/or ISDN. We do not want to pass information through the

Public Switched Telephone Network (PSTN) unencrypted. To secure our traffic we will be using the CiscoSecure VPN client, v. 1.1. The CiscoSecure VPN client is a software program that is loaded on any hosts needing access to corporate through a VPN tunnel using the client. It will be used to create a tunnel between the host dialing in and the NAS. The VPN tunnel will terminate on the asynchronous interface we use to dial in on. The VPN client's use is not limited to dial-up. It can be used across any type of network interface running TCP/IP.

After the Cisco VPN client is installed, it will run automatically whenever you start your computer. If you look in the right-hand corner of your system tray, you will see its icon. You can double-click this icon, or right-click, and choose Policy Editor to add, change, or delete policy configurations.

Let's begin our configuration on the NAS router.

Figure 4.4 Enterprise dial-up VPN.



Configuring IPSec on the Network Access Server

Create the IPSec transform set.

```
RouterNAS(config)# crypto ipsec transform-set vpnclient esp-des esp-sha-hmac
```

Create the ISAKMP policy.

```
RouterNAS(config)# crypto isakmp policy 100
RouterNAS(config-isakmp)#hash md5
RouterNAS(config-isakmp)#authentication pre-share
```

Configure a shared key and identify the peer.

```
RouterNAS(config)# Crypto isakmp key dialclient address 10.1.1.1
```

Configure an access list defining the traffic to be encrypted. This list will specify that any inside host with a destination of the VPN client (10.1.1.1) will get encrypted.

```
RouterNAS(config)# Access-list 130 permit ip any host 10.1.1.1
```

Create a crypto map and associate the previous configurations.

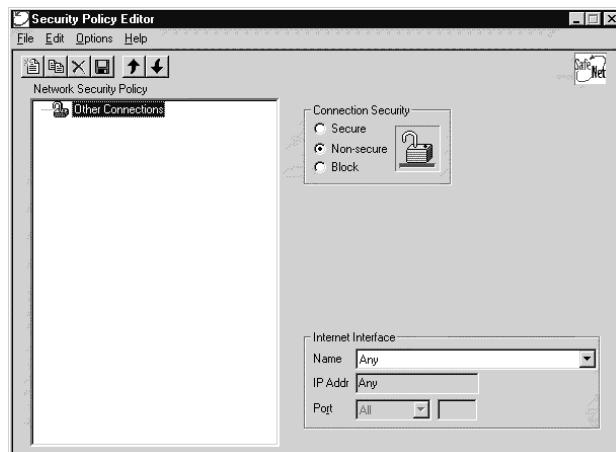
```
RouterNAS(config)#crypto map dialclient 10 ipsec-isakmp
RouterNAS(config-crypto-map)# set peer 10.1.1.1
RouterNAS(config-crypto-map)#set transform-set vpnclient
RouterNAS(config-crypto-map)#match address 130
```

Apply the crypto map to the interface.

```
RouterNAS(config-if)# Crypto map dialclient
```

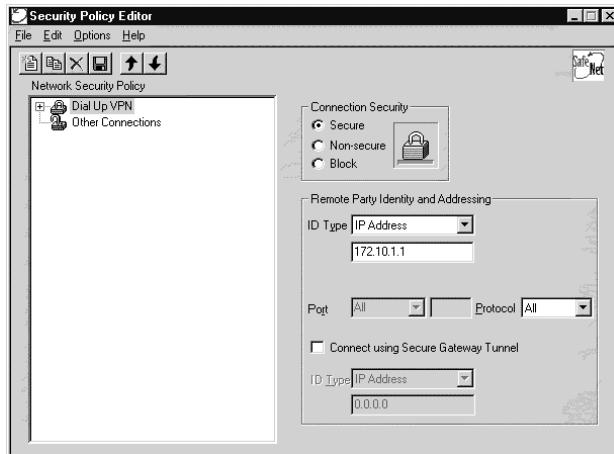
Now that you have configured the NAS router, you should configure the VPN client. Open the VPN client by double-clicking its icon in the lower right-hand corner of the system tray. You will see a screen like the one in Figure 4.5.

Figure 4.5 Creating a new connection.



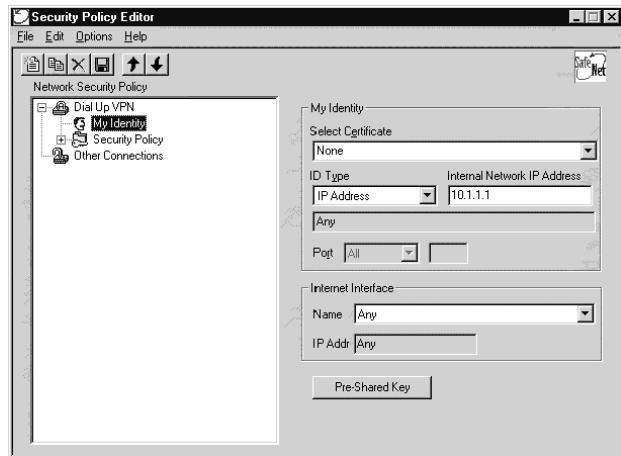
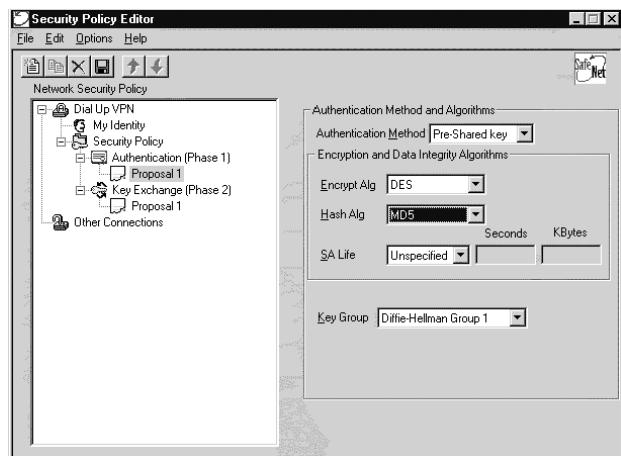
In this window you can specify which interface the VPN client will operate on, as well as the type of connection security. Normally you will leave these at the default values. You can start configuring a new IPSec policy by choosing New Connection from the File menu. After choosing New Connection, you will see a screen like the one in Figure 4.6.

Figure 4.6 Naming the connection and identifying peer.



Name your connection, as shown in Figure 4.6. Then you must identify your peer in the Remote Party Identity and Addressing text box. Use the address of the asynchronous interface you dial into on the NAS (172.10.1.1). That is all the configuration that is necessary on this page. Expand the connection by clicking the plus symbol next to your connection. After the connection is expanded, you will see the My Identity caption. Click My Identity to get the My Identity page, shown in Figure 4.7. Notice we have entered an Internal Network IP Address of 10.1.1.1, which matches the access list we created on the NAS. This is the identity of your VPN client.

Click the Pre-Shared Key button and enter the key (dialclient) configured on the NAS. Now that you have identified the client and set the pre-shared key, you can configure your security policies for authentication and encryption. Click the Security Policy caption. Click the Enable Replay Detection check box for enhanced security. There is nothing else to be configured on this page, so go ahead and expand the Authentication caption. This is where you set the ISAKMP policy for authentication. These must match the configuration policies set on the NAS. When finished, the VPN client authentication should look like the window in Figure 4.8.

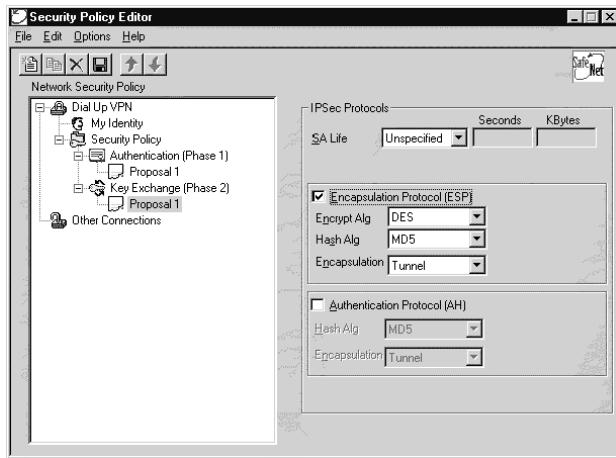
Figure 4.7 Pre-Shared Key.**Figure 4.8** Phase I proposal.

Now that you have a matching ISAKMP policy for your client, you need to create an IPSec policy to match the IPSec policy configured on the NAS. When you finish, the client will look the window in Figure 4.9.

Since you are using ESP, and not Authentication Header, leave the AH check box unchecked.

You are now ready to dial the NAS. To aid in troubleshooting the VPN client, you can right-click the VPN Client icon in the system tray and choose Log Viewer from the menu. This is similar to the debug function on the router. It will show you, step by step, as the client negotiates ISAKMP and IPSec with the NAS peer.

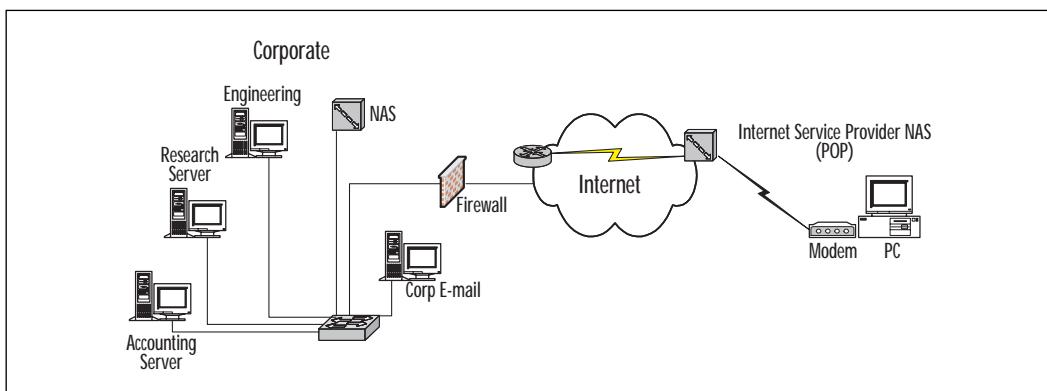
Figure 4.9 Phase 2 proposal.



Service Provider Solution

In this next scenario you will terminate a tunnel on the PIX. Look at Figure 4.10 to get an idea of the network. This scenario allows a user needing access to the corporate network to dial any Internet service provider (ISP) and create a tunnel over the Internet with the outside interface of your PIX firewall. This has the advantage of allowing an authorized person to connect to the inside of the corporate office from anywhere in the world, as long as Internet access is available. This works regardless of the choice of ISP, so long as there is a valid route to the outside interface of the PIX. This is great, because there are thousands of Internet dial-up points of presence all over the world, allowing an authorized person access to the corporate office from almost anywhere in the field.

Figure 4.10 Securing Internet dial-up.



This scenario will introduce you to *IKE mode config*. IKE mode config allows the PIX to assign the CiscoSecure VPN client an address from a pool of addresses defined by you, the administrator. You will also configure a “wildcard” pre-shared key. The wildcard pre-shared key allows any VPN client with the right ISAKMP policy and pre-shared key to connect and negotiate an SA, and to have its address assigned to it. This allows great flexibility in managing the CiscoSecure client. Using this configuration will allow you to automatically assign addresses, as opposed to manually tracking IP addresses manually assigned to VPN clients. This scenario will not configure an access list to define what is encrypted. Instead, it specifies that any connections using IPSec and the ISAKMP policy wildcard key get encrypted. You will then define an access list stating that the defined source and destination can communicate without the use of NAT. At first it may seem that this is a big security risk, considering you don’t explicitly define what is encrypted, or your peer. What you have actually done, however, is configure your connection so that everything between the PIX and the VPN client is encrypted, and a client cannot connect without the proper pre-shared key and authentication and encryption policies. Let’s look at how to configure the PIX for IKE mode configuration and then move on to configuring the VPN client.

Configuring ISAKMP

The first thing you want to do is allow IPSec connections to the PIX.

```
Pixfirewall(config)#sysopt connection permit-ipsec
```

Next, let’s define your wildcard pre-shared key. This basically says that any client can attempt to create an SA with the PIX. The SA will not work if the client is not using the correct authentication policies and the correct pre-shared key.

```
Pixfirewall(config)#isakmp key secretkey address 0.0.0.0 netmask 0.0.0.0
```

Configure ISAKMP identities to use the IP address.

```
pixfirewall(config)#isakmp identity address
```

Now define a pool of addresses to be assigned to the clients. Since your network is not large, assign a block of 100 IP addresses to be used.

```
Pixfirewall(config)#ip local pool test 192.168.56.0-192.168.56.100
```

You need to configure the PIX to allow the inside network to communicate with the addresses assigned to the VPN clients. You do this with an access list.

```
Pixfirewall(config)#access-list 110 permit ip 10.2.3.0 255.255.255.0  
192.168.56.0 255.255.255.0
```

You also want to tell the PIX that communication between the VPN clients and the inside network can be done without using address translations. This is done with the **nat 0** command.

```
Pixfirewall(config)#nat (inside) 0 list 110
```

You must configure ISAKMP to get IP addresses from the pool configured previously. This command tells the PIX to get the addresses from the local address pool called test.

```
Pixfirewall(config)#isakmp client configuration address-pool local test  
outside
```

Configure the ISAKMP policy to use the pre-shared key when authenticating its peer.

```
Pixfirewall(config)#isakmp policy 10 authentication pre-share
```

Configure the ISAKMP policy to use 56-bit encryption when swapping info.

```
Pixfirewall(config)#isakmp policy 10 encryption des
```

ISAKMP should use the MD5 hash.

```
Pixfirewall(config)#isakmp policy 10 hash md5
```

Use Diffie-Hellman 1.

```
Pixfirewall(config)#isakmp policy 10 group 1
```

ISAKMP SAs will expire and be renegotiated after 86400 seconds.

```
Pixfirewall(config)#isakmp policy 10 lifetime 86400
```

Configure ISAKMP to be enabled on the outside interface.

```
Pixfirewall(config)# isakmp enable outside
```

Configuring IPSec

Now that you have properly configured ISAKMP, you can move on to creating your crypto maps and configuring IPSec.

Configure a transform set to be used by your crypto map.

```
Pixfirewall(config)#crypto ipsec transform-set myset esp-des esp-md5-hmac
```

Define a dynamic crypto map to use the transform set. Remember that dynamic crypto maps are used to apply standard settings to a range of peers.

```
Pixfirewall(config)#crypto dynamic-map dynmap 15 set transform-set myset
```

Now we configure our crypto map to use ISAKMP for IPSec key exchange and SAs using the information contained in your dynamic map.

```
Pixfirewall(config)#crypto map mymap 15 ipsec-isakmp dynamic dynmap
```

You need to tell the PIX that it can initiate the giving of a dynamic address assignment and/or respond to the request for an address. To do this, use the following two commands, before you apply the crypto map to the interface.

```
Pixfirewall(config)#crypto map mymap client configuration address initiate  
Pixfirewall(config)#crypto map mymap client configuration address respond
```

Now apply the crypto map to the outside interface.

```
Pixfirewall(config)#crypto map mymap interface outside
```

Configuring the VPN Client

Choose New Connection from the File menu of the VPN client. In Figure 4.11, you can see I have named the connection Internet VPN. Notice that the Remote Party Identity and Addressing boxes are a little different from when we configured the client for NAS operation. I have also checked the Connect using Secure Gateway Tunnel check box and filled in an IP address. This is the address of the outside interface you will connect to on the PIX. The Remote Party Identity and Addressing reflects the inside network of the PIX and is the subnet identified in the PIX configuration. It is the source address list in access list 110 on the PIX.

You now configure authentication and encryption policies that match those defined on the PIX. You also add the pre-shared key we defined when configuring the PIX. When finished, the VPN client will look like Figures 4.12 and 4.13.

Now save your policies. Right-click the VPN client icon in the system tray and make sure the third item on the menu says “Deactivate Security Policy.” If you see this, it means the policies are active (not that they are in use, but that they are ready and “turned on”). To deactivate the policy, click the Deactivate Security Policy option, and the caption will change to Activate Security Policy. Make sure your security policies are active, open the log viewer and dial your Internet connection. Once you have been

assigned an IP by your Internet provider, you should see your client begin negotiation with the PIX in the log viewer. You can watch as the PIX assigns your client an IP address and SA negotiation is completed. Once negotiation is complete, you should have access to the inside network. You can test this by pinging inside addresses.

Figure 4.11 Configuring the VPN client for connection to PIX.

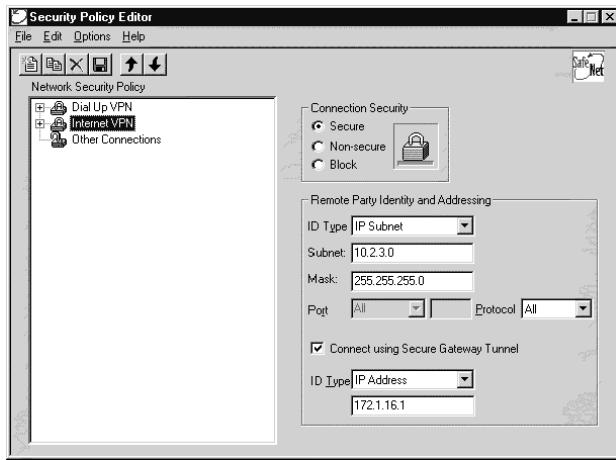


Figure 4.12 Configuring the authentication policy.

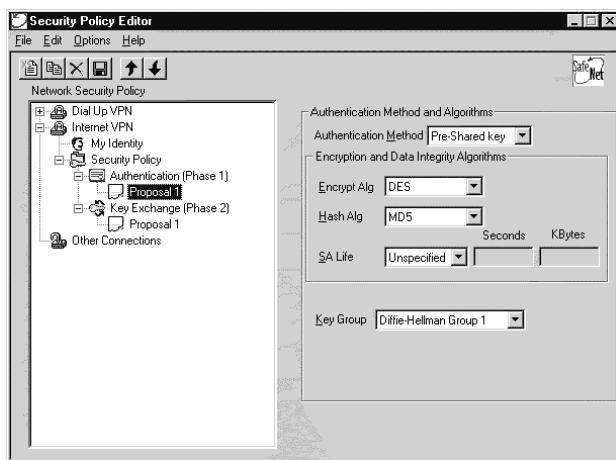
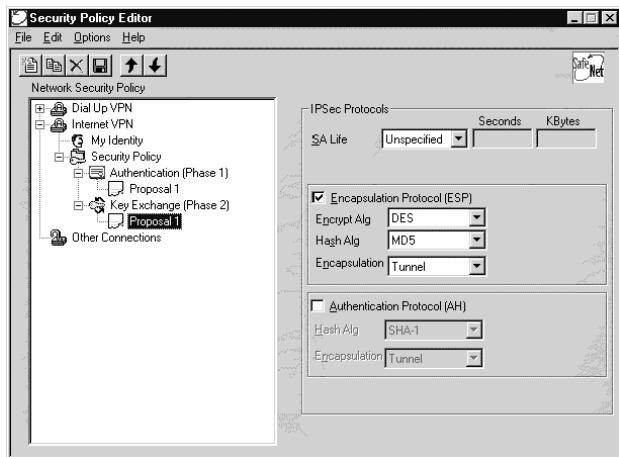


Figure 4.13 Configuring the key exchange.

Verifying and Debugging VPN Operation

You can verify and debug VPN operations using a combination of debug and show commands. Explore these on your device so that you become familiar with all the information available to you for troubleshooting and operation verification. We have already gone over a couple show commands in previous sections of the chapter. Here I want to cover the show commands that will help you verify that the VPN is operating. The show and debug output shown below is not from the configurations we have performed previously.

The **show crypto ipsec sa** command shows the security associations created for IPSec operation. It can be used to verify that the IPSec SA exists and that encryption is taking place.

```
show crypto ipsec sa
interface: Ethernet0
Crypto map tag: test1, local addr. 192.168.0.2
local ident (addr/mask/prot/port): (192.168.0.2/255.255.255.255/0/0)
remote ident (addr/mask/prot/port): (192.168.0.20/255.255.255.255/0/0)
```

You can see here that we have a peer and identify who the peer is.

```
current_peer: 192.168.0.20
PERMIT, flags={origin_is_acl,transport_parent,}
```

The following output shows us that we are encapsulating and encrypting outbound packets, as well as decapsulating and decrypting inbound packets. This verifies encryption operations and indicates that IPSec is operating between peers. This would be enough verification that a successful tunnel had been created, but let's go ahead and look at the rest of the show.

```
#pkts encaps: 77, #pkts encrypt: 76, #pkts digest 76
#pkts decaps: 88, #pkts decrypt: 88, #pkts verify 88
#send errors 0, #recv errors 0
```

This shows us where your VPN tunnel is terminating locally, as well as the peer terminating point. You can also see the transform set in use and can tell that replay detection is on.

```
local crypto endpt.: 192.168.0.2, remote crypto endpt.: 192.168.0.20
path mtu 1500, media mtu 1500
current outbound spi: 1694080F
inbound esp sas:
spi: 0xF3F17E1(255793121)
transform: esp-des esp-sha-hmac ,
in use settings ={Transport, }
slot: 0, conn id: 2, crypto map: test1
sa timing: remaining key lifetime (k/sec): (4607998/57)
IV size: 8 bytes
replay detection support: Y
spi: 0x8CC2053(147595347)

[ further output omitted... ]
```

Another good indicator of successful VPN operations is the **show crypto engine connections** command. The following example shows both the command and the output it produces.

```
show crypto engine connections active
ID Interface IP-Address State Algorithm Encrypt Decrypt
46 Ethernet0 172.21.230.67 set HMAC_MD5+DES_56_CB 0 4
47 Ethernet0 172.21.230.67 set HMAC_MD5+DES_56_CB 4 0
```

In this example you can see that Ethernet0 has an active crypto connection. It has encrypted and sent four packets and has decrypted four packets that it has received. In a single peer-to-peer VPN relationship, this would indicate that a good VPN operation is taking place.

Let's look at some sample debug outputs. Let's start by looking at an ISAKMP debug. Here we can watch as ISAKMP negotiates first its own security association, then looks for and negotiates a matching IPSec transform set and does the IPSec security association.

```
debug crypto isakmp
20:26:58: ISAKMP (8): beginning Main Mode exchange
20:26:58: ISAKMP (8): processing SA payload. message ID = 0
```

ISAKMP starts trying to match ISAKMP policy. Once a policy match is made, the peers will begin the authentication phase, where they authenticate each other.

```
20:26:58: ISAKMP (8): Checking ISAKMP transform 1 against priority 10
policy
20:26:58: ISAKMP:      encryption DES-CBC
20:26:58: ISAKMP:      hash SHA
20:26:58: ISAKMP:      default group 1
20:26:58: ISAKMP:      auth pre-share
20:26:58: ISAKMP (8): atts are acceptable. Next payload is 0
```

IKE has found a compatible policy in the output above and will begin authenticating the peer in the output below.

```
20:26:58: ISAKMP (8): SA is doing pre-shared key authentication
20:26:59: ISAKMP (8): processing KE payload. message ID = 0
20:26:59: ISAKMP (8): processing NONCE payload. message ID = 0
20:26:59: ISAKMP (8): SKEYID state generated
20:26:59: ISAKMP (8): processing ID payload. message ID = 0
20:26:59: ISAKMP (8): processing HASH payload. message ID = 0
20:26:59: ISAKMP (8): SA has been authenticated
```

Now that the ISAKMP security association has been established, ISAKMP will begin negotiating IPSec transform sets and key exchange.

```
20:26:59: ISAKMP (8): beginning Quick Mode exchange, M-ID of 767162845
20:26:59: ISAKMP (8): processing SA payload. message ID = 767162845
20:26:59: ISAKMP (8): Checking IPSec proposal 1
```

```
20:26:59: ISAKMP:      transform 1, ESP_DES
20:26:59: ISAKMP:      attributes in transform:
20:26:59: ISAKMP:      encaps is 1
20:26:59: ISAKMP:      SA life type in seconds
20:26:59: ISAKMP:      SA life duration (basic) of 600
20:26:59: ISAKMP:      SA life type in kilobytes
20:26:59: ISAKMP:      SA life duration (VPI) of 0x0 0x46 0x50 0x0
20:26:59: ISAKMP:      authenticator is HMAC-MD5
20:26:59: ISAKMP (8): atts are acceptable.
```

ISAKMP has found a matching transform set and will begin negotiating the security association. A security association will be made in both directions: one for inbound IPSec traffic, and one for outbound traffic.

```
20:26:59: ISAKMP (8): processing NONCE payload. message ID = 767162845
20:26:59: ISAKMP (8): processing ID payload. message ID = 767162845
20:26:59: ISAKMP (8): processing ID payload. message ID = 767162845
20:26:59: ISAKMP (8): Creating IPSec SAs
20:26:59:           inbound SA from 192.168.55.1 to 192.168.55.2
(proxy 192.168.55.1 to 192.168.55.2)
20:26:59:           has spi 454886490 and conn_id 9 and flags 4
20:26:59:           lifetime of 600 seconds
20:26:59:           lifetime of 4608000 kilobytes
20:26:59:           outbound SA from 192.168.55.2 to 192.168.55.1
(proxy 192.168.55.2 to 192.168.55.1)
20:26:59:           has spi 75506225 and conn_id 10 and flags 4
20:26:59:           lifetime of 600 seconds
20:26:59:           lifetime of 4608000 kilobytes
```

We now have a successful tunnel. These are some of the show and debug commands that I find most useful. There are plenty of others for you to explore that you may find easier to use. Explore them all, as any of them can prove to be useful in troubleshooting VPN.

Advantages and Disadvantages of VPN

One advantage to VPN technology is that it has become highly scalable through digital certificates and public key infrastructure (PKI). Digital certificates are a means of authenticating a user or device. The certificate

is created and signed by a trusted third party who verifies that the user or device is who they say they are. PKI systems, such as the Rivest, Shamir, and Adelman (RSA) system, use a public and a private key pair. The private key is kept by the device, and the public key is made available to remote devices. An association takes place when a device encrypts and sends data using its private key. The receiver then decrypts the information using the peer's public key. The fact that the information could be decrypted using the sender's public key is a verification that the information must have originated from that sending device, as only the public key of that device could decrypt information created with the sender's private key.

Using digital certificates and PKI allows ease of management and scales to thousands of devices and/or users. Certificate technology uses certificate revocation lists to revoke the certificate of devices that are no longer being used, may have been compromised, or have been administratively cancelled. A device using digital certificates will check the revocation list, and if a certificate is no longer valid, authentication will not take place. This eases management tremendously, as certificate tracking, validation, and revocation are handled by the trusted third party, which allows engineers and administrators to focus on other tasks.

Another advantage is in the ease of installation. Most companies already have plenty of leased lines and an Internet connection, which makes installation of remote access networks incredibly easy, as that's all that is usually necessary to configure the peers. As you can see from this chapter, configuration is not a difficult task. Businesses can use resources that are already in place, saving both time and money. As we all know, most of the time spent in getting remote access to a new site is in provisioning leased lines. With VPN technology, a remote access network can be built in minutes by companies, if they have a connection to the Internet; their bandwidth on those lines to the Internet can be increased with just a phone call to the carrier.

Although VPN technology is sure to change the face of networking, careful consideration must be made when using it as a solution—it may not always present a viable solution. One disadvantage is the use of Quality of Service (QoS). QoS cannot be guaranteed over most public infrastructures like the Internet because of the varying paths data must take to get to a destination. The various paths fall under different companies' administrative control and may not implement the same or a compatible (if any) QoS policy. Care should be taken when considering a VPN over a public infrastructure for time-sensitive data.

Cisco's VPN Solutions

This section introduces some Cisco VPN concentrators and other products designed to enable secure communication and manage powerful, scalable VPN solutions. Cisco provides solutions for all levels of organization from small offices/home offices (SOHOs) to enterprise and carrier class companies.

FW Solution (HW Accelerator)

Hardware acceleration takes the process of encrypting and decrypting traffic off the central processor and moves it to the processor on the add-in card. This allows scalability for VPN on Cisco's standard product line of routers and firewalls without the immediate need for VPN concentrators. The need for hardware acceleration is punctuated by the more processor-intensive algorithms being deployed, such as 3DES. The act of running data through an encryption scheme can eat a lot of processor cycles, affecting the routing or security functions of a firewall or router. When performance suffers from utilizing VPN technology, it is time to look at a hardware accelerator or VPN concentrator.

3000 Series Product Line

The Cisco 3000 Series product line is a series of five different VPN concentrators meant to meet the needs of small- to medium-business VPN solutions. The 3000 series has high availability features and is highly scalable using field-swappable components, allowing the upgrade to be performed by the customer.

The 3005 is for small- to medium-sized organizations and supports up to full-duplex T1 or E1 connections and has 4 Mbps encryption performance. This box will support 100 users, and encryption is done through software. There are no modular slots, and system memory is fixed at 32 MB. There is no dual-power-supply option.

The 3015 also supports 100 users, its encryption is also done in software, and it has the same encryption performance as the 3005. This box has four expansion slots, is upgradable, comes with 64 MB of RAM, and has an optional dual power supply as well as optional multichassis redundancy.

On the higher end of the 3000 line are the 3030, 3060, and 3080. These concentrators use hardware for encryption and have much higher encryption performance.

The 3030 can support up to 1500 users with encryption throughput at 50 Mbps. This box has optional redundant power supplies and encryption hardware. It comes with 128 MB of RAM and three expansion slots.

The 3060 can support up to 5000 users with 100 Mbps encryption throughput. It has two encryption modules, with an option for a redundant encryption module. It has two expansion slots, comes with 256 MB of RAM, and has optional redundant power supplies and optional multi-chassis redundancy.

The top of the line 3080 can support up to 10,000 users at 100 Mbps encryption throughput. It uses four encryption modules and has a redundant hardware encryption module and redundant power supply as part of the standard package. It comes with 256 MB of RAM but no expansion slots.

VPN concentrators can be used side-by-side with a firewall, as shown in Figure 4.14; inline with a firewall, as shown in Figure 4.15; or stand-alone, without the use of a firewall. The third option is the most secure form of communication, as only authenticated, encrypted traffic can traverse the concentrator—leaving no open holes for hackers to explore.

Figure 4.14 VPN concentrator side-by-side with a firewall.

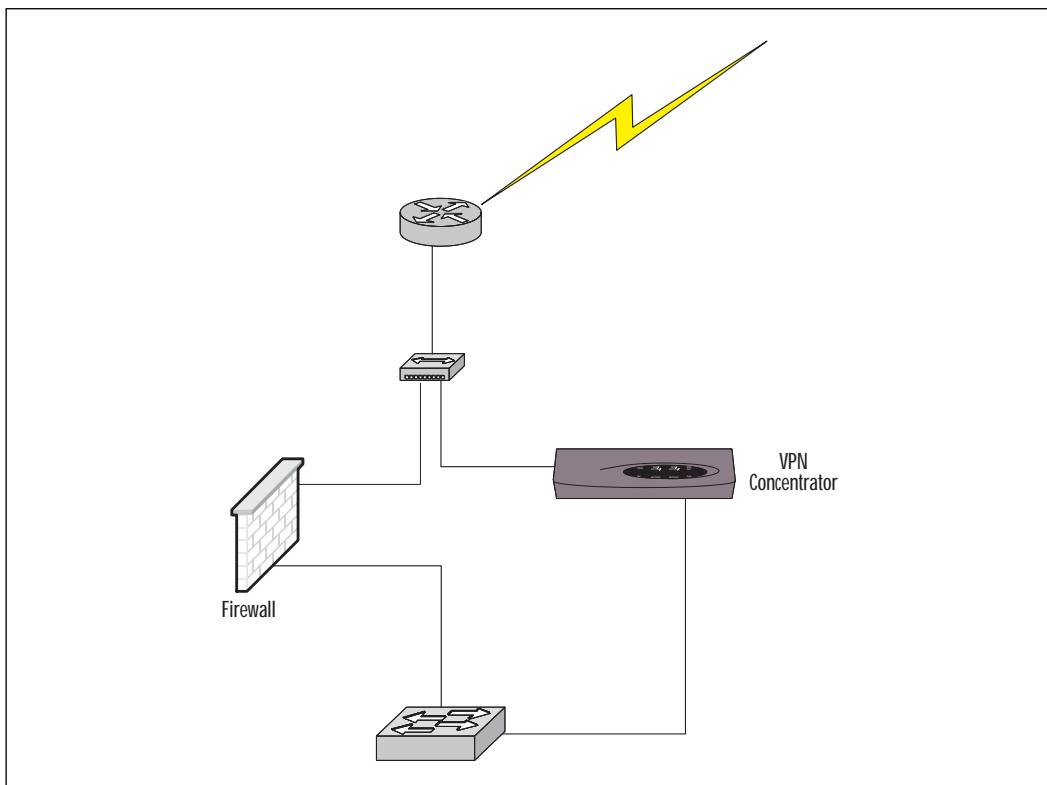
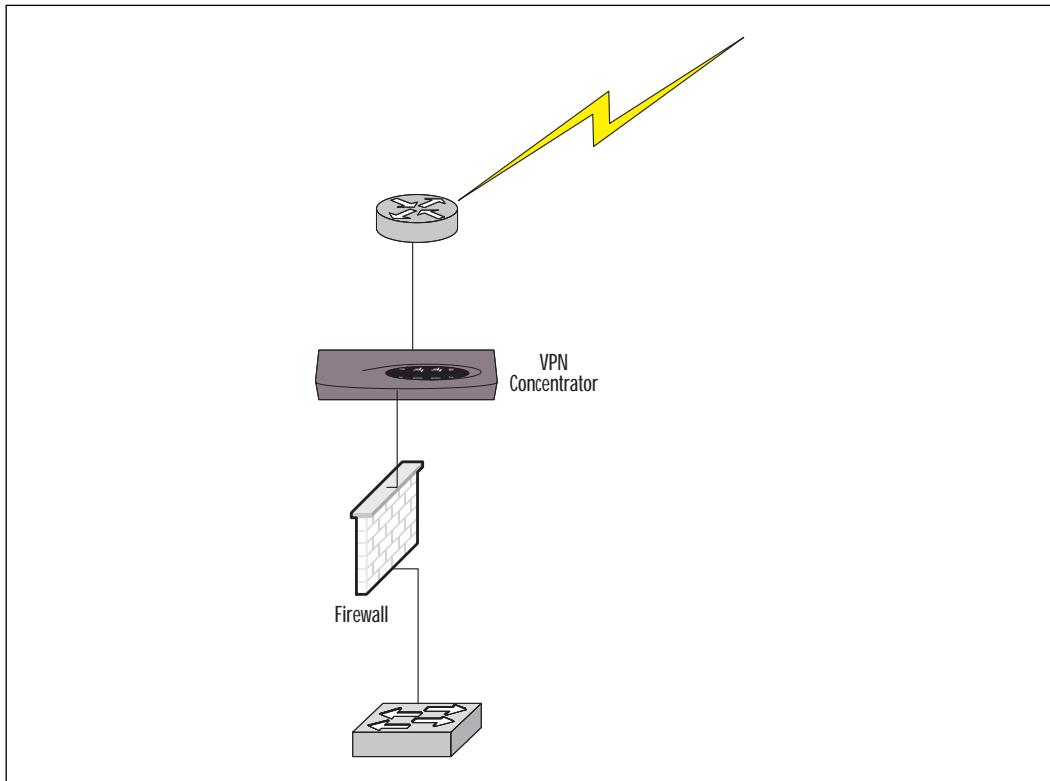


Figure 4.15 VPN concentrator inline with a firewall.



Traditional Router with FW Feature Set

The Cisco Firewall feature set is used primarily on perimeter routers as a first line of defense before traffic arrives at the firewall. It is also used in smaller environments to secure the entire network. The firewall feature set implements many of the capabilities of a standard firewall and can do encryption with the FW Plus IPSec.

Policy Manager 2.x (VPN Configuration and Management)

The CiscoSecure Policy Manager provides a graphical interface for defining, distributing, and enforcing security policies across the enterprise. It enhances productivity by giving an “overhead” view of the security in the enterprise and allows configuration of the whole enterprise, as opposed to a device-by-device approach. CiscoSecure Policy Manager is a scalable

security policy management system that provisions security policy throughout the organization. It can be used to configure firewalls and VPNs in a uniform manner. CiscoSecure Policy Manager allows you to define, distribute, enforce, and audit enterprise-wide security policies from a central location including perimeter access control, Network Access Translation, and IPSec VPNs.

By using the graphical interface, you can build your network topology, define your security policies, push them out to the devices, and then use the monitoring and reporting tools to keep an ongoing audit and generate on-demand reports of enterprise security.

Some of the features and benefits of the Policy Manager are:

Scalability Support for up to 500 firewalls and VPN routers.

Multiple topologies Internet, intranet, or extranet.

Secure communications Local and remote management via IPSec tunnels or proprietary PIX Secure Telnet method.

Templates Supplies templates to assist security administrators in creating policy, and provides IPSec VPN templates for assistance in creating VPN tunnels.

Offline configuration support Configure and test security policies offline.

NAT Easy NAT implementation.

Consistency checking Checks policy integrity prior to distribution.

Rollback mechanism Auto rollback to previous working policy.

Summary

VPN technology can be used to create remote access networks over various public or private infrastructures, from the Public Switched Telephone Network using dial-up connections, to secure communications across the Internet and point-to-point leased lines. VPN technology can be used to leverage current topology and in-place communications lines, is easy to configure, and has minimal costs associated with implementation. VPN technology can be used to secure communications between hosts, between host and network, or between networks. Because VPN technology is highly scalable in both hardware and software—and because it is easily managed, easy to implement, and minimal in cost—we will continue to see growth in VPN networks. Implementation will likely accelerate at an increased rate as more personnel gain the knowledge to configure these networks.

FAQs

Q: Can I allow remote users to access the DMZ on a PIX firewall using IPSec?

A: Yes, by changing the access-list and nat 0 statement to reflect the DMZ you want to give access to.

Q: Is CISCO's IPSec compatible with devices running the older Cisco Private Link encryption?

A: Yes. On the firewall you would issue the **sysopt ipsec pl-compatible** command.

Q: Can I configure a VPN tunnel between two devices that may not be from Cisco, or between a Cisco and non-Cisco device?

A: If both devices follow the IPSec standard, then yes—however, some vendors do not follow the standard explicitly, so you must be careful and ask the vendor. In most instances, I have found that if both vendors do follow IPSec, than you can create a tunnel. In some instances you may not be able to use ISAKMP between the devices, because of various implementations of the open framework; however, this is overcome by doing IPSec with manual keys, as opposed to using ISAKMP.

Chapter 5

Using ISDN and DDR to Enhance Remote Access Connectivity

Solutions in this chapter:

- ISDN
- DDR overview
- Legacy DDR

Introduction

ISDN stands for “Integrated Services Digital Network,” and is an International Telecommunication Union Telecommunication Standardization (ITU-T) term for a digital technology that replaces traditional analog telephone equipment with new high-speed digital equipment. While previous chapters in this book were about using analog communications to provide remote connectivity, this chapter will concentrate on how to take advantage of ISDN and dial-on-demand routing (DDR) to enhance remote connectivity.

DDR can be used with technologies such as ISDN and Public Switched Telephone Networks (PSTN), and allows connections to be established and disconnected on an as-needed basis, which can result in substantial cost savings. There are two types of DDR configuration: legacy DDR and dialer profiles. This chapter will concentrate on legacy DDR configuration, and Chapter 6 will deal with optimizing DDR with rotary groups and dialer profiles.

Because costs are incurred when dial-up connections are established, it is generally not advisable to run the same dynamic routing protocols on DDR links as on permanent links. The final section of this chapter discusses the routing issues that occur when implementing DDR solutions, and the various options available to us for maintaining routing tables without a permanent connection.

ISDN Overview

ISDN is different from standard telephone service in that it is a digital network, whereas the standard telephone, or PSTN, is an analog network. There are several disadvantages to the PSTN. One key disadvantage is the fact that computers must convert digital data into an analog stream to transmit over the PSTN, and then re-convert back into digital data at the other end. Another disadvantage of the PSTN is that it was developed purely for transmission of voice communications, limiting its data bandwidth and transmission quality. The maximum speed for analog data transfer across PSTN networks is 33.6 Kbps. In addition, analog modem connections require a significant amount of time to establish.

ISDN was developed to fix the problems encountered in the PSTN. In order to make ISDN a public network, standards had to be developed for all companies to follow. The International Consultative Committee for Telegraph and Telephone (CCITT) developed the ISDN standards and specifications. The ITU-T replaced the CCITT.

NOTE

In order to get 56 Kbps modem connections, one end (typically the receiving end) must be completely digital. When dialing into an Internet service provider (ISP) and connecting at speeds greater than 33.6 Kbps, a Primary Rate Interface (PRI) line is most likely being used at the ISP end.

ISDN is a group of digital services allowing high-speed transmissions of data, voice, and video. It is an end-to-end digital services network. The ITU-T developed groups of standard protocols separated by content. The first group is called the *E series*. The E series protocols deal with telephone network standards for ISDN. The second group is called the *I series*. The I series protocols deal with various aspects of the ISDN standard. The I series is separated into the following groups:

- I.100 General Concepts and Terminology
- I.200 Service Aspects
- I.300 Network Aspects
- I.400 User-Network Interfaces
- I.500 Internetwork Interfaces
- I.600 Maintenance Principles

The third group is the *Q series*. The Q series standards deal with call setup and switching processes. For a complete list of each of these standards, as well as all other ITU-T standards, go to www.itu.int/itu-t/rec.

The ISDN standards focus on how the end-user communicates with the network. In addition to the ITU-T, there are several other organizations involved in setting the standards for ISDN. These organizations work together and, through the American National Standards Institute (ANSI), develop the standards for ISDN.

ISDN is composed of a group of channels distinguished by function and bit rate. There are three different channels in the ISDN service model: B-channel, D-channel, and H-channel. ISDN lines can be ordered in several different configurations of grouping of these channels. Basic Rate Interface (BRI) and PRI are two of the most common groupings.

Below are details of the three channel types. The following section covers the BRI and PRI lines in more detail.

The B-channel is used for user services including data, audio, and video, and operates at 64 Kbps (56 Kbps in older equipment) in full-duplex mode.

NOTE

One key difference between ISDN and analog transmission is the duplex mode. Analog transmissions operate at half-duplex; they can only send data or receive data, not both at the same time. Digital transmissions operate at full-duplex; they can send and receive data at the same time.

The D-channel is used for signaling between the user and the network, and can carry user packet mode data. The D-channel operates at either 16 Kbps or 64 Kbps in full duplex, depending on the interface in use. Both the B- and D-channels are fully digitized.

The H-channel is used in applications that require bit rates higher than the 64 Kbps offered in the B-channel. There are four H-channels: H0, H10, H11, and H12. H0 is equivalent to six B-channels operating at 384 Kbps. The H10 channel is equivalent to 23 B-channels operating at 1.472 Mbps. The H10 channel has been defined by ANSI but is the only H-channel not standardized by the ITU-T. The H11 channel is used when the circuit is a T1 line. The H11 channel is equivalent to 24 B-channels operating at 1.536 Mbps. The H12 channel is used when the circuit is an E1 line. The H12 channel is equivalent to 30 B-channels operating at 1.92 Mbps.

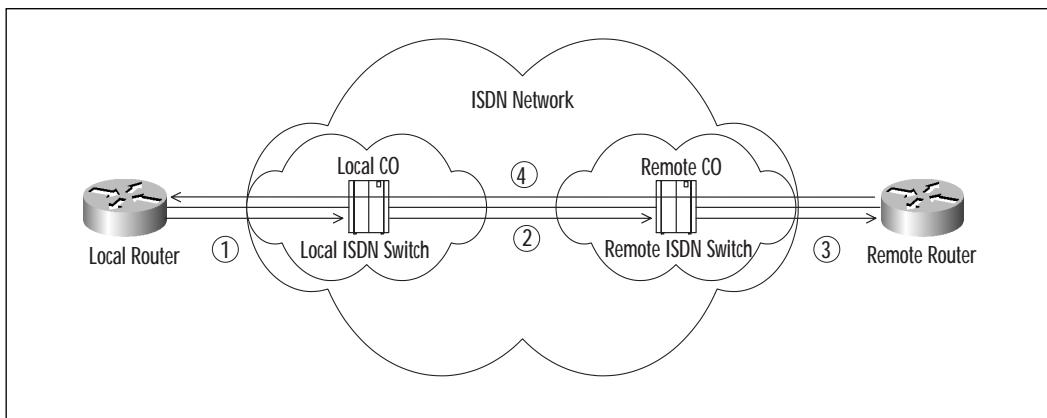
Basic Rate Interface (BRI)

Small businesses and home users typically use the BRI for remote connectivity and the Internet. Another use for BRI lines is as a backup connection should a primary wide area network (WAN) link fail. The BRI is comprised of two B-channels and one D-channel. It is referred to as 2B+D. The available bandwidth of the BRI is $2 \times 64 \text{ Kbps} + 16 \text{ Kbps} = 144 \text{ Kbps}$. There are an additional 48 Kbps of bandwidth required to allow the physical connection to operate, giving a total bit rate of 192 Kbps. However, in most cases, the usable bandwidth for data across a BRI line is 128 Kbps.

BRI Call Setup

Figure 5.1 shows how a BRI call is set up. Only the D-channel is involved in setting up and breaking down an ISDN call.

Figure 5.1 ISDN BRI call setup process.



The following describes what happens at each numbered step in the call setup process shown in Figure 5.1.

1. The D-channel initiates a call. The called number is sent to the Central Office (CO) ISDN switch.
2. The CO ISDN switch sets up a path to the destination switch using the SS7 protocol.
3. The remote switch sends a signal to the remote D-channel activating the remote end.
4. The remote end answers the call and establishes a data session through the B-channel.

BRI Reference Points and Functional Groups

ISDN reference points identify architectural separations at the customer's site. The functional groups identify the equipment involved in ISDN BRI circuits. Figure 5.2 visually shows the reference points in relation to the functional groups.

The functional groups from Figure 5.2 are:

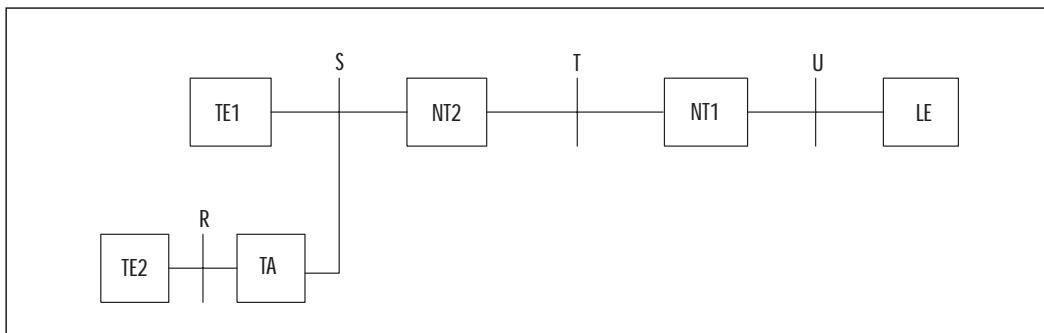
- **TE2** Terminal Equipment 2 is a device that is not compatible with ISDN, such as an analog telephone or a router without an ISDN interface.
- **TA** The Terminal Adapter converts standard electrical signals from non-ISDN devices into a form compatible with ISDN. The TA is the link between non-ISDN equipment and the ISDN network.

- **TE1** Terminal Equipment 1 is a device that is compatible with ISDN, such as a digital telephone or a router with an ISDN interface.
- **NT2** The Network Termination 2 device directs traffic to and from the user devices and the NT1, such as a private branch exchange (PBX).
- **NT1** The Network Termination 1 device connects the ISDN wiring (four-wire ISDN wiring) to the conventional local loop (two-wire standard wiring).
- **LE** The Local Exchange is the ISDN switch residing in the CO.

The reference points from Figure 5.2 are:

- **R (Rate)** Reference point between TA and non-ISDN device.
- **S (System)** Reference point between NT2 and TE1 or TA that connects the terminals to the ISDN network. The System reference point is the most important point for users.
- **T (Terminal)** Reference point between NT2 and NT1. Both the T and S reference points use the same characteristics and are often represented as S/T.
- **U (User)** Reference point between NT1 and LE, which is only specified by ANSI (not by CCITT) and is only used in North America.

Figure 5.2 ISDN BRI reference points and functional groups.



Primary Rate Interface (PRI)

PRI lines are used where more bandwidth is required. They are also used as a dial-up access line giving an organization up to 30 (23 in North America and Japan) 64 Kbps dial-in lines. There are several different con-

figurations for the PRI. In North America and Japan, the configuration is noted as 23B+D, or 23 B-channels and one D-channel operating at 64 Kbps. The bit rate of this type of PRI is 24×64 Kbps = 1.544 Mbps.

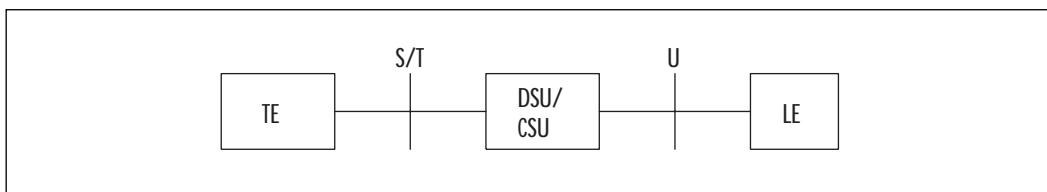
Another configuration of the PRI is noted as 30B+D. This PRI offers a bit rate of 2.048 Mbps and is commonly offered in Europe and Australia.

PRI Reference Points and Functional Groups

The reference points for PRI lines are simpler than for BRI lines. The functions of the reference points are the same as in the BRI line. The major difference is that PRI does not support multiple ISDN devices on the same line, whereas a BRI network supports connecting multiple devices to the same line.

As shown in Figure 5.3, in PRI lines the Terminal Equipment (TE) connects directly to the Data Service Unit/Channel Service Unit (DSU/CSU), which then connects to the Local Exchange (LE). The DSU/CSU is similar to a modem but does not convert digital signals into analog signals. Since there is no support for non-ISDN multiple devices, the reference points and functional groups for the PRI line can be kept simple.

Figure 5.3 ISDN PRI reference points and functional groups.



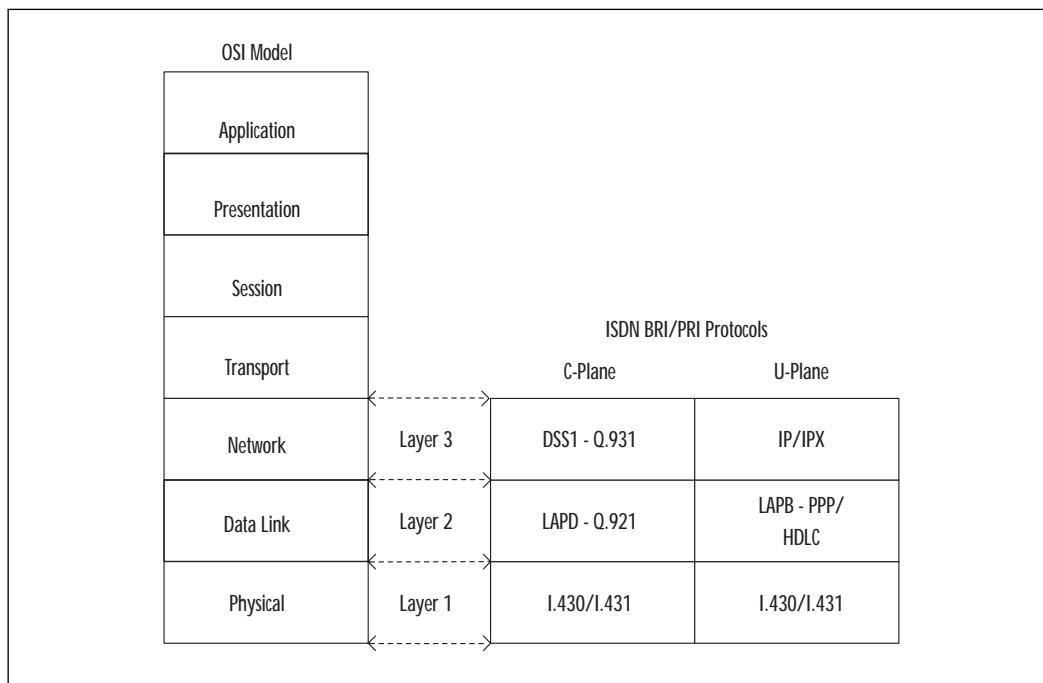
ISDN Protocol Layers

ISDN uses several different protocols for both control signaling and user data. The protocols can be correlated to the Open System Interconnection (OSI) reference model. The OSI reference model regulates all communication between systems to ensure interoperability between vendors. The OSI reference model consists of seven functional layers including: Physical, Data Link, Network, Transport, Session, Presentation, and Application. Since signaling protocols and user data protocols are different, yet still operate in the same OSI layers, it further divides the OSI model into protocol planes. The user plane (U-plane) contains the protocols required for sending user data such as voice, video and data. The control plane (C-plane) contains the protocols necessary for exchanging control signaling. Finally, the management plane (M-plane) controls the flow of traffic.

between the U-plane and C-plane. All of these planes can operate on the same layers of the OSI model simultaneously. ISDN services or bearer services operate at the first three layers of the OSI model (see Figure 5.4). These services allow for processing information for user-to-user communication and for transmitting all processed information. The actual processing of information takes place at Layers 4 through 7 of the OSI model, which are the responsibility of the computer, not the network.

As mentioned earlier, the B-channel carries user data that directly correlates to the U-plane, and the D-channel carries signaling information that directly correlates to the C-plane. In the next section, we will discuss the three layers that ISDN uses and we will discuss the relevance of both the U-plane and the C-plane.

Figure 5.4 OSI reference model and ISDN protocols.



U-plane

At Layer 1, or the physical layer, the B-channel is specified by both I.430 for BRI functionality and I.431 for PRI functionality. At this layer, the B-channel performs circuit switching, packet switching, and leased circuitry. For both circuit-switched and leased circuits, control signals set up the circuit and the ISDN network does not need to use any Layer 2 or 3 proto-

cols. When a packet-switched circuit is set up, the X.25 protocols run at Layers 2 and 3 allowing the exchange of data. The Layer 2 protocol for packet-switched circuits is known as Link Access Procedure for the B-channel (LAPB). Once LAPB establishes the Layer 2 connection, the Layer 3 connection can be established. Layer 3 protocols on the B-channel can be any OSI Layer 3 protocol such as Internet Protocol (IP) or Internetwork Packet Exchange (IPX).

C-plane

The D-channel operates at the same physical medium as the B-channel. Because of this, its physical layer protocols are the same as B-channel on both the BRI and PRI. For the D-channel, the Layer 2 protocol for packet-switched circuits is known as Link Access Procedure for the D-channel (LAPD). LAPD is specified under ITU-T Q.920 and Q.921 standards. The CCITT did not make LAPD a requirement, only a recommendation (I.440 and I.441). The D-channel has several Layer 3 protocols to choose from. The most commonly used Layer 3 protocol is Q.931.

ISDN Call Setup and Teardown

Figure 5.5 shows how the call setup process takes place using the Q.931 protocol. Not every ISDN switch uses the same procedures for both call setup and teardown. Figures 5.5 and 5.6 show the setup and teardown of a typical ISDN switch. In addition to the steps shown, an optional progress message can also pass through the system. Not all of these messages are required to take place when placing an ISDN call.

Dial-on-Demand Routing (DDR)

DDR is a technology that routers use to dynamically initiate and close a circuit-switched session to remote routers on demand. Once these sessions have been connected, data as well as routing updates can be exchanged between routers. In order for the router to initiate this session, it must first know when to dial. This is done through what is called *interesting traffic*. Once the call has been established, data can pass to the other end. The DDR session is typically not broken until there is a period of inactivity called *idle-time*. Multiple locations can be configured to dial based on routing destination. There are several features built into DDR that enhance its operation. Most of the more popular features, such as PPP Multilink and Dial Timers, will be covered in the remainder of this section and in Chapter 6.

DDR typically runs on an as-needed basis, meaning the session is not connected until necessary. By running DDR on an as-needed basis, companies can save significant WAN usage costs. DDR operates over circuit-switched networks like ISDN and PSTN. Some of the methods using DDR are legacy DDR, dialer profiles, dial backup, and snapshot routing. All of these methods will be covered later in this chapter.

Figure 5.5 ISDN D-channel call setup.

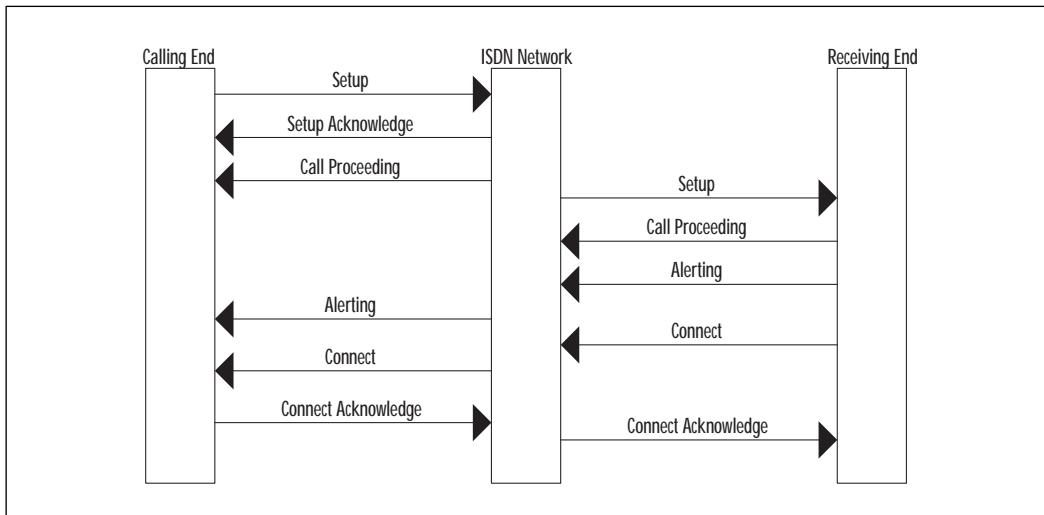
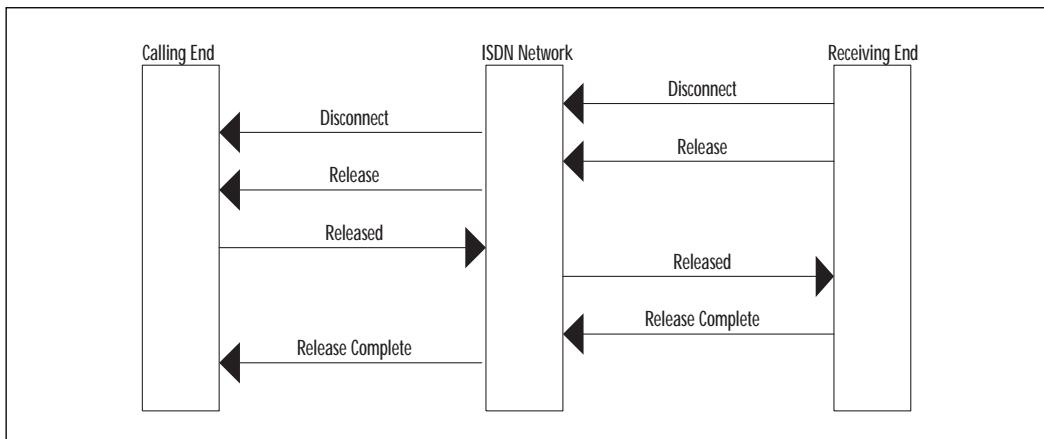


Figure 5.6 ISDN D-channel call teardown.



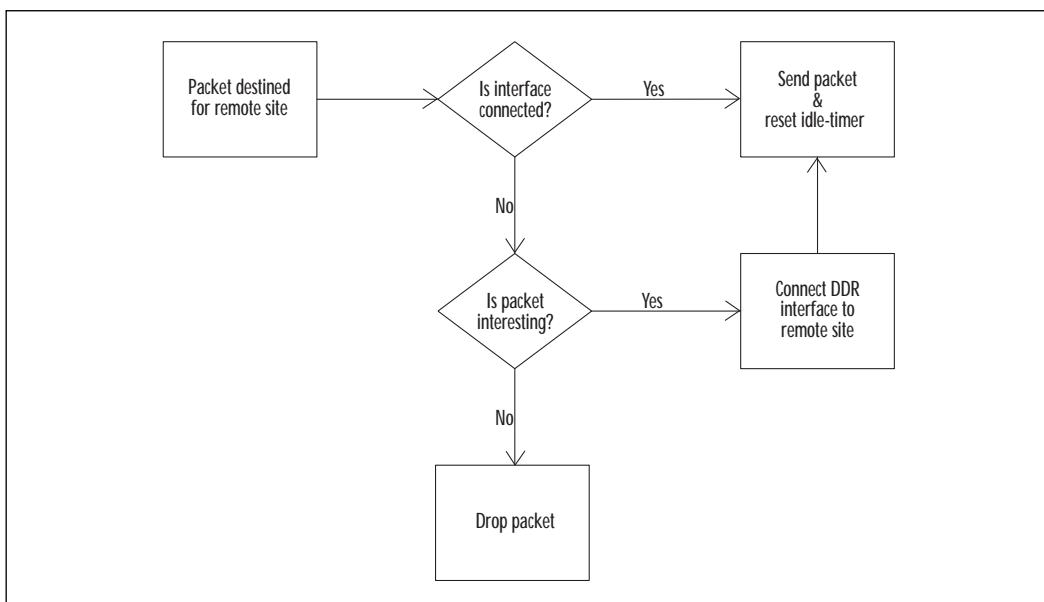
Interesting Traffic

The mechanism that allows DDR to function is the definition of interesting traffic. Interesting traffic is defined as traffic the router deems important (based on access lists); all other traffic is deemed *uninteresting*. When interesting traffic enters the router destined for a remote network, the router establishes a call to the remote network and sends the data (see Figure 5.7). Once the circuit is connected, all traffic (including uninteresting traffic) can flow through the circuit. In the event of uninteresting traffic coming into the router destined for a remote network, the router will not establish a new call and the uninteresting traffic will be dropped.

Interesting traffic is configured on the router with the dialer-list command. The dialer-list command is then associated with a protocol and then permitted, denied, or matched to an access list. An example of an interesting traffic definition is **dialer-list 1 protocol ip permit**. This would allow IP traffic entering the router and destined for the remote network or networks to trigger a DDR session. Another example is:

- dialer-list 2 protocol ip list 101
- dialer-list 2 protocol ipx list 901
- dialer-list 2 protocol appletalk deny

Figure 5.7 Dial-on-demand logic.



The previous dialer-list would deny all Appletalk traffic from initiating the DDR session, and would look at access list 101 for matches on IP traffic and access list 901 for matches on IPX traffic. If an IP or IPX match were found, the DDR interface would dial. One reason you would want to configure an access list permitting only specific traffic to initiate a DDR call would be for permitting only e-mail and Web traffic. In that instance, other traffic such as routing updates and broadcasts would not initiate a DDR session. If dynamic routing protocols were allowed to trigger the DDR interface, the link would stay connected all the time. The limit on the number of dialer-lists in a router is 10, but each list can have multiple entries. It is important to remember to use an access list when using DDR and dynamic routing to prevent routing updates or hello packets from opening and keeping the link active.

NOTE

Once a DDR connection has been made, any traffic passing through the interface (including uninteresting traffic) will keep the session open.

Topologies

There are three topology designs possible under DDR. The topology chosen depends on the number of sites in the design and the amount of traffic between the sites. The three possible topologies are:

- Point-to-point
- Fully meshed
- Hub-and-spoke

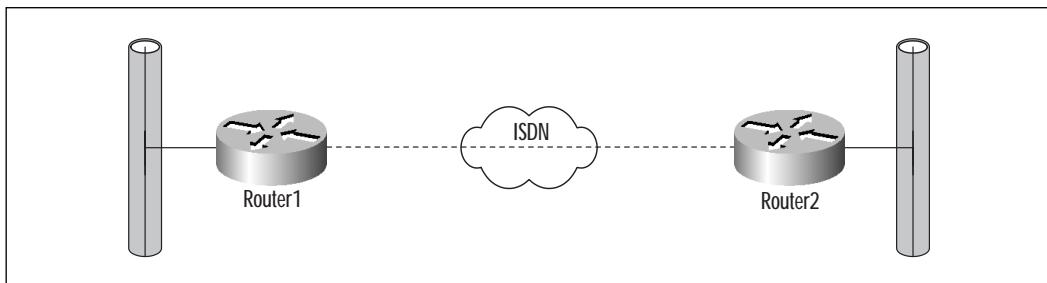
Point-to-Point Topology

If there are only two sites involved in the design, point-to-point topology should be used. For point-to-point topology to work, each site is configured to dial the other. Another option is to use multiple links to give additional bandwidth. Figure 5.8 shows a point-to-point topology.

Fully Meshed Topology

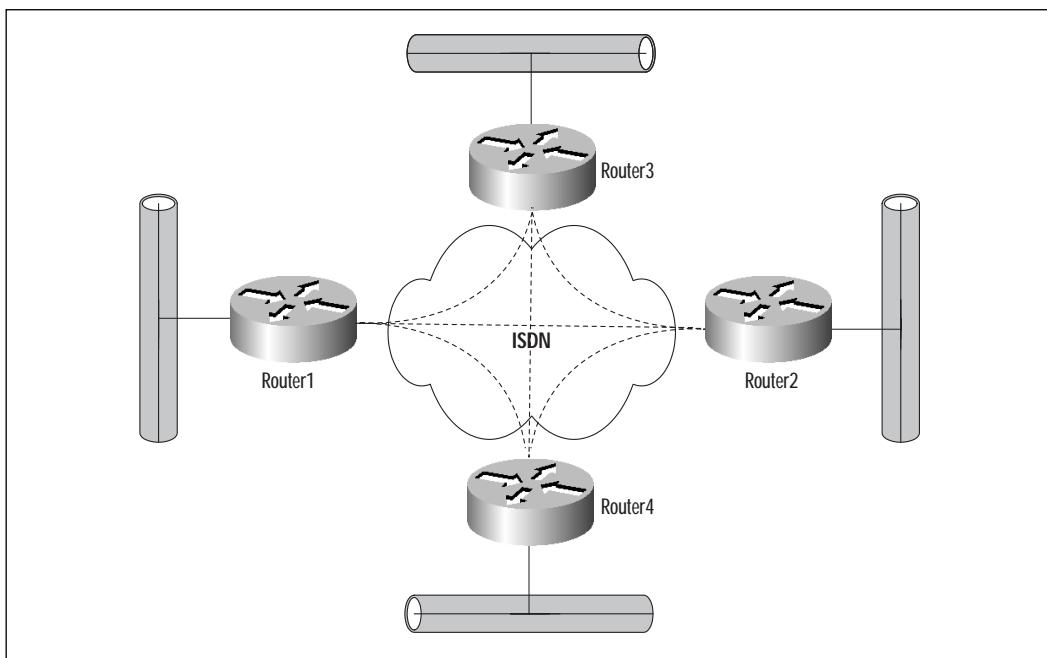
A fully meshed network topology is only recommended for a very small DDR network. In the fully meshed design, each router is configured to dial every other router in the network. An advantage of this design is that it

Figure 5.8 Point-to-point DDR topology.



allows each site to communicate directly with each other site instead of going through a central site. However, with this design, the scalability is severely limited. You must also take the number of available ports and circuits into consideration. If you have the network shown in Figure 5.9, and Router1 is connected to Router2, and Router3 is connected to Router4, then data cannot pass between Router1 and Router3 or Router4, and cannot pass between Router2 and Router3 or Router4. Just like any fully meshed topology, the amount of resources required to maintain a full mesh grows exponentially with the number of devices.

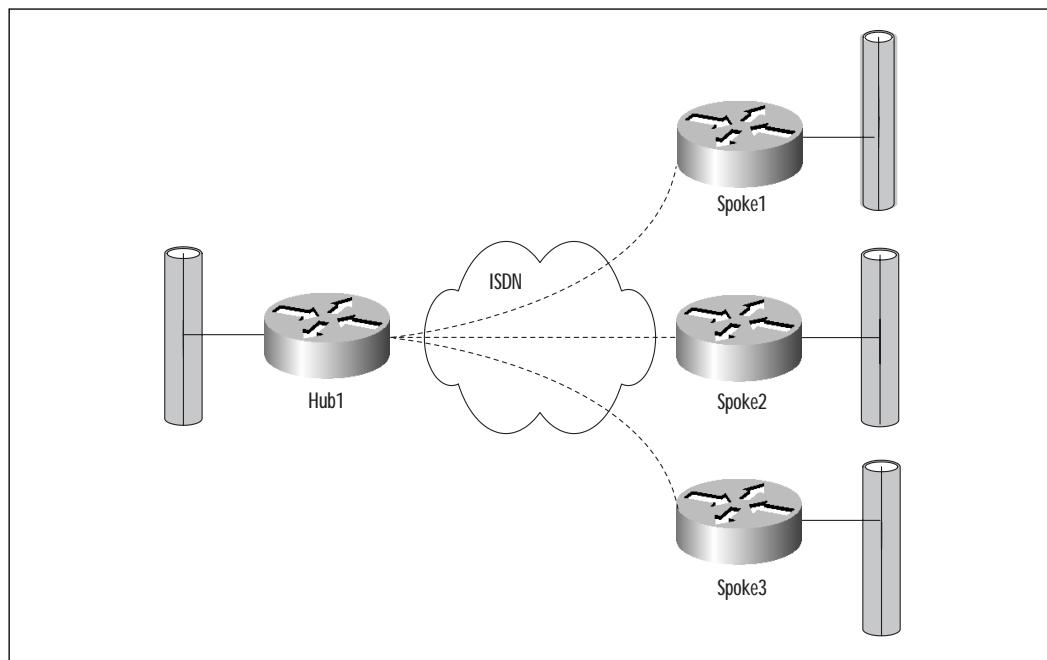
Figure 5.9 Fully meshed DDR topology.



Hub-and-Spoke Topology

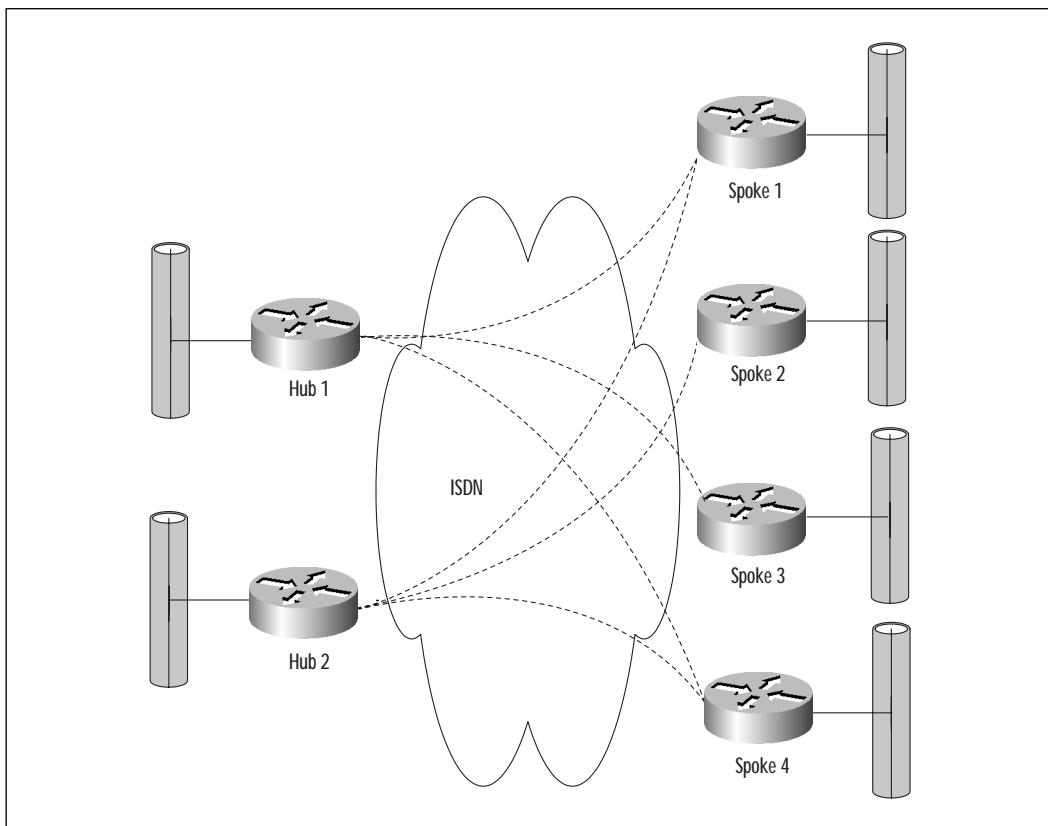
A hub-and-spoke network topology is different from the fully meshed design, in that all traffic is sent to a central site and then re-routed to the final destination. For example, in Figure 5.10, if a computer on Spoke2's Ethernet interface wanted to send an e-mail to a computer on Spoke3's Ethernet segment, Spoke2 would dial Hub1 (assuming that the e-mail was configured as interesting traffic), which would then dial Spoke3 and send the data. Hub1 would be taking in the data from Spoke2 and sending it out to Spoke3. This type of design is more suitable for large-scale DDR networks. In order for this type of design to scale properly, the only site that needs to have significant available resources is the hub. Contrary to the exponential growth in resources (circuits and ports) required in a fully meshed design, the hub-and-spoke design only needs resources two times the number of DDR sites. Another advantage of the hub-and-spoke design is that it is easy to configure and troubleshoot. The complexity of the design is constrained to the hub router; the spoke routers have very simple configurations. One key disadvantage to this design (but not to the fully meshed topology) is that there is now a single point of failure in the network. If the hub router goes down, then none of the hub sites are able to communicate with the rest of the network.

Figure 5.10 Hub-and-spoke DDR topology.



One popular solution to overcome this potential failure issue is to design a dual-hub-and-spoke network. This works well on large networks, retains the advantages of the hub-and-spoke design, and overcomes the issue of a single point of failure. Figure 5.11 shows a dual-hub-and-spoke design.

Figure 5.11 Dual-hub-and-spoke DDR topology.



Dialer Interfaces

There are a few different interfaces that Cisco routers can use as a dialer interface: ISDN BRI, synchronous serial, and asynchronous. In order to have an understanding of dialer interfaces, it is important to have an understanding of dialer profiles, dialer rotary groups, dialer addressing, dialer mapping, encapsulation, and supported interfaces. The following sections cover these concepts.

Dialer Profiles

Dialer profiles were introduced into the IOS to offer design flexibility in DDR networks. They are key to the function of dialer interfaces. Dialer profiles are based on separate logical interface configurations being bound to physical interfaces. They involve configuring a profile, which is kept separate from the physical interface. Once the profile has been configured, it is then bound to the physical interface. Multiple profiles can then be linked to one interface, allowing multiple sites to be called from the same interface. Additionally, one profile can be linked to multiple interfaces, allowing greater bandwidth per call. Chapter 6 gives more details on dialer profiles, including configuration examples.

Dialer Rotary Groups

Dialer rotary groups are used when there are multiple physical interfaces placing a call. In the event one interface is busy, the rotary group will use the next available interface to make the call. A dialer rotary group does not need to be configured for either BRI or PRI interfaces; the multiple B-channels in either interface are automatically placed into a dialer rotary group. Chapter 6 gives more details.

Dialer Addressing

There are two different ways to assign dialer interface addresses: using unnumbered interfaces and shared subnetting.

Unnumbered interfaces are similar to assigning a point-to-point line an unnumbered address; the address of another interface on the router is used on the dialer interface. Using unnumbered dialer interfaces works because the links are always point-to-point.

In using shared subnetting, the dialer interface is similar to assigning a subnet to a LAN or multipoint WAN to share. For shared subnetting, each site in the dialer cloud would get a unique address from a subnetted pool. Using shared subnetting is much simpler than using unnumbered addresses; however, it consumes extra addresses.

Dialer Mapping

Dialer maps translate telephone numbers into next-hop addresses. DDR cannot function without statically configured dialer maps. In addition to translating telephone numbers to next-hop addresses, dialer maps control whether an interface passes broadcast messages. Dialer maps can also control the speed of the call, and can link names for PPP authentication. If a site is only going to receive calls and not make any outgoing calls, the phone number can be left off the dialer map statement. Examples B

through F in the “Configuring ISDN and DDR” section all contain examples of dialer maps.

Encapsulation

Once a connection is established between two DDR devices, datagrams must be encapsulated and framed before being sent across the media.

There are several methods of encapsulation available on Cisco routers, and depending on the interface being used, not all methods are available. Cisco routers support Point-to-Point Protocol (PPP), Serial Line Internet Protocol (SLIP), X.25 data-link, and High-Level Data Link Control (HDLC).

SLIP is the predecessor to PPP. SLIP works only over asynchronous interfaces and supports only IP. Additionally, there is no support for authentication or dynamic address assignment. SLIP is not a recommended encapsulation method.

PPP is the recommended encapsulation method for Cisco routers. PPP was developed to overcome problems with SLIP, such as its inability to operate over synchronous serial lines and its lack of dynamic configuration support. PPP supports several protocols and can be used for synchronous serial, asynchronous serial, and ISDN interfaces. PPP also supports authentication and address resolution and is supported by other vendors as well. X.25 is supported on both synchronous serial interfaces and ISDN B-channels.

HDLC is supported on both synchronous serial interfaces and ISDN interfaces. HDLC supports multiple protocols like PPP. Unlike PPP, HDLC does not support authentication and is not vendor-independent.

Supported Interfaces

As mentioned earlier, there are three Cisco interfaces that support ISDN.

ISDN Interfaces

There are two ISDN BRI interfaces used on Cisco routers. One has the NT1 device built in and the other does not. The NT1 device terminates a four-wire ISDN bus and connects it to the two-wire local loop. The reason Cisco offers ISDN interfaces with or without an NT1 device is mainly because a Telco may or may not provide the NT1 device (most in the United States do not). To determine whether the interface has an NT1, all you need to do is look at the RJ-45 port on the router. If the port is labeled U then it has an NT1 built in; if the port is labeled S/T then it does not. If the router has an S/T port then you must connect it to an external NT1 in order to operate over ISDN. Multilink PPP is commonly used in conjunction with ISDN BRI lines. Multilink PPP bonds multiple B-channels together, providing greater bandwidth. Both ISDN BRI and PRI interfaces are automatically configured as dialer in-band interfaces.

An in-band interface is simply an interface that sends dialing information over the same connection that carries the data. ISDN interfaces support PPP, HDLC, X.25, and V.120 encapsulation.

Synchronous Serial Interfaces

There are two ways that synchronous serial interfaces can initiate dialing. V.25bis dialing is the ITU standard for in-band dialing and is used with devices such as synchronous modems, ISDN terminal adapters (TA), and switched 56 Kbps DSU/CSUs. Data Terminal Ready (DTR) dialing is the other method for synchronous serial interface dialing. DTR does not support incoming calls. DTR does, however, allow for lower cost devices to be used when there is only one number that interface calls.

Synchronous serial interfaces support PPP, HDLC, and X.25 encapsulation. To convert a synchronous serial interface into a dialer interface, use the Cisco command **dialer in-band** or **dialer dtr**.

Asynchronous Modem Connections

Asynchronous connections are made through the auxiliary (Aux) port on a router or through the asynchronous ports on a communications server, such as a Cisco 2511 router. Just as with synchronous serial interfaces, you must use the **dialer in-band** or **dialer dtr** command on the interface for DDR operation. Asynchronous DDR connections can support multiple protocols and encapsulations. Some disadvantages of asynchronous DDR designs are they require more time to establish connections than ISDN, and have much lower bandwidth capability than ISDN or synchronous serial connections. If bandwidth and call establishment time are not important, asynchronous DDR can be a cost-effective solution.

In order to use asynchronous DDR, chat scripts must be configured so that dialing and login commands get sent to the remote end. The chat script sends the modem the proper dialing and login commands. Multiple chat scripts can be assigned to dialer maps to allow for additional flexibility. In addition to chat scripts, modem scripts for configuring outbound modems and logon scripts for remote system logon information can be used. There are two examples in the “Configuring ISDN and DDR” section that show how to configure an asynchronous serial interface.

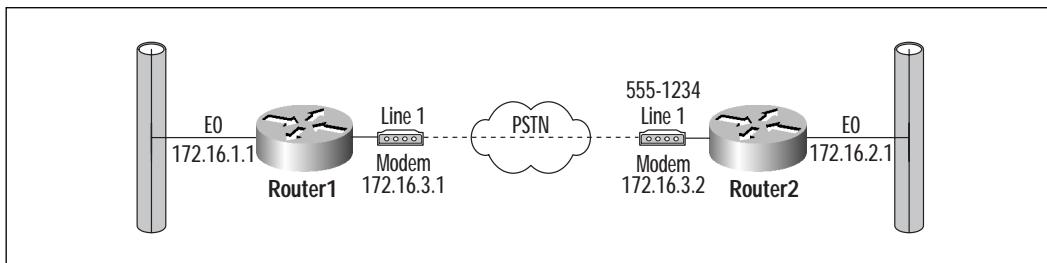
Configuring ISDN and DDR

This section illustrates how to configure the various pieces of DDR and ISDN.

In Example A (Figure 5.12 and Router1 configuration), Router1 will be calling into Router2 through asynchronous interface Line 1. As mentioned earlier, the configuration for a synchronous serial interface would be the

same as an asynchronous serial interface. The configuration of Router1 is shown in Example A with an explanation of each command in Table 5.1. Only the commands required to set up and initiate the call are shown. This example introduces how to configure an interface for DDR operation. Examples B through D expand on DDR operation and introduce ISDN configuration. Each of the examples shows only partial router configurations. For a fully configured router example, refer to the “Walkthrough” section at the end of the chapter.

Figure 5.12 (Example A) Asynchronous one-to-one.



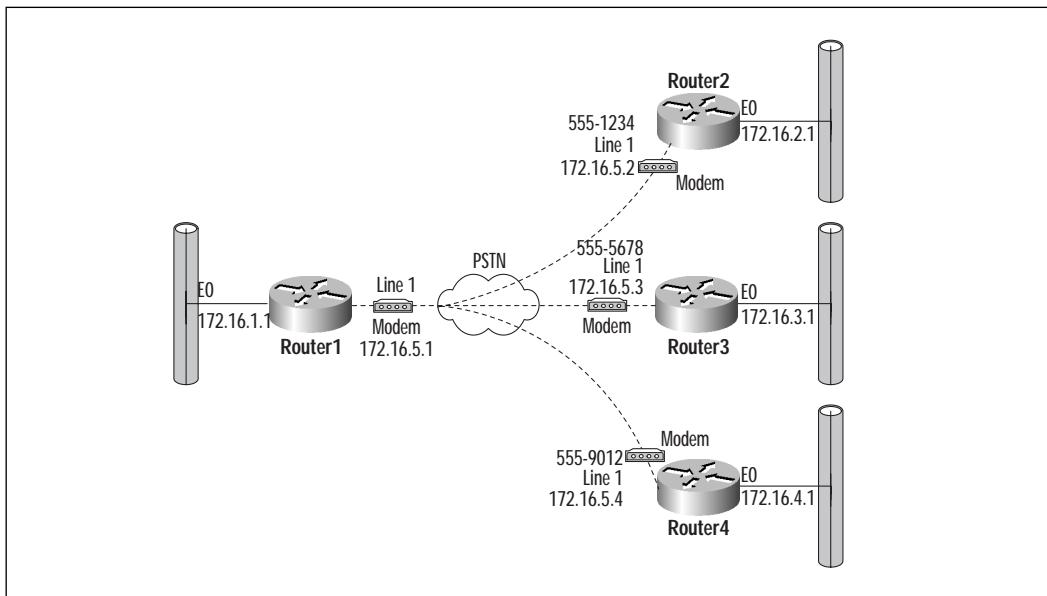
Example A Router1 configuration.

```
Router1(config)#ip route 172.16.2.0 255.255.255.0 172.16.3.2
Router1(config)#dialer-list 1 protocol ip permit
Router1(config)#interface async 1
Router1(config-if)#dialer in-band
Router1(config-if)#ip address 172.16.3.1 255.255.255.0
Router1(config-if)#dialer string 5551234
Router1(config-if)#dialer-group 1
Router1(config-if)#encapsulation ppp
```

Example B (Figure 5.13 and Router1 configuration) shows how to configure a router to dial into several different locations using the same phone line. Commands are explained in Table 5.2. In this example, if the line was connected to Router2 and traffic came into Router1 destined for Router4, the traffic would be dropped. It would be important to control the amount of time the phone line was used to prevent this situation. One command that can help control this is **dialer idle-timeout**, which is covered in Example D.

Table 5.1 Command Descriptions

Command	Description
ip route 172.16.2.0 255.255.255.0 172.16.3.2	This command tells the router to send all traffic destined for the 172.16.2.0 network to the 172.16.3.2 interface. Static routes (or a dynamic routing protocol) must be defined in order for the router to know where to send non-local traffic. Additionally, the other end must have a route back to your network or networks. Dynamic routing will be covered later in this chapter.
dialer-list 1 protocol ip permit	This is the command that specifies the interesting traffic that can initiate dialing. In this example, the interesting traffic has been identified as all IP traffic. The next example shows how you can limit the interesting traffic to a specific set of protocols.
interface async 1	This command enters the sub-interface configuration mode for the asynchronous interface.
dialer in-band	This command enables DDR on the asynchronous interface. By default, only ISDN interfaces have this command automatically enabled.
ip address 172.16.3.1 255.255.255.0	This command configures the asynchronous interface with IP address 172.16.3.1.
dialer string 5551234	The dialer string command tells the router what phone number to dial. In this example, the remote site phone number is 555-1234.
dialer-group 1	The dialer-group command identifies what dialer list to use for interesting traffic on that interface. It is possible to have several dialer lists configured on the router and each interface can point to different dialer lists.
encapsulation ppp	This command tells the router to use PPP encapsulation on the interface.

Figure 5.13 (Example B) Asynchronous one-to-many.**Example B Router1 configuration.**

```
Router1(config)#ip route 172.16.2.0 255.255.255.0 172.16.5.2
Router1(config)#ip route 172.16.3.0 255.255.255.0 172.16.5.3
Router1(config)#ip route 172.16.4.0 255.255.255.0 172.16.5.4
Router1(config)#dialer-list 1 protocol ip list 101
Router1(config)#username Router2 password cisco
Router1(config)#username Router3 password cisco
Router1(config)#username Router4 password cisco
Router1(config)#interface async 1
Router1(config-if)#dialer in-band
Router1(config-if)#ip address 172.16.3.1 255.255.255.0
Router1(config-if)#dialer map ip 172.16.5.2 name Router2 5551234
Router1(config-if)#dialer map ip 172.16.5.3 name Router3 5555678
Router1(config-if)#dialer map ip 172.16.5.4 name Router4 5559012
Router1(config-if)#dialer-group 1
Router1(config-if)#encapsulation ppp
Router1(config-if)#ppp authentication chap
```

```

Router1(config)#access-list 101 permit tcp any any eq www
Router1(config)#access-list 101 permit tcp any any eq smtp
Router1(config)#access-list 101 permit tcp any any eq pop3
Router1(config)#access-list 101 permit icmp any any

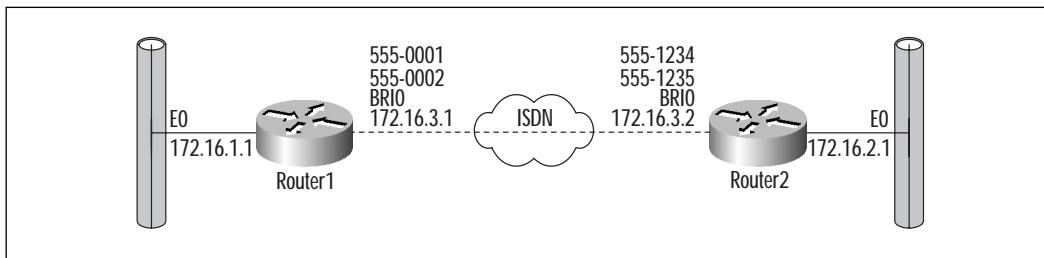
```

Table 5.2 Command Descriptions

Command	Description
dialer-list 1 protocol ip list 101	As in Example A, this command identifies what traffic will be considered interesting. This example identifies IP traffic, which passes the access list 101 as interesting traffic.
username Router2 password cisco	The username command is required for authentication. This command identifies the shared secret password required when challenged by the remote router.
dialer map ip 172.16.5.2 name Router2 5551234 dialer map ip 172.16.5.3 name Router3 5555678 dialer map ip 172.16.5.4 name Router4 5559012	The dialer map command maps an IP address to the remote router name to the phone number to be dialed. Along with IP route commands, all traffic destined for the 172.16.2.0 network will go through this dialer map. For the authentication to function, the name option must also be used.
ppp authentication chap	This command tells the router to use CHAP authentication on this interface. For CHAP authentication to pass, the remote routers must have this router in their username list and have CHAP authentication configured.
access-list 101 permit tcp any any eq www access-list 101 permit tcp any any eq smtp access-list 101 permit tcp any any eq pop3 access-list 101 permit icmp any any	access-list 101 permits all WWW, SMTP, POP3, and ICMP traffic. The explicit Deny All will deny all other types of IP traffic. With this access list and the dialer-list command, only WWW, SMTP, POP3, or ICMP traffic can initiate the DDR session.

Example C (Figure 5.14 and Router1 configuration) introduces ISDN connectivity. This example is very similar to Example A. Only the new commands are explained in Table 5.3. One difference between ISDN and analog telephone lines is that ISDN lines have two B-channels. When you obtain an ISDN line from the telephone company, they give you two phone numbers, one for each B-channel. With ISDN, you can configure your Cisco router to dial both of the B-channels and bond them together, giving you 128 Kbps of bandwidth. Example C explains how to accomplish this.

Figure 5.14 (Example C) ISDN BRI one-to-one.



Example C Router1 configuration.

```

Router1(config)#isdn switch-type basic-nil
Router1(config)#ip route 172.16.2.0 255.255.255.0 172.16.3.2
Router1(config)#dialer-list 1 protocol ip permit
Router1(config)#interface bri 0
Router1(config-if)#ip address 172.16.3.1 255.255.255.0
Router1(config-if)#isdn spid1 0913555000101
Router1(config-if)#isdn spid2 0913555000201
Router1(config-if)#dialer map ip 172.16.3.2 5551234
Router1(config-if)#bandwidth 128
Router1(config-if)#dialer load-threshold 127 either
Router1(config-if)#dialer-group 1
Router1(config-if)#encapsulation ppp
Router1(config-if)#ppp multilink

```

Table 5.3 Command Descriptions

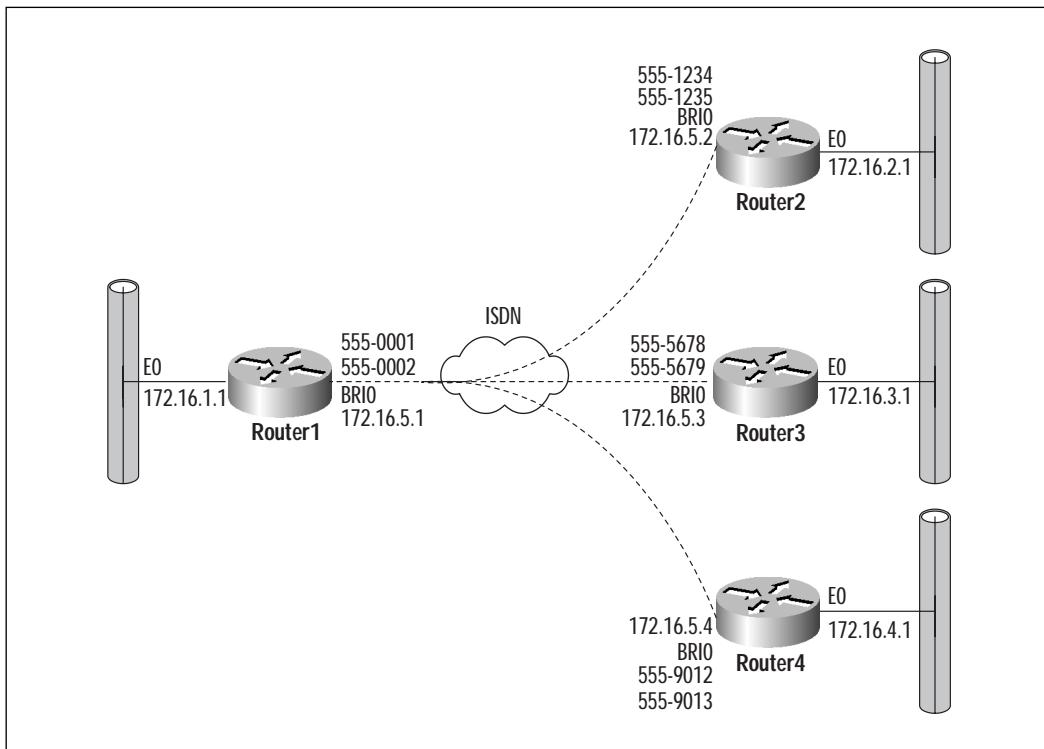
Command	Description
isdn switch-type basic-ni1	This command configures the ISDN switch type into the router. The telephone company should provide this information to you when installing an ISDN line.
isdn spid1 0913555000101 isdn spid2 0913555000201	This command configures your Service Profile Identifiers (SPIDs) into the router. The SPID is not required on all ISDN switch types. Your telephone company should provide SPIDs when installing an ISDN line.
bandwidth 128	This command tells the router how much bandwidth is available on the interface. The bandwidth command is used in calculating the load threshold.
dialer load-threshold 127 either	The dialer load-threshold command configures the router to initiate a second call once the threshold has been met. The value is a number between 1 and 255 and is a percent of the total bandwidth of the line. 127 is equivalent to approximately 50 percent or 64 Kbps, of data. Once traffic reaches this data rate, the second number is dialed (through the D-channel), connecting both B-channels. In this example, only one dialer map statement had to be issued for the threshold to operate correctly. Certain ISDN switches automatically recognize when a second call is incoming and re-route the call to the second B-channel. If the switch in this example did not support this, there would have been a second dialer map statement pointing the same IP address to the second B-channel number.
ppp multilink	This command bonds both B-channels together to provide for double the bandwidth of a B-channel.

WARNING

If you need to change the ISDN switch type on a Cisco router, the change will not take place until you reboot the router.

Example D (Figure 5.15 and Router1 configuration) shows how to configure an ISDN connection to dial into multiple sites. Example B identified the **dialer idle-timeout** command to allow for faster disconnection of DDR lines. Example D explains that command. Table 5.4 explains the benefit of the **dialer idle-timeout 5 either** command.

Figure 5.15 (Example D) ISDN BRI one-to-many.



Example D Router1 configuration.

```
Router1(config)#isdn switch-type basic-nii
Router1(config)#ip route 172.16.2.0 255.255.255.0 172.16.5.2
```

```

Router1(config)#ip route 172.16.3.0 255.255.255.0 172.16.5.3
Router1(config)#ip route 172.16.4.0 255.255.255.0 172.16.5.4
Router1(config)#dialer-list 1 protocol ip permit
Router1(config)#username Router2 password cisco
Router1(config)#username Router3 password cisco
Router1(config)#username Router4 password cisco
Router1(config)#interface bri 0
Router1(config-if)#ip address 172.16.5.1 255.255.255.0
Router1(config-if)#isdn spid1 0913555000101
Router1(config-if)#isdn spid2 0913555000201
Router1(config-if)#dialer map ip 172.16.5.2 name Router2 5551234
Router1(config-if)#dialer map ip 172.16.5.3 name Router3 5555678
Router1(config-if)#dialer map ip 172.16.5.4 name Router4 5559012
Router1(config-if)#bandwidth 128
Router1(config-if)#dialer load-threshold 127 either
Router1(config-if)#dialer-group 1
Router1(config-if)#encapsulation ppp
Router1(config-if)#ppp multilink
Router1(config-if)#dialer idle-timeout 5 either

```

Table 5.4 Command Descriptions

Command	Description
dialer idle-timeout 5 either	This command configures the router to disconnect the ISDN interface after 5 seconds of inactivity in either direction. Configuring this command can improve online usage.

ISDN and DDR commands

The following section covers the various ISDN and DDR commands covered in the previous examples. This is a list of some of the commands and their associated optional parameters.

1. **dialer-list *dialer-list-number* protocol *protocol operator***
 - The dialer-list command is used to define interesting traffic.
 - **dialer-list-number** – A number between 1 and 10.

- protocol – Can be any of the following (depending on IOS):
appletalk, bridge, clns, decnet, ip, ipx, llc2, netbios, vines, xns.
 - operator – Can be either permit, deny, or list with list number.
2. dialer map *protocol next-hop-address [name hostname] [speed speed] [modem-script script_name] [system-script script_name] [spcl] [class map_class] [broadcast] dial-string*
 - The dialer map command is used to map a protocol and next hop address to a phone number. This command is useful when dialing to more than one location.
 - protocol next-hop-address – Specifies the protocol and next hop router address.
 - name hostname – Specifies the destination router's host name.
 - speed speed – Specifies either 56K or 64K bits per second.
 - modem-script script_name – Specifies a modem chat script to be used for making the connection.
 - system-script script_name – Specifies a system chat script to be used for system login to the destination host.
 - spc – Specifies whether the connection is semi-permanent.
 - class map_class – Specifies a map class for the map.
 - broadcast – Specifies whether broadcast packets for the given protocol should be sent to the next hop address.
 - dial-string – Specifies the telephone number to be used for dialing out when a packet destined for the next hop address arrives.
 3. dialer in-band – Enables the interface for DDR operation. Sets the interface for V.25bis dialing.
 4. dialer string *phone_number* – Specifies the telephone number to be dialed.
 5. dialer-group *dialer-list-number* – Assigns the interface to the specified dialer list.
 - dialer-list-number – Value from 1 to 10.
 6. encapsulation *type* – Sets the encapsulation type for the interface. See the “Encapsulation” section earlier in the chapter for an explanation of types.

