



Official Cert Guide

Learn, prepare, and practice for exam success



- ▶ Master **CCNA Security 640-554** exam topics
- ▶ Assess your knowledge with **chapter-opening quizzes**
- ▶ Review key concepts with **exam preparation tasks**
- ▶ Practice with **realistic exam questions** on the CD-ROM

CCNA Security 640-554

KEITH BARKER, CCIE® No. 6783
SCOTT MORRIS, CCIE No. 4713
KEVIN WALLACE, CCIE No. 7945
MICHAEL WATKINS

CCNA Security 640-554

Official Cert Guide

Keith Barker, CCIE No. 6783

Scott Morris, CCIE No. 4713

Cisco Press

800 East 96th Street

Indianapolis, IN 46240

CCNA Security 640-554 Official Cert Guide

Keith Barker, CCIE No. 6783

Scott Morris, CCIE No. 4713

Copyright© 2013 Pearson Education, Inc.

Published by:

Cisco Press

800 East 96th Street

Indianapolis, IN 46240

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without written permission from the publisher, except for the inclusion of brief quotations in a review.

Printed in the United States of America 1 2 3 4 5 6 7 8 9 0

First Printing July 2012

Library of Congress Cataloging-in-Publication data is on file.

ISBN13: 978-1-58720-446-3

ISBN: 1-58720-446-0

Warning and Disclaimer

This book is designed to provide information about selected topics for the CCNA Security 640-554 exam. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an “as is” basis. The authors, Cisco Press, and Cisco Systems, Inc. shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the author and are not necessarily those of Cisco Systems, Inc.

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers' feedback is a natural continuation of this process. If you have any comments about how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through email at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Corporate and Government Sales

Cisco Press offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales. For more information, please contact: U.S. Corporate and Government Sales 1-800-382-3419 corpsales@pearsontechgroup.com

For sales outside of the U.S., please contact: International Sales international@pearsoned.com

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc. cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Publisher: Paul Boger	Manager, Global Certification: Erik Ullanderson
Associate Publisher: Dave Dusthimer	Business Operation Manager, Cisco Press: Anand Sundaram
Executive Editor: Brett Bartow	Technical Editors: Brandon Anastasoff and David Burns
Managing Editor: Sandra Schroeder	Development Editor: Andrew Cupp
Senior Project Editor: Tonya Simpson	Editorial Assistant: Vanessa Evans
Indexer: Heather McNeill	Copy Editor: Keith Cline
Book Designer: Gary Adair	Compositor: Mark Shirar



Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV
Amsterdam, The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

CCDE, CCENT, Cisco Eos, Cisco HealthPresence, the Cisco logo, Cisco Lumin, Cisco Nexus, Cisco StadiumVision, Cisco TelePresence, Cisco WebEx, DCE, and Welcome to the Human Network are trademarks; Changing the Way We Work, Live, Play, and Learn and Cisco Store are service marks; and Access Registrar, Aironet, AsyncOS, Bringing the Meeting To You, Catalyst, CCDA, CCDP, CCIE, CCOIP, CCNA, CCNP, CCSP, CCVP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Cisco Unity, Collaboration Without Limitation, EtherFast, EtherSwitch, Event Center, Fast Step, Follow Me Browsing, FormShare, GigaDrive, HomeLink, Internet Quotient, IOS, iPhone, iQuick Study, IronPort, the IronPort logo, LightStream, Linksys, MediaTone, MeetingPlace, MeetingPlace Chime Sound, MGX, Networkers, Networking Academy, Network Registrar, PCNow, PIX, PowerPanels, ProConnect, ScriptShare, SenderBase, SMARTnet, Spectrum Expert, StackWise, The Fastest Way to Increase Your Internet Quotient, TransPath, WebEx, and the WebEx logo are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the United States and certain other countries.

All other trademarks mentioned in this document or website are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0812R)

About the Authors

Keith Barker, CCIE No. 6783 (R&S and Security), is a 27-year veteran of the networking industry. He currently works as a network engineer and trainer for Copper River IT. His past experience includes EDS, Blue Cross, Paramount Pictures, and KnowledgeNet, and he has delivered CCIE-level training over the past several years. As part of the original set of Cisco VIPs for the Cisco Learning Network, he continues to give back to the community in many ways. He is CISSP and CCSI certified, loves to teach, and keeps many of his video tutorials at <http://www.youtube.com/keith6783>. He can be reached at Keith.Barker@CopperRiverIT.com or by visiting <http://www.CopperRiverIT.com>.

Scott Morris, CCIE No. 4713 (R&S, ISP/Dial, Security, and Service Provider), has more than 25 years in the industry. He also has CCDE and myriad other certifications, including nine expert-level certifications spread over four major vendors. Having traveled the world consulting for various enterprise and service provider companies, Scott currently works at Copper River IT as the chief technologist. He, too, has delivered CCIE-level training and technology training for Cisco Systems and other technology vendors. Having spent a “past life” (early career) as a photojournalist, he brings interesting points of view from entering the IT industry from the ground up. As part of the original set of Cisco VIPs for the Cisco Learning Network, he continues to give back to the community in many ways. He can be reached at smorris@CopperRiverIT.com or by visiting <http://www.CopperRiverIT.com>.

About the Contributing Authors

Kevin Wallace, CCIE No. 7945, is a certified Cisco instructor holding multiple Cisco certifications, including CCSP, CCVP, CCNP, and CCDP. With Cisco experience dating back to 1989, Kevin has been a network design specialist for the Walt Disney World Resort, a senior technical instructor for SkillSoft/Thomson NETg/KnowledgeNet, and a network manager for Eastern Kentucky University. Kevin holds a bachelor of science degree in electrical engineering from the University of Kentucky. Kevin has also authored or co-authored multiple books for Cisco Press, including: *CCNP TSHOOT 642-832 Cert Kit*, *CCNP TSHOOT 642-832 Official Certification Guide*, *CCNP ROUTE 642-902 Cert Kit*, and *CCNP Routing and Switching Official Certification Library*, all of which target the current CCNP certification.

Michael Watkins, CCNA/CCNP/CCVP/CCSP, is a full-time senior technical instructor with SkillSoft. With 12 years of network management, training, and consulting experience, Michael has worked with organizations such as Kraft Foods, Johnson and Johnson, Raytheon, and the United States Air Force to help them implement and learn the latest network technologies. In addition to holding over more than 20 industry certifications in the areas of networking and programming technologies, Michael holds a bachelor of arts degree from Wabash College.

About the Technical Editors

Brandon Anastasoff has been a systems engineer with Cisco Systems since October 2007, when he moved from a lead network architect role in a major newspaper-publishing firm. He has spent more than 20 years in the industry, focusing on security for the past 10 and obtaining certifications inside and outside of Cisco, with his CISSP, CCSP, and most recently, the Security CCIE. After studying in the United Kingdom, Brandon took a year off in Saudi Arabia to see what a real job would be like before proceeding to college, but found the lure of an income too irresistible and never went back for the degree. Brandon had to make a choice early in his career to either follow the art of computer animation or the up-and-coming PC networking boom, and he has never regretted the decision to enter networking. He moved from early versions of Windows and Macintosh operating systems through Novell's NetWare, and then moved more into the infrastructure side, focusing mostly on Cisco LAN/WAN equipment. After Y2K, the focus became more security oriented, and Brandon became familiar with virus and Trojan analysis and forensic investigations. Today, Brandon is glad to be where he is and enjoys talking about security whenever the opportunity presents itself.

David Burns has in-depth knowledge of routing and switching technologies, network security, and mobility. He is currently a systems engineering manager for Cisco covering various U.S. service provider accounts. In July 2008, Dave joined Cisco as a lead systems engineer in a number of areas, including Femtocell, Datacenter, MTSO, and Security Architectures working for a U.S.-based SP Mobility account. He came to Cisco from a large U.S.-based cable company where he was a senior network and security design engineer. Dave held various roles before joining Cisco during his 10-plus years in the industry, working in SP operations, SP engineering, SP architecture, enterprise IT, and U.S. military intelligence communications engineering. He holds various sales and industry/Cisco technical certifications, including the CISSP, CCSP, CCDP, and two associate-level certifications. Dave recently passed the CCIE Security Written, and is currently preparing for the CCIE Security Lab. Dave is a big advocate of knowledge transfer and sharing and has a passion for network technologies, especially as related to network security. Dave has been a speaker at Cisco Live on topics such as Femtocell (IP mobility) and IPS (security). Dave earned his Bachelor of Science degree in telecommunications engineering technology from Southern Polytechnic State University, Georgia, where he currently serves as a member of the Industry Advisory Board for the Computer & Electrical Engineering Technology School.

Dedications

From Keith:

To my parents for bringing me into this world, to my children for perpetuating this world, and to my wonderful wife, Jennifer, for making my current world a better place. I love you, Jennifer.

From Scott:

The variety of inspirations and muses that affect a person's life vary over time. Every one of them affects us in different ways to help shape or drive us to where we are today. I certainly enjoy all the influences that have helped to shape (or warp) me to where I currently am. To my friend and co-author Keith, for convincing me that this was a good idea and a lot of fun to do (and gently "reminding" me of that along the way). To my dear friend Amy (who is smarter than I am) for continuing to tell me that I need to get my CCIE Voice taken care of and prodding me along now and then, motivating me to be something more than what I am currently. To my dear friend Angela, who enjoys keeping me both sane and humble by poking holes in my plans and helping me make things even better while keeping my sense of humor intact. And to my two little girls, who help keep my perspective on the world both healthy and a little off-kilter.

Acknowledgments

We want to thank many people for helping us put this book together.

The Cisco Press team: Brett Bartow, the executive editor, was the catalyst for this project, coordinating the team and ensuring that sufficient resources were available for the completion of the book. Andrew Cupp, the development editor, has been invaluable in producing a high-quality manuscript. His great suggestions and keen eye caught some technical errors and really improved the presentation of the book. We would also like to thank Tonya Simpson and the production team for their excellent work in shepherding this book through the editorial process and nipping at our heels where necessary. Many thanks go to Keith Cline for going the extra mile during the copy edit.

The technical reviewers: We want to thank the technical reviewers of this book, Brandon Anastasoff and David Burns, for their thorough, detailed review and very valuable input.

Our families: Of course, this book would not have been possible without the constant understanding and patience of our families. They have lived through the long days and nights it took to complete this project, and have always been there to poke, prod, motivate, and inspire us. We thank you all.

Each other: Last, but not least, this book is a product of work by two co-workers and colleagues, who have worked together at three different companies over the past 5 years and still manage to stay friends, which made it even more of a pleasure to complete.

Contents at a Glance

Introduction xxv

Part I Fundamentals of Network Security 3

Chapter 1 Networking Security Concepts 5

Chapter 2 Understanding Security Policies Using a Lifecycle Approach 23

Chapter 3 Building a Security Strategy 37

Part II Protecting the Network Infrastructure 47

Chapter 4 Network Foundation Protection 49

Chapter 5 Using Cisco Configuration Professional to Protect the Network Infrastructure 63

Chapter 6 Securing the Management Plane on Cisco IOS Devices 91

Chapter 7 Implementing AAA Using IOS and the ACS Server 137

Chapter 8 Securing Layer 2 Technologies 175

Chapter 9 Securing the Data Plane in IPv6 199

Part III Mitigating and Controlling Threats 219

Chapter 10 Planning a Threat Control Strategy 221

Chapter 11 Using Access Control Lists for Threat Mitigation 235

Chapter 12 Understanding Firewall Fundamentals 267

Chapter 13 Implementing Cisco IOS Zone-Based Firewalls 291

Chapter 14 Configuring Basic Firewall Policies on Cisco ASA 327

Chapter 15 Cisco IPS/IDS Fundamentals 371

Chapter 16 Implementing IOS-Based IPS 389

Part IV Using VPNs for Secure Connectivity 421

Chapter 17 Fundamentals of VPN Technology 423

Chapter 18 Fundamentals of the Public Key Infrastructure 441

Chapter 19 Fundamentals of IP Security 465

Chapter 20 Implementing IPsec Site-to-Site VPNs 495

Chapter 21 Implementing SSL VPNs Using Cisco ASA 529

Chapter 22 Final Preparation 559

Part V Appendixes 565

A Answers to the “Do I Know This Already?” Quizzes 567

B CCNA Security 640-554 (IINSv2) Exam Updates 573

Glossary 577

Index 587

CD-Only Appendixes

C Memory Tables 3

D Memory Tables Answer Key 33

Contents

Introduction xxv

Part I Fundamentals of Network Security 3

Chapter 1 Networking Security Concepts 5

- “Do I Know This Already?” Quiz 5
- Foundation Topics 8
- Understanding Network and Information Security Basics 8
 - Network Security Objectives 8
 - Confidentiality, Integrity, and Availability 8
 - Cost-Benefit Analysis of Security 9
 - Classifying Assets 10
 - Classifying Vulnerabilities 11
 - Classifying Countermeasures 12
 - What Do We Do with the Risk? 12
- Recognizing Current Network Threats 13
 - Potential Attackers 13
 - Attack Methods 14
 - Attack Vectors 15
 - Man-in-the-Middle Attacks 15
 - Other Miscellaneous Attack Methods 16
- Applying Fundamental Security Principles to Network Design 17
 - Guidelines 17
 - How It All Fits Together 19
- Exam Preparation Tasks 20
- Review All the Key Topics 20
- Complete the Tables and Lists from Memory 20
- Define Key Terms 20

Chapter 2 Understanding Security Policies Using a Lifecycle Approach 23

- “Do I Know This Already?” Quiz 23
- Foundation Topics 25
- Risk Analysis and Management 25
 - Secure Network Lifecycle 25
 - Risk Analysis Methods 25
 - Security Posture Assessment 26
 - An Approach to Risk Management 27
 - Regulatory Compliance Affecting Risk 28

	Security Policies	28
	Who, What, and Why	28
	Specific Types of Policies	29
	Standards, Procedures, and Guidelines	30
	Testing the Security Architecture	31
	Responding to an Incident on the Network	32
	Collecting Evidence	32
	Reasons for Not Being an Attacker	32
	Liability	33
	Disaster Recovery and Business Continuity Planning	33
	Exam Preparation Tasks	34
	Review All the Key Topics	34
	Complete the Tables and Lists from Memory	34
	Define Key Terms	34
Chapter 3	Building a Security Strategy	37
	“Do I Know This Already?” Quiz	37
	Foundation Topics	40
	Securing Borderless Networks	40
	The Changing Nature of Networks	40
	Logical Boundaries	40
	SecureX and Context-Aware Security	42
	Controlling and Containing Data Loss	42
	An Ounce of Prevention	42
	Secure Connectivity Using VPNs	43
	Secure Management	43
	Exam Preparation Tasks	44
	Review All the Key Topics	44
	Complete the Tables and Lists from Memory	44
	Define Key Terms	44
Part II	Protecting the Network Infrastructure	47
Chapter 4	Network Foundation Protection	49
	“Do I Know This Already?” Quiz	49
	Foundation Topics	52
	Using Network Foundation Protection to Secure Networks	52
	The Importance of the Network Infrastructure	52
	The Network Foundation Protection (NFP) Framework	52

	Interdependence	53
	Implementing NFP	53
	Understanding the Management Plane	55
	First Things First	55
	Best Practices for Securing the Management Plane	55
	Understanding the Control Plane	56
	Best Practices for Securing the Control Plane	56
	Understanding the Data Plane	57
	Best Practices for Protecting the Data Plane	59
	Additional Data Plane Protection Mechanisms	59
	Exam Preparation Tasks	60
	Review All the Key Topics	60
	Complete the Tables and Lists from Memory	60
	Define Key Terms	60
Chapter 5	Using Cisco Configuration Professional to Protect the Network Infrastructure	63
	“Do I Know This Already?” Quiz	63
	Foundation Topics	65
	Introducing Cisco Configuration Professional	65
	Understanding CCP Features and the GUI	65
	The Menu Bar	66
	The Toolbar	67
	Left Navigation Pane	68
	Content Pane	69
	Status Bar	69
	Setting Up New Devices	69
	CCP Building Blocks	70
	Communities	70
	Templates	74
	User Profiles	78
	CCP Audit Features	81
	One-Step Lockdown	84
	A Few Highlights	84
	Exam Preparation Tasks	88
	Review All the Key Topics	88
	Complete the Tables and Lists from Memory	88
	Define Key Terms	88
	Command Reference to Check Your Memory	89

Chapter 6 Securing the Management Plane on Cisco IOS Devices 91

“Do I Know This Already?” Quiz	91
Foundation Topics	94
Securing Management Traffic	94
What Is Management Traffic and the Management Plane?	94
Beyond the Blue Rollover Cable	94
Management Plane Best Practices	95
Password Recommendations	97
Using AAA to Verify Users	97
AAA Components	98
Options for Storing Usernames, Passwords, and Access Rules	98
Authorizing VPN Users	99
Router Access Authentication	100
The AAA Method List	101
Role-Based Access Control	102
Custom Privilege Levels	103
Limiting the Administrator by Assigning a View	103
Encrypted Management Protocols	103
Using Logging Files	104
Understanding NTP	105
Protecting Cisco IOS Files	106
Implement Security Measures to Protect the Management Plane	106
Implementing Strong Passwords	106
User Authentication with AAA	108
Using the CLI to Troubleshoot AAA for Cisco Routers	113
RBAC Privilege Level/Parser View	118
Implementing Parser Views	120
SSH and HTTPS	122
Implementing Logging Features	125
Configuring Syslog Support	125
SNMP Features	128
Configuring NTP	131
Securing the Cisco IOS Image and Configuration Files	133
Exam Preparation Tasks	134
Review All the Key Topics	134
Complete the Tables and Lists from Memory	135
Define Key Terms	135
Command Reference to Check Your Memory	135

Chapter 7	Implementing AAA Using IOS and the ACS Server	137
	“Do I Know This Already?” Quiz	137
	Foundation Topics	140
	Cisco Secure ACS, RADIUS, and TACACS	140
	Why Use Cisco ACS?	140
	What Platform Does ACS Run On?	141
	What Is ISE?	141
	Protocols Used Between the ACS and the Router	141
	Protocol Choices Between the ACS Server and the Client (the Router)	142
	Configuring Routers to Interoperate with an ACS Server	143
	Configuring the ACS Server to Interoperate with a Router	154
	Verifying and Troubleshooting Router-to-ACS Server Interactions	164
	Exam Preparation Tasks	171
	Review All the Key Topics	171
	Complete the Tables and Lists from Memory	171
	Define Key Terms	171
	Command Reference to Check Your Memory	172
Chapter 8	Securing Layer 2 Technologies	175
	“Do I Know This Already?” Quiz	175
	Foundation Topics	178
	VLAN and Trunking Fundamentals	178
	What Is a VLAN?	178
	Trunking with 802.1Q	180
	Following the Frame, Step by Step	181
	The Native VLAN on a Trunk	181
	So, What Do You Want to Be? (Says the Port)	182
	Inter-VLAN Routing	182
	The Challenge of Using Physical Interfaces Only	182
	Using Virtual “Sub” Interfaces	182
	Spanning-Tree Fundamentals	183
	Loops in Networks Are Usually Bad	184
	The Life of a Loop	184
	The Solution to the Layer 2 Loop	184
	STP Is Wary of New Ports	187
	Improving the Time Until Forwarding	187

Common Layer 2 Threats and How to Mitigate Them	188
Disrupt the Bottom of the Wall, and the Top Is Disrupted, Too	188
Layer 2 Best Practices	189
Do Not Allow Negotiations	190
Layer 2 Security Toolkit	190
Specific Layer 2 Mitigation for CCNA Security	191
<i>BPDU Guard</i>	191
<i>Root Guard</i>	192
<i>Port Security</i>	192
Exam Preparation Tasks	195
Review All the Key Topics	195
Complete the Tables and Lists from Memory	195
Review the Port Security Video Included with This Book	196
Define Key Terms	196
Command Reference to Check Your Memory	196

Chapter 9 Securing the Data Plane in IPv6 199

“Do I Know This Already?” Quiz	199
Foundation Topics	202
Understanding and Configuring IPv6	202
Why IPv6?	202
The Format of an IPv6 Address	203
<i>Understanding the Shortcuts</i>	205
<i>Did We Get an Extra Address?</i>	205
<i>IPv6 Address Types</i>	206
Configuring IPv6 Routing	208
Moving to IPv6	210
Developing a Security Plan for IPv6	210
Best Practices Common to Both IPv4 and IPv6	210
Threats Common to Both IPv4 and IPv6	212
The Focus on IPv6 Security	213
New Potential Risks with IPv6	213
IPv6 Best Practices	214
Exam Preparation Tasks	216
Review All the Key Topics	216
Complete the Tables and Lists from Memory	216
Define Key Terms	217
Command Reference to Check Your Memory	217

Part III	Mitigating and Controlling Threats 219
Chapter 10	Planning a Threat Control Strategy 221
	“Do I Know This Already?” Quiz 221
	Foundation Topics 224
	Designing Threat Mitigation and Containment 224
	The Opportunity for the Attacker Is Real 224
	Many Potential Risks 224
	The Biggest Risk of All 224
	Where Do We Go from Here? 225
	Securing a Network via Hardware/Software/Services 226
	Switches 227
	Routers 228
	ASA Firewall 230
	Other Systems and Services 231
	Exam Preparation Tasks 232
	Review All the Key Topics 232
	Complete the Tables and Lists from Memory 232
	Define Key Terms 232
Chapter 11	Using Access Control Lists for Threat Mitigation 235
	“Do I Know This Already?” Quiz 235
	Foundation Topics 238
	Access Control List Fundamentals and Benefits 238
	Access Lists Aren’t Just for Breakfast Anymore 238
	Stopping Malicious Traffic with an Access List 239
	What Can We Protect Against? 240
	The Logic in a Packet-Filtering ACL 241
	Standard and Extended Access Lists 242
	Line Numbers Inside an Access List 243
	Wildcard Masks 244
	Object Groups 244
	Implementing IPv4 ACLs as Packet Filters 244
	Putting the Policy in Place 244
	Monitoring the Access Lists 255
	To Log or Not to Log 257
	Implementing IPv6 ACLs as Packet Filters 259
	Exam Preparation Tasks 263
	Review All the Key Topics 263

Complete the Tables and Lists from Memory	263
Review the NAT Video Included with This Book	263
Define Key Terms	264
Command Reference to Check Your Memory	264

Chapter 12 Understanding Firewall Fundamentals 267

“Do I Know This Already?” Quiz	267
Foundation Topics	270
Firewall Concepts and Technologies	270
Firewall Technologies	270
Objectives of a Good Firewall	270
Firewall Justifications	271
The Defense-in-Depth Approach	272
Five Basic Firewall Methodologies	273
<i>Static Packet Filtering</i>	274
<i>Application Layer Gateway</i>	275
<i>Stateful Packet Filtering</i>	276
<i>Application Inspection</i>	277
<i>Transparent Firewalls</i>	277
Using Network Address Translation	278
NAT Is About Hiding or Changing the Truth About Source Addresses	278
Inside, Outside, Local, Global	279
Port Address Translation	280
NAT Options	281
Creating and Deploying Firewalls	283
Firewall Technologies	283
Firewall Design Considerations	283
Firewall Access Rules	284
Packet-Filtering Access Rule Structure	285
Firewall Rule Design Guidelines	285
Rule Implementation Consistency	286
Exam Preparation Tasks	288
Review All the Key Topics	288
Complete the Tables and Lists from Memory	288
Define Key Terms	288

Chapter 13 Implementing Cisco IOS Zone-Based Firewalls 291

“Do I Know This Already?” Quiz	291
Foundation Topics	294

Cisco IOS Zone-Based Firewall	294
How Zone-Based Firewall Operates	294
Specific Features of Zone-Based Firewalls	294
Zones and Why We Need Pairs of Them	295
Putting the Pieces Together	296
Service Policies	297
The Self Zone	300
Configuring and Verifying Cisco IOS Zone-Based Firewall	300
First Things First	301
Using CCP to Configure the Firewall	301
Verifying the Firewall	314
Verifying the Configuration from the Command Line	315
Implementing NAT in Addition to ZBF	319
Verifying Whether NAT Is Working	322
Exam Preparation Tasks	324
Review All the Key Topics	324
Review the Video Bonus Material	324
Complete the Tables and Lists from Memory	324
Define Key Terms	325
Command Reference to Check Your Memory	325
Chapter 14 Configuring Basic Firewall Policies on Cisco ASA	327
“Do I Know This Already?” Quiz	327
Foundation Topics	330
The ASA Appliance Family and Features	330
Meet the ASA Family	330
ASA Features and Services	331
ASA Firewall Fundamentals	333
ASA Security Levels	333
The Default Flow of Traffic	335
Tools to Manage the ASA	336
Initial Access	337
Packet Filtering on the ASA	337
Implementing a Packet-Filtering ACL	338
Modular Policy Framework	338
Where to Apply a Policy	339
Configuring the ASA	340
Beginning the Configuration	340
Getting to the ASDM GUI	345

Configuring the Interfaces	347
IP Addresses for Clients	355
Basic Routing to the Internet	356
NAT and PAT	357
Permitting Additional Access Through the Firewall	359
Using Packet Tracer to Verify Which Packets Are Allowed	362
Verifying the Policy of No Telnet	366
Exam Preparation Tasks	368
Review All the Key Topics	368
Complete the Tables and Lists from Memory	368
Define Key Terms	369
Command Reference to Check Your Memory	369
Chapter 15 Cisco IPS/IDS Fundamentals	371
“Do I Know This Already?” Quiz	371
Foundation Topics	374
IPS Versus IDS	374
What Sensors Do	374
Difference Between IPS and IDS	374
Sensor Platforms	376
True/False Negatives/Positives	376
Positive/Negative Terminology	377
Identifying Malicious Traffic on the Network	377
Signature-Based IPS/IDS	377
Policy-Based IPS/IDS	378
Anomaly-Based IPS/IDS	378
Reputation-Based IPS/IDS	378
When Sensors Detect Malicious Traffic	379
Controlling Which Actions the Sensors Should Take	381
Implementing Actions Based on the Risk Rating	382
IPv6 and IPS	382
Circumventing an IPS/IDS	382
Managing Signatures	384
Signature or Severity Levels	384
Monitoring and Managing Alarms and Alerts	385
Security Intelligence	385
IPS/IDS Best Practices	386
Exam Preparation Tasks	387
Review All the Key Topics	387

	Complete the Tables and Lists from Memory	387
	Define Key Terms	387
Chapter 16	Implementing IOS-Based IPS	389
	“Do I Know This Already?” Quiz	389
	Foundation Topics	392
	Understanding and Installing an IOS-Based IPS	392
	What Can IOS IPS Do?	392
	Installing the IOS IPS Feature	393
	Getting to the IPS Wizard	394
	Working with Signatures in an IOS-Based IPS	400
	Actions That May Be Taken	405
	Best Practices When Tuning IPS	412
	Managing and Monitoring IPS Alarms	412
	Exam Preparation Tasks	417
	Review All the Key Topics	417
	Complete the Tables and Lists from Memory	417
	Define Key Terms	417
	Command Reference to Check Your Memory	418
Part IV	Using VPNs for Secure Connectivity	421
Chapter 17	Fundamentals of VPN Technology	423
	“Do I Know This Already?” Quiz	423
	Foundation Topics	426
	Understanding VPNs and Why We Use Them	426
	What Is a VPN?	426
	Types of VPNs	427
	<i>Two Main Types of VPNs</i>	427
	Main Benefits of VPNs	427
	<i>Confidentiality</i>	428
	<i>Data Integrity</i>	428
	<i>Authentication</i>	430
	<i>Antireplay</i>	430
	Cryptography Basic Components	430
	Ciphers and Keys	430
	<i>Ciphers</i>	430
	<i>Keys</i>	431
	Block and Stream Ciphers	431
	<i>Block Ciphers</i>	432

- Stream Ciphers* 432
- Symmetric and Asymmetric Algorithms 432
 - Symmetric* 432
 - Asymmetric* 433
- Hashes 434
 - Hashed Message Authentication Code 434
 - Digital Signatures 435
 - Digital Signatures in Action* 435
 - Key Management 436
 - IPsec and SSL 436
 - IPsec* 436
 - SSL* 437
- Exam Preparation Tasks 439
- Review All the Key Topics 439
- Complete the Tables and Lists from Memory 439
- Define Key Terms 439

Chapter 18 Fundamentals of the Public Key Infrastructure 441

- “Do I Know This Already?” Quiz 441
- Foundation Topics 444
- Public Key Infrastructure 444
 - Public and Private Key Pairs 444
 - RSA Algorithm, the Keys, and Digital Certificates 445
 - Who Has Keys and a Digital Certificate?* 445
 - How Two Parties Exchange Public Keys* 445
 - Creating a Digital Signature* 445
 - Certificate Authorities 446
 - Root and Identity Certificates 446
 - Root Certificate* 446
 - Identity Certificate* 448
 - Using the Digital Certificates to get the Peer’s Public Key* 448
 - X.500 and X.509v3 Certificates* 449
 - Authenticating and Enrolling with the CA 450
 - Public Key Cryptography Standards 450
 - Simple Certificate Enrollment Protocol 451
 - Revoked Certificates 451
 - Uses for Digital Certificates 452
 - PKI Topologies 452
 - Single Root CA* 453

<i>Hierarchical CA with Subordinate CAs</i>	453
<i>Cross-Certifying CAs</i>	453
Putting the Pieces of PKI to Work	453
Default of the ASA	454
Viewing the Certificates in ASDM	455
Adding a New Root Certificate	455
Easier Method for Installing Both Root and Identity certificates	457
Exam Preparation Tasks	462
Review All the Key Topics	462
Complete the Tables and Lists from Memory	462
Define Key Terms	463
Command Reference to Check Your Memory	463
Chapter 19 Fundamentals of IP Security	465
“Do I Know This Already?” Quiz	465
Foundation Topics	468
IPsec Concepts, Components, and Operations	468
The Goal of IPsec	468
The Play by Play for IPsec	469
<i>Step 1: Negotiate the IKE Phase 1 Tunnel</i>	469
<i>Step 2: Run the DH Key Exchange</i>	471
<i>Step 3: Authenticate the Peer</i>	471
<i>What About the User’s Original Packet?</i>	471
<i>Leveraging What They Have Already Built</i>	471
<i>Now IPsec Can Protect the User’s Packets</i>	472
<i>Traffic Before IPsec</i>	472
<i>Traffic After IPsec</i>	473
Summary of the IPsec Story	474
Configuring and Verifying IPsec	475
Tools to Configure the Tunnels	475
Start with a Plan	475
Applying the Configuration	475
Viewing the CLI Equivalent at the Router	482
Completing and Verifying IPsec	484
Exam Preparation Tasks	491
Review All the Key Topics	491
Complete the Tables and Lists from Memory	491
Define Key Terms	492
Command Reference to Check Your Memory	492

Chapter 20 Implementing IPsec Site-to-Site VPNs 495

- “Do I Know This Already?” Quiz 495
- Foundation Topics 498
- Planning and Preparing an IPsec Site-to-Site VPN 498
 - Customer Needs 498
 - Planning IKE Phase 1 500
 - Planning IKE Phase 2 501
- Implementing and Verifying an IPsec Site-to-Site VPN 502
 - Troubleshooting IPsec Site-to-Site VPNs 511
- Exam Preparation Tasks 526
- Review All the Key Topics 526
- Complete the Tables and Lists from Memory 526
- Define Key Terms 526
- Command Reference to Check Your Memory 526

Chapter 21 Implementing SSL VPNs Using Cisco ASA 529

- “Do I Know This Already?” Quiz 529
- Foundation Topics 532
- Functions and Use of SSL for VPNs 532
 - Is IPsec Out of the Picture? 532
 - SSL and TLS Protocol Framework 533
 - The Play by Play of SSL for VPNs 534
 - SSL VPN Flavors 534
- Configuring SSL Clientless VPNs on ASA 535
 - Using the SSL VPN Wizard 536
 - Digital Certificates 537
 - Authenticating Users 538
 - Logging In 541
 - Seeing the VPN Activity from the Server 543
- Configuring the Full SSL AnyConnect VPN on the ASA 544
 - Types of SSL VPNs 545
 - Configuring Server to Support the AnyConnect Client 545
 - Groups, Connection Profiles, and Defaults 552
 - One Item with Three Different Names 553
 - Split Tunneling 554
- Exam Preparation Tasks 556
- Review All the Key Topics 556
- Complete the Tables and Lists from Memory 556
- Define Key Terms 556

Chapter 22 Final Preparation 559

Tools for Final Preparation 559

Pearson IT Certification Practice Test Engine and Questions on the
CD 559

Installing the Software from the CD 560

Activating and Downloading the Practice Exam 560

Activating Other Exams 560

Premium Edition 561

The Cisco Learning Network 561

Memory Tables 561

Chapter-Ending Review Tools 561

Videos 562

Suggested Plan for Final Review/Study 562

Using the Exam Engine 562

Summary 563

Part V Appendixes 565

A Answers to the “Do I Know This Already?” Quizzes 567

B CCNA Security 640-554 (IINSv2) Exam Updates 573

Glossary 577

Index 587

On the CD

C Memory Tables 3

D Memory Tables Answer Key 33

Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in the IOS Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italic* indicates arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets ([]) indicate an optional element.
- Braces ({ }) indicate a required choice.
- Braces within brackets ([{ }]) indicate a required choice within an optional element.

Introduction

Congratulations! If you are reading this, you have in your possession a powerful tool that can help you to

- Improve your awareness and knowledge of network security
- Increase your skill level related to the implementation of that security
- Prepare for the CCNA Security certification exam

When writing this book, it was done with you in mind, and together we will discover the critical ingredients that make up the recipe for a secure network and work through examples of how to implement these features. By focusing on both covering the objectives for the CCNA Security exam and integrating that with real-world best practices and examples, Scott Morris and I created this content with the intention of being your personal tour guides, as we take you on a journey through the world of network security.

The 640-554 *Implementing Cisco IOS Network Security (IINSv2)* exam is required for the CCNA Security certification. The prerequisite for CCNA Security is the CCNA Route/Switch certification (or any CCIE certification). The CCNA Security exam tests your knowledge of securing Cisco routers and switches and their associated networks, and this book prepares you for that exam. This book covers all the topics listed in Cisco's exam blueprint, and each chapter includes key topics and preparation tasks to assist you in mastering this information. The CD that accompanies this book also includes bonus videos to assist you in your journey toward becoming a CCNA in Security. Of course, the CD included with the printed book also includes several practice questions to help you prepare for the exam.

About the 640-554 Implementing Cisco IOS Network Security (IINSv2) Exam

Cisco's objective of the CCNA Security exam is to verify the candidate's understanding, implementation, and verification of security best practices on Cisco hardware and software. The focus points for the exam (which this book prepares you for) are as follows:

- **Cisco routers and switches**
 - Common threats, including blended threats, and how to mitigate them.
 - The lifecycle approach for a security policy
 - Understanding and implementing network foundation protection for the control, data, and management planes
 - Understanding, implementing, and verifying AAA (authentication, authorization, and accounting), including the details of TACACS+ and RADIUS
 - Understanding and implementing basic rules inside of Cisco Access Control Server (ACS) Version 5.x, including configuration of both ACS and a router for communications with each other

- Standard, extended, and named access control lists used for packet filtering and for the classification of traffic
- Understanding and implementing protection against Layer 2 attacks, including CAM table overflow attacks, and VLAN hopping
- **Cisco firewall technologies**
 - Understanding and describing the various methods for filtering implemented by firewalls, including stateful filtering. Compare and contrast the strengths and weaknesses of the various firewall technologies.
 - Understanding the methods that a firewall may use to implement Network Address Translation (NAT) and Port Address Translation (PAT).
 - Understanding, implementing, and interpreting a Zone-Based Firewall policy through Cisco Configuration Professional (CCP).
 - Understanding and describing the characteristics and defaults for interfaces, security levels, and traffic flows on the Adaptive Security Appliance (ASA).
 - Implementing and interpreting a firewall policy on an ASA through the GUI tool named the ASA Security Device Manager (ASDM).
- **Intrusion prevention systems**
 - Comparing and contrasting intrusion prevention systems (IPS) versus intrusion detection systems (IDS), including the pros and cons of each and the methods used by these systems for identifying malicious traffic
 - Describing the concepts involved with IPS included true/false positives/negatives
 - Configuring and verifying IOS-based IPS using CCP
- **VPN technologies**
 - Understanding and describing the building blocks used for virtual private networks (VPN) today, including the concepts of symmetrical, asymmetrical, encryption, hashing, Internet Key Exchange (IKE), public key infrastructure (PKI), authentication, Diffie-Hellman, certificate authorities, and so on
 - Implementing and verifying IPsec VPNs on IOS using CCP and the command-line interface (CLI)
 - Implementing and verifying Secure Sockets Layer (SSL) VPNs on the ASA firewall using ASDM

As you can see, it is an extensive list, but together we will not only address and learn each of these, but we will also have fun doing it.

You can take the exam at Pearson VUE testing centers. You can register with VUE at <http://www.vue.com/cisco/>.

640-554 IINSv2 Exam

Table I-1 lists the topics of the 640-554 IINSv2 exam and indicates the parts in the book where these topics are covered.

Table I-1 640-554 CCNA Security (IINSv2) Exam Topics

Exam Topic	Part
Common Security Threats	
Describe common security threats	I, II, III
Security and Cisco Routers	
Implement security on Cisco routers	II, III
Describe securing the control, data, and management plane	II
Describe Cisco Security Manager	II, III
Describe IPv4 to IPv6 transition	II
AAA on Cisco Devices	
Implement AAA (authentication, authorization, and accounting)	II
Describe TACACS+	II
Describe RADIUS	II
Describe AAA	II
Verify AAA functionality	II
IOS ACLs	
Describe standard, extended, and named IP IOS access control lists (ACLs) to filter packets	III
Describe considerations when building ACLs	III
Implement IP ACLs to mitigate threats in a network	III
Secure Network Management and Reporting	
Describe secure network management	II
Implement secure network management	II
Common Layer 2 Attacks	
Describe Layer 2 security using Cisco switches	II
Describe VLAN security	II
Implement VLANs and trunking	II
Implement spanning tree (securely)	II
Cisco Firewall Technologies	

Exam Topic	Part
Describe operational strengths and weaknesses of the different firewall technologies	III
Describe stateful firewalls	III
Describe the types of NAT used in firewall technologies	III
Implement zone-based policy firewall using CCP	III
Implement the Cisco Adaptive Security Appliance (ASA)	III
Implement Network Address Translation (NAT) and Port Address Translation (PAT)	III
Cisco IPS	
Describe Cisco Intrusion Prevention System (IPS) deployment considerations	III
Describe IPS technologies	III
Configure Cisco IOS IPS using CCP	III
VPN Technologies	
Describe the different methods used in cryptography	IV
Describe VPN technologies	IV
Describe the building blocks of IPsec	IV
Implement an IOS IPsec site-to-site VPN with pre-shared key authentication	IV
Verify VPN operations	IV
Implement Secure Sockets Layer (SSL) VPN using ASA Device Manager	IV

About the Implementing Cisco IOS Network Security (IINSv2) 640-554 Official Cert Guide

This book maps to the topic areas of the 640-554 exam and uses a number of features to help you understand the topics and prepare for your exam.

Objectives and Methods

This book uses several key methodologies to help you discover the exam topics for which you need more review, to help you fully understand and remember those details, and to help you prove to yourself that you have retained your knowledge of those topics. So, this book does not try to help you pass the exams only by memorization, but by truly learning and understanding the topics. This book is designed to assist you in the exam by using the following methods:

- Using a conversational style that reflects the fact that we wrote this book as if we made it just for you, as a friend, discussing the topics with you, one step at a time

- Helping you discover which exam topics you may want to invest more time studying, to really “get it”
- Providing explanations and information to fill in your knowledge gaps
- Supplying three bonus videos (on the CD) to reinforce some of the critical concepts and techniques that you have learned from in your study of this book
- Providing practice questions to assess your understanding of the topics

Book Features

To help you customize your study time using this book, the core chapters have several features that help you make the best use of your time:

- **“Do I Know This Already?” quiz:** Each chapter begins with a quiz that helps you determine how much time you need to spend studying that chapter.
- **Foundation Topics:** These are the core sections of each chapter. They explain the concepts for the topics in that chapter.
- **Exam Preparation Tasks:** After the “Foundation Topics” section of each chapter, the “Exam Preparation Tasks” section lists a series of study activities that you should do when you finish the chapter. Each chapter includes the activities that make the most sense for studying the topics in that chapter:
 - **Review All the Key Topics:** The Key Topic icon appears next to the most important items in the “Foundation Topics” section of the chapter. The “Review All the Key Topics” activity lists the key topics from the chapter, along with their page numbers. Although the contents of the entire chapter could be on the exam, you should definitely know the information listed in each key topic, so you should review these.
 - **Complete the Tables and Lists from Memory:** To help you memorize some lists of facts, many of the more important lists and tables from the chapter are included in a document on the CD. This document lists only partial information, allowing you to complete the table or list.
 - **Define Key Terms:** Although the exam is unlikely to ask a “define this term” type of question, the CCNA exams do require that you learn and know a lot of networking terminology. This section lists the most important terms from the chapter, asking you to write a short definition and compare your answer to the glossary at the end of the book.
 - **Command Reference to Check Your Memory:** Review important commands covered in the chapter.
- **CD-based practice exam:** The companion CD contains an exam engine that enables you to review practice exam questions. Use these to prepare with a sample exam and to pinpoint topics where you need more study.

How This Book Is Organized

This book contains 21 core chapters. Chapter 22 includes some preparation tips and suggestions for how to approach the exam. Each core chapter covers a subset of the topics on the CCNA Security exam. The core chapters are organized into parts. They cover the following topics:

Part I: Fundamentals of Network Security

- **Chapter 1, “Networking Security Concepts”:** This chapter covers the need for and the building blocks of network and information security, threats to our networks today, and fundamental principles of secure network design.
- **Chapter 2, “Understanding Security Policies Using a Lifecycle Approach”:** This chapter covers risk analysis and management and security policies.
- **Chapter 3, “Building a Security Strategy”:** This chapter covers securing borderless networks and controlling and containing data loss.

Part II: Protecting the Network Infrastructure

- **Chapter 4, “Network Foundation Protection”:** This chapter covers introduction to securing the network using the *network foundation protection (NFP)* approach, the management plane, the control plane, and the data plane.
- **Chapter 5, “Using Cisco Configuration Professional to Protect the Network Infrastructure”:** This chapter covers introduction to Cisco Configuration Professional, CCP features and the GUI, setting up a new devices, CCP building blocks, and CCP audit features.
- **Chapter 6, “Securing the Management Plane on Cisco IOS Devices”:** This chapter covers management traffic and how to make it more secure and the implementation of security measures to protect the management plane.
- **Chapter 7, “Implementing AAA Using IOS and the ACS Server”:** This chapter covers the role of Cisco Secure ACS and the two primary protocols used with it, RADIUS and TACACS. It also covers configuration of a router to interoperate with an ACS server and configuration of the ACS server to interoperate with a router. The chapter also covers router tools to verify and troubleshoot router-to-ACS server interactions.
- **Chapter 8, “Securing Layer 2 Technologies”:** This chapter covers VLANs and trunking fundamentals, spanning-tree fundamentals, and common Layer 2 threats and how to mitigate them.
- **Chapter 9, “Securing the Data Plane in IPv6”:** This chapter covers IPv6 (basics, configuring, and developing a security plan for IPv6).

Part III: Mitigating and Controlling Threats

- **Chapter 10, “Planning a Threat Control Strategy”:** This chapter covers the design considerations for threat mitigation and containment and the hardware, software, and services used to implement a secure network.

- **Chapter 11, “Using Access Control Lists for Threat Mitigation”:** This chapter covers the benefits and fundamentals for *access control lists (ACL)*, implementing IPv4 ACLs as packet filters, and implementing IPv6 ACLs as packet filters.
- **Chapter 12, “Understanding Firewall Fundamentals”:** This chapter covers firewall concepts and the technologies used by them, the function of *Network Address Translation (NAT)*, including its building blocks, and the guidelines and considerations for creating and deploying firewalls.
- **Chapter 13, “Implementing Cisco IOS Zone-Based Firewalls”:** This chapter covers the operational and functional components of the IOS Zone-Based Firewall and how to configure and verify the IOS Zone-Based Firewall.
- **Chapter 14, “Configuring Basic Firewall Policies on Cisco ASA”:** This chapter covers the *Adaptive Security Appliance (ASA)* family and features, ASA firewall fundamentals, and configuring the ASA.
- **Chapter 15, “Cisco IPS/IDS Fundamentals”:** This chapter compares intrusion *prevention systems (IPS)* to *intrusion detection systems (IDS)* and covers how to identify malicious traffic on the network, manage signatures, and monitor and manage alarms and alerts.
- **Chapter 16, “Implementing IOS-Based IPS”:** This chapter covers the features included in IOS-based IPS (in software) and installing the IPS feature, working with signatures in IOS-based IPS, and managing and monitoring IPS alarms.

Part IV: Using VPNs for Secure Connectivity

- **Chapter 17, “Fundamentals of VPN Technology”:** This chapter covers what VPNs are and why we use them and the basic ingredients of cryptography.
- **Chapter 18, “Fundamentals of the Public Key Infrastructure”:** This chapter covers the concepts, components, and operations of the *public key infrastructure (PKI)* and includes an example of putting the pieces of PKI to work.
- **Chapter 19, “Fundamentals of IP Security”:** This chapter covers the concepts, components, and operations of IPsec and how to configure and verify IPsec.
- **Chapter 20, “Implementing IPsec Site-to-Site VPNs”:** This chapter covers planning and preparing to implement an IPsec site-to-site VPN and implementing and verifying the IPsec site-to-site VPN.
- **Chapter 21, “Implementing SSL VPNs Using Cisco ASA”:** This chapter covers the functions and use of SSL for VPNs, configuring SSL clientless VPN on the ASA, and configuring the full SSL AnyConnect VPN on the ASA.
- **Chapter 22, “Final Preparation”:** This chapter identifies tools for final exam preparation and helps you develop an effective study plan.

Appendixes

- **Appendix A, “Answers to the ‘Do I Know This Already?’ Quizzes”:** Includes the answers to all the questions from Chapters 1 through 21.

- **Appendix B, “CCNA Security 640-554 (IINSv2) Exam Updates”:** This appendix provides instructions for finding updates to the exam and this book when and if they occur.

CD-Only Appendixes

- **Appendix C, “Memory Tables”:** This CD-only appendix contains the key tables and lists from each chapter, with some of the contents removed. You can print this appendix and, as a memory exercise, complete the tables and lists. The goal is to help you memorize facts that can be useful on the exams. This appendix is available in PDF format on the CD; it is not in the printed book.
- **Appendix D, “Memory Tables Answer Key”:** This CD-only appendix contains the answer key for the memory tables in Appendix C. This appendix is available in PDF format on the CD; it is not in the printed book.

Premium Edition eBook and Practice Test

This Cert Guide contains a special offer for a 70% discount off the companion CCNA Security 640-554 Official Cert Guide Premium Edition eBook and Practice Test. The Premium Edition combines an eBook version of the text with an enhanced Pearson IT Certification Practice Test. By purchasing the Premium Edition, you get access to two eBook versions of the text: a PDF version and an EPUB version for reading on your tablet, eReader, or mobile device. You also get an enhanced practice test that contains an additional two full practice tests of unique questions. In addition, all the practice test questions are linked to the PDF eBook, allowing you to get more detailed feedback on each question instantly. To take advantage of this offer, you will need the coupon code included on the paper in the CD sleeve. Just follow the purchasing instructions that accompany the code to download and start using your Premium Edition today!

This page intentionally left blank

CCNA Security exam topic covered in this part:

- Describe common security threats

Part I: Fundamentals of Network Security

Chapter 1: Networking Security Concepts

**Chapter 2: Understanding Security Policies Using a Lifecycle
Approach**

Chapter 3: Building a Security Strategy



This chapter covers the following subjects:

- Understanding network and information security basics
- Recognizing current network threats
- Applying fundamental security principles to network design

Networking Security Concepts

Security has been important for a long time, with an increasing focus on it over the years. When LANs connecting personal computers began to emerge back in the early 1980s, security was not goal number one, and maybe not even in the top two or three when implementing a network. It was more of an afterthought. Today, however, security for corporate networks is at or near the top of the list.

One challenge to network security is that the threats to a network constantly change. You can deal with this in a couple of ways. One way is to just stick your head in the sand and hope attackers do not harm your network. An alternative approach is to design the network with the best practices for security, and then monitor your current security and vigilantly update it.

The concept of *location of data* is becoming blurred by concepts of cloud computing and content-data networks and global load balancing. As we strive to empower employees around the world with ubiquitous access to important data, it is increasingly important to remain constantly vigilant about protecting data and the entities using it (individuals, businesses, governments, and so on).

This chapter covers the fundamental building blocks of network security (implementing and improving), an essential topic that you are ready to master now that you better understand its importance.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 1-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 1-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Understanding Network and Information Security Basics	1–5
Recognizing Current Network Threats	6–7
Applying Fundamental Security Principles to Network Design	8–10

1. Which security term refers to a person, property, or data of value to a company?
 - a. Risk
 - b. Asset
 - c. Threat prevention
 - d. Mitigation technique
2. Which asset characteristic refers to risk that results from a threat and lack of a countermeasure?
 - a. High availability
 - b. Liability
 - c. Threat prevention
 - d. Vulnerability
3. Which three items are the primary network security objectives for a company?
 - a. Revenue generation
 - b. Confidentiality
 - c. Integrity
 - d. Availability
4. Which data classification label is usually *not* found in a government organization?
 - a. Unclassified
 - b. Classified but not important
 - c. Sensitive but unclassified
 - d. For official use only
 - e. Secret
5. Which of the following represents a physical control?
 - a. Change control policy
 - b. Background checks
 - c. Electronic lock
 - d. Access lists

6. What is the primary motivation for most attacks against networks today?
 - a. Political
 - b. Financial
 - c. Theological
 - d. Curiosity
7. Which type of an attack involves lying about the source address of a frame or packet?
 - a. Man-in-the-middle attack
 - b. Denial-of-service attack
 - c. Reconnaissance attack
 - d. Spoofing attack
8. Which two approaches to security provide the most secure results on day one?
 - a. Role based
 - b. Defense in depth
 - c. Authentication
 - d. Least privilege
9. Which of the following might you find in a network that is based on a defense-in-depth security implementation? (Choose all that apply.)
 - a. Firewall
 - b. IPS
 - c. Access lists
 - d. Current patches on servers
10. In relation to production networks, which of the following are viable options when dealing with risk? (Choose all that apply.)
 - a. Ignore it
 - b. Transfer it
 - c. Mitigate it
 - d. Remove it

Foundation Topics

Understanding Network and Information Security Basics

Security is important, and the lack of it risks financial implications. This section covers some of the concepts, terms, and methodologies used in preparing for and working with secure networks.

Network Security Objectives

When considering networks, you can view them from different perspectives. For example, senior management might view the network as a business tool to facilitate the goals of the company. Network technicians (at least some) might consider their networks to be the center of the universe. End users might consider the network to be just a tool for them to get their job done, or possibly as a source for recreation.

Not all users appreciate their role in keeping data safe, and unfortunately the users of the network represent a significant vulnerability, in that they have usernames and passwords (or other credentials, such as one-time password token generators) that allow them access to the network. If a user is compromised or an unauthorized individual gains access, the security of the network may still fail as a result, even after you apply all the concepts that you learn in this book. So, an important point to remember is that the users themselves represent a security risk and that training users is a key part of a comprehensive security policy.

Confidentiality, Integrity, and Availability

Network security objectives usually involve three basic concepts:



- **Confidentiality:** There are two types of data: data in motion as it moves across the network; and data at rest, when data is sitting on storage media (server, local workstation, in the cloud, and so forth). Confidentiality means that only the authorized individuals/systems can view sensitive or classified information. This also implies that unauthorized individuals should not have any type of access to the data. Regarding data in motion, the primary way to protect that data is to encrypt it before sending it over the network. Another option you can use with encryption is to use separate networks for the transmission of confidential data. Several chapters in this book focus on these two concepts.
- **Integrity:** Integrity for data means that changes made to data are done only by authorized individuals/systems. Corruption of data is a failure to maintain data integrity.

- **Availability:** This applies to systems and to data. If the network or its data is not available to authorized users—perhaps because of a *denial-of-service (DoS)* attack or maybe because of a general network failure—the impact may be significant to companies and users who rely on that network as a business tool. The failure of a network generally equates to loss of revenue.

Perhaps thinking of these security concepts as the *CIA* might help you remember them: *confidentiality, integrity, and availability*.

Cost-Benefit Analysis of Security

Network security engineers must understand not only what they protect, but also from whom. *Risk management* is the key phrase that you will hear over and over, and although not very glamorous, it is based on specific principles and concepts related to both asset protection and security management.

What is an *asset*? It is anything that is valuable to an organization. These could be tangible items (people, computers, and so on) or intangible items (intellectual property, database information, contact lists, accounting info). Knowing the assets that you are trying to protect and their value, location, and exposure can help you more effectively determine the time and money to spend securing those assets.

A *vulnerability* is an exploitable weakness in a system or its design. Vulnerabilities can be found in protocols, operating systems, applications, and system designs. Vulnerabilities abound, with more discovered every day.

A *threat* is any potential danger to an asset. If a vulnerability exists but has not yet been exploited, the threat is *latent* and not yet realized. If someone is actively launching an attack against your system and successfully accesses something or compromises your security against an asset, the threat is *realized*. The entity that takes advantage of the vulnerability is known as the *threat agent* or *threat vector*.

A *countermeasure* is a safeguard that somehow mitigates a potential risk. It does so by either reducing or eliminating the vulnerability, or at least reduces the likelihood of the threat agent to actually exploit the risk. For example, you might have an unpatched machine on your network, making it highly vulnerable. If that machine is unplugged from the network and ceases to have any interaction with exchanging data with any other device, you have successfully mitigated all of those vulnerabilities. You have likely rendered that machine no longer an asset, though; but it is safer.

Note that thresholds apply to how we classify things. We do not spend more than the asset is worth to protect it because doing so makes no sense. For example, purchasing a used car for \$200 and then spending \$2000 on a secure garage facility so that nobody can harm the car or \$1500 on an alarm system for that car seems to be a fairly silly proposition.

If you identify the data with the greatest value/worth, you usually automatically identify where the greatest effort to secure that information will be. Keep in mind, however, that beyond a company's particular view about the value of any data, regulatory entities might also be involved (government regulations or laws, business partner agreements, contractual agreements, and so forth).

Just accepting the full risk (the all-or-nothing approach) is not really acceptable. After all, you can implement security measures to mitigate the risk. In addition, those same security devices, such as firewalls and *intrusion prevention systems (IPS)*, can protect multiple devices simultaneously, thus providing a cost benefit. So, you can reduce risk by spending money on appropriate security measures, and usually do a good job of protecting an asset. You can never completely eliminate risk, so you must find the balance.

Table 1-2 describes a number of security terms and the appliances to which they relate.



Table 1-2 *Security Terms*

Vocabulary Term	Explanation
Asset	An asset is an item that is to be protected and can include property, people, and information/data that have value to the company. This includes intangible items such as proprietary information or trade secrets and the reputation of the company. The data could include company records, client information, proprietary software, and so on.
Vulnerability	A vulnerability is an exploitable weakness of some type. That exploitation might result from a malicious attack, or it might be accidentally triggered because of a failure or weakness in the policy, implementation, or software running on the network.
Threat	This is what you are protecting against. A threat is anything that attempts to gain unauthorized access to, compromise, destroy, or damage an asset. Threats are often realized via an attack or exploit that takes advantage of an existing vulnerability. Threats today come in many varieties and spread more rapidly than ever before. Threats can also morph and be modified over time, and so you must be ever diligent to keep up with them.
Risk	Risk is the <i>potential</i> for unauthorized access to, compromise, destruction, or damage to an asset. If a threat exists, but proper countermeasures and protections are in place (it is your goal to provide this protection), the potential for the threat to be successful is reduced (thus reducing the overall risk).
Countermeasure	A countermeasure is a device or process (a safeguard) that is implemented to counteract a potential threat, which thus reduces risk.

Classifying Assets

One reason to classify an asset is so that you can take specific action, based on policy, with regard to assets in a given class. Consider, for example, *virtual private networks (VPN)*. We classify (that is, identify) the traffic that should be sent over a VPN tunnel. By classifying data and labeling it (such as labeling “top secret” data on a hard disk), we

can then focus the appropriate amount of protection or security on that data: more security for top secret data than for unclassified data, for instance. The benefit is that when new data is put into the system, you can classify it as confidential or secret and so on and it will then receive the same level of protection that you set up for that type of data. Table 1-3 lists some common asset classification categories.

Table 1-3 *Asset Classifications*

Governmental classifications	Unclassified
	Sensitive but unclassified (SBU)
	Confidential
	Secret
	Top secret
Private sector classifications	Public
	Sensitive
	Private
	Confidential
Classification criteria	Value
	Age
	Replacement cost
	Useful lifetime
Classification roles	Owner (the group ultimately responsible for the data, usually senior management of a company)
	Custodian (the group responsible for implementing the policy as dictated by the owner)
	User (those who access the data and abide by the rules of acceptable use for the data)



Classifying Vulnerabilities

Understanding the weaknesses/vulnerabilities in a system or network is a huge step toward correcting the vulnerability or putting in appropriate countermeasures to mitigate threats against those vulnerabilities. Potential network vulnerabilities abound, with many resulting from one or more of the following:

- Policy flaws
- Design errors
- Protocol weaknesses
- Misconfiguration

- Software vulnerabilities
- Human factors
- Malicious software
- Hardware vulnerabilities
- Physical access to network resources

Cisco and others have created databases that categorize threats in the public domain. The *Common Vulnerabilities and Exposures (CVE)* is a dictionary of publicly known security vulnerabilities and exposures. A quick search using your favorite search engine will lead you to their website. There is also a *National Vulnerability Database (NVD)*, which is a repository of standards-based vulnerability information; you can do a quick search for it, too. (URLs change over time, so it is better to advise you to just do a quick search and click any links that interest you.)

Classifying Countermeasures

After a company has identified its assets and considered the risks involved to that asset from a threat against a vulnerability, the company can then decide to implement countermeasures to reduce the risk of a successful attack. Common control methods used to implement countermeasures include the following:



- **Administrative:** These consist of written policies, procedures, guidelines, and standards. An example would be a written *acceptable use policy (AUP)*, agreed to by each user on the network. Another example is a change control process that needs to be followed when making changes to the network. Administrative controls could involve items such as background checks for users, as well.
- **Physical:** Physical controls are exactly what they sound like, physical security for the network servers, equipment, and infrastructure. An example is providing a locked door between users and the wiring closet on any floor (where the switches and other gear exists). Another example of a physical control is a redundant system (for instance, an uninterruptible power supply).
- **Logical:** Logical controls include passwords, firewalls, intrusion prevention systems, access lists, VPN tunnels, and so on. Logical controls are often referred to as *technical controls*.

Not all controls are created equal, and not all controls have the same purpose. Working together, however, the controls should enable you to prevent, detect, correct, and recover, all while acting as a deterrent to a threat.

What Do We Do with the Risk?

You can deal with risk in several ways, one of which is remove it. For example, by not placing a web server on the Internet, you eliminate any risk of that nonexistent web server being attacked. (This does not work very well for companies that do want the web server.)

An option for avoiding the web server altogether is to transfer the risk to someone else. For example, instead of hosting your own server on your own network, you could outsource that functionality to a service provider. The service provider could take full responsibility (the risk) for attacks that might be launched against that server of theirs and provide a service level agreement and guarantees to the customer.

So, the service provider now has the risk. How do they handle it? They do exactly what you're learning in this book: They reduce risk by implementing appropriate countermeasures. By applying the correct patches and using the correct firewalls and IPSs and other safeguards, they reduce their own risk. If risk is purely financial, insurance can be purchased that helps manage the risk. Attacks against networks today are primarily motivated by the desire for financial gain.

Another option is for a company to put up its own web server and just assume the risk. Unfortunately, if they take no security precautions or countermeasures against potential threats, the risk could be high enough to damage the company and put it out of business. Most people would agree that this is not acceptable risk.

Recognizing Current Network Threats

Threats today are constantly changing, with new ones emerging. Moving targets are often difficult to zero in on, but understanding the general nature of threats can prepare you to deal with new threats. This section covers the various network threat categories and identifies some strategies to stay ahead of those threats.

Potential Attackers

We could devote an entire book to attacks that have been launched in the past 15 minutes somewhere in the world against a network resource. Instead of trying to list the thousands of attacks that could threaten vulnerable networks, let's begin by looking at the types of adversaries that may be behind attacks:

- Terrorists
- Criminals
- Government agencies
- Nation-states
- Hackers
- Disgruntled employees
- Competitors
- Anyone with access to a computing device (sad, but true)

Different terms are used to refer to these individuals, including hacker/cracker (criminal hacker), script-kiddie, hactivists, and the list goes on. As a security practitioner, you want to “understand your enemy.” This is not to say that everyone should learn to be a hacker

or write malware, because that is really not going to help. Instead, the point is that it is good to understand the motivations and interests of the people involved in breaking all those things you seek to protect.

Some attackers seek financial gain (as mentioned previously). Others might want the notoriety that comes from attacking a well-known company or brand. Sometimes attackers throw their net wide and hurt companies both intended and unintended.

Back in the “old days,” attacks were much simpler. We had basic intrusions, war-dialing, and things like that. Viruses were fairly new. But it was all about notoriety. The Internet was in its infancy, and people sought to make names for themselves. In the late 1990s and early 2000s, we saw an increase in the number of viruses and malware, and it was about fame.

More recently, many more attacks and threats revolve around actual theft of information and damage with financial repercussions. Perhaps that is a sign of the economy, maybe it is just an evolution of who is computer literate or incited to be involved. Attackers may also be motivated by government or industrial espionage.

Attack Methods

Most attackers do not want to be discovered and so they use a variety of techniques to remain in the shadows when attempting to compromise a network, as described in Table 1-4.

Table 1-4 *Attack Methods*

Action	Description
Reconnaissance	This is the discovery process used to find information about the network. It could include scans of the network to find out which IP addresses respond, and further scans to see which ports are open. This is usually the first step taken, to discover what is on the network and to determine potential vulnerabilities.
Social engineering	<p>This is a tough one because it leverages our weakest (very likely) vulnerability in a secure network: the user. If the attacker can get the user to reveal information, it is much easier for the attacker than using some other method of reconnaissance. This could be done through email or misdirection of web pages, which results in the user clicking something that leads to the attacker gaining information. Social engineering can also be done in person or over the phone.</p> <p><i>Phishing</i> presents a link that looks like a valid trusted resource to a user. When the user clicks it, the user is prompted to disclose confidential information such as usernames/passwords.</p> <p><i>Pharming</i> is used to direct a customer’s URL from a valid resource to a malicious one that could be made to appear as the valid site to the user. From there, an attempt is made to extract confidential information from the user.</p>

Action	Description
Privilege escalation	This is the process of taking some level of access (whether authorized or not) and achieving an even greater level of access. An example is an attacker who gains user mode access to a router and then uses a brute-force attack against the router, determining what the enable secret is for privilege level 15 access.
Back doors	<p>When attackers gain access to a system, they usually want future access, as well, and they want it to be easy. A backdoor application can be installed to either allow future access or to collect information to use in further attacks.</p> <p>Many back doors are installed by users clicking something without realizing the link they click or the file they open is a threat. Back doors can also be implemented as a result of a virus or a worm (often referred to as <i>malware</i>).</p>

Attack Vectors

Be aware that attacks are not launched only from individuals outside your company. They are also launched from people and devices inside your company who have current user accounts. Perhaps the user is curious, or maybe a back door is installed on the computer that the user is on. In either case, it is important to implement a security policy that takes nothing for granted, and to be prepared to mitigate risk at several levels.

You can implement a security policy that takes nothing for granted by requiring authentication from users before their computer is allowed on the network (for which you could use 802.1x and Cisco *Access Control Server [ACS]*). This means that the workstation the user is on must go through a profiling before being allowed on the network. You could use *Network Admission Control (NAC)* or an *Identity Service Engine (ISE)* to enforce such a policy. In addition, you could use security measures at the switch port, such as port security and others. We cover many of these topics, in great detail, in later chapters.

Man-in-the-Middle Attacks

A man-in-the-middle attack results when attackers place themselves in line between two devices that are communicating, with the intent to perform reconnaissance or to manipulate the data as it moves between them. This can happen at Layer 2 or Layer 3. The main purpose is eavesdropping, so the attacker can see all the traffic.

If this happens at Layer 2, the attacker spoofs Layer 2 MAC addresses to make the devices on a LAN believe that the Layer 2 address of the attacker is the Layer 2 address of their default gateway. This is called *ARP poisoning*. Frames that are supposed to go to the default gateway are forwarded by the switch to the Layer 2 address of the attacker on the same network. As a courtesy, the attacker can forward the frames to the correct



destination so that the client will have the connectivity needed and the attacker now sees all the data between the two devices. To mitigate this risk, you could use techniques such as *Dynamic Address Resolution Protocol (ARP) Inspection (DAI)* on switches to prevent spoofing of the Layer 2 addresses.

The attacker could also implement the attack by placing a switch into the network and manipulating the *Spanning Tree Protocol (STP)* to become the root switch (and thus gain the ability to see any traffic that needs to be sent through the root switch). You can mitigate this through techniques such as root guard and other spanning-tree controls discussed later in this book.

A man-in-the-middle attack can occur at Layer 3 by a rogue router being placed on the network and then tricking the other routers into believing that the new router has a better path. This could cause network traffic to flow through the rogue router and again allow the attacker to steal network data. You can mitigate attacks such as these in various ways, including routing authentication protocols and filtering information from being advertised or learned on specific interfaces.

To safeguard data in motion, one of the best things you can do is to use encryption for the confidentiality of the data in transit. If you use plaintext protocols for management, such as Telnet or HTTP, an attacker who has implemented a man-in-the-middle attack can see the contents of your cleartext data packets, and as a result will see everything that goes across the attacker's device, including usernames and passwords that are used. Using management protocols that have encryption built in, such as SSH and HTTPS, is considered best practice, and using VPN protection for cleartext sensitive data is also considered a best practice.

Other Miscellaneous Attack Methods

No standards groups for attackers exist, so not all the attacks fit clearly in one category. In fact, some attacks fit into two or more categories at the same time. Table 1-5 describes a few additional methods attackers might use.

Table 1-5 *Additional Attack Methods*



Method	Description
Covert channel	This method uses programs or communications in unintended ways. For example, if the security policy says that web traffic is allowed but peer-to-peer messaging is not, users can attempt to tunnel their peer-to-peer traffic inside of HTTP traffic. An attacker may use a similar technique to hide traffic by tunneling it inside of some other allowed protocol to avoid detection. An example of this is a backdoor application collecting keystroke information from the workstation and then slowly sending it out disguised as <i>Internet Control Message Protocol (ICMP)</i> . This is a covert channel. An overt channel is the legitimate use of a protocol, such as a user with a web browser using HTTP to access a web server.

Method	Description
Trust exploitation	If the firewall has three interfaces, and the outside interface allows all traffic to the <i>demilitarized zone (DMZ)</i> , but not to the inside network, and the DMZ allows access to the inside network from the DMZ, an attacker could leverage that by gaining access to the DMZ and using that location to launch his attacks from there to the inside network. Other trust models, if incorrectly configured, may allow unintentional access to an attacker including active directory and <i>NFS (network file system in UNIX)</i> .
Password attacks	These could be brute force, where the attacker's system attempts thousands of possible passwords looking for the right match. This is best protected against by specifying limits on how many unsuccessful authentication attempts may occur within a specified time frame. Password attacks can also be done through malware, man-in-the-middle attacks using packet sniffers, or by using key loggers.
Botnet	A botnet is a collection of infected computers that are ready to take instructions from the attacker. For example, if the attacker has the malicious backdoor software installed on 10,000 computers, from his central location he could instruct those computers to all send TCP SYN requests or ICMP echo requests repeatedly to the same destination. To add insult to injury, he could also spoof the source IP address of the request so that reply traffic is sent to yet another victim. A covert channel is generally used by the attacker to manage the individual devices that make up the botnet.
DoS and DDoS	Denial-of-service attack and distributed denial-of-service attack. An example is using a botnet to attack a target system. If an attack is launched from a single device with the intent to cause damage to an asset, the attack could be considered a DoS attempt, as opposed to a DDoS. Both types of attacks want the same result, and it just depends on how many source machines are used in the attack as to whether it is called a DoS or DDoS.

Applying Fundamental Security Principles to Network Design

This section examines the holistic approach to improve the security posture of your network before, during, and after your network implementation.

Guidelines

You want some basic principles and guidelines in place in the early stages of designing and implementing a network. Table 1-6 describes such key guidelines.

**Table 1-6** *Guidelines for Secure Network Architecture*

Guideline	Explanation
Rule of least privilege	<p>This rule states that minimal access is only provided to the required network resources, and not any more than that. An example of this is an access list applied to an interface for filtering that says deny all. Before this, specific entries could be added allowing only the bare minimum of required protocols, and only then between the correct source and destination addresses.</p>
Defense in depth	<p>This concept suggests that you have security implemented on nearly every point of your network. An example is filtering at a perimeter router, filtering again at a firewall, using IPSs to analyze traffic before it reaches your servers, and using host-based security precautions at the servers, as well. This is defense in depth. Using authentication and authorization mechanisms could also be part of a defense-in-depth approach.</p> <p>The concept behind defense in depth is that if a single system fails, it does not mean that security has completely been removed from the equation.</p>
Separation of duties	<p>By placing specific individuals into specific roles, there can be checks and balances in place regarding the implementation of the security policy. Rotating individuals into different roles periodically will also assist in verifying that vulnerabilities are being addressed, because a person who moves into a new role will be required to review the policies in place.</p>
Auditing	<p>This refers to accounting and keeping records about what is occurring on the network. Most of this can be automated through the features of <i>authentication, authorization, and accounting (AAA)</i> (covered later in this book). When events happen on the network, the records of those events can be sent to an accounting server. When the separation-of-duties approach is used, those who are making changes on the network should not have direct access to modify or delete the accounting records that are kept on the accounting server.</p>

How It All Fits Together

This book explains how to implement security products from Cisco to mitigate or reduce the amount of risk that our companies and customers face. If there were a single magic button that we could press that both allowed the functionality we need and provided adequate security at the same time, that button would be a hot seller. Unfortunately, no magic button exists. However, what we do have are solid principles and guidelines that we can use to implement security on our networks today.

As you work through the rest of this book, keep in mind the concepts of confidentiality, data integrity, and availability (remember, CIA) for every single concept discussed. For example, the section on VPNs focuses on the different types of VPNs and how to implement them. You even learn how to configure one of them. It is easy to get wrapped up in the details of how to do this or how to do that, but what you want to look at is which aspect of CIA a specific technology implements. In the case of VPNs, they protect the data with encryption, and so the concept of confidentiality applies. VPNs can also hash, which implements data integrity. If you are reading this book in hard-copy format, you might want to take three highlighters, one for each of the letters in CIA, and mark the technologies that address these exact issues as you encounter them in your reading of this book.