7. dialer idle-timeout *time* [either] – Specifies the amount of traffic inactivity time on the interface before disconnecting it.
 - *time* – A value between 1 and 2147483 seconds. The default is 120 seconds.
 - *either* – Tells the interface to monitor inbound and outbound traffic inactivity.
8. dialer hold-queue size [*timeout seconds*] – Specifies the output hold queue on the DDR interface. This command tells the router to hold a specified number of packets while the interface is being connected and transmitted once the session is established.
 - *size* – Number of packets from 0 to 100 to be held before dropping.
 - *timeout seconds* – The length of time the packets will be held before being dropped.
9. dialer load-threshold *percent-load* [*direction*] – This command identifies when to place an additional call based on the percent of bandwidth used on the interface. When an ISDN call is initiated, only 1 B-channel is dialed. When configuring this command, you can tell the router how soon to dial the second B-channel.
 - *percent-load* – A value from 1 to 255. A value of 127 would be 49.8% of the line, or 63.75 Kbps.
 - *direction* – Determines what direction of traffic flow is monitored before activating the additional line. This optional parameter can be set to inbound, outbound, or either.
10. isdn switch-type *type* – Sets the type of ISDN switch connected to your router.
 - *switch-type* – Several different types of ISDN switches are supported including:

a) basic-1tr6	1TR6 switch type for Germany
b) basic-5ess	AT&T 5ESS switch type for the U.S.
c) basic-dms100	Northern DMS-100 switch type
d) basic-net3	NET3 switch type for UK and Europe
e) basic-ni1	National ISDN-1 switch type
f) basic-nwnet3	NET3 switch type for Norway
g) basic-nznet3	NET3 switch type for New Zealand
h) basic-ts013	TS013 switch type for Australia
i) ntt	NTT switch type for Japan

- j) vn2 VN2 switch type for France
- k) vn3 VN3 and VN4 switch types for France

11. `isdn spid1 spid phone_number` – This command sets the Service Profile Identifier (SPID) for the BRI interface. The phone company provides the SPID, which is usually the phone number with a few numbers added to the front or back or both. 0913555123401 is an example of a SPID.

Caller ID Screening

One of the features supported with ISDN is caller ID. With caller ID, you can have your router accept calls only from specific numbers. This is referred to as caller ID screening. Caller ID screening is configured by using the **isdn caller** command. You can also configure a wildcard digit or digits when configuring the numbers by replacing the digit with an x. Each interface can be configured to screen up to 64 different numbers. Example F in the “Walkthrough” section at the end of this chapter gives an example of how to configure caller ID screening.

In addition to caller ID screening, Cisco has implemented a feature called caller ID callback. Caller ID callback allows a router to receive a call from a client, hang up the line, and then call the originating caller back. This feature can be used to save money, and allows the central location to pay for expensive ISDN calls. An example of caller ID callback is also shown in Example F.

WARNING

In order for caller ID screening to work, the local switch must be capable of delivering the caller ID to the router. If you configure caller ID screening and the switch does not support caller ID, calls will not be accepted by the router.

Routing Issues with DDR

All of the previous examples used static entries for routing. Static routing is not always the best option; there are many different types of routing designs that can be implemented when dealing with DDR. Cisco has

developed several methods of overcoming the following problems of implementing a dynamic routing protocol across a DDR line.

Static and Default Routes

Static routing is the most simple of the DDR routing options. All of the examples in this chapter so far have used static routing. Configuring static routing for DDR is the same as configuring static routing for any other Cisco interface. The command **ip route destination-address subnet-mask next-hop-address** will configure a static route on the router. In order for static routing to function, the remote network must also have a route back to you. To configure a default route, use the command **ip default-network default-network-address**.

The “gateway of last resort” is the route to use if there are no specific routes to a specified network. When configuring a single-homed connection to the Internet, gateway-of-last-resort routes are typically used. To configure the gateway of last resort, use the **ip route 0.0.0.0 0.0.0.0 next-hop-address** command.

Snapshot Routing

Static routing works well on small networks and in areas where a DDR link is the end of a routed network (Stub network). If you have a medium-sized network, maintaining the static routing table can be time-consuming and tedious. Snapshot routing is one method of overcoming the shortfalls of static routing.

Snapshot routing allows dynamic routing protocols to run across DDR links without requiring the line to remain connected. Snapshot routing works by having an active period when the link is active and routing information passed between neighboring routers, and then having a quiet period when the routing tables are frozen. The active period can be initiated by either user data triggering the DDR link, or by the quiet period timer expiring. Once in the active period, both routers exchange routing information, updating their routing tables. After the active period, the link is terminated, and the routers enter the quiet period and freeze their routing tables. Once the quiet period begins, a timer starts counting down to zero. As soon as the timer hits zero, the routers enter the active state and initiate a DDR connection.

Both the active and quiet periods are user-configurable values. Snapshot routing supports all periodic update routing protocols:

- Internet Protocol–Routing Information Protocol (IP–RIP) and Interior Gateway Routing Protocol (IGRP)

- Internetwork Packet Exchange–RIP (IPX–RIP) and Service Advertising Protocol (SAP)
- Appletalk–Routing Table Maintenance Protocol (RTMP)
- Vines–Routing Table Protocol (RTP)

Snapshot routing does not support link state routing protocols because of the way that they exchange routing information. Link state protocols—Intermediate System to Intermediate System Protocol (IS-IS), Open Shortest Path First (OSPF), Netware Link Service Protocol (NLSP), and Cisco's Enhanced IGRP (EIGRP)—exchange information between neighboring routers every 5 to 10 seconds. This update period would essentially require the link to remain active indefinitely for the routing protocol to function properly.

TIP

Snapshot routing has been designed to work for hub-and-spoke topologies. If you have a fully or partially meshed topology, static routing or OSPF on-demand routing would be a better choice of routing design.

To configure snapshot routing, configure the routing protocol and DDR interface as normal. Additionally, use the **snapshot server active-time [dialer]** command on the interface of the router receiving the call, and the **snapshot client active-time quiet-time [suppress-statechange-updates] [dialer]** command on the interface of the dialing router. The active time parameter is a value from 5 to 100 minutes, and the quiet time value is from 8 to 100,000 minutes; the dialer optional parameter allows the router to dial if not already connected, and the optional parameter **suppress-statechange-updates** allows the router to exchange routing updates if the connection is established through interesting traffic. The **suppress-state-change-updates optional** command is on by default when configuring snapshot routing. For the dialer parameter to function, you need to configure a dialer map for snapshot routing. An example of snapshot routing is provided at the end of this chapter.

OSPF On-demand Circuits

As mentioned in the previous section, snapshot routing does not support OSPF. Cisco developed support for RFC 1793 “Extending OSPF to Support Demand Circuits” to overcome the lack of link state routing support across DDR networks. OSPF on-demand works by initially bringing up the DDR

line when the routers exchange LSA information for the first time, and when a change occurs during normal operation. As long as the network topology is stable, the circuit does not need to be connected.

Configuring OSPF on-demand circuits is fairly simple. In addition to the normal OSPF and DDR configuration, use the **ip ospf demand-circuit** command in the interface configuration mode. In order for this feature to work, all routers in the area must have it loaded. Additionally, only one of the routers needs to configure this command. If using a point-to-point topology, either end can be configured with this command. If using a point-to-multipoint topology, the hub (or multipoint end) must be configured with this command. Example F in the “Walkthrough” section shows an example OSPF on-demand configuration.

TIP

It is recommended that you put OSPF on-demand circuits into stub areas or Not So Stubby Areas (NSSAs) to isolate as many of the topology changes as possible.

Route Redistribution

When configuring DDR networks, it is important to remember to redistribute the remote networks into the rest of your network. Whichever way the DDR network is configured, it is recommended you redistribute the static, OSPF on-demand, or snapshot networks into the rest of your network. To do this, use the **redistribute routing-protocol** command within the primary network routing protocol process.

Monitoring and Troubleshooting ISDN and DDR

The following section covers some of the various show and debug commands for ISDN and DDR. The screenshots used in these examples are taken from the two examples in the following “Walkthrough” section.

Monitoring the ISDN Interface

The command **show interface bri 0** (Figure 5.16) displays information about the BRI interface. It gives you information about the D-channel of

the interface. This command is only valid on routers with internal BRI interfaces. If you are not using an internal BRI interface, then you would issue the command **show interface serial** to obtain similar information.

Figure 5.16 The *show interface bri 0* command.

```
Router1#show interface bri 0
BRI0 is up, line protocol is up (spoofing)

Hardware is BRI
Internet address is 172.16.3.1/30
MTU 1500 bytes, BW 128 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation PPP, loopback not set
Last input 00:00:01, output 00:00:01, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/64/0 (size/max total/threshold/drops)
  Conversations 0/1/256 (active/max active/max total)
  Reserved Conversations 0/0 (allocated/max allocated)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  4723 packets input, 25063 bytes, 0 no buffer
  Received 4 broadcasts, 0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  4957 packets output, 23463 bytes, 0 underruns
  0 output errors, 0 collisions, 7 interface resets
  0 output buffer failures, 0 output buffers swapped out
  5 carrier transitions
```

Looking at Figure 5.16, the second line shows that the interface is up and the protocol is up (spoofing). Spoofing is used to trick the router into believing the interface is permanently connected. This is done so that DDR will function properly. When an interface is down, any entries in the routing table pointing to that interface will be removed. DDR requires that routing table entries be intact in order to initiate dialing. DDR tells the BRI interface to remain in a spoofing state to maintain the routing entries for that interface or network. This command is primarily used to verify that the interface is responding and that the IP address has been configured

correctly. Also, when identifying problems, the input and output rates and errors are useful.

As you can see in Figure 5.17, the command **show interface bri 0 1 2** gives details of both B-channels of the BRI interface. You can quickly identify whether either or both of the B-channels are up or down, as well as determine the encapsulation protocol. Other useful data is the various input and output information.

Figure 5.17 The *show interface bri 0 1 2* command.

```
Router1#show interface bri 0 1 2
BRI0:1 is down, line protocol is down
  Hardware is BRI
    MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
    Encapsulation PPP, loopback not set, keepalive set (10 sec)
    LCP Closed, multilink Closed
    Closed: IPCP, CDPCP
    Last input 00:00:17, output 00:00:17, output hang never
    Last clearing of "show interface" counters never
    Queueing strategy: fifo
    Output queue 0/40, 0 drops; input queue 2/75, 0 drops
    5 minute input rate 0 bits/sec, 0 packets/sec
    5 minute output rate 0 bits/sec, 0 packets/sec
      6764 packets input, 273534 bytes, 0 no buffer
      Received 6764 broadcasts, 0 runts, 0 giants, 0 throttles
      0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
      6826 packets output, 283850 bytes, 0 underruns
      0 output errors, 0 collisions, 7 interface resets
      0 output buffer failures, 0 output buffers swapped out
      231 carrier transitions
BRI0:2 is down, line protocol is down
  Hardware is BRI
    MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
    Encapsulation PPP, loopback not set, keepalive set (10 sec)
    LCP Closed, multilink Closed
    Closed: IPCP, CDPCP
```

```
Last input 07:12:56, output 07:12:56, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    72 packets input, 2468 bytes, 0 no buffer
    Received 72 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    74 packets output, 2480 bytes, 0 underruns
    0 output errors, 0 collisions, 7 interface resets
    0 output buffer failures, 0 output buffers swapped out
    2 carrier transitions
```

A quick way to identify whether the BRI and B-channels are up is to use the **show ip interface brief** command. This command shows whether the interface is up, whether the protocol is up, and also shows the IP address of the interface. Notice in Figure 5.18 that the BRI0 interface is the only BRI interface that has an IP address assigned to it.

Figure 5.18 The *show ip interface brief* command.

```
Router2#show ip interface brief
Interface          IP-Address      OK? Method Status   Protocol
BRI0              172.16.3.2    YES  NVRAM   up      up
BRI0:1            unassigned     YES  unset    up      up
BRI0:2            unassigned     YES  unset    up      up
Ethernet0          172.16.2.1    YES  NVRAM   up      up
Virtual-Access1   unassigned     YES  unset    up      up
Virtual-Access2   unassigned     YES  unset    down    down
```

The **show isdn status** command gives information on all three layers of the ISDN interface. It identifies the ISDN switch type configured, and gives information on SPIDs and active calls. You can see information on all three ISDN layers in Figure 5.19.

Figure 5.19 The *show isdn status* command.

```
Router1#show isdn status
The current ISDN Switchtype = basic-nil
ISDN BRI0 interface
  Layer 1 Status:
    ACTIVE
  Layer 2 Status:
    TEI = 118, Ces = 1, SAPI = 0, State =
MULTIPLE_FRAME_ESTABLISHED
    TEI = 119, Ces = 2, SAPI = 0, State =
MULTIPLE_FRAME_ESTABLISHED
  Spid Status:
    TEI 118, ces = 1, state = 5(init)
      spid1 configured, no LDN, spid1 sent, spid1 valid
      Endpoint ID Info: epsf = 0, usid = 2, tid = 1
    TEI 119, ces = 2, state = 5(init)
      spid2 configured, no LDN, spid2 sent, spid2 valid
      Endpoint ID Info: epsf = 0, usid = 4, tid = 1
  Layer 3 Status:
    1 Active Layer 3 Call(s)
    Activated dsl 0 CCBs = 1
    CCB:callid=0x8076, sapi=0x0, ces=0x1, B-chan=1
    Total Allocated ISDN CCBs = 1
```

Monitoring the Dialer

The dialer is responsible for making and maintaining DDR connections. The command in Figure 5.20 can be used to verify proper dialing and connectivity.

Figure 5.20 The *show dialer* command.

```
Router1#show dialer
BRI0 - dialer type = ISDN

Dial String      Successes   Failures   Last called   Last status
```

```
8358661           235           1   00:01:53      successful
0 incoming call(s) have been screened.
0 incoming call(s) rejected for callback.
```

```
BRI0:1 - dialer type = ISDN
Idle timer (120 secs), Fast idle timer (20 secs)
Wait for carrier (30 secs), Re-enable (15 secs)
Dialer state is multilink member
Dial reason: snapshot
Connected to 8358661 (Router2)
```

```
BRI0:2 - dialer type = ISDN
Idle timer (120 secs), Fast idle timer (20 secs)
Wait for carrier (30 secs), Re-enable (15 secs)
Dialer state is idle
```

The **show dialer** command gives information on the phone number being dialed and the number of successful and failed calls to that number. It also gives specific information on the interface performing the dialing such as “Idle timer,” “Fast idle timer,” “Wait for carrier,” and “Re-enable.” The Idle timer shows how long the router waits to disconnect after not receiving traffic. The Fast idle timer is triggered if there is traffic destined for a different number. This timer will disconnect the circuit, allowing the data destined for the different network to be passed. In Figure 5.20, all of the timers are configured as default values.

The command **show dialer maps** displays all static dialer maps configured on that router and the interface where they are configured. In Figure 5.21, there are two dialer maps configured on the BRI0 interface.

Figure 5.21 The *show dialer maps* command.

```
Router1#show dialer maps
Static dialer map ip 172.16.3.2 name Router1 broadcast (8358661) on BRI0
Static dialer map snapshot 1 name Router2 broadcast (8358661) on BRI0
```

Monitoring PPP Multilink

PPP Multilink allows for multiple circuits to be bonded together to allow for greater bandwidth. The command in Figure 5.22 can be used to verify PPP multilink operation.

Figure 5.22 The *show ppp multilink* command.

```
Router1#show ppp multilink

Bundle Router2, 2 members, Master link is Virtual-Access2
Dialer Interface is BRI0
  0 lost fragments, 0 reordered, 0 unassigned, sequence 0xC/0xE rcvd/sent
  0 discarded, 0 lost received, 1/255 load

Member Links: 2 (max not set, min not set)
BRI0:2
BRI0:1
```

The **show ppp multilink** command gives information on the status of the multilink session. It identifies the remote router and the interface connecting to it. In Figure 5.22, both B-channels are in the same multilink bundle.

The command **show interface bri 0 1 2** not only gives information about a BRI interface, it also gives information on PPP multilink. If you look at the fifth line of the output in Figure 5.23, it identifies that multilink is open, which means the PPP multilink session has been established.

Figure 5.23 The *show interface bri 0 1 2* command.

```
Router1#show interface bri 0 1 2

BRI0:1 is up, line protocol is up
  Hardware is BRI
  MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation PPP, loopback not set, keepalive set (10 sec)
  LCP Open, multilink Open
  Last input 00:00:02, output 00:00:02, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
```

```
Output queue 0/40, 0 drops; input queue 2/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    6825 packets input, 276786 bytes, 0 no buffer
    Received 6825 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    6888 packets output, 287236 bytes, 0 underruns
    0 output errors, 0 collisions, 7 interface resets
    0 output buffer failures, 0 output buffers swapped out
    234 carrier transitions

BRI0:2 is up, line protocol is up

Hardware is BRI

MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation PPP, loopback not set, keepalive set (10 sec)
LCP Open, multilink Open
Last input 00:00:07, output 00:00:07, output hang never
Last clearing of "show interface" counters never
Queueing strategy: fifo

Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    87 packets input, 3084 bytes, 0 no buffer
    Received 87 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    90 packets output, 3240 bytes, 0 underruns
    0 output errors, 0 collisions, 7 interface resets
    0 output buffer failures, 0 output buffers swapped out
    3 carrier transitions
```

Monitoring Snapshot Routing

The following commands can be useful in monitoring snapshot routing features. Both commands are the same; Figure 5.24 is taken from the snapshot server and Figure 5.25 is taken from the snapshot client.

When issued from the server, the **show snapshot** command (Figure 5.24) shows how long the server has been configured for the active period. It also shows the interface being used for the snapshot routing.

When issued from the client router, the **show snapshot** command gives more information about the snapshot session, such as the amount of active time and quiet time, as well as the interfaces being used for the snapshot session. In Figure 5.25, the active time has been set to 10 minutes and the quiet time is set to 3 minutes ($13 \text{ minutes} - 10 \text{ minutes} = 3 \text{ minutes}$).

Figure 5.24 The *show snapshot* command.

```
Router2#show snapshot
BRI0 is up, line protocol is upSnapshot server
Options: dialer support
Length of active period: 10 minutes
```

Figure 5.25 The *show snapshot* command.

```
Router1#show snapshot
BRI0 is up, line protocol is upSnapshot client
Options: dialer support
Length of active period: 10 minutes
Length of quiet period: 13 minutes
Length of retry period: 13 minutes
For dialer address 1
Current state: active, remaining/exchange time: 9/0 minutes
Connected dialer interfaces:
BRI0:1, BRI0:2
```

Troubleshooting ISDN and DDR

The following debug commands allow you to troubleshoot any problems you encounter with DDR and ISDN interfaces. The following examples display only a few lines of the debug results. To obtain a better understanding of these debug commands, it is recommended you perform them in a laboratory environment.

The command **debug isdn q921** displays all information that passes between the local router and the ISDN switch. Figure 5.26 gives an example of this command.

Figure 5.26 The *debug isdn q921* command.

```
Router1#debug isdn q921
ISDN Q921 packets debugging is on
02:47:01: %LINK-3-UPDOWN: Interface BRI0:1, changed state to down
02:47:01: %LINK-3-UPDOWN: Interface BRI0:2, changed state to down
02:47:02: %LINK-3-UPDOWN: Interface BRI0, changed state to up
02:47:02: ISDN BR0: TX -> SABMEp sapi = 0 tei = 79
02:47:02: ISDN BR0: RX <- IDREM ri = 0 ai = 127
02:47:02: ISDN BR0: RX <- IDCKRQ ri = 0 ai = 79
02:47:02: %ISDN-6-LAYER2DOWN: Layer 2 for Interface BRI0, TEI 80
changed to down
02:47:02: %ISDN-6-LAYER2DOWN: Layer 2 for Interface BRI0, TEI 79
changed to down
02:47:02: %ISDN-6-LAYER2DOWN: Layer 2 for Interface BR0, TEI 79 changed
to down
02:47:02: %SYS-5-CONFIG_I: Configured from console by console
02:47:02: ISDN BR0: TX -> IDREQ ri = 44940 ai = 127
02:47:03: ISDN BR0: RX <- IDCKRQ ri = 0 ai = 79
02:47:04: ISDN BR0: RX <- IDREM ri = 0 ai = 79
02:47:04: ISDN BR0: TX -> IDREQ ri = 43085 ai = 127
02:47:05: ISDN BR0: RX <- IDASSN ri = 43085 ai = 81
02:47:05: ISDN BR0: TX -> SABMEp sapi = 0 tei = 81
02:47:05: ISDN BR0: RX <- UAf sapi = 0 tei = 81
02:47:05: %ISDN-6-LAYER2UP: Layer 2 for Interface BR0, TEI 81 changed
to up
02:47:05: ISDN BR0: TX -> INFOC sapi = 0 tei = 81 ns = 0 nr = 0 i
= 0x08007B3A0A303
02:47:05: ISDN BR0: RX <- INFOC sapi = 0 tei = 81 ns = 0 nr = 1 i
= 0x08007B3B02828
02:47:05: ISDN BR0: TX -> INFOC sapi = 0 tei = 81 ns = 1 nr = 1 i
= 0x08012705040288
02:47:05: ISDN BR0: TX -> IDREQ ri = 11550 ai = 127
02:47:05: ISDN BR0: RX <- INFOC sapi = 0 tei = 81 ns = 1 nr = 2 i
= 0x0801A702180189
02:47:05: ISDN BR0: RX <- IDASSN ri = 11550 ai = 82
```

```

02:47:05: ISDN BR0: TX -> RRr sapi = 0 tei = 81 nr = 2
02:47:05: ISDN BR0: TX -> SABMEp sapi = 0 tei = 82
02:47:05: ISDN BR0: RX <- UAf sapi = 0 tei = 82
02:47:05: %ISDN-6-LAYER2UP: Layer 2 for Interface BR0, TEI 82 changed
to up
02:47:05: ISDN BR0: TX -> INFOc sapi = 0 tei = 82 ns = 0 nr = 0 i
= 0x08007B3A0A3038
02:47:05: ISDN BR0: RX <- INFOc sapi = 0 tei = 81 ns = 2 nr = 2 i
= 0x0801A707
02:47:05: ISDN BR0: TX -> RRr sapi = 0 tei = 81 nr = 3
02:47:05: %LINK-3-UPDOWN: Interface BRI0:1, changed state to up
02:47:05: ISDN BR0: TX -> INFOc sapi = 0 tei = 81 ns = 2 nr = 3 i
= 0x0801270F
02:47:05: ISDN BR0: RX <- INFOc sapi = 0 tei = 82 ns = 0 nr = 1 i
= 0x08007B3B028481
02:47:05: ISDN BR0: TX -> RRr sapi = 0 tei = 82 nr = 1
02:47:05: ISDN BR0: RX <- RRr sapi = 0 tei = 81 nr = 3
02:47:05: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to
up
02:47:06: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0:1,
changed state to up
02:47:06: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-
Access1, changed state to up

```

Q921 information is a Layer 2 protocol. If you need to verify Layer 3 connectivity, use the **debug isdn q931** command. This command, as shown in Figure 5.27, displays all call setup and teardown information across the D-channel. Both Q921 and Q931 display information on the D-channel. If you need to obtain information on the B-channel you should use either the **debug dialer** or **debug ip packet** command.

Figure 5.27 The *debug isdn q931* command.

```

Router1#debug isdn q931
ISDN Q931 packets debugging is on
02:50:03: ISDN BR0: TX -> INFORMATION pd = 8 callref = (null)
    SPID Information i = '0835866201'
02:50:03: ISDN BR0: RX <- INFORMATION pd = 8 callref = (null)

```

```

ENDPOINT IDent i = 0x8281
02:50:03: ISDN BR0: TX -> SETUP pd = 8 callref = 0x28
02:50:03:             Bearer Capability i = 0x8890
02:50:03:             Channel ID i = 0x83
02:50:03:             Keypad Facility i = '8358661'
02:50:03: ISDN BR0: RX <- CALL_PROC pd = 8 callref = 0xA8
02:50:03:             Channel ID i = 0x89
02:50:03:             Locking Shift to Codeset 5
02:50:03:             Codeset 5 IE 0x2A i = 0x809402, '=' , 0x8307,
'8358661', 0x8E0B, ' Telton 1 '
02:50:03: %ISDN-6-LAYER2UP: Layer 2 for Interface BR0, TEI 84 changed
to up
02:50:03: ISDN BR0: TX -> INFORMATION pd = 8 callref = (null)
              SPID Information i = '0835866401'
02:50:03: ISDN BR0: RX <- CONNECT pd = 8 callref = 0xA8
02:50:03: %LINK-3-UPDOWN: Interface BRI0:1, changed state to up
02:50:03: ISDN BR0: TX -> CONNECT_ACK pd = 8 callref = 0x28
02:50:03: ISDN BR0: RX <- INFORMATION pd = 8 callref = (null)
              ENDPOINT IDent i = 0x8481
02:50:03: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to
up
02:50:03: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0:1,
changed state to up
02:50:03: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-
Access1, changed state to up

```

Figure 5.28 shows the **debug dialer command.** This command is useful for identifying DDR events such as the dialing cause and phone number being dialed.

Figure 5.28 The *debug dialer* command.

```

Router1#debug dialer
Dial on demand events debugging is on
02:55:27: BRI0: Dialing cause ip (s=172.16.3.1, d=172.16.3.2)
02:55:27: BRI0: Attempting to dial 8358661
02:55:27: %ISDN-6-LAYER2UP: Layer 2 for Interface BR0, TEI 87 changed
to up

```

```
02:55:27: %ISDN-6-LAYER2UP: Layer 2 for Interface BR0, TEI 88 changed  
to up  
02:55:27: %LINK-3-UPDOWN: Interface BRI0:1, changed state to up  
02:55:27: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to  
up  
02:55:27: dialer Protocol up for Vil  
02:55:28: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0:1,  
changed state to up
```

If you are using snapshot routing, you can use the **debug snapshot** command to verify its operation. Figure 5.29 shows the transition from quiet to active time, which causes the interface to dial and establish a connection.

Figure 5.29 The *debut snapshot* command.

```
Router1#debug snapshot  
Snapshot support debugging is on  
03:20:02: SNAPSHOT: BRI0[1]: Move to active queue (Post active timeout)  
03:20:02: SNAPSHOT: BRI0[1]: moving to active queue  
03:20:03: %ISDN-6-LAYER2UP: Layer 2 for Interface BR0, TEI 89 changed  
to up  
03:20:03: %ISDN-6-LAYER2UP: Layer 2 for Interface BR0, TEI 90 changed  
to up  
03:20:03: %LINK-3-UPDOWN: Interface BRI0:1, changed state to up  
03:20:03: SNAPSHOT: BRI0[1]: Avoiding active: in active queue (Dial  
connection set)  
03:20:03: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to  
up  
03:20:04: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0:1,  
changed state to up  
03:20:04: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-  
Access1, changed state to up  
03:20:09: %ISDN-6-CONNECT: Interface BRI0:1 is now connected to 8358661  
Router2
```

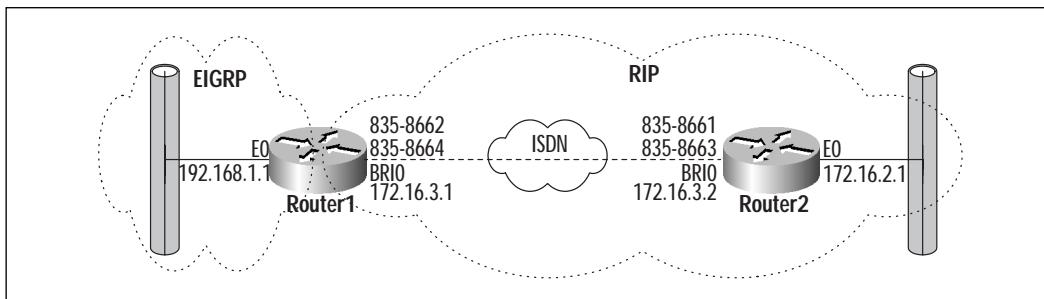
If you are using PPP multilink, you can use the **debug ppp multilink events** command to verify its operation. Figure 5.30 shows that multilink is being used and uses both B-channels on the BRI0 interface.

Figure 5.30 The *debug ppp multilink events* command.

```
Router1#debug ppp multilink events
Multilink events debugging is on
03:28:34: %ISDN-6-LAYER2UP: Layer 2 for Interface BR0, TEI 95 changed
to up
03:28:34: %ISDN-6-LAYER2UP: Layer 2 for Interface BR0, TEI 96 changed
to up
03:28:34: %LINK-3-UPDOWN: Interface BRI0:1, changed state to up
03:28:34: BR0:1 MLP: Multilink up event pending
03:28:34: Vil MLP: Added to huntgroup BR0
03:28:34: Vil MLP: Clone from BR0
03:28:34: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to
up
03:28:35: BR0:1 MLP: Router2, multilink up, first link
03:28:35: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0:1,
changed state to up
03:28:35: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-
Access1, changed state to up
03:28:37: %LINK-3-UPDOWN: Interface BRI0:2, changed state to up
03:28:37: BR0:2 MLP: Multilink up event pending
03:28:37: BR0:2 MLP: Router2, multilink up
03:28:38: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0:2,
changed state to up
```

Walkthrough

Previous examples showed the basics of connecting a BRI to a BRI. The following examples show how to configure some of the more advanced DDR features covered in this chapter. Example E (Figure 5.31, Router1 configuration, and Router2 configuration) shows a BRI-to-BRI configuration using snapshot routing and route redistribution. In this example, both router configurations are displayed. Example F shows how to configure a BRI to a PRI using OSPF on-demand routing, caller ID callback, and caller ID screening.

Figure 5.31 (Example E) Snapshot routing with route redistribution.**Example E Router1 configuration.**

```

hostname Router1
isdn switch-type basic-nil
dialer-list 1 protocol ip permit
!
interface Ethernet0
  ip address 192.1681.1 255.255.255.0
!
interface BRI0
  ip address 172.16.3.1 255.255.255.252
  encapsulation ppp
  bandwidth 128
  dialer map ip 172.16.3.2 name Router2 broadcast 8358661
  dialer map snapshot 1 name Router2 broadcast 8358661
  dialer load-threshold 127 either
  dialer-group 1
  isdn spid1 0835866201
  isdn spid2 0835866401
  snapshot client 10 13 dialer
  ppp multilink
!
router eigrp 6243
  redistribute rip metric 64 10 255 127 1500
  network 192.168.1.0

```

```
!  
router rip  
version 2  
redistribute eigrp 6243 metric 2  
network 172.16.0.0  
neighbor 172.16.3.2
```

Example E Router2 configuration.

```
hostname Router2  
isdn switch-type basic-nil  
dialer-list 1 protocol ip permit  
!  
interface Ethernet0  
ip address 172.16.2.1 255.255.255.0  
!  
interface BRI0  
ip address 172.16.3.2 255.255.255.252  
encapsulation ppp  
bandwidth 128  
dialer map ip 172.16.3.1 name Router1 broadcast 8358662  
dialer map snapshot 1 name Router1 broadcast 8358662  
dialer load-threshold 127 either  
dialer-group 1  
isdn spid1 0835866101  
isdn spid2 0835866301  
snapshot server 10 dialer  
ppp multilink  
!  
router rip  
version 2  
network 172.16.0.0  
neighbor 172.16.3.1  
no auto-summary
```

Example E shows how to configure snapshot routing as well as route redistribution. To configure snapshot routing, you simply need to configure one router as the client and the other as the server. The snapshot client is the end that controls the active and quiet timers. You will notice in the Router1 configuration that the dialer parameter has also been used. In the event the dialer parameter is used, a dialer map must be made between the snapshot process and the phone number. This will allow the snapshot update to initiate a DDR session if there is no interesting traffic to bring up the link.

Looking at the Router1 configuration, EIGRP is redistributing routes learned from RIP, and RIP is redistributing routes learned from EIGRP. This is commonly referred to as *mutual redistribution*.

Figures 5.32 and 5.33 show the routing table for Router1 before and after the DDR connection is established. In Figure 5.32, notice that before the connection is established, the only routes in the routing table are the ones directly connected to the router (192.168.1.0, 172.16.3.2, and 172.16.3.0). After the connection is established (Figure 5.33), the routing table also shows the 172.16.2.0 network and that it was learned via RIP. Once the ISDN connection to Router2 is disconnected, the route to 172.16.2.0 stays in the routing table for the quiet period configured in the snapshot command.

Figure 5.32 Router1 routing table before DDR connection.

```
Router1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B -
BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default
U - per-user static route, o - ODR

Gateway of last resort is not set
```

172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks

```
C        172.16.3.2/32 is directly connected, BRI0
C        172.16.3.0/30 is directly connected, BRI0
C        192.168.1.0/24 is directly connected, Ethernet0
```

Figure 5.33 Router1 routing table after DDR connection.

```

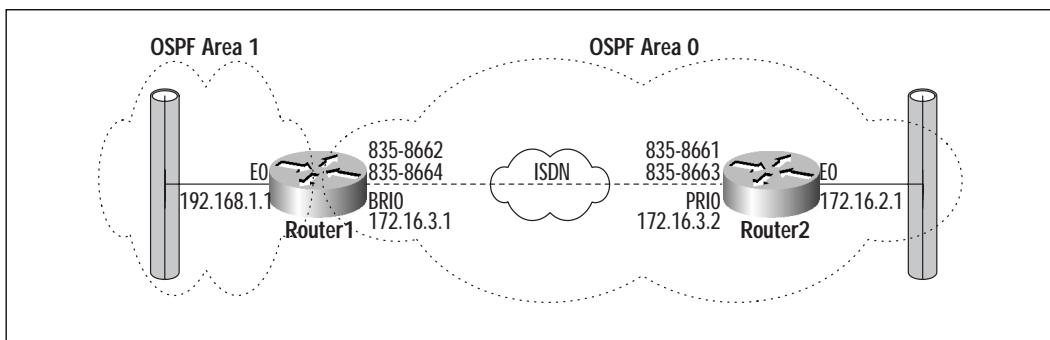
Router1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B -
BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default
U - per-user static route, o - ODR

Gateway of last resort is not set

172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
R        172.16.2.0/24 [120/1] via 172.16.3.2, 00:00:13, BRI0
C        172.16.3.0/30 is directly connected, BRI0
C        192.168.1.0/24 is directly connected, Ethernet0

```

Example F (Figure 5.34, Router1 configuration, and Router2 configuration) shows a BRI-to-PRI connection. In this example, the PRI is also performing caller ID authentication. Both routers are running OSPF on-demand routing across the ISDN link. Following the configurations for Router1 and Router2 is an output of Router1's routing table before and after the ISDN connection is established.

Figure 5.34 (Example F) PRI OSPF on-demand with caller ID screening.

Example F Router1 configuration.

```
hostname Router1
isdn switch-type basic-nil
dialer-list 1 protocol ip permit
!
interface Ethernet0
 ip address 192.168.1.1 255.255.255.0
!
interface BRI0
 ip address 172.16.3.1 255.255.255.252
 encapsulation ppp
 bandwidth 128
 ip ospf demand-circuit
 dialer map ip 172.16.3.2 name Router1 broadcast 8358661
 dialer load-threshold 127 either
 dialer-group 1
 isdn spid1 0835866201
 isdn spid2 0835866401
 ppp multilink
!
router ospf 2177
 network 172.16.3.0 0.0.0.255 area 0
 network 192.168.1.0 0.0.0.255 area 1
```

Example F Router2 configuration.

```
hostname Router2
isdn switch-type primary-5ess
dialer-list 1 protocol ip permit
ip local pool dialup 172.16.3.129 172.16.3.152
!
controller T1 0
 framing esf
 clock source line primary
```

```
linecode b8zs
pri-group timeslots 1-24
!
interface Ethernet0
 ip address 172.16.2.1 255.255.255.0
!
interface Serial0:23
 no ip address
 encapsulation ppp
 dialer rotary-group 1
 dialer-group 1
 isdn switch-type primary-5ess
 isdn incoming-voice modem
 no fair-queue
!
interface Dialer1
 ip address 172.16.3.2 255.255.255.0
 encapsulation ppp
 dialer in-band
 ip ospf demand-circuit
 dialer map ip 172.16.3.1 name Router1 broadcast
 dialer-group 1
 peer default ip address pool dialup
 isdn caller 8358662 callback
 isdn caller 8358664 callback
 ppp multilink
!
router ospf 2177
 network 172.16.2.0 0.0.0.255 area 0
 network 172.16.3.0 0.0.0.255 area 0
```

Caller ID authentication with callback is simple to configure. As seen in the configuration of Router2, it requires only two commands: **isdn caller 8358662 callback**, and **isdn caller 8358664 callback**. These commands will only allow an incoming call to be connected if its number is 835-8662 or 835-8664. The callback parameter instructs Router2 to hang up and call Router1 back.

TIP

Remember that you can replace any number or numbers with an “x” to mean “I don’t care.” If, in this example, the **isdn caller 835866x** command were used, the following incoming numbers would be allowed:

8358660	8358661	8358662	8358663	8358664
8358665	8358666	8358667	8358668	8358669

Both Router1 and Router2 are running OSPF and have configured their ISDN interfaces for OSPF on-demand operation. It only takes one **ip ospf demand-circuit** command (in addition to the normal OSPF configuration) to configure the routers for OSPF on-demand. Additionally, since Router2 has a PRI interface, the controller and the serial0:23 interface must be configured. To configure the controller, you need to know the type of controller (T1 or E1), framing, linecode, and number of channels to be used, as well as where the clock source is. The serial0:23 interface is the D-channel on a T1 PRI. Notice that the Dialer1 interface is being used for this example. Dialer interfaces are covered in detail in the next chapter.

Figures 5.35 and 5.36 show the route table for Router1 before and after the ISDN connection is established.

Figure 5.35 Router1 routing table before DDR connection.

```
Router1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B -
BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default
U - per-user static route, o - ODR

Gateway of last resort is not set

172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
C      172.16.3.2/32 is directly connected, BRI0
```

```
C      172.16.3.0/30 is directly connected, BRI0
C      192.168.1.0/24 is directly connected, Ethernet0
```

Figure 5.36 Router1 routing table after DDR connection.

```
Router1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B -
BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default
U - per-user static route, o - ODR

Gateway of last resort is not set

172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
O      172.16.2.0/24 [110/791] via 172.16.3.2, 00:00:18, BRI0
C      172.16.3.0/30 is directly connected, BRI0
C      192.168.1.0/24 is directly connected, Ethernet0
```

Before the connection is first brought up, Router1 has no route to Router2's Ethernet network (172.16.2.0). Notice that after the connection is made (see Figure 5.36), there is now an OSPF route to network 172.16.2.0 through the BRI 0 interface. Even after the ISDN line is disconnected, the route to 172.16.2.0 remains in the routing table for Router1.

Summary

ISDN was developed to overcome problems with the PSTN analog network. The CCITT, which was later replaced by the ITU-T, developed the standards for ISDN. The standards are split into three categories: the E series, which deal with telephone standards for ISDN; the I series, which deal with concepts and terminology of ISDN; and the Q series, which deal with call setup and switching processes. ISDN is composed of a group of channels including the B-channel (64 Kbps), D-channel (16 Kbps or 64 Kbps), and H-channel (384 Kbps up to 1.92 Mbps). The B-channel and D-channel are the most commonly used. Combining the channels into bundles gives two different interfaces. BRI and PRI are the two most common bundled ISDN

interfaces. The BRI is composed of two B-channels for data and one D-channel for signaling, with a combined bandwidth of 144 Kbps. The PRI is provisioned differently in different parts of the world. In the United States and Japan, the PRI is composed of 23 B-channels and one D-channel with a total bandwidth of 1.544 Mbps. In Europe and Australia, the PRI is composed of 30 B-channels and one D-channel, which provide 2.048 Mbps of bandwidth. There are several different functional groups and reference points for both BRI and PRI interfaces that identify architectural separations at the customer's premises.

The ISDN protocol layers can be mapped to the lower three layers of the OSI model and then further split into the user plane (U-plane), control plane (C-plane), and management plane (M-plane). There are several protocols that operate at each layer within each plane, such as Q.931, which controls call setup and teardown.

DDR allows routers to dynamically open and close a circuit-switched session for data connectivity. The key to DDR functioning is the definition of interesting traffic. Interesting traffic is traffic that has been identified by the router administrator and can be an entire protocol such as IP or can be linked to specific access lists. When using DDR, several different topologies can be used to build the network; point-to-point, fully meshed, and hub-and-spoke topologies are the most common. DDR can operate over ISDN, synchronous serial, and asynchronous interfaces.

One common problem encountered with ISDN is how to route over large, complex networks. Using static routes works for small networks that do not change often; however, if the network has many routes or changes frequently (more than once a month), a dynamic routing protocol is preferred. There are two methods for allowing dynamic routing protocols across DDR interfaces: snapshot routing and OSPF on-demand routing. Snapshot routing works with RIP and IGRP, and OSPF on-demand works with OSPF. Both methods operate by exchanging routing updates when the link is active. When the link is disconnected the routing tables remain unchanged. The link can be configured to connect at certain time intervals to refresh routing information, or connect when interesting traffic establishes the link.

Throughout this chapter there have been configuration examples along with the IOS commands used. These examples cover basic scenarios and give a good basis for understanding the various ISDN and DDR functionality. See Chapter 6 for more detail on DDR functionality.

FAQs

Q: How can I tell if my Cisco router supports ISDN?

A: There are several ways to verify whether your Cisco router will support ISDN. First, any Cisco router with a serial port can support an ISDN connection; however, it will not support many of the features of ISDN, such as caller ID. Second, if your router supports native ISDN, you can check for a BRI port in the back of the router. If there is a BRI port, you need to check to see whether it is labeled S/T or U. If it is labeled S/T, you need an external NT1 device and if it is labeled U, you do not need an external NT1 device. A third way to determine if your router supports native ISDN is to issue the **show version** command and look near the bottom for the list of interfaces.

Q: Is there any way to determine what type of ISDN switch type I am connected to?

A: The best way to identify the type of ISDN switch you are connected to is to contact your telephone company.

Q: What is the best topology to use when designing an ISDN DDR network?

A: That depends on the number of sites you need to connect. If there are only two sites you need to connect, the only choice is the point-to-point topology. If you have several sites you need to connect, there are several other choices, depending on these factors:

What is the total number of sites?

- If this number is greater than three or four, the fully meshed topology is probably too expensive. The hub-and-spoke or dual-hub-and-spoke topology would probably be best.

How much will each site use the link (for example, as a backup line, less than 30 minutes a day, more than 2 hours a day, etc.)?

- If the DDR lines will not be used often, and there are multiple sites to be connected, you could use dialer interfaces and configure a hub-and-spoke topology.

How important is uptime?

- If these sites must be connected as much as possible, the dual-hub-and-spoke topology is probably the best solution.

Q: How do I know whether to use a dialer interface or to configure legacy DDR?

A: Chapter 6 covers DDR in more detail, specifically dialer interfaces. Once you understand dialer interfaces, you can choose the best solution for your network.

Q: How do I determine what routing design to use for my DDR network?

A: This can be a very complex question. The answer really depends on several different factors:

What is the topology being used?

- For a fully meshed design, snapshot routing will not work.
- If it is a hub-and-spoke design, any of the routing methods will work.

Are the DDR link connections to stub networks?

- If the DDR link is between two routed networks, static routing can be cumbersome. Snapshot and OSPF on-demand routing would probably be better solutions in this case.

How long is the DDR link usually connected/disconnected?

- If the connection is usually kept open longer than five minutes, then snapshot routing is essentially free. If the connection times were quick (under one minute), then snapshot routing would have to keep the link established for a longer period of time, adding to the cost of the line.

How often does the routing table change for the rest of the network?

- If the network changes frequently (several times a day/week), then maintaining the routing tables statically would be time consuming. Running either on-demand or snapshot routing would be a better solution.
- If the network is stable, any of the routing methods will work.

What is the size of the network?

- If the network is relatively small, static routing might be the easiest solution. If the network is medium to large, snapshot or on-demand designs would be more efficient.

Enabling Dial-on-Demand Routing (DDR)

Solutions in this chapter:

- Dialer rotary groups
- Dialer profiles
- Virtual profiles
- Fine tuning connections

Introduction

In Chapter 5, we looked at using Integrated Services Digital Network (ISDN) and dial-on-demand routing (DDR) to enhance on-demand connectivity, using legacy DDR configurations. In this chapter, we will take a more in-depth look at DDR and how to optimize DDR connectivity using rotary groups, dialer profiles, and virtual profiles. The final section of this chapter will look at fine-tuning connections using dialer lists and dialer timers.

Rotary groups and dialer profiles are ways in which we can separate the logical and physical interface configurations. Although they both use dialer interfaces, rotary groups are used with legacy DDR, and dialer profiles provide us with a more flexible and scalable way of configuring DDR than legacy DDR.

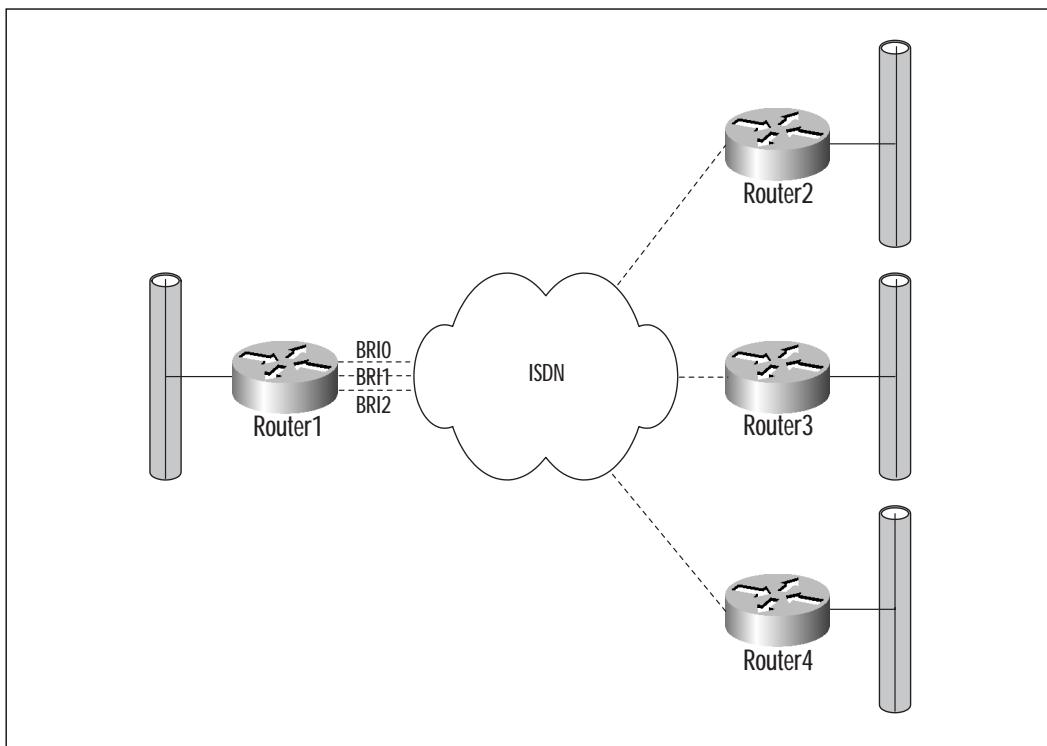
Dialer Rotary Groups

Chapter 5 introduced dialer rotary groups as a method of grouping multiple physical interfaces for use with DDR. When configuring either a Basic Rate Interface (BRI) or Primary Rate Interface (PRI) into a dialer rotary group, multiple B-channels are automatically put into the same rotary group.

Figure 6.1 shows an example of a dialer rotary group. There are three BRI interfaces on Router1 connecting to multiple sites. With the dialer rotary group configured, if BRI0 is connected to Router2 and interesting traffic destined for Router3 enters Router1, the rotary group will allow the router to connect to Router3 using the next available interface. The dialer group allows all three sites to be dialed from any of the interfaces, based on availability. Additionally, since each BRI contains two B-channels, Router1 could be connected to as many as six sites at one time. Configuring one interface to dial several different locations is called a dialer profile, which is covered in the next section of this chapter.

Configuring Dialer Rotary Groups

Configuring a dialer rotary group is fairly simple. For each physical interface you want in your rotary group, you enter the **dialer rotary-group group-number** command. Once each interface has been configured as part of a rotary group, you configure the dialer interface. The dialer interface is a virtual interface used with DDR. It contains most of the configuration for establishing the DDR link.

Figure 6.1 Dialer rotary group example.

To enter the dialer interface, use the **interface dialer group-number** command from the global configuration mode. The value range for the dialer group is between 0 and 255. Figure 6.2 shows an example of a rotary group router configuration.

Figure 6.2 Dialer rotary group configuration.

```
hostname Router1
isdn switch-type basic-nil
dialer-list 1 protocol ip permit
!
interface Ethernet0
ip address 192.168.1.1 255.255.255.0
!
interface BRI0
no ip address
```

Continued

```
encapsulation ppp
dialer rotary-group 1
!
interface BRI1
no ip address
encapsulation ppp
dialer rotary-group 1
!
interface dialer 1
ip address 172.16.3.1 255.255.255.252
encapsulation ppp
bandwidth 128
dialer in-band
dialer map ip 172.16.3.2 name Router2 broadcast 8358661
dialer load-threshold 127 either
dialer-group 1
ppp multilink
```

As you can see in Figure 6.2, both BRI0 and BRI1 are configured for rotary group 1, no IP address, and PPP encapsulation. The **dialer rotary-group 1** command defines the rotary group, enabling either interface to be used to dial Router2.

Interface dialer 1 is then configured for the remainder of the DDR information to make the call. As mentioned earlier, dialer interfaces are logical interfaces that are linked to a physical interface (or multiple interfaces in this example) for actual dialing.

NOTE

The only configuration needed on a physical interface when using dialer profiles is the encapsulation type and bonding to either rotary groups or dialer pools.

Dialer Profiles

The previous section on rotary groups briefly introduced dialer profiles. A dialer profile is a logical interface bound to a physical interface. In the instance of rotary groups, a single dialer profile can be bound to multiple physical interfaces. Additionally, you can have multiple dialer profiles bound to a single physical interface. A key difference between rotary groups and dialer profiles is that a physical interface can participate in only one dialer profile, whereas in a dialer profile configuration, a physical interface can participate in multiple dialer profiles. This means that you can configure one interface with multiple configurations. If you are using legacy DDR with BRI for a dial backup solution and have three different sites to back up, you need three BRI interfaces and three ISDN lines. If you are using dialer profiles, you need one BRI interface and one ISDN line. In many complex designs, using dialer profiles can save both time and money over using legacy DDR.

There are many items that need to be considered prior to configuring a dialer profile. Below is a list of items you will need to determine prior to configuring the actual dialer profile.

- **Physical interface** The interface that is linked to the dialer interface
- **Dialer list** Indicates interesting traffic (traffic that you need in order to keep the interface up)
- **Dialer interface** The interface that holds the configuration for dialing
- **Dialer pool** Allows group physical interfaces to dialer interfaces
- **Map class** This optional item simplifies configuration by grouping similar interface configurations into a single map class

NOTE

It is important to remember that when configuring the access list for defining interesting traffic, dynamic routing protocol updates are not considered interesting traffic.

Physical Interface

The physical interface is the interface that will physically connect and establish a “line up, protocol up” status. As mentioned in Chapter 5, this can be an ISDN BRI or PRI interface, an asynchronous serial interface or a synchronous serial interface. As with the rotary group, only a limited amount of configuration information needs to be put on the physical interface.

Dialer List

The dialer list is what identifies interesting traffic, which causes the router to initiate dialing and keep the interface in an “up, up” status. For a detailed overview of dialer lists, see Chapter 5.

NOTE

With the use of dialer profiles, you can have multiple dialer lists, each configured for a different profile.

Dialer Interface

Also introduced in Chapter 5, the dialer interface is the logical interface that holds the bulk of the configuration for use in both dialer profiles and rotary groups.

Dialer Pool

The dialer pool is used to group multiple physical interfaces to one dialer interface. The pool can be a value from 1 to 255 and can have multiple physical interfaces configured on one router. Additionally, a physical interface can belong to multiple dialer pools. There are two commands used to configure a dialer pool: **dialer pool number** and **dialer pool-member number**. The **dialer pool** command is placed on the dialer interface and the **dialer pool-member** command is placed on the physical interface.

Map Class

The dialer map class is an optional item that contains configuration commands used in more than one interface. If you have three dialer interfaces with similar timer settings, you can configure a map class to cut down on the number of times you need to configure them. The command needed to

configure a map class is **map-class dialer *class-name***. The commands you can configure under the map class are: **dialer isdn [speed 56|spc]**, **dialer idle-timeout *seconds***, **dialer fast-idle *seconds***, and **dialer wait-for-carrier-time *seconds***.

The following section details the procedures involved in configuring a dialer profile.

Configuring Dialer Profiles

The following example covers all the requirements needed for a dialer profile.

Figure 6.3 shows the setup for the following dialer profile example. In this example, Router1 has two BRI interfaces—one to connect to Router4 and the other to be used as backup for the Frame Relay connection to Router2 and Router3. Figure 6.4 shows the configuration of Router1.

Figure 6.3 Dialer profile example.

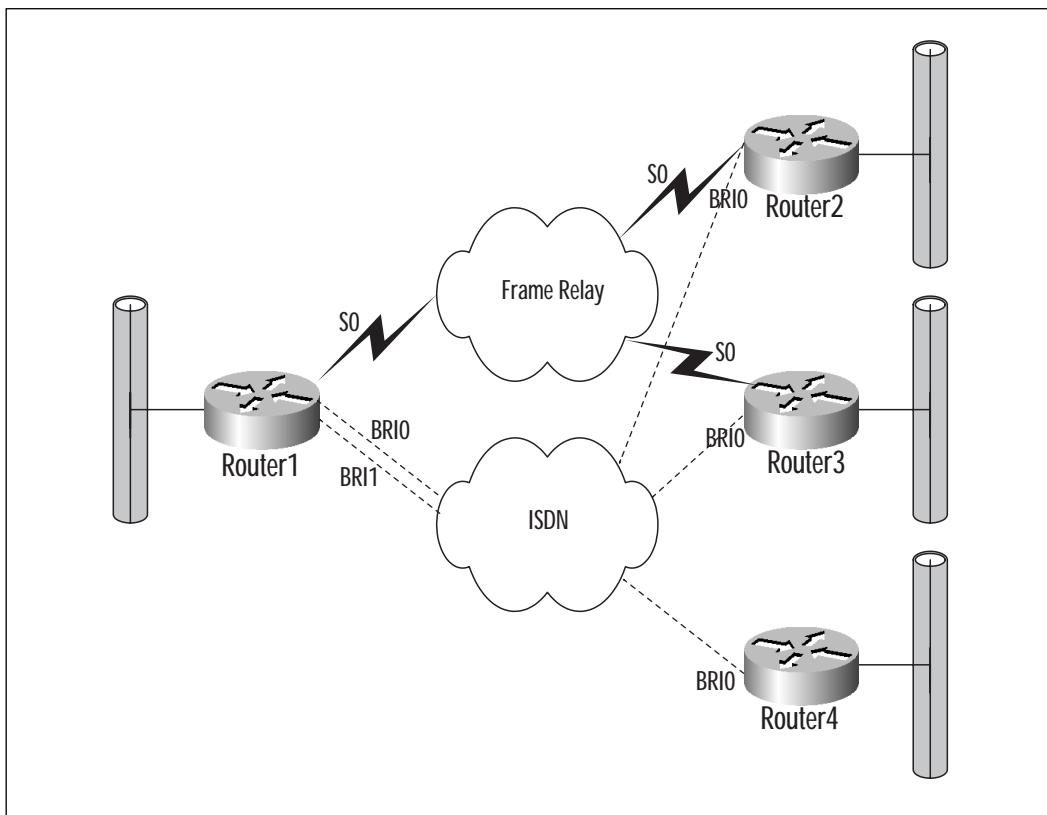


Figure 6.4 Dialer profile configuration.

```
hostname Router1
isdn switch-type basic-nil
dialer-list 1 protocol ip permit
!
interface BRI0
no ip address
encapsulation ppp
dialer pool-member 1 priority 50
dialer pool-member 2 priority 50
!
interface BRI1
ip address 172.16.5.1 255.255.255.252
encapsulation ppp
dialer map ip 172.16.5.2 name Router4 broadcast 8358662
dialer load-threshold 127 either
dialer-group 1
ppp multilink
!
interface dialer 1
ip address 172.16.3.1 255.255.255.252
encapsulation ppp
bandwidth 64
dialer in-band
dialer pool 1
dialer remote-name Router2
dialer string 8358661 class backup
dialer load-threshold 127 either
dialer-group 1
ppp multilink
!
interface dialer 2
ip address 172.16.4.1 255.255.255.252
encapsulation ppp
```

Continued

```
bandwidth 64
dialer in-band
dialer pool 2
dialer remote-name Router3
dialer string 8358661 class backup
dialer load-threshold 127 either
dialer-group 1
ppp multilink
!
map-class dialer backup
dialer fast-idle 30
dialer hold-queue 20
dialer idle-timeout 180
```

The first two bold commands in Figure 6.4 configure the BRI0 interface to be a member of dialer pools 1 and 2. The optional priority parameter can be used to specify that one pool receive priority over another. The priority range is from 0 (lowest) to 255 (highest) with a default value of 0.

The next two bold commands configure interfaces dialer1 and dialer2 to be members of dialer pools 1 and 2, respectively. Finally, the map class backup has been configured. You can see that under the dialer string commands in dialer pools 1 and 2, the class backup parameter has been used. The class parameter associates the map class backup with that interface when that string is dialed.

Virtual Profiles

The virtual profile feature of DDR is a method of customizing each dial-up connection with its own virtual interface. When using virtual profiles, as each user dials in to the network, he is assigned his own unique interface. This feature allows for a more scalable dial-up network. Some of the virtual profiles work if you are using DDR dialer profiles or legacy DDR, or even if DDR is not configured. One use of a virtual profile is for a specific user to get a specific IP address and/or routing entries.

NOTE

In the event you are using a dialer profile for a specific user, the virtual profile will override the configuration.

There are two components of a virtual profile: the generic component, which is information common to all dial-up users, including some router configuration; and the user-specific component with information about each user obtained from an authentication, authorization, and accounting (AAA) server. (See Chapter 8 for an overview of AAA.) When creating a virtual profile, you can use either the generic component (Case 1), the user-specific component (Case 2), or both (Case 3). Each of these cases is explained in the following section.

Case 1: Create a Virtual Profile Using the Virtual Template

In this first example, the virtual profile is created by applying the virtual template and a subset of the configuration obtained from the AAA server; the router will apply the configuration commands in the virtual interface to the physical interface. If the physical interface has been configured for legacy DDR or a dialer profile with no specific user, the virtual interface configuration will override the existing configuration. If, however, the interface has been configured with user information and a dialer profile, it will override the virtual profile. When the virtual interface is used, the router applies the configuration commands to the physical interface the user dialed into, whether it is an ISDN line, a serial line, or an asynchronous serial line.

Once the virtual interface commands have been applied, the router checks for user-specific information on the AAA server. If the AAA server contains interface-specific information for that user, it is ignored. Only non-interface-specific information is applied, such as access lists, routes, address pools, and route filters.

If you are using ISDN with virtual interfaces, the virtual interface is applied to the B-channel as opposed to the D-channel. This allows separate configurations on each B-channel for different users.

Configure a Virtual Profile Using Virtual Templates

To configure a virtual profile using a virtual template you need to perform the following steps:

1. Configure a virtual template interface
2. Group the virtual template interface with the virtual profile

Configure a Virtual Template Interface

The virtual template is a serial interface, which means you can configure the same commands on it as on any other serial interface, except shutdown and dialer commands. Figure 6.5 shows an example of a virtual template interface.

Figure 6.5 Configuration for virtual template interface.

```
Interface virtual-template 1
ip unnumbered ethernet 0
encapsulation ppp
ppp authentication chap
```

As you can see, the configuration for the virtual template is very simple; in addition to the commands above, you can configure many additional commands.

Group the Virtual Template Interface with the Virtual Profile

Grouping the virtual template with the virtual profile is done by issuing the **virtual-profile virtual-template number** command. The virtual templates can range from 1 to 30. With this method of creating a virtual profile, all interface-specific AAA commands are ignored and all other AAA commands such as routes and access lists are not. With this method of creating a virtual profile, there is no requirement for using AAA. If AAA is not used, all users that need access to the router must be specifically created in the router configuration.

Case 2: Create a Virtual Profile Using the AAA Server

In this case, the virtual profile is created solely from the configuration obtained from the AAA server. When a user establishes a Point-to-Point Protocol (PPP) session, the router contacts the AAA server and obtains user-specific information, which is then applied to the virtual profile for that user. The information is interpreted as IOS commands—as if the AAA server were directly connected to the router making configuration changes. Both interface and non-interface commands can be included in the information from the AAA server.

Once the router gets the commands from the AAA server, it applies them to the interface, overriding any previous configurations for that interface. When the PPP session is terminated, the virtual profile is deleted and the interface is restored to default configuration.

Configure a Virtual Profile Using the AAA Server

To configure a virtual profile using an AAA server, you need to perform the following steps:

1. Configure AAA on the router
2. Specify AAA as the virtual profile source
3. Configure the per-user configurations on the AAA server

Configure AAA on the Router

For details on configuring AAA on the router, refer to Chapter 8, “Securing Your Remote Access Network.”

Specify AAA as the Virtual Profile Source

To specify AAA as the virtual profile source you need to use the **virtual-profile aaa** command from the global configuration mode.

Configure the Per-user Configurations on the AAA Server

The following example contains an excerpt from both the AAA server and the router running per-user configurations. Figure 6.6 contains a per-user configuration for users Mike and Dan. For more details on per-user configurations on the AAA server, refer to Cisco’s Web site at www.cisco.com. In this example, two users are configured for authentication on the AAA server, and the router is configured to use AAA authentication.

Figure 6.6 AAA server configuration for virtual profile using AAA server.

```
AAA Configuration for Mike and Dan
mike Password = "ekimpass"
    User-Service-Type = Framed-User,
    Framed-Protocol = PPP,
    cisco-avpair = "interface_config=ip address 172.16.1.100
255.255.255.0,"
dan Password = "danssecret"
    User-Service-Type = Framed-User,
    Framed-Protocol = PPP,
    cisco-avpair = "interface_config=ip address 172.16.2.100
255.255.255.0"
```

The router in Figure 6.7 is configured to reference the AAA server for its virtual profile information. In this example, Mike would get IP address 172.16.1.100 when he dials in, and Dan would get IP address 172.16.2.100.

Figure 6.7 Router configuration for virtual profile using AAA server.

```
Router Configuration
aaa new-model
aaa authentication ppp default radius
aaa authorization network radius
virtual-profile aaa
!
interface dialer 0
ip address 10.0.1.1 255.255.255.0
encapsulation ppp
dialer map ip 10.0.1.2 name mike 8348661
dialer map ip 10.0.1.3 name dan 8348662
dialer-group 1
ppp authentication chap
```

Case 3: Create a Virtual Profile Using Both the Virtual Template and AAA Server

The configuration from the AAA server and the virtual interface template together make up Case 3. When using both AAA and virtual templates, the router processes a new PPP session in the following order:

1. The virtual profile is dynamically created from the information contained in the virtual template
2. The AAA server information is obtained and applied to the virtual profile

Just as in Case 2, if there is conflicting information in either the AAA server or the virtual template with the router, the router configuration is overwritten. This case offers the most customizable configuration possible. Specific user information as well as generic information can be combined to create user-unique profiles.

Configure a Virtual Profile Using Both the Virtual Template and AAA Server

To configure a virtual profile using both a virtual template and an AAA server, you need to perform the following steps:

1. Configure a virtual interface template
2. Configure AAA on the router
3. Configure the per-user configurations on the AAA server
4. Specify the virtual profile by both virtual templates and AAA

Steps 1, 2, and 3 are similar to the steps in the previous two cases.

Step 4 is a combination of Cases 1 and 2. Figures 6.8 and 6.9 show all four steps on both the AAA server and the router.

Figure 6.8 AAA server configuration for virtual profile using both virtual template and AAA server.

```
AAA Configuration for Mike and Dan
mike Password = "ekimpass"
    User-Service-Type = Framed-User,
    Framed-Protocol = PPP,
    cisco-avpair = "interface_config=ip address 172.16.1.100
255.255.255.0,"
dan Password = "danssecret"
    User-Service-Type = Framed-User,
    Framed-Protocol = PPP,
    cisco-avpair = "interface_config=ip address 172.16.2.100
255.255.255.0"
```

Figure 6.8 is an excerpt from the AAA server and is the same as the AAA server configuration used in the example on configuring a virtual profile using AAA.

Figure 6.9 Router configuration for virtual profile using both virtual template and AAA server.

```
aaa new-model
aaa authentication ppp default radius
aaa authorization network radius
virtual-profile virtual-template 1
```

Continued

```
virtual-profile aaa
!
interface Virtual-Template 1
ip unnumbered ethernet 0
encapsulation ppp
ppp authentication chap
!
interface dialer 0
ip address 10.0.1.1 255.255.255.0
encapsulation ppp
dialer map ip 10.0.1.2 name mike 8348661
dialer map ip 10.0.1.3 name dan 8348662
dialer-group 1
ppp authentication chap
```

Figure 6.9 is an excerpt from the router configuration for creating the virtual profile by both AAA and virtual templates. The two commands in bold group the virtual profile to both AAA and the virtual template. Creating the virtual template and configuring AAA are the same as in the previous examples.

Fine Tuning Connections

DDR has several options available for fine-tuning its connections. The biggest expense in DDR is the cost of the link, so most of the options available directly address timers used in maintaining and terminating DDR sessions. Another way of keeping costs down is by limiting when and how often the line gets established. This is done through dialer lists. By now you should have a good understanding of what the dialer list is and how to configure one. The next section reiterates this and gives more examples of dialer lists with additional information on setting specific dialing and disconnecting timers.

Dialer Lists

Interesting traffic is defined as traffic that the router deems important. The way to define this is by configuring an access list. All traffic destined for a DDR interface must pass through the dialer list before being marked “interesting.” When interesting traffic comes into the router destined for a remote network, the router establishes a call to the remote network and

sends the data. Once the circuit is connected, all traffic (including uninteresting traffic) can flow through the circuit. Once your defined interesting traffic stops (for a specified/configurable amount of time) the call will be disconnected.

NOTE

When the circuit has been connected, traffic that is marked interesting will reset the idle timer.

The idle timer is what causes the link to be terminated. Because the dialer list is tied to how long the line is kept open, it is important to configure the dialer list carefully. The limit on the number of dialer lists in a router is 10, but each list can have multiple entries. Figures 6.10 and 6.11 are examples of dialer lists; they are followed by a brief explanation of what traffic will be permitted or denied.

Figure 6.10 Dialer list example 1.

```
dialer-list 1 protocol ip list 101
!
access-list 101 permit tcp any any eq smtp
access-list 101 permit tcp any any eq www
access-list 101 permit tcp any any eq pop3
access-list 101 permit tcp any any eq telnet
access-list 101 permit icmp any any
access-list 101 deny any any
```

The dialer list in Figure 6.10 permits only IP traffic that passes access list 101. Access list 101 allows only e-mail, WWW, Telnet and ICMP traffic.

Figure 6.11 Dialer list example 2.

```
dialer-list 1 protocol ip permit
dialer-list 1 protocol appletalk permit
dialer-list 1 protocol ipx permit
dialer-list 1 protocol decnet permit
```

The example in Figure 6.11 allows IP, AppleTalk, IPX, and DECNET traffic to initiate a connection. This type of dialer list would be costly if the line being used was measured by how long it was connected.

Dialer Timers

In addition to dialer lists, dialer timers are another way of keeping DDR costs down. There are several different timers associated with DDR. The timers are:

- Enable-timeout
- Fast-idle
- Hold-queue
- Idle-timeout
- Wait-for-carrier-time

The enable-timeout timer sets the amount of time that an interface stays down before it is capable of dialing. The command syntax is **dialer enable-timeout seconds**, where seconds is a value between 1 and 2147483. The default is 15 seconds.

The fast-idle timer is a timer that overrides the idle-timeout timer. If an interface is connected to location A and traffic destined for location B enters the router and the interface cannot dial, the fast-idle timer starts counting down to 0. Once the fast-idle timer reaches 0, the interface is reset, allowing the traffic destined for location B to be sent. The syntax for the fast-idle timer is **dialer fast-idle seconds**, where seconds is a value between 1 and 2147483. The default value for the dialer fast-idle time is 20 seconds.

The hold-queue is a queue that the interface maintains. If the interface is not connected and interesting traffic comes in, the hold-queue holds a specified amount of packets while the interface is brought up. Once the interface is connected, the hold-queue is emptied and any future traffic can flow directly through the interface. The syntax for the hold-queue is **dialer hold-queue packets [timeout seconds]**, where packets is the number of packets to be held from 0 to 100 and the optional timeout parameter is how long the packets will be kept while the interface is being connected. By default, the hold queue is 0, which means that during a call establishment all incoming packets will be dropped.

As mentioned earlier, the idle-timeout is the amount of time the router waits between seeing interesting traffic and disconnecting the line. Once an interface is connected, the idle-timeout timer is started. Once the timer reaches 0, the interface is disconnected. If interesting traffic enters the

router during the call, the idle-timeout timer is reset. The syntax for the command is **dialer idle-timeout seconds [either]** where seconds is the amount of time before disconnecting the line (between 1 and 2147483 seconds) and either informs the router to count both inbound and outbound traffic for the idle-timeout. The default idle-timeout is 120 seconds.

The wait-for-carrier-time timer is how long the router will wait for a carrier to come up before dialing. The syntax for this command is **dialer wait-for-carrier-time seconds**, where seconds is a value between 1 and 2147483. The default wait-for-carrier-time is 30 seconds.

Walkthrough

The following walkthrough shows how to configure a router to make multiple connections over the same physical interface. In this example, a 3640 router is used with PRI, FastEthernet, and Digital modem modules. The 3640 is configured to accept analog and ISDN dial-up connections as well as a connection to a remote 3620 router, all through the PRI interface.

Figure 6.12 shows the network diagram. Figure 6.13 is the router configuration for the 3640.

Figure 6.12 PRI with ISDN dialup, ISDN dialout, and analog dialup.

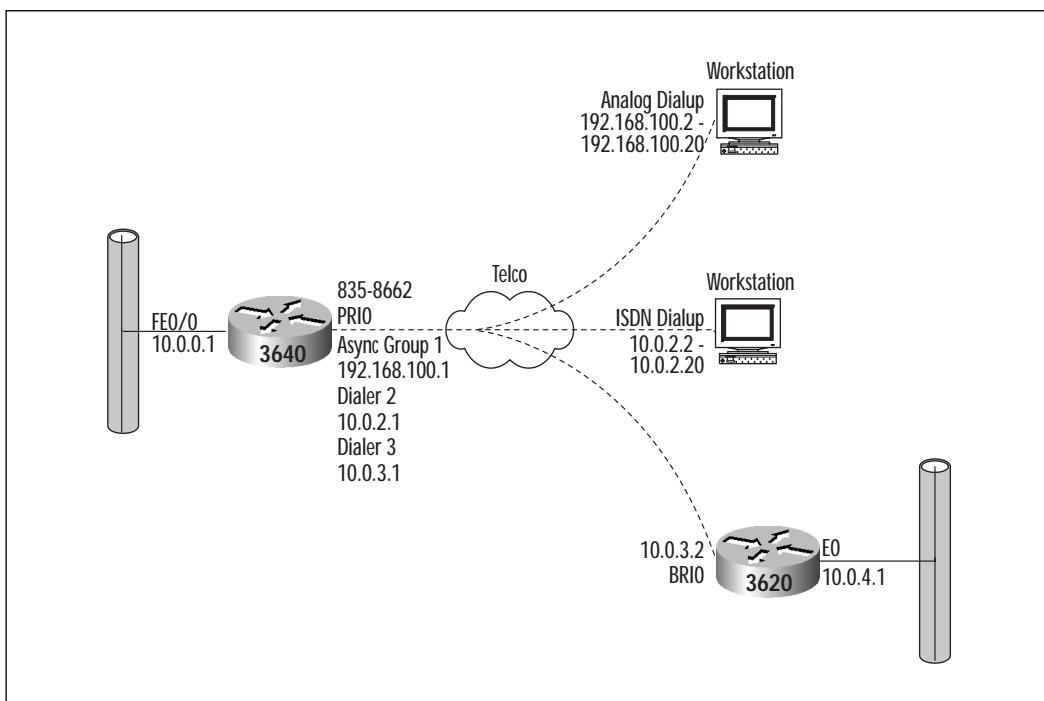


Figure 6.13 3640 router configuration.**(Section 1)**

```
hostname Cisco3640
!
username alicia password alicia
username andy password andy
username brad password brad
username chad password chad
username jeff password jeff
username john password john
username Cisco3620 password chappass
!
isdn switch-type primary-dms100
!
```

(Section 2)

```
controller T1 0/0
framing esf
linecode b8zs
pri-group timeslots 1-24
!
interface FastEthernet 0/0
ip address 10.0.0.1 255.255.255.0
!
```

(Section 3)

```
interface Serial 0/0:23
description PRI D-channel
no ip address
encapsulation ppp
dialer pool-member 2
dialer pool-member 3
!
```

Continued

(Section 4)

```
interface Group-Async 1
description connected to Dial-inPCs(modem)
ip address 192.168.100.1 255.255.255.0
encapsulation ppp
dialer in-band
dialer idle-timeout 180
async mode dedicated
group-range 33 64
ppp authentication chap pap callin
peer default ip address pool analogdialup
!
```

(Section 5)

```
interface Dialer 2
description connected to Dial-inPCs(ISDN)
ip address 10.0.2.1 255.255.255.224
encapsulation ppp
dialer in-band
dialer idle-timeout 180
dialer pool 2
ppp authentication chap pap callin
ppp multilink
peer default ip address pool isdndialup
!
```

(Section 6)

```
interface Dialer 3
description connected to Cisco3620
ip address 10.0.3.1 255.255.255.252
encapsulation ppp
dialer idle-timeout 120
dialer remote-name Cisco3620
dialer-group 1
dialer string 8358665
```

Continued

```
dialer hold-queue 20
dialer idle-timeout 60
dialer fast-idle 4
dialer pool 3
ppp authentication chap
snapshot server 15 dialer
!
```

(Section 7)

```
dialer-list 1 protocol ip list 101
ip local pool isdn-dialup 10.0.2.2 10.0.2.20
ip local pool analog-dialup 192.168.100.2 192.168.100.20
!
```

(Section 8)

```
access-list 101 permit tcp any any eq smtp
access-list 101 permit tcp any any eq www
access-list 101 permit tcp any any eq telnet
access-list 101 permit tcp any any eq pop3
access-list 101 permit icmp any any
access-list 101 deny any any
!
router rip
version 2
network 10.0.0.0
network 192.168.100.0
!
```

(Section 9)

```
line 33 64
exec
autoselect ppp
autoselect during-login
login local
modem InOut
transport input all
```

Figure 6.13 shows the router configuration for the 3640. The following is an explanation of the numbered sections in Figure 6.13:

Section 1 sets up the dial-up user names and passwords. It also configures the router name for the connection to the 3620 and its Challenge Handshake Authentication Protocol (CHAP) password.

Section 2 is the configuration for the PRI controller. The framing has been configured as Extended Superframe (esf), the linecode is set to binary eight zero signaling (b8zs), and all 24 time slots are being made available to the controller.

Section 3 is the configuration for the D-channel of the PRI interface. The last channel of a T1 circuit is typically the D-channel. The encapsulation is being set to ppp and the two dialer pools (2 and 3) are being identified. Once the dialer pools have been identified, the router will know what physical interface to use to establish calls for that dialer.

Section 4 is the configuration for analog dial-up users. In this interface, the IP address, encapsulation, PPP authentication, and dialer options are configured. Of the dialer options, the idle-timeout is set to 180 seconds, which will disconnect any dial-up users after 180 seconds of no activity. The **group-range 33 64** command identifies what lines to use for this interface. The lines for the modems will vary depending on the physical configuration of the router. The IP address pool for this interface is also identified as the analogdialup pool. Section 7 contains the configuration of the pool.

Section 5 is the configuration for the dial-up ISDN connections. This interface (Dialer 2) shares many of the same commands as the Group-Async 1 interface. The differences are the IP address pool (ISDN dialup versus analog dialup), PPP multilink, the **group range 33 64** command, and the reference to the dialer pool (dialer pool 2).

Section 6 is the configuration for the DDR connection to the 3620 remote router. This interface also shares many commands with the previous two interfaces. The additional commands configure snapshot routing (snapshot server 15 dialer) and set the fast-idle time to 4 seconds (dialer fast-idle 4). The fast-idle setting will allow the router to quickly hang up the line to make it available for a dial-up user.

Section 7 contains the dialer list for identifying interesting traffic and the IP address pools for the two dial-up configurations. The interesting traffic has been identified as IP traffic which passes IP access list 101. (**Section 8** describes the access list.) The two IP address pools identify IP addresses that will be assigned to dial-up clients when they establish a connection. This access list allows all SMTP, POP, WWW, Telnet, and ICMP traffic to establish a connection to the 3620 remote router.

Section 9 is the configuration for the digital modems for analog dial-up users. This configuration allows users either to connect directly to the router (exec) or to establish a PPP session (autoselect ppp) and connect to the Internet.

This example shows how one physical interface can be configured to perform multiple tasks based on some of the advanced DDR commands covered in this chapter.

Summary

This chapter covered rotary groups, dialer profiles, virtual profiles, and fine-tuning DDR connections.

The rotary group is used when there are multiple physical interfaces through which to place a call. In the event that one interface is busy, the rotary group will use the next available interface to make a call. A dialer rotary group does not need to be configured for both BRI and PRI B-channels; the multiple B-channels in either interface are automatically placed into a dialer rotary group.

Dialer profiles are based on separate logical interface configurations bound to physical interfaces. They involve configuring a profile, which is kept separate from the physical interface. Once the profile has been configured, it is bound to the physical interface. Multiple profiles can then be linked to one interface, allowing multiple sites to be called from the same interface. Additionally, one profile can be linked to multiple interfaces, allowing greater bandwidth per call.

Virtual profiles are used in dial-up networks to configure unique interfaces for each individual user. You can use a virtual interface, AAA server, or both to create a virtual profile. The virtual interface contains information that will be applied to all users, such as encapsulation type and dial timers, and the AAA server contains user-specific information such as access lists and routes.

DDR has several different methods of keeping connection times short and deciding how often the line is brought up. Dialer lists and dialer timers are two methods. Dialer lists are used to determine what kind of traffic is interesting, which tells the router to make a DDR connection. Dialer timers can be used to make the connection hang up more quickly and queue packets while the connection is being made.

FAQs

Q: I have a hub-and-spoke Frame Relay network and need to set up a backup solution. I have decided to use ISDN to accomplish this. Do I need to use dialer profiles or can I use legacy DDR?

A: The answer depends mainly on how many sites you need to back up. If you are backing up one site, you can use legacy DDR. If you are backing up more than one site and do not want to pay for two ISDN lines for each office, you can use dialer profiles. If you are backing up enough sites, you may want to use a PRI line at the hub site. If you are using a PRI line, you can configure either legacy DDR or dialer profiles, depending on how complex your network is. The most important thing to keep in mind is that dialer profiles allow you to configure one interface to dial out with multiple different configurations; if your hub is going to be receiving calls, a dialer profile will not be necessary.

Q: I need to set up virtual profiles, but do not have an AAA server. How hard is it to configure an AAA server?

A: If you want to use virtual profiles you do not have to use AAA. Remember you can use a virtual interface template for virtual profiles. But to answer your question, Cisco has an AAA server called the Access Control Server. More information can be obtained from Cisco's Web site at www.cisco.com.

Q: Can I configure both a rotary group and a dialer profile on the same router?

A: Yes and no. You can configure both a rotary group and a dialer profile on the same router; the same physical interfaces cannot be used for both. If you have BRI0 as a member for rotary group 1, it cannot be a member of a dialer profile.

Configuring and Backing Up Permanent Connections

Solutions in this chapter:

- Configuring point-to-point connections
- Understanding and configuring X.25 connections
- Configuring Frame Relay connections
- Configuring and troubleshooting ATM connections
- Backing up permanent connections

Introduction

When analyzing the traffic requirements between remote offices and your central site, you may find it is not cost-effective to use an on-demand connection. Under these circumstances, you need to implement a permanent connection.

This chapter will explore several ways of providing permanent connections: point-to-point links (leased lines), X.25, Frame Relay, and Asynchronous Transfer Mode (ATM). Although X.25 is perhaps not the perfect choice for implementing a new network, there are times when you may need to extend or connect to an existing X.25 network, so this chapter will look at X.25 technology. Frame Relay is currently the most common method used to connect a wide area network (WAN); ATM is also commonly used for WAN connections. We will look at these technologies and see how they can be used to connect remote sites to a central site.

As organizations become more reliant on their network infrastructure, network engineers are required to provide a higher level of service. The final section of this chapter will look at ways of back up these connections to provide different levels of resilience.

Configuring Point-to-Point Connections

In today's WAN arena, point-to-point networks are a very common method for connecting a remote site to another site. When implementing point-to-point connections there are many options to choose from. A point-to-point link can be a simple dial-up connection, a dedicated serial link, or an Integrated Services Digital Network (ISDN) connection. Regardless of the type of link, you'll need a protocol to allow communication over that link. Let's look at two protocols that can be implemented over point-to-point links: Point-to-Point Protocol (PPP) and High-Level Data Link Control (HDLC).

PPP is designed for links that transport packets between two peers. PPP can operate across asynchronous, synchronous, ISDN, and dial-up point-to-point implementations. PPP links provide a simultaneous, full-duplex, bi-directional operation, and are assumed to deliver packets in order. PPP encapsulates higher-layer protocol packets—such as Internet Protocol (IP), Internetwork Packet Exchange (IPX), and AppleTalk—into PPP packets for transmission across the link on a first-come, first-served basis. PPP is a standard international protocol, and can be used in multi-vendor environments.

HDLC is a widely-used protocol for encapsulation techniques on point-to-point dedicated links. HDLC is derived from IBM's Synchronous Data

Link Control (SDLC) protocol suite. HDLC specifies the encapsulation method in point-to-point synchronous links, and it is the default encapsulation for Cisco serial interfaces.

The following diagram and configurations provide details on how to configure a simple point-to-point network.

Figure 7.1 A simple point-to point-network.

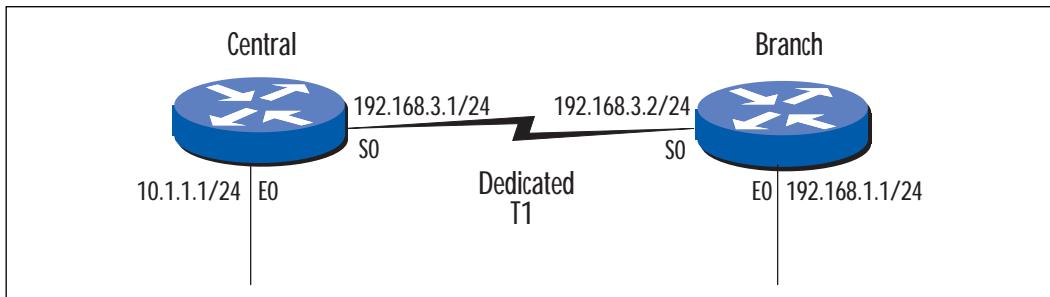


Figure 7.2 Point-to-Point Configurations.

```
Central#  
  
!  
version 11.3  
!  
hostname Central  
!  
interface Ethernet0  
ip address 10.1.1.1 255.255.255.0  
!  
interface Serial0  
ip address 192.168.3.1 255.255.255.0  
no shutdown  
!  
router rip  
network 192.168.3.0  
network 10.0.0.0  
!
```

Continued

Figure 7.2 Continued.

```
end

Branch
!
version 11.3

!
hostname Branch
!
interface Ethernet0
ip address 192.168.1.1 255.255.255.0
no shutdown
!
interface Serial0
ip address 192.168.3.2 255.255.255.0
no shutdown
!
!
router rip
network 192.168.3.0
network 10.0.0.0
```

Notice that Figure 7.2 did not specify an encapsulation on any of the serial interfaces. This means that the encapsulation would be HDLC, the default encapsulation on serial interfaces in Cisco routers.

If you wanted to use PPP instead of HDLC, you would enter the following command in interface configuration mode for each of the connected serial interfaces:

```
Central(config-if)# encapsulation ppp
```

Keep in mind that the encapsulation must be the same on both sides of the link, or no communication will be possible over that link.

X.25 Connections

X.25 technology was developed in the early days of computer networking, and was designed for unreliable and slow-speed networks. During the days in which X.25 was commonly used, people didn't have the option of running multimedia, voice, or any other high-bandwidth application over a data network. This accounts for the differences between X.25 and some of the newer technologies that are currently available like ATM, Frame Relay, or ISDN. The following sections review some of the advantages and disadvantages of using X.25. At the same time, it uses X.25 to introduce some of the more common solutions currently in place.

X.25 Overview

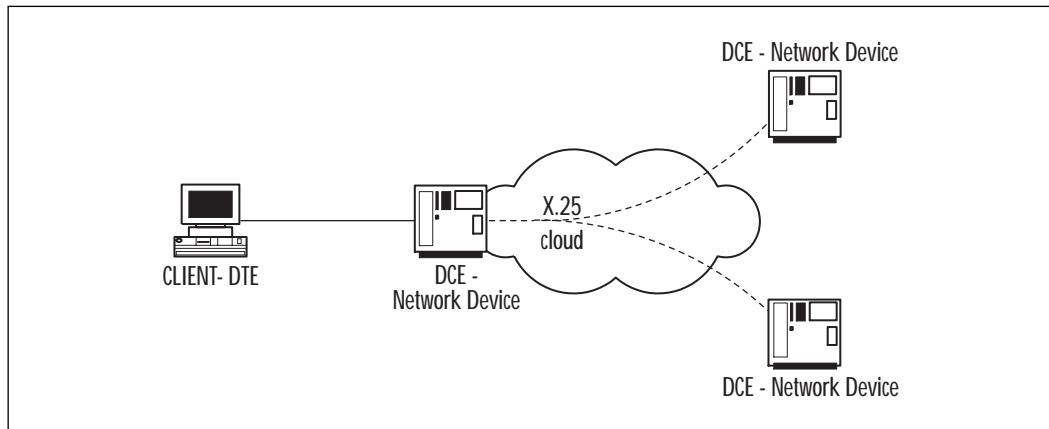
X.25 was developed when some of the newer technologies were yet to be discovered. It's a protocol that runs up to Layer 3 in the Open System Interconnection (OSI) stack, which means it can be routed. Today we're seeing that X.25 is being replaced with faster technologies such as ATM, Frame Relay, or ISDN. One of the primary benefits of X.25 was its ability to provide error checking, which was needed when most data networks were running over slow, error-prone public networks. This benefit, however, has also been looked upon as a disadvantage, due to the delay added as errors are being checked. X.25 defines the first three layers in an ISO network model:

- **Layer 1 (the physical layer)** is concerned with electrical or signaling functions. It includes several standards such as X.21 and other serial cable standards like V.35 and RS232.
- **Layer 2 (the data-link layer)** Link Access Procedure Balanced (LAPB) is a data-link layer protocol that provides an error-free link between two connected devices. LAPB is derived from the HDLC standard of ISO.
- **Layer 3 (the network layer)** is referred to as the X.25 Packet Layer Protocol (PLP) and is primarily concerned with network routing functions and the multiplexes permanent virtual circuits (PVCs), switched virtual circuits (SVCs) type of logical connections over a single physical connection.

Data Terminal Equipment (DTE) and Data Circuit-Terminating Equipment (DCE)

X.25 utilizes a connection-oriented service, which ensures that packets are transmitted in order. The end-user connection is called DTE and the connection on the network (carrier) side is called DCE (see Figure 7.3). The user (DTE) can communicate with multiple users simultaneously on a single physical line, with multiple logical channels. On one physical line there can be as many as 4096 logical channels.

Figure 7.3 X.25 DTE and DCE connectivity.



Packet Assembler/De-assembler (PAD)

In the early 80s, a majority of data processing was done utilizing asynchronous terminals, which are character-oriented. These asynchronous terminals are then connected to a device called a Packet Assembler/De-assembler (PAD), which collects characters and sends them as a packet through the X.25 network. In Figure 7.3, in place of a client DTE device, a PAD that connects to asynchronous terminals would be used.

Frames in X.25

Frames in X.25 are defined into three categories, Information Frames (IF), Supervisory Frames (SF), and Unnumbered Frames (UF). IFs carry the user data and sequence numbers to tell the other end what is received and what is expected. SFs handle flow and error control; they also indicate the final packet (no data to send). UFIs control Mode setting commands and responses. They are carried over LAPB frame format (see Table 7.1). LAPB frames include the following fields:

A header flag of 01111110 delimiting the beginning of the frame.

The address field (1 byte or 2 bytes), really used for link commands and responses—the real addressing is done at the packet layer. (The packet layer address is called Data Network Identification Code (DNIC)). The address field simply indicates whether the frame is a command frame or a response frame.

A control byte, which specifies whether the frame is an Information frame (IF), Supervisory frame (SF), or an Unnumbered frame (UF).

The information field follows the control field. The information field contains the upper layer data (encapsulated in a PLP packet).

The FCS field (frame check sequence) provides error checking and guarantees the integrity of the transmitted data.

The trailer flag (also 01111110) delimits the end of the frame.

Table 7.1 X.25 Packet Format

Flag	Address	Control	Information	FCS	Flag
01111110	8 bits	8 or 16 bits	Variable no. of bytes	16-bit check sum	01111110

The X.25 protocol is defined in three parts, corresponding to the lower three layers of the OSI model.

X.21 defines physical layer characteristics and maps to the physical layer in the OSI model.

LAP-B mode maps to the data-link layer in the OSI model.

Packet Layer Protocol provides connection-oriented transport over virtual circuits and maps to the network layer in the OSI model.

The other protocols related to X.25 are: X.3, X.29, X.75, and X.121. These are also called International Telecommunication Union Telecommunication Standardization Sector (ITU-T) standards for the X.25 series.

X.3 Specifies the parameters for PAD terminal handling. X.3 controls such elements as the baud rate, flow control, local echo, and cursor style.

X.29 Specifies the multiplexing and de-multiplexing of characters into an X.25 packet. It sends these packets to an asynchronous terminal, via asynchronous lines, connected to the PAD.

X.75 Specifies the interoperability between two or more public switching X.25 networks.

X.121 Specifies the X.25 addressing standard. It is also called the DNIC (Data Network Identification Code) address.

X.25 Virtual Circuits

A virtual circuit is simply a logical circuit that provides reliable connectivity between two DTE devices. Physically, the connection may pass through many different intermediate nodes along the way, but logically it appears to be a single link between the two communicating devices.

X.25 supports two types of virtual circuits, switched virtual circuits (SVCs), and permanent virtual circuits (PVCs).

SVCs provide a temporary link to transmit data; they are established and terminated on an as-needed basis. During a data transfer, the DTE devices are required to establish, maintain, and terminate the session. This has to happen each time the two devices need to communicate. An SVC would be useful in a situation that requires sporadic data transfers.

PVCs, on the other hand, are permanently in place and always ready to transfer data. The session is always active. A PVC is useful in a situation that requires frequent and consistent data transfers.

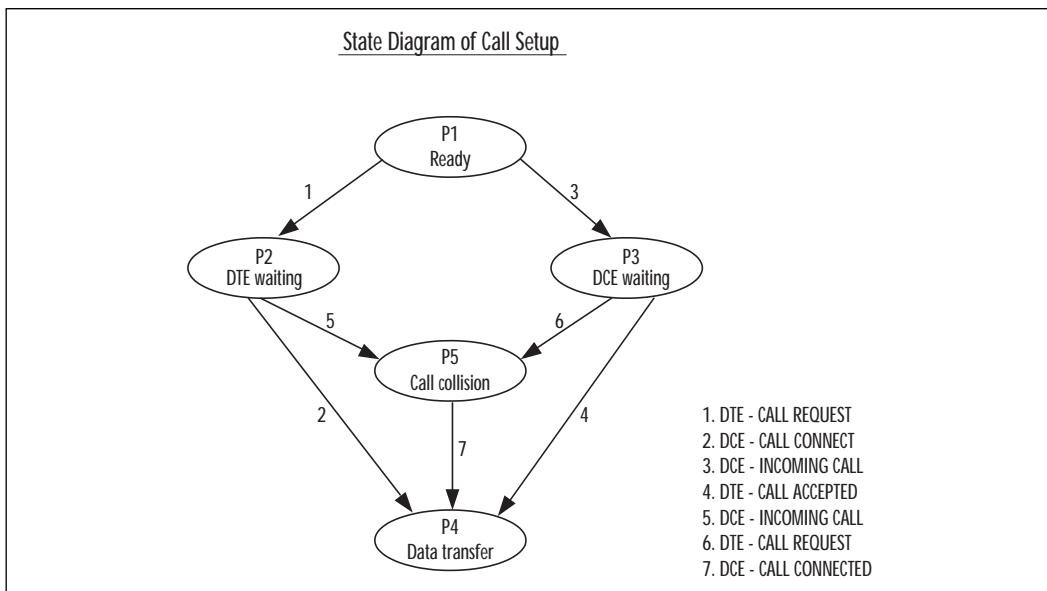
X.25 Call Setup and Disconnection

In the network layer, the packet is defined with a general format ID, logical channel group number, Logical Channel Number (LCN), and packet type. The establishment and termination of a virtual circuit (PVCs and SVCs) occurs at the packet level. Sliding windows, flow control per virtual circuits (VC), and recovery functions also occur at the packet level.

Table 7.2 and Figure 7.4 illustrate the call setup and disconnection process.

Table 7.2 Call Setup and Disconnection

Call Request	→	Incoming Call
Call Connected	←	Call Accepted
Data	→	Data
Data	←	Data
Clear Request	→	Clear Indication
Clear Confirmation	←	Clear Confirmation
...		...

Figure 7.4 X.25 Call Setup.

Configuring X.25

This section describes how to configure an X.25 network. First, you need to understand a little bit about how X.25 addressing works. X.25 networks use the X.121 addressing format. X.121 addresses are used by X.25 to establish virtual circuits. Table 7.3 illustrates the X.121 address format.

Table 7.3 X.121 Address Format

International Data Number (IDN)	
DNIC 4 digits	NTN up to 10 digits
Country 3 digits	PSN 1 digit

An X.121 address consists of the International Data Number (IDN), which in turn consists of two sub-fields: the DNIC, and the National Terminal Number (NTN).

The four-digit DNIC portion of the X.121 address consists of two sub-fields: the country code (three digits), which identifies the country in which the destination network resides (the code for the United States is 311), and the Packet Switched Network (PSN), a single digit that basically identifies the X.25 provider (AT&T or Tymnet, for example).

The NTN portion of the X.121 address specifies the unique identifier that is assigned the exact DTE device for which the packet is destined. The NTN field may vary in length.

Now that you understand the addressing, let's look at a sample X.25 implementation. Refer to Figure 7.5. We will use two routers, Central-1, and Branch-1. Central-1 is a hub site, which is where the majority of corporate hosts, (servers, mainframes, etc.) are located. The remote site will tie into the central site via an X.25 connection. Look at the hub site X.25 (X.121) addresses below. Remember, the first three digits (311) are the US country code. The fourth digit (0) is the X.25 service provider ID. In this case, let's pretend AT&T is assigned the zero ID. The last four digits (1234) are the unique ID of the DTE device/hub site router. The same rules apply to the remote site address. Check out the figure and the accompanying configurations:

Hub site X25 – address = 31101234

Remote site X25 – address = 31103456

Figures 7.5, 7.6, and 7.7 show additional configuration detail. Figure 7.5 is a simple example of an X.25 implementation.

Figure 7.5 Example of an X.25 network.