By keeping your mind open to the reasons for each and every technology we discuss, you can more easily build an overall fortress of security that everybody wants for mission-critical networks.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 1-7 lists these key topics.



Table 1-7 *Key Topics*

Key Topic Element	Description	Page Number
Text	Confidentiality, integrity, and availability	8
Table 1-2	Security terms	10
Table 1-3	Asset classifications	11
Text	Classifying countermeasures	12
Text	Man-in-the-middle attacks	15
Table 1-5	Additional attack methods	16
Table 1-6	Guidelines for secure network architecture	18

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables” (found on the CD), or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

asset, vulnerability, threat, risk

This page intentionally left blank



This chapter covers the following subjects:

- Risk analysis and management
- Security policies

Understanding Security Policies Using a Lifecycle Approach

A security policy means different things to different people, and so this chapter focuses on the details of where security policies come from, who is responsible for creating and implementing them, and how to use a lifecycle approach for the continuous job of designing, implementing, and managing security.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 2-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 2-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Risk Analysis and Management	1–3
Security Policies	4–8

1. Which of the following are methods of risk analysis? (Choose all that apply.)
 - a. Risk acceptance
 - b. Risk avoidance
 - c. Quantitative
 - d. Qualitative
2. What are the primary reasons for documenting the value of an asset, in combination with the vulnerabilities of that asset? (Choose all that apply.)
 - a. Identifying risk
 - b. Identifying countermeasures
 - c. Return on investment (ROI)
 - d. Vulnerability

- 3.** Which of the following drivers compel a company to implement security? (Choose all that apply.)

 - a.** Revenue generation
 - b.** Regulatory Compliance
 - c.** Liability
 - d.** Confidential data
- 4.** Which portion of an incident management process involves becoming aware that an incident occurred?

 - a.** Preparation
 - b.** Detection and analysis
 - c.** Containment and recovery
 - d.** Forensics
- 5.** Who is ultimately responsible for the data and security on the network?

 - a.** Senior management
 - b.** IT staff
 - c.** End users
 - d.** Customers
- 6.** Which policy does the senior executive team create?

 - a.** Acceptable use policy
 - b.** Unacceptable use policy
 - c.** Governing policy
 - d.** Procedure documents
- 7.** Which of the following is an example of a technical or end-user policy? (Choose all that apply.)

 - a.** Email policy
 - b.** Network policy
 - c.** Application policy
 - d.** Life insurance policy
- 8.** When a network device is no longer being used, which lifecycle elements are implemented?

 - a.** Media sanitation
 - b.** Initiation
 - c.** Disposition
 - d.** Operations

Foundation Topics

Risk Analysis and Management

This section examines some of the guiding factors of risk analysis, including regulatory compliance and the effects of risk analysis and risk management.

Secure Network Lifecycle

Understanding exactly how a security device fits into the network and how it can possibly mitigate risk is an important aspect of network security. You learn all about that in this book. What is also important to understand is that security is never completely done; new attacks and vulnerabilities continuously crop up. So, within the context of having a secure network, you can use a lifecycle approach to security that implies continuation, meaning it is never absolutely complete. Here are the five phases in the security lifecycle:



- **Initiation:** This involves preliminary risk assessments and categorizing of risk, such as with labels of low, medium, or high. These assessments and labels can assist you in prioritizing security measures by focusing on the high-risk items first.
- **Acquisition and development:** This involves a more detailed risk assessment, acquiring the products and tools needed to implement the countermeasures needed to reduce risk, and testing those countermeasures (usually on a closed network or as a pilot program) to verify their correct implementation.
- **Implementation:** This is the actual point where the rubber meets the road, where you put the countermeasures in place on the production network.
- **Operations and maintenance:** This involves monitoring and with the care and feeding of our network security devices (and incident handling when issues arise).
- **Disposition:** All things come to an end, and disposing of network gear (including sanitizing/formatting/destroying media storage devices) is part of this.

It is also interesting to note that this lifecycle has many starting points: as new assets are acquired or when a new countermeasure is brought to market.

Risk Analysis Methods

The preceding chapter discussed the interaction between assets and vulnerabilities of those assets and possible threats that result in risk. When a company has a valuable asset, where significant vulnerabilities are associated with that asset, and very few countermeasures are positioned against possible threats, the resulting risk is very high for that asset.

One way to determine the total risk and the financial impact involved with the risk is to just wait until it happens. Then after the damage has been done to both the asset and to

the reputation of the company, you can tally up the figures to determine the total risk (or the cost of that risk). That is a silly way to approach this; after all, most companies are driven by revenue and like to avoid risk. Instead of waiting to see what happens, most companies put appropriate countermeasures in place against perceived threats to protect against the vulnerabilities that otherwise would exist in their networks.

The concern then is to calculate the impact or risk value of an asset before it is compromised. Many factors can go into this, including the value of the reputation of the company that might be harmed, legal fees, investigation fees, new software and hardware, loss of business, loss of customers, and so on. There is no simple formula to calculate the exact cost of a loss, which includes all the possible variables. However, instead of just hoping for the best, and not really understanding the possible financial impact, you can make educated estimates using a couple different methods:

- **Qualitative:** In this method, the data is gathered by an individual, who likely is a subject matter expert (in this case as to the asset's value, its vulnerabilities, potential threats, and the impact or risk based on those factors).
- **Quantitative:** In this method, you use raw data, numbers, and statistics to determine the risk.

In a typical environment, we use both methods to help assess the value of an asset, vulnerabilities that exist, and potential threats and their likelihood, which are then used to determine a risk value, also referred to as a *risk score*. The benefit of doing this exercise in determining the risk is to help justify the cost of the mitigation techniques. A company does not want to spend \$20,000 on a mitigation technique that is protecting only against a risk that might have a \$2000 impact. However, they would likely purchase a \$100,000 security and mitigation solution designed to mitigate against several million dollars of risk.

Security Posture Assessment

Assessing the security of a particular asset needs to be an ongoing process as conditions change in the network. By assessing all the aspects of a company's networks, you can help identify vulnerabilities and improve your ability to implement the appropriate countermeasures. Table 2-2 describes activities involved in documenting the current security posture of a network.

Table 2-2 *Assessing the Current Security Posture of Network Devices*

Key Activity	Explanation
General security posture assessment	This provides a high-level idea about the security state of network devices, including servers, desktops, and data storage. This should involve assessing security from multiple perspectives, with the intent to identify relevant vulnerabilities.

Key Activity	Explanation
Internal assessment	Attacks are likely to come from users inside the network, as well, and the internal assessment is designed to see how well protected you are from the inside attacks.
External assessment	This is to assess the security risk associated with attacks from external devices on networks that connect to you (for example, from devices over the Internet).
Wireless assessment	Wireless assessment identifies the vulnerabilities and weaknesses associated with the wireless implementation. This includes the range of the access points that might go beyond the walls of the building and provide a potential opportunity for a threat.
Analysis and documentation	This combines the details about vulnerabilities that may exist from all the assessments completed. This report should include countermeasures and recommended solutions to mitigate the risk involved from an attack.

An Approach to Risk Management

Part of the secret of enabling good security is applying it in the correct places and the correct ways. Obviously, our critical servers and network devices deserve the most attention in mitigating potential threats because those servers have the highest risk if compromised. Here are some things to consider when considering any specific asset:



- Value of the asset
- Vulnerabilities
- Potential threats
- Compliance issues
- Business requirements

By working through a list like this, you will likely remember the various items you need to consider. However, a checklist alone does not help you know what the exact vulnerabilities are or what the exact compliance regulations are. Those two items require constant vigilance and expertise that you must update regularly. Usually senior management identifies vital compliance issues, and once they do, the management and IT staff can research and implement those specific compliance requirements.

New assets are often added to a network. If mitigation is already in place, adding a new similar device to the same network as other resources (similar to the new asset) will not require extensive analysis. For each new asset (for which you have not calculated a risk), do the following:

- Using qualitative/quantitative approaches, identify the risk (value of asset, vulnerability, potential threats = risk).

- Take action regarding the risk (which could include transferring the risk, accepting the risk, or reducing the risk using countermeasures).
- Monitor the risk. This includes verification that the countermeasures are reducing the risk and making adjustments regarding that risk if it is increasing or decreasing based on changes in the network, changes in access to data, or changes based on new types of threats.

Regulatory Compliance Affecting Risk

It is easy to overlook regulatory compliance as a risk. However, consider the impact of not complying with a government regulation and then having a compromise in your network. The risk is not only the current business practices of that company, but also now includes the government, which might have the right to completely close down the business (a huge risk, indeed). So, part of the overall plan for security and managing risk is to implement whatever regulatory compliance is required in your local community, state, or country. If you do business internationally, you will likely need to comply with government agencies beyond your own borders. A couple of examples are Sarbanes-Oxley (SOX) and Health Insurance Portability and Accountability Act (HIPAA) from the United States.

Security Policies

This section examines who is responsible for the creation of security policies, the implementation of security policies, and the lifecycle of a security policy.

Who, What, and Why

Security policies mean different things to different people. The senior management team, for example, who should be creating the overall security policy, specifies what regulations need to be met and what the overall security requirements are. (This does not mean that the senior management team knows how to implement that security; it just means they want it.) To the end user, a security policy might seem restrictive, in that it does not allow them to do anything they want on a corporate computer that is attached to the network.

Table 2-3 describes various aspects of security policies that should be in place to secure information today.

Table 2-3 *The Who, What, and Why of Security Policies*

Security Policies	Explanation
Who creates security policies?	<p>The executive senior management team is ultimately responsible for the data and the networks that carry the data for their company. From a technician's perspective, this might seem a bit odd that the senior management team is creating a security policy, but that is who specifies the overall goals of the policy. The high-level security policy is often referred to as a <i>governing policy</i>.</p> <p>It is up to the management teams and staff who have the skills to implement the appropriate controls (which include physical, logical, and administrative controls). At this level, we often use technical policies to implement the security responsibilities based on the roles the staff are filling.</p> <p>It is up to the end users to agree to and abide by the policies set forth by the company. This is referred to as an <i>end-user policy</i>, which is sometimes called <i>acceptable use policy (AUP)</i>.</p> <p>Policies may also apply to individuals outside of the company, including customers, suppliers, contractors, and so on.</p>
What is in a security policy?	<p>In a security policy, a primary aspect is risk management. In that light, it could include items such as access controls, backups, virus protection, incident handling, encryption, monitoring, password requirements, disposing of resources, inspections and reviews, personal/physical security, system-configured change process, auditing, security awareness and training, documentation, AUP (and the list goes on).</p> <p>A security policy should begin with a general overview about why the policy was written and what it covers and what it does not cover. This is often referred to as the <i>scope of the policy</i>.</p>
Why do we have security policies?	<p>Besides risk management, security policies are also used to educate users, staff, and other workers about what the policy of the company is. It can also be used to establish a baseline for which security measures must be implemented to protect assets. Without a security policy in place, the risk (which is a factor of assets that are vulnerable being attacked and resulting in a loss) is too great.</p>

Specific Types of Policies

If you look at every asset and consider the risk for each one, and then implement those items in policy, you are going to end up with a lot of policies. This is normal. Policies that are usually found in high-security networks include the following:

- **Guideline policies:** These include AUPs, audit policies, password policies, risk assessment policies, web server policies, and so on.

- **Email policies:** These include email-forwarding policies, spam policies, and so on.
- **Remote-access policies:** These include *virtual private network (VPN)* remote access, dial-in access, and minimum requirements for remote access with regard to virus scanning or patches.
- **Telephony policy:** This includes guidance about the acceptable use of telephony services related to voice/data over that media.
- **Application policies:** These include minimum security requirements that need to be included in new applications that are added to the network and restrictions about what end users are allowed to install and or run on their local computers.
- **Network policies:** These include standards for access via wireless or wired connections, and could include minimum requirements for the PCs that are connecting, such as minimum service packs, specific antivirus properties (such as current antivirus that has been updated in the past 4 days), and other network-related activity.

Standards, Procedures, and Guidelines

To ensure a secure network environment, a lot of rules must be followed. With regard to policies, sometimes it is not clear to the user (or to the person the policy applies to) what is absolutely required versus what is just a good idea. It is better to be aware of what is critical and must be done long before the opportunity for action occurs. Table 2-4 describes additional documents that you may use to enforce the security policy as a whole.

Table 2-4 *Standards, Procedures, and Guidelines*

Security Practice Vocabulary	Explanation
Standards	A standard specifies the use of specific technologies as a countermeasure. These help the IT staff be consistent in their approach to mitigating a specific risk.
Procedures	This is a detailed document about the standards and guidelines, which helps staff to implement security for the network. Using a procedural list, an implementation on the network can be done by any one of the staff, and if the procedure is followed in a consistent manner, the result will be the same each time. Having good procedures that are easily followed to implement network security correctly is an important aspect of a secure network.
Guidelines	Guidelines are simply suggestions and are not mandatory. They usually represent best practice techniques, but are not actually required to be used. If policy and procedure and standards are vague, following the guidelines provided will be a good indication of what to do to maintain and continue the spirit of the security policy. (When in doubt, check with the manager for clarification before implementing any changes outside of procedure or standards.)

Security Practice Vocabulary	Explanation
Policies	The policies themselves are high level in nature and come from the senior management team. They usually do not include the technical details about how to implement the policy. (The implementation is left up to their staff.) Ultimately, the senior executive team is responsible as the owner of the data, and is also responsible to ensure that staff implements the policies.

Testing the Security Architecture

Companies often spend quite a large amount of money implementing security even though they are not really 100 percent certain that it is actually working. This is unacceptable if you truly want to manage risk. To test your security, you can use several techniques, including the following:

- Network scanning
- Vulnerability scanning
- Password cracking
- Penetration testing
- Social engineering attempts

All these items represent literal threats to most networks today, so you should attempt to use these techniques in a nonmalicious way against your own networks from the inside and from the outside. It is also really important to make sure that policy allows for these types of internal tests to be run. (Otherwise, you might be considered the attacker, which may have negative consequences personally against you.)

You can run these tests at various levels. For instance, a network manager might want to test his network capabilities and staff response time in a known drill, or perhaps a CEO or executive group wants to test an entire staff and network's approach to handling serious attacks in an unannounced and unplanned fashion. In either case, the responses should be the same, as if a real set of threats were converging on a system. The benefit to either, of course, is that you get good reports and analysis from outside personnel on how things can be improved, as opposed to hearing about your own company on the 6 o'clock news.

When running these tests (attacks) against your network, it is important to look at the log files or any other reporting mechanisms that you currently have that should or may identify a problem. If a specific attack is not detected, and it is determined that that attack indeed is a risk to an asset on your network, the next logical thing to do is document the issue and then recommend a mitigation technique against that threat. This goes right back to the lifecycle for the secure network: the initiation, acquisition and development, implementation, operations and management, and disposition at the end of the security technique's useful lifetime.

Responding to an Incident on the Network

After you implement security measures, test them, and are now in operation with them, attacks may still occur, but hopefully your security countermeasures mitigate them. If an attack succeeds, there should be a policy in place about exactly how to handle the incident. The word *incident* in the context of this security policy means an event on the network that has negative impact (risk involved).

An incident response policy should

- Assist in the recovery of business operations, while at the same time preserving any evidence about the attack that might be needed for forensics.
- Document all possible details of the incident, including what systems were involved, when it occurred, who was involved, and any of the details that might assist in the clear documentation of what occurred.
- Prevent, if possible, future incidents similar to the one just experienced. This is yet another way to reduce future risk.

The implementation of an incident management policy really involves direction from senior management to provide the authority, preparation on behalf of the IT staff and management, the ability to detect and analyze an incident when it does occur (this could be from log files, intrusion prevention systems, and so on), the ability to contain and recover from the attack, and finally, post-incident activities such as forensics investigation.

Collecting Evidence

The attacker, if detected, would love it if all the evidence were accidentally wiped away, leaving no trace of the attack. It is your job to have policy in place, including the procedures, that relates to the collection of evidence for forensic purposes. This is not simple because so many different types of systems and hardware exist. If possible, however, if you experience an attack that involves disk storage, you should take a drive image (like a giant snapshot) before the drive is powered off, disconnected, or modified in any other way. Often, a checksum is used that can prove the image taken exactly represents the original disk immediately after the attack.

Any equipment involved should be photographed before it is disconnected or moved. Having a chain of evidence, which is used in court as a way to prove that the evidence has not been tampered with, is a critical aspect for prosecution. Without a clear chain of evidence, the evidence might not be seen as credible and as a result potentially may not be used in court.

Reasons for Not Being an Attacker

Many laws in place today are designed, by and large, to protect the common good. Stealing is stealing, regardless of your country, state, or place of birth. Stealing is punishable in most parts of the world. Many of the laws that govern computer networks are

in place to protect information, both public and private, and because they are law, they can be enforced. Breaches of criminal law incur penalties such as fines, imprisonment, or both. Civil law focuses on activities that are considered wrong but are not breaking the criminal law. Breaches of civil law usually give rise to financial liability. If an attacker launches an attack against a network that is not his own, it is possible that the attacker could be subject to both criminal and civil penalties.

A moral compass, or ethics, is an important character trait in today's world, including when working on today's networks. With regard to what you do on a network, you should act as if the information assurance officer is looking over your shoulder (she just might be). If all your interactions are done in an ethical manner, you will never have to second-guess your decisions from a moral perspective.

Liability

The objective of most companies is to generate revenue. The reason companies invest money on security is to comply with regulations imposed on them and to reduce the risk of vulnerable systems that support the company. The company also has liability (which also equates to risk) related to information they store about customers (for example, credit card information, health information, or other sensitive data). Reducing risk also reduces the potential liability that comes with that risk.

Disaster Recovery and Business Continuity Planning

Disasters do occur, and if such occurs to a data network, the results can prove devastating, both financially and functionally. Examples include tornadoes, earthquakes, large-scale physical attacks, and so on. Part of an overall policy that addresses availability should be the ability to recover from a disaster and keep the business moving. This is often referred to as *disaster recovery (DR)* and *business continuity (BC)*.

Companies that can afford no downtime might have two completely operational fault-tolerant sites in geographically different locations. If one site goes down, the other site is seamlessly used. The downside of this is that it is expensive. Other options include having sites that are ready to bring up, that could be fully functional within 24 hours or 48 hours or some other period of time.

In planning this, decisions would be made about the cost of the fault tolerance or backup site compared to the risk (financial loss) of not having a solution in place. Some of the BC factors are the *maximum tolerable downtime (MTD)*, the *recovery time objective (RTO)*, and the *recovery point objective (RPO)*. The RTO is the number of hours or days set as the objective for resuming the business process in the event of a disaster. The RPO refers to the state at which the data is being restored. For example, an RPO of 4 hours restores data to the point where it was 4 hours earlier than the incident that triggered the RPO.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 2-5 lists these key topics.



Table 2-5 *Key Topics*

Key Topic Element	Description	Page Number
Text	Secure network lifecycle	25
Text	An approach to risk management	27
Table 2-3	The who, what, and why of security policies	29

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

qualitative, quantitative, regulatory compliance

This page intentionally left blank



This chapter covers the following subjects:

- Securing borderless networks
- Controlling and containing data loss

Building a Security Strategy

In today's networks, with so many access methods, including *virtual private networks* (VPN), remote employees, partner intranets, public access, and so on), we really need to consider all these types of access while building a security strategy. One term used often is *borderless networks*, which suggests the network does not simply start at one location and end at another location, but instead provides access without physical borders. As the world looks for uninterrupted and ubiquitous access, this concept shows how one person's dream can be another person's nightmare.

This chapter examines a high-level strategy for securing this type of "borderless" network.

"Do I Know This Already?" Quiz

The "Do I Know This Already?" quiz helps you determine your level of knowledge of this chapter's topics before you begin. Table 3-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 3-1 "Do I Know This Already?" Section-to-Question Mapping

Foundation Topics Section	Questions
Securing Borderless Networks	1–3
Controlling and Containing Data Loss	4–8

1. In which single area of the borderless network would we be primarily concerned with things such as viruses and malware?
 - a. Borderless end zone
 - b. Borderless Internet
 - c. Borderless data center
 - d. Borderless bookstore

- 2.** Which of the following methods or resources enable you to qualify a device (verify the workstation meets minimum requirements) before letting the device access the network? (Choose all that apply.)

 - a.** Port security
 - b.** NAC
 - c.** ISE
 - d.** VPN
- 3.** On a physical switch, you can use technical controls for traffic flows between devices. How can you best implement a similar policy between two virtual devices that you are running logically in the data center?

 - a.** Cannot be done
 - b.** Virtual switch
 - c.** Virtual ACLs
 - d.** Route the traffic out of the virtual to a physical switch, enforce the security there, and then route the traffic back into the virtual environment
- 4.** Which of the following elements can you use as part of the Cisco SecureX architecture/strategy? (Choose all that apply.)

 - a.** IPS appliances
 - b.** AnyConnect
 - c.** ASA firewalls
 - d.** SIO
- 5.** Which concept refers to granting access based on multiple conditions, including the identity of the user, the device the user is connecting from, and how secure the workstation is the user is connecting from?

 - a.** Context awareness
 - b.** ACS
 - c.** ISE
 - d.** NAC

- 6.** How does AnyConnect provide confidentiality?
 - a.** It encrypts data on the disk, file by file.
 - b.** It encrypts the entire disc.
 - c.** It implements encryption at Layer 2.
 - d.** It implements SSL or IPsec.

- 7.** TrustSec uses which of the following to identify a specific policy that should be applied to traffic?
 - a.** Security group tag
 - b.** Group domain of interpretation
 - c.** DSCP
 - d.** IP precedence

- 8.** Which cloud-based service could you use as an early-warning system for a threat that might be coming your way via the Internet?
 - a.** SIO
 - b.** IOS
 - c.** ISO
 - d.** SOI

Foundation Topics

Securing Borderless Networks

This section covers how our networks have changed over the years and current strategies for securing the borderless networks of today.

The Changing Nature of Networks

The borders of today's networks are dissolving. We have mobile workers, customers with multiple access methods, and cloud services that blur the traditional dividing lines between network applications and functions. This is convenient for users because they do not really have to know or even care where their data is; they can just pick up their nearest computing device (tablet, smart phone, and so on) and access data. The data could be sitting on the device itself or could be retrieved dynamically through several online resources.

Companies that do not want to host their own servers or install their own applications can purchase infrastructure for these as a cloud service. The person providing those services needs to be aware that security is critical, however, and have the ability to implement the security required. Often, we are responsible for this, as we implement the networks that make up the cloud that other customers will use. Whether it is a public cloud for others to use, or a private cloud for a few, or some storm system in between, the lack of borders around our data highlights our need to be security conscious to protect our assets (data) wherever they might travel.

Even with all these new terms and architectures, the principles and rules related to security do not change. The location of our assets may change. The specific places that we might think about adding a security device or technology (as a mitigation technique) may change, but the basic rules and considerations (what to look for and what to think about) discussed in the preceding chapter have not changed at all.

If it is our own network, completely under our control, we know what to do. If it is a multifacility network involving multiple service providers and cloud providers, it just *seems* more complicated. Any complicated scenario can be broken down into simple parts. So, as we focus on individual parts and apply the same security-conscious principles as successfully implemented before, we can continue to implement and maintain security.

Logical Boundaries

In a traditional infrastructure, clients connect to the access layer of the network. The access layer devices (typically Layer 2 switches) connect to distribution layer devices

(normally using switches with both Layer 2 and Layer 3 functionality). This makes up a switch block. A switch block can be connected to other switch blocks through a core that could be high-speed Layer 2 or Layer 3. This model has worked wonderfully for many years as a framework for putting corporate networks together.

With borderless networks (with servers, clients, and resources less constrained as to connections), we can still categorize (classify into discrete areas) devices that comprise the network and apply appropriate security concepts to each particular area. To do so, you want a working knowledge of borderless network security terminology. Table 3-2 describes many of the terms that you need to know.



Table 3-2 *Borderless Network Components*

Component	Explanation
Borderless end zone	This is where devices connect to the network. It is here that we are concerned with viruses, malware, and other malicious software. Using techniques such as <i>Network Admissions Control (NAC)</i> and <i>Identity Services Engine (ISE)</i> , we can properly interrogate devices before they are allowed onto the network to verify they meet certain minimum requirements (installations of virus scanning tools, service packs, patch revision levels, and so on).
Borderless data center	This represents a cloud-driven business environment that could provide services. It is in this borderless data center where we implement firewalls such as the <i>Adaptive Security Appliance (ASA)</i> and <i>intrusion prevention systems (IPS)</i> to protect network resources there. Virtual tools can also be used inside virtual environments in the data center, such as virtual switches that can enforce policy on virtual devices that are connected to that virtual switch.
Borderless Internet	This represents the biggest IP network on the planet, which we are all familiar with. Service providers and other individuals connected to the Internet use various techniques for security, including IPSs, firewalls, and protocol inspection (all the way from Layer 2 to Layer 7 of the OSI model).
Policy management point	In a perfect environment, we would have a single point of control that could implement appropriate security measures across the entire network. <i>Cisco Security Manager (CSM)</i> is an example of one of these enterprise tools. Another example is <i>Cisco Access Control Server (ACS)</i> , which provides contextual access. For example, if you want to allow administrators full access to a router only if they are logging in from a specific location, you could enforce that with ACS and <i>authentication, authorization, accounting (AAA)</i> rules. Under that same system, administrators could also potentially gain access from other locations.

SecureX and Context-Aware Security



SecureX architecture is a strategy, so do not look for the label on any specific security product. The core elements of this architecture include the following:

- **Context awareness:** This just what it sounds like: being aware of context. For example, you might want to confirm a basic set of parameters (who users are, how they are accessing a network, the condition of the computer they are using to access the network, and so on) before giving users access. Actual tools to implement this include ISE, NAC, and AAA.
- **AnyConnect Client:** With AnyConnect Client, you can establish *Secure Sockets Layer (SSL)* or IPsec VPNs for clients. VPNs provide for confidentiality of the data in motion and the integrity of that data.
- **TrustSec:** This creates a distributed access policy enforcement mechanism, and can also use encryption to provide confidentiality. The intent is to provide and control end-to-end security, based on who, what, where, and how users are connected to the network. Endpoint systems are analyzed to verify they meet corporate security requirements. Actual tools to implement this include ISE, NAC, and AAA. If *security group tags (SGT)* are used, devices involved in forwarding the traffic can implement the appropriate security based on the tag. Data can be encrypted for confidentiality, as well.
- **Security Intelligence Operations:** *Security Intelligence Operations (SIO)* is a cloud-based service that Cisco manages. This service identifies and correlates real-time threats so that customers can leverage this information to better protect their networks. An example is learning about an attack that is propagating through the Internet before it reaches your network, thus enabling you to place additional security measures in preparation for its arrival at your perimeter.

Controlling and Containing Data Loss

This section covers Cisco tools you can use to implement/maintain confidentiality, integrity, and availability of data in networks today.

An Ounce of Prevention



It has been said that an ounce of prevention is worth a pound of cure. Although the ratio might not be exact, the concept is true. Cisco has a suite of tools that you can use for threat mitigation and monitoring, including the following:

- **ASA firewalls:** The Cisco ASA provides packet filtering, stateful filtering (all covered in the firewall section), support for IPsec remote-access and IPsec site-to-site VPNs, and VPN support for SSL remote-access users. An additional module can provide intrusion prevention services, as well.
- **Integrated Services Routers (ISR):** Building on the routing infrastructure, you can integrate additional security into the router itself using features such as zone-based firewalls and IPSs (in software or IPS modules installed through an available option slot in the chassis). Routers support VPNs, authenticated routing protocols, packet filtering, and a wide variety of other security features.

- **Intrusion prevention systems (IPS):** An IPS is implemented as a standalone appliance, or you can implement it as a module that goes into a Cisco ASA firewall or router. In addition, you can place a blade module in a 6500 series switch. An IPS, using primarily signature matching, can identify malicious traffic and prevent attacks from being forwarded into the network.
- **IronPort Email Security Appliances and IronPort Web Security Appliances (WSA):** These appliances provide granular control of email and, in the case of web traffic and WSA, can track thousands of applications and enforce security policies to protect networks against threats.
- **ScanSafe:** ScanSafe web security can dynamically categorize search engine results to prevent access to undesired sites or content, and can also look for malicious content, thus offering protection for zero-day attacks that have not been identified through traditional IPS signatures.

Secure Connectivity Using VPNs

One thing you might have noticed is that both the router and firewall support VPN services. Some management protocols, such as *Secure Shell (SSH)* and *Hypertext Transfer Protocol Secure (HTTPS)*, use encryption to protect the confidentiality of the data that is being carried over the network. Other plaintext protocols, such as Telnet and FTP, do not provide that same level of security.

Any of these protocols can be better secured by using them in combination with a VPN tunnel. The VPN tunnel offers confidentiality by encrypting the data. Popular options for VPNs include IPsec and SSL (HTTPS). You could use a site-to-site VPN to protect the confidentiality of data moving between two sites, such as a headquarters office and a branch office. You could use a remote-access VPN for a single user at a random public location, such as from a hotel room, who needs secure access back to the resources of the corporate network.

Secure Management

Anytime you access a device to gain information or manage the device itself, you should use a secured management protocol such as SSH or HTTPS. The graphical tools for managing network devices keep getting better all the time. GUIs currently built in to security products include the *ASA Adaptive Security Device Manager (ASDM)*, which can be used with the ASA firewall; *Cisco Configuration Professional (CCP)*, which can be used to manage IOS routers; and *IPS Device Manager (IDM)* and *IDM Express (IME)*, which can both be used to manage the IPS functionality on various platforms that support intrusion prevention. Some overlap exists, too. For instance, you could use either CCP or IME to manage the IPS implementation done in software on the router.

In an enterprise environment, you can purchase a separate single-console management tool for most of these security and network devices. An example is *Cisco Security Manager (CSM)*. CSM is also a GUI tool that enables you to configure, manage, and monitor IOS routers, ASA firewall appliances, IPS sensors, and Catalyst series switches. For a current list of the exact platforms supported and managed by CSM, check Cisco.com.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 3-3 lists these key topics.



Table 3-3 *Key Topics*

Key Topic Element	Description	Page Number
Table 3-2	Borderless network components	41
List	SecureX and context-aware security	42
List	An ounce of prevention	42

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

SecureX, context-aware security, ASA, IPS, AnyConnect

This page intentionally left blank

CCNA Security exam topics covered in this part:

- Describe common security threats
- Implement security on Cisco routers
- Describe securing the control, data, and management plane
- Describe Cisco Security Manager
- Describe IPv4 to IPv6 transition
- Implement AAA (authentication, authorization, and accounting)
- Describe TACACS+
- Describe RADIUS
- Describe AAA
- Verify AAA functionality
- Describe secure network management
- Implement secure network management
- Describe Layer 2 security using Cisco switches
- Describe VLAN security
- Implement VLANs and trunking
- Implement spanning tree (securely)

Part II: Protecting the Network Infrastructure

Chapter 4: Network Foundation Protection

Chapter 5: Using Cisco Configuration Professional to Protect the Network Infrastructure

Chapter 6: Securing the Management Plane on Cisco IOS Devices

Chapter 7: Implementing AAA Using IOS and the ACS Server

Chapter 8: Securing Layer 2 Technologies

Chapter 9: Securing the Data Plane in IPv6



This chapter covers the following subjects:

- Using network foundation protection to secure networks
- Understanding the management plane
- Understanding the control plane
- Understanding the data plane

Network Foundation Protection

The network infrastructure primarily consists of routers and switches and their interconnecting cables. The infrastructure has to be healthy and functional if we want to be able to deliver network services.

If we break a big problem down into smaller pieces, such as security and what an attacker might do, we can then focus on individual components and parts. By doing this, the work of implementing security becomes less daunting. That is what *network foundation protection (NFP)* is all about: breaking the infrastructure down into smaller components, and then systematically focusing on how to secure each of those components.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 4-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 4-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Using Network Foundation Protection to Secure Networks	1–3
Understanding the Management Plane	4–5
Understanding the Control Plane	6–8
Understanding the Data Plane	9–10

1. Which of the following is not a core element addressed by NFP (network foundation protection)?
 - a. Management plane
 - b. Control plane
 - c. Data plane
 - d. Executive plane

- 2.** If you add authentication to your routing protocol so that only trusted authorized routers share information, which plane in the NFP are you securing?

 - a.** Management plane
 - b.** Control plane
 - c.** Data plane
 - d.** Executive plane
- 3.** If you use authentication and authorization services to control which administrators can access which networked devices and control what they are allowed to do, which primary plane of NFP are you protecting?

 - a.** Management plane
 - b.** Control plane
 - c.** Data plane
 - d.** Executive plane
- 4.** Which of the following is not a best practice to protect the management plane? (Choose all that apply.)

 - a.** HTTP
 - b.** Telnet
 - c.** HTTPS
 - d.** SSH
- 5.** Which of the following is a way to implement role-based access control related to the management plane? (Choose all that apply.)

 - a.** Views
 - b.** AAA services
 - c.** Access lists
 - d.** IPS
- 6.** What do CoPP and CPPr have in common? (Choose all that apply.)

 - a.** They both focus on data plane protection.
 - b.** They both focus on management plane protection.
 - c.** They both focus on control plane protection.
 - d.** They both can identify traffic destined for the router that will likely require direct CPU resources to be used by the router.

7. Which type of attack can you mitigate by authenticating a routing protocol?
(Choose all that apply.)
- a. Man-in-the-middle attack
 - b. Denial-of-service attack
 - c. Reconnaissance attack
 - d. Spoofing attack
8. What is a significant difference between CoPP and CPPr?
- a. One works at Layer 3, and the other works at Layer 2.
 - b. CPPr can classify and act on more-specific traffic than CoPP.
 - c. CoPP can classify and act on more-specific traffic than CPPr.
 - d. One protects the data plane and the other protects the management plane.
9. Which of the following enables you to protect the data plane?
- a. IOS Zone-Based Firewall
 - b. IPS
 - c. Access lists
 - d. Port security
10. DHCP snooping protects which component of NFP?
- a. Management plane
 - b. Control plane
 - c. Data plane
 - d. Executive plane

Foundation Topics

Using Network Foundation Protection to Secure Networks

This section covers a strategic approach to hardening the network so that you can manage it and allow it to correctly maintain the routing tables, and most important, so that the network stays “healthy” and can forward traffic.

The Importance of the Network Infrastructure

Many pieces and parts make up a network, and even one simple component that is not working can cause a failure of the network. If a network does not work, revenue and productivity suffer. In a nutshell, if you have vulnerabilities such as weak passwords (or no passwords), software vulnerabilities, or misconfigured devices, that leaves the door open to attackers. The impact of a down network is huge; it normally affects the work force and other systems and customers that rely on that network. The NFP framework is designed to assist you to logically group functions that occur on the network and then focus on specific security measures you can take with each of these functions.

The Network Foundation Protection (NFP) Framework



For Cisco IOS routers and switches, the NFP framework is broken down into three basic planes (also called sections/areas):

- **Management plane:** This includes the protocols and traffic that an administrator uses between his workstation and the router or switch itself. An example is using a remote management protocol such as *Secure Shell (SSH)* to monitor or configure the router or switch. The management plane is listed here first because until the device is configured (which occurs in the management plane), the device will not be too functional in a network. If a failure occurs in the management plane, it may result in losing the ability to manage a network device.
- **Control plane:** This includes protocols and traffic that the network devices use on their own without direct interaction from an administrator. An example is a routing protocol. A routing protocol can dynamically learn and share routing information that the router can then use to maintain an updated routing table. If a failure occurs in the control plane, a router may lose the ability to share or correctly learn dynamic routing information, and as a result not have the routing intelligence to be able to route for the network.
- **Data plane:** This includes traffic that is being forwarded through the network (sometimes called transit traffic). An example is a user on one part of the network who is accessing a server; the data plane represents the traffic that is either being switched or forwarded by the network devices between the client and server. A

failure of some component in the data plane results in the customer's traffic not being able to be forwarded. Other times, based on policy, you might want to deny specific types of traffic on the data plane.

Interdependence

Some interdependence exists between these three planes. For example, if the control plane fails, and the router does not know how to forward traffic, this scenario impacts the data plane because the user's traffic cannot be forwarded. Another example is a failure in the management plane that might allow an attacker to configure devices and as a result could cause both a control plane and data plane failure.

Implementing NFP

You learn more about each of these three planes later in this chapter. Before that, however, Table 4-2 describes security measures you can use to protect each of the three planes.

Table 4-2 *Components of a Threat Control and Mitigation Strategy*

Plane	Security Measures	Protection Objectives
Management plane	<i>Authentication, authorization, accounting (AAA)</i> <i>Authenticated Network Time Protocol (NTP)</i> <i>Secure Shell (SSH)</i> <i>Secure Sockets Layer/Transport Layer Security (SSL/TLS)</i> Protected syslog <i>Simple Network Management Protocol Version 3 (SNMPv3)</i> Parser views	Authenticate and authorize any administrators. Protect time synchronization by using authenticated NTP. Use only encrypted remote-access protocols such as SSH for CLI and SSL/TLS for GUI tools, and use secure versions of SNMP. If plaintext tools are used (such as syslog or Telnet), they should be protected by encryption protocols such as IPsec or should be used out of band (a separate network just for management traffic). A parser "view" is a way to limit what a specific individual, based on his role, can do on the router.



Plane	Security Measures	Protection Objectives
Control plane	<p><i>Control plane policing (CoPP)</i> and <i>control plane protection (CPPr)</i></p> <p>Authenticated routing protocol updates</p>	<p>The control plane tools can be implemented to limit the damage an attacker can attempt to implement directly at the router's IP address (traffic addressed directly to the router, which the router must spend CPU resources to process).</p> <p>Routing protocol updates should be authenticated to remove the possibility of an attacker manipulating routing tables by putting a rogue router running the same routing protocol on your network. The attacker could be doing reconnaissance to learn the routes, or the attacker could be attempting to manipulate the resulting data plane by changing the routing on the network.</p>
Data plane	<p><i>Access control lists (ACL)</i></p> <p>Layer 2 controls, such as private VLANs, <i>Spanning Tree Protocol (STP)</i> guards</p> <p>IOS IPS, Zone-Based Firewall</p>	<p>ACLs, when applied as filters on interfaces, can control which traffic (transit traffic) is allowed on the data plane.</p> <p>At Layer 2, by protecting the infrastructure there, you can avoid a rogue switch from becoming the root of your spanning tree, which would affect the data plane at Layer 2.</p> <p>Firewall filtering and services can also control exactly what traffic is flowing through your network. An example is using an IOS Zone-Based Firewall to implement policy about the data plane and what is allowed.</p>

NFP, as you might have noticed is not a single feature, but rather is three components of the infrastructure, with recommendations about protecting each one using a suite of features that you can implement across your network.

A command-line utility called **auto secure** implements security measures (several in each category) across all three of the planes. You will see the equivalent of **auto secure** in *Cisco Configuration Professional (CCP)*, the GUI tool that you can use to manage router) in the next chapter.

When implementing the best practices described by NFP, does that mean your network is going to be up forever and not have any problems? Of course not. If the network is designed poorly, with no fault tolerance, for example, and a device fails (because of a mechanical or software failure or a physical problem or because cables were removed),

if you do not have the failovers in place to continue to move traffic, your data plane is going to suffer. Other factors, such as lack of change control or an administrator accidentally putting in the incorrect configuration, are, of course, ongoing potential opportunities for the network to stop functioning.

Understanding the Management Plane

This section examines what you can do to protect management access and management protocols used on the network.

First Things First

As mentioned earlier, the management plane is covered first in this discussion. After all, without a configured router (whether configured through the console port or through an IP address with a secure remote-access tool such as SSH), the network device is not much good without a working configuration that either an administrator or some other type of management system such as *Cisco Security Manager (CSM)* has put in place. (A basic Layer 2 switch with all ports in the same VLAN would be functional, but this is unlikely to be the desired configuration for that device.)

Best Practices for Securing the Management Plane

To secure the management plane, adhere to these best practices:

- Enforce password policy, including features such as maximum number of login attempts and minimum password length.
- Implement *role-based access control (RBAC)*. This concept has been around for a long time in relation to groups. By creating a group that has specific rights, and then placing users in that group, you can more easily manage and allocate administrators. With RBAC, we can create a role (like a group) and assign that role to the users who will be acting in that role. With the role comes the permissions and access. Ways to implement RBACs include using *Access Control Server (ACS)* and CLI parser views (which restrict the commands that can be issued in the specific view the administrator is in). Custom privilege level assignments are also an option to restrict what a specific user may do while operating at that custom privilege level.
- Use AAA services, and centrally manage those services on an ACS server. With AAA, a network router or switch can interact with a centralized server before allowing any access, before allowing any command to be entered, and while keeping an audit trail that identifies who has logged in and what commands they executed while there. Your policies about who can do what can be configured on the central server, and then you can configure the routers and switches to act as clients to the server as they make their requests asking whether it is okay for a specific user to log in or if it is okay for a specific user to issue a specific command.



- Keep accurate time across all network devices using secure NTP.
- Use encrypted and authenticated versions of SNMP, which includes Version 3 (and some features from Version 2).
- Control which IP addresses are allowed to initiate management sessions with the network device.
- Lock down syslog. Syslog is sent in plain text. On the infrastructure of your network, only permit this type of traffic between the network device's IP address and the destinations that the network device is configured to send the syslog messages to. In practice, not too many people are going to encrypt syslog data, although it is better to do so. Short of doing encryption, we could use an *out-of-band (OOB)* method to communicate management traffic between our network devices and the management stations. An example is a separate VLAN that user traffic never goes on to, and using that separate VLAN just for the management traffic. If management traffic is sent in-band, which means the management traffic is using the same networks (same VLANs, for instance), all management traffic needs to have encryption, either built in or have it protected by encryption (such as using IPsec).

Understanding the Control Plane

This section reviews what you can do to protect network devices in the event of attacks involving traffic directed *to* (nontransit traffic) the network device itself.

The route processor, the CPU on a router, can only do so much. So, whenever possible, the router is going to cache information about how to forward packets (transit packets going from one device on the network to some other device). By using cached information when a packet shows up that needs to be forwarded, the CPU has to expend little effort. Forwarding of traffic is the data plane, and that is what really benefits from using cached information.

So, what has that got to do with the control plane? If a packet, such as a Telnet packet, is being sent to the router's IP address, it is no longer a transit packet that can be simply forwarded by looking up information in a route cache of some type. Instead, because the packet is addressed to the router itself, the router has to spend some CPU cycles to interpret the packet, look at the application layer information, and then potentially respond. If an attacker sends thousands of packets like these to the router, or if there is a *botnet* of hundreds of thousands of devices, each configured to send these types of packets to the router, the router could be so busy just considering all these requests that it might not have enough resources to do its normal work. Control plane security is primarily guarding against attacks that might otherwise negatively impact the CPU, including routing updates (which are also processed by the CPU).

Best Practices for Securing the Control Plane

Table 4-3 describes three ways to implement security of the control plane.

Table 4-3 *Three Ways to Secure the Control Plane*

Feature	Explanation
CoPP	<p>Control plane policing. You can configure this as a filter for any traffic destined to an IP address on the router itself. For example, you can specify that management traffic, such as SSH/HTTPS/SSL and so on, can be rate-limited (policed) down to a specific level. This way, if an attack occurs that involves an excessive amount of this traffic, the excess traffic above the threshold set could simply be ignored and not have to be processed directly by the CPU. Another way to think of this is as applying <i>quality of service (QoS)</i> to the valid management traffic and policing to the bogus management traffic.</p> <p>This is applied to a logical control plane interface (not directly to any Layer 3 interface) so that the policy can be applied globally to the router.</p>
CPPr	<p>Control plane protection. This allows for a more detailed classification of traffic (more than CoPP) that is going to use the CPU for handling. The three specific subcategories that can be classified are traffic to one of the physical or logical interfaces of the router, certain data plane traffic that requires CPU intervention before forwarding (such as IP options), and <i>Cisco Express Forwarding (CEF)</i> exceptions (traffic related to network operations, such as keepalives or packets with <i>Time-To-Live (TTL)</i> mechanisms that are expiring) that have to involve the CPU.</p> <p>The benefit of CPPr is that you can rate-limit and filter this type of traffic with a more fine-toothed comb than CoPP.</p> <p>This is also applied to a logical control plane interface, so that regardless of the logical or physical interface the packets come in on, the router processor can still have the protection.</p>
Routing protocol authentication	<p>Most routing protocols support authentication. If you use authentication, a rogue router on the network will not be believed by the authorized network devices (routers). The attacker may have intended to route all the traffic through his device, or perhaps at least learn details about the routing tables and networks.</p>

Using CoPP or CPPr, you can specify which types of management traffic are acceptable at which levels. For example, you could decide and configure the router to believe that SSH is acceptable at 100 packets per second, syslog is acceptable at 200 packets per second, and so on. Traffic that exceeds the thresholds can be safely dropped if it is not from one of your specific management stations. You can specify all those details in the policy.

Understanding the Data Plane

This section covers the methods available for implementing policy related to traffic allowed *through* (transit traffic) network devices.

For the data plane, this discussion concerns traffic that is going *through* your network device rather than *to* a network device. This is traffic from a user going to a server, and the router is just acting as a forwarding device. This is the data plane. Table 4-4 describes some of the prevalent ways to control the data plane (which may be implemented on an IOS router).



Table 4-4 *Protecting the Data Plane*

Feature	Explanation
ACLs used for filtering	There are many types of ACLs and many ways to apply them for filtering. Note that an ACL can be used as a classification mechanism used in other features, such as an IOS firewall, identifying traffic for control plane protection, identifying who is allowed to connect to a vty line, where SNMP is allowed, and so on. In the discussion of protecting the data plane, we focus primarily on ACLs applied directly to interfaces for the purpose of filtering.
IOS firewall support	The firewall features on an IOS router have grown over the years. The older technology for implementing a firewall on IOS routers was called <i>context-based access control (CBAC)</i> . CBAC has been replaced with the more current Zone-Based Firewall on the IOS.
IOS IPS	IOS IPS is a software implementation of an <i>intrusion prevention system (IPS)</i> that is overlaid on top of the existing routing platform, to provide additional security. IOS IPS uses signature matches to look for malicious traffic. When an alert goes off because of a signature match, the router can prevent the packet from being forwarded, thus preventing the attack from reaching the final destination.
TCP Intercept	This tool allows the router to look at the number of half-formed sessions that are in place and intervene on behalf of the destination device. This can protect against a destination device from a SYN-flood attack that is occurring on your network. The Zone-Based Firewall on an IOS router includes this feature.
Unicast Reverse Path Forwarding	<i>Unicast Reverse Path Forwarding (uRPF)</i> can mitigate spoofed IP packets. When this feature is enabled on an interface, as packets enter that interface the router spends an extra moment considering the source address of the packet. It then considers its own routing table, and if the routing table does not agree that the interface that just received this packet is also the best egress interface to use for forwarding to the source address of the packet, it then denies the packet. This is a good way to limit IP spoofing.

Best Practices for Protecting the Data Plane

To secure the data plane, adhere to these best practices:

- Block unwanted traffic at the router. If your corporate policy does not allow TFTP traffic, just implement ACLs that deny traffic that is not allowed. You can implement ACLs inbound or outbound on any Layer 3 interface on the router. With extended ACLs, which can match based on the source and or destination address, placing the ACL closer to the source saves resources because it denies the packet before it consumes network bandwidth and before route lookups are done on a router that is filtering inbound rather than outbound. Filtering on protocols or traffic types known to be malicious is a good idea.
- Reduce the chance of *denial-of-service (DoS)* attacks. Techniques such as TCP Intercept and firewall services can reduce the risk of SYN-flood attacks
- Reduce spoofing attacks. For example, you can filter (deny) packets trying to enter your network (from the outside) that claim to have a source IP address that is from your internal network.
- Provide bandwidth management. Implementing rate-limiting on certain types of traffic can also reduce the risk of an attack (*Internet Control Message Protocol [ICMP]*, for example, which would normally be used in small quantities for legitimate traffic).
- IPS. When possible, use an IPS to inhibit the entry of malicious traffic into the network.



Additional Data Plane Protection Mechanisms

Normally, for data plane protection we think of Layer 3 and routers. Obviously, if traffic is going through a switch, a Layer 2 function is involved, as well. Layer 2 mechanisms that you can use to help protect the data plane include the following:

- Port security to protect against MAC address flooding and CAM (*content-addressable memory*) overflow attacks. When a switch has no more room in its tables for dynamically learned MAC addresses, there is the possibility of the switch not knowing the destination Layer 2 address (for the user's frames) and forwarding a frame to all devices in the same VLAN. This might give the attacker the opportunity to eavesdrop.
- *Dynamic Host Configuration Protocol (DHCP)* snooping to prevent a rogue DHCP server from handing out incorrect default gateway information and to protect a DHCP server from a starvation attack (where an attacker requests all the IP addresses available from the DHCP server so that none are available for clients who really need them).
- *Dynamic ARP inspection (DAI)* can protect against *Address Resolution Protocol (ARP)* spoofing, ARP poisoning (which is advertising incorrect IP-to-MAC address mapping information), and resulting Layer 2 man-in-the-middle attacks.
- IP source guard, when implemented on a switch, verifies that IP spoofing is not occurring by devices on that switch.



Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 4-5 lists these key topics.



Table 4-5 *Key Topics*

Key Topic Element	Description	Page Number
List	The Network Foundation Protection (NFP) framework	52
Table 4-2	Components of a threat control and mitigation strategy	53
List	Best practices for securing the management plane	55
Table 4-3	Securing the control plane	57
Table 4-4	Protecting the data plane	58
List	Best practices for protecting the data plane	59
List	Additional data plane protection mechanisms	59

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

management plane, control plane, data plane, NFP, Unicast Reverse Path Forwarding

This page intentionally left blank



This chapter covers the following subjects:

- Introducing Cisco Configuration Professional
- Understanding CCP features and the GUI
- Setting up new devices
- CCP building blocks
- CCP audit features

Using Cisco Configuration Professional to Protect the Network Infrastructure

There have been several flavors of *graphical user interface (GUI)* tools over the years, including *Security Device Manager (SDM)*, CiscoWorks, and others. This chapter introduces you to another GUI tool called *Cisco Configuration Professional (CCP)*, also sometimes shown as Cisco CP. Even if you believe that the command-line interface is the only “true” way to go, some nice features are built in to CCP such as smart wizards that can make the configuration of the Cisco router faster and less error-prone. Another benefit of CCP is that it does not take a lot of expertise to get a configuration in place on the router. Regardless of which type of tool is used to put a configuration in place, it is important to understand the actual commands that are instructing the router to behave in a secure manner. If you need another reason to be interested in CCP, here it is: It is relevant to the certification. With all those facts in mind, let’s get started.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 5-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 5-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Understanding CCP Features and the GUI	2
Setting up New Devices	3
CCP Building Blocks	1, 4, 6–7
CCP Audit Features	5

1. Which is *not* an option for opening the Manage Community window within CCP?
 - a. Click the Manage Community icon on the toolbar
 - b. Click Manage Community from the left navigation pane
 - c. Choose Manage Community from the Applications menu
 - d. Open Cisco Configuration Professional for the first time

- 2.** What information could the status bar provide?
 - a.** Version of IOS on the router
 - b.** Health status of the router
 - c.** Connectivity being used to manage the router
 - d.** Summarized community information
- 3.** Which of the following commands are required to allow CCP to communicate securely with the router? (Choose all that apply.)
 - a.** `ip http secure-server`
 - b.** `ip http server`
 - c.** `ip http authentication local`
 - d.** `ip ssh ver 2`
- 4.** Which component of CCP allows replication of the working portions of an existing configuration to save time when configuring additional devices?
 - a.** User profiles
 - b.** Templates
 - c.** Security Audit
 - d.** One-Step Lockdown
- 5.** Which element that might be fixed by the Security Audit Wizard could potentially stop spoofed packets from entering the network?
 - a.** Unicast RPF
 - b.** CDP
 - c.** Proxy ARP
 - d.** IP redirects
- 6.** Which of the following are default protocols available for use within CCP to manage a router? (Choose all that apply.)
 - a.** RDP
 - b.** HTTPS
 - c.** LDP
 - d.** Telnet
- 7.** Which tool enables you to restrict a CCP user from seeing specific configuration options within CCP when managing a router?
 - a.** Templates
 - b.** User profiles
 - c.** Security Audit
 - d.** Communities

Foundation Topics

Introducing Cisco Configuration Professional

Cisco Configuration Professional is an application that you can run from your computer. The files for the program may be local on the computer or on the flash file system of the router. CCP provides a GUI and acts as a device management tool for working with and monitoring Cisco routers. It is typically installed on the local hard drive of the Windows machine you are using to run the program. Through the use of wizards and menu systems, CCP can simplify the implementation of routing, firewalls, *intrusion prevention systems (IPS)*, *virtual private networks (VPN)*, unified communications, and many other features on an IOS router. CCP also includes monitoring functions, which can assist in troubleshooting a router that is not operating correctly. CCP also enables administrators to easily organize and manage multiple routers at a single site by grouping those routers together into what CCP calls a *device community*.



Another flavor of the CCP GUI tool is called CCP Express; it is a watered-down version of the full CCP. This version may be preinstalled on flash, and may be run as a Java applet from the computer that is connecting to the router. This version is preinstalled from the factory on some Cisco routers.

CCP does not require a separate license (other than the router license) and may be downloaded from Cisco.com with a valid CCO account.

Understanding CCP Features and the GUI

This section walks you through the interface and many of the relevant features of CCP.

The GUI is intended to make it simple to manage the features on an IOS router, as shown in Figure 5-1.

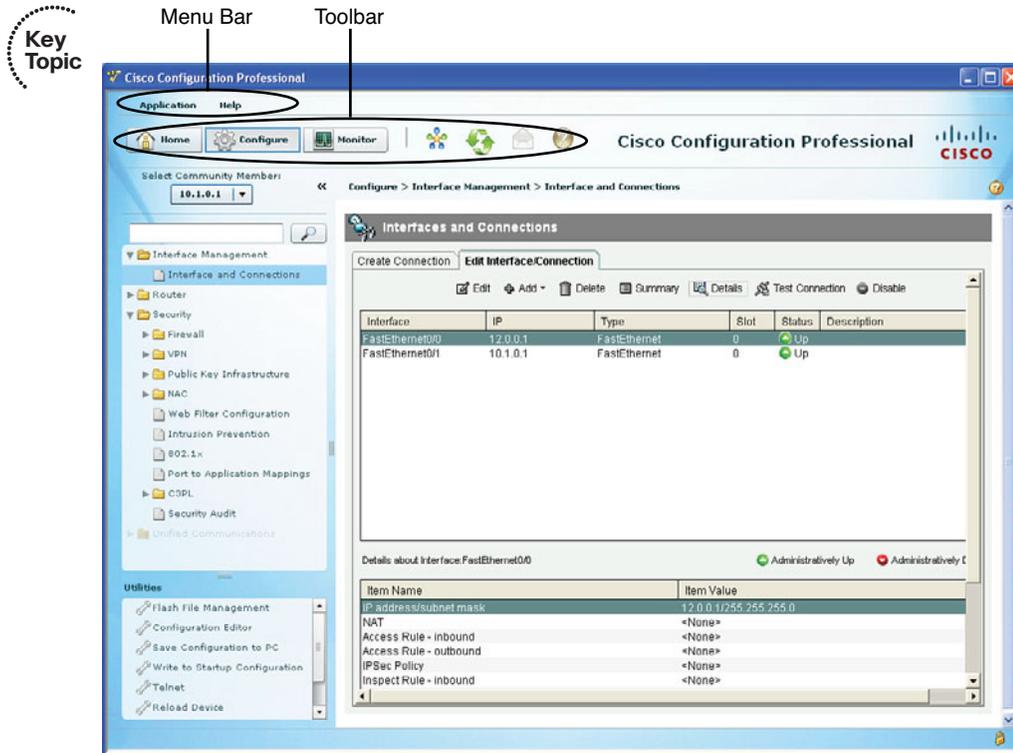


Figure 5-1 Layout of the CCP GUI

The following sections cover the major components of the interface.

The Menu Bar

This menu bar at the top of the window contains two options, Application and Help, as shown in Table 5-2.

Table 5-2 *CCP Menu Bar*

Menu Item	Options
Application	<p>The Application menu contains the following options in its drop-down menu:</p> <p>Manage Community: Enables you to create a new community or choose an existing community.</p> <p>Setup New Device: Enables you to set up a new device.</p> <p>Create User Profile: Enables you to restrict users from using all the features that are available in the left navigation pane.</p> <p>Import User Profile: Enables you to import a user profile.</p> <p>Options: Enables you to set user preferences such as log level, show community at startup, and show CLI preview parameters.</p> <p>Template: Enables you to create, edit, or apply a template.</p> <p>Work Offline: Enables you to work with CCP in offline mode.</p> <p>Exit: Exits the CCP application.</p>
Help	<p>The Help menu contains the following options:</p> <p>Help Contents: Displays the online help contents, which includes online help topics and links to screencasts</p> <p>Feedback: Displays a feedback form allowing you to provide feedback on CCP</p> <p>About: Displays information about Cisco CP, such as the version number, and allows you to view the end-user license agreement</p>

The Toolbar

The toolbar is a row of icons directly below the menu bar. The toolbar is a handy gathering place for many of the functions that you will be using with CCP. Table 5-3 describes the toolbar items.

Table 5-3 *Properties of the Toolbar*

Tool Name	Description
Home button	Click this button to display what is called the Community View page. This information summarizes the community information and allows you to add, edit, and even discover new devices. You can also use the Home button to see the device status of each device.



Tool Name	Description
Configure button	If you want to make a change to the configuration or view the existing configuration of the router, you use this Configure button to get to the correct area. From the drop-down list, you can make sure you are configuring the correct router based on its IP address, and then using features selected from the navigation pane on the left, configure the specific elements of the router you want to view or change. Not all features are available for configuration. For example, if a feature such as voice is not supported on a device, that feature is not displayed as a configurable option. Another reason that some of the options may not be configurable is because of the individual who is logged in. With <i>role-based access control (RBAC)</i> , not every user has to be given full access to configure everything. You can restrict what the administrator can see or configure by using user profiles, as covered later in this chapter.
Monitor button	This button displays the router and security features that you can monitor on a specific router. A list of items that can be selected for monitoring is presented in the left navigation pane.
Manage community icon	If you want to view or edit your existing communities, or create a new one, clicking this icon provides those options. From the Manage Community pop-up window, you can also request CCP to “discover” those routers, which means it will log in to them and read the running configuration.
Refresh icon	Clicking the refresh icon instructs CCP to reach out and request the current running configuration from the specified device. This is especially important if changes have been made at the command line of the router after CCP discovered the device. This refresh allows CCP to correctly display the configured settings, including those that were done at the command line, outside of CCP.
Provide feedback to Cisco icon	This icon opens the CCP feedback form, which you can use to send feedback about this product to Cisco Systems.
Help icon	The help icon, which looks like a question mark, opens context-sensitive help that is relevant for the active window.
Search icon	The search feature opens up a new browser window and enables you to search the help documents based on a keyword.

Left Navigation Pane

This is where you can select the item you want either to create or to manage on the IOS router. Many of the options here expand to show additional subsets.

Content Pane

This is to the right of the navigation pane, and is where most of the action occurs. Here, you enter parameters that you want to configure on the router, and see information about the router depending on which item you are looking at (which is selected by using the navigation pane on the left).

Status Bar

The status bar is located at the bottom and displays information about CCP. One example of what you might find on a status bar is the padlock icon. A secure connection (HTTPS) is indicated by the locked padlock, and an unsecure (HTTP) connection is represented by an open or unlocked padlock icon.

The primary objective in this chapter is an introduction of CCP to manage security-related features on the IOS router, such as security audits, firewalls, VPNs, and IPSs.

Setting Up New Devices

This section identifies the required basic configuration to allow CCP to communicate with the router.

There is usually quite a bit of excitement about using a new GUI such as CCP, but it is difficult to use until the computer you are sitting at can use CCP to communicate with the router. So, here are some items that you want to ensure are in place on the router so that CCP can work with that router.

Make sure that the router is powered on and reachable from the computer you are sitting at. You can verify this with a simple ping to the IP address of the router.

On a router that has CCP express preinstalled on the flash, you could use a crossover cable and connect directly from a PC Ethernet interface to the first Ethernet port on the router. You configure your PC to obtain an IP address via a *Dynamic Host Configuration Protocol (DHCP)* server, and then open a browser to 10.10.10.1. If your PC is not configured to use a DHCP server, you can assign a static address in that same 10.10.10.0/24 address space. You want to avoid using the same default IP address as the router, which is 10.10.10.1.

If you are going to manage an existing router that does not have CCP installed on the flash, you can download CCP from Cisco.com and install it on your local computer. You want to make sure the following items are in place:

- The router should be enabled to support HTTP or HTTPS.
- The authentication for HTTP/S should be set to use the local database (the running-config) on the router.
- Username with privilege level 15 rights should be created on the router.

Example 5-1 shows the configuration of those items.



Example 5-1 *Preparing the Router to Accept HTTP/HTTPS Connections from CCP*

```
! Enable HTTP services on the router to be managed and discovered (less
! secure)
R1(config)# ip http server

! Enable HTTPS services on the router to be managed and discovered (more
! secure)
R1(config)# ip http secure-server

! Create a local user account on the router with "Level 15" permissions
! (privileged
! mode), and creates an MD5 hashed password
R1(config)# username admin privilege 15 secret cisco

! Tell the router that when people connect via HTTP or HTTPS, request a
! user name and password, and use the local running-configuration (also
! called the local database) to verify the username and password supplied
! during authentication to verify if the username and password are correct,
! before allowing access
R1(config)# ip http authentication local
```

From a browser, you then connect via HTTP or HTTPS to the reachable IP address on the router's interface. To log in, you use the username and password configured in the running configuration of the router. In this example, this is the user admin with the password of cisco. If you connect via HTTPS, you are prompted as to whether you want to accept the self-signed certificate from the router, which is required if you want to continue using HTTPS. This prompting about the certificate results because of your browser not trusting the self-signed certificate the router created for HTTPS sessions. This basic connectivity paves the way for CCP, which will also use either HTTP or HTTPS connectivity to the router from your computer.

CCP Building Blocks

This section describes tools that you can use inside of CCP for efficient security policy deployment and configuration.

Communities



Before administering a router via CCP, you must first create a community that includes the router you want to manage. The first time you start CCP, it prompts you to add IP address and login information about a router that will be part of a new community.

A *community* is a group of routers that share something in common. That “something in common” could be the routers at a single geographic location or a similar function, such as that they are all running firewall services. The concept of having a community makes it easier for the administrator to work with a group of devices from one common interface.

A single community can contain a maximum of 10 devices. So, if you have 15 routers that you want to manage, you must create at least two communities to support that many devices. Perhaps two communities, one named internal routers and the other named edge routers would make sense. The actual grouping of your routers into communities is for the convenience of the administrator, and the ultimate decision on which routers to put into which of the communities is up to you. Figure 5-2 shows a community.

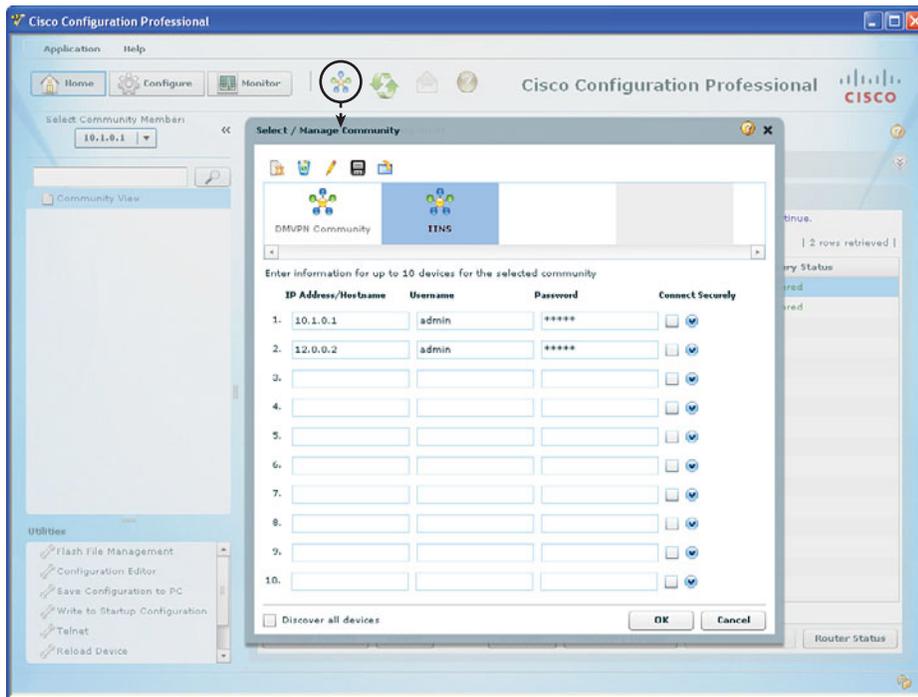


Figure 5-2 *Manage Community Pop-Up Window*

To create a community, add devices to it, and discover all the devices in a community, follow these steps:

Step 1. Use the Manage Community dialog box to create communities. The Manage Community dialog box automatically displays when you start CCP, and a community called New Community is created by default. You can change the default community name. You can also open the Manage Community dialog box in the following ways:

- From the toolbar, click the **Manage Community** icon.
- From the menu bar, choose **Application > Manage Community**.

Step 2. In the Manage Community dialog box, enter the IP address or hostname and the username and password information for the devices to configure.

If you enter the default username `cisco` and default password `cisco` (as would be the case on a brand new router that has CCP express preinstalled), the Change Default Credentials dialog box opens. For security reasons, you must change the default credentials to new credentials.

Step 3. To have CCP connect securely with the device, check the **Connect Securely** check box. To view the port information, click the **down arrow** next to the Connect Securely check box.

Step 4. To change the default port information, click the **down arrow** to the right of the device, and enter a new port value, as shown in Figure 5-3.

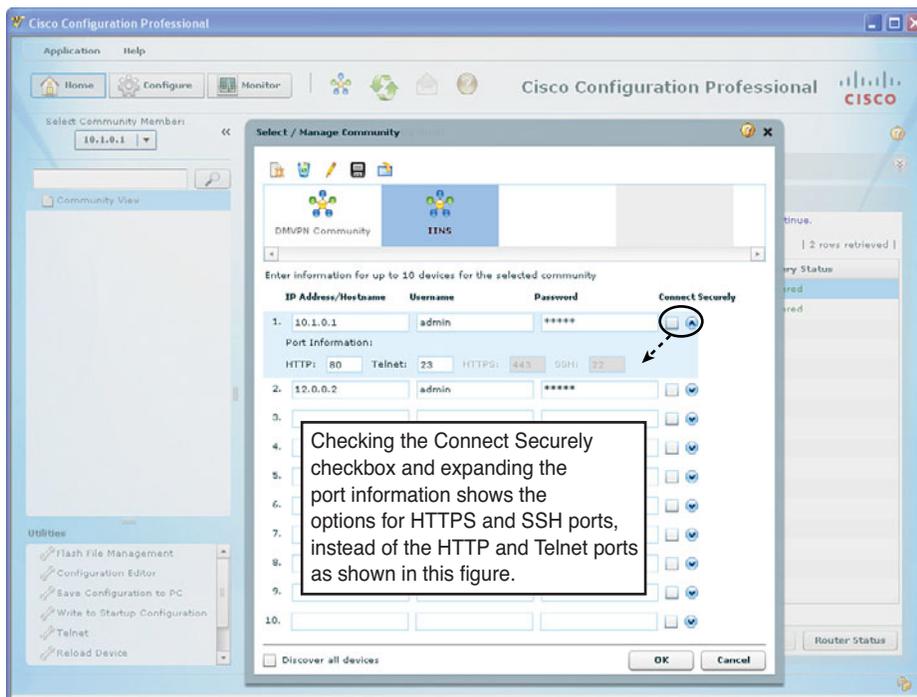


Figure 5-3 Connectivity Options, and Custom Port Choices for Connectivity to Routers

Note Make sure that CCP can access the device at the specified secure or unsecure ports that you have configured, or at the defaults depending on which options you have chosen. The defaults for unsecure HTTP and Telnet are TCP 80 and TCP 23, respectively. The defaults for the secure protocols of HTTPS and SSH are TCP 443 and 22, respectively.

- Step 5.** To have CCP discover all the devices in a community, check the **Discover All Devices** check box, as shown in Figure 5-4. You can choose to discover the devices later, from the Community View page.

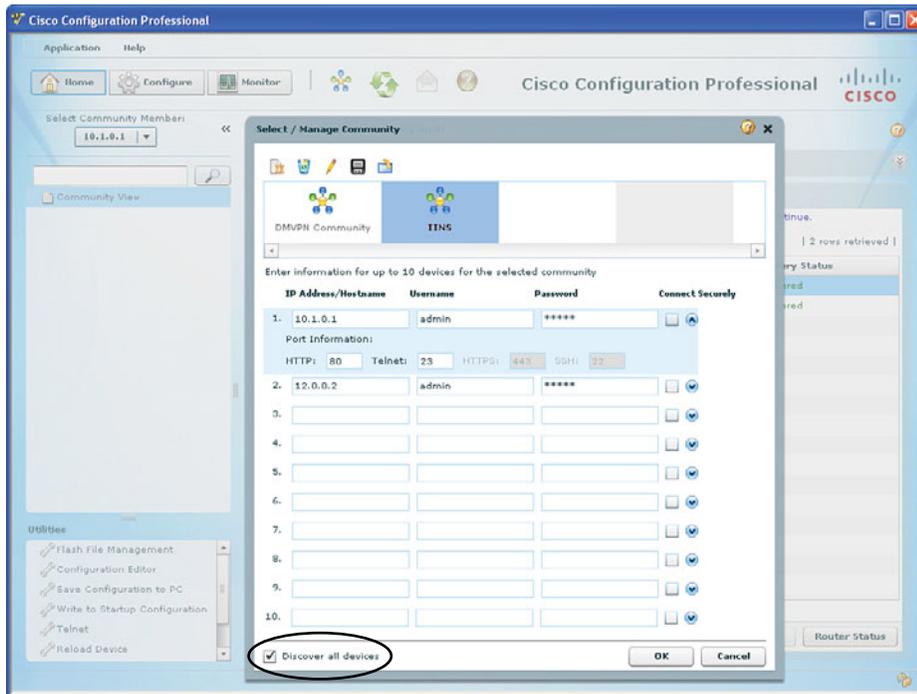


Figure 5-4 Launching the “Discover” Feature to Allow CCP to Read Router Configurations

- Step 6.** Click OK. The Community View page opens. It displays the information about the devices in the community, as shown in Figure 5-5.

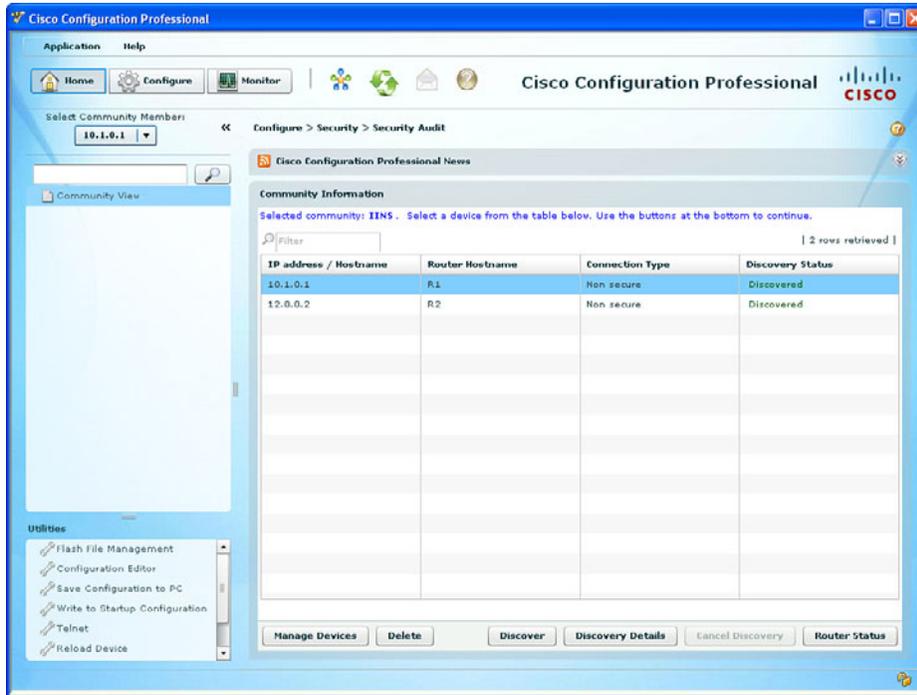


Figure 5-5 Results After Successful Discovery of Devices in the Community

Templates



If you are going to do the same type of configuration over and over again, why not do it once and then just copy/paste for the rest of them? That is where templates can come in handy.

Suppose, for instance, that you have the perfect router configuration and want to replicate that configuration to five additional routers. What are the challenges with this? First of all, you do not want all the routers to have exactly the same name, and it is also very likely that you are not going to use the same IP addresses on each device. So, the template feature enables you to identify parts of a configuration that you need to change before putting the configuration on a second or third router. The elements you are going to change, such as the hostname, the template turns into a variable, then as you apply the template to new devices, you can just swap out those variables with the new values you want to use (for example, a new hostname for the third router and a different hostname again for the fourth router). Using this strategy, we could use the same template over and over again. The process of identifying the individual components that will change from router to router and converting them to variables is done through a process called *parameterizing*.

To create and apply a template, follow these steps:

- Step 1.** Select **Application** from the menu bar, and from the drop-down list choose **Template**, and then **Create**.
- Step 2.** Choose one of your discovered routers from the **Discovered Router** drop-down list, or select a file that that can be accessed from your PC to use as the source for the template that is being created. Figure 5-6 shows an example. Click **Next** to continue.

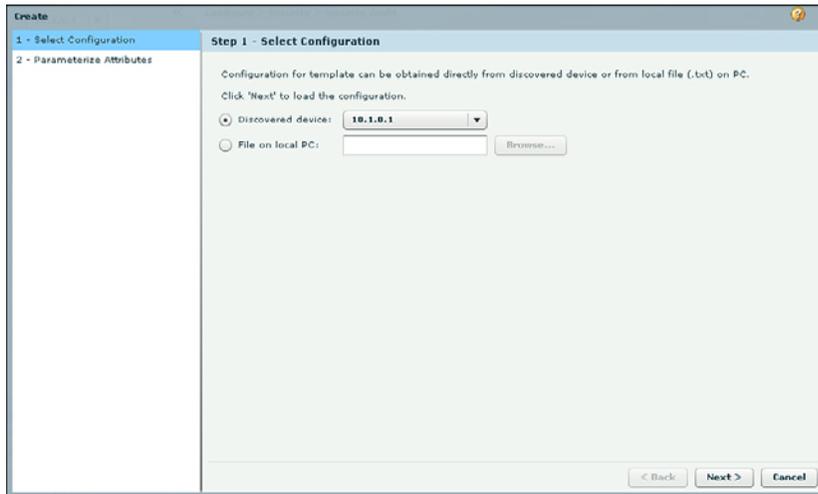


Figure 5-6 *Choosing the Source Content for the New Template*

- Step 3.** Highlight the items that need to be replaced, before placing the configuration on another router, such as hostname and IP addresses. After highlighting each item, click the **Parameterize** button. This causes those items to be identified as a variable that would need to be replaced before applying the configuration to another router. Figure 5-7 shows an example of this, where the hostname and IP addresses have been parameterized and have the double set of curly brackets. Just delete any content you do not want included in part of the new template, and then click **Finish**.

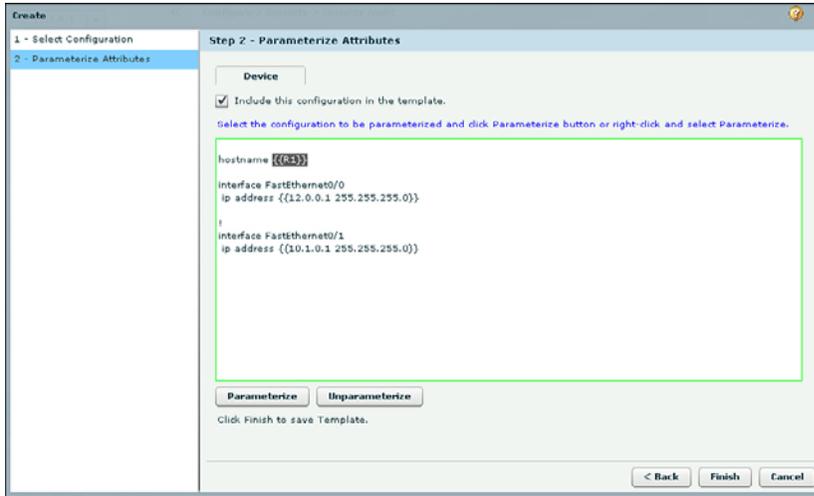


Figure 5-7 *Specifying Which Items Should be Parameterized*

- Step 4.** Save the file, using an extension of your choice on a file system that is reachable from your computer, such as the hard drive. You import this file in a later step.

Note You need to apply the template that was just created, for it to be used. That happens in the next step. It is easier to find the saved document if you save it with a .txt extension as part of step 4.

- Step 5.** Select **Application** from the menu bar, and from the drop-down list choose **Template**, and then **Apply**.
- Step 6.** Browse for and select the previously saved template file, and then click **Next**. Click the **Find Parameterized Attribute** button to search for and identify the previously identified variables and replace them with the values you want to use for the router that will receive this configuration. Then click **Next** to continue.

Note If, after clicking the Find Parameterized Attribute button, you hover over the config, a pop-up will remind you of which parameterized values are from the template, as shown in Figure 5-8.

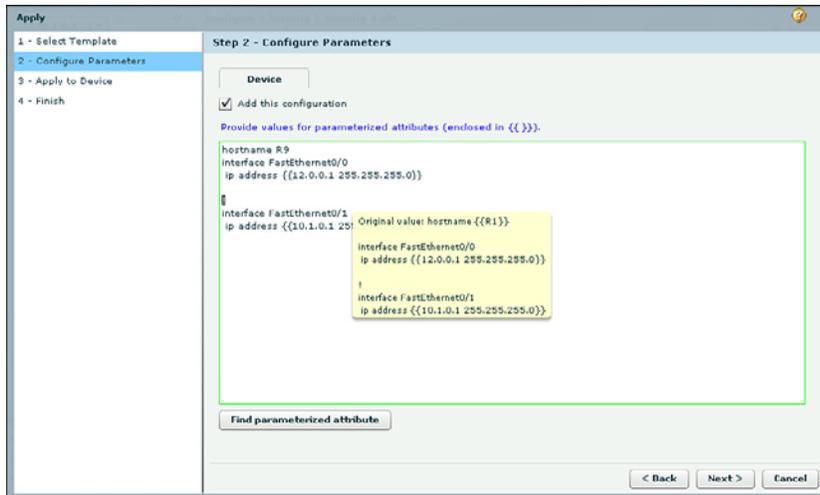


Figure 5-8 Replacing the Parameterized Items with the Values You Want for the New Router

- Step 7.** From the drop-down list, select the previously discovered router that you want to apply this new configuration to, as shown in Figure 5-9. You can select to merge the configuration with the existing router configuration, or you can choose to completely override the existing configuration of the router receiving the template. Click **Next** to continue, followed by the **Finish** button.

Note Some parameters in a router configuration do not allow for duplicates. An example is a hostname. A router can have only one hostname for itself at a time. The hostname, if part of the template, overrides a previous hostname, even if you select a “merge” option. Other items from a template, such as a unique access list that does not exist in the new router yet, can be added or “merged” with the existing configuration so that any existing and the new access list from the template will exist in the final router configuration when you merge them.

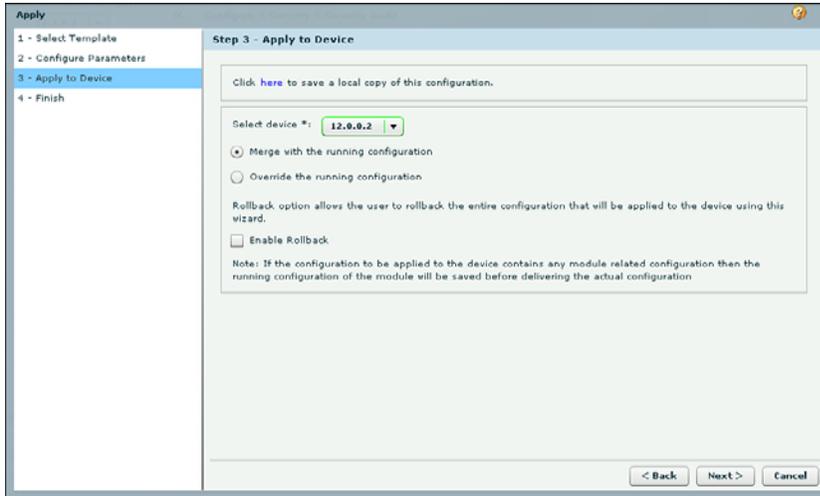


Figure 5-9 Selecting the Destination Device, and the Merge or Override Option

Note If you select the Override option, you are prompted about the router being reloaded for the new configuration to take effect. There is also a Rollback option provided, to allow restoration of a previous configuration.

User Profiles



The user profile feature enables you to restrict which features show up as available in the left side navigation pane of CCP. The profile controls which options are shown, based on which devices the user is managing. For example, you might want to hide the configuration options for a group of *Border Gateway Protocol (BGP)* routers from the CCP installation that is running on the help desk computers.

The process of user profile implementation includes creating a user profile and saving it to a computer's file system, and then importing that profile into the CCP installations where those restrictions should be implemented.

Note Restrictions implemented through CCP user profiles restrict access only via CCP. If that same user opens a *Secure Shell (SSH)* session to the router directly, the restrictions via CCP do not apply because CCP is not used in a direct session between the administrator and the router.

To create and implement a user profile, follow these steps:

- Step 1.** Select **Applications > Create User Profile**.
- Step 2.** Read the welcome screen and click **Next**.
- Step 3.** Using the check boxes, select the routers that this user profile will have an effect on, as to which features will be available for the administrator to configure. Figure 5-10 shows an example of this. Click **Next**.

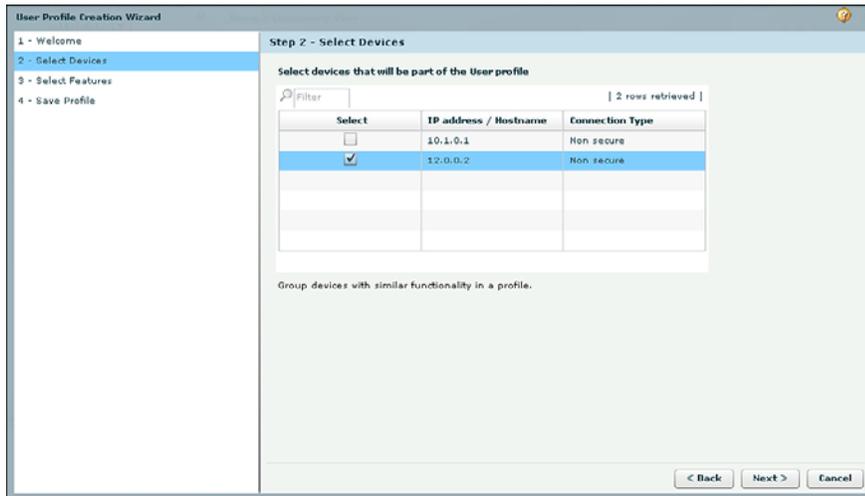


Figure 5-10 *Selecting the Routers that Will Have Navigation Pane Restrictions*

- Step 4.** Expand the contents of the folders by clicking the triangle to the left of each item. Repeat this process to expand and see the items that make up the viewable options in the navigation pane. Select the navigation pane permissions you want to provide for each option by clicking the icon and select what level of permissions to this item you want to give to the user who will be using this profile. Figure 5-11 shows an example. When done with your selections, click **Next**.

Note Although not apparent in black and white in this text, the navigation options are color coded to reflect the permissions you have chosen for each. Green means full permissions, blue means view only permissions, and red indicates that the option will not be available to the user who is working with CCP with this associated user profile.

Not all options in the navigation pane will have the read-only option available to choose from. Some may have the option to only hide the screen from the user who is running CCP and using this profile.

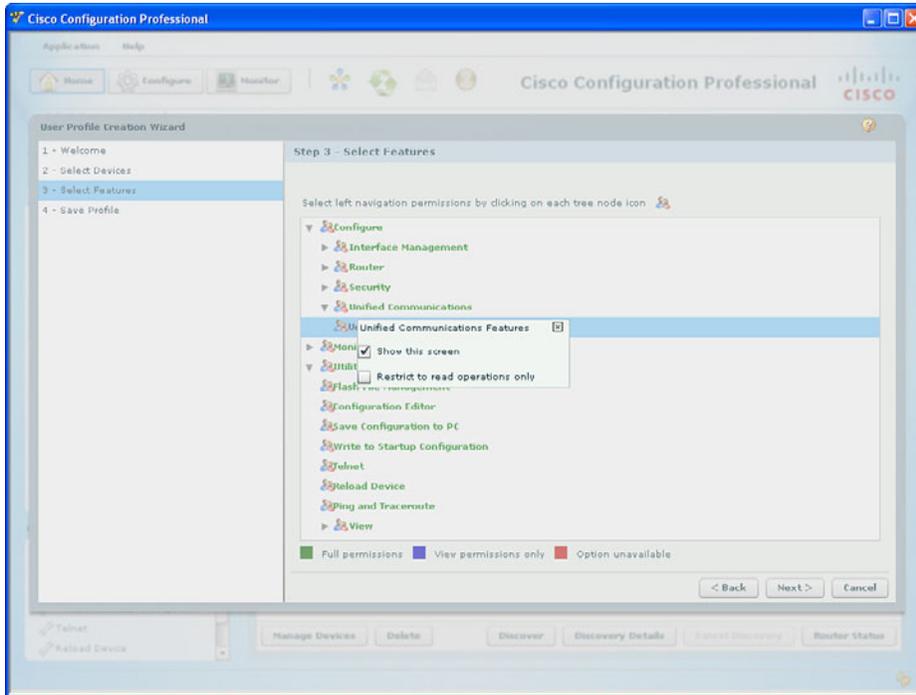


Figure 5-11 *Choosing the Navigation Pane Options to Be Available to the User*

- Step 5.** Click the **Save User Profile** button, and then specify where you want to save this profile. After the user profile is saved as a file, click **Finish**.

Note This profile will not be applied until it is called upon from the file system and applied to CCP. That step is coming up.

- Step 6.** On the computer where you want to apply the restrictive user profile for CCP, from the Application menu, choose **Import User Profile**.
- Step 7.** Click the **Browse** button and open the previously saved user template file. Click **Next**. From here, you can confirm your settings for this template, then click **Next** to apply this user profile to this installation of CCP, and then click **Finish**.

Note To verify that the user profile is correctly restricting based on your settings, from the Managed Community window rediscover the routers and then verify that those routers have the limitations set by going into configuration through CCP, to verify which options are available in the navigation pane.

CCP Audit Features

This section describes how to use the Security Audit feature integrated inside a CCP.

The CCP Security Audit can look at your current router configuration and then make recommendations on how it could be more secure. The CCP Security Audit feature is based on the command-line IOS `auto secure` feature, and can perform an almost identical list of tasks as its CLI counterpart.

Security Audit can operate in one of two ways. You can use an interactive wizard to choose which potential security threats may be changed via configuration. You can also use an option called One Step-Lockdown, which takes a subset of the features that the audit would do, most of which will not require user intervention, and then modifies the configuration to implement those security measures.

To perform a security audit, follow these steps:

- Step 1.** On the toolbar, click **Configure**. In the navigation pane, go to **Security > Security Audit**.
- Step 2.** Click **Perform Security Audit**, read the welcome page, and then click **Next**. If a one-step lockdown is desired, that option appears on this page, as shown in Figure 5-12.

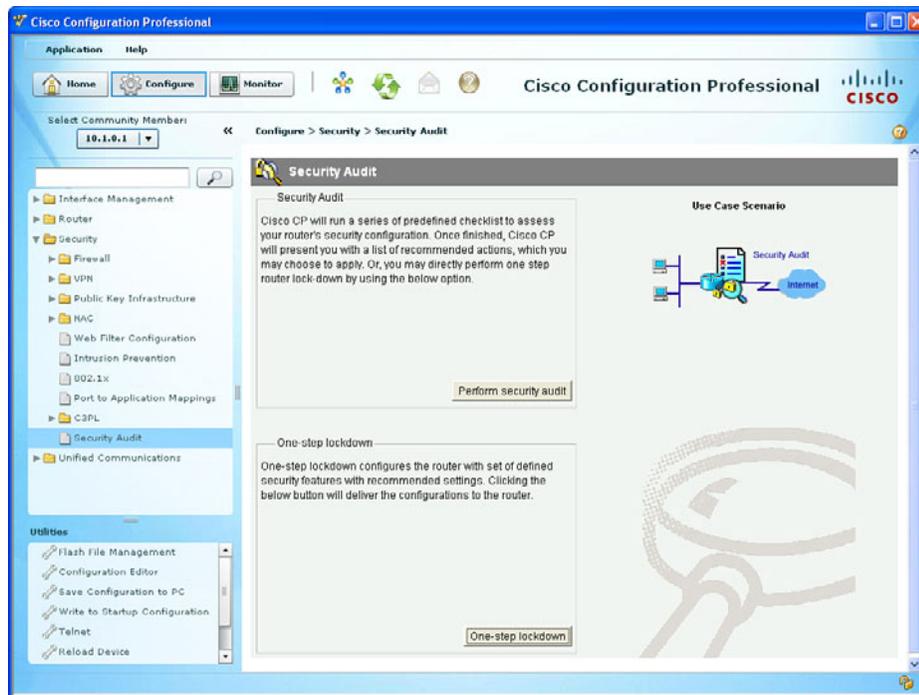


Figure 5-12 *The Security Audit Window, with Detailed and One-Step Lockdown Choices*



Step 3. For each interface listed, check either the **Inside** or **Outside** check box to indicate where the interface connects, and then click **Next**.

Note The Security Audit Wizard needs to know which of your router interfaces connect to your inside network and which connect outside your network. The wizard uses this information if firewall services need to be implemented.

Step 4. The Security Audit Wizard checks the router configuration to identify which possible security problems may exist. A screen showing the progress of this action appears, and it lists all the configuration options being tested for and whether the current router configuration passes those tests. The Security Audit Report Card screen appears, showing a list of possible security problems, as shown in Figure 5-13. Click **Close** to continue.

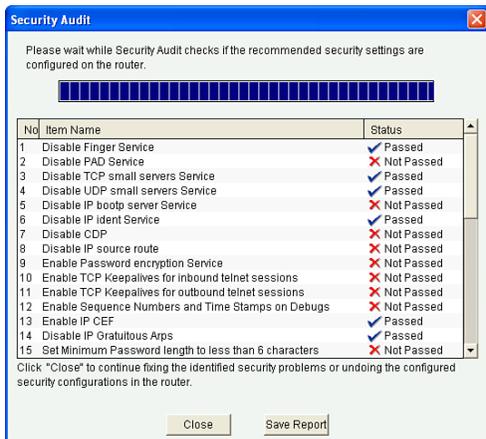


Figure 5-13 Security Audit Report Card

Note If you want to save this report to a file, click **Save Report**.

Step 5. Check the **Fix It** boxes next to any problems that you want CCP to fix, as shown in Figure 5-14. After you have identified what to correct, click **Next**.

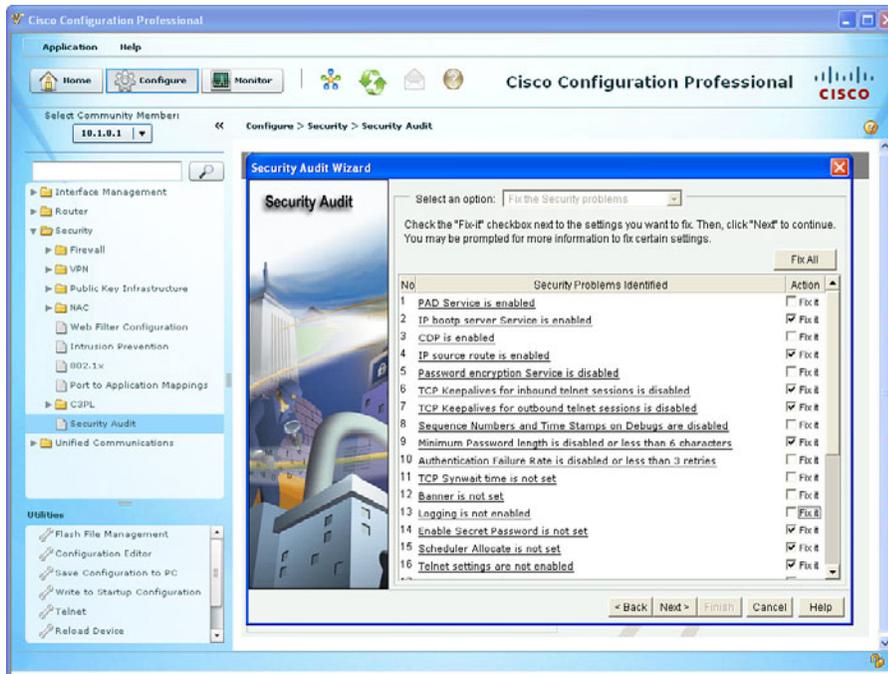


Figure 5-14 Option Screen for Fixing Identified Potential Problems or for Leaving Them “As Is”

Note For a description of the problem and a list of the Cisco IOS commands that will be added to your configuration, click the hyperlink title of the identified security problem to display a help page about that problem.

- Step 6.** The CCP Security Audit Wizard may display one or more screens requiring you to enter information to fix certain problems, such as a banner message or details about implementing a firewall, depending on which items are being corrected. Enter the information as required and click the **Next** button for each of those screens.
- Step 7.** The Summary page of the wizard shows a list of all the configuration changes that Security Audit will make. Click **Finish** to deliver those changes to your router.

Note You may or may not receive an additional pop-up window that identifies the actual commands that will be delivered. This is based on the options set under the application menu item inside of CCP. This will be true of later CCP tasks done in this book. Using the option to preview the literal commands being delivered to the router will prove useful for learning and for identifying a potential error before the configuration is deployed.

One-Step Lockdown

The other option, instead of using the interactive Security Audit Wizard, is the One-Step Lockdown feature. It addresses several features, particularly those that do not require the administrator to provide input. It provides only a subset of security measures that the full interactive Security Audit feature can perform.

Note Any item that may require input from the administrator will not be part of the implementation that One-Step Lockdown provides, because requiring input would defeat the spirit of a one-step lockdown.

A Few Highlights

Both the full security audit that is done through the wizard and the watered-down one-step lockdown enable you to implement dozens of commands with just a few clicks. This section describes a few of the more relevant Security Audit features that you can implement to mitigate security weaknesses. For a complete list of every single option, see the built-in help feature inside of CCP or the online documentation at Cisco.com. “Fix it” items offered by the Security Audit Wizard include the following:

- **Disable Finger Service:** Finger is used to find out which users are logged in to a network device. Although this information is not usually tremendously sensitive, it can sometimes be useful to an attacker.
- **Disable TCP and UDP Small Servers Service:** By default, older IOS enables the “small services”: echo, chargen, and discard. (Small services are disabled by default in the current IOS. These services are rarely used for legitimate purposes, but they can be used to launch *denial-of-service [DoS]* and other attacks.)
- **Disable IP BOOTP Server Service:** BOOTP allows both routers and computers to automatically configure from a centrally maintained server upon startup, including downloading Cisco IOS software. As a result, BOOTP can potentially be used by an attacker to download a copy of a router’s Cisco IOS software. This service is also vulnerable to DoS attacks, which could affect the router overall.
- **Disable IP Identification Service:** Identification support allows you to query a TCP port for identification. It is potentially dangerous to allow any system on a directly connected segment to learn that the router is a Cisco device and to determine the model number and the Cisco IOS software version being run. This information may be used to design attacks against the router.
- **Disable CDP:** This is dangerous in that it allows any system on a directly connected segment to learn that the router is a Cisco device and to determine the model number and the Cisco IOS software version being run. This information may be used to design attacks against the router.

- **Disable IP Source Route:** The IP protocol supports source routing options that allow the sender of an IP datagram to control the route that the datagram takes toward its ultimate destination, and generally the route that any reply takes. These options are rarely used for legitimate purposes in networks. Allowing the sender of a packet to control the path through your network is not a good idea.
- **Enable Password Encryption Service:** Providing a light encryption of a plaintext password prevents an individual who may be glancing at the configuration from being able to know what the actual plaintext password is.
- **Enable TCP Keepalives for Inbound and Outbound Telnet Sessions:** Enabling TCP keepalives causes the router to generate periodic keepalive messages, letting it detect and drop broken Telnet connections, and not leaving them open for potential unauthorized access.
- **Enable IP CEF:** Routes configured for *Cisco Express Forwarding (CEF)* perform better under SYN attacks than routers using the traditional cache.
- **Disable IP Gratuitous ARPs:** A gratuitous *Address Resolution Protocol (ARP)* is an ARP broadcast in which the source and destination IP addresses are the same. It is used primarily by a host to inform the network about its IP address. *Gratuitous* means that an ARP reply is sent without an initial request from another station. A spoofed gratuitous ARP message can cause network mapping information to be stored incorrectly, or the wrong devices to be used, causing network malfunctions.
- **Set Minimum Password Length to More Than 6 Characters:** One method attackers use to crack passwords is to try all possible combinations of characters until the password is discovered. Longer passwords have exponentially more possible combinations of characters, making this method of attack much more difficult.
- **Set Authentication Failure Rate to Less Than 3 Retries:** One method of cracking passwords, called the “dictionary” attack, is to use software that attempts to log in using every word in a dictionary. This configuration causes access to the router to be locked for a period of 15 seconds after three unsuccessful login attempts, disabling the dictionary method of attack. In addition to locking access to the router, this configuration causes a log message to be generated after three unsuccessful login attempts, warning the administrator of the unsuccessful login attempts.
- **Set TCP SYN-Wait Time to 10 Seconds:** A SYN-flood attack sends repeated connection requests to a host, but never sends the acceptance of acknowledgments that complete the connections, creating increasingly more incomplete connections at the host. This can overwhelm and disable the host. Setting the TCP SYN-Wait time to 10 seconds causes the router to shut down an incomplete connection after 10 seconds, preventing the buildup of incomplete connections at the host.
- **Set Banner:** This is to provide a banner informing unauthorized users that their use is in fact unauthorized.
- **Enable Logging:** Because it gives detailed information about network events, logging is critical in recognizing and responding to security events.

- **Set Enable Secret Password:** The `enable secret` command is used to set the password that grants privileged administrative access to the Cisco IOS system.
- **Disable SNMP:** Version 1 of the *Simple Network Management Protocol (SNMP)* is the most commonly used and is often a security risk because it uses authentication strings (passwords) called *community strings* that are stored and sent across the network in plain text. It is an easy-to-spoof protocol. SNMP is a valid protocol to run on your network to properly manage it, but it is considered a more advanced topic at this point because you should take extra precautions to secure the protocol and properly manage a network. For now, disabling it is the safest option (until you can revisit it later).
- **Set Scheduler Interval:** When a router is fast-switching a large number of packets, it is possible for the router to spend so much time responding to interrupts from the network interfaces that no other work gets done. Some very fast packet floods can cause this condition. It may stop administrative access to the router, which is dangerous when the device is under attack. Tuning the scheduler interval ensures that management access to the router is always available by causing the router to run system processes after the specified time interval, even when CPU usage is at 100 percent.
- **Set Scheduler Allocate:** On routers that do not support the command `scheduler interval`, Security Audit configures the `scheduler allocate` command whenever possible.
- **Set Users:** Security Audit secures the console, AUX, vty, and tty lines by configuring Telnet user accounts to authenticate access to these lines whenever possible. Security Audit displays a dialog box in which you can define user accounts and passwords for these lines.
- **Enable Telnet Settings:** Security Audit secures the console, AUX, vty, and tty lines by implementing the `transport input` and `transport output` commands to define which protocols can be used to connect to those lines. It also sets the exec-timeout value to ten minutes on the console and AUX lines, causing an administrative user to be logged out from these lines after ten minutes of no activity.
- **Disable IP Redirects:** ICMP supports IP traffic by relaying information about paths, routes, and network conditions. ICMP redirect messages instruct an end node to use a specific router as its path to a particular destination. In a properly functioning IP network, a router sends redirects only to hosts on its own local subnets; no end node ever sends a redirect, and no redirect ever traverses more than one network hop. However, an attacker may violate these rules; some attacks are based on this. Disabling ICMP redirects causes no operational impact to the network, and it eliminates this possible method of attack.
- **Disable IP Proxy ARP:** ARP is used by the network to convert IP addresses into MAC addresses. Normally, ARP is confined to a single LAN, but a router can act as a proxy for ARP requests, making ARP queries available across multiple LAN segments. Because it breaks the LAN security barrier, you should use proxy ARP only between two LANs with an equal security level, and only when necessary.

- **Disable IP Directed Broadcast:** An IP directed broadcast is a datagram that is sent to the broadcast address of a subnet to which the sending machine is not directly attached. The directed broadcast is routed through the network as a unicast packet until it arrives at the target subnet, where it is converted into a link-layer broadcast. By sending a continuous stream of request packets, using a spoofed source address, the attacker can create a much larger stream of replies, which can completely inundate the host whose address is being falsified.
- **Disable MOP Service:** *Maintenance Operations Protocol (MOP)* is used to provide configuration information to the router when communicating with DECnet networks. MOP is vulnerable to various attacks, and it is unlikely that anyone is running DECnet any longer.
- **Disable IP Unreachables:** ICMP supports IP traffic by relaying information about paths, routes, and network conditions. ICMP host unreachable messages are sent out if a router receives a nonbroadcast packet that uses an unknown protocol, or if the router receives a packet that it is unable to deliver to the ultimate destination because it knows of no route to the destination address. These messages can be used by an attacker to gain network mapping information.
- **Disable IP Mask Reply:** ICMP mask reply messages are sent to the device requesting the information by devices that have the requested information. An attacker can use these to gain network mapping information.
- **Enable Unicast RPF on Outside Interfaces:** *Reverse Path Forwarding (RPF)* causes the router to check the source address of any packet against the interface through which the packet entered the router. If the input interface is not a feasible path to the source address according to the routing table, the packet is dropped. This source address verification is used to defeat IP spoofing.
- **Enable Firewall on All of the Outside Interfaces:** The IOS firewall filters packets and performs session tracking, based on application layer information, such as the type of session such as HTTP or DNS requests. Traffic specified by the firewall policy is allowed to flow through the firewall.
- **Set Access Class on HTTP Server Service and vty Lines:** Security Audit limits access to the HTTP service and vty lines by configuring an access class that permits access only from directly connected network nodes.
- **Enable SSH for Access to the Router:** *Secure Shell (SSH)* is the preferred secure method for remote-access command-line access to the router because the session is encrypted.
- **Enable AAA:** Cisco IOS *Authentication, Authorization, and Accounting (AAA)* service provides a modular way of performing authentication, authorization, and accounting services. Authentication using the running configuration (a user that is in the running configuration) is required for HTTP, console, and vty access to the router.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 5-4 lists these key topics.



Table 5-4 *Key Topics*

Key Topic Element	Description	Page Number
Text	Introducing Cisco Configuration Professional	65
Figure 5-1	Understanding CCP features and the GUI	66
Table 5-3	The CCP toolbar	67
Example 5-1	Setting up new devices	70
Text	Communities	70
Text	Templates	74
Text	User profiles	78
Text	CCP Security Audit	81

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

CCP, audit, user profiles, templates, communities

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 5-5 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 5-5 *Command Reference*

Command	Description
ip http server	Enable HTTP services on the router to be managed and discovered (less secure than HTTPS)
ip http secure-server	Enable HTTPS services on the router to be managed and discovered (more secure than HTTP)
username admin privilege 15 secret cisco	Create a local user account on the router with “level 15” permissions and an MD5 hashed password
ip http authentication local	Tell the router to request a username when people connect via HTTP or HTTPS, and to check the username and password against the usernames and password (or secrets) in the running-config



This chapter covers the following subjects:

- Securing management traffic
- Implementing security measures to protect the management plane

Securing the Management Plane on Cisco IOS Devices

Accessing and configuring Cisco devices is a common occurrence for an administrator. Malicious router management traffic from an unauthorized source can pose a security threat. For example, an attacker could compromise router security by intercepting login credentials (such as the username and password). This chapter introduces the concept of the *management plane* (which is a collection of protocols and access methods we use to configure, manage, and maintain a network device) and examines how to protect it.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 6-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 6-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Securing Management Traffic	1–4, 6
Implementing Security Measures to Protect the Management Plane	5, 7–10

1. Which one of the following follows best practices for a secure password?
 - a. ABC123!
 - b. SIE3peR1#
 - c. tough-passfrazze
 - d. InterEstIng-PaSsWoRd

- 2.** When you connect for the first time to the console port on a new router, which privilege level are you using initially when presented with the command-line interface?

 - a.** 0
 - b.** 1
 - c.** 15
 - d.** 16
- 3.** Which of the following is *not* impacted by a default login authentication method list?

 - a.** AUX line
 - b.** HDLC interface
 - c.** Vty line
 - d.** Console line
- 4.** You are trying to configure a method list, and your syntax is correct, but the command is not being accepted. Which of the following might cause this failure? (Choose all that apply.)

 - a.** Incorrect privilege level
 - b.** AAA not enabled
 - c.** Wrong mode
 - d.** Not allowed by the view
- 5.** Cisco recommends which version of Simple Network Management Protocol (SNMP) on your network if you need it?

 - a.** Version 1
 - b.** Version 2
 - c.** Version 3
 - d.** Version 4
- 6.** How can you implement role-based access control (RBAC)? (Choose all that apply.)

 - a.** Provide the password for a custom privilege level to users in a given role
 - b.** Associate user accounts with specific views
 - c.** Use access lists to specify which devices can connect remotely
 - d.** Use AAA to authorize specific users for specific sets of permissions

7. Which of the following indirectly requires the administrator to configure a host name?
 - a. Telnet
 - b. HTTP
 - c. HTTPS
 - d. SSH
8. What are the two primary benefits of using NTP along with a syslog server? (Choose all that apply.)
 - a. Correlation of syslog messages from multiple different devices
 - b. Grouping of syslog messages into summary messages
 - c. Synchronization in the sending of syslog messages to avoid congestion
 - d. Accurate accounting of when a syslog message occurred
9. Which of the following commands result in a secure bootset? (Choose all that apply.)
 - a. `secure boot-set`
 - b. `secure boot-config`
 - c. `secure boot-files`
 - d. `secure boot-image`
10. What is a difference between a default and named method list?
 - a. A default method list can contain up to four methods.
 - b. A named method list can contain up to four methods.
 - c. A default method list must be assigned to an interface or line.
 - d. A named method list must be assigned to an interface or line.

Foundation Topics

Securing Management Traffic

It is tricky to fix a problem if you are unaware of the problem. So, this first section starts by classifying and describing management traffic and identifying some of the vulnerabilities that exist. It also identifies some concepts that can help you to protect that traffic. This chapter then provides implementation examples of the concepts discussed earlier.

What Is Management Traffic and the Management Plane?

When you first get a new router or switch, you connect to it for management using a blue rollover cable that connects from your computer to the console port of that router or switch. This is your first exposure to the concept of management traffic. By default, when you connect to a console port you are not prompted for a username or any kind of password. By requiring a username or password, you are taking the first steps toward improving what is called the *management plane* on this router or switch.

The management plane includes not only configuration of a system, but also who may access a system and what they are allowed to do while they are logged in. The management plane also includes messages to or from a Cisco router or switch that is used to maintain or report on the current status of the device, such as a management protocol like *Simple Network Management Protocol (SNMP)*.

Beyond the Blue Rollover Cable

Using the blue rollover cable directly connected to the console port is fairly safe. Unfortunately, it is not very convenient to require the use of a console port when you are trying to manage several devices that are located in different buildings, or on different floors of the same building. A common solution to this problem is to configure the device with an IP address that you can then use to connect to that device remotely. It is at this moment that the security risk goes up. Because you are connecting over IP, it might be possible for an unauthorized person to also connect remotely. The management plane, if it were secure, would enable you to control who may connect to manage the box, when they may connect, what they may do, and report on anything that they did. At the same time, you want to ensure that all the packets that go between the device being managed and the computer where the administrator is sitting are encrypted so that anyone who potentially may capture the individual packets while going through the network could not interpret the contents of the packets (which might contain sensitive information about the configuration or passwords used for access).

Management Plane Best Practices

When implementing a network, remember the following best practices. Each one, when implemented, improves the security posture of the management plane for your network:



- **Strong passwords:** Make passwords very difficult to break. Whenever you use passwords, make them complex and difficult to guess. An attacker can break a password in several ways, including a dictionary and/or a brute force attack. A dictionary attack automates the process of attempting to log in as the user, running through a long list of words (potential passwords); when one attempt fails, the attack just tries the next one (and so on). A brute-force attack doesn't use a list of words, but rather tries thousands or millions of possible character strings trying to find a password match (modifying its guesses progressively if it incorrectly guesses the password or stops before it reaches the boundary set by the attacker regarding how many characters to guess, with every possible character combination being tried.). A tough password takes longer to break than a simple password.
- **User authentication and AAA:** Require administrators to authenticate using usernames and passwords. This is much better than just requiring a password and not knowing exactly who the user is. To require authentication using usernames and passwords, you can use a method *authentication, authorization, and accounting (AAA)*. Using this, you can control which administrators are allowed to connect to which devices and what they can do while they are there, and you can create an audit trail (accounting records) to document what they actually did while they were logged in.
- **Role-based access control (RBAC):** Not every administrator needs full access to every device, and you can control this through AAA and custom privilege levels/parser views. For example, if there are junior administrators, you might want to create a group that has limited permissions. You could assign users who are junior administrators to that group; they then inherit just those permissions. This is one example of using RBAC. Another example of RBAC is creating a custom privilege level and assigning user accounts to that level. Regardless of how much access an administrator has, a change management plan for approving, communicating, and tracking configuration changes should be in place and used before changes are made.
- **Encrypted management protocols:** When using either in-band or out-of-band management, encrypted communications should be used, such as *Secure Shell (SSH)* or *Hypertext Transfer Protocol Secure (HTTPS)*. *Out-of-band (OOB)* management implies that there is a completely separate network just for management protocols and a different network for end users and their traffic. In-band management is when the packets used by your management protocols may intermingle with the user packets (considered less secure than OOB). Whether in-band or OOB, if a plaintext management protocol must be used, such as Telnet or HTTP, use it in combination with a *virtual private network (VPN)* tunnel that can encrypt and protect the contents of the packets being used for management.

- **Logging:** Logging is a way to create an audit trail. Logging includes not only what administrators have changed or done, but also system events that are generated by the router or switch because of some problem that has occurred or some threshold that has been reached. Determine the most important information to log, and identify logging levels to use. A logging level simply specifies how much detail to include in logging messages, and may also indicate that some less-serious logging messages do not need to be logged. Because the log messages may include sensitive information, the storage of the logs and the transmission of the logs should be protected to prevent tampering or damage. Allocate sufficient storage capacity for anticipated logging demands. Logging may be done in many different ways, and your logging information may originate from many different sources, including messages that are automatically generated by the router or switch and sent to a syslog server. A syslog server is a computer that is set up to receive and store syslog messages generated from network devices. If SNMP is used, preferably use Version 3 because of its authentication and encryption capabilities. You can use SNMP to change information on a router or switch, and you can also use it to retrieve information from the router or switch. An *SNMP trap* is a message generated by the router or switch to alert the manager or management station of some event.
- **Network Time Protocol (NTP):** Use NTP to synchronize the clocks on network devices so that any logging that includes time stamps may be easily correlated. Preferably, use NTP Version 3, to leverage its ability to provide authentication for time updates. This becomes very important to correlate logs between devices in case there is ever a breach and you need to reconstruct (or prove in a court of law) what occurred.
- **Secure system files:** Make it difficult to delete, whether accidentally or on purpose, the startup configuration files and the IOS images that are on the file systems of the local routers and switches. You can do so by using built-in IOS features discussed later in this chapter.

Note Even though OOB management is usually preferred over in-band management, some management applications benefit from in-band management. For example, consider a network management application that checks the reachability of various hosts and subnets. To check this reachability, an application might send a series of pings to a remote IP address, or check the availability of various Layer 4 services on a remote host. To perform these “availability” checks, the network management application needs to send traffic across a production data network. Also, in-band network management often offers a more economic solution for smaller networks. Even if using in-band management, it should be a separate subnet/VLAN, and one that only a select few people/devices have access to get to. This reduces your footprint for possible attack vectors.

Password Recommendations

Using passwords is one way to provide access. Using passwords alone is not as good as requiring a user ID or login name associated with the password for a user.

Here are some guidelines for password creation:

- It is best to have a minimum of eight characters for a password; bigger is better. This rule can be enforced by the local router if you are storing usernames and passwords on the router in the running config. The command **security passwords min-length** followed by the minimum password length enforces this rule on new passwords that are created, including the enable secret and line passwords on the vty, AUX, and console 0. Preexisting passwords will still operate even if they are less than the new minimum specified by the command.
- Passwords can include any alphanumeric character, a mix of uppercase and lowercase characters, and symbols and spaces. As a general security rule, passwords should not use words that may be found in a dictionary, because they are easier to break. Leading spaces in a password are ignored, but any subsequent spaces, including in the middle or at the end of a password, literally become part of that password and are generally a good idea. Another good practice is using special characters or even two different words (that are not usually associated with each other) as a passphrase when combined together. Caution should be used to not require such a complex password that the user must write it down to remember it, which increases the chance of it becoming compromised.
- Passwords in a perfect environment should be fairly complex, and should be changed periodically. The frequency of requiring a change in passwords depends on your security policy. Passwords changed often are less likely to be compromised.
- From a mathematical perspective, consider how many possibilities someone would need to try to guess a password. If only capital letters are used, you have 26 possibilities for each character. If your password is one character long, that is 26, or 26 possible variants. If you have a two-character password, that is 26², or 676 possible variants. If you start using uppercase (26) and lowercase (26), numerals (10), and basic special characters (32), your starting set becomes 94 possible variants per character. Even if we look at using an eight-character password, that is 94⁸ or 6,095,689,385,410,816 (6.1 quadrillion) possibilities.

Using AAA to Verify Users

Unauthorized user access to a network creates the potential for network intruders to gain information or cause harm or both. Authorized users need access to their network resources, and network administrators need access to the network devices to configure and manage them. AAA offers a solution for both. In a nutshell, the goal of AAA is to identify who users are before giving them any kind of access to the network, and once they are identified, only give them access to the part they are authorized to use, see, or manage. AAA can create an audit trail that identifies exactly who did what and when

they did it. That is the spirit of AAA. User accounts may be kept on the local database or on a remote server. The *local database* is a fancy way of referring to user accounts that are created on the local router and are part of the running configuration.

AAA Components



Providing network and administrative access in a Cisco environment—regardless of whether it involves administrators managing the network or users getting access through network resources—is based on a modular architecture composed of the following three functional components:

- **Authentication:** Authentication is the process by which individuals prove that they are who they claim to be. The network environment has a variety of mechanisms for providing authentication, including the use of a username and password, token cards, and challenge and response. A common use is authenticating an administrator's access to a router console port, auxiliary port, or vty lines. An analogy is a bank asking you to prove that you are who you say you are before allowing you to make a transaction. As an administrator, you can control how a user is authenticated. Choices include referring to the local running configuration on the router to look for the username, going to an external server that holds the username and password information, and other methods. To specify the method to use, you create an authentication “method list” that specifies how to authenticate the user. There can be custom named method lists or default method lists. Examples of each are shown later in this chapter.
- **Authorization:** After the user or administrator has been authenticated, authorization can be used to determine which resources the user or administrator is allowed to access, and which operations may be performed. In the case of the average user, this might determine what hours that user is allowed on the network. In the case of an administrator, it could control what the administrator is allowed to look at or modify. An analogy is a bank (after having already authenticated who you are) determining whether you are authorized to withdraw some amount of money (probably based on your balance in your account at the bank). You can create authorization method lists to specify how to authorize users on the network.
- **Accounting and auditing:** After being authenticated and possibly authorized, the user or administrator begins to access the network. It is the role of accounting and auditing to record what the user or administrator actually does with this access, what he accesses, and how long he accesses it. This is also known as *creating an audit trail*. An analogy is a bank documenting and debiting your account for the money you withdraw. You can create and assign accounting method lists to control what is accounted for and where the accounting records will be sent.

Options for Storing Usernames, Passwords, and Access Rules



Cisco provides many ways to implement AAA services for Cisco devices, many of which use a centralized service to keep usernames, passwords, and configured rules about who can access which resources. Over the years, there have been many names and access methods

associated with the central server, including calling it an authentication server, AAA server, ACS server, TACACS server, or RADIUS server. These all refer to the same type of function: a server that contains usernames, passwords, and rules about what may be accessed. A router or switch acts like a client to this server and can send requests to the server to verify the credentials of an administrator or user who is trying to access a local router or switch. The following list describes a few of these centralized server types:

- **Cisco Secure ACS Solution Engine:** This is a dedicated server that contains the usernames, their passwords, and other information about what users are allowed to access and when. In the past, this was sold as a server appliance with the *Access Control Server (ACS)* software preinstalled. A router or switch becomes a client to the server. The router can be configured to require authentication from a user or administrator before providing access, and the router sends this request to the ACS server and lets the ACS server make the decision about allowing the user or administrator to continue. The protocol used between the router and the ACS server is normally TACACS+ if you are authenticating an administrator who is seeking command-line access. The protocol used between the router and the ACS server is normally RADIUS if you are authenticating an end user for network access. These are not hard-and-fast rules, and you can use either of the two protocols for similar features in many cases.
- **Cisco Secure ACS for Windows Server:** This software package may be used for user and administrator authentication. AAA services on the router or *network access server (NAS)* contact an external Cisco Secure ACS (running on a Microsoft Windows system). This is an older flavor of ACS, but may still be relevant to the certification exams.
- **Current flavors of ACS functionality:** The most common way that ACS services are implemented today is through a virtual machine running on some flavor of VMware. Another up-and-coming service to support similar services to ACS is called the *Cisco Identity Services Engine (ISE)*, which can be bundled in a single physical or logical device or appliance.
- **Self-contained AAA:** AAA services may be self-contained in the router itself. Implemented in this fashion, this form of authentication and authorization is also known as *local* authentication and authorization. The database that contains the usernames and passwords is the running configuration of the router or IOS device, and from a AAA perspective is referred to as the *local database* on the router. So, if you create a user locally on the router, you can also say that you created a user in the local database of the router. It is the same thing. In this case, because the router is acting as its own AAA server, you do not use TACACS+ or RADIUS as a protocol to connect to a remote ACS server, because you are not using an ACS server.

Authorizing VPN Users

One common implementation of AAA is its use in authenticating users accessing the corporate LAN through a remote-access IPsec VPN.

Let's see how authentication and authorization applies to users who are trying to access our network through a VPN. The first step is to authenticate users to find out who they are, and after we find out who they are, we can then control what they are authorized for. For example, if a user connects via a VPN, that user may or may not be allowed access to certain portions of the network based on who the user is. This type of access is sometimes called *packet mode*, as in a user attempting to send packets through the network instead of trying to get a *command-line interface (CLI)* like an administrator would. A user connecting over a dial-up connection (older technology) could very likely be authenticated via a PPP connection using the same concepts. In either case, we authenticate the users by asking for their username and password, and then check the rules to see what they are authorized to access. If we use the remote *Access Control Server (ACS)* server for the authentication and authorization for an end user, we would very likely use the RADIUS protocol between the router and the AAA server.

AAA access control is supported using either a local username-password database or through a remote server (such as an ACS server). To provide access to a small group of network users, or as a backup in case the ACS server cannot be reached, a local security database can be configured in the router using the **username** command.

Router Access Authentication



Note that we must choose authentication first if we want to also use authorization for a user or administrator. We cannot choose authorization for a user without knowing who that user is through authentication first.

Typically, if we authenticate an administrator, we also authorize that administrator for what we want to allow him to do. Administrators traditionally are going to need access to the CLI. When an administrator is at the CLI, that interface is provided by something called an EXEC shell. If we want to authorize the router to provide this CLI, that is a perfect example of using AAA to first authenticate the user (in this case, the administrator) and then authorize that user to get a CLI prompt (the EXEC shell) and even place the administrator at the correct privilege level. This type of access (CLI) could also be referred to as *character mode*. Simply think of an administrator at a CLI typing in characters to assist you in remembering that this is “character” mode. With the administrator, we would very likely authenticate his login request and authorize that administrator to use an EXEC shell. If we were using a remote ACS server for this authentication and authorization of an administrator, we would very likely use TACACS+ (between the router and the ACS server) because it has the most granular control, compared with RADIUS, which is the alternative. TACACS+ and RADIUS are both discussed in another chapter of this book in greater detail.

Table 6-2 identifies some of the terms that refer to the type of access and the likely protocols used between the router acting as a client and the ACS server acting as the AAA server.

Table 6-2 AAA Components to Secure Administrative and Remote LAN Access

Access Type Mode	Mode	Where These Are Likely to Be Used	AAA Command Element
Remote administrative access Usually TACACS+ between the router and the ACS	Character (line or EXEC mode)	Lines: vty, AUX console, and tty	login, enable, exec
Remote network access end users Usually RADIUS between the router and the ACS	Packet (interface mode) such as an interface with PPP requiring authentication	Interfaces: async, group-async, BRI, PRI, Other functionality: VPN user authentication	ppp, network, vpn groups



The AAA Method List

To make implementing AAA modular, we can specify individual lists of ways we want to authenticate, authorize, and account for the users. To do this, we create a *method list* that defines what resource will be used (such as the local database, an ACS server via TACACS+ protocol or an ACS server via RADIUS protocol, and so forth). To save time, we can create a default list or custom lists. We can create method lists that define the authentication methods to use, authorization method lists that define which authorization methods to use, and accounting method lists that specify which accounting method lists to use. A default list, if created, applies to the entire router or switch. A custom list, to be applied, must be both created and then specifically referenced in line or interface configuration mode. You can apply a custom list over and over again in multiple lines or interfaces. The type of the method list may be authentication, authorization, or accounting.

The syntax for a method list is as follows:

```
aaa type {default | list-name} method-1 [method-2 method-3 method-4]
```

The commands for a method list, along with their descriptions, are shown in Table 6-3.

Table 6-3 Method List Options

Command Element	Description
<i>type</i>	Identifies the type of list being created. Relevant options are authentication , authorization , or accounting .
default	Specifies the default list of methods to be used based on the methods that follow this argument. If you use the keyword default , a custom name is not used.



Command Element	Description
<i>list-name</i>	Used to create a custom method list. This is the name of this list, and is used when this list is applied to a line, such as to vty lines 0–4.
<i>method</i>	<p>At least one method must be specified. To use the local user database, use the local keyword. A single list can contain up to 4 methods, which are tried in order, from left to right.</p> <p>In the case of an authentication method list, methods include the following:</p> <p>enable: The enable password is used for authentication. This might be an excellent choice as the last method in a method list. This way, if the previous methods are not available (such as the AAA server, which might be down or not configured), the router times out on the first methods and eventually prompts the user for the enable secret as a last resort.</p> <p>krb5: Kerberos 5 is used for authentication.</p> <p>krb5-telnet: Kerberos 5 Telnet authentication protocol is used when using Telnet to connect to the router.</p> <p>line: The line password (the one configured with the password command, on the individual line) is used for authentication.</p> <p>local: The local username database (running config) is used for authentication.</p> <p>local-case: Requires case-sensitive local username authentication.</p> <p>none: No authentication is used.</p> <p>group radius: A RADIUS server (or servers) is used for authentication.</p> <p>group tacacs+: A TACACS+ server (or servers) is used for authentication.</p> <p>group group-name: Uses either a subset of RADIUS or TACACS+ servers for authentication as defined by the aaa group server radius or aaa group server tacacs+ command.</p>

Role-Based Access Control

The concept of *role-based access control (RBAC)* is to create a set of permissions or limited access and assign that set of permissions to users or groups. Those permissions are used by individuals for their given roles, such as a role of administrator or a role of a help desk person and so on. There are different ways to implement RBAC, including creating custom privilege levels and creating parser views (coming up later in this section). In either case, the custom level or view can be assigned the permissions needed for a specific

function or role, and then users can use those custom privilege levels or parser views to carry out their job responsibilities on the network, without being given full access to all configuration options.

Custom Privilege Levels

When you first connect to a console port on the router, you are placed into user mode. User mode is really privilege level 1. This is represented by a prompt that ends with `>`. When you move into privileged mode by typing the `enable` command, you are really moving into privilege level 15. A user at privilege level 15 has access and can issue all the commands that are attached to or associated with level 15 and below. Nearly all the configuration commands, and the commands that get us into configuration mode, are associated by default with privilege level 15.

By creating custom privilege levels (somewhere between levels 2 and 14, inclusive), and assigning commands that are normally associated with privilege level 15 to this new level, you can give this subset of new commands to the individual who either logs in at this custom level or to the user who logs in with a user account that has been assigned to that level.

Limiting the Administrator by Assigning a View

Working with individual commands and assigning them to custom privilege levels is tedious at best, and it is for that reason that method is not used very often. So, what can be done if we need users to have a subset of commands available to them, but not all of them? In an earlier chapter, we looked at how *Cisco Configuration Professional (CCP)* could restrict the visibility of the features in the navigation pane by using user profiles. This technique, however, did not protect the router against a user connecting with Telnet or SSH, and if that user had level 15 permissions, the router would once again be unprotected at the CLI.

A solution to this is to use *parser views*, also referred to as simply a *view*. You can create a view and associate it with a subset of commands. When the user logs in using this view, that same user is restricted to only being able to use the commands that are part of his current view. You can also associate multiple users with a single view.



Encrypted Management Protocols

It is not always practical to have console access to the Cisco devices you manage. There are several options for remote access via IP connectivity, and the most common is an application called Telnet. The problem with Telnet is that it uses plain text, and anyone who gets a copy of those packets can identify our usernames and passwords used for access and any other information that goes between administrator and the router being managed (over the management plane). One solution to this is to not use Telnet. If Telnet must be used, it should only be used out of band, or placed within a VPN tunnel for privacy, or both.



Secure Shell provides the same functionality as Telnet, in that it gives you a CLI to a router or switch; unlike Telnet, however, SSH encrypts all the packets used in the session. So, with SSH, if a packet is captured and viewed by an unauthorized individual, it will not have any meaning because the contents of each packet are encrypted and the attacker or unauthorized person will not have the keys or means to decrypt the information. The encryption provides the feature of confidentiality.

With security, bigger really is better. With SSH, Version 2 is bigger and better than Version 1. Either version, however, is better than the unencrypted Telnet protocol. When you type in `ip ssh version 2`, (to enable version 2), the device may respond with a Version “1.99” is active. This is a function of a server that runs 2.0 but also supports backward compatibility with older versions. For more information, see RFC4253, section 5.1. You should use SSH rather than Telnet whenever possible.

For GUI management tools such as CCP, use HTTPS rather than HTTP because it encrypts the session which provides confidentiality for the packets in that session.

Using Logging Files



I still recall an incident on a customer site when a database server had a failed disk and was running on its backup. It was like that for weeks until they noticed a log message. If a second failure had occurred, the results would have been catastrophic. Administrators *should*, on a regular basis, analyze logs, especially from their routers, in addition to logs from other network devices. Logging information can provide insight into the nature of an attack. Log information can be used for troubleshooting purposes. Viewing logs from multiple devices can provide event correlation information (that is, the relationship between events occurring on different systems). For proper correlation of events, accurate time stamps on those events are important. Accurate time can be implemented through *Network Time Protocol (NTP)*.

Cisco IOS devices can send log output to a variety of destinations, including the following:

- **Console:** A router’s console port can send log messages to an attached terminal (such as your connected computer, running a terminal emulation program).
- **vty lines:** Virtual tty (vty) connections (used by SSH and Telnet connections) can also receive log information at a remote terminal (such as an SSH or Telnet client). However, the **terminal monitor** command should be issued to cause log messages to be seen by the user on that vty line.
- **Buffer:** When log messages are sent to a console or a vty line, those messages are not later available for detailed analysis. However, log messages can be stored in router memory. This “buffer” area can store messages up to the configured memory size, and then the messages are rotated out, with the first in being the first to be removed. When the router is rebooted, these messages in the buffer memory are lost.
- **SNMP server:** When configured as an SNMP device, a router or switch can generate log messages, in the form of SNMP traps and send them to an SNMP manager (server).

- **Syslog server:** A popular choice for storing log information is a syslog server, which is easily configured and can store a large volume of logs. Syslog messages can be directed to one or more syslog servers from the router or switch.

A syslog logging solution consists of two primary components: syslog servers and syslog clients. A syslog server receives and stores log messages sent from syslog clients such as routers and switches.

Not all syslog messages are created equal. Specifically, they have different levels of severity. Table 6-4 lists the eight levels of syslog messages. The higher the syslog level, the more detailed the logs. Keep in mind that more-detailed logs require a bit more storage space, and also consider that syslog messages are transmitted in clear text. Also consider that the higher levels of syslog logging consume higher amounts of CPU processing time. For this reason, take care when logging to the console at the debugging level.

Table 6-4 *Syslog Severity Levels*

Level	Name	Description
0	Emergencies	System is unusable.
1	Alerts	Immediate action needed.
2	Critical	Critical conditions.
3	Errors	Error conditions.
4	Warnings	Warning conditions.
5	Notifications	Normal, but significant conditions.
6	Informational	Informational messages.
7	Debugging	Highly detailed information based on current debugging that is turned on.

The syslog log entries contain time stamps, which are helpful in understanding how one log message relates to another. The log entries include severity level information in addition to the text of the syslog messages. Having synchronized time on the routers, and including time stamps in the syslog messages, makes correlation of the syslog messages from multiple devices more meaningful.

Understanding NTP

Network Time Protocol (NTP) uses UDP port 123, and it allows network devices to synchronize their time. Ideally, they would synchronize their time to a trusted time server. You can configure a Cisco router to act as a trusted NTP server for the local network, and in the same way, that trusted NTP server could turn around and be an NTP client to a trusted NTP server either on the Internet or reachable via network connectivity. NTP Version 3 supports cryptographic authentication between NTP devices, and for this reason its use is preferable over any earlier versions.

One benefit of having reliable synchronized time is that log files and messages generated by the router can be correlated. In fact, if we had 20 routers, and they were all reporting various messages and all had the same synchronized time, we could very easily correlate the events across all 20 routers if we looked at those messages on a common server. A common server that is often used is a syslog server.

Protecting Cisco IOS Files

Similar to the computers that we use every day, a router also uses an operating system. The Cisco operating system on the router is called *IOS*. When a router first boots, it performs a power-on self-test, and then looks for an image of IOS on the flash. After loading the IOS into RAM, the router then looks for its startup configuration. If for whatever reason an IOS image or the startup configuration cannot be found or loaded properly, the router will effectively be nonfunctional as far as the network is concerned.

To help protect a router from accidental or malicious tampering of the IOS or startup configuration, Cisco offers a resilient configuration feature. This feature maintains a secure working copy of the router IOS image and the startup configuration files at all times. Once enabled, the administrator cannot disable the features remotely (but can if connected directly on the console). The secure files are referred to as a *secure bootset*.

Implement Security Measures to Protect the Management Plane

The first section of this chapter covered some best practices to protect the management plane. With that in mind, you can now leverage what you have learned and look at some practical examples of implementing those best practices. It requires both the understanding and implementation of these best practices to secure your networks.

Implementing Strong Passwords

The privileged EXEC secret (the one used to move from user mode to privileged mode) should not match any other password that is used on the system. Many of the other passwords are stored in plain text (such as passwords on the vty lines). If an attacker discovers these other passwords, he might try to use them to get into privileged mode, and that is why the enable secret should be unique. Service password encryption scrambles any plaintext passwords as they are stored in the configuration. This is useful for preventing someone who is looking over your shoulder from reading a plaintext password that is displayed in the configuration on the screen. Any new plaintext passwords are also scrambled as they are stored in the router's configuration.

Example 6-1 shows the use of strong passwords.

Example 6-1 *Using Strong Passwords*

```

! Use the "secret" keyword instead of the "password" for users
! This will create a secured password in the configuration by default
! The secret is hashed using the MD5 algorithm as it is stored in the
! configuration
R1(config)# username admin secret CeyeSc01$24

! At a minimum, require a login and password for access to the console port
! Passwords on lines, including the console, are stored as plain text, by
! default, in the configuration
R1(config)# line console 0
R1(config-line)# password k4(1fmMsS1#
R1(config-line)# login
R1(config-line)# exit

! At a minimum, require a login and password for access to the VTY lines which
! is where remote users connect when using Telnet
! Passwords on lines, including the vty lines, are stored as plain text, by
! default, in the configuration
R1(config)# line vty 0 4
R1(config-line)# password 8wT1*eGP5@
R1(config-line)# login

! At a minimum, require a login and password for access to the AUX line
! and disable the EXEC shell if it will not be used
R1(config-line)# line aux 0
R1(config-line)# no exec
R1(config-line)# password 1wT1@ecP27
R1(config-line)# login
R1(config-line)# exit

! Before doing anything else, look at the information entered.
R1(config)# do show run | include username
username admin secret 5 $1$XJdX$9hqvG53z3lesP5BLOqgg0.
R1(config)#
R1(config)# do show run | include password
no service password-encryption
password k4(1fmMsS1#
password 8wT1*eGP5@
password 1wT1@ecP27
R1(config)#

```

```

! Notice that we can not determine the admin user's password, since
! it is automatically hashed using the MD5 algorithm because of using
! the secret command, however, we can still see all the other plain text
! passwords.

! Encrypt the plain text passwords so that someone reading the configuration
! won't know what the passwords are by simply looking at the configuration.
R1(config)# service password-encryption

! Verify that the plain text passwords configured are now scrambled due to the
! command "service password-encryption"
R1(config)# do show run | begin line
line con 0
  password 7 04505F4E5E2741631A2A5454
  login
line aux 0
  no exec
  login
  password 7 075E36781F291C0627405C
line vty 0 4
  password 7 065E18151D040C3E354232
  login
!
end

```

User Authentication with AAA

Example 6-2 shows the use of method lists, both named and default.



Example 6-2 *Enabling AAA Services and Working with Method Lists*

```

! Enable aaa features, if not already present in the running configuration
R1(config)# aaa new-model

! Identify a AAA server to be used, and the password it is expecting with
! requests from this router. This server would need to be reachable and
! configured as a TACACS+ server to support R1's requests
R1(config)# tacacs-server host 50.50.4.101
R1(config)# tacacs-server key ToUgHPaSSw0rD-1#7

! configure the default method list for the authentication of character
! mode login (where the user will have access to the CLI)
! This default method list, created below has two methods listed "local"
! and "enable"

```

```

! This list is specifying that the local database (running-config) will
! be used first to look for the username.  If the username isn't in the
! running-config, then it will go to the second method in the list.
! The second method of "enable" says that if the user account isn't found
! in the running config, then to use the enable secret to login.
! This default list will apply to all SSH, Telnet, VTY, AUX and Console
! sessions unless there is another (different) custom method list that is
! created and directly applied to one of those lines.
R1(config)# aaa authentication login default local enable

! The next authentication method list is a custom authentication
! method list named MY-LIST-1.This method list says that the first attempt
! to verify the user's name and password should be done through one of the
! tacacs servers (we have only configured one so far), and then if that server
! doesn't respond, use the local database (running-config), and if the
! username isn't in the running configuration to then use the enable secret
! for access to the device.  Note: this method list is not used until
! applied to a line elsewhere in the configuration.
R1(config)# aaa authentication login MY-LIST-1 group tacacs local enable

! These next method lists are authorization method lists.
! We could create a default one as well, using the key
! word "default" instead of a name.  These custom method lists for
! authorization won't be used until we apply them
! elsewhere in the configuration, such as on a VTY line.
! The first method list called TAC1 is an authorization
! method list for all commands at user mode (called privilege level 1).
! The second method list called TAC15 is an
! authorization method list for commands at level 15 (privileged exec mode).
! If these method lists are applied to a line, such as the
! console or VTY lines, then before any commands
! are executed at user or privileged mode, the router will check
! with an ACS server that is one of the "tacacs+" servers, to see if the user
! is authorized to execute the command.  If a tacacs+ server isn't
! reachable, then the router will use its own database of users (the local
! database) to determine if the user trying to issue the command
! is at a high enough privilege level to execute the command.
R1(config)# aaa authorization commands 1 TAC1 group tacacs+ local
R1(config)# aaa authorization commands 15 TAC15 group tacacs+ local

```

```

! The next 2 method lists are accounting method lists that will record the
! commands issued at level 1 and 15 if the lists are applied to a line, and
! if an administrator connects to this device via that line.
! Accounting method lists can have multiple methods, but can't log to the
! local router.
R1(config)# aaa accounting commands 1 TAC-act1 start-stop group tacacs+
R1(config)# aaa accounting commands 15 TAC-act15 start-stop group tacacs+

! Creating a user with level 15 access on the local router is a good idea,
! in the event the ACS server can't be
! reached, and a backup method has been specified as the local database.
R1(config)# username admin privilege 15 secret 4Je7*1swEsf

! Applying the named method lists is what puts them in motion.
! By applying the method lists to the VTY lines
! any users connecting to these lines will be authenticated by the
! methods specified by the lists that are applied
! and also accounting will occur, based on the lists that are applied.
R1(config)# line vty 0 4
R1(config-line)# login authentication MY-LIST-1
R1(config-line)# authorization commands 1 TAC1
R1(config-line)# authorization commands 15 TAC15
R1(config-line)# accounting commands 1 TAC-act1
R1(config-line)# accounting commands 15 TAC-act15

! Note: on the console and AUX ports, the default list will be applied,
! due to no custom method list being applied
! directly to the console or AUX ports.

```

Using **debug** as a tool to verify what you think is happening is a good idea. In Example 6-3, we review and apply AAA and perform a **debug** verification.

Example 6-3 *Another Example of Creating and Applying a Custom Method List to vty Lines*

```

! Creating the method list, which has 3 methods. First the local database
! (if the username exists in the configuration, and if not
! then the enable secret (if configured), and if not then no
! authentication required
! (none)
R2(config)# aaa authentication login MY-AUTHEN-LIST-1 local enable none

! Applying the method list to the VTY lines 0-4
R2(config)# line vty 0 4
R2(config-line)# login authentication MY-AUTHEN-LIST-1
R2(config-line)# exit

```

```
! Creating a local username in the local database (running-config)
R2(config)# username bob secret ciscobob

! Setting the password required to move from user mode to privileged mode
R2(config)# enable secret ciscoenable
R2(config)# interface loopback 0

! Applying an IP address to test a local telnet to this same local router
! Not needed if the device has another local IP address that is in use
R2(config-if)# ip address 2.2.2.2 255.255.255.0
R2(config-if)# exit

! Enable logging so we can see results of the upcoming debug
R2(config)# logging buffered 7
R2(config)# end

! Enabling debug of aaa authentication, so we can see what the router is
! thinking regarding aaa authentication
R2# debug aaa authentication
AAA Authentication debugging is on

R2# clear log
Clear logging buffer [confirm]

! Telnet to our own address
R2# telnet 2.2.2.2
Trying 2.2.2.2 ... Open

User Access Verification

Username: bob
AAA/BIND(00000063): Bind i/f
AAA/AUTHEN/LOGIN (00000063): Pick method list 'MY-AUTHEN-LIST-1'
Password: [ciscobob] password not shown when typing it in

R2>

! We can see that bob is connected via line vty 0, and that from the debug
! the correct authentication list was used.
R2>who
      Line      User      Host(s)      Idle      Location
    0 con 0
*  2 vty 0      bob       idle         00:00:00  2.2.2.2
R2> exit
```

```
! If we exit back out, and remove all the users in the local database,  
! (including bob) then the same login authentication will fail on the first  
! method of the "local" database (no users there), and will go to the second  
! method in the list, which is "enable", meaning use the enable secret if  
! configured.
```

```
! As soon as I supply a username, the router discovers that there are no  
! usernames  
! configured in running configuration (at least none that match the user  
! who is trying to  
! login), and fails on the first method "local" in the list  
! It then tries the next method of just caring about the enable secret.
```

```
R2# telnet 2.2.2.2
```

```
Trying 2.2.2.2 ... Open
```

```
User Access Verification
```

```
AAA/BIND(00000067): Bind i/f
```

```
AAA/AUTHEN/LOGIN (00000067): Pick method list 'MY-AUTHEN-LIST-1'
```

```
! Note: bertha is not a configured user in the local database on the router
```

```
Username: bertha
```

```
Password: [ciscoenable] not shown while typing. This is the enable secret  
we set.
```

```
AAA/AUTHEN/ENABLE(00000067): Processing request action LOGIN
```

```
AAA/AUTHEN/ENABLE(00000067): Done status GET_PASSWORD
```

```
R2>
```

```
AAA/AUTHEN/ENABLE(00000067): Processing request action LOGIN
```

```
AAA/AUTHEN/ENABLE(00000067): Done status PASS
```

```
R2> exit
```

```
! One more method exists in the method list we applied to the VTY lines.  
! If the local fails, and the enable secret fails (because neither of these  
! is configured on the router, then the third method in the method list  
! 'MY-AUTHEN-LIST-1' will be tried. The third method we specified is none,  
! meaning no authentication required, come right in. After removing the  
! enable secret, we try once more.
```

```
R2# telnet 2.2.2.2
```

```
Trying 2.2.2.2 ... Open
```

```
User Access Verification
```

```

AAA/BIND(00000068): Bind i/f
AAA/AUTHEN/LOGIN (00000068): Pick method list 'MY-AUTHEN-LIST-1'
Username: doesn't matter
R2>
AAA/AUTHEN/ENABLE(00000068): Processing request action LOGIN
AAA/AUTHEN/ENABLE(00000068): Done status FAIL - secret not configured
R2>
! No password was required. All three methods of the method list were
! tried.
! The first two methods failed, and the third of "none" was accepted.