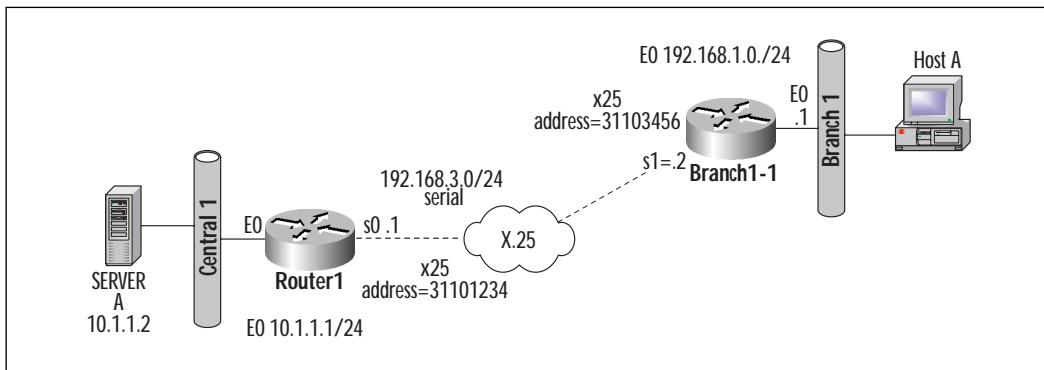


Figure 7.6 Central Router Configuration.

```
Central-1 #
!
version 11.3
!
```

Continued

Figure 7.6 Continued.

```
hostname Central-1

!
interface Ethernet0
ip address 10.1.1.1 255.255.255.0
no ip route-cache
no ip mroute-cache
!

!
interface Serial1
ip address 192.168.3.1 255.255.255.224
no ip route-cache
no ip mroute-cache

x25 address 31101234
!specify the node address given by X.25 service provider
!
x25 map ip 192.168.3.2 31103456 broadcast
!          Map statement provides mapping between remote
!          X.121 address and tcp/ip address. The broadcast option
provides a mechanism to send broadcasts to remote interface.
!
! encapsulation x25 dce
! clockrate 56000

These two statements are needed while using
back-to-back routers to simulate an x.25 network.

no shutdown
!
!
router rip
network 192.168.3.0
network 10.0.0.0
```

Continued

Figure 7.6 Continued.

```
!
ip classless
!
line con 0
!
end
```

Figure 7.7 Branch Router Configuration

```
Branch1-1 #
!
version 11.3

!
hostname Branch1-1

!
interface Ethernet0
ip address 192.168.1.1 255.255.255.0
no shutdown

!
interface Serial0
ip address 192.168.3.2 255.255.255.0
encapsulation x25
no ip route-cache
no ip mroute-cache
x25 address 31103456
x25 map ip 192.168.3.1 31101234 broadcast
no shutdown

!
!
! The statement below activates ip routing for specific networks
using rip
```

Continued

Figure 7.6 Continued.

```
router rip
    network 192.168.3.0
    network 10.0.0.0
!
ip classless
!
line con 0
end
```

Verifying and Troubleshooting X.25 Connections

The Cisco IOS provides many tools for monitoring X.25 connections. Some of the important commands are:

show interface Serial nn Displays information about serial interface and X.25 parameters.

show x25 interface serial nn Displays information about VCs.

show x25 map Displays information about address maps between IP and X.121 addresses.

show x25 vc Displays information about active SVCs and PVCs.

clear x25 Used to clear an SVC, or to reset a PVC.

debug x25 events Provides cause and diagnostic codes, which in turn provide information on why a call is rejected, disconnected, etc.

Additional X.25 troubleshooting information can be found at:

www.cisco.com/univercd/cc/td/doc/cisintwk/itg_v1/tr1919.htm

www.cisco.com/univercd/cc/td/doc/product/software/ios113ed/dbook/dx25.htm

Some common areas in troubleshooting X.25 networks include serial line encapsulation (making sure you have the correct encapsulation set on the serial interface of both connected devices), physical cabling (the physical connection/wires can sometimes be the root of connectivity problems), and X.121 address to LAN protocol address mapping (make sure the X.25 address is mapped to the correct LAN protocol (IP) address).

The **show interfaces serial exec** command provides useful information for identifying problems in X.25 networks.

```
Central1# show interfaces serial 1
LAPB state is SABMSENT, T1 3000, N1 12056, N2 20, k7,Protocol ip
VS 0, VR 0, RCNT 0, Remote VR 0, Retransmissions 2
IFRAMES 0/0 RNRs 0/0 REJs 0/0 SABMs 3/0 FRMRs 0/0 DISCs 0/0
```

The following fields of the **show interfaces serial** command provide particularly important information when troubleshooting X.25 networks:

REJs Number of rejects

SABMs Number of Set Asynchronous Balance Mode requests

RNRs Number of Receiver Not Ready events

FRMRs Number of protocol frame errors

DISCs Number of disconnects

Using the **show x25 interface** command, one can monitor virtual channel activity on the link.

```
Central1#sho x25 int s1
SVC 1024, State: D1, Interface: Serial1
Started 00:14:28, last input 00:00:02, output 00:00:22
Connects 31103456 <-> ip 192.168.3.2 (Examine the x25 address and ip
address)
Call PID ietf, Data PID none
Window size input: 2, output: 2
Packet size input: 128, output: 128
PS: 2 PR: 3 ACK: 2 Remote PR: 2 RCNT: 1 RNR: no
P/D state timeouts: 0 timer (secs): 0
data bytes 2468/1960 packets 34/35 Resets 0/0 RNRs 0/0 REJs 0/0 INTs
0/0
```

The **show x25 map** command displays information about address maps between TCP/IP and X.121 addresses. Upon examining the X.121 address and TCP/IP address closely, one can identify if there are any misconfigurations on the map.

```
Central1#show x25 map
Serial1: X.121 31103456 <-> ip 192.168.3.2
permanent, broadcast, 1 VC: 1024
```

Show x25 vc provides information regarding the virtual channels.

```
Central1#show x25 vc (the virtual channels are 1-1024, which provide a
logical path)

SVC 1024, State: D1, Interface: Serial1
Started 00:14:44, last input 00:00:18, output 00:00:10
Connects 31103456 <-> ip 192.168.3.2
Call PID ietf, Data PID none
Window size input: 2, output: 2
Packet size input: 128, output: 128
PS: 3 PR: 3 ACK: 3 Remote PR: 2 RCNT: 0 RNR: no
P/D state timeouts: 0 timer (secs): 0
data bytes 2560/1960 packets 35/35 Resets 0/0 RNRs 0/0 REJs 0/0 INTs
0/0
```

Show x25 services provides information about what services are available (like reverse-charging the telephone call, and what VCs are allocated). By using this information, one can establish if the X.25 service contributor is providing the contracted services and channels.

```
Central1#show x25 services
X.25 software, Version 3.0.0.

2 configurations supporting 2 active contexts
VCs allocated, freed and in use: 53 - 49 = 4
VCs active and idle: 2, 2
```

Debug x25 provides information about X.25 state transitions while the call is being set up, and reasons (if any) why the call is not being set up.

```
Central1# debug x25
Serial1: X.25 I R/Inactive Restart (5) 8 lci 0
Cause 0, Diag 27 (DTE originated/Packet too long)
Facilities: (0)
Call User Data (4): 0xCC000000 (ip)
Cause 0, Diag 26 (DTE originated/Packet too short)
Serial1: X.25 O P7 Clear Confirm (3) 8 lci 1
```

This command provides cause and diagnostic codes, provided in Table 7.4.

Table 7.4 Sample Cause Codes

Cause Code (Hex)	Description
00	DTE originated
01	Number Busy
05	Network Congestion
Diagnostic Code	Description
26	Packet too short
27	Packet too long

Frame Relay Connections

Over the past three to five years, Frame Relay has been the wide area service of choice. It provides an efficient, low-cost communication technology.

There are two types of Frame Relay connections: User-Network Interface (UNI), and Network-to-Network Interface (NNI). UNI defines the signaling between the end-user network device and the Frame Relay switch. NNI defines the signaling between the trunks connecting two different public Frame Relay clouds (like a connection between AT&T and MCI WorldCom). NNI is needed to provide end-to-end connectivity to a customer whose remote sites could be anywhere in the world (because a specific service provider may not have coverage in a given geographic area).

Frame Relay Overview

Frame Relay is packet-switching technology at the data-link level. The Frame Relay protocol originally had been part of the ISDN suite of protocols. In the late 80's and early 90's, Frame Relay became a separate protocol. It uses a simpler protocol suite than X.25, because it assumes the transport media is very clean. Any error checking and retransmissions are handled by upper-layer protocols, which make Frame Relay faster than X.25.

X.25 provides error-detection and error-correction algorithms at data-link and network layers. Error detection at the data-link layer is provided through cyclic redundancy check (CRC) checksum algorithms.

Frame Relay offers a high-speed version of packet switching, with many of the same techniques being employed to provide a complete network service. Data is forwarded in variable-length frames, and is multiplexed onto the transmission links. Frame Relay has the potential of operating effectively at much higher speeds (up to 45 Mbps) than existing packet switching systems like X.25. It is well suited to high-speed data applica-

tions, such as LAN connectivity, but is not well suited to delay-sensitive applications (voice, video), because of the variable length of the frames within the network.

A Frame Relay frame is transmitted to its destination by way of virtual circuits (logical paths from an originating point in the network) to a destination point. Virtual circuits may be one of two types: permanent virtual circuits (PVCs) or switched virtual circuits (SVCs).

A PVC is a permanently established connection between two endpoints on a Frame Relay network. A PVC can be used in a case where data transfers occur frequently and require fairly constant connectivity. PVCs do not require the time-consuming call setup and tear down procedures utilized in SVCs. Configuring a PVC requires only one-time setup by the network administrator, and the connection is permanently available, whereas SVCs are established and terminated on a call-by-call basis.

An SVC differs from a permanent virtual circuit in that SVCs only provide a temporary data transmission path. SVCs can be used in situations where only sporadic connectivity is required. Each time data needs to be transmitted, a new SVC must be established. After the transmission is complete, the SVC is terminated.

Table 7.5 is an example of the fields contained in a Frame Relay packet.

Table 7.5 Frame Relay Packet Format

Flag	Link Layer/Frame Relay Header	User Data	FCS	Flag
------	-------------------------------	-----------	-----	------

The Frame Relay packet format is designed based on low bit-error rates (1 in $10^{**}10$), with upper layers requesting retransmission of dropped packets or lost packets. The main functionality provided by the Frame Relay Switch is threefold:

1. **Error Checking** FCS uses 32-bit polynomial to check CRC and drop the packet if the checksum doesn't match.
2. **Addressing** Switch-checks the routing information in the packet and forwards it through the appropriate output port/PVC.
3. **Congestion Notification** If the switch buffers are full, it sends the congestion notification (forward or backward) depending on how the output/input buffers are filling up.

Let's refer to Table 7.5 and take a closer look at the fields contained in the Frame Relay packet.

Flag is an eight-bit sequence with bit stuffing, to identify the “start, end, start” sequence to delimit each packet.

Link Layer/Frame Relay Header contains addressing and error-checking functionality for Frame Relay. Take a look at Table 7.6. It shows the fields that are contained in the Frame Relay header.

Table 7.6 Frame Relay Header Format

DLCI	C/R	EA	FECN	BECN	DE
------	-----	----	------	------	----

Still referring to Table 7.6, let’s look at each of these fields in a little more detail.

DLCI Addressing in Frame Relay is called DLCI (Data Link Connection Identifier). A DLCI is a 10-bit, Layer 2 address (up to 1,024) that identifies a virtual circuit. Frame Relay networks assign each end of a connection with a Data Link Connection Identifier from a pool of locally unused numbers. The service provider’s Frame Relay network then maps one DLCI to the other, using a look-up table. Locally significant DLCIs have become the primary method of addressing because the same address can be used in several different locations while still referring to different connections. Thus, local addressing prevents a customer from running out of DLCIs as the network grows.

C/R The command response bit, which is not used in most Frame Relay networks.

EA The Extended Address field signifies up to two additional bytes in the Frame Relay header, thus greatly expanding the number of possible addresses.

FECN The Forward Explicit Congestion Notification bit lets the receiving router know that congestion exists in the path that the frame came from.

BECN The Backward Explicit Congestion Notification bit lets the receiving router know that congestion exists in the reverse of the path that the frame came from.

DE If the Discard Eligibility bit is set on a frame, it means that this frame is eligible to be discarded if the Frame Relay network becomes congested.

Let’s look at FECN, BECN, and DE in a little more detail.

When the network becomes congested to the point that it cannot process new data transmissions, it begins to discard frames (frames with the DE bit set to 1). These discarded frames are retransmitted, thus causing more congestion. In an effort to prevent this situation, several mechanisms

have been developed to notify user devices at the onset of congestion, so that the offered load may be reduced.

Two bits in the Frame Relay header are used to notify the user device that congestion is occurring on the line. They are the Forward Explicit Congestion Notification (FECN) bit and the Backward Explicit Congestion Notification (BECN) bit. The FECN is changed to 1 as a frame is sent downstream toward the destination location when congestion occurs during data transmission. In this way, all downstream nodes and the attached user device learn about congestion on the line. The BECN is changed to 1 in a frame traveling back toward the source of data transmission on a path where congestion is occurring. Thus the source node is notified to slow down transmission until the congestion subsides.

Now that you have looked at the Frame Relay header, refer back to Table 7.4 and look at the last two fields contained in the Frame Relay packet.

User Data contains the upper-layer data encapsulated in the Frame Relay packet. This field can vary in length.

FCS is used, upon receipt of the packet, to check the data for any errors that may have occurred during transmission. The value is computed by the transmitting station before transmission. The receiving station will then do the same computation and verify the value.

Committed Information Rate (CIR)

Committed Information Rate (CIR) is the minimum bandwidth consumed by the user at all times. CIR is usually less than the physical interface speed. The user could have a T-1 port, with a CIR 256K. The user can have data traffic bursting up to T-1, but guaranteed 256K all the time. A Frame Relay network keeps track of the number of packets for a delta time. When the data rate exceeds CIR in the delta period, the Frame Relay network sets the rest of the packets with DE (Discard Eligibility) bit, until the delta expires. If the network is congested, it will start dropping the packets with DE bits, otherwise they will pass through the network.

CIR is needed to guarantee certain bandwidth for normal data transmission needs. Certain applications like file services, application services at a central location, or workstations at a branch location, need to communicate continuously to maintain network drive mappings and application database connections. These applications need certain guaranteed bandwidth. The provisioning of a Frame Relay circuit with CIR guarantees bandwidth needed for standard applications. Provisioning different PVCs with a different CIR is possible and recommended. For example, Central office connects to two branch offices. Branch1 has 100 users, and

Branch2 has 10 users. Central office can connect to Frame Relay T-1, with PVC1 to Branch1 at 512K CIR, and PVC2 to Branch2 with 64K CIR.

NOTE

CIR rates can be set to minimize the cost of a Frame Relay circuit. The lower the CIR, the lower the cost you'll be able to receive from the Frame Relay provider.

Local Management Interface (LMI)

Local Management Interface (LMI) is a signaling (polling) protocol between a service provider network and an end-user device. Poll and acknowledgement (status) messages are exchanged between the user and network at regular intervals (similar to keep-alives on an Ethernet network.). In addition to the polling mechanism, which verifies connectivity, LMI is responsible for providing the end station with its local DLCI address, and keeping an eye on the status of the assigned DLCIs.

LMI Type

When configuring a router supporting Frame Relay, it is very important that the LMI type is correct—if it is incorrect, the Frame Relay circuit will not function properly. LMI signaling comes in three options:

ANSI Annex D defined by American National Standards Institute (ANSI) standard T1.617. ANSI uses DLCI 0 to pass status information between the service provider's Frame Relay switch and the connected router.

Cisco An LMI type developed jointly by StrataCom, Northern Telecom, DEC, and Cisco. This LMI type uses DLCI 1023 to pass status information between the service provider's Frame Relay switch and the connected router.

Q933a ITU-T Q.933 Annex A. This LMI type also uses DLCI 0 to pass status information between the service provider's switch and the connected router. In addition, Q933a provides CIR information for each configured PVC.

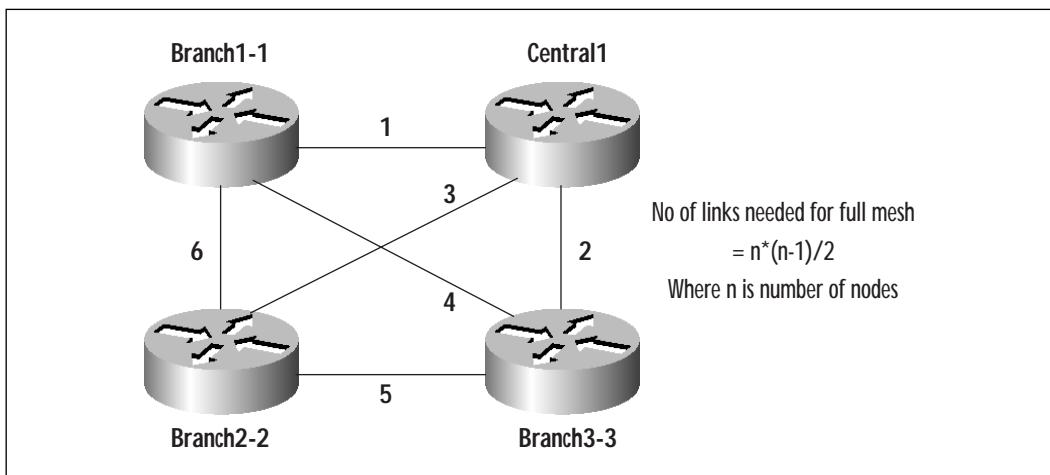
All three LMI types accomplish the same thing—they each just do it a little differently. The important point here is to make sure that you find out from your provider what your LMI should be set to on your router. Remember, if the LMI type between communicating devices is different, the virtual circuit will not establish, DLCIs will not be assigned, and communication over the link will not be possible.

Frame Relay Topologies

Frame Relay technology provides various mechanisms to connect many remote sites efficiently and economically. When every remote site has a direct connection to every other site, it is called a *fully meshed* network. This type of topology provides connectivity to every site, but it is rarely cost-justified and tends to be a lot harder to support.

Figure 7.8 illustrates four remote sites connected together in a fully meshed network. Six network connections are needed to make it fully meshed. As the number of remote sites increase, the number of Frame Relay circuits increases exponentially; thus your monthly charge for Frame Relay will increase.

Figure 7.8 A fully meshed network.



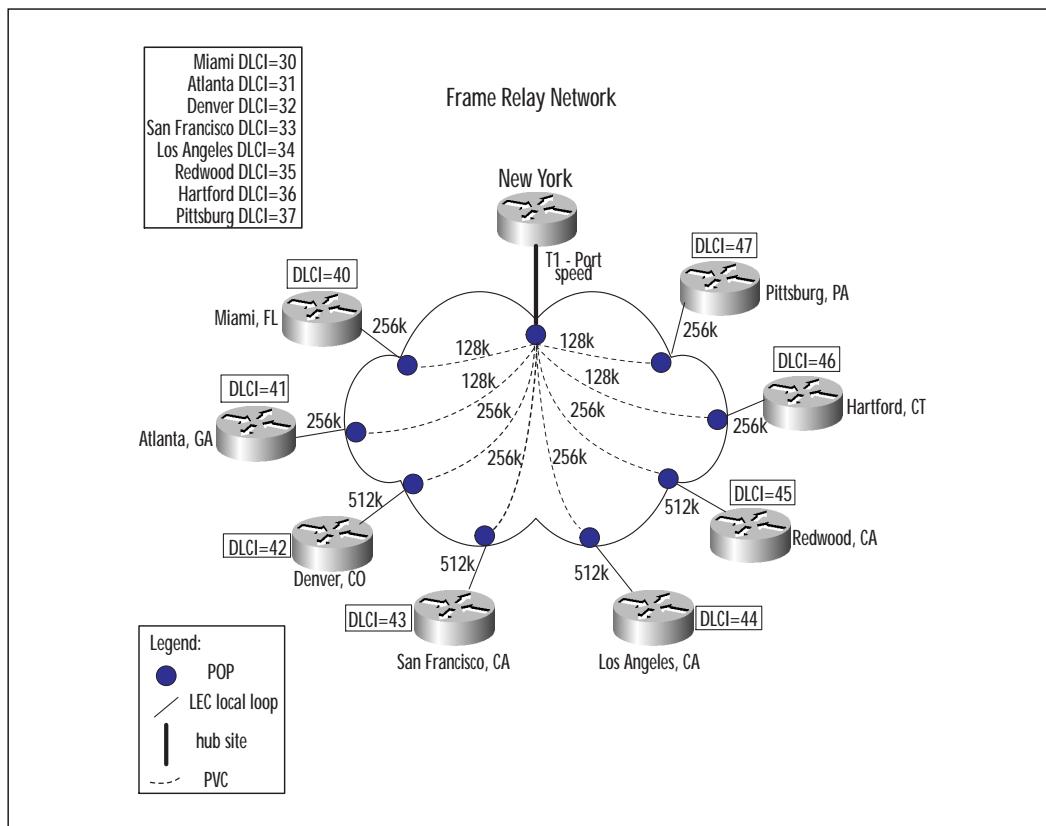
An alternative approach to the design is to implement a *partially meshed* network. A partially meshed network is also called a *hub-and-spoke network*. This kind of topology can be connected with **n-1** connections. All the traffic comes to one central location and then is re-routed back to the appropriate branch location.

Hub-and-spoke designs are more efficient because the full connectivity can be achieved through a minimum number of connections. Hubs can be headquarters and spokes can be branch offices; Figure 7.9 shows New York as Headquarters with branches in various cities. New York connects through a physical T-1 to a Frame Relay cloud. The branch sites connect at 256K-port speed, with total subscription of two T-1s. Over-subscription by 100 percent is recommended, with CIR matching the hub port speed.

Every branch is guaranteed committed information rate (CIR), with a burst of up to physical port speed.

If remote site connectivity (or redundancy) is an issue, a more cost effective method (instead of a fully meshed infrastructure) is to implement ISDN dial backup. Your remote-site router will have an ISDN interface that sits dormant until the Frame Relay circuit to the hub site goes down. Once the Frame Relay circuit is down, the ISDN circuit will activate and dial into the hub site. When the Frame Relay circuit comes back up, your ISDN will disconnect, giving you a stable infrastructure back to the hub site.

Figure 7.9 Hub-and-spoke design for a Frame Relay network.



For Managers

Creating a Frame Relay Spreadsheet

A spreadsheet with all the remote sites, port speeds, and CIR requirements, as shown in Table 7.7, is helpful in negotiating the rates with Frame Relay providers. This spreadsheet also provides information for determining if the hub site is oversubscribed.

Table 7.7 Frame Relay Provisioning

Source site	Source DLCI	Destination site	Destination DLCI	Port Speed	CIR/PVC
Miami, FL	40	New York	30	256K	128K
Atlanta, GA	41	New York	31	256K	128K
Denver, CO	42	New York	32	512K	256K
San Francisco, CA	43	New York	33	512K	256K
Los Angeles, CA	44	New York	34	512K	256K
Redwood, CA	45	New York	35	512K	256K
Hartford, CT	46	New York	36	256K	128K
Pittsburgh, PA	47	New York	37	256K	128K
New York, NY (Hub Site)	Multiple	Multiple	Multiple	T1	T1

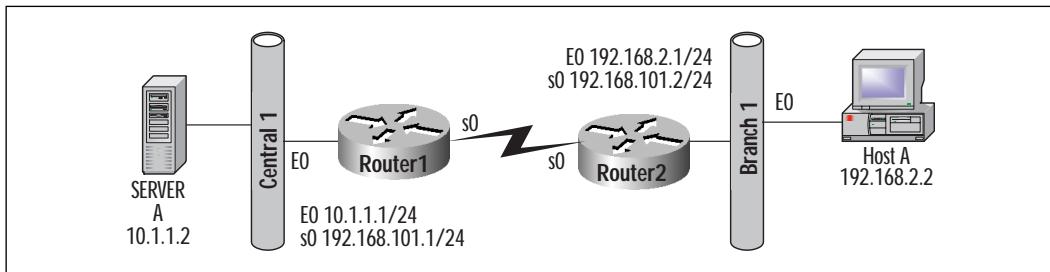
Split Horizon and Poison Reverse

Split horizon and poison reverse are features designed to prevent routing loops.

Routing loops occur when a route becomes unusable due to failure of a router or a network. In principle, the adjacent routers detect failures; they then send routing updates that show the old route as unusable. However, it is possible for updates not to reach some parts of the network at all, or to be delayed in reaching certain routers. A router that still believes the old route is good can continue spreading that information, thus reentering the failed route into the system. Eventually this information will propagate through the network and come back to the router that re-injected it. The result is a circular route.

The split horizon rule is based on the concept that it never makes sense to send a route back in the direction from which it came. Consider the example in Figure 7.10. Router1 will tell Router2 that it has a route to network 10.1.1.0. When Router2 sends updates to Router1, there is no reason for it to mention network 10.1.1.0. Since Router1 is closer to 10.1.1.0, there is no reason for it to consider going via Router2. The split horizon rule says a separate update message should be generated for each neighboring network. The update for a given neighbor should omit routes that point to that neighbor. This rule prevents loops between adjacent routers. For example: suppose Router1's interface to network 10.1.1.0 fails. Without the split horizon rule, Router2 would be telling Router1 that it can get to 10.1.1.0 via its serial 0 interface. Since it no longer has a real route, Router1 might choose that route. In that case, Router1 and Router2 would both have routes to 10.1.1.0. But Router1 would point to Router2 and Router2 would point to Router1. Since there is no reason to send information back to the place it came from, split horizon will help preventing loops. In addition to its role to prevent loops, split horizon keeps down the size of update messages.

Figure 7.10 Split Horizon Example.



The poison reverse rule is intended to break larger loops. The rule simply states that updates for a route will be sent out the same interface on which the route was learned; however, the metric for the route will be set to infinity (destination unreachable).

The Routing Table Manager (RTM) within Cisco IOS code monitors outgoing updates for each interface and removes any updates that were learned through that interface (split horizon). It also monitors incoming routing updates and their metrics, and then applies poison-reverse related controls. These two are turned on by default. Split horizon can be disabled on a per-interface basis:

```
Router1 #
  Interface s0
    No ip split-horizon
```

Subinterfaces

Subinterfaces are logical interfaces within a physical interface. Sub-interfaces are ideal for mapping PVCs in Frame Relay, and VCs in ATM. With their help, you can convert nonbroadcast multiaccess (NBMA) networks like Frame Relay into point-to-point networks. With subinterfaces, the split horizon issue in Frame Relay networks is resolved.

A single, physical interface can be logically divided into multiple, virtual subinterfaces. The subinterface may be defined as either a *point-to-point* or *multipoint* connection. A point-to-point subinterface would provide all the advantages of direct point-to-point links. Point-to-point links provide complete control over the traffic, like filtering through access-list and broadcast control.

Multipoint subinterfaces (see Figure 7.11) provide nonbroadcast multi-access (NBMA). In multipoint situations all interfaces will be part of a single subnet. Pinging your own IP address on a multipoint Frame Relay interface does not work, because Frame Relay multipoint subinterfaces are non-broadcast (unlike Frame Relay point-to-point sub-interfaces).

Figure 7.11 Frame Relay Multipoint using subinterfaces.

Central1 Router Configuration

```
Central1#
!
interface Serial0
  ip address 192.168.101.1 255.255.255.0
  encapsulation frame-relay
  frame-relay lmi-type cisco
  frame-relay map 192.168.101.2 32 !Maps PVC to Branch1-1
  frame-relay map 192.168.101.3 33 !Maps PVC to Branch2-2
```

Branch1-1 Router Configuration

```
Branch1-1#
!
!
```

Continued

Figure 7.11 Continued.

```
interface Serial0
    no ip address           ! the ip address is supplied on the
    subinterface
    encapsulation frame-relay
    frame-relay lmi-type cisco
!
interface Serial0.1 multipoint
    ip address 192.168.101.2 255.255.255.0 !Notice the ip addresses for
    main, branch1-1, branch2-2 are in the same subnet
    frame-relay map 192.168.101.1 30 ! Remote ip address maps to local
    DLCI

Branch2-2 Router Configuration
Branch2-2#
!
interface Serial0
    no ip address
    encapsulation frame-relay
    frame-relay lmi-type cisco
!
interface Serial0.1 multipoint
    ip address 192.168.101.3 255.255.255.0
    frame-relay map 192.168.101.2 31
end
```

NOTE

The number of subinterfaces on a given router is limited to 230. The number of DLCIs is limited to maximum of 796. The Cisco 2500 series router can have 60 DLCIs and Cisco 7500 series provides a maximum of up to 720 DLCIs.

Configuring Frame Relay

Cisco routers can be configured as a Frame Relay switch (carrier side) or Frame Relay Customer Premise Equipment (CPE). Usually the only reason a Cisco router is configured as a Frame Relay switch is for lab and/or testing purposes. In carrier networks, Frame Relay switches like Cisco (Stratacom) or Lucent (Ascend) switches are used. A router as DTE connecting to the Frame Relay cloud is a more popular scenario.

The following is an example of Frame Relay hub-and-spoke configuration. (These configurations are based on Figure 7.9 and Table 7.7.)

Hub Sites Configuration

```
NewYork>
```

```
interface Serial0
  description Hub site Frame Relay T-1 circuit# 123456
  no ip address
  encapsulation frame-relay
  frame-relay lmi-type ansi
!
interface Serial0.30 point-to-point
  description 128k PVC to Miami, Fl - circuit 30
  ip address 192.168.30.1 255.255.255.0
  frame-relay interface-dlci 30
!
interface Serial0.31 point-to-point
  description 128k PVC to Atlanta - circuit 31
  ip address 192.168.31.1 255.255.255.0
  frame-relay interface-dlci 31
!
interface Serial0.32 point-to-point
  description 256k PVC to Denver ,CO- circuit 32
  ip address 192.168.32.1 255.255.255.0
  frame-relay interface-dlci 32
!
interface Serial0.33 point-to-point
```

```
description 256k PVC to San Francisco, CA - circuit 33
ip address 192.168.33.1 255.255.255.0
frame-relay interface-dlci 33
!
interface Serial0.34 point-to-point
description 256k PVC to Los Angels, CA - circuit 34
ip address 192.168.34.1 255.255.255.0
frame-relay interface-dlci 34
!
interface Serial0.35 point-to-point
description 256k PVC to Redwood CA - circuit 35
ip address 192.168.35.1 255.255.255.0
frame-relay interface-dlci 35
!
interface Serial0.36 point-to-point
description 128k PVC to Hartford , CN - circuit 36
ip address 192.168.36.1 255.255.255.0
frame-relay interface-dlci 36
!
interface Serial0.37 point-to-point
description 128k PVC to Pittsburgh, PA - circuit 37
ip address 192.168.37.1 255.255.255.0
frame-relay interface-dlci 37
!
```

Remote Sites Configuration

```
Miami>
!
interface Serial0
no ip address
encapsulation frame-relay
frame-relay lmi-type ansi
!
```

```
interface Serial0.40 point-to-point
description 128k PVC to New York From Miami, Fl - circuit 40
ip address 192.168.30.2 255.255.255.0
frame-relay interface-dlci 40
!
Atlanta>
!
interface Serial0
no ip address
encapsulation frame-relay
frame-relay lmi-type ansi
!
interface Serial0.41 point-to-point
description 128k PVC to New York From Atlanta - circuit 41
ip address 192.168.31.2 255.255.255.0
frame-relay interface-dlci 41
!
Denver>
interface Serial0
no ip address
encapsulation frame-relay
frame-relay lmi-type ansi
!
interface Serial0.42 point-to-point
description 256k PVC to New York From Denver ,CO- circuit 42
ip address 192.168.32.2 255.255.255.0
frame-relay interface-dlci 42
!
SanFrancisco>
!
interface Serial0
no ip address
encapsulation frame-relay
frame-relay lmi-type ansi
```

```
!  
interface Serial0.43 point-to-point  
description 256k PVC to New York From San Francisco, CA - circuit 43  
ip address 192.168.33.2 255.255.255.0  
frame-relay interface-dlci 43  
!  
LA>  
!  
interface Serial0  
no ip address  
encapsulation frame-relay  
frame-relay lmi-type ansi  
!  
interface Serial0.44 point-to-point  
description 256k PVC to New York From Los Angels, CA - circuit 44  
ip address 192.168.34.2 255.255.255.0  
frame-relay interface-dlci 44  
!  
REDWOOD>  
!  
interface Serial0  
no ip address  
encapsulation frame-relay  
frame-relay lmi-type ansi  
!  
interface Serial0.45 point-to-point  
description 256k PVC to New York From Redwood CA - circuit 45  
ip address 192.168.35.2 255.255.255.0  
frame-relay interface-dlci 45  
!  
HARTFORD>  
!  
interface Serial0  
no ip address
```

```
encapsulation frame-relay
frame-relay lmi-type ansi
!
interface Serial0.46 point-to-point
description 128k PVC to New York From Hartford , CN - circuit 46
ip address 192.168.36.2 255.255.255.0
frame-relay interface-dlci 46
!
PITTSBURGH>
!
interface Serial0
no ip address
encapsulation frame-relay
frame-relay lmi-type ansi
!
interface Serial0.47 point-to-point
description 128k PVC to New York from Pittsburg, PA - circuit 47
ip address 192.168.37.2 255.255.255.0
frame-relay interface-dlci 47
```

Verifying and Troubleshooting Frame Relay

Troubleshooting begins at the physical layer and then moves up to the network layer.

- Layer 1 (physical layer)
- Layer 2 (data-link layer, circuit level)
- Layer 3 (network layer)

Physical Layer Troubleshooting

If you see the serial protocol up, line protocol up, then the interface is physically up and running. You may not have any physical level problems. If you see serial protocol up, line protocol down, then the interface is up from a software configuration point of view, but no physical signal connectivity is established. At this point, look at the leads like CTS and RTS to see if they are up.

Use the **show interface serial 0** command to see the interface statistics for serial 0. The first line in the command output will indicate whether the interface and protocol are up or down. (In the following output example, both serial 0 and line protocol are up. This would indicate that there are no problems with the circuit.)

```
Show interface serial 0

Serial0 is up, line protocol is up
Hardware is PowerQUICC Serial
Description: Frame Relay circuit 12345
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation FRAME-RELAY, loopback not set ! Makesure the
encapsulation is frame relay.

Keepalive set (10 sec)

LMI enq sent 328460, LMI stat recv 328460, LMI upd recv 0, DTE
LMI up - LMI should be up

LMI enq recv 0, LMI stat sent 0, LMI upd sent 0
LMI DLCI 0 LMI type is ANSI Annex D frame relay DTE ! Compare with
remote LMI type , both ! should be same

Broadcast queue 0/64, broadcasts sent/dropped 87762/0, interface
broadcasts 37

378

Last input 00:00:03, output 00:00:08, output hang never
Last clearing of "show interface" counters 5w3d
Queueing strategy: fifo
Output queue 0/40, 0 drops; input queue 0/75, 0 drops
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
392750 packets input, 24814146 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
10 input errors, 3 CRC, 3 frame, 0 overrun, 0 ignored, 4 abort
429748 packets output, 29450130 bytes, 0 underruns
0 output errors, 0 collisions, 1 interface resets !crc errors and
interface resets indicate there is      !an      issue with phyical line.
0 output buffer failures, 0 output buffers swapped out
```

```
0 carrier transitions
DCD=up  DSR=up  DTR=up  RTS=up  CTS=up
```

Also, verify that the cable is physically secure and connected. If this is a new installation, you might also want to verify that this is the correct cable. You can verify that by entering the following command:

```
Show Frame PVC
```

This command provides PVC status, network congestion details, etc. See Figure 7.12 for an example of **show frame pvc**.

Figure 7.12 The show frame pvc command.

```
Central1#
Show frame pvc
PVC Statistics for interface Serial0 (Frame Relay DTE)

DLCI = 30, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0

      input pkts 20          output pkts 12376          in bytes 28400
      out bytes 17462536      dropped pkts 0          in FECN pkts 0
      in BECN pkts 0          out FECN pkts 0          out BECN pkts 0
      in DE pkts 0           out DE pkts 0
      pvc create time 5:22:21  last time pvc status changed 5:20:20

DLCI = 31, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE = Serial0

      input pkts 30          output pkts 250          in bytes 42600
      out bytes 355000         dropped pkts 0          in FECN pkts 0
      in BECN pkts 0          out FECN pkts 0          out BECN pkts 0
      in DE pkts 0           out DE pkts 0
      pvc create time 10:22:21  last time pvc status changed 10:20:20

show controller serial 0

Router# show control serial 0
Interface Serial0
```

Continued

Figure 7.12 Continued.

```
Hardware is PowerQUICC MPC860
DTE V.35 TX and RX clocks detected.
idb at 0x8087CD18, driver data structure at 0x80882C28
SCC Registers:
```

If you don't see DTE V.35, there is no cable connected between the router to the CSU/DSU, or to the Smart Jack. The Smart Jack is an RJ48 jack provided by the Telco and installed at the customer site. If you don't see the clock, you may have to provide the clock from an external source like the CSU/DSU. The clock is usually provided by the network (Carrier); it maintains the transmit and receive signal.

When you have verified all of the following and the problem still exists, verify the CRC counters, input errors, output errors, and carrier transitions.

!

```
10 input errors, 3 CRC, 3 frame, 0 overrun, 0 ignored, 4 abort
429748 packets output, 29450130 bytes, 0 underruns
0 output errors, 0 collisions, 1 interface resets !crc errors and
interface resets indicate there is      !an      issue with physical line.
```

You may have a faulty line if these counters are consistently incrementing. This type of situation will need to be corrected by the carrier.

Loopback Tests

You can perform loopback tests to verify Frame Relay connectivity at the physical layer. These tests will help you to isolate a problem with the Frame Relay circuit. You would typically run two types of loopback tests: local loopback and remote loopback.

Local Loopback

Local loopback will check the connection between the local CSU/DSU and the local router.

Setup the near end CSU/DSU in local loopback, and check to see if the line comes up. If it does not come up, the potential areas to look at are:

1. Faulty cable from router to CSU/DSU
2. Faulty CSU/DSU
3. Faulty router

Remote Loopback

Remote loopback will check the connection between the local CSU/DSU and the remote router (the router on the other end of the Frame Relay circuit). Configure the local CSU/DSU to provide remote loopback. Monitor the far end router.

If the line comes up, local router, CSU/DSU, serial circuit up to the remote CSU/DSU are functioning normally. In this situation one of these three could be faulty:

1. Remote CSU/DSU
2. Remote cable
3. Remote router

Frame Relay Problems

Once you have verified that the physical line is not causing the problem, the next step is to begin looking into the data-link layer (Layer 2) statistics.

The first item to verify in troubleshooting Frame Relay Layer 2 is whether the Frame Relay LMI type matches Frame Relay service provider settings. Remember if the LMI type differs between the two devices, communication will not take place. Using the **show frame-relay lmi** command, you should see what your LMI is set to and that status messages are being sent and received.

```
Show frame-relay lmi  
Central-1# show frame-relay lmi
```

```
LMI Statistics for interface Serial0 (Frame Relay DTE) LMI TYPE = ANSI  
Invalid Unnumbered info 0 Invalid Prot Disc 0  
Invalid dummy Call Ref 0 Invalid Msg Type 0  
Invalid Status Message 0 Invalid Lock Shift 0  
Invalid Information ID 0 Invalid Report IE Len 0  
Invalid Report Request 0 Invalid Keep IE Len 0  
Num Status Enq. Sent 328601 Num Status msgs Rcvd 328601  
Num Update Status Rcvd 0 Num Status Timeouts 0
```

The **debug frame relay lmi** Command

The debug command **central-1# debug frame-relay lmi** provides a variety of information, such as: Is the PVC active? Does the DLCI configured on the router match the DLCI broadcast by the carrier? Is the LMI type the same in the Frame Relay local switch and router?

Monitor the keep-alives on the debug output.

```
!
*Jun  9 18:18:18.819: KA IE 3, length 2, yourseq 121, myseq 123
*Jun  9 18:18:18.819: PVC IE 0x7 , length 0x3 , dlci 31, status 0x2
(indicates pvc status is active)
*Jun  9 18:18:18.819: KA IE 3, length 2, yourseq 122, myseq 124
```

Check to see if the PVC is active.

```
Central-1#sho frame pvc
```

```
PVC Statistics for interface Serial0 (Frame Relay DTE)
```

	Active	Inactive	Deleted	Static
Local	1	1	0	0
Switched	0	0	0	0
Unused	1	2	0	0

```
DLCI = 30, DLCI USAGE = LOCAL, PVC STATUS = INACTIVE,
INTERFACE = Serial0.30
```

input pkts 0	output pkts 0	in bytes 0
out bytes 0	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 0	out bcast bytes 0	
pvc create time 5w0d, last time pvc status changed 5w0d		

```
DLCI = 31, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE =
Serial0.31
```

input pkts 64061	output pkts 101345	in bytes 18983910
out bytes 24863788	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 64061	out DE pkts 0	
out bcast pkts 87814	out bcast bytes 19797798	
pvc create time 5w0d, last time pvc status changed 03:24:11		

The following URL at Cisco Online provides Frame Relay troubleshooting links:

www.cisco.com/univercd/cc/td/doc/cisintwk/itg_v1/tr1918.htm

You can use the following commands to further identify problems related to specific protocols like IP, Novell, Appletalk, and DECNET:

debug frame-relay shows the packets coming into the router

debug frame-relay packet shows the packets going out of the router

debug frame-relay lmi shows the lmi status packets

debug frame-relay events provides information about frame relay ARP replies

The **debug frame-relay** Command

The following debugging scenario shows Frame Relay packets received by the Frame Relay interface. The data shows what protocol type of packet was received, on what DLCI, and the length of packet.

```
Router1# debug frame-relay
Router1#
Router1#debug frame
Frame Relay debugging is on
Router1#
Serial2(i): dlci 102(0x1861), pkt type 0x800, datagramsize 96 !traffic
coming in on dlci 102, packet      type is ip
Serial2(i): dlci 100(0x1841), pkt type 0x800, datagramsize 116
Serial2.30: Broadcast on DLCI 100 link 65(CDP) !Cisco discovery
protocol packet recieived
Serial2.30(o): dlci 100(0x1841), pkt type 0x2000(CDP), datagramsize 282
Serial2.32: Broadcast on DLCI 102 link 65(CDP)
Serial2.32(o): dlci 102(0x1861), pkt type 0x2000(CDP), datagramsize 282
broadcast dequeue
Serial2.30(o):Pkt sent on dlci 100(0x1841), pkt type 0x2000(CDP) ,
datagramsize 282
```

The **debug frame-relay packet** Command

Debug frame-relay packet displays the packets being transmitted through the interface. The router is queuing Cisco Discovery Protocol (CDP) packets for broadcasting on the serial link. The output also shows IP packets (0x800) being transmitted.

```

Router1# debug frame-relay packet

Serial2.30(o):Pkt sent on dlci 100(0x1841), pkt type 0x2000(CDP),
datagramsize 282

broadcast dequeue

Serial2.32(o):Pkt sent on dlci 102(0x1861), pkt type 0x2000(CDP),
datagramsize 282

Serial2.30: broadcast search

Serial2.30(o): dlci 100(0x1841), pkt type 0x800(IP), datagramsize 96
broadcast dequeue

Serial2.30(o):Pkt sent on dlci 100(0x1841), pkt type 0x800(IP),
datagramsize 96

Serial2.32: broadcast search

Serial2.32(o): dlci 102(0x1861), pkt type 0x800(IP), datagramsize 96
broadcast dequeue

```

The debug frame-relay lmi Command

Debug frame-relay lmi displays the sending and receiving of LMI status messages. Notice the serial2(in) and serial2(out) statements at the beginning of each debug message. This indicates that you are successfully sending and receiving LMI status messages on interface serial2.

```

Router1# debug frame-relay LMI

Frame Relay LMI debugging is on

Displaying all Frame Relay LMI data

Serial2(out): StEnq, myseq 5, yourseen 4, DTE up →Myseq - provides the
router sequence number being sent out

datagramstart = 0x647D20, datagramsize = 14

FR encaps = 0x00010308

00 75 95 01 01 01 03 02 05 04

Serial2(in): Status, myseq 5

RT IE 1, length 1, type 1

KA IE 3, length 2, yourseq 5 , myseq 5

Serial2(out): StEnq, myseq 6, yourseen 5, DTE up →yourseen provides
the acknowledgment from remote router, which is normally - myseq -1 =
(6-1=5) under normal operation

datagramstart = 0x647D20, datagramsize = 14

FR encaps = 0x00010308 →Shows type of encapsulation being used

```

```
00 75 95 01 01 03 02 06 05
```

```
Serial2(in): Status, myseq 6
RT IE 1, length 1, type 1
KA IE 3, length 2, yourseq 6 , myseq 6
Serial2(out): StEnq, myseq 7, yourseen 6, DTE up
datagramstart = 0x647D20, datagramsize = 14
FR encaps = 0x00010308
00 75 95 01 01 00 03 02 07 06
```

```
Serial2(in): Status, myseq 7
RT IE 1, length 1, type 0
KA IE 3, length 2, yourseq 7 , myseq 7
PVC IE 0x7 , length 0x3 , dlci 100, status 0x2 →Shows PVC is active
PVC IE 0x7 , length 0x3 , dlci 102, status 0x2
Serial2(out): StEnq, myseq 8, yourseen 7, DTE up
datagramstart = 0x647D20, datagramsize = 14
FR encaps = 0x00010308
00 75 95 01 01 01 03 02 08 07
Router1#undebug all
All possible debugging has been turned off
```

Frame Relay Traffic Shaping (FRTS)

Frame Relay traffic shaping is a way of controlling traffic in a Frame Relay network. It is necessary because Frame Relay allows oversubscription of circuits above the CIR. Any traffic above the CIR can be discarded if the Frame Relay network is congested. The benefits are that the end-user device can transmit data up to the port speed of the physical port and speed is reduced only when there is congestion in the network.

Frame Relay uses various methods in controlling the traffic:

Discard Eligibility bit (DE) lets you control which packets to discard during congestion.

FECN Forward Explicit Control Notification.

BECN Backward Explicit Control Notification.

DLCI priority levels DLCI priority levels provide a way to define multiple parallel DLCIs for different types of traffic.

(FECN, BECN, and Discard Eligibility were discussed earlier in the chapter.) The Cisco IOS provide some generic traffic control mechanisms that can be used in Frame Relay traffic shaping:

Default Queuing (FIFO) Cisco uses First In, First Out queuing by default. If no special configuration is done on the serial interface of a router, it uses FIFO.

Custom Queuing Custom queuing reserves a percentage of an interface's available bandwidth for each selected traffic type. If a particular type of traffic is not using the bandwidth reserved for it, then other traffic types may use the remaining reserved bandwidth.

Priority Queuing Priority queuing provides priority to important traffic. Priority queuing can flexibly prioritize according to network protocol (such as IP or DECnet), incoming interface, packet size, source/destination address, etc.

FRTS provides dynamic traffic control through BECN, FECN on a per-VC basis. When a BECN is received in a packet, the outbound traffic is automatically reduced by the transmitting router. When the congestion recedes, and there are no BECN indicators arriving, the router will increase the outgoing traffic to its normal speeds permitted for that interface.

Enable Frame Relay Traffic Shaping (FRTS) on the Interface

Enabling FRTS on an interface enables both traffic shaping and per-VC queuing on all the interface's PVCs and SVCs. Traffic shaping enables the router to control the circuit's output rate and react to congestion notification information.

Configuring Traffic Shaping

Enabling Frame Relay traffic shaping on an interface requires a two-step process. First, you must enable FRTS on a specific interface. To do this, use the following command in interface configuration mode:

```
Central(config-if)#>frame-relay traffic-shaping
```

Second, you will need to define a map class on the router, and assign that map class to the traffic-shaping interface. The map class will define the various settings that will control how traffic travels over the Frame Relay link. The following is an example of some of the commands that you can use to define the map class:

```
Central (config)#> map-class frame-relay test (test is the name of map class)
```

```

Central (config-map-class)#> frame-relay adaptive-shaping becn (Enables
bebn for traffic shaping)

Central (config-map-class)#> frame-relay cir 56000 (Sets the cir value
for traffic shaping)

Central (config-map-class)#> frame-relay mincir 1500 (Sets the minimum
cir value for traffic shaping)

Central (config-map-class)#> frame-relay bc 1100 (Defines the committed
burst size. Should match the providers setting to prevent the
discarding of packets with DE bit set)

Central (config-map-class)#> frame-relay be 2000 (Defines the excess
burst size. Should also match the providers setting to prevent the
discarding of packets with DE bit set)

```

Once the map class is defined, it can be assigned to the interface by entering the following command in interface configuration mode:

```
Central(config-if)#> frame-relay class test (Test is the name of the map
class we defined)
```

Figure 7.13 illustrates a simple implementation of Frame Relay traffic shaping. In the example, traffic shaping is enabled in router 1's configuration. Check out the map class statement and how the map class is assigned to interface serial 2.1.

Figure 7.13 Frame Relay Traffic Shaping.

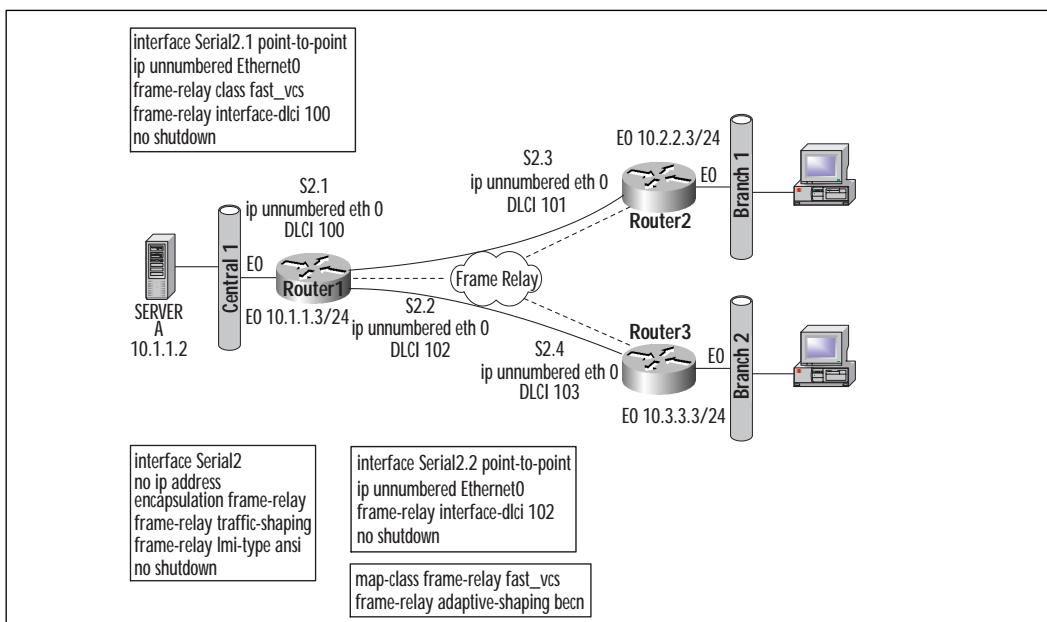


Figure 7.14, 7.15, and 7.16 provide the configurations for each of the routers represented in Figure 7.13.

Figure 7.14 Router1 configuration.

```
Router1
!
version 11.3
!
hostname Router_1
!
interface Ethernet0
ip address 10.1.1.3 255.255.255.0
no ip route-cache
no ip mroute-cache
!
interface Serial2
no ip address
encapsulation frame-relay
no ip route-cache
no ip mroute-cache
frame-relay traffic-shaping - Applies traffic shaping to the interface
frame-relay lmi-type ansi
no shutdown
!
interface Serial2.1 point-to-point
description frame relay to router b
ip unnumbered Ethernet0
no ip route-cache
frame-relay class fast_vcs - Traffic shaping applied to this pvc
frame-relay interface-dlci 100
no shutdown
!
interface Serial2.2 point-to-point
description frame relay to router c
ip unnumbered Ethernet0
```

Continued

Figure 7.14 Continued.

```
no ip route-cache
frame-relay interface-dlci 102
no shutdown
!
interface BRI0
  no ip address
  no ip route-cache
  no ip mroute-cache
  shutdown
!
router eigrp 100
  network 10.0.0.0
!
ip classless
!
map-class frame-relay fast_vcs
  frame-relay adaptive-shaping becn - Traffic shaping parameter used is
  BECN
!
banner motd ^C
Establish a Frame Relay PVCs on three routers and control
traffic flow. - Router_1
^C
!
line con 0
  exec-timeout 0 0
  password xxxx
  login
line aux 0
  password xxxx
  login
  transport input all
line vty 0 4
```

Continued

Figure 7.14 Continued.

```
password xxxx
login
!
end
```

Figure 7.15 Router2 configuration.

```
Router2
!
version 11.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service udp-small-servers
service tcp-small-servers
!
hostname Router_2
!
enable password xxxx
!
!
interface Ethernet0
ip address 10.2.2.3 255.255.255.0
no ip route-cache
no ip mroute-cache
!
interface Serial2
no ip address
encapsulation frame-relay
no ip route-cache
no ip mroute-cache
frame-relay lmi-type ansi
no shutdown
```

Continued

Figure 7.15 Continued.

```
!
interface Serial2.3 point-to-point
description frame relay to router 1
ip unnumbered Ethernet0
no ip route-cache
frame-relay interface-dlci 101
no shutdown
!
interface Serial3
no ip address
no ip route-cache
no ip mroute-cache
shutdown
!
interface BRI0
no ip address
no ip route-cache
no ip mroute-cache
shutdown
!
router eigrp 100
network 10.0.0.0
!
ip classless
!
!
banner motd ^C
Establish a Frame Relay PVCs on three routers and control
traffic flow. - Router_2
^C
!
line con 0
exec-timeout 0 0
```

Continued

Figure 7.15 Continued.

```
password xxxx
login
line aux 0
password xxxx
login
transport input all
line vty 0 4
password xxxx
login
!
end
```

Figure 7.16 Router3 configuration.

```
Router3
!
version 11.3
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
service udp-small-servers
service tcp-small-servers
!
hostname Router_3
!
enable password xxxx
!
!
interface Ethernet0
ip address 10.3.3.3 255.255.255.0
no ip route-cache
no ip mroute-cache
!
```

Continued

Figure 7.16 Continued.

```
interface Serial0
    no ip address
    no ip route-cache
    no ip mroute-cache
    shutdown
!
interface Serial2.4 point-to-point
    description frame relay to router a
    ip unnumbered Ethernet0
    no ip route-cache
    frame-relay interface-dlci 103
    no shutdown
!
router eigrp 100
    network 10.0.0.0
!
ip classless
!
!
banner motd ^C
Establish a Frame Relay PVCs on three routers and control
traffic flow. - Router_3
^C
!
line con 0
exec-timeout 0 0
password xxxx
login
line aux 0
password xxxx
login
transport input all
line vty 0 4
```

Continued

Figure 7.16 Continued.

```
password xxxx
login
!
end
```

Verifying Traffic Shaping

The functioning of traffic shaping configurations can be monitored through various show and debug commands. These are:

- show frame-relay pvc
- show frame-relay lmi
- show interface
- show ip route
- show traffic shap
- show frame-relay map
- debug frame-relay lmi

Let's look at the related output that each of these commands produces.

```
Router1#show frame-relay pvc
```

```
PVC Statistics for interface Serial2 (Frame Relay DTE)
```

```
DLCI = 100, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE =
Serial2.1
```

```
      input pkts 21          output pkts 24          in bytes 2014
      out bytes 2066         dropped pkts 0          in FECN pkts 0
      in BECN pkts 0         out FECN pkts 0          out BECN pkts 0
→ shows BECN packets count
      in DE pkts 0          out DE pkts 0
      out bcast pkts 22       out bcast bytes 1838
      pvc create time 00:12:17, last time pvc status changed 00:01:19
DLCI = 102, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE =
Serial2.2
```

```
      input pkts 11          output pkts 15          in bytes 804
      out bytes 1750         dropped pkts 0        in FECN pkts 0
      in BECN pkts 0         out FECN pkts 0        out BECN pkts 0
      in DE pkts 0          out DE pkts 0
      out bcast pkts 12     out bcast bytes 1198
      pvc create time 00:11:03, last time pvc status changed 00:00:40
```

```
Router1#show frame-relay traffic
```

```
Frame Relay statistics:
```

```
    ARP requests sent 0, ARP replies sent 0
    ARP request recv 0, ARP replies recv 0
```

```
Router1#sh frame lmi
```

```
LMI Statistics for interface Serial2 (Frame Relay DTE) LMI TYPE = ANSI
      Invalid Unnumbered info 0          Invalid Prot Disc 0
      Invalid dummy Call Ref 0         Invalid Msg Type 0
      Invalid Status Message 0        Invalid Lock Shift 0
      Invalid Information ID 0       Invalid Report IE Len 0
      Invalid Report Request 0       Invalid Keep IE Len 0
      Num Status Enq. Sent 14        Num Status msgs Rcvd 14
      Num Update Status Rcvd 0       Num Status Timeouts 0
```

```
Router1#show interfaces s2
```

```
Serial2 is up, line protocol is up
  Hardware is CD2430 in sync mode
  MTU 1500 bytes, BW 115 Kbit, DLY 20000 usec, rely 255/255, load 1/255
  Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)
  LMI enq sent 15, LMI stat recv 15, LMI upd recv 0, DTE LMI up
  LMI enq recv 0, LMI stat sent 0, LMI upd sent 0
  LMI DLCI 0  LMI type is ANSI Annex D  frame relay DTE
  FR SVC disabled, LAPF state down
  Broadcast queue 0/64, broadcasts sent/dropped 52/0, interface
  broadcasts 46
  Last input 00:00:00, output 00:00:03, output hang never
```

```
Last clearing of "show interface" counters never
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/64/0 (size/max total/threshold/drops)
    Conversations 0/1/256 (active/max active/max total)
    Reserved Conversations 0/0 (allocated/max allocated)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    66 packets input, 4724 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    75 packets output, 5758 bytes, 0 underruns
    0 output errors, 0 collisions, 5 interface resets
0 output buffer failures, 0 output buffers swapped out
    2 carrier transitions
    DCD=up  DSR=up  DTR=up  RTS=up  CTS=up
```

Router1#**sh ip route**

```
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B -
BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * -
candidate default
U - per-user static route, o - ODR
```

Gateway of last resort is not set

10.0.0.0/24 is subnetted, 5 subnets

```
D      10.2.2.0 [90/2195456] via 10.140.1.1, 00:03:45, Serial1
D      10.3.3.0 [90/2195456] via 10.140.2.2, 00:03:46, Serial0
C      10.1.1.0 is directly connected, Ethernet0
C      10.140.2.0 is directly connected, Serial0
C      10.140.1.0 is directly connected, Serial1
```

```
Router1#sh frame-relay traffic  
Frame Relay statistics:  
    ARP requests sent 0, ARP replies sent 0  
    ARP request recv 0, ARP replies recv 0
```

```
Router1# show traffic-shape → shows traffic shaping related statistics  
          Access Target      Byte   Sustain   Excess   Interval  
Increment Adapt  
I/F       List     Rate      Limit   bits/int   bits/int   (ms)  
(bytes)   Active  
Se2.1           56000     875      7000      0          125  
875       BECN → shows BECN is active  
Se2.2           56000    7875     56000     56000      125  
875       BECN  
=====  
    Num Update Status Rcvd 0      Num Status Timeouts 0  
=====  
Router1#show frame-relay pvc
```

PVC Statistics for interface Serial2 (Frame Relay DTE)

DLCI = 100, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE =
Serial2.1

input pkts 215	output pkts 217	in bytes 17440
out bytes 17428	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 215	out bcast bytes 17200	

Shaping adapts to BECN → shows what type of traffic shaping used
pvc create time 00:26:06, last time pvc status changed 00:15:07

DLCI = 102, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE =
Serial2.2

```

input pkts 205          output pkts 209          in bytes 16230
out bytes 17176         dropped pkts 0          in FECN pkts 0
in BECN pkts 0          out FECN pkts 0          out BECN pkts 0
in DE pkts 0            out DE pkts 0
out bcast pkts 206     out bcast bytes 16624
pvc create time 00:24:51, last time pvc status changed 00:14:28
=====

```

```

Router1#debug frame-relay lmi
Frame Relay LMI debugging is on
Displaying all Frame Relay LMI data

```

```

05:37:40: Serial2(out): StEnq, myseq 108, yourseen 107, DTE up
05:37:40: datagramstart = 0x647D20, datagramsize = 14
05:37:40: FR encapsulation = 0x00010308
05:37:40: 00 75 95 01 01 01 03 02 6C 6B
05:37:40:
05:37:40: Serial2(in): Status, myseq 108
05:37:40: RT IE 1, length 1, type 1
05:37:40: KA IE 3, length 2, yourseq 108, myseq 108
05:37:51: Serial2(out): StEnq, myseq 109, yourseen 108, DTE up
05:37:51: datagramstart = 0x647D20, datagramsize = 14
05:37:51: FR encapsulation = 0x00010308
05:37:51: 00 75 95 01 01 01 03 02 6D 6C
05:37:51:
05:37:51: Serial2(in): Status, myseq 109
05:37:51: RT IE 1, length 1, type 1
05:37:51: KA IE 3, length 2, yourseq 109, myseq 109 frame
=====
Router1#debug frame-relay
05:38:00: Serial2(out): StEnq, myseq 110, yourseen 109, DTE up
05:38:00: datagramstart = 0x647D20, datagramsize = 14
05:38:00: FR encapsulation = 0x00010308

```

```
05:38:00: 00 75 95 01 01 00 03 02 6E 6D
05:38:00:
05:38:00: Serial2(in): Status, myseq 110
05:38:00: RT IE 1, length 1, type 0
05:38:00: KA IE 3, length 2, yourseq 110, myseq 110
05:38:00: PVC IE 0x7 , length 0x3 , dlci 100, status 0x2
05:38:00: PVC IE 0x7 , length 0x3 , dlci 102, status 0x2 ip
05:38:10: Serial2(out): StEnq, myseq 111, yourseen 110, DTE up
05:38:10: datagramstart = 0x647D20, datagramsize = 14
05:38:10: FR encap = 0x00010308
05:38:10: 00 75 95 01 01 01 03 02 6F 6E
05:38:42: Serial2.2(o):Pkt sent on dlci 102(0x1861), pkt type
0x800(IP), datagramsize 64
05:38:43: Serial2.1: broadcast search
05:38:43: Serial2.1(o): dlci 100(0x1841), pkt type 0x800(IP),
datagramsize 64
05:38:43: broadcast dequeue
05:38:43: Serial2.1(o):Pkt sent on dlci 100(0x1841), pkt type
0x800(IP), datagramsize 64
05:38:46: Serial2(i): dlci 100(0x1841), pkt type 0x800, datagramsize 64
05:38:46: Serial2.1: Broadcast on DLCI 100 link 65(CDP)
05:38:46: Serial2.1(o): dlci 100(0x1841), pkt type 0x2000(CDP),
datagramsize 279
05:38:46: broadcast dequeue
05:38:46: Serial2.1(o):Pkt sent on dlci 100(0x1841), pkt type
0x2000(CDP), datagramsize 279
05:38:46: Serial2(i): dlci 102(0x1861), pkt type 0x800, datagramsize 64
05:38:46: Serial2.2: broadcast search
05:38:46: Serial2.2(o): dlci 102(0x1861), pkt type 0x800(IP),
datagramsize 64
05:38:46: broadcast dequeue
05:38:46: Serial2.2(o):Pkt sent on dlci 102(0x1861), pkt type
0x800(IP), datagramsize 64
05:38:47: Serial2.1: broadcast search
05:38:47: Serial2.1(o): dlci 100(0x1841), pkt type 0x800(IP),
datagramsize 64
```

```
05:38:47: Serial2(i): dlci 100(0x1841), pkt type 0x2000, datagramsize  
279  
05:38:47: broadcast dequeue  
05:38:47: Serial2.1(o):Pkt sent on dlci 100(0x1841), pkt type  
0x800(IP), datagramsize 64  
05:38:48: Serial2(i): dlci 102(0x1861), pkt type 0x2000, datagramsize  
279  
=====  
Router1#show interfaces  
Ethernet0 is up, line protocol is up  
Hardware is Lance, address is 0010.7be8.7e84 (bia 0010.7be8.7e84)  
Internet address is 10.1.1.3/24  
MTU 1500 bytes, BW 10000 Kbit, DLY 1000 usec, rely 255/255, load  
1/255  
Encapsulation ARPA, loopback not set, keepalive set (10 sec)  
ARP type: ARPA, ARP Timeout 04:00:00  
Last input never, output 00:00:03, output hang never  
Last clearing of "show interface" counters never  
Queueing strategy: fifo  
Output queue 0/40, 0 drops; input queue 0/75, 0 drops  
5 minute input rate 0 bits/sec, 0 packets/sec  
5 minute output rate 0 bits/sec, 0 packets/sec  
    0 packets input, 0 bytes, 0 no buffer  
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles  
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort  
    0 input packets with dribble condition detected  
    887 packets output, 72833 bytes, 0 underruns  
    0 output errors, 0 collisions, 3 interface resets  
    0 babbles, 0 late collision, 0 deferred  
    0 lost carrier, 0 no carrier  
    0 output buffer failures, 0 output buffers swapped out  
=====  
Serial2 is up, line protocol is up  
Hardware is CD2430 in sync mode  
MTU 1500 bytes, BW 115 Kbit, DLY 20000 usec, rely 255/255, load 1/255
```

```
Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)
LMI eng sent 138, LMI stat recv 138, LMI upd recv 0, DTE LMI up
LMI eng recv 0, LMI stat sent 0, LMI upd sent 0
LMI DLCI 0 LMI type is ANSI Annex D frame relay DTE
FR SVC disabled, LAPF state down
Broadcast queue 0/64, broadcasts sent/dropped 623/0, interface
broadcasts 577
Last input 00:00:01, output 00:00:00, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/64/0 (size/max total/threshold/drops)
    Conversations 0/1/256 (active/max active/max total)
    Reserved Conversations 0/0 (allocated/max allocated)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    759 packets input, 51726 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    768 packets output, 52560 bytes, 0 underruns
    0 output errors, 0 collisions, 5 interface resets
    0 output buffer failures, 0 output buffers swapped out
    2 carrier transitions
    DCD=up DSR=up DTR=up RTS=up CTS=up
Serial2.1 is up, line protocol is up
Hardware is CD2430 in sync mode
Description: frame relay to router b
Interface is unnumbered. Using address of Ethernet0 (10.1.1.3)
MTU 1500 bytes, BW 115 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation FRAME-RELAY
Serial2.2 is up, line protocol is up
Hardware is CD2430 in sync mode
Description: frame relay to router c
Interface is unnumbered. Using address of Ethernet0 (10.1.1.3)
MTU 1500 bytes, BW 115 Kbit, DLY 20000 usec, rely 255/255, load 1/255
```

```

Encapsulation FRAME-RELAY
Serial3 is administratively down, line protocol is down
Hardware is CD2430 in sync mode
MTU 1500 bytes, BW 115 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation HDLC, loopback not set, keepalive set (10 sec)
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/64/0 (size/max total/threshold/drops)
  Conversations 0/0/256 (active/max active/max total)
  Reserved Conversations 0/0 (allocated/max allocated)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 no buffer
  Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  0 packets output, 0 bytes, 0 underruns
  0 output errors, 0 collisions, 3 interface resets
  0 output buffer failures, 0 output buffers swapped out
  0 carrier transitions
  DCD=down  DSR=down  DTR=down  RTS=down  CTS=down

```

Router1#**show frame-relay lmi**

```

LMI Statistics for interface Serial2 (Frame Relay DTE) LMI TYPE = ANSI
  Invalid Unnumbered info 0          Invalid Prot Disc 0
  Invalid dummy Call Ref 0        Invalid Msg Type 0
  Invalid Status Message 0       Invalid Lock Shift 0
  Invalid Information ID 0       Invalid Report IE Len 0
  Invalid Report Request 0      Invalid Keep IE Len 0
  Num Status Enq. Sent 139       Num Status msgs Rcvd 139
  Num Update Status Rcvd 0       Num Status Timeouts 0

```

Router1#**sh frame-relay pvc**

```
PVC Statistics for interface Serial2 (Frame Relay DTE)
```

```
DLCI = 100, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE =
Serial2.1
```

input pkts 321	output pkts 324	in bytes 25944
out bytes 25996	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 322	out bcast bytes 25768	
Shaping adapts to BECN		
pvc create time 00:33:42, last time pvc status changed 00:22:43		

```
DLCI = 102, DLCI USAGE = LOCAL, PVC STATUS = ACTIVE, INTERFACE =
Serial2.2
```

input pkts 311	output pkts 314	in bytes 24734
out bytes 25401	dropped pkts 0	in FECN pkts 0
in BECN pkts 0	out FECN pkts 0	out BECN pkts 0
in DE pkts 0	out DE pkts 0	
out bcast pkts 311	out bcast bytes 24849	
pvc create time 00:32:27, last time pvc status changed 00:22:04		

```
Router1#sh frame-relay map
Serial2.1 (up): point-to-point dlci, dlci 100(0x64,0x1840), broadcast
shows the mapping of DLCI
    status defined, active
Serial2.2 (up): point-to-point dlci, dlci 102(0x66,0x1860), broadcast
    status defined, active
```

```
Router1#show frame-relay traffic
```

Frame Relay statistics:

```
    ARP requests sent 0, ARP replies sent 0
    ARP request recv 0, ARP replies recv 0
```

```
Router1#
```

ATM Connections

This section covers ATM connectivity in wide area networks, where ATM is very widely used. Major telecommunication carriers build voice and data backbones using ATM technology. The major benefits of ATM are Quality of Service (QoS), which is required for voice and video traffic. ATM provides fixed size cells of 53 bytes. These cells consist of a 5-byte header and a 48-byte payload. The fixed cell size provides predictability and allows ATM to operate extremely efficiently. ATM is especially useful for time-delay sensitive applications such as voice and video.

ATM Overview

ATM is the building block of Broadband ISDN (B-ISDN) services. The development of optical technologies was a major consideration in its technologies. ATM is a technology developed to address the needs of both voice and data technologies; in voice technologies, there should be guaranteed bandwidth on a per call basis for a call to be reliable. In data technologies, the traffic is bursty. Voice packets are usually small compared to data packets. To address the requirements of both, ATM Forum and other standards organizations agreed to 53-byte cell, with a 5-byte header, and 48-byte payload. ATM technologies scales well at higher speeds like OC-3, OC-12, etc.

Some of the features of ATM are:

- The edge devices provide error and flow control.
- There is no error control on data field within the network, due to low transmission error rates on fiber.
- There is no flow control on links within the network.
- It is connection-oriented at the lowest level.
- All information is transferred in a virtual circuit assigned for the duration of the connection.
- A fixed cell (packet) size permits high-speed switching nodes.
- There is no constraint on data services (segmentation).
- It has an efficient cell structure for bandwidth allocation, and quality of service.

ATM Packet Format

Table 7.8 depicts the ATM cell format.

Table 7.8 ATM Cell Format

Header 5 bytes (8 bits = 1 byte)						Payload 48 bytes
GFC 4 bits	VPI 8 bits	VCI 16 bits	PTI 3 bits	CLP 1 bit	HEC 8 bits	Data

GFC Generic Flow

VPI Virtual Path Identifier—VPI is 8 bits, which gives 256 virtual paths

VCI Virtual Circuit Identifier—VCI is 16 bits, which gives 65K virtual circuits

PTI Payload Type Indicator

CLP Cell Loss—CLP is the cell loss priority bit, which if set, can discard the packet. This is similar to the DE (Discard Eligibility) bit in Frame Relay

HEC Header Error Control—HEC is the check sum error control on the header itself. HEC is also used as a synchronizing delimiter; after three HEC matches the transmission is synchronized

Payload Data

ATM Adaptation Layer (AAL)

The ATM Adaptation Layer (AAL) provides mapping of higher layer application data to and from the ATM cell. The services AAL provides are a SAR (Segmentation Assembly and Re-assembly) layer; also it detects lost cells and errors in cells through a 4-bit sequence number protection. Several AAL types are defined, with each type consisting of a separate SAR sublayer:

- **AAL Type 1** Used for connection-oriented, constant-bit-rate services and is used for circuit emulation.
- **AAL Type 2** Used for connection-oriented, variable-bit-rate services, and is used for video applications.
- **AAL Type 3/4** AAL Type 3 and 4 are combined; they are designed for data applications and support both connectionless and connection oriented applications.
- **AAL Type 5** A more commonly used protocol, applied to VBR (Variable Bit Rate) type traffic. AAL Type 5 is used for signaling and frame relay over ATM.

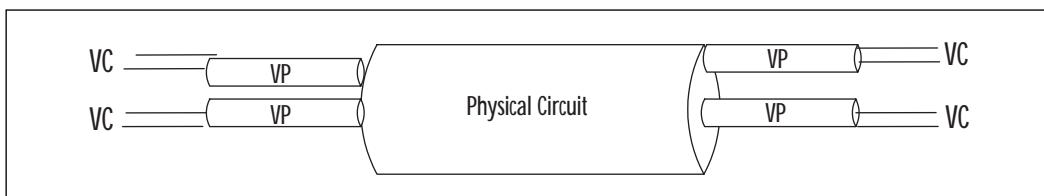
The AAL provides the benefits of error detection, circuit emulation, and connectionless or connection-oriented services depending on the type of AAL used.

ATM Virtual Circuits

ATM virtual circuits are built on top of a VPI/VCI combination. A VC bundle inside of a VP is used to differentiate traffic (like voice, video, and data). VPI/VCI are significant on a physical link between a pair of ATM switches. These circuits are unidirectional, and need mapping in reverse directions to complete conversation between two end-node devices. Circuits can be established as PVCs or SVCs. More popularly used circuits are PVCs, which need mapping and configuration at each ATM switch along the path. SVCs are more dynamic; hence they build and tear the sessions automatically.

Figure 7.17 illustrates that on a given physical ATM network, the VP are the virtual paths that are uniquely identified through VPI. In every virtual path, multiple virtual channels can be defined. VPI is 8 bits long (256 virtual paths), and VCI is 16 bits long (64K circuits), thus providing 256^*64k circuits. The number of channels available gives the granularity needed to provide QoS. Each circuit is a VPI/VCI combination. VPI zero (VPI=0) is reserved.

Figure 7.17 VPI/VCI circuit emulation.



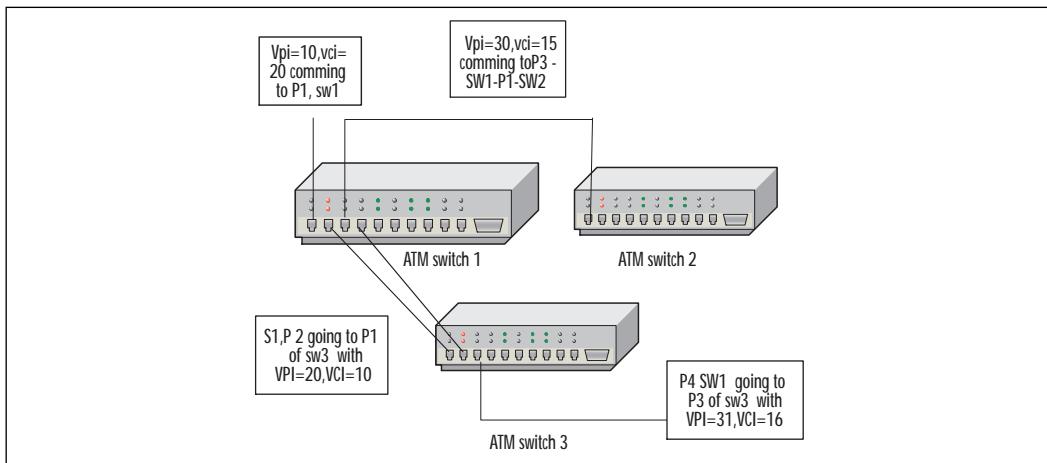
PVC Mapping and Circuit Buildup

Table 7.8 and Figure 7.18 demonstrate the process of PVC mapping and circuit buildup. Notice how the Ports, VCI, and VPI in the table relate and map to the switch diagram. Remember PVCs need to be manually configured on each switch.

Table 7.8 PVC mapping and circuit emulation.

Input Port	VPI	VCI	Port	Output VPI	VCI
1	10	20	2	20	10
2	20	10	1	10	20
3	30	15	4	31	16
4	40	16	3	30	15

Figure 7.18 PVC mapping and circuit buildup.



In the case of Cisco routers with an AIP ATM interface, the PVCs are mapped point-to-point, or point-to-multipoint.

Configuring ATM

Configuring routers for ATM is similar to any other interface on Cisco routers. Set up the interface subsystem in the configuration mode, by typing the interface-related detailed syntax. Figure 7.19 illustrates how to build an ATM network; configurations follow in Figures 7.20 and 7.21.

Figure 7.19 ATM network.

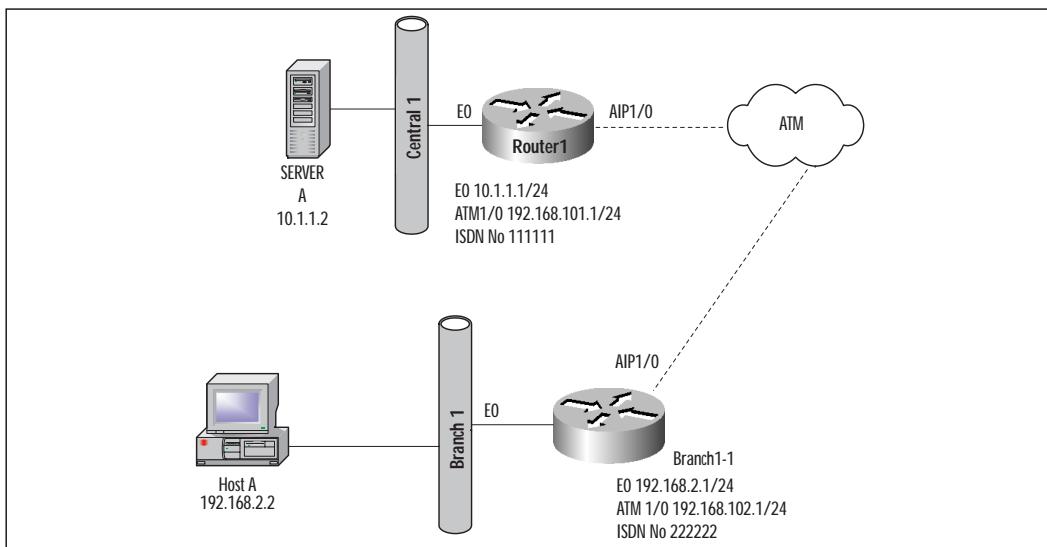


Figure 7.20 Router1 configuration.

```
Router1
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router1
!
!
network-clock base-rate 56k
ip subnet-zero
no ip domain-lookup
!
controller T1 0
framing esf
clock source internal
linecode b8zs
mode atm
!
!
process-max-time 200
!
interface Ethernet0
ip address 10.0.2.1 255.255.255.0
no ip directed-broadcast
!
interface ATM0 → This command provides the configuration mode for atm interface
no ip address
no ip directed-broadcast
no atm ilmi-keepalive
!
interface ATM0.1 point-to-point - defines atm sub interface
```

Continued

Figure 7.20 Continued.

```
ip address 10.0.23.2 255.255.255.0
no ip directed-broadcast
pvc my-data-pvc 0/100 - creates PVC
ubr 64
encapsulation aal5snap
!
no ip address
no ip directed-broadcast
shutdown
!
router igrp 1
network 10.0.0.0
!
ip classless
no ip http server
!
end
```

Figure 7.21 Router2 configuration.

```
Router2
!
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname Router2
!
!
network-clock base-rate 56k
ip subnet-zero
no ip domain-lookup
!
```

Continued

Figure 7.20 Continued.

```
controller T1 0
  framing esf
  linecode b8zs
  mode atm
!
!
process-max-time 200
!
interface Ethernet0
  ip address 10.0.3.1 255.255.255.0
  no ip directed-broadcast
!
interface ATM0
  no ip address
  no ip directed-broadcast
  no atm ilmi-keepalive
!
interface ATM0.1 point-to-point
  ip address 10.0.23.3 255.255.255.0
  no ip directed-broadcast
  pvc my-data-pvc 0/100
  ubr 64
  encapsulation aal5snap
!
!
!
router igrp 1
  network 10.0.0.0
!
ip classless
no ip http server
!
end
```

Verifying and Troubleshooting ATM Connections

The methodology applied in troubleshooting ATM networks is by using show and debug commands relevant to ATM. Various commands that can be used to monitor an ATM network include the following:

```
Router1#show atm ?
arp-server      ATM ARP Server Table
class-links     ATM vc-class links
ilmi-configuration   Display Top level ILMI
ilmi-status     Display ATM Interface ILMI information
interface       Interfaces and ATM information
map             ATM static mapping
pvc              ATM PVC information
signalling      ATM Signaling commands
svc              ATM SVC information
traffic         ATM statistics
vc               ATM VC information
vp               ATM VP information

Router1#show int atm 0
ATM0 is up, line protocol is up
  Hardware is PQUICC Atom1
  MTU 1500 bytes, sub MTU 1500, BW 1536 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ATM, loopback not set → shows the encapsulation mode
on the interface
  Keepalive not supported
  Encapsulation(s):, PVC mode
1024 maximum active VCs, 2 current VCCs → shows Virtual channels
supported
  VC idle disconnect time: 300 seconds
  Last input 00:00:00, output never, output hang never
  Last clearing of "show interface" counters never
  Input queue: 0/75/0 (size/max/drops); Total output drops: 0
  Queueing strategy: weighted fair
```

```

Output queue: 0/1000/64/0 (size/max total/threshold/drops)
  Conversations 0/0/256 (active/max active/max total)
  Reserved Conversations 0/0 (allocated/max allocated)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
  13 packets input, 1008 bytes, 0 no buffer
  Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
  0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
  15 packets output, 1166 bytes, 0 underruns
  0 output errors, 0 collisions, 2 interface resets
  0 output buffer failures, 0 output buffers swapped out

```

The following command shows the details on the sub-interface atm 0.1.

```

Router1# show int atm 0.1
ATM0.1 is up, line protocol is up
  Hardware is PQUICC Atom1
  Internet address is 10.0.23.2/24
  MTU 1500 bytes, BW 1536 Kbit, DLY 20000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation ATM
  12 packets input, 874 bytes
  15 packets output, 1106 bytes
  0 OAM cells input, 0 OAM cells output

```

The following command shows traffic across the ATM link.

```

Router1#SHOW ATM traffic
13 Input packets
14 Output packets
0 Broadcast packets
0 Packets received on non-existent VC
0 Packets attempted to send on non-existent VC
0 OAM cells received
F5 InEndloop: 0, F5 InSegloop: 0, F5 InAIS: 0, F5 InRDI: 0
F4 InEndloop: 0, F4 InSegloop: 0, F4 InAIS: 0, F4 InRDI: 0
0 OAM cells sent

```

```
F5 OutEndloop: 0, F5 OutSegloop: 0,      F5 OutRDI: 0  
F4 OutEndloop: 0, F4 OutSegloop: 0,      F4 OutRDI: 0  
0 OAM cell drops
```

The following command shows the PVC status.

```
Router1#show atm pvc  
On ATM 0.1 interface , my-data-pvc has VPI=0, VCI =100, encapsulation  
is SNAP.
```

```
Router1#show atm pvc  
          VCD /                                Peak   Avg/Min Burst  
Interface    Name        VPI    VCI   Type     Encaps   SC     Kbps   Kbps  
Cells       Sts  
0.1          my-data-pv  0      100   PVC      SNAP     UBR    64      UP  
0.2          my-voice-p  0      200   PVC      VOICE    VBR    384     192  
48          UP (192)
```

The following command shows the mapping between IP address and PVC.

```
Router1#show atm pvc map  
Map list ATM0.1_ATM_INARP : DYNAMIC  
ip 10.0.23.3 maps to VC 1, VPI 0, VCI 100, ATM0.1
```

ATM Debug Commands:

```
Router1#debug atm ?  
aal-crc      Display CRC error packets  
arp          Show ATM ARP events  
compress     ATM Compression  
errors       ATM errors  
events       ATM or FUNI Events  
ilmi         Show ILMI events  
oam          Dump OAM Cells  
packet      ATM or FUNI packets  
pvcd        Show PVCD events  
sig-all     ATM Signalling all  
sig-api     ATM Signalling api  
sig-error   ATM Signalling errors  
sig-events  ATM Signalling events
```

```
sig-ie      ATM Signalling information elements
sig-packets ATM Signalling packets
smap-all    ATM Signalling Static Map all
smap-error   ATM Signalling Static Map errors
smap-events  ATM Signalling Static Map events
state       ATM or FUNI VC States
```

Let's look at some ATM debug commands that will further aid in troubleshooting ATM implementations.

The debug atm packet Command

The **debug atm packet** command will display all ATM packets.

```
Router1#debug atm packet
ATM packets debugging is on
Displaying all ATM packets
Router1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router1(config)#int atm 0
Router1(config-if)#shut
Router1(config-if)#no shut
Router1(config-if)#exit
Router1(config)#exit

Router1#sho log
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
  Console logging: disabled
  Monitor logging: level debugging, 0 messages logged
  Buffer logging: level debugging, 351 messages logged
  Trap logging: level informational, 47 message lines logged

Log Buffer (4096 bytes):

04:45:47: %SYS-5-CONFIG_I: Configured from console by console
```

```
04:46:06: %LINK-5-CHANGED: Interface ATM0, changed state to administratively down
04:46:07: %LINEPROTO-5-UPDOWN: Line protocol on Interface ATM0, changed state to down
04:46:21: ATM0.1(O):
VCD:0x1 VPI:0x0 VCI:0x64 DM:0x100 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x56
Shows the 1/100 (64hex) pvc sending an ip packet type0800.
04:46:21: 45C0 004A 0000 0000 0209 96EA 0A00 1702 FFFF FFFF 1105 0001 0003 0000 0000
04:46:21: 53C9 0002 0000 0064 0003 E805 DCFF 0100 0003 00FF FFFF 0100 0501 1043 6973
04:46:21: 0017 0000 07D0 0019 6E05 DCFF 0100
04:46:21:
04:46:22: ATM0.1(O): -o -Outgoing packet
VCD:0x1 VPI:0x0 VCI:0x64 DM:0x100 SAP:AAAA CTL:03 OUI:000000 TYPE:0806 Length:0x20
Arp packet type 0806
04:46:22: 0013 0800 0000 0008 0400 0004 0A00 1702 0000 0000
04:46:22:
04:46:22: ATM0.1(I):
VCD:0x1 VPI:0x0 VCI:0x64 Type:0x0 SAP:AAAA CTL:03 OUI:000000 TYPE:0806 Length:0x20
04:46:22: 0013 0800 0000 0009 0400 0004 0A00 1703 0A00 1702
04:46:22:
04:46:23: %LINK-3-UPDOWN: Interface ATM0, changed state to up
04:46:24: %LINEPROTO-5-UPDOWN: Line protocol on Interface ATM0, changed state to up
04:46:25: %SYS-5-CONFIG_I: Configured from console by console
04:46:34: ATM0.1(I):
VCD:0x1 VPI:0x0 VCI:0x64 Type:0x0 SAP:AAAA CTL:03 OUI:000000 TYPE:0800 Length:0x56
04:46:34: 45C0 004A 0000 0000 0109 97E9 0A00 1703 FFFF FFFF 1101 0001 0003 0000 0000
```

```
04:46:34: 8030 0002 0000 0834 0019 6E05 DCFF 0101 0003 0000 0064 0003  
E805 DCFF 0100  
04:46:34:  
04:47:54:  
04:48:02: %SYS-5-CONFIG_I: Configured from console by console  
Router1# conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
Router1(config)#exit  
Router1#no debug atm packet  
ATM packets debugging is off
```

The debug atm state Command

Use the **debug atm state** command to see changes in the state of the ATM VCs.

```
Router1#debug atm state  
ATM VC States debugging is on  
Router1#conf t  
Enter configuration commands, one per line. End with CNTL/Z.  
Router1(config)#int atm 0  
Router1(config-if)#shut  
Router1(config-if)#no shut  
Router1(config-if)#exit  
Router1(config)#exit  
Router1#sho log  
  
Log Buffer (4096 bytes):  
  
04:48:02: %SYS-5-CONFIG_I: Configured from console by console  
04:48:18: %SYS-5-CONFIG_I: Configured from console by console  
04:48:40: %LINK-5-CHANGED: Interface ATM0, changed state to  
administratively down  
04:48:41: %LINEPROTO-5-UPDOWN: Line protocol on Interface ATM0, changed  
state to down  
04:49:12: %LINK-3-UPDOWN: Interface ATM0, changed state to up
```

```
04:49:13: %LINEPROTO-5-UPDOWN: Line protocol on Interface ATM0, changed state to up  
04:49:18: %SYS-5-CONFIG_I: Configured from console by console
```

The following conversation provides ATM VC states.

```
04:51:08: Changing vc 0/100vc-state to ATM_VC_SHUTTING_DOWN  
04:51:08: Changing vc 0/100vc-state to ATM_VC_NOT_IN_SERVICE  
04:51:08: Changing vc 0/100vc-state to ATM_VC_NOT_IN_SERVICE  
04:51:08: Changing vc 0/200vc-state to ATM_VC_SHUTTING_DOWN  
04:51:08: Changing vc 0/200vc-state to ATM_VC_NOT_IN_SERVICE  
04:51:08: Changing vc 0/200vc-state to ATM_VC_NOT_IN_SERVICE  
04:51:10: %LINK-5-CHANGED: Interface ATM0, changed state to administratively down  
04:51:11: %LINEPROTO-5-UPDOWN: Line protocol on Interface ATM0, changed state to down  
04:51:41: Changing vc 0/100 vc-state to ATM_VC_NOT_VERIFIED  
04:51:41: Changing vc 0/100 vc-state to ATM_VC_ESTABLISHING_VC  
04:51:41: Changing vc 0/100 vc-state to ATM_VC_NOT_VERIFIED  
04:51:41: Changing vc 0/100 vc-state to ATM_VC_UP  
04:51:41: Changing vc 0/200 vc-state to ATM_VC_NOT_VERIFIED  
04:51:41: Changing vc 0/200 vc-state to ATM_VC_ESTABLISHING_VC  
04:51:41: Changing vc 0/200 vc-state to ATM_VC_NOT_VERIFIED  
04:51:41: Changing vc 0/200 vc-state to ATM_VC_UP  
  
04:51:43: %LINK-3-UPDOWN: Interface ATM0, changed state to up  
04:51:44: %LINEPROTO-5-UPDOWN: Line protocol on Interface ATM0, changed state to up  
04:51:46: %SYS-5-CONFIG_I: Configured from console by console
```

The debug atm ilmi Command

The **debug atm ilmi** command provides ilmi conversations.

```
Router1#debug atm ilmi  
Setting ILMI debug for all interfaces.  
Router1#conf t  
Enter configuration commands, one per line. End with CNTL/Z.
```

```
Router1(config)#int atm 0
Router1(config-if)#shut
Router1(config-if)#no shut
Router1(config-if)#exit
Router1(config)#exit

Router1#sho log
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
  Console logging: disabled
  Monitor logging: level debugging, 0 messages logged
  Buffer logging: level debugging, 529 messages logged
  Trap logging: level informational, 67 message lines logged
Log Buffer (4096 bytes):
ILMI conversation starts here
tion error on o/g ILMI Pdu <ilmi_send_pkt> (ATM0)
04:57:33: ILMI: Unable to Send Pdu out <ilmi_send_trap> sends an SNMP trap
04:57:35: ILMI(ATM0): Sending ilmiColdStart trap
04:57:35: ILMI(ATM0): No ILMI VC found
04:57:35: ILMI: Encapsulation error on o/g ILMI Pdu <ilmi_send_pkt> (ATM0)
04:57:35: ILMI: Unable to Send Pdu out <ilmi_send_trap>
04:57:37: ILMI(ATM0): Sending ilmiColdStart trap
04:57:37: ILMI(ATM0): No ILMI VC found
04:57:37: ILMI: Encapsulation error on o/g ILMI Pdu <ilmi_send_pkt> (ATM0)
04:57:37: ILMI: Unable to Send Pdu out <ilmi_send_trap>
04:57:38: ILMI(ATM0): Received Interface Down. Shutting down ILMI
04:57:40: %LINK-5-CHANGED: Interface ATM0, changed state to administratively down
04:57:41: %LINEPROTO-5-UPDOWN: Line protocol on Interface ATM0, changed state to down
04:58:01: ILMI(ATM0): Received Interface Up
04:58:01: ILMI(ATM0): Sending ilmiColdStart trap
04:58:01: ILMI(ATM0): No ILMI VC found
```

```
04:58:01: ILMI: Encapsulation error on o/g ILMI Pdu <ilm_i_send_pkt>  
(ATM0)  
04:58:01: ILMI: Unable to Send Pdu out <ilm_i_send_trap>  
04:58:03: %LINK-3-UPDOWN: Interface ATM0, changed state to up  
Router1#no debug all
```

Backing up Permanent Connections

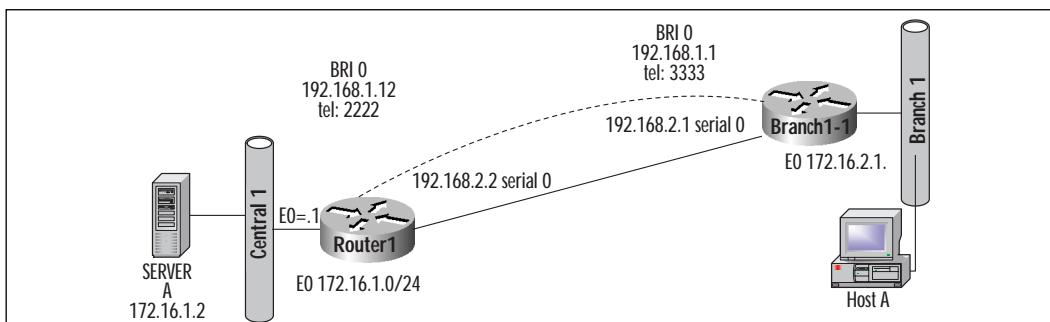
Permanent connections provide connectivity between local and remote sites. Although we call them *permanent connections*, we all know that nothing is ever really permanent, right? Like any other physical entity, these permanent connections are susceptible to failure. The problem with these connections is that if and when they fail, all connectivity is lost, resulting in costly downtime for the remote users. In order to provide fault tolerance to the remote site, you must have a backup connection in place in case the permanent connection does fail. In the event of a permanent connection failure, the backup connection should be able to kick in (transparent to the end-users) without any administrative intervention, and pick up right where the failed link left off. Let's take a look at some of the ways in which we can provide this type of backup connection.

Backup Interface

The backup interface is one of the mechanisms that provides redundancy in wide area networks. The backup interface is configured in the primary interface configuration; when the primary goes down, it recognizes the loss of signal on the primary and raises DTR on the secondary interface.

Figure 7.22 illustrates how to configure the backup interface on a point-to-point link.

Figure 7.22 A point-to-point permanent connection with an ISDN backup connection.



```
! central site

version 11.3
!
hostname Central-1
!
isdn switch-type basic-dms100
!
interface Ethernet0
ip address 172.16.1.1 255.255.255.0
no ip route-cache
no ip mroute-cache
!
interface Serial0
backup delay 30 never
backup interface BRI0
backup load 70 40
ip address 192.168.2.2 255.255.255.0
no ip route-cache
no ip mroute-cache
bandwidth 64
no shutdown
!
interface BRI0
ip address 192.168.1.2 255.255.255.0
encapsulation ppp
no ip route-cache
no ip mroute-cache
bandwidth 64
dialer idle-timeout 1
dialer map ip 192.168.1.1 name Branch1 3333
dialer load-threshold 180 outbound
dialer-group 10
isdn switch-type basic-dms100
```

```
ppp authentication chap
no shutdown
!
ip classless
!
access-list 120 permit ip 172.16.2.0 0.0.0.255 host 192.168.1.1
dialer-list 10 protocol ip permit
!
end
```

Branch-1

```
!
version 11.3
!
hostname Branch1
!
isdn switch-type basic-dms100
!
interface Ethernet0
ip address 172.16.2.1 255.255.255.0
no ip route-cache
no ip mroute-cache
!
interface Serial0
backup delay 60 20 →!When primary fails, it waits for 60 sec,
!                               When primary comes back, the backup link waits
!                               for 20 sec before shutting down
backup interface BRI0 →BRI 0 will be activated in case of s0 failure
backup load 80 30
ip address 192.168.2.1 255.255.255.0
no ip route-cache
no ip mroute-cache
bandwidth 64
no shutdown
```

```

!
interface BRI0
  ip address 192.168.1.1 255.255.255.0
  encapsulation ppp
  no ip route-cache
  no ip mroute-cache
  bandwidth 64
  dialer idle-timeout 180
  dialer map ip 192.168.1.2 name CENTRAL-1 1111 ! Dialer string points
to remote
  !
  !                                         side of the link
  dialer load-threshold 1 either
  dialer-group 10
  isdn switch-type basic-dms100 ! ISDN switch type provided by TELCO
  ppp authentication chap
  no shutdown
!
ip classless
ip route 17.16.1.0 255.255.255.0 192.168.1.2
!
access-list 120 permit ip 172.16.1.0 0.0.0.255 host 192.168.1.2
access-list 120 permit tcp any any established
dialer-list 10 protocol ip list 120
!
end

```

The backup load Command

The **backup load** command allows you to use a secondary link when a set utilization has been reached. This command will enable or *bring up* a second interface, while the primary is still up and running, giving you additional bandwidth as needed. This is desirable when there is heavy traffic on the primary link. For example:

```

interface serial 1
!
backup interface bri 0

```

```
!  
backup load 85 10 →
```

If the primary link is 85 percent utilized, the backup line comes up. If the primary line's available bandwidth is less than 10 percent of the utilization of the backup link, the backup comes down.

Floating Static Routes and Default Routes

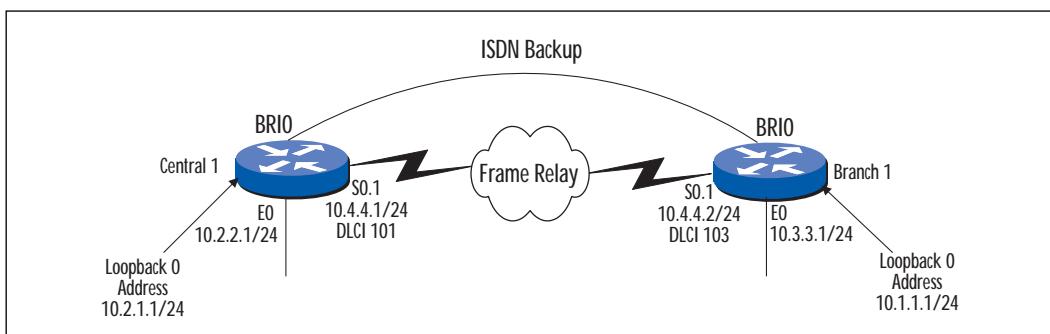
Floating static routing is another method of providing redundancy in a network. Similar to the **backup interface** command, it is a more dynamic method that provides a higher level of guarantee. To understand the way a floating static route works, you must first understand routing metrics.

Metrics in a routing environment provide a mechanism for the routing table manager (RTM) to decide which route to prefer. Each routing protocol has a default metric, like EBGP 20 and Open Shortest Path First (OSPF) 110. If a route can be reached via both EBGP and OSPF, the preferred route will be through EBGP, because it has a low-cost route. Static routes by default have a zero metric. A floating static route provides a mechanism to increase the cost to reach a specific route; therefore, the dynamic routing protocol route is preferred.

A floating static route is more efficient than a backup interface, because a floating static route is already installed in the routing table. There is no convergence time required for a floating static route to be active. In case of the need for a backup interface, the router IOS has to activate the backup interface, make a connection via a dial-up to an ISDN or similar physical line. The router has to start sending interesting packets, sending routing updates on the new route. The new route injection into the network will take time depending upon the convergence times, the diameter of the network, etc.

Figure 7.23 shows a floating static configuration.

Figure 7.23 Frame Relay network with ISDN backup.



Frame Relay Configuration with ISDN Backup

```
Central-1
CENTRAL-1
!
version 11.3
!
hostname Central-1
!
isdn switch-type basic-5ess
!
interface Loopback0
 ip address 10.2.1.1 255.255.255.0
!
interface Ethernet0
 ip address 10.2.2.1 255.255.255.0
 no shutdown
!
interface Serial0
 no ip address
 encapsulation frame-relay
 no shutdown
!
interface Serial0.1 point-to-point
 ip address 10.4.4.1 255.255.255.0
 frame-relay interface-dlci 101
 no shutdown
!
interface BRI0
 ip unnumbered Ethernet0
 encapsulation ppp
 no ip route-cache
 no ip mroute-cache
 dialer map ip 10.1.1.1 name branch1 3333
 dialer-group 1
```

```
isdn switch-type basic-5ess
ppp authentication chap callin
ppp chap hostname Central-1
ppp chap password 7 070C285F4D06
hold-queue 75 in
no shutdown
!
router eigrp 100
network 10.0.0.0
!
ip classless
ip route 10.1.1.1 255.255.255.255 BRI0 180 !floating static
!                                         metric 180 will be active when
primary fails

ip route 10.3.3.0 255.255.255.0 10.1.1.1 180! Floating static
!
access-list 101 deny    ip any host 255.255.255.255
access-list 101 deny    eigrp any any
access-list 101 permit ip any any
dialer-list 1 protocol ip list 101
!
end
```

```
Branch1
!
version 11.3
!
hostname Branch1
isdn switch-type basic-5ess
!
```

```
interface Loopback0
 ip address 10.1.1.1 255.255.255.0
!
interface Ethernet0
 ip address 10.3.3.1 255.255.255.0
 no ip route-cache
 no ip mroute-cache
 no shutdown
!
interface Serial0
 no ip address
 encapsulation frame-relay
 no ip route-cache
 no ip mroute-cache
 no shutdown
!
interface Serial0.1 point-to-point
 ip address 10.4.4.2 255.255.255.0
 no ip route-cache
 no ip mroute-cache
 frame-relay interface-dlci 103
 no shutdown
!
interface BRI0
 ip unnumbered Ethernet0
 encapsulation ppp
 no ip route-cache
 no ip mroute-cache
 dialer map ip 10.2.1.1 name Central-1 2222
 dialer-group 1
 isdn switch-type basic-5ess
 ppp authentication chap callin
 ppp chap hostname Branch1
 hold-queue 75 in
```

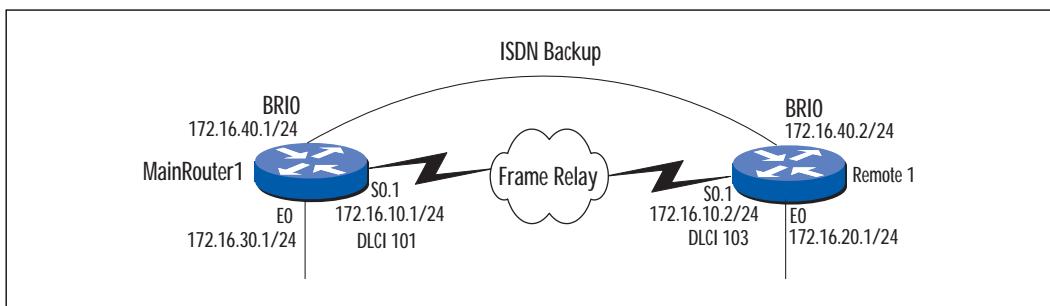
```

no shutdown
!
router eigrp 100
  network 10.0.0.0
!
ip classless
ip route 10.2.1.1 255.255.255.255 BRIO 180
ip route 10.2.2.0 255.255.255.0 10.2.1.1 180
!
access-list 101 deny ip any host 255.255.255.255
access-list 101 deny eigrp any any
access-list 101 permit ip any any
dialer-list 1 protocol ip list 101
!
end

```

The route table for routers with floating static routes will contain all of the learned and connected routes as well as the floating static. Take a look at Figure 7.24.

Figure 7.24 ISDN backup and floating static route.



MainRouter has a floating static route configured to reach the 172.16.20.0/24 network in the event that the frame relay link fails. The following is an example of what you would see in the route table of MainRouter with its floating static route, prior to the primary link failure:

```
MainRouter#show ip route
```

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
 N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
 E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
 i - IS-IS, L1 - ISIS level-1, L2 - ISIS level-2, * - candidate default
 U - per-user static route, o - ODR

Gateway of last resort is not set

```
172.16.0.0/16 is variably subnetted, 4 subnets, 1 mask
D      172.16.20.0/24 [90/2195456] via 172.16.10.2, 00:07:28, Serial0
C      172.16.30.0/24 is directly connected, Ethernet0
C      172.16.10.0/24 is directly connected, Serial0
C      172.16.40.0/24 is directly connected, BRI0
S*     172.16.20.0/24 [180/0] via 172.16.40.1
```

Notice that MainRouter has learned about the 172.16.20.0/24 network via EIGRP. The floating static route has an asterisk specifying that it is candidate default (standby mode). It has an administrative distance of 180 (EIGRP administrative distance is 90), which means that currently the EIGRP route is preferred. If the EIGRP route disappears for any reason, the floating static route will take over. Now let's see what happens to the route table after the primary route to the 172.16.20 network disappears due to a frame relay link failure:

MainRouter#show ip route

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
 N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
 E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
 i - IS-IS, L1 - ISIS level-1, L2 - ISIS level-2, * - candidate default
 U - per-user static route, o - ODR

```
Gateway of last resort is not set
```

```
172.16.0.0/16 is variably subnetted, 4 subnets, 1 mask
C         172.16.30.0/24 is directly connected, Ethernet0
C         172.16.10.0/24 is directly connected, Serial0
C         172.16.40.0/24 is directly connected, BRI0
S         172.16.20.0/24 [180/0] via 172.16.40.1
```

Notice that the EIGRP route has disappeared and that the static route is no longer in standby mode (it has no asterisk next to it). It has taken over and is providing a route to the 172.16.20.0 network through the BRI0 interface. It will be the primary route to that network until such a time when the router learns a new route via a routing protocol with a lower administrative distance. The static route would then return to standby mode (candidate default).

Dialer Watch

Dialer Watch or Dial Backup is used in a DDR environment to monitor an active interface. An ISDN BRI interface can be configured to monitor a Frame Relay interface or a Frame Relay DLCI. The monitoring interface becomes activated when the monitored interface or the DLCI reaches a down status. This method keeps the monitoring interface (ISDN) in a perpetual state of down and cannot be used to send/receive any traffic until the monitored interface goes down or the configured load threshold has been exceeded.

To configure a BRI0 interface to backup a serial0 interface, in the s0 interface configuration mode, type:

```
Int s0
Backup interface bri0
Backup delay timel time2
```

Time1 is the time in seconds that the backup interface waits before going into activation after the primary line went down, and *time2* is the time in seconds the backup interface waits before going into standby mode after the primary line is up.

Another method to Dial Backup is to configure a Floating Static Route. Static routes are usually preferred to dynamic routes. In order for a dynamic route to be preferred over a static route, higher administrative distance value is given to the static route. When the primary interface fails, the dynamic route is aged out. At this point, the static route will be used

to get to the remote network. This static route is usually configured on the BRI interface. The key issue is that the static route must be redistributed into a routing process that provided the dynamic route.

To configure a floating static route for the above scenario, on the BRI interface, configure:

```
Int bri0
Ip address and other configuration parameters, and on the primary
interface s0, configure
Int s0
Ip address and other parameters,
Run a routing process, e.g. router rip
Redistribute static
Distance xx, where xx is an integer and must be greater than the rip
administrative distance.
From global command mode type
Ip route remote net bri0 or ip route remote net ip address of bri0
```

The two methods described so far keep the backup interface in perpetual standby mode until activated by a primary interface failure.

Configuring a Dialer Profile

Another method is to configure a dialer profile. This is a logical interface that can be configured with most of the parameters of a physical interface. The logical interface now monitors the primary or active interface, and activates the physical ISDN interface only when the active interface fails. This means that the ISDN line could be used to send/receive traffic.

The dialer profile configuration follows a similar ISDN configuration except that it is a logical interface.

```
Int dialer n ! where n is an integer
Ip address and other configuration parameters.
Dialer pool x where x is an integer
Dialer string where string is the remote phone number
```

On the BRI0 interface, configure:

```
!
Int bri0
Encapsulation ppp
Ppp authentication chap
```

```
Dialer pool-member x
```

On the serial interface, configure:

```
Int s0
Ip address and other parameters
Backup interface dialern
Backup delay timel time2
```

With the dialer watch configuration, the ISDN interface is only used when needed and released after use.

Verifying and Troubleshooting Backup Connections

Let's look at some of the commands that can be used to troubleshoot and verify your ISDN backup connections (see Figure 7.25).

Figure 7.25 The show controller command.

Show controller provides the physical level information on the line.

```
Central1# show controller
BRI unit 0
```

!On BRI , ISDN the channels are divided into 2B+D. Here in the output below these channels !show layer1 is activated. The message activated ensures, that the BRI interface is !successfully communicated with carrier ISDN switch.

D Chan Info:

Layer 1 is ACTIVATED

```
    idb 0x148D68, ds 0x15BE88, reset_mask 0x8
    buffer size 1524
    RX ring with 2 entries at 0x2101600 : Rxhead 1
        00 pak=0x15C41C ds=0x614FEC status=D000 pak_size=0
        01 pak=0x15C614 ds=0x6156AC status=F000 pak_size=0
    TX ring with 2 entries at 0x2101640: tx_count = 0, tx_head = 0, tx_tail
        = 0
        00 pak=0x000000 ds=0x000000 status=00 pak_size=0
        01 pak=0x000000 ds=0x000000 status=00 pak_size=0
```

Continued

Figure 7.25 Continued.

```
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 memory errors
0 transmitter underruns
0 d channel collisions

B1 Chan Info:
Layer 1 is ACTIVATED
idb 0x14E8CC, ds 0x15BF60, reset_mask 0x0
buffer size 1524
RX ring with 8 entries at 0x2101400 : Rxhead 0

00 pak=0x15E108 ds=0x61AE6C status=D000 pak_size=0
01 pak=0x15DF10 ds=0x61A7AC status=D000 pak_size=0
02 pak=0x15DD18 ds=0x61A0EC status=D000 pak_size=0
03 pak=0x15DB20 ds=0x619A2C status=D000 pak_size=0
04 pak=0x15D928 ds=0x61936C status=D000 pak_size=0
05 pak=0x15D730 ds=0x618CAC status=D000 pak_size=0
06 pak=0x15D538 ds=0x6185EC status=D000 pak_size=0
07 pak=0x15D340 ds=0x617F2C status=F000 pak_size=0

TX ring with 2 entries at 0x2101440: tx_count = 0, tx_head = 0, tx_tail
= 0
00 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
01 pak=0x000000 ds=0x000000 status=7C00 pak_size=0
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 memory errors
0 transmitter underruns
0 d channel collisions

B2 Chan Info:
Layer 1 is ACTIVATED
idb 0x154430, ds 0x15C038, reset_mask 0x2
buffer size 1524
RX ring with 8 entries at 0x2101500 : Rxhead 0

00 pak=0x1601E4 ds=0x621A6C status=D000 pak_size=0
01 pak=0x15FFEC ds=0x6213AC status=D000 pak_size=0
02 pak=0x15FDF4 ds=0x620CEC status=D000 pak_size=0
```

Continued

Figure 7.25 Continued.

```
03 pak=0x15FBFC ds=0x62062C status=D000 pak_size=0
04 pak=0x15FA04 ds=0x61FF6C status=D000 pak_size=0
05 pak=0x15F80C ds=0x61F8AC status=D000 pak_size=0
06 pak=0x15F614 ds=0x61F1EC status=D000 pak_size=0
07 pak=0x15F41C ds=0x61EB2C status=F000 pak_size=0
TX ring with 2 entries at 0x2101540: tx_count = 0, tx_head = 0, tx_tail
= 0
00 pak=0x000000 ds=0x000000 status=5C00 pak_size=0
01 pak=0x000000 ds=0x000000 status=7C00 pak_size=0
0 missed datagrams, 0 overruns, 0 bad frame addresses
0 bad datagram encapsulations, 0 memory errors
0 transmitter underruns
0 d channel collisions
```

Show interface BRI 0 to find most of the details about ISDN link.

```
Central1# show interface BRI0
BRI0 is up, line protocol is up (spoofing) ! Spoofing indicates that BRI
interface is acting as      backup interface
Hardware is BRI
Internet address is 10.2.2.1/24
MTU 1500 bytes, BW 64 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation PPP, loopback not set
Last input 00:00:00, output 00:00:00, output hang never
Last clearing of "show interface" counters never
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: weighted fair
Output queue: 0/1000/64/0 (size/max total/threshold/drops)
    Conversations 0/1/256 (active/max active/max total)
    Reserved Conversations 0/0 (allocated/max allocated)
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
108 packets input, 524 bytes, 0 no buffer
```

Continued

Figure 7.25 Continued.

```

Received 20 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
106 packets output, 508 bytes, 0 underruns
0 output errors, 0 collisions, 12 interface resets

! interface resets and carrier transitions do occur, as backup
interface comes up when !activated, and goes to spoofing after primary
comes_back

0 output buffer failures, 0 output buffers swapped out
3 carrier transitions

Show isdn status,
Centrall1#sh isdn status
Global ISDN Switchtype = basic-5ess !Identifies the switch type used
ISDN BRI0 interface
    dsl 0, interface ISDN Switchtype = basic-5ess
Layer 1 Status:
    ACTIVE
Layer 2 Status: !show that layer 2 is active and bonding at 64k on
the B1 channel
    TEI = 64, Ces = 1, SAPI = 0, State =
MULTIPLE_FRAME_ESTABLISHED
Layer 3 Status:
    0 Active Layer 3 Call(s)
Activated dsl 0 CCBs = 0
Total Allocated ISDN CCBs = 0

```

Usually the initial setup problems are a switch type mismatch (like 5ESS or DMS100), or a wrong SPID number. SPID numbers might need a leading or trailing zero, depending what the Telco has programmed.

```

Debug ISDN Event
BRI0: ISDN Event: incoming ces value = 1
BRI0: received HOST_TERM_REGISTER_NACK - invalid EID/SPID or TEI not
assigned
Cause i = 0x8082 - No route to specified network

```

To verify that TEI is assigned, show ISDN status. Also look at the ISDN CSU/DSU (ISDSU) to see if it has a TEI link up. On ADTRAN ISU128 models, the ISDN configuration is provided through the key pad. Using the key pad, check the status of the line, which will provide the TEI line link-up details.

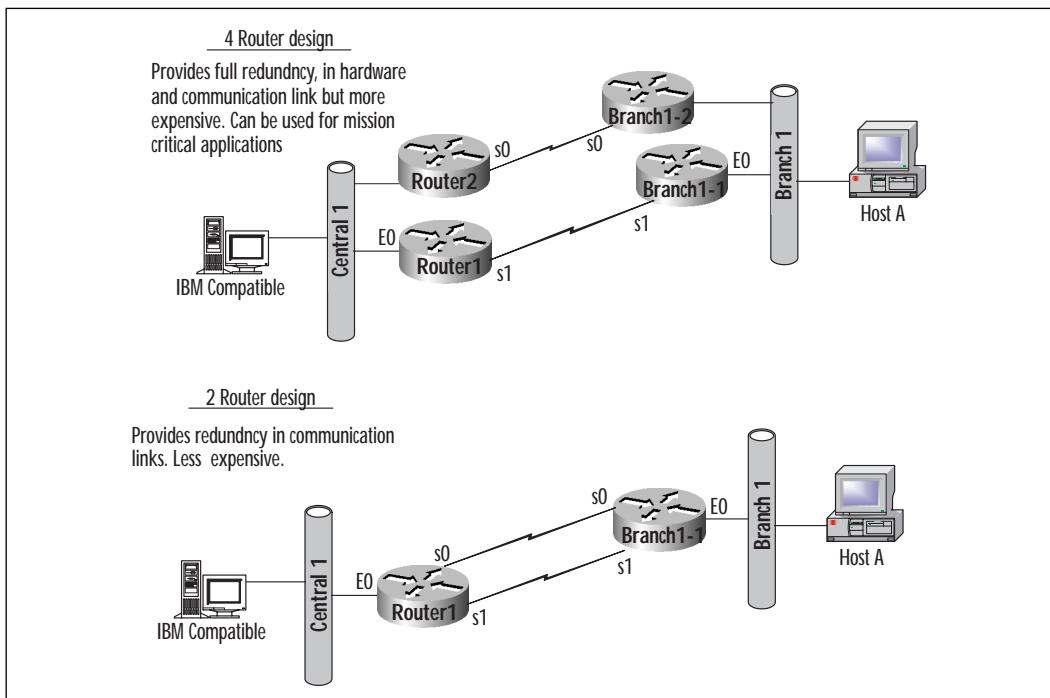
Routing Issues

Make sure the backup interface, or the floating static, is properly functioning, by disconnecting the primary cable and looking at the routing tables. As the ISDN comes up, see if it is getting activated on the interesting packet.

Redundant Hardware and Links/ Design and Performance Issues

The network could be designed with built-in redundancy like two T-1s between two sites. In this type of scenario we might see two routers at each physical location (site), with a total of four. Another scenario may be one router with two serial links (with a total of two). See Figure 7.26 for an illustration of the two scenarios.

Figure 7.26 Redundant hardware and link designs for backups.



The usual practice entails one router on each end with two serial links, because of the cost of the hardware (two routers instead of four), and a better throughput due to the load balancing.

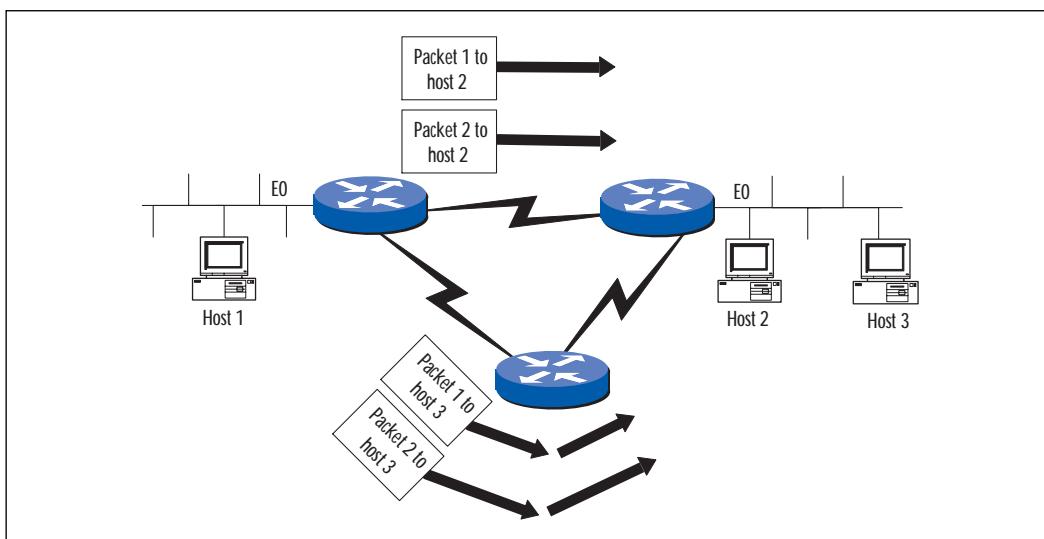
Load Balancing

Cisco routers can support two types of load balancing (sometimes referred to as *load sharing*): per-destination load balancing and per-packet load balancing. Let's look at each of these in detail.

Per-Destination Load Balancing

By default, Cisco routers are in a *fast switching mode*. This means that the first time a router receives a packet addressed to a particular destination, it will perform a route-table lookup and select the route. That information is then stored in the fast switching cache so that any subsequent packets bound for the same destination can be immediately switched and sent over the predetermined route without having to perform another lookup. This means that all packets destined for a particular host will take the same route. All packets destined for another host on the same destination network can and will take a different route. The balance is decided on a per-destination basis. Refer to Figure 7.27. Notice that there are two packets destined for each of the two hosts (Host 2 and Host 3). The path that each packet takes is dependent on which destination it is bound for.

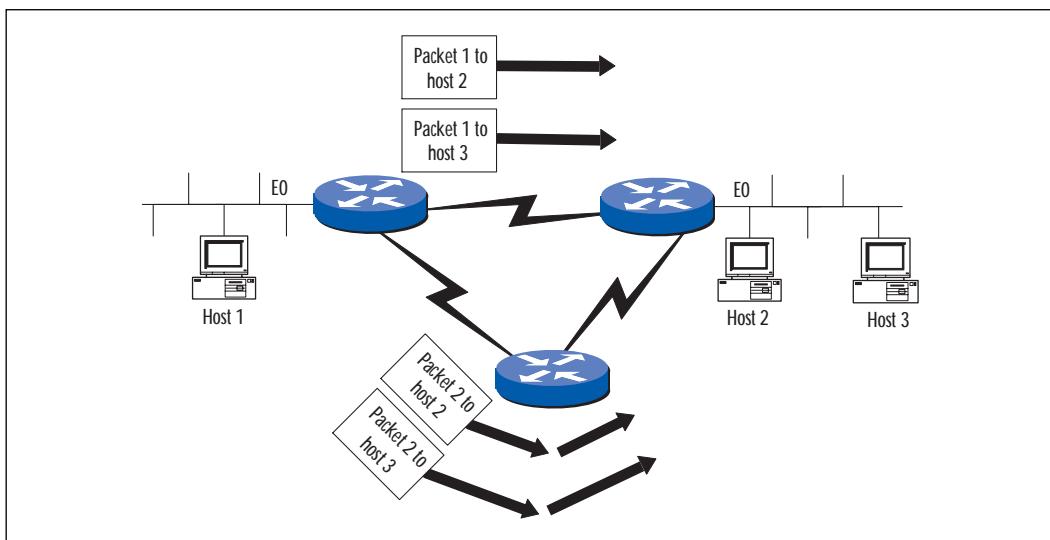
Figure 7.27 Per-destination load balancing.



Per-packet Load Balancing

By implementing the command **no ip route-cache** on a Cisco router, two things on the router change. First, the router will load-balance traffic across two equal cost paths on a packet-by-packet basis. Second, the router will switch from the default setting of fast switching to process switching. Process switching simply means that the router will do a route-table lookup for each packet it must process. Because each route decision is independent, packets will be distributed evenly across the two equal cost paths. (See Figure 7.28.) Per-packet load balancing results in more evenly balanced traffic over the equal cost links—however, there are a couple of drawbacks to this method. The switching process is not as fast as fast switching (hence the name) and there is added overhead on the CPU. You must consider this when selecting the load balancing method for a particular network. Refer to Figure 7.28. Notice that, regardless of the destination, the packets are evenly distributed over the two links.

Figure 7.28 Per packet load balancing.



Summary

We covered a *lot* of material in this chapter! It examined point-to-point connections and their benefit, the related protocols, and a simple point-to-point configuration.

It presented a simple X.25 connection and explained how X.25 addressing works. It described X.25 virtual circuits, outlined a simple X.25 configuration, and ran through the basics of troubleshooting an X.25 implementation.

Next it covered Frame Relay packets and the fields they contain. It explained LMI, CIR, FRTS, and sub interfaces as well as Frame Relay topologies and configurations. It described troubleshooting in a Frame Relay environment, the related troubleshooting commands, and some common problems.

ATM connections and the fixed cell length of 53 bytes were described next. We talked about the fact that the fixed cell length cuts down on latency and is much more efficient for transmitting voice or video data. The discussion covered ATM packets and virtual circuits. We looked at some ATM configurations and talked about troubleshooting ATM networks.

The chapter concluded by describing what it takes to provide some level of fault tolerance to your connections. We looked at backup interfaces, backup ISDN circuits, floating static routes, redundant hardware, and load balancing. All of these elements can provide more dependable network connectivity in the event of a link failure.

FAQs

Q: How many DLCIs can be configured on a Cisco 2500 with a T-1 interface?

A: Up to 60 DLCI can be configured on the 2500 series router.

Q: Is ATM suitable for WAN technology or for LAN technology?

A: ATM is more appropriate for WAN technologies, as it provides QoS functions, which are critical for meeting SLAs.

Q: Where is X.25 technology still in use, and why?

A: X.25 is still in use in Europe and other countries outside of the US. Most of the telecommunication links outside the US are low-speed, error-prone lines. X.25 is well suited for these types of lines.

Q: What are the advantages of using Dialer Watch?

A: Dialer Watch lets you use one ISDN backup line for backing up many permanent connections.

Q: Which technologies are better suited for voice and data?

A: ATM technology has been the most efficient, but with IP, QoS technologies (in Cisco IOS) other media technologies (Ethernet, Frame, etc.) are now capable of providing a very high quality of service.

Securing your Remote Access Network

Solutions in this chapter:

- Cisco Firewall Feature Set
- AAA overview
- CiscoSecure
- Authentication, Authorization and Accounting (AAA)
- Virtual profiles
- Per-user configuration

Introduction

One of the problems facing today's network administrators is that of security and access control. As networks expand, and more networking devices need to be managed, scalability issues arise, particularly if access to these devices is to be centrally managed. As telecommuting becomes more popular, remote access solutions such as dial-up Public Switched Telephone Network (PSTN) and Integrated Services Digital Network (ISDN) connections on network access servers (NAS) also need to be managed.

As businesses become more competitive, the need to keep information internal and private is becoming an absolute necessity. This can be accomplished by implementing a security solution known as a *firewall*, which determines what type of traffic can enter or leave your network and who can get into your network from the outside. In this chapter we will see how Cisco has made it possible to run a software package that includes a built-in firewall.

Access control is another method of adding security into a network infrastructure. Access control, while complementing firewall technology, is a way to manage which users can access your network server (authentication), what services they are allowed to use once they have that access (authorization), and logging of that access (accounting). These components, called AAA for short, provide an architectural framework for configuring the three independent security functions of authentication, authorization, and accounting in a consistent manner.

Although AAA can be configured with local security functions, security protocols such as Remote Authentication Dial-in User Service (RADIUS), or Terminal Access Controller Access Control System Plus (TACACS+), allow us to provide a centrally managed, scalable access control solution. In this chapter we will be looking at how we can use AAA to scale access control in an expanding network.

What is a Firewall?

A firewall is a network device that controls and monitors access to areas of a network. It can be a dedicated hardware device, such as the Cisco PIX Firewall, or software loaded onto an existing network device. The most common use of a firewall is to protect the network of an organization that is connected to the Internet, by monitoring and filtering network traffic at network entrance points. A firewall is also used to provide additional protection to sensitive areas within an organization's intranet, such as finance and research departments, and to secure entrance points to the networks of customers or suppliers (extranets). If there are multiple access points to a network, then multiple firewalls are required.

Cisco IOS Firewall Feature Set

The Cisco Firewall Feature Set is supported in Cisco IOS version 11.2(11)P and later, with additional features added in version 12.0(5)T. It is a software option that runs on a variety of Cisco hardware platforms and adds firewall features, enhances current IOS security, and improves intrusion detection. It seamlessly integrates with existing security features to allow the Cisco router to behave as a full-featured firewall, which offers security and policy enforcement throughout intranets, extranets, and connections to the Internet.

The Firewall Feature Set is scalable and will run on Cisco 1600, 1720, 2500, 2600, 3600, and 7200 routers.

When used with Internet Protocol Security (IPSec), and other Cisco features such as Quality of Service (QoS) and Layer 2 Tunneling Protocol (L2TP) tunneling, it can provide a comprehensive virtual private network (VPN) solution. This enables telecommuters, customers, and suppliers to securely connect to your private network using public networks.

The Cisco IOS Firewall Feature Set will protect the internal network, monitor and filter traffic through network boundaries, and enable secure WWW commerce.

Firewall Feature Set Benefits and Features

The current Firewall Feature Set was developed in three phases, with each phase offering additional and enhanced features. Phase I is available only on IOS 11.2(11)P and later, 11.3(3)T and later, 12.0, 12.0(1)T-12.0(4)T, and 12.0(4)XE. Phase I+ and Phase II are available only on IOS 12.0(5)T and later.

Phase I

Phase I's *initial features* consist of Context-based Access Control (CBAC), Java blocking, denial of service detection and prevention, and real-time alerts and audit trail features.

Phase I+

Phase I+'s *enhanced features* include all of the features of Phase I, plus dynamic port mapping, configurable alerts and audit trail, Simple Mail Transfer Protocol (SMTP) attack detection and prevention, and MS Netshow support.

Phase II (Full Features)

Phase II's *full features* consist of all of the features of Phase I and Phase I+, plus intrusion detection (59 signatures), and dynamic per-use authentication and authorization (authentication proxy).

Key Benefits

The following list summarizes the key benefits offered by the Cisco Firewall Feature Set.

Scalability The feature set can be scaled to meet with any performance and bandwidth requirements, and is available on a number of Cisco platforms.

Protection of investment Current investment in Cisco technology and skills is protected, by building on the currently-used Cisco hardware and software.

VPN support When used with other Cisco IOS features, the Firewall Feature Set offers a full VPN solution. This allows remote users secure and cost-effective access to the organization's network over public networks.

Flexibility Using the Firewall Feature Set on a Cisco router enables you to perform a comprehensive suite of routing and firewall functions. These include multiprotocol routing, intrusion detection, authentication and authorization, VPN support, and perimeter security.

Ease of management Cisco ConfigMaker can be used to configure all Cisco features from a remote central console. This includes all common router features, as well as the Firewall Feature Set.

The IOS Firewall Feature Set is a comprehensive security solution for networks with existing Cisco devices. It allows organizations to build on their current investment in Cisco technology, while enhancing existing security features and adding a full-featured router-based firewall solution.

AAA Overview

The letters AAA stand for authentication, authorization, and accounting. AAA is a framework that allows you to control and monitor who is allowed to access particular services on your network. The AAA framework separates the three components into independent security functions.

The authentication feature validates the identity of the user through usernames and passwords; it also utilizes challenge/response, messaging, and encryption. The authorization feature determines what access level is available to a particular user, group, system, or process. The accounting feature collects and distributes security information used for billing, auditing, and reporting. It records actions such as logon/logout time, commands executed, and traffic sent, and will distribute this information to the appropriate locations.

By using these three separate functions, you have a flexible, modular solution to control access to your network. AAA allows you to easily configure the level of access given on a per-user, or per-service (IPX, AppleTalk, etc.) basis.

AAA offers the following benefits:

Scalability AAA can be scaled to control access to networks of all sizes. Further access control can easily be added when required.

Greater flexibility and control Access can be controlled on a per-user, per-group, or per-service basis, allowed actions can be tightly controlled, and detailed accounting information can be recorded.

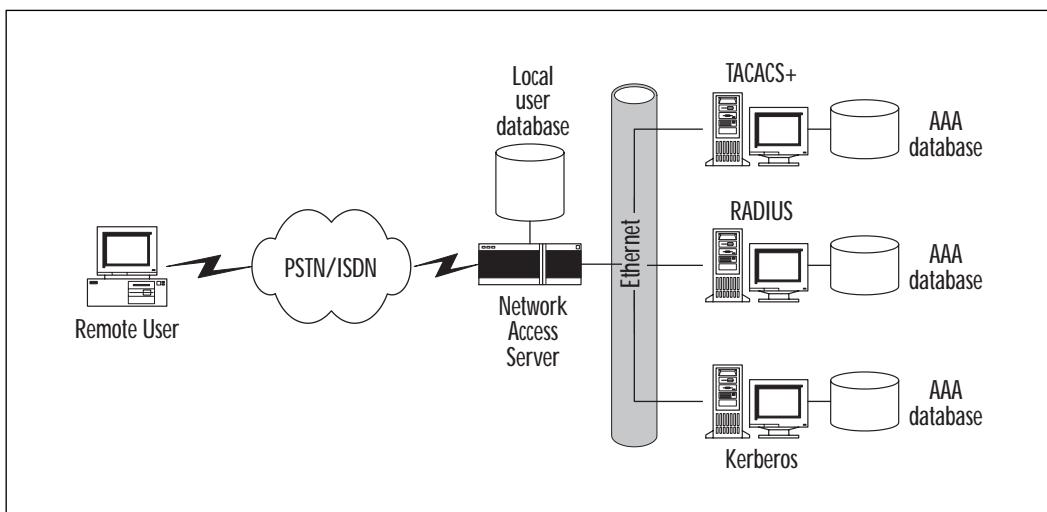
Standard authentication methods Established authentication standards such as RADIUS, TACACS+, and Kerberos may be used.

Multiple backup systems Many AAA servers can be used to provide access control, and security information may be replicated amongst these servers to provide redundancy.

AAA Servers

When using AAA, a network access server (NAS) or router must be able to access security information for a specific user to provide authentication, authorization, and accounting services. The network administrator has two main options for where to hold this information—locally, or on a remote AAA server (see Figure 8.1).

Figure 8.1 AAA servers.



If AAA information is held locally, user AAA account information is held on the router or access server itself. These accounts are created through the Cisco IOS and are used to permit or deny user access. When using this solution, AAA negotiation is performed internally within Cisco IOS, and is therefore protocol-independent. However, only a limited number of Cisco-specific security attribute values are supported.

When using server-based remote AAA, the router or network access server negotiates with a remote AAA security server to determine whether a user is to be allowed access. User and group information is held on the AAA security server, along with accounting records. An access server uses a standard security protocol (such as TACACS+, RADIUS, or Kerberos) and supports a wide range of security attribute values—not only Cisco-specific attributes. Using an access server for AAA services also allows for fault-tolerance and redundancy. Multiple security servers might be used to authenticate users with user information stored on several servers. If one security server becomes inaccessible, the user could be authenticated via another source.

CiscoSecure

CiscoSecure is a suite of access control software applications that enable the centralization of security policies, while integrating Cisco IOS software features. There are a variety of different products available; what you choose will depend on the hardware platform and the scale of your security requirement. The CiscoSecure range is Cisco's AAA server solution. It runs on UNIX or Windows NT platforms and supports common security protocols such as TACACS+ and RADIUS. CiscoSecure will enable secure-dial network solutions for corporations, service providers, and medium and small businesses.

The CiscoSecure products currently available are CiscoSecure ACS for Windows NT, CiscoSecure ACS for UNIX, and CiscoSecure Global Roaming Server for UNIX.

CiscoSecure ACS for Windows NT

This product is designed for workgroups and enterprises that need a standard security policy throughout a Windows NT infrastructure. Its main features are:

- Easy-to-use Access Control Server (ACS) running on Windows NT
- Windows NT or flat-file database
- TACACS+ and RADIUS support
- Unlimited NAS support

CiscoSecure ACS for UNIX

This product is aimed more at the larger corporation and Internet service providers (ISPs), offering increased security and reliability as well as extra features required by such organizations. Its key features include:

- Powerful ACS running on UNIX
- Relational database
- TACACS+ and RADIUS support
- Unlimited NAS support

CiscoSecure Global Roaming Server for UNIX

CiscoSecure Global Roaming Server (GRS) is a solution for ISPs, enabling them to offer secure dial VPN and Internet roaming solutions to their customers. Using GRS, VPN and Internet users will be able to access a global roaming network using many ISPs' existing TACACS+ and RADIUS servers. CiscoSecure GRS features include:

- TACACS+ and RADIUS proxy
- Relational or flat-file databases
- TACACS+ and RADIUS translation

Authentication

Authentication is a method of validating the claimed identity of users, before allowing them access to the network. It works by stepping through a predefined list of authentication methods that have been applied to the interface the user is accessing through. These lists are known as method-lists, and are sequenced lists of authentication methods defined by the administrator—named and applied to a specific interface. Interfaces with no user-defined method-lists automatically use a default method-list, which is (not surprisingly) named *default*. Any user-defined method-lists will automatically override the default list.

Authorization

Authorization determines the actions that an authorized user, group, system or service is allowed to perform. AAA generates a set of attributes that identify the actions a user is allowed to perform. This set is then compared with an entry in a security database specific to the user, which determines exactly what the user is authorized to perform. Attribute-value (AV) pairs defining the user rights are associated with the user to determine the specific user rights.

The authorization database holds authorization information for users accessing the network, and can be held on the access server or router itself, or on a remote security server such as TACACS+ or RADIUS.

Authorization methods are defined through AAA in a similar manner to authentication. You must define a named list of sequenced authorization methods, and apply the list to an interface.

Accounting

Accounting tracks resources used by a user, and the network resources that they consume. This information can then be sent back to a security server in the form of an accounting record for further analysis. This information is used by network administrators for security auditing, network management, and billing purposes. Account records are made up of accounting AV pairs, which are stored on the access server or router.

As with authentication and authorization, accounting methods must be defined through AAA. A named list of accounting methods can be defined and then applied to an interface.

Method-Lists

A method-list is a sequenced list of authentication, authorization, or accounting methods. Each entry in the list is tried in order to provide the required AAA service. For example, when a user attempts to authenticate, the access server contacts each authentication source specified in the authentication method-list, in sequence, to try to authenticate the user. One or more security servers may be specified to offer fault-tolerance and backup of authentication databases.

A security server may respond to an authentication request with either a PASS or FAIL message; no response is treated as an ERROR. If the authentication receives a PASS message, then the user is considered authenticated and may access the system; no further entries in the access list need be processed. A FAIL message means the user is not authenticated and is not allowed access; no further entries in the access list are processed. An ERROR means that there was no entry found for that user using that particular method; the next entry in the method-list is processed and the authentication process begins again. If all entries in a method-list have been processed without the user obtaining a PASS message, access is denied.

The following is an example method-list configured on a Cisco router:

```
router(config)#aaa authentication login default tacacs+ radius local  
none
```

It is an AAA authentication method-list named **default** used to verify a user **login**. The method-list consists of three entries, **tacacs+**, **radius**, and **local**. This means that initially the network access server will try to authenticate the login by TACACS+, and if this does not respond, RADIUS is attempted; if RADIUS does not respond, a local database is interrogated. If all these authentication methods fail, access is denied. Detailed command syntax is discussed later in this chapter.

Security Protocols

Security protocols provide access control for routers, network access servers, and other networked computing devices via one or more centralized servers. You can choose between two major security protocols, depending on the requirements of your particular environment. These are RADIUS and TACACS+. The following section outlines the key features of each and suggests which might be more appropriate for different environments.

Remote Authentication Dial-in User Service (RADIUS)

RADIUS is a connectionless, client-server protocol used for security authentication and authorization. Network access servers generally act as clients, where the server is usually the RADIUS process running on a UNIX or Microsoft Windows NT server. The RADIUS server can also act as a proxy to other RADIUS servers or other kinds of authentication servers. RADIUS uses User Datagram Protocol (UDP) for its client-server communications, and is therefore a connectionless protocol. As UDP uses best-effort delivery, all retransmissions are handled by the RADIUS devices, not by the transmission protocol.

RADIUS was developed by Livingstone Enterprises Inc., and has gained wide industry acceptance by many ISPs as the favored security protocol—primarily because of its relatively small CPU and memory requirements. Request for Comments (RFC) 2138 details the RADIUS protocol specification, and RFC 2139 is an informational document detailing the RADIUS accounting standard.

Terminal Access Controller Access Control System Plus (TACACS+)

TACACS was originally developed by BBN for the MILNET, but has since been extended several times by Cisco. It provides separate authentication, authorization, and accounting services using the connection-oriented

Transmission Control Protocol (TCP). Although it provides all three AAA services, not all have to be used in a particular implementation, because they are separate processes. By separating authentication from authorization, it is possible to create a dynamic authorization process, which can be integrated with other security negotiations such as Point-to-Point Protocol (PPP).

There are many TACACS+ servers available, but the AAA server was designed specifically to be scalable and compatible with Cisco routers. TACACS+ supports 16 privilege levels, and controls a greater range of service than other security protocols. It can control enable, shell, and standard login—along with PPP, AppleTalk Remote Access Protocol (ARAP), remote command (RCMD), firewall proxy, and Novell Asynchronous Services Interface (NASI). TACACS+ can also block services from certain ports, and control which router commands a particular user or group is allowed to perform.

Comparing TACACS+ and RADIUS

TACACS+ offers a much wider range of Cisco-specific security features than RADIUS, and should be seriously considered for use in a predominantly Cisco environment. However, RADIUS has a wide industry acceptance and continues to be the security protocol of choice for many ISPs. RADIUS benefits from increased vendor interoperability, as well as reduced CPU and memory requirements. Although RADIUS does not guarantee vendor interoperability, there are about 45 standard RADIUS attributes that enhance the likelihood of interoperability. Table 8.1 provides a summary of the key differences between TACACS+ and RADIUS.

Table 8.1 TACACS+ and RADIUS Comparison

TACACS+	RADIUS
Connection-oriented, uses TCP	Connectionless, uses UDP
Encrypts entire body of packet (more secure)	Encrypts only the password in an access-request packet (less secure)
Uses AAA, with separate authentication, authorization, and accounting processes	Combines authentication and authorization
Multiprotocol support	Limited protocol support; does not support NetBIOS Frame Protocol Control Protocol, Appletalk Remote Access Protocol, Novell Asynchronous Services Interface, or X.25 PAD connections

Continued

Table 8.1 Continued

TACACS+	RADIUS
Can control commands used on the router on a per-user, or per-group basis	Cannot control which commands a user can execute on a router
More memory- and processor-intensive Cisco proprietary	Less memory- and processor-intensive Industry standard

For Managers**Choosing a Security Server**

It's important to take great care when deciding which security protocol and security servers are suitable for your particular environment. Although the decision must ultimately be made by the manager, it is essential to have input from all technical professionals, such as network engineers, server administrators, and security analysts. You must clearly define your security needs, and the ability of your existing network and server hardware to support those needs. For example, you must ensure that the security protocol you choose can support all protocols you are likely to be using. There is no point in choosing RADIUS if you are using Apple Macintosh computers with Appletalk Remote Access Protocol. Also, you must ensure your team has the appropriate skills to support all aspects of the implementation.

There are many flavors of security server software available, many of which support both RADIUS and TACACS+, and others. However, these vary enormously in the security features they offer, even though they may appear similar. Most of these are available for free download, usually with a limited evaluation period, or license count, which may then be activated to the full version. Use these trial versions to build the products you are considering into your test environment—to ensure they suit your current infrastructure and meet your expectations.

Using RADIUS and TACACS+ for AAA Services

Both RADIUS and TACACS+ can be used to provide authentication, authorization, and accounting services to Cisco network access servers. The three functions are independent with TACACS+, but authentication and authorization are combined with RADIUS.

AAA information is stored on the RADIUS or TACACS+ server, which is queried by the NAS when a user attempts to authenticate or perform an action. If accounting is configured, information on all defined accounting events is sent to the security server.

The IP addresses or names of security servers are configured on the router—along with other parameters—and each is tried when a particular method of AAA is required. For example, all defined TACACS+ servers are attempted for providing authentication services when TACACS+ is specified as an accounting method.

There are many TACACS+ and RADIUS daemons available from most major networking equipment suppliers.

Configuring AAA

The AAA configuration process takes place in a number of distinct stages. First, AAA must be enabled on the router, then method-lists for each of the AAA components must be defined, then these method-lists must be associated with interfaces or lines.

Enabling AAA

To be able to use any of the AAA network security services you must enable AAA. Once AAA has been enabled you can no longer use commands to configure the older protocols, TACACS, or extended TACACS. AAA must be enabled in global configuration mode.

To enable AAA use:

```
router(config)#aaa new-model
```

Configuring the RADIUS or TACACS+ Parameters

Configuration of TACACS+ and RADIUS both use a single required command, followed by a number of optional commands—depending on your specific requirements.

Configuring TACACS+ Parameters

The **tacacs-server** command is used to set TACACS+ server parameters in global configuration mode. With this command you can set the IP address of the TACACS+ server, the encryption key used by the server, client-server timeouts, maximum number of failed attempts at executing commands, and other server-specific settings.

Defining a TACACS+ Server Host

The optional **timeout** keyword sets the amount of time a server will wait for a host to reply before timing out. The optional **key** keyword sets the encryption key used between the access server and the TACACS+ daemon. Any **timeout** or **key** settings made here for this specific host will override any global settings for these values.

```
router(config)#tacacs-server host name [timeout integer] [key string]
```

Optional TACACS+ Commands

Table 8.2 details optional configuration commands that might suit your security requirements.

Table 8.2 Optional TACACS+ Commands

Command	Description
router(config)#tacacs-server retransmit <i>retries</i>	Specifies the number of times the server searches the list of TACACS+ servers before stopping.
router(config)#tacacs-server timeout <i>seconds</i>	Sets the amount of time a server will wait for a host to reply before timing out.
router(config)#tacacs-server attempts <i>count</i>	Sets the number of login attempts that can be made on the line.
router(config)#tacacs-server key <i>key</i>	Sets the encryption key used between the access server and the TACACS+ daemon.

Configuring RADIUS Parameters

The **radius-server** command is used to set RADIUS server parameters in global configuration mode.

Defining a RADIUS Server Host

The **auth-port** and **acct-port** keywords specify port numbers used for authentication and accounting, respectively.

```
router(config)#radius-server host {hostname | ip-address} [auth-port
port-number] [acct-port port-number]
```

Optional TACACS+ Commands

Table 8.3 lists optional RADIUS configuration commands.

Table 8.3 Optional RADIUS Commands

Command	Description
router(config)#radius-server key <i>string</i>	Specifies the shared secret string used between the router and RADIUS server.
router(config)#radius-server retransmit <i>retries</i>	Specifies the number of times the server searches the list of RADIUS servers before stopping. The default is 3.
router(config)#radius-server timeout <i>seconds</i>	Sets the amount of time a server will wait for a host to reply before timing out.
router(config)#radius-server deadtime <i>minutes</i>	Sets the amount of time a RADIUS server will continue to be used if no authentication requests are acknowledged.
router(config)#radius-server vsa send [accounting authentication]	Enables the NAS to use and recognize RADIUS IETF attribute 26 vendor-specific-attributes. This allows more Cisco-specific attribute-value pairs to be recognized by RADIUS.

Configuring AAA Authentication

There are many different authentication types defined by AAA—including login, enable, arap, nasi, and ppp. The following are the most commonly used types of authentication.

The **aaa authentication login** Command

The **aaa authentication login** command is used to enable AAA authentication, regardless of the authentication method you decide to use. With this command, you define a list of one or more login authorization methods that will be tried when a user logs in, and then apply this list to an interface.

To create a local login authentication list use:

```
router(config)#aaa authentication login {default | list-name} method1  
[method2...]
```

The *list-name* is a character string used to identify the method-list. It is this name you use when you apply the list to a line.

There can be one or more *methods* that identify which authentication methods are attempted and in which order. If you want to allow a user access even if all authentication methods fail, add the **none** keyword at the end of the method-list. Table 8.4 lists supported methods and their descriptions.

Table 8.4 AAA Authentication Login Method Types

Keyword	Description
Enable	Use enable password for authentication.
If-needed	Do not authenticate if a user has already been authenticated on a TTY line.
Krb5	Use Kerberos version 5 for authentication.
Krb5-telnet	User Kerberos 5 Telnet authentication when using Telnet to connect to the router. If used, must be the first method in the method-list.
Line	Use line password for authentication.
Local	Use local username for authentication.
None	Use no authentication.
Radius	Use RADIUS authentication.
Tacacs+	Use TACACS+ authentication.

To apply an authentication login list to a line or set of lines, use:

```
router(config)#line [aux | console | tty | vty] line number [end-line-number]
router(config-line)#login authentication {default | list-name}
```

The following configuration is an example of how a router may be configured to use AAA login authentication. The authentication list is first defined, then applied to the appropriate interfaces.

```
router(config)#aaa new-model
router(config)#aaa authentication login default tacacs+ radius
router(config)#aaa authentication login customers tacacs+ radius local
none
router(config)#line 0
router(config-line)#login authentication default
router(config-line)#exit
router(config)#line 1-16
router(config-line)#login authentication customers
```

The aaa authentication ppp Command

The **aaa authentication ppp** command is used to specify authentication methods for use on serial interfaces using PPP. To create a ppp authentication list, use:

```
router(config)#aaa authentication ppp {default | list-name} method1
[method2..]
```

Table 8.5 details the methods supported by **aaa authentication ppp**.

Table 8.5 AAA Authentication PPP Method Types

Keyword	Description
Local	Local username database used for authentication.
Krb5	Kerberos 5 used for authentication (PAP only).
If-needed	Does not authenticate if user has already been authenticated on a TTY line.
None	No authentication used.
Radius	RADIUS used for authentication
Tacacs+	TACACS+ used for authentication.

The method-list is then applied to an interface using:

```
router(config)#interface interface-type interface-number
router(config-line)#ppp authentication {chap | pap | chap pap | pap chap } [if-needed] {default | list-name} [callin]
```

The following configuration is an example of how a router may be configured to use AAA PPP authentication. The authentication list is first defined, then applied to serial interface 0.

```
router(config)#aaa new-model
router(config)#aaa authentication ppp default tacacs+ radius
router(config)#interface s0
router(config-if)#encapsulation ppp
router(config-if)#ppp authentication chap default
```

In the example above, a default PPP authentication method-list has been created. Initially, TACACS+ is used to try to authenticate the user, then RADIUS is used. If both authentication methods fail, authentication fails. The default method-list is then applied to interface serial 0.

The aaa authentication enable default Command

The **aaa authentication enable default** command is used to determine whether a user can access the privileged-command level.

```
router(config)#aaa authentication enable default [method1 [method2...]]
```

Table 8.6 lists methods supported by **aaa authentication enable**; if no method is specified then no authentication is used. Therefore, access is always allowed.

Table 8.6 AAA Authentication Enable Method Types

Keyword	Description
Line	Line password used for authentication.
If-needed	Does not authenticate if user has already been authenticated on a TTY line.
None	No authentication used.
Radius	RADIUS used for authentication
Tacacs+	TACACS+ used for authentication.

Configuring AAA Authorization

Once the user has been authenticated, authorization is used to restrict access. The **aaa authorization** global command is used to configure AAA authorization. AAA supports four types of authorization:

Network This applies to network connections, including PPP, ARAP, or Serial Line Internet Protocol (SLIP).

EXEC Applies to the user EXEC terminal session.

Commands Applies to EXEC mode commands issued by a user.

Authorization is attempted for all EXEC mode commands associated with a particular access level.

Reverse access Applies to reverse Telnet sessions.

AAA supports six authorization methods used to determine a user's access to each of the authorization types:

If authenticated The user is allowed to access the requested feature if successfully authenticated.

Local The access server uses its local database to provide authorization for the requested feature. The local database is defined using the **username** command and can only be used to authorize certain functions.

None Authorization is not performed for this function.

RADIUS A RADIUS server is used to provide authorization functions. This is performed by associating attributes held in the RADIUS database with a particular user.

TACACS+ A TACACS+ server is used to provide authorization functions. Authorization is performed by associating a user with attribute-value pairs stored in the TACACS+ security database.

Kerberos instance map The instance defined by the **kerberos instance map** command is used.

When using basic AAA authorization only a single method is used to attempt to authorize a user. If this method fails, no authorization is granted.

```
router(config)#aaa authorization {network | exec | commands level |  
reverse-access} {if-authenticated | local | none | radius | tacacs+ |  
krb5-instance }
```

For example, the command **aaa authorization exec tacacs+** would cause the access server to use a TACACS+ database to provide authentica-

tion for EXEC mode commands. By using an authorization method-list, several authorization methods may be used in sequence to attempt to authorize a user to carry out a particular function.

```
router(config)#aaa authorization {network | exec | commands level | reverse-access}{default | list-name} [method1 [method2...]]
```

The authorization method-list is assigned to a line as follows:

```
router(config)#line [aux | console | tty | vty] line-number [ending-line-number]
router(config-line)#authorization {arap | commands level | exec | reverse-access} {default | list-name}
```

The authorization method-list is assigned to an interface as follows:

```
router(config)#interface interface-type interface-number
router(config-if)#ppp authorization {default | list-name}
```

The following sample shows how a router can be configured to use AAA authorization:

```
router(config)#aaa new-model
router(config)#aaa authorization network default tacacs+ local if-authenticated
router(config)#aaa authorization exec admins tacacs+ local
router(config)#interface serial 0
router(config-if)#ppp authorization default
router(config)#line console 0
router(config-line)#authorization admins
```

In the example above, two authorization method-lists are defined, a network ‘default’, and ‘admins.’ The ‘default’ network list attempts authorization by TACACS+, and then checks the NAS database. If both these methods fail, the **if-authenticated** keyword will cause the user to be granted authorization only if they have been successfully authenticated. The ‘admins’ exec list attempts to authorize access to an EXEC session first by TACACS+, then by the local user database. If both fail, authorization is denied.

The ‘default’ network method-list is applied to interface serial 0. The ‘admins’ method-list is applied to the console line.

Configuring AAA Accounting

Accounting is a very powerful network auditing feature, allowing user-activity information to be collected and stored on your security server. The **aaa accounting** global command is used to configure AAA accounting. AAA supports five types of accounting:

Network Will monitor and report information on network connections, including PPP, ARAP, or SLIP. Information recorded includes items such as byte or packet count, protocol used, username, and start and stop times.

EXEC Reports on information about user EXEC terminal sessions on the NAS. Information includes start and stop times, IP address of the NAS, and the number that dialed in for dial-up users.

Commands Reports on all EXEC terminal commands executed by a user, recording information such as the command used, privilege level of the command, and username. Cisco command accounting can be used only with TACACS+ security servers.

System System accounting reports on all system level events, such as reboots and when accounting is turned on or off. Cisco system accounting can only be used with TACACS+ security servers, and does not support named method-lists (default only).

Connection Reports on outbound connections made from the NAS, such as Telnet, local-area transport (LAT), packet assembler/disassembler (PAD), TN3270, and rlogin.

AAA supports only two accounting methods:

RADIUS A RADIUS server is used to record accounting information. Only limited types of accounting are supported.

TACACS+ A TACACS+ server is used to record accounting information.

Basic AAA accounting is enabled using the following command:

```
router(config)#aaa accounting {system | network | connection | exec |
commands level } {start-stop | wait-start | stop-only} {tacacs+ |
radius}
```

Table 8.7 lists the options used when an accounting record is to be generated.

For example, the **aaa accounting connection stop-only tacacs+** global configuration command would report on outbound connections from the NAS to a TACACS+, only when the event has ended.

By using an accounting method-list, accounting records may be sent to several accounting servers.

Table 8.7 AAA Accounting Report Triggers

Keyword	Description
Start-stop	An accounting record is sent when a process to be reported on starts, and again when it ends.
Wait-start	An accounting record is sent when a process to be reported on starts. The security server must acknowledge that the record has been received before the user can continue with the process.
Stop-only	An accounting record is only sent at the end of the process to be reported on.

```
router(config)#aaa accounting {system | network | connection | exec |
commands level } {default | list-name} {start-stop | wait-start | stop-
only} [method1 [method2...]]
```

The following commands apply an accounting method-list to a line:

```
router(config)#line [aux | console | tty | vty] line-number [ending-
line-number]
router(config-line)#accounting {arap | commands level | exec |
connection} {default | list-name}
```

Using the **arap** keyword will report on network accounting events.

The following commands are used to apply an accounting method-list to an interface:

```
router(config)#interface interface-type interface-number
router(config-if)#ppp accounting {default | list-name}
```

The following configuration commands show how accounting can be configured on a router and then applied to a group of lines.

```
router(config)#aaa new-model
router(config)#aaa accounting connection sessions stop-only tacacs+
router(config)#aaa accounting network users wait-start tacacs+
router(config)#aaa accounting commands 10 admins start-stop tacacs+
radius
router(config)#line tty 8 16
router(config-line)#accounting connection sessions
router(config-line)#accounting arap users
router(config-line)#accounting commands 10 admins
```

In the example above, three accounting method-lists are defined: sessions, users, and admins. Sessions reports outbound connections from the NAS to a TACACS+ server on their completion. The users method-list reports network events to a TACACS+ server; however the TACACS+ server must acknowledge receipt of the accounting record before the user may proceed. Admins reports information on privilege level 10 commands when they begin, and when they end. A TACACS+ server is sent records first, and a RADIUS server is used if TACACS+ fails. The three method-lists are applied to TTY lines 8 through 16.

Virtual Profiles and AAA

Virtual profiles are an exceptionally powerful feature, allowing per-user configurations defined on central security servers to be applied to dialer interfaces. This is a PPP-specific feature, and can be used in conjunction with dialer profiles to provide a unique interface to each user. Virtual profiles are totally independent of the media used for the dial-in call; Integrated Services Digital Network (ISDN) and Public Switched Telephone Network (PSTN) dial-in users, for example, could use the same profiles.

Virtual profile configuration can be derived from a virtual interface configuration, per-user configuration stored on an AAA security server, or from a combination of the two.

Virtual profiles are used to overcome current network scalability limitations:

AAA implementation Currently per-user configuration is limited by the AV pairs defined by the AAA implementation. Virtual profiles allow more Cisco-specific attributes to be used.

Media Each interface currently can be accessed only by statically defined users associated with that interface. Using virtual profiles allows a user configuration to be dynamically bound to an interface when it is accessed.

Network protocols When using virtual profiles, network numbers are assigned dynamically on dial-in.

Dial-on-demand routing (DDR) DDR is designed to add routers when a temporary link comes up, but not remove them when they are torn down. Dynamically adding and removing routes improves scalability.

Dialer profiles Dialer profiles solve some of the limitations of legacy DDR, but are limited by the number of physical interfaces on the router. Virtual profiles can scale to many thousands of dial-in users.

ISDN Currently AAA user configurations are applied to the ISDN D-channel, and both B-channels. Using virtual profiles allows you to bind user configurations to individual B-channels.

However, there are some limitations on virtual profiles, in that they do not support fast-switching, virtual private dial-up network (VPDN), or Layer 2 Forwarding Protocol (L2F) tunneling.

When using virtual profiles, per-user configuration is separated into two logical parts:

Generic A generic virtual interface template is used to specify an interface configuration that is common to all dial-in users. A virtual interface template overrides any physical interface configuration.

User-dependent User-specific configuration is stored in a file on the AAA security server. This information is sent to a network access server when a user is authenticated, and can override any previous configuration information.

The two parts can be used independently, or combined, allowing for three possible configuration scenarios.

Figure 8.2 shows how virtual profiles and configuration commands are added to a virtual interface when a user dials in.

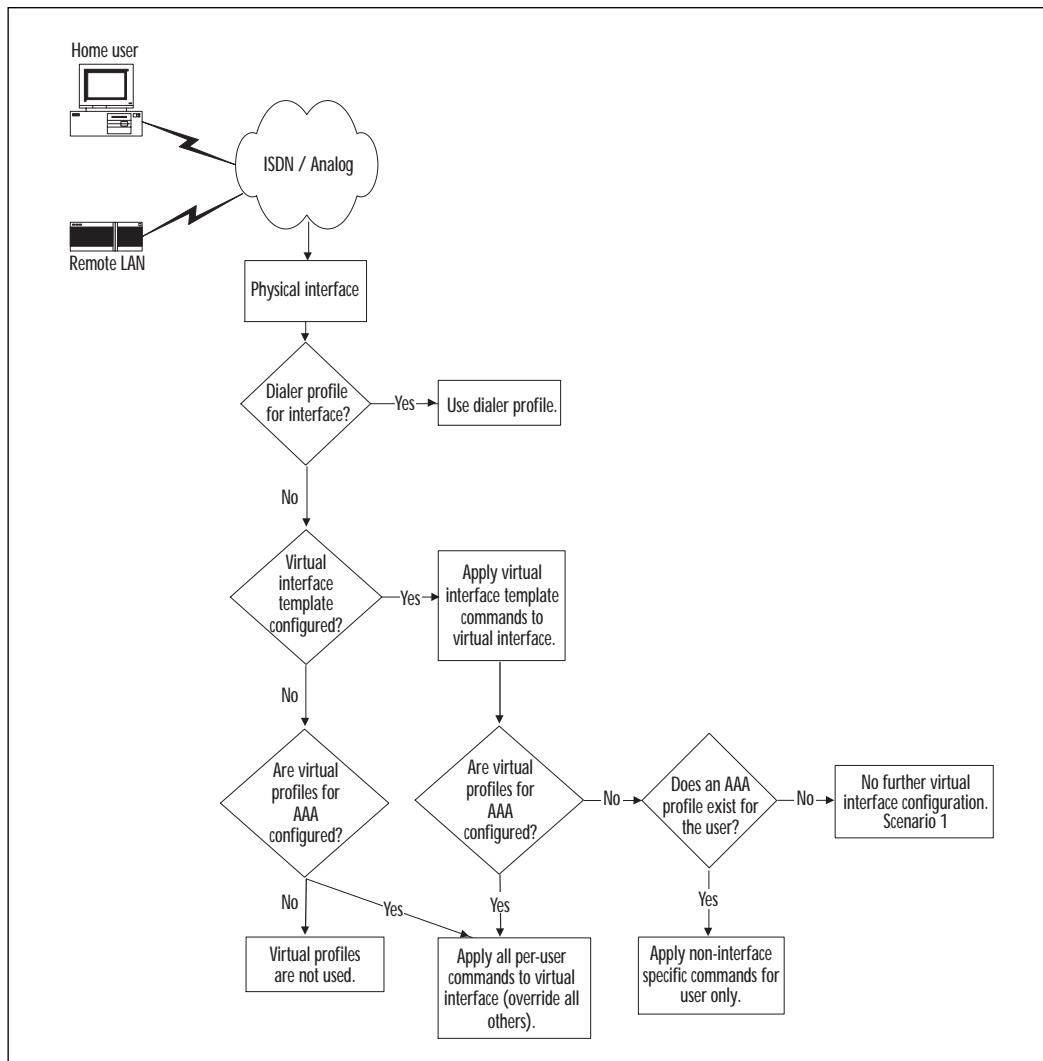
Scenario 1: Virtual template and subset of user configuration from AAA server are applied.

Scenario 2: All user configuration from AAA server is applied.

Scenario 3: Virtual template and all user configuration from AAA server are applied.

Scenario 1: Virtual Profiles Using Virtual Templates

This solution uses a combination of dialer profiles, virtual templates, and AAA user configuration. When using virtual profiles using virtual templates, the system checks to see if the physical interface is configured for dialer profiles; if it is, the router looks for a dialer profile for the user dialing in. If a dialer profile exists for this user, then it is used and the virtual profiles are not used. If a dialer profile for that user does not exist, the system uses a virtual template to create a virtual access interface for the user.

Figure 8.2 Virtual profile access process.

Scenario 2: Virtual Profiles Using AAA Configuration

This solution uses no dialer profiles or virtual templates; only virtual profiles by AAA are defined on the router. The AAA authorization response from a security server contains user-specific command-line configuration commands that are then applied to the interface. These virtual profile commands override existing configuration commands.

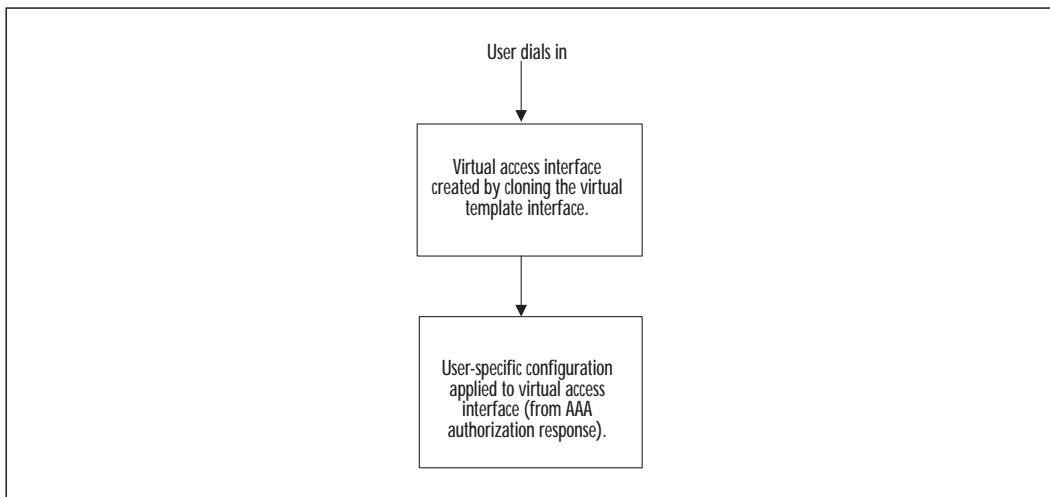
Scenario 3: Virtual Profiles Using Virtual Templates and AAA Configuration

No DDR dialer profile is defined for the user; a virtual template for virtual profiles is defined, virtual profiles by AAA are enabled on the router, and a per-user configuration entry for the user is defined on the AAA server.

The router dynamically creates a virtual access interface by cloning the virtual template defined for virtual profiles. The user-specific configuration received in the AAA authorization response is applied to the virtual access interface.

Figure 8.3 shows how virtual profiles are used to add user-specific commands to a virtual access interface when a user dials in.

Figure 8.3 Virtual profiles using virtual templates and AAA.



Configuring Virtual Profiles

There are several ways of using virtual profiles, depending on your specific needs. Each method requires different configuration commands.

Configuring Virtual Profiles Using Virtual Templates

A virtual template interface is a serial interface, and can therefore support all commands that may be applied to such an interface except **shutdown** and **dialer**.

Table 8.8 shows the commands necessary to configure a virtual interface and specify the interface to be used for virtual profiles.

Table 8.8 Configuring a Virtual Interface

Command	Description
router(config)#interface virtual-template <i>number</i>	Creates a virtual interface template and enters virtual template configuration mode.
router(config-if)#ip unnumbered ethernet 0	Enables IP without applying an IP address to the interface.
router(config-if)#encapsulation ppp	Enables PPP encapsulation.
router(config)#virtual-profile virtual-template <i>number</i>	Specifies the virtual template to be used for virtual profiles. The template number can range from 1 to 30.

Example of Virtual Profiles Using Virtual Templates

This code listing shows an example of how virtual profiles might be configured to support virtual templates on a typical router.

```
! Enable AAA
aaa new-model
aaa authentication ppp default tacacs
aaa authorization network tacacs
!
! Specify virtual-template 1 to be used for virtual profiles
virtual-profile virtual-template 1
!
! Configure virtual-template 1
interface virtual-template 1
ip unnumbered ethernet 0
encapsulation ppp
ppp authentication chap
!
interface serial 0
encapsulation ppp
no ip route-cache
```

```
ppp authentication chap
dialer in-band
dialer rotary-group 0
!
interface bri 0
encapsulation ppp
no ip route-cache
dialer rotary-group 0
ppp authentication chap
!
interface bri 1
encapsulation ppp
no ip route-cache
dialer pool-member 1
ppp authentication chap
!
interface dialer 0
ip address 10.26.1.1 255.255.255.0
encapsulation ppp
dialer in-band
no ip route-cache
dialer map ip 10.26.1.2 bud 1234
dialer map ip 10.26.1.3 simon 5678
dialer-group 1
ppp authentication chap
```

In the example above, users dialing in on interface serial 0 or bri 0 would have the virtual template interface applied to their virtual access interface. Any non-interface-specific configuration commands defined on the TACACS+ server for the user would also be applied. Interface bri 1 would not use virtual profiles as a dialer profile defined through the **dialer pool-member** command.

Configuring Virtual Profiles Using AAA Configuration

To use virtual profiles using AAA configuration, per-user configurations for each user must be defined on the AAA security server. This is discussed further in the “Per-user Configuration Example,” section of this chapter. AAA must be configured on the router, and AAA must be specified as the source of virtual profiles.

Table 8.9 details the command necessary to configure per-user configuration using AAA.

Table 8.9 Per-user Configuration Using AAA

Command	Description
router(config)#virtual-profile aaa	Specifies the source of the per-user configuration as AAA.

Example of Virtual Profiles Using AAA Configuration

This following router code shows that the virtual profile will use AAA for per-user configuration.

```
! Enable AAA
aaa new-model
aaa authentication ppp default tacacs
aaa authorization network tacacs
!
! Specify virtual profile configuration by AAA
virual-profiles aaa
!
```

Configuring Virtual Profiles Using Virtual Templates and AAA Configuration

As explained earlier, to use virtual profiles using AAA configuration, per-user configurations for each user must be defined on the AAA security server. AAA must be configured on the router, a virtual interface template must be defined and specified as a source of AAA virtual profiles, and AAA must be specified as a source of virtual profiles.

Table 8.10 details the commands necessary to configure virtual profiles using a combination of virtual templates and AAA.

Table 8.10 Virtual Profiles Using Virtual Templates and AAA

Command	Description
router(config)#interface virtual-template <i>number</i>	Creates a virtual interface template and enters virtual template configuration mode.
router(config-if)#ip unnumbered ethernet 0	Enables IP without applying an IP address to the interface.
router(config-if)#encapsulation ppp	Enables PPP encapsulation.
router(config)#virtual-profile virtual-template <i>number</i>	Specifies the virtual template to be used for virtual profiles. The template number can range from 1 to 30.
router(config)#virtual-profile aaa	Specifies the source of the per-user configuration as AAA.

Example of Virtual Profiles Using Virtual Templates and AAA Configuration

The following router configuration shows how a router might be configured to use both virtual templates and AAA for per-user configuration.

```
! Enable AAA
aaa new-model
aaa authentication ppp default tacacs
aaa authorization network tacacs
!
! Specify virtual-template 1 to be used for virtual profiles
virtual-profile virtual-template 1
! Specify that virtual profiles are to be used
virtual-profile aaa
!
! Configure virtual-template 1
interface virtual-template 1
ip unnumbered ethernet 0
encapsulation ppp
```

```

ppp authentication chap
!
interface bri0
encapsulation ppp
ppp authentication chap
no ip route-cache
!
```

In the example above, virtual profiles using both virtual templates and AAA configuration are defined. Users dialing into bri 0 will have the virtual interface configuration applied to their virtual access interface, and then if they have a user entry on the AAA server, their user-specific configuration will also be applied. Any configuration commands defined on the AAA server will override those of the virtual interface.

Per-User Configuration Example

As we have already seen, by using per-user configuration with virtual profiles we have a flexible and scalable solution for dial-in user access. The AAA authorization response holds all per-user configuration information (if any), formatted in AV pairs. The AV pairs available depend on the type of security server you choose to use.

The following example shows the application of a user named ‘remote’ dialing into a Cisco router named ‘central’; the virtual template interface is cloned to produce a unique virtual access interface, then further per-user configuration commands are applied to this interface.

User ‘Remote’ RADIUS Configuration

The following is the user’s configuration entry on a typical RADIUS server.

```

remote Password = "entry"
        User-Service-Type = Framed-User,
        Framed-Protocol = PPP,
        Cisco-avpair = "ip:route=40.0.0.0 255.0.0.0",
        Cisco-avpair = "ip:route=50.0.0.0 255.0.0.0",
        Cisco-avpair = "ip:inac1#2=10.26.2.1"
```

Network Access Server Configuration (Central)

The Cisco router at the central site is configured as follows.

```
version 11.2
service timestamps debug datetime localtime
service udp-small-servers
service tcp-small-servers
!
hostname central
!
aaa new-model
aaa authentication ppp default radius
aaa authorization network radius
enable secret 5 $1$IIN8$6BG9B9q8.Qi7mwBKDwF5D1
enable password digest
!
username remote password 0 entry
isdn switch-type basic-net3
!
interface Ethernet0
 ip address 10.26.1.1 255.255.255.0
 no ip mroute-cache
!
interface Virtual-Template1
 ip unnumbered Ethernet0
 no cdp enable
!
interface BRI0
 ip unnumbered Ethernet0
 no ip mroute-cache
 encapsulation ppp
 no ip route-cache
 dialer idle-timeout 300
 dialer map ip 10.26.2.1 name remote broadcast 20842254
dialer-group 1
```

```
no fair-queue
ppp authentication chap
!
no ip classless
ip route 0.0.0.0 0.0.0.0 10.26.1.254
!
virtual-profile vtemplate 1
dialer-list 1 protocol ip permit
radius-server host 10.26.1.10
radius-server key rabbit
```

The following debug shows the per-user configuration values being applied to the virtual-access interface configuration when the user dials in. The IP routes to networks 40.0.0.0/8 and 50.0.0.0/8 are added with a next hop of 10.26.2.1 (the IP address of the dialing-in interface), along with an access list denying traffic from 10.26.2.1.

```
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: (0): send AV
protocol=ip
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: (0): send AV
addr*10.26.2.1
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: (9876735263):
Method=RADIUS
*Jul 19 04:37:23: AAA/AUTHOR (9876735263)uthorization status = PASS_ADD
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV
service=ppp
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV
protocol=ip
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV
addr*10.26.2.1
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV
route=40.0.0.0 255.0.0.0
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV
route=50.0.0.0 255.0.0.0
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: Processing AV
inacl#5=deny 20.0.0.1
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: authorization
succeeded
```

```
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: done: her address  
20.0.0.1, we want 20.0.0.1  
*Jul 19 04:37:23: AAA/AUTHOR/IPCP: Virtual-Access1: authorization  
succeeded  
*Jul 19 04:37:23: AAA/AUTHOR: Virtual-Access1: parse_cmd 'ip route  
40.0.0.0 255.0.0.0 10.26.1.2' ok (0)  
*Jul 19 04:37:23: AAA/AUTHOR: Virtual-Access1: enqueue peruser IP  
txt=no ip route 40.0.0.0 255.0.0.0 10.26.2.1  
*Jul 19 04:37:23: AAA/AUTHOR: Virtual-Access1: parse_cmd 'ip route  
50.0.0.0 255.0.0.0 10.26.2.1' ok (0)  
*Jul 19 04:37:23: AAA/AUTHOR: Virtual-Access1: enqueue peruser IP  
txt=no ip route 50.0.0.0 255.0.0.0 10.26.2.1  
*Jul 19 04:37:23: AAA/AUTHOR: parse 'ip access-list standard Virtual-  
Access1#0' ok (0)  
*Jul 19 04:37:23: AAA/AUTHOR: parse 'deny 10.26.2.1' ok (0)
```

central# **show ip access-lists**

Standard IP access list Virtual-Access1#0 (per-user)

 deny 10.26.2.1

central# **show ip route**

Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B -
BGP

D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, * - candidate
default

U - per-user static route, o - ODR

Gateway of last resort is 10.26.1.254 to network 0.0.0.0

U 40.0.0.0/8 [1/0] via 10.26.2.1

U 50.0.0.0/8 [1/0] via 10.26.2.1

 10.26.2.0/24 is subnetted, 1 subnets

C 10.26.2.1 is directly connected, Virtual-Access1

 10.26.2.0/24 is subnetted, 1 subnets

C 10.26.1.1 is directly connected, Ethernet0

S* 0.0.0.0/0 [1/0] via 10.26.1.254

Monitoring and Verifying AAA Access Control

Because AAA is such a powerful method of securing your network resources, inappropriate configuration can cause serious problems for users trying to access those resources. It is therefore very important to be able to use the wide range of Cisco IOS commands available to monitor and resolve such problems. Cisco **debug** commands can be used to give detailed information on dynamic security processes, and **show** commands can be used to check current configuration values.

AAA Debug and Show Commands

debug ppp authentication will give detailed information on authentication transactions between the NAS and dial-in client. This is usually a good starting point if access is being denied by the NAS. In the following example you can see that the remote client ‘mark’ is successfully authenticating to a NAS named ‘3260’ via BRI0/0.

```
3620#
00:07:04: %LINK-3-UPDOWN: Interface BRI0/0:1, changed state to up
00:07:04: %ISDN-6-CONNECT: Interface BRI0/0:1 is now connected to unknown
00:07:04: BRI0/0:1 PPP: Treating connection as a callin
00:07:04: BRI0/0:1 CHAP: O CHALLENGE id 5 len 25 from "3620"
00:07:05: BRI0/0:1 CHAP: I RESPONSE id 5 len 25 from "mark"
00:07:06: BRI0/0:1 CHAP: O SUCCESS id 5 len 4
00:07:06: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
00:07:06: V1 PPP: Treating connection as a dedicated line
00:07:07: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0/0:1,
changed state to up
00:07:07: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-
Access1, changed state to up
00:07:10: %ISDN-6-CONNECT: Interface BRI0/0:1 is now connected to mark
3620#
```

debug aaa authentication shows the authentication process between a NAS and AAA security. It can be used with **debug ppp authentication** to locate the source of authentication problems.

debug aaa authorization gives information on how a NAS is trying to provide authorization to a user request. It gives information on the inter-

face the user is connecting to, the username, the resource requiring authorization, the method-list being used by the interface, and the actual methods that are used. It will also indicate if authorization is successful or not.

In the following example, you can see that the user ‘mark’ dials into BRI0/0 using PPP encapsulation. The interface identifies the ‘general’ method-list as being the network method-list for this interface. A RADIUS server then gives an authorization PASS reply to the requesting user.

```
3620#  
00:08:55: %LINK-3-UPDOWN: Interface BRI0/0:1, changed state to up  
00:08:55: %ISDN-6-CONNECT: Interface BRI0/0:1 is now connected to unknown  
00:08:56: BR0/0:1 AAA/AUTHOR/FSM: (0): LCP succeeds trivially  
00:08:56: AAA: parse name=BRI0/0:1 idb type=14 tty=-1  
00:08:56: AAA: name=BRI0/0:1 flags=0x55 type=2 shelf=0 slot=0 adapter=0  
port=0 channel=1  
00:08:56: AAA: parse name=<no string> idb type=-1 tty=-1  
00:08:56: AAA/MEMORY: create_user (0x61DD835C) user='mark' ruser=''  
port='BRI0/0  
:1' rem_addr='isdn/842633' authen_type=CHAP service=PPP priv=1  
00:08:58: BR0/0:1 AAA/AUTHOR/LCP: Authorize LCP  
00:08:58: BR0/0:1 AAA/AUTHOR/LCP (3064768274): Port='BRI0/0:1'  
list='general' service=NET  
00:08:58: AAA/AUTHOR/LCP: BR0/0:1 (3064768274) user='mark'  
00:08:58: BR0/0:1 AAA/AUTHOR/LCP (3064768274): send AV service=ppp  
00:08:58: BR0/0:1 AAA/AUTHOR/LCP (3064768274): send AV protocol=lcp  
00:08:58: BR0/0:1 AAA/AUTHOR/LCP (3064768274): found list "general"  
00:08:58: BR0/0:1 AAA/AUTHOR/LCP (3064768274): Method=radius (radius)  
00:08:58: BR0/0:1 AAA/AUTHOR (3064768274): Post authorization status =  
PASS_REPL  
00:08:58: BR0/0:1 AAA/AUTHOR/LCP: Processing AV service=ppp  
00:08:59: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up  
00:08:59: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0/0:1,  
changed state to up  
00:09:00: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-  
Access1, changed state to up  
00:09:01: %ISDN-6-CONNECT: Interface BRI0/0:1 is now connected to mark  
3620#
```

debug aaa accounting shows information on AAA accounting events as they occur.

debug virtual-template will give detailed information on how a virtual template interface is cloned to produce a virtual access interface when a user dials in. This is an extremely useful way to learn which commands are being bound to a virtual access interface, and in what order. This would be a good place to look when a virtual access interface is not behaving as expected.

```
3620#
00:13:20: %LINK-3-UPDOWN: Interface BRI0/0:1, changed state to up
00:13:20: %ISDN-6-CONNECT: Interface BRI0/0:1 is now connected to unknown
00:13:21: Vil VTEMPLATE: Reuse Vil, recycle queue size 0
00:13:21: Vil VTEMPLATE: Hardware address 0010.7b1b.c761
00:13:21: Vil VTEMPLATE: Has a new cloneblk vtemplate, now it has
vtemplate
00:13:21: Vil VTEMPLATE: ***** CLONE VACCESS1 *****
00:13:21: Vil VTEMPLATE: Clone from Virtual-Template1
interface Virtual-Access1
default ip address
no ip address
encap ppp
ip unnumbered Dialer5
no ip directed-broadcast
peer default ip address pool lab
end
```

debug tacacs gives more detailed information on security transactions with TACACS+ security server than either **debug aaa authentication** or **debug aaa authorization**. The output includes all TACACS+ packets exchanged, along with PASS or FAIL results.

debug radius is similar to the **debug tacacs** command and gives detailed information on RADIUS-specific transactions. The following output shows a successful RADIUS authentication request, and the exchange of RADIUS attributes.

```
00:14:18: RADIUS: Initial Transmit BRI0/0:1 id 8 10.26.2.1:1645,
Access-Request,
len 83
00:14:18:          Attribute 4 6 0A1A0202
```

```
00:14:18:      Attribute 5 6 00007531
00:14:18:      Attribute 61 6 00000002
00:14:18:      Attribute 1 6 6D61726B
00:14:18:      Attribute 30 8 38343236
00:14:18:      Attribute 3 19 09F5D352
00:14:18:      Attribute 6 6 00000002
00:14:18:      Attribute 7 6 00000001
00:14:18: RADIUS: Received from id 8 10.26.2.1:1645, Access-Accept, len
126
00:14:18:      Attribute 2 8 6A6F7264
00:14:18:      Attribute 6 6 00000002
00:14:18:      Attribute 7 6 00000001
00:14:18:      Attribute 26 38 0000000901062269
00:14:18:      Attribute 6 6 00000002
00:14:18:      Attribute 7 6 00000001
00:14:18:      Attribute 8 6 FFFFFFFE
00:14:18:      Attribute 18 30 0A417574
```

show interface virtual-access number shows the configuration of the virtual-access interface dynamically created when a user dials in. You can see from the following example that the IP address is displayed along with other protocol characteristics.

```
Virtual-Access1 is up, line protocol is up
Hardware is Virtual Access interface
Interface is unnumbered. Using address of Dialer5 (192.1.1.1)
MTU 1500 bytes, BW 100000 Kbit, DLY 100000 usec,
    reliability 255/255, txload 1/255, rxload 1/255
Encapsulation PPP, loopback not set
Keepalive set (10 sec)
DTR is pulsed for 5 seconds on reset
LCP Open
Open: IPCP
Last input never, output never, output hang never
Last clearing of "show interface" counters 00:01:08
Queueing strategy: fifo
Output queue 1/40, 0 drops; input queue 0/75, 0 drops
```

```
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    14 packets input, 580 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    27 packets output, 1062 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 output buffer failures, 0 output buffers swapped out
    0 carrier transitions
```

Walkthrough

The following router is configured to use most of the AAA functions we have discussed to provide secure remote access to a Microsoft Windows 95 or NT remote client. Commands relevant to AAA are annotated in the listing.

```
version 12.0
service timestamps debug uptime
service timestamps log uptime
no service password-encryption
!
hostname 3620
!
! configure the router for AAA services
aaa new-model
! create a default login authentication method-list using a TACACS+
! server, then a local database.
aaa authentication login default group tacacs+ local
! create an authentication method-list for PPP connections named
! 'general' using only TACACS+ for authentication
aaa authentication ppp general group tacacs+
! create an authorization method-list for network connections named
! 'general' using only TACACS+
aaa authorization network general group tacacs+
! create an accounting method-list for network activity reporting to a
```

```
! TACACS+ server. Events are reported when they begin and when they end.
aaa accounting network monitor start-stop group tacacs+
enable secret 5 $1$IIN8$6BG9B9q8.Qi7mwBKDwF5D1
enable password digest
!
username master password 0 letmein
!
ip subnet-zero
no ip domain-lookup
!
! specify that virtual templates are to be used for virtual profiles
virtual-profile virtual-template 1
isdn switch-type basic-net3
isdn voice-call-failure 0
cns event-service server
!
interface Loopback0
  ip address 1.1.1.1 255.255.255.255
  no ip directed-broadcast
!
interface Ethernet0/0
  ip address 10.26.2.2 255.255.255.0
  no ip directed-broadcast
!
interface Serial0/0
  no ip address
  no ip directed-broadcast
  shutdown
  no fair-queue
!
interface BRI0/0
  no ip address
  no ip directed-broadcast
  encapsulation ppp
```

```
no ip route-cache
no ip mroute-cache
dialer rotary-group 5
isdn switch-type basic-net3
!
interface TokenRing0/0
no ip address
no ip directed-broadcast
shutdown
ring-speed 16
!
! specify the configuration of the virtual template
interface Virtual-Template1
ip unnumbered Dialer5
no ip directed-broadcast
peer default ip address pool lab
!
interface Dialer5
ip address 192.1.1.1 255.255.255.0
no ip directed-broadcast
encapsulation ppp
no ip route-cache
no ip mroute-cache
dialer in-band
dialer-group 1
peer default ip address pool lab
! use the 'general' method-list for PPP authentication
ppp authentication chap general
! use the 'general' method-list for PPP authorization
ppp authorization general
! use the 'monitor' method-list for PPP accounting
ppp accounting monitor
!
ip local pool lab 192.1.1.10 192.1.1.20
```

```
no ip classless
no ip http server
!
dialer-list 1 protocol ip permit
!
! specify the IP address of the TACACS+ server to be used
tacacs-server host 10.26.2.1
! specify the shared secret to be used by the TACACS+ server and NAS
tacacs-server key rabbit
!
line con 0
  transport input none
line aux 0
line vty 0 4
  password forward
  transport input lat pad v120 mop telnet rlogin udptn nasi
!
end
```

The configuration above will use the TACACS+ server at address 10.26.2.1 for all authentication, authorization, and accounting processes. If a user dials in on BRI0/0, the ‘general’ authentication method-list will be used to authenticate the user. This will first try authentication via the TACACS+ server; if this fails, access will be denied. Any network operations the user attempts to perform will be authorized through the ‘general’ authorization method-list, again using the TACACS+ server. All networking processes used by that user will be reported to the TACACS+ server.

When a user successfully dials in, the interface virtual-template 1 is cloned to provide a virtual-access interface. Any per-user configuration commands held on the TACACS+ server are sent in the authorization reply packet. In this configuration, only non-interface-specific, per-user commands will be applied for the user.

The following debug shows a successful authentication and authorization of a Windows NT client dialing into a Cisco 3620. From this we can

see the user ‘mark’ is dialing into port BRI0/0, and that the TACACS+ server at IP address 10.26.1.1 is being used to provide authentication and authorization. We can see that virtual template 1 has been cloned as virtual access interface 1, and we can see the specific commands that have been applied to that interface. After this cloning takes place, the per-user configuration parameters are applied to the interface. Further down the configuration we can see that a start accounting message has been sent by the NAS and received by the TACACS+ server.

```
3620#  
00:58:52: %LINK-3-UPDOWN: Interface BRI0/0:1, changed state to up  
00:58:52: %ISDN-6-CONNECT: Interface BRI0/0:1 is now connected to unknown  
00:58:54: AAA: parse name=<no string> idb type=-1 tty=-1  
00:58:54: AAA/MEMORY: create_user (0x61D47724) user='mark' ruser=''  
port='BRI0/0'  
:1' rem_addr='isdn/842633' authen_type=CHAP service=PPP priv=1  
00:58:54: TAC+: send AUTHEN/START packet ver=193 id=3590112425  
00:58:54: TAC+: Using default tacacs server-group "tacacs+" list.  
00:58:54: TAC+: Opening TCP/IP to 10.26.2.1/49 timeout=5  
00:58:54: TAC+: Opened TCP/IP handle 0x61E6C798 to 10.26.2.1/49  
00:58:54: TAC+: 10.26.2.1 (3590112425) AUTHEN/START/LOGIN/CHAP queued  
00:58:54: TAC+: (3590112425) AUTHEN/START/LOGIN/CHAP processed  
00:58:54: TAC+: ver=193 id=3590112425 received AUTHEN status = PASS  
00:58:54: TAC+: Closing TCP/IP 0x61E6C798 connection to 10.26.2.1/49  
00:58:54: BRO/0:1 AAA/AUTHOR/LCP (3464581390): found list "general"  
00:58:54: AAA/AUTHOR/TAC+: (3464581390): user=mark  
00:58:54: AAA/AUTHOR/TAC+: (3464581390): send AV service=ppp  
00:58:54: AAA/AUTHOR/TAC+: (3464581390): send AV protocol=lcp  
00:58:54: TAC+: using previously set server 10.26.2.1 from group tacacs+  
00:58:54: TAC+: Opening TCP/IP to 10.26.2.1/49 timeout=5  
00:58:54: TAC+: Opened TCP/IP handle 0x61E6D654 to 10.26.2.1/49  
00:58:54: TAC+: Opened 10.26.2.1 index=1  
00:58:54: TAC+: 10.26.2.1 (3464581390) AUTHOR/START queued  
00:58:54: TAC+: (3464581390) AUTHOR/START processed  
00:58:54: TAC+: (3464581390): received author response status = PASS_ADD  
00:58:54: TAC+: Closing TCP/IP 0x61E6D654 connection to 10.26.2.1/49  
00:58:54: Vi1 VTEMPLATE: Reuse Vi1, recycle queue size 0
```

```
00:58:54: Vil VTEMPLATE: Hardware address 0010.7b1b.c761
00:58:54: Vil VTEMPLATE: Has a new cloneblk vtemplate, now it has
vtemplate
00:58:54: Vil VTEMPLATE: ***** CLONE VACCESS1 *****
00:58:54: Vil VTEMPLATE: Clone from Virtual-Template1
interface Virtual-Access1
default ip address
no ip address
encap ppp
ip unnumbered Dialer5
no ip directed-broadcast
peer default ip address pool lab
end

00:58:54: TAC+: using previously set server 10.26.2.1 from group tacacs+
00:58:54: TAC+: Opening TCP/IP to 10.26.2.1/49 timeout=5
00:58:54: %LINK-3-UPDOWN: Interface Virtual-Access1, changed state to up
00:58:54: Vil PPP: Treating connection as a dedicated line
00:58:54: TAC+: Opened TCP/IP handle 0x61E6CA14 to 10.26.2.1/49
00:58:54: TAC+: Opened 10.26.2.1 index=1
00:58:54: TAC+: 10.26.2.1 (4089160280) ACCT/REQUEST/START queued
00:58:55: Vil AAA/AUTHOR/PER-USER: Event IP_UP
00:58:55: Vil AAA/AUTHOR: IP_UP
00:58:55: Vil AAA/PER-USER: processing author params.
00:58:55: %LINEPROTO-5-UPDOWN: Line protocol on Interface BRI0/0:1,
changed state to up
00:58:55: %LINEPROTO-5-UPDOWN: Line protocol on Interface Virtual-
Access1, changed state to up
00:58:57: TAC+: (4089160280) ACCT/REQUEST/START processed
00:58:57: TAC+: (4089160280): received acct response status = SUCCESS
00:58:57: TAC+: Closing TCP/IP 0x61E6CA14 connection to 10.26.2.1/49
```

Summary

This chapter describes many of the more advanced security features of Cisco products. It shows the value of implementing a firewall to protect an organization's assets, and illustrates how the Cisco IOS Firewall Feature Set is a comprehensive security solution for current Cisco installations. The Cisco IOS Firewall Feature Set builds on existing IOS features, adds new security features, and provides a scalable and flexible router-based firewall solution.

The chapter describes various implementations of authentication, authorization, and accounting (AAA), and shows how network access servers (NAS) communicate with remote security servers to perform AAA functions. The two major security server protocols are RADIUS, which has been adopted by many ISPs as an industry standard, and TACACS+, which was developed by Cisco and includes many Cisco-proprietary features. The CiscoSecure product line is Cisco's own security server offering, and runs on Windows NT or UNIX—offering both RADIUS and TACACS+ support.

Detailed information on the definitions of authentication, authorization, and accounting is included, as well as details of their configuration on a Cisco network access server. Cisco TACACS+ and RADIUS configuration commands are included.

Finally, it describes how virtual profiles might be used to provide a unique per-user configuration for each dial-in user. This is a particularly powerful feature that uses attribute values stored in AAA authentication response packets, virtual-template configurations, or a combination of both to provide specific interface configuration for each user.

By using a combination of the security features outlined in the chapter, you can create a comprehensive, flexible, and scalable security solution that builds on existing Cisco security features.

FAQs

Q: Should I use TACACS+ or RADIUS on my security server?

A: That depends on your current network infrastructure. RADIUS is preferred by many as being the industry standard, and is perceived as being less vendor-specific, more feature-rich, and less resource-intensive than TACACS+. However, because TACACS+ is a Cisco proprietary protocol, it has more Cisco-specific features, and integrates fully with the Cisco IOS. TACACS+ offers improved security through full packet encryption, as well as multiprotocol support, separate AAA functions, and the ability to restrict commands executed on a Cisco IOS router.

Q: If I have both virtual profiles and dialer profiles configured, which will be used?

A: Dialer profiles will take precedence over virtual profiles. If a user had a dialer profile configured, it would be used; the virtual profile would then be ignored.

Q: Where can I find RADIUS and TACACS+ server software?

A: Lucent, Shiva, DEC, and Microsoft produce such software, along with Cisco's CiscoSecure product range.

Q: Why should I use AAA security services?

A: AAA separates authentication, authorization, and accounting into three distinct functions. This gives you a flexible and modular security solution that allows individual components to be altered without affecting the others. You can control access on a per-user, per-group, or per-service basis—allowing strict control of actions performed. AAA uses a variety of established security protocols such as RADIUS, TACACS+, and Kerberos to provide these services. AAA is also very scalable. Security servers may easily be added or removed, and access control features can simply be added when necessary. Also, AAA allows multiple security systems to serve the same groups of users. By replicating user information across these servers, you can provide redundancy among your security servers.

Q: How do I enable AAA on my network access server?

A: The **aaa new-model** global command enables AAA on the router. The **aaa authentication**, **aaa authorization**, and **aaa accounting** global commands will then enable each individual AAA feature, as discussed within this chapter. These features are applied to each line or interface you want to secure.

Q: What is a method-list?

A: A method-list is a sequenced list of authentication, authorization, or accounting methods. The system tries each entry in the list in order to provide the required AAA service. If the first method fails, the system tries the next until the list ends. If that happens, authentication or authorization is denied, or accounting is not performed.

Q: I want to use virtual profiles on my Cisco network access servers. What is the minimum Cisco IOS requirement?

A: Any IOS release supporting Multilink PPP with one of the following hardware platforms will support virtual profiles: Cisco 1003, 1004, 2500, and 4000 series; AS5200; 7000, 7200, and 7500 series.

Optimizing Network Performance with Queuing and Compression

Solutions in this chapter:

- WAN connection requirements
- WAN topology and specifications
- Network planning and design
- Considerations before installation
- Selecting Cisco access servers and routers
- Implementation considerations

Introduction

Today's networks are coping with ever-increasing traffic and applications that require more bandwidth and faster response times. As we start connecting these networks together and allow remote users to dial in and access them, we are unlikely to have unlimited bandwidth available, due to cost constraints. It is the network designer's job to ensure that the applications running across these links can maintain a satisfactory level of performance and responsiveness, as well as make efficient use of the available bandwidth.

To improve responsiveness in congested networks, Cisco has provided congestion management and avoidance techniques. Congestion management, or queuing features, include first-in, first-out queuing (FIFO), priority queuing (PQ), custom queuing (CQ), and weighted fair queuing (WFQ). IOS version 12.x also introduces a new class-based weighted fair queuing feature (CBWFQ), Versatile Interface Processor (VIP), and distributed weighted fair queuing (DWFQ) for the Cisco 7000 series products.

In addition, Cisco empowers network architects with congestion avoidance techniques. These mechanisms monitor the traffic load in an attempt to anticipate and avoid bottlenecks before they occur. This is accomplished using random early detection (RED) algorithms.

In this chapter, we will also look at using compression. Compression is an effective way to make more efficient use of bandwidth by reducing the amount of data that needs to be transmitted between endpoints. Cisco provides a number of different compression techniques and options, which will be covered in this chapter.