```

Using the CLI to Troubleshoot AAA for Cisco Routers

One tool you can use when troubleshooting AAA on Cisco routers is the **debug** command. You may use three separate **debug** commands to troubleshoot the various aspects of AAA:



- **debug aaa authentication:** Use this command to display debugging messages for the authentication functions of AAA.
- **debug aaa authorization:** Use this command to display debugging messages for the authorization functions of AAA.
- **debug aaa accounting:** Use this command to display debugging messages for the accounting functions of AAA.

Each of these commands is executed from privileged EXEC mode. To disable debugging for any of these functions, use the **no** form of the command, such as **no debug aaa authentication**.

Example 6-4 shows an example of debugging login authentication, EXEC authorization, and commands at level 15 authorization. As shown in the example, you can use **debug** not only for verification, as in the preceding example, but also as a troubleshooting method.

Example 6-4 Using debug Commands

```

! R4 will have a loopback, so we can telnet to ourselves to test
R4(config-if)# ip address 4.4.4.4 255.255.255.0
R4(config-if)# exit

! Local user in the database has a privilege level of 15
R4(config)# username admin privilege 15 secret cisco

```



```
! This method list, if applied to a line, will specify local authentication
R4(config)# aaa authentication login AUTHEN_Loc local

! This next method list, if applied to a line, will require authorization
! before giving the administrator an exec shell.  If the user has a valid
! account in the running configuration, the exec shell will be created for
! the authenticated
! user, and it will place the user in their privilege level automatically
R4(config)# aaa authorization exec AUTHOR_Exec_Loc local

! This method list, if applied to a line, will require authorization for
! each and every level 15 command issued.  Because the user is at
! privilege level 15 the router will say "yes" to any level 15 commands
! that may be issued by the user
R4(config)# aaa authorization commands 15 AUTHOR_Com_15 local

! Next we will apply the 3 custom method lists to vty lines 0-4, so that
! when anyone connects via these vty lines, they will be subject to the
! login authentication, the exec authorization, and the level 15 command
! authorizations for the duration of their session.

R4(config)# line vty 0 4
R4(config-line)# login authentication AUTHEN_Loc
R4(config-line)# authorization exec AUTHOR_Exec_Loc
R4(config-line)# authorization commands 15 AUTHOR_Com_15
R4(config-line)# exit
R4(config)#
R4(config)# do debug aaa authentication
AAA Authentication debugging is on
R4(config)# do debug aaa authorization
AAA Authorization debugging is on
R4(config)# exit

! Now test to see it all in action.
R4# telnet 4.4.4.4
Trying 4.4.4.4 ... Open
User Access Verification

Username: admin
Password: [cisco] password not displayed when entering

! It picked the login authentication list we specified
AAA/BIND(00000071): Bind i/f
AAA/AUTHEN/LOGIN (00000071): Pick method list 'AUTHEN_Loc'
```

```

! It picked the authorization list we specified for the exec shell
R4#
AAA/AUTHOR (0x71): Pick method list 'AUTHOR_Exec_Loc'
AAA/AUTHOR/EXEC(00000071): processing AV cmd=
AAA/AUTHOR/EXEC(00000071): processing AV priv-lvl=15
AAA/AUTHOR/EXEC(00000071): Authorization successful

! It picked the command level 15 authorization list, when we issued the
! configure terminal command, which is a level 15 command.
R4# config t
Enter configuration commands, one per line. End with CNTL/Z.
R4(config)#
AAA/AUTHOR: auth_need : user= 'admin' ruser= 'R4' rem_addr= '4.4.4.4' priv=
 15 list=
'AUTHOR_Com_15' AUTHOR-TYPE= 'command'
AAA: parse name=tty2 idb type=-1 tty=-1
AAA: name=tty2 flags=0x11 type=5 shelf=0 slot=0 adapter=0 port=2 channel=0
AAA/MEMORY: create_user (0x6A761F34) user='admin' ruser='R4' ds0=0
  port='tty2'
rem_addr='4.4.4.4' authen_type=ASCII service=NONE priv=15 initial_task_
 id='0',
vrf= (id=0)
tty2 AAA/AUTHOR/CMD(1643140100): Port='tty2' list='AUTHOR_Com_15'
  service=CMD
AAA/AUTHOR/CMD: tty2(1643140100) user='admin'
tty2 AAA/AUTHOR/CMD(1643140100): send AV service=shell
tty2 AAA/AUTHOR/CMD(1643140100): send AV cmd=configure
tty2 AAA/AUTHOR/CMD(1643140100): send AV cmd-arg=terminal
tty2 AAA/AUTHOR/CMD(1643140100): send AV cmd-arg=<cr>
tty2 AAA/AUTHOR/CMD(1643140100): found list "AUTHOR_Com_15"
tty2 AAA/AUTHOR/CMD(1643140100): Method=LOCAL
AAA/AUTHOR (1643140100): Post authorization status = PASS_ADD
AAA/MEMORY: free_user (0x6A761F34) user='admin' ruser='R4' port='tty2'
rem_addr='4.4.4.4' authen_type=ASCII service=NONE priv=15 vrf= (id=0)
R4(config)#
! It made a big splash, with lots of debug output, but when you boil it all
! down it means the user was authorized to issue the configure terminal
! command.

```

There is also a **test aaa** command that is very useful when verifying connectivity with a remote ACS server.

This section walked you through the details of AAA using the command line with very exact examples because you need to understand how it works. Now that you have taken

a look at how it works, you should know that you can also use CCP as a GUI to implement the AAA.

Let's take a moment to review where you can find the AAA elements inside CCP. In the configuration section, using the navigation pane on the left, go to **Configure > Router > AAA > AAA Summary**. You will see there an overview of what authentication policies have been created on a router and any authorization or accounting policies, as shown in Figure 6-1.

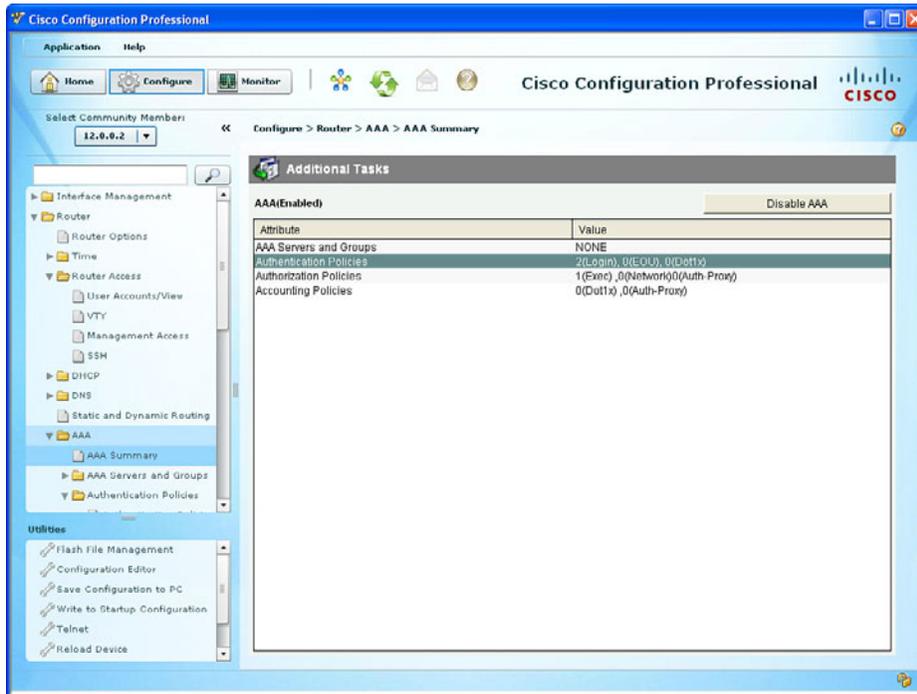


Figure 6-1 Using CCP to View AAA Policies

If you wanted to add, edit, or modify your authentication policies, you just navigate to **Configure > Router > AAA > Authentication Policies > Login**, as shown in Figure 6-2.

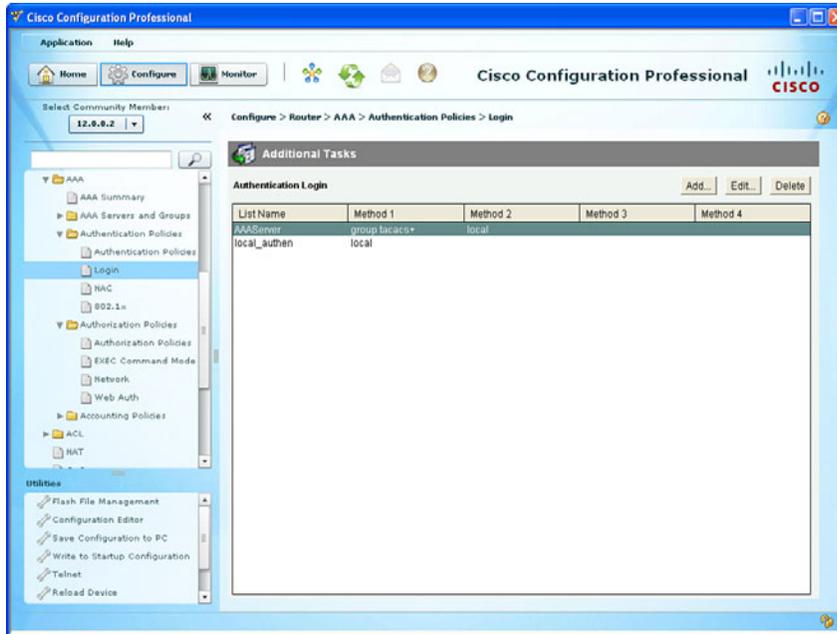


Figure 6-2 Using CCP to See Method Lists for Login

If you want to see which method lists were applied to your vty lines, just navigate to **Configure > Router > Router Access > VTY**, as shown in Figure 6-3.

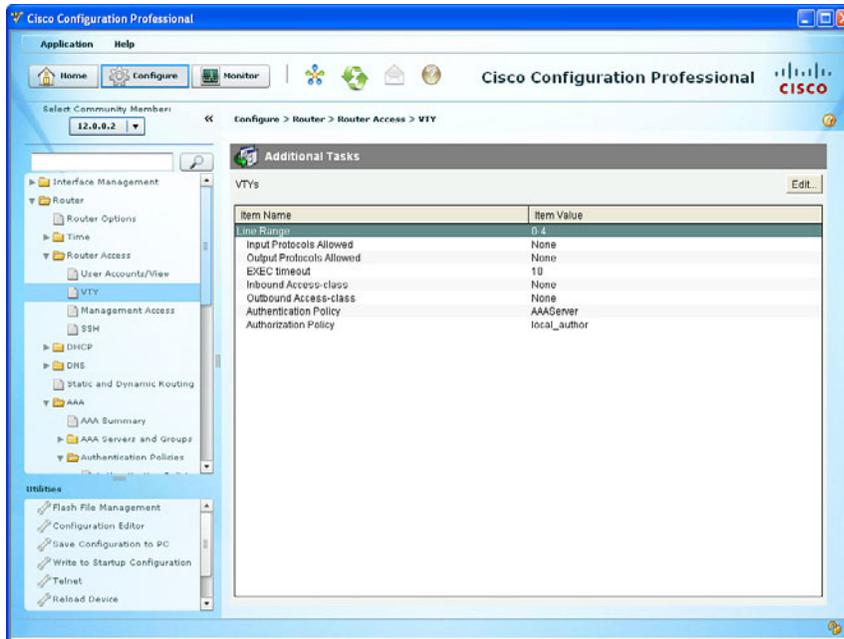


Figure 6-3 Using CCP to See Which Methods Have Been Applied to the vty Lines

From here, you can also modify which AAA policies are applied to vty lines by clicking **Edit**, which prompts the opening of an Edit VTY Lines dialog, as shown in Figure 6-4.

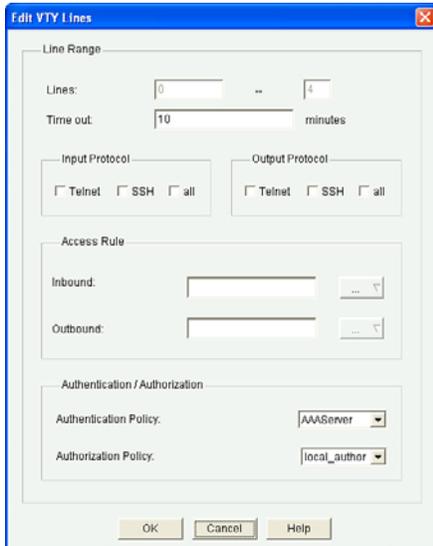


Figure 6-4 Using CPP to Edit vty Line Properties, Including AAA Method Lists Applied

RBAC Privilege Level/Parser View

You may implement RBAC through AAA, with the rules configured on an ACS server, but you may implement it in other ways, too, including creating custom privilege levels and having users enter those custom levels where they have a limited set of permissions, or creating a *parser view* (also sometimes simply called a *view*), which also limits what the user can see or do on the Cisco device. Each options can be tied directly to a username, so that once users authenticate they may be placed at the custom privilege level, or in the view that is assigned to them.

Let's implement a custom privilege level first, as shown in Example 6-5. The example includes explanations throughout.

Example 6-5 Creating and Assigning Commands to a Custom Privilege Level

Key
Topic

```
! By default, we use privilege level 1 (called user mode), and privilege
! level 15 (called privileged mode). By creating custom levels, (between
! 1-15) and assigning commands to those levels, we are creating custom
! privilege levels
! A user connected at level 8, would have any of the new commands
! associated with level 8, as well as any commands that have been custom
! assigned or defaulted to levels 8 and below. A user at level 15 has
! access to all commands at level 15 and below.
```

```

! This configuration assigns the command "configure terminal" to privilege
! level 8
R2(config)# privilege exec level 8 configure terminal

! This configuration command assigns the password for privilege level 8
! the keyword "password" could be used instead of secret, but is less secure
! as the "password" doesn't use the MD5 hash to protect the password
! The "0" before the password, implies that we are inputting a non-hashed
! (to begin with) password. The system will hash this for us, because we
! used the enable "secret" keyword.
R2(config)# enable secret level 8 0 NewPa5s123&
R2(config)# end
R2#
%SYS-5-CONFIG_I: Configured from console by console

! To enter this level, use the enable command, followed by the level you want
! to enter. If no level is specified, the default level is 15
R2# disable
! Validate that user mode is really privilege level 1
R2> show privilege
Current privilege level is 1
! Context sensitive help shows that we can enter a level number after the
! word enable
R2> enable ?
  <0-15> Enable level
  view   Set into the existing view
  <cr>

R2> enable 8
Password: [NewPa5s123&] ! note: password doesn't show when typing it in
R2# show privilege
Current privilege level is 8
! We can go into configuration mode, because "configure terminal" is at our
! level
R2# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
! Notice we don't have further ability to configure the router, because
! level 8 doesn't include the interface configuration or other router
! configuration commands.
R2(config)# ?
Configure commands:
  beep      Configure BEEP (Blocks Extensible Exchange Protocol)
  call      Configure Call parameters
  default   Set a command to its defaults

```

end	Exit from configure mode
exit	Exit from configure mode
help	Description of the interactive help system
netconf	Configure NETCONF
no	Negate a command or set its defaults
oer	Optimized Exit Routing configuration submodes
sasl	Configure SASL
wsma	Configure Web Services Management Agents

If we are requiring login authentication, we can associate a privilege level with a given user account, and then when users authenticate with their username and password they will automatically be placed into their appropriate privilege level. Example 6-6 shows an example of this.

Example 6-6 *Creating a Local User and Associating That User with Privilege Level 8 and Assigning Login Requirements on the vty Lines*

```
! Create the user account in the local database (running-config) and
! associate that user with the privilege level you want that user to use.
R2(config)# username Bob privilege 8 secret Cisco123
R2(config)# line vty 0 4

! "login local" will require a username and password for access if the "aaa
! new-model" command is not present. If we have set the aaa new-model,
! then we would also want to create a default or named method list that
! specifies we want to use the local database for authentication.
R2(config-line)# login local

! Note: Once bob logs in, he would have access to privilege level 8 and
! below, (including all the normal show commands at level 1)
```

Implementing Parser Views



To restrict users without having to create custom privilege levels, you can use a *parser* view, also referred to as simply a *view*. A view can be created with a subset of privilege level 15 commands, and when the user logs in using this view, that same user is restricted to only being able to use the commands that are part of his current view.

To create a view, an enable secret password must first be configured on the router. AAA must also be enabled on the router (**aaa new-model** command).

Example 6-7 shows the creation of a view.

Example 6-7 *Creating and Working with Parser Views*

```

! Set the enable secret, and enable aaa new-model (unless already in
! place)
R2(config)# enable secret aBc!2#&iU
R2(config)# aaa new-model
R2(config)# end

! Begin the view creation process by entering the "default" view, using the
! enable secret
R2# enable view
Password: [aBc!2#&iU] note password not shown when typed

R2#
%PARSER-6-VIEW_SWITCH: successfully set to view 'root'.
R2# configure terminal

! As the administrator in the root view, create a new custom view
R2(config)# parser view New_VIEW
%PARSER-6-VIEW_CREATED: view 'New_VIEW' successfully created.

! Set the password required to enter this new view
R2(config-view)# secret New_VIEW_PW

! Specify which commands you want to include as part of this view.
! commands "exec" refer to commands issued from the command prompt
! commands "configure" refer to commands issued from privileged mode
R2(config-view)# commands exec include ping
R2(config-view)# commands exec include all show
R2(config-view)# commands exec include configure

! This next line adds the ability to configure "access-lists" but nothing
! else
R2(config-view)# commands configure include access-list
R2(config-view)# exit
R2(config)# exit

! Test the view, by going to user mode, and then back in using the new view
R2# disable
R2>enable view New_VIEW
Password: [New_VIEW_PW] Password not shown when typed in

! Console message tells us that we are using the view
%PARSER-6-VIEW_SWITCH: successfully set to view 'New_VIEW'.

```

```

! This command reports what view we are currently using
R2# show parser view
Current view is 'New_VIEW'

! We can verify that the commands assigned to the view work
! Note: we only assigned configure, not configure terminal so we have to
! use the configure command, and then tell the router we are configuring
! from the terminal. We could have assigned the view "configure terminal"
! to avoid this
R2# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

! Notice that the only configuration options we have are for access-list,
! per the view
R2(config)# ?
Configure commands:
  access-list  Add an access list entry
  do           To run exec commands in config mode
  exit        Exit from configure mode

```

We could also assign this view to a user account, so that when users log in with their username and password, they are automatically placed into their view, as shown in Example 6-8.

Example 6-8 *Associating a User Account with a Parser View*

```
R2(config)# username Lois view New_VIEW secret cisco123
```

Note This creation of a username and assigning that user to a view needs to be done by someone who is at privilege level 15.

SSH and HTTPS

Because Telnet sends all of its packets as plain text, it is not secure. SSH allows remote management of a Cisco router or switch, but unlike Telnet, SSH encrypts the contents of the packets to protect it from being interpreted if they fall into the wrong hands.

To enable SSH on a router or switch, the following items need to be in place:

- Hostname other than the default name of “router”
- Domain name
- Generating a public/private key pair, used behind the scenes by SSH
- Requiring user login via the vty lines, instead of just a password. Local authentication or authentication using an ACS server are both options.

- Having at least one user account to log in with, either locally on the router, or on an ACS server

Example 6-9 shows how to implement these components, along with annotations and examples of what happens when the required parts are not in place. If you have a non-production router or switch handy, you might want to follow along.

Example 6-9 *Preparing for SSH*



```

! To create the Public/Private key pair used by SSH, we would issue the
! following command. Part of the key pair, will be the hostname and the
! domain name.
! If these are not configured first, the crypto key generate command will
! tell you as shown in the next few lines.
Router(config)# crypto key generate rsa
% Please define a hostname other than Router.
Router(config)# hostname R1
R1(config)# crypto key generate rsa
% Please define a domain-name first.
R1(config)# ip domain-name cisco.com

! Now with the host and domain name set, we can generate the key pair
R1(config)# crypto key generate rsa
The name for the keys will be: R1.cisco.com
Choose the size of the key modulus in the range of 360 to 2048 for your
General Purpose Keys. Choosing a key modulus greater than 512 may take
a few minutes.

! Bigger is better with cryptography, and we get to choose the size for the
! modulus
! The default is 512 on many systems, but you would want to choose 1024 or
! more to improve security. SSH has several flavors, with version 2 being
! more secure than version 1. To use version 2, you would need at least a
! 1024 size for the key pair
How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable... [OK]

R1(config)#
%SSH-5-ENABLED: SSH 1.99 has been enabled
! Note the "1.99" is based on the specifications for SSH from RFC 4253
! which indicate that an SSH server may identify its version as 1.99 to
! identify that it is compatible with current and older versions of SSH.

! Create a user in the local database
R1(config)# username Keith secret Ci#kRk*ks

```

```
! Configure the vty lines to require user authentication
R1(config)# line vty 0 4
R1(config-line)# login local

! Alternatively, we could do the following for the requirement of user
! authentication
! This creates a method list which points to the local database, and then
! applies that list to the VTY lines
R1(config)# aaa new-model
R1(config)# aaa authentication login Keith-List-1 local
R1(config)# line vty 0 4
R1(config-line)# login authentication Keith-List-1

! To test this we could SSH to ourselves from the local machine, or from
! another router that has IP connectivity to this router.

R1# ssh ?
-c    Select encryption algorithm
-l    Log in using this user name
-m    Select HMAC algorithm
-o    Specify options
-p    Connect to this port
-v    Specify SSH Protocol Version
-vrf  Specify vrf name
WORD  IP address or hostname of a remote system

! Note: one of our local IP addresses is 10.1.0.1
R1# ssh -l Keith 10.1.0.1

Password: <password for Keith goes here>

R1>
! to verify the current SSH session(s)
R1>show ssh
Connection Version Mode Encryption Hmac      State      Username
0           2.0      IN   aes128-cbc hmac-sha1  Session started Keith
0           2.0      OUT  aes128-cbc hmac-sha1  Session started Keith
%No SSHv1 server connections running.
R1>
```

Perhaps you want to manage a router via HTTPS. If so, you can use CCP or a similar tool and implement HTTPS functionality, as shown in Example 6-10.

Example 6-10 *Preparing for HTTPS*

```
! Enable the SSL service on the local router.  If it needs to generate
! keys for this feature, it will do so on its own in the background.
R1(config)# ip http secure-server

! Specify how you want users who connect via HTTPS to be authenticated
R1(config)# ip http authentication ?
aaa      Use AAA access control methods
enable   Use enable passwords
local    Use local username and passwords

R1(config)# ip http authentication local

! If you are using the local database, make sure you have at least one user
! configured in the running-config so that you can login.  To test, open
! a browser to HTTPS://a.b.c.d where a.b.c.d is the IP address on the
! router.
```

Implementing Logging Features

Logging is important as a tool for discovering events that are happening in the network and for troubleshooting. Correctly configuring logging so that you can collect and correlate events across multiple network devices is a critical component for a secure network.

Configuring Syslog Support

Example 6-11 shows a typical syslog message and how to control what information is included with the message.

Example 6-11 *Using Service Time Stamps with Syslog Events*

```
R4(config)# interface fa0/0
R4(config-if)# shut
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to administra-
tively down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed
state to down
R4(config-if)#

! If we add timestamps to the syslog messages, those timestamps can assist it
! correlating events that occurred on multiple devices
```

```

R4(config)# service timestamps log datetime
R4(config)# int fa0/0
R4(config-if)# no shutdown

! These syslog messages have the date of the event, the event (just after
! the %) a description, and also the level of the event. The first is 3,
! the second is 5 in the example shown
*Nov 22 12:08:13: %LINK-3-UPDOWN: Interface FastEthernet0/0, changed state
to up
*Nov 22 12:08:14: %LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/0, changed state to up

```

To configure logging, you just tell CCP what the IP address of your syslog server is and which level of logging you want to do to that IP address. As a reminder, level 7, also known as debug level, sends all syslog alerts at level 7 and lower. To configure logging, navigate to **Configure > Router > Logging**, as shown in Figure 6-5.

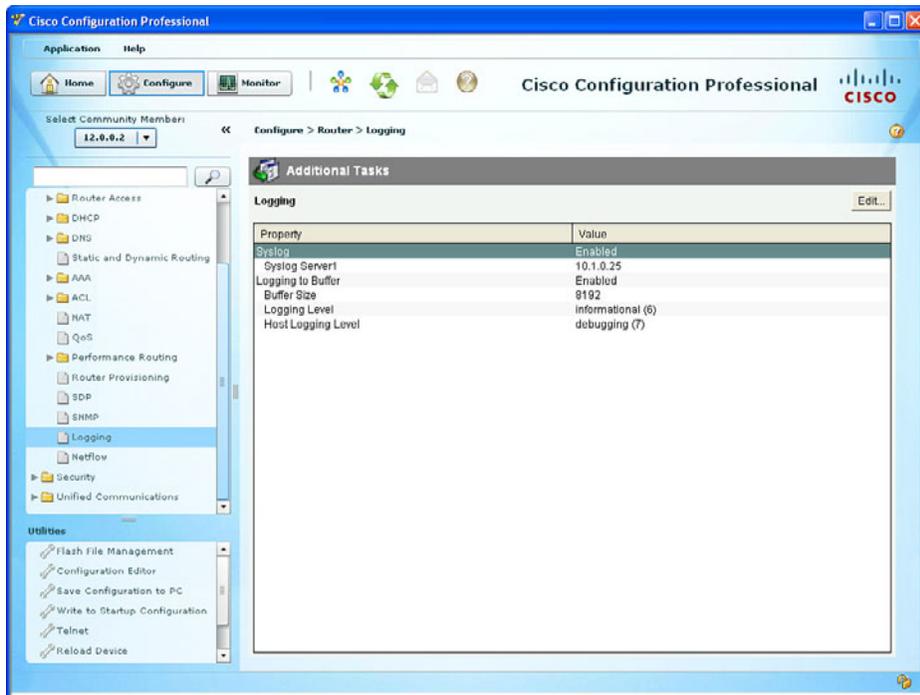


Figure 6-5 Viewing the Logging Configuration

To modify any of the logging settings, click the **Edit** button, as shown in Figure 6-6.

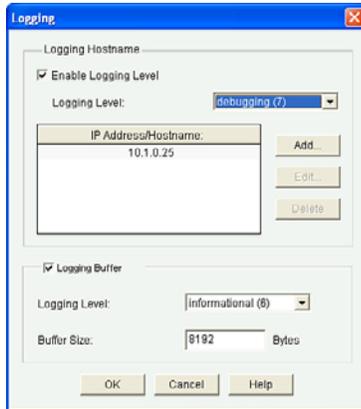


Figure 6-6 Using CCP to Edit the Logging Settings

In Figure 6-6, we have configured level 7 logging (debugging level) to a syslog server at the IP address of 10.1.0.25, and we have specified that the logging level to the buffer on the router is level 6 (informational level). The memory buffer to hold syslog messages is 8192 bytes. Beyond the 8192 bytes worth of messages in memory, any new messages will replace the oldest messages in a *first in, first out (FIFO)* manner. An example of a syslog server is syslog software running on a PC or dedicated server in your network.

The CCP (for the preceding scenario) creates the equivalent output at the CLI, as shown in Example 6-12.

Example 6-12 CLI Equivalent Generated by CCP

```
logging 10.1.0.25
logging trap debugging
logging buffered 8192 informational
```

Figure 6-7 shows the syslog output from the router being collected on the syslog server computer.

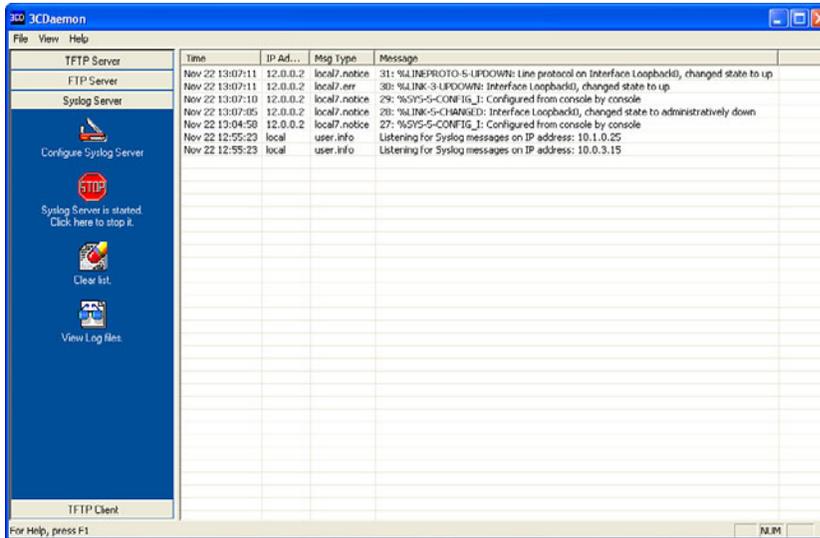


Figure 6-7 Sample Output Viewed on a Syslog Server

SNMP Features



Simple Network Management Protocol (SNMP) has become a de facto standard for network management protocols. The intent of SNMP is to manage network nodes, such as network servers, routers, switches, and so on. SNMP versions range from version 1 to 3, with some intermediate steps in between. The later the version, the more security features it has. Table 6-5 describes some of the components of SNMP.

Table 6-5 Components of SNMPv1 and SNMPv2c Network Management Solutions

Component	Description
SNMP manager	An SNMP manager runs a network management application. This SNMP manager is sometimes called a <i>Network Management Server (NMS)</i> .
SNMP agent	An SNMP agent is a piece of software that runs on a managed device (such as a server, router, or switch).
Management Information Base	Information about a managed device's resources and activity is defined by a series of <i>objects</i> . The structure of these management objects is defined by a managed device's <i>Management Information Base (MIB)</i> . This can be thought of as a collection of unique numbers associated with each of the individual components of a router.

An SNMP manager can send information to, receive request information from, or receive unsolicited information (called a trap) from a managed device (a router). The managed device runs an SNMP agent and contains the MIB.

Even though multiple SNMP messages might be sent between an SNMP manager and a managed device, consider the three broad categories of SNMP message types:

- **GET:** An SNMP GET message is used to retrieve information from a managed device.
- **SET:** An SNMP SET message is used to set a variable in a managed device or to trigger an action on a managed device.
- **Trap:** An SNMP trap message is an unsolicited message sent from a managed device to an SNMP manager. It can be used to notify the SNMP manager about a significant event that occurred on the managed device.

Unfortunately, the ability to get information from or send configuration information to a managed device poses a potential security vulnerability. Specifically, if an attacker introduces a rogue NMS into the network, the attacker's NMS might be able to gather information about network resources by polling the MIBs of managed devices. In addition, the attacker might launch an attack against the network by manipulating the configuration of managed devices by sending a series of SNMP SET messages.

Although SNMP does offer some security against such an attack, the security integrated with SNMPv1 and SNMPv2c is considered weak. Specifically, SNMPv1 and SNMPv2c use *community strings* to gain read-only access/read-write access to a managed device. You can think of a community string much like a password. Also, be aware that multiple SNMP-compliant devices on the market today have a default read-only community string of "public" and a default read-write community string of "private."

The security weaknesses of SNMPv1 and SNMPv2c are addressed in SNMPv3. SNMPv3 uses the concept of a security model and a security level:

- **Security model:** A security model defines an approach for user and group authentications.
- **Security level:** A security level defines the type of security algorithm performed on SNMP packets. Three security levels are discussed here:
 - **noAuthNoPriv:** The noAuthNoPriv (no authentication, no privacy) security level uses community strings for authentication and does not use encryption to provide privacy.
 - **authNoPriv:** The authNoPriv (authentication, no privacy) security level provides authentication using *Hashed Message Authentication Code (HMAC)* with *message digest algorithm 5 (MD5)* or *Secure Hash Algorithm (SHA)*. However, no encryption is used.
 - **authPriv:** The authPriv (authentication, privacy) security level offers HMAC MD5, or SHA authentication and also provides privacy through encryption. Specifically, the encryption uses the *Cipher Block Chaining (CBC) Data Encryption Standard (DES) (DES-56)* algorithm.

As summarized in Table 6-6, SNMPv3 supports all three of the previously described security levels. Notice that SNMPv1 and SNMPv2 support only the noAuthNoPriv security level.

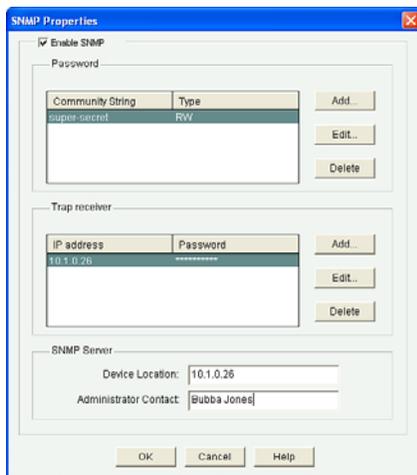
**Table 6-6** *Security Models and Security Levels Supported by Cisco IOS*

Security Model	Security Level	Authentication Strategy	Encryption Type
SNMPv1	noAuthNoPriv	Community string	None
SNMPv2c	noAuthNoPriv	Community string	None
SNMPv3	noAuthNoPriv	Username	None
	authNoPriv	MD5 or SHA	None
	authPriv	MD5 or SHA	CBC-DES (DES-56)

Through the use of the security algorithms, as shown in Table 6-6, SNMPv3 dramatically increases the security of network management traffic as compared to SNMPv1 and SNMPv2c. Specifically, SNMPv3 offers three primary security enhancements:

- **Integrity:** Using hashing algorithms, SNMPv3 can ensure that an SNMP message was not modified in transit.
- **Authentication:** Hashing allows SNMPv3 to validate the source of an SNMP message.
- **Encryption:** Using the CBC-DES (DES-56) encryption algorithm, SNMPv3 provides privacy for SNMP messages, making them unreadable by an attacker who might capture an SNMP packet.

To configure SNMP on the router is simple, especially with CCP. If you know the community strings to use, and the IP address of the SNMP manager, you can configure it on the router by navigating to **Configure > Router > SNMP** and from there use the **Edit** button to add, change, or remove any of the SNMP-related settings. CCP enables command-line editing through the Utilities menu, but currently the SNMP Properties window does not support the configuration of SNMPv3. You can configure the basic SNMPv1 information, as shown in Figure 6-8.

**Figure 6-8** *Using CCP to Configure SNMPv1 Information*

The command-line output for this GUI would look similar to that shown in Example 6-13.

Example 6-13 *Output Created by CCP for Implementing SNMPv1*

```
snmp-server location 10.1.0.26
snmp-server contact Bubba Jones
snmp-server community super-secret RW
snmp-server host 10.1.0.26 trap cisK0tRap^
```

Configuring NTP

Because time is such an important factor, you should use *Network Time Protocol (NTP)* to synchronize the time in the network so that events that generate messages and time stamps can be correlated. You can use CCP to implement the NTP in addition to using the CLI. Let's take a look at both right now.

To configure the NTP, you first need to know what the IP address is of the NTP server you will be working with, and you also want to know what the authentication key is and the key ID. NTP authentication is not required to function, but is a good idea to ensure that the time is not modified because of a rogue NTP server sending inaccurate NTP messages using a spoofed source IP address.

Armed with the NTP server information, in CCP you go to **Configure > Router > Time > NTP and SNTP** and click **Add** and put in the information about the server you will be getting the time from. When done, you click **OK** to close the dialog box. It may take anywhere between 5 and 15 minutes for the router to synchronize its clock. In Figure 6-9, this router is being told that the NTP server is at 55.1.2.3, that it should source the NTP requests from its IP address on its local Fast Ethernet 0/0 interface, and that it should use key number 1, and the password associated with that key. If multiple NTP servers were configured, the Prefer option is used to identify the preference of which NTP server to use.

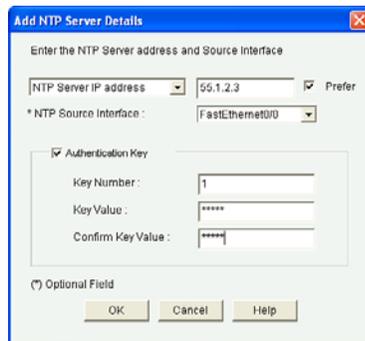


Figure 6-9 *Configuring a Router to Use an NTP Server*

NTP supports authentication on a Cisco router because the router supports NTPv3. Example 6-14 shows the effective equivalent syntax that is created and delivered to the router.

Example 6-14 *Using Authentication via Keys with NTPv3*

```
ntp update-calendar
ntp authentication-key 1 md5 pAs5w0rd!3@
ntp authenticate
ntp trusted-key 1
ntp server 55.1.2.3 key 1 source FastEthernet0/0 prefer
```

To verify the status on this router acting as a NTP client, you could use the commands from the CLI as shown in Example 6-15.

Example 6-15 *Verifying Synchronization from the NTP Client*

```
R2# show ntp status
Clock is synchronized, stratum 4, reference is 55.1.2.3
nominal freq is 250.0000 Hz, actual freq is 250.0000 Hz, precision is 2**24
reference time is D27619E3.7317ACB3 (12:53:55.449 UTC Tue Nov 22 2011)
clock offset is 0.0140 msec, root delay is 0.00 msec
root dispersion is 0.97 msec, peer dispersion is 0.43 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000053 s/s
system poll interval is 64, last update was 130 sec ago.

R2# show ntp association
  address  ref clock      st  when  poll  reach  delay  offset  disp
*~55.1.2.3 127.127.1.1  3   4     64   77    0.000  14.090  190.28
* sys.peer, # selected, + candidate, - outlyer, x falseticker,
~ configured
R2#
```

Note NTP uses UDP port 123. If NTP does not synchronize within 15 minutes, you may want to verify that connectivity exists between this router and the NTP server that it is communicating to. You also want to verify that the key ID and password for NTP authentication are correct

Securing the Cisco IOS Image and Configuration Files

If a router has been compromised, and the flash file system and NVRAM have been deleted, there could be significant downtime as the files are put back in place before restoring normal router functionality. The Cisco Resilient Configuration feature is intended to improve the recovery time by making a secure working copy of the IOS image and startup configuration files (which are referred to as the *primary bootset*) that cannot be deleted by a remote user.

To enable and save the primary bootset to a secure archive in persistent storage, follow Example 6-16.

Example 6-16 *Creating a Secure Bootset*

```

! Secure the IOS image
R6(config)# secure boot-image
%IOS_RESILIENCE-5-IMAGE_RESIL_ACTIVE: Successfully secured running image

! Secure the startup-config
R6(config)# secure boot-config
%IOS_RESILIENCE-5-CONFIG_RESIL_ACTIVE: Successfully secured config archive
[flash:.runcfg-20111222-230018.ar]

! Verify the bootset
R6(config)# do show secure bootset
IOS resilience router id FTX1036A13J

IOS image resilience version 12.4 activated at 23:00:10 UTC Thu Dec 22 2011
Secure archive flash:c3825-advipservicesk9-mz.124-24.T.bin type is image
(elf) []
  file size is 60303612 bytes, run size is 60469256 bytes
  Runnable image, entry point 0x80010000, run from ram

IOS configuration resilience version 12.4 activated at 23:00:18 UTC Thu Dec
22 2011
Secure archive flash:.runcfg-20111222-230018.ar type is config
configuration archive size 1740 bytes

! Note: to undo this feature, (using the "no" option in front of the command)
! you must be connected via the console. This prevents remote users from
! disabling the feature.

```



Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 6-7 lists these key topics.

Table 6-7 *Key Topics*

 Key Topic Element	Description	Page Number
Text	Management plane best practices	95
Text	AAA components	98
Text	Storing usernames, passwords, and access rules	98
Text	Router access authentication	100
Table 6-2	AAA components to secure administrative and remote LAN access	101
Text	The AAA method list	101
Table 6-3	Method list options	101
Text	Limiting the administrator by assigning a view	103
Text	Encrypted management protocols	103
Text	Using logging files	104
Text	User authentication in AAA	108
Text	Using the CLI to troubleshoot AAA for Cisco routers	113
Example 6-4	Using debug commands	113
Example 6-5	Creating and assigning commands to custom privilege levels	118
Text	Implementing parser views	120
Example 6-7	Creating and working with parser views	121
Example 6-9	Preparing for SSH	123
Text	SNMP features	128
Table 6-6	Security models and security levels supported by Cisco IOS	130
Example 6-16	Creating a secure bootset	133

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

AAA, method list, custom privilege level, parser view, SSH, syslog, SNMP, NTP, secure bootset

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 6-8 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 6-8 *Command Reference*

Command	Description
<code>service password-encryption</code>	Encrypt most plaintext passwords in the configuration.
<code>aaa new-model</code>	Enable AAA features.
<code>aaa authentication login default local</code>	Create a default method list for character mode login that will use the local database (running config) on the router or switch.
<code>enable view</code>	Enter the root parser view, from where you can create additional views. This requires that <code>aaa new-model</code> already be in place in the configuration.
<code>privilege exec level 8 show startup-config</code>	Assign a <code>show startup-config</code> command to a custom privilege level 8.
<code>crypto key generate rsa</code>	Create the public/private key pair required for SSH.
<code>secure boot-image</code>	Secure the IOS image on flash
<code>aaa authentication bubba local enable</code>	Create an authentication method list called bubba that will use the local database first, and if the username does not exist, will require the enable secret to allow login.
<code>line console 0</code>	Apply the method list named bubba to the console port.
<code>login authentication bubba</code>	



This chapter covers the following subjects:

- Cisco Secure ACS, RADIUS, and TACACS
- Configuring routers to interoperate with an ACS server
- Configuring the ACS server to interoperate with a router
- Verifying and troubleshooting router-to-ACS server interactions

Implementing AAA Using IOS and the ACS Server

As you learned in the preceding chapter, using *authentication, authorization, and accounting (AAA)* to verify the identity of a user, and what that user is authorized to do, is a great way to secure the management plane on a router or switch. The challenge, however, is that most companies have many network devices. If a single administrator needs access to 10 different routers, and you are using the local database only for the username and password of that administrator (remember, the local database means the running configuration on that specific router), you must create that same user account 10 different times, once on each router. If he ever needs to change the password, it also requires going back to all those 10 devices and manually changing it on each one. This solution does not scale well in environments with multiple administrators and many devices.

A solution to this is to have a centralized database where all the usernames and passwords are kept for authentication and what the individual users are allowed to do (the *authorization* portion of AAA). This is primarily what the *Access Control Server (ACS)* server can provide. It is a two-part process. The first part is to configure on the ACS server information about the users and their passwords and what those users are allowed to do. The second part is to tell the router that it should refer any of its decisions about authentication or authorization to the ACS server.

One other note about the word *users*. Often when we refer to the management plane, and we refer to users, those users are very likely administrators who need access to the *command-line interface (CLI)*. Also be aware that end users will not need CLI access, but will need access to network services and to have their packets allowed through the router. You can use the ACS server to authenticate either type of user, and you can call on it for authorization for these users. In addition, you can use the ACS server as a destination for logging (called accounting), noting which users access the system and what they do while there.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 7-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 7-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Cisco Secure ACS, RADIUS, and TACACS	1–3
Configuring Routers to Interoperate with an ACS Server	4–6
Configuring the ACS Server to Interoperate with a Router	7–8
Verifying and Troubleshooting Router-to-ACS Server Interactions	9–10

1. Which of the following are most likely to be used for authentication of a network administrator accessing the CLI of a Cisco router? (Choose all that apply.)
 - a. TACACS+
 - b. Diameter
 - c. RADIUS
 - d. ACS
2. Which of the following allows for granular control related to authorization of specific Cisco IOS commands that are being attempted by an authenticated and authorized Cisco router administrator?
 - a. RADIUS
 - b. Diameter
 - c. TACACS+
 - d. ISE
3. Which devices or users would be clients of an ACS server? (Choose all that apply.)
 - a. Routers
 - b. Switches
 - c. VPN users
 - d. Administrators
4. On the router, what should be created and applied to a vty line to enforce a specific set of methods for identifying who a user is?
 - a. RADIUS server
 - b. TACACS+ server
 - c. Authorization method list
 - d. Authentication method list
5. What is the minimum size for an effective TACACS+ group of servers?
 - a. 1
 - b. 2

- c.** 5
 - d.** 6
- 6.** With what can you configure AAA on the router? (Choose all that apply.)
- a.** ACS
 - b.** CCP
 - c.** CLI
 - d.** TACACS+
- 7.** Which statement is true for ACS 5.x?
- a.** User groups are nested in network device groups.
 - b.** Authorization policies can be associated with user groups that are accessing specific network device groups.
 - c.** There must be at least one user in a user group.
 - d.** User groups can be used instead of device groups for simplicity.
- 8.** Where in the ACS do you go to create a new group of administrators?
- a.** Users and Identity Stores > Identity Groups
 - b.** Identity Stores > Identity Groups
 - c.** Identity Stores and Groups > Identity Groups
 - d.** Users and Groups > Identity Groups
- 9.** From the router, which method tests the most about the ACS configuration, without forcing you to log in again at the router?
- a.** ping
 - b.** traceroute
 - c.** test aaa
 - d.** telnet
- 10.** Which of the following could likely cause an ACS authentication failure, even when the user is using the correct credentials? (Choose all that apply.)
- a.** Incorrect secret on the ACS
 - b.** Incorrect IP address of the ACS configured on the router
 - c.** Incorrect routing
 - d.** Incorrect filtering between the ACS and the router

Foundation Topics

Cisco Secure ACS, RADIUS, and TACACS

This section discusses how you can use a centralized authentication server such as ACS and the protocols it uses to communicate with its clients, which are routers and switches. This information is relevant to both certification and for the implementation of AAA using ACS.

Why Use Cisco ACS?



Most midsize and large companies using Cisco equipment are also going to use ACS servers so that they can centrally manage the users and control what those users are authorized to do. By configuring users locally on the ACS server, and then having the dozens or hundreds of routers and switches act as clients to the ACS server, you can use the Cisco ACS server as a central clearinghouse for the authentication of users. This way, you can create a user account one time on the ACS server, and configure the routers and switches to use the ACS server for any type of user, whether an administrator trying to access the router for configuration or an end user who just needs access through a router for some network application or service such as browsing the web. If all your network devices use the ACS server, you can avoid having to create that same user account on each of the individual routers' and switches' local database (in their running config).

Most companies using ACS servers have many users, and it is time-consuming to create all the user accounts manually in ACS. One convenient feature of an ACS server is that all the users do not have to be locally configured on the ACS server, either; instead, the ACS server can use an external database that already exists that contains the usernames and passwords. An example is Microsoft Active Directory, where all the users and their credentials are already in place. The chain of events goes something like this: A user connects to a router, and the router prompts the user for authentication. In this example, assume it is an administrator who wants CLI access to the router. The router being configured to use the ACS server prompts the user for his username and password. After getting the username and password, the router sends those credentials to the AAA server (in this case, the ACS server) and waits for a reply. At the ACS server, if it is configured to use an external database such as Microsoft Active Directory, the ACS server makes an inquiry out to Active Directory to validate whether the username and password that the user provided are accurate. If they are, Active Directory can indicate that to the ACS server, and the ACS server in turn can indicate that the credentials are correct back to the router, and then the router can provide the access to the user. If there were no Active Directory, the ACS server would consult its own local configuration to verify the username and password instead of handing it off to Active Directory. That's it in a nutshell. ACS could use multiple external databases for these lookups, and the basic concept is that if the users are already defined in some database, ACS can leverage that database and not have to re-create all users.

What Platform Does ACS Run On?

ACS has a few different flavors, as you learned in a preceding chapter. They include older versions that can be installed on top of an existing Windows server, a dedicated physical appliance can be purchased from Cisco that is installed in a rack at the customer site and has ACS software preinstalled, and the most popular option moving forward is to install the ACS server logically in a VMware environment such as an ESXi server with ACS running as a virtual machine. Regardless of which implementation you choose, the core functionality of having a centralized database of users, along with authorization rules about what users are allowed to do, is the basic premise of ACS.

What Is ISE?

A product called *Identity Services Engine (ISE)* is an identity and access control policy platform that can validate that a computer meets the requirements of a company's policy related to virus definition files, service pack levels, and so on before allowing the device on the network. This solution leverages many AAA-like features, but is not a 100 percent replacement for ACS. For the near future, customers who want the features of ISE will likely use ACS for the authentication and authorization components and use ISE (in addition) for the posturing and policy-compliance checking for hosts.

Protocols Used Between the ACS and the Router

The next couple of sections discuss how to configure the router to forward authentication questions to the ACS server and examine how to tell the ACS server to work with the router. But right now, you need to understand the “language of love” used to communicate between the ACS server and the router (with a router acting as a client to the ACS server).

Two main protocols may be used between the ACS server and its client (such as a router whose using the ACS server to verify authentication requests): TACACS+ (pronounced TACK-AXE, you do not need to say the +), and RADIUS (pronounced RAY-D-US).

TACACS+ stands for *Terminal Access Control Access Control Server*, and that is why we just use the acronym. There have been earlier versions of TACACS+, which had slightly varying names, such as XTACACS and TACACS (without the plus). Because the only version now used is TACACS+, anytime we refer to the term pronounced TACK-AXE, it is accepted and understood that we are referring to the currently implemented TACACS+ (even without saying the + at the end). TACACS+ is Cisco proprietary, which means its primary usage will likely be seen as a protocol used between a Cisco device and a Cisco ACS server. If you configure the router and the ACS server to use TACACS+, all the AAA packets that are sent between the router and the ACS server use the TACACS+ protocol, which encrypts each packet before it is sent on the network.

The other possible protocol that could be used between the router and the ACS server for the purpose of AAA services is RADIUS, which stands for *Remote Authentication Dial-In User Service*. RADIUS is an open standard, which means that not only ACS supports it but also that other vendors' implementations of AAA and their servers (such as



Microsoft) can support communications with a client (such as a router) using this protocol. RADIUS encrypts only passwords, but not the other RADIUS packets being sent between the ACS server and the network device.

Protocol Choices Between the ACS Server and the Client (the Router)

Traditionally, and in common practice, if you are authenticating and authorizing administrators for command-line access, it is likely that you will configure TACACS+ on both the ACS server and the router for their communication with each other. A large reason for this is because TACACS+ has clearly defined and separate techniques and configurations for each aspect of AAA. For example, if you want to tell the router to check authorization for each individual command before allowing an administrator to put that command in, and only give the administrator a subset or portion of commands, TACACS+ and its authorization component allows extremely granular control in communicating which commands would be allowed. RADIUS, however, does not have the same level of granular control as TACACS+ command-by-command authorization.

If you are authenticating and authorizing end users who just want their packets to go through a network device (when authentication and authorization is required), it is likely that you are using RADIUS as the communications method between the ACS server on the router. RADIUS is an open IETF standard and is used by most vendors, including Cisco, when doing AAA for end users. You may configure the router and ACS server to use both TACACS+ and RADIUS simultaneously between the ACS server and its client, the router.

Table 7-2 compares these two protocols.



Table 7-2 *TACACS+ Versus RADIUS*

	TACACS+	RADIUS
Functionality	Separates AAA functions into distinct elements. Authentication is separate from authorization, and both of those are separate from accounting.	Combines many of the functions of authentication and authorization together. Has detailed accounting capability when accounting is configured for use.
Standard	Cisco proprietary, but very well known.	Open standard, and supported by nearly all vendors' AAA implementation.
L4 protocol	TCP.	UDP.
Replacement coming	None officially planned.	Possibly Diameter (named to imply that RADIUS is only half as much, pun intended).

	TACACS+	RADIUS
Confidentiality	All packets are encrypted between ACS server and the router (which is the client).	Only the password is encrypted with regard to packets sent back and forth between the ACS server and the router.
Granular command by command authorization	This is supported, and the rules are defined on the ACS server about which commands are allowed or disallowed.	No explicit command authorization checking rules can be implemented.
Accounting	Provides accounting support.	Provide accounting support, and generally acknowledged as providing more detailed or extensive accounting capability than TACACS+.

Configuring Routers to Interoperate with an ACS Server

This section covers the detailed commands required for a router to use a central authentication server such as ACS. From the earlier discussion, you know that both the ACS server, which will have the usernames and passwords available to it, and the router that will be communicating with the ACS server need to be configured for them to work together. This section examines the router component.

The good news is that most of what you learned in the preceding chapter about AAA on the router still applies here. The biggest difference on the router is that in method lists, the router can be told to use the local database (which you now know is the running config on the local router) for verification of a username and password, or the router can be told to check with an ACS server to ask that server whether or not the username and password are valid.

On the router, you could use the CLI or *Cisco Configuration Professional (CCP)* for the configuration. Because you should know both (and you might need both depending on the certification environment), both methods are covered here. You first learn about the CLI version, followed by CCP. In this section, the configuration is based on a router that has not yet been set up for any type of AAA. As done previously, comments about each of the commands and what their purpose is are included. A few of these may be a review, too, because of the preceding chapter, but a little repetition will help reinforce these concepts.

Another key factor in any implementation is to have a plan, before beginning to configure the router. So, here is the plan. We want the router to implement the following:

- Administrators/users who are accessing the router via the vty lines, regardless if they are using Telnet or *Secure Shell (SSH)*, the router should check with a TACACS+ server (the ACS server using TACACS+ to communicate with this router) for the authentication check (username/password).

- Authenticated users need to be authorized to have access to a CLI (EXEC) session, including the privilege level they should be placed into. The authorization check should be done by the router referring to the ACS server, using TACACS+.

Example 7-1 shows the configuration to implement these objectives.



Example 7-1 *Using the CLI to Configure IOS for Use with ACS*

```

! This command enables the configuration of the rest of the AAA
! If it is in the configuration, it doesn't need to be put in again.
! On most IOS systems, the default has aaa new-model disabled.
R1(config)# aaa new-model

! This authentication method list, when applied to a line such as the VTY
! lines will tell the router to prompt the user who is accessing that line
! for a username and password in order for that user to login.
! When the user supplies the username and password at the login prompt
! the router will send the credentials to a configured TACACS+ server
! and then the server can reply with a pass or fail message.
! This command indicates "group tacacs+" as the first method
! as there could be more than one server configured. If no ACS server
! responds
! after a short timeout the router will then try the second method in the
! method list which is "local" which means the router will then check the
! running
! config to see if there is a username and matching password
R1(config)# aaa authentication login AUTHEN_via_TACACS group tacacs+ local

! This next authorization method list, when applied to a line, will cause
! the router
! to check with the AAA server to verify that the user is authorized
! to gain access to the CLI. The CLI represents an Exec Shell.
! Not only can the ACS indicate to the router whether or not the user is
! authorized
! but it can also indicate what privilege level the user is placed into.
! Both the username and password will need to be created on the ACS server
! for the previous authentication method, and the authorization
! for a CLI will also need to be configured on that same ACS server.
! This authorization list will use one or more configured ACS servers
! via TACACS+, and if there are no servers that respond, then the router
! will check locally regarding whether the command is authorized for this
! user based on privilege level of the user, and privilege level of the
! command being attempted.
R1(config)# aaa authorization exec Author-Exec_via_TACACS group tacacs+
local

```

```

! It is important to note that before we apply either of these method lists
! to the VTY lines, we should create at least one local user as a backup
! in the event the ACS server is unreachable, or not yet configured.
! In the example below it will create a user on the local database of the
! router
! including a username, password as well as a privilege level for that user
R1(config)# username admin privilege 15 secret cisco

! Next we need to create a least one ACS server that the router should try
! to use
! via TACACS+. This is the equivalent of crating a server group of one.
! The password is used as part of the encryption of the packets, and
! whatever
! password we configure here, we also need to configure on the ACS server.
R1(config)# tacacs-server host 192.168.1.252 key cisco123

! Verifying that the IP addresses reachable is a test that can be done
! even before the full ACS configuration is complete on the AAA server
R1(config)# do ping 192.168.1.252

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.252, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 8/13/28 ms

! Next, for the authentication method list and authorization method list
! to be used we would need to apply them. In the example below
! we are applying both method lists to the first five VTY lines.
R1(config)# line vty 0 4
R1(config-line)# authorization exec Author-Exec_via_TACACS
R1(config-line)# login authentication AUTHEN_via_TACACS
! users connecting to these vty lines will now be subject to both authenti-
! cation
! and authorization, based on the lists that are applied to these lines

```

With the authentication and authorization method lists created and applied, you could attempt to log in through one of the five vty lines, and here is what you would expect: You should be prompted for username and password, the router should not be able to successfully contact the ACS server (because you have not configured the ACS part of it yet on that server), and then after a short timeout, the router would use the second method in each of its lists, which indicates to use the local database for the authentication and the authorization. Because you do have a local user with a password and a privilege level assigned to that user, it should work. By enabling a **debug**, and attempting to log in, you can see exactly what is happening, as shown in Example 7-2.

Example 7-2 *Verifying AAA*

```
R1# debug tacacs
TACACS access control debugging is on

! Telnet to an IP address on the local router.
R1# telnet 10.0.0.1
Trying 10.0.0.1 ... Open

TPLUS: Queuing AAA Authentication request 102 for processing
TPLUS: processing authentication start request id 102 TPLUS:
Authentication start packet created for 102()
TPLUS: Using server 192.168.1.252 TPLUS(00000066)/0/
NB_WAIT/6812DC64: Started 5 sec timeout

User Access Verification

! Timing out on TACACS+ regarding authentication because no server is
responding
TPLUS(00000066)/0/NB_WAIT/6812DC64: timed out
TPLUS(00000066)/0/NB_WAIT/6812DC64: timed out, clean up
TPLUS(00000066)/0/6812DC64: Processing the reply packet

! Now moving to the local database on the router
Username: admin
Password: cisco

! Timing out on TACACS+ regarding authorization due to no server responding.
TPLUS: Queuing AAA Authorization request 102 for processing
TPLUS: processing authorization request id 102
TPLUS: Protocol set to None .....Skipping
TPLUS: Sending AV service=shell
TPLUS: Sending AV cmd*
TPLUS: Authorization request created for 102(admin)
TPLUS: Using server 192.168.1.252 TPLUS(00000066)/0/NB_WAIT/
6812DC64: Started 5 sec timeout TPLUS(00000066)/0/NB_WAIT/
6812DC64: timed out TPLUS(00000066)/0/NB_WAIT/6812DC64: timed
out, clean up TPLUS(00000066)/0/6812DC64: Processing the reply
packet

! After timing out, the router again uses its local database for
! authorization and appropriate privilege level for the user.

! If we exit, and change the debugs slightly, and do it again, it will give
! us yet another perspective.
```

```
R1# debug aaa authentication
AAA Authentication debugging is on
R1# debug aaa authorization
AAA Authorization debugging is on

Telnet
R1# telnet 10.0.0.1
Trying 10.0.0.1 ... Open

AAA/BIND(00000067): Bind i/f

! Notice it shows using the authentication list we assigned to the VTY
! lines
AAA/AUTHEN/LOGIN (00000067): Pick method list 'AUTHEN_via_TACACS'
! Not shown here, but indeed the ACS server is timing out, due to not yet
! being configured, which causes the second entry in the list "local" to be
! used.

User Access Verification

Username: admin
Password: cisco

! Now the authorization begins, using the method list we configured for the
! lines
AAA/AUTHOR (0x67): Pick method list 'Author-Exec_via_TACACS'

R1#
AAA/AUTHOR/EXEC(00000067): processing AV cmd=
AAA/AUTHOR/EXEC(00000067): processing AV priv-lvl=15
AAA/AUTHOR/EXEC(00000067): Authorization successful
R1#

!
```

So what has happened so far? Table 7-3 describes the steps to configure the router for ACS.

**Table 7-3** *Configuring the Router to Use ACS via TACACS+*

Task	How to Do It
Decide what the policy should be (for example, which vty lines should require authentication/authorization, and which methods (ACS, local, none) should be used.	This step is done way before you ever begin configuring the router, and is based on your security policy for your network. It is the concept of what you want to accomplish for authentication and authorization.
Enable the ability to configure AAA.	aaa new-model is not enabled by default. If you want to use the services of ACS, you must enable the feature of AAA as the very first step of configuration on a new router.
Specify the address of an ACS server to use.	Use the tacacs-server host command, including the IP address of the ACS server and the password.
Create a named method list for authentication and another for authorization, based on your policy.	Each method list is created in global configuration mode, specifying which methods this list uses, in order, from left to right.
Apply the method lists to the location that should use those methods	In vty line configuration mode, specify the authentication and authorization method lists that you created in the preceding step.

Now that we have implemented your policy using the CLI, let's take a look at implementing a nearly identical policy but this time using CCP to implement it.

For this configuration, we have removed all AAA-related method lists, and have left only the command **aaa new-model**, as a starting point. The intent of this configuration using CCP is to familiarize you with the locations inside the GUI for used to configure the method lists for authentication and authorization and how to apply those to the vty lines.

In the configuration section of CCP, having selected the router that you want to configure, go to **Configure > Router > AAA > AAA Servers and Groups > Servers**, click the **Add** button, and provide the relevant information about your ACS server, as shown in Figure 7-1.

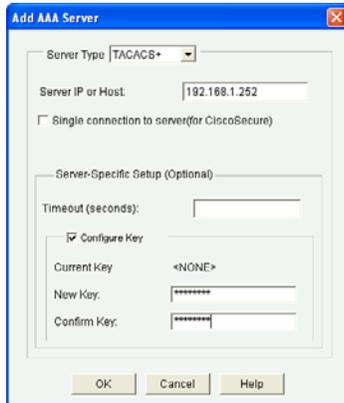


Figure 7-1 Adding an ACS Server to the IOS Router via CCP

You then click **OK** and any other confirmation buttons that are presented to apply this configuration to the router. Now that the router knows the IP address of the ACS server, and which protocol to use and the secret key used to encrypt the packets it sends to the server, the next step is to create the method lists. Just like at the CLI, you create one method list for authentication of logins, and a second method list for authorization of the EXEC session. Each method list specifies that the ACS server should be used first, and if for whatever reason the ACS server fails to respond, the second method the router should use is the local configuration (the running config).

To create the method list for authentication, in CCP you go to **Configure > Router > AAA > Authentication Policies > Login**, click **Add**, and specify the details of the authentication method list, including its name and the methods from top to bottom that this method list will call on. The dialog looks similar to what Figure 7-2 shows.

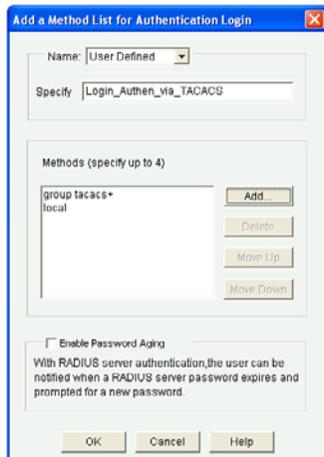


Figure 7-2 Creating an Authentication Method List

From within the pop-up window, the **Add** button enables you to add the individual methods to be used by this method list. There is also the option of moving the methods up or down based on the order you want this method list to call on. As before, you would click the **OK** button and any other confirmation buttons you are prompted with until the configuration is delivered to the router.

Now that your authentication method list is created, you can see this list and any other authentication lists that have been created on this same screen as shown in Figure 7-3. This includes the methods in order from left to right.

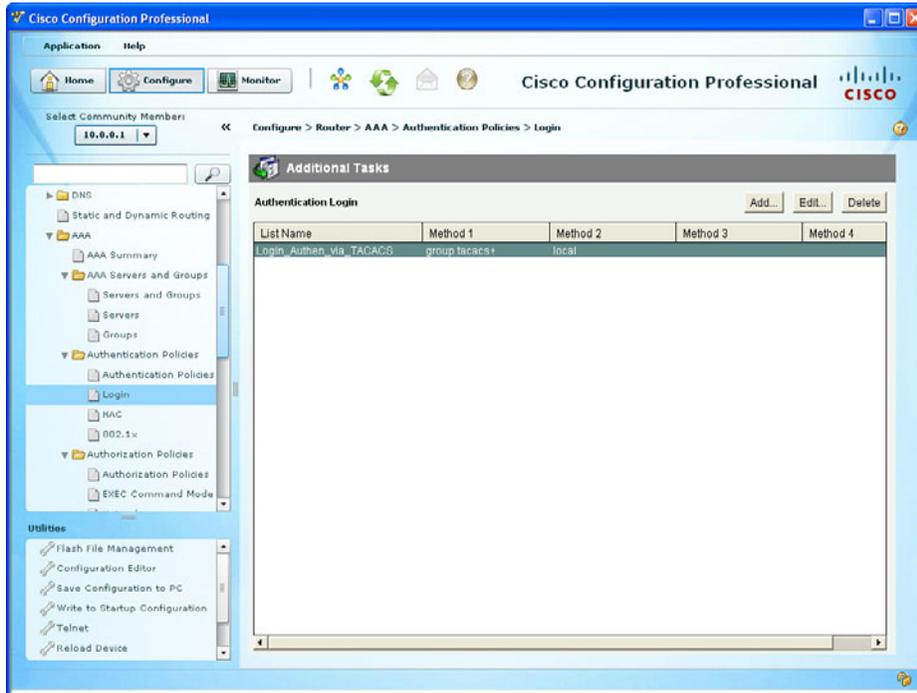


Figure 7-3 Viewing the Configured Authentication Method Lists

Now that you have created your authentication method list, you also need to create an authorization method list based on your policy. Again for this example, we are implementing the same exact policy we did earlier from the command line. To create the authorization method list, go to **Configure > Router > AAA > Authorization Policies > EXEC Command Mode** and click **Add**, as shown in Figure 7-4.

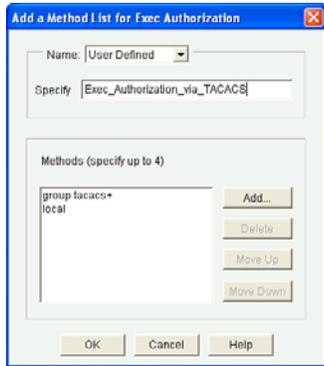


Figure 7-4 *Creating a New Authorization Method List*

Using a similar process as you did with the first method list, you choose **User Defined** (meaning that you are not going to set this method as the global default, but instead are only creating a method list that will not be used until this method list is associated somewhere else in the configuration, such as configured on a vty line). In addition to naming the authorization method list, you would also click the **Add** button from this pop-up to select the individual methods to be used. Just as before, you click the **OK** button and any other confirmation buttons presented until CCP finally delivers the configuration to the router, at which point you can see a summary of your authorization method list, as shown in Figure 7-5.

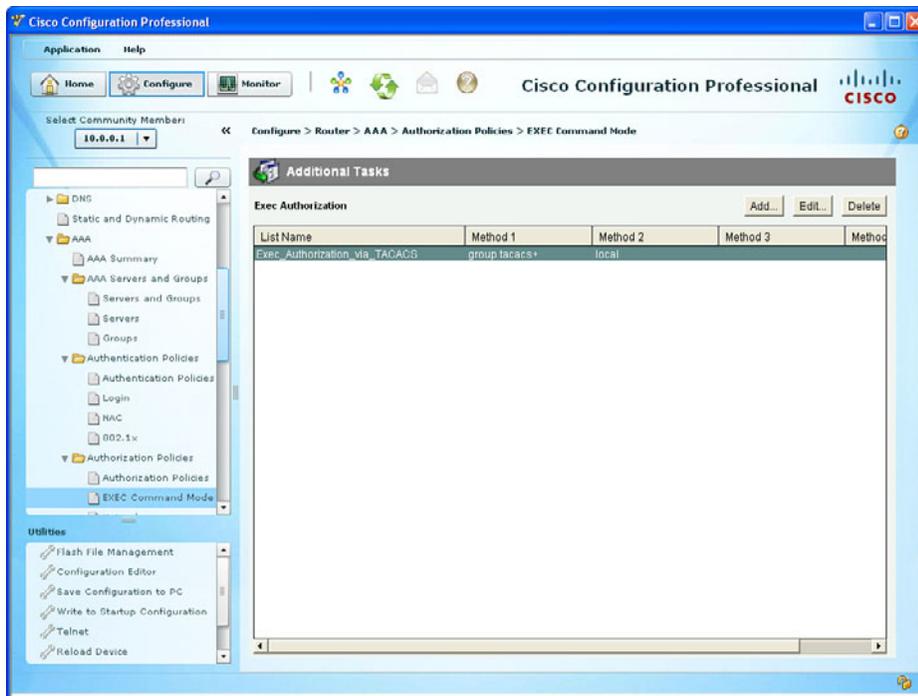


Figure 7-5 *List of Authorization Method Lists Configured on the Router*

At this point, with a method list configured for authentication and a second method list configured for authorization, it is time to put those method lists to work. Now what does that mean? Currently, those method lists are just wasting space in the configuration. If you want those method lists to be used, you need to specifically apply those methods lists. Based on the policy, you want the method list for authentication and the method list for authorization to both be applied to the vty lines of the router.

To apply the method lists, we leave the AAA section and go to **Configure > Router > Router Access > VTY**. From there, click **Edit** and use the drop-down box to select the method lists that we want to use. The only method lists that exist for authentication and authorization are the ones that you create. In this case, we select the authentication and authorization method lists that we previously created, as shown in Figure 7-6.

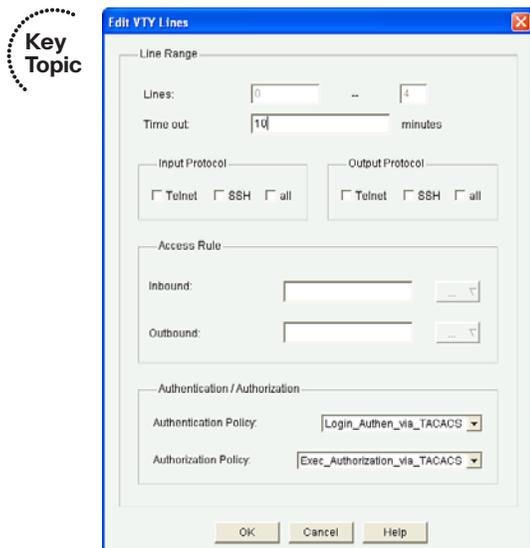


Figure 7-6 *Applying the Newly Created Method Lists*

You click **OK** and then click any confirmation buttons presented until CCP deploys the configuration to the router. Once it deploys the configuration, it shows you a summary of how your vty lines are configured, as shown in Figure 7-7.