Network Performance

Managing congestion over wide area network (WAN) links is important due to the mismatch in speed between input ports (10 Mbps Ethernet) and output ports (56 Kbps serial link). One way that network devices can handle overflow of arriving traffic is to use a queuing algorithm to sort and prioritize outbound traffic, and then prioritize the traffic on the output link as indicated. It is important to note that queuing/prioritization works most effectively on WAN links that experience bursty traffic. If a WAN link is congested 100 percent of the time, queuing/prioritization may not remedy the issue—look to additional bandwidth instead.

The Cisco IOS software includes the following queuing tools:

- FIFO
- WFQ

- PQ
- CQ
- CBWFQ

Each queuing algorithm was designed to solve a specific network traffic problem and each will have a different effect on network performance. As described in the following sections, queuing is an effective way to control the order of traffic.

Queuing Overview

Many applications currently in use are of an interactive, transaction-based or time-sensitive nature. These applications are commonly referred to as *real-time* applications. An example of a real-time application is *Voice over X (VoX)*. VoX can refer to voice over IP, voice over Frame Relay, or voice over Asynchronous Transfer Mode (ATM). Voice traffic does not tolerate excessive delays because it is transported between endpoints. Therefore, Quality of Service (QoS) mechanisms need to be provisioned to reduce end-to-end delay or jitter.

Cisco routers route IP packets from input ports to output ports based on the most specific route entry found in the routing table. During periods when interface traffic volumes are low, packets traverse a given interface in a first-in, first-out manner. As packets arrive faster than they can be forwarded out of an interface, they are placed in a queue. Therefore, queuing happens when network congestion occurs (that is, the queue depth is greater than or equal to 1), otherwise all packets are forwarded out an interface as they arrive.

Various queuing methods have been developed and implemented for the Cisco series of routers. This chapter will explain how the queuing algorithms work, and how each method increases performance, allowing improved access to the outgoing interface. Queuing algorithms allow different traffic streams to be prioritized on network interfaces. These queuing algorithms can allow real-time traffic to be transmitted before other, less time-sensitive traffic. By using queuing techniques, the network manager can optimize network traffic flow resulting in better traffic management and support of all end-user applications.

Queuing Methods and Configuration

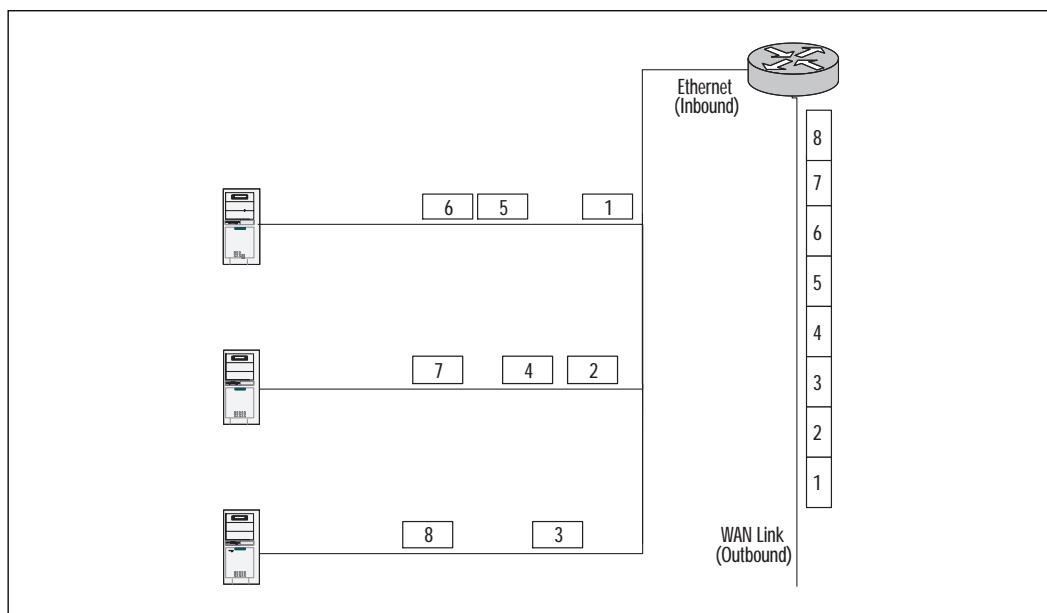
There are five network queuing techniques described in this chapter: FIFO, WFQ, PQ, CQ, and CBWFQ. Each of these queuing techniques has advantages and disadvantages pertaining to the design and configuration of each

individual network. We will examine the way each queuing method works, then develop a flowchart for selecting which queuing scheme should be enabled.

First-In, First-Out Queuing (FIFO)

The first queuing method is FIFO. Packets arrive in sequential order at the network interface. They are then inserted into the output buffer in the order in which they were received, and processed in the exact order that they arrive at the buffer. The packet buffer or processor on the interface does not give precedence to the type of packets or traffic arriving or when it needs to exit the interface. All packets exit the interface sequentially, in the same order in which they arrived. This is the default queuing method for all interfaces, except for serial interfaces operating at a rate of 2.048 Mbps and slower. Figure 9.1 illustrates FIFO queuing.

Figure 9.1 FIFO queuing.



When designing router hardware and software, a methodology had to be derived to allow all packet flows to have fair access to an outgoing interface. A packet flow can be described as a conversation between two end stations. Problems occur when large continuous packet transfers, sometimes called packet troops or packet trains (for example, a large file transfer), consume the majority of network resources and prevent other

traffic from using the link. (Under sustained heavy utilization, time-sensitive traffic like voice, video, and Telnet may not reach its destination in a timely manner. Failure to reach a destination on time may cause unacceptable user results. In theory, this file transfer could decrease its utilization of the network link and allow time-sensitive traffic fair access to interface bandwidth. The four queuing algorithms described in the next sections were implemented to give network managers the ability to balance interface bandwidth allocation between multiple applications and assign priority to mission-critical applications.

Weighted Fair Queuing (WFQ)

WFQ is a queuing method that automatically provides even allocation of bandwidth to high-bandwidth traffic flows, and prioritizes low-bandwidth connections to each network resource. This algorithm dynamically tracks traffic flows and allocates bandwidth accordingly. WFQ is the default queuing mechanism for all serial interfaces operating below 2.048 Mbps that do not use Linked Access Procedure, Balanced (LAPB), X.25, and Synchronous Data Link Control (SDLC) encapsulations.

WFQ interweaves low-volume traffic flows with high-volume traffic flows, resulting in the breakup of packet trains that restrict lower bandwidth traffic's access to network resources. WFQ automatically places interactive low-volume traffic at the front of the queue (to reduce response time) and allows high-volume traffic to compete for the remaining capacity. When WFQ is running in conjunction with Frame Relay, the algorithm will adjust the queuing schedule to compensate for link congestion, as identified by the receipt of forward explicit congestion notification (FECN) and backward explicit congestion notification (BECN) frames. This function is enabled by default and requires no manual configuration.

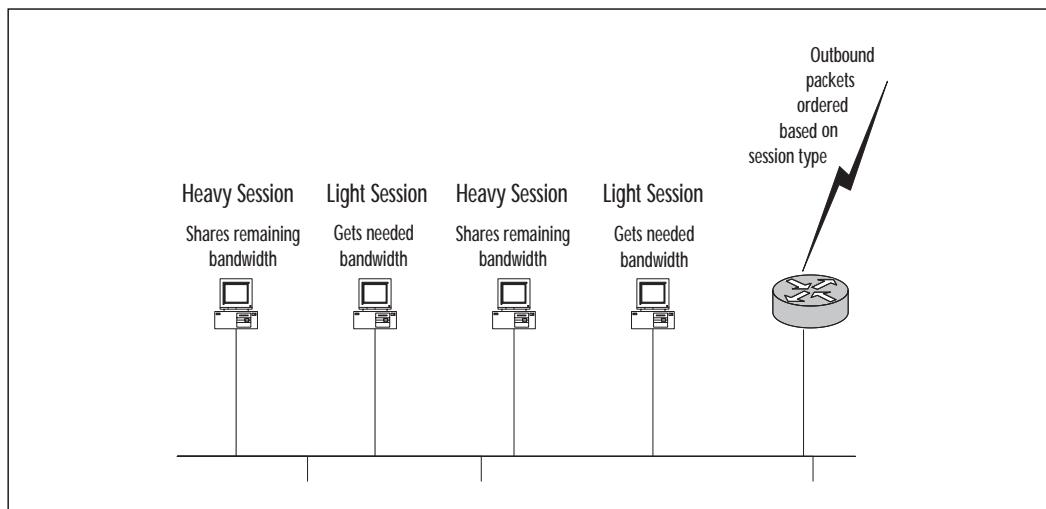
For example, assume we have a mid-sized hub-and-spoke network design topology for a national retail chain. The links between the hub and spokes are T1 circuits. Users primarily use FTP for batch processing and Telnet for access to their order entry system located at the hub site. Remote users have been complaining about intermittent response time problems resulting in a loss of revenue. Assuming the role of network operator, we suspect batch processing is degrading performance of the more interactive applications. By enabling WFQ, Telnet is automatically given priority over FTP, resulting in improved response time.

WFQ has three important user-configurable parameters. The interface command used to enable and configure WFQ is **fair queue**. It has several optional parameters such as congestive discard threshold, number of dynamic queues, and number of reservable queues.

The congestive discard threshold is set to 64 by default. This means that once 64 messages (packets) are queued as part of a flow, new packets

belonging to that flow will be discarded. A network manager/administrator can select to change this parameter to an integer based on a power of 2 from a range of 16 to 4096. Changing this parameter should be considered only after completion of a traffic analysis. If this parameter is changed, the router should be carefully monitored for memory issues. In most networks, this variable should remain unchanged.

Figure 9.2 WFQ.



Dynamic queues are used to support best-effort conversations. The default number of dynamic queues allocated is directly proportional to the configured interface bandwidth, as listed in Table 9.1:

Table 9.1 Allocation of Dynamic Queues

Bandwidth Range	Number of Dynamic Queues
Less than or equal to 64 Kbps	16
More than 64 Kbps and less than or equal to 128 Kbps	32
More than 128 Kbps and less than or equal to 256 Kbps	64
More than 256 Kbps and less than or equal to 512 Kbps	128
More than 512 Kbps	256

The last parameter (reservable queues) is used to define the number of flows reserved for features such as Resource Reservation Protocol (RSVP). The default value is determined by dividing the configured interface bandwidth value by 32 Kbps. The value can be statically defined as an integer from 0 to 1000. In practice, this value should not be changed unless an accurate traffic analysis has been performed. The following is an example of a serial interface being configured for WFQ using all default configuration values:

```
interface Serial0
    ip address 10.10.10.1 255.255.255.252
    fair-queue
```

The next example illustrates a serial interface configured for a congestive discard of 100 and 128 dynamic queues:

```
interface Serial0
    ip unnumbered Ethernet0
    bandwidth 384
    fair-queue 100
```

In summary, WFQ can identify and prioritize mixed traffic streams to more fairly allocate access to an interface rather than just servicing packets in FIFO fashion. WFQ is designed to minimize configuration efforts and automatically adapt to changing network traffic conditions.

Resource Reservation Protocol (RSVP)

RSVP is an Internet Protocol (IP) service that guarantees, or “reserves,” bandwidth across a network. RSVP is an ideal QoS method for real-time traffic (audio and video). Real-time traffic is consistent and very sensitive to latency; therefore, it requires a guaranteed network consistency. Without this consistency, there is risk of jitter, delay variations, and information loss due to insufficient bandwidth.

RSVP supports two types of real-time traffic: *multicast traffic*, primarily a flow in one direction from a single host sending packets to many hosts, and *unicast traffic*, for guaranteed bandwidth between two hosts.

There are three RSVP-supported reservation styles: wildcard-filter style, fixed-filter style, and shared-explicit style. A reservation style is a set of control options that specify a number of supported parameters. There are two groups of reservation styles: *distinct* and *shared*. A distinct reservation notes each individual flow, as in a video stream. A shared reservation notes a group of flows, as in an audio environment.

The three types of reservation styles are:

- **Wildcard-filter (WF) style** is a shared reservation style. A single reservation is created, into which flows from all upstream senders are mixed. The reservation is extended to new senders.
- **Fixed-filter (FF) style** is a distinct reservation style. A distinct reservation request is created for data packets from a particular sender.
- **Shared-explicit (SE) style** is a shared reservation style. A single reservation is created, into which flows from all upstream senders are mixed. The scope is explicitly specified by the receiver.

As discussed in the previous section, WFQ is RSVP-aware. The bandwidth reserved by WFQ can be statically defined or dynamically allocated.

For IT Professionals

Planning Considerations

How much bandwidth is needed for your application? If you are running VoIP and using a G.729a codec, then, depending on your configuration, you may need from 6.3 Kbps to 17.2 Kbps. As you can see, if you plan for 10 Kbps you could be shocked when it is time to test.

How much bandwidth is available? The default for a Cisco router is 75 percent of available bandwidth is reservable.

How much bandwidth is needed for other data traffic? You do not want to squelch your other traffic.

WFQ and IP Precedence

When queuing IP traffic, WFQ uses the IP precedence field from the QoS portion of the IP packet header in its algorithm to allocate bandwidth. The IP precedence bits are located in the type of service (TOS) field of an IP packet and have a value between 0 (default/low) and 7 (high). In practice, the precedence values of 6 and 7 are reserved. Please review Table 9.2.

As IP precedence values increase, the algorithm allocates more bandwidth to the flow. This results in higher-precedence traffic being served in the queue before lower-precedence traffic. Once the IP header has this value set, the value will traverse the network intact unless explicitly changed. This allows packets with higher precedence/priority to be serviced throughout a network (end-to-end) based on their IP precedence.

A benefit of using IP precedence is that WFQ is IP precedence-aware. The higher the value of IP precedence, the more bandwidth allocated to the IP traffic flow by WFQ. Non-real-time traffic flows normally have an IP precedence value of 0. Assigning real-time applications an IP precedence value greater than 0 ensures they will be serviced as high priority by the queuing algorithm.

Table 9.2 IP Precedence Values

Precedence Number	Value Name
0	Routine
1	Priority
2	Immediate
3	Flash
4	Flash-override
5	Critical
6	Internet
7	Network

The method that WFQ uses to calculate flow priority is complex. The following examples should help to simplify understanding.

In WFQ, each IP flow is given a percentage of the total interface bandwidth based on precedence level and the number of flows assigned to each precedence level. The following formula simplifies the issue:

Percentage of interface bandwidth assigned to a flow =

Precedence level + 1

The sum of [(each flow's precedence level + 1) *
(the number of flows at that precedence level)]

To further clarify, consider the following two examples. In the first example, our object is to determine what percentage of bandwidth is assigned each flow with a precedence value of 0 and 4. We have eight flows using precedence levels 0 through 7 with one flow allocated per precedence level.

$$\frac{\text{precedence level} + 1}{(0+1)*1 + (1+1)*1 + (2+1)*1 + (3+1)*1 + (4+1)*1 + (5+1)*1 + (6+1)*1 + (7+1)*1}$$

To determine the bandwidth for precedence 0, we will insert 0 for precedence level and calculate the lower half of the formula:

$$\frac{0+1}{1+2+3+4+5+6+7+8=36}$$

To determine the bandwidth for precedence 4, we will insert 4 for precedence level and calculate the lower half of the formula:

$$\frac{4+1}{1+2+3+4+5+6+7+8=36}$$

In the formulas above, precedence 0 traffic will be allocated 1/36 of the interface bandwidth and precedence 4 will receive 5/36 of the interface bandwidth.

In our next example, we have adjusted the formula to represent 12 traffic flows and three individual precedence levels. Our objective is to determine the amount of interface bandwidth assigned to a single flow at each precedence level.

Example criteria:

Five flows with a precedence of 0

Ten flows with a precedence of 2

Two flows with a precedence of 4

$$\frac{\text{precedence level}+1}{(0+1)^*5+(2+1)^*10+(4+1)^*2}$$

To determine the bandwidth for precedence 0, we will insert 0 for precedence level and calculate the lower half of the formula:

$$\frac{0+1}{(1^*5)+(3^*10)+(5^*2)=45}$$

To determine the bandwidth for precedence 2, we will insert 2 for precedence level and calculate the lower half of the formula:

$$\frac{2+1}{(1^*5)+(3^*10)+(5^*2)=45}$$

To determine the bandwidth for precedence 4, we will insert 4 for precedence level and calculate the lower half of the formula:

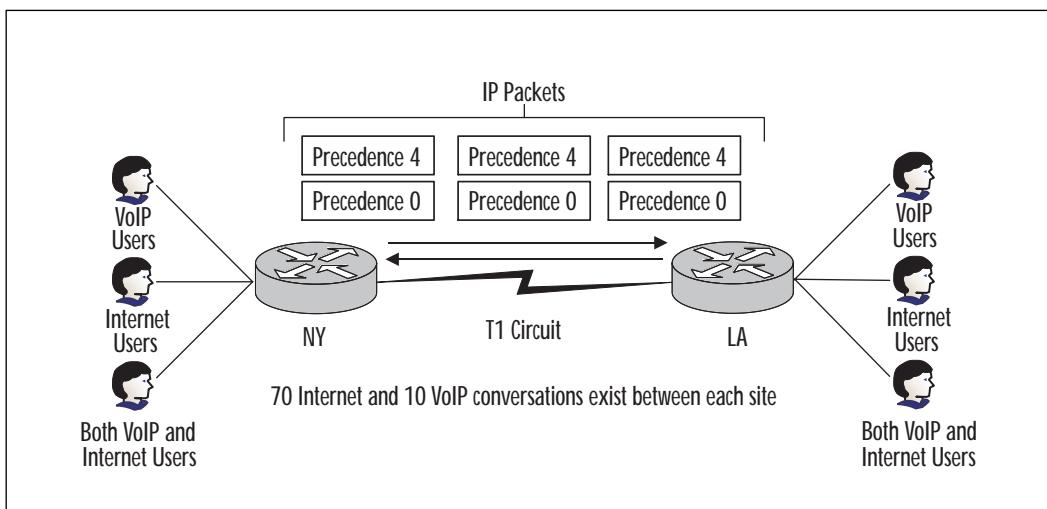
$$\frac{4+1}{(1*5)+(3*10)+(5*2)=45}$$

The output of the formula states that precedence 0 flows receive 1/45 of the interface bandwidth, precedence 2 flows receive 3/45 of the interface bandwidth, and precedence 4 flows receive 5/45 of the interface bandwidth.

For example, assume we have two locations interconnected via a T1 circuit, as illustrated in Figure 9.3. By default, WFQ is enabled on each WAN interface. Traffic is distributed between Voice over IP (VoIP) and Internet traffic, with 10 flows of VoIP traffic and 70 flows of Internet traffic, respectively.

All voice traffic has been assigned an IP precedence value of 4 and all Internet traffic a precedence value of 0. During periods of congestion, using the formula above, WFQ will allocate 1/120 of the interface bandwidth to each precedence 0 flow and 5/120 or 1/24 of the interface bandwidth to each Internet flow. This equates to about 64 Kbps per VoIP session and 12.8 Kbps per Internet session.

Figure 9.3 IP precedence used to allocate more bandwidth to voice traffic.



VIP DWFQ

VIP DWFQ can be described as the high-speed version of WFQ. This version requires the use of a Cisco 7000 series router using VIP2-40s or later

interface processors. Although the VIP2-40 is the minimum required interface processor to run DFWQ, it is recommended to deploy VIP2-50s when the aggregate port speed on the VIP exceeds 45 Mbps. In addition, distributed Cisco express forwarding (dCEF) is required to run DWFQ.

dCEF provides increased packet routing performance because the entire route forwarding information base (FIB) is resident on each VIP card. Therefore, routing table lookups happen locally on the VIP card without querying the centralized route switch processor.

In flow-based DWFQ, all traffic flows are equally weighted and guaranteed equal access to the queue. This queuing method guarantees fair access to all traffic streams, thus preventing any single flow from monopolizing resources.

To enable DWFQ, activate fair queuing by enabling “IP CEF” in global configuration mode and “fair-queue” under the VIP2 interface configuration.

Review the following example:

```
version 12.1
!
ip cef
!
interface FastEthernet0/0
  ip address 172.20.10.2 255.255.255.0
  full-duplex
!
interface Hssi4/0
  ip address 172.20.20.2 255.255.255.0
  fair-queue
!
router ospf 100
  network 172.20.0.0 0.0.255.255 area 0
!
router#
```

DWFQ also has the following limitations:

- Can be configured only on main interfaces; per IOS 12.1.0, there is no sub-interface support.

- Can be configured only on an ATM interface with AAL5SNAP encapsulation. Per IOS 12.1.0, there is no support for AAL5MUX or AAL5NLPID encapsulations.
- Is not supported on any virtual, tunnel, or Fast EtherChannel interfaces.
- Cannot be configured in conjunction with RSP-based WFQ, PQ, or CQ.

Priority Queuing (PQ)

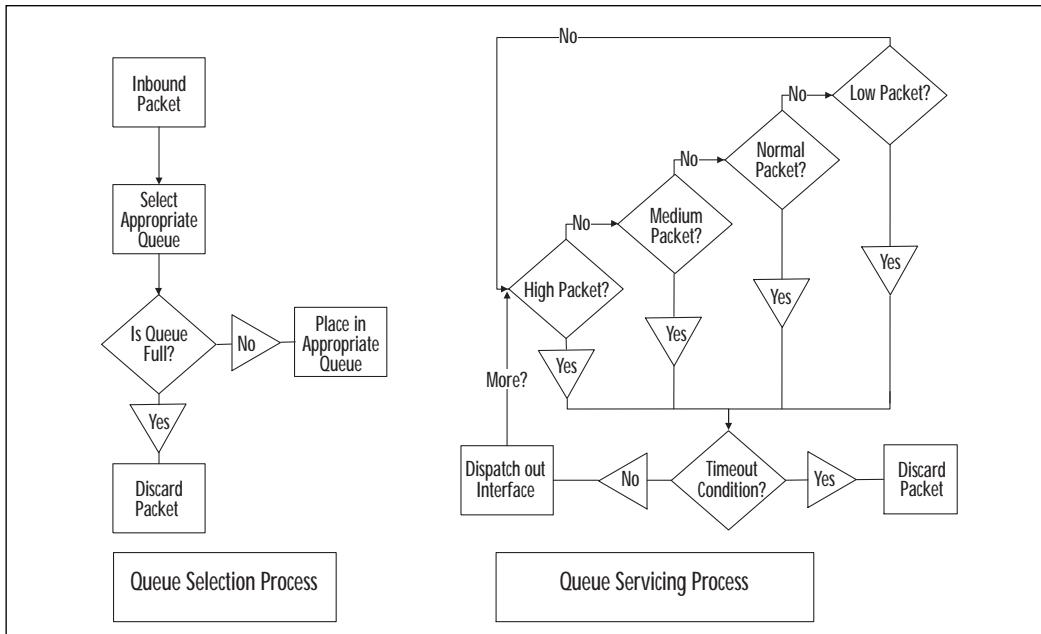
PQ provides a granular means for the network administrator to determine which traffic must be queued and serviced first. With priority queuing techniques, the network administrator must understand all the traffic flows within the network. This type of control is important when specific mission-critical traffic must receive servicing. The network administrator has the control to create different interface packet queues that are serviced in a hierarchical order. Each network flow can be categorized by the following:

- Protocol or sub-protocol type
- Incoming interface
- Packet size
- Fragments
- Access lists

The queues are known as high, medium, normal, and low. The router services the queues from highest to lowest priority. The service order on the four queues works such that if the high queue has traffic in it, the normal queue cannot forward any packets until all packets in the high-priority queue are transmitted. This is a major issue when designing a queuing strategy for a network. The network administrator may inadvertently starve a certain network stream, making users unable to use applications and services on the network. However, this may be ideal for networks in which critical applications are not able to run because network users are running “less important” applications.

Figure 9.4 illustrates the PQ packet flow.

When using PQ, packets are compared with a statically defined priority list. If there is any capacity in the priority queue associated with the incoming traffic, the packet is placed in the designated queue and waits to be serviced out the interface. If there is no room left in the queue, the packet is dropped.

Figure 9.4 PQ packet flow.

WARNING

Packets that are dropped do not go into another queue.

Since the definitions for queues are defined, a packet either fits into that queue, or it does not. Even though packets are sent into queues, there is no guarantee they will be processed in time to reach their destination. This process enables network administrators to control the priority of mission-critical network traffic, but also requires a good understanding of its effect on the flow of other network traffic. Networks implementing priority queuing require constant reassessment, since traffic pattern requirements may change as well. Traffic that was once considered high priority may become a low priority at some point.

It is important to note that priority queuing can affect CPU utilization. Cisco routers will process switch packets on interfaces that have priority queuing enabled. The packet-switching performance will be degraded compared with other interfaces using caching schemes. Also note that priority queuing is not supported on tunnel interfaces.

Priority Queuing Examples

In a mainframe environment, there may be a lot of users “surfing” the Web and downloading files, causing performance problems with time-sensitive Software Network Architecture (SNA) traffic and other tn3270 (Telnet) traffic. The following situation allows the SNA traffic (using Data-Link Switching (DLSw)) and the Telnet traffic to have high priority where the reset of traffic is considered low. There may be some exceptions that can be controlled using an access list to make a normal priority.

```
!  
priority-list 1 protocol ip normal list 100  
priority-list 1 protocol ip high tcp telnet  
priority-list 1 protocol dlsw high  
priority-list 1 default low  
!
```

To use an extended access list to make specific IP traffic have normal priority on the interface, the **priority-list 1 protocol ip normal list 100** command is used.

To configure Telnet traffic as high priority, the **priority-list 1 protocol ip high tcp telnet** command is used.

To configure DLSw traffic as high priority, the **priority-list 1 protocol dlsw high** command is used.

To configure traffic that does not match any of the previous statements, the **priority-list 1 default low** command will set a default priority. If no default queue is defined the normal queue is used.

```
!  
interface Serial0  
priority-group 1  
!
```

The interface **priority-group 1** command is configured under the whole interface to specify that priority list 1 is used for that interface.

```
c2507#show interface serial 0  
Serial0 is up, line protocol is up  
Hardware is HD64570  
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load  
1/255  
Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)
```

```

LMI enq sent 0, LMI stat recv 0, LMI upd recv 0, DTE LMI up
LMI enq recv 0, LMI stat sent 0, LMI upd sent 0
LMI DLCI 1023 LMI type is CISCO frame relay DTE
Broadcast queue 0/64, broadcasts sent/dropped 0/0, interface
broadcasts 0

Last input 00:00:03, output 00:00:03, output hang never
Last clearing of "show interface" counters 00:00:03
Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Queueing strategy: priority-list 1
Output queue (queue priority: size/max/drops):
    high: 0/20/0, medium: 0/40/0, normal: 0/60/0, low: 0/80/0
5 minute input rate 0 bits/sec, 0 packets/sec
5 minute output rate 0 bits/sec, 0 packets/sec
    0 packets input, 0 bytes, 0 no buffer
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
    0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
    0 packets output, 0 bytes, 0 underruns
    0 output errors, 0 collisions, 0 interface resets
    0 output buffer failures, 0 output buffers swapped out
    0 carrier transitions
    DCD=up  DSR=up  DTR=up  RTS=up  CTS=up

```

c2507#

Using the **show interface serial 0** command, the type of queuing is displayed on the queuing strategy line of the interface output. The syntax for queues is size/max/drops, where size is the current used depth of the queue, max is the maximum depth of the queue before packets are dropped, and drops is the number of packets dropped after the max has been reached. The size and drops reset to 0 when the counters are cleared.

```

!
priority-list 1 queue-limit 30 60 60 90
!
```

The command **priority-list 1 queue-limit <high> <med> <norm> <low>** configures the different queues to different depths.

Custom Queuing (CQ)

CQ is a method used to statically define your own queuing parameters. Before enabling CQ, a traffic analysis needs to be performed. To define CQ parameters you need to know the packet sizes being used for each application. This data is necessary to configure CQ effectively.

CQ is the next progression of PQ. It guarantees some level of service to all created queues. With PQ, you can end up servicing only your high priority queue and never service the low priority queue. CQ takes the other queues into consideration, allowing a percentage of the other queues' traffic to be processed. The percentage can be defined by the protocol, source/destination address, or incoming interface. This ability to assign a percentage of the output interface ensures that each queue will be serviced regularly and guaranteed some level of bandwidth.

There are 17 queues defined in CQ. Queue 0 is reserved for system messages such as keep alives and signaling, and queues 1 through 16 are available for custom configuration. The system queue is always serviced first. The algorithm will allow you to specify the number of bytes to be serviced by the queue and/or the number of packets to be forwarded by the queue before moving to the next sequential queue. The result is a queuing mechanism that services each queue sequentially for the predetermined byte and/or packet count before cycling to the next queue. Bandwidth to each queue is indirectly configured in terms of byte count and queue length. When using CQ, no application receives more bandwidth than configured in the custom queue under congestive conditions.

It is important to set the byte count parameters correctly to achieve predictable results. Assume that you want to engineer a custom queue that divides the effective interface bandwidth evenly across four different applications. Now, also assume that you have not performed any traffic analysis and have configured four CQs with a byte count of 250 under the assumption that all the applications are similar. Now suppose that each application transmits 100-, 300-, 500-, and 700-byte frames consecutively. The net result is not a 25/25/25/25 ratio. When the router services the first queue, it forwards three 100-byte packets; when it services the second queue, it forwards one 300-byte packet; when it services the third queue, it forwards one 500-byte packet; and when it services the fourth queue, it forwards one 700-byte packet. The result is an uneven distribution of traffic flowing through the queue. You must pre-determine the packet size used by each flow or you will not be able to configure your bandwidth allocations correctly.

To determine the bandwidth that a custom queue will receive, use the following formula:

(queue byte count / total byte count of all queues) * bandwidth capacity of the interface.

Custom Queuing Examples

In an environment where there is a low-speed serial connection handling all of the network traffic and more control over the different traffic types is necessary, CQ may be most suitable. In an environment where users are having problems getting Dynamic Host Configuration Protocol (DHCP) information when booting up, create a configuration that allows for DHCP traffic to have a higher priority. The following configuration shows Telnet and bootpc with the highest priority and an access list with the lowest priority.

```
!
queue-list 1 protocol ip 1 list 100
queue-list 1 protocol ip 2 tcp telnet
queue-list 1 protocol ip 3 udp bootpc
queue-list 1 default 4
!
```

To use an extended access list to make specific IP traffic flow into queue 1, the **queue-list 1 protocol 1 list 100** command is used.

To configure Telnet traffic to flow into queue 2, the **queue-list 1 protocol 2 tcp telnet** command is used.

To configure UDP bootpc to flow into queue 3, the **queue-list 1 protocol 3 udp bootpc** command is used.

For all other traffic not defined in any of the CQs, a default queue should be configured as in the **queue-list 1 default 4** command. If there is no default queue configured, the router will assume that queue 1 is the default.

```
!
queue-list 1 queue 1 byte-count 1000
queue-list 1 queue 2 byte-count 4000
queue-list 1 queue 3 byte-count 4000
queue-list 1 queue 4 byte-count 2000
!
```

Queue 1 has been configured for 1000 bytes to be drained per cycle, queue 2 has been configured for 4000 bytes, queue 3 has been configured

for 4000 bytes, and default queue 4 has been configured for 2000 bytes. Configuring the byte count of the different queues controls which queue has high priority. The higher the byte count, the more bandwidth is dedicated to that queue.

```
!  
interface Serial 0  
    custom-queue-list 1  
!
```

To apply CQ to a specific interface, the **custom-queue-list 1** command is used.

```
c2507# show interface serial 0  
Serial0 is up, line protocol is up  
    Hardware is HD64570  
    MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load  
    1/255  
    Encapsulation FRAME-RELAY, loopback not set, keepalive set (10 sec)  
    LMI enq sent 0, LMI stat recv 0, LMI upd recv 0, DTE LMI down  
    LMI enq recv 0, LMI stat sent 0, LMI upd sent 0  
    LMI DLCI 1023 LMI type is CISCO frame relay DTE  
    FR SVC disabled, LAPF state down  
    Broadcast queue 0/64, broadcasts sent/dropped 0/0, interface  
    broadcasts 0  
    Last input 00:00:07, output 00:00:07, output hang never  
    Last clearing of "show interface" counters 00:00:03  
    Input queue: 0/75/0 (size/max/drops); Total output drops: 0  
    Queueing strategy: custom-list 1  
    Output queues: (queue #: size/max/drops)  
        0: 0/20/0 1: 0/20/0 2: 0/20/0 3: 0/20/0 4: 0/20/0  
        5: 0/20/0 6: 0/20/0 7: 0/20/0 8: 0/20/0 9: 0/20/0  
        10: 0/20/0 11: 0/20/0 12: 0/20/0 13: 0/20/0 14: 0/20/0  
        15: 0/20/0 16: 0/20/0  
    5 minute input rate 0 bits/sec, 0 packets/sec  
    5 minute output rate 0 bits/sec, 0 packets/sec  
        0 packets input, 0 bytes, 0 no buffer  
    Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
```

```

0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
0 packets output, 0 bytes, 0 underruns
0 output errors, 0 collisions, 1 interface resets
0 output buffer failures, 0 output buffers swapped out
2 carrier transitions

DCD=up  DSR=up  DTR=up  RTS=up  CTS=uph

```

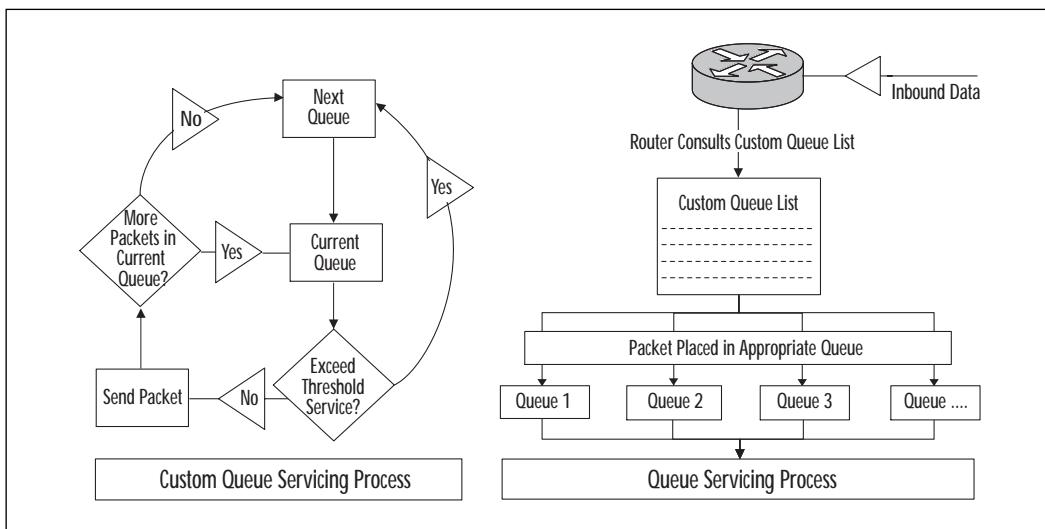
c2507#

```

!
queue-list 1 queue 1 limit 40
!
```

The **queue-list <list> queue <queue#> limit <depth>** command configures the queue depth for each custom queue.

Figure 9.5 The CQ servicing process.



Class-Based Weighted Fair Queuing (CBWFQ)

CBWFQ is an extended version of the standard WFQ functionality, with support for user-defined traffic classes added. With CBWFQ, the network administrator has the ability to separate traffic and place it into queues based on criteria such as protocol, access control lists (ACLs), or origi-

nating interface. Each packet is analyzed in an effort to match a defined traffic class. The packet is then forwarded to the appropriate queue for servicing.

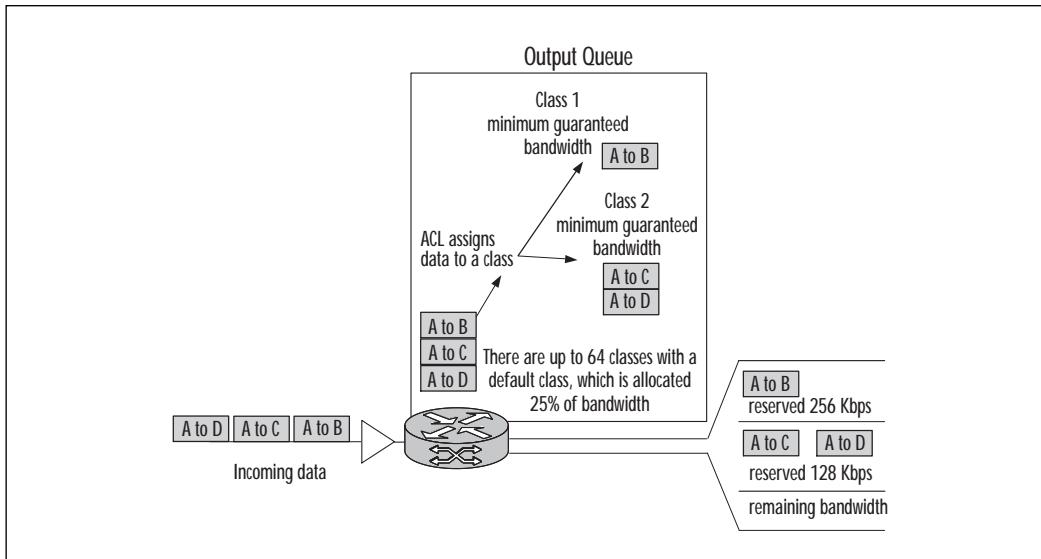
Classes are defined by parameters called *class characteristics*. Examples of class characteristics are bandwidth, weight, and maximum packet limit. The bandwidth assigned is the minimum bandwidth required for that specific class of service during periods of congestion. The weight value is derived from the bandwidth value assigned to each class. In addition, the weight value is used to help calculate the average queue length and packet limit. The packet limit defines the queue depth in packets. The queue is designed to drop all packets that exceed the configured queue depth or packet limit unless a policy is applied to the class. An example of such a policy is weighted random early detection (WRED), discussed later in this chapter.

CBWFQ does not allow more than 75 percent of the interface bandwidth to be assigned to classes. The additional 25 percent is reserved for overhead such as routing updates. The network administrator can override this threshold, but must first take into account all the bandwidth required for routing protocol updates.

A good example is an ATM-based interface. This network administrator would need to take into account the overhead required to package data into ATM cells at Layer 2, in addition to any control packet flows traversing the link.

The advantage to using CBWFQ is that it is not bound to packet flows. In CBWFQ, up to 64 classes can be defined to a more granular level than traditional WFQ. CBWFQ is not affected by the total number of flows traversing an interface, and classes do not compete for bandwidth with other classes. The caveat is that multiple flows can compete for bandwidth within a defined class; therefore, significant thought is required when defining your queuing strategy.

CBWFQ is not supported in conjunction with traffic shaping or ATM unspecified bit rate (UBR) permanent virtual circuits. Please review Figure 9.6, which illustrates CBWFQ operation. CBWFQ allocates bandwidth to a queue by guaranteeing the minimum amount of bandwidth defined for each class. There are 64 definable queues; WFQ is used to allocate bandwidth within each class or queue, unlike CQ, which services each queue defined in a FIFO manner.

Figure 9.6 CBWFQ.

Selecting a Cisco IOS Queuing Method

Steps 1 through 6 should be followed when determining which queuing option to implement:

1. Is the WAN link congested with network traffic? If there is no congestion on the link, there is no need to sort the traffic into queues. If the link is consistently congested, traffic queuing may not resolve the problem. If the link is only congested for short periods of time, queuing may resolve the flows.
2. What type of traffic is traversing the network and is it congested? The network administrator must learn traffic flows and study the link during peak usage. This will help determine what traffic is utilizing the link and what can be done with that traffic. The network administrator needs to determine whether control over individual streams has to be enforced and/or if generic protocols need to be queued to improve response time. Remember, traffic utilization is dynamic and will need to be analyzed often to determine whether changes are required.
3. After the traffic analysis is completed, can traffic be serviced by WFQ? This step is done to determine whether packet trains are utilizing the link during peak times. If so, automatic queuing provided by WFQ may be able to meet current needs. Remember,

traffic patterns are dynamic and subject to change. It is recommended that a regular traffic analysis be performed to determine whether queuing optimization is required.

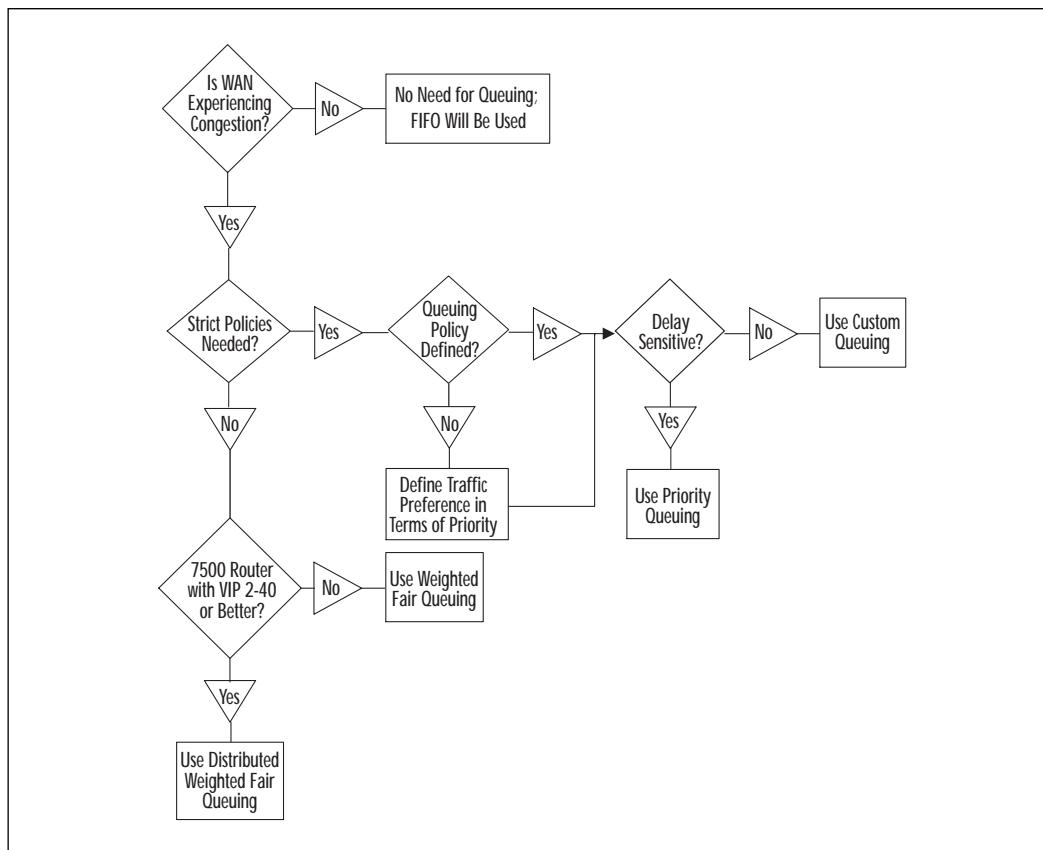
4. What is your organization's queuing policy? Queuing policies are based on application requirements in conjunction with a detailed traffic study. All interfaces require basic queuing configuration. These configuration values may need to be adjusted based on application requirement or location.
5. Does control over individual streams need to be taken into account? If certain applications are failing but enough bandwidth exists, CQ, WFQ, or CBWFQ can be utilized. This will allow the network administrator to select the critical traffic to be serviced while the other network flows will utilize the remaining bandwidth.
6. Can network delay be tolerated? If so, the network administrator can develop PQ schemes. The network administrator will need to determine which flows need servicing first and then determine how the other flows can be divided into the remaining queues. If the network cannot handle delays in packet arrival, then CQ can be used. CQ can guarantee that all applications gain some access to the link. Please review the queuing selection flow chart in Figure 9.7.

NOTE

When addressing congestion on links that have very low physical bandwidth, consider the amount of bandwidth being used by the routing protocol selected. For locations that are stub sites (have only one link connected to the backbone), consider using a default route or gateway of last resort. This will avoid the overhead associated with dynamic routing protocols.

Other things to consider are dynamic routing protocol selection, such as Routing Information Protocol (RIP) versus Open Shortest Path First (OSPF). Distance Vector protocols such as RIP will propagate the entire routing table every 30 seconds, requiring more bandwidth than link state protocols such as OSPF, which propagate changes in a given topology as they occur.

Table 9.3 provides a comparison of queuing techniques.

Figure 9.7 Queuing selection.**Table 9.3** Queuing Technique Selection

Weighted Fair Queuing	Priority Queuing	Custom Queuing
No queue lists	4 queues	16 queues
Low volume given priority	High queue serviced first	Round-robin service
Conversation dispatching	Packet dispatching	Threshold dispatching
Interactive traffic gets priority	Critical traffic gets through	Allocation of available bandwidth
File transfer gets balanced access	Designed for low-bandwidth links	Designed for higher-speed, low-bandwidth links
Enabled by default	Must configure	Must configure

Verifying Queuing Operation

To properly verify queuing operation, use the **show queuing** command to identify discards in both the input and output queues.

```
Router1#show queuing

Current fair queue configuration:
Interface Serial 0

Input queue: 0/75/0 (size/max/drops); Total output drops: 0
Output queue: 18/64/30 (size/threshold/drops)
    Conversations 2/8 (active/max active)
    Reserved Conversations 0/0 (allocated/max allocated)
    (depth/weight/discards) 3/4096/30
    Conversation 117, linktype: ip, length: 556, flags: 0x280
    source: 172.16.128.110, destination: 172.16.58.90, id: 0x1069, ttl:
59,
    TOS: 0 prot: 6, source port 514, destination port 1022
    (depth/weight/discards) 14/4096/0
    Conversation 150, linktype: ip, length: 1504, flags: 0x280
    source: 172.16.128.110, destination: 172.16.58.90, id: 0x104D, ttl:
59,
    TOS: 0 prot: 6, source port 20, destination port 1554
```

Weighted Random Early Detection (WRED) Overview

WRED is Cisco's version of RED. When this service is used, routers will attempt to anticipate and subsequently avoid network congestion. This differs from queuing techniques that attempt to control congestion after it has occurred on an interface.

RED is designed to make packet-switched networks aware of congestion before it becomes a problem. RED tries to control the average queue size while indicating to the end host if it should stop sending packets using Transmission Control Protocol's (TCP's) congestion control mechanisms.

RED will randomly drop packets during periods of high congestion. This action causes the source machine to decrease its transmission rate. Since TCP restarts quickly once a packet is lost, it can adapt its transmission rate to one the network can support.

RED is recommended only for TCP/IP networks. It is not recommended for protocols such as AppleTalk or Internetwork Packet Exchange/ Sequenced Packet Exchange (IPX/SPX), which respond to dropped packets by retransmitting the packets at the original rate.

Tail Drop

Tail dropping occurs when the egress queues become so congested that no more packets can enter the queue. These packets have nowhere to go so they are dropped from the tail end of the queue. Once packets start to tail-drop, the current network session will go to timeout mode. These timeouts can cause each sender to simultaneously retransmit. Since all TCP sessions restart at the same time, more packets get congested in the queue at approximately the same interval, essentially causing a cyclic effect. In other words, traffic can go through a wave of congestion that increases and decreases at regular intervals, and is commonly referred to as a *global synchronization problem*.

Weighted Random Early Detection (WRED)

WRED tries to overcome the problem seen with tail dropping by randomly discarding packets before the buffers get congested. WRED determines when to start dropping packets based on the average queue length. Once the packet count within the queue exceeds the defined upper queue threshold, WRED begins dropping packets in the upper queue range. The dropping of packets is totally indiscriminate to the network flow. Since packets are dropped at random within the queue, this causes only a few sessions to restart. This gives the network a chance to drain the queues. Since the remaining sessions are still flowing, the buffers can empty and allow other TCP sessions a chance to recover.

NOTE

WRED, CQ, PQ, and WFQ are mutually exclusive on an interface. The router software produces an error message if you configure WRED and any one of these queuing strategies simultaneously.

Flow-Based WRED

Flow-based WRED takes into account the types of packets and protocols it attempts to drop while keeping track of flow states. If it needs to drop any

flows, it will look for new flows within the queue rather than sacrificing a currently connected flow.

To allow for irregular bursty traffic, a scaling factor is applied to the common incoming flows. This value allows each active flow to reserve a number of packets in the output queue. The value is used for all currently active flows. When the scaling factor is exceeded, the probability of packets being dropped from the flow is increased.

Flow-based WRED provides a more fair method in determining which packets are tail-drops during periods of congestion. WRED automatically tracks flows to ensure that no single flow can monopolize resources. This is accomplished by actively monitoring traffic streams, learning which flows are not slowing down packet transmission, and fairly treating flows that do slow down packet transmission.

Data Compression Overview

Traffic optimization is a strategy that a network designer or operator seeks when trying to reduce the cost and prolong the link life of a WAN—in particular, improving link utilization and throughput. Many techniques are used to optimize traffic flow, which include PQs (as described earlier in this chapter), filters, and access lists. However, more effective techniques are found in data compression. Data compression can significantly reduce frame size and therefore reduce data travel time between endpoints. Some compression methods reduce the packet header size, while others reduce the payload. Moreover, these methods ensure that reconstruction of the frames happens correctly at the receiving end. The types of traffic and the network link type and speed need to be considered when selecting the data compression method to be applied. For example, data compression techniques used on voice and video differ from those applied to file transfers.

In the following sections, we will review these compression methods and explain the differences between them.

The Data Compression Mechanism

Data compression works by providing a coding scheme at both ends of a transmission link. The coding scheme at the sending end manipulates the data packets by replacing them with a reduced number of bits, which are reconstructed back to the original data stream at the receiving end without packet loss.

The scheme for data compression is referred to as a *lossless compression algorithm*, and is required by routers to transport data across the network. In comparison, voice and video compression schemes are referred to as *lossy* or *nonreversible compression*. The nature of voice or video data streams is that retransmission due to packet loss is not required. The latter type of compression allows for some degradation in return for greater

compression and, therefore, more benefits. The Cisco IOS supports teleconferencing standards such as Joint Photographic Experts Group (JPEG) and Moving Picture Experts Group (MPEG).

Lossless compression schemes use two basic encoding techniques:

- Statistical compression
- Dictionary compression

Statistical compression is a fixed, non-adaptive encoding scheme that suits single applications where data is consistent and predictable. Today's router environments are neither consistent nor predictable; therefore, this scheme is rarely used.

Dictionary compression is based on the Lempel-Ziv (LZ) algorithm, which uses a dynamically encoded dictionary to replace a continuous bit stream with codes. The symbols represented by the codes are stored in memory in a dictionary-style format. The code and the original symbol vary as the data patterns change. Hence, the dictionary changes to accommodate the varying needs of traffic. Dictionaries vary in size from 32,000 bytes to much larger, to accommodate higher compression optimization. The compression ratios are expressed as ratio $x:1$, where x is the number of input bytes divided by the number of output bytes.

Dictionary-based algorithms require the dictionaries at the sending and receiving ends to remain synchronized. Synchronization through the use of a reliable data link such as X.25 or a reliable Point-to-Point Protocol (PPP) mode ensures that transmission errors do not cause the dictionaries to diverge.

Additionally, dictionary-based algorithms are used in two modes—continuous and packet. Continuous mode refers to the ongoing monitoring of the character stream to create and maintain the dictionary. The data stream consists of multiple network protocols (for example, IP and DECnet). Synchronization of end dictionaries is therefore important. Packet mode, however, also monitors a continuous stream of characters to create and maintain dictionaries, but limits the stream to a single network packet. Therefore, the synchronization of dictionaries needs to occur only within the packet boundaries.

Header Compression

TCP/IP header compression is supported by the Cisco IOS, which adheres to the Van Jacobson algorithm defined in RFC 1144. This form of compression is most effective with data streams of smaller packets where the TCP/IP header is disproportionately large compared with the payload. Even though this can successfully reduce the amount of bandwidth required, it is quite CPU-intensive and not recommended for WAN links larger than 64 Kbps.

To enable TCP/IP header compression for Frame Relay encapsulation:

```
router(config-if)# frame-relay ip tcp header-compression [passive]
```

(for interface configuration). Or, on a per dlc basis:

```
router(config-if)# frame-relay map ip ip-address dlc [broadcast] cisco  
tcp header-compression {active | passive}
```

Another form of header compression, Real-time Transport Protocol (RTP), is used for carrying packets of audio and video traffic over an IP network, and provides the end-to-end network transport for audio, video, and other network services.

The minimal 12 bytes of the RTP header, combined with 20 bytes of IP header and 8 bytes of User Datagram Protocol (UDP) header, create a 40-byte IP/UDP/RTP header. The RTP packet has a payload of about 20 to 150 bytes for audio applications that use compressed payloads. This is clearly inefficient in that the header has the possibility of being twice the size of the payload. With RTP header compression, the 40-byte header can be compressed to a more reasonable 2 to 5 bytes.

To enable RTP header compression for PPP or high-data-rate digital subscriber line (HDSL) encapsulations:

```
router(config-if)# ip rtp header-compression [passive]
```

If the **passive** keyword is included, the software compresses outgoing RTP packets only if incoming RTP packets on the same interface are compressed. If the command is used without the **passive** keyword, the software compresses all RTP traffic.

To enable RTP header compression for Frame Relay encapsulation:

```
router(config-if)# frame-relay ip rtp header-compression [passive]  
router(config-if)# frame-relay map ip ip-address dlc [broadcast] rtp  
header-compression {active | passive}  
router(config-if)# frame-relay map ip ip-address dlc [broadcast]  
compress (enables both RTP and TCP header compression)
```

Link and Payload Compression

Variations of the LZ algorithm are used in many programs such as STAC (Lempel Ziv Stac, or LZS), ZIP and UNIX compress utilities. Cisco internetworking devices use the STAC (LZS) and Predictor compression algorithms. LZS is used on Cisco's Link Access Procedure, High-Level Data Link Control (HDLC), X.25, PPP, and Frame Relay encapsulation types. Predictor and Microsoft Point-to-Point Compression (MPPC) are only supported under PPP.

STAC (LZS) or Stacker was developed by STAC Electronics. This algorithm searches the input for redundant strings of data and replaces them with a token of shortened length. STAC uses the encoded dictionary method to store these string matches and tokens. This dictionary is then used to replace the redundant strings found in new data streams. The result is a reduced number of packets transmitted.

The Predictor compression algorithm tries to predict the incoming sequence of data stream by using an index to look up a sequence in the compression dictionary. The next sequence in the data stream is then checked for a match. If it matches, that sequence replaces the looked-up sequence in the dictionary. If not, the algorithm locates the next character sequence in the index and the process begins again. The index updates itself by hashing a few of the most recent character sequences from the input stream.

A third and more recent form of compression supported by Cisco IOS is MPPC. MPPC, as described under RFC 2118, is a PPP-optimized compression algorithm. MPPC, while it is an LZ-based algorithm, occurs in Layer 3 of the OSI model. This brings up issues of Layer 2 compression as used in modems today. Compressed data does not compress—it expands.

STAC, Predictor, and MPPC are supported on the 1000, 2500, 2600, 3600, 4000, 5200, 5300, 7200, and 7500 Cisco platforms. To configure software compression, use the **compress** interface configuration command. To disable compression on the interface, use the “no” form of this command, as illustrated below.

```
router(config-if)# compress {stac | predictor | mppc(ignore-pfc)}  
router(config-if)# no compress {stac | predictor | mppc(ignore-pfc)}
```

Another form of payload compression used on Frame Relay networks is FRF.9. FRF.9 is a compression mechanism for both switched virtual circuits (SVC) and permanent virtual circuits (PVC). Cisco currently supports FRF.9 mode 1 and is evaluating mode 2, which allows more parameter configuration flexibility during the LCP compression negotiation.

To enable FRF.9 compression on a Frame Relay interface:

```
router(config-if)# frame-relay payload-compress frf9 stac  
or  
router(config-if)# frame-relay map payload-compress frf9 stac
```

Per-Interface Compression (Link Compression)

This technique is used to handle larger packets and higher data rates. It is applied to the entire data stream to be transported—that is, it compresses the entire WAN link as if it were one application. The per-interface compression algorithm uses STAC or Predictor to compress the traffic, which in turn is encapsulated in a link protocol such as PPP or LAPB. This last step applies error correction and ensures packet sequencing.

Per-interface compression adds delay to the application at each router hop due to compression and decompression on every link between the endpoints. To unburden the router, external compression devices can be used. These devices take in serial data from the router, compress it, and send data out onto the WAN. Other compression hardware types are integrated on routers. Integrated compression software applies compression on existing serial interfaces. In this case, a router must have sufficient CPU and RAM for compression and dictionaries, respectively.

Per-Virtual Circuit Compression (Payload Compression)

Per-virtual circuit compression is usually used across virtual network services such as X.25 (Predictor or STAC) and Frame Relay (STAC). The header is unchanged during per-virtual circuit compression. The compression is therefore applied to the payload packets. It lends itself well to routers with a single interface but does not scale well in a scenario with multiple virtual circuit destinations (across a packet cloud).

Continuous-mode compression algorithms cannot be applied realistically due to the multiple dictionary requirements of the multiple virtual circuit destinations. In other words, it puts a heavy load on router memory. Therefore, packet-mode compression algorithms, which use fewer dictionaries and less memory, are more suited across packet networks.

Performing compression before or after WAN encapsulation on the serial interface is a consideration for the designer. Applying compression on an already encapsulated data payload reduces the packet size but not the number of packets. This suits Frame Relay and Switched Multimegabit Data Service (SMDS). In comparison, applying compression before WAN serial encapsulation will benefit the user from a cost perspective when using X.25, where service providers charge by the packet. This method reduces the number of packets transmitted over the WAN.

Hardware Compression

Cisco has developed hardware compression modules to take the burden of compression off of the primary CPU. On the 2600 and 3660 series of

routers there is an Advanced Integration Module (AIM) slot, which currently can be populated with compression modules. For the 7000, 7200, and 7500 series routers there are Compression Service Adapters (CSAs) that offload the compression from the primary CPU. Note that CSAs require a VIP2 model VIP2-40 or above and that the 7200 VXR series does not support CSA-based compression.

The 2600 can populate its AIM slot with an AIM-COMP2= and increase its compression capabilities from 256 Kbps to 8 Mbps of compressed data throughput. On the 3660, if you populate the AIM slot with an AIM-COMPR4= module, the 3660 detects an increase from 1024 Kbps to 16 Mbps.

There are two available modules for the 7000, 7200, and 7500 series routers: the SA-COMP/1 and the SA-COMP/4. Their function is identical, but the SA-COMP/4 has more memory to maintain a larger dictionary. The SA-COPMP/1 and SA-COMP/4, while supporting 16 Mbps of bandwidth, can support up to 64 and 256 compression contexts, respectively. One context is essentially one bi-directional reconstruction dictionary pair. This may be a point-to-point link or a point-to-point Frame Relay sub-interface.

Selecting a Cisco IOS Compression Method

Network managers look at WAN transmission improvements as one of their goals. Due to ever-increasing bandwidth requirements, capacity planning is key to maintaining good throughput and keeping congestion to a minimum. Capacity planners and network operators have to consider additional factors when trying to add compression to their arsenal. Below are some of the considerations.

- **CPU and memory utilization** When utilizing link compression, Predictor tends to use more memory, but STAC uses more CPU power. Payload compression uses more memory than link compression; however, link compression will be more CPU-intensive.
- **WAN topology** With the increased number of remote sites (more point-to-point connections), additional dedicated memory is required due to the increased number of dictionary-based compression algorithms.
- **Latency** Latency is increased when compression is applied to the data stream. It remains a function of the type of algorithm used and the router CPU power available.

NOTE

Encrypted data cannot be compressed; it will actually expand if run through a compression algorithm. By definition, encrypted data has no repetitive pattern.

Verifying Compression Operation

To verify and monitor the various compression techniques, use the following Cisco commands:

For IP header compression:

```
router# show ip tcp header-compression  
router# debug ip tcp header-compression
```

For RTP header compression:

```
router# show ip rtp header-compression  
router# debug ip rtp header-compression  
router# debug ip rtp packets
```

For payload compression:

```
router# show compress {detail-ccp}  
router# debug compress
```

Summary

As a network grows in size and complexity, managing large amounts of traffic is key to maintaining good performance. Some of the many considerations in improving application performance and throughput are compression, queuing, and congestive avoidance techniques.

When selecting a queuing or congestion-avoidance algorithm, it is best to first perform a traffic analysis to better understand the packet size, latency, and end-to-end flow requirements for each application. Armed with this information, network administrators can select the best QoS mechanism for their specific environment.

There are three viable compression methods to increase network performance: header, payload, and link. These use various algorithms such as Van Jacobson algorithm for header compression, STAC, and Predictor for

the payloads and link compression. Hardware compression modules are used in the routers to offload CPU processing due to the heavy burden of compression algorithms.

FAQs

Q: Where can I find more information about queuing and QoS?

A: You can start online at Cisco's Web site: www.cisco.com/univercd/cc/td/doc/cisintwk/ito_doc/qos.htm

Some related RFCs are:

RFC 2309: Recommendations on Queue Management and Congestion Avoidance in the Internet

RFC 2212: Specification of Guaranteed Quality of Service

RFC 1633: Integrated Services in the Internet Architecture: An Overview

Q: Are there any basic rules of thumb or "gotchas" that affect congestion management technologies?

A: Yes, some common rules of thumb are:

1. WFQ will not work on interfaces using LAPB, X.25, Compressed PPP, or SDLC encapsulations.
2. If the WAN link's average bandwidth utilization is 80 percent or more, additional bandwidth may be more appropriate than implementing a queuing policy.

Q: How can I verify queue operation?

A: The following debug commands can be useful (note that performing debug on a production router should be carefully weighed and the potential repercussions analyzed beforehand):

```
debug custom-queue  
debug priority
```

Q: How can I verify queue operation?

A: The following show commands can be useful:

```
show queue <interface and #>  
show queuing
```

where, for example, interface and # could stand for Ethernet 0.

Q: If both CBWFQ and CQ are available, which one should I use?

A: It is preferred you use CBWFQ over CQ because it will perform WFQ within each class-based queue. In other words, interactive applications such as Telnet are serviced before more bandwidth-intensive traffic within each statically defined queue. This results in better user response time than a custom queue using a FIFO method of draining the queue.

Q: When selecting a compression method, should I use hardware or software compression?

A: Use hardware compression over software compression when possible. Software compression can effect CPU utilization and needs to be monitored accordingly to avoid performance degradation. Hardware-based compression modules offload the main CPU by performing compression on a separate processing card. The end result is improved performance and throughput.

Requirements for Network Address Translation in Remote Access Networks

Solutions in this chapter:

- Network Address Translation (NAT) overview
- Translating inside source addresses
- Address overloading
- Overlapping address
- Transmission Control Protocol (TCP) load distribution
- NAT timeouts
- NAT to an Internet service provider (ISP)

Introduction

In this chapter we will be looking at Network Address Translation (NAT) and why it can be essential to today's remote access networks. With more and more organizations connecting to the Internet, two key problems are IP address depletion and scaling in routing. As it is not currently possible to allocate enough globally unique IP addresses to every organization whose systems require access to the Internet, other solutions are being developed.

NAT is a feature within the Cisco IOS that permits an organization's IP address structure to appear differently to outside networks than the actual address space it is using. This allows organizations to connect to the Internet without having to use globally unique addressing schemes internally.

Another challenge that can face today's network administrators is overlapping networks. Following mergers and acquisitions, or when simply requiring to connect to a partner organization, it is possible that both organizations may be using the same address space. NAT can help overcome this problem without the need for renumbering IP addresses.

NAT Overview

Over the past few years, available registered IP addresses have become increasingly scarce. Companies have been required to either reserve many small blocks of IP subnets or use addresses from the reserved block, as outlined in RFC 1918. The security of these addresses is also a key concern for companies as they are forced to devise mechanisms to avoid advertising internal Intranet addresses to the Internet. NAT is a solution to both of these problems. It can be used to translate addresses between private Intranets and public Internets. A company can use RFC 1918 addresses internally and use NAT to access the Internet. In this manner, only a few registered addresses are required from the ISP, and IP address depletion within an organization becomes a non-issue. The NAT router will be responsible for translating all internal non-registered addresses to one or more registered addresses. In achieving this, the organization has also protected their internal IP addressing scheme from being broadcast out to the Internet, thus providing an added layer of network security.

The following three address blocks are reserved for use on private networks (see RFC 1918):

10.0.0.0–10.255.255.255 (255.0.0.0 Subnet Mask)

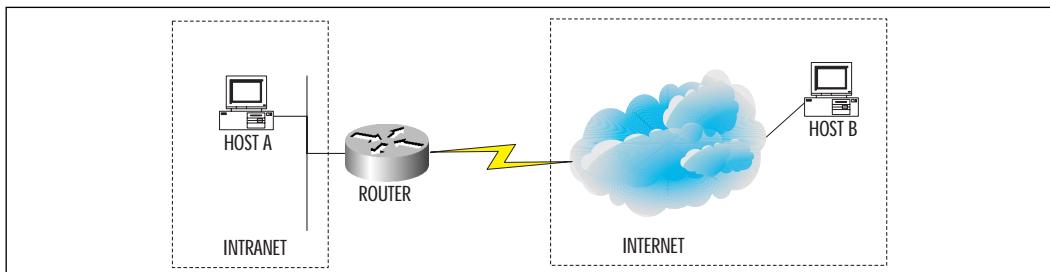
172.16.0.0–172.31.255.255 (255.240.0.0 Subnet Mask)

192.168.0.0–192.168.255.255 (255.255.0.0 Subnet Mask)

NAT converts IP addresses from the private address space to the public address space. When a device performing NAT receives a packet from the Intranet, it changes the source IP address, recomputes the appropriate checksums, and sends it to the Internet. In this fashion, anyone receiving the packet on the Internet will not be able to determine the original sender's IP address.

In Figure 10.1, Host A is on the Intranet, Host B is on the Internet, and the router is performing NAT translations. When Host A has data to send to Host B, the router will use NAT to translate Host A's IP address to an address from the public address space, and then forward the data to Host B. Host B will think it is communicating with the router and not with a host behind the router. All traffic from Host B will be directed to the router, and the router will forward the data to Host A.

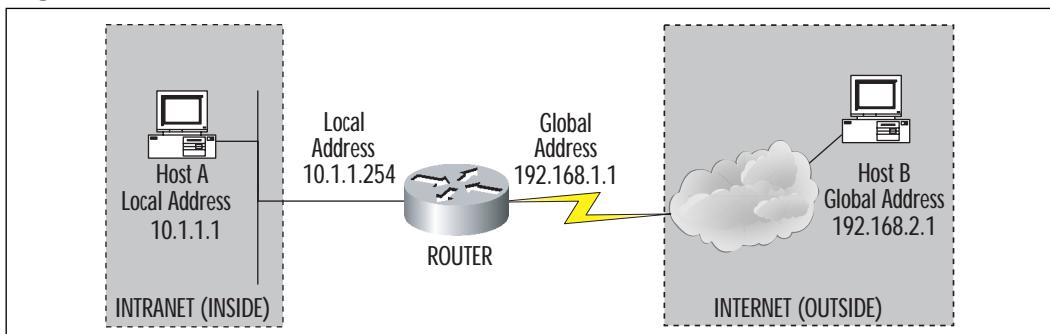
Figure 10.1 NAT overview.



Terminology

Cisco uses specific NAT terminology for referring to hosts in the Intranet and the Internet, both prior to translating and after translating. Figure 10.2 illustrates those terms. Host A is on the inside of an organization and the router is running NAT and connects to the Internet. Host A is communicating with Host B on the Internet.

Figure 10.2 NAT terminology.



The following list highlights each of the components in Figure 10.2.

- **Inside** The administrative domain that is controlled by an organization (the Intranet). This includes all hosts, servers, and networks that are internal to a company, such as Host A.
- **Outside** The administrative domain that is not controlled by an organization (the Internet).
- **Local address** An IP address that is used within the company and is the address that will be NAT-translated. Host A's local address is 10.1.1.1, and the router's local address is 10.1.1.254. These addresses will not traverse the Internet (outside) and therefore are considered local to (or inside) an organization.
- **Global address** A registered and legitimate IP address that can traverse the Internet. The diagram shows that there are two global addresses: the router's global address is 192.168.1.1 and Host B's global address is 192.168.2.1.
- **Inside local address** An IP address that is assigned to a host residing on the inside. This address can be either a registered IP address assigned by the ISP or Network Information Center (NIC), or an IP address assigned from RFC 1918. Host A has an inside local address. Note that the inside local address is the same as the local address.
- **Inside global address** An IP address that is allocated by either the ISP or NIC that's assigned to an *inside local address* (see definition above) after a NAT translation. This is the IP address of the inside host or hosts as it appears on the outside network. When Host A communicates with Host B, the router will assign Host A a registered global address to use over the Internet. It is possible to configure the NAT router to assign Host A an IP address of 192.168.1.1, thus making it seem as if all conversations are being sourced from the router. Please see the "Address Overloading" section later in this chapter for more information.
- **Outside local address** An IP address of a host on the outside as it appears on the inside after a NAT translation. Host B's IP address (192.168.2.1) can be NAT-translated to a different IP address prior to traversing the inside network. This IP address can be in the same address pool as the company's internal IP addresses. This makes it seem as if Host B is on the inside of a network instead of

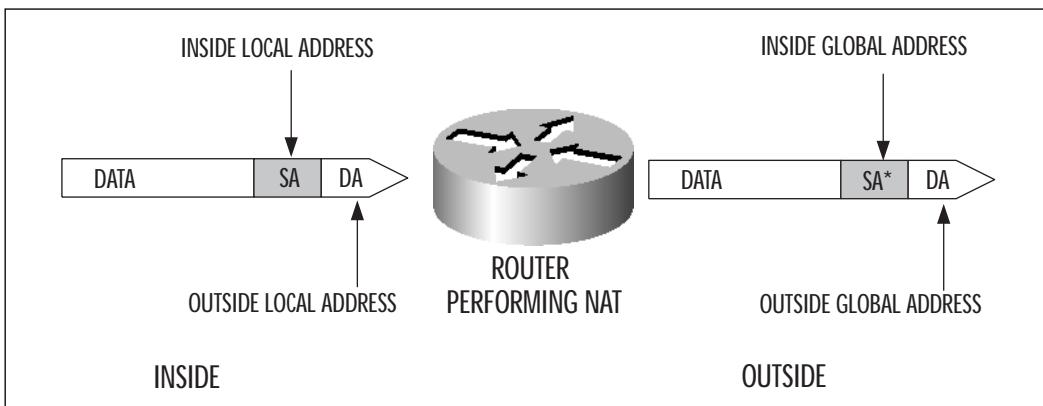
the outside. The hosts on the inside do not even realize that Host B is really located on the outside.

- **Outside global address** An IP address of a host on the outside. Host B's outside global address is 192.168.2.1.

NAT Operation

Figure 10.3 illustrates a router performing NAT translation on a packet being sent from the inside to the outside. The source address of the packet on the inside is depicted as SA, and the source address after the NAT translation is shown as SA*. SA is known as the inside local address, and SA* is the inside global address. The router actually serves two purposes: first, to translate all inside addresses to outside addresses and second, to translate outside addresses to inside addresses. The network engineer has the flexibility of configuring the router to convert all inside addresses and mapping them to one global address (known as Address Overloading or Port Address Translation (PAT), whereby making it seem as if all traffic is being sourced from one host; or reserving a pool of registered addresses on the router to use for conversion. Note in the diagram that SA* can either be the IP address of the router's physical interface or any other IP address that the engineer has configured.

Figure 10.3 Packet conversion through a NAT router.



The router maintains a table of all NAT translations, which is continually updated as new connections are made and old connections are timed out. The timeout parameters can be configured on the router and typically range from minutes to hours of inactivity. IP address timeouts are necessary because they ensure that the router can reallocate these addresses to

other hosts. If there were no timeouts, it is easy to see how quickly router resources would be depleted.

NOTE

In the above discussion, the NAT router is being used to translate addresses between the inside and the outside. However, NAT can also be used internally to a company's own organization; in fact, a NAT router can be used between any two routers where address translation is required. This also implies that the IP subnets being translated can belong to any address space and may or may not be registered.

Traffic Types Supported

NAT was first supported in the Cisco IOS release 11.2 plus image. The base image did not provide support for any NAT features; however, address overloading (PAT) was added to the base image starting with release 11.3, and full NAT functionality was added in release 12.0.

Phase 1 of Easy IP was available in release 11.3 and phase 2 from 12.0T.

Multiple hardware platforms were supported with each release of the Cisco IOS. A complete list can be obtained from the Cisco Web site at www.cisco.com.

The Cisco IOS NAT function supports multiple traffic types and protocols. Any Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) data stream that does not carry any source or destination IP addresses in the application layer can be NAT-translated. Additionally, native support is provided for Hypertext Transfer Protocol (HTTP), Trivial File Transfer Protocol (TFTP), File Transfer Protocol (FTP), Telnet, archie, finger, Network File System (NFS), rlogin, csh, Internet Control Message Protocol (ICMP), IP Multicast, and many others. Note that NAT not only translates IP addresses at the network layer (in the IOS model), but also translates application-level embedded IP addresses, such as for FTP. Applications that cannot be translated, include routing table updates, Simple Network Management Protocol (SNMP), Domain Name Server (DNS) zone transfers, Bootstrap Protocol (BOOTP), and others.

NAT Commands

Several commands are available to monitor, maintain, and troubleshoot NAT. The list below outlines the majority of the commands and will be used in examples throughout the chapter. There are different commands to show NAT translations and statistics, clear NAT translations, and perform extensive troubleshooting using the debug commands.

1. Clear all dynamic NAT translations from the NAT table before they timeout.

```
router prompt> clear ip nat translation *
```

2. Clear a dynamic translation that contains an inside translation.

```
router prompt> clear ip nat translation inside global-ip local-ip
```

3. Clear a dynamic translation entry containing an outside translation.

```
router prompt> clear ip nat translation outside local-ip global-ip
```

4. Clear a PAT translation.

```
router prompt> clear ip nat translation protocol inside global-ip
global-port local-ip local-port [outside local-ip local-port global-ip
global-port]
```

5. Display all active NAT translations. The verbose option displays how long ago the translation was created and used.

```
router prompt> show ip nat translations [verbose]
```

6. Display all NAT translation statistics, such as what is configured as the outside and inside interfaces, the total number of translations, IP address pools, and so on.

```
router prompt> show ip nat statistics
```

7. Debug IP NAT translations. This command can be used to display information about every packet that is NAT-translated. The *access list* option is the number of a standard access list that defines a set of IP addresses to be included in the debug. The *detailed* option provides a description of each packet considered for NAT translation, error information, and failure conditions.

```
router prompt> debug ip nat [access-list|detailed]
```

8. Display PAT statistics and the active sessions on a 700 series router.

```
router prompt> show ip pat
```

Translate Inside Source Addresses

This section discusses how to use static and dynamic NAT translations, and also addresses overloading to convert inside local addresses.

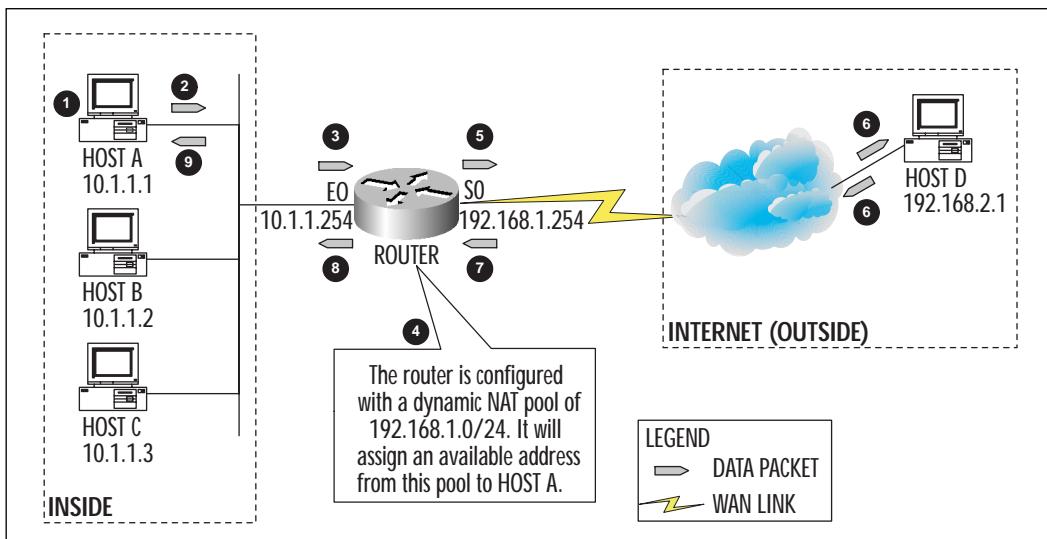
Dynamic Translation

Suppose a company wants its employees to be able to access the Internet (the outside), but it has a limited set of global IP addresses to assign to everyone on the inside. In this case, only those employees who have a global address can access the Internet. As a solution, the company decides to re-address all inside hosts using IP addresses from RFC 1918. In this fashion, the company ensures that all inside hosts are using consistent IP addresses from the same class (for example, 10.0.0.0). Since the company has re-addressed all inside hosts, the registered global IP addresses can now be used for access to the Internet. Here's how it works: The router that connects to the Internet is configured for NAT translation. It is set up to use all of the global IP addresses (called a "pool") to translate all inside addresses. So, if an inside host wants to access the Internet, the NAT router will detect this and assign an IP address from the global pool to the inside host. The NAT-translated packet can now be sent over the Internet, since the translated source address is from a global pool of IP addresses. Similarly, any additional inside hosts will also be NAT-translated by the router prior to accessing the Internet. The router will maintain a translation table that lists the local IP address and global IP address assigned to it. For the duration of a host's conversation, the router will continue to use the same global IP address. When the host is finished accessing the Internet, the router will detect that the inside host has not sent any data to the outside for some time, and will remove the NAT translation entry from its translation table. This global IP address can now be assigned to other hosts. Note that the router is only configured with a limited supply of global IP addresses, which will probably be lower in number than the number of hosts on the inside. If the router has used all of the global IP addresses for translations, any additional hosts that wish to access the Internet will be denied, since the router does not have any IP addresses to assign. The user will have to wait until an address is returned to the pool.

Figure 10.4 illustrates a diagram depicting the above. The router is performing dynamic NAT translations from the inside to the outside. It is con-

figured with a pool of addresses from the 192.168.1.0/24 network. The hosts on the inside are using IP addresses from the 10.0.0.0 network and the outside Host D is on the Internet. Following the diagram is an example that walks through how Host A on the inside would communicate with Host D on the outside.

Figure 10.4 Dynamic NAT translation.



NOTE

In the example below (and all subsequent examples), the step number corresponds to the step number in the diagram above. For example, Step 2 discusses Host A sending traffic to Host D; this corresponds to the same step number in Figure 10.4.

Here's how the translation proceeds:

1. Inside Host A wishes to communicate with outside Host D on the Internet.
2. Host A sends all traffic to Host D with source IP 10.1.1.1 (the inside local IP address) and destination IP address 192.168.2.1 (the outside global address).

3. Upon receiving the packet, the router consults its NAT table and determines that IP address 10.1.1.1 (Host A's inside local address) has not been mapped to an inside global address.
4. The router was configured with a global NAT pool consisting of a IP address from the 192.168.1.0/24 subnet. The router chooses an available inside global address (192.168.1.1) from its NAT pool and dynamically maps it to Host A's inside local IP address (10.1.1.1). If the router does not have an address available to assign to Host A, then it will refuse the connection to the outside.
5. The router changes the source IP address in the packet to 192.168.1.1 and leaves the destination IP address as 192.168.2.1.
6. Host D receives the packet and replies to Host A using Host D's source address (192.168.2.1) and Host A's destination (192.168.1.1).
7. Upon receiving the packet, the router consults its NAT table and determines that 192.168.1.1 is dynamically mapped to 10.1.1.1.
8. The router changes the destination IP address to 10.1.1.1 and leaves the source IP address as 192.168.2.1.
9. Host A receives the packet and continues the conversation.
10. When Host A and Host D complete their conversation, the NAT software within the router detects this and, after some time (user configurable), deallocates IP address 192.168.1.1 and returns it to the NAT pool.

Configuring Dynamic NAT

Dynamic NAT translations use standard Cisco access lists to specify which addresses on the inside can be translated. This list comprises inside local IP addresses and only those addresses for which translations are permitted. Additionally, for dynamic NAT, NAT pools are created by name that consist of ranges of IP addresses that will be used for the translation. These will be the list of inside global addresses that are permitted to traverse the Internet.

Here are the steps that are involved in configuring a dynamic NAT translation:

1. Create an access list with the list of inside local IP addresses that are permitted to be NAT translated (for example, allowed to access the outside). The source below is the IP address on the inside that is permitted to access the outside. If an entire subnet is to be permitted, the source-wildcard parameter can be used to define a mask (for example, if the entire 10.0.0.0/8 network is to be configured to access the outside, then the source-wildcard would be 0.255.255.255).

```
router prompt> access-list access-list-number permit source [source-wildcard]
```

2. Define NAT pools by name. Create as many pools as necessary to accommodate all inside local hosts requiring simultaneous access (for example, if the total number of addresses in the NAT pools is 100 and 150 hosts on the inside require access, then the first 100 hosts that request access will be granted it while the remaining 50 will have to wait until some of the first 100 hosts are finished).
3. The **name** parameter below is a name given to the pool of addresses on the router. A pool for the marketing department could be called **marketing** and a pool for the engineering department could be **engineering**, and so on. **Start-ip** and **end-ip** are the beginning and the ending IP addresses of the NAT pools, respectively. The **netmask** and the **prefix-length** parameters are used to indicate the subnet mask of the IP addresses within the pool.

```
router prompt> ip nat pool name start-ip end-ip {netmask netmask|prefix-length prefix-length}
```

4. Link the NAT pools to the Access lists by specifying which pool should use which Access list. Use the name of the pool chosen above and the Access list number configured in Step 1 above.

```
router prompt> ip nat inside source list access-list-number pool name
```

5. Next, identify the interface from which the inside local addresses in the Access lists are being sourced; this will be referred to as the “inside” interface. The **interface-number** below should be of the form **Ethernet0**, **Serial0**, and so on.

```
router prompt> interface interface-number
```

6. At this stage, the router is not aware of which interface is the inside interface and which is the outside interface. The following command will denote the interface above as the inside interface:

```
router prompt> ip nat inside
```

7. Repeat the steps above for the outside interface (the interface from which traffic will exit after the NAT translation):

```
router prompt> interface interface-number
router prompt> ip nat outside
```

The completed config file would look like:

```
access-list 1 permit 10.1.1.0 0.0.0.255
!
ip nat pool employees 192.168.1.1 192.168.1.254 netmask 255.255.255.0
ip nat inside source list 1 pool employees
!
interface ethernet0
  ip address 10.1.1.254 255.255.255.0
  ip nat inside
!
interface serial0
  ip address 192.168.1.254 255.255.255.0
  ip nat outside
```

Dynamic NAT Translation Screen Captures

The configuration file above was used to configure the NAT router in Figure 10.4. The screen captures below illustrate the output from executing **show** and **debug** commands on the router.

In the screen capture below, output from executing the **show ip nat translation** command is shown. Hosts A, B, and C were used to send PINGS to Host D on the outside to set up the translations. Note how the router has assigned each host its own inside global address.

```
NATRouter#show ip nat translations
Pro Inside global      Inside local      Outside local      Outside global
-- 192.168.1.1          10.1.1.1           --              --
-- 192.168.1.2          10.1.1.2           --              --
-- 192.168.1.3          10.1.1.3           --              --
```

The screen capture below shows the output from executing the **show ip nat translation verbose** command. The **create** field specifies how long ago the translation was created. The **use** field specifies how long ago the trans-

lation was last used. The *left* field shows how much time is remaining before the entry is deleted.

```
NATRouter#show ip nat translations verbose
Pro Inside global      Inside local       Outside local      Outside global
-- 192.168.1.1          10.1.1.1           --                --
    create 00:07:54, use 00:02:04, left 23:57:55, flags: none
-- 192.168.1.2          10.1.1.2           --                --
    create 00:04:57, use 00:04:57, left 23:55:02, flags: none
-- 192.168.1.3          10.1.1.3           --                --
    create 00:04:32, use 00:04:31, left 23:55:28, flags: none
```

The output below shows the result of typing the **show ip nat statistics** command:

```
NATRouter#show ip nat statistics
Total active translations: 3 (0 static, 3 dynamic; 0 extended)
Outside interfaces:
  Serial1
Inside interfaces:
  Serial0
Hits: 47 Misses: 3
Expired translations: 0
Dynamic mappings:
- Inside Source
  access-list 1 pool employees refcount 3
    pool employees: netmask 255.255.255.0
      start 192.168.1.1 end 192.168.1.254
      type generic, total addresses 254, allocated 3 (1%), misses 0
```

The screen capture below shows the output from using the NAT **debug** command. Host A (10.1.1.1) was used to send five PINGS to Host D (192.168.2.1). The debug output shows that an ICMP packet is being NAT-translated from either the inside or the outside. If the packet is sourced from the inside, it is shown as an *i* and if it is from the outside it is shown as an *o*. Note that when outside Host D (192.168.2.1) responds to the inside host it uses IP address 192.168.1.1, which is the address that the NAT router has assigned to inside Host A.

```
NATRouter#debug ip nat detailed
IP NAT detailed debugging is on
```

```
NATRouter#
NATRouter#
01:51:38: NAT: i: icmp (10.1.1.1, 8328) -> (192.168.2.1, 8328) [60]
01:51:38: NAT: o: icmp (192.168.2.1, 8328) -> (192.168.1.1, 8328) [60]
01:51:38: NAT: i: icmp (10.1.1.1, 8329) -> (192.168.2.1, 8329) [61]
01:51:38: NAT: o: icmp (192.168.2.1, 8329) -> (192.168.1.1, 8329) [61]
01:51:38: NAT: i: icmp (10.1.1.1, 8330) -> (192.168.2.1, 8330) [62]
01:51:38: NAT: o: icmp (192.168.2.1, 8330) -> (192.168.1.1, 8330) [62]
01:51:38: NAT: i: icmp (10.1.1.1, 8331) -> (192.168.2.1, 8331) [63]
01:51:38: NAT: o: icmp (192.168.2.1, 8331) -> (192.168.1.1, 8331) [63]
01:51:38: NAT: i: icmp (10.1.1.1, 8332) -> (192.168.2.1, 8332) [64]
01:51:39: NAT: o: icmp (192.168.2.1, 8332) -> (192.168.1.1, 8332) [64]
```

The screen capture below shows how to clear a NAT translation. Note that the inside global IP address has to be specified first and then the inside local address. After clearing the entry for 10.1.1.1, the **show ip nat translation verbose** command is typed to verify that the translation no longer exists.

```
NATRouter#clear ip nat translation inside 192.168.1.1 10.1.1.1
01:58:34: NAT: deleting alias for 192.168.1.1
NATRouter#
NATRouter#
NATRouter#show ip nat translation verbose
Pro Inside global      Inside local      Outside local      Outside
global
-- 192.168.1.2          10.1.1.2          --          --
      create 00:13:01, use 00:13:01, left 23:46:58, flags: none
-- 192.168.1.3          10.1.1.3          --          --
      create 00:12:36, use 00:12:35, left 23:47:24, flags: none
```

The output below shows how to clear all NAT translations.

```
NATRouter#clear ip nat translation *
01:58:57: NAT: deleting alias for 192.168.1.2
01:58:57: NAT: deleting alias for 192.168.1.3
NATRouter#
NATRouter#show ip nat translation verbose
NATRouter#
```

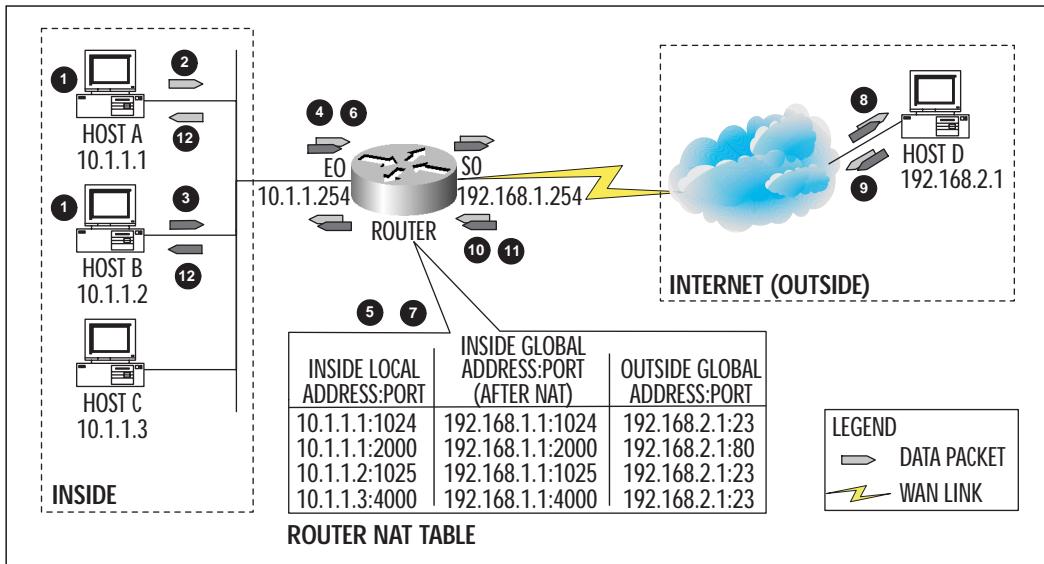
Address Overloading

Another implementation of NAT involves using one inside global address to translate all inside hosts that require access to the outside. Frequently, a company will have only a few global addresses with which to connect to the Internet, either by design or because the ISP only allocated a small number of IP addresses. A company may choose to have a few IP addresses so as to protect the IP address space on the inside. For example, if registered IP addresses are being used on the inside, then the organization would have to advertise the associated subnets to the Internet. This opens up many security risks that the organization may want to avoid.

With address overloading, multiple inside local addresses can all be translated to the *same* inside global address. All conversations are distinguished using either the TCP or UDP source port numbers. Therefore, all hosts permitted to access the outside will be able to do so, without the NAT router running out of IP addresses to allocate. As mentioned previously, if dynamic NAT is configured, then the router can only permit the number of hosts that it has IP addresses for from the pool; with address overloading, all hosts can access the outside using just one IP address. In that respect, address overloading is also known as PAT since the router recognizes different conversations using port numbers. Figure 10.5 illustrates a router performing address overloading using an inside global address of 192.168.1.1. From the router's NAT translation table, it can be seen that all conversations are unique. Each inside host is mapped to the same global address, and the router uses TCP or UDP port numbers to distinguish each conversation.

Here's how the address overloading translation proceeds:

1. Inside Hosts A and B on the Intranet wish to Telnet to outside Host D on the Internet.
2. Host A sends all traffic to Host D with source IP 10.1.1.1, source port number 1024, destination IP address 192.168.2.1, and destination port number 23.
3. Host B sends all traffic to Host D with source IP 10.1.1.2, source port number 1025, destination IP address 192.168.2.1, and destination port number 23.
4. Upon receiving the packet from Host A, the router consults its NAT table and determines that IP address 10.1.1.1 (Host A's inside local address) has not been mapped to a global address.
5. The router changes the source IP address of Host A to 192.168.1.1 (the inside global address) and updates its NAT table. Note that the

Figure 10.5 Address overloading.

router does not alter the source port number and the destination IP address of Host D.

6. The router also receives the packet from Host B and performs the same tasks. It consults its NAT table and determines that IP address 10.1.1.2 (Host B's inside local address) has not been mapped to a global address.
7. The router changes the source IP address of Host B to 192.168.1.1 also (the inside global address) and updates its NAT table. Note that both Host A and Host B have been mapped to an inside global address of 192.168.1.1.
8. Host D receives both packets and assumes that they were sent from the same host, since the source IP addresses are the same.
9. Host D replies to both packets using Host D's source address 192.168.2.1 and a destination of 192.168.1.1. Host D does not alter any port numbers.
10. Upon receiving the first packet from Host D, the router consults its NAT table and determines that 192.168.1.1:1024 is mapped to Host A 10.1.1.1:1024. The router changes the destination IP to 10.1.1.1 and sends the packet to Host A.
11. Upon receiving the second packet from Host D, the router consults its NAT table again and determines that 192.168.1.1:1025 is

mapped to Host B 10.1.1.2:1025. The router changes the destination IP to 10.1.1.2 and forwards the packet to Host B.

12. Hosts A and B receive their respective packets and continue the conversation.
13. When Hosts A, B, and D complete their conversation, the NAT software within the router detects this after some time (user configurable), and removes the mapping.

Configuring Address Overloading

The configuration for address overloading is similar to the configuration for dynamic NAT except that the parameter “overload” is specified when linking the NAT pool to the access list.

Here are the steps that are involved:

1. Create an access list with the list of inside local IP addresses that will be NAT-translated.

```
router prompt> access-list access-list-number permit source [source-wildcard]
```

2. Define NAT pools by name. Create as many pools as are necessary to accommodate all inside local hosts requiring simultaneous access. Note that in the case of address overloading, the **start-ip** and **end-ip** can be the same IP address, which will be used for all inside hosts.

```
router prompt> ip nat pool name start-ip end-ip {netmask netmask|prefix-length prefix-length}
```

3. Link the NAT pools to the access lists by specifying which pool should use which access list. Note the parameter **overload**, which specifies that the router should use TCP and UDP port numbers to distinguish multiple conversations.

```
router prompt> ip nat inside source list access-list-number pool name overload
```

4. Next, identify the interface from which the inside local addresses in the access lists are being sourced; this will be referred to as the “inside” interface. The *interface-number* below should be of the form *Ethernet0*, *Serial0*, and so on.

```
router prompt> interface interface-number
```

5. The router at this stage is not aware of which interface is the inside interface and which is the outside interface. The following command will denote the interface above as the inside interface.

```
router prompt> ip nat inside
```

6. Repeat the steps above for the outside interface (the interface from which traffic will exit after the NAT translation).

```
router prompt> interface interface-number
```

```
router prompt> ip nat outside
```

The config file looks like:

```
access-list 1 permit 10.1.1.0 0.0.0.255
!
ip nat pool employees 192.168.1.1 192.168.1.1 netmask 255.255.255.0
ip nat inside source list 1 pool employees overload
!
interface ethernet0
  ip address 10.1.1.254 255.255.255.0
  ip nat inside
!
interface serial0
  ip address 192.168.1.254 255.255.255.0
  ip nat outside
```

Address Overloading Screen Captures

The network in Figure 10.5 was set up in a lab, and the following screen captures were taken on the NAT router.

The screen shot below shows the output from the **show ip nat translation** command. Hosts A, B, and C were used to PING Host D. Note how each inside local address is mapped to the same inside global address 192.168.1.1.

```
NATRouter#show ip nat translations
Pro Inside global      Inside local      Outside local      Outside
global
icmp 192.168.1.1:1141  10.1.1.1:1141    192.168.2.1:1141
192.168.2.1:1141
```

```
icmp 192.168.1.1:7915 10.1.1.2:7915      192.168.2.1:7915
192.168.2.1:7915

icmp 192.168.1.1:95      10.1.1.3:95      192.168.2.1:95
192.168.2.1:95
```

The output from the **show ip nat translation verbose** command is shown below.

```
NATRouter#show ip nat translations verbose
Pro Inside global      Inside local      Outside local      Outside
global

icmp 192.168.1.1:1141  10.1.1.1:1141      192.168.2.1:1141
192.168.2.1:1141

      create 00:00:06, use 00:00:06, left 00:00:54, flags: extended
icmp 192.168.1.1:91    10.1.1.3:91      192.168.2.1:91
192.168.2.1:91

      create 00:00:08, use 00:00:08, left 00:00:51, flags: extended
icmp 192.168.1.1:7915  10.1.1.2:7915      192.168.2.1:7915
192.168.2.1:7915

      create 00:00:42, use 00:00:42, left 00:00:17, flags: extended
```

The output from the **debug** command is shown below. Host A (10.1.1.1) was used to PING Host D (192.168.2.1). Again, observe that outside Host D (192.168.2.1) is using IP address 192.168.1.1 to respond to Host A.

```
NATRouter#debug ip nat detailed
IP NAT detailed debugging is on
NATRouter#
02:11:56: NAT: i: icmp (10.1.1.1, 813) -> (192.168.2.1, 813) [95]
02:11:56: NAT: ipnat_allocate_port: wanted 813 got 813
02:11:56: NAT: o: icmp (192.168.2.1, 813) -> (192.168.1.1, 813) [95]
```

Static Translation

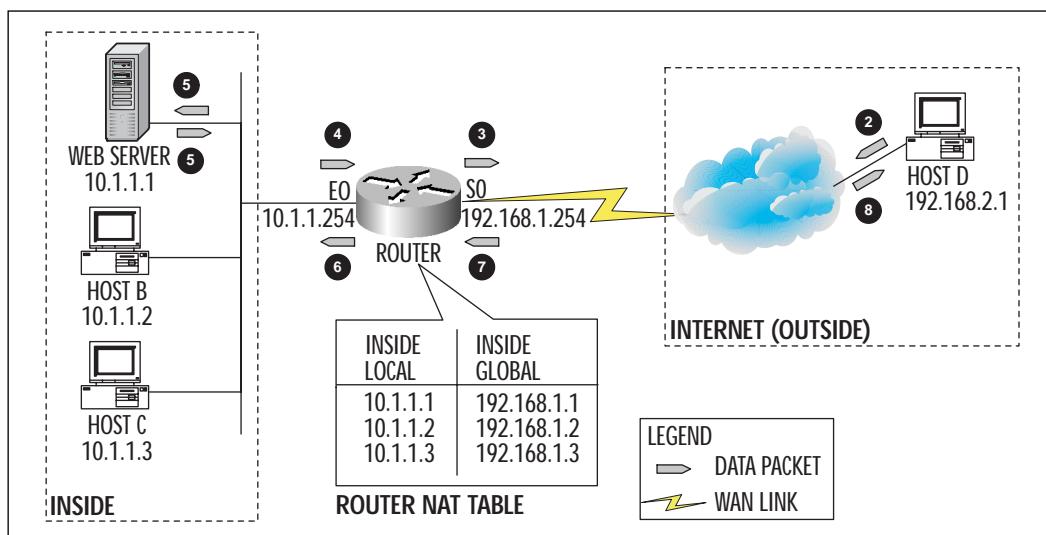
Static NAT translation is similar to dynamic NAT translation, except that the router is not configured with a pool of addresses to assign to inside hosts. The router is instead configured with one-to-one IP address mappings between inside local addresses and inside global addresses. These static entries ensure that the mappings are never timed out and the global IP addresses are not allocated to other hosts from the inside network.