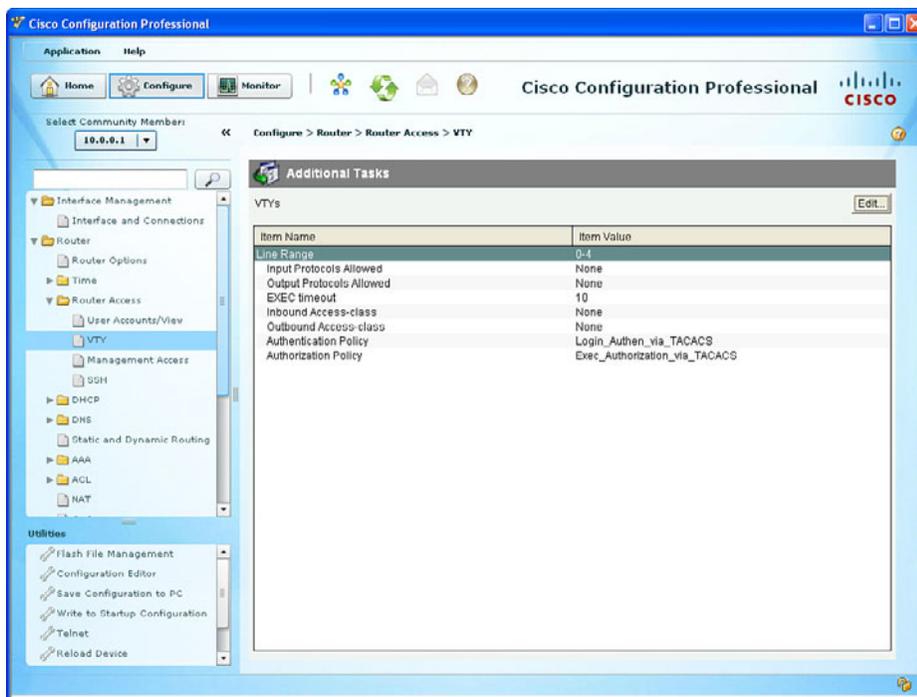


Figure 7-7 Summary of vty Configuration, Showing the Method Lists Applied

At this point in the configuration, you might actually cause yourself some grief if you accidentally log out. If you are connected via the console port, you might be okay because you have not applied the method list to the console port. But if you are connected remotely using Telnet or SSH, which both use the logical vty lines, you might not be able to reconnect if the ACS server is not reachable (because it is not configured yet) and the router falls back to the local configuration and that is where the problem is. We have not yet created a local user in this demonstration. The secret is to make sure there is at least one locally configured user with administrative privileges (privilege level 15) so that you can always get back into the router. You saw earlier how to create a user account from the CLI. You should also do the equivalent here in CCP. To create a local user, go to **Configure > Router > Router Access > User Accounts/View** and click **Add**, as shown in Figure 7-8.



Figure 7-8 Adding a Local User with Privilege Level 15

This user account should have a difficult password, and you might want to consider not giving this username a name that might be recognized or guessed at by a would-be attacker. After you've added the information, including the privilege level 15 access, click **OK** and any other confirmation pop-up confirmation messages until CCP delivers the configuration to the router.

So now you have seen how to configure the router portion for AAA integration within ACS server using TACACS+. The second part of getting AAA working between an ACS server and a router is to configure the ACS portion. And that is what you do in the next section.

Configuring the ACS Server to Interoperate with a Router

This section covers the GUI on the ACS server, which enables it to communicate with a client, such as a router.

Before examining the configuration of the ACS server itself, let's first review a few things. The ACS server has literally thousands of bells and whistles and options that may be configured and tuned. The goal in this section is to make sure that you are comfortable with the basic concept that the ACS server can be a centralized clearinghouse for user authentication/authorization and a repository for accounting records of what those users actually did. From an administrator's perspective, this includes which administrator issued which commands on which devices.

One challenge that large organizations face is having several administrators with different areas of responsibility. For example, one administrator may be responsible for the perimeter routers that are running Zone-Based Firewall services. A different administrator might be responsible for the routers that are providing *virtual private network (VPN)* services, and the list goes on. In situations such as these, it is unwise to give every

administrator full administrative rights to every single router. Instead, it makes sense to provide access only to those individuals who need it. For example, administrators who manage the perimeter routers should not have access, or at least full access, to the VPN devices that they do not manage. In this light, ACS can group the routers together into logical organizations called *device groups*. This way, you put specific routers into a group, and then on the ACS put the administrators who are currently responsible for those routers into a user group and assign that group an authorization role that includes administrative rights of full access for that specific group of routers. This scenario does require a bit more effort for the initial configuration of ACS, but after it is set up you can just add new administrators and put them into specific groups within ACS, and they automatically receive the rights and access levels they need.

Table 7-4 describes the key components for this type of configuration.

Table 7-4 *Key Components for Configuring ACS*

Component of ACS	How It Is Used
Network device groups	Groups of network devices, normally based on routers or switches with similar functions/devices managed by the same administrators.
Network devices (ACS clients/routers/switches)	The individual network devices that go into the device groups.
Identity groups (user/admin groups)	Groups of administrators, normally based on users who will need similar rights and access to specific groups of network devices.
User accounts	Individual administrator/user accounts that are placed in Identity groups.
Authorization profiles	These profiles control what rights are permitted. The profile is associated with a network device group and a user/administrator identity group.



For the demonstration here, we create the following:

- Device group for border routers
- A single router that belongs to the device group
- Two groups, an Admin group and a Monitor group
- Two users (an administrator belonging to the Admin group, and a help desk account belonging to the Monitor group)
- Two authorization policies (the first stating that members of the Admin group who are accessing devices in the device group should get full privilege level 15 access, the second policy stating that users who are members of the Monitor group will only have privilege level 1 access to the devices in the device group)

So with this policy in mind, the first thing is to open a browser window from your local computer to the IP address that is running the ACS server. The URL is `https://a.b.c.d/acsadmin`, where *a.b.c.d* is the actual IP address of your server. On a new installation of ACS, the default password is default. Initially, the ACS server is using an SSL self-signed certificate, and you may get a pop-up asking you whether you want to confirm your session to this device, even though your browser does not trust the certificate. You need to agree and continue if you want to manage the ACS server. Figure 7-9 shows the initial login screen.

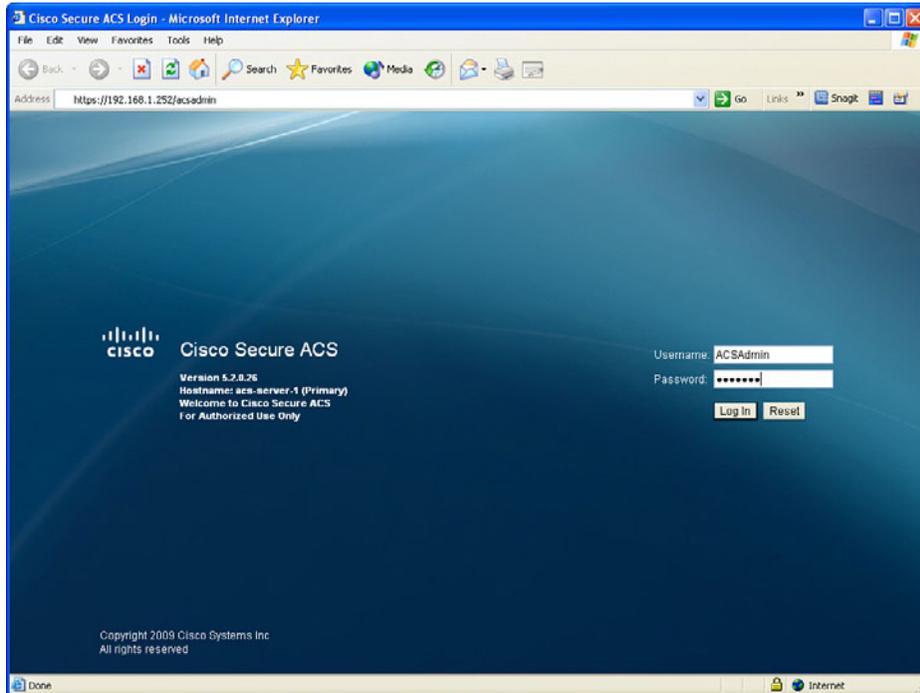


Figure 7-9 Initial Login Screen for ACS

A newly installed ACS also requires proper licensing. The licensing information is provided along with the purchased product. Evaluation licenses are also available for individuals interested in evaluating the product. Contacting your Cisco representative to obtain the software is probably the easiest way to get it, and with your CCO account you can register online for an evaluation license.

The first step is to create a device group. You do so by navigating to **Network Resources > Network Device Groups > Device Type** and clicking **Create**, as shown in Figure 7-10.

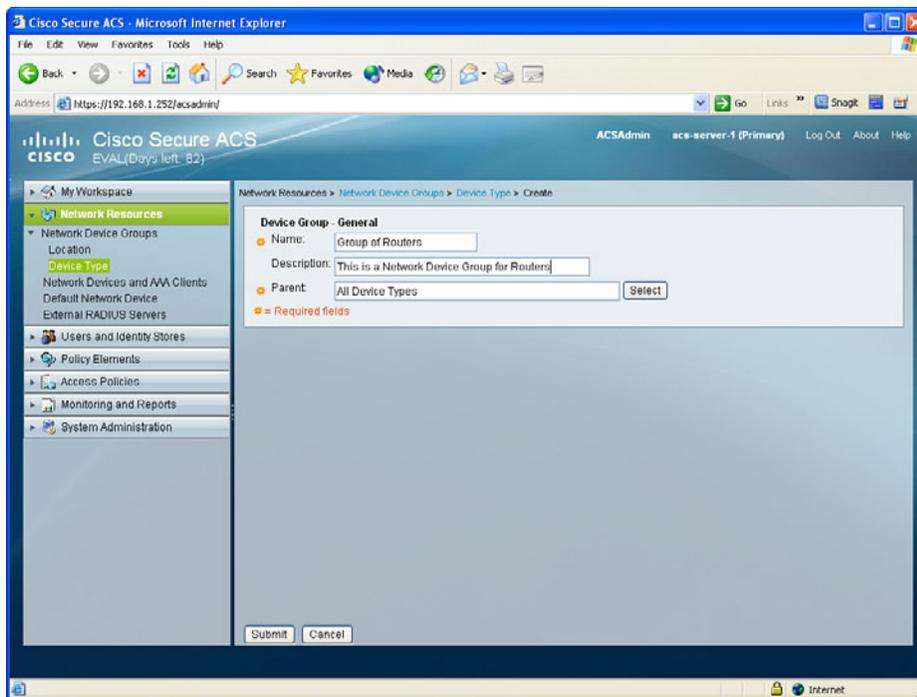


Figure 7-10 *Creating a Network Device Group*

After adding information about this group, click **Submit** to implement the new network device group. The problem with this device group is that by default there are no network devices in it. To fix that, we add as an example a single router (the router we configured earlier) to be included in this network device group on the ACS server. This is done by navigating to **Network Resources > Network Devices and AAA Clients** and clicking **Create**, as shown in Figure 7-11.

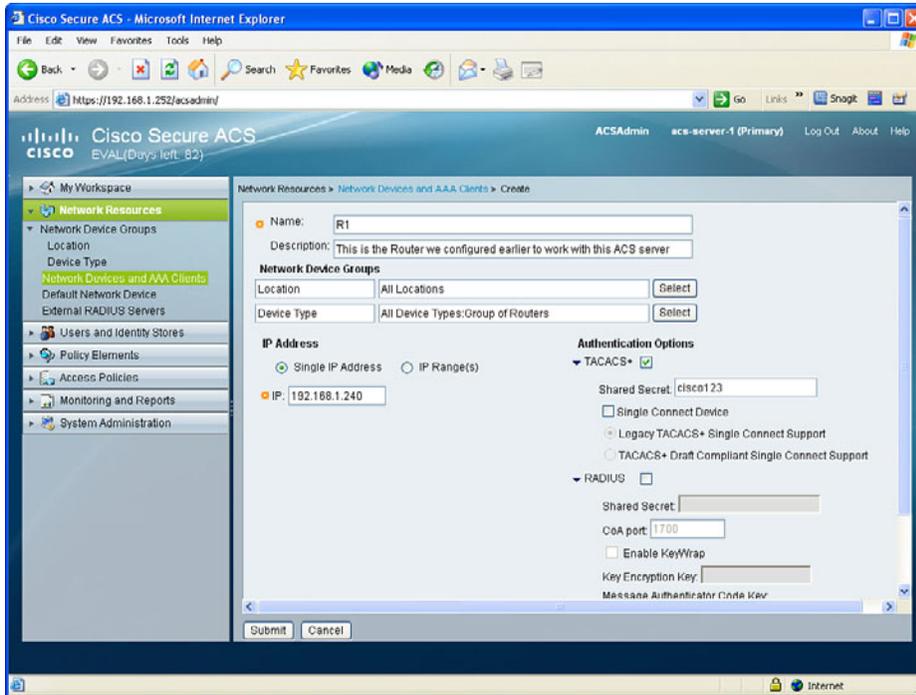


Figure 7-11 Adding a Network Device to the Device Group

In this dialog box, you click **Select** button to the right of the device type and select the device group created from the previous step. In addition to that, you configure the name that the ACS server will know the router by. This name does not have to match the real name of the router, but it is a good idea for it to match so that someone looking at the ACS would know which client (the router) is being referred to in the configuration of the ACS. The IP address of this client (the router) is the reachable address of the router from the perspective of the ACS server. Clicking the box next to TACACS+ lets the ACS server know which protocol to expect from this client, and having the correct password (the one that matches the password configured earlier on the router) is also required for successful communication. After reviewing the information to confirm it is accurate, click **Submit**.

So, we have created a network device group, and added router R1 as the first network device (ACS client) in this group. The next step is to create a user group, and then create some users in those groups. The two groups we are going to create are an Admin group and a Monitor group. To create these groups, navigate to **Users and Identity Stores > Identity Groups** and click **Create**, as shown in Figure 7-12.

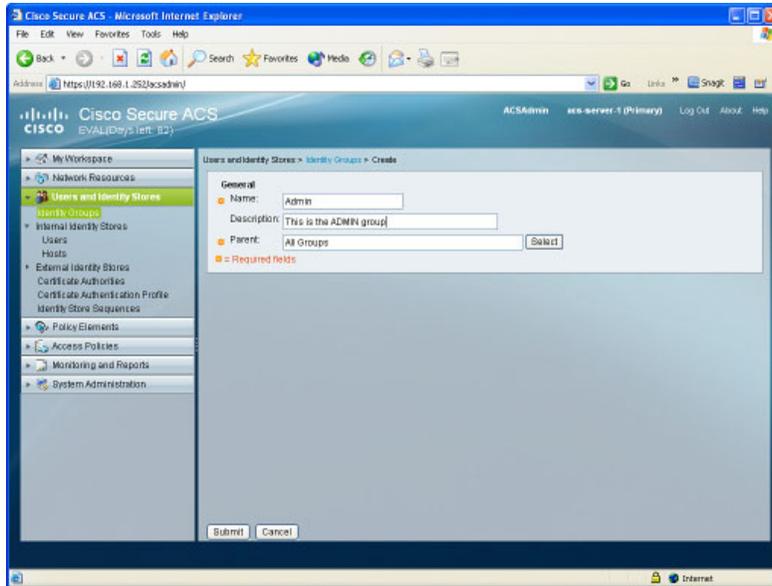


Figure 7-12 *Creating User Groups*

Complete the dialog box by providing the name of the group you are going to create, and then click **Submit**. You could repeat this process for any additional groups. For this discussion, we create two groups: one named Admin and the other named Monitor. After you click **Submit**, a summary of your existing groups displays, as shown in Figure 7-13.

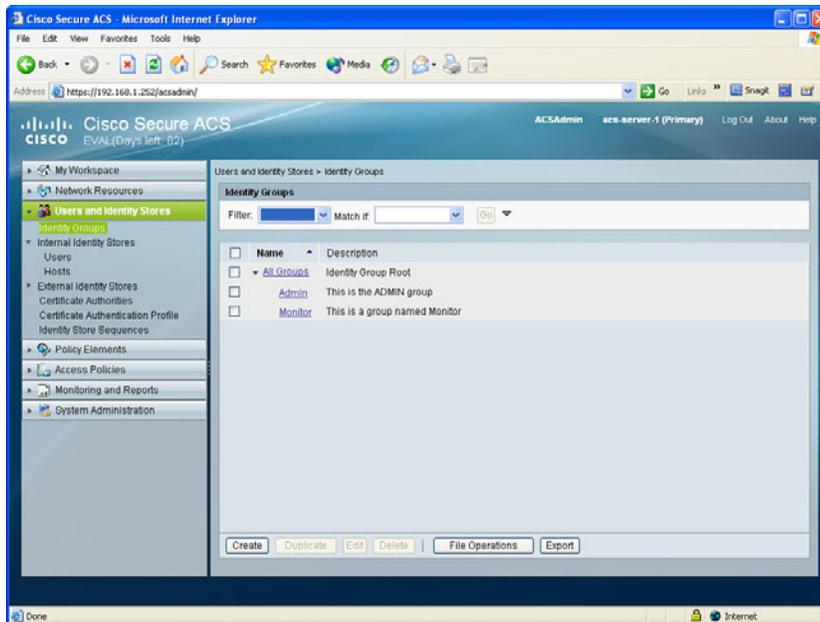


Figure 7-13 *Summary of Identity Groups*

These new groups have no users in them by default, and have no special permissions by default. The first step to fixing that is to create a couple user accounts and place at least one user account into each group. To create individual users, navigate to **Users and Identity Stores > Internal Identity Stores > Users** and click **Create**, as shown in Figure 7-14.

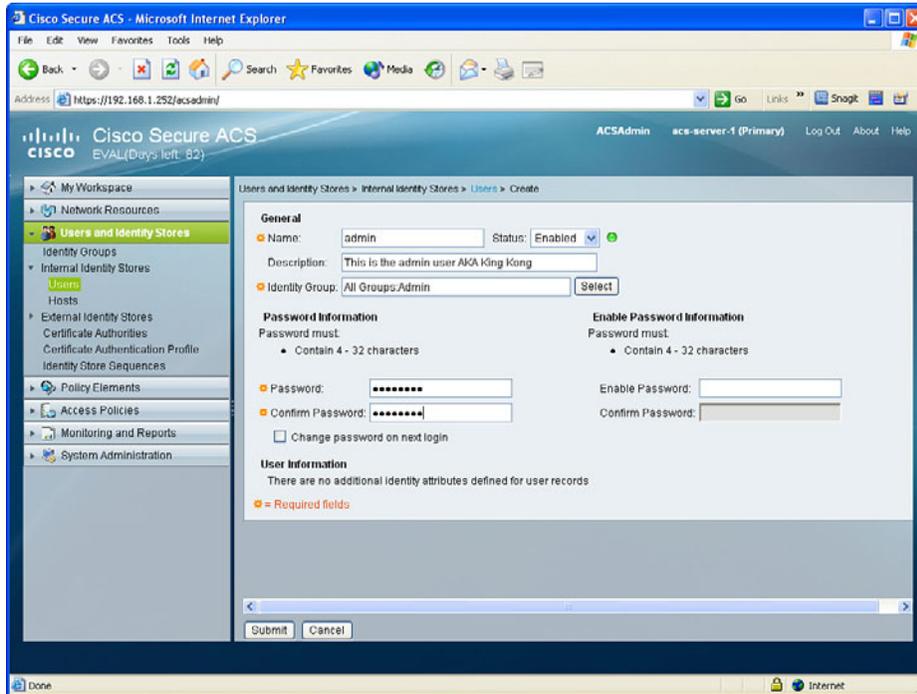


Figure 7-14 *Creating New User Accounts in ACS*

After entering in the name of this user, and a description (if desired), click the **Select** button from this pop-up window to select which user group you want this user to be a member of. It also specifies the password for this administrator. After verifying the details are correct, click **Submit**. In this scenario, we are creating one user named **admin** that belongs to the **Admin** group, and a second user named **help-desk** that belongs to the **Monitor** group. After you click **Submit**, a summary of your configured users configured on the ACS server displays, as shown in Figure 7-15.

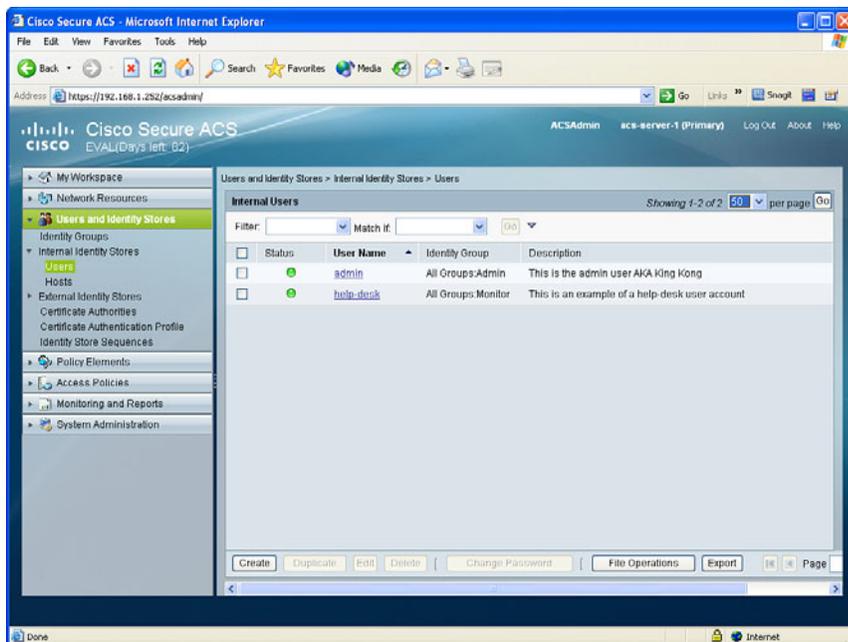


Figure 7-15 *Users Created on the ACS Server*

The next step is to configure authorization policies that give full access to users in the Admin group who are trying to access routers in the network device group we created. We also want to give limited access to users in the Monitor group who are trying to access the same devices. We can do this with authorization policies. To create and assign the reservation policies, first navigate to **Access Policies > Access Services > Default Device Admin > Authorization** and click **Create**, as shown in Figure 7-16.

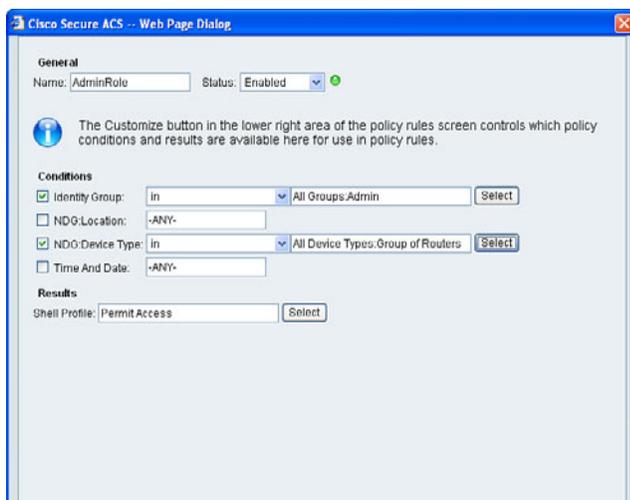


Figure 7-16 *Creating an Authorization Policy*

In the dialog window, indicate the name of this policy, called in this example AdminRole, and check the box next to the conditions next to identity group, and click the **Select** button to choose the admin group created earlier. Use the same process, checking that box next to NDG Device Type (NDG stands for *network device group*) and then using the **Select** button, to indicate the device belongs to the group of routers device group that was created earlier.

This is setting up a condition so that if a user who is a member of the Admin group is attempting to access a device that is a member of the specific router group, then as a result we can provide specific access based on a custom shell profile that we can create. To do that, click the **Select** button next to the Shell Profile option, and you are presented with the screen shown in Figure 7-17.

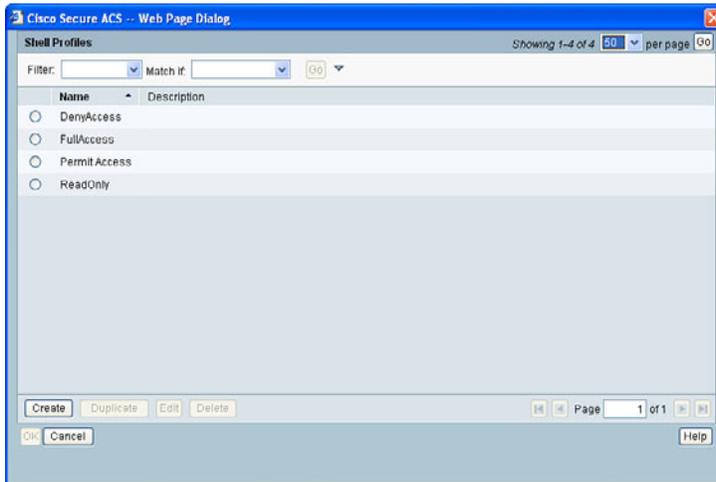


Figure 7-17 *Selecting the Profile to Assign*

You could assign one of the preconfigured profiles, or you could create your own profile and assign it to this group of users. To create a custom profile, click the **Create** button, and from the new window that is brought up name the profile in the dialog box provided, and then display the Common Tasks tab and change the default privilege level to **Static**, and assign the privilege level of **15**, as shown in Figure 7-18.

Click **Submit**, and then confirm any dialog boxes presented to you from ACS until the configuration is applied. By using these steps, any users in the Admin group accessing any of the devices in the specified device group will not only be able to authenticate but also be automatically authorized for and placed into privilege level 15 after successfully authenticating on those routers. We would repeat this process for the Monitor group, assigning a static privilege level of 1.

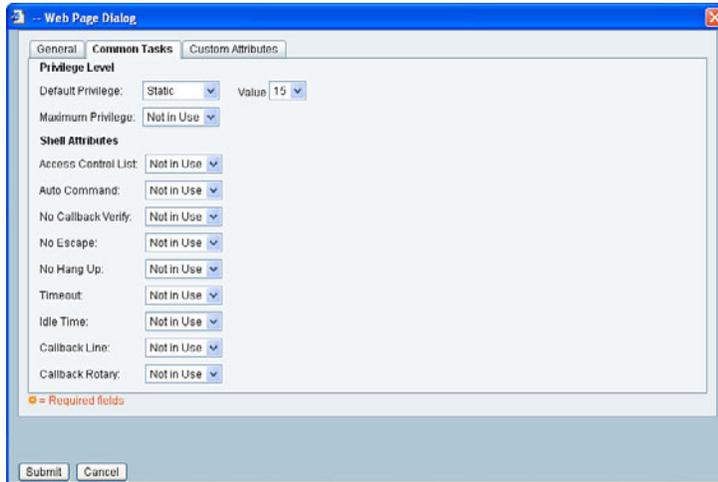


Figure 7-18 *Creating a Custom Authorization Profile*

After saving the changes, you can view a summary of the authorization profiles in this same location. Figure 7-19 shows two custom authorization profiles. One applies for admin users in the Admin group accessing devices in the router group, the other applies for help desk users who are members of the Monitor group accessing the same devices.

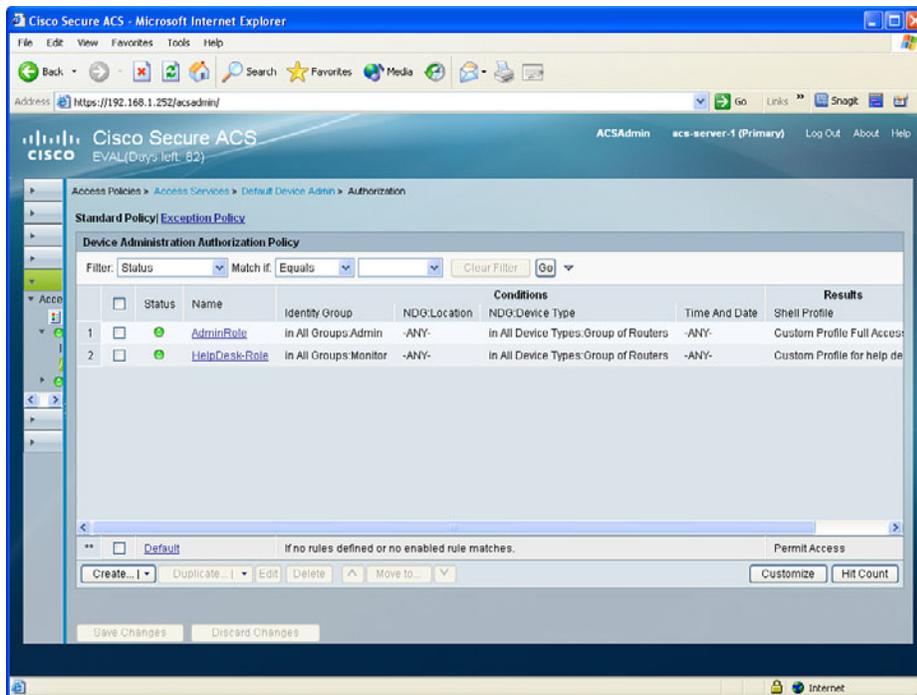


Figure 7-19 *Custom Authorization Policies*

In this section, we created device groups and added individual routers, or a least one in this case, to that device group. We also created user groups and put users (in this case, at least one per group) into those groups. We then created custom authorization profiles that indicate which profiles to be applied based on which users in which groups are accessing which devices. The final piece to the puzzle is to verify that it actually works. Let's do that right now in the next section.

Verifying and Troubleshooting Router-to-ACS Server Interactions

This section discusses the commands that enable you to verify/troubleshoot AAA when the router is using the ACS server to authenticate or authorize the users who are trying to connect to the router.

The chances that everything is configured perfectly the very first time on both the router and the ACS server to allow the router to call upon the ACS server for authentication of users and authorization of users are not very good. The good news is that after some practice and good documentation skills implementing ACS and Cisco router configurations your ability will improve. Whether you are an experienced veteran or brand-new to ACS, the tools covered right now will prove helpful in troubleshooting and verifying the configuration.

Back at the router, one of the first things you might want to do if you have not done so already is verify that you have reachability between the router and the ACS server. You might want to consider using ping to verify the connectivity, as shown in Example 7-3.

Example 7-3 Verifying Basic Connectivity

```
R1# ping 192.168.1.252

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.252, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/21/32 ms
R1#
```

If the ping was not successful, it could be due to access control filtering that is denying *Internet Control Message Protocol (ICMP)* between the router and the ACS server, the ACS server may physically be powered off or its network cable may be disconnected, the ACS server may be connected to a switch port that is misconfigured and is in the wrong VLAN, or it may be a general routing issue or the network is not fully converged or able to route correctly. Verifying the basic routing and connectivity is a fantastic start, and after that is in place, here is the very next tool you should use, called **test** (see Example 7-4).

Example 7-4 *Testing AAA Between the Router and the ACS*

```
R1# test aaa group tacacs+ admin cisco123 legacy
Attempting authentication test to server-group tacacs+ using tacacs+
User was successfully authenticated.
```



In the syntax for the AAA test, we include the group (in this case, a group of one TACACS+ server and a username and the password for that user). The keyword **legacy** is also used as part of the syntax for the test. This is a cool tool because it enables you to verify that the ACS to router authentication component is working, before testing your authentication method list with Telnet. Another great thing to do when troubleshooting is to look at the reports on the ACS server that may indicate a reason as to why a problem occurred. You can find these reports by navigating to **Monitoring & Reports > Reports > Favorites**. Figure 7-20 shows an example.

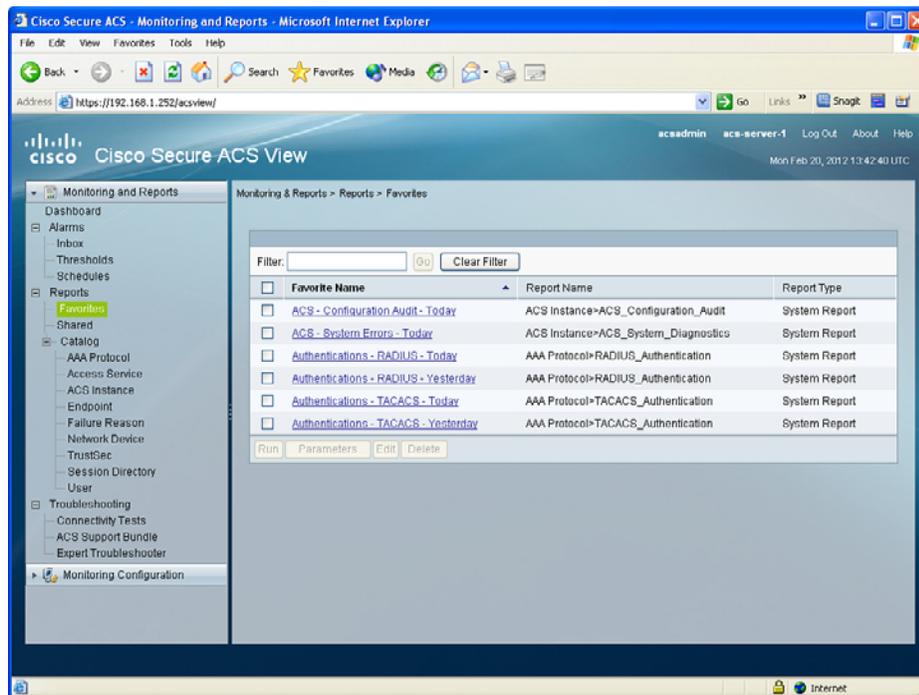


Figure 7-20 *Reporting Options from Within ACS*

From here, click the **Authentications – TACACS – Today** link for information and indications as to why errors may be occurring, as shown in Figure 7-21.

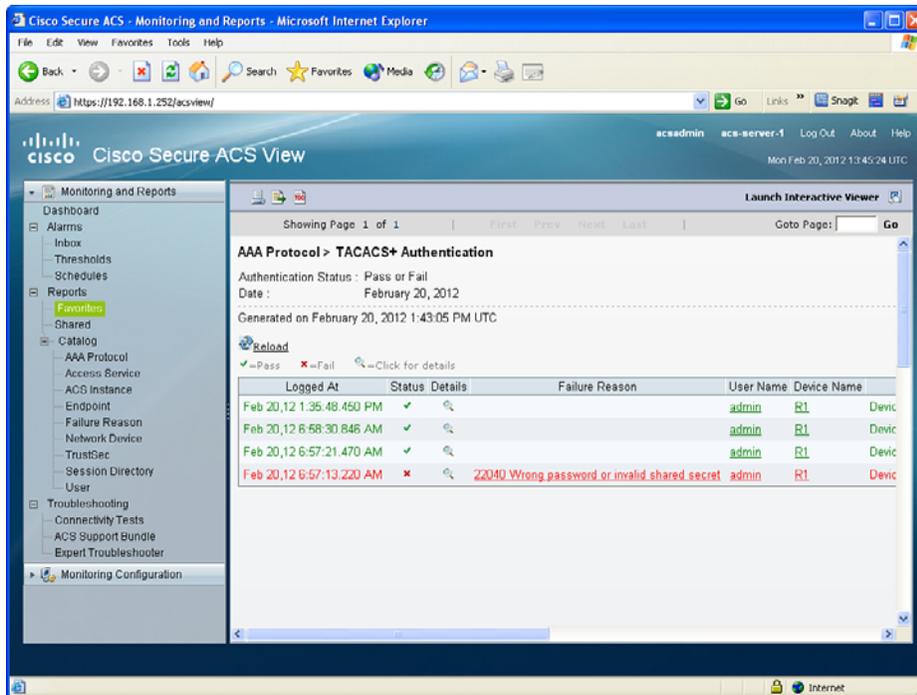


Figure 7-21 Detailed Error Messages from ACS

One common occurrence is that after the reports are looked at, there are no error messages about the ACS client (the router) that we believe is trying to use the ACS. In cases such as these, you want to verify no filters are blocking the traffic from the router to the ACS and vice versa, and verify that in the router config it has the correct IP address of the ACS server. If the router does not have the correct IP address of the ACS server, there will never be any records on the ACS server about that misconfigured router.

Now that we know we have functional AAA connectivity between the router and the server, let's test the method lists for authentication and authorization that we placed on the vty lines.

A simple Telnet to that router can do the job. It is often easier to start on the router, perhaps from a console port, and telnet back to that same router. A Telnet session, regardless of its source, should trigger the authentication, and with some **debug** commands in place, we can verify that ACS is working correctly. In this example, we use a login from a remote workstation and look at the **debug** messages on the console of the router, as shown in Example 7-5.

Example 7-5 Using debug Commands to Verify Functionality

```
! Verifying what debugging is currently in place on the router
R1# show debug
General OS:
    TACACS access control debugging is on
```

```
AAA Authentication debugging is on
AAA Authorization debugging is on

! on a remote machine, we telnet and authenticate as the user admin, and
! simply view the debug output on the console of the router receiving the
! telnet session

R1#
AAA/BIND(00000083): Bind i/f

! the session came in on a VTY line, which triggered the authentication
! method list associated with that line
AAA/AUTHEN/LOGIN (00000083): Pick method list 'Login_Authen_via_TACACS'
TPLUS: Queuing AAA Authentication request 131 for processing
TPLUS: processing authentication start request id 131
TPLUS: Authentication start packet created for 131()
TPLUS: Using server 192.168.1.252

! Sending a TACACS+ request to contact the server
TPLUS(00000083)/0/NB_WAIT/68BD742C: Started 5 sec timeout
TPLUS(00000083)/0/NB_WAIT: socket event 2
TPLUS(00000083)/0/NB_WAIT: wrote entire 33 bytes request
TPLUS(00000083)/0/READ: socket event 1
TPLUS(00000083)/0/READ: Would block while reading
TPLUS(00000083)/0/READ: socket event 1
TPLUS(00000083)/0/READ: read entire 12 header bytes (expect 16 bytes data)
R1#
TPLUS(00000083)/0/READ: socket event 1
TPLUS(00000083)/0/READ: read entire 28 bytes response

! Router got a message back from ACS
! Router will now prompt the user for their username
TPLUS(00000083)/0/68BD742C: Processing the reply packet
TPLUS: Received authen response status GET_USER (7)
R1#
TPLUS: Queuing AAA Authentication request 131 for processing
TPLUS: processing authentication continue request id 131
TPLUS: Authentication continue packet generated for 131
TPLUS(00000083)/0/WRITE/68BD742C: Started 5 sec timeout
TPLUS(00000083)/0/WRITE: wrote entire 22 bytes request
TPLUS(00000083)/0/READ: socket event 1
TPLUS(00000083)/0/READ: read entire 12 header bytes (expect 16 bytes data)
TPLUS(00000083)/0/READ: socket event 1
TPLUS(00000083)/0/READ: read entire 28 bytes response
TPLUS(00000083)/0/68BD742C: Processing the reply packet
```

```
! Router will now prompt user for the user password
TPLUS: Received authen response status GET_PASSWORD (8)
R1#
TPLUS: Queuing AAA Authentication request 131 for processing
TPLUS: processing authentication continue request id 131
TPLUS: Authentication continue packet generated for 131 TPLUS(00000083)/
0/WRITE/68BD742C: Started 5 sec timeout TPLUS(00000083)/0/WRITE: wrote
entire 25 bytes request TPLUS(00000083)/0/READ: socket event 1
TPLUS(00000083)/0/READ: read entire 12 header bytes (expect 6 bytes data)
TPLUS(00000083)/0/READ: socket event 1
TPLUS(00000083)/0/READ: read entire 18 bytes response TPLUS(00000083)/
0/68BD742C: Processing the reply packet

! The ACS server said YES to the username/password combination.
TPLUS: Received authen response status PASS (2)

! The router now begins the authorization process for the user
! using the authorization methods in the list associated with the VTY lines
AAA/AUTHOR (0x83): Pick method list 'Exec_Authorization_via_TACACS'
TPLUS: Queuing AAA Authorization request 131 for processing
TPLUS: processing authorization request id 131
TPLUS: Protocol set to None . . .Skipping
TPLUS: Sending AV service=shell
TPLUS: Sending AV cmd*
TPLUS: Authorization request created for 131(admin)
TPLUS: using previously set server 192.168.1.252 from group tacacs+
TPLUS(00000083)/0/NB_WAIT/68BD742C: Started 5 sec timeout TPLUS(00000083)/
0/NB_WAIT: socket event 2
TPLUS(00000083)/0/NB_WAIT: wrote entire 57 bytes request TPLUS(00000083)/
0/READ: socket event 1
TPLUS(00000083)/0/READ: Would block while reading TPLUS(00000083)/0/READ:
socket event 1
TPLUS(00000083)/0/READ: read entire 12 header bytes (expect 18 bytes data)
TPLUS(00000083)/0/READ: socket event 1
TPLUS(00000083)/0/READ: read entire 30 bytes response TPLUS(00000083)/
0/68BD742C: Processing the reply packet

! Got the reply from the ACS server saying yes to authorization
! and that the user should be placed at privilege level 15
TPLUS: Processed AV priv-lvl=15
TPLUS: received authorization response for 131: PASS AAA/
AUTHOR/EXEC(00000083): processing AV cmd= AAA/AUTHOR/
EXEC(00000083): processing AV priv-lvl=15 AAA/AUTHOR/
EXEC(00000083): Authorization successful
R1#
```

```

R1# show users
      Line      User      Host(s)      Idle      Location
  2 vty 0      admin      idle         00:00:51 10.0.0.25

! We could do the same test again, except this time, login as the user
! "help-desk"
! The results will be nearly identical, with the exception that the user
! will be provided with an exec shell (CLI) at privilege level 1

R1#
AAA/BIND(00000084): Bind i/f
AAA/AUTHEN/LOGIN (00000084): Pick method list 'Login_Authen_via_TACACS'
TPLUS: Queuing AAA Authentication request 132 for processing
TPLUS: processing authentication start request id 132
TPLUS: Authentication start packet created for 132()
TPLUS: Using server 192.168.1.252
TPLUS(00000084)/0/NB_WAIT/68793774: Started 5 sec timeout
TPLUS(00000084)/0/NB_WAIT: socket event 2
TPLUS(00000084)/0/NB_WAIT: wrote entire 33 bytes request
TPLUS(00000084)/0/READ: socket event 1
TPLUS(00000084)/0/READ: Would block while reading
TPLUS(00000084)/0/READ: socket event 1
TPLUS(00000084)/0/READ: read entire 12 header bytes (expect 16 bytes data)
R1#
TPLUS(00000084)/0/READ: socket event 1
TPLUS(00000084)/0/READ: read entire 28 bytes response
TPLUS(00000084)/0/68793774: Processing the reply packet
TPLUS: Received authen response status GET_USER (7)
R1#
TPLUS: Queuing AAA Authentication request 132 for processing
TPLUS: processing authentication continue request id 132
TPLUS: Authentication continue packet generated for 132
TPLUS(00000084)/0/WRITE/68793774: Started 5 sec timeout
TPLUS(00000084)/0/WRITE: wrote entire 26 bytes request
TPLUS(00000084)/0/READ: socket event 1
TPLUS(00000084)/0/READ: read entire 12 header bytes (expect 16 bytes data)
TPLUS(00000084)/0/READ: socket event 1
TPLUS(00000084)/0/READ: read entire 28 bytes response
TPLUS(00000084)/0/68793774: Processing the reply packet
TPLUS: Received authen response status GET_PASSWORD (8)
R1#
TPLUS: Queuing AAA Authentication request 132 for processing

```

```

TPLUS: processing authentication continue request id 132
TPLUS: Authentication continue packet generated for 132 TPLUS(00000084)/
0/WRITE/68793774: Started 5 sec timeout TPLUS(00000084)/0/WRITE: wrote
entire 25 bytes request TPLUS(00000084)/0/READ: socket event 1
TPLUS(00000084)/0/READ: read entire 12 header bytes (expect 6 bytes data)
TPLUS(00000084)/0/READ: socket event 1
TPLUS(00000084)/0/READ: read entire 18 bytes response TPLUS(00000084)/
0/68793774: Processing the reply packet
TPLUS: Received authen response status PASS (2)
AAA/AUTHOR (0x84): Pick method list 'Exec_Authorization_via_TACACS'
TPLUS: Queuing AAA Authorization request 132 for processing
TPLUS: processing authorization request id 132
TPLUS: Protocol set to None .....Skipping
TPLUS: Sending AV service=shell
TPLUS: Sending AV cmd*
TPLUS: Authorization request created for 132(help-desk)
TPLUS: using previously set server 192.168.1.252 from group tacacs+
TPLUS(00000084)/0/NB_WAIT/68793774: Started 5 sec timeout TPLUS(00000084)/
0/NB_WAIT: socket event 2
TPLUS(00000084)/0/NB_WAIT: wrote entire 61 bytes request TPLUS(00000084)/
0/READ: socket event 1
TPLUS(00000084)/0/READ: Would block while reading TPLUS(00000084)/0/READ:
socket event 1
TPLUS(00000084)/0/READ: read entire 12 header bytes (expect 17 bytes data)
TPLUS(00000084)/0/READ: socket event 1
TPLUS(00000084)/0/READ: read entire 29 bytes response TPLUS(00000084)/
0/68793774: Processing the reply packet
TPLUS: Processed AV priv-lvl=1
TPLUS: received authorization response for 132: PASS AAA/AUTHOR/
EXEC(00000084): processing AV cmd= AAA/AUTHOR/EXEC(00000084): processing
AV priv-lvl=1 AAA/AUTHOR/EXEC(00000084): Authorization successful
R1#

R1# show users
      Line      User      Host(s)      Idle      Location
  2 vty 0      help-desk  idle          00:01:24  10.0.0.25

```

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 7-5 lists these key topics.

Table 7-5 *Key Topics*

Key Topic Element	Description	Page Number
Text	Why use Cisco ACS	140
Text	Protocols used between the ACS and the router	141
Table 7-2	TACACS+ versus RADIUS	142
Example 7-1	Using the CLI to configure IOS for use with ACS	144
Table 7-3	Configuring the router to use ACS via TACACS+	148
Figure 7-6	Applying the newly created method lists	152
Table 7-4	Key components for configuring ACS	155
Example 7-4	Testing AAA between the router and the ACS	165



Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

ACS, RADIUS, TACACS+, AAA server, authentication method list, authorization method list

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 7-6 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 7-6 *Command Reference*

Command	Description
aaa new-model	Enable the configuration of method lists and other AAA-related elements, including the use of ACS.
test aaa group tacacs+ admin cisco123 legacy	Allow verification of the authentication function working between the AAA client (the router) and the ACS server (the AAA server).
aaa authentication login MYLIST1 group tacacs+ local	Create an authentication method list, that when applied elsewhere in the configuration, requests the services of an ACS server via TACACS+, and if no server responds, the next method “local” (which is the local router configuration) is checked to verify the credentials of the user.
aaa authorization exec MYLIST2 group tacacs+ none	Create an authorization method list, that when applied to a vty line, requests the services of an ACS server (via TACACS+). If no server responds, the second method “none” is used. This result in no username prompt being provided to the user, and authentication is not required.
tacacs-server host 192.168.1.252 key cisco123	Places a server into the “group” of ACS servers the router can use for TACACS+ requests. It includes the IP address and the secret used to encrypt packets between this router (the client) and the ACS server.

This page intentionally left blank



This chapter covers the following subjects:

- VLAN and trunking fundamentals
- Spanning-Tree fundamentals
- Common Layer 2 threats and how to mitigate them

Securing Layer 2 Technologies

We often take for granted Layer 2 in the network because it just works. *Address Resolution Protocol (ARP)* and Layer 2 forwarding on Ethernet are all proven technologies that work very well. This certification, the CCNA Security, was built with the presumption that candidates would have a CCNA in route/switch or equivalent knowledge. With this knowledge, your understanding of the details about VLANs, trunking, and inter-VLAN routing is presumed. However, so that you absolutely understand these fundamental concepts, this chapter begins with a review.

The first two sections of this chapter deal with ARP and DHCP. It is important to make sure that the basics are in place so that you can fully understand the discussion about protecting Layer 2 in the last section of this chapter, which covers the really important “stuff.” That section focuses on just a few Layer 2-related security vulnerabilities and explains exactly how to mitigate threats at Layer 2. If you are currently comfortable with VLANs, trunking, and routing between VLANs, you might want to jump right to the last section.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 8-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 8-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
VLAN and Trunking Fundamentals	1, 6–7
Spanning-Tree Fundamentals	2
Common Threats and How to Mitigate Them	3–5, 8–10

- 1.** Which is the primary Layer 2 mechanism that allows multiple devices in the same VLAN to communicate with each other even though those devices are physically connected to different switches?
 - a.** IP address
 - b.** Default gateway
 - c.** Trunk
 - d.** 802.1D
- 2.** How does a switch know about parallel Layer 2 paths?
 - a.** 802.1Q
 - b.** BPDU
 - c.** CDP
 - d.** NTP
- 3.** When implemented, which of the following helps prevent CAM table overflows?
 - a.** 802.1w
 - b.** BPDU guard
 - c.** Root guard
 - d.** Port security
- 4.** Which of the following is *not* a best practice for security?
 - a.** Leaving the native VLAN as VLAN 1
 - b.** Shutting down all unused ports and placing them in an unused VLAN
 - c.** Limiting the number of MAC addresses learned on a specific port
 - d.** Disabling negotiation of switch port mode
- 5.** What is the default number of MAC addresses allowed on a switch port that is configured with port security?
 - a.** 1
 - b.** 5
 - c.** 15
 - d.** Depends on the switch model

6. Which two items normally have a one-to-one correlation?
 - a. VLANs
 - b. Classful IP networks
 - c. IP subnetworks
 - d. Number of switches
 - e. Number of routers
7. What is a typical method used by a device in one VLAN to reach another device in a second VLAN?
 - a. ARP for the remote device's MAC address
 - b. Use a remote default gateway
 - c. Use a local default gateway
 - d. Use trunking on the PC
8. Which two configuration changes prevent users from jumping onto any VLAN they choose to join?
 - a. Disabling negotiation of trunk ports
 - b. Using something else other than VLAN 1 as the "native" VLAN
 - c. Configuring the port connecting to the client as a trunk
 - d. Configuring the port connecting to the client as an access port
9. If you limit the number of MAC addresses learned on a port to five, what benefits do you get from the port security feature? (Choose all that apply.)
 - a. Protection for DHCP servers against starvation attacks
 - b. Protection against IP spoofing
 - c. Protection against VLAN hopping
 - d. Protection against MAC address spoofing
 - e. Protection against CAM table overflow attacks
10. Why should you implement root guard on a switch?
 - a. To prevent the switch from becoming the root
 - b. To prevent the switch from having any root ports
 - c. To prevent the switch from having specific root ports
 - d. To protect the switch against MAC address table overflows

Foundation Topics

VLAN and Trunking Fundamentals

You must understand the basics of how VLANs and trunking operate before you can learn to secure those features. This section reviews how VLANs and trunking are configured and how they operate.

Figure 8-1 serves as a reference for the discussion going forward. You might want to bookmark this page or take a moment to make a simple drawing of the topology. You will want to refer to this illustration often during the discussion.

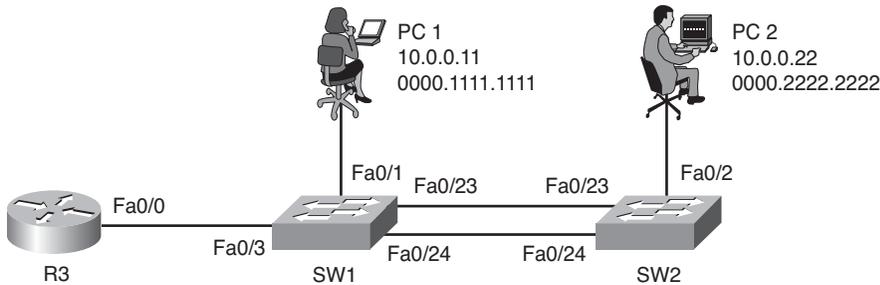


Figure 8-1 Base Topology for Most of the Discussion in This Chapter

What Is a VLAN?



One way to identify a local-area network is to say that all the devices in the same LAN have a common Layer 3 IP network address and that they also are all located in the same Layer 2 broadcast domain. A *virtual LAN (VLAN)* is another name for a *Layer 2 broadcast domain*. VLANs are controlled by the switch. The switch also controls which ports are associated with which VLANs. In Figure 8-1, if the switches are in their default configuration, all ports by default are assigned to VLAN 1, and that means all the devices, including the two users and the router, are all in the same broadcast domain, or VLAN.

As you start adding hundreds of users, you might want to separate groups of users into individual subnets and associated individual VLANs. To do this, you assign the switch ports to the VLAN, and then any device that connects to that specific switch port is a member of that VLAN. Hopefully, all the devices that connect to switch ports that are assigned to a given VLAN also have a common IP network address configured so that they can communicate with other devices in the same VLAN. Often, DHCP is used to assign IP addresses from a common subnet range to the devices in a given VLAN.

If you want to move the two users in Figure 8-1 to a new common VLAN, you create the VLAN on the switches, and then assign the individual access ports that connect the users to the network to that new VLAN, as shown in Example 8-1.

Example 8-1 *Creating a New VLAN and Placing Switch Ports into That VLAN*



```

! Create the new VLAN
SW1(config)# vlan 10

! Assign the port as an access port belonging to VLAN 10
SW1(config-vlan)# interface fa0/1
SW1(config-if)# switchport mode access
SW1(config-if)# switchport access vlan 10

! Verify the VLAN exists, and that Fa0/1 has been assigned to it.
SW1(config-if)# do show vlan brief

VLAN Name                Status            Ports
-----
1    default                active           Fa0/2, Fa0/3, Fa0/4, Fa0/5
                                Fa0/6, Fa0/7, Fa0/8, Fa0/9
                                Fa0/10, Fa0/11, Fa0/12, Fa0/13
                                Fa0/14, Fa0/15, Fa0/16, Fa0/17
                                Fa0/18, Fa0/19, Fa0/20, Fa0/21
                                Fa0/22, Fa0/23, Fa0/24, Gi0/1
                                Gi0/2
10   VLAN0010                active           Fa0/1
<snip>

! Another way to verify the port is assigned the VLAN:
SW1# show vlan id 10
VLAN Name                Status            Ports
-----
10   VLAN0010                active           Fa0/1

! One more way to verify the same thing:
SW1# show interfaces fa0/1 switchport
Name: Fa0/1
Switchport: Enabled
Administrative Mode: static access
Operational Mode: static access
<snip>
Access Mode VLAN: 10 (VLAN0010)
<snip>

```

Note You would perform a similar configuration on SW2 with regard to creating VLAN 10 and assigning port Fa0/2 on SW2 as an access port in that VLAN.

Trunking with 802.1Q

Key Topic

One problem with having two users in the same VLAN but not on the same physical switch is how SW1 tells SW2 that a broadcast or unicast frame is supposed to be for VLAN 10. The answer is simple. For connections between two switches, you configure specific trunk ports instead of configuring access ports. If the two switches are configured as trunks, they include additional information called a *tag* that identifies which VLAN each frame belongs to. 802.1Q is the standard protocol for this tagging. The most critical piece of information (for this discussion) in this tag is the VLAN ID.

Currently, the two users cannot communicate because they are in the same VLAN (VLAN 10) but the interswitch links (between the two switches) are not configured as trunks. To configure both sets of interfaces as trunks, you would specify the trunk method of 802.1Q, and then turn on the feature, as shown in Example 8-2.

Key Topic

Example 8-2 Configure Interfaces as Trunk Ports

```
SW2(config)# interface range fa0/23-24
SW2(config-if-range)# switchport trunk encapsulation dot1q
SW2(config-if-range)# switchport mode trunk
SW2(config-if-range)#

! To verify the trunks:
SW2(config-if-range)# do show interface trunk

Port          Mode          Encapsulation  Status      Native vlan
Fa0/23        on            802.1Q         trunking    1
Fa0/24        on            802.1Q         trunking    1

Port          Vlans allowed on trunk
Fa0/23        1-4094
Fa0/24        1-4094

Port          Vlans allowed and active in management domain
Fa0/23        1,10
Fa0/24        1,10

Port          Vlans in spanning tree forwarding state and not pruned
Fa0/23        1,10
Fa0/24        none
SW2(config-if-range)#
```

```

! Another way to verify the trunk:
SW2# show interface fa0/23 switchport
Name: Fa0/23
Switchport: Enabled
Administrative Mode: trunk
Operational Mode: trunk
Administrative Trunking Encapsulation: dot1q
Operational Trunking Encapsulation: dot1q
Negotiation of Trunking: On
Access Mode VLAN: 1 (default)
Trunking Native Mode VLAN: 1 (default)
Administrative Native VLAN tagging: enabled
<snip>

```

Following the Frame, Step by Step

A broadcast frame sent from PC1 and received by SW1 would forward the frame over the trunk tagged as belonging to VLAN 10 to SW2. SW2 would see the tag, know it was a broadcast associated with VLAN 10, remove the tag, and forward the broadcast to all other interfaces associated with VLAN 10, including the switch port that is connected to PC2. These two core components (access ports being assigned to a single VLAN, and trunk ports that tag the traffic so that a receiving switch knows which VLAN a frame belongs to) are the core building blocks for Layer 2 switching, where a VLAN can extend beyond a single switch.

The Native VLAN on a Trunk

From the output in the earlier example, we verified our trunk interfaces between the two switches. One option shown in the output was a native VLAN. By default, the native VLAN is VLAN 1. So, what does this mean, and why do we care? If a user is connected to an access port that is assigned to VLAN 1 on SW1, and that user sends a broadcast frame, when SW2 forwards that broadcast to SW2, because the frame belongs to the native VLAN (and both switches agree to using the same native VLAN), the 802.1Q tagging is simply left off. This works because when the receiving switch receives a frame on a trunk port, if that frame is missing the 802.1Q tag completely, the receiving switch assumes that the frame belongs to the native VLAN (in this case, VLAN 1).

This is not a huge problem until somebody tries to take advantage of this, as discussed later in this chapter. In the meantime, just know that using a specific VLAN as the native VLAN (different from the default of VLAN 1) and never using that same VLAN for user traffic is a prudent idea.



So, What Do You Want to Be? (Says the Port)

Trunks can be automatically negotiated between two switches, or between a switch and a device that can support trunking. Automatic negotiation to determine whether a port will be an access port or a trunk port is risky because an attacker could potentially negotiate a trunk with a switch; then the attacker could directly access any available VLANs simply by illegally tagging the traffic directly from his PC.

Inter-VLAN Routing

Key
Topic

Our two users (PC1 and PC2) communicate with each other, and they can communicate with other devices in the same VLAN (which is also the same IP subnet), but they cannot communicate with devices outside their local VLAN without the assistance of a default gateway. A router could be implemented with two physical interfaces, one connecting to an access port on the switch that is been assigned to VLAN 10, and another physical interface connected to yet a different access port that is been configured for yet a different VLAN. With two physical interfaces and a different IP address on each, the router could perform routing between the two VLANs.

The Challenge of Using Physical Interfaces Only

So here is the problem: What if you have 50 VLANs? Purchasing 50 physical interfaces for the router would be pricey, let alone the fact that you would also be using 50 physical interfaces on the switch. One solution is to use a technique called *router on a stick*. Consider Figure 8-1. R3 has one physical interface physically connected to the switch. So, from a physical topology perspective, it looks like the router is a lollipop, and that is where it gets its name. There is a video example about router on a stick and inter-VLAN routing available on my YouTube channel at www.YouTube.com/Keith6783.

Using Virtual “Sub” Interfaces

To use one physical interface, we have to play a little game, where we tell the switch that we are going to do trunking out to the router, which from the switch perspective looks exactly like trunking to another switch. And on the router, we tell the router to pay attention to the 802.1Q tags, and assign frames from specific VLANs, based on the tags, to logical subinterfaces. Each subinterface has an IP address from different subnets, as shown in Example 8-3.

Key
Topic

Example 8-3 *Configuring Router on a Stick and Switch Support for the Router*

```
! Enable trunking on the switchport connected to the router
SW1(config)# int fa 0/3
SW1(config-if)# switchport trunk encapsulation dot1q
SW1(config-if)# switchport mode trunk

! Move to R3:
! Make sure the physical interface isn't shutdown
```

```

R3(config)# int fa 0/0
R3(config-if)# no shutdown

! Create a logical sub interface, using any number following the .
R3(config-if)# int fa 0/0.1

! Tell the router to process any dot1q frames tagged with VLAN ID 10 with this
! logical interface
R3(config-subif)# encapsulation dot1q 10

! Provide an IP address that is in the same subnet as other devices in VLAN 10
R3(config-subif)# ip address 10.0.0.1 255.255.255.0

! Verify that this router can ping devices in VLAN 10, namely PC1 and PC2
R3(config-subif)# do ping 10.0.0.11

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.11, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
R3(config-subif)# do ping 10.0.0.22

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.0.0.22, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
R3(config-subif)#

```

Note The PC1 and PC2 computers need to configure R3's address of 10.0.0.1 as their default gateway if those computers want to use R3 to reach nonlocal networks.

You can repeat the process of creating additional subinterfaces on the router to support more VLANs until the router has a subinterface in every VLAN you want.

Spanning-Tree Fundamentals

This section discusses the basics of how the *Spanning Tree Protocol (STP)* can avoid loops at Layer 2 of the OSI model. It is important to understand how it works so that you can fully understand correct mitigation techniques.

This discussion references Figure 8-1 again.

Loops in Networks Are Usually Bad

Without STP, whenever we have parallel connections between Layer 2 devices, such as the connections between SW1 and SW2, we would have Layer 2 loops. Let's take a look at this using the network configured in the previous section. STP is on by default on most Cisco switches, but for the purposes of this discussion, assume that STP is not running, at least for now.

The Life of a Loop

If PC1 sends an ARP request into the network, SW1 receives it and knows that this frame belongs to VLAN 10 because of the access port it came in on, and forwards it out all other ports that are also assigned to VLAN 10, in addition to any trunk ports that are allowing VLAN 10. By default, trunk ports allow all VLAN traffic. So, this broadcast is tagged as belonging to VLAN 10, and is sent down ports 23 and 24.

Just for a moment, let's follow just one of those ports. So, the traffic is being sent down port 23, and SW2 sees it and decides it needs to forward it out all other ports that are assigned to VLAN 10, which includes port number 2, which is an access port assigned to VLAN 10, and also the trunk port 24. So, now SW2 sends the same broadcast to SW1 on port 24. SW1 repeats the process and sends it out port 23, and there would be a loop. The loop happens in the other direction, as well. Besides having a loop, both switches become confused about which port is associated with the source MAC address of PC1. Because a looping frame is seen inbound on both ports 23 and 24, because of the loop going both directions, MAC address flapping occurs in the dynamically learned MAC address table of the switch.

The Solution to the Layer 2 Loop

STP, or 802.1D, was developed to identify parallel Layer 2 paths and block on one of the redundant paths so that a Layer 2 loop would not occur. A single switch with the lowest bridge ID becomes the *root bridge*, and then all the other nonroot switches determine whether they have parallel paths to the root and block on all but one of those paths. STP communicates using *bridge protocol data units (BPDU)*, and that is how negotiation and loop detection is accomplished.

Example 8-4, which contains annotations, allows you to both review how STP operates and see the commands to verify it at the same time; it uses the topology from the beginning of this chapter.

Example 8-4 STP Verification and Annotations

```

SW1# show spanning-tree vlan 10

VLAN0010
! This top part indicates the Bridge ID of the root bridge, which is made
! up of the
! bridge priority, then the base mac address for the switch, lower is better
! This switch is claiming victory over the other switch (SW2)
! This is due to this switch having a lower bridge ID

Spanning tree enabled protocol ieee
Root ID    Priority    32778
           Address    0019.060c.9080
           This bridge is the root
           Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec

! This is the output about the local switch.  Because this is the root
! switch,
! this information will be identical to the information above regarding the
! bridge ID, which is a combination of the Priority and Base MAC address
  Bridge ID Priority    32778 (priority 32768 sys-id-ext 10)
  Address    0019.060c.9080
  Hello Time  2 sec  Max Age 20 sec  Forward Delay 15 sec
  Aging Time  300 sec

! This specifies the state of each interface, and the default costs associated
! with each interface if trying to reach the root switch.  Because this
! switch
! is the root bridge/switch, the local costs are not relevant.
! This also shows the forwarding or blocking state.  All ports on the root
! switch
! will be forwarding, every time, for the VLAN that it is the root bridge for.
Interface          Role Sts Cost          Prio.Nbr Type
-----
Fa0/1              Desg FWD 19           128.3   P2p
Fa0/3              Desg FWD 19           128.5   P2p
Fa0/23             Desg FWD 19           128.25  P2p
Fa0/24             Desg FWD 19           128.26  P2p

! Road trip over to SW2, who didn't win the STP election
SW2# show spanning-tree vlan 10

! This first part identifies who the root is, and the cost for this switch to get
! to the root switch.  SW1 advertised BPDUs that said the cost to reach me is 0,
! and then this switch SW2, added that advertised cost to it's only local
! interface cost to get to 19 as the cost for this switch to reach the root
! bridge.

```

```

VLAN0010
  Spanning tree enabled protocol ieee
  Root ID    Priority    32778
            Address    0019.060c.9080
            Cost      19
            Port      25 (FastEthernet0/23)
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec

! This part identifies the local switch STP information. If you compare the
! bridge ID of this switch, to the bridge ID of SW1 (the root switch), you
! will notice that the priority values are the same, but SW1's MAC address
! is slightly lower, and as a result has a lower Bridge ID, which caused
! SW1 to win the election for root bridge of the spanning tree for VLAN 10
  Bridge ID Priority    32778 (priority 32768 sys-id-ext 10)
            Address    0019.0617.6600
            Hello Time 2 sec  Max Age 20 sec  Forward Delay 15 sec
            Aging Time 300 sec

! This is the port forwarding/blocking information for SW2. SW2 received
! BPDUs from root bridge on both 23 and 24, and so it knew there was a
! loop. It decided to block on port 24. The cost was the same on both
! ports, and STP used the advertised port priority of the sending switch,
! and chose the lower value. In STP lower is always preferred. By
! default, lower numbered physical ports, have lower numbered port
! priorities.
Interface          Role Sts Cost      Prio.Nbr Type
-----
Fa0/2              Desg FWD 19        128.4   P2p
Fa0/23             Root FWD 19        128.25  P2p
Fa0/24             Altn BLK 19        128.26  P2p

! The blocking on port 24 is also reflected in the output of the show
! commands for trunking. Notice that port 23 is forwarding for both
! VLAN 1 and 10, while port 24 is not forwarding for either VLAN.
SW2# show interfaces trunk

Port      Mode          Encapsulation  Status      Native vlan
Fa0/23    on            802.1Q         trunking    1
Fa0/24    on            802.1Q         trunking    1

Port      Vlans allowed on trunk
Fa0/23    1-4094
Fa0/24    1-4094

```

Port	Vlans allowed and active in management domain
Fa0/23	1, 10
Fa0/24	1, 10
Port	Vlans in spanning tree forwarding state and not pruned
Fa0/23	1, 10
Fa0/24	none

STP is on by default, and will have a separate instance for each VLAN. So, if you have five VLANs, you have five instances of STP. Cisco calls this default implementation *Per-VLAN Spanning Tree Plus (PVST+)*.

STP Is Wary of New Ports

When an interface is first brought up and receives a link signal from a connected device, such as a PC or router that is connected, STP is cautious before allowing frames in on the interface. If another switch is attached, there is a possible loop. STP cautiously waits for 30 seconds on a recently brought up port before letting frames go through that interface; 15 seconds of that is the listening state, where STP is seeing whether any BPDUs are coming in. During this time, it does not record source MAC addresses in its dynamic table. The second half of the 30 seconds (15 more) is then still looking for BPDUs, but STP also begins to populate the MAC address table with the source MAC addresses it sees in frames. This is called the *learning state*. After listening and learning have completed (full 30 seconds), the switch can begin forwarding frames. If a port is in blocking state at first, an additional 20-second delay might occur as the port determines that the parallel path is gone, before moving to listening and learning.

For most administrators and users, this delay is too long. When configured, enhancements to STP, including the PortFast feature, can tell the switch to bypass the listening and learning stage and go right to forwarding. This leaves a small window for a loop if a parallel path is injected in the network.

Improving the Time Until Forwarding

Cisco had some proprietary improvements to the 802.1D (traditional STP) that allowed faster convergence in the event of a topology change and included many features such as the PortFast, UplinkFast, and BackboneFast. Many of these features were used in a newer version of STP called *Rapid Spanning Tree* (also known as 802.1w). Enabling PortFast for traditional STP and configuring Rapid Spanning Tree globally are shown in Example 8-5.



Example 8-5 *Configuring PortFast, Then Rapid Spanning Tree*

```

SW2(config)# interface fa0/2
SW2(config-if)# spanning-tree portfast
%Warning: portfast should only be enabled on ports connected to a single
  host. Connecting hubs, concentrators, switches, bridges, etc... to this
  interface when portfast is enabled, can cause temporary bridging loops.
  Use with CAUTION

%Portfast has been configured on FastEthernet0/2 but will only
  have effect when the interface is in a non-trunking mode.
SW2(config-if)#

! or we could do it globally
SW2(config)# spanning-tree portfast default
%Warning: this command enables portfast by default on all interfaces. You
  should now disable portfast explicitly on switched ports leading to hubs,
  switches and bridges as they may create temporary bridging loops.

! To change the STP from 802.1D to 802.1w, it is this one command
SW2(config)# spanning-tree mode rapid-pvst

! The show command will display rstp, instead of the original ieee for the
! mode
SW2# show spanning-tree vlan 10

VLAN0010
  Spanning tree enabled protocol rstp
<snip>

```

Common Layer 2 Threats and How to Mitigate Them

This section discusses many security threats that focus on Layer 2 technologies, and addresses how to implement countermeasures against those risks. This is relevant to the “security” portion of the CCNA Security certification.

Disrupt the Bottom of the Wall, and the Top Is Disrupted, Too

Everything at Layer 3 and higher is encapsulated into some type of Layer 2 frame. If the attacker can interrupt, copy, redirect, or confuse the Layer 2 forwarding of data, that same attacker can also disrupt any type of upper-layer protocols that are being used.

Layer 2 Best Practices

Let's begin with best practices for securing your switches, and then discuss in more detail which best practice mitigates which type of attack.

Best practices for securing your infrastructure, including Layer 2, include the following:

- Select an unused VLAN (other than VLAN 1) and use that for the native VLAN for all your trunks.
- Avoid using VLAN 1 anywhere, because it is a default.
- Administratively configure access ports as access ports so that users cannot negotiate a trunk and disable the negotiation of trunking (no *Dynamic Trunking Protocol [DTP]*).
- Limit the number of MAC addresses learned on a given port with the *port security* feature.
- Control spanning tree to stop users or unknown devices from manipulating spanning tree. You can do so by using the BPDU guard and root guard features.
- Turn off *Cisco Discovery Protocol (CDP)* on ports facing untrusted or unknown networks that do not require CDP for anything positive. (CDP operates at Layer 2 and may provide attackers information we would rather not disclose.)
- On a new switch, shut down all ports and assign them to a VLAN that is not used for anything else other than a parking lot. Then bring up the ports and assign correct VLANs as the ports are allocated and needed.



To control whether a port is an access port or a trunk port, you can revisit the commands used earlier in this chapter, including the ones shown in Example 8-6.