Static translations are most useful when a host from the outside requires a

fixed (static) IP address for a host from the inside. If an organization has a Web server on the inside and wants to ensure that users from the outside can always access the server, then it can configure a static NAT entry on the router for the server. This mapping will guarantee that the global address assigned to the server is not reallocated to another host.

Figure 10.6 illustrates a router performing a static NAT translation. Host D from the outside (Internet) is accessing the Web server on the inside. The NAT translation table on the router is configured to assign an inside global address of 192.168.1.1 to the server. This will guarantee that this IP address is not assigned to other inside hosts.

Figure 10.6 Static NAT translation.



Here's how the translation proceeds:

1. Outside Host D on the Internet wishes to communicate with the Web server on the inside of an organization.
2. Host D sends all traffic to the Web server with source IP address 192.168.2.1 and destination IP address 192.168.1.1.
3. Upon receiving the packet, the router consults its NAT table and determines that IP address 192.168.1.1 (the Web server's inside global address) is statically mapped to 10.1.1.1 (the Web server's inside local address).
4. The router changes the destination IP address in the packet to 10.1.1.1 and leaves the source IP address as 192.168.2.1 and

sends the packet to the Web server. This effectively converts the server's IP address from an inside global address to an inside local address.

5. The Web server receives the packet and replies to Host D using Host D's destination address 192.168.2.1 and the Web server's source 10.1.1.1.
6. Upon receiving the packet, the router consults its NAT table and determines that 10.1.1.1 is statically mapped to 192.168.1.1.
7. The router changes the source IP address to 192.168.1.1 and leaves the destination IP address as 192.168.2.1.
8. Host D receives the packet and continues the conversation.

TIP

The inside global IP address does not have to be in the IP subnet ranges defined on the router; it can be any global IP address for which there is a route on the router. For example, in Figure 10.6, the router is configured with addresses from the 10.1.1.0 and 192.168.1.0 subnets and the hosts are statically mapped to the addresses in the 192.168.1.0 subnet. However, the hosts could have been mapped to any global subnet, such as addresses from the 172.20.1.0 subnet, as long as the router can route to it.

Configuring Static NAT Translations

Defining static NAT translations on the router is quite straightforward. Simply enter all addresses for which static mappings are required one at a time, and mark which interfaces are outside and inside.

Here are the steps that are involved in configuring a static NAT translation:

1. Specify that this is a static translation by identifying the inside local address and the inside global address that the local address will map to:

```
router prompt> ip nat inside source static inside-local-ip-address  
inside-global-ip-address
```

2. Next, identify the interface from which the inside local address is being sourced; this will be referred to as the “inside” interface. The *interface-number* below should be of the form *Ethernet0*, *Serial0*, and so on.

```
router prompt> interface interface-number
```

3. At this stage, the router is not aware of which interface is the inside interface and which is the outside interface. The following command will denote the interface above as the inside interface.

```
router prompt> ip nat inside
```

4. Repeat the steps above for the outside interface (the interface from which traffic will exit after the NAT translation).

```
router prompt> interface interface-number
```

```
router prompt> ip nat outside
```

So, for Figure 10.6, the config file would look like:

```
ip nat inside source static 10.1.1.1 192.168.1.1
ip nat inside source static 10.1.1.2 192.168.1.2
ip nat inside source static 10.1.1.3 192.168.1.3
!
interface ethernet0
  ip address 10.1.1.254 255.255.255.0
  ip nat inside
!
interface serial0
  ip address 192.168.1.254 255.255.255.0
  ip nat outside
```

Static NAT Translation Output

The network in Figure 10.6 was set up in a lab and the following screen captures were taken on the NAT router.

The screen shot below shows the output from the **show ip nat translation** command. The Web server and Hosts B and C were configured with static IP addresses. The screen capture shows that Web server 10.1.1.1 is mapped to 192.168.1.1; Host B (10.1.1.2) is mapped to 192.168.1.2; and Host C (10.1.1.3) is mapped to 192.168.1.3. Each time the Web server and the hosts need to access services on the outside, the router will use their respective static NAT IP addresses and forward the data.

```
NATRouter#show ip nat translation
Pro Inside global      Inside local       Outside local      Outside global
-- 192.168.1.1          10.1.1.1           --                  --
-- 192.168.1.2          10.1.1.2           --                  --
-- 192.168.1.3          10.1.1.3           --                  --
```

The output from the **show ip nat translation verbose** command is shown below. It can be seen that each of the entries above are static entries (denoted by the *static* parameter for the *flags* field). Also note that the NAT translation *create* time and *use* time are the same. This implies that the static entry is never timed out. Previously, for the dynamic NAT translations, there was a *left* field that denoted how much time was remaining before the entry would be removed from the table.

```
NATRouter#show ip nat translation verbose
Pro Inside global      Inside local       Outside local      Outside global
-- 192.168.1.1          10.1.1.1           --                  --
      create 00:02:21, use 00:02:21, flags: static
-- 192.168.1.2          10.1.1.2           --                  --
      create 00:02:12, use 00:02:12, flags: static
-- 192.168.1.3          10.1.1.3           --                  --
      create 00:02:04, use 00:02:04, flags: static
```

The **show ip nat statistics** command output is shown below.

```
NATRouter#show ip nat statistics
Total active translations: 3 (3 static, 0 dynamic; 0 extended)
Outside interfaces:
  Serial1
Inside interfaces:
  Serial0
Hits: 22 Misses: 3
Expired translations: 0
Dynamic mappings:
```

The debug output is shown below.

```
NATRouter#debug ip nat detailed
IP NAT detailed debugging is on
NATRouter#
```

```
12:34:39: NAT: i: icmp (10.1.1.1, 6416) -> (192.168.2.1, 6416) [10]
12:34:39: NAT: o: icmp (192.168.2.1, 6416) -> (192.168.1.1, 6416) [10]
```

The behavior of the **clear ip nat translation *** command is interesting. After the command is executed and the NAT translations are shown again, we see that none of the translations were cleared. This is because the translations were configured as static, and the only method of clearing a static translation is to remove it from the config file.

```
NATRouter#clear ip nat translation *
NATRouter#show ip nat translation
Pro Inside global      Inside local       Outside local      Outside global
-- 192.168.1.1          10.1.1.1           --                --
-- 192.168.1.2          10.1.1.2           --                --
-- 192.168.1.3          10.1.1.3           --                --
```

Dual Address Translation (Overlapping Networks)

Overlapping networks are two or more autonomous networks that share the same IP address space. Two different autonomous systems (ASs) can share the same address space if both of them use the same non-registered addresses from RFC 1918, or if one of the ASs selects an address space that is registered to the other AS. When both of these networks are connected together, a routing problem exists because the routers within each AS will not be able to determine if a particular network is reachable from within its AS, or from the other AS.

For IT Professionals

Autonomous System (AS)

An AS is defined as any collection of networks that is administered by a central organization (for example, when a company connects to the Internet, it is connecting its internal AS with the ISP's AS). Note that it is also possible for a company to have multiple ASs within its own organization, where each AS may be under the administrative control of a different IS group.

Suppose we have a situation in which two organizations decide to merge with each other. Company A from AS A is using IP addresses from the 10.0.0.0 network, and Company B from AS B is also using addresses from the 10.0.0.0 network. When both of these AS networks are connected together, the same scenario described above will occur, in which routers will not be able to determine where specific networks are located. So how can both of these organizations communicate electronically?

Cisco offers a solution to this problem using a concept called dual-address translation. With dual-address translation, the router interconnecting both companies will be performing NAT translations and there will be a DNS server used to resolve host names to IP addresses. Here's how it works: For hosts in Company A to communicate with hosts in Company B, all hosts in Company B have to be registered in Company B's DNS server. Essentially, communication between hosts in Company A and Company B is only permitted for hosts that have DNS entries. We will consider all hosts in Company A as being on the inside and all hosts in Company B as being on the outside. So, when a host on the inside is ready to communicate, it will send a DNS query to the DNS server (located on the outside) for a name to IP address resolution for the host that it wishes to communicate with. If the DNS server can resolve the name (belonging to hosts on the outside network) it will send a DNS response back to the inside host. Remember that the response will include an IP address of the outside host that will be from the duplicate network space as the originating host on the inside. The NAT router, upon receiving the DNS response, will modify the packet and translate the offered outside address with a preconfigured address (the outside local address) from a NAT pool. The host on the inside of Company A can now use this address to communicate with the host on the outside in Company B. Note that it is very important to realize that a DNS server is necessary for communication to occur. If an outside address is not registered, then an inside host cannot communicate with it.

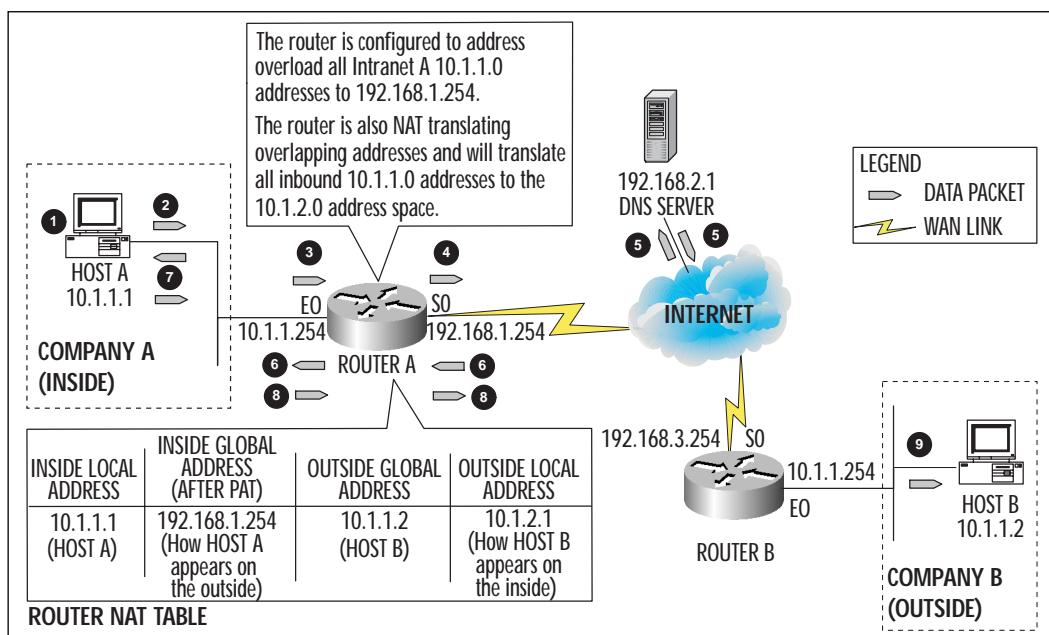
Figure 10.7 illustrates Company A communicating with Company B. Both companies are using subnets from the 10.0.0.0 network. Host A from Company A and Host B from Company B are both on the 10.1.1.0/24 subnet. The DNS server (192.168.2.1) is located somewhere on the Internet and includes all host name to IP address mappings for Company B. We will concentrate on the operation of Router A, which is performing dual address translation. The router connects Company A to the Internet and is actually serving a dual NAT function:

1. Router A will NAT translate all packets going to the Internet (for example, from the inside to the outside) using address overloading. It will map all 10.1.1.0/24 addresses from Company A to IP address 192.168.1.254 for outside communication. So, effectively, the 10.1.1.0/24 addresses within Company A are inside local addresses and the 192.168.1.254 address is the inside global address.
2. Router A will also NAT translate the DNS replies from the DNS server using dual-address translation. When the DNS server sends a DNS response packet to Company A, Router A will recognize this and NAT will translate the IP address mapping to an address from the 10.1.2.0 pool. (See the discussion below on how the translation proceeds.)

NOTE

For the sake of simplicity, the operation of Router B is not discussed. It is assumed that Router B is not participating in any NAT translations and will forward all packets without IP address conversions.

Figure 10.7 Dual address translation (overlapping networks).



Here's how the translation proceeds:

1. Host A in Company A wishes to communicate with Host B in Company B. The only information that Host A has is Host B's hostname *hostb*. Note that both Host A and Host B are on the same IP subnet.
2. In order to obtain the IP address for Host B, Host A sends a DNS query to DNS server 192.168.2.1 to resolve Host B's name to an IP address. The source address of the DNS query will be Host A's IP address (10.1.1.1), and the destination address will be the DNS Service's IP address (192.168.2.1).
3. Upon receiving the packet, the router determines that IP address 10.1.1.1 from the inside should be NAT-translated. The router is configured for address overloading and will use 192.168.1.254 for all outbound traffic. The router translates IP address 10.1.1.1 (Host A's inside local address) to 192.168.1.254 (inside global address).
4. The router changes the source IP address of the packet to 192.168.1.254 and leaves the destination IP address as 192.168.2.1 (the DNS server).
5. The DNS server receives the packet and determines that it has a hostname to IP address mapping entry for Host B. The DNS server sends a DNS reply back to destination IP address 192.168.1.254. The information in the DNS reply packet specifies that Host B's IP address is 10.1.1.2, which is in the same IP subnet space as Host A.
6. Router A receives the DNS reply and interprets the data within the packet. It recognizes that the IP address specified within the DNS reply belongs to the 10.1.1.0 subnet, which belongs to Company B and Company A (for example, since the router is performing dual address translations, it determines that there is an address overlap). The router is configured to translate all overlapping addresses from the 10.1.1.0 network to addresses from the 10.1.2.0 network so that there is no overlap. The router will translate the data portion of the DNS reply packet by changing the IP address of Host B to 10.1.2.1 from 10.1.1.2. The router does not change the source and destination IP addresses of the packet, only the data within the packet. The source address remains that of the DNS server (192.168.2.1) and the destination address that of Host A (10.1.1.1).

7. Host A receives the DNS reply and notes that Host B's IP address is 10.1.2.1. Host A opens a session to Host B using a source address of 10.1.1.1 and a destination address of 10.1.2.1.
8. The router receives the packet and determines that both the source and destination addresses have to be NAT-translated. The router will translate source address 10.1.1.1 to 192.168.1.254 because it is performing address overloading. The router will also translate the destination address 10.1.2.1 to 10.1.1.2 because it is performing dual-address translation. Router A will forward the translated packet to Host B using a source address of 192.168.1.254 and a destination address of 10.1.1.2.
9. Host B receives the packet and continues the conversation using source address 10.1.1.2 and destination address 192.168.1.254.

Configuring Overlapping Networks

To configure dual-address translation, first configure PAT for the inside local addresses, and then configure dual-address translation for the outside global addresses.

Here are the steps that are involved:

1. Create an access list with the list of inside local IP addresses that will be NAT-translated. These include all addresses from Company A that are permitted to access services on the outside (in our case, Company B).

```
router prompt> access-list access-list-number permit source [source-wildcard]
```

2. Define NAT pools by name. Create as many pools as necessary to accommodate all inside local hosts requiring simultaneous access.

```
router prompt> ip nat pool name start-ip end-ip {netmask netmask|prefix-length prefix-length}
```

3. Link the NAT pools to the access lists by specifying which pool should use which access list. Note the word *overload*, which specifies to use address overloading.

```
router prompt> ip nat inside source list access-list-number pool name overload
```

4. Next, identify the interface from which the inside local addresses in the access lists are being sourced; this will be referred to as the

“inside” interface. The *interface-number* below should be of the form *Ethernet0*, *Serial0*, and so on.

```
router prompt> interface interface-number
```

5. At this stage, the router is not aware of which interface is the inside interface and which is the outside interface. The following command will denote the interface above as the inside interface:

```
router prompt> ip nat inside
```

6. Repeat the steps above for the outside interface (the interface from which traffic will exit after the NAT translation).

```
router prompt> interface interface-number
```

```
router prompt> ip nat outside
```

The steps above defined PAT for the inside addresses; the following steps outline how to configure dual-address translation for outside global addresses. Global addresses can be NAT-translated using either static or dynamic translations. In our example, we will use dynamic translations.

1. Create another access list with the list of outside global IP addresses that will be NAT-translated. These will include addresses from the IP address space that contains Host B.

```
router prompt> access-list access-list-number permit source [source-wildcard]
```

2. Define another NAT pool by name. These addresses will be used to translate the outside global addresses.

```
router prompt> ip nat pool name start-ip end-ip {netmask netmask|prefix-length prefix-length}
```

3. Link the NAT pools to the access lists by specifying which pool should use which access list. Note that in this case, translation is required for the outside.

```
router prompt> ip nat outside source list access-list-number pool name
```

Putting it all together, the completed config file would look like:

```
access-list 1 permit 10.1.1.0 0.255.255.255
access-list 2 permit 10.1.1.0 0.255.255.255
!
ip nat pool CompanyA 192.168.1.254 192.168.1.254 netmask 255.255.255.0
ip nat inside source list 1 pool CompanyA overload
```

```
!
ip nat pool CompanyB 10.1.2.1 10.1.2.254 netmask 255.255.255.0
ip nat outside source list 2 pool CompanyB
!
interface ethernet0
  ip address 10.1.1.254 255.255.255.0
  ip nat inside
!
interface serial0
  ip address 192.168.1.254 255.255.255.0
  ip nat outside
```

TCP Load Distribution

TCP load distribution can be used to load-balance TCP traffic across multiple servers using a virtual IP address and NAT translation. Suppose an organization has an application server that is overutilized. The server could be providing application services such as HTTP or FTP. Now suppose that the company decides to distribute the load across multiple servers instead of one server. NAT's TCP load distribution feature can be used to accomplish this. The collection of multiple servers can be assigned one virtual IP address and a single host name. In this fashion, when outside hosts need access to the servers, they can use the single IP address and hostname to connect. The cluster of servers will be front-ended by a router running NAT services and configured for TCP load distribution. The router will be configured with the virtual IP address assigned to the cluster of servers, and a pool of addresses that include the real IP addresses of the servers. So, when outside hosts need to communicate with the servers, they will direct their traffic to the virtual IP address; the router will recognize this and send the data to the servers. The router will direct traffic to each server using a round-robin mechanism. Each time a new connection is received by the NAT router, it will forward the packet to the next server in line.

The advantage to TCP load distribution is that traffic from one overutilized server is load-distributed across several servers serving similar content. However, the router will not take into account the respective utilizations of each server. It will simply use a round-robin mechanism to forward data. Each time a new connection is requested, it will note this and send the connection to the next server in line.

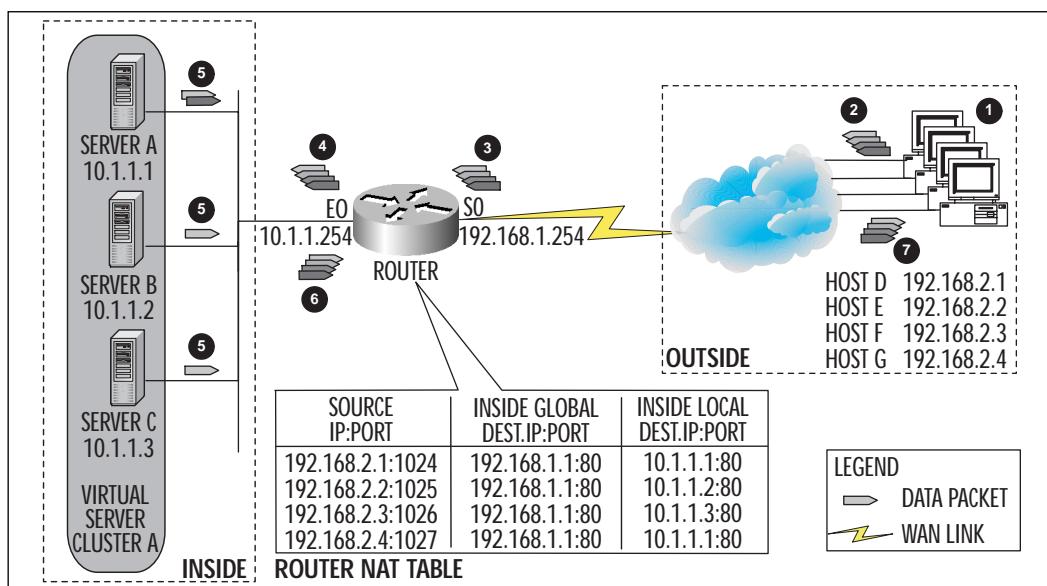
NOTE

Only TCP traffic is supported, and non-TCP frames such as UDP will pass through the NAT router untranslated. However, if additional NAT rules are in effect, they may impact the non-TCP packets.

Additionally, TCP load distribution can be used within an organization as well as from the Internet. A company may decide that it needs to load-balance its internal intranet Web servers. Or the company may choose to load-balance servers that are accessed from the Internet.

Figure 10.8 illustrates a router implementing the TCP load distribution function. The organization has configured Web servers A, B, and C to serve similar Web content. The router is configured to use a virtual IP address of 192.168.1.1. There are four hosts (Hosts D, E, F, and G) on the Internet that wish to connect to the company's Web site. These hosts are not aware that the company has multiple Web servers serving the same purpose. They simply use the company's Web site address to connect. The company has registered this Web site address with their ISP's DNS server using the virtual IP address. So, when the hosts from the Internet connect to the Web site, they send all traffic to the virtual IP address, and the router passes the data to the respective Web server.

Figure 10.8 TCP load distribution.



1. Hosts D, E, F, and G on the Internet want to connect to the company's Web site (comprised of servers A, B, and C). The hosts are considered to be on the outside, and the servers are on the inside. Note that the users on the Internet are not aware that the server that they are accessing is actually a cluster of several servers; therefore, we are referring to them collectively as a virtual server.
2. Each of the hosts on the Internet (D, E, F, and G) send traffic to the virtual server's IP address 192.168.1.1 with a destination port of 80 (HTTP). Remember, the hosts are using a DNS server located somewhere on the Internet that will resolve the company's Web site address to the virtual IP address.
3. Upon receiving the packets, the router consults its NAT table and determines that 192.168.1.1 (the inside global address) is set up as a virtual IP address that is mapped to servers A, B, and C.
4. The router changes the destination IP address to 10.1.1.1 for the packet received from Host D, and forwards it to server A. Similarly, it translates Host E's destination IP address to 10.1.1.2 and forwards it to server B translates Host F's destination IP address to 10.1.1.3 and forwards it to server C and translates Host G's destination address to 10.1.1.1 and forwards it to server A. Note that the server is implementing a round-robin mechanism to distribute the load across the servers.
5. Server A will respond to Hosts D and G using a destination IP address of 192.168.2.1 for Host D and 192.168.2.4 for Host G, and a source IP address of 10.1.1.1. Server B responds with a destination IP address of 192.168.2.2 and a source address of 10.1.1.2. Server C responds with a destination IP address of 192.168.2.3 and a source address of 10.1.1.3. Each of the servers will use its real IP address as the source address to respond.
6. The router will receive all of these packets, consult its NAT table, and translate the source addresses for all four of these packets to 192.168.1.1.
7. Hosts D, E, F, and G receive their respective packets and continue the conversation.

Configuring TCP Load Distribution

Configuration of load distribution involves using access lists and NAT pools. The access list is used to configure the virtual IP address, and the

NAT pool contains a list of the real IP addresses of the servers. Here's how the configuration procedure proceeds:

1. Create an access list that permits the virtual IP address. This is the address that is assigned to the cluster of servers.

```
router prompt> access-list access-list-number permit source [source-wildcard]
```

2. Define a NAT pool that contains all the inside local IP addresses of the servers that will be load-distributing. If the list of hosts is not consecutive, then multiple pools can be set up. Note the parameter *rotary*, which specifies that this is a load distribution.

```
router prompt> ip nat pool name start-ip end-ip {netmask netmask|prefix-length prefix-length} type rotary
```

3. Link the NAT pool(s) to the access list by specifying which pool should use which access list.

```
router prompt> ip nat inside destination list access-list-number pool name
```

4. Next, identify the interface from which the real hosts can be accessed. This will be referred to as the “inside” interface. The *interface-number* below should be of the form *Ethernet0*, *Serial0*, and so on.

```
router prompt> interface interface-number
```

5. At this stage, the router is not aware of which interface is the inside interface and which is the outside interface. The following command will denote the interface above as the inside interface:

```
router prompt> ip nat inside
```

6. Repeat the steps above for the outside interface (the interface from which traffic will be received from the Internet or outside):

```
router prompt> interface interface-number
```

```
router prompt> ip nat outside
```

The completed config file looks like:

```
access-list 1 permit 192.168.1.1 255.255.255.255
!
ip nat pool webservers 10.1.1.1 10.1.1.3 netmask 255.255.255.0 type
rotary
```

```

ip nat inside destination list 1 pool webservers
!
interface ethernet0
  ip address 10.1.1.254 255.255.255.0
  ip nat inside
!
interface serial0
  ip address 192.168.1.254 255.255.255.0
  ip nat outside

```

Output Showing TCP Load Distribution

The network in Figure 10.8 was set up in a lab and the following screen captures were taken on the NAT router.

Hosts D, E, F, and G were used to Telnet to the Web servers using a virtual IP address of 192.168.1.1. The screen capture below shows output for the **show ip nat translation** command. The router forwards Host D's (192.168.2.1) session to Web server A (10.1.1.1); forwards Host E's (192.168.2.2) session to Web server B (10.1.1.2); forwards Host F's (192.168.2.3) session to Web server C (10.1.1.3); and forwards Host G's (192.168.2.4) session back to Web server A (10.1.1.1). The router is using a round-robin algorithm.

```

NATRouter#show ip nat translation
Pro Inside global      Inside local       Outside local      Outside global
tcp 192.168.1.1:23    10.1.1.1:23       192.168.2.1:11018  192.168.2.1:11018
tcp 192.168.1.1:23    10.1.1.3:23       192.168.2.2:11017  192.168.2.2:11017
tcp 192.168.1.1:23    10.1.1.2:23       192.168.2.3:11016  192.168.2.3:11016
tcp 192.168.1.1:23    10.1.1.1:23       192.168.2.4:11015  192.168.2.4:11015

```

The output from the **show ip nat translation verbose** command is shown below. The *extended* parameter specifies that the NAT router is using one global IP address (192.168.1.1) to mask several inside local addresses (10.1.1.1–10.1.1.3). The *dest* parameter indicates that the entry is being used for a TCP load distribution. And the *timing-out* parameter means that the entry will no longer be used because a TCP RST or FIN bit was received.

```
NATRouter#show ip nat translation verbose
```

Pro Inside global global	Inside local	Outside local	Outside
tcp 192.168.1.1:23 192.168.2.1:11018	10.1.1.1:23	192.168.2.1:11018	
	create 00:00:09, use 00:00:07, left 00:00:52, flags: extended, dest, timing-out		
tcp 192.168.1.1:23 192.168.2.2:11017	10.1.1.2:23	192.168.2.2:11017	
	create 00:00:13, use 00:00:10, left 00:00:49, flags: extended, dest, timing-out		
tcp 192.168.1.1:23 192.168.2.3:11016	10.1.1.3:23	192.168.2.3:11016	
	create 00:00:17, use 00:00:14, left 00:00:44, flags: extended, dest, timing-out		
tcp 192.168.1.1:23 192.168.2.4:11015	10.1.1.1:23	192.168.2.4:11015	
	create 00:00:22, use 00:00:20, left 00:00:39, flags: extended, dest, timing-out		

The output from the **show ip nat statistics** command is shown below.

```
NATRouter#show ip nat statistics
Total active translations: 4 (0 static, 4 dynamic; 4 extended)
Outside interfaces:
  Serial1
Inside interfaces:
  Serial0
Hits: 799  Misses: 22
Expired translations: 9
Dynamic mappings:
- Inside Destination
  access-list 1 pool webservers refcount 4
    pool webservers: netmask 255.255.255.0
      start 10.1.1.1 end 10.1.1.3
      type rotary, total addresses 3, allocated 4 (133%), misses 0
```

The debug output is shown below. Host D (192.168.2.1) was used to Telnet to the Web servers using a virtual IP address of 192.168.1.1. Note that when Host D opens the session to the virtual IP address, Web server 10.1.1.2 responds. This is because this particular conversation was passed to Web server 10.1.1.2 by the NAT router.

```
NATRouter#debug ip nat detailed  
IP NAT detailed debugging is on  
NATRouter#  
NATRouter#  
12:51:13: NAT: o: tcp (192.168.2.1, 11019) -> (192.168.1.1, 23) [0]  
12:51:13: NAT: i: tcp (10.1.1.2, 23) -> (192.168.2.1, 11019) [0]  
12:51:13: NAT: o: tcp (192.168.2.1, 11019) -> (192.168.1.1, 23) [1]  
12:51:13: NAT: o: tcp (192.168.2.1, 11019) -> (192.168.1.1, 23) [2]  
12:51:13: NAT: o: tcp (192.168.2.1, 11019) -> (192.168.1.1, 23) [3]  
12:51:13: NAT: i: tcp (10.1.1.2, 23) -> (192.168.2.1, 11019) [1]  
12:51:13: NAT: o: tcp (192.168.2.1, 11019) -> (192.168.1.1, 23) [4]  
12:51:13: NAT: o: tcp (192.168.2.1, 11019) -> (192.168.1.1, 23) [5]  
12:51:13: NAT: o: tcp (192.168.2.1, 11019) -> (192.168.1.1, 23) [6]  
. . .  
. . .  
. . .
```

The screen capture below shows how to clear a rotary NAT translation. Observe how the specific port numbers have to be specified. All NAT translations can also be cleared using the **clear ip nat translation *** command.

```
NATRouter#show ip nat translation  
Pro Inside global           Inside local           Outside local           Outside  
global  
tcp 192.168.1.1:23      10.1.1.3:23      192.168.1.253:11269  
192.168.1.253:11269  
NATRouter#clear ip nat translation tcp inside 192.168.1.1 23 10.1.1.3  
23 outside 192.168.1.253 11269 192.168.1.253 11269
```

Changing NAT Timeouts

A router will typically be configured with a limited number of pool addresses to use for NAT (for example, for dynamic NAT, the router will have only a limited address pool from which to allocate addresses). If all of the addresses are depleted, the router will refuse all new connections, even if some of the hosts that have been allocated addresses are not using them any longer. The solution is to implement some type of a NAT timeout policy that states that in the event there is no activity between NAT hosts for a predefined period of time, the NAT addresses will be returned to the pool for reallocation to other hosts.

The following commands outline how to configure NAT timeouts. All of these are configured in global configuration mode and use seconds as the unit of measure.

1. **Dynamic translations** time out at a default of **86400 seconds or 24 hours**. This does not affect hosts that were translated using address overloading. A dynamic translation would occur if the router were configured with a NAT pool. For example, if the number of addresses in the dynamic pool is much smaller than the number of hosts on the inside that require translation, then this parameter can be used to change the address timeout from 24 hours to maybe 3 hours. Using a smaller number will ensure that the IP address for any host not accessing the outside is returned to the pool for reallocation more frequently.

```
router prompt> ip nat translation timeout seconds
```

2. **Static translations** do not have any timeout values because the NAT table is “statically” configured with one-to-one address mappings. These types of mappings are typically allocated to servers that need to be accessible with a well-known inside global address.
3. If **address overloading** is configured, then additional control is provided for the following types of traffic:
 - a. **UDP** traffic has a default time out of **300 seconds or 5 minutes**. This includes types such as TFTP and SNMP.

```
router prompt> ip nat translation udp-timeout seconds
```

- b. **DNS** queries and responses time out within **60 seconds**. Note that DNS traffic uses UDP as its transport; however, it has a different timeout than other UDP traffic.

```
router prompt> ip nat translation dns-timeout seconds
```

- c. TCP traffic has a default timeout value of **86400 seconds or 24 hours**. This includes any TCP traffic for which a TCP FIN or a TCP RST packet has not been received.

```
router prompt> ip nat translation tcp-timeout seconds
```

For IT Professionals

TCP FIN and RST

A TCP FIN packet is sent by a host that wishes to end a TCP conversation—such as when logout is typed at a Telnet prompt.

A TCP RST packet is generated when a break occurs in the conversation (for example, if a Telnet session is active and CTRL-C is pressed several times).

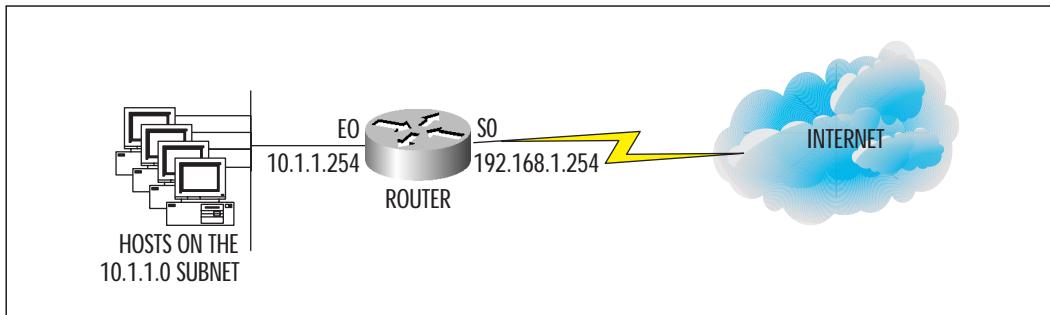
The default timeout for **TCP SYN** and **TCP RST** packets is **60 seconds**.

```
router prompt> ip nat translation finrst-timeout seconds
```

NAT to an ISP

Let's walk through an example of an organization that wants to connect to the Internet. Refer to Figure 10.9 below. The company wants everyone within the organization to have access to the Internet. It has assigned all internal hosts' IP addresses from the 10.0.0.0 network. Recognizing that they cannot connect to the Internet using the 10.0.0.0 network, they decided to use NAT translation. They contact their ISP and obtain a single outside global address, 192.168.1.254. The network engineers configure the router that connects to the ISP to use NAT and decide to use address overloading (PAT). Now, any host from inside the company that connects to the Internet will appear to be sourced from IP address 192.168.1.254.

Figure 10.9 NAT to an ISP.



The NAT configuration for the router would be as follows:

```
access-list 1 permit 10.1.1.0 0.0.0.255
!
ip nat pool employees 192.168.1.254 192.168.1.254 netmask 255.255.255.0
ip nat inside source list 1 pool employees overload
!
interface ethernet0
  ip address 10.1.1.254 255.255.255.0
  ip nat inside
!
interface serial0
  ip address 192.168.1.254 255.255.255.0
  ip nat outside
```

NAT to an ISP using Easy IP

Easy IP is a term given to a combination of features and functionality that facilitate the configuration of a Cisco router being used in remote office locations and home offices. The router will typically have only one local area network (LAN) interface and one wide area network (WAN) interface connecting back to a central site. Easy IP includes the following configuration tasks for the remote router:

- Dynamic Host Configuration Protocol (DHCP) server options
- Defining PAT parameters
- PPP/IP Control Protocol (IPCP) setup for the WAN interface

The router is configured as a DHCP server and provides automatic IP addresses to clients on the local LAN. It is configured with a pool of IP addresses to choose from. Parameters such as which DNS servers to use, lease time options, client's subnet masks, and a list of default gateways can all be configured and passed to the requesting node. Using the router as a DHCP server resolves the issue of administering IP address allocations at remote locations. Clients can be configured to use DHCP services and obtain all networking parameters from the Easy IP router.

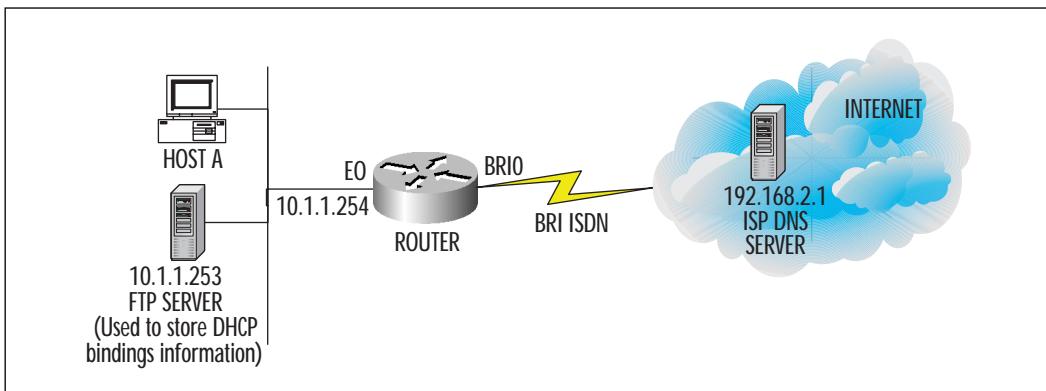
The second requirement for Easy IP is to define NAT parameters on the router. Specifically, PAT is used, and all clients are configured to use one IP address to access network services outside of their location. An Access list is created that defines which pool of inside local IP addresses is permitted to traverse the WAN. The PAT address will typically be an inside global address; therefore, the local pool of addresses can be chosen from one of the unregistered addresses in RFC 1918 (see the “NAT Overview” section at the beginning of this chapter for a list of addresses).

Finally, for Easy IP, Point-to-Point Protocol/Internet Protocol Control Protocol (PPP/IPCP) has to be set up on the WAN interface. Using PPP/IPCP, a router can dynamically obtain its IP address from a DHCP server located at the central site. So, when the remote router has any data to send over the WAN circuit, it will first try to obtain an inside global address from the central site DHCP server using PPP/IPCP to negotiate its IP address. Once it obtains this address, it will NAT-translate the local LAN IP addresses to this global address and forward the data.

Aside from being used in remote office locations and home offices, Easy IP can also benefit ISPs. The ISP can offer an ISDN dial-up service to the customer and not have to allocate a static IP address to every ISDN modem. The modems can be configured to use Easy IP and obtain IP addresses dynamically from the ISP. Easy IP is supported as two phases within the Cisco IOS: Phase I includes the PAT and PPP/IPCP functionality only, and Phase II includes PAT, PPP/IPCP, and DHCP server features. Phase I Easy IP is supported in Cisco IOS versions 11.3, 11.3T, and 12.0, and Phase II only in version 12.0T.

Easy IP Operation

Consider the situation in Figure 10.10. An organization wants to use Easy IP to provide Internet access to all of its employees. The router uses a dial-up BRI ISDN connection to the ISP. The steps involved in configuring the router are outlined below.

Figure 10.10 Easy IP.

1. First, let's configure the Easy IP router as a DHCP server to allocate IP addresses to local hosts on the 10.1.1.0 subnet.

NOTE

The DHCP server functionality is available starting in release 12.0(1)T. The following routers and access servers are supported: Cisco 700, 1000, 1600, 1700 [requires 12.0(2)T], 2500, 3600, 3800, 4000, AS5100, AS5200, 7000, and 7200 series.

2. The router will store the DHCP bindings information on a remote FTP server (TFTP and remote copy protocol (RCP) servers can also be used). In our example, the FTP server's IP address is 10.1.1.1; the login name on the server is *ftpserver* and the password is *ftpserver*; the filename that stores the bindings information is called *binding_info*.
3. We will also specify that the DHCP server cannot allocate any IP addresses in the 10.1.1.200–10.1.1.254 range, since these will be reserved for servers that may require static IP addresses or hosts that cannot participate in DHCP services.
4. The domain name for the organization is *orga.com*, and it will use the ISP's DNS server 192.168.2.1 for address resolution.

```
ip dhcp database ftp://root:ftpserver@10.1.1.1/binding_info
!
ip dhcp excluded-address 10.1.1.200 10.1.1.254
```

```
!
ip dhcp pool engineering
network 10.1.1.0 255.255.255.0
domain-name orga.com
dns-server 192.168.2.1
```

- The second step involves configuring PAT parameters on the Easy IP router. All hosts on the 10.1.1.0 subnet will use the inside global address that will be assigned to the router's serial interface to communicate over the Internet. The router will only request an IP address for the serial interface when there is traffic to be sent over the WAN circuit to the Internet. Therefore, when configuring address overloading, we specify using the IP address assigned to the BRI dialer interface. The Access list permits the 10.1.1.0 subnet access to the Internet.

```
access-list 1 permit 10.1.1.0 0.0.0.255
!
ip nat inside source list 1 interface dialer1 overload
!
interface e0
ip address 10.1.1.254 255.255.255.0
ip nat inside
```

- The final step for Easy IP is to configure the ISDN interface and use PPP/IPCP for address negotiation. Note that the BRI interface is not configured with an IP address; rather the dialer interface is configured to negotiate its IP address with the ISP. The command **ip address negotiated** specifies using PPP/IPCP negotiation for the dialer1 interface. The CHAP secret is *orga* and the phone number to dial to connect to the ISP is (800) 555-1234. The default route points to the dialer1 interface.

```
interface dialer1
ip address negotiated
encapsulation ppp
dialer remote-name orga
dialer string 8005551234
dialer pool 1
```

```
dialer group 1
dialer-list 2 protocol ip list 1
ip nat outside
!
interface bri0
encapsulation ppp
dialer pool-member 1
no ip address
!
isdn switch-type basic-5ess
isdn tei first-call
!
ip route 0.0.0.0 0.0.0.0 dialer1
```

PAT to an ISP Using a Cisco 700 Series Router

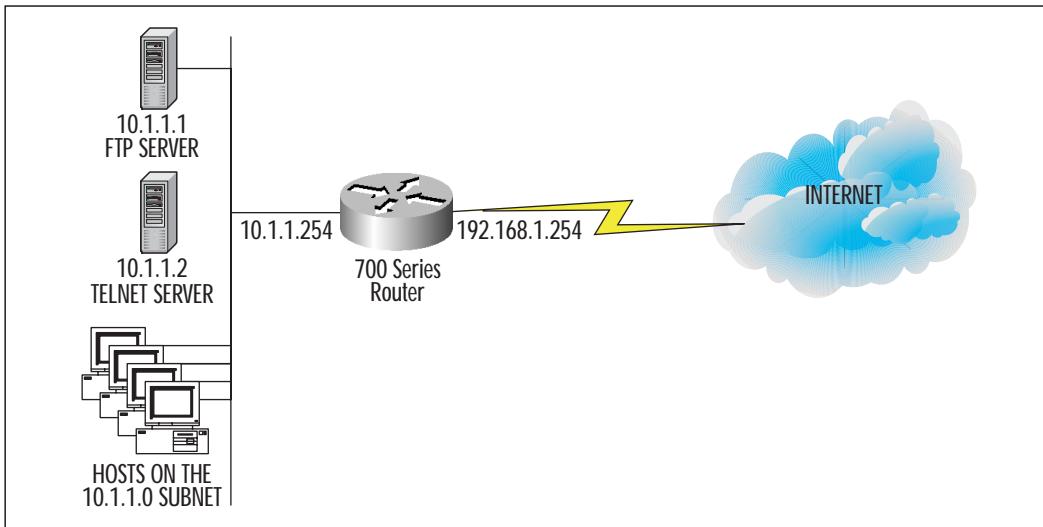
Cisco 700 series routers provide Internet access using an ISDN BRI connection. It is designed for small and home-based offices (SOHO) and can support up to five users. The routers can only support the PAT functionality of NAT, and bind all internal IP addresses to the WAN ISDN interface's IP address.

The 700 series routers also require the configuration of port handlers. A port handler statement on the router informs the router of the IP address of a server on the inside that can service a specific protocol or traffic type. Only one server of each type of protocol is allowed to pass through (for example, if a Telnet port handler is configured, then there can only be one Telnet server on the internal network that can be accessed from the outside). If there are no handlers set up for the specific traffic type, then the router itself will process the traffic. So, if a user from the outside wants to Telnet to a server on the inside, the router will, upon receiving the packet, check to see if there is a Telnet server (or handler) configured. If so, the router will forward the traffic to the Telnet server.

The following traffic types can be configured: Telnet, FTP, SMTP, WINS, and HTTP. A default port handler is also provided that will process all traffic for which a port handler has not been configured. Additionally, specific port numbers can also be defined in the event that well-known port numbers (such as 23 for Telnet) are not being used.

Figure 10.11 below shows a small office connecting to the Internet using a 700 series router.

Figure 10.11 PAT on a Cisco 700 series router.



Here are the steps involved in configuring PAT on the router:

1. Enable PAT on the router using the following command:

```
router prompt> set ip pat on
```

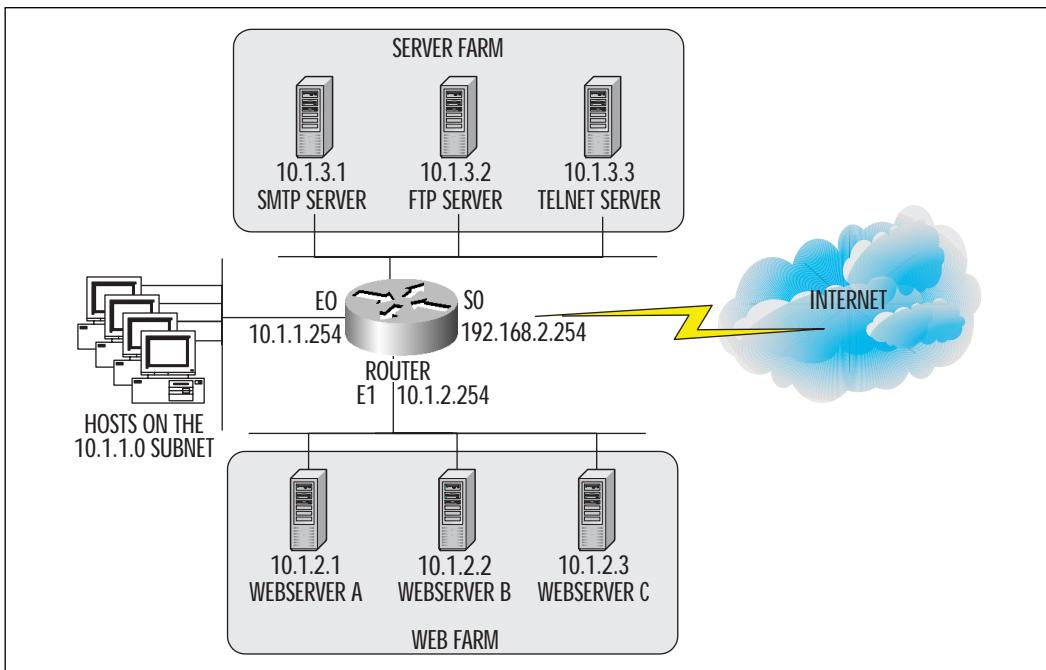
2. Set up port handlers as necessary. The following commands will configure the router to forward all ftp traffic to 10.1.1.1 and all Telnet traffic to 10.1.1.2:

```
router prompt> set ip pat porthandler ftp 10.1.1.1  
router prompt> set ip pat porthandler Telnet 10.1.1.2
```

Walkthrough

Suppose we have the scenario shown in Figure 10.12 below. An organization has several hosts that need to access the Internet and servers that are accessed from the Internet. The SMTP, FTP, and Telnet servers are required to have static IP addresses, and the HTTP Web server farm should be configured for load distribution. The following steps outline the configuration.

Figure 10.12 Sample network walkthrough.



1. The first step involves planning how to allocate the IP addresses and what scheme to use for NAT translation. We will assume that the ISP has given us the 192.168.1.0/24 subnet that can be used for allocating inside global addresses.

The company has more than 1,000 employees that need to access the Internet, and there are only 255 addresses available for use. To overcome this limitation, address overloading can be used as a solution. This will leave enough addresses to allocate to the servers.

For the server farm, we will use static address translation and register the addresses with the ISP's DNS server. Internet users can then connect to the servers using well-known names and use DNS services for name resolution.

The servers in the Web farm all provide the same content, and the NAT router will be configured for TCP load distribution. All of the Web servers can then be accessed using one IP address.

In total, we will only have used 5 of the 255 possible addresses and provided Internet access for over 1,000 employees and server access from the Internet. This leaves 250 addresses for future use.

2. Configure address overloading to permit all internal hosts access to the Internet.

```
access-list 1 permit 10.0.0.0 0.255.255.255
!
ip nat pool employees 192.168.1.1 192.168.1.1 netmask 255.255.255.0
ip nat inside source list 1 pool employees overload
!
interface ethernet0
 ip address 10.1.1.254 255.255.255.0
 ip nat inside
!
interface serial0
 ip address 192.168.2.254 255.255.255.0
 ip nat outside
```

Note that the access list permits the entire 10.0.0.0 subnet outbound access and that the pool is named employees. All internal hosts will appear to be source from 192.168.1.1.

3. Configure static NAT translation for the SMTP, FTP, and Telnet servers. Each server will have its own inside global address for communication.

```
ip nat inside source static 10.1.3.1 192.168.1.2
ip nat inside source static 10.1.3.2 192.168.1.3
ip nat inside source static 10.1.3.3 192.168.1.4
```

4. The three Web servers will be masked using a virtual IP address of 192.168.1.5.

```
access-list 2 permit 192.168.1.5
!
ip nat pool webservers 10.1.2.1 10.1.2.3 netmask 255.255.255.0 type rotary
ip nat inside destination list 2 pool webservers
!
interface ethernet1
```

```
ip address 10.1.2.254 255.255.255.0  
ip nat inside
```

Summary

NAT is an effective method of conserving a company's IP address space, while at the same time providing an added level of network security. NAT is used to translate an IP address from one address space to another address space, and is usually implemented on routers and firewalls. It can convert addresses within an organization's own internal network or between an internal and an external network (such as the Internet).

A company can allocate internal IP addresses from the reserved block of IP addresses outlined in RFC 1918, or from registered blocks obtained from the ISP. NAT can then be used to convert all internal addresses, whether registered or not, and connect to the Internet. This not only provides for a larger address space, but also ensures that internal addresses are not used on the Internet, thereby providing for a more secure network.

NAT has many configurations, and the one implemented depends on the outcome desired. A static translation is used for a one-to-one mapping of an inside address to an inside global address. An inside global address is the registered IP address that is used for the NAT translation. Static mappings ensure that the same inside global address is always used to access the same internal host.

A dynamic mapping involves configuring the NAT router with a pool of inside global addresses from which to allocate addresses. When an inside host wishes to communicate over the Internet, it will be assigned an available address from the global pool. After a user-configurable time period, the address is returned to the address pool and can be reused for other hosts.

Multiple internal hosts can be mapped to one inside global address using the address overloading feature of NAT. The router maps an internal host's IP address and TCP or UDP port number to the internal global address. Using the socket information, the router is able to distinguish traffic from one host to another or even within the same host. Address overloading is also known as PAT, since it uses ports to track data flow.

Dual address translation can be used where two autonomous systems are connected together and share the same IP addresses. With the use of a DNS server and a NAT router, hosts within one AS can access the hosts

within another AS. The NAT router can intercept DNS responses and insert a preassigned IP address prior to forwarding to the end host. In this fashion, the end hosts will use this new IP address to access services in the other AS.

Finally, NAT also features TCP load distribution. Traffic can be distributed across multiple servers all serving the same content. A virtual IP address is assigned to all of the servers, so when the NAT router receives data traffic, it uses a round-robin algorithm and sequentially passes any new connections to the next server in line. In this fashion, multiple similar servers can replace one overloaded server.

Cisco uses a concept called Easy IP to provide easy Internet access to small remote offices. Basically, the router at the premises is configured as a DHCP server to automatically allocate IP addresses to the hosts. In this manner, the router handles administration of addresses automatically. The router is next configured to use PAT to NAT all the internal hosts to a single inside global address. The global address itself is obtained automatically from a remote access server only if there is traffic to send. So, using Easy IP, IP address allocation, NAT translation, and router IP address selection are all handled automatically.

FAQs

Q: I'm using address overloading; why do I have such a large number of ICMP translations when I view the NAT translation table?

A: Each time a PING (which uses ICMP) is sent through the NAT router, the router will create an entry for every PING packet corresponding to the ICMP sequence number. So, if 10 PINGs are sent through the router, there will be 10 ICMP NAT translation entries—one for every PING.

Q: Why can't I retrieve SNMP MIB information from devices after my address has been NAT translated?

A: Responses to SNMPGET queries typically embed IP address information in the payload, which the NAT function cannot translate. The NAT router will only translate IP addresses at the network layer.

Q: What happens when all of the addresses in my dynamic address pool have been assigned to internal hosts?

A: Since all of the dynamic IP addresses have been allocated and there are no additional IP addresses available, any host that has not been assigned an address from the pool will not be NAT translated. The host will not be able to access any services on the outside until an address becomes available, which can happen automatically if a previously allocated address times out, or a network administrator deletes an allocated entry from the NAT translation table.

Q: Can I configure my NAT router to use static, dynamic, and address overloading simultaneously?

A: Yes. To achieve this, configure the router with all static NAT entries first. Then, a dynamic pool can be created with the remaining IP addresses (or IP addresses from a different IP network). Finally, address overloading can be configured using the desired IP address. All of these are simultaneously possible, as long as none of the allocated IP addresses conflict with each other.

Q: I have two routers that connect to the Internet and want both of them to perform NAT translations. I only have a few global IP addresses remaining. Can I assign the same IP address pool to both routers to conserve addresses?

A: No. The IP address pool can only be allocated to one NAT device on the network. If the same set of IP addresses from the pool is also allocated to another device, then IP address assignments by the routers will conflict with each other. One solution is to use address overloading on both routers. If more than one global address is available, then assign one IP address to one router and assign the other address to the other router.

Q: Where can I obtain additional information on NAT?

A: Cisco's NAT Web site address is www.cisco.com/warp/public/732/nat and includes items such as white papers and technical tips. Additional information can also be obtained from the official NAT RFC 1631.

Private Addressing and Subnetting Large Networks

Solutions in this chapter:

- Discovering the motivation for using private addresses
- Calculating address allocation efficiency
- Examining RFC 1918 private address ranges
- Developing strategies for subnetting private addresses

Introduction

You've heard it said: "We're running out of IP Addresses!" Really? In the IP (version 4) architecture, we use 32-bit address fields. With 32-bits in our addresses, there are 2^{32} unique addresses available. That's over four *billion* addresses. We know that the Internet has experienced exponential growth over the last few years, but even with continued growth, it's unlikely that we'll see anywhere near four billion machines on the Internet any time soon.

So where's the problem? The problem exists in the granularity of address allocation. Prior to Classless Inter-Domain Routing (CIDR), addresses were allocated in classful blocks. That is, if you needed more addresses than a Class C network provided, you got a Class B network address; if you needed more than a Class B provided, you got a Class A network address. Those were the only three choices. (Not many organizations actually got Class A addresses, of course.)

Although there are indeed over 4 billion unique IP addresses available with the current version of IP, the number of unique *network numbers* is much fewer. In fact, there are only 126 Class A networks, about 16,000 Class B networks, and about 2 million Class C networks. This design has led to widespread waste of globally unique IP addresses.

Strategies to Conserve Addresses

In the 1970s, the architects of the Internet envisioned an internetwork with dozens of networks and hundreds of nodes. They developed a design where any node on the internetwork was reachable by any other node. Back then, no one could have guessed the effect new applications like the World Wide Web and vastly increased bandwidth would have on the number of people interested in participating in "the Net." In the Internet today, there are tens of thousands of networks and millions of nodes. Unfortunately, the original design has not scaled well. The increased number of networks joining the Internet has strained router technology, and the sheer number of participants has strained the limits of IP addressing as it was originally designed. Some compromises had to be made to allow the Internet to continue its growth.

Several strategies have been developed and implemented to help the Internet community cope with its growing pains. They help reduce the load on the Internet routers and help us use globally unique IP addresses more efficiently. These strategies include:

- CIDR
- Variable-Length Subnet Masking (VLSM)
- Private Addressing

Classless Inter-Domain Routing (CIDR)

Classless Inter-Domain Routing (CIDR), specified in RFCs 1517, 1518, and 1519, was introduced in September 1993 as a way to reduce router table growth. As a side effect, it has helped reduce the waste of IP Addresses by reducing the granularity of allocation. Now, instead of full Class A, B, or C networks, organizations can be allocated any number of addresses.

(Normally, addresses are allocated in even powers of two to allow CIDR to realize its maximum benefit, but in reality, any number of addresses can be allocated.)

For example, if you needed 3,000 addresses for your network, a single class C network (256 addresses) would be insufficient. If, however, you were assigned a Class B network (65,536 addresses), there would be over 62,000 addresses wasted! With CIDR, you can be allocated a block of 4,096 addresses—equivalent to 16 class C networks (a /20 in CIDR notation). This block of addresses will cover your addressing needs now, allow room for growth, and use global addresses efficiently.

Variable-Length Subnet Mask (VLSM)

Variable-Length Subnet Mask (VLSM) is a technique used to conserve IP addresses by tailoring the mask to each subnet. Subnets that need many addresses will use a mask that provides many addresses. Those that need fewer addresses will use a different mask. The idea is to assign “just the right amount” of addresses to each subnet.

Many organizations have point-to-point WAN links. Normally, these links comprise a subnet with only two addresses required. But that would never do for a typical LAN where there are dozens (if not hundreds) of hosts in a subnet. By using a routing protocol that supports VLSM, we can use a block of addresses much more efficiently.

Private Addresses

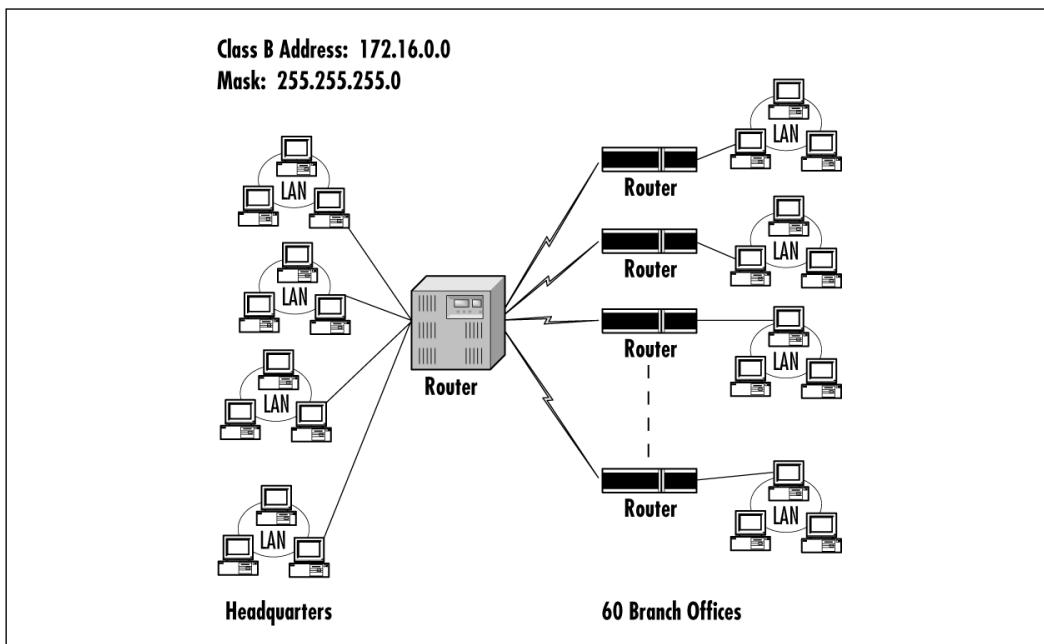
By far, the most effective strategy for conserving globally unique (public) IP addresses involves not using any at all! If your enterprise network will be using TCP/IP protocols, but will not be communicating with hosts in the global Internet, you don't need to use public IP addresses. The Internet Protocol simply requires that all hosts in the interconnected network have unique addresses. If the internetwork is limited to your organization, then the IP addresses need only be unique within your organization.

Today, many (if not most) organizations want to have at least some ability to communicate over the Internet. Does that mean these organizations must use public addresses? Yes it does—but it does not mean that *all* of the devices in that network must have public addresses. Such networks can still use private addresses and a technique called Network Address Translation (NAT) to convert those private (inside) addresses to public (outside) addresses.

Addressing Economics

IPv6 is fixing the problem with the limited address space of IPv4. Until IPv6 is fully deployed, we must make use of the IP addressing system we have. Sometimes, the networks we must support are not IP-address friendly. For example, consider the sample network in Figure 11.1.

Figure 11.1 A sample network.



In the network shown in Figure 11.1, we have multiple LANs at the headquarters location and several branch offices that each have one LAN. The headquarters router is acting as a “collapsed backbone,” connecting all the headquarters LANs and, via leased lines, the branch office routers. The organization has been assigned class B address 172.16.0.0, which provides 65,536 unique addresses.

As we mentioned earlier, the serial links connecting routers need their own IP addresses. In a point-to-point network such as the dedicated leased lines shown in the figure, each of the links is an individual subnet.

For IT Professionals

Using Frame Relay Network as WAN Technology

When you use Frame Relay networks as your WAN technology, the entire Frame Relay “cloud” is one subnet, and each router interface will have an address appropriate for that subnet.

Table 11.1 lists the various subnets and the addressing requirements for each.

Table 11.1 Sample Network Addressing Needs

Location	# Subnets	# Hosts
Headquarters	1	50
	1	110
	1	190
	1	150
	1	150
Branches	60	30
WAN Links	60	2

In this example, the network is using RIP (version 1) as the routing protocol, so each subnet must use the same mask. Identify the largest subnet in our network: One of the subnets at the Headquarters location needs 190 addresses. Consulting our resources, we see that 255.255.255.0 is the most appropriate mask to use because it provides 254 unique addresses in each subnet. Table 11.2 shows just how inefficient it can be to use a single, fixed mask for all subnets.

Table 11.2 Sample Network Address Analysis

Location #	Subnets	Interfaces	Subnet Unused	Total Unused
Headquarters	1	50	204	204
	1	110	144	144
	1	190	64	64
	1	150	104	104
	1	150	104	104
Branches	60	30	224	13,440
WAN Links	60	2	252	15,120

The Headquarters subnets are sized appropriately, even allowing for some growth. The branch office subnets provide many more addresses than will actually be used. The biggest waste occurs in the WAN links. Since the sample network is using point-to-point links between headquarters and the branches, we will never need more than two addresses in each subnet. If you add up the numbers, there are a total of 2,570 addresses needed, but we are allocating 125 subnets with 254 addresses each for a total of 31,750 addresses. As you can see, we're not using our Class B network address very efficiently. The situation is even worse than it first appears. We see there are over 29,000 unused addresses in the subnets we are using; we're only using 125 of the possible 256 subnets available. If you include the *other* 131 subnets with 254 possible addresses each we have a grand total of 62,454 unused addresses. In other words, we're using just under 4 percent of the total addresses provided by our Class B network number. This inefficient use of addresses is one of the main causes of IP address exhaustion.

If we could use VLSM, the subnets would be sized more appropriately, but the larger problem remains. We would still be using only about 4 percent of our total Class B space.

An Appeal

RFC 1917, published in February 1996, is titled “An Appeal to the Internet Community to Return Unused IP Networks to the IANA.” It cites the growing problem of IP address exhaustion and asks administrators to be good “netizens” and return blocks of IP addresses to the Internet Assigned Numbers Authority for reallocation. It suggests three alternatives:

- If you aren’t going to connect to the public Internet, you don’t need globally unique addresses. Use private addresses instead.

- If you have a portable block of addresses, return the block to the IANA and use addresses supplied by your upstream Internet Service Provider.
- If you have a large block of public addresses, but only need a small portion of them, return the large block to IANA and request a smaller block of addresses. This would be the appropriate action for our example network considered earlier.

Public vs Private Address Spaces

The Internet Protocol requires that each interface on a network have a unique address. If the scope of your network is global, then the addresses must be globally unique. Such is the case with the Internet. Since global uniqueness must be assured, a centralized authority must be responsible for making sure IP address assignments are made correctly and fairly.

For the last few years, this has been the function of the IANA. The Internet has been rapidly expanding in both number of connected networks and number of new applications. The 1990s have seen both the commercialization and the internationalization of the Internet. To meet the demands of a growing Internet community, the IANA is being replaced by the Internet Corporation for Assigned Names and Numbers (ICANN).

NOTE

More information about the ICANN can be found at www.icann.com.

If an organization wants to use IP protocols and applications in its network, but has no intention of connecting its network to the global Internet, the IP addresses it uses need not be globally unique. A network of this type is called a private network, and the addresses used are called private addresses.

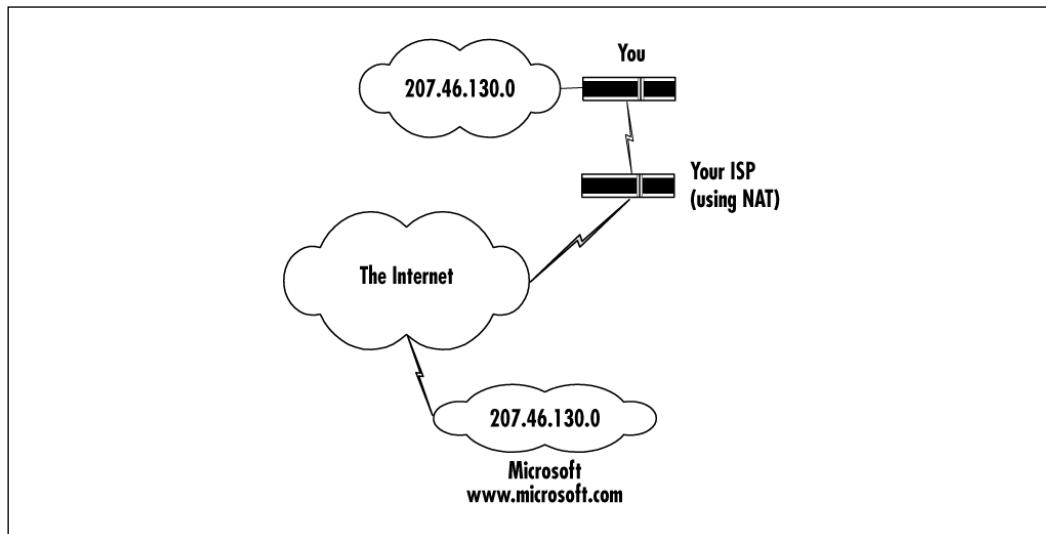
Can I Pick My Own?

If you are deploying IP on a private network, you can use any IP addresses you wish, as long as you adhere to the normal IP addressing rules. Before you go crazy and use an entire Class A address for each subnet, consider the following possibilities:

- Most organizations will eventually choose to implement *some* kind of connection to the Internet—if for no other reason than to exchange e-mail.
- There may be a merger or acquisition in your future that might require joining your network to one or more other networks.

As an example, suppose you needed a Class C address for a small network that will not be connected to the Internet (see Figure 11.2). You chose to use 207.46.130.0 as your network address and configured all your devices accordingly. As soon as you finish getting everything set up, your boss decides to implement Internet e-mail. You consult your friendly neighborhood ISP who tells you not to worry. They can use a trick called Network Address Translation (see Chapter 10) that will allow you to keep using your addresses *and* give you access to the Internet. Great! Everything works just fine except for one thing—you can't access www.microsoft.com.

Figure 11.2 The danger of picking your own addresses.



The Class C address 207.46.130.0 has been officially assigned to Microsoft, which uses it in its Web server farm. When you try to access the Microsoft Web site, DNS (the Domain Name System) resolves the name to IP address 207.46.130.14. When your browser sends an HTTP request to the target address, the IP software thinks (rightly so) that the address is inside your network and does not forward it to the router.

The lesson here is that there is a risk in dreaming up your own IP addresses—even if you never intend to connect to the global Internet.

RFC 1918—Private Network Addresses

In the midst of the explosive Internet growth in the early 1990s, RFC 1597 suggested a way to help conserve globally unique IP addresses. The idea was to set aside three blocks of addresses that would never be officially allocated to any organization. These blocks could then be used in any and every private network without fear of duplicating any officially assigned IP addresses in other organizations.

NOTE

Not everyone agreed with this plan. The authors of RFC 1627 (June 1994) complained that an Internet policy decision was made without the normal peer review and public comment process. They also point out that the original ideal of the Internet architecture, worked out over decades, was to have every host uniquely addressable. They argue that RFC 1597 violates this ideal. Ultimately, of course, the proponents of private addressing prevailed.

In February 1996, RFC 1597 was updated and made obsolete by RFC 1918, and was assigned the “Best Current Practice” status.

The Three Address Blocks

RFC 1918 designates three ranges of IP addresses as private:

- 10.0.0.0–10.255.255.255
- 172.16.0.0–172.31.255.255
- 192.168.0.0–192.168.255.255

The first of these address blocks is equivalent to a traditional Class A address. In CIDR notation, it would be 10.0.0.0/8. RFC 1918 calls it a 24-bit block of addresses because only 8 of the 32 bits is fixed; the other 24 bits are available for local administration. Either way, the range contains 16,777,216 unique addresses—enough to supply even the largest networks.

The second block is called a 20-bit block and is equivalent to 16 traditional Class B networks, or a /12 block in CIDR terminology. This block contains 1,048,576 addresses.

Finally, the third block is known as a 16-bit block and is equivalent to 256 Class C networks. This 16-bit prefix supplies 65,536 different IP addresses.

Table 11.3 summarizes the private address blocks defined by RFC 1918.

Table 11.3 Private IP Address Blocks

Address Block	Classful Equivalent	Prefix Length	Number of Addresses
10.0.0.0– 10.255.255.255	1 Class A 256 Class B 65,536 Class C	/8	16,777,216
172.16.0.0– 172.31.255.255	16 Class B 4,096 Class C	/12	1,048,576
192.168.0.0– 192.168.255.255	1 Class B 256 Class C	/16	65,536

Considerations

Anyone can use any of the address blocks in Table 11.3 in any network at any time. The main thing to remember is that devices using these addresses will not be able to communicate with other hosts on the Internet without some kind of address translation.

Here are some things to think about when deciding to use private addressing in your network:

Number of addresses One of the main benefits of using private addresses is that you have plenty to work with. Since you are not using globally unique addresses (a scarce resource), you don't need to be conservative. In the example network shown in Figure 11.1, you could use an entire class B equivalent address block without feeling guilty. Even though you would be using only 4 percent of the available addresses, you are not hoarding a valuable commodity.

Security Using private addresses can also enhance the security of your network. Even if part of your network is connected to the Internet, no one outside your network will be able to reach your devices. Likewise, no one from inside your network will be able to reach hosts on the Internet. RFC 1918 specifies that “...routing information about private networks shall not be propagated on inter-enterprise links, and packets with private source or destination addresses should not be forwarded across such links. Routers in networks not using private address space, especially those of Internet service providers, are expected to be configured to reject (filter out) routing information about private networks.”

For Managers

Security Breaches from Within

Although the preceding information about security and privacy may be comforting, don't let it lull you into complacency. Security experts estimate that anywhere from 50 to 70 percent of all attacks on computer systems come from *inside* the organization. Private network addressing cannot protect against insider attacks.

Limited scope The reason you have all these addresses available is that your network will not be connected to the global Internet. If, later, you wish to communicate over the Internet, you must obtain official (globally unique and routable) addresses and either renumber your devices or use NAT.

Renumbering Anytime you switch to or from private addressing, you will need to renumber (change the IP address of) all your IP devices. Many organizations are setting up their user workstations to obtain IP addresses automatically when booting up rather than assigning a fixed IP address to the workstations. This facility requires that at least one Dynamic Host Configuration Protocol (DHCP) server be set up for the organization. DHCP is described in RFC 2131.

Joining Networks If you join your network with another that has used private addressing, you may find that some devices have conflicting addresses. For example, let's say you chose to use the 24-bit block of private addresses (network 10). You assigned the address 10.0.0.1 to the first router on the first subnet. Now you merge with another organization and must join your networks. Unfortunately, the administrator of the other network chose to assign address 10.0.0.1 to one of its routers. According to IP addressing rules, both devices cannot use the same address. Further, the two routers are probably on different subnets, so not only do you have to assign a different address to the router, you must assign different subnet addresses as well. Again, the solutions include renumbering and NAT.

Which to Use When

According to RFC 1918:

"If a suitable subnetting scheme can be designed and is supported by the equipment concerned, it is advisable to use the 24-bit block (class A

network) of private address space and make an addressing plan with a good growth path. If subnetting is a problem, the 16-bit block (class C networks), or the 20-bit block (class B networks) of private address space can be used.”

The concept of subnetting was introduced into the IP world in August 1985 (RFC 950). Since most IP software modules in use today were developed after that time, they do understand how to do subnetting. So go ahead and use the 10 network for private addressing unless you have good reasons to do otherwise. By using the 24-bit block, you have 24 bits to play with when designing a private addressing scheme.

Strategy for Subnetting a Class A Private Network

When it comes to developing an addressing plan for a private network, the rules are exactly the same as for any other IP network. Our goals for the addressing plan are as follows:

Simplicity We want the plan to be as simple as possible so that as many people as possible can understand it. When we look at the IP address of a particular device, we should be able to easily deduce what kind of device it is and where it is in our network without having to refer to volumes of documentation.

Ease of Administration We want the plan to be easy to implement and maintain. The plan should allow room for anticipated growth and, if possible, make room for unanticipated growth or other changes.

Router Efficiency As nice as it is for the plan to be understandable by the humans that have to maintain it, the routers have to live with the plan every time a packet needs to be forwarded to another subnet. Therefore, the plan should not place a heavy burden on the resources of our routers. Ideally, the plan should build in addressing hierarchies that allow the routing tables to be kept at a relatively small size.

Documentation We want to be able to describe the plan in a few short statements without a lot of exceptions.

We now present an example of a large organization that has decided to implement private IP addressing in its internetwork. The procedure is the same—choose a mask, allocate the subnet bits, and determine the range of addresses for each subnet.

The Network

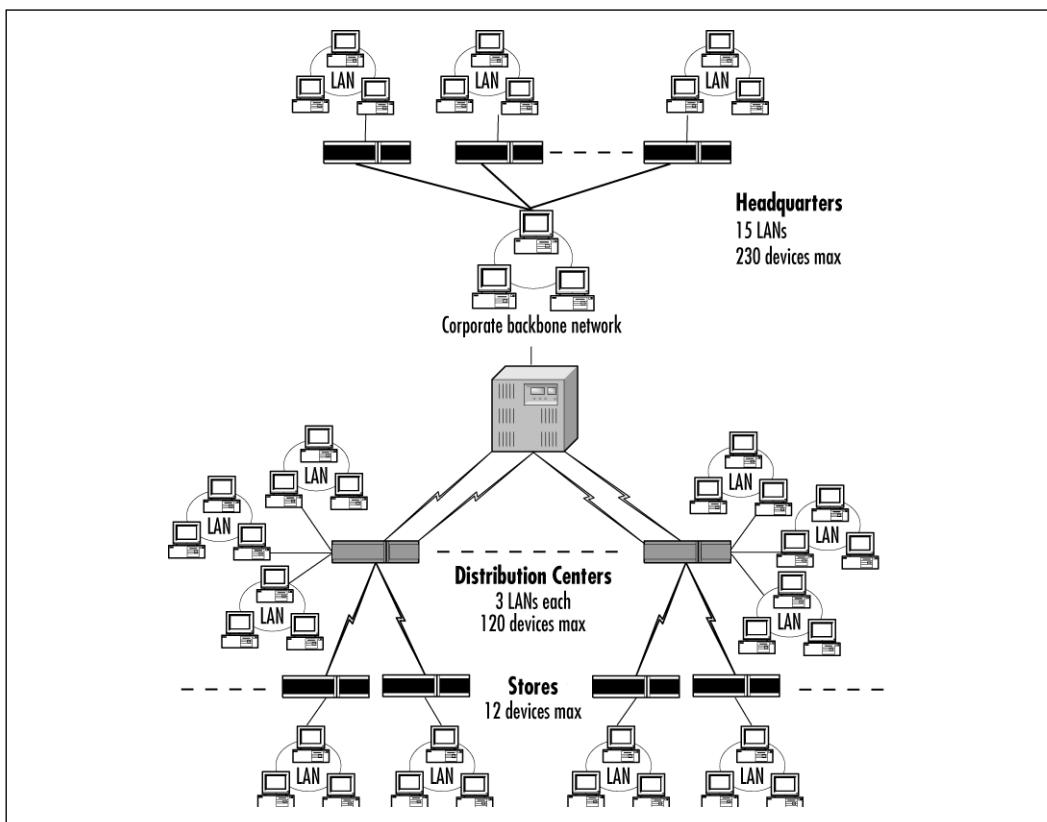
The network that we'll study here is relatively stable. There are about 3000 retail stores owned by the company and no store has more than 12 IP devices in it. Reports from management consultants indicate that this number should suffice for the medium term. Each store is connected to its regional distribution center via a leased point-to-point line.

There are currently 18 regional distribution centers, with each center supporting no more than 200 stores. Distribution centers have two physical networks for administration, and one supporting the warehouse. The largest of the admin LANs has 80 IP devices on it, and the warehouse LAN needs 120 addresses. Each distribution center is connected back to headquarters via two parallel T3 links.

The headquarters campus has 14 LANs connected by routers to the corporate backbone network. The largest of the headquarters LANs has 230 IP devices on it.

Figure 11.3 shows a high-level overview of the corporate network.

Figure 11.3 A large network.



We can summarize the addressing needs of the network in Table 11.4. From the information in Table 11.4 we can obtain the number of subnets needed (7305) and the number of addresses needed in the largest subnet (230).

Table 11.4 Sample Network Addressing Analysis

Location	# Subnets	Max Addresses
Headquarters LANs	15	230
HQ – DC links	$18 \times 2 = 36$	2
Dist. Ctr. LANs	$18 \times 3 = 54$	120
DC – Store links	$18 \times 200 = 3600$	2
Store LANs	$18 \times 200 = 3600$	12
Total Subnets Needed:	7305	
Max Subnet Size:		230

The Strategy

There are many correct solutions to this addressing problem, and arguments can be made for all of them. Since our first goal is simplicity, we'll try to keep the plan as simple as possible. Since all the software we're using understands subnetting, we'll follow the advice given in RFC 1918 and use the 24-bit block—that is, network 10.

Now that we know we have 24 bits to work with, how shall we allocate them? We look for clues in the structure of the network we are studying. There seem to be three levels of hierarchy:

- Headquarters
- Distribution Centers
- Stores

Can we somehow fit that hierarchy into our addressing scheme? Before we delve too deeply into this, we need to decide a couple of things. First, will we use fixed- or variable-length subnet masks? Using the “keep it simple” strategy, let's try using the fixed mask approach, since it is easier to design and maintain.

Our next step is to decide on a mask to use. Looking at our Class A subnetting tables, we decide on 255.255.255.0. Could we have picked another? Sure, but most people would agree that 255.255.255.0 is the easiest mask to work with. The tables tell us we now have 65,535 subnets to

work with, each supplying 254 addresses. This should work nicely. Now we have our IP address structure laid out before us:

- Network ID: 8 bits
- Subnet ID: 16 bits
- Host ID: 8 bits

Sixteen bits is represented in dotted decimal notation as two decimal numbers. Perhaps we can reduce the company network hierarchy to two levels: Region and Store. We can do this if we call the headquarters “Region 0.” Using this approach, we can try to make our IP addresses look something like this:

10.R.S.H

where R is the region number, S is the store number, and H is the host ID. If we can make this work, the IP addresses will be almost self-documenting—a very desirable feature indeed.

Address Assignment

Let’s get down to business. In Table 11.3 we identified five subnet groups. Looking at each group, we must decide on what the IP addresses should look like.

The Headquarters LANs

We stated that we should call the headquarters “Region 0.” There are 15 LANs in this group. Let’s use 10.0.L.0 for this group, where L is 0 for the backbone, and 1–14 for the administrative LANs. The LANs at the headquarters location are summarized in Table 11.5.

Table 11.5 Headquarters Subnets

Description	Address Range
Backbone	10.0.0.1–10.0.0.254
LAN 1	10.0.1.1–10.0.1.254
LAN 2	10.0.2.1–10.0.2.254
...	...
LAN 14	10.0.14.1–10.0.14.254

The WAN Links from Headquarters to the Distribution Centers

Again, there are a number of ways to assign this group of addresses. Let's use 10.100+R.0.0 and 10.200+R.0.0 for the two WAN links to each regional distribution center. Here, R is the region number. Table 11.6 summarizes these assignments.

Table 11.6 Headquarters WAN Links

Description	Addresses
HQ to Region 1	10.101.0.1 & 10.101.0.2 10.201.0.1 & 10.201.0.2
HQ to Region 2	10.102.0.1 & 10.102.0.2 10.202.0.1 & 10.202.0.2
...	...
HQ to Region 18	10.118.0.1 & 10.118.0.2 10.218.0.1 & 10.218.0.2

The Distribution Center LANs

We don't want to collide with the store LANs here, so we'll start our allocation from the top of the list. The three DC LANs will be addressed using the forms 10.R.255.0, 10.R.254.0, and 10.R.253.0. Table 11.7 shows the plan.

Table 11.7 Distribution Center Subnets

Description	Address Range
Region 1, Admin 1	10.1.255.1–10.1.255.254
Region 1, Admin 2	10.1.254.1–10.1.254.254
Region 1, Warehouse	10.1.253.1–10.1.253.254
Region 2, Admin 1	10.2.255.1–10.2.255.254
Region 2, Admin 2	10.2.254.1–10.2.254.254
Region 2, Warehouse	10.2.253.1–10.2.253.254
...	...
Region 18, Admin 1	10.18.255.1–10.18.255.254
Region 18, Admin 2	10.18.254.1–10.18.254.254
Region 18, Warehouse	10.18.253.1–10.18.253.254

The WAN Links from the DC to the Stores

Following the lead of the HQ-DC links, the link from region R to store S will look like 10.100+R.S.0 (Table 11.8).

Table 11.8 Distribution Center WAN Links

Description	Addresses
Region 1 to Store 1	10.101.1.1 & 10.101.1.2
Region 1 to Store 2	10.101.2.1 & 10.101.2.2
...	...
Region 1 to Store 200	10.101.200.1 & 10.101.200.2
Region 2 to Store 1	10.102.1.1 & 10.102.1.2
Region 2 to Store 2	10.102.2.1 & 10.102.2.2
...	...
Region 2 to Store 200	10.102.200.1 & 10.102.200.2
...	...
Region 18 to Store 1	10.118.1.1 & 10.118.1.2
Region 18 to Store 2	10.118.2.1 & 10.118.2.2
...	...
Region 18 to Store 200	10.118.200.1 & 10.118.200.2

The Store LANs

Finally, we're down to the largest group. Since this is the largest group, we'll make these addresses as straightforward as possible. As we stated earlier, the LAN in store S in region R will have the address 10.R.S.0. Table 11.9 shows some samples of store LAN addresses.

Table 11.9 Store Subnets

Description	Address Range
Region 1, Store 1	10.1.1.1–10.1.1.254
Region 1, Store 2	10.1.2.1–10.1.2.254
Region 1, Store 200	10.1.200.1–10.1.200.254
Region 6, Store 107	10.6.107.1–10.6.107.254
Region 18, Store 5	10.18.5.1–10.18.5.254

Results

The plan seems to work. Here again are the goals we established earlier, and some discussion of how well our plan meets the goals.

Simplicity, ease of administration, and documentation We're using the same net mask (255.255.255.0) in every subnet. We have a single structure for each of the five types of subnets in our network. Because we are using private addressing, we have plenty of addressing space to work with. We have used this space to give our addresses some intelligence. Some noteworthy features of our plan are:

- Any address with a zero in the second byte refers to a device at the headquarters location.
- Any address with a three-digit value in the second byte refers to a WAN link between a distribution center and either a store (third byte > 0) or the headquarters location (third byte = 0).
- All other addresses refer to devices on LANs either in the DC or in a store.

Router Efficiency Will each router in the company's internetwork need to list all 7305 subnets? We sure hope not! Our addressing scheme needs to allow for *route summarization*. To take full advantage of route summarization and keep our routing tables down to their absolute minimum size, the structure of our addresses needs to follow exactly the actual hierarchy of physical connections. Unfortunately, this is not the case with the addressing plan we have just developed. Let's look again at the plan in Table 11.10.

Table 11.10 Sample Network Address Structure

Subnet Group	IP Address Structure
Headquarters LANs	10.0.1.0–10.0.15.0
HQ – DC links	10.100+R.0.0
DC LANs	10.R.253.0–10.R.255.0
DC – Store links	10.100+R.S.0
Store LANs	10.R.S.0

In the ideal case, the corporate router would need to have only 19 entries: one for the corporate backbone, and one for each of the regions. To make that happen, all of the addresses associated with a region would

have to share a common prefix. That is, they must all have the first several bits in common. This is not the case in our plan. For example, the distribution LAN in region 5 would have the address 10.5.255.0. The link from that distribution center to store 17 would be 10.105.17.0. The only prefix these two addresses have in common is the network ID (10) itself—not very helpful.

Does this mean we have to abandon our plan? No, it doesn't. Although our plan is not *ideal* for route summarization, it well may be good enough. With some careful configuration of the regional routers, we can represent each region with three entries in the corporate router's table. One entry would represent all of the DC and store LANs, and there would be one entry for each of the WAN links between the corporate router and the DC. The central router would then have less than a hundred entries in its routing table—a very reasonable number.

The routers at each distribution center would have an entry for each of the WAN links, store LANs, and DC LANs, totaling a bit over 400 entries. Current router technology is able to handle that number of entries very easily.

Given that the routers will not be overwhelmed by the routing table sizes, and given that the addressing plan presented has some desirable features, we will go ahead and deploy the plan as presented.

BGP Requirements

Border Gateway Protocol (BGP) is the de-facto standard for routing between Autonomous Systems in the Internet. BGP was developed to address the limitations with Exterior Gateway Protocol (EGP), which was not the strongest routing protocol, although it was widely used. BGP can be thought of as the next generation of EGP. All communications between Internet service providers (ISP) is handled via BGP-4, which is *required* for CIDR. BGP-4 differs from BGP-3 just as RIP-2 differs from RIP-1. BGP-4 is also known as BGP4 without the hyphen.

BGP allows the use of announcements of classless routes, routes that are not strictly on Class A, Class B, or Class C networks. These classless routes can be subnets or supernets.

The primary purpose of BGP is to advertise routes to other networks, which are called Autonomous Systems (AS). BGP is also useful for advertising routes to upstream providers about what routes are available inside your network. When you are communicating with another ISP over the Internet, you are communicating with their network, or autonomous system, which is the more appropriate wording when speaking of routing with BGP. The border routers separate your AS from their AS. Every router

in your AS should know the route to that destination AS. All AS routers in your area should contain the same routing information, and you should be advertising only routes that you know how to get to. The sin of BGP routing is advertising routes that you do not know how to reach.

There are three types of configurations in a network:

- **Stub areas** Always end points. This is usually a single, statically routed connection from a central site, such as an ISP, to a remote location such as a home or office. BGP is not needed in stub area configurations.
- **Multihomed areas** Central sites with at least two statically-defined or dynamically routed connections to remote locations. Data will only flow to and from the remote locations. BGP is also not needed in this multihomed configuration.
- **Transit areas** Central sites with at least two connections to remote locations. One connection is to a remote location with an Internet connection, and another connection is to an additional Internet connection. Each of these locations is an autonomous system (AS). BGP is required in this configuration.

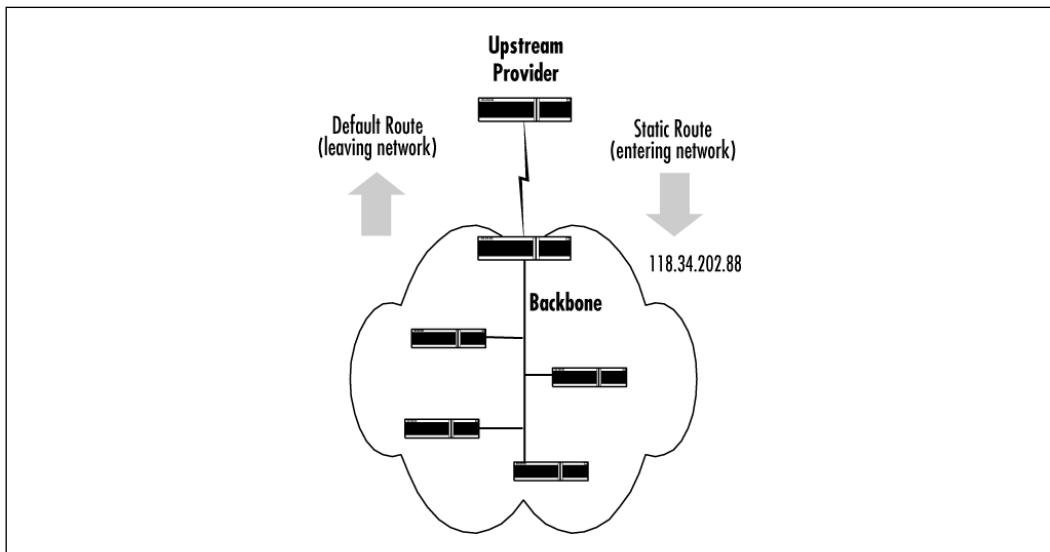
BGP is needed in the configuration if the customer has multiple locations with multiple routers, but they do not want each location's routing tables to affect the others. Defining these autonomous systems makes it possible to use these trusted paths between locations. This is the strategy that is used on the Internet to ensure better reliability and higher performance.

Figure 11.4 should clearly illustrate the purpose of BGP single-homed connections to an upstream provider.

You can see how the default route for the AS is routed through the default route. This default route makes perfect sense on a singularly homed network, with only one connection to an upstream provider. From the upstream provider, it is also much easier, because your AS does not have a multihomed link to more than one upstream provider. This upstream provider can configure a static route to your AS. It would make no sense to configure this connection between the two ASs with a dynamic routing protocol, because this link between the ASs will rarely change. If this IP address to your AS were to change, you would simply have the upstream provider change the static routing address to your AS.

You have been hearing about the autonomous system—now we need to describe the autonomous system number, which is used to represent the autonomous system to the Internet. Most networks will have only one autonomous system number. When you are exchanging routes with

Figure 11.4 Routing BGP in single-homed connections.



another router speaking BGP (called a *peering session*), it will start out like the following:

```
router BGP 14290
neighbor 204.118.35.166 remote-as 802
<the rest is omitted>
```

This communication starts out by saying “I would like to connect to ASN (autonomous system number) 14290 using BGP.” The list of commands that would initiate the routing table transfer is omitted.

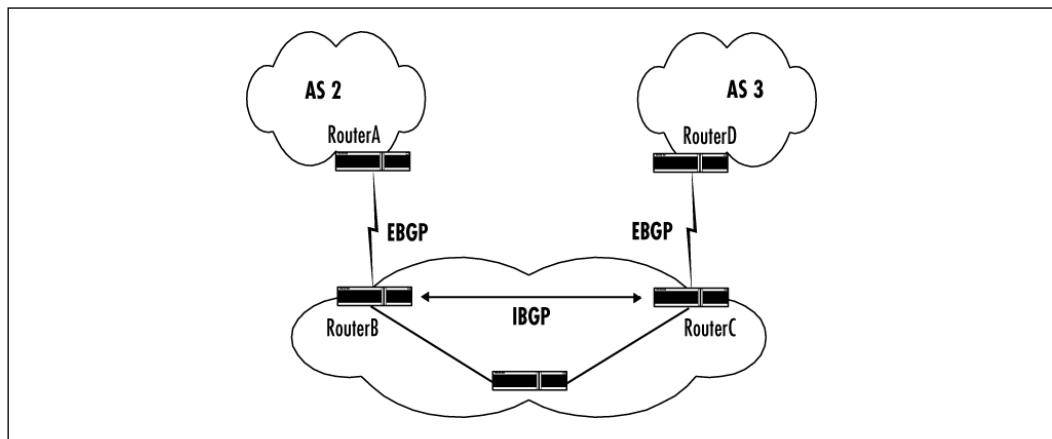
If a node wishes to connect with BGP peer node, the node will open a connection on TCP port 179, which is the default port. A significant amount of information is transferred, such as the identification numbers, authentication information, and protocol version numbers before the BGP update of the routing tables can take place. The update will not take place if the authentication has not been successful. If the update is successful, the changes will then be propagated to neighboring BGP routers.

When you communicate to other hosts and routers using BGP, you can make semi-intelligent routing decisions, which include the best path to reach a destination. This route contains more than just the first router to route the packet to; it can include the complete route to the destination. You can also advertise your routes to neighboring routers, and have those routers in turn advertise your routes to their neighboring routers.

BGP selects only one path as the best path to a destination. This path is now propagated to the neighboring BGP routers. Unlike some routing protocols, BGP does not need a periodic routing table refresh. The initial exchange between two BGP routers is the full routing table, but from then on only the optimal paths are advertised in update messages to the neighboring BGP routers. This makes long running sessions between BGP routers more efficient than short sessions, because the amount of times the full routing table is exchanged on initial contact is less.

There are actually two types of BGP that differ in terms of advertising routing information. The first is EBGP, basically referred to as BGP, which is what we have been discussing thus far. This is used to advertise routes to different autonomous systems, whereas IBGP is used to advertise routes within the same autonomous system. Figure 11.5 demonstrates the use of both types of BGP protocols and the autonomous system.

Figure 11.5 Differentiating between interior and exterior routing with IBGP and EBGP.



In the network example shown in Figure 11.5, BGP first makes sure that networks within the interior AS are reachable. Then border routers can exchange routing information with each other regarding the status of networks within their autonomous systems. EBGP is used to communicate with border routers, and IBGP is used within the AS.

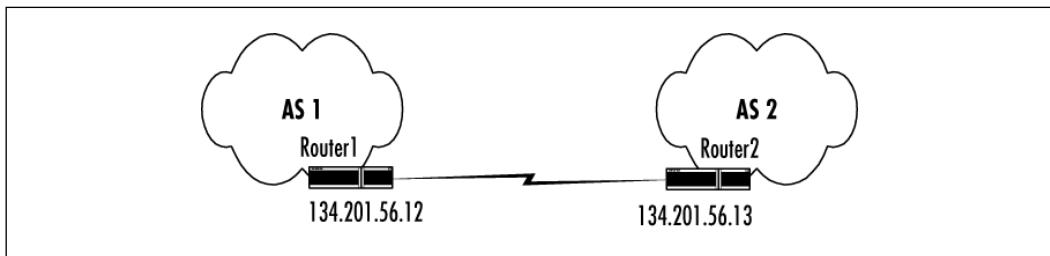
Just like RIP, IBGP is an interior routing protocol that can be used for active routing within your network. IBGP does not distribute routes as much as EBGP. Each router in an IBGP configuration must be configured to peer into every other router to exchange this information, whereas this is not needed with straight BGP. However, IBGP is more flexible and provides a more efficient means of controlling and exchanging the routing information from *within* an AS.

IBGP and EBGP Requirements

BGP requires a combination of hardware and software to support. The most commonly used implementations of BGP are with Cisco routers, Nortel routers, UNIX variants, BSD, and Linux. Nortel and Cisco routers are by far the most common types of routers currently supporting BGP.

We will now discuss the steps required to enable and configure BGP. First, we will assume that we want two routers to communicate using BGP. These routers will be called Router1 and Router2. These routers belong in two unique autonomous systems, called AS 1 and AS 2, as illustrated in Figure 11.6.

Figure 11.6 An example of routing between two separate autonomous systems.



We now need to enable BGP on the routers one at a time, starting with Router1:

```
router bgp 1
```

and now the same step on Router2:

```
router bgp 2
```

These statements enable BGP on the router for the AS in which they belong. We will now define the neighbors that we wish to communicate with via BGP. Establishing a connection between two neighbors, or peers, via BGP is made possible by the TCP protocol. The TCP connection is essential for the BGP routers to establish a connection and exchange routing updates.

The *neighbor* command is used to establish a TCP connection:

```
router bgp 1
```

```
neighbor 134.201.56.13 remote-as 2
```

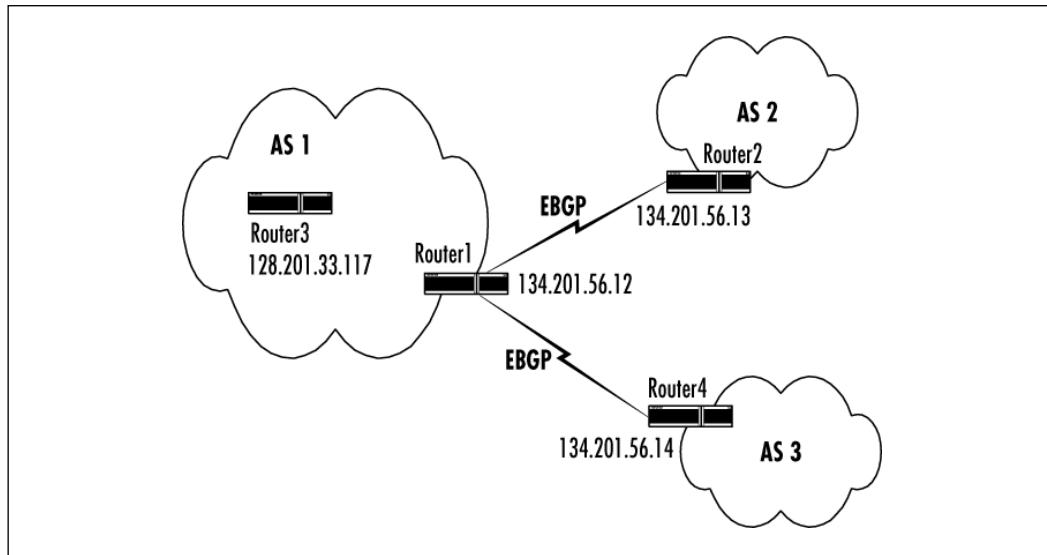
```
router bgp 2
```

```
neighbor 134.201.56.12 remote-as 1
```

These statements use the TCP/IP address of the directly connected routers for the EBGP connection. Note that EBGP will be used because we are communicating with an external autonomous system.