Example 8-6 Administratively Locking Down Switch Ports

```
SW2(config)# int fa0/2

! Specifies that this is an access port, not a trunk, and specifies VLAN
! association
SW2(config-if)# switchport mode access
SW2(config-if)# switchport access VLAN 10

! Disables the ability to negotiate, even though we hard coded the port as
! an access port This command disables DTP, which otherwise is still on
! by default, even for an interface configured as an access port.
SW2(config-if)# switchport nonegotiate

SW2(config-if)# int fa 0/23
! Specifies the port as a trunk, using dot1q
```



```

SW2(config-if)# switchport trunk encapsulation dot1q
SW2(config-if)# switchport mode trunk

! Specify a VLAN that exists, but isn't used anywhere else, as the native
! VLAN
SW2(config-if)# switchport trunk native vlan 3

! Disables the ability to negotiate, even though we hard coded the port as
! a trunk
SW2(config-if)# switchport nonegotiate

! Note, negotiation packets are still being sent unless we disable
! negotiation

```

Do Not Allow Negotiations

The preceding example prevents a user from negotiating a trunk with the switch, maliciously, and then having full access to each of the VLANs by using custom software on the computer that can both send and receive dot1q tagged frames. A user with a trunk established could perform “VLAN hopping” to any VLAN he desired by just tagging frames with the VLAN of choice. Other malicious tricks could be done, as well, but forcing the port to an access port with no negotiation removes this risk.

Layer 2 Security Toolkit

Cisco has many tools for protecting Layer 2, including those described in Table 8-2.

Table 8-2 *Tool Kit for L2 Security*



Tool	Description
Port security	Limits the number of MAC addresses to be learned on an access switch port, as covered later in this chapter.
BPDU guard	If BPDUs show up where they should not, the switch protects itself, as covered in this chapter.
Root guard	Control which ports are not allowed to become root ports to remote root switches, as covered in this chapter.
Dynamic ARP inspection	Prevents spoofing of Layer 2 information by hosts.
IP source guard	Prevents spoofing of Layer 3 information by hosts.
802.1x	Authenticates users before allowing their data frames into the network.
DHCP snooping	Prevents rogue DHCP servers from impacting the network.

Tool	Description
Storm control	Limits the amount of broadcast or multicast traffic flowing through the switch.
Access control lists	Traffic control to enforce policy. Access control is covered in another chapter.

The key Layer 2 security technologies we focus on in CCNA Security include port security, BPDU guard, root guard, DHCP snooping, and access lists. The other topics from the table you can save for your *CCNP* Security studies.

Specific Layer 2 Mitigation for CCNA Security

With a review of the switching technologies and how they operate now in mind, let's take a specific look at implementing security features on our switches.

BPDU Guard

When you enable BPDU guard, a switch port that was forwarding stops and disables the port if a BPDU is seen inbound on the port. A user should never be generating legitimate BPDUs. This configuration, applied to ports that should only be access ports to end stations, helps to prevent another switch (that is sending BPDUs) from being connected to the network. This could prevent manipulation of your current STP topology. Example 8-7 shows the implementation of BPDU guard.



Example 8-7 Implementing BPDU Guard on a Switch Port

```
SW2(config-if)# int fa 0/2
SW2(config-if)# spanning-tree bpduguard enable

! Verify the status of the switchport
SW2# show int fa0/2 status
```

Port	Name	Status	Vlan	Duplex	Speed	Type
Fa0/2		connected	10	a-full	a-100	10/100BaseTX

```
SW2#
```

A port that has been disabled because of a violation shows a status of **err-disabled**. To bring the interface back up, issue a **shutdown** and then a **no shutdown** in interface configuration mode.

You can also configure the switch to automatically bring an interface out of **err-disable**, based on the reason it was placed there and how much time has passed before bringing the interface back up. To enable this for a specific feature, follow Example 8-8.

Example 8-8 *Configuring the Switch to Automatically Restore Err-Disabled Ports*

```

SW2(config)# errdisable recovery cause bpduguard

! err-disabled ports will be brought back up after 30 seconds of no bpdu
! violations
SW2(config)# errdisable recovery interval 30

! You can also see the timeouts for the recovery

SW2# show errdisable recovery
ErrDisable Reason          Timer Status
-----
arp-inspection             Disabled
bpduguard                  Enabled
<snip>

Timer interval: 30 seconds
Interfaces that will be enabled at the next timeout:

SW2#

```

Root Guard

Your switch might be connected to other switches that you do not manage. If you want to prevent your local switch from learning about a new root switch through one of its local ports, you can configure root guard on that port, as shown in Example 8-9. This will also help in preventing tampering of your existing STP topology.

Example 8-9 *Controlling Which Ports Face the Root of the Spanning Tree*

```

SW1(config)# int fa 0/24
SW1(config-if)# spanning-tree guard root
%SPANTREE-2-ROOTGUARD_CONFIG_CHANGE: Root guard enabled on port
FastEthernet0/24.

```

Port Security

How many MAC addresses should legitimately show up inbound on an access port?

Port security controls how many MAC addresses can be learned on a single switch port. This feature is implemented on a port-by-port basis. A typical user uses just a single MAC address. Exceptions to this may be a virtual machine or two that might use different MAC addresses than their host, or if there is an IP phone with a built-in switch, which may also account for additional MAC addresses. In any case, to avoid a user

connecting dozens of devices to a switch, that is then connected to their access port, you can use port security to limit the number.

This also protects against malicious applications that may be sending thousands of frames into the network, with a different bogus MAC address for each frame, as the user tries to exhaust the limits of the dynamic MAC address table on the switch, which might cause the switch to forward all frames to all ports within a VLAN so that the attacker can begin to sniff all packets. This is referred to as a *CAM table overflow attack*. *Content-addressable memory (CAM)* is a fancy way to refer to the MAC address table on the switch.

Port security also prevents the client from depleting DHCP server resources, which could have been done by sending thousands of DHCP requests, each using a different source MAC address.

With the port security feature, the default violation action is to shut down the port. Alternatively, we can configure the violation response to be to “protect,” which will not shut down the port but will deny any frames from new MAC addresses over the set limit. The “restrict” action does the same as protect but generates a syslog message, as well.

To implement port security, follow Example 8-10.

Example 8-10 Implementing Port Security

```
SW2(config-if)# int fa 0/2

! Enable the feature
SW2(config-if)# switchport port-security

! Set the maximum to desired number. Default is 1. If we administra
! tively
! set the maximum to 1, the command won't show in the running configuration
! because the configuration matches the default value. It is handy to know
! this behavior, so you won't be surprised by what may seem to be a missing
! part of your configuration.
SW2(config-if)# switchport port-security maximum 5

! Set the violation action. Default is err-disable. Protect will simply
! not allow
! frames from MAC addresses above the maximum.
SW2(config-if)# switchport port-security violation protect

! This will cause the dynamic mac addresses to be placed into running
! config to save them to startup config, use copy run start
SW2(config-if)# switchport port-security mac-address sticky
```



```

! To verify settings, use this command
SW2# show port-security
Secure Port  MaxSecureAddr  CurrentAddr  SecurityViolation  Security Action
              (Count)          (Count)          (Count)
-----
      Fa0/2             5             1             0             Protect
-----
Total Addresses in System (excluding one mac per port)      : 0
Max Addresses limit in System (excluding one mac per port) : 6144

! This can also provide additional information about port security:

SW2# show port-security interface fa0/2
Port Security          : Enabled
Port Status            : Secure-up
Violation Mode         : Protect
Aging Time             : 0 mins
Aging Type             : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses  : 5
Total MAC Addresses    : 1
Configured MAC Addresses : 0
Sticky MAC Addresses   : 1
Last Source Address:Vlan : 0000.2222.2222:10
Security Violation Count : 0

```

For a video demonstration of port security, see the video on that topic that accompanies this book.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 8-3 lists these key topics.

Table 8-3 *Key Topics*

Key Topic Element	Description	Page Number
Text	What is a VLAN?	178
Example 8-1	Creating a new VLAN and placing switch ports into that VLAN	179
Text	Trunking with 802.1Q	180
Example 8-2	Configure Interfaces to be trunk ports	180
Text	The native VLAN on a trunk	181
Text	Inter VLAN routing	182
Example 8-3	Configuring router on a stick and switch support for the router	182
Example 8-5	Configuring PortFast, then Rapid Spanning Tree	188
List	Layer 2 best practices	189
Example 8-6	Administratively locking down switch ports	189
Table 8-2	Layer 2 security toolkit	190
Text	BPDU guard	191
Text	Root guard	192
Text	Port security	192
Example 8-10	Implementing port security	193



Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Review the Port Security Video Included with This Book

Watch the video, and if possible practice the port security commands, including the commands to verify port security. If you do not have access to simulated or practice gear, verify that you can write out the commands, on paper, without assistance.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

access port, trunk port, inter-VLAN routing, router on a stick, STP, root guard, port security, BPDU guard

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 8-4 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 8-4 *Command Reference*

Command	Description
<code>switchport mode access</code>	Assign a switch port as an access port
<code>switchport access vlan 10</code>	Control the VLAN assignment for the device connecting to this port, and associate that device with a single specific VLAN of 10
<code>show interfaces fa0/1 switchport</code>	Verify the current configuration and operating status of a switch port
<code>switchport trunk encapsulation dot1q</code>	Specify the trunking encapsulation to be used, if doing trunking
<code>switchport mode trunk</code>	Specify that this port should be a trunk
<code>switchport trunk native vlan 3</code>	Specify the native VLAN should be 3, if the port is acting as a trunk port
<code>switchport nonegotiate</code>	Disable negotiation between the switch and the device connected to the device related to trunking
<code>spanning-tree bpduguard enable</code>	Protect the switch port against being connected on this port to another device that is generating any type of BPDUs
<code>spanning-tree guard root</code>	Protect this switch port against believing the root bridge is reachable via this port
<code>switchport port-security</code>	Protect the switch (on this port at least) against a MAC address table flooding attack (CAM table overflow) and prevent a DHCP starvation attack from being launched from the device connected to this port

This page intentionally left blank



This chapter covers the following subjects:

- Understanding and configuring IPv6
- Configuring IPv6 routing
- Developing a security plan for IPv6

Securing the Data Plane in IPv6

IPv6 is definitely in the future for those who will be working with IP. The prerequisite for CCNA Security is the CCNA in Route/Switch. As a result, it is assumed that Security candidates are familiar with IPv6. To make certain that you are, this chapter explains how IPv6 works and how to configure it, and then you learn how to develop a security plan for IPv6. If you already feel comfortable with IPv6, you can go directly to the “Developing a Security Plan for IPv6” section, later in this chapter.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 9-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 9-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Understanding and Configuring IPv6	1
Understanding and Configuring IPv6 Routing	2
Developing a Security Plan for IPv6	3–10

1. Which of the following are the valid first four characters of a globally routable IPv6 address? (Choose all that apply.)
 - a. 1234
 - b. 2345
 - c. 3456
 - d. 4567

- 2.** Which of the following are the valid first four characters of a link local address?
 - a.** FE80
 - b.** FF02
 - c.** 2000
 - d.** 3000

- 3.** What is the default method for determining the interface ID for a link local address on Ethernet?
 - a.** EUI-64
 - b.** MAC address with FFFE at the end
 - c.** MAC address with FFFE at the beginning
 - d.** Depends on the network address being connected to

- 4.** How many groups of four hexadecimal characters does an IPv6 address contain?
 - a.** 4
 - b.** 8
 - c.** 16
 - d.** 32

- 5.** Which of the following routing protocols have both an IPv4 and IPv6 version? (Choose all that apply.)
 - a.** Routing Information Protocol
 - b.** Enhanced Interior Gateway Routing Protocol
 - c.** Open Shortest Path First
 - d.** Interior Gateway Routing Protocol

- 6.** Which best practices apply to networks that run both IPv4 and IPv6? (Choose all that apply.)
 - a.** Physical security
 - b.** Routing protocol authentication
 - c.** Authorization of administrators
 - d.** Written security policy

7. Which of protocols, if abused, could impair an IPv6 network, but not IPv4?
(Choose all that apply.)
 - a. ARP
 - b. NDP
 - c. Broadcast addresses
 - d. Solicited node multicast addresses
8. If a rogue IPv6 router is allowed on the network, which information could be incorrectly delivered to the clients on that network? (Choose all that apply.)
 - a. IPv6 default gateway
 - b. IPv6 DNS server
 - c. IPv6 network address
 - d. IPv6 ARP mappings
9. Why is tunneling any protocol (including IPv6) through another protocol a security risk?
 - a. The innermost contents of the original packets may be hidden from normal security filters.
 - b. The tunnels, if they extend beyond the network perimeter, may allow undesired traffic through the tunnel.
 - c. Functionality might need to be sacrificed when going through a tunnel.
 - d. Quality of service, for the underlying protocol, might be compromised.
10. What is one method to protect against a rogue IPv6 router?
 - a. Port security
 - b. Static ARP entries
 - c. DHCPv6
 - d. RA guard

Foundation Topics

Understanding and Configuring IPv6

When compared to IPv4, both similarities and differences exist as to how IPv6 operates. Certification requires that you to know the fundamentals of IPv6, and that is the focus in this section, which first reviews IPv6 basics and then shows you how to configure it.

Why IPv6?

Two good reasons to move to IPv6 are as follows:

- IPv6 has more address space available.
- We are running out of public IPv4 addresses.

For more than a decade, the requirement to implement IPv6 has been threatened as imminent. The lifetime of its predecessor (IPv4) was extended more than a decade because of features such as *Network Address Translation (NAT)*, which enables you to hide thousands of users with private IP addresses behind a single public IP address.

With IPv6, upper-layer applications still work like you are used to with IPv4. The biggest change is that we are doing a forklift upgrade to Layer 3 of the OSI model. Along with this change, there are some modifications as to how IPv6 interacts with the rest of the protocol stack and some modifications to its procedures for participating on the network.

Table 9-2 describes a few of the notable differences and some similarities between IPv4 and IPv6.

Table 9-2 *IPv4 Versus IPv6*



IPv4	IPv6
32-bit (4-byte) address supports 232 4,294,967,296 addresses.	128-bit (16-byte) address supports 2128 (about 3.4 × 1038) addresses (340 undecillion addresses altogether, or roughly 42 octillion addresses per person on the planet! Or 438 quintillion addresses per square inch of land on Earth. (Or <i>lots!</i>)
You can use NAT to extend address space limitations.	Does not support NAT by design (and has plenty of addresses for everyone).
Administrators must use <i>Dynamic Host Configuration Protocol (DHCP)</i> or static configuration to assign IP addresses to hosts.	Hosts can use stateless address autoconfiguration to assign an IP address to themselves, but can also use DHCP features to learn more information, such as about <i>Domain Name System (DNS)</i> servers.

IPv4	IPv6
IPsec support is an optional add-on concept to protect IP packets through encryption, validating a peer, data integrity, and antireplay support.	IPsec support is supposed to be “required.” This really means that it is supported for IPv6 from the beginning, but IPv6 does not require it to be configured for IPv6 to work.
Multiple pieces in an IPv4 header.	Simplified (but larger) IPv6 header, with options for header extensions as needed.
Uses broadcasts for several functions, including <i>Address Resolution Protocol (ARP)</i> .	Does not use any broadcasts and does not use ARP. Instead, it uses multicast addresses and <i>Neighbor Discovery Protocol (NDP)</i> also called <i>ND</i> . ND replaces ARP. Devices can automatically discover the IPv6 network address and many other housekeeping features such as discovering any routers on the network. ND uses IPv6’s version of <i>Internet Control Message Protocol (ICMP)</i> as the workhorse behind most of its functions.
Both support common Layer 4 protocols such as TCP, UDP.	Both support common Layer 4 protocols such as TCP, UDP.
Both support common application layer protocols, such as HTTP, FTP, and so on, that are encapsulated in their respective Layer 4 protocols.	Both support common application layer protocols, such as HTTP, FTP, and so on, that are encapsulated in their respective Layer 4 protocols.
Both support common Layer 2 technologies such as Ethernet standards and WAN standards.	Both support common Layer 2 technologies, such as Ethernet standards and WAN standards.
Both contain two parts in the IP address: the network on the left side of the address, and the host part on the right side of the address.	Both contain two parts in the IP address: the Network on the left side of the address, and the host part on the right side of the address. In IPv6, the host part is also called the <i>host ID</i> .
Both use a network mask to identify which part of the address is the network (on the left) indicated by the mask bits that are on” (or 1), and the rest of the bits to the right represent the host part of the IPv4 address.	Both use a network mask to identify which part of the address is the network (on the left) indicated by the mask bits that are on” (or 1), and the rest of the bits to the right represent the host part or host ID or interface ID of the IPv6 address.

The Format of an IPv6 Address

Understanding the basic format of an IPv6 address is important for certification and for the actual implementation of IPv6. A few key details about IPv6 and its format are as follows:

- **Length:** IPv6 addresses are 128 bits long.
- **Groupings:** IPv6 addresses are segmented into eight groups of four hex characters.

- **Separation of groups:** Each group is separated by a colon (:).
- **Length of mask:** Usually 50 percent (64 bits long) for network ID, which leaves 50 percent (also 64 bits) for interface ID (using a 64-bit mask).
- **Number of networks:** The network part is allocated by Internet registries 2^{64} (1.8×10^{19})

This allows room for billions of networks.

Hexadecimal only takes one character to represent 4 bits and is used to represent IPv6 addresses (4 bits at a time). Table 9-3 shows the conversion between decimal, binary, and hexadecimal.

Table 9-3 *Conversion Charts Between Decimal, Binary and Hexadecimal*

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

We can represent an IPv6 the hard way or the easier way. The hard way is to type in every hexadecimal character for the IP address. In Example 9-1, we put in a 128-bit IPv6 address, typed in as 32 hexadecimal characters, and a /64-bit mask.

Example 9-1 *An IPv6 Address Configured the Hard Way*

```

R1(config-if)# ipv6 address 2001:0db8:0000:0000:1234:0000:0052:0001/64

! The output reflects how we could have used some shortcuts in representing
! the groups of zeros.
R1(config-if)# do show ipv6 interface brief
FastEthernet0/1          [up/up]
    FE80::C800:41FF:FE32:6
    2001:DB8::1234:0:52:1
R1(config-if)#

```

Understanding the Shortcuts

Example 9-1 shows the address being configured and the abbreviated address from the output of the **show** command. When inserting a group of four hexadecimal numbers, you can limit your typing a bit. For example, if any leading characters in the group are 0, you can omit them (just as the 0 in front of DB8 is in the second group from the left). In addition, if there are one or more consecutive groups of all 0s, you can input them as a double colon (::). The system knows that there should be eight groups separated by seven colons, and when it sees ::, it just looks at how many other groups are configured and assumes that the number of missing groups plus the existing groups that are configured totals eight. In the example, the first two groups of consecutive 0s are shortened in the final output. This shortcut may be done only once for any given IPv6 address. This example contains three groupings of 0s, and if you use the shortcut twice, the system will not know whether there should be four 0s after the DB8: and eight 0s (or two groups) after the 1234: or vice versa.

Did We Get an Extra Address?

Besides the IPv6 global address configured in Example 9-1, the system automatically configured for itself a second IPv6 address known as a *link local* address that begins with FE80. A link local address is an IPv6 address that you can use to communicate with other IPv6 devices on the same local network (local broadcast domain). If an IPv6 device wants to communicate with a device that is remote, it needs to use its global and routable IPv6 address for that (not the link local one). To reach remote devices, you also need to have a route to that remote network or a default gateway to use to reach the remote network.

The following section covers the other types of addresses that you will work with in IPv6 networks.

IPv6 Address Types

In IPv6, you must be familiar with several types of addresses. Some are created and used automatically; others you must configure manually. These address types include the following:



- **Link local address:** Link local addresses may be manually configured, but if they are not, they are dynamically configured by the local host or router itself. These always begin with the characters FE80. The last 64 bits are the host ID (also referred to as the *interface ID*), and the device uses the modified EUI-64 format (by default) to create that. The modified EUI-64 uses the MAC address (if on Ethernet; and if not on Ethernet, it borrows the MAC address of another interface), and realizes it is only 48 bits. To get to 64 bits for the host ID, it inserts four hexadecimal characters of FFFE, (which is the 16 more bits we need) and injects those into the middle of the existing MAC address to use as the 64-bit host ID. It also looks at the seventh bit from the left (of the original MAC address) and inverts it. If it is a 0 in the MAC address, it is a 1 in the host ID and vice versa. To see an example of this, look back at Example 9-1, at the output of the **show** command there, focusing on the address that begins with FE80.
- **Loopback address:** In IPv4, this was the 127 range of IP addresses. In IPv6, the address is ::1 (which is 127 0s followed by a 1).
- **All-nodes multicast address:** In IPv6, multicasts begin with FFxx: (where the x = some other hex number). 02 happens to designate a multicast address that is link local in scope. There are other preset scopes, but you do not have to worry about them here. The IPv6 multicast group that all IPv6 devices join is FF02::1. If any device needs to send a packet/frame to all other local IPv6 devices, it can send the packet to the multicast address of FF02::1, which translates to a specific multicast Layer 2 address, and then all the devices that receive those frames continue to de-encapsulate those frames. If a device receives a frame, and the receiving device determines that the Layer 2 destination in the frame is not destined for itself and not destined for any multicast groups that the local device has joined, it discards the frame, instead of continuing to de-encapsulate it to find out what is inside (in the upper layers).
- **All-routers multicast address:** In addition to the group address of FF02::1 that is joined by all devices configured for IPv6, routers that have had routing enabled for IPv6 also join the group FF02::2. By doing so, any client looking for a router can send a request to this group address and get a response if there is a router on the local network. You might have noticed a pattern here: FF02 is just like 224.0.0.x in IPv4 multicast. 224.0.0.1 = all devices. 224.0.0.2 = all routers.
- **Unicast and anycast addresses (configured automatically or manually):** A global IPv6 address, unlike a link local address, is routable and can be reached through one or more routers that are running IP routing and that have a correct routing table. Global IPv6 unicast addresses have the first four characters in the range of 2000 to 3FFF, and may be manually configured, automatically discovered by issuing a router solicitation request to a local router, or be learned via IPv6

Dynamic Host Configuration Protocol (DHCP). An anycast address can be a route or an IP address that appears more than one time in a network, and then it is up to the network to decide the best way to reach that IP. Usually, two DNS servers, if they both use the same anycast address, are functional to the users, so that regardless of which DNS server that packets are forwarded to, the client gets the DNS response it needs.

- **Solicited-node multicast address for each of its unicast and anycast addresses:** When a device has global and link local addresses, it joins a multicast group of FF02::1:FFxx:xxxx. The x characters represent the last 24 bits of the host ID being used for the addresses. If a device needs to learn the Layer 2 address of a peer on the same network, it can send out a neighbor solicitation (request) to the multicast group that the device that has that address should have joined. This is the way IPv6 avoids using broadcasts.
- **Multicast addresses of all other groups to which the host belongs:** If a router has enabled IPv6 routing, it joins the FF02::2 group (all routers), as mentioned earlier. If a router is running RIPng (the IPv6 flavor), it joins the multicast group for RIPng, which is FF02::9, so that it will process updates sent to that group from other RIP routers. Notice again some similarities. RIPv2 in IPv4 uses 224.0.0.9 as the multicast address!

Example 9-2 shows the output for a router that has been enabled for IPv6 routing, RIPng, and has an IPv6 global address.

Example 9-2 IPv6 Interface Information

```
! MAC address, for reference, that is currently used on the Fa0/1
! interface
R1# show interfaces fa0/1 | include bia
    Hardware is i82543 , address is ca00.4132.0006 (bia ca00.4132.0006)

R1# show ipv6 interface fa0/1
FastEthernet0/1 is up, line protocol is up

! Link local address, beginning with FE80::
! and using modified EUI-64 for the host ID
! Notice that CA from the MAC address is C8 in the host ID
! due to inverting the 7th bit for the modified EUI-64 formatting
IPv6 is enabled, link-local address is FE80::C800:41FF:FE32:6

! Global addresses have the first group range of 2000-3fff
Global unicast address(es):
    2001:DB8::1234:0:52:1, subnet is 2001:DB8::/64

! Multicast begins with FFxx:
Joined group address(es):
```



```
! Because we are enabled for IPv6 on this interface
  FF02::1

! Because we are enabled for IPv6 routing
  FF02::2

! Because we are enabled for RIPng
  FF02::9

! Because our link local address ends in 32:0006
! This is a solicited node multicast group
  FF02::1:FF32:6

! Because our global address ends in 52:0001
! This is a solicited node multicast group
  FF02::1:FF52:1

<snip>
```

Configuring IPv6 Routing

To support multiple IPv6 networks and allow devices to communicate between those networks, you need to tell the routers how to reach remote IPv6 networks. You can do so through static routes, IPv6 routing protocols, and default routes. For the router to route any customer's IPv6 traffic, you need to enable unicast routing for IPv6 from global configuration mode. If you fail to do this, the router can send and receive its own packets, but it will not forward packets for others, even if it has the routes in its IPv6 routing table.

IPv6 can use the new and improved flavors of these dynamic routing protocols with their versions that support IPv6:

- RIP, called *RIP next generation (RIPng)*
- OSPFv3
- EIGRP for IPv6

One difference with the interior gateway routing protocols for IPv6 is that none of them support **network** statements. To include interfaces of the routing process, you use **interface** commands. For EIGRP, you also need to issue the **no shutdown** command in EIGRP router configuration mode. Example 9-3 shows the enabling of unicast routing and the configuring of IPv6 routing protocols.

Example 9-3 *Enabling IPv6 Routing and Routing Protocols*

```

! Enables IPv6 routing of other devices packets
R1(config)# ipv6 unicast-routing

! Enabling all 3 IGPs on interface Fa0/1
! Note: that in a production network, we would only need 1 routing protocol
! on a given interface. If we did have multiple identical learned routes
! the Administrative Distance (same as in IPv4) would determine which
! routing protocols would be the "best" and be placed in the routing table.
R1(config)# int fa 0/1

! Enabling RIPng on the interface
! Simply create a new "name" for the process. I called this one "MYRIP"
! Use the same name on all the interfaces on the local router where you
! want RIPng to be used on that same router.
R1(config-if)# ipv6 rip MYRIP enable

! Enabling OSPFv3 on the interface
! Syntax is the keywords ipv6 ospf, followed by the process ID, then the
! area information
R1(config-if)# ipv6 ospf 1 area 0

! Enabling IPv6 EIGRP on the interface
R1(config-if)# ipv6 eigrp 1
R1(config-if)# exit

! Bringing the EIGRP routing process out of its default shutdown state
! This is not needed for RIPng or OSPFv3
R1(config)# ipv6 router eigrp 1
R1(config-rtr)# no shutdown

! Verify which routing protocols are running
R1# show ipv6 protocol
IPv6 Routing Protocol is "connected"
IPv6 Routing Protocol is "rip MYRIP"
  Interfaces:
    FastEthernet0/1
  Redistribution:
    None
IPv6 Routing Protocol is "ospf 1"
  Interfaces (Area 0):
    FastEthernet0/1
  Redistribution:
    None

```

```

IPv6 Routing Protocol is "eigrp 1"
  EIGRP metric weight K1=1, K2=0, K3=1, K4=0, K5=0
  EIGRP maximum hopcount 100
  EIGRP maximum metric variance 1
  Interfaces:
    FastEthernet0/1
  Redistribution:
    None
  Maximum path: 16
  Distance: internal 90 external 170

```

The command **show ipv6 route** outputs the IPv6 routes the router knows how to reach, including the ones learned through dynamic routing protocols.

Moving to IPv6

Moving to IPv6 will be more of a transition or migration, than a one-time event. As such, there are mechanisms in place to support co-existence between IPv4 and IPv6 including the ability for a router or network device to run both protocol stacks at the same time, and the ability to perform tunneling. A tunneling example would be when there are two isolated portions of the network that want to run IPv6, and between them, there is a big patch of IPv4 only. Tunneling would take the IPv6 packets, and re-encapsulate them into IPv4 for transport across the IPv4 portion of the network. At the other end of the tunnel, the router would de-encapsulate the IPv6 from the IPv4 shell, and then continue forwarding the IPv6 packet on toward its final destination.

Developing a Security Plan for IPv6

Most security risks associated with the older IPv4 are the same security risks associated with the newer IPv6. Now what does that mean to you? It means that you need to make sure that you have considered and implemented security controls to address both protocol stacks. This section discusses many security threats common to both IPv4 and IPv6 (and some specific to IPv6) and how to address them.

Best Practices Common to Both IPv4 and IPv6



For both protocol stacks, here are some recommended best practices, which is a great place to start your network configuration:

- **Physical security:** Keep the room where the router is housed free (safe) from electrostatic and magnetic interference. It should also be temperature and humidity controlled. There should be controlled and logged access to that physical room. Redundant systems for electricity that feed into the routers are part of this, as well.

- **Device hardening:** Disable services that are not in use and features and interfaces that are not in use. You learned about this concept in an earlier chapter with regard to *Cisco Configuration Professional (CCP)*.
- **Control access between zones:** Enforce a security policy that clearly identifies which packets are allowed between networks (using either simple access list controls or more advanced controls such as stateful inspection that leverages firewall features on a router or a dedicated firewall appliance, all of which are covered extensively in other chapters in this book).
- **Routing protocol security:** Use authentication with routing protocols to help stop rogue devices from abusing the information being used in routing updates by your routers.
- **Authentication, authorization, and accounting (AAA):** Require AAA so that you know exactly who is accessing your systems, when they are accessing your systems, and what they are doing. You learned about AAA in earlier chapters. *Network Time Protocol (NTP)* is a critical part to ensure that time stamps reflect reality. Check log files periodically. All management protocols should be used with cryptographic services. *Secure Shell (SSH)* and *Hypertext Transfer Protocol Secure (HTTPS)* include these features. Place Telnet and HTTP inside of an encrypted *virtual private network (VPN)* tunnel to meet this requirement.
- **Mitigating DoS attacks:** *Denial of service* refers to willful attempts to disrupt legitimate users from getting access to the resources they intend to. Although no complete solution exists, administrators can do specific things to protect the network from a DoS attack and to lessen its effects and prevent a would-be attacker from using a system as a source of an attack directed at other systems. These mitigation techniques include filtering based on bogus source IP addresses trying to come into the networks and vice versa. Unicast reverse path verification is one way to assist with this, as are access lists. Unicast reverse path verification looks at the source IP address as it comes into an interface, and then looks at the routing table. If the source address seen would not be reachable out of the same interface it is coming in on, the packet is considered bad and is dropped. Another technique to mitigate well-known DoS attacks is some type of TCP Intercept, which can mitigate SYN-flood attacks.
- **Have and update a security policy:** A security policy should be referenced and possibly updated whenever major changes occur to the administrative practices, procedures, or staff. If new technologies are implemented, such as a new VPN or a new application that is using unique protocols or different protocols than your current security policy allows, this is another reason to revisit the security policy. Another time a security policy might need to be updated is after a significant attack or compromise to the network has been discovered.

Threats Common to Both IPv4 and IPv6



The following threats and ways to mitigate them apply to both IPv4 and IPv6:

- **Application layer attacks:** An attacker is using a network service in an unexpected or malicious way. To protect against this, you can place filters to allow only the required protocols through the network. This will prevent services that aren't supposed to be available from being accessed through the network. You can also use application inspection, done through the ASA or the IOS zone-based firewall, or IPS functionality to identify and filter protocols that are not being used in their intended way. Being current on system and application patches will also help mitigate an application layer attack.
- **Unauthorized access:** Individuals not authorized for access are gaining access to network resources. To protect against this, use AAA services to challenge the user for credentials, and then authorize that user for only the access they need. This can be done for users forwarding traffic through the network and for administrators who want to connect directly for network management. Accounting records can create a detailed record of the network activity that has taken place.
- **Man-in-the-middle attacks:** Someone or something is between the two devices who believe they are communicating directly with each other. The “man in the middle” may be eavesdropping or actively changing the data that is being sent between the two parties. You can prevent this by implementing Layer 2 Dynamic ARP Inspection (DAI) and STP guards to protect spanning tree. You can implement it at Layer 3 by using routing protocol authentication. Authentication of peers in a VPN is also a method of preventing this type of attack.
- **Sniffing or eavesdropping:** An attacker is listening in on the network traffic of others. This could be done in a switched environment, where the attacker has implemented a CAM table overflow, causing the switch to forward all frames to all other ports in the same VLAN. To protect against this, you can use switch port security on the switches to limit the MAC addresses that could be injected on any single port. In general, if traffic is encrypted as it is transported across the network, either natively or by a VPN, that is a good countermeasure against eavesdropping.
- **Denial-of-service (DoS) attacks:** Making services that should be available unavailable to the users who should normally have the access/service. Performing packet inspection and rate limiting of suspicious traffic, physical security, firewall inspection, and IPS can all be used to help mitigate a DoS attack.
- **Spoofed packets:** Forged addressing or packet content. Filtering traffic that is attempting to enter the network is one of the best first steps to mitigating this type of traffic. Denying inbound traffic that is claiming to originate from inside the network will stop this traffic at the edge of the network. Reverse path checks can also help mitigate this type of traffic.

- **Attacks against routers and other network devices:** Turning off unneeded services and hardening the routers, similar to what the CCP security audit can do, will help the router be less susceptible to attacks against the router itself. Implement the techniques you learned in the NFP chapter to protect the control, management, and data planes.

The Focus on IPv6 Security

With IPv6, you do have a few advantages related to security. If an attacker issues a ping sweep of your network, he will not likely find all the devices via a traditional ping sweep to every possible address, and so reconnaissance will be tougher for the attacker using that method (because there are potentially millions of addresses on each subnet [264 possibilities, or about 18 quintillion]). Be aware, however, that this is a double-edged sword, because each device on an IPv6 network joins the multicast group of FF02::1. So, if the attacker has local access to that network, he could ping that local multicast group and get a response that lets him know about each device on the network. FF02::1 is local in scope, so the attacker cannot use this technique remotely; he would have to be on the local network.

The scanners and worms that used to operate in IPv4 will still very likely be able to operate in IPv6, but they will just use a different mechanism to do it. Customers unaware that IPv6 is even running on their workstations represent another security risk. They could be using IPv4 primarily but still have an active IPv6 protocol stack running. An attacker may leverage a newfound vulnerability in some aspect of IPv6 and then use that vulnerability gain access to the victim's computer. Disabling an unused protocol stack (in this case, the unused IPv6 stack) would appropriately mitigate this risk.

New Potential Risks with IPv6

Any new feature or way of operating could be open to a new form of attack. Here is a list of features that are implemented differently or have slightly different methods than IPv4, and as a result, any manipulation of how the feature works could result in a compromise of the network:

- **Network Discovery Protocol:** Clients discover routers using NDP, and if a rogue router is present, it could pretend to be a legitimate router and send incorrect information to the clients about the network, the default gateway, and other parameters. This could also lead to a man-in-the-middle attack, where the rogue router now has the opportunity to see all packets from the hosts that are being sent to remote networks.
- **DHCPv6:** A rogue router that has fooled a client about being a router could also manipulate the client into using incorrect DHCP-learned information. This could cause a man-in-the-middle attack because the host could be using the address of the rogue router as the default gateway.



- **Hop-by-hop extension headers:** With IPv4, there were IP options that could be included in IP headers. Malicious use of these headers could cause excessive CPU utilization on the routers that receive or forward these packets, in addition to dictating the path the packet should take through the network. There are no IP options in IPv6; instead, there are IPv6 extensions, which can also be misused. One of the IPv6 extension headers is the Routing Header, type 0 (also referred to as RH0). RH0 can be used to identify a list of one or more intermediate nodes to be included on the path toward the final destination. This can enable an attacker to dictate the path a packet can take through the network. By default, Cisco IOS disables the processing of RH type 0 headers on most of its current versions of IOS.
- **Packet amplification attacks:** Using multicast addresses rather than IPv4 broadcast addresses could allow an attacker to trick an entire network into responding to a request. An example is to send a neighbor solicitation request (which is part of the NDP) to the all-hosts multicast address of FF02::1, which would cause all devices to respond. Another example is if a packet is sent with the header extensions set so that a packet is just looped around the network until the *Time-To-Live (TTL)* mechanism expires, and perhaps injecting thousands of these to consume bandwidth and resources on the network devices forwarding them.
- **ICMPv6:** This protocol is used extensively by IPv6 as its NDP. Much potential harm may result from manipulation of this protocol by an attacker.
- **Tunneling options:** Tunneling IPv6 through IPv4 parts of a network may mean that the details inside the IPv6 packet might not be inspected or filtered by the IPv4 network. Filtering needs to be done at the edges of the tunnel to ensure that only authorized IPv6 packets are successfully sent end to end.
- **Autoconfiguration:** Because an IPv6 host can automatically configure an IP address for itself, any trickery by a rogue router could also cause the host's autoconfiguration to be done incorrectly, which could cause a failure on the network or a man-in-the-middle attack as the client tries to route all traffic through the rogue router.
- **Dual stacks:** If a device is running both IPv4 and IPv6 at the same time, but is aware of only one (or is primarily only using one), the other protocol stack if not secured is a potential way that an attacker could remotely access the device. Once access is obtained this way, the attacker could then change one or both IP settings or other configuration options based on what the attacker wants to do next.
- **Bugs in code:** Any newer software has the potential to have bugs, including the software that is supporting the IPv6 features in the network or end-station devices.

IPv6 Best Practices



Implementing security measures at the beginning of a deployment improves the initial security posture instead of waiting until after an attack has occurred. IPv6 best practices include the following:

- **Filter bogus addresses:** Drop, at the end of your network, any addresses that should never be valid source or destination addresses. These are also referred to as *bogon addresses*.

- **Filter non-local multicast addresses:** If you are not running multicast applications, you should never need multicast to be forwarded beyond a specific VLAN. Local multicast is often used by IPv6 (for example, in routing updates and neighbor discovery).
- **Filter ICMPv6 traffic that is not needed on your specific networks:** Normal NDP uses ICMPv6 as its core protocol. A path's *maximum transmission unit (MTU)* is also determined by using ICMP. Outside of its normal functionality, you want to filter the unused parts of ICMP so that an attacker cannot use it against your network.
- **Drop routing header type 0 packets:** Routing header 0, also known as *RH0*, may contain many intermediate next hops, and if followed an attacker could control the path of a packet through a network. The attacker could also use this to create an amplification attack that could loop until the TTL expires on the packet. Cisco routers, by default, drop packets with this type of header.
- **Use manual tunnels rather than automatic tunnels:** If tunneling, do not use automatic tunnel mechanisms such as automatic 6to4, because you cannot control all of them. (They are dynamic.) With the manual tunnels, avoid allowing the tunnels to go through the perimeter of your network, as you will not have tight controls on the contents of the tunneled packets.
- **Protect against rogue IPv6 devices:** *Secure Neighbor Discovery (SEND)* and *router advertisement guard (RA guard)* can help in mitigating rogue routers.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 9-4 lists these key topics.



Table 9-4 *Key Topics*

Key Topic Element	Description	Page Number
Table 9-2	IPv4 versus IPv6	202
Example 9-1	An IPv6 address configured the hard way	205
List	Address types with IPv6	206
Example 9-2	IPv6 interface information	207
Example 9-3	Enabling IPv6 routing and routing protocols	209
List	Best practices common to both IPv4 and IPv6	210
List	Threats common to both IPv4 and IPv6	212
Text	New potential risks with IPv6	213
Text	IPv6 best practices	214

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

amplification attack, RS, RA, NS, NA, eavesdropping, man-in-the-middle attack, spoofing, EUI-64

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 9-5 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 9-5 *Command Reference*

Command	Description
<code>ipv6 address</code>	Apply an IPv6 address to an interface
<code>ipv6 unicast-routing</code>	Enable the router to forward IPv6 packets on behalf of other devices
<code>ipv6 ospf 1 area 0</code>	Enable the interface for OSPF process 1, in area 0

CCNA Security exam topics covered in this part:

- Describe common security threats
- Implement security on Cisco routers
- Describe Cisco Security Manager
- Describe standard, extended, and named IP IOS access control lists (ACL) to filter packets
- Describe considerations when building ACLs
- Implement IP ACLs to mitigate threats in a network
- Describe operational strengths and weaknesses of the different firewall technologies
- Describe stateful firewalls
- Describe the types of NAT used in firewall technologies
- Implement zone-based policy firewall using CCP
- Implement the Cisco Adaptive Security Appliance (ASA)
- Implement Network Address Translation (NAT) and Port Address Translation (PAT)
- Describe Cisco Intrusion Prevention System (IPS) deployment considerations
- Describe IPS technologies
- Configure Cisco IOS IPS using CCP

Part III: Mitigating and Controlling Threats

Chapter 10: Planning a Threat Control Strategy

Chapter 11: Using Access Control Lists for Threat Mitigation

Chapter 12: Understanding Firewall Fundamentals

Chapter 13: Implementing Cisco IOS Zone-Based Firewalls

Chapter 14: Configuring Basic Firewall Policies on Cisco ASA

Chapter 15: Cisco IPS/IDS Fundamentals

Chapter 16: Implementing IOS-Based IPS



This chapter covers the following subjects:

- Designing threat mitigation and containment
- Securing networks via hardware/software/services

Planning a Threat Control Strategy

It is no secret that security is one of the top priorities for most commercial and government networks today. Over the past couple of decades, the methods, speed, and technical skills have improved for those individuals who maliciously are trying to gain access to data that is not authorized for them, to disrupt services for authorized users, or in general to cause harm.

Today, the security design is a mission-critical aspect of the network. This chapter covers design considerations with focus on threat mitigation and containment. The chapter then reviews the software, hardware, and services that corporations have at their disposal to design and implement a secure network.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 10-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 10-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Designing Threat Mitigation and Containment	1–5
Securing Networks via Hardware/Software/Services	6–10

1. Which of the following are defense-in-depth approaches? (Choose all that apply.)
 - a. Performing filtering at the router and the firewall
 - b. Requiring authentication for the administrator to connect to the router, and providing authentication before allowing privileged mode access
 - c. Implementing multiple security features on the firewall only, because of the dedicated appliance having the CPU and resources to implement all of them
 - d. Using an IPS on the network, and using an agent on each of the servers to look for malicious traffic

- 2.** Which is true about a security policy? (Choose all that apply.)
 - a.** Senior management is responsible for creating it.
 - b.** Administrators are responsible for implementing it.
 - c.** Users are responsible for abiding by it.
 - d.** Routers, firewalls, and switches may all be used to enforce it.
- 3.** Why is application layer visibility an important security feature? (Choose all that apply.)
 - a.** So that licensing can be done for those applications
 - b.** To identify any abuse of application layer protocols
 - c.** Because knowing what applications are in use can assist with bandwidth management
 - d.** To identify an attempt to embed unauthorized content in an authorized protocol
- 4.** Which one of the following poses the single greatest risk to an otherwise secured network?
 - a.** Unauthorized device connected to a switch port
 - b.** Rogue DHCP server
 - c.** User clicking a suspicious link and confirming OK
 - d.** MAC address spoofing
- 5.** What is the term for tricking a user into revealing sensitive or confidential information, including information about user credentials?
 - a.** Eavesdropping
 - b.** Cross-site scripting
 - c.** Denial of service
 - d.** Social engineering
- 6.** What tool enables you to centrally manage various models of security appliances and devices?
 - a.** Cisco Security Monitoring, Analysis, and Response System (MARS)
 - b.** Cisco Security Manager (CSM)
 - c.** Cisco Configuration Professional (CCP)
 - d.** Adaptive Security Device Manager (ASDM)

7. Which device can analyze network traffic in real time, generate alerts, and even prevent the first malicious packet from entering the network?
 - a. IPS
 - b. CSM
 - c. IDS
 - d. CCP
8. Which device uses class maps, policy maps, and service policies to implement policies and perform application layer inspection? (Choose all that apply.)
 - a. IOS Zone-Based Firewall
 - b. IOS context-based access control (CBAC)
 - c. ASA
 - d. CSM
9. Which device can provide VPN termination for both remote-access (IPsec and SSL) and site-to-site (IPsec) VPNs?
 - a. IOS router
 - b. IPS
 - c. ASA
 - d. ACS
10. What Cisco service investigates threats and publishes their findings for the benefit of security administrators?
 - a. SIO
 - b. IOS
 - c. OSI
 - d. ISO

Foundation Topics

Designing Threat Mitigation and Containment

You have already read about the various threats that exist in the world today and myriad vulnerabilities within many of the systems inside networks. This section covers some of the guiding principles that, when followed and implemented, can mitigate the threats that networks face today.

The Opportunity for the Attacker Is Real

There are so many opportunities for an attack against a network. Data that is going over the network, if it is not encrypted, is subject to eavesdropping. An attacker who wants to do reconnaissance can do so in many different ways. It could happen at Layer 2 by spoofing the Layer 2 addresses between a device and what that device believes is a Layer 2 address of the default gateway. It could happen if the attacker merges a rogue router into a network and, by manipulating routing protocols, causes traffic to be routed through it. It could happen from a service provider who has full access to the traffic being sent over the service provider's backbone between two areas, or as the data travels from a remote user to the central office. And that is just scratching the surface as far as reconnaissance goes.

Many Potential Risks

Beyond access to raw packets, there are so many ways to access networks today (along with the data they carry), and so many service protocols running (many of which are running unbeknownst to the owner of the computer or server), that software used for reconnaissance, access, or manipulation of the current system is a whole other world of opportunity for an attacker. (We can use the keyword *attacker* to represent any unauthorized individual or system that may be intentionally or unintentionally gaining access to data, manipulating data or systems, or causing harm to the network.) Most of the significant attacks today are financially driven. The leaking of data or loss of data or systems can have a catastrophic financial impact on companies whose networks are compromised.

The Biggest Risk of All

Even after all you can do related to security of the systems and software that make up your network, a huge security vulnerability still exists: end users (ourselves included, as administrators). Users have been provided with credentials to access the network resources, and if an attacker can obtain those credentials, the security of the network is compromised. Social engineering is designed to get information (usually through trickery) from the user that the attacker can then use. Tricking the user by showing a login menu (which is not a real login menu), having the user click a link, or having the user

install a program that implements a vulnerability or back door are common occurrences. Therefore, user training and signed user agreements are critical. User awareness about security risks and their willingness and ability to comply with company policy are some of the toughest challenges in security today.

Where Do We Go from Here?

If you are designing a network from scratch, you might have the rare opportunity to provide input about the technical controls that can be used to implement the initial security policy. Most of the time, the networks we work with have been in place a long time before we arrived, and the security policy has been implemented and usually will have morphed over time (a normal thing) to support the applications and services that were initially on the network or that have been added to the network. Whether you are on a new network or an existing network, be aware every network should have the key components described in Table 10-2.

Table 10-2 *Threat Control and Mitigation Strategy Components*

Feature	Description
Formal process for policy creation, implementation, and review	Senior management, ultimately, is responsible for policy. The job of the network administrator is to implement and enforce through technical and logical controls the policy that has been mandated. When there are changes to be made, a formal procedure, including change control and a written signoff by the person authorized for that change, should be in place. The documented history of change control should be kept. Auditing records detailing which administrators have accessed which systems, along with any changes that were made, should be kept and stored. This can be automated through services such as <i>authentication, authorization, and accounting (AAA)</i> .
Mitigation policies and techniques	A policy should be in place specifying the course of action in response to an attack or threat. Ideally, an automated system could retaliate against the attacker's packets or activities to stop the attack and quarantine the area if possible. Reporting features should be such that the highest-ranking threats can be easily seen and addressed by both systems and administrators. An example of an automated response to an attacker may be a system that dynamically sends out blocking requests to perimeter routers that deny future packets from the attacker, or perhaps sending a TCP reset to the attacker and the proposed victim. A feature such as this could be implemented on an <i>intrusion prevention system (IPS)</i> .



Feature	Description
End-user education and awareness	Because of smart phones, <i>virtual private networks (VPN)</i> , and almost instant access everywhere, data is available at lots of different places. Having an end-user policy, educating the end users, and periodically reviewing and verifying that the end users are aware of their role in protecting the data is a critical piece in mitigating threats. An ounce of prevention here is worth a pound of forensics later if a compromise has occurred.
Defense in depth	Defense in depth refers to a layered security approach, where multiple devices may have overlapping security responsibilities. This is desired so that a single failure of a given system does not represent a significant vulnerability of the entire network.
Centralized monitoring and analysis	Ideally, with multiple devices providing security, you would have centralized monitoring where all the information can be easily sorted and viewed. This information coming in from all these devices should be correlated (which should be an automated process), for an enterprise-wide view of what is really happening. After seeing that an attack is taking place, a centralized tool may have the ability to recommend or even implement an appropriate countermeasure against the attack. (An example is to re-write an <i>access control list [ACL]</i> and apply it on a router that is in the path of the attacker.) External information from global correlation systems could also be used to harden the network against a global threat that may be headed in the direction of the current network.
Application layer visibility	Well-known protocols, in the hands of an attacker, could be manipulated to cause harm to your network in the form of an attack. Application layer visibility is critical to verify whether protocol abuse is occurring (such as malformed requests, tunneling, and so on) so that the network can respond if needed and prevent the malicious traffic or application from proceeding further.
Incident response	The policy should state what will happen and how it will happen as incidents occur on a network. Having a formal policy for this removes the ambiguity of how incidents are reported and the follow-up process documented after the event. The response may be administrative (manual) and automated.

Securing a Network via Hardware/Software/Services

Now that you understand *why* we need network security (from discussions here and in earlier chapters), and now that you have been introduced to *how* to achieve network security (via a security policy), it is time to start looking more at the *where* of network security (that is, within specific pieces of the network infrastructure). This section covers

some of the security-related functionality provided by routers, switches, firewalls, and other specialized appliances and services.

Nearly every network device Cisco makes fits into the category of something (hardware/software) you could use to implement a secure network. Many of these things are discussed in other chapters (in greater detail, including configuration) in this book. In this section, we take a high-level look only.

Switches

Let's start with Layer 2 and work our way up from there. Table 10-3 describes security features available on a Cisco switch. Note that features available on a given switch depend on the model of that switch, the firmware it is using, and the IOS version and which feature set is running.

Table 10-3 *Security Features on Cisco Switches*

Feature	Description
Port security	Limits the number of MAC addresses that a port can learn. This protects against a CAM <i>content-addressable memory</i> (CAM) (also known as the <i>MAC table</i>) overflow. An attacker may attempt to flood bogus source MAC addresses in an attempt to consume all the memory in the table, which would cause the switch to forward unicast frames out all ports in the same VLAN. By launching this attack, the attacker is hoping to see all frames on the VLAN and perform an eavesdropping reconnaissance against the network.
DHCP snooping	An attacker who attempts to place a rogue DHCP server on the network could potentially hand out incorrect <i>Dynamic Host Configuration Protocol</i> (DHCP) information, including the default gateway for the clients to use, which could cause a man-in-the-middle attack and allow eavesdropping by the attacker. DHCP snooping only allows server responses from specifically trusted ports that lead to your authorized DHCP servers. This also protects the DHCP server by rate-limiting how many TCP requests can be sent per interval. This is useful if somewhere an attacker is requesting thousands of IP addresses in an attempt to consume the entire pool on the DHCP server.
Dynamic Address Resolution Protocol (ARP) inspection	Using the information from DHCP snooping or from manually configuring it, a switch can confirm that your traffic includes accurate MAC address information in ARP communications, to protect against an attacker trying to perform Layer 2 spoofing.
IP source guard	This can be used to verify the client on a given port is not doing Layer 3 spoofing (IP address spoofing).
Root guard, BPDU guard, BPDU filtering	These features enable you to control your spanning-tree topology, including resisting a rogue switch's attempt to become root of the spanning tree.



Feature	Description
Storm control	This feature allows the switch to begin clamping on traffic at configurable levels. For example, broadcast storm control could tell the switch to stop forwarding broadcast traffic (or limit it) if broadcasts ever reach more than 50 percent utilization (for example) of the switch capacity.
Additional modules	Modules are supported on various networking devices, which add functionality to that device. Examples include IPS modules, VPN modules, firewall modules, anti-malware modules, and so on. You can expand security services on many network devices, such as routers, switches, and even add on to the functionality of firewalls.

Note that on many multilayer switches, Layer 3 and higher services for security may also be present, such as routing protocol authentication, packet filtering, and so on.

Routers

Routers can add additional security features above and beyond the routing and forwarding they perform on the network. Table 10-4 describes router security features. As with switches, the exact security features available on any router depend on both the hardware and software being used.

Table 10-4 *Security Features of IOS Routers*



Feature	Description
Reflexive access lists	This is mostly for historical purposes, but this was one of the early attempts on Cisco IOS to perform stateful filtering. We discuss stateful filtering in detail in the firewall chapters, but the concept is to not allow any traffic in from the outside world (if it is initiated from the outside). If a user on the inside of your network sends traffic out to a server on the outside network, the reflexive access lists looks at that flow of traffic, creates an <i>access control entry (ACE)</i> , which is the mirror image (swapping the source and destination IP addresses and ports), and dynamically applies that so that the return traffic from the server is allowed. Reflexive access lists are not used much anymore.
<i>Context-based access control (CBAC)</i>	This was the evolution of the IOS router to now support stateful filtering, without creating reflexive access lists. This used to be called the IOS Firewall, because CBAC was the primary feature of the IOS Firewall feature set.

Feature	Description
Zoned-Based Firewall	<p>This replaced CBAC, and is the current recommended way to implement stateful filtering on IOS routers. An entire chapter in this book covers this topic. Zone-Based Firewalls use class maps to identify traffic, policy maps to specify actions to take on that traffic, and a service policy set of commands to put the policy in place.</p> <p>Among other things, a Zone-Based Firewall can do application layer inspection and URL filtering and has other security-related features.</p>
Packet-filtering ACLs	<p>Using standard and extended ACLs, you can implement your policy of what traffic is allowed or denied through the interfaces of the router.</p>
AAA	<p>AAA stands for <i>authentication, authorization, and accounting</i>. The IOS router has extensive support for each of these features and to work with external servers relevant to these features if desired.</p>
VPNs	<p>IOS supports remote-access VPNs using <i>Secure Sockets Layer (SSL)</i> or IPsec. It also supports VPNs in a site-to-site configuration when using IPsec. (SSL is not generally used for site-to-site VPNs.)</p>
IPS	<p>The IOS router can implement an <i>intrusion prevention system (IPS)</i> in software or by using a hardware module in an available option slot. With an IPS function on the router, you can leverage the added security that the routing function currently provides.</p>
Routing protocol authentication	<p>This provides security that prevents an unauthorized router from being trusted or believed as it sends routing updates with an attempt to influence or learn the routing information from another router.</p>
Control plane protection and control plane policing	<p>This enables you to set thresholds and limits for traffic that is directed to the router. In an attempt to overwhelm the router, an attacker might send thousands of packets directly to the router, which by default would have to be processed by the router itself (as opposed to forwarding the packet somewhere else as in the case of the transit packet). The protection and policing set limits on these packets so that CPU can be preserved.</p>
Secure management protocols	<p><i>Secure Shell (SSH)</i> and SSL are supported for managing the router.</p>

ASA Firewall

Some network devices were built for special-purpose functions (for example, the firewall appliance called the *Adaptive Security Appliance [ASA]*). Table 10-5 describes several of the ASA's security features. The exact features available on the ASA depend on both the hardware and software being used.

Table 10-5 *Security Features of ASA Firewalls*



Feature	Description
Stateful filtering	This allows the ASA to remember the state of a connection (for example, a client going out to a web server) and dynamically allow the return traffic back to the client. The firewall can be implemented as a Layer 2 or Layer 3 device and in either case can analyze traffic all the way up to the application layer.
<i>Modular policy framework (MPF)</i>	Used by the ASA (via class maps, policy maps, and service policy rules) to perform simple protocol and application layer inspection and policy enforcement.
URL filtering	Working with statically configured URLs or with a third-party system, the ASA can control which URLs are allowed to be accessed by users through this firewall.
Packet-filtering ACLs	Using standard and extended ACLs, you can implement your policy of what traffic is allowed or denied through the interfaces of the router.
AAA	AAA stands for <i>authentication, authorization, and accounting</i> . The ASA has extensive support for each of these features and can work with external servers related to these features (such as an <i>Access Control Server [ACS]</i> server).
VPNs	ASA supports remote-access VPNs using SSL or IPsec. It also supports VPNs in a site-to-site configuration when using IPsec. (SSL is not generally used for site-to-site VPNs.)
IPS	The ASA can implement an IPS by adding a hardware module to an available option slot on the ASA.
Routing protocol authentication	This provides security that prevents a rogue router from being trusted or believed as it sends routing updates with an attempt to influence or learn the routing information from another router.
Secure management protocols	SSH and SSL are supported for managing the ASA.

Other Systems and Services

Table 10-6 describes other appliances or services that are likely to be used as part of an overall security implementation.

Table 10-6 *Other Appliances and Services Used to Implement a Security Policy*

Device or System	Explanation
IPS	An IPS analyzes network traffic, can report on traffic that it deems malicious or harmful, and can take countermeasures against the offending traffic. This can be implemented as an appliance, as a blade in a 6500 switch, or as a module in an ASA or IOS router. The primary method for identifying problem traffic is through signature matching.
<i>Cisco Security Manager (CSM)</i>	This is an enterprise-level configuration tool that you can use to manage most security devices.
<i>Cisco Security Intelligence Operations (SIO) Service</i>	The SIO researches and analyzes threats and provides real-time updates and best practices related to these threats. They can dynamically deliver the latest breaking news right when it happens. There is also an application for smart phones. You can learn more at http://www.cisco.com/go/sio .



Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 10-7 lists these key topics.



Table 10-7 *Key Topics*

Key Topic Element	Description	Page Number
Table 10-2	Threat control and mitigation strategy components	225
Table 10-3	Security features on Cisco switches	227
Table 10-4	Security features of IOS routers	228
Table 10-5	Security features of ASA firewalls	230
Table 10-6	Other appliances and services used to implement security policy	231

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

Cisco SIO, ASA, stateful filtering

This page intentionally left blank



This chapter covers the following subjects:

- Access control list fundamentals and benefits
- Implementing IPv4 ACLs as packet filters
- Implementing IPv6 ACLs as packet filters

Using Access Control Lists for Threat Mitigation

You can use an *access control list (ACL)* in many different ways, including identifying traffic in a class map for a Zone-Based Firewall on the IOS router or in a class map on an *Adaptive Security Appliance (ASA)* firewall. You can also use an ACL to identify traffic that should be sent over a *virtual private network (VPN)*, given priority treatment, or have *Network Address Translation (NAT)* performed. Many features rely on the ability of an ACL to identify (classify) traffic. ACLs can also be used to apply filtering. Access lists applied for the purpose of packet filtering require a straight yes or no decision about each packet based on a list of rules that are processed from top to bottom. At the interface of the router (or firewall, for that matter), a packet is compared against the list (that you created and applied), and as soon as a match occurs, the packet is then permitted or denied (based on the instructions in the list), and then the router processes the next packet.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 11-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 11-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Access Control List Fundamentals and Benefits	1–4
Implementing IPv4 ACLs as Packet Filters	5–8
Implementing IPv6 ACLs as Packet Filters	9–10

1. Which of the following are advantages of an extended access list over a standard access list when used for packet filtering?
 - a. It can filter based on source address.
 - b. It can filter based on destination address.
 - c. It can filter based on application layer information.
 - d. Logging can be performed.

2. What method is used to indicate that a portion of an IP address in the source packet does not need to be compared to an access list entry?
 - a. Subnet mask
 - b. Mask
 - c. Wildcard mask
 - d. Full IP address required
3. What technique enables you to match on a range of subnets using a single access list entry, without using object groups?
 - a. Wildcard mask, so that matches are done only for the summary of those networks
 - b. Reflexive ACLs
 - c. Time-based ACLs
 - d. Extended named ACLs
4. What happens when an access list has 100 lines and a match occurs on line 14?
 - a. Lines 15 through 100 are parsed as a group object.
 - b. The ACL acts on the packet, and no further list processing is done for that packet.
 - c. The ACL is processed all the way through line 100, to see whether there is a more strict policy that should be applied.
 - d. There cannot be a line 14 because the only lines permitted start with 10 and increment by 10.
5. Which of the following are valid options for creating and applying ACLs in CCP? (Choose all that apply.)
 - a. Use the ACL Editor.
 - b. Go to Interface Configuration.
 - c. Use the ACL Wizard from the Tools menu.
 - d. ACLs may be created in CCP, but they have to be applied using the CLI.
6. What is the benefit of a network object group as it relates to access lists?
 - a. A single object group, that contains many hosts, can simplify the implementation of an ACL.
 - b. Object groups refer only to services such as TCP or UDP ports.
 - c. Object groups can be used as an alternative to ACLs.
 - d. Network object groups, when implemented, use less CPU and resources from the router when implementing access controls that contain them.

7. Which one of the following is probably the single most significant benefit of managing existing ACLs using CCP rather than via the command line?
 - a. Applying access lists to interfaces
 - b. Creating brand-new access lists
 - c. Looking at hit counts on the access list entries
 - d. Rearranging the order of the access list entries
8. What does the **log** keyword do when added at the end of an access list entry?
 - a. It sends an SNMP message.
 - b. It sends an SDEE message.
 - c. It generates a syslog message.
 - d. It causes hit counts to be displayed when viewing access lists.
9. With IPv6, what is significantly different about applying a packet filter to an interface compared to IPv4?
 - a. The syntax is the same at the interface.
 - b. You do not use the keywords for **in** or **out**.
 - c. You use the command **ipv6 access-list** rather than **access-group**.
 - d. You use the command **traffic-filter** instead of **access-group**.
10. If you accidentally implement an IPv6 filtering policy that explicitly denies all inbound IPv6 traffic, which protocol in the IPv6 suite will most likely cause a failure in the network first?
 - a. IPv6 ICMP
 - b. IPv6 UDP
 - c. IPv6 TCP
 - d. Impossible to implement a **deny any any** statement in IPv6 ACLs

Foundation Topics

Access Control List Fundamentals and Benefits

This section covers some of the uses of ACLs, with a focus on the function of filtering. We also look at the options and flavors of ACLs and the mechanics of how they operate.

Access Lists Aren't Just for Breakfast Anymore

Many people who are new to Cisco think that an ACL is just a mechanism used for packet filtering, and that is certainly what most of this chapter is all about. However, a wide range of other functions and features can use the services of an ACL to identify specific traffic based on the matching capabilities that make up the access list. Note that ACLs are also referred to simply as *access lists*, too; the terms are used interchangeably. In general, what you will see is that access lists are used to give a yes or no decision about something. Table 11-2 describes a few of the features that can leverage an access list.

Table 11-2 *Features That Can Use an Access List*

Feature	Description
IOS Inspect class map	This is a class map that is designed to be used with the Cisco IOS router Zone-Based Firewall. The class map can refer to an access list for the purpose of identifying traffic that matches and is permitted in the access list. Traffic that is permitted in the access list is considered a match for the purposes of the class map.
IOS class map	A typical class map could be used for features such as policy-based routing. This class map also has the ability to refer to an access list for the classification (identification) of specific types of traffic.
Routing protocols	Routing protocols, such as <i>Open Shortest Path First (OSPF)</i> , <i>Enhanced Interior Gateway Routing Protocol (EIGRP)</i> , and <i>Border Gateway Protocol (BGP)</i> , can all use the functions of an access list to control the behavior of various aspects of the routing protocol.
<i>Quality of service (QoS)</i>	High-priority can be assigned to specific traffic that is classified (identified) using an access list.
VPN	You can use an access list to identify which traffic is “interesting” (meaning which traffic should be sent through a VPN tunnel). This would be part of a VPN configuration, and traffic that does not match a permit statement in the access list would be forwarded normally and not be sent through the VPN tunnel.

Feature	Description
ASA Firewall Modular Policy Framework	The ASA firewall, which also uses class maps to identify traffic, can have those class maps refer to access lists for the identification of specific traffic.
Network/Port Address Translation (NAT/PAT)	Using policy-based NAT, or even just qualifying which source IP addresses should be translated, you could use an ACL to identify those devices that require translation.
Packet filtering	Packet filtering refers to applying an access list as a filter on an interface to allow the device to control what traffic is allowed through that interface. This is the primary focus for the rest of this chapter.

Stopping Malicious Traffic with an Access List

In reality, the packet-filtering access lists (meaning an access list applied to an interface with the intention of filtering some of the traffic) is not just used to filter malicious traffic. You can also apply it to an interface to enforce a policy. Figure 11-1 serves as a reference point for most of the discussion throughout the rest of this chapter.

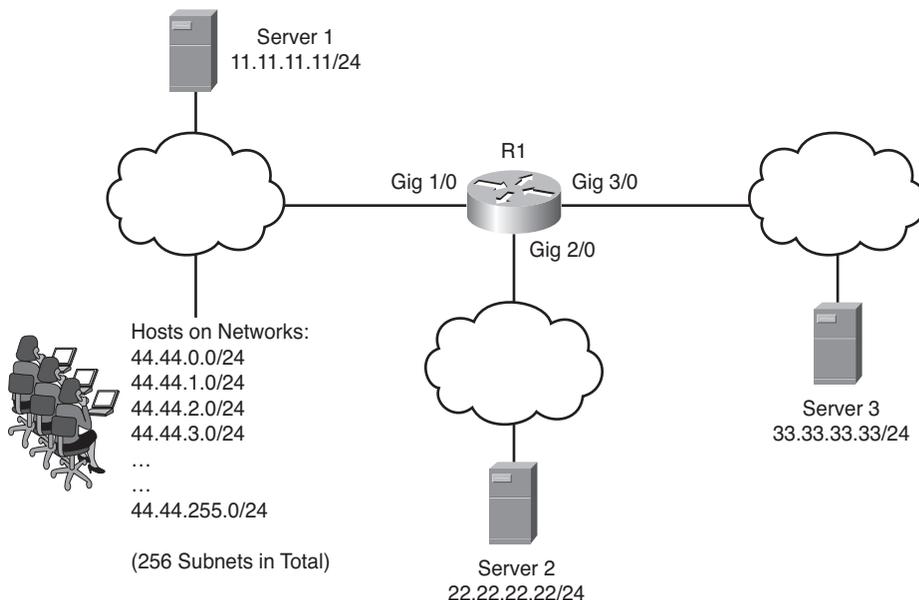


Figure 11-1 *Topology for Packet-Filtering Discussion*

To begin with, we make a few assumptions about this network topology. First, the router has a routing table that provides the knowledge of how to forward to any of the

networks shown. If a router does not have a route, it will not matter what type of access lists we apply because the router will be unable to forward packets out of any interface because it does not know which interface to use.

What Can We Protect Against?



So, with routing in place and the functional working router, let's identify a few types of malicious traffic the router could filter by leveraging a well-written access list either as a filter or used as a classifier along with another feature. Attacks that could be mitigated include the following:

- **IP address spoofing:** An example of this is a device sending a packet but lying about its source IP address. Let's pretend that all the networks in the bottom left of the diagram are under our control, meaning we are responsible for them. If we know that all those networks that begin with 44.44, as shown in the diagram, live off of the gig 1/0 interface, an access list could be created and applied outbound on that interface that denies any packets claiming to come from one of those 44.44 networks. (This would stop packets that come in on the g3/0 or g2/0 that were claiming to be coming from 44.44.x.x, because the ACL would deny the packet outbound on G1/0.) This technique has lots of variations, but the concept is the same: filter the bogus, spoofed traffic, and not forward it. Instead of applying the access list that denies source IP addresses from the 44.44 range outbound on gig 1/0, we also could have applied that same access list inbound on gig 3/0 and gig 2/0. By applying the access list closer to the source of the attacker, we are saving some resources on the router, because the router does not have to do a route lookup, decide which interface to forward out of, only to then deny that packet because of an access list on that outbound interface. Ideally, you should place access lists as close to the source as possible (as in the case of extended access lists) as long as the placement of that access list is implementing the policy that you want to happen. In case you were wondering why, it is because you do not want to forward a packet all the way through your network when you know you are going to kill it anyway.
- **TCP SYN-flood attacks:** This type of an attack is a *denial-of-service (DoS)* attack. These are designed to take down a device, often a server. It is done by sending the initial sequence for a three-way handshake and either spoofing the source address so that the server is trying to reply to yet another victim or having the attacker intentionally fail to reply in an attempt to tie up resources on the server. Firewall techniques, such as the IOS Zone-Based Firewall or the ASA Firewall (both covered later in this book), could be used to mitigate that type of attack, but there are other features such as TCP Intercept that the router can use (in conjunction with an access list to identify the servers to protect) to mitigate these types of attacks, as well.
- **Reconnaissance attacks:** By configuring a filtering ACL and applying it to an interface, you could deny certain types of *Internet Control Message Protocol (ICMP)* or *User Datagram Protocol (UDP)* traffic that may be used for an attacker to learn additional details about the networks behind the router. Traffic that could be denied may include the protocols used by traceroute and ping, among other traffic

that might not be critical for your network and that might be safely denied without any impact to the production network.

- **General vulnerabilities:** By implementing an ACL that uses the attitude of least permission, which means that nothing is allowed except for what you explicitly permit, you can remove a whole range of potential vulnerabilities and weaknesses in your network from the eyes and potential access of the attacker. If the ports and protocols are not available to go through the router, the attacker cannot leverage those ports and protocols against a system on the other side of the router.

The Logic in a Packet-Filtering ACL

When I think about computer networks, many times I will internalize the process that a router or switch is going through, and think of the device as having human characteristics. That idea lends itself to a great analogy about access lists. Take a moment and consider the router from our earlier diagram as being a building. This building has three doors, and each door is labeled. The doors are gig 1/0, gig 2/0, and gig 3/0. As you visualize this, imagine that each of those doors opens and that people can walk in to the building in any one of the doors or out of the building through any of the doors. Now suppose that you want to control which people can come into the building through which specific doors. The first thing we want to do is have a plan of who you want to allow to come in through which doors. Using the diagram from earlier, let's consider just the door gig 1/0. For this example, let's decide that anybody who is wearing a blue hat is not allowed in that door. Let's also say that if somebody is wearing a green hat, he is also not allowed in the door. For everyone else who has either a different colored hat, or no hat at all, we will let them in.



So, that is the policy that we created. But how do you enforce this policy? The first thing you have to do is write out the policy on paper, which would look something like this:

- Line 1. Deny people with blue hats
- Line 2. Deny people with green hats
- Line 3. Permit everyone else regardless of hat color
- Line 4. Permit everyone else not wearing any hat

So, now that you have written out the policy on a piece of paper, which is a great place to start, you need to enforce it. The idea that I have is to hire a security guard. What we can do is take the security guard over to this door named gig 1/0, and tell the security guard to use this list and check every single person who is trying to come into the building through that door from outside. For every person who shows up and is trying to come into that door from the outside, the security guard should take the list, look at line 1, compare that to the person's hat color, which is blue for line 1, and if it matches put the list away and deny access to that person. If another person comes up and has a green hat, take out the list start at line 1, and if it matches go ahead and deny the user, but if it is not a match for line 1, continue on with line 2. If line 2 is a match, based on a green hat, deny that user, as well. So, for every single person who is trying to come into

the building from the outside through that door, the security guard is going to go from top to bottom through the list until either there is a match (based on hat color), which will be followed by denying that user, or a match on line 3 that permits everybody else with a hat in. If someone has no hat, technically it would not match as written, so those people would be denied. If line 4 did not exist, there would be an implicit **deny** (because of the lack of a match that said “permit”).

This is an example of an access list being applied inbound on a specific interface. From this example, does the security guard care what users are trying to leave the building or go out of that interface? The answer is no, because we did not apply a security list outbound on the interface, only inbound. Most of this, of course, you learned in your CCNA certifications, but it is important to realize that access lists when applied as filters can be applied inbound and or outbound. The goal is for the final result to implement your desired security policy.

If an empty access list is applied to an interface, it will not deny any traffic. The implicit deny takes effect only when there is at least one configured line in the ACL. Another interesting quirk is that if configured ACL is applied outbound on an interface, the rules in the ACL apply only to outbound traffic that is being routed through the router (transit traffic) and doesn't have any effect on traffic generated by the router itself (such as a routing update) that is exiting that same interface.

Standard and Extended Access Lists

Now that you know, or at least have revisited, the basic function of an access list for filtering purposes, it is time to review a couple of categories of access lists. A standard access list can only match packets based on their source IP address information. That is it, end of story. An extended access list can match a packet based on the source or destination IP address and most of the content that is contained in the Layer 4 protocol, such as TCP or UDP ports and other Layer 4-relevant information or specific Layer 4 protocols.

Both standard and extended access lists can be identified via a unique number or a unique name, and the functional implementation about what they do does not change based on choosing to use a number rather than a name.

Consider the initial diagram again. If our goal is to prevent the users on the 44.44 networks from reaching server 3, and we want to use a standard access list that filters based on their source addresses, we need to apply that on interface gig 3/0 in the outbound direction. It is unfortunate because all the packets going into the router have to be compared and looked up in the routing table before the router even knows the traffic should be forwarded out of gig 3/0. It is only after the route lookup that the router discovers the **deny** for that source IP address and drops the traffic. We cannot place this standard ACL on interface gig 1/0 because that would prevent those users from getting to server 2. If instead we use an extended access list that specifies that we want to deny the traffic from the users on the 44.44 network only if that traffic is destined for the IP address of server 3, we can apply that access list closer to the users on interface gig 1/0. This will save resources on the router because it does not have to do a route lookup because

traffic going inbound on gig 1/0 would be denied right there. Table 11-3 describes and compares standard and extended IPv4 ACLs.

Table 11-3 *Standard ACLs Versus Extended ACLs*

	Standard ACL	Extended ACL
Numeric range	1–99, 1300–1999.	100–199, 2000–2699.
Option for using names for the ACL instead of numbers	Yes.	Yes.
What they can match on	Source IP only of the packet being compared to the list.	Source or destination IP, plus most Layer 4 protocols, including items in the Layer 4 header of the packet being compared.
Where to place	Unfortunately, these need to be placed relatively close to the destination. Applying these access lists too close to the source may limit that source from reaching other destinations that were not intended to be limited.	Because the extended ACL has the granularity of matching on specific source and destination, you can place these very close to the source of the host who is generating the packet, because it will only deny the traffic to the specific destination and will not cause a loss of service to other destinations that are still being permitted.



Line Numbers Inside an Access List

An access list is a collection of entries, or lines. Sometimes, these are called *access control entries (ACE)*. By default, adding a new line to an access list places that line at the bottom of the list. Based on your policy, that might not be the position where you want this entry in the list (which is processed by the security guard from top to bottom). By default, the router automatically assigns sequence numbers to each line. Normally they begin with 10, and increment by 10 for each new line. If you want to manually put an access list entry strategically in between two already present lines, you can do that by modifying the new sequence number for your new entry. For example, if you have an access list entry at 30, and another one at 40, and you want to insert a new line between those two, you can edit the ACL either in the GUI or at the CLI (using the names access lists editing mode) and specify the new entry with a line number in the range of 31 to 39.

Wildcard Masks

Key Topic

The prerequisites for the CCNA Security include CCNA for Routing & Switching. So, much of this content about access lists might very well be a review for you, which never hurts. One item you have probably come across is the concept of a wildcard mask. The wildcard mask is a binary representation that in essence says that wherever there is a bit on in the wildcard mask, the corresponding bit from the IP address being looked at (the source or destination IP address of the packet being compared) does not have to match. So, if you have an IP address that is 32 bits long, and a wildcard mask that is 0.0.0.255, it means that the last 8 bits of the IP address being checked are not being compared. Anything goes for those last 8 bits, which might be why it is called a *wildcard*. Quite simply, a 0 bit means the initial value must not change, and a 1 bit means you do not care what value goes there. The upcoming configuration section reviews several scenarios to reinforce your knowledge of how wildcard masks operate and how you might use them to create a filter that will match on a range of IP addresses with a single entry by using the appropriate wildcard mask to match a summary of networks or hosts.

Object Groups

Key Topic

When creating access lists, you can run into problems. Suppose, for instance, that you have 15 different servers and that they are not all on the same subnet. So, you cannot create a nice summary range for them. You also need to permit both web and email traffic to these 15 servers. You could create object groups that identify those 15 servers, and then use those object groups embedded into entries in your access list. So, to allow 2 protocols to 15 different servers, you could create an object group for the 15 servers and implement your access list with 2 lines, one for web traffic and one for email traffic to the object group. In reality, this is just a timesaver for the administrator, because behind the scenes the router is still logically checking an access list, which would be 30 lines long, about 2 specific services to each of the 15 different devices. So, in short, object groups are convenient for the configuration of policy and a benefit to the administrator for that reason. We look at an example later.

Implementing IPv4 ACLs as Packet Filters

This section covers how to implement the ACLs using a variety of methods, including *Cisco Configuration Professional (CCP)* and the *command-line interface (CLI)*.

Putting the Policy in Place

The policy is whatever you, as the administrator, have either created or have been asked to implement. The policy states what you want to do, and the implementation of that policy (in our case, using access lists for packet filtering) is what enforces the policy. Refer to Figure 11-1 (shown at the beginning of this chapter), which we use for both our scenarios and the implementation of policy through CCP and the CLI.

Let's begin with a simple warm-up exercise, with the following requirements:

- Must use a standard access list (may be named or numbered).
- Traffic from the subnet where Server 1 is located (Network 11.11.11.0/24) should not be allowed access to Server 3 (Network 33.33.33.0/24).
- Other traffic should not be restricted anywhere.

Armed with this information, we know the standard access list can filter based only on the source address of the subnet where Server 1 is. We can use a wildcard mask in the access list to identify this source subnet. We also know that if we place this access list inbound on gig 1/0 that it will stop traffic that is from Server 1's subnet from reaching Server2, so we cannot place it inbound on interface gig 1/0. Our only other alternative is to place it outbound on interface gig 3/0. At the end of every access list is an implicit **deny**, sort of like the security guard going through the entire list and not seeing the person's name on the list and by default denying entry for that person (that would be for an inbound ACL, or denying exit for an outbound applied ACL). So, to not deny everybody else's traffic, we also need to implement a second entry (*access control entry, ACE*) that permits all the other traffic.

In reality, a standard ACL causes additional filtering because any traffic from the subnet of Server 1 will not be allowed out of interface gig 3/0, regardless of its destination. However, given the requirements, it is our best option (short of using an extended ACL) about placement.

To create and apply an access list using CCP, after selecting the appropriate router navigate to **Configure > Router > ACL > ACL Summary**. It is an excellent idea to see whether any access lists are currently in place before creating new ones, and the Summary page is a great place to determine whether any are present, as shown in Figure 11-2.

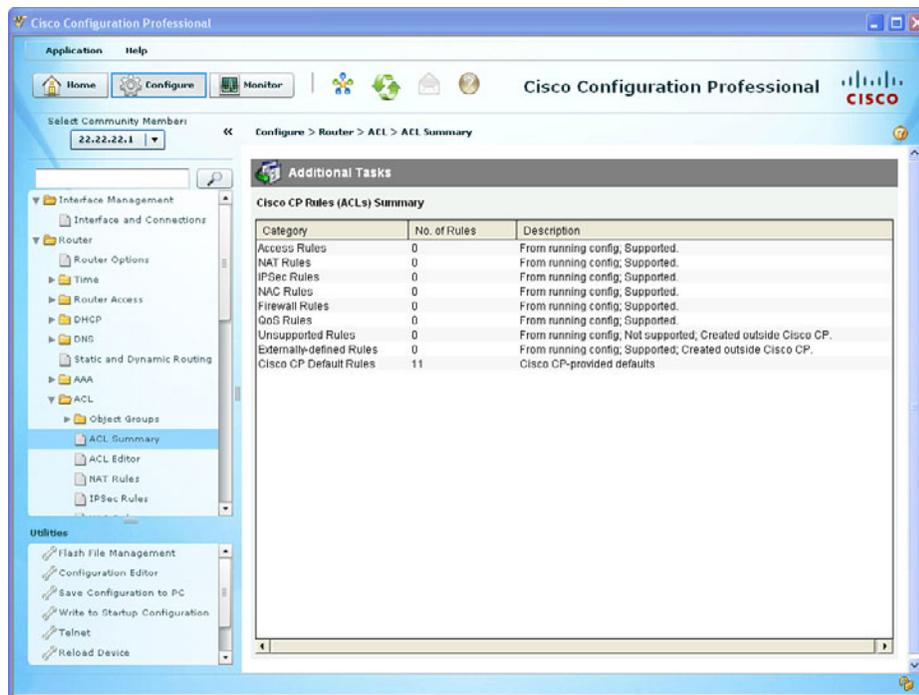


Figure 11-2 ACL Summary Page in CCP

After viewing the current access lists that are in place, if any, you are ready to create a new access list. To do that, in the navigation bar on the left select **ACL Editor**, which is right below the ACL Summary option and click **Add**; you can now create a new rule. You can specify the name or number of this rule and whether it is a standard or extended rule by using the drop-down for the type. By default, there are no entries (individual lines) in this access list, but you can add them by clicking the **Add** button and putting the details for the first entry in the standard access list. Figure 11-3 shows adding an entry to the access list.

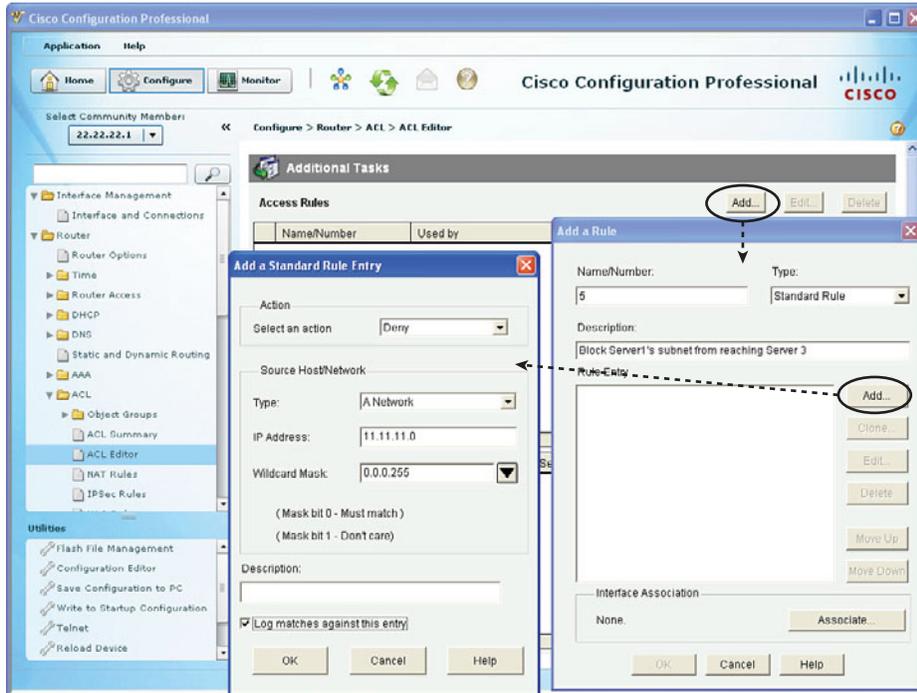


Figure 11-3 Creating a New Standard Access List

You repeat this add process to add a second line in the ACL to permit all other source IP addresses. Figure 11-4 shows the second line.

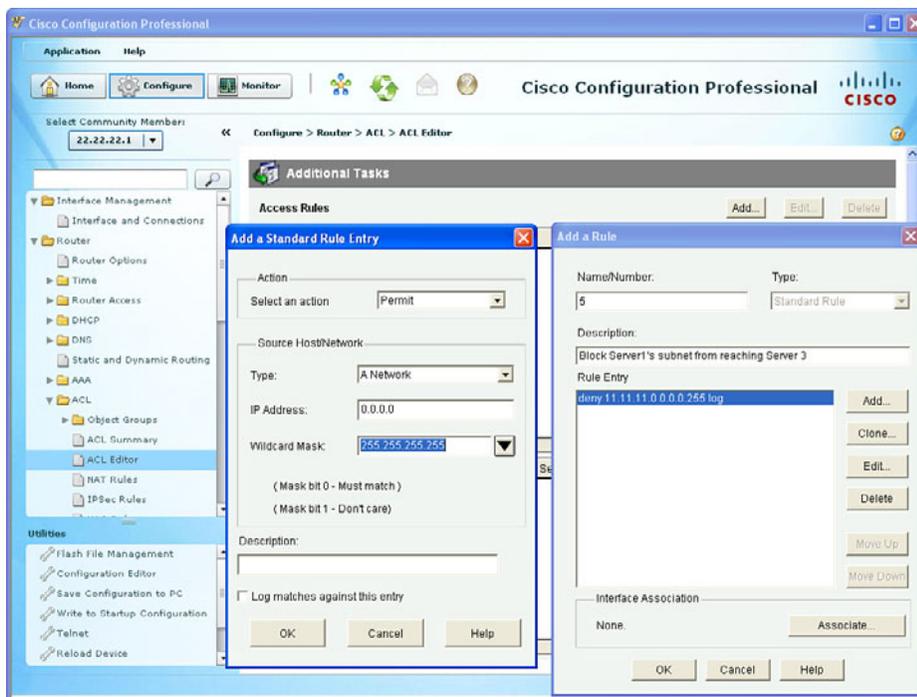


Figure 11-4 Adding a Second Line in the ACL

To modify the order of an access list, you just highlight an entry and click the **Move Up** or **Move Down** button, as appropriate, as shown in Figure 11-5.

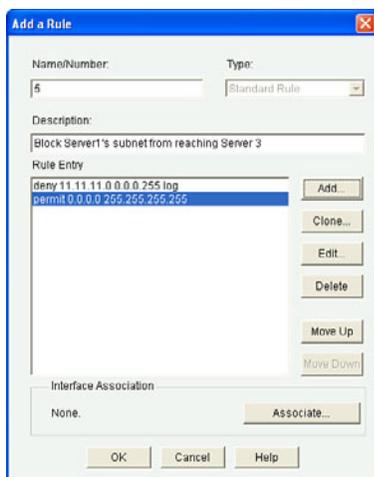


Figure 11-5 Moving an Entry in the Access List Up or Down

After confirming the access list is the way you want it, click **OK** to deliver the configuration to the router. Example 11-1 shows how to implement this access list using the CLI.



Example 11-1 Using the CLI to Implement an Access List

```

! adding comments in the form of remarks is helpful in remembering
! what a specific portion of the access list was intended for
R1(config)# access-list 5 remark Block Server1's subnet from reaching Server 3

! using the log keyword at the end of the ACL entry (ACE) will create
! syslog messages regarding this line is being matched. The syslog messages
! could be viewed wherever they are being sent, such as from the buffer memory
! or at a syslog server.
R1(config)# access-list 5 deny 11.11.11.0 0.0.0.255 log

! This last line of the access list is critical, to permit any traffic
! that wasn't
! previously denied. Without this last line, all other traffic would be denied
! where this access list is applied ( based on the direction of the traffic and
! which direction the access list is studying the traffic
R1(config)# access-list 5 permit 0.0.0.0 255.255.255.255

```

The access list is shown here in the editor with the top portion showing where it is applied (we have not applied it anywhere yet), and the bottom portion of the screen shows the detailed entries in the access list, as shown in Figure 11-6.

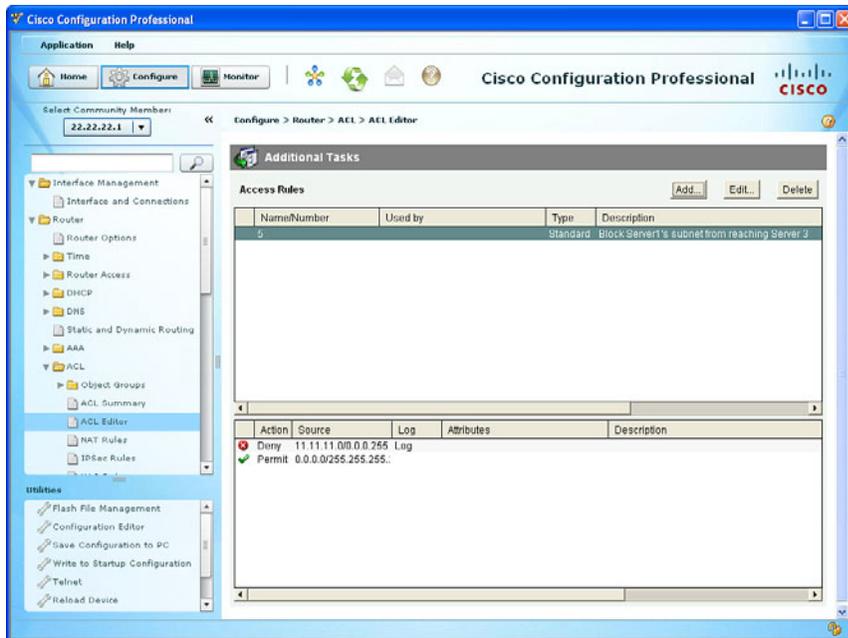


Figure 11-6 Viewing Access List Details

To complete this task, we need to apply this access list outbound on gig 3/0, per our plan. There are a couple of options for applying the access list to an interface. The simplest method, if you have just created an access list, is to click the **Associate** button while editing an access list. From the pop-up window, you can specify the access list you want to apply to the interface, including in which direction, as shown in Figure 11-7.

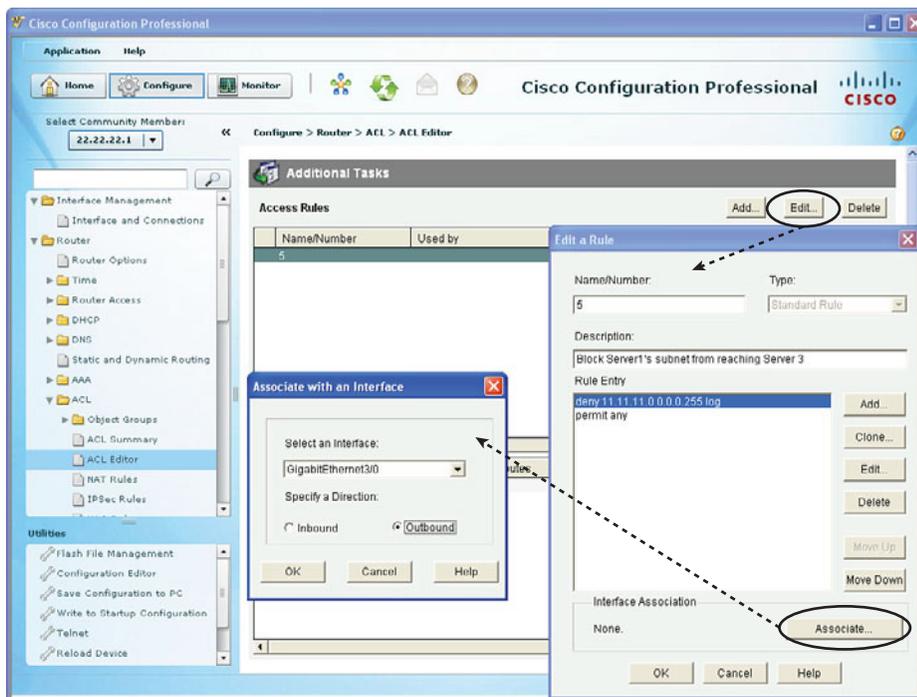


Figure 11-7 Applying the Access List to an Interface

To apply this access list to the interface using the CLI, use the syntax shown in Example 11-2.

Example 11-2 Apply the Access List to an Interface

```
! Move to interface configuration mode
R1(config)# interface GigabitEthernet3/0

! apply the access list using the access group command, with the keyword out
R1(config-if)# ip access-group 5 out
```



Another option is to navigate to **Configure > Interface Management > Interface and Connections**, edit the properties of an interface, and then select the access list from a drop-down menu, as shown in Figure 11-8.

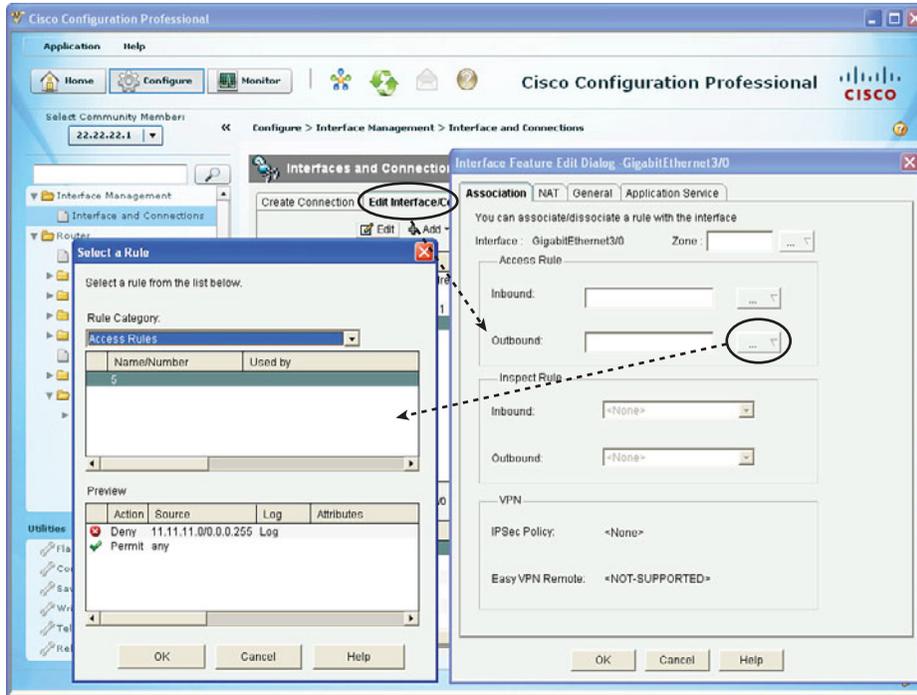


Figure 11-8 Applying an Access List to an Interface

Let's work through another scenario together with a new set of requirements, including the following:

- Devices on a network 44.44.1.0/24 should be allowed only web access to Server 2 and Server 3.
- Devices on the other subnets of 44.44 should not be allowed *any* access to Server 2 and Server 3.
- All other traffic should be permitted.

To implement this, you use an extended access list because it best fits the need of filtering based on both source and destination addresses. You are likely going to apply this inbound on gig 1/0 because that is closest to the source. Because you are doing similar filtering for the two servers, you could create an object group that represents those two servers, and use it in the ACL. Using object groups, you can implement this policy with three lines. To deny all the other 44 networks, you use a wildcard mask to summarize all those networks into a single line for the **deny** statement.

You can choose one of two ways in the GUI to do this. You can start in the Interface and Connections section and edit the interface parameters, which will enable you to additionally create an ACL and apply it, or you could go to the ACL Editor, create the ACL there, and associate it with the appropriate interface.

You saw both options in the previous figures, and because I am currently in interface configuration mode, we are going to start right there. So, navigate to **Configure > Interface Management > Interface and Connections** and click the **Edit** button after highlighting the interface you want to apply it to, or just double-click the interface to bring up the Edit menu. In our case, gig 1/0 currently has no access list associated with that interface. We now are going to apply the rule inbound on gig 1/0, so we click the **Options** button across from the keyword Inbound. This gives us the options to create a new ACL, remove a rule association from this interface, or select an existing rule. In our case, we create a new rule, as shown in Figure 11-9.

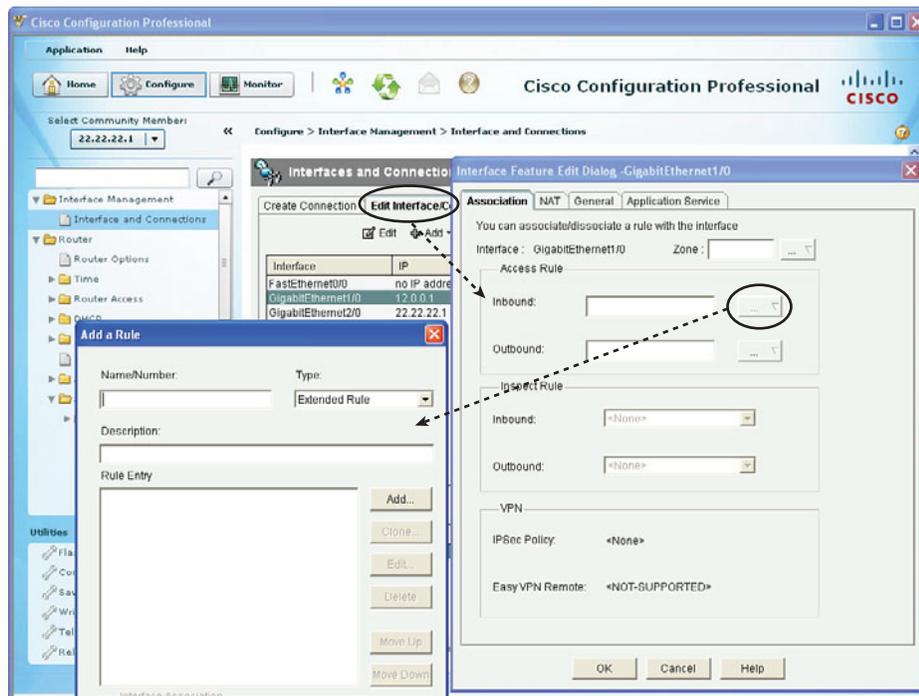


Figure 11-9 *Creating a New Rule*

Just as before, we click the **Add** button and select the ports and IP addresses and protocols that we want to implement in this access list. If you put them in the wrong order, you can use the **Move Up** and **Move Down** options to reorder the entries in the access list as discussed previously.

Before we started creating rules, we decided that we wanted to use an object group to represent the two servers. We really should create the object group before we try to reference that object group in an access list. Some interfaces (for example, on the ASA, covered in a later chapter) provide pop-up options for creating object groups on-the-fly. It makes sense to have every bit planned out and to create object groups before referring to them in an access list. Everything you plan out related to security should be well thought out, so this is a good exercise.

To create an object group, navigate to **Configure > Router > ACL > Object Groups > Network Object Groups**. For some service object groups, you can identify different protocol and port numbers in the group for the same reason (thus simplifying the creation of access lists), or you might need to permit or deny the same types of services (web, email, and so on) by putting all these into an object group configuration. For now, we create a network object group that has the IP address of two hosts: Server2 and Server 3.

To create an object group from this location, click **Create**, as shown in Figure 11-10.

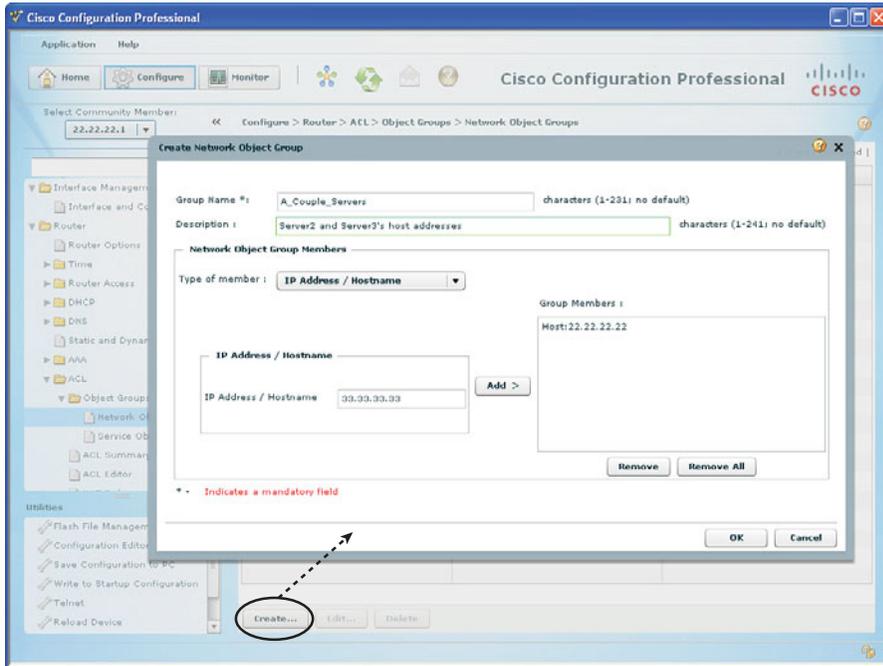


Figure 11-10 *Creating a Network Object Group*

From the Type of Member drop-down list, you can select an individual hostname, a network, a range of IP addresses, or even an existing network object group that could be nested into this new object group you are creating. In this example, we specify that we are adding individual IP address/hostname, and add the two individual IP addresses of 22.22.22.22 and 33.33.33.33. After they are both listed as group members on the right, we click **OK** to continue. There are two main categories for object groups. There are network object groups, which identify devices based on IP address, and could be a network, host, or a range of hosts. The other type of object group is called a service object group, and they can specify TCP, UDP, and a collection of ports that represent common services such as TCP port 22 for *Secure Shell (SSH)*, TCP port 443 for *Secure Sockets Layer (SSL)*, and so on. There is also support for ICMP-related messages, and again the goal is the same: to simplify the creation of access lists by lumping the common services that are needed into an object group and refer to that object group in an entry in the access list.

To create this same object group at the CLI as with the CCP, use the syntax shown in Example 11-3.

Example 11-3 *Create a Network Object Group*



```

! name the object group, and the type (in this case it's a network
! object group)
R1(config)# object-group network A_Couple_Servers

! add a description if desired
R1(config-network-group)# description Server2 and Server3's host addresses

! and the two hosts that will be identified by this object group
R1(config-network-group)# host 33.33.33.33
R1(config-network-group)# host 22.22.22.22

```

Now we are ready to call on this object group from within the access list configuration. As we create the access list and add entries, one of the options when supplying the source or destination address information is to select one of the object groups that is currently on the system. (We have created only one so far). Example 11-4 shows the access list required to implement our solution, which includes the object group we just created.

Example 11-4 *Using Object Groups as Part of the ACL*



```

! create the named or numbered access list, as long as it is extended
! in this example we're using a named access list
R1(config)# ip access-list extended IINS_Extended_ACL_Example

! you can add comments using the remark command to your ACL's if desired
R1(config-ext-nacl)# remark This ACL uses object groups

! this entry permits TCP traffic from the 44.44.1.0/24 network if the
! traffic is destined for the two servers identified by the object group,
! and if the destination port is TCP 80 ( Web services)
! we could add logging with a login keyword at the end of each entry if
! desired
R1(config-ext-nacl)#permit tcp 44.44.1.0 0.0.0.255 object-group A_Couple_
Servers eq www

! next we deny all the other 44.44 networks including 44.44.1 from any further
! traffic to the servers. Because the access list is ordered from top to bottom,
! this next deny statement would be too late to stop the desired Web traffic
! from the previous line, which is the desired result.
R1(config-ext-nacl)# deny ip 44.44.0.0 0.0.255.255 object-group A_Couple_
Servers

```

```

! now we have a permit for all other traffic that was not previously
! matched.
R1(config-ext-nacl)# permit ip any any
R1(config-ext-nacl)# exit

! Applying this access list inbound on the correct interface is what puts
! the policy into action
R1(config)# interface GigabitEthernet1/0
R1(config-if)# ip access-group IINS_Extended_ACL_Example in

```

To verify the configuration of the ACLs and to see where they are applied, you can visit the ACL Editor again, to see the details, as shown in Figure 11-11.



The screenshot shows the Cisco Configuration Professional ACL Editor interface. The left sidebar displays a tree view of configuration options, with 'ACL Editor' selected under the 'ACL' category. The main window displays the 'Additional Tasks' section, which includes a table of 'Access Rules'. The table has columns for Name/Number, Used by, Type, and Description. Two rules are listed: a standard ACL (5) and an extended ACL (IINS_Extended_ACL_Example). The extended ACL is highlighted, and its details are shown in a table below. The details table has columns for Action, Source, Destination, Service, Log, and Attributes.

Name/Number	Used by	Type	Description
5	GigabitEthernet1/0 Outbound	Standard	Block Server1's subnet from reaching Serv
IINS_Extended_ACL_Example	GigabitEthernet1/0 Inbound	Extended	This ACL uses object groups

Action	Source	Destination	Service	Log	Attributes
Permit	44.44.1.0/0.0.0.255	A_Couple_Servers	dest www/tcp	Log	
Deny	44.44.0.0/0.0.255.255	A_Couple_Servers	ip	Log	
Permit	any	any	ip		

Figure 11-11 *Verifying the Details of the ACLs*

Here, we can see the two ACLs that we created, and as we highlight each one, the bottom portion of the screen shows us the entries in the ACL. The top portion also indicates where the ACL is applied.

Monitoring the Access Lists

At the CLI, we can see the details about the access lists, including their sequence numbers and how many times each entry in the access list has been matched. We can also view interface-related details, to see whether access lists are applied to those interfaces. Example 11-5 shows the monitoring of these ACLs.

Example 11-5 *Monitoring ACLs from the CLI*

```

! the command show access-list, or show ip access-list will show all of your
! ACLs that you have configured. If you have access control lists other than
! for IPv4, using the IP keyword with the show command will filter the output
! and only show the IP access lists for IP version 4

! at the end of each entry, if there have been matches for that entry they will
! show up inside parentheses
R1# show access-lists
Standard IP access list 5

! notice the sequence numbers starting with 10
 10 deny 11.11.11.0, wildcard bits 0.0.0.255 log (3711 matches)
 20 permit any (33 matches)

! the output is now showing us the next access list which is the named
! extended access list
Extended IP access list IINS_Extended_ACL_Example
 10 permit tcp 44.44.1.0 0.0.0.255 object-group A_Couple_Servers eq www
    log (7 matches)
 20 deny ip 44.44.0.0 0.0.255.255 object-group A_Couple_Servers log (8
    matches)
 30 permit ip any any (4624 matches)

! to view the IP related information on an interface, use the following
! command
! in the output to indicate whether or not there is a filtering ACL
! applied, and if so which direction it is applied
R1# show ip int g3/0
GigabitEthernet3/0 is up, line protocol is up
Internet address is 13.0.0.1/24
Broadcast address is 255.255.255.255
Address determined by non-volatile memory
MTU is 1500 bytes
Helper address is not set
Directed broadcast forwarding is disabled
Multicast reserved groups joined: 224.0.0.10
Outgoing access list is 5

```

```
Inbound access list is not set
Proxy ARP is enabled
Local Proxy ARP is disabled
Security level is default
Split horizon is enabled
ICMP redirects are always sent
ICMP unreachable are always sent
ICMP mask replies are never sent
IP fast switching is enabled
IP fast switching on the same interface is disabled
IP Flow switching is disabled
IP CEF switching is enabled
IP CEF switching turbo vector
IP CEF turbo switching turbo vector
IP multicast fast switching is enabled
IP multicast distributed fast switching is disabled
IP route-cache flags are Fast, CEF
Router Discovery is disabled
IP output packet accounting is disabled
IP access violation accounting is disabled
TCP/IP header compression is disabled
RTP/IP header compression is disabled
Policy routing is disabled
Network address translation is disabled
BGP Policy Mapping is disabled
Input features: MCI Check
Output features: IPsec or interface ACL checked on pre-encrypted clear-
text packets
WCCP Redirect outbound is disabled
WCCP Redirect inbound is disabled
WCCP Redirect exclude is disabled
```

```
R1# show ip int g1/0
GigabitEthernet1/0 is up, line protocol is up
Internet address is 12.0.0.1/24
Broadcast address is 255.255.255.255
Address determined by non-volatile memory
MTU is 1500 bytes
Helper address is not set
Directed broadcast forwarding is disabled
Multicast reserved groups joined: 224.0.0.10
Outgoing access list is not set
Inbound access list is IINS_Extended_ACL_Example
Proxy ARP is enabled
Local Proxy ARP is disabled
Security level is default
```

```

Split horizon is enabled
ICMP redirects are always sent
ICMP unreachable are always sent
ICMP mask replies are never sent
IP fast switching is enabled
IP fast switching on the same interface is disabled
IP Flow switching is disabled
IP CEF switching is enabled
IP CEF switching turbo vector
IP CEF turbo switching turbo vector
IP multicast fast switching is enabled
IP multicast distributed fast switching is disabled
IP route-cache flags are Fast, CEF
Router Discovery is disabled
IP output packet accounting is disabled
IP access violation accounting is disabled
TCP/IP header compression is disabled
RTP/IP header compression is disabled
Policy routing is disabled
Network address translation is disabled
BGP Policy Mapping is disabled
Input features: Access List, MCI Check
WCCP Redirect outbound is disabled
WCCP Redirect inbound is disabled
WCCP Redirect exclude is disabled
R1#

```

If you want to clear the counters on the lines of the access list, you can use the command **clear ip access-list counters**. The counters will reset to zero, and this is handy in troubleshooting real-time traffic in relation to access lists. The counters increment regardless of whether the keyword **log** is added as part of the entry.

To Log or Not to Log

The **log** keyword generates a syslog message when the line is matched. In an attempt to not overwhelm any device monitoring syslog messages, a syslog message is generated the first time the match is made, and then after a waiting period it sends a summary syslog message indicating the total number of hits that entry had since the waiting period began, and then it starts the waiting period again. (You can change these defaults, if desired, to trigger on every packet that is matched.) If you use the default logging summarization, an ACL entry with the **log** keyword that is matched 4000 times within 1 minute generates significantly fewer syslog messages than 4000 individual messages. Any destinations that you have configured on the router to send syslog messages to will receive a copy of the syslog messages. As you learned in an earlier chapter, to configure syslog destinations, you navigate to **Configure > Router > Logging** and click **Edit**. You

can then configure the details for logging, including destination syslog servers and what level of syslog messages you intend to send. Figure 11-12 shows an example of this.

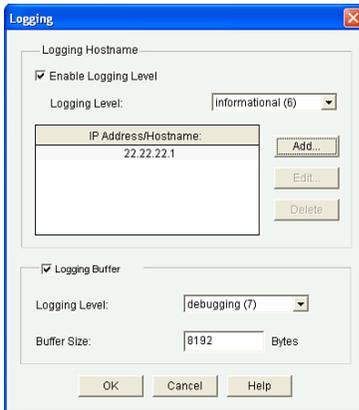


Figure 11-12 Specifying Syslog Destinations

To view the log files that are stored in memory on the router, either issue the command `show log` at the CLI, or from within CCP, navigate to **Monitor > Router > Logging**, display the **Syslog** tab, and scroll through the syslog messages in memory on the router there, and even use the search features to look for a specific syslog message, or filter based on the logging level. Figure 11-13 shows an example of this.

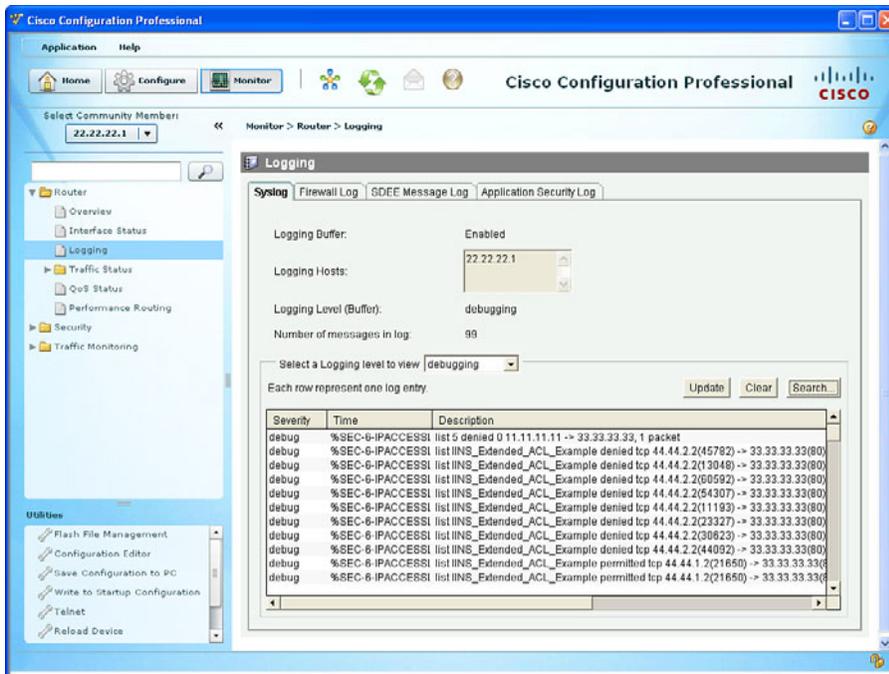


Figure 11-13 Viewing Logs on the Router from Within CCP

The next tab to the right is the Firewall Log tab, and this shows the log of attempted connections that were denied by the firewall, and in the bottom half you can see details about the top attack ports (based on what was denied) or the top attackers (based on the source IP addresses that were denied in the packets), as shown in Figure 11-14.

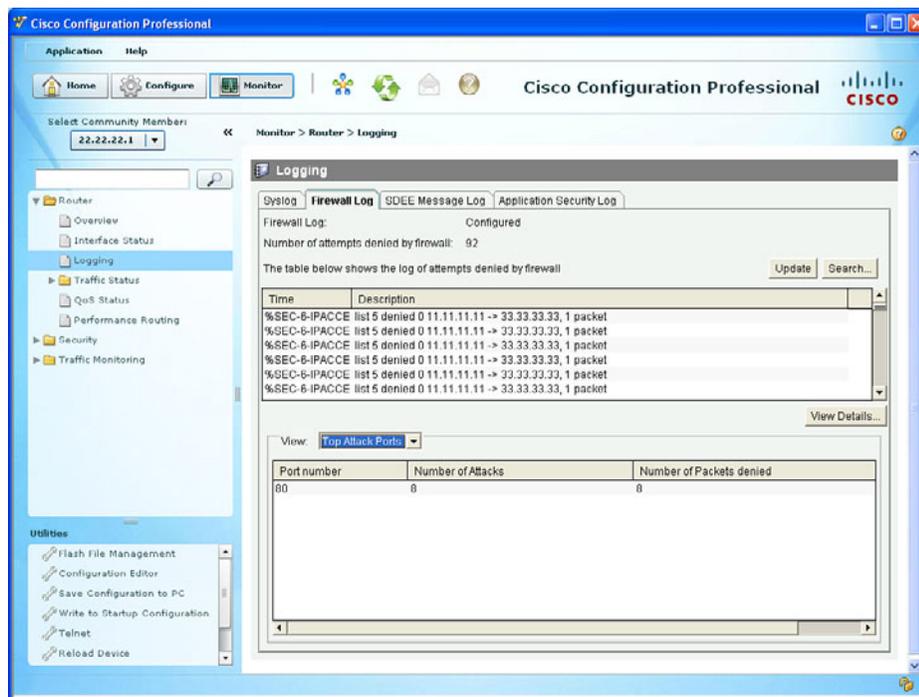


Figure 11-14 Firewall Log Details

The logged ACLs in the Firewall Log tab show up even if a fully configured Zone-Based Firewall is not configured (because the `log` keyword is used with the ACL).

It is important to remember that anything you log and how you log it is an action that the router, switch, or firewall has to take based on your configuration. Although you might have a burning desire to log everything “just in case,” keep in mind that it is possible to overwhelm the device based on what you are logging and exactly how often that occurs based on traffic load. Log what is necessary to wherever you will actually *look* for the information that you have logged. Monitoring is a significant part of the security design and ongoing maintenance for a network. So, make sure you pay attention to what you have decided to log.

Implementing IPv6 ACLs as Packet Filters

Because IPv6 is a different protocol than its predecessor (IPv4), the commands used to implement packet filtering differ slightly. This section examines how to implement packet filtering for IPv6.

The implementation of packet filtering for IPv6 is done for the same reasons you do it for IPv4: to filter traffic that should not be allowed based on policy, and to restrict traffic that is obviously spoofed. Because IPv6 has different ways of operating, compared to its predecessor, you also need to be aware that the attackers may leverage some of this new functionality for reconnaissance or access attacks. Current versions of Cisco routers implement defaults that attempt to mitigate problems such as not allowing IPv6 packets with routing header (RH0) extensions. If some of the routers in the network process these headers, filters to stop these packets should be implemented. Tunneling is a necessary step for many networks as they merge IPv6 into existing networks. When tunneling traffic, the inside packet details are not as visible during the traffic's transit over the tunnel, so filtering based on the internal packet content can be tricky.

The objective here is to familiarize you with the filtering process for IPv6 and to show an example doing just that. IPv6 packet-filtering highlights include the following:

- Can filter based on source and destination addresses.
- Can filter based on source and destination ports.
- Can filter based on the presence of a next header.
- There is an implicit **deny** at the end of the access list, with the exception of neighbor solicitation (NS) and neighbor advertisement (NA) packets. NS and NA packets are implicitly allowed.. (Note that if including an explicit **deny** you should explicitly permit these (NS and NA), before your deny if IPv6 is to function properly.)
- If an empty access list (and access lists without any entries, which is really just a name) is applied to an interface as a filtering access list, it will not deny any traffic. This is the exact same behavior as IPv4 packet-filtering access lists. This can happen if a valid access list is applied to an interface and then the access list is deleted but the interface configuration still shows that it is applied. In this scenario, that access list will not filter any traffic; instead, it behaves as if no access list is in force at all.
- Reflexive and time-based access lists are supported, just like on the IOS for IPv4. A reflexive access list was an attempt at performing stateful inspection, using ACLs that created dynamic entries based on the initial traffic and what the expected return traffic would look like. The dynamic entries permit the reply traffic in a similar way as stateful firewall do by default today. You learn more about stateful packet inspection in the upcoming firewall chapters.
- You can filter on IPv6 extension headers.

For our simple example of filtering with IPv6, let's use Figure 11-15.

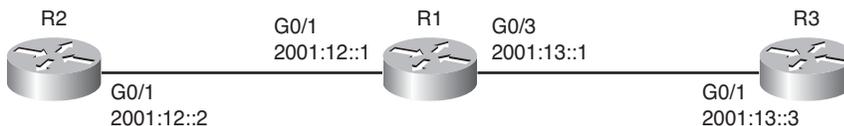


Figure 11-15 *Topology for This IPv6 Filtering Discussion*

R1 is directly connected to the two IPv6 networks in the diagram. R1 knows that any packets coming in on G0/3 (which would be from R3, and possibly any networks that R3 is also connected to) should never have an inbound packet there with a source address of 2001:12::/64 because that network lives off of the G0/1 interface (it is directly connected). To filter any IPv6 packets that contain bogus source address, we can create a filter and apply it inbound on G0/3I, as shown in Example 11-6.

Example 11-6 *Creating an IPv6 Access List and Applying It as a Filter*



```

! Create the access list
R1(config)# ipv6 access-list BOGUS_SOURCE_FILTER

! specify the source network address that should be denied in any packet
! this is because that network is directly connected to g0/1
R1(config-ipv6-acl)# deny 2001:12::/64 any

! permit all other IPv6 traffic
R1(config-ipv6-acl)# permit any any
! go to the interface where we will be applying this traffic filter

R1(config)# int g0/3

! the syntax for applying an IPv6 packet filter, uses different syntax
! than IPv4. The name of the filter is still there, as well as the
! direction
R1(config-if)# ipv6 traffic-filter BOGUS_SOURCE_FILTER in

! we can verify the ACL is applied
R1(config-if)# do show ipv6 int g0/3
GigabitEthernet3/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::C800:A8FF:FE41:54
  No Virtual link-local address(es):
  Global unicast address(es):
    2001:13::1, subnet is 2001:13::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::9
    FF02::1:FF00:1
    FF02::1:FF41:54
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachable are sent
  Input features: Access List
  Inbound access list BOGUS_SOURCE_FILTER
  ND DAD is enabled, number of DAD attempts: 1

```

```
ND reachable time is 30000 milliseconds (using 34967)
ND advertised reachable time is 0 (unspecified)
ND advertised retransmit interval is 0 (unspecified)
ND router advertisements are sent every 200 seconds
ND router advertisements live for 1800 seconds
ND advertised default router preference is Medium
Hosts use stateless autoconfig for addresses.

! we can verify the access list, see the sequence numbers
! as well as number matches. The matches on sequence 20
! are very likely due to routing protocol updates, and possible neighbor
! discovery
R1# show ipv6 access-list
IPv6 access list BOGUS_SOURCE_FILTER
    deny ipv6 2001:12::/64 any sequence 10
    permit ipv6 any any (12 matches) sequence 20
```

Be careful when filtering IPv6 ICMP because ICMP is a key protocol used for dozens of functions, including neighbor discovery, router solicitations, router advertisements, and many more. One way to test an ACL before telling it to deny traffic is to include it as a **permit** statement, including logging, and then you can investigate the details of the traffic that otherwise you might have inadvertently denied. It is better to be safe than sorry.

If you do intend on locking down the ICMP IPv6 traffic, see RFC 4890, *ICMPv6 Filtering Recommendations*.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 11-4 lists these Key Topics.

Table 11-4 *Key Topics*

Key Topic Element	Description	Page Number
Text	What can we protect against?	240
Text	The logic in a packet-filtering ACL	241
Table 11-3	Standard ACLs versus extended ACLs	243
Text	Wildcard masks	244
Text	Object groups	244
Example 11-1	Using the CLI to implement an access list	248
Example 11-2	Using the CLI to apply the access list to an interface	249
Example 11-3	Using the CLI to create a network object group	253
Example 11-4	Using object groups as part of the ACL	253
Figure 11-11	Verifying the details of the ACLs	254
Example 11-6	Creating an IPv6 access list and applying it as a filter	261



Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Review the NAT Video Included with This Book

Review the bonus video regarding NAT and object groups, and practice the techniques shown in the video.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

packet filtering, spoofed address, SYN-flood attack, standard/extended ACL, numbered/named ACL

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 11-5 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 11-5 *Command Reference*

Command	Description
<code>ipv6 traffic-filter <i>BOGUS_SOURCE_FILTER</i> in</code>	Apply the named IPv6 ACL inbound in interface configuration mode
<code>object-group network <i>A_Couple_Servers</i></code>	Create a named network object group and move to object group configuration mode
<code>permit tcp 44.44.1.0 0.0.0.255 object-group <i>A_Couple_Servers</i> eq www</code>	Permit source traffic from any hosts whose IP address begins with 44.44.1, and allow TCP access to any hosts that are members of the object group, if the destination TCP port is 80 (www)
<code>ip access-group <i>IINS_Extended_ACL_Example</i> in</code>	Apply the named IPv4 access list inbound in interface configuration mode

This page intentionally left blank



This chapter covers the following subjects:

- Firewall concepts and technologies
- Using Network Address Translation
- Creating and deploying firewalls

Understanding Firewall Fundamentals

A firewall in a race car is designed to separate the engine compartment from the driver so that in the event of a problem the driver can be protected from what goes on in the engine compartment. A firewall on a computer network is very much the same concept, and that is to separate one portion of the network from another.

Complete separation means that no network connectivity exists, which does not serve anyone very well. By allowing specific traffic through the firewall, you can implement a balance of the required connectivity and security. Traffic that may be identified as harmful is any traffic that compromises confidentiality, data integrity, or availability for the intended users.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 12-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 12-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Firewall Concepts and Technologies	1–4
Using Network Address Translation	5–6
Creating and Deploying Firewalls	7–8

1. Which firewall methodology requires the administrator to know and configure all the specific ports, IPs, and protocols required for the firewall?
 - a. AGL
 - b. Packet filtering
 - c. Stateful filtering
 - d. Proxy server

- 2.** Which technology dynamically builds a table for the purpose of permitting the return traffic from an outside server, back to the client, in spite of a default security policy that says no traffic is allowed to initiate from the outside networks?

 - a.** Proxy
 - b.** NAT
 - c.** Packet filtering
 - d.** Stateful filtering
- 3.** What does application layer inspection provide?

 - a.** Packet filtering at Layer 5 and higher
 - b.** Enables a firewall to listen in on a client/server communication, looking for information regarding communication channels
 - c.** Proxy server functionality
 - d.** Application layer gateway functionality
- 4.** Which one of the following is true about a transparent firewall?

 - a.** Implemented at Layer 1
 - b.** Implemented at Layer 2
 - c.** Implemented at Layer 3
 - d.** Implemented at Layer 4 and higher
- 5.** What is the specific term for performing Network Address Translation for multiple inside devices but optimizing the number of global addresses required?

 - a.** NAT-T
 - b.** NAT
 - c.** PAT
 - d.** PAT-T
- 6.** What term refers to the internal IP address of a client using NAT as seen from other devices on the same internal network as the client?

 - a.** Inside local
 - b.** Inside global
 - c.** Outside local
 - d.** Outside global

7. Which of the following describes a rule on the firewall which will never be matched because of where the firewall is in the network ?
 - a. Orphaned rule
 - b. Redundant rule
 - c. Shadowed rule
 - d. Promiscuous rule

8. What is the long-term impact of providing a promiscuous rule as a short-term test in an attempt to get a network application working?
 - a. The promiscuous rule may be left in place, leaving a security hole.
 - b. The rule cannot be changed later to more accurately filter based on the business requirement.
 - c. It should be a shadowed rule.
 - d. Change control documentation may not be completed for this test.

Foundation Topics

Firewall Concepts and Technologies

These concepts apply to IOS routers performing firewall services and to dedicated firewalls, which are purpose built for security. This section covers the concepts of firewalls, their strengths and weaknesses, and why they are used in specific situations. This information is used in nearly all networks where security is a concern.

Firewall Technologies

A firewall is a concept that can be implemented by a single device, a group of devices, or even simply software running on a device such as a host or a server. As mentioned in the introduction to this chapter, the function of a firewall primarily is to deny unwanted traffic from crossing the boundary of the firewall. For network traffic, this means that a firewall, in its basic form, could be implemented by the following:

- A router or other Layer 3 forwarding device that has an access list or some other method used to filter traffic that is trying to go between two of its interfaces. This is the primary method that is implemented by an IOS router (using firewall features) or the ASA Firewall.
- A switch that has two VLANs, without any routing in between them which would absolutely keep traffic from the two different networks separated (by not being able to have inter-VLAN communications).
- Hosts or servers that are running software that prevents certain types of received traffic from being processed and controls which traffic can be sent. This is an example of a software firewall.

Objectives of a Good Firewall

Here are some common properties that a good firewall should possess:

- **It must be resistant to attacks:** If a firewall can be brought down or compromised to the point where it allows unwanted access, it thus fails to implement policy correctly. If the firewall is a victim of a *denial-of-service (DoS)* attack, to the point where it cannot provide normal access for users, that is also problem. If there is some vulnerability that an attacker can leverage with an exploit, thus enabling the attacker to modify the firewall configuration, that (of course) is also a problem.
- **Traffic between networks must be forced through the firewall:** If multiple paths exist between network A and network B, and a firewall is controlling the traffic for these connections, but if there are alternative paths, the malicious traffic has the potential to avoid the firewall. So, if there are multiple paths, each of those paths should have the same firewall policy, and very likely will have the same firewall methodology at each point.



- The firewall enforces the access control policy of the organization:** Many times, unfortunately, the tail wags the dog as new firewalls are put into place. Rules are made for that firewall about traffic allowed through the firewall, and then as a result we document the policy. What ideally should happen is that a policy would be created on paper first that identifies the business requirements for which traffic should be allowed through the firewall, and then the rules should be created and applied to the firewall to enforce that policy, in that order.

Firewall Justifications

Companies often make a significant investment in security, including the monies they spend on firewalls. Table 12-2 describes some of the items a firewall can help protect against.

Table 12-2 *Protective Measures Provided by a Firewall*

Reduces the Risk Of	Explanation
Exposure of sensitive systems to untrusted individuals	By hiding most of the functionality of a host or network device, and permitting only the minimum required connectivity to that given system, the firewall reduces the exposure for that system. An example is only allowing web traffic to a specific IP address of a web server on your <i>demilitarized zone (DMZ)</i> . Even if that web server has other services running, those services will not be available to users trying to access those services through the firewall.
Exploitation of protocol flaws	You can configure a firewall to inspect protocols to ensure compliance with the standards for that protocol at multiple layers of the protocol stack. It can also control the amount of time it will allow for a normal connection sequence before saying enough is enough. An example is only allowing a specific amount of time between a <i>Domain Name System (DNS)</i> request and the DNS reply. Another example is the three-way handshake for TCP and the firewall only allowing so much time for that to complete between a client on one side of the firewall and a server being accessed on the other side of the firewall.
Unauthorized users	By using authentication methods, a firewall could control which users traffic is allowed through the firewall and be configured to block on all other traffic based on policy. For example, a firewall could leverage <i>authentication, authorization, and accounting (AAA)</i> services using its local configuration or an <i>Access Control Server (ACS)</i> server.
Malicious data	A firewall can detect and block malicious data, which would stop traffic from reaching the intended destination. This function could also be provided by an <i>intrusion prevention system (IPS)</i> .

We could hope that by just purchasing a firewall and putting it in place that all of our security issues can be put to rest. It should be understood that having a firewall and implementing correctly the policies for an organization is a mitigation step for reducing risk but does not completely eliminate risk. In addition, firewalls do have some limitations, as described in Table 12-3.



Table 12-3 *Potential Firewall Limitations*

Limitation	Explanation
Configuration mistakes have serious consequences.	The firewall's job is to implement a policy. Based on a specific firewall you are dealing with, there are specific ways of implementing features such as <i>access control lists (ACL)</i> , packet inspection, Network Address Translation, authentication, and so on. If the firewall rules are not implemented correctly, it might not do the job of implementing the policy as intended. It takes good technical gear and a good technical configuration on that gear for a successful policy implementation.
Not all network applications were written to survive going through the firewall.	If there is some type of custom application that simply will not work based on the combination of what the application is doing and what the current firewall rules in place are, the choice is to rewrite the application or make an exception in the firewall policy for the application.
Individuals who are forced to go through a firewall might try to engineer a way around it.	If there is a firewall policy, for example, that specifies no instant messaging file transfers are allowed, it might be possible for a user to circumvent that rule by getting creative. Options such as hiding the forbidden instant messaging file transfers inside of another protocol so that it looks like standard HTTP or HTTPS is one thing they may try. This concept is called <i>tunneling</i> and can be done with a variety of protocols. A firewall that is capable of application layer inspection may be able to identify and prevent this type of malicious tunneling.
Latency being added by the firewall.	If a firewall is given a huge job of analyzing all the traffic, it might take a few milliseconds or more per packet for the analysis, and as a result some slight delay may be added to the overall network traffic delivery time.

The Defense-in-Depth Approach

Having just one single point of control/security for your entire network is not wise because if that one single point is misconfigured or fails to implement policy, the network is wide open to all the negative impact that the firewall is trying to prevent. One solution, which is really more an idea than a solution, is to use a defense-in-depth approach or what is known as a layered approach to security.

Let's take a look at an example of a defense-in-depth approach for an average company that has a web server that is publicly available to access. We, as the end user on the outside global Internet, open up a browser and type in the name of the server. Behind the scenes, our browser facilitates a DNS request to find out the IP address of that server. Once we have the IP address of the server, we initiate a TCP session with that server.

As our packets go over the Internet toward that company, the perimeter router at that company is their first line of defense. The router could be checking the source IP address of our packet to verify that it is not a spoofed source IP address (such as the IP address space that is internal to their company, which should never be in the Source IP Address field of the packet coming into their network). On the perimeter router, they also might have some limited access control lists that may immediately drop well-known malicious traffic right there at the edge of their network. Because our packet is not malicious (at least not according to the router), the packet is forwarded toward the corporate network and through the firewall, which is the second line of defense.

The firewall can do all sorts of things at this point before the packet is ever forwarded to the intended server. The firewall could pretend that it is the server and communicate back and forth with us for the three-way handshake to verify that we are not trying to do a SYN-flood attack. The firewall could require us to authenticate before forwarding our traffic any further, and many other rules could be implemented on that firewall. Assuming we pass all the rules for that firewall, our packet is then forwarded to the server.

The connectivity between their firewall and their server is very likely going through a switch on the company's local network or wherever the server is hosted. That switch, in addition, could have filtering implemented as another layer of defense. The packets as they finally arrive at the server may or may not be processed by the server based on software firewall rules that are also running in software on the individual server.

So although that explanation is a little bit involved, it is important for you to understand how you can use a defense-in-depth or layered approach to defend your network. In short, it cannot be just a single device protecting all of your network; it needs to be a team effort by nearly all the devices.

Another reason for this defense-in-depth approach is that not all attacks or malicious traffic is coming from outside networks. Much of it is sourced from internal devices that are already on the inside of the networks, such as end-user machines.

The goal of the firewall, or firewalls, is to reduce risk. It is part of an overall strategy for keeping the network up and keeping data reliable and available. Firewalls do not replace the need for other systems such as a backup or disaster recovery plans, which are additional pieces needed for overall business continuity.

Five Basic Firewall Methodologies

So why don't all firewalls cost the same amount of money? One answer is capacity: The more sessions, packets, and resources a firewall can deal with, the more expensive it is (most likely). The various firewalls also contain different features and have different methods for implementing services.



Here are some basic methodologies for implementing a firewall. Many firewalls can perform one or more of these features simultaneously:

- Static packet filtering
- Proxy server (also known as *application layer gateway [ALG]*)
- Stateful packet filtering
- Application inspection
- Transparent firewall

Let's take a closer look at each one of these, beginning with static packet filtering.

Static Packet Filtering

Static packet filtering is based on Layer 3 and Layer 4 of the OSI model. An example of a firewall technology that uses static packet filtering is a router with an access list applied to one or more of its interfaces for the purpose of permitting or denying specific traffic. One of the challenges with static packet filtering is that the administrator must know exactly what traffic needs to be allowed through the firewall, which can be tricky if you have many users that need to access many servers.

Table 12-4 lists some of the advantages and disadvantages of static packet filtering.

Table 12-4 *Advantages and Disadvantages of Packet Filters*

Advantages	Disadvantages
Based on simple sets of permit or deny entries	Susceptible to IP spoofing. If the access list allows traffic from a specific IP address, and someone is spoofing the source IP address, the access list permits that individual packet.
Have a minimal impact on network performance	Does not filter fragmented packets with the same accuracy as nonfragmented packets.
Are simple to implement	Extremely long access control lists are difficult to maintain.
Configurable on most routers	Stateless (does not maintain session information for current flows of traffic going through the router).
Can perform many of the basic filtering needs without requiring the expense of a high-end firewall	Some applications jump around and use many ports, some of which are dynamic. A static access list may be required to open a very large range of ports to support application that may only use a few of them.

Because packet filtering uses a simple rule set (a packet that comes in or out of an interface where there is an access list applied for filtering), there is a check against the packet with the entries in the access list from top to bottom. As soon as a match occurs,

the access list stops processing the rest of the list and implements the action against the packet, which is either a permit or deny. An extended access list on a Cisco router can use many matching criteria against the Layer 3 and Layer 4 headers, including the following:

- Source IP address
- Destination IP address
- Source port
- Destination port
- TCP synchronization information

Application Layer Gateway

Application layer firewalls, which are also sometimes called *proxy firewalls* or *application gateways*, can operate at Layer 3 and higher in the OSI reference model. Most of these proxy servers include specialized application software that takes requests from a client, puts that client on hold for a moment, and then turns around and makes the requests as if it is its own request out to the final destination.

A proxy firewall acts as an intermediary between the original client and the server. No direct communication occurs between the client and the destination server. Because the application layer gateway can operate all the way up to Layer 7, it has the potential to be very granular and analytical about every packet that the client and server exchange and can enforce rules based on anything the firewall sees.

Table 12-5 lists the advantages and disadvantages of application layer gateways.

Table 12-5 *Advantages and Disadvantages of Application Layer Gateways*

Advantages	Disadvantages
Very tight control is possible, due to analyzing the traffic all the way to the application layer.	Is processor intensive because most of the work is done via software on the proxy server.
It is more difficult to implement an attack against an end device because of the proxy server standing between the attacker and potential victim.	Not all applications are supported, and in practice it might support a specific few applications.
Can provide very detailed logging.	Special client software may be required.
May be implemented on common hardware.	Memory and disk intensive at the proxy server. Could potentially be a single point of failure in the network, unless fault tolerance is also configured.

Stateful Packet Filtering



Stateful packet filtering is one of the most important firewall technologies in use today. It is called *stateful* because it remembers the state of sessions that are going through the firewall. Here is a great example. Suppose that you and I go to an amusement park, and halfway through the day we realize that we forgot something in the car. So, on our way out to retrieve the item, we wonder (at the gate) if we have to pay to get back in. The nice person at the gate explains that she will stamp our hand with a code so that when we return we can show the code and they will let us back in for free. Let's say, for our example, that they also write our names on a list of people who were already on the inside and were going outside to the parking lot with the intention of returning. When we want to come back inside, the people at the gate check the list and see that we have already been on the inside and that we left temporarily. So, they allow us back in.

With a stateful packet-filtering device, for customers on the inside of the corporate network, as they are trying to reach resources on the outside public networks, their packets go to the firewalls on the way out. The firewalls take a look at the source IP address, destination IP address, the ports in use, and other layer for information and remember that information in what is known as a *stateful database* on the firewall. (like writing the names down, from the amusement park example). It is called *stateful* because the firewall is remembering the state of the session (that it was on the way outside, including the ports and IP addresses involved). By default, this same firewall does not allow any traffic from the outside and untrusted networks back into the private trusted inside network. The exception to this is for return traffic that exactly matches the expected return traffic based on the stateful database information on the firewall. In short, the reply traffic goes back to the users successfully, but attackers on the outside trying to initiate sessions are denied by default.

Table 12-6 lists some advantages and disadvantages of stateful packet-filtering devices.

Table 12-6 *Advantages and Disadvantages of Stateful Packet-Filtering Devices*

Advantages	Disadvantages
Can be used as a primary means of defense by filtering unwanted or unexpected traffic	Might not be able to identify or prevent an application layer attack.
Can be implemented on routers and dedicated firewalls	Not all protocols contain tightly controlled state information, such as <i>User Datagram Protocol (UDP)</i> and <i>Internet Control Message Protocol (ICMP)</i> .
Dynamic in nature compared to static packet filtering	Some applications may dynamically open up new ports from the server, which if a firewall is not analyzing specific applications or prepared for this server to open up a new port, it could cause a failure of that application for the end user. If a firewall also supports application layer inspection, it may be able to predict and allow this inbound connection.

Advantages	Disadvantages
Provides a defense against spoofing and <i>denial-of-service</i> (DoS) attacks	Stateful technology, by itself, does not support user authentication. This, however, does not prevent a firewall that implements stateful packet filtering from also implementing authentication as an additional feature.

Application Inspection

An application inspection firewall can analyze and verify protocols all the way up to Layer 7 of the OSI reference model, but does not act as a proxy between the client and the server being accessed by the client. Table 12-7 lists some of the advantages of an application inspection firewall.

Table 12-7 *Advantages of an Application Inspection Firewall*

Feature	Explanation
Can see deeper into the conversations, to see secondary channels that are about to be initiated from the server	If an application is negotiating dynamic ports, and the server is about to initiate one of these dynamic ports to the client, the application inspection could have been analyzing that conversation and dynamically allowed that connection from the server to allow it through the firewall and to the client. This would allow the application to work for the client (through the firewall).
Awareness of the details at the application layer	If there is a protocol anomaly which is a deviation from the standard, an application layer firewall could identify this and either correct the packet or deny the packet from reaching the destination.
Can prevent more kinds of attacks than stateful filtering on its own	Current firewalls today, such as the ASA and Cisco IOS Zone-Based Firewall solutions, have the ability of packet filtering, stateful filtering, and application inspection in a single device. With the additional features, more types of traffic can be classified and then permitted or denied based on policy.

Transparent Firewalls

A transparent firewall is more about how we inject the firewall into the network as opposed to what technologies it uses for filtering. A transparent firewall can use packet-based filtering, stateful filtering, application inspection as we discussed earlier, but the big difference with transparent firewalls are that they implemented at Layer 2.

Most traditional firewalls are implemented as a Layer 3 hop in the network (similar to a router hop), meaning that packets have to go through this device at Layer 3. In a Layer 3 firewall, each of the interfaces has an IP address on a different network, and traffic from one subnet to another that goes through the firewall has to pass the rules on the firewall.

With a transparent firewall, we still have two interfaces, but we do not assign IP addresses to those interfaces, and those two interfaces act more like a bridge (or a switch with two ports in the same VLAN). Traffic from one segment of a given subnet is going to be forced through the transparent firewall if those frames want to reach the second segment behind the firewall. A transparent firewall has a management IP address so that we can remotely access it, but that is all. Users accessing resources through the firewall will not be aware that it is even present, and one of the biggest advantages of using a transparent firewall is that we do not have to re-address our IP subnets to put a transparent firewall in-line on the network.

Even though this is implemented as a Layer 2 device, it still sees all packets that go between its interfaces, and it can still apply all the rules of a normal Layer 3 firewall related to permitting traffic, building a stateful database, and performing application inspection.

Using Network Address Translation

Network Address Translation (NAT) is an important feature that is often implemented on firewalls. This section provides a detailed look at the options that exist for NAT. You can use NAT in combination with the other firewall features previously discussed.

Figure 12-1 serves as a reference point for this discussion.

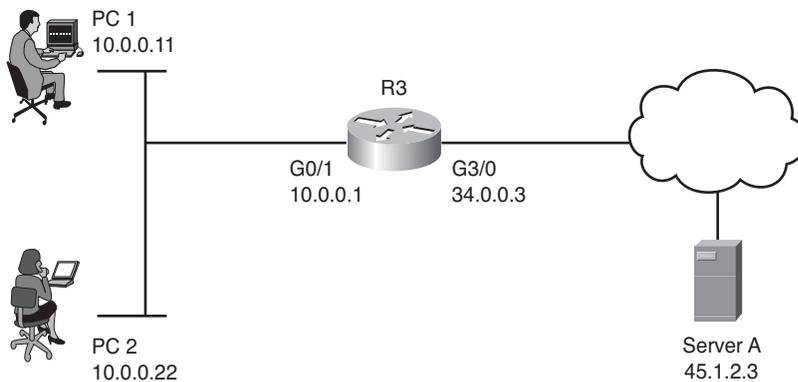


Figure 12-1 Reference Diagram Used for This NAT Discussion

NAT Is About Hiding or Changing the Truth About Source Addresses

It is really true, lying is what NAT does for a living. Let's look at the referenced diagram. The two users are using private address space on the 10 network along with the G0/1 interface of the router. R3 is also connected to the Internet through a service provider through its G3/0 interface. From a security perspective (and a NAT perspective), the

G0/1 interface connects to the “trusted” or “inside” network (from the company’s perspective), and the G3/0 interface connects to the “untrusted” or “outside” network. Now the router itself does not have a problem with IP connectivity to the Internet because the router has a globally reachable IP address (34.0.0.3) in this example. The users are not so fortunate, however, because they are using private IP address space, and that kind of address is not allowed directly on the Internet by the service providers. So, if the users want to access a server on the Internet, they forward their packets to the default gateway, which in this case is R3, and if configured to do so R3 modifies the IP headers in those packets and swaps out the original source IP addresses with either its own global address or a global address from a pool of global addresses (which R3 is responsible for managing, meaning that if a packet was destined to one of those addresses, the routing to those addresses on the Internet would forward the packets back to R3). These are global addresses assigned by the service provider for R3’s use.