If we were to make the configuration more difficult, we could add another router called Router3 *within* our AS 1, and create another AS called AS 3, as illustrated in Figure 11.7.

Figure 11.7 An example of routing between three autonomous systems.



We need to modify the statements on the routers as follows:

```
Router1#
router bgp 1
neighbor 134.201.56.13 remote-as 2

neighbor 134.201.56.14 remote-as 3
```

```
Router2#
router bgp 2
neighbor 134.201.56.12 remote-as 1
```

```
Router4#
router bgp 3
neighbor 134.201.56.12 remote-as 1
```

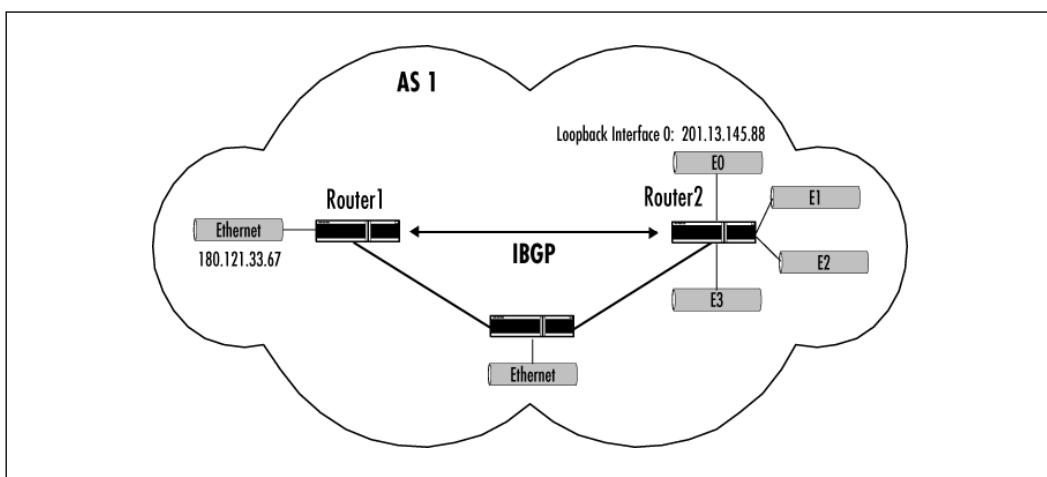
In the preceding example, Router1, Router2, and Router4 are running EBGP. Router1 and Router3 are running IBGP. The difference between running IBGP and EBGP is that the **remote-as** number is pointing to an external or internal AS.

Notice also that Router1 and Router3 are not directly connected, which is the case for Router1 being directly connected to Router2 and Router4. This is acceptable because the router is within your AS. As long as there is some IGP running to connecting the neighboring routers within the same AS this is acceptable.

Loopback Interfaces

Another feature of IBGP is the use of loopback interfaces, which eliminate a dependency that occurs when you use the IP address of a router (the physical interface to the route). Figure 11.8 illustrates the use of a loopback interface specified on Router2.

Figure 11.8 Specifying the loopback interface for reliable routing.



In Figure 11.8, Router1 and Router2 are both running IBGP in AS 1. If Router1 were to communicate with Router2 by specifying the IP address of the Ethernet interface 0, 1, 2, or 3 (as shown in the figure as “E” for Ethernet—E0, E1, E2, and E3), and if the specified interface is not available, a TCP connection was not possible. These two routers could not communicate. To prevent this from happening, Router1 would specify the loopback interface that is defined by Router2. When this loopback interface is used, BGP does not have to rely on the physical interface availability when making TCP connections. The following commands on both of the routers illustrate the use of specifying a loopback interface.

```
Router1#
router bgp 1
neighbor 201.13.145.88 remote-as 1

Router2#
loopback interface 0
IP address 201.13.145.88 255.255.255.0
router bgp 1
neighbor 180.121.33.67 remote-as 1
neighbor 180.121.33.67 update-source loopback 0
```

Router1 will specify the address of the loopback interface (201.13.145.88) of Router2 in the **neighbor remote-as** configuration command. The use of this loopback interface requires that Router2 also includes the **neighbor update-source** router configuration command in its own configuration. When this **neighbor <IP address> update-source loopback** command is used, the source of the BGP TCP connections for this specified neighbor is the IP address of the loopback interface, and not the IP address of the physical interface.

Summary

The designers of the Internet Protocol never dreamed that there would be millions of hosts on over 100,000 networks participating in the Internet. At the time, a fixed 32-bit address looked like it would be more than enough to serve the addressing needs of the Internet for years to come. And it has. However, as the Internet continues to grow, more and more pressure is being put on the user community to use globally unique IP addresses efficiently. This pressure has lead to policy changes at the Internet Registries and to new techniques to conserve addresses.

One of those techniques is to use private addresses as specified in RFC 1918. There are both benefits and drawbacks to using private addresses.

FAQs

Q: How do I know which one of the private address blocks to use?

A: Unless there is a good reason—such as a specific learning objective, or to force your router into certain behaviors—use “network 10.”

Q: Can I use VLSM in private networks?

A: Absolutely! There's no harm in using addresses wisely, even if you have a very large supply.

Q: Why is network 10 included in the private address ranges?

A: Class A network 10 was the address used by the old ARPANET, the precursor of today's Internet. Network 10 was decommissioned in the 1980s and we use it today to honor its auspicious beginnings.

Q: Can I use private addresses and public addresses in my network?

A: Yes. Since the public and private addresses use different network prefixes, they will need to be on separate ports of a router. In other words, they would need to be separate subnets of your network. The devices with public addresses will be able to communicate on the Internet, those with private addresses will not.

Q: I've got a network with private addresses. Now I want to connect to the Internet. Can I?

A: Yes, you have two options. First, you can obtain public addresses and renumber your IP devices. Second, you (or your ISP) can implement Network Address Translation (NAT) to translate your private addresses to public addresses. NAT is covered in Chapter 10.

Appendix

Implementing the Windows 2000 Servers

Solutions in this appendix:

- Understanding the installation options for Windows 2000
- Installing Windows 2000 Active Directory
- Configuring services on Windows 2000 servers

Introduction

One of the interesting things about a Cisco and Microsoft Windows 2000 network is that both Cisco routers and Windows 2000 servers can perform routing. Remote access and routing are tightly integrated functions. A remote access server is, essentially, a router. When a remote user dials into a remote access server, access to the rest of the network must be granted by routing the remote user's requests to the various requested resources. Because of this tight integration, it is not uncommon to see routing and remote access services combined on a single network component. Remote access servers also utilize modems in the same way as a network interface—again, making them, effectively, routers.

You can find this appendix as a chapter in Syngress Media's *Building a Cisco Network for Windows 2000* (available at www.syngress.com); it is provided here as an introductory resource on Active Directory, terminal services, and configuring remote access services, for BCNAN readers.

Network infrastructure can be dissected into three layers: Backbone; Shared systems or the security layer; Workstation systems or the access layer.

The infrastructure backbone is a high-speed freeway for data transmission. All network segments should be capable of accessing the network backbone, even if they are not directly attached to that backbone.

A backbone can exist within each building or campus of a global network, and then a connection to other buildings or campuses leads off of it. The backbone does not have computers directly attached to it. It should not connect directly to the Internet or any other public network. It should not have any extraneous applications or security filters preventing traffic from flowing speedily through it. Routers are the main backbone infrastructure components.

The shared systems area represents all the network segments that connect directly to the backbone. These segments have significant security placed upon them, with firewalls, access list filters, and login authentication required. Connections to public networks and the Internet should occur in this area. Servers are connected to these segments, as well as any secured resources. You will find routers and high-speed switches at this level.

The access layer of the internetwork represents each segment that includes workstations and workgroup printers. These segments are connected to the shared systems segments, making them two hops down from the backbone. You should find only hubs, switches, and bridges at this level.

So where does Windows 2000 fit into all of this? Windows 2000 (all versions) is a network operating system. It was designed to work on a network and to interact with other computers. This interaction—whether it is logging on, looking for resources, using a database, accessing a mainframe, reading e-mail, sending print jobs to a network printer, Web browsing, or downloading files from a server—consumes bandwidth by causing traffic on the network. You will need to implement your Windows 2000 servers while considering how Windows 2000 usage will affect your network. The design that you have started with will take shape, but you need to remain flexible enough to test that design and to ensure that the results meet your business requirements.

Installing Windows 2000

Windows 2000 installation is not a difficult process to undertake. In fact, it is fairly simple to install Windows 2000 directly from the installation program. If you intend to roll out multiple Windows 2000 Professional workstations, however, you should investigate unattended installation methods either using a script or disk duplication:

- Unattend.txt
- SYSPREP
- RIPREP

Automating Windows 2000 installation is one way to save costs. Automating the installation significantly reduces the time spent at each workstation or server, and only a minor amount of time is spent in a lab creating the automated setup. The method of installation should be selected according to your environment. When you have multiple types of hardware, a scripted installation probably will be best. If you have a few workstations, then a disk duplication method is best. For remote installation, you need to make certain that your workstations are equipped with the right network interface cards and you have a spare Windows 2000 server available.

Workstation installations can be automated either way; however, it is typical for servers to be installed in an attended mode. There are times when scripting a server installation makes sense:

- Server hardware is standardized
- Operating system configuration is standardized
- Many different people can install servers

When hardware or operating system configurations are standardized, then automating an installation of the operating system becomes a time saver. When several different people are installing your servers, then automation ensures a standard result, avoids errors, reduces the need for assistance, and saves time overall. (You can find in-depth coverage of installing Windows 2000 in the book *Deploying Windows 2000 with Support Tools* published by Syngress Media.)

Overview of a Scripted Installation

Windows 2000 inherits the same scripted capabilities that Windows NT included. An administrator can create a custom script to answer the Setup executable's questions so that there is no user input needed. A sample script called unattend.txt can be used for testing how scripts work. The script sometimes is called an *answer file* because it answers setup questions.

The unattend.txt script is typical of an installation script in that it contains various sections that supply information for the installation. Each section has a section heading *[section]* followed by parameters and their values in the form of *parameter=value*.

You can create a script using two different methods:

- Manually, where you edit a text file and type the various section headings and parameter/value pairs
- Using Setup Manager, where you use the Setup Manager application, found in the CD:\support\tools\deploy.cab file, to configure a script and output the text file

There are two ways of executing Windows 2000 setup. The executable that you select is entirely dependent on the operating system currently running on the machine. If using DOS, then the command is WINNT.EXE with the following parameters:

```
Winnt /S:PathToSourceFiles /T:TempDriveLetter /U:YourScriptFile
```

If you are using a 32-bit version of Windows (Windows 95, Windows 98, Windows NT), the command is WINNT32.EXE with the following parameters:

```
Winnt32 /s:PathToSourceFiles /tempdrive:TempDriveLetter /unattend:  
YourScriptFile
```

There are additional parameters for both of these setup executables, described in Table A.1.

Not only can the Windows 2000 installation be automated, it can prompt additional application installations. The administrator can add commands using cmdlines.txt, or the administrator can place a setup file (or batch file containing multiple setup files within it) in the [GuiRunOnce] section of the answer file. In addition, the administrator can run the Windows Installer Service for any compatible applications, or use a third-party tool that is intended to automate an application's installation.

Table A.1 Windows 2000 Setup File Switches

Command	Parameter	Used For	Example
Winnt	/S	States the source location for the Windows 2000 installation files	/S:e:\i386
Winnt	/T	States the location for temporary files used during the installation process	/T:d
Winnt	/U	States the name of the script file	/U:e:\myscript.txt
Winnt	/R[x]	Identifies a directory to be created, or copied if using the "x" parameter	/R:c:\myfolder
Winnt	/E	Executes a command after Windows 2000 is installed	/E:e:\myfile.exe
Winnt32	/s	States the source location for the Windows 2000 installation files; up to 8 separate /s switches can be used to provide multiple source file locations	/s:e:\i386
Winnt32	/tempdrive	States the location for temporary files used during the installation process	/tempdrive:d

Continued

Table A.1 Continued

Command	Parameter	Used For	Example
Winnt32	/unattend	States the name of the script file	/unattend:e:\myscript.txt
Winnt32	/copydir	Copies a directory of files on the hard drive	/copydir:c:\myfolder
Winnt32	/copysource	States the source directory to be copied	/copysource:e:\myfold
Winnt32	/cmd	Executes a command after Windows 2000 is installed	/cmd:e:\myfile.exe
Winnt32	/debug	Runs the Windows 2000 installation in debug mode	
Winnt32	/udf:[id]	Pulls the specific information related to the given id out of a file with multiple users' information	/udf:jmar,e:\udf.txt
Winnt32	/syspart	Creates the system partition on the stated drive letter	/syspart:d
Winnt32	/noreboot	Suppresses a PC from restarting	/noreboot
Winnt32	/makelocalsource	Copies source files to the local drive	/makelocalsource
Winnt32	/checkupgradeonly	Determines whether the given PC can be upgraded	/checkupgradeonly
Winnt32	/m	Copies replacement files from a different source location first and, if files are not present, uses files from the default location	/m:c:\folder

Overview of Disk Duplication Methods

You can use two different types of disk duplication methods for Windows 2000. Although they are nearly identical as far as the Windows 2000 setup method, the setup initialization is completely different. Disk duplication is a good choice for identically installed workstations—applications and settings included.

SYSPREP

SYSPREP is the short form of System Preparation, which refers to the process of preparing a Windows 2000 Professional system (SYSPREP does not work with the Server versions of Windows 2000) for duplication on multiple computers. SYSPREP disk duplication is a spectacularly shorter process than a scripted installation. Not only that, but it works in conjunction with third-party disk duplication applications. The process is straightforward:

1. Select a master computer that uses the same hardware (specifically, the same Hardware Access Layer (HAL), Advanced Configuration and Power Interface (ACPI), and storage controllers) as the system on which you will be duplicating Windows 2000. If you are using SYSPREP v1.1, it is not necessary to have identical storage controllers. Instead, you can specify storage controllers in the sysprep.inf file.
2. Install Windows 2000 on the master computer.
3. Install applications on the master computer that you will want on all the duplicated computers.
4. Configure desktop and system settings that should appear on each of the duplicated computers.
5. Remove data that you do not want to be copied to the target computers, such as temporary Internet files, log files, document histories, pagefile.sys (pagefile.sys can be deleted from the image only at a DOS prompt, not while the system is running and loaded into RAM), etc. Copy the sysrep.exe, sysprep.inf, and setupc1.exe files from the Windows 2000 CD:\Support\tools\deploy.cab file to C:\SYSPREP folder.
6. Verify that the Windows 2000 image is exactly what you want to appear on all duplicated computers.
7. Run SYSPREP on the master computer. SYSPREP can be run with three parameters: -quiet runs SYSPREP without user prompts. -nosidgen runs SYSPREP but doesn't keep the Security Identifiers (SIDs) that were on the master computer. -reboot runs SYSPREP

with an automatic restart when SYSPREP has completed. If you are running SYSPREP v1.1, you have an additional parameter, -pnp, which forces Plug and Play to discover new hardware on the next reboot.

8. Boot the computer from a floppy disk and connect to a server. Copy the contents of the master computer's hard drive to the server.
9. Connect to the server from the target computers and copy the image down to the hard drive.
10. Starting the target computers will run SETUPC1.EXE to generate new SIDs for the target computers and to start the mini-setup wizard. (The mini-setup wizard can be automated with the sysprep.inf. It prompts the user for the license agreement, regional settings, and other configuration information.)

TIP

Many of the tools needed to deploy Windows 2000 are not part of the Windows 2000 operating system. Some tools can be found on the Windows 2000 CD in the Support\Tools directory and are installed by running the setup file in that directory. Others can be found on Microsoft's Web site under Windows 2000 downloads.

www.microsoft.com/windows2000/downloads/deployment/sysprep/ is the location for downloading the SYSPREP tool.

RIPREP

RIPREP is the executable associated with the Remote Operating System installation. This is a disk duplication method that uses a Pre-boot-Execution-Environment (PXE)-capable network interface card on the target computer, and a Remote Installation Service (RIS) running on a server. The RIS server can deliver an image to a workstation without anyone having to boot the target computer with a boot disk, as you would have to use with the SYSPREP method.

The same hardware limitations for SYSPREP apply to RIPREP—the HAL, ACPI, and storage controllers must be identical for the master and target computers. It is likely that many images will be needed for an enterprise that has multiple hardware types. In addition, if different users

require different application sets, then multiple images will be required. RIS requires the service to be running on a Windows 2000 server. The RIS process is as follows:

1. The Remote Operating System Installation service is installed on a Windows 2000 server.
2. A master computer is selected, and Windows 2000 is installed on it. The master computer is the one that you will be duplicating on other computers.
3. Applications are installed on the master computer.
4. Settings and configuration changes are made to the master computer.
5. The image is validated.
6. RIPREP is run on the master computer to package the image for delivery.
7. The RIS server is configured to provide the image to target computers.
8. The client creates a RIS service request based on the Dynamic Host Configuration Protocol (DHCP) discovery process.
9. The RIS remote boot request is forwarded to both a DHCP and a RIS server, or the request will fail.
10. If passing through a router, the router must be configured to forward DHCP broadcasts (DHCP is based on User Datagram Protocol (UDP); and UDP packets typically are not forwarded).
11. Upon receiving the request, the RIS server checks the Active Directory for a computer account with a Globally Unique Identifier (GUID) matching the GUID in the service request. If the GUID exists and a RIS server has been configured for it, the response to the client includes the RIS server.
12. The configured RIS server answers the request and delivers the image.

WARNING

When you install Windows 2000 from a server share to multiple target computers, you create a huge traffic load for the network. The method that you use to install—whether SYSPREP, RIPREP, or scripted installation—is inconsequential to the resulting impact on the network. To coun-

teract the impact, you should follow a few simple rules: Never install workstations during business hours. If the business is open 24 hours a day, 7 days a week, then you should never install workstations during busy hours. Also, never install workstations across slow, unreliable, or wide area network links. If at all possible, do install workstations on the same network segment to isolate the traffic. Limit the servers from which specific remote installation clients can download an image.

Configuring Remote Installation Services starts in the Control Panel by opening the Add Remove Programs icon and then selecting Add/Remove Windows Components. The option to Configure Remote Installation Services should be listed, or you will need to add the component. When you select this option, you will see the Welcome screen shown in Figure A.1. Both a valid Domain Name System (DNS) and DHCP server are required, as well as the Windows 2000 Professional CD-ROM. (Like SYSPREP, RIPREP is intended to install only Windows 2000 Professional.)

Figure A.1 RIS Welcome screen.



The next screen prompts you for the location of the RIS file structure. You should indicate a directory on the local file server but it cannot be the same drive as the server's operating system and it *must* be an NTFS formatted partition. After selecting your location and clicking Next, you are

prompted for the response settings of your RIS server. You can select whether the RIS server will respond to client requests, and whether those requests must come from known clients. This screen is shown in Figure A.2.

Next, you are prompted for the location of the Windows 2000 files. This can be copied from a network share or from the original CD-ROM. The default location will be the drive letter of your CD-ROM drive. After this, the RIS service copies the files into the new remote installation share. After completing the RIS Wizard, you are ready to begin RIPREP on your selected master computer.

Figure A.2 Client support settings.



Windows 2000 Setup Phases

When you install Windows 2000, you will encounter three phases during setup:

- WINNT
- Text mode
- Graphical User Interface (GUI) mode (including SYSSETUP and OCMANAGER)

WINNT Phase

During this setup phase, the Winnt32.exe (or winnt.exe) executable file is in control of the processes running on the computer. The following processes are completed:

- Files that are listed in DOSNET.INF are copied to two directories: C:\\$WIN_NT\$.~BT and C:\\$WIN_NT\$.~LS
- A new boot sector is written to C:\\$WIN_NT\$.~BT\BOOTSEC.DAT
- An entry is placed in the BOOT.INI file to boot to BOOTSEC.DAT

Text Mode

After the WINNT phase is completed, the text mode phase begins. Text mode is indicated by a blue screen with white lettering. The primary purpose of the text mode phase is to install the basic Windows 2000 operating system. During this phase:

- The end-user license agreement (EULA) must be accepted by the end-user or scripted as agreed
- The HAL type is detected or selected by the installer and installed
- Power management is detected or selected by the installer and installed
- Storage controllers are detected or selected by the installer and installed

GUI Mode

Text mode ends with a reboot and leads into the GUI (Graphical User Interface) mode. There are two parts of GUI mode:

- SysSetup (System Setup) installs the remaining Windows 2000 operating system
- OCManger (Optional Component Manager) installs optional components

The optional components are installed in an orderly fashion. The list of the components that can be selected are found in SYSOC.INF. Security is set up, then Plug and Play (PNP) devices are installed. Regional settings are selected. Then the name of the user, the company, and the computer are added. Finally, licenses are added.

For IT Professionals**Network Monitor**

Many times, Network Monitor is not installed as part of the base operating system. This is a unique tool since it requires both a driver and an application to be set up on a computer. To install Network Monitor after the server's installation is complete, make certain to have a copy of the Windows 2000 CD's i386 directory available in either the original installation location or another known location.

1. Click Start | Settings | Control Panel.
2. Double-click the Add/Remove Programs icon.
3. Double-click Add/Remove Windows Components.
4. Click Next.
5. Click Management and Monitoring Tools.
6. Click the Details button.
7. Check the Network Monitor Tools check box.
8. Click OK.
9. Click Next.
10. Click Finish and Close to exit, then exit out of Control Panel.

Both the network monitor and the network monitor tools are now installed on the computer. You can now use this tool to measure network traffic for the computer.

Installing the Active Directory

The Active Directory is installed on each and every domain controller. Those servers that do not need to be domain controllers are simply members of the Active Directory, dependent upon the domain controllers to provide them with Active Directory access, so they do not need to have it installed on them.

There are many decisions to be made about the Active Directory installation. You need to know which domain to install first. You need to know whether a domain needs to be upgraded, and which of the upgraded servers to migrate first and thereafter.

Which Domain First?

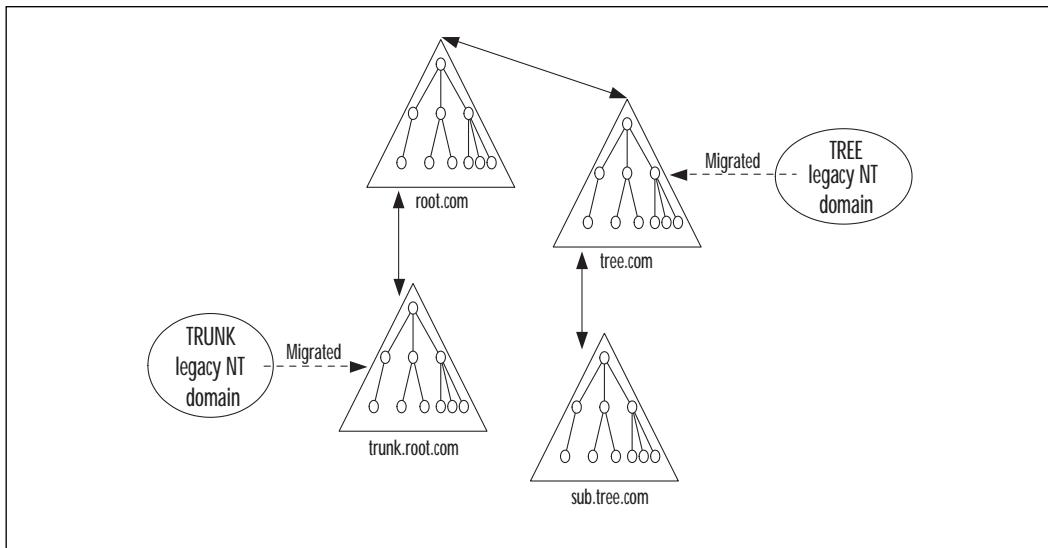
Regardless of whether you are upgrading an existing Windows NT network, or you are installing a fresh Windows 2000 network, you will need to decide which domain to install first. You should follow these rules in determining which domain to install, and in what order, for each forest:

- The root domain should always be installed or migrated first.
- If you are upgrading an existing domain to Windows 2000, you must migrate the PDC first. Then you should upgrade all the BDCs and add any new domain controllers before moving to the next domain.
- If there are any subdomains of the root domain, do those next. Follow the namespace down the tree from the root until it ends for all subdomains before installing the next namespace.
- If there are other namespaces in the forest, select one and begin with the root domain of that namespace.
- Follow that namespace until it ends for each subdomain, and until they are all migrated or installed.
- Repeat this process for each namespace until the forest is fully installed.

For example, imagine that you have a forest as shown in Figure A.3. In this forest you will migrate the domain `trunk.root.com` and the second namespace, `tree.com`, from existing NT domains that are named TRUNK and TREE, respectively, with NetBIOS naming. In addition, you will install a new `root.com` domain and a domain called `sub.tree.com` with new domain controllers.

1. Install the new `root.com` domain, and ensure that the domain controllers are all installed within it.
2. Migrate the existing TRUNK NT domain to `trunk.root.com`. Ensure that the PDC and all BDCs are migrated before migrating the next domain.
3. Migrate the existing TREE NT domain to `tree.com`. Ensure that the PDC and all BDCs are migrated before going to the next domain in the forest.
4. Install a new `sub.tree.com` domain, and ensure that the domain controllers are all installed within it.

Figure A.3 Migration and installation of a forest.



Which Server First?

Once you're ready to install the first server in a domain, you need to know which one to start with. The first server to install into a domain is a domain controller (DC). No matter what, the first DC in a domain will take on the five Flexible Single Master of Operations (FSMO) roles, and is a Global Catalog server.

If you are migrating a Windows NT domain, the first NT server to upgrade is the PDC. The PDC will become an Active Directory DC and take on the FSMO roles, including the PDC Emulator FSMO role. The NT domain user and machine account data will be migrated into the Active Directory, too.

Migrations require some preparation. Because the users, groups, and machine accounts are upgraded from the Windows NT domain Security Accounts Manager (SAM) database, the Active Directory domain will inherit any problem accounts that exist. Problem accounts can be those that do not have passwords, have not been used in a lengthy period of time, groups without members, and accounts that do not use the enterprise naming convention. Preparation steps include:

1. Review the existing user accounts and delete any that have not been used in over 90 days, and remove their associated home directories.

2. Review the existing group accounts and delete any empty ones. Also, consolidate any groups that can be consolidated.
3. Remove machine accounts that have not been used in over 90 days.
4. Back up the PDC twice.
5. Verify that the hardware is compatible with Windows 2000.
6. Verify that applications are compatible with Windows 2000.
7. Download any Beginning Input Output System (BIOS) updates from the BIOS manufacturer's or server vendor's Web site.
8. Download any Windows 2000 drivers required for legacy hardware.
9. Validate the domain security policies.
10. Simplify protocols to TCP/IP only, if possible.
11. Update existing Windows NT OS and the applications with service packs.
12. Convert the file system to NT File System (NTFS).
13. Delete any unnecessary files on the server, such as temporary Internet files, etc.

DCPromo

Installing the Active Directory is done through the DCPromo.exe application, also known as the Active Directory Wizard. The Active Directory Wizard will both promote a member server to a domain controller and demote a domain controller to a member server. If you've inadvertently installed a server into the wrong domain, you can uninstall the Active Directory with DCPROMO.EXE at any time—effectively demoting a domain controller to a member server—without having to reinstall the server completely. For those who are familiar with Windows NT, this is a major improvement!

The difference between a domain controller (DC) and a member server is that a DC carries a copy of the domain partition of the Active Directory locally, but a member server must contact a DC in order to access it.

When you run DCPROMO, you will need to know what role the DC will be playing in the Active Directory forest and other information.

- Is this the first DC in the domain?
- Is this the root domain for a domain namespace?
- Is this the first domain in the forest (e.g., the root domain)?

- Will this server be a DNS server?
- Is the server a DNS client?
- If a new domain, what is its DNS domain name?
- If a new domain, what will be the NetBIOS name?
- Where will the Active Directory files be located?
- Where will the system volume be located?
- Will security be relaxed for NT Remote Access Service (RAS) backward compatibility?
- What is the password to be used on this server to restore Active Directory?

The first screen that you reach when running DCPROMO, the Welcome screen, is shown in Figure A.4. You can bypass this screen by clicking Next.

The second dialog lets you select whether this DC is the first in a new domain, or if it will be a new DC in an existing domain. From this screen forward, the answer you give to the question will determine what screen you will encounter next. For example, if this is the first DC that you install in the Active Directory forest and it will run the DNS service, you will come across the dialogs and answers in Table A.2.

Figure A.4 DCPromo Welcome screen.

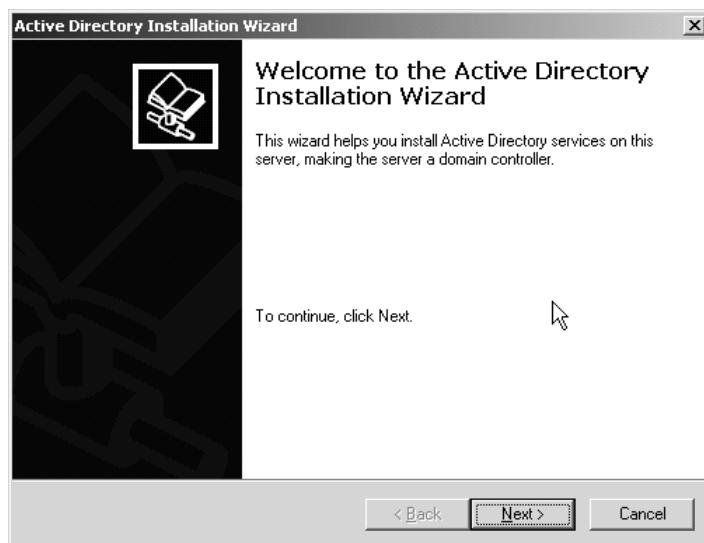


Table A.2 Active Directory Wizard

Dialog Screen	Options	Actions
Welcome Screen	None	Click Next.
Domain Controller Type	<ul style="list-style-type: none"> ■ First domain controller in a new domain. ■ Domain controller in an existing domain. 	Select First DC in New domain and click Next.
Create Tree or Child Domain	<ul style="list-style-type: none"> ■ First domain in a new domain tree. ■ Child domain of an existing domain tree. 	Select First domain in a domain tree and click Next.
Create or Join Forest	<ul style="list-style-type: none"> ■ Create a new forest. ■ Place this domain tree in an existing forest. 	Select Create a new forest and click Next.
DNS configuration	<ul style="list-style-type: none"> ■ Configure this computer as a DNS client. ■ Install this server as a DNS server. 	Select the option to install the server as a DNS server and click Next.
DNS domain name	<ul style="list-style-type: none"> ■ Enter the domain name in the DNS namespace format of domain.com. 	Type in the domain name and click Next.
NetBIOS domain name	<ul style="list-style-type: none"> ■ Enter the NetBIOS domain name in the NET-BIOS format of DOMAIN. This will automatically default to the first 15 characters of the first section of the DNS domain name provided in the previous dialog. 	If changing the name, type in the domain name and click Next.
Active Directory files location	<ul style="list-style-type: none"> ■ Enter the location for the database and logging files for Active Directory. This will default to the system partition within the WINNT directory. It is recommended that the database and logging files should exist on separate disks for recoverability. 	If changing the location as recommended, type in the new path(s) and click Next.

Continued

Table A.2 Continued

Dialog Screen	Options	Actions
System Volume	<ul style="list-style-type: none"> ■ Enter the location for the system volume. This defaults to WINNTSYSVOL. The system volume is replicated to all DCs. It will grow over time because it holds scripts, group policies, and other files that enable logon, so place the directory on a partition with room for growth. 	If changing the location, type in the new path and click Next.
Security	<ul style="list-style-type: none"> ■ Standard Windows 2000 security ■ Relaxed permissions for backwards compatibility with Windows NT 4 <p>Remote Access Servers (RAS). Select this option only if planning to use Windows NT 4.0 RAS.</p>	As a new Windows 2000 domain, there is no need for backward compatibility, so select the standard security and click Next.
Directory Services Restore Password	<ul style="list-style-type: none"> ■ Enter and confirm the administrative password that will be used to restore the Active Directory on this DC. 	Type in the password and verify it, then click Next.
Summary page	<ul style="list-style-type: none"> ■ Provides a summary of the options selected during the Active Directory Installation. 	Review the page and click Next.

Installing the Recovery Console

Windows 2000 Server provides a way to restore a domain controller without having to rely on reinstalling the server and restoring data from a tape backup—the Recovery Console. The Recovery Console is a command line console that you can use to

- Format a hard disk
- Manipulate files on an NTFS hard disk

- Reconfigure a service
- Start or stop a service

The Recovery Console is not installed by default. If you do not install the Recovery Console, you can still access it by booting the Windows 2000 Setup program from the Windows 2000 CD-ROM. When you install the Recovery Console, it is listed as an option in the boot menu. To install it:

1. Log on Windows 2000 Server as an administrator.
2. Open a command prompt window by clicking the Start menu, select the Run option, type **CMD** into the Run box, and press Enter.
3. From the Windows 2000 source files in the i386 directory, type **WINNT32 /CMDCONS**.
4. Follow the dialog screens until the command console is installed.

The Recovery Console will be available as an option in the boot menu the next time the server is started. To run it, simply select that option at startup.

Populating a Domain with Organizational Units (OUs) and Objects

Each domain has a number of default containers. These default containers are not intended to be the only containers in the domain, nor are they appropriate for placing your new users and groups. The default containers play the roles listed in Table A.3.

Table A.3 Default Containers for a Domain

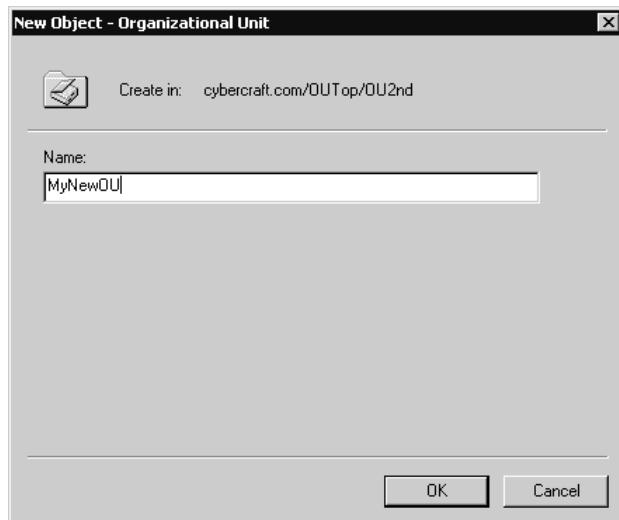
Container Name	Role
Builtin	Contains backward-compatible (Windows NT) security groups.
Computers	Contains member computers of the domain after their domain is upgraded from Windows NT. Can contain new member computers, as needed.
Domain controllers	Contains all Active Directory DCs belonging to that domain.
ForeignSecurity Principals	Active Directory places objects representing security principals (usually just users and groups) from domains outside the forest that are trusted through a trust relationship.
Users	Contains the initial default users and group accounts. Upgraded user and group accounts are initially placed here.

Creating an OU

To create an OU, you start in the Active Directory User and Computers console. The first OU you will create should be at the top level of the domain.

1. Right-click the domain object at the top of the tree hierarchy.
2. Click New from the pop-up menu options.
3. Select Organizational Unit.
4. The New object dialog will appear, as shown in Figure A.5.
5. Type a name in the box for the OU.
6. Click OK.

Figure A.5 Creating a new OU.



Create an OU for Hidden Objects

If you do not want objects to be seen by every user, you can use an OU to hide them. This can be useful if there are some highly secure objects (for example, objects containing confidential HR data) that no one should have access to except certain users. To hide objects in an OU:

1. Enable the Advanced Features of the Active Directory Users and Computers console by clicking the View menu and selecting Advanced Features.

2. In the Active Directory Users and Computers console, right-click the OU that will contain the hidden objects.
3. Select Properties.
4. Click on the Security tab.
5. Remove all permissions.
6. Click Advanced.
7. Clear the checkbox for Inherit permissions from parent.
8. Click OK.
9. You should be back in the Security dialog. Add the groups and users who need rights to the OU to see its contents.

Delegating Authority

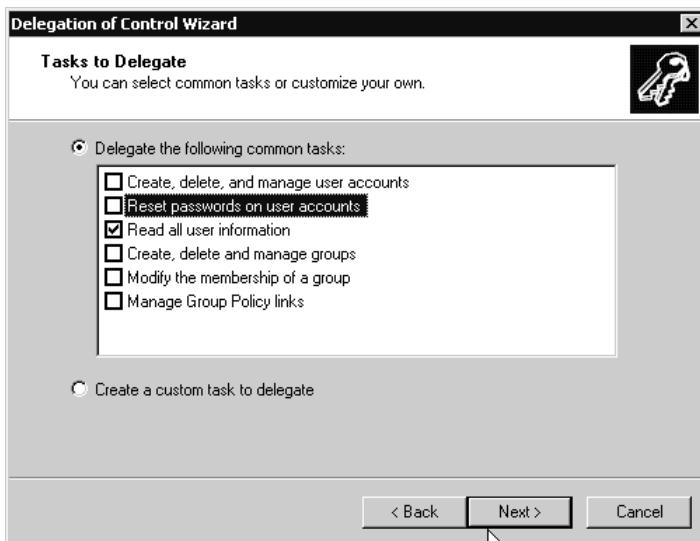
Once you have an OU, you can delegate the authority to manage that OU and its contents to other users. Many legacy Windows NT domains were created in order to create boundaries between different administrative groups. This was the only way to achieve that separation without a third-party add-on. However, with the Active Directory, authority can be delegated within the OU hierarchy, and separation of domains is no longer required.

The Active Directory Users and Computers console contains a Delegation of Control Wizard, shown in Figure A.6. This wizard simplifies the process of delegating authority, including both predefined roles and the ability to customize what is being delegated. To delegate control:

1. Start the Active Directory Users and Computers console.
2. Right-click on the OU that will be delegated.
3. Select Delegate Control from the pop-up menu.
4. At the Welcome dialog, click Next.
5. In the Group or User Selection dialog, click Add.
6. Select the group or user to whom you are granting authority.
7. You will be returned to the Group or User Selection dialog; click Next.
8. In the Predefined Delegations window, select the role that the group or user will play, or select Do customized delegation. Click Next.

9. If you are doing customized delegation, you can select the entire OU, or a specific list of objects within it. You will also be prompted to select which permissions are being granted.
10. When complete, click Finish at the Summary dialog.

Figure A.6 Delegation of Control Wizard.



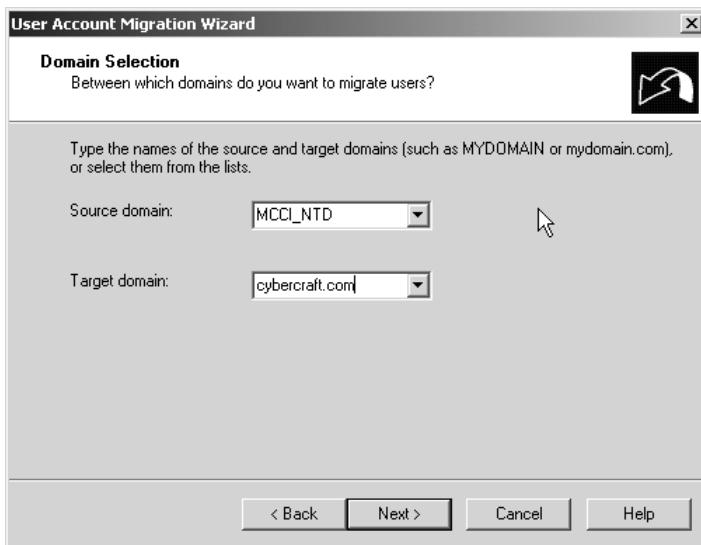
One type of permission is in high demand for large enterprises—resetting passwords. Why? Large enterprises require strict security because they have so many end users. They execute this security by requiring passwords on all their systems, by requiring frequent password changes, and by requiring lengthy passwords with odd characters. The end result is that users forget their long, odd-character-including, recently changed passwords—especially if they have more than one password to access more than one computer system. To become productive, those users need to log on to the network; so, the enterprise needs to provide a way for those users to get a new password to replace their forgotten one. It is not productive for the highest level administrators to change passwords, nor is it secure if everyone who is given the right to change passwords is also granted full administrative authority. This is where Delegation of Control Wizard comes in. One of the predefined roles is set for Reset Passwords. For all the Call Center, Help Desk, or other IT folks who need to reset passwords, simply create a group, grant that group the Reset Password predefined role, and then add those users to that group. If only the rest of IT were this easy.

Creating a User Account

For users to begin logging on, they need user accounts. If you have upgraded a Windows NT Primary Domain Controller (PDC) to Windows 2000 Active Directory DC, those users will be migrated into the Users container of the domain. Then move them into your OU hierarchy.

If you created a new Active Directory domain and want to migrate user accounts from a legacy Windows NT domain, then you need to use a tool such as the Active Directory Migration Tool (ADMT). ADMT provides a GUI interface and migration wizards for domain components; the user migration wizard, shown in Figure A.7, is used to migrate objects from an existing NT domain to a Windows 2000 domain. This is helpful when merging domains. ADMT is downloadable from Microsoft's Web site at <http://download.microsoft.com/download/win2000srv/Install/1.0/NT5/EN-US/ADMT.exe>.

Figure A.7 ADMT.



Once you install the Active Directory Migration Tool, you can select the components that you want to migrate using the wizards for those components. For example, to migrate member computers, select the Action menu and then the Computer Migration Wizard.

ClonePrincipal is another tool that you can use to migrate multiple user accounts. The ClonePrincipal tool uses a customizable Visual Basic script for migrating security principals from an existing domain to an Active Directory domain.

If you are adding a few users to the Active Directory, the best way to go about it is to create them manually in the Active Directory Users and Computers console. To create a new user account:

1. Right-click on the OU that will contain the new user.
2. Select New from the pop-up menu.
3. Select User from the sub-pop-up menu.
4. Type in the user's name—first, initial, last—and logon names in the dialog shown in Figure A.8.
5. Click Next.
6. Type the user's password and verify it.
7. Select the user's password options in the dialog shown in Figure A.9.
8. Click Next.
9. Review the summary and click Finish.
10. To configure the user's detailed properties, double-click the new user object and make any changes or additions to the dialog shown in Figure A.10.

Figure A.8 User creation dialog.

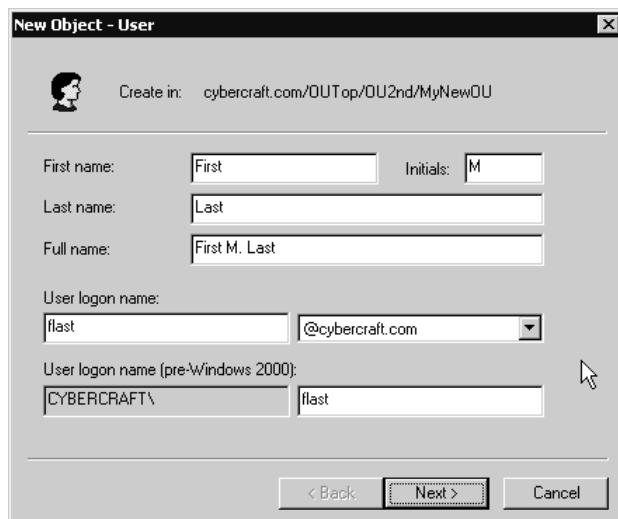
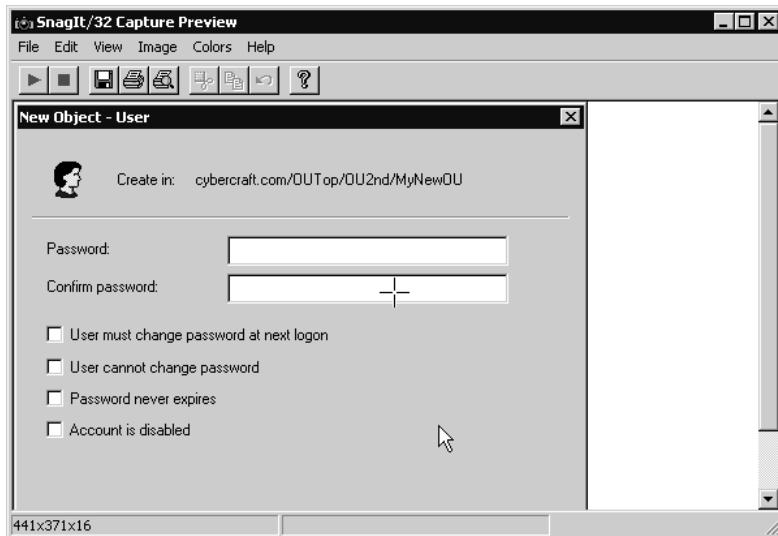
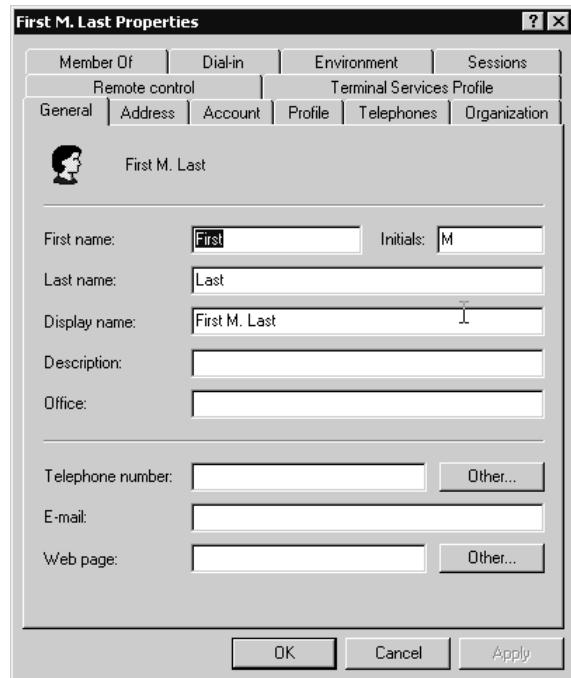


Figure A.9 User password dialog.**Figure A.10** User properties dialog.

Creating Groups

When you create a group, you need to know whether your domain is in mixed mode or native mode. Mixed mode is the default state of a domain—it accepts Windows NT Backup Domain Controllers (BDCs) as part of the domain. A domain will remain in mixed mode even if all the servers are Windows 2000 servers until the network administrator switches it to native mode. Once a domain has been changed to native mode, it cannot be changed back to mixed mode. To switch a domain to native mode:

1. In the left pane of the Active Directory Users and Computers console, click on the domain.
2. Select the Action menu.
3. Select the Properties option.
4. Click the General Tab.
5. Click the Change Mode button.

If you intend to use nested groups—where groups are members of other groups—you must switch your domain to native mode. Nested groups are not applicable to NT domains, so they cannot be used in mixed mode. There are new types of groups available in the Active Directory domains, too:

Local groups These groups can be granted access to any resource on a particular server. They can contain members from anywhere in the forest or trusted domains. A local group in native mode can contain global groups and universal groups as members. Local groups work the same as they did in Windows NT when the domain is in mixed mode.

Domain local groups These groups can be granted access to any resource in a single domain. They can contain members from anywhere in the forest or trusted domains. A domain local group in native mode can contain global groups and universal groups as members.

Global groups These groups can be granted access to resources in any domain in the forest or a trusted domain. They can contain members only of the local domain. In native mode, a global group can contain other global groups or universal groups as members. Global groups have the same capabilities as they did in Windows NT when in mixed mode.

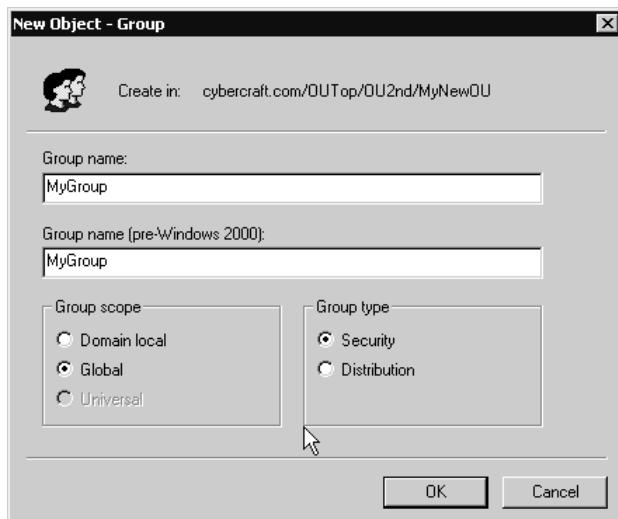
Universal groups These groups can be granted access to any resource in a trusted domain that is in native mode. A universal group can contain members of any domain within the forest.

To create a group:

1. In the Active Directory Users and Computers console, right-click the OU where the group will reside.
2. Select New.
3. Select Group from the pop-up menu.
4. Specify the type of group you want. To assign permissions to a group, do not select Distribution as the Group Type. Specify the name of the group. The group dialog is shown in Figure A.11.
5. Click OK.

To nest a group, you simply make one group a member of another group. This is done through the members tab when you view a group's properties. As you can see from Figure A.11, a group is defined through its scope and its type. The scope refers to whether the group is a domain local, global, or universal group. (Local groups are available on member servers and workstations, and the rest are available on DCs in the Active Directory.) The type refers to whether a group is a security group (it will be granted permissions to resources) or a distribution group used for e-mail lists. Since you will most likely be using your groups for resource access, select Security. If you want to use the universal group, switch your domain to native mode first.

Figure A.11 New Group dialog box.

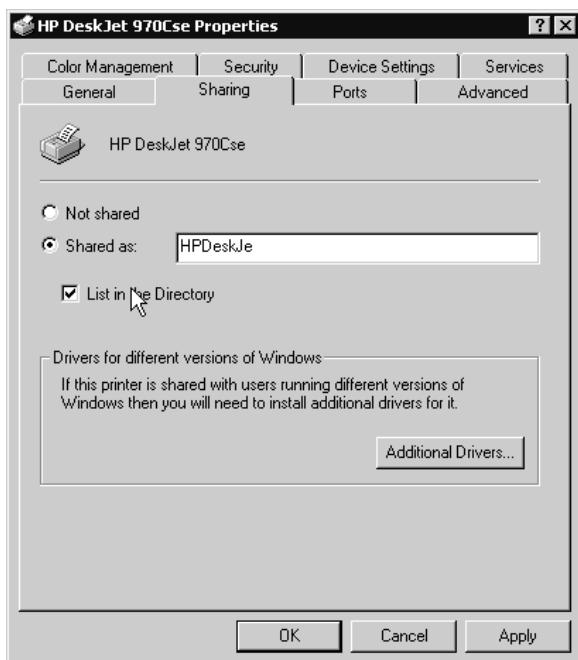


Publishing Printers

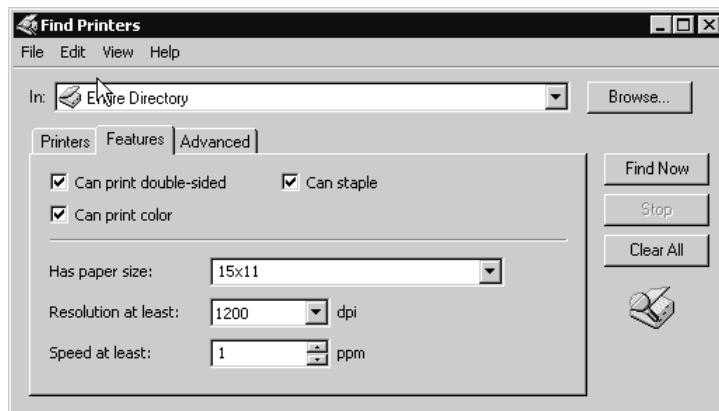
Printers are published in the Active Directory, by default, through the Add Printer Wizard. Publishing a printer enables it to be searched for using an Active Directory executed query. If the printer is not published in the Active Directory, follow this procedure:

1. On the server from where the printer is shared, open the printers folder.
2. Right-click on the printer object.
3. Select Properties.
4. Click the Sharing tab.
5. Check the box for List in Directory shown in Figure A.12.

Figure A.12 Publishing a printer in the Active Directory.



After a printer has been published, a user can query the directory and look for certain characteristics for that printer. These include duplex printing, color, location in the network, and so on. The user's query will look similar to Figure A.13.

Figure A.13 Querying the directory for printers.

Publishing Folders

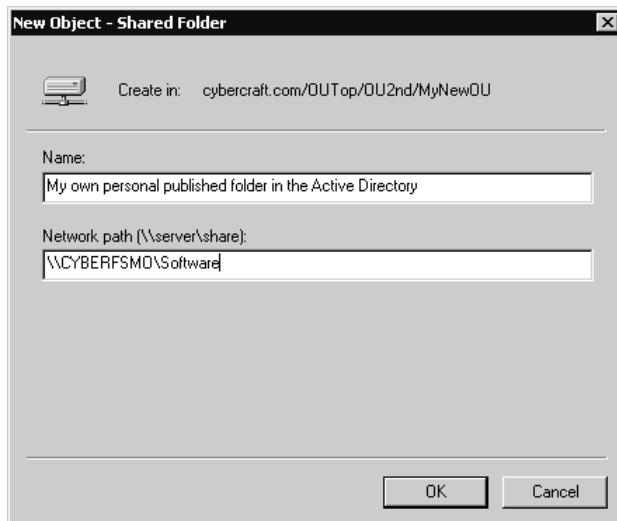
Unlike printers, folders are not published in the Active Directory by default. Once a folder is published in the Active Directory, users can search for it in the Active Directory through queries or even browsing. To publish a folder, it must first be shared. Sharing folders is a simple matter of right-clicking the folder in the Windows Explorer and selecting the Sharing tab, then setting the permissions for specific users and groups. To further publish the folder in the Active Directory:

1. In the Active Directory Users and Computers console, right-click on the OU where the folder will be published.
2. Select New.
3. Select Shared Folder.
4. Type the Universal Naming Convention (UNC) for the share—it's in the format of \\server\share, as shown in Figure A.14.
5. Click OK.

TIP

If you want to publish folders that are not online, you can go ahead and do so. The Active Directory does not check for the folder when it publishes. This is helpful for offline publishing, but it can also be an issue if you have a typo. Make certain to double check that you've typed the published folder name correctly if it is offline. If the folder is available online, then verify that it was published correctly by browsing for it in the Active Directory and expanding it.

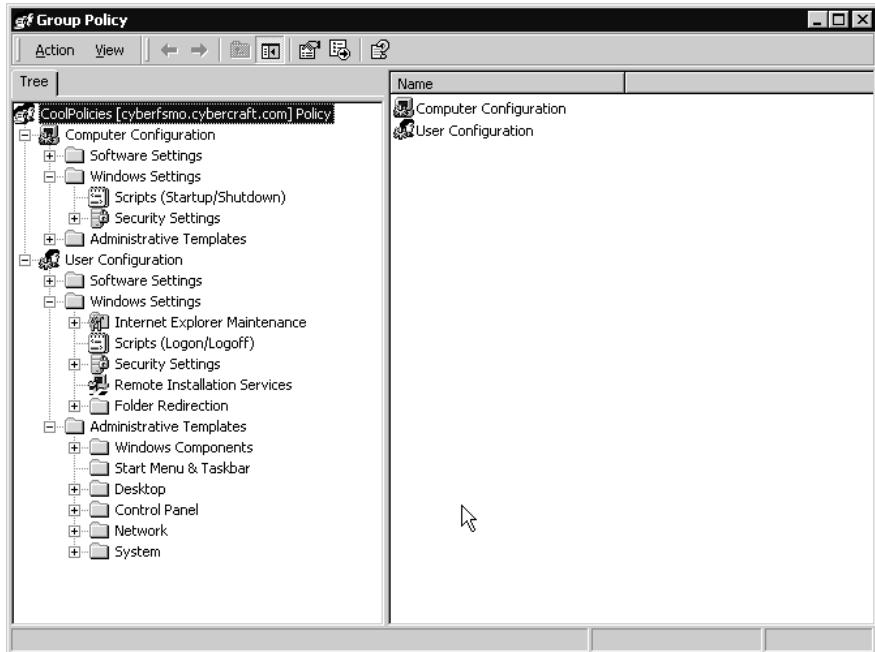
Figure A.14 Published folder dialog.



Applying a Group Policy

Group policies can be applied to a domain, a site, or an OU. If you've created an OU hierarchy that is functionally designed for a nested group policy application, then you will most likely and most often apply group policies to OUs. To create a group policy:

1. Right-click the OU in the Active Directory Users and Computers console. (Alternatively, you can right-click the domain or site in the Active Directory Sites and Services console and follow this same sequence to create a Group Policy.)
2. Select Properties.
3. Click the Group Policy tab.
4. Click New.
5. Name the policy.
6. Click Edit.
7. The Group Policy Editor will start as shown in Figure A.15. Configure the policy options for this Group Policy.
8. Exit the Group Policy Editor after you have configured all the options you require. You return to the Group Policy tab for the OU.
9. Click Close.

Figure A.15 Group Policy Editor.

Setting Up Sites

IP subnets can be assigned to sites immediately following the installation of the first DC. The value in doing this step first is realized when further Windows 2000 computers are installed. Each Windows 2000 computer contacts the Active Directory to find a DC that is within its own site for authentication purposes. When the Active Directory is installed, it automatically will install the DC into the site that is associated with the server's own IP subnet. If a site is not identified with the server's IP subnet, then the server is placed within the default site. For a company with a site in Florence, Italy and another in Sydney, Australia, a DC installed into the wrong site could cause logon delays and excessive wide area network bandwidth consumption. It is best to ensure that all DCs are placed into the appropriate sites at the earliest opportunity. To associate IP subnets with a site, you first need to create the sites and then create the IP subnets within each one.

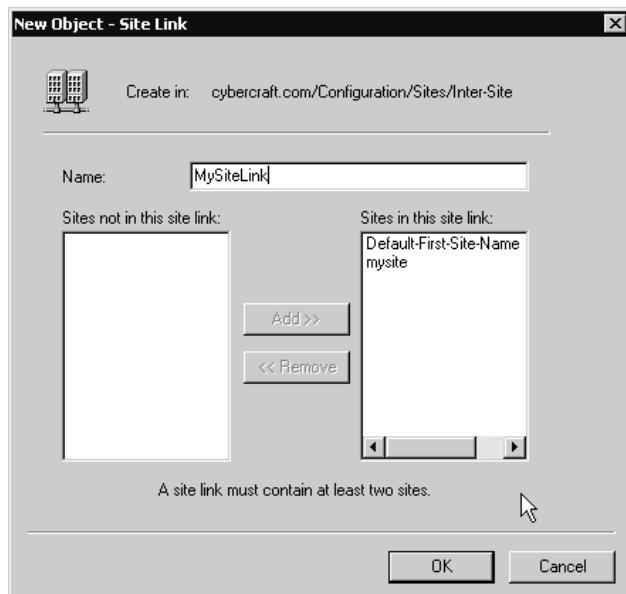
1. In the Active Directory Sites and Services console, right-click on Sites.
2. Select New Site.
3. Type the name for the site.

4. Select a site link (you can change this later, if you need to).
5. Click OK.
6. Right-click on Subnets.
7. Select New Subnet.
8. Type the IP subnet address and subnet mask.
9. Click OK.

When you have multiple sites, you need to create site links, site link bridges, and connection objects to enable them to transfer information. To create the site link:

1. In the Active Directory Sites and Services console, navigate below the Sites container to the Inter-Site Transports.
2. There are two transports listed—IP and SMTP. Right-click on the transport you will use. Most often, you will only use IP.
3. Select New Site Link from the pop-up menu.
4. In the New Site Link dialog, select the sites that will participate in this site link and type the name of the site, as shown in Figure A.16. You must place at least two sites in each site link.
5. Click OK.

Figure A.16 New site link.



For IT Professionals

Connection Object Management

Even though you have created site links, your DCs will need to have connection objects in order to synchronize updates across the site link. Think of a site link like a road for traffic, but without any cars. The connection objects are like the cars that carry traffic across the road.

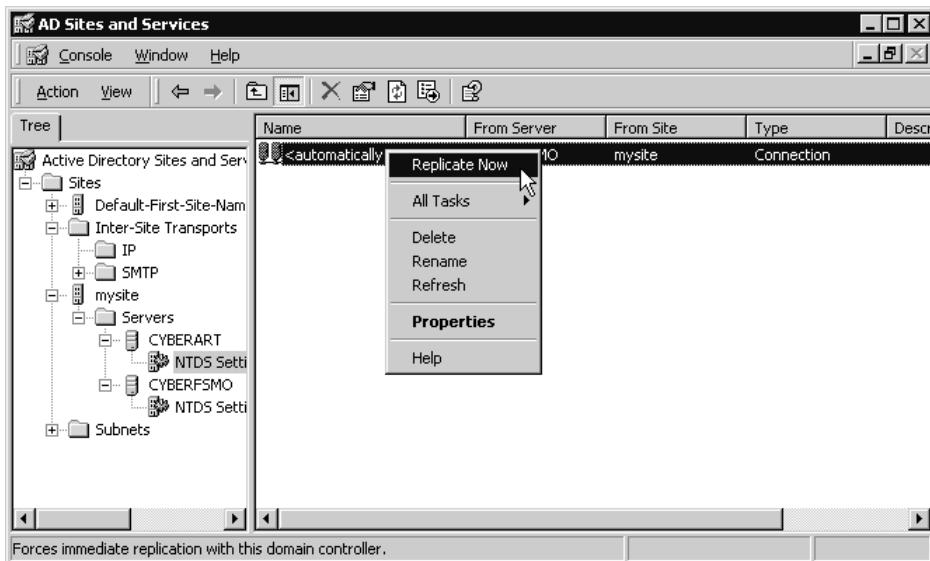
It is easy to ignore connection object management because objects are generated automatically by the Knowledge Consistency Checker (KCC) within any particular site. They are *not* generated automatically across sites.

Be careful when you move servers from one site to another! If you move a server from one site to another, the connection objects that were created by the KCC will move with it and never be changed thereafter. These connection objects may not be desirable if you want to manage traffic over that site link with bridgehead servers or by reducing the number of intersite connections.

If you are creating bridgehead servers, you will need to check each server in each site to ensure that there are no connection objects created between nonbridgehead servers in the different sites. You will also need to make sure that there is only one connection object in the bridgehead server's NTDS Settings object pointing from the other site's bridgehead server. NTDS stands for NT Directory Service. Each domain controller has an NTDS Settings object.

An administrator may wish to force replication to make recent changes synchronize throughout the forest. To do this, the administrator can use the Active Directory Sites and Services console to access the Replicate Now option, shown in Figure A.17. Replication is forced by right-clicking the connection object below the NTDS Site Settings of the server that you want to have synchronized.

Figure A.17 Replicate Now.



Installing and Configuring Windows 2000 Components

Once the Active Directory is installed on the Domain Controllers, your work is still not done. You will need to install or configure other Windows 2000 services such as the Domain Name System, Remote Access Services, and Terminal Services.

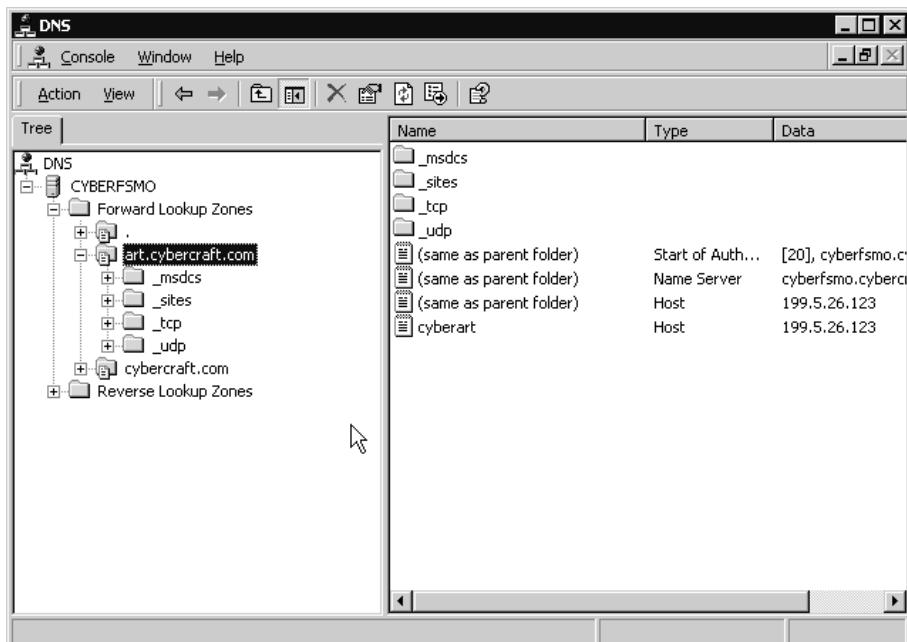
Configuring DNS

To start configuring DNS, you will want to start the DNS Manager, located in the Administrative Tools menu.

1. In the DNS Manager, shown in Figure A.18, select the server that will be configured for DNS.
2. Click the Action menu.
3. Select Configure the server.
4. The Configure DNS Server wizard will start. Click Next at the Welcome dialog.
5. Select whether the server is the first DNS server on the network or not. Click Next.

6. Create a Forward Lookup Zone. This is the domain name of the zone that the server will manage.
7. Select whether this zone is Active Directory Integrated, Standard Primary, or Secondary. If the server is not a DC, you will see that the first option, Active Directory Integrated, is grayed out. Click Next.
8. State the domain name for the zone and click Next.
9. You are then prompted to create a reverse lookup zone. For DNS experts, this is an In.Addr.Arpa zone, which can look up an IP address and find the domain name—the reverse of a standard zone. It is not necessary to create a reverse lookup zone for Windows 2000 Active Directory to function correctly.
10. The Configure DNS Server wizard completes with a summary page. Click Finish.

Figure A.18 DNS Manager.



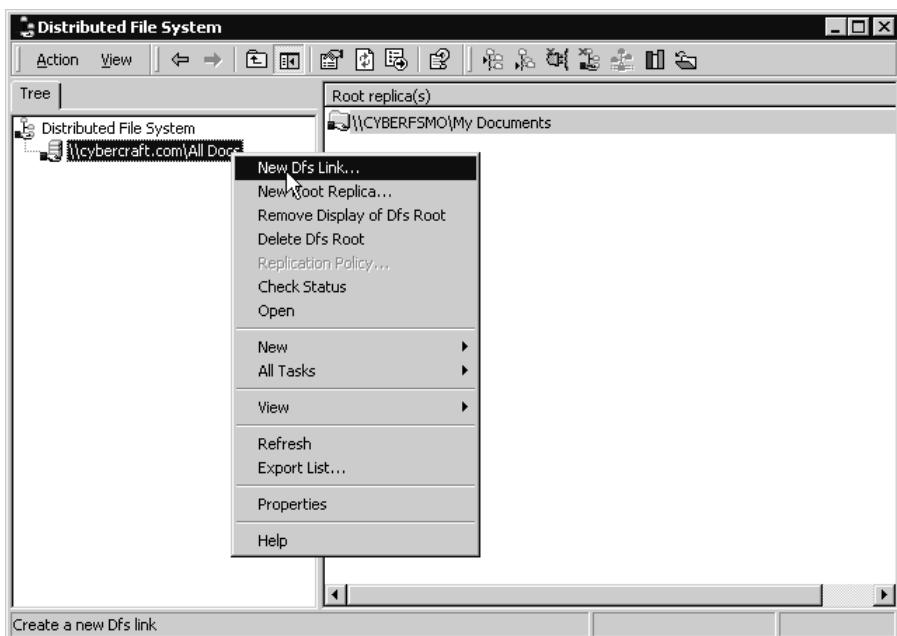
Configuring the Distributed File System

The Distributed file system (Dfs) can be configured in two ways—as an Active Directory stored system, or as a standalone system. To create the Dfs root, start the Distributed file system console from the Administrative Tools menu. When you start the configuration wizard, you will be prompted for the type of system. To store the Dfs topology in the Active Directory, select the Create a Domain Dfs Root option. You will be prompted for the domain that will host Dfs, the server to host Dfs, a shared folder for the Dfs root, and a name for the Dfs root. The summary page of the wizard is shown in Figure A.19.

Figure A.19 Dfs Configuration wizard.



Dfs creates a full mesh topology between all the replicas. Each new replica and every other member of the replica set will share a link. This can create a lot of traffic on the network. To optimize Dfs, you can delete the connections that you don't really need in the Active Directory Users and Computers console. Otherwise, Dfs is managed in the Distributed file system console shown in Figure A.20.

Figure A.20 Dfs MMC.

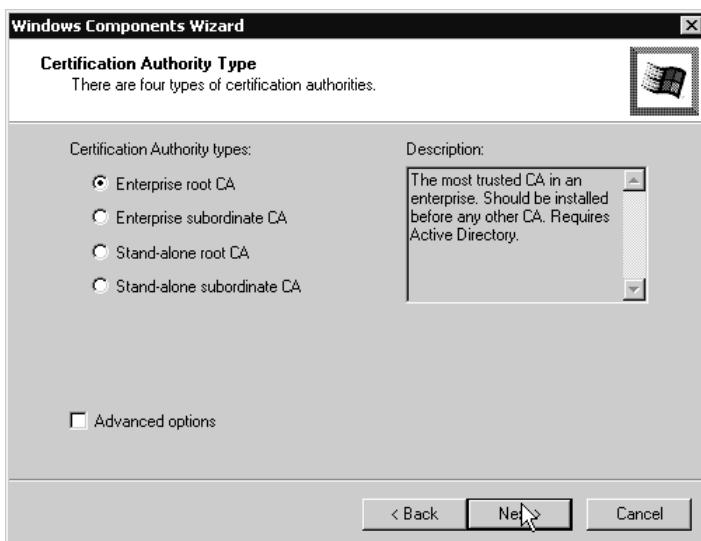
Public Key Infrastructure

The Public Key Infrastructure (PKI) is an authentication method based on digital certificates and certification authority (CA) servers. Windows 2000 provides CA services natively. Once you install a server with CA services, you will not be able to change the role of the server, or the domain to which it belongs. The implementation process of PKI is:

1. Install one or more root CAs in the top-level domains of each Windows 2000 domain tree in the forest. The root CA is placed at the top of a CA hierarchy and is self-signed. It should be configured to issue only subordinate CA certificates. When you install the CA server, you will not be able to rename the server or change its domain membership (whether it is a DC or member server, or which domain it belongs to). You are given four choices for installing the server at the CA services installation, depicted in Figure A.21.
2. Install subordinate CA servers in the child domains to implement certificate policy. Subordinate CAs are issued their certificates from the root CA. These CA servers request a certificate from the root CA. When you install a CA on a subdomain, then the Enterprise Root CA option is grayed out.

3. Configure the CA servers to issue certificates for users. Issuing CA servers should be configured to issue appropriate certificates such as user certificates or session certificates.
4. Configure certificate revocation lists.
5. Configure Group Policy.
6. Configure certificate renewal and enrollment.
7. Issue certificates.

Figure A.21 Creating a CA server.



To create a CA on a Windows 2000 server:

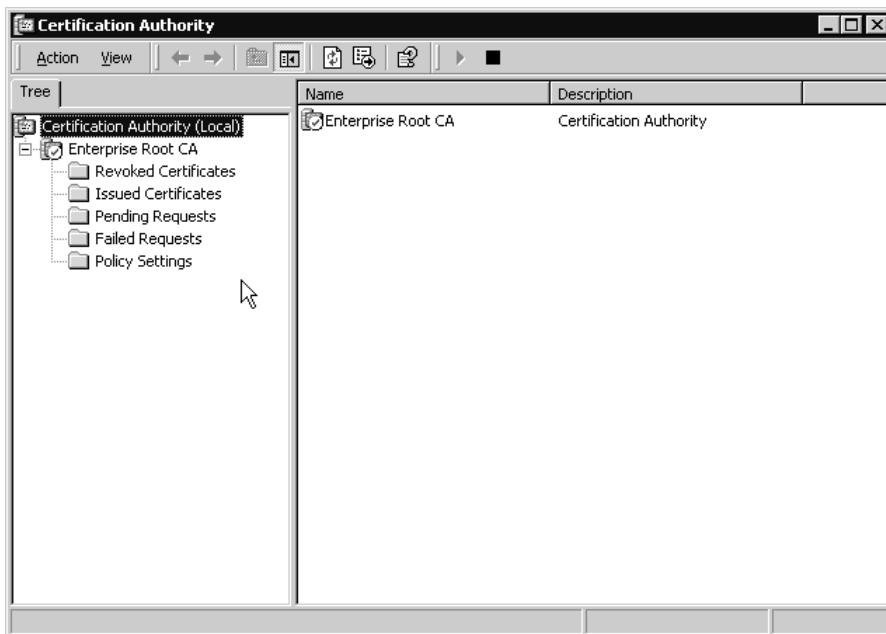
1. Open the Control Panel.
2. Double-click Add/Remove Programs.
3. Select Add/Remove Windows Components.
4. Add Certificate Services.
5. Install an enterprise root CA.
6. You can optionally select Advanced options to specify whether the server is going to be a Cryptographic Service Provider (CSP)—which is responsible for creating and destroying keys and performing cryptographic operations. You can also change the hash algorithm, which detects modifications in message data. You can choose to

use existing public and private keys, and set the key length. When you complete your selections, click Next.

7. Type the name of the CA server and its detailed information and click Next.
8. Specify the Validity Duration for the server. This value states when the CA expires, so carefully consider how long this server will remain in service. Click Next.
9. State the location for the CA database and log files and shared folder. Click Next.
10. If you have IIS running, you will be prompted to stop it. Click OK.

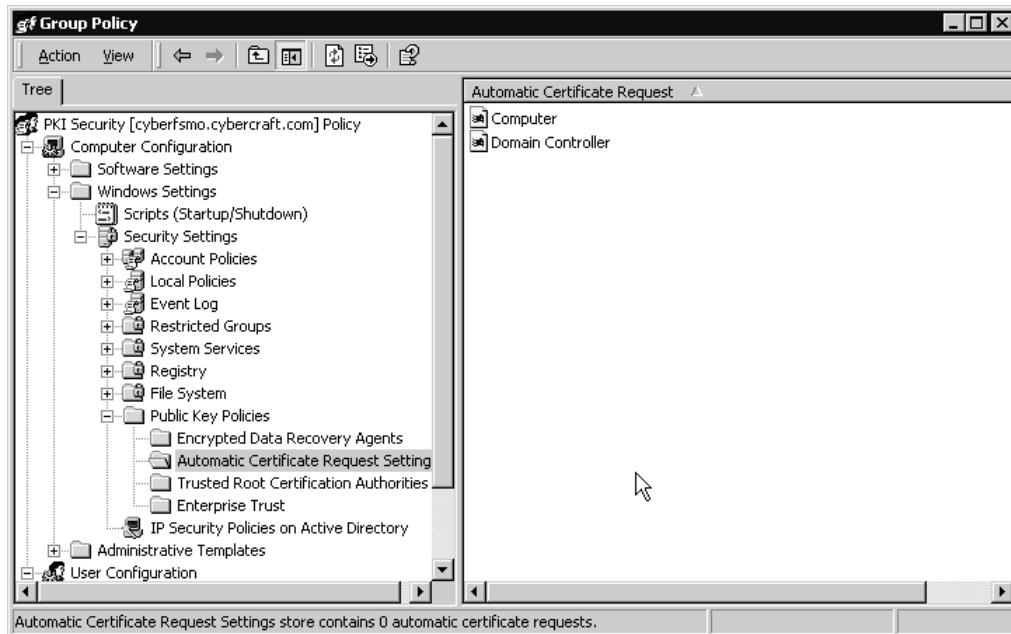
The CA server is managed using the Certification Authority console that is found in the Administrative Tools menu and shown in Figure A.22.

Figure A.22 Certificates management.



PKI policies can be established through Group Policy. These policies are located in the Computer Configuration group policy under Windows Settings\Security Settings\Public Key Policies. This group policy section is illustrated in Figure A.23.

Figure A.23 PKI group policies.



Internet Information Services

Internet Information Services (IIS) is installed by default on every Windows 2000 server, but must be installed as an option on Windows 2000 Professional workstations. To add IIS to a machine that does not have it, use the Add/Remove Programs icon in the Control Panel.

When it is used to serve files to the Web, IIS can create a tremendous load on a server. You can optimize IIS by selecting one of the application protection options for IIS processing of your directory:

- **High (Isolated)** means that the application runs in a separate process.
- **Medium (Pooled)** means that many applications share the same process, thus improving reliability (the default option).
- **Low (IIS Process)** means that the HTML application runs in the same process as IIS. Selecting this can cause IIS to fail if the HTML application fails.

To configure this option for the Web, open the IIS console, shown in Figure A.24. Select the Properties for the Web site.

Click on the Home Directory tab and select the Application Protection drop-down box shown in Figure A.25.

Figure A.24 Internet Services Manager.

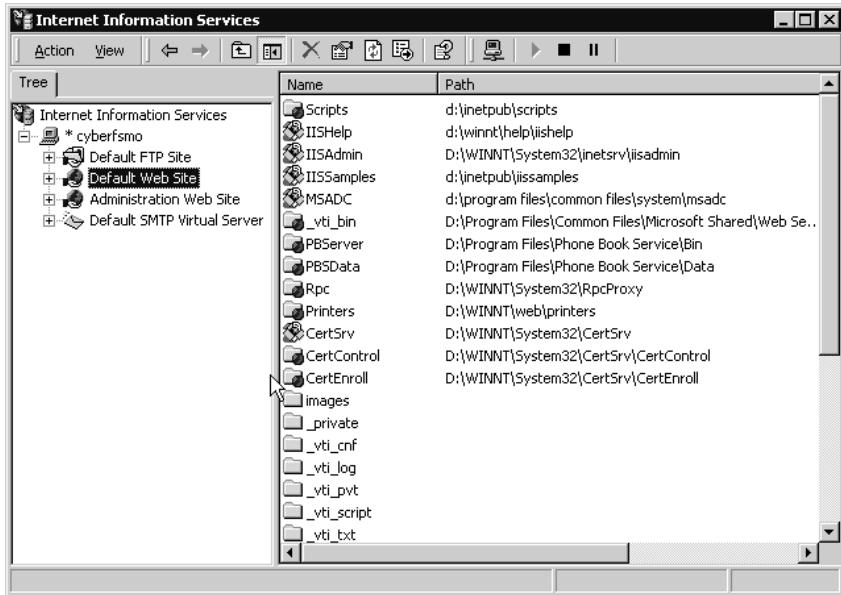
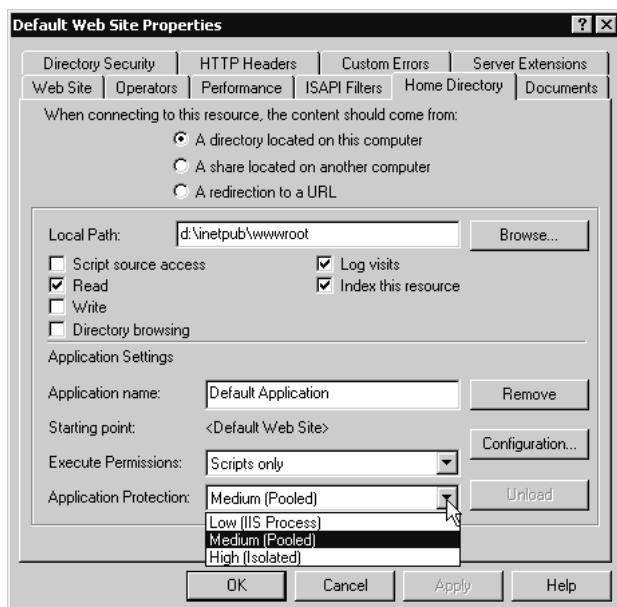


Figure A.25 Configuring IIS bandwidth throttling.



Asynchronous Transfer Mode

Asynchronous Transfer Mode (ATM) is a protocol that is based on cell switching. Cells are small frames, in this case 53 bytes in length. Cell switching is faster than standard packet switching because the small cells do not need to be written to disk as they are being switched throughout an internetwork. Instead, they can stay in random access memory (RAM). ATM typically is implemented as a wide area network backbone technology, but it is slowly permeating local area networks as well.

Windows 2000 supports ATM natively. You can install ATM from the backbone to the workstation. To enable IP over ATM:

1. Open the Control Panel.
2. Double-click Network and Dial-up Connections.
3. Select the Properties tab of the Network Connections dialog box.
4. Double-click the ATM adapter.
5. Select the TCP/IP Protocol and click Enable.

If you are connecting directly to an ATM permanent virtual circuit (PVC), you must configure the Asynchronous Transfer Mode Address Resolution Protocol (ATMARP) client:

1. Open the Control Panel.
2. Double-click Network and Dial-up Connections.
3. Right-click ATM Connection.
4. Choose the Properties tab.
5. Select ATM Call Manager and then its Properties tab.
6. Click Add.
7. Enter the PVC name and Virtual Channel Identifier (VCI) number.
8. Change the Application Type to Default ATMARP.

Terminal Services

Terminal Services are an optional Windows 2000 Server component. In Windows NT 4.0, there was a special *Terminal Server Edition* that was required to run this application service. Now, all Windows 2000 Server editions—Server, Advanced Server, and DataCenter Server—are equipped with an option to run Terminal Services. You can install Terminal Services from the Control Panel using the Add/Remove Programs icon and selecting the Add/Remove Windows Components option.

You should install Terminal Services with one of two situations in mind:

Remote administration Enables servers to be managed remotely from any Terminal Services client over TCP/IP connections. Two Terminal Services connections are included without any licensing requirements or configuration needed.

Application services Enables applications to be available over TCP/IP connections. Terminal Services connections must be configured and licensed in order to be available to users.

The effect of Terminal Services being enabled on a server for remote administration is minimal. However, providing applications to users can create a processing load that increases incrementally for each simultaneously attached terminal services client. Reasons for using the application services can be simply to provide a specific application, to provide a line of business applications to remote offices, or even to create a full desktop of applications for all users to access. You will need to configure the items listed in Table A.4 depending on which way you deploy Terminal Services.

Table A.4 Terminal Services Configuration Requirements

Configured Option	Remote Administration	Application Services
Terminal Services Licensing	Not required	Required
Terminal Licenses Server	Not required	Required
User security	Required for administrators only	Required for all application users
Connections	Not required	Required
Application installation	Not required	Required for each application

To begin, you must install the Terminal Services License Server. If you have the Active Directory installed, you must install the license server on a DC. Otherwise, it can be installed on any Windows 2000 server. To install and configure the Terminal Services License Server:

1. In the Control Panel, open the Add/Remove Programs icon.
2. Select Add/Remove Windows Components.
3. Check the box for Terminal Services Licensing.
4. Select your entire enterprise.

5. Click Next.
6. Click Finish.
7. When complete, you can configure licensing by executing the Terminal Services Licensing console from the Administrative Tools menu.
8. Terminal Services Licensing will locate all Terminal Services servers and list them in its window, shown in Figure A.26.
9. To activate a server, right-click on the server and select Activate from the pop-up menu.
10. You can change licensing options by right-clicking on a server and selecting Properties from the pop-up menu, illustrated in Figure A.27.

Figure A.26 Terminal Services Licensing.

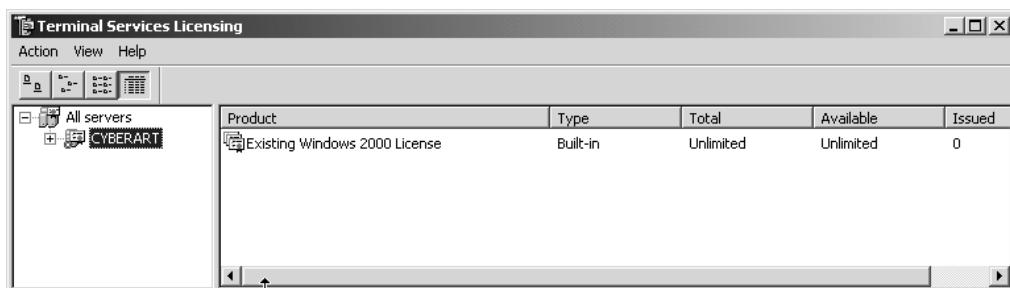
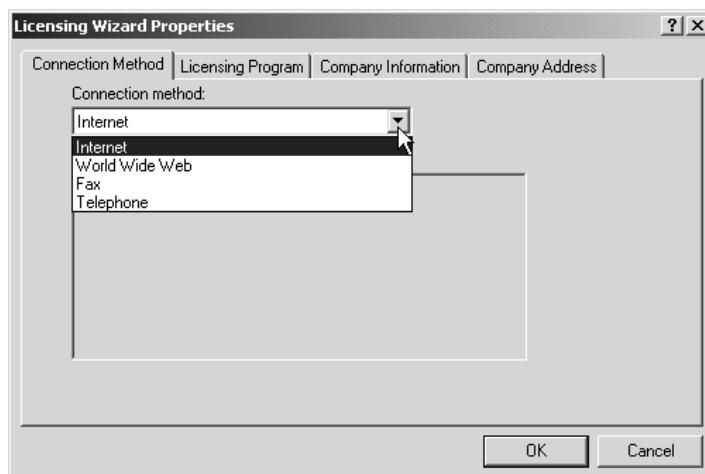


Figure A.27 Server licensing properties.



Next you must configure routers and firewalls. Configuration may not be necessary, however, unless the existing configuration would block the passage of Terminal Services traffic. You should ensure that the Remote Desktop Protocol (RDP) port is not blocked on any routers and firewalls that are placed between the Terminal Servers, the Terminal Services License Server and the Terminal Services clients. RDP uses TCP port 3389. In addition, you must ensure that the IP addresses of your servers and clients are not blocked on any routers or firewalls either. If you have an application layer firewall, you should make certain that there is a filter defined for RDP.

Then, install the Terminal Service on the Windows 2000 Servers that will provide remote administration or application services. This can be executed during the server's installation, or afterward using the Control Panel. To install Terminal Services:

1. In the Control Panel, open the Add/Remove Programs icon.
2. Select Add/Remove Windows Components.
3. Check the box for Terminal Services.
4. Click Next.
5. Select the mode—Remote administration or Application server.
6. Click Next.
7. Click Finish.

Connections are configured in the Terminal Services Configuration console found in the Administrative Tools menu. You must add connections for each simultaneous user. To add a connection, select the Action menu and then the Create New Connection option. The Terminal Services Connection Wizard will start. Select the following during the wizard:

Connection type RDP 5.0.

Encryption level Medium is default.

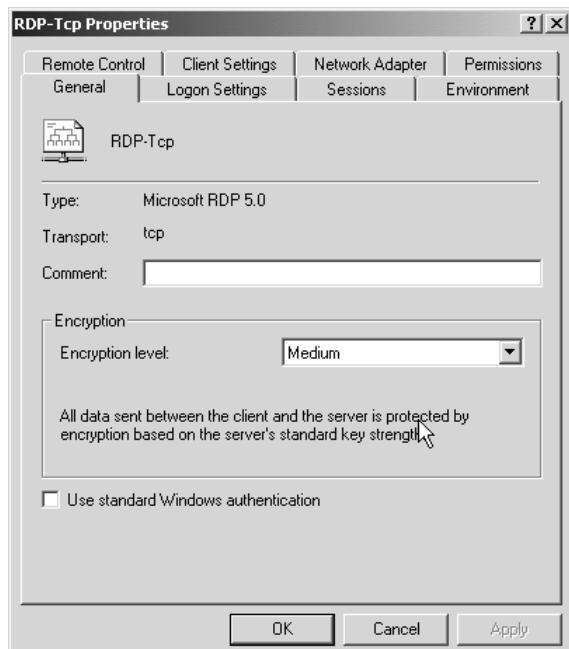
Remote control settings for shadowing user actions on this connection The default is to depend on each user's settings for shadowing the connection.

Transport type Type the connection name and select the Transport type for TCP.

Network adapter Select the adapter that users can use to access this connection and how many connections can be established over that adapter. If you have a server that is connected to the Internet as well as an internal network, you may wish only internal users to access the server. In this case, select only the adapter connected to the internal network.

You can change a connection's properties after initial creation by right-clicking the connection and selecting Properties. The Properties dialog is illustrated in Figure A.28.

Figure A.28 Connection properties.

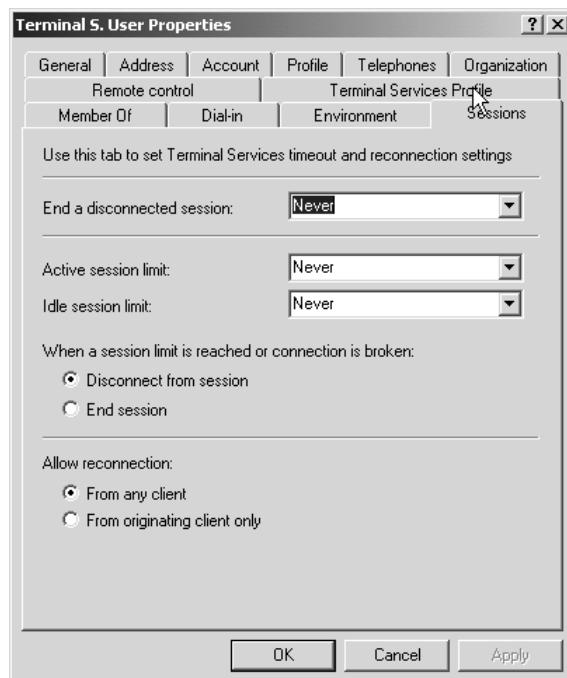


User Security is configured through the Active Directory Users and Computers console for domain-participating Terminal Servers. To change a user's Terminal Services properties, right-click the user account and select Properties. The three tabs that directly affect how a user's terminal connection works are Sessions, Remote Control, and Terminal Services Profile.

- Sessions, shown in Figure A.29, allows you to manage how a connection will work for the user. This includes whether to disconnect or end a session. Disconnected sessions can be connected later, so an application can be left open at a certain point even if there is an error in transmission between the client and the server. An ended session, on the other hand, goes away completely.

- Remote Control allows you to configure whether the user's session can be shadowed by another user. For example, if you configured Terminal Services for a classroom, you would enable remote control without user's permission but with interaction for all students, but disable remote control for all teachers. This would enable a teacher to look at what a student was doing remotely, and then demonstrate how to execute some function within the application.
- Terminal Services Profile allows you to configure a different profile for terminal connections than would be used on a standard Windows desktop. This is particularly helpful when you provide a standard desktop environment through application services, but you want to enable each user to maintain a different profile on their own computer.

Figure A.29 Configuring user sessions.

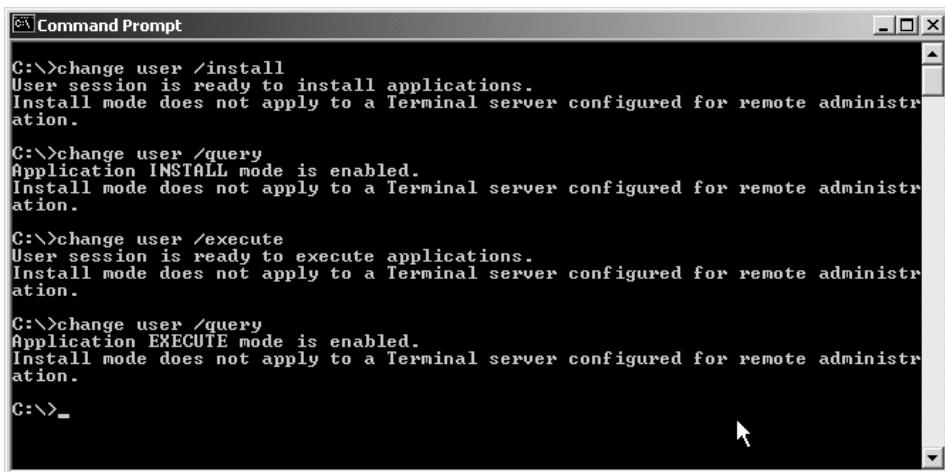


Installing applications on the server requires the server's mode to be changed. In a case such as Office 2000, there may be a special scripted installation specifically meant for Terminal Servers. Applications are installed differently on a Terminal Services server than they are on a stan-

dard server in order to place user files in multiple user locations rather than a single multiple-access directory. In this way, users can have separate preferences for their applications. To install an application:

1. Open a command prompt by clicking Start | Run, typing **cmd**, and then pressing Enter.
2. Change to the directory from which you will install.
3. Type **change user/install** and press Enter.
4. Install the application.
5. When the installation is complete, type **change user/execute** and press Enter to change back to the standard mode. If the application requires the server to reboot, or at any time, you can check the mode that the server is in at reboot by typing **change user/query** at a command prompt. The **change user** command is illustrated in Figure A.30.

Figure A.30 Change user command for application installation.



The screenshot shows a Windows Command Prompt window titled "Command Prompt". The window contains the following text output from the command line:

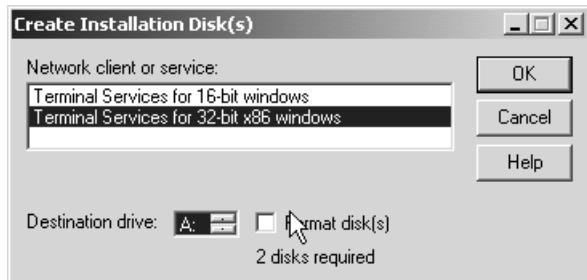
```
C:\>change user /install
User session is ready to install applications.
Install mode does not apply to a Terminal server configured for remote administration.

C:\>change user /query
Application INSTALL mode is enabled.
Install mode does not apply to a Terminal server configured for remote administration.

C:\>change user /execute
User session is ready to execute applications.
Install mode does not apply to a Terminal server configured for remote administration.

C:\>_
```

Creating Terminal Services clients is the final step in the Terminal Services deployment. There is a utility in the Administrative Tools menu of each Terminal Services server called Terminal Services Client Creator. This utility will create diskettes for deploying the Terminal Server client to either 16-bit Windows workstations, or to 32-bit Windows workstations, as shown in Figure A.31. You can use the Setup executable on the diskette to install the client on workstations so that they can access the Terminal Services server.

Figure A.31 Terminal Services Client Creator.

Configuring Routing and Remote Access Services

Routing and remote access is configured through the Routing and Remote Access console available in the Administrative Tools menu. You must configure routing and remote access when you use a server to provide routing between network segments, to provide remote access services to dial-up users, or to provide virtual private network (VPN) services to Internet users. To configure a server:

1. Start the Routing and Remote Access Server (RRAS) console on the Windows 2000 Server.
2. Right-click on the server in the left-hand pane.
3. Select Configure and Enable Routing and Remote Access from the pop-up menu.
4. The RRAS Setup Wizard will start. Click Next.
5. Select the type of services that the server will provide. To provide custom settings, select the Manually configured server option. Otherwise, select the settings that match the role for your new server.
6. Depending on which option you select, the wizard will walk you through the requirements for that option. For example, if you select Remote access server, the next screen allows you to select the remote access protocols, shows how to assign IP addresses (as shown in Figure A.32), and asks whether you will use Remote Authentication Dial-In User Service (RADIUS) for central remote access authentication.
7. After you make your selections, click Finish. The service will start and the RRAS console will show configurable options below your new RRAS server.

Figure A.32 Remote Access Services IP address assignment.



DHCP

You can configure Dynamic Host Configuration Protocol (DHCP) scopes on any Windows 2000 Server through the DHCP console in the Administrative Tools menu. This console is shown in Figure A.33. There are two steps to this process:

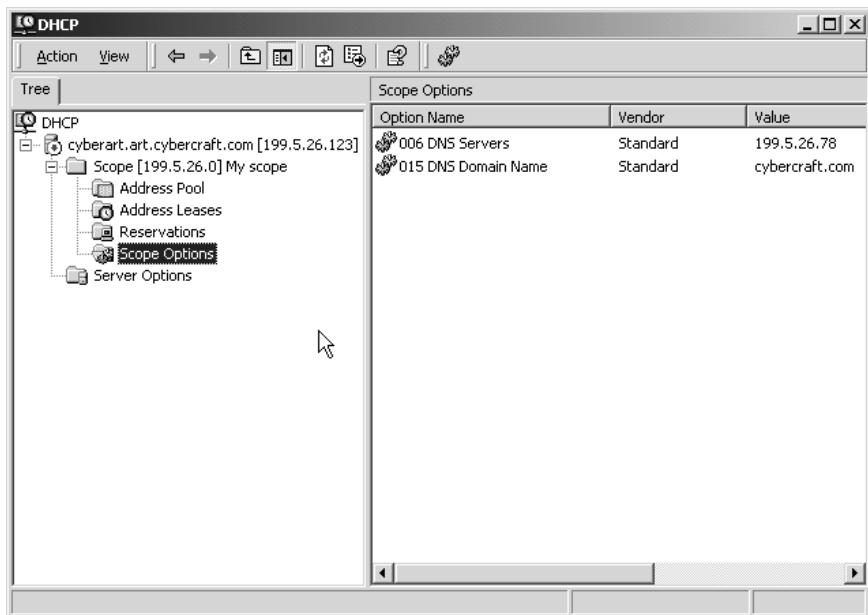
1. Create a DHCP scope of IP addresses to be assigned to computers requesting a dynamic address.
2. Authorize the DHCP server as a security precaution to ensure that it can run on the Windows 2000 network.

To create a DHCP scope:

1. In the DHCP console, right-click the server.
2. Select New Scope.
3. The New Scope Wizard will start. Click Next.
4. Type a name and description for the scope and click Next.
5. Type the IP address range for this scope and the subnet mask. Click Next.
6. If you have statically assigned IP addresses that should be excluded from the scope, add them in at the next wizard dialog. Click Next.

7. Specify the duration for the DHCP lease. Click Next.
8. Select the option to configure the DHCP options and click Next. Options are the additional address information that is passed on to DHCP clients, such as the default gateway that enables the clients to access other IP subnets.
9. Type the address of the Default gateway and click Next.
10. Type the DNS name of the domain to which these DHCP clients will belong, and then provide the DNS server names and IP addresses to contact, and in which order to contact them. Click Next.
11. Type the names and IP addresses of all the WINS servers on the network, if any. Click Next.
12. Select Yes to activate the scope. Click Next.
13. Click Finish to complete the DHCP wizard.

Figure A.33 DHCP console.



To authorize the new DHCP server:

1. Click the Action menu in the DHCP console.
2. Select Authorize.

WINS

Windows Internet Naming Service (WINS) is a leftover from Windows NT. If you have member servers or clients that require WINS, you will want to maintain at least two WINS servers on your network. To configure WINS, simply start the WINS console and add a server using the Action menu. Then configure replication partners for each WINS server.

Case Studies

Both ABC Chemical and West Coast Accounting need to install different types of servers throughout their enterprises. We're going to walk through the installation and configuration of a selected server for ABC Chemical Company first and then follow with West Coast Accounting's installation.

ABC Chemical Company

In the ABC Chemical Company, there are three sites—one for the main campus and one for each warehouse. We will walk through the installation for the domain controller located in the WestSite, which is a Windows 2000 server. This will be a secondary DNS server, as well as a DC for the ABCChem.com that serves as a Global Catalog server and as the Schema FSMO. This will be an upgraded server from Windows NT 4.0.

Before the server is installed, the site structure should be created. This will be done from the first DC installed using the Active Directory Sites and Services console. The first DC will be installed into the Default-First-Site-Name. The three sites that need to be created are HQ, EastWarehouse, and WestWarehouse.

1. Instead of creating a new site for HQ, rename Default-First-Site-Name. Just right-click on the Default-First-Site-Name, select Rename from the pop-up menu, and type the new name **HQ**.
2. Create the two other sites by right-clicking on the sites' containers and selecting New Site from the pop-up menu. Type the EastWarehouse site name and select the DefaultIPSiteLink to create the site. Repeat this to create the WestWarehouse site.
3. Create two site links, WestWarehouse-HQ and EastWarehouse-HQ, and one site link bridge, East-WestBridge. Because the sites each belong to the same domain, they require IP site links. To generate the site links, right-click on the IP container below Inter-Site Transports and select New Site Link. Type the name of the site link—**EastWarehouse-HQ**—and select the two sites, EastWarehouse and HQ, to participate in the link.

4. Double-click the site link to display its properties, change the Cost to 5, the frequency to 60 minutes, and set the schedule so that the link is not available between 10 AM to 2 PM Monday through Friday.
5. To create a site link bridge, right-click on the IP container and select New Site Link Bridge.
6. Type the name for the bridge, **East-WestBridge**, and select the two site links to participate in the bridge.
7. To add the correct IP subnets to each site, right-click the Subnets container and select New Subnet from the pop-up menu.
8. Type the address and subnet mask for a subnet in the EastWarehouse site, select the EastWarehouse site from the Site Name box, and click OK. Repeat this for each IP subnet in each site.

Now it's time to upgrade your NT server. Begin by placing the CD-ROM into the computer and executing the command **D:\I386\WINNT32 /CHECKUPGRADEONLY** (where D: represents the letter of your CD-ROM drive) to determine whether the server can be upgraded. Once this is acceptable, you can run the D:\I386\WINNT32 command. Using the information that you have for the server hardware, you can easily run the upgrade. After the upgrade is complete, the Active Directory must be installed. Since the NT Server was a Windows NT 4 BDC, the DCPROMO application will start automatically. Configure the DC to belong to an existing domain in an existing forest, placing the log files and the database files on separate hard disks. When complete, the server will prompt to be rebooted.

The new DC requires that DNS be configured with a secondary zone. On the server, start the DNS console. Select the Action menu and the option to Configure this server. When the configuration wizard starts, select the creation of a forward lookup zone and then select Secondary for the type and ABCChem.com as the zone name. After the zone is installed, right-click it and select Properties, then click Yes to Allow Dynamic Updates.

The new DC will need to be changed to a Global Catalog server. Open the Active Directory Sites and Services console. Expand the Sites container, the EastWarehouse site, and then the server object within that. Right-click on the NTDS Settings object and select Properties from the pop-up menu. On the General tab, check the box for Global Catalog.

This server must also be designated as the Schema Master FSMO. First, the Schema Manager must be enabled on the server with the REGSRVR command. Then, open the Schema Manager console, right-click

on the root, and select Operations Master. Click the Change button and select the new DC. Select the Schema May be Modified on this server.

West Coast Accounting

In the West Coast Accounting offices, the administrator decides to install Windows 2000 Professional on workstations using a scripted installation method. In this way, the West Coast Accounting administrator can send a few things to a remote office's administrator and ensure that all desktop clients are installed in a consistent manner. These things include:

- The script, or answer file
- An installation batch file
- Source files for Windows 2000 Professional on CD-ROM
- Instructions

The West Coast Accounting administrator can drastically reduce the work involved if the image is identical for each workstation, as well as if the hardware involved is identical. The administrator creates a script that looks like the following:

```
[Data]
Unattendedinstall = Yes
Msdosinitiated = "0"
AutoPartition = 1

[Unattended]
UnattendMode = FullUnattended
OemPreinstall = Yes
TargetPath = Winpro
FileSystem = LeaveAlone
OemSkipEula = Yes

[GuiUnattended]
TimeZone = "004"
AdminPassword = xx3rILacc88
AutoLogon = Yes
AutoLogonCount = 1
OemSkipWelcome = 1
OemSkipRegional = 1
```

```
[UserData]
FullName = "West Coast Accounting"
OrgName = "West Coast Accounting, LLC"
ComputerName = "WCA001"
ProductId = "askjf-sajio-ajkl3-1233j-jakls"

[Display]
BitsPerPel = 8
XResolution = 800
YResolution = 600
VRefresh = 60

[Networking]
InstallDefaultComponents = Yes

[Identification]
JoinDomain = wcacctg.com
DomainAdmin = Administrator
DomainAdminPassword = ask88abc
```

In addition to the script, the administrator creates a batch file that consists of essentially one command to install Windows 2000 from a computer booted with a DOS-formatted diskette. This command is:

```
WINNT /S:d:\i386 /T:c: /U:a:\unattend.txt /E:a:\setupapp.bat
```

Summary

Implementing Windows 2000 involves more than just throwing a CD into a CD-ROM drive and running SETUP. In fact, there is no traditional setup.exe file; instead, there's your choice of winnt.exe and winnt32.exe, which execute either for DOS or 32-bit Windows, respectively.

WINNT (and WINNT32) can be executed with a script to automate an installation. Organizations benefit from automating the installation of an operating system since identical operating systems will have a smaller range of problems with applications than those installed with different

options. Unattend.txt is the default name for a script file. It typically is used in migrating workstations to Windows 2000 Professional, but can also be used for Windows 2000 servers.

Disk duplication is available with two methods—System Preparation (SYSPREP) and Remote Installation Services (RIPREP).

Disk duplication is limited to rolling out Windows 2000 Professional. The difference between SYSPREP and RIPREP is that SYSPREP requires a manual way (usually a boot disk) to access the image on the network, whereas RIPREP can be delivered automatically using a Remote Installation Server to Preboot-Execution-Environment (PXE)-capable network adapters.

There are three phases of the Windows setup process. It begins with the WINNT phase, which begins copying necessary files to the hard drive, and then moves to a Text mode portion. Text mode gathers information about the hardware access layer (HAL), power, and storage, and begins the basic operating system installation. GUI mode occurs next and completes the installation with specific information and optional component installation.

When installing a new Windows 2000 network or upgrading an existing NT network, you need to decide which domain to begin with, and then which server within that domain. The rules are simple:

1. Start with the root domain—if it is a new domain, begin installing its first new domain controller. If it is an upgraded domain, begin by upgrading its PDC.
2. Move onto any child domains of the root domain namespace. If your root domain is root.com, then you would install or migrate sub.root.com before installing or migrating tree.com.
3. Complete the root domain tree until all the root namespace domains are migrated before beginning a new namespace.
4. Migrate each additional namespace within the forest starting with the top of the namespace and installing each subdomain in order.

When you upgrade an NT domain controller, the Active Directory installation wizard begins automatically. However, when you install a new server and wish to make it a domain controller, you must run the Active Directory installation wizard using the DCPROMO.EXE file. If you have an existing DC that you wish to transform into a member or standalone server, then you can also run DCPROMO to demote it.

After the Active Directory is installed in each domain, you create the Organizational Unit hierarchy, and then populate it with users and groups.

These tasks are all completed using the Active Directory Users and Computers console.

Before installing or migrating all domains and servers to Windows 2000, you should establish the sites structure, creating

- Sites
- Connection objects
- Site links
- Site link bridges
- IP subnets

After installing the remaining domains and DCs, you can create bridge-head servers between the sites to manage the traffic traveling across site links.

Just installing and configuring the Active Directory does not complete the installation of a Windows 2000 server. There are other components to install and configure, depending on the role that your server will play in the internetwork, as shown in Table A.5.

Table A.5 Configuration of Various Windows 2000 Server Roles

Server Role	Component	Configuration Method
DNS Server	DNS	Configure the server using the DNS console in the Administrative Tools menu. Configure a forward lookup zone and enable dynamic updates.
File Server	Windows 2000	Right-click a folder in the Windows Explorer and select the Sharing tab. Assign rights and permissions appropriate to the share.
Print Server	Windows 2000	Use the Printers icon in the Control Panel to start the Add Printer wizard. Right-click the printer after creation to change the rights and permissions assigned to it. Select the List in Directory option to publish the printer in the Active Directory.
Dfs root	Dfs	Configure a shared folder to be the root. Then add a root on a DC using the Distributed file system console in the Administrative Tools menu, indicating the shared folder that you created.

Continued

Table A.5 Continued

Server Role	Component	Configuration Method
Certificate Authority	PKI	Create a hierarchy of CA servers on the internet-work. Start by installing the root CA, adding the sub-CA, and finally the issuing CA server. Install CA services using the Add/Remove Program icon in the Control Panel from the Add/Remove Windows Components option. Configure the CA services using the Certificate Authority console to create certificates and issue them. Further configure Group Policies for CA services using the Group Policy tab in the Active Directory properties for OUs, domains, or sites.
Web Server	IIS	IIS is installed by default. Configure the server using the Internet Services Manager in the Administrative Tools menu.
Terminal Services License server	Terminal Services Licensing	Install Terminal Services Licensing using the Add/Remove Programs icon in the Control Panel and selecting the Add/Remove Windows Components option. Configure the licenses using the Terminal Services Licensing console in the Administrative Tools menu.
Terminal Server	Terminal Services	Install Terminal Services using the Add/Remove Programs icon in the Control Panel and selecting the Add/Remove Windows Components option. Configure connections using the Terminal Services Configuration console in the Administrative Tools menu. Create client diskettes using the Terminal Services Client Creator in the Administrative Tools menu and install those on workstations. Configure users individual sessions using the Sessions, Remote Control, and Terminal Services Profile tabs in the Active Directory properties for each user object. Manage active connections using the Terminal Services Manager once Terminal Services are up and running.
Remote Access Server	Routing and Remote Access	Configure remote access using the Routing and Remote Access console in the Administrative Tools menu. Select a Remote Access Server to access the most common remote access needs.

Continued

Table A.5 Continued

Server Role	Component	Configuration Method
VPN Server	Routing and Remote Access	Configure VPN using the Routing and Remote Access console in the Administrative Tools menu. Select VPN Server to access the most common VPN needs during the configuration wizard.
Router	Routing and Remote Access	Configure a Windows 2000 Server to act as a router using the Routing and Remote Access console in the Administrative Tools menu. Select Router to access the most common routing needs during the configuration wizard.
DHCP Server	DHCP	Configure DHCP using the DHCP console in the Administrative Tools menu. Create a scope of IP addresses to be assigned dynamically to clients, identify the IP addresses that should be excluded from the scope, and provide the additional IP addressing information to be delivered to the DHCP clients when they request an IP address, such as default gateway and DNS server information.
WINS Server	WINS	Configure WINS using the WINS console in the Administrative Tools menu. Add a WINS Server and any WINS Replication partners.

FAQs

Q: Can I use disk duplication to copy one server to other servers on my network? I want to make sure that the installation does not veer from the company standards.

A: Disk duplication is not supported by Windows 2000 Server, Advanced Server, or DataCenter Server. You cannot use SYSPREP or RIPREP to deploy the server version of Windows 2000. You can use an unattend.txt file to script the installation of a Windows 2000 Server. This will reduce the time it takes and will manage the installation to reduce operator input errors.

Q: I'm going to have a mixed domain of Windows NT 4 and Windows 2000 DCs for at least a year, and I don't plan to upgrade my Windows 98 or Windows NT 4 clients until two years after that. I am currently not using WINS because we deployed NWLink (IPX compatible) protocols on the network. Do I need to deploy WINS in my network?

A: Since you will need to upgrade all your clients and servers to TCP/IP in order to participate with the Windows 2000 DCs, and since the older versions of Windows depend on NetBIOS naming, you should deploy WINS. WINS will map the new IP addresses to the NetBIOS names of the computers on the network.

Q: I want to install DHCP for a group of workstations, but I want to statically assign the server IP addresses. I may be adding new servers in the future to the same subnet, too. These are all on the same IP subnet. How do I make certain that the DHCP server doesn't give out one of the servers' IP addresses?

A: When you configure the DHCP scope, you can specify which IP addresses are excluded from the scope during the DHCP Configuration wizard. An excluded set of IP addresses will not be handed out to a DHCP client. If you install a new server and you need to reserve an IP address that was previously part of the DHCP scope, you can do so by right-clicking on the Address Pool object under the Scope container and selecting New Exclusion Range from the pop-up menu. Then you can specify the IP address(es) that you want to exclude from the scope.

Index

3Com, 35
3DES. *See* Triple DES
10BaseT Ethernet ports, 14
700 series, 14
800 series, 14–15
900 series, 15
1000 series, 15
1400 series, 15
1600 series, 15
1700 series, 16
 modular routers, 15
2500 series, 16
2600 Series, 16, 39, 130
 modular router, 15
3000 Series product line, 145–147
3000 VPN Concentrators, 16
3600 Series, 16–17, 130
 modular router, 15
7000 Series, 16
7100 Series, 17
7200 Series, 17
7500 Series, 17

A

AAA. *See* Authentication authorization and accounting
aaa authentication enable default (command), 341
aaa authentication login (command), 339–340
aaa authentication ppp (command), 340–341
AAL. *See* Asynchronous Transfer Mode Adaptation Layer
Access control lists (ACLs), 7, 390
Access layer, 486
Access lists, 383, 434, 435

creation, 439
Access network services, utilization, 67
Access server, 71
ACCM. *See* Asynchronous Control Character Map
Accounting, 332. *See also* Authentication authorization and accounting
ACK, 79
ACLs. *See* Access control lists
ACPI. *See* Advanced Configuration and Power Interface
Active Directory
 domain, 508
 executed query, 513
 installation, 497–519
Active Directory Migration Tool (ADMT), 508
Address blocks, 465–466
Address flag, 239
Address overloading, 410, 421–430, 443
 configuration, 423–424
 screen captures, 424–425
Addresses, 78. *See also* Private network addresses
 assignment. *See* Internetwork conservation, strategies, 458–460
 number, 466
 personal selection, 463–465
 renumbering, 467
 space, contrast. *See* Public address space
Addressing, 249. *See also* Private addressing
 economics, 460–465
 hierarchies, 468
Administrative LANs, 471

- ADMT. *See* Active Directory Migration Tool
- ADSL. *See* Asymmetric Digital Subscriber Line
- ADTRAN, 321
- Advanced Configuration and Power Interface (ACPI), 491
- Advanced Integration Module (AIM) slots, 16, 402
- AH. *See* Authentication Header
- AIM. *See* Advanced Integration Module
- Aironet series, 6
- All-digital format, 7
- American National Standards Institute (ANSI), 156, 252
terminal standard, 65
- Analog lines, 13
- Analog modems, connections, 6–7
- Analog ports, 14, 15
- Analog transmissions, 154
- Analog-to-digital conversion process, 32
- ANSI. *See* American National Standards Institute
- Answer file, 488
- AppleTalk, 12, 77, 81, 181, 225, 234, 269, 329
protocol, 396
- AppleTalk Control Protocol (ATCP), 82
- AppleTalk Remote Access (ARA) Protocol (ARAP), 334, 335
advanced remote connectivity, 82
contrast. *See* Point-to-Point Protocol
- Application services, 528
- ARAP. *See* AppleTalk Remote Access Protocol
- ARQ. *See* Automatic repeat request
- AS. *See* Autonomous system
- AS/400, 57
- AS5000 Series, 17
- AS5200, 39
- AS5300, 39, 130
- ASCII
character, 68
text characters, 81
- ASN. *See* Autonomous system number
- Asymmetric Digital Subscriber Line (ADSL), 14–15
modem, 15
- Asynchronous communications, 36
- Asynchronous connection, 77
configuration, 38–56
- Asynchronous connectivity, 30
- Asynchronous Control Character Map (ACCM), 86
- Asynchronous DDR, 55
- Asynchronous dial-in terminal services, providing, 56–73
- Asynchronous dialup, 15
- Asynchronous framing, 43
- Asynchronous interface, 85, 165
- Asynchronous lines, 55, 94
- Asynchronous modem connections, 168
- Asynchronous remote access connection
configuration, 29
FAQs, 74
introduction, 30
- Asynchronous serial, 16
interface, 168
ports, 8
- Asynchronous Transfer Mode Adaptation Layer (AAL), 291
AAL5NLPID, 383
- Asynchronous Transfer Mode Address Resolution Protocol (ATMARP), 527
- Asynchronous Transfer Mode (ATM), 11–13, 17, 234, 257, 527
ATM25 interface, 15
- ATM-based interface, 391
- availability, 237

- configuration, 293–296
 connections, 290–305
 troubleshooting/verification, 297–305
 debug commands, 300
 network, 292, 297
 overview, 290–291
 packet format, 290–291
 virtual circuits, 292–293
AT commands, 50
ATCP. See AppleTalk Control Protocol
ATDT\T command, 56
ATH command, 56
ATM. See Asynchronous Transfer Mode
ATMARP. See Asynchronous Transfer Mode Address Resolution Protocol
AT&T, 30, 241, 242, 248
Attribute-value (AV), 331
 pairs, 354
AUI, 16
Authentication, 12, 77, 331. *See also*
 Caller ID; Challenge Handshake Authentication Protocol;
 Password Authentication Protocol; Point-to-Point Protocol
 failures, 91–96
 methods, 122, 329, 341
 phase, 142
 protocol, 80
Authentication authorization and accounting (AAA), 326
 access control, monitoring/verification, 358–362
 accounting, configuration, 344–346
 authentication, configuration, 339–341
 authorization, configuration, 342–343
 commands, 219
 configuration, 336–346
 usage. *See* Virtual profiles
 debug/show commands, 358–362
 enabling, 336
 implementation, 346
 overview, 328–333
 RADIUS, usage, 336
 servers, 218, 220–222, 329–331
 TACACS+, usage, 336
 templates, 223
 virtual profiles, usage, 346–357
 walkthrough, 362–367
Authentication Header (AH), 115, 134
Authority, delegation. *See*
 Organizational units
Authorization, 331–332. *See also*
 Authentication authorization and accounting
Auto-answer, 45
Autocommand, 66–67
Autodiscovery process, 53
Automatic repeat request (ARQ), 36
Autonomous system (AS), 430, 475, 476. *See also* External AS; Internal AS
Autonomous system number (ASN), 477
Autoselect, 84
AUX port, 33, 34, 39, 49
 cabling, 33–34
AV. *See* Attribute-value
- ## B
- Backbone**, 393. *See also* Collapsed backbone; Corporate backbone infrastructure components, 486
Backout plans, 24
Backup
 connection, 305
 troubleshooting/verification, 317–323
 interface, 305–309
 services, 77
 systems, 329
Backup Domain Controller (BDC), 498, 511, 538

- backup load (command), 308–309
- Backward Explicit Congestion Notification (BECN)**, 250, 271, 272
- bit, 251
- frames, 375
- BACP**. See **Bandwidth Allocation Control Protocol**
- Bandwidth, 380, 389. *See also* Precedence planning, 378
- Bandwidth Allocation Control Protocol (BACP)**, 95
- Basic Rate Interface (BRI)**, 8, 32, 96, 154–156, 210
- BRI-to-BRI configuration, 195
 - BRI-to-PRI connection, 199
 - call setup, 154–155
 - circuits, 155
 - connection, 195, 449
 - functional groups, 155–156
 - functionality, 158
 - interface, 99, 101, 165–167, 213–215, 315, 316, 448
 - support, 97
 - leased line, 15
 - lines, 153
 - ports, 14
 - reference points, 155–156
 - usage, 195
- Baud rate, 239
- B-channel**, 153, 154, 173, 184, 185, 188
- information, 192
 - selection, 218
 - usage, 194
- BCRAN**, 486
- FAQs, 26–27
- introduction, 2
- BDC**. See **Backup Domain Controller**
- BECN**. See **Backward Explicit Congestion Notification**
- Beginning Input Output System (BIOS)**, 500. *See also* NetBIOS
- Best-effort conversations, 376
- BGP**. *See* **Border Gateway Protocol**
- Bi-directional reconstruction dictionary pair, 402
- BIOS**. *See* **Beginning Input Output System**
- B-ISDN**. *See* **Broadband ISDN**
- Bonding**. *See* **Communications links**
- action, 78
- BOOTP**, 412
- bootpc**. *See* **User Datagram Protocol**
- Border Gateway Protocol (BGP)**
- peer node, 477
 - requirements, 475–478
 - routers, 478
- Branch office, 21
- BRI**. *See* **Basic Rate Interface**
- Broadband ISDN (B-ISDN)**, 290
- Brute force attacks, 88
- BSD**, 479
- BSD UNIX**
- environment, 57
 - rlogin program, 60
- Built-in interfaces, 11
- Built-in NT1line, 14
- Business operation, 20

C

- CA**. *See* **Certification authority**
- Cable modems**, 15
- Cabling**. *See* **AUX port; Modems**
- Call setup**, 8. *See also* **Basic Rate Interface; Integrated Services Digital Network; X.25**
- Call teardown**, 8
- Callback**. *See* **EXEC; Microsoft Callback; Point-to-Point Protocol**
- accepting, 92

- number, 72
- verification, 71
- Caller ID
 - authentication, 199, 201
 - screening, 179
- Carrier Detect (CD), 33
- Carrier transitions, 266
- CBAC. *See* Context-based Access Control
- CBWFQ. *See* Class-based Weighted Fair Queuing
- CCITT. *See* International Telegraph and Telephone Consultative Committee
- CD. *See* Carrier Detect
- CDP. *See* Cisco Discovery Protocol
- CEF. *See* Distributed Cisco express forwarding; Internet Protocol
- Cell Loss Priority (CLP), 291
- Central Office (CO), 3, 155
- Central site, 21
- Certification authority (CA), 522
 - servers, 523, 524
- Challenge Handshake Authentication Protocol (CHAP), 12, 76, 80, 87–90, 448. *See also* Microsoft CHAP
 - authentication, 86–90
 - debugging, 103
 - password, 230
- Challenge/response, 328
- Change control procedures, 24
- Channel service units/data service units (CSU/DSU), 3, 5, 8, 15, 27, 31, 157. *See also* Remote CSU/DSU; Switched DSU/CSU connection, 266
- Channelized T3 line, 17
- CHAP. *See* Challenge Handshake Authentication Protocol
- Chat scripts, 55–56, 71
- CIDR. *See* Classless Inter-Domain Routing
- CIPX. *See* Compressed IPX
- CIR. *See* Committed Information Rate
- Circuit level, 263
- Circuit-switched connections, 4, 6–9
- Circuit-switched network, 7
- Cisco 700 Series router, usage. *See* Internet Service Provider
- Cisco 1700, 327
- Cisco 1720, 327
- Cisco 2500, 327
- Cisco 2600, 327
- Cisco 3600, 327
- Cisco 3640, 45, 49
- Cisco 7000 series router, 381
- Cisco 7200, 327
- Cisco 7500, 258
- Cisco access
 - routers, selection, 14–17
 - servers, 2, 64
 - selection, 14–17
 - usage. *See* Point-to-Point Protocol
- Cisco command accounting, 344
- Cisco console, 33–34
- Cisco debug commands, 103
- Cisco Discovery Protocol (CDP) packets, 269
- Cisco equipment, details, 26
- Cisco IOS, 14, 56, 61, 65, 67, 245, 272
 - 11.3(2)T, 91, 93
 - account creation, 330
 - code monitors, 256
 - commands, 358
 - compression method, selection, 402–403
 - features, 328
 - firewall feature set, 327, 368
 - queuing method, selection, 392–394
 - support, 397, 398
 - versions, 446

- Cisco remote access solutions
FAQs, 26–27
introduction, 2
- Cisco routers, 33, 58, 62, 83, 104, 293, 384
PPP/ISDN connections, 99–103
support, 5
- Cisco VPN terminology, 117–119
- CiscoSecure, 330–331
ACS. See UNIX; Windows NT
Global Roaming Server. See UNIX
- CiscoSecure ACS. See Windows NT
- CiscoSecure Policy Manager, 147, 148
- Class A address, 458, 463
- Class A networks, 458, 467–468, 475
- Class A private network, subnetting strategy, 468–475
- Class A subnetting tables, 470
- Class B networks, 458, 468, 475
- Class C address, 464
- Class C networks, 458, 466, 475
- Class characteristics, 391
- Class-based Weighted Fair Queuing (CBWFQ), 372, 373, 390–392
- Classless Inter-Domain Routing (CIDR), 458, 459, 465, 475
- clear crypto sa (command), 126
- Clear To Send (CTS), 43, 44, 263
usage, 45
- Clients, configuration. See Virtual Private Network
- Client-server communications, 333
- Client-server protocol, 333
- Client-to-client, 76
- CLNS. See Connectionless Network Service
- CLP. See Cell Loss Priority
- CO. See Central Office
- Collapsed backbone, 460
- Command/response (C/R), 250
- Committed Information Rate (CIR), 251–254, 271
- Communication lines, multiple protocols, 77
- Communications links, bonding, 78
- Competitive bids, 21
- Compressed IPX (CIPX), 82
- Compression. See Hardware; Header compression; Point-to-Point Protocol
effects, 94
method, selection. See Cisco IOS
operation, verification, 403
usage, 12. See also Network performance optimization
- Compression Service Adapter (CSA), 402
- Computer Migration Wizard, 508
- Concentrators. See 3000 VPN Concentrators
- Confidentiality, 115
- Congestion notification, 249
- Connectionless Network Service (CLNS), 12
- Connections. See Analog modems;
Circuit-switched connections;
Dedicated connections; Integrated Services Digital Network; Packet-switched connections; Point-to-point connection
backup. See Permanent connections
configuration. See Asynchronous connection; Permanent connections
object management, 518
requirements. See Wide Area Network troubleshooting. See Windows connections
type, 530
types, 4
- Consistency checking, 148
- Context-based Access Control (CBAC), 327

Continuous-mode compression algorithms, 401
Control byte, 239
Corporate backbone, 474
Corporate IT department, 104
CPE. See Customer Premise Equipment
C-plane, 158, 159
CPU utilization, 402
CQ. See Custom Queuing
C/R. See Command/response
CRC. See Cyclic redundancy check
Crypto Map, 118, 123, 124. *See also* Dynamic Crypto Map
policy number, 125
usage, 128
Crypto map entries, 124
CSA. See Compression Service Adapter
CSU/DSU. See Channel service units/data service units
CTS. See Clear To Send
Cursor style, 239
Custom Queuing (CQ), 272, 372, 373, 387–390
examples, 388–390
Customer Premise Equipment (CPE), 3, 259
Customers, benefit, 21
Cyclic redundancy check (CRC), 248
counters, 266

D

Data circuit-terminating equipment (DCE), 5, 238
Data Communications Equipment (DCE), 31, 32
Data compression
mechanism, 397–398
methods, 35–38
overview, 397–401
protocols, 37–38

Data Encryption Standard (DES), 116, 118, 120. *See also* Triple DES; Triple pass DES
Data Link Connection Identifier (DLCI), 250, 258
address, 252
broadcast, 267
priority levels, 271
Data Network Identification Code (DNIC), 239–241
Data Set Ready (DSR), 32, 46
usage, 47
Data Terminal Equipment (DTE), 31, 32, 238
Data Terminal Ready (DTR), 32–33, 305
dialing, 168
signal, 45
Data-link layer, 237, 263
protocol, 8, 77
Data-Link Switching (DLSw), 385
DC. *See* Distribution centers; Domain controller
DCE. *See* Data circuit-terminating equipment; Data Communications Equipment
dCEF. *See* Distributed Cisco express forwarding
D-channel, 153–155, 158, 192
selection, 218
DCPromo / DCPROMO, 500–503, 541
DDR. *See* Dial-on demand routing
DE. *See* Discard Eligibility
debug aaa accounting (command), 360
debug aaa authentication (command), 358
debug aaa authorization (command), 358–359
debug atm ilmi (command), 303–305
debug atm packet (command), 300–302

debug atm state (command), 302–303
debug (commands), 358
debug dialer (command), 193
debug frame relay lmi (command),
 267–269
debug frame-relay (command), 269
debug frame-relay lmi (command),
 270–271, 280
debug frame-relay packet (command),
 269–270
debug modem (command), 99
debug ppp authentication (command),
 99, 358
debug PPP error (command), 99
debug ppp multilink (command), 194
debug PPP negotiation (command), 99
debug radius (command), 360–361
debug snapshot (command), 194
debug tacacs (command), 360
debug virtual-template (command),
 360
Debug x25 (command), 247
DEC. See Digital Equipment
 Corporation
Dedicated connections, 4–6
Default queuing, 272
Default routes, 180, 309–315
Default stopbits, 43
Demarcation (Demarc) point, 3
Demilitarized zone (DMZ), 126
De-multiplexing, 239
DES. See Data Encryption Standard
Destination IP address, 415, 421, 422,
 438
Dfs. See Distributed file system
DHCP. See Dynamic Host
 Configuration Protocol
Dial backup, 160
Dialer, 349. *See also* War dialer
 addressing, 165, 166
Backup, 315
interfaces, 165–168
list, 83, 84
mapping, 166–167
monitoring, 186–187
profiles, 160, 166, 214, 346, 347
rotary groups, 165, 166
Watch, 315–317, 324
dialer dtr (command), 168
dialer idle-timeout (command), 169,
 175
dialer in-band (command), 168
dialer-list 1 protocol ip permit, 161
Dial-in access, 77
Dial-in clients, 85
Dial-in lines, 156
Dial-in modem, 66
Dial-in service, 55, 56
Dial-in terminal services, providing.
 See Asynchronous dial-in termi-
 nal services
Dial-on-demand routing (DDR), 7, 33,
 159–168, 346. *See also*
 Asynchronous DDR
commands, 176–179
configuration, 168–179. *See also*
 Legacy DDR
connections, 186, 198
enabling, 209
FAQs, 205–207, 232
interface, 190
introduction, 152, 210
link, 180
monitoring, 182, 190–195
networks. *See* Large-scale DDR net-
 works
routing issues, 179–182
services, 77
sites, 164
troubleshooting, 182, 190–195
usage, 151

- walkthrough, 195–203
- Dial-out service, 56
- Dial-up connection, 2. *See also* Upstream traffic
- Dial-up network, 217
- Dial-Up Networking (DUN), 95, 104, 107
 - software, 105
- Dial-up PSTN, 326
- Dictionary compression, 398
- Dictionary pair. *See* Bi-directional reconstruction dictionary pair
- Diffie-Hellman, 137
- Digital certificates, 522
- Digital communications, 30
- Digital Equipment Corporation (DEC), 369
 - DECNET, 225, 269
 - DECnet, 12, 77, 398
 - LAT, 57
 - VMS hosts, 60–62
- Digital modems, 32
- Digital Subscriber Line (DSL). *See* Asymmetric Digital Subscriber Line; Integrated Services Digital Network Digital Subscriber Line
- Discard Eligibility (DE)**, 250
 - bit, 251, 271
- Disconnects (DISCs), 246
- Disk duplication methods, overview, 491–495
- Distance Vector protocols, 393
- Distinct reservation style, 377
- Distributed Cisco express forwarding (dCEF), 382
- Distributed file system (Dfs), configuration, 521–522
- Distribution centers (DC)**, 472, 473
 - LANs, 472
- DLCI**. *See* Data Link Connection Identifier
- DLSw**. *See* Data-Link Switching
- DMZ**. *See* Demilitarized zone
- DNIC**. *See* Data Network Identification Code
- DNS**. *See* Domain Name System
- Documentation**, 468, 474
- Domain controller (DC)**, 499, 500, 512, 516
- Domain local groups**, 511
- Domain Name System (DNS)**, 412, 494
 - client, 501
 - configuration, 519–520
 - Manager, 519
 - name, 57
 - query, 433, 443
 - reply, 433, 434
 - responses, 443
 - server, 431–433, 437–438, 446–447, 451, 501
 - Service, 85, 501
- Domains**
 - installation, 498–499
 - populating, 504–516
- DSL**. *See* Digital Subscriber Line
- DSR**. *See* Data Set Ready
- DSU**. *See* Channel service units/data service units
- DTE**. *See* Data Terminal Equipment
- DTR**. *See* Data Terminal Ready
- Dual address translation, 430–436
- Dual-hub-and-spoke network, 165
- DUN**. *See* Dial-Up Networking
- DWFQ**. *See* VIP DWFQ
- Dynamic Crypto Map**, 118, 138
- Dynamic Host Configuration Protocol (DHCP)**, 81, 85, 108, 467, 535–536. *See also* Third-party DHCP
 - discovery process, 493
 - information, 388
 - server, 446, 494

- options, 445
 - services, 109
 - traffic, 388
 - Dynamic NAT, 415
 - configuration, 416–418
 - translation, 416, 429
 - screen captures, 418–420
 - Dynamic queues, 376
 - Dynamic routing protocols, 162
 - Dynamic translations, 414–416, 443
- E**
- E series, 153
 - E1 connections, 145
 - E1 line, 154. *See also* Multichannel E1 line
 - E3 line, 4, 10
 - EA. *See* Extended Address
 - Easy IP
 - operation, 446–449
 - usage. *See* Internet Service Provider
 - EBGP. *See* Exterior Border Gateway Protocol
 - EDE. *See* Encrypt, Decrypt, Encrypt
 - EGP. *See* Exterior Gateway Protocol
 - EIA/TIA-232, 5, 31
 - port, 32
 - EIA/TIA-449, 5, 31
 - EIA/TIA-530, 5
 - EIGRP, 198
 - route, 314, 315
 - Enable-timeout, 225
 - Encapsulating Security Payload (ESP), 115, 118, 134
 - Encapsulation, 167, 236. *See also* Layer 2 encapsulation; Multiple protocols
 - protocols, 184. *See also* Wide Area Network
 - type, 212
 - Encrypt, Decrypt, Encrypt (EDE), 116
 - Encryption, 137, 328
 - key, 337
 - level, 530
 - End-to-end connectivity, 248
 - End-user applications, 373
 - End-user license agreement (EULA), 496
 - Equipment
 - impact, 19
 - usage. *See* Sites
 - Error checking, 36, 249
 - Error control, 35–38
 - Error detection, 7, 78
 - E-series, 34
 - ESP. *See* Encapsulating Security Payload
 - Ethernet, 15, 17. *See also* Fast Ethernet
 - connections, 15
 - Hub, 16
 - interfaces. *See* Gigabit Ethernet interfaces
 - network, 203
 - ports, 15. *See also* 10BaseT Ethernet ports
 - EULA. *See* End-user license agreement
 - EXEC, 342, 344
 - callback, 69–73
 - command, 67
 - mode command, 66, 342, 343
 - prompt, 57
 - session, 70, 343
 - terminal commands, 344
 - terminal session, 342
 - Exit configuration mode, 43
 - Extended Address (EA), 250
 - Exterior Border Gateway Protocol (EBGP), 309, 478
 - requirements, 479–482
 - Exterior Gateway Protocol (EGP), 475
 - External AS, 481
 - External consultants, 25

Extranets, 148
solution, 126–130

F

FAIL message, 332
FAIL result, 360
Fast Ethernet, 17, 226
Fast switching mode, 322
Fast-idle, 225
FCC. *See* Federal Communications Commission
FCS. *See* Frame Check Sequence
FDDI. *See* Fiber Distributed Data Interface
Feature-rich menu system, 69
FECN. *See* Forward Explicit Congestion Notification
Federal Communications Commission (FCC), 6
FF. *See* Fixed-filter
FIB. *See* Forwarding information base
Fiber Distributed Data Interface (FDDI), 17
FIFO. *See* First In First Out
FIFQ, 372
File Transfer Protocol (FTP), 375, 413, 436, 449
server, 447, 450, 452
FIN. *See* Transmission Control Protocol
Firewalls, 119, 148
definition, 326
feature set. *See* Cisco IOS
benefits/features, 327–328
proxy, 334
First In First Out (FIFO), 272, 372, 374–375, 391
Fixed-configuration routers, 15
Fixed-filter (FF) style, 378
Fixed-LAN interfaces, 16
Fixed-length cells, 13
Flags, 78, 250
Flash memory cards, 15
Flexible Single Master of Operations (FSMO), 537
role, 499
Floating static routes, 309–315
Flow control, 239
Flow-based WRED, 396–397
flowcontrol hardware (command), 45
Folders, publishing, 514–515
Forward Explicit Congestion Notification (FECN), 250, 271, 272
bit, 251
receipt, 375
Forward Lookup Zone, 520
Forwarding information base (FIB), 382
Fractional T1 line, 4, 15
Fragments, 383
Frame Check Sequence (FCS), 77, 78, 251
field, 239
Frame Relay, 2, 10–12, 114, 234, 257
circuit, 254
cloud, 461
configuration, 259–263
ISDN backup, usage, 310–315
connections, 215, 248–288
connectivity, 266
encapsulation types, 399
FRF, 400
header, 250, 251
interface, 269
network, usage, 461
overview, 248–252
problems, 267
spreadsheet, creation, 255
technologies, 237
topologies, 253–258
traffic shaping, 273
troubleshooting, 263–266
verification, 263–266

Frame Relay Traffic Shaping (FRTS),
251–272
enabling, 272

Frames. *See X.25*

Framing, 202

FRF. *See Frame Relay*

FRMRs. *See Protocol frame errors*

FRTS. *See Frame Relay Traffic Shaping*

FSMO. *See Flexible Single Master of Operations*

FTP. *See File Transfer Protocol*

Fully meshed infrastructure, 254

Fully meshed network, 253

Fully meshed topology, 162–163

Functional groups. *See Basic Rate Interface; Primary Rate Interface*

FW feature set. *See Routers*

FW Plus IPSec, 147

FW solution, 145

G

Gateway of the last resort, 180

Generic Flow (GFC), 291

GFC. *See Generic Flow*

Gigabit Ethernet interfaces, 17

Global address, 410. *See also Inside global address; Outside global address*

Global Catalog server, 499

Global command, 342

Global configuration mode, 83–84, 92–93, 336, 443

Global groups, 511

Global IP addresses, 414, 425, 435, 440

Global Roaming Server (GRS). *See UNIX*

Global synchronization problem, 396

Globally Unique IDentifier (GUID), 493

Graphical interface, 148

Graphical User Interface (GUI), 495, 508, 541
mode, 496

Group Policy Editor, 515

Groups. *See Domain local groups; Global groups; Local groups; Universal groups*

codes, 61

creation, 511–512

policy, application, 515–516

GRS. *See UNIX*

G-series, 34

GUI. *See Graphical User Interface*

GUID. *See Globally Unique IDentifier*

H

HAL. *See Hardware Access Layer*

Hardware. *See Servers*

compression, 401–402

Hardware Access Layer (HAL), 491, 541

Hash algorithm, 115

usage, 121, 122

H-channel, 153, 154

HDLC. *See High-level Data Link Control*

Header compression, 78, 398–399

Header Error Control (HEC), 291

Header flag, 239

Headquarters

- LAN, 460, 471
- subnet, 462
- WAN links, 472

HEC. *See Header Error Control*

Heuristic hacking attempts, 87

Hidden objects, 505–506

High Speed Transfer (HST), 35

High-level Data Link Control (HDLC), 2, 11–12, 86, 167, 234–236

- encapsulation, 168
- types, 399
- High-speed links, 6
- High-Speed Serial Interface (HSSI), 17, 31
- High-speed technologies, 6
- Hold-queue, 225
- Hop address, 84
- HSSI. *See* High-Speed Serial Interface
- HST. *See* High Speed Transfer
- HTML application, 525
- HTTP. *See* HyperText Transfer Protocol
- Hub-and-spoke configuration, 259
- Hub-and-spoke network design topology, 375
- Hub-and-spoke topology, 164–165, 205
- HW accelerator, 145
- HyperText Transfer Protocol (HTTP), 412, 436, 449
 - Web server, 450
- I**
- IANA, 462, 463
- IBGP. *See* Interior Border Gateway Protocol
- IBM
 - 3270 clients, 57
 - 3278 terminal, 65
 - hosts, 65
 - TN3270, 344
 - emulation services, 65
 - services, 64
- ICANN. *See* Internet Corporation for Assigned Names and Numbers
- ICMP, 412
 - packet, 419
 - traffic, 224, 230
- Idle timer, 224
- Idle-time, 159
- Idle-timeout, 225, 226
- IDN. *See* International Data Number
- IDS. *See* Integrated Services Digital Network
- Digital Subscriber Line (IDSL)
- IF. *See* Information Frame
- IGRP. *See* Interior Gateway Routing Protocol
- IIS. *See* Internet Information Services
- IKE. *See* Internet Key Exchange
- Incoming interface, 383
- Information field, 239
- Information Frame (IF), 239
- Information Technology (IT)
 - budget, 6
 - department. *See* Corporate IT department
 - professional advice, 88, 115, 378, 430, 444, 461, 497, 518
- Initialization strings, 50
- Input errors, 266
- Inside components, 410
- Inside global address, 410, 416, 418, 425
- Inside global IP address, 420
- Inside interface, 428
- Inside local address, 410, 425
- Inside source addresses, translation, 414–421
- Integrated NT1 line, 15
- Integrated Services Digital Network
 - Digital Subscriber Line (IDSL), 14, 76
- Integrated Services Digital Network (ISDN), 2, 346, 347
 - backup
 - connections, 317
 - usage. *See* Frame Relay
 - call setup/teardown, 159
 - circuits, 32, 254
 - commands, 176–179
 - configuration, 168–179
 - connections, 7–9, 175, 230, 234, 326. *See also* Cisco routers
 - device. *See* Non-ISDN device

- dial backup, 254
- dial-up connections, 226
- FAQs, 205–207
- interfaces, 83, 167–168, 190, 316
 - monitoring, 182–186
 - support, 167–168
- introduction, 152
- lines, 91, 173, 203, 218
- link, 199
- monitoring, 182, 190–195
- network, 158
- overview, 152–157
- protocol layers, 157–159
- suite, 248
- support, 14
- switch, 100, 101, 155
 - type, 185
- terminal adapter, 8
- troubleshooting, 182, 190–195
- usage, 151
- walkthrough, 195–203
- Interesting traffic, 7, 159, 161–162, 213
- Interfaces.** See *Built-in interfaces*; *Incoming interface*; *Inside Interface*; *Loopback*; *Outside interface*; *Serial interfaces*; *Synchronous serial interfaces*
- closing, virtual template, 97
- configuration mode, 83, 92, 93, 104
- statistics, 264
- support. See *Integrated Services Digital Network*
- Interior Border Gateway Protocol (IBGP),** 478
 - requirements, 479–482
- Interior Gateway Routing Protocol (IGRP),** 180
- Internal AS,** 481
- International Data Number (IDN),** 241
- International Telecommunications Union-Telecommunication Standardization Sector (ITU-T),** 34, 35, 37, 152, 153, 239
- International Telegraph and Telephone Consultative Committee (CCITT),** 34, 152, 156
- Internet,** 148
 - connection, 21
 - dedicated connections, 5
 - standard, 76
- Internet Corporation for Assigned Names and Numbers (ICANN),** 463
- Internet Information Services (IIS),** 525–526
- Internet Key Exchange (IKE),** 114–115, 118–120
 - configuration, 120–123
 - IKE mode config, 136
 - operation, 123
- Internet Packet Exchange Control Protocol (IPXCP),** 77, 81
- Internet Protocol Control Protocol (IPCP),** 77, 81, 445, 448
 - setup. See *Wide Area Network*
- Internet Protocol (IP),** 77, 81, 234, 269, 463
 - address space, 430
 - CEF, 382
 - header, 378
 - compression, 403
 - Multicast, 412
 - packets, 373
 - precedence, 378, 379
 - usage. See *Weighted Fair Queuing*
 - subnets, 516
 - ranges, 427
 - traffic, 224, 378, 385
- Internet Protocol (IP) addresses,** 57–59, 66, 84, 85, 108, 137, 245, 361.
 - See also *Destination IP address*; *Global IP addresses*; *Local IP address*; *Real IP address*; *Source IP address*; *Static IP addresses*; *Virtual IP address*

- configuration, 183
- resolution, 431
- timeouts, 411
- translation, 409
- usage, 221, 415, 419
- Internet Protocol Security (IPSec),** 3, 115, 327
 - client, 16
 - configuration, 123–126. *See also* Network access server
 - connections, 127
 - policy, 134
 - protocol, 127
 - security association (SA), 116, 118, 140
- Internet Protocol-Routing Information Protocol (IP-RIP),** 180
- Internet Security Association and Key Management Protocol (ISAKMP),** 114–115, 118
 - configuration, 120–123, 136–137
 - policy, 133
 - security association (SA), 117
- Internet Service Provider (ISP),** 76, 96, 104, 135, 331, 408, 475
- Cisco 700 Series router,** usage, 449–450
 - Easy IP, usage, 445–449
 - NAT usage, 444–445
 - PAT usage, 449–450
- Internetwork,** 469–475
 - address assignment, 471–473
 - results, 474–475
 - strategy, 470–471
- Internetwork Packet Exchange (IPX),** 12, 76, 77, 81, 234, 329, 545.
 - See also* Compressed IPX
 - network number, 84
 - protocol, 84
- Internetwork Packet Exchange-Routing Information Protocol (IPX-RIP),** 181
- Internetwork Packet Exchange/Sequenced Packet Exchange (IPX/SPX),** 104, 396
- Inter-VLAN routing,** 16, 17
- Intranets,** 148, 410, 421, 437
 - solution, 119–120
- Investment, protection,** 328
- IOS,** 52–54, 69, 166, 327. *See also* Cisco IOS
 - commands, 219
- IPCP.** *See* Internet Protocol Control Protocol
- IP-RIP.** *See* Internet Protocol-Routing Information Protocol
- IPSec.** *See* Internet Protocol Security
- IPX.** *See* Internetwork Packet Exchange
- IPXCP.** *See* Internet Packet Exchange Control Protocol
- IPX-RIP.** *See* Internetwork Packet Exchange-Routing Information Protocol
- IPX/SPX.** *See* Internetwork Packet Exchange/Sequenced Packet Exchange
- IS department,** 23
- IS staff,** 17
- ISAKMP.** *See* Internet Security Association and Key Management Protocol
- ISDN.** *See* Integrated Services Digital Network
- isdn caller (command),** 179
- I-series,** 34
- ISO,** 12
- ISP.** *See* Internet Service Provider
- IT.** *See* Information Technology
- ITU-T.** *See* International Telecommunications Union-Telecommunication Standardization Sector

J

Joint Photographic Experts Group (JPEG), 398
JPEG. See *Joint Photographic Experts Group*

K

KCC. See *Knowledge Consistency Checker*
Kerberos, 329
 instance map, 342
keymap (command), 65
K-Flex56 standard, 56
Knowledge Consistency Checker (KCC), 518

L

LAN. See *Local Area Network*
LAPB. See *Link Access Procedure on the B channel*
LAPD. See *Link Access Procedure on the D channel*
LAPM. See *Link Access Procedure for Modems*
Large networks
 FAQs, 482–483
 introduction, 458
 subnetting, 457
Large-scale DDR networks, 164
LAT, 56, 62, 63, 327. *See also Digital Equipment Corporation*
 functionality, 60
 host, 67
 network, 60, 61
Latency, 402
 type, 19
LATHOST, 61
Layer 1, 237, 263
Layer 2, 237, 263

encapsulation, 13
 protocol, 192
Layer 3, 237, 263
Layer 3 problems, 102
LCN. See *Logical Channel Number*
LCP. See *Link Control Protocol*
LE. See *Local Exchange*
Learned services, 60
Legacy DDR, 160, 213, 346
 configuration, 152
Legacy Windows NT domains, 506
Lempel-Ziv (LZ) algorithm, 399
Line configuration, 39, 53, 71
 mode, 84
Linecode, 202
Link
 compression, 399–401
 configuration/management/termination, 79
 configuration/negotiation, 77
 control, options. *See Point-to-Point Protocol*
 design, 321–322
 layer, 250
 utilization, 397
Link Access Procedure encapsulation type, 399
Link Access Procedure for Modems (LAPM), 37
Link Access Procedure on the B channel (LAPB), 159, 401
 mode, 239
Link Access Procedure on the D channel (LAPD), 8
Link Control Protocol (LCP), 12, 79–81, 400
 option negotiation, 95
 packets, 87
Linux, 479
Livingstone Enterprises, Inc., 333
LMI. See *Local Management Interface*

- Load balancing, 322–323. *See also* Per destination load balancing; Per packet load balancing
- Load distribution. *See* Transmission Control Protocol
- Load sharing, 322
- Local address, 410. *See also* Inside local address; Outside local address
- Local Area Network (LAN), 2, 114, 166, 446, 459, 469. *See also* Administrative LANs; Distribution centers; Headquarters; Store LANs; Virtual LAN
- addresses, 473
- connectivity, 249
- interfaces, 445. *See also* Fixed-LAN interfaces
- protocol address mapping, 245
- routing. *See* Inter-VLAN routing
- Local echo, 239
- Local Exchange (LE), 156, 157
- Local groups, 511
- Local IP address, 415–417, 434
- Local loop, 3
- Local loopback, 266
- Local Management Interface (LMI), 252
- status messages, 270
- type, 252, 267
- Local POP, 3
- Logical Channel Number (LCN), 240
- Login authentication, 340
- list, 339
- Login authorization methods, 339
- Loopback. *See* Local loopback; Remote loopback
- interfaces, 49, 481–482
- tests, 266–271
- Lossless compression
- algorithm, 397
- schemes, 398
- Lossy compression, 397
- Lucent, 259, 369
- LZ. *See* Lempel-Ziv
- LZS, 399, 400
- ## M
- MainRouter, 313, 314
- Management, ease, 328
- Maximum Receive Unit (MRU), 79, 86
- Maximum Received Reconstruction Unit (MRRU), 95
- MCI WorldCom, 248
- MD5. *See* Message Digest 5
- Media, 346
- Memory utilization, 402
- Menus, 67–69
- Message Digest 5 (MD5), 118, 129
- Messaging, 327
- Method-list, 332–333, 345, 359, 369. *See also* User-defined method-lists
- MIB information, 454
- Microcom networking protocol (MNP), 36–37
- MNP 1, 36
- MNP 2, 36
- MNP 3, 36, 38
- MNP 4, 37
- MNP 5, 37
- MNP 6/7/9/10, 37
- Microsoft Callback (MSCB), 91, 93, 104, 108
- Microsoft CHAP (MS-CHAP), 77, 89, 104
- Microsoft Point-to-Point Compression (MPPC), 93, 400
- encapsulation types, 399
- MILNET, 333
- Mirror access list, 123
- Mirror image access list, 124
- Mission-critical applications, 19

- MMP. *See* Multichassis multilink PPP
- MNP. *See* Microcom networking protocol
- modem autodiscovery (command), 53
- modem callin (command), 45
- modem dialin (command), 45, 46
- modem inout (command), 45, 47
- Modemcap
- database, 52, 53
 - entry, 53
- Modems. *See* Asymmetric Digital Subscriber Line; Cable modems; Digital modems; Link Access Procedure for Modems
- automatic configuration, 51–55
 - configuration, 48–55
 - connections. *See* Analog modems; Asynchronous modem connections
 - data flow, 42
 - manual configuration, 48–51
 - modulation standards, 34–35
 - overview, 30–32
 - signaling/cabling, 32–33
 - speed, 38
 - vendors, 35
- Modulation standards. *See* Modems
- Motion Pictures Experts Group (MPEG), 398
- MP. *See* Multilink PPP; Multilink Protocol
- MPEG. *See* Motion Pictures Experts Group
- MPPC. *See* Microsoft Point-to-Point Compression
- MRRU. *See* Maximum Received Reconstruction Unit
- MRU. *See* Maximum Receive Unit
- MSCB. *See* Microsoft Callback
- MS-CHAP. *See* Microsoft CHAP
- Multicast. *See* Internet Protocol traffic, 377
- Multichannel E1 line, 17
- Multichannel T1 line, 17
- Multichassis multilink PPP (MMP), 96–98
- Multihomed areas, 476
- Multilink, 12
- monitoring. *See* Point-to-Point Protocol
- Multilink PPP (MP), 94–96. *See also* Multichassis multilink PPP
- Multilink Protocol (MP), 82, 97
- Multiple protocols. *See* Communication lines
- encapsulation, 12
 - Multiple topologies, 148
 - Multiple-access directory, 533
 - Multiplexing, 239
 - Multipoint subinterfaces, 257
 - Multipoint WAN, 166
 - Mutual redistribution, 198
- ## N
- NACK, 79
- NAS. *See* Network access server
- NASI. *See* Novell Asynchronous Services Interface
- NAT. *See* Network Address Translation
- National Terminal Number (NTN), 241, 242
- NBMA. *See* Nonbroadcast multiaccess
- NCP. *See* Network Control Protocol
- NetBEUI, 104
- NetBIOS, 498
- NetSys program, 24
- Network. *See* Internetwork; Large networks; Overlapping networks
- adapter, 530
 - administrator, 391
 - analysis, 18–21
 - business justifications, 20–21

- connections, 253. *See also Point-to-Point Protocol*
- cost, 19–20
- downtime, 21
- installation
 - considerations. *See Remote access network*
 - plan, 20
 - verification/troubleshooting, 25
- interruption, minimization, 25
- joining, 467
- layer, 237, 263
- monitor, 497
- needs, 18–19
- planning/design, 18–24
- protocols, 346
- resources, 20
- size, 207
- time frame, 19
- traffic conditions, 377
- training, 20
- Network access server (NAS), 326, 344, 346, 358
 - configuration, 355–357
 - database, 343
 - IPSec configuration, 131–135
 - router, 132
- Network Address Translation (NAT), 126, 148. *See also Dynamic NAT; Static NAT*
 - commands, 413–414
 - FAQs, 454–455
 - introduction, 408
 - operation, 411–412
 - overview, 408–412
 - pools, 417, 434, 443
 - requirements, 407. *See also Remote access networks*
 - router, 424, 442, 455
 - software, 423
 - terminology, 409–411
 - timeouts, changing, 443–444
 - usage. *See Internet Service Provider*
- walkthrough, 450–453
- Network Control Protocol (NCP), 12, 77, 81
- Network Information Center (NIC), 410
- Network performance, 372–373
- Network performance optimization
 - FAQs, 404–405
 - introduction, 372
 - queuing/compression, usage, 371
- Network Termination
 - 1 (NT1), 156
 - device, 167
 - line, 8. *See also Built-in NT1 line; Integrated NT1 line*
 - 2 (NT2), 156
- Network-to-Network Interface (NNI), 248
- Next-hop addresses, 166
- NFS, 412
- NIC. *See Network Information Center*
- NNI. *See Network-to-Network Interface*
- no ip route-cache (command), 323
- Nonbroadcast multiaccess (NBMA), 257
- Non-interface specific configuration commands, 351
- Non-ISDN device, 156
- Non-registered addresses, 408
- Nonreversible compression, 397
- Non-TCP frames/packets, 437
- Nonvolatile RAM (NVRAM), 50, 51, 83
- Not So Stubby Area (NSSA), 182
- Novell, 269
 - standard, 76–77
- Novell Asynchronous Services Interface (NASI), 334
- NSSA. *See Not So Stubby Area*
- NT1. *See Network Termination*
- NT2. *See Network Termination*
- NTDS Site Settings, 518, 538
- NTFS. *See Windows NT*

NTN. *See* National Terminal Number
 NVRAM. *See* Nonvolatile RAM
 Nyquist, Harry, 31

O

Oakley protocol, 115
 Objects, 504–516. *See also* Hidden objects
 management. *See* Connections
 OC-3, 16, 290. *See also* Packet OC-3
 OC-12, 290
 OCMANAGER, 495
 OCManger. *See* Optional Component Manager
 Offline configuration, support, 148
 Off-peak times, 25
 On-demand circuits. *See* Open Shortest Path First
 On-demand reports, 148
 One-to-one address mappings, 443
 One-to-one IP address mappings, 425
 One-way hash, 88
 Open Shortest Path First (OSPF), 309, 393
 configuration, 182
 on-demand circuits, 181–182
 on-demand configuration, 182
 on-demand operation, 202
 on-demand routing, 181, 195, 199
 route, 203
 Open System Interconnection (OSI), 12
 Layer 2 frame, 77
 Layer 4 ports, 81
 layers, 157
 model, 31, 99, 158, 239
 Operating system, configuration, 487, 488
 Optional Component Manager (OCManger), 496
 Organizational units (OU), 504–516

authority, delegation, 506–507
 creation, 505–506
 hierarchy, 506
 OSI. *See* Open System Interconnection
 OSPF. *See* Open Shortest Path First
 OU. *See* Organizational units
 Outbound traffic, 143
 Output errors, 266
 Outside components, 410
 Outside global address, 411
 Outside interface, 435, 439
 Outside local address, 410–411
 Overlapping networks, 430–436
 configuration, 434–436
 Ownership, total cost, 21

P

Packet
 header size, 397
 size, 383
 switching, 9, 10
 trains, 392
 transfers, 374
 Packet Assembler/De-assembler (PAD), 238, 344
 Packet Layer Protocol (PLP), 237, 239
 Packet OC-3, 17
 Packet Switched Network (PSN), 241
 Packetized Ensemble Protocol (PEP), 35
 Packet-switched connections, 4, 10–11
 PAD. *See* Packet Assembler/De-assembler
 PAP. *See* Password Authentication Protocol
 PASS message, 332
 PASS reply, 359
 PASS result, 360
 Password Authentication Protocol (PAP), 12, 76, 80
 authentication, 86–90

- PAT. *See* Port Address Translation
- Payload**, 291
- compression, 399–401, 403
- Payload Type Indicator (PTI)**, 291
- PCMCIA**. *See* Personal Computer Memory Card International Association
- PDC**. *See* Primary Domain Controller
- Peer**, 89, 118, 120
- authentication, 123
 - defining, 129
 - identity, 121, 132
 - interface, 86
 - node. *See* Border Gateway Protocol
- Peer-to-peer VPN relationship**, 142
- PEP**. *See* Packetized Ensemble Protocol
- Per-destination load balancing**, 322
- Per-packet load balancing**, 323
- Performance**
- issues, 321–322
- Per-interface compression**, 401
- Permanent connections**
- backup, 305–323
 - backup/configuration, 233
 - FAQs, 324
 - introduction, 234
- Permanent virtual circuit (PVC)**, 10, 237, 240, 249, 400
- activity, 267
 - circuit buildup, 292–293
 - mapping, 257, 292–293
 - name, 527
 - status, 265
- Per-service basis**, 329
- Personal Computer Memory Card International Association (PCM-CIA)**, 15
- Per-use authentication**, 327
- Per-user configuration**, 220, 347, 352, 353
- example, 354–357
- Per-virtual circuit compression**, 401
- Phase I**, 327
- Phase II**, 327
- Physical interface**, 212, 214
- Physical layer**, 99, 100, 237
- troubleshooting, 263–266
- Physical location**, 321
- PING**, 419, 424, 425, 454
- ping (command)**, usage, 25
- PIX Firewall**, 119, 126, 135, 137, 139
- Secure Telnet method, 148
 - usage, 136
- PKI**. *See* Public key infrastructure
- PLP**. *See* Packet Layer Protocol
- Plug and Play (PNP)**, 496
- PNP**. *See* Plug and Play
- Point of Presence (POP)**, 3. *See also* Local POP
- Point-to-multipoint connections**, 11
- Point-to-point connections**, 4
- configuration, 234–236
- Point-to-point dedicated links**, 234
- Point-to-point links**, 76, 462
- Point-to-point network**, 235, 257, 461
- Point-to-Point Protocol (PPP)**, 2, 11, 12, 97, 234, 334, 446. *See also* Multichassis multilink PPP; Multilink PPP
- addressing methods, 84–85
 - authentication, 166
 - callback, 91–93
 - Cisco access servers, usage, 99
 - compression, 93
 - configuration, 83–98
 - connections. *See* Cisco routers
 - dialer map access, 92
 - enabling, 84
 - encapsulation, 76, 168
 - features, 77–78
 - frame, 77
 - link control options, 86–90

- listing, 14
- mode, 398
- multilink, monitoring, 188–189
- negotiation, 448
- network connections, 30
- options (Microsoft specific), 104–105
- overview, 76–82
- packet, 93
- protocol, 82
- session, 219
- setup. *See* Wide Area Network
- SLIP/ARAP, contrast, 81–82
- troubleshooting, 99–103
- usage. *See* Remote network access
- verification, 99–103
- Point-to-point subinterface, 257
- Point-to-point topology, 162
- Point-to-Point Tunneling Protocol (PPTP), 16
- Point-to-point WAN links, 459
- Poison reverse, 255–257
- Policy Manager 2.x, 147–148
- Pools, 414, 439. *See also* Network Address Translation
- POP, 230. *See* Point of Presence
- Port Address Translation (PAT), 411, 435
 - configuration, 450
 - overloading, 412
 - parameters, 448
 - definition, 445
 - statistics, 414
 - usage, 446. *See also* Internet Service Provider
- PPP. *See* Point-to-Point Protocol
- PPTP. *See* Point-to-Point Tunneling Protocol
- PQ. *See* Priority Queuing
- Pre-boot-Execution Environment (PXE), 492
- Precedence. *See* Internet Protocol bandwidth, 380, 381
- Predictor, 12, 401
 - compression algorithm, 399, 400
- Pre-shared keys, 121, 130, 136, 137
- PRI. *See* Primary Rate Interface
- Price-per-performance solution, 20
- Primary Domain Controller (PDC), 498–500, 508
- Primary Rate Interface (PRI), 8, 32, 156–157, 210
 - functional groups, 157
 - functionality, 158
 - interface, 166, 167, 202, 214, 226
 - support, 97
 - lines, 153
 - reference points, 157
 - usage, 195
- Printers, publishing, 513–514
- Priority Queuing (PQ), 272, 373, 383–387, 397
 - examples, 385–386
- Private addresses, 459–460, 482, 483
 - space, contrast. *See* Public address space
- Private addressing, 457
 - FAQs, 482–483
 - introduction, 458
- Private network, subnetting strategy.
 - See* Class A private network
- Private network addresses, 465–468
 - considerations, 466–467
 - usage, timing, 467–468
- Protocol frame errors (FRMRs), 246
- Protocol next-hop-address, 177
- Protocols, 78. *See also* Data-link layer
 - protocol; Integrated Services Digital Network; Security protocols; Wide Area Network
 - operator, 176
 - type, 383
 - usage, 19
- PSN. *See* Packet Switched Network

PSTN. *See* Public Switched Telephone Network
 PTI. *See* Payload Type Indicator
 Public address spaces, private address space contrast, 463
 Public key infrastructure (PKI), 143, 522–525
 policies, 524
 systems, 144
 Public Switched Telephone Network (PSTN), 2, 30, 31, 91, 131, 160, 346. *See also* Dial-up PSTN
 analog network, 203
 PVC. *See* Permanent virtual circuit
 PXE. *See* Pre-boot-Execution Environment

Q

Q series, 153
 Q921 information, 192
 Q.931, 159
 Q933a, 252
 Q-series, 34
 Quality of Service (QoS), 144, 327, 373, 378, 404
 mechanisms, 373
 Queues. *See* Dynamic queues; Reversible queues
 definitions, 384
 operation, verification, 404
 Queuing. *See* Custom queuing; Default queuing; Priority queuing
 configuration, 373–394
 methods, 373–394
 selection. *See* Cisco IOS
 operation, verification, 395
 overview, 373–395
 policy, 393
 usage. *See* Network performance optimization

R

RADIUS. *See* Remote Authentication Dial-in User Service
 Random early detection (RED), 372, 395. *See also* Weighted random early detection
 RAS. *See* Remote Access Service
 RCMD. *See* Remote command RCP server, 447
 RDP. *See* Remote Desktop Protocol
 Ready To Send (RTS), 33, 43, 263
 Real IP address, 436, 439
 Real-time traffic, 377
 Real-time Transport Protocol (RTP), 399
 header compression, 403
 Receive (RX), 33
 Receiver Not Ready events (RNRs), 246
 Recovery Console, installation, 503–504
 RED. *See* Random early detection
 redistribute routing-protocol (command), 182
 Redundancy, importance, 19
 Redundant hardware, 321–322
 Reference points. *See* Basic Rate Interface; Primary Rate Interface
 regexp. *See* Regular expression
 Regular expression (regexp), 56
 Rejects (REJs), 246
 Reliability, 19
 Remote access connectivity, 113, 151
 FAQs, 149, 205–207
 introduction, 114, 152
 Remote access networks
 FAQs, 368–370
 implementation considerations, 24–25
 installation considerations, 17–24
 introduction, 326–328

- NAT, requirements, 407
- securing, 325
- Remote Access Service (RAS), 104, 519
 - usage. *See* Windows clients
- Remote access VPN, 130–143
- Remote administration, 528
- Remote Authentication Dial-in User Service (RADIUS), 326, 329, 333, 342, 344, 534
 - authentication request, 360
 - comparison. *See* Terminal Access Controller Access Control System Plus
 - configuration. *See* User Remote RADIUS configuration
 - database, 342
 - login, 333
 - parameters, configuration, 336–338
 - protocol, 330
 - servers, 331, 342, 344, 359
 - host, defining, 338
 - support, 335
 - usage, 341. *See also* Authentication authorization and accounting
- Remote command (RCMD), 334
- Remote control, 532
 - settings, 530
- Remote CSU/DSU, 267
- Remote Desktop Protocol (RDP), 530
- Remote Installation Service (RIS), 492
 - server, 495
 - service request, 493
- Remote loopback, 267–271
- Remote network access
 - FAQs, 112
 - introduction, 76
 - PPP usage, 75
- Remote Operating System installation, 492, 493
- Remote router host name/password, 83
- Remote site, 305
 - connectivity, 254
- Remote system logon information, 168
- Renumbering. *See* Addresses
- Request For Comments (RFC)
 - 1332–1334, 82
 - 1378, 82
 - 1517–1519, 459
 - 1552/1553, 81, 82
 - 1570, 82
 - 1597, 465
 - 1627, 465
 - 1631, 455
 - 1633, 404
 - 1661, 82
 - 1717, 94
 - 1793, 181
 - 1917, 462–463
 - 1918, 408, 410, 430, 465–468, 470
 - 1990, 82
 - 1994, 82
 - 2131, 467
 - 2138, 3233
 - 2212, 404
 - 2309, 404
- Resource Reservation Protocol (RSVP), 377–378
- Resources. *See* Network coordination, 24, 25
- Response time, 19
- Reversible queues, 377
- Reverse access, 342
- Reverse Telnet session, 49
- RFC. *See* Request For Comment
- RI. *See* Ring Indicator
- Ring Indicator (RI), 33, 46
- RIP, 198, 393
 - version 1, 461, 475
- RIPREP, 487, 492–495, 541
- RIS. *See* Remote Installation Service
- Rivest, Shamir, Adelman (RSA) system, 144

- Rlogin, 57, 59, 327, 412
 program. *See* BSD UNIX
- RNRs. *See* Receiver Not Ready events
- Rollback mechanism, 148
- Root domain, 498
- Rotary groups, 212
- Route summarization, 474
- Router-based VPN, 128
- Routers. *See* Cisco routers
- configuration, 39–48
 - efficiency, 468, 474–475
 - FW feature set, 147
 - platforms, 22
 - selection. *See* Cisco access
- Router-to-router, 76
- Routes. *See* Default routes; Static routes
- redistribution, 182
- Routing. *See* Snapshot routing
- issues, 321. *See also* Dial-on demand routing
 - loops, 255
- Routing and Remote Access Services (RRAS), configuration, 534–535
- Routing Table Maintenance Protocol (RTMP), 181
- Routing Table Manager (RTM), 256, 309
- Routing Table Protocol (RTP), 181
- RRAS. *See* Routing and Remote Access Services
- RSA. *See* Rivest, Shamir, Adelman
- RST. *See* Transmission Control Protocol
- RSVP. *See* Resource Reservation Protocol
- RTM. *See* Routing Table Manager
- RTMP. *See* Routing Table Maintenance Protocol
- RTP. *See* Real-time Transport Protocol; Routing Table Protocol
- RTS. *See* Ready To Send
- RX. *See* Receive
- ## S
- S register, 51
- SA. *See* Internet Protocol Security; Internet Security Association and Key Management Protocol; Security association
- SABMs. *See* Set Asynchronous Balance Mode requests
- Sales departments, 21
- SAM. *See* Security Accounts Manager
- Sampling Theorem, 31
- SAP. *See* Service Advertising Protocol
- SAR. *See* Segmentation Assembly and Re-assembly
- Scalability, 148, 328, 329
- Scripted installation, overview, 488–490
- Scripts. *See* Chat scripts
- command, 55
- SDLC. *See* Synchronous Data Link Control
- SE. *See* Shared-explicit
- Secure Gateway Tunnel, 138
- Secure Hash Algorithm (SHA), 118
- Security, 466
- breaches, 467
 - protocols, 333–335
 - server, choice, 335
- Security Accounts Manager (SAM), 499
- Security association (SA), 120, 136, 411. *See also* Internet Protocol Security; Internet Security Association and Key Management Protocol
- negotiation, 129, 139
- Segmentation, 290
- Segmentation Assembly and Re-assembly (SAR), 291

- Serial interfaces, 219, 340
Serial Line Internet Protocol (SLIP), 11, 12, 167
 contrast. See Point-to-Point Protocol
Serial link, 372
Serial ports. See Asynchronous serial ports
Servers
 hardware, 487
 installation, 487, 499–500
 selection. See Cisco access usage. See Point-to-Point Protocol
Service Advertising Protocol (SAP), 181
Service announcements, 60
Service Profile IDentifier (SPID), 179, 185, 320
Service providers, 21
 solution, 135–140
Services, 60. *See also* Learned services
Sessions, 531
Set Asynchronous Balance Mode requests (SABMs), 246
SF. *See* Supervisory Frames
SGBP. *See* Stack Group Bidding Protocol
SHA. *See* Secure Hash Algorithm
Shared reservation style, 377
Shared-explicit (SE) style, 378
Shiva, 369
show caller (command), 99
show (commands), 358
show crypto engine connections (command), 141
show crypto ipsec sa (command), 140
show crypto isakmp policy (command), 121
show crypto map (command), 125
show dialer (command), 187
show frame-relay lmi (command), 267, 280
show frame-relay map (command), 280
show frame-relay pvc (command), 280
show interface (command), 13, 280
 usage, 25
show interface serial (command), 183
show interface virtual-access number (command), 361–362
show interfaces serial exec (command), 246
show ip route (command), 280
show isdn status (command), 185, 186
show lat sessions (command), 62
show modem log (command), 99
show modemcap name (command), 52
show modemcap usr_courier (command), 52
show snapshot (command), 190
show x25 interface (command), 246
show x25 map (command), 246
show x25 services (command), 247
show x25 vc (command), 247
Shutdown, 349
SIDs, 492
Signaling. *See* Modems
Simple Mail Transfer Protocol (SMTP), 58, 230, 327, 449, 517
 server, 450, 452
Single-homed connections, 476
Sites. *See* Central site
Sites, equipment usage, 21–23
Sites, setup, 516–519
Site-to-site VPN, 119–130
Skeme protocol, 115
SLDC. *See* Synchronous Data Link Control
SLIP. *See* Serial Line Internet Protocol
Small Office/Home Office (SOHO), 2, 14, 21, 145, 449
 environments, 15
Smart Jack, 266
Smart Serial port, 15

- SMTP. *See Simple Mail Transfer Protocol*
- SNA. *See Software Network Architecture; Systems Network Architecture*
- Snapshot routing, 180–181, 206
monitoring, 189–190
- SNMP, 412, 443, 454
- Software Network Architecture (SNA), 385
- SOHO. *See Small Office/Home Office*
- SONET. *See Synchronous Optical Network*
- Source address, 422
- Source IP address, 427
- Source node, 251
- SPID. *See Service Profile IDentifier*
- Split horizon, 255–257
- Spoofing, 319
state, 183
- S/T interface, 15
- STAC, 400, 401
compress utilities, 399
- Stack group, 97
- Stack Group Bidding Protocol (SGBP), 97
- Stacker, 12, 400
- Static IP addresses, 450
- Static NAT
translation, 426
configuration, 427–428
output, 428–430
- Static routes, 83, 180. *See also Floating static routes*
- Static translation, 425–427, 430, 443
- Statistical compression, 398
- Store LANs, 473
- Stratacom, 259
- Stub areas, 476
- Subinterfaces, 257–258. *See also Multipoint subinterfaces*
- Sublayers, 78
- Subnet, 474
bits, 468
- Sub-protocol type, 383
- Supervisory Frames (SF), 238, 239
- SVC. *See Switched virtual circuit*
- Switched DSU/CSU, 168
- Switched virtual circuit (SVC), 10, 237, 240, 249, 400
- Synchronous communications, 36
- Synchronous connection, 77
- Synchronous Data Link Control (SDLC), 11, 234–235
- Synchronous dial-up, 15
- Synchronous lines, 94
- Synchronous Optical Network (SONET), 13
- Synchronous serial interfaces, 16, 165, 168
- SYSPREP, 487, 491–494, 541
- SYSSETUP, 495
- SysSetup, 496
- Systems Network Architecture (SNA), 11

T

- T1 circuits, 230
- T1 connections, 145
- T1 line, 4, 15. *See also Fractional T1 line; Multichannel T1 line*
- T1 PRI, 202
- T3 line, 4, 10. *See also Channelized T3 line*
- TA. *See Terminal Adapter*
- TACACS+. *See Terminal Access Controller Access Control System Plus*
- Tail drop, 396
- TAR file, 37
- TCP. *See Transmission Control Protocol*

- TCP/IP. *See* Transmission Control Protocol/Internet Protocol
- TE1. *See* Terminal Equipment
- TE2. *See* Terminal Equipment 2
- TEI. *See* Terminal Endpoint Identifier
- Telebit, 35
- Telecommuter/mobile user, 21–23
- Telecommuters, 14
- Telnet, 59, 224, 230, 327, 375, 412
 configuration, 57
 packets, 60
 prompt, 444
 reversing, 30
 servers, 449, 450, 452
 sessions, 67, 342. *See also* Reverse Telnet session
 traffic, 385, 388
- Templates, 148
- Terminal Access Controller Access Control System Plus (TACACS+), 326, 329, 333–334, 342, 344
- authorization, 343
- login, 333
- optional commands, 337, 338
- packets, 360
- parameters, configuration, 336–338
- protocol, 330
- RADIUS, comparison, 334
- security servers, 344
- servers, 331, 342, 346, 351, 365, 366
 host, defining, 337
 support, 335, 368
 usage, 341. *See also* Authentication authorization and accounting
- Terminal Adapter (TA), 155. *See also* Integrated Services Digital Network
- Terminal Endpoint Identifier (TEI), 100, 101, 103
- Terminal Equipment. *See* Data Terminal Equipment
- 1 (TE1), 156, 321
- 2 (TE2), 155
- Terminal Services, 57–66, 519, 527–534
- clients, creation, 533
- Profile, 532
- providing. *See* Asynchronous dial-in terminal services
- Test lab, staging/testing, 23–24
- Text mode, 495, 496
- TFTP. *See* Trivial File Transfer Protocol
- Third-party DHCP, 85
- Third-party dialing program, 104
- Throughput, 19, 397
- Time frame. *See* Network
- Timeouts, 178, 443. *See also* Internet Protocol addresses
 changing. *See* Network Address Translation
- Time-sensitive traffic, 375
- TN3270. *See* IBM
- Token Ring, 16, 17
- Topologies, 162. *See also* Frame Relay; Fully meshed topology; Hub-and-spoke topology; Multiple topologies; Point-to-point topology; Wide Area Network
 choice, 21, 205
 usage, 206
- TOS. *See* Type of service
- traceroute (command), usage, 25
- Traffic. *See* Interesting traffic; Internet Protocol; Multicast; Real-time traffic; Time-sensitive traffic; Unicast traffic
 conditions. *See* Network types, support, 412
- Traffic shaping. *See* Frame Relay Traffic Shaping
- configuration, 272–280
- verification, 280–289
- Trailer flag, 239

Transform set, 118, 123
 Transit areas, 476
Translation. See *Dual address translation; Dynamic NAT; Dynamic translation; Inside source addresses; Network Address Translation; Static NAT; Static translation*
Transmission Control Protocol (TCP), 334, 395, 396, 412
 connections, 479, 481
 FIN bit, 440
 FIN packet, 444
 load distribution, 436–442, 438–440, 451
 output, 440–442
 port 179, 477
 port 3389, 530
 protocol, 479
 RST bit, 440
 RST packet, 444
 sessions, 396
 source port numbers, 421
Transmission Control Protocol/Internet Protocol (TCP/IP), 12
 connections, 528
 header compression, 398
 login, 57
 protocols, 459
 routing table, 67
 running, 131
 settings, 105, 106
 support, 104
Transmission errors, 77
Transmit (TX), 33
Transmitting router, 272
Transport type, 530
Triple DES (3DES), 116, 118, 145
Triple pass DES, 116
Trivial File Transfer Protocol (TFTP), 412, 443

server, 447
Troubleshooting, 24, 25. *See also Dial-on demand routing; Integrated Services Digital Network; Network; Point-to-Point Protocol; Virtual Private Network; Windows connections*
Trusted host, 57
TTY lines, 38, 39
TTYcap (command), 65
TX. *See Transmit*
Tymnet, 241
Type of service (TOS), 378

U

U interface, 8
UBR, 391
UDP. *See User Datagram Protocol*
UF. *See Unnumbered Frames*
Unattend.txt, 487
UNI. *See User-Network Interface*
Unicast traffic, 377
Uninteresting traffic, 161
Universal groups, 511
UNIX
 CiscoSecure ACS, 330, 331
 CiscoSecure Global Roaming Server (GRS), 330, 331
 clients, 56
 compress utilities, 399
 environment. *See BSD UNIX*
 hosts, 66
 server, 333
 variants, 479
Unnumbered Frames (UF), 238, 239
Unshielded twisted-pair (UTP), 76
U-plane, 158–159
Upstream traffic, dial-up connection, 15
U.S. Robotics, 35

- User account, creation, 508–510
 User actions, shadowing, 530
 User data, 251
 User Datagram Protocol (UDP), 333, 437
 bootpc, 388
 header, 399
 packet, 493
 source port numbers, 421
 traffic, 443
 User growth, 19
 User Remote RADIUS configuration, 354–357
 User-defined method-lists, 331
 User-defined traffic, 390
 User-dependence, 347
 Username line, 91
 User-Network Interface (UNI), 248
 User-specific command-line configuration commands, 348
 UTP. *See* Unshielded twisted-pair
- V**
- V.22, 34
 V.22bis, 34
 V.25bis, 168
 V.32, 34
 V.32bis, 34, 35
 V.32ter, 34
 V.34, 34
 V.35, 5, 31, 35
 V.42, 35
 V.42bis, 35, 37
 v.90, 35
 V.120 encapsulation, 120
 Variable-Length Subnet Mask (VLSM), 459, 462, 483
 VC. *See* Virtual circuit
- VCI. *See* Virtual Channel Identifier; Virtual Circuit Identifier
 Vendors, 20, 21. *See also* Modems
 Versatile Interface Processor (VIP), 372
 Vines, 77, 181
 VIP. *See* Versatile Interface Processor
 VIP DWFQ, 381–383
 VIP2, 402
 Virtual access interface, 354, 361
 configuration, 356
 Virtual Channel Identifier (VCI) number, 527
 Virtual channels, 247
 Virtual Circuit Identifier (VCI), 291, 292
 Virtual circuit (VC), 10, 240, 257. *See also* Asynchronous Transfer Mode; Permanent virtual circuit; Switched virtual circuit; X.25
 compression. *See* Per-virtual circuit compression
 Virtual IP address, 436, 438, 442
 Virtual LAN (VLAN), 16
 routing. *See* Inter-VLAN routing
 Virtual Path Identifier (VPI), 291, 292
 Virtual private dial-up network (VPDN), 347
 Virtual Private Network (VPN), 3, 21, 113, 534. *See also* Remote access VPN; Router-based VPN; Site-to-site VPN
 advantages/disadvantages, 143–144
 Cisco solutions, 145–148
 client, 16
 configuration, 138–140
 concentrators, 119, 126, 145, 146
 configuration/management, 147–148
 FAQs, 149
 information, 26
 introduction, 114

- operation, 116–117
 verification/debugging, 140–143
- relationship. *See* Peer-to-peer VPN relationship
- support, 328
- technology, 114–119
- terminology. *See* Cisco VPN terminology
- troubleshooting, 143
- Virtual profiles**
- AAA configuration, usage, 348, 349, 352–354
 - configuration, 349–354
 - usage. *See* Authentication authorization and accounting
 - virtual templates, usage, 347–354 examples, 350–351
- Virtual templates.** *See* Interfaces interface, 360
- usage. *See* Virtual profiles
- Virtual type terminal (VTY) ports**, 38
- VLAN.** *See* Virtual LAN
- VLSM.** *See* Variable-Length Subnet Mask
- VMS hosts.** *See* Digital Equipment Corporation
- VoATM.** *See* Voice over ATM
- VoFR.** *See* Voice over Frame Relay
- Voice over ATM (VoATM)**, 373
- Voice over Frame Relay (VoFR)**, 373
- Voice over Internet Protocol (VoIP)**, 373
- Voice over X (VoX)**, 373
- VoIP.** *See* Voice over Internet Protocol
- VoX.** *See* Voice over X
- VPDN.** *See* Virtual private dial-up network
- VPI.** *See* Virtual Path Identifier
- VPN.** *See* Virtual Private Network
- V-series**, 34
- VT100 terminal emulation**, 65
- VTY.** *See* Virtual type terminal
- VXR**, 402
- ## W
- Wait-for-carrier-time**, 225, 226
- WAN.** *See* Wide Area Network
- War dialer**, 86
- Web server**, 426–428, 437, 440, 452 farm, 464
- Weighted Fair Queuing (WFQ)**, 372, 375–383. *See also* Class-based Weighted Fair Queuing; VIP DWFQ
- functionality, 390
- IP precedence, usage, 378–381
- Weighted Random Early Detection (WRED)**, 391. *See also* Flow-based WRED
- overview, 395–403
- WF.** *See* Wildcard-filter
- WFQ.** *See* Weighted Fair Queuing
- WIC.** *See* Wide Area Network Interface Card
- Wide Area Network Interface Card (WIC)**, 15, 16
- Wide Area Network (WAN)**, 10, 234, 397, 401. *See also* Multipoint WAN
- circuit, 446
- connection requirements, 2–3
- encapsulation protocols, 11–14
- growth, 21
- interface, 445
- links, 154, 372, 398, 462, 473–475. *See also* Headquarters; Point-to-point WAN links
- PPP/IPCP setup**, 445
- specifications, 3–11
- technology, 11, 14, 461
- topology, 3–11, 402

- usage costs, 160
 - Wildcard-filter (WF) style, 378**
 - Windows 95, 16, 362, 488**
 - clients, 105
 - Windows 98, 16, 95, 488**
 - clients, 105–107
 - Windows 2000, 91, 104, 488**
 - clients, 108–110
 - components, installation/ configuration, 519–537**
 - desktop, 108**
 - FAQs, 544–545**
 - installation, 487–497**
 - introduction, 486–487**
 - servers, 487, 535**
 - case studies, 537–540
 - implementation, 485
 - setup phases, 495–497
 - Windows clients, remote access servers**
 - usage, 104
 - Windows connections, troubleshooting, 110–111**
 - Windows Explorer, 514**
 - Windows Internet Naming Service (WINS), 449, 537**
 - Windows NT, 91, 511**
 - CiscoSecure ACS, 330
 - client dialing, 365
 - domains, 499. *See also* Legacy Windows NT domains
 - File System (NTFS), 500, 503
 - platforms, 330
 - remote client, 362
 - server, 333
 - Windows NT4 clients, 107–108**
 - WINNT, 488, 495, 540**
 - phase, 496
 - WINS. *See* Windows Internet Naming Service**
 - Wireless bridges, 6**
 - Wireless options, 6**
 - Workforce efficiency, 20**
 - Workstation installations, 487**
 - WRED. *See* Weighted Random Early Detection**
- X**
- X2 protocols, 35**
 - X.3, 239**
 - X.21, 5, 31, 239**
 - X.25, 11–13, 234, 401**
 - call setup, 240–241
 - configuration, 241–245
 - connections, 237–248
 - troubleshooting/verification, 245–248
 - data-link, 167**
 - disconnection, 240–241**
 - encapsulation, 168**
 - types, 399
 - events, 245**
 - frames, 238–240**
 - network, 238**
 - overview, 237–240**
 - service contributor, 247**
 - state transitions, 247**
 - technology, 324**
 - virtual circuits, 240–241**
 - X.29, 239**
 - X.75, 240**
 - X.121, 240–242**
 - address, 245, 246
 - X-ON/X-OFF, 33**
 - X-series, 34**
- Z**
- ZIP compress utilities, 399**
 - ZIP file, 37**

The Global Knowledge Advantage

Global Knowledge has a global delivery system for its products and services. The company has 28 subsidiaries, and offers its programs through a total of 60+ locations. No other vendor can provide consistent services across a geographic area this large. Global Knowledge is the largest independent information technology education provider, offering programs on a variety of platforms. This enables our multi-platform and multi-national customers to obtain all of their programs from a single vendor. The company has developed the unique Competus™ Framework software tool and methodology which can quickly reconfigure courseware to the proficiency level of a student on an interactive basis. Combined with self-paced and on-line programs, this technology can reduce the time required for training by prescribing content in only the deficient skills areas. The company has fully automated every aspect of the education process, from registration and follow-up, to "just-in-time" production of courseware. Global Knowledge through its Enterprise Services Consultancy, can customize programs and products to suit the needs of an individual customer.

Global Knowledge Classroom Education Programs

The backbone of our delivery options is classroom-based education. Our modern, well-equipped facilities staffed with the finest instructors offer programs in a wide variety of information technology topics, many of which lead to professional certifications.

Custom Learning Solutions

This delivery option has been created for companies and governments that value customized learning solutions. For them, our consultancy-based approach of developing targeted education solutions is most effective at helping them meet specific objectives.

Self-Paced and Multimedia Products

This delivery option offers self-paced program titles in interactive CD-ROM, videotape and audio tape programs. In addition, we offer custom development of interactive multimedia courseware to customers and partners. Call us at 1-888-427-4228.

Electronic Delivery of Training

Our network-based training service delivers efficient competency-based, interactive training via the World Wide Web and organizational intranets. This leading-edge delivery option provides a custom learning path and "just-in-time" training for maximum convenience to students.

Global Knowledge Courses Available

Microsoft

- Windows 2000 Deployment Strategies
- Introduction to Directory Services
- Windows 2000 Client Administration
- Windows 2000 Server
- Windows 2000 Update
- MCSE Bootcamp
- Microsoft Networking Essentials
- Windows NT 4.0 Workstation
- Windows NT 4.0 Server
- Windows NT Troubleshooting
- Windows NT 4.0 Security
- Windows 2000 Security
- Introduction to Microsoft Web Tools

Management Skills

- Project Management for IT Professionals
- Microsoft Project Workshop
- Management Skills for IT Professionals

Network Fundamentals

- Understanding Computer Networks
- Telecommunications Fundamentals I
- Telecommunications Fundamentals II
- Understanding Networking Fundamentals
- Upgrading and Repairing PCs
- DOS/Windows A+ Preparation
- Network Cabling Systems

WAN Networking and Telephony

- Building Broadband Networks
- Frame Relay Internetworking
- Converging Voice and Data Networks
- Introduction to Voice Over IP
- Understanding Digital Subscriber Line (xDSL)

Internetworking

- ATM Essentials
- ATM Internetworking
- ATM Troubleshooting
- Understanding Networking Protocols
- Internetworking Routers and Switches
- Network Troubleshooting
- Internetworking with TCP/IP
- Troubleshooting TCP/IP Networks
- Network Management
- Network Security Administration
- Virtual Private Networks
- Storage Area Networks
- Cisco OSPF Design and Configuration
- Cisco Border Gateway Protocol (BGP) Configuration

Web Site Management and Development

- Advanced Web Site Design
- Introduction to XML
- Building a Web Site
- Introduction to JavaScript
- Web Development Fundamentals
- Introduction to Web Databases

PERL, UNIX, and Linux

- PERL Scripting
- PERL with CGI for the Web
- UNIX Level I
- UNIX Level II
- Introduction to Linux for New Users
- Linux Installation, Configuration, and Maintenance

Authorized Vendor Training

Red Hat

- Introduction to Red Hat Linux
- Red Hat Linux Systems Administration
- Red Hat Linux Network and Security Administration
- RHCE Rapid Track Certification

Cisco Systems

- Interconnecting Cisco Network Devices
- Advanced Cisco Router Configuration
- Installation and Maintenance of Cisco Routers
- Cisco Internetwork Troubleshooting
- Designing Cisco Networks
- Cisco Internetwork Design
- Configuring Cisco Catalyst Switches
- Cisco Campus ATM Solutions
- Cisco Voice Over Frame Relay, ATM, and IP
- Configuring for Selsius IP Phones
- Building Cisco Remote Access Networks
- Managing Cisco Network Security
- Cisco Enterprise Management Solutions

Nortel Networks

- Nortel Networks Accelerated Router Configuration
- Nortel Networks Advanced IP Routing
- Nortel Networks WAN Protocols
- Nortel Networks Frame Switching
- Nortel Networks Accelar 1000
- Comprehensive Configuration
- Nortel Networks Centillion Switching
- Network Management with Optivity for Windows

Oracle Training

- Introduction to Oracle8 and PL/SQL
- Oracle8 Database Administration

Custom Corporate Network Training

Train on Cutting Edge Technology

We can bring the best in skill-based training to your facility to create a real-world hands-on training experience. Global Knowledge has invested millions of dollars in network hardware and software to train our students on the same equipment they will work with on the job. Our relationships with vendors allow us to incorporate the latest equipment and platforms into your on-site labs.

Maximize Your Training Budget

Global Knowledge provides experienced instructors, comprehensive course materials, and all the networking equipment needed to deliver high quality training. You provide the students; we provide the knowledge.

Avoid Travel Expenses

On-site courses allow you to schedule technical training at your convenience, saving time, expense, and the opportunity cost of travel away from the workplace.

Discuss Confidential Topics

Private on-site training permits the open discussion of sensitive issues such as security, access, and network design. We can work with your existing network's proprietary files while demonstrating the latest technologies.

Customize Course Content

Global Knowledge can tailor your courses to include the technologies and the topics which have the greatest impact on your business. We can complement your internal training efforts or provide a total solution to your training needs.

Corporate Pass

The Corporate Pass Discount Program rewards our best network training customers with preferred pricing on public courses, discounts on multimedia training packages, and an array of career planning services.

Global Knowledge Training Lifecycle

Supporting the Dynamic and Specialized Training Requirements of Information Technology Professionals

- Define Profile
- Assess Skills
- Design Training
- Deliver Training
- Test Knowledge
- Update Profile
- Use New Skills

Global Knowledge

Global Knowledge programs are developed and presented by industry professionals with "real-world" experience. Designed to help professionals meet today's interconnectivity and interoperability challenges, most of our programs feature hands-on labs that incorporate state-of-the-art communication components and equipment.

ON-SITE TEAM TRAINING

Bring Global Knowledge's powerful training programs to your company. At Global Knowledge, we will custom design courses to meet your specific network requirements. Call (919)-461-8686 for more information.

YOUR GUARANTEE

Global Knowledge believes its courses offer the best possible training in this field. If during the first day you are not satisfied and wish to withdraw from the course, simply notify the instructor, return all course materials and receive a 100% refund.

REGISTRATION INFORMATION

In the US:

call: (888) 762-4442

fax: (919) 469-7070

visit our website:

www.globalknowledge.com

Get More at **access.globalknowledge**

The premier online information source for IT professionals

You've gained access to a Global Knowledge information portal designed to inform, educate and update visitors on issues regarding IT and IT education.

Get what you want when you want it at the
access.globalknowledge site:

Choose personalized technology articles related to *your* interests. Access a new article, review, or tutorial regularly throughout the week customized to what you want to see.

Keep learning in between Global courses by taking advantage of chat sessions with other users or instructors. Get the tips, tricks and advice that you need today!

Make your point in the Access.Globalknowledge community with threaded discussion groups related to technologies and certification.

Get instant course information at your fingertips. Customized course calendars showing you the courses you want when and where you want them.

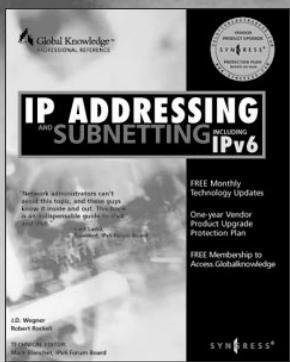
Get the resources you need with online tools, trivia, skills assessment and more!

All this and more is available now on the web at
access.globalknowledge. VISIT TODAY!



<http://access.globalknowledge.com>

SYNGRESS SOLUTIONS...



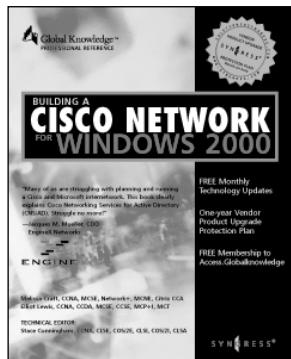
AVAILABLE NOW!
ORDER at
www.syngress.com

IP ADDRESSING AND SUBNETTING INCLUDING IPV6

Internet Protocol (IP) is the chosen protocol for the revolutionary convergence of telephony and data. The impact of a poorly designed addressing architecture on an enterprise-wide network can be catastrophic. This book provides you with complete coverage of the latest strategies, configuration scenarios, tips, techniques and warnings to successfully deploy an IP Addressing and Subnetting scheme on your network.

ISBN: 1-928994-01-6

\$59.95



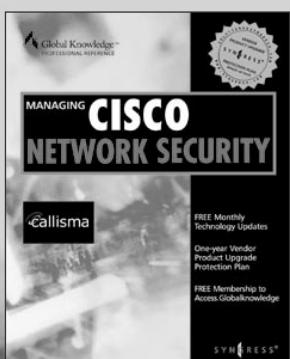
AVAILABLE NOW!
ORDER at
www.syngress.com

BUILDING A CISCO NETWORK FOR WINDOWS 2000

This book is a practical guide to designing and building Cisco-based networks running Windows 2000. It focuses on how to promote system efficiency and improve network service by using DEN to centralize network management, using the directory as a repository for all policies that govern network usage. The book also describes how to manage Cisco Internetworking Operating System (IOS) networking features and network resources in the same unified way as other enterprise resources, such as user access controls, files, and printers.

ISBN: 1-928994-00-8

\$59.95



AVAILABLE
NOVEMBER 2000
www.syngress.com

MANAGING CISCO NETWORK SECURITY

Developed for IT professionals, *Managing Cisco Network Security* details the strategies, tactics, and methods for designing, configuring, and maintaining CiscoSecure networks. It includes thorough discussions on hot topics ranging from secure VPNs and intranets, to protected LANs and WANs. It also covers the full range of CiscoSecure hardware and software solutions, including PIX Firewall, Intrusion Detection System, Access Client/Server (ACS) software, and Authentication Agent.

ISBN: 1-928994-17-2

\$59.95

solutions@syngress.com

SYNGRESS®