The primary device that does NAT is either a router or a firewall. NAT not only allows us to have thousands of users hide behind a single IP address (using *Port Address Translation [PAT]*, discussed momentarily), but it also offers protection of those users because the outside world does not know exactly which devices have been dynamically assigned to which IPs, so initiating a connection from the outside to an inside device is more difficult, (not even mentioning that the ACLs that might additionally prevent the packets from the outside making it back to the inside).

Inside, Outside, Local, Global

Some terms that are used with NAT are often confusing, such as the words *inside*, *outside*, *local*, and *global*. Before looking at these words specifically, here are some options that you can do with NAT that will make these terms more meaningful.

You can translate the IP address of PC1 as its packets go out to the Internet. This is an example of something often referred to as *inside* NAT (performing translation for a packet coming from an inside host). You can also translate the source IP address of device that lives on the outside as its packets come into your local network. This is commonly referred to as *outside* NAT. So, if you were going to do both inside NAT and outside NAT, for PC1 on the inside and Server A on the outside, the router would have to track all that information, including the source IPs on both sides and the translated addresses for both sides. To track all the addresses, the terms shown in Table 12-8 would apply to a single router performing all the NAT.

Table 12-8 NAT Terminology

NAT Term	Description
Inside local	The real IP configured on an inside host, such as PC1.
Inside global	The mapped/global address that the router is swapping out for the inside host during NAT. The outside world sees PC1 coming from this mapped/global address.



NAT Term	Description
Outside local	If performing NAT on outside devices (outside NAT), this is the mapped address of the outside device (such as Server A) as it would appear to inside hosts. If not doing outside NAT on the router, this appears as the normal outside device's IP address to the inside devices.
Outside global	The real IP configured on an outside host, such as the IP on Server A.

Terms that we use are all about establishing a reference point. The *inside* and *outside* have to do with where an IP exists before we touch it. It is either inside our network and control or it is not. The *local* and *global* then have to do with the appearance of the address and may therefore be pre- or post-NAT manipulation. The inside local address as shown in Table 12-8 originated inside our control, and is what the address looks like locally on our network (pre-NAT). The same address post-NAT is inside global, again originating inside our network, but looking at it from a global perspective.

Example 12-1 shows the output for two static NAT entries. One is for PC1, and the other is for Server A. In this example, Server A is made to appear (mapped) as 10.0.0.3 to users on the inside. PC1 is made to appear (mapped) as 34.0.0.11 to users on the outside.

Example 12-1 *One Inside and One Outside Static NAT Translation*

```
R3# show ip nat translations
Pro Inside global      Inside local      Outside local      Outside global
--- ---              ---              10.0.0.3          45.1.2.3
--- 34.0.0.11         10.0.0.11        ---              ---
```

Besides the security provided by NAT, NAT is also used to allow communications between two networks that otherwise would have incompatible IP addressing (such as overlapping addresses), and with the use of *Port Address Translation (PAT)*, we have been able to extend the lifetime of IPv4 for a least a decade longer than it should have been used (because it was running out of IP addresses).

Port Address Translation

PAT is a subset of NAT, and it is still swapping out the source IP address as traffic goes through the NAT/PAT device, except with PAT everyone does not get their own unique translated address. Instead, the PAT device keeps track of individual sessions based on port numbers and other unique identifiers, and then forwards all packets using a single source IP address, which is shared. This is often referred to as *NAT with overload*; we are hiding multiple IPs on a single global address. When all those packets come back, they return to unique port numbers. Fortunately, the PAT device has been keeping track of which clients from which IP addresses initiated which sessions. So, referring to that table, the PAT device can rewrite the reply packet with all the correct information and return the packet to the client so that the client is not even aware that NAT/PAT even

happened. Example 12-2 shows a router's NAT table when performing PAT for both PC1 and PC2 as they connect out to Server A using Telnet.

Example 12-2 *PAT Table for Both PC1 and PC2, Using the Same Global Address*

```
R3# show ip nat translations
Pro Inside global      Inside local      Outside local      Outside global
! Note: source ports 56177 and 41300 are source port numbers that are
! dynamically chosen by PC1 and PC2 respectively.
! the destination port of 23 is the well-known port for telnet, which is
! what both PCs are connecting to the server with.

! Entry for the translation for PC1, to the global address of 34.0.0.3
! which is also the outside interface address of the router
tcp 34.0.0.3:56177      10.0.0.11:56177   45.1.2.3:23        45.1.2.3:23

! Entry for the translation for PC2, mapped to the global address of
! 34.0.0.3
! It is sharing this mapping with PC1. R3 is tracking all the addresses
! and ports involved, and knows which return packets from the server
! should be forwarded to which PC, and on which ports.
tcp 34.0.0.3:41300      10.0.0.22:41300   45.1.2.3:23        45.1.2.3:23
```

As you see in the example, PAT preserves the same source port that the PCs chose to initially use. If by chance the PCs both initiate sessions to the same server and use identical source ports, the router dynamically uses a different source port for one of the sessions. This is perfectly okay because the PAT router keeps track of all the sessions and all the ports, and as long as the return traffic can be correctly forwarded to the correct PC, it provides the functionality desired. Dynamic port numbers are unlikely to be completely used because the range begins at 1024 and goes all the way to 65,535. If for some reason PAT cannot find an available port, and if multiple IP global addresses are available for PAT, the router can decide to move to the next available global IP address and start performing PAT using that. For instance, the router could be configured for PAT and reference a pool of global addresses to use rather than a single IP for use with PAT.

NAT Options

Table 12-9 describes the several flavors and options related to NAT configuration and deployment.

**Table 12-9** NAT Deployment Options

Option	Description
Static NAT	This is a one-to-one permanent mapping. If you have 100 internal users and 100 global addresses, a one-to-one mapping can be done for every user, and every user would have a dedicated global address associated with his inside address. We do not usually have enough global addresses for each user. A typical use of a static mapping is this: We have a server on the inside of our network, or perhaps on a <i>demilitarized zone (DMZ)</i> interface off of our firewall, and we want to allow devices on the Internet access to that specific device. By creating a static mapping for that one server to a global IP address, that global IP address can be used in the <i>Domain Name System (DNS)</i> tables, and users on the Internet can reach our server by name (for example, <i>www.server.com</i>).
Dynamic NAT	Dynamic NAT involves having a pool of global addresses, and only mapping those global addresses to inside devices when those inside devices have and need to go out to the Internet. For example, a printer on the inside network may need to send packets out to the Internet, and as a result we will not use any global addresses (NAT or PAT) for that printer because the printer will never initiate traffic from the inside network going to the Internet. With dynamic NAT applied to inside hosts, after a period of time of a host not using a dynamically assigned IP address, the router or firewall performing the NAT reclaims that IP address and can use it for another device that might need it in the future.
Dynamic PAT (NAT with overload)	This is the feature that is used for most users who access the Internet. It combines the benefits of dynamically assigning global addresses only when needed, and it uses overload so that literally thousands of inside devices can be translated to the same global IP address, and as previously discussed, the router is tracking all ports and IP addresses in use to maintain the translation tables. The PAT global addresses include a single IP address that is already assigned to the outside interface of the router, a single global address dedicated for the use of PAT, or a pool of global addresses allocated for the use of PAT.
Policy NAT/PAT	Policy-based NAT is based on a set of rules, such as what is the source IP address, what is the destination IP address, and which ports are used that would qualify that packet to have NAT/PAT applied to it. Rules can be set up that only specific source IP addresses which are destined for specific destination addresses and or specific ports will be translated, and that is what the <i>policy</i> part means. Traffic outside the policy is simply forwarded based on normal routing forwarding without translation.

Creating and Deploying Firewalls

Various rules and options apply to firewalls, but this section covers the best practices for implementing a firewall.

Firewall Technologies

We know that a firewall has a goal of separating two entities, and specifically controlling access between those two, such as two networks. Most commercial firewalls today can do packet filtering, application layer inspection, stateful packet filtering, NAT (in all its flavors), AAA functions, and perform *virtual private network (VPN)* services. A good example of a device such as this is the Cisco *Adaptive Security Appliance (ASA)*, which is a dedicated firewall appliance. Many of these same features can all be implemented in software and on top of an IOS router whose license, memory, and CPU can support it. A dedicated firewall appliance is considered more secure and therefore preferable to simply using a standalone router. A defense-in-depth approach suggests that perhaps you can run these features on more than a single device for the added levels of protection.

Firewall Design Considerations

Here is a partial list of best practices for firewall deployment:

- Firewalls should be placed at security boundaries, such as between two networks that have different levels of trust (from the perspective of your organization). An example is your internal network compared to the Internet.
- Firewalls should be a primary security device, but not the only security device or security measure on the network.
- A policy that starts with a “deny all” attitude and then specifically only permits traffic that is required is a better security posture than a default “permit all” attitude first and then denying traffic specifically not wanted.
- Leverage the firewall feature that best suits the need. For example, if you know you have thousands of users who need access to the Internet, you can implement dynamic NAT/PAT for those users, along with stateful filtering and deny all inbound traffic coming from the Internet. This stops users on the Internet from initializing sessions to your users because of the deny on the outside interface. It allows users to access the Internet because you are performing NAT dynamically for them. Return traffic coming back from the Internet is allowed into the firewall because the stateful filtering is being done and the firewall can dynamically allow the return traffic. If you want to allow only specific users access to the Internet, you can additionally enable AAA.
- Make sure that physical security controls and management access to the firewall devices, and the infrastructure that supports them such as cables and switches, are secure.



- Have a regularly structured review process looking at the firewall logs. Many tools enable you to review syslog messages and look for anomalies and messages that might indicate a need for further investigation.
- Practice change management for any configuration modification on the firewalls. AAA and proper documentation is important to have a record of which administrator made which changes and when they were made. The accounting records (or least a copy of these accounting records regarding changes) should be forwarded to at least one server that is out of the administrative control of the admin group. This protects the company from administrators who might make malicious (or innocent) changes to the configuration and cause a network problem and then try to delete the accounting logs.

Firewall Access Rules

As mentioned before, the appropriate method for implementing firewall rules is based on a policy. The policy (on paper) drives what the firewall configuration should be. You can implement many different types of access rules on a typical firewall, some of which are described in Table 12-10.



Table 12-10 *Firewall Access Rules*

Rule	Description
Rules based on service control	These rules are based on the types of services that may be accessed through the firewall, inbound or outbound. An example is that access to web servers, both HTTP or HTTPS, is allowed while all other types of traffic are denied.
Rules based on address control	These rules are based on the source/destination addresses involved, usually with a permit or deny based on specific entries in an access control list.
Rules based on direction control	These rules specify where the initial traffic can flow. For example, a rule might say that traffic from the inside going to the outside (which we could also call outbound traffic) is permitted. Traffic initiated from the outside going to inside resources (which we could call inbound traffic) would be denied. Note that stateful filtering, with its stateful database, could dynamically allow the return traffic back to the inside users. These types of rules could very easily be combined (and usually are) with various protocols/services (such as HTTP, HTTPS, and so on).
Rules based on user control	These rules control access based on knowing who the user is and what that user is authorized to do. This can be implemented via AAA services.
Rules based on behavior control	These rules control how a particular service is used. For example, a firewall may implement an email filter to protect against spam.

Packet-Filtering Access Rule Structure

In the context of packet filtering, an ACL is applied to an interface either inbound or outbound on that interface. If applied inbound, all packets attempting to go through that interface must be permitted by the entries in the access list. Access lists are processed in a top-down fashion. As soon as the firewall identifies a match from a single entry in the ACL, it then implements the action of permit or deny (based on what that entry in the access list says to do) on the packet, and then the firewall moves on to the next packet and does the list again from top to bottom, or at least from the top until a match occurs. If there's no match in the access list, the packet-filtering function assumes the worst and denies the packet.

Firewall Rule Design Guidelines

Regardless of which type of rules you choose to implement, here are some guidelines for the creation of those rules:

- Use a restrictive approach as opposed to a permissive approach for all interfaces and all directions of traffic. By using this as a starting point, you can then permit only traffic that you specify while denying everything else. This might take a little while to fine-tune because many administrators often discover additional required protocols for the functionality of their networks that may not have initially been considered, such as routing protocols, network management protocols, and so on.
- Presume that your internal users' machines may be part of the security problem. If you blindly trust all devices on the inside to access resources through the firewall, this may also include an attacker who has physical access to the building or malicious code that is unknown to the user running on one of their PCs.
- Be as specific as possible in your **permit** statements, such as avoiding the use of the keyword **any** or **all** IP protocols if possible.
- Recognize the necessity of a balance between functionality and security. Customers have a network for a reason, and they need to allow traffic through the firewalls to meet their business needs. At some point, you might need to point out a potential security weakness based on allowing something through your firewall but allow the traffic anyway based on the business need. It is usually up to someone higher up in the political food chain to make those final decisions.
- Filter bogus traffic, and perform logging on that traffic. Some packets should never be allowed into your network. For example, if your network is the 23.1.2.0/24 network, there should never be a packet that is entering your network (from a remote network) which (based on its source address) claims it is also from the 23.1.2.0/24 network. Traffic from the RFC 1918 private address space is unlikely to be legitimate traffic if coming in from the Internet. Bogus traffic, such as the two examples just provided, should be filtered at the edges of the network. Even if you think your service provider will deny the traffic, you should implement the same filtering on your perimeter routers as well.



- Periodically review the policies that are implemented on the firewall to verify that they are current and correct. Obsolete rules that are no longer in use should be removed or least updated through documented change control.

Rule Implementation Consistency

For any changes that will be made to a firewall, there should be a change control procedure that identifies exactly what is going to be done, why it is going to be done, and the approval of the person in charge of making that authorization for the work to be done. The change control documentation should include what the possible impact might be to the network related to this change and what the restore procedure and timeframe would be in the event the changes need to be backed out.

Unfortunately, if there is not a consistent and well-considered process for implementing the rules, some negative results may follow regarding the rules that end up on the firewall that is supposed to be implementing the policy for the company. Table 12-11 describes some of these rules that may inadvertently show up.



Table 12-11 *Results of Inconsistent or Ill-Considered Rule Implementation*

Rule	Description
Rules that are too promiscuous	These types of rules allow more access than is necessary for the business requirement. Often, a rule may be implemented in an attempt to get a network application working, and the keyword of any is allowed for either the addresses or the IP keyword for the entire protocol stack. Unfortunately, if this rule is put in as a temporary test, and the application begins to work, it will be very difficult later in the production environment to narrow the scope of the access and still allow the application to function. Rules that are too promiscuous are significant holes in a security policy.
Redundant rules	ACLs are processed from top to bottom. If a rule is already in place as allowing a specific flow of traffic, a second rule for that does not need to be added to the control lists. Unfortunately, if an ACL is thousands of lines long, or is using object groups that are not understood by the administrator, additional unnecessary entries may be inadvertently added by the administrator. This does not necessarily cause an additional security risk, but it does create rules that are unnecessarily long (or at least longer).
Shadowed rules	A shadowed rule is basically incorrect order placement in the access list. For example, if you want to deny a specific source IP address from going to a specific web server, and you add the entry for that to an access list, one would think that that access is now filtered. However, access control entries are added by default to the bottom of an ACL. As a result, if a previous line specifically permits all web traffic to any web server, that entry permits this individual device to go to the specified web server before the new ACL entry is ever considered.

Rule	Description
Orphaned rules	This most likely results from a configuration error that is referencing incorrect IPs that would never be seen by the firewall. For example, if an access list intended to filter traffic from inside users includes a source IP address range that does not exist on the inside of the network, that access control list entry will never be matched. Orphaned rules are simply taking up space in the configuration and are never matched.
Incorrectly planned rules	This may result from an error that is made as the business requirements are being translated to the technical and logical controls that the firewall will implement. This may be due to a lack of understanding what protocols (and or ports) are really used by the devices in the network with the applications in use.
Incorrectly implemented rules	This results from an administrator implementing the incorrect port, protocol, or IP information on the firewall.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 12-12 lists these key topics.



Table 12-12 *Key Topics*

Key Topic Element	Description	Page Number
List	The objectives of a good firewall	270
Table 12-3	Potential firewall limitations	272
Text	Five basic firewall methodologies	273
Text	Stateful packet filtering	276
Table 12-8	NAT terminology	279
Table 12-9	NAT deployment options	282
List	Firewall design considerations	283
Table 12-10	Firewall access rules	284
List	Firewall rules design guidelines	285
Table 12-11	Results of inconsistent or ill-considered implementation	286

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

packet filtering, stateful filtering, transparent firewall, NAT, PAT

This page intentionally left blank



This chapter covers the following subjects:

- Cisco IOS Zone-Based Firewall
- Configuring and verifying Cisco IOS Zone-Based Firewall

Implementing Cisco IOS Zone-Based Firewalls

Cisco has implemented some form of firewall protection in their IOS for many years. You can implement packet filtering with access lists applied to interfaces. The older firewall feature set named *context-based access control (CBAC)* provided stateful filtering. More recently, that CBAC has been replaced by a newer method for stateful filtering and application inspection called the *Zone-Based Firewall (ZBF)*. This chapter is all about understanding and implementing the ZBF feature on an IOS-based router.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 13-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 13-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Cisco IOS Zone-Based Firewall	1–4
Configuring and Verifying Cisco IOS Zone-Based Firewall	5–8

1. Which zone is implied by default and does not need to be manually created?
 - a. Inside
 - b. Outside
 - c. DMZ
 - d. Self

- 2.** If interface number 1 is in zone A, and interface number 2 is in zone B, and there is no **policy** or **service** commands applied yet to the configuration, what is the status of transit traffic that is being routed between these two interfaces?

 - a.** Denied
 - b.** Permitted
 - c.** Inspected
 - d.** Logged
- 3.** When creating a specific zone pair and applying a policy to it, policy is being implemented on initial traffic in how many directions?

 - a.** 1
 - b.** 2
 - c.** 3
 - d.** Depends on the policy
- 4.** What is the default policy between an administratively created zone and the self zone?

 - a.** Deny
 - b.** Permit
 - c.** Inspect
 - d.** Log
- 5.** What is one of the added configuration elements that the Advanced security setting has in the ZBF Wizard that is not included in the Low security setting?

 - a.** Generic TCP inspection
 - b.** Generic UDP inspection
 - c.** Filtering of peer-to-peer networking applications
 - d.** NAT

6. Why is it that the return traffic, from previously inspected sessions, is allowed back to the user, in spite of not having a zone pair explicitly configured that matches on the return traffic?
 - a. Stateful entries (from the initial flow) are matched, which dynamically allows return traffic.
 - b. Return traffic is not allowed because it is a firewall.
 - c. Explicit ACL rules need to be placed on the return path to allow the return traffic.
 - d. A zone pair in the opposite direction of the initial zone pair (including an applied policy) must be applied for return traffic to be allowed.
7. What does the keyword *overload* imply in a NAT configuration?
 - a. NAT is willing to take up to 100 percent of available CPU.
 - b. PAT is being used.
 - c. NAT will provide “best effort” but not guaranteed service, due to an overload.
 - d. Static NAT is being used.
8. Which of the following commands shows the current NAT translations on the router?
 - a. `show translations`
 - b. `show nat translations`
 - c. `show ip nat translations`
 - d. `show ip nat translations *`

Foundation Topics

Cisco IOS Zone-Based Firewall

This section examines the logic and structural components that make up the IOS-based Zone-Based Firewall (ZBF).

How Zone-Based Firewall Operates



With ZBFs, interfaces are placed into zones. Zones are created by the network administrator, using any naming convention that makes sense (although names such as inside, outside, and DMZ are quite common). Then policies are specified as to what transit (user) traffic is allowed to be initiated, (for example, from users on the inside destined to resources on the outside) and what action the firewall should take, such as inspection (which means to do stateful inspection of the traffic). After traffic is inspected, the reply traffic is allowed back through the firewall because of the stateful filtering feature. The policies are implemented in a single direction (for example, inside to outside). If you want to allow initial traffic in both directions, you create two unidirectional policies for traffic to be allowed and inspected from the inside to the outside, and also from the outside to the inside. You implement two separate policies because the policies themselves are unidirectional.

One benefit of this modular approach is that after policies are in place, if you add additional interfaces, all you need to do is add those interfaces to existing zones, and your policies will automatically be in place.

Specific Features of Zone-Based Firewalls



The ZBF major features include the following:

- Stateful inspection
- Application inspection
- Packet filtering
- URL filtering
- Transparent firewall (implementation method)
- Support for *virtual routing and forwarding (VRF)*
- *Access control lists (ACL)* are not required as a filtering method to implement the policy

Many of these features we have addressed in a previous chapter, and some of the concepts are new. Let's consider a few not yet discussed. URL filtering refers to the ability to control what traffic is permitted or denied (mostly denied) based on the URL that is

trying to be accessed by the client. VRFs are virtual routing tables on a Cisco router than can be used to compartmentalize the routing tables on the router instead of keeping all the routes in the global (primary) routing tables. A transparent firewall is implemented at Layer 2 but can still perform analysis of traffic at Layer 3 and higher. You learn more about these details as you progress through this chapter.

Zones and Why We Need Pairs of Them

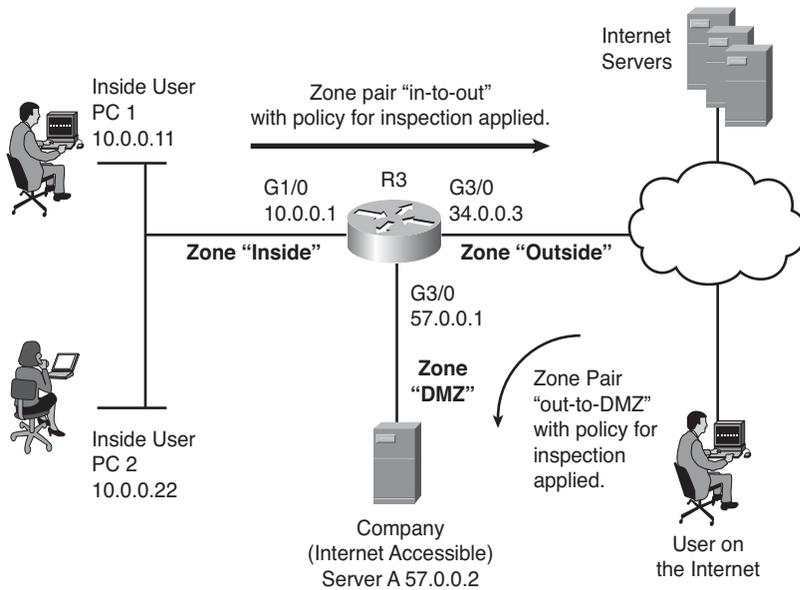
A zone is first created by the administrator, and then interfaces can be assigned to zones. A zone can have one or more interfaces assigned to it. Any given interface can belong to only a single zone. There is a default zone, called the *self zone*, which is a logical zone. For any packets directed to the router directly (the destination IP represents the packet is for the router), the router automatically considers that traffic to be entering the self zone. In addition, any traffic initiated by the router is considered as leaving the self zone. By default, any traffic to or from the self zone is allowed, but you can change this policy.

For the rest of the administrator-created zones, no traffic is allowed between interfaces in different zones. For interfaces that are members of the same zone, all traffic is permitted by default. So, here is the catch. If you want to allow traffic between two zones, such as between the inside zone (using interfaces facing the inside network) and the outside zone (interfaces facing the internet or less trusted networks), you must create a policy for traffic between the two zones, and that is where a zone pair comes into play. A zone pair, which is just a configuration on the router, is created identifying traffic sourced from a device in one zone and destined for a device in the second zone. The administrator then associates a set of rules (the policy) for this unidirectional zone pair, such as to inspect the traffic, and then applies that policy to the zone pair.

A small company, with users on the inside network, with the only other connection being the Internet, may want to create two zones, one for the inside and one for the outside. Then they would assign the inside interface to the inside zone, and the outside interface to the outside zone. Then, a policy could be created that specifies that traffic that is initiated from the inside users and going out to the Internet should be inspected and that information should be placed in the stateful database. A zone pair identifying traffic from the inside to the outside would have the policy applied to it, letting it know that the stateful inspection should be done.

A larger company that has a public facing server may have three interfaces and three zones. The zones may be inside, outside, and DMZ. Compared to the small company, this medium-sized company creates an additional zone pair (from outside to DMZ), and then applies a policy to that zone pair to allow outside users to access the servers on the DMZ.

Figure 13-1 shows an example of a medium-sized company with a DMZ.



Any time traffic is inspected, it is allowed between the zones and the session information is placed into the stateful database on the zone-based firewall. This allows the return traffic to be allowed, even without a zone pair and policy in the direction of the return traffic.

Figure 13-1 *Topology for This ZBF Discussion*

Putting the Pieces Together



Cisco uses a language called the *Cisco Common Classification Policy Language (C3PL)* for the implementation of the policy. This process has three primary components:

- **Class maps:** These are used to identify traffic, such as traffic that should be inspected. Traffic can be matched based on Layer 3 through Layer 7 of the OSI model, including application-based matching. Class maps can also refer to *access control lists (ACL)* for the purpose of identifying traffic or even call upon other class maps. Class maps can have multiple **match** statements. A class map can specify that all **match** statements have to match (which is a *match-all condition*) or can specify that matching any of the entries is considered a match (which is a *match-any condition*). A system defined class map named *class-default* can be used that represents all traffic not matched in a more specific (administratively configured) class map.
- **Policy maps:** These are the actions that should be taken on the traffic. Policy maps call on the class maps for the classification of traffic. Policy maps with multiple sections are processed in order. The primary actions that can be implemented by the policy map are *inspect* (which means that stateful inspection should happen), *permit* (which means that traffic is permitted but not inspected), *drop*, or *log*.

- **Service policies:** This is where you apply the policies, identified from a policy map, to a zone pair. This step actually implements the policy.

If a policy map contains multiple actions, based on different class map-identified traffic, the policy map is processed from top to bottom, applying the actions as traffic matches the class maps. If a specific section of a policy map matches, the action is taken. If traffic does not match, the packet is compared against the next section of the policy map. If none of the sections match the traffic, the default behavior action is taken. The default policy for traffic that is trying to be initiated between two zones (starting in one zone and going to a device in another zone) is an implicit deny. (The exception to this default deny is traffic to or from the built-in “self” zone, which is allowed by default.)

Table 13-2 describes the actions that you may specify in a policy map.

Table 13-2 *Policy Map Actions*

Policy Action	Description	When to Use It
Inspect	Permit and statefully inspect the traffic	This should be used on transit traffic initiated by users who expect to get replies from devices on the other side of the firewall.
Pass	Permits/allows the traffic but does not create an entry in the stateful database	Traffic that does not need a reply. Also in the case of protocols that do not support inspection, this policy could be applied to the zone pair for specific outbound traffic, and be applied to a second zone pair for inbound traffic.
Drop	Deny the packet	Traffic you do not want to allow between the zones where this policy map is applied.
Log	Log the packets	If you want to see log information about packets that were dropped because of policy, you can add this option.



Service Policies

A service policy is applied to a zone pair. The zone pair represents a unidirectional flow of traffic between two zones. A specific zone pair can have only a single service policy assigned to it. Because the zone pair is unidirectional, the policy map applied to the zone pair (using the **service-policy** command) applies to traffic initiated in one zone going to the other zone in one direction. If reply traffic is desired, the **inspect** action in the policy map should be applied, which will allow stateful inspection, and the reply traffic from the servers will be dynamically allowed (because of the stateful database being referenced).

When a router receives a packet, it normally makes a routing decision, and then forwards that packet on its way. If ZBF is configured, the router may or may not forward the packet, based on the stateful table and the policies that are in place. Table 13-3 describes

the flow of traffic (packets) being routed between interfaces in various zones, depending on the configuration. This is a good table to commit to memory; it will assist anyone troubleshooting ZBFs. *Ingress* refers to a packet going into an interface of the router, and *egress* refers to a packet that is being sent out of an interface of the router.



Table 13-3 *Traffic Interaction Between Zones*

Ingress Interface Member of Zone	Egress Interface Member of Zone	Zone Pair Exists, with Applied Policy	Result
No	No	Does not matter	Traffic is forwarded.
No	Yes (any zone)	Does not matter	Traffic is dropped.
Yes (zone A)	Yes (zone A)	Does not matter	Traffic is forwarded.
Yes (zone A)	Yes (zone B)	No	Traffic is dropped.
Yes (zone A)	Yes (zone B)	Yes	Policy is applied. If policy is inspect or pass, the initial traffic is forwarded. If the policy is drop, the initial traffic is dropped.

If there is a zone pair that identifies traffic between two zones, and the policy is not applied to the zone pair, the default behavior is to drop traffic as if no zone pair even existed.

Before we go any further, I want to show you a configuration that includes the following ZBF components:

- Zones
- Interfaces that are members of zones
- Class maps that identify traffic
- Policy maps that use class maps to identify traffic and then specify the actions which should take place
- Zone pairs, which identify a unidirectional traffic flow, beginning from devices in one zone and being routed out an interface in a second zone.
- Service policy, which associates a policy map with a zone pair

Now that you know all the pieces, it is time to take a look at the commands for the policy of allowing users on the inside to access the Internet (as shown earlier in Figure 13-1). Example 13-1 both shows and explains this.

Example 13-1 *Components That Make Up The ZBF*

```

! The class map "classifies" or "identifies" the traffic
! In this example, this class map will match on either TELNET traffic or
! any type of ICMP traffic
R3(config)# class-map type inspect match-any MY-CLASS-MAP
R3(config-cmap)# match protocol telnet
R3(config-cmap)# match protocol icmp
R3(config-cmap)# exit

! The policy map calls on a specific class map that it wants to use
! to identify which traffic the policy applies to, and then specifies the
! policy action. In this example, it is to inspect the traffic
R3(config)# policy-map type inspect MY-POLICY-MAP
R3(config-pmap)# class type inspect MY-CLASS-MAP
R3(config-pmap-c)# inspect
R3(config-pmap-c)# exit
R3(config-pmap)# exit

! Next we create the security zones, they can be named whatever you want to
! name them. In this example, I named them inside and outside.
R3(config)# zone security inside
R3(config-sec-zone)# exit
R3(config)# zone security outside
R3(config-sec-zone)# exit

! Create the zone-pair, specifying the zones and the direction (from where
! to where)
R3(config-sec-zone)# zone-pair security in-to-out source inside destination
outside

! Use the service-policy command in zone-pair configuration mode to apply
! the policy map you want to use for traffic that matches this zone-pair
R3(config-sec-zone-pair)# service-policy type inspect MY-POLICY-MAP
R3(config-sec-zone-pair)# exit

! Configure the interfaces, so they become members of the respective zones
R3(config)# interface GigabitEthernet3/0
R3(config-if)# description Belongs to outside zone
R3(config-if)# zone-member security outside
R3(config-if)# exit
R3(config)# interface GigabitEthernet1/0
R3(config-if)# description Belongs to inside zone
R3(config-if)# zone-member security inside
R3(config-if)# exit
R3(config)#

```

The preceding policy performs stateful inspection for traffic from the inside users for traffic going to the Internet if that traffic is Telnet traffic (which is TCP port 23) or is *Internet Control Message Protocol (ICMP)* traffic. ACLs can be used by the class map for matching and generic protocol matches such as UDP or TCP. Application-specific matching adds the ability for the firewall to detect additional communication channels that may be initialized by the outside devices, such as in the case of inspecting FTP, where the server may initiate the data connection on a port mutually agreed to by the client and the FTP server.

The Self Zone

Traffic directed to the router itself (as opposed to traffic going through the router as transit traffic that is not destined directly to the router) involves the self zone. Traffic destined to the router, regardless of which interface is used, is considered to be going to the self zone. Traffic being sourced from the router is considered to be coming from the self zone. By default, all traffic to the self zone or from the self zone (which really means all traffic from the router or to the router) is allowed. However, if you want to create policies related to traffic to or from this self zone, you do it the same way by creating zone pairs and assigning a policy to the zone pair. Table 13-4 describes self zone traffic behavior.



Table 13-4 *Self Zone Traffic Behavior*

Source Traffic Member of Zone	Destination Traffic Member of Zone	Zone Pair Exists, with a Policy Applied	Result
Self	Zone A	No	Traffic is passed.
Zone A	Self	No	Traffic is passed.
Self	Zone A	Yes	Policy is applied.
Zone A	Self	Yes	Policy is applied.

Regarding the self zone, if there is a zone pair but no policy is applied, the default behavior is to forward all traffic (which is different from the traffic between manually created zones).

Configuring and Verifying Cisco IOS Zone-Based Firewall

This section examines configuring the IOS ZBF feature from both *Cisco Configuration Professional (CCP)* and the *command-line interface (CLI)*.

First Things First

As discussed earlier in this book, having a policy or a plan in place first is a good idea before you configure anything. Remember that the policy you implement on the firewall is supposed to reflect the business needs of the company. The previous section examined a basic implementation of ZBFs, and that example (Example 13-1) is a fantastic reference.

You can use CCP for an easy walkthrough of the configuration of ZBF. We use the wizard for the configuration, and then we use the CLI for the verification. One thing you will notice is that the wizard adds a lot of additional configuration that may not apply directly to your network. One recommendation, if you are going to use CCP, is to use the wizards but take the recommended commands that CCP is about to apply and put those into a text editor. Edit out the pieces that are not needed or wanted, and then manually paste the configuration in after you have verified that it is perfect. For certification, it is recommended that you become familiar with interpreting a ZBF configuration, using CCP, to be able to determine what actions will be taken on a packet going through the firewall.

We use the same topology as shown earlier in the chapter (refer to Figure 13-1).

Using CCP to Configure the Firewall

In CCP, you select the router that you want to configure from the drop-down list and navigate to **Configure > Security > Firewall > Firewall**, as shown in Figure 13-2.

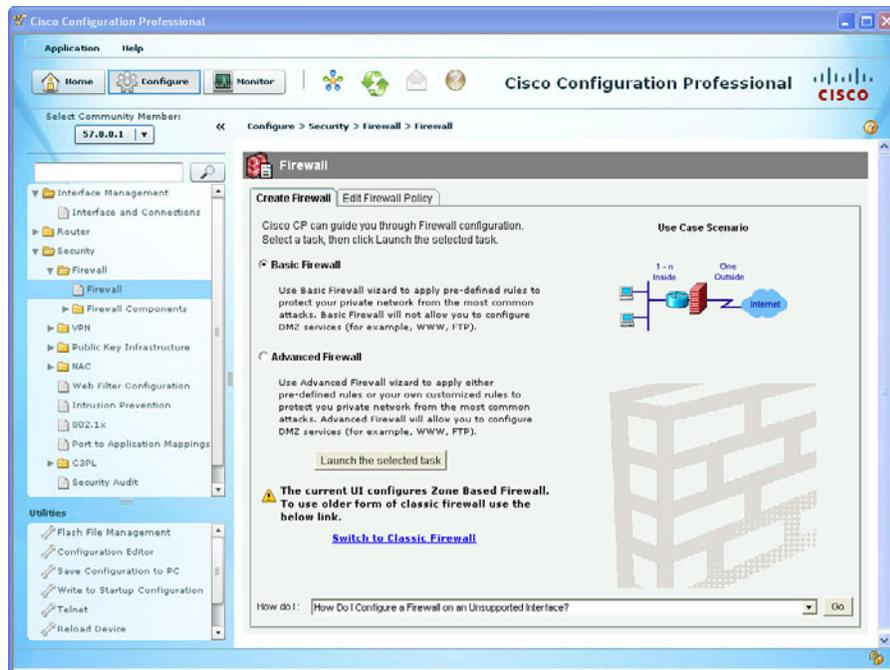


Figure 13-2 Firewall Wizard Page in CCP

The Basic Firewall option involves two interfaces, which are in different zones. The Advanced Firewall option enables you to apply predefined rules and allow you to configure a third zone such as a DMZ. The basic concept of a ZBF is that we're dealing with only two interfaces at a time. Each pair of interfaces has a zone pair for each direction for which you want to apply policy. Working with multiple interfaces and multiple zones is simply repeating the process focusing on one zone pair at a time, applying the policies you want to each zone pair.

There is an option to configure the older method of IOS-based firewall, which is the CBAC. You can select it by clicking the **Switch to Classic Firewall** link on this page. We are not going backward today, and so we click the **Launch of the Selected Task** button; and because the **Basic Firewall** radio button was selected, the wizard presents the welcome screen shown in Figure 13-3.

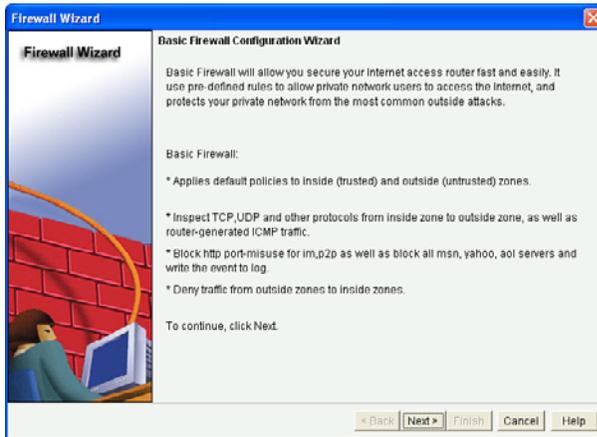


Figure 13-3 Welcome Screen to Basic Firewall Wizard

From here, click the **Next** button, at which point you are asked which interfaces are facing the trusted network and which interfaces are facing the untrusted network, as shown in Figure 13-4.

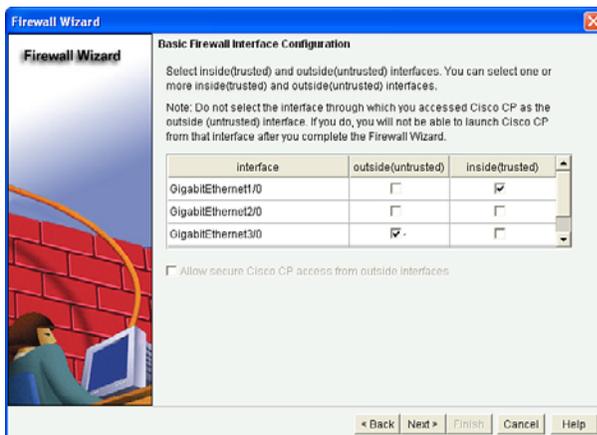


Figure 13-4 Specifying Which Interfaces Connect to Trusted or Untrusted Networks

Note that if an interface is not identified as being part of a zone, no traffic is allowed between the unspecified interface and interfaces in any other zones. In the preceding example, interface Gigabit Ethernet 2/0 cannot forward traffic to (route between) either of the other two interfaces. If that is your intention, you are okay. This configuration is such in this example for demonstration purposes only. After you click **Next**, you are given a notification that this interface is not a member of the zone, as shown in Figure 13-5.

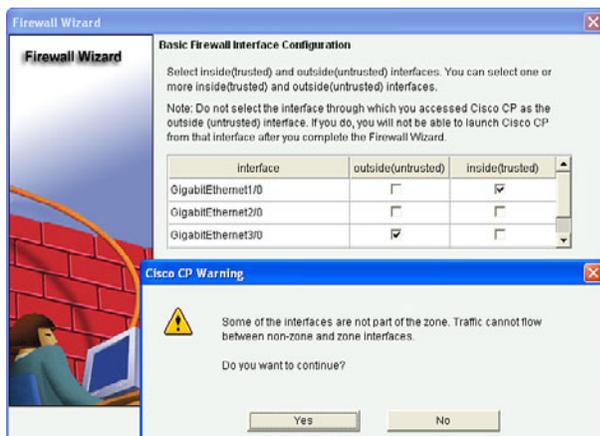


Figure 13-5 Warning Message About an Interface Not Belonging to a Zone

After you click **Yes** to continue, if your router supports *Call Manager Express (CME)*, even if it is not configured, you may be presented a warning message, as shown in Figure 13-6.

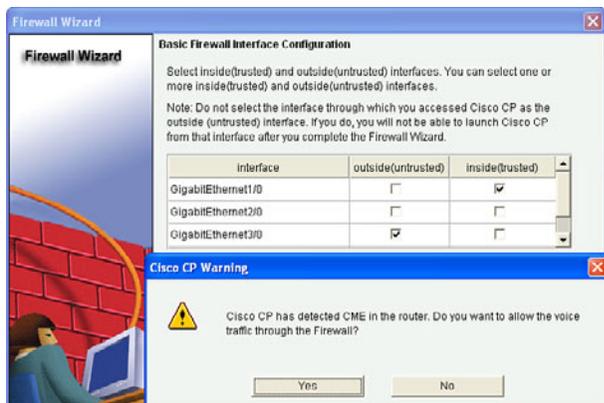


Figure 13-6 Warning Message About CME

If this CME is not applicable to your network, you can click the **No** button. If it is applicable, you can click **Yes**, and CCP will configure the additional policies to allow this traffic through the firewall. After answering this dialog box, you are next presented with a warning message letting you know that CCP cannot access this router through interfaces connected to the untrusted networks. In our case, that is G3/0, as indicated in the warning in Figure 13-7.

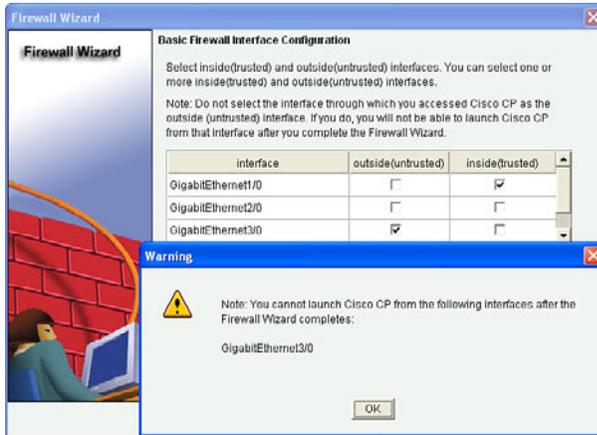


Figure 13-7 Warning Message That Untrusted Interfaces Will Not Support CCP Access

After you click the OK button, the wizard asks you which level of security you want to implement, as shown in Figure 13-8.

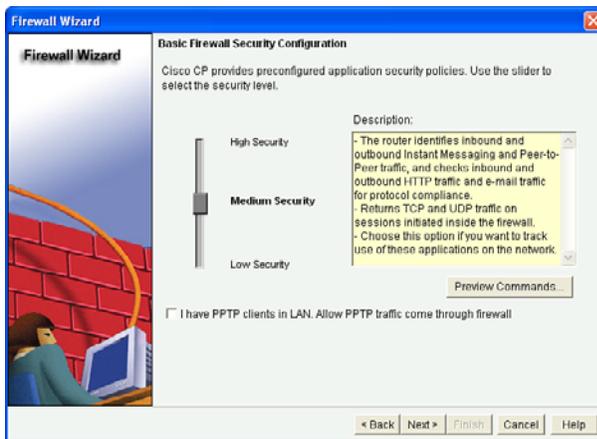


Figure 13-8 Choosing the Security Level to Implement

When configuring the ZBF Wizard, you can choose from three security levels:



- **High Security:** With this setting, the firewall identifies and drops instant messaging and peer-to-peer traffic. It does application inspection for web and email traffic and drops noncompliant traffic. It does generic inspection of TCP and UDP applications.
- **Medium Security:** This is similar to the High Security option, but it does not check web and email traffic for protocol compliance.
- **Low Security:** The router does not perform any application layer inspection. It does do generic TCP and UDP inspection.

It is important to note that the details of exactly what is implemented from each of these options are totally under the control of CCP. It is quite possible that in CCP any future updates may modify the details of all or any of these settings from the wizard. As the administrator of your network, it is up to you to confirm exactly what configurations will be applied to the router.

For this example, we chose the **Medium Security** option from the wizard. After you click **Next**, if a *Domain Name System (DNS)* server is not configured, the wizard asks you to provide the IP address of a DNS server for the router to use, as shown in Figure 13-9.

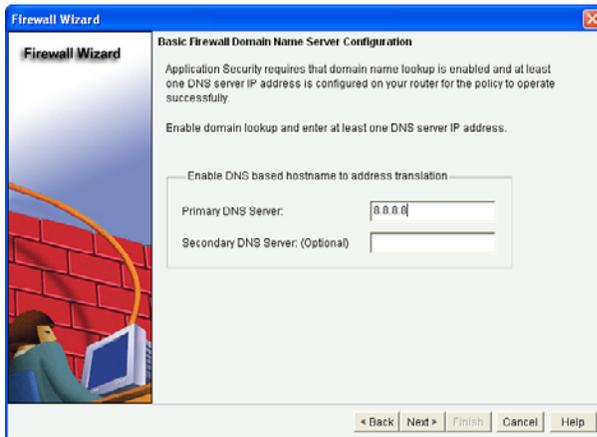


Figure 13-9 *Selecting a DNS Server*

After you click **Next**, a summary of the features to be implemented are shown, as displayed in Figure 13-10.

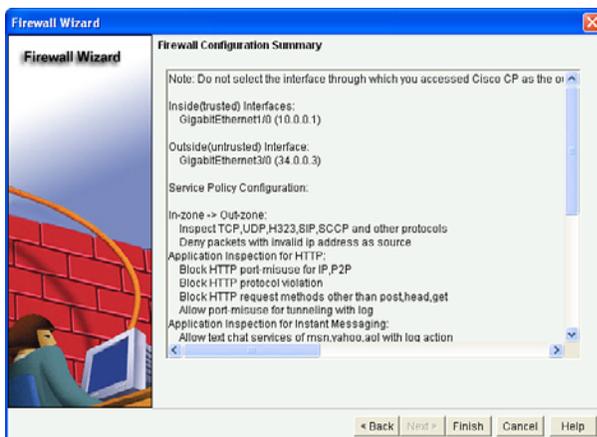


Figure 13-10 *Summary Page from ZBF Wizard*

After you click **Finish**, the wizard may provide one or more warnings about traffic that may be filtered by the policy and then delivers the configuration to the router.

Now for the interesting news. CCP created several pages of commands to implement the policy. Example 13-2 shows the literal commands that CCP created and applied.

Example 13-2 *Literal CLI Commands That the CCP Basic Zone-Based Firewall Creates*

```
! although this is an incredible amount of output, the basic concepts are
! the same as shown in more basic example 13-1.

access-list 100 remark CCP_ACL Category=128
access-list 100 permit ip host 255.255.255.255 any
access-list 100 permit ip 127.0.0.0 0.255.255.255 any
access-list 100 permit ip 34.0.0.0 0.0.0.255 any
ip domain lookup
ip name-server 8.8.8.8
parameter-map type protocol-info yahoo-servers
  server name scs.msg.yahoo.com
  server name scsa.msg.yahoo.com
  server name scsb.msg.yahoo.com
  server name scsc.msg.yahoo.com
  server name scsd.msg.yahoo.com
  server name cs16.msg.dcn.yahoo.com
  server name cs19.msg.dcn.yahoo.com
  server name cs42.msg.dcn.yahoo.com
  server name cs53.msg.dcn.yahoo.com
  server name cs54.msg.dcn.yahoo.com
  server name ads1.vip.scd.yahoo.com
  server name radio1.launch.vip.dal.yahoo.com
  server name in1.msg.vip.re2.yahoo.com
  server name data1.my.vip.sc5.yahoo.com
  server name address1.pim.vip.mud.yahoo.com
  server name edit.messenger.yahoo.com
  server name messenger.yahoo.com
  server name http.pager.yahoo.com
  server name privacy.yahoo.com
  server name csa.yahoo.com
  server name csb.yahoo.com
  server name csc.yahoo.com
  exit
parameter-map type protocol-info aol-servers
  server name login.oscar.aol.com
  server name toc.oscar.aol.com
  server name oam-d09a.blue.aol.com
  exit
parameter-map type protocol-info msn-servers
  server name messenger.hotmail.com
```

```
server name gateway.messenger.hotmail.com
server name webmessenger.msn.com
exit
class-map type inspect match-any ccp-cls-protocol-im
match protocol ymsgr yahoo-servers
match protocol msnmsgr msn-servers
match protocol aol aol-servers
exit
class-map type inspect edonkey match-any ccp-app-edonkeydownload
match file-transfer
exit
class-map type inspect match-any ccp-h323annexe-inspect
match protocol h323-annexe
exit
class-map type inspect http match-any ccp-http-blockparam
match request port-misuse im
match request port-misuse p2p
match req-resp protocol-violation
exit
class-map type inspect aol match-any ccp-app-aol-otherservices
match service any
exit
class-map type inspect match-all ccp-protocol-pop3
match protocol pop3
exit
class-map type inspect msnmsgr match-any ccp-app-msn
match service text-chat
exit
class-map type inspect match-any ccp-cls-icmp-access
match protocol icmp
match protocol tcp
match protocol udp
exit
class-map type inspect match-all ccp-icmp-access
match class-map ccp-cls-icmp-access
exit
class-map type inspect match-all ccp-protocol-imap
match protocol imap
exit
class-map type inspect match-any ccp-cls-insp-traffic
match protocol cuseeme
match protocol dns
match protocol ftp
match protocol https
match protocol icmp
match protocol imap
match protocol pop3
match protocol netshow
```

```
match protocol shell
match protocol realmedia
match protocol rtsp
match protocol smtp extended
match protocol sql-net
match protocol streamworks
match protocol tftp
match protocol vdolive
match protocol tcp
match protocol udp
exit
class-map type inspect aol match-any ccp-app-aol
match service text-chat
exit
class-map type inspect edonkey match-any ccp-app-edonkey
match file-transfer
match text-chat
match search-file-name
exit
class-map type inspect kazaa2 match-any ccp-app-kazaa2
match file-transfer
exit
class-map type inspect fasttrack match-any ccp-app-fasttrack
match file-transfer
exit
class-map type inspect match-any ccp-h323-inspect
match protocol h323
exit
class-map type inspect match-any ccp-h323nxg-inspect
match protocol h323-nxg
exit
class-map type inspect match-all ccp-insp-traffic
match class-map ccp-cls-insp-traffic
exit
class-map type inspect edonkey match-any ccp-app-edonkeychat
match search-file-name
match text-chat
exit
class-map type inspect match-any ccp-h225ras-inspect
match protocol h225ras
exit
class-map type inspect msnmsgr match-any ccp-app-msn-otherservices
match service any
exit
class-map type inspect ymsgr match-any ccp-app-yahoo-otherservices
match service any
exit
class-map type inspect http match-any ccp-app-httpmethods
```

```
match request method bcopy
match request method bdelete
match request method bmove
match request method bpropfind
match request method bproppatch
match request method connect
match request method copy
match request method delete
match request method edit
match request method getattribute
match request method getattributenames
match request method getproperties
match request method index
match request method lock
match request method mkcol
match request method mkdir
match request method move
match request method notify
match request method options
match request method poll
match request method propfind
match request method proppatch
match request method put
match request method revadd
match request method revlabel
match request method revlog
match request method revnum
match request method save
match request method search
match request method setattribute
match request method startrev
match request method stoprev
match request method subscribe
match request method trace
match request method unedit
match request method unlock
match request method unsubscribe
exit
class-map type inspect match-any ccp-skinny-inspect
match protocol skinny
exit
class-map type inspect match-all ccp-protocol-im
match class-map ccp-cls-protocol-im
exit
class-map type inspect match-any ccp-cls-protocol-p2p
match protocol edonkey signature
match protocol gnutella signature
match protocol kazaa2 signature
```

```
match protocol fasttrack signature
match protocol bittorrent signature
exit
class-map type inspect http match-any ccp-http-allowparam
match request port-misuse tunneling
exit
class-map type inspect gnutella match-any ccp-app-gnutella
match file-transfer
exit
class-map type inspect match-any ccp-sip-inspect
match protocol sip
exit
class-map type inspect match-all ccp-invalid-src
match access-group 100
exit
class-map type inspect ymsgr match-any ccp-app-yahoo
match service text-chat
exit
class-map type inspect pop3 match-any ccp-app-pop3
match invalid-command
exit
class-map type inspect imap match-any ccp-app-imap
match invalid-command
exit
class-map type inspect match-all ccp-protocol-p2p
match class-map ccp-cls-protocol-p2p
exit
class-map type inspect match-all ccp-protocol-http
match protocol http
exit
policy-map type inspect http ccp-action-app-http
class type inspect http ccp-http-blockparam
log
reset
exit
class type inspect http ccp-app-httpmethods
log
reset
exit
class type inspect http ccp-http-allowparam
log
allow
exit
exit
policy-map type inspect imap ccp-action-imap
class type inspect imap ccp-app-imap
log
exit
```

```
exit
policy-map type inspect pop3 ccp-action-pop3
  class type inspect pop3 ccp-app-pop3
    log
    exit
  exit
policy-map type inspect p2p ccp-action-app-p2p
  class type inspect edonkey ccp-app-edonkeychat
    log
    allow
    exit
  class type inspect edonkey ccp-app-edonkeydownload
    log
    allow
    exit
  class type inspect fasttrack ccp-app-fasttrack
    log
    allow
    exit
  class type inspect gnutella ccp-app-gnutella
    log
    allow
    exit
  class type inspect kazaa2 ccp-app-kazaa2
    log
    allow
    exit
  exit
policy-map type inspect im ccp-action-app-im
  class type inspect aol ccp-app-aol
    log
    allow
    exit
  class type inspect msnmsgr ccp-app-msn
    log
    allow
    exit
  class type inspect ymsgr ccp-app-yahoo
    log
    allow
    exit
  class type inspect aol ccp-app-aol-otherservices
    log
    reset
    exit
  class type inspect msnmsgr ccp-app-msn-otherservices
    log
    reset
```

```
    exit
class type inspect ymgr ccp-app-yahoo-otherservices
  log
  reset
  exit
exit
policy-map type inspect ccp-inspect
class type inspect ccp-invalid-src
  drop log
  exit
class type inspect ccp-protocol-http
  no drop
  inspect
  service-policy http ccp-action-app-http
  exit
class type inspect ccp-protocol-imap
  no drop
  inspect
  service-policy imap ccp-action-imap
  exit
class type inspect ccp-protocol-pop3
  no drop
  inspect
  service-policy pop3 ccp-action-pop3
  exit
class type inspect ccp-protocol-p2p
  no drop
  inspect
  service-policy p2p ccp-action-app-p2p
  exit
class type inspect ccp-protocol-im
  no drop
  inspect
  service-policy im ccp-action-app-im
  exit
class type inspect ccp-insp-traffic
  no drop
  inspect
  exit
class type inspect ccp-sip-inspect
  no drop
  inspect
  exit
class type inspect ccp-h323-inspect
  no drop
  inspect
  exit
class type inspect ccp-h323annexe-inspect
```

```
no drop
inspect
exit
class type inspect ccp-h225ras-inspect
no drop
inspect
exit
class type inspect ccp-h323nxg-inspect
no drop
inspect
exit
class type inspect ccp-skinny-inspect
no drop
inspect
exit
exit
policy-map type inspect ccp-permit
class class-default
exit
policy-map type inspect ccp-permit-icmpreply
class type inspect ccp-icmp-access
no drop
inspect
exit
class class-default
no drop
pass
exit
exit
zone security in-zone
zone security out-zone
zone-pair security ccp-zp-out-self source out-zone destination self
service-policy type inspect ccp-permit
exit
zone-pair security ccp-zp-in-out source in-zone destination out-zone
service-policy type inspect ccp-inspect
exit
zone-pair security ccp-zp-self-out source self destination out-zone
service-policy type inspect ccp-permit-icmpreply
exit
interface GigabitEthernet3/0
description $FW_OUTSIDE$
zone-member security out-zone
exit
interface GigabitEthernet1/0
description $FW_INSIDE$
zone-member security in-zone
exit
```

Verifying the Firewall

You can verify the firewall from both the CCP and the command line. In the video bonus material included with this book, there is a video on using CCP to interpret the configuration of a ZBF. This video reviews the concepts and walks through the details of how and where to find all the components. You should review the video and practice with CCP yourself to confirm that you can interpret a ZBF policy by navigating through and looking at the GUI of CCP.

To edit or verify the policy within CCP, navigate to **Configure > Security > Firewall > Firewall** and click **Edit**, as shown in Figure 13-11.

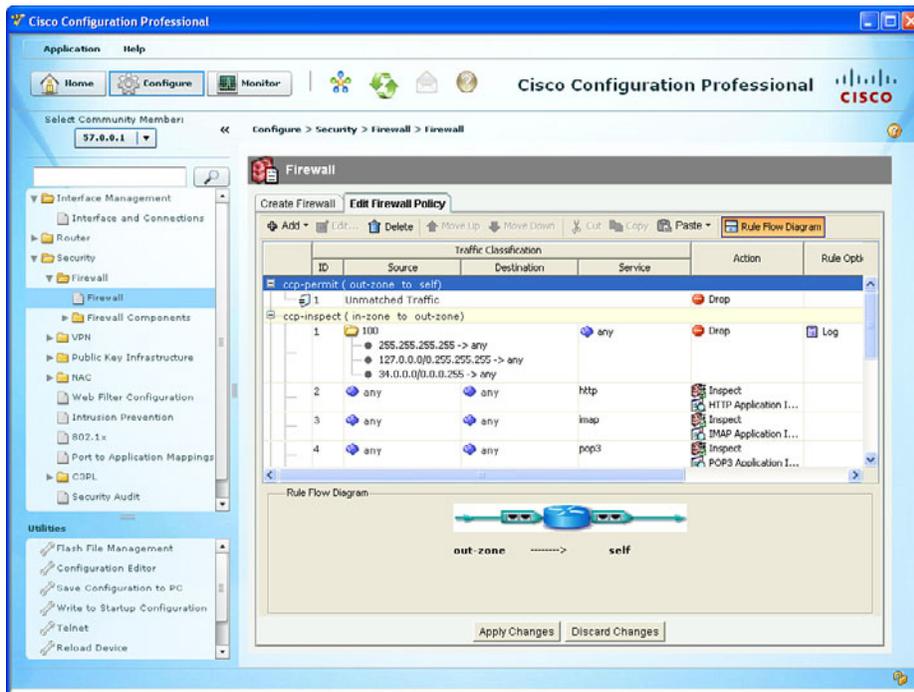


Figure 13-11 Verifying the Firewall Configuration from Within CCP

If logging is enabled and you are logging to a syslog server, you could verify any logged events by looking at those logs at the server. You learned how to enable logging in general through CCP in Chapter 5, “Using Cisco Configuration Professional to Protect the Network Infrastructure.” You can also view log messages on the router under the Log Monitoring section by navigating to **Monitor > Router > Logging**.

To see the firewall status and current activity, you can use the Monitor feature within CCP. To access this, navigate to **Monitor > Security > Firewall Status** and select the options you want to monitor. Figure 13-12 shows an example of this feature.

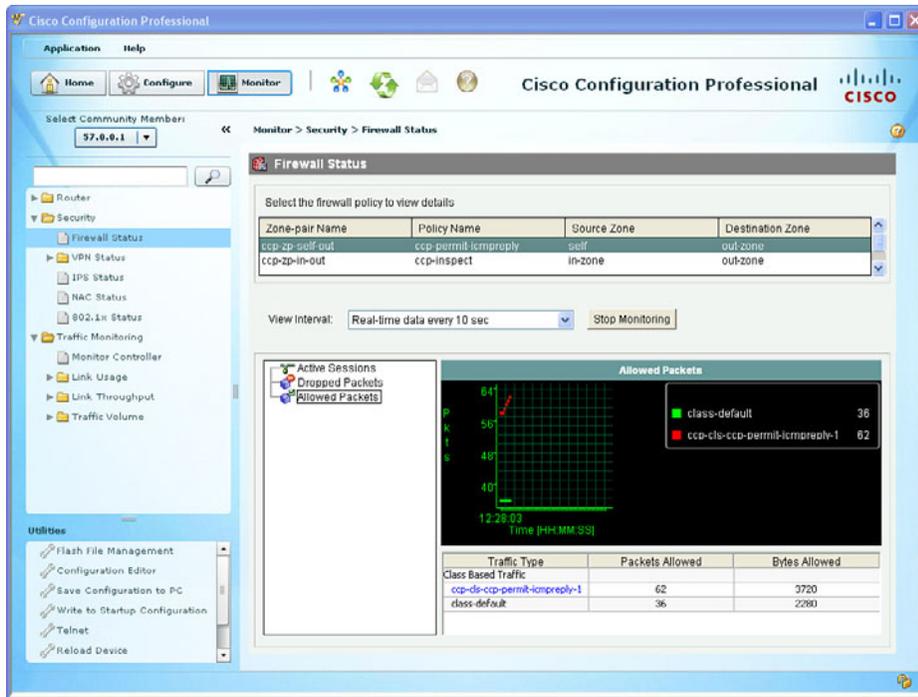


Figure 13-12 Monitoring the Firewall Through CCP

A button enables you to stop and start the monitoring (located on the page shown in Figure 13-12). The output shown in the figure about allowed packets and a custom class map is just an example from a customized firewall policy and may not reflect the defaults that are implemented by the ZBF Wizard. It is shown here for the purpose of demonstrating where to go in the GUI to monitor traffic going through the firewall.

Verifying the Configuration from the Command Line

For certification purposes, the focus of ZBFs is from the perspective of CCP. So, in preparation for the exam, be sure to practice with the GUI. Having the knowledge of the CLI is more relevant to a live environment, and for that purpose, it is also important to know. From the command line, you can use the commands as shown in the Example 13-3 to verify your ZBF components.

Example 13-3 Verifying the Configuration from the Command Line

```
R3# show class-map type inspect
Class Map type inspect match-any ccp-cls-protocol-p2p (id 27)
  Match protocol edonkey signature
  Match protocol gnutella signature
  Match protocol kazaa2 signature
  Match protocol fasttrack signature
  Match protocol bittorrent signature
```

```
Class Map type inspect match-any ccp-skinny-inspect (id 25)
  Match protocol skinny

Class Map type inspect match-all ccp-insp-traffic (id 19)
  Match class-map ccp-cls-insp-traffic

Class Map type inspect match-any ccp-h323nxg-inspect (id 18)
  Match protocol h323-nxg

Class Map type inspect match-any ccp-cls-icmp-access (id 8)
  Match protocol icmp
  Match protocol tcp
  Match protocol udp

Class Map type inspect match-any ccp-cls-protocol-im (id 1)
  Match protocol ymsgr yahoo-servers
  Match protocol msnmsgr msn-servers
  Match protocol aol aol-servers

Class Map type inspect match-all ccp-protocol-pop3 (id 6)
  Match protocol pop3

Class Map type inspect match-any ccp-h225ras-inspect (id 21)
  Match protocol h225ras

Class Map type inspect match-any ccp-h323annexe-inspect (id 3)
  Match protocol h323-annexe

Class Map type inspect match-any ccp-cls-insp-traffic (id 12)
  Match protocol cuseeme
  Match protocol dns
  Match protocol ftp
  Match protocol https
  Match protocol imap
  Match protocol pop3
  Match protocol netshow
  Match protocol shell
  Match protocol realmedia
  Match protocol rtsp
  Match protocol smtp extended
  Match protocol sql-net
  Match protocol streamworks
  Match protocol tftp
  Match protocol vdolive
  Match protocol tcp
  Match protocol udp
```

```
Class Map type inspect match-all ccp-protocol-p2p (id 35)
  Match class-map ccp-cls-protocol-p2p

Class Map type inspect match-any ccp-h323-inspect (id 17)
  Match protocol h323

Class Map type inspect match-all ccp-protocol-im (id 26)
  Match class-map ccp-cls-protocol-im

Class Map type inspect match-all ccp-icmp-access (id 9)
  Match class-map ccp-cls-icmp-access

Class Map type inspect match-all ccp-invalid-src (id 31)
  Match access-group 100

Class Map type inspect match-any ccp-sip-inspect (id 30)
  Match protocol sip

Class Map type inspect match-all ccp-protocol-imap (id 11)
  Match protocol imap

Class Map type inspect match-all ccp-protocol-http (id 36)
  Match protocol http
```

R3#

```
! Note, although there is lot of output, the objective is to understand
! the commands that allow you to see what is happening from the CLI
! In the example content below, we see the detailed information
! regarding a telnet session that is currently going through the firewall,
! as well as a PING that is being sent through the firewall.
```

R3# **show policy-map type inspect zone-pair ccp-zp-in-out sessions**

```
policy exists on zp ccp-zp-in-out
```

```
Zone-pair: ccp-zp-in-out
```

```
Service-policy inspect : ccp-inspect
```

```
<snip>
```

```
Inspect
```

```
Class-map: ccp-insp-traffic (match-all)
```

```
Match: class-map match-any ccp-cls-insp-traffic
```

```
Match: protocol cuseeme
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol dns
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol ftp
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol https
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol imap
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol pop3
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol netshow
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol shell
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol realmedia
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol rtsp
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol smtp extended
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol sql-net
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol streamworks
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol tftp
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol vdolive
  0 packets, 0 bytes
  30 second rate 0 bps
Match: protocol tcp
```

```

    0 packets, 0 bytes
    30 second rate 0 bps
Match: protocol udp
    0 packets, 0 bytes
    30 second rate 0 bps

Inspect

Number of Established Sessions = 2
Established Sessions
Session 673BBD00 (10.0.0.11:29333)=>(34.0.0.4:23) tacacs:tcp SIS_
OPEN
    Created 00:02:20, Last heard 00:00:37
    Bytes sent (initiator:responder) [39:273572]
Session 673BC100 (10.0.0.22:8)=>(34.0.0.4:0) icmp SIS_OPEN
    Created 00:00:40, Last heard 00:00:00
    ECHO request
    Bytes sent (initiator:responder) [69912:69912]

<snip>

```

Implementing NAT in Addition to ZBF

You can add *Network Address Translation (NAT)* to the same router that is performing ZBF. The simplest way to implement NAT is to use the outside interface of the firewall and overload on that address (*Port Address Translation [PAT]*). All packets going through the firewall toward the Internet would appear to the Internet as coming from the single global address of the router. The router may have obtained its global IP address via *Dynamic Host Configuration Protocol (DHCP)* from the service provider, or it may be statically configured.. You can verify the address by looking at the details of the interface to determine how the IP address was obtained and what the current IP address is.

To configure NAT, navigate to **Configure > Router > NAT**, and from there, launch the basic NAT Wizard, as shown in Figure 13-13.

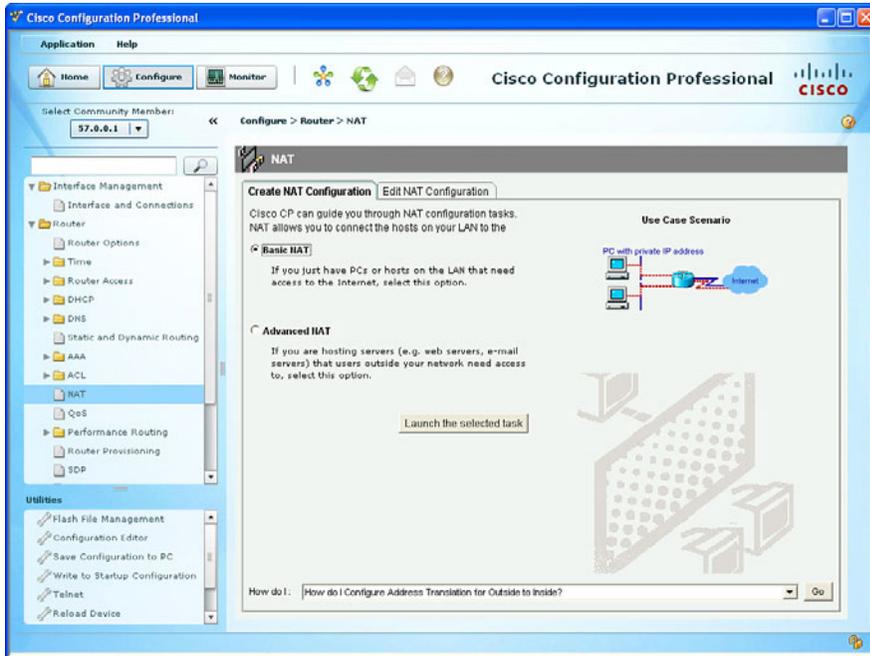


Figure 13-13 Basic NAT Wizard

If you have a DMZ or are hosting a server (on a third interface) and that server needs specific address translation so that users on the outside may access it, you could use the Advanced NAT Wizard to configure that in addition to the Basic NAT for the inside users. Basic NAT is just for inside users accessing outside resources.

For this example, we use the Basic NAT Wizard. When we click the **Launch the Selected Task** button, we are presented with the welcome screen for the Basic NAT Wizard, as shown in Figure 13-14.

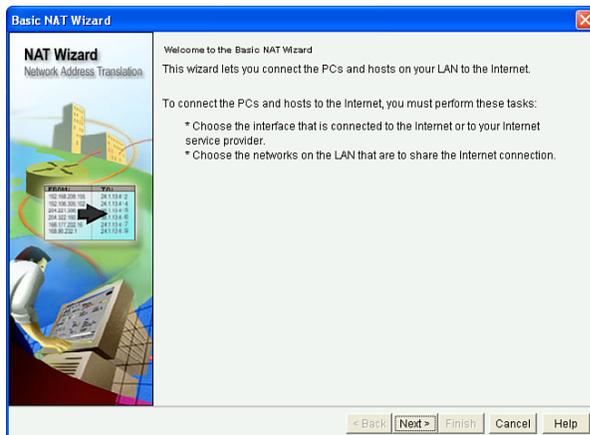


Figure 13-14 Welcome Screen of the Basic NAT Wizard

When we click **Next** to continue, the router needs to know which of our interfaces are connecting to the Internet. (This will be our untrusted interface, if we have configured ZBF.) The wizard also wants to know which internal networks will be permitted to be translated as packets from devices on those networks are routed through the firewall and out toward the Internet. Figure 13-15 shows an example of selecting these interfaces.

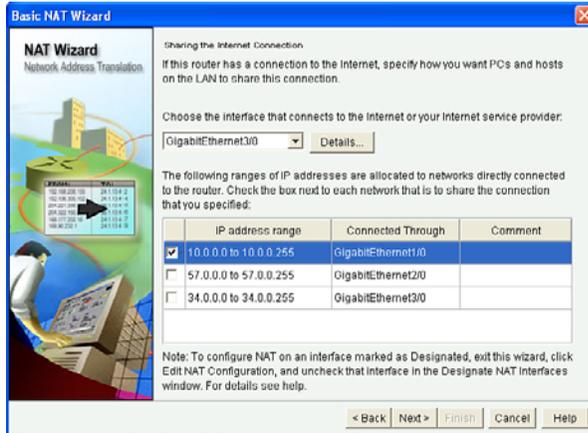


Figure 13-15 *Selecting the Appropriate Interfaces for NAT*

If we want to allow traffic from the DMZ network to also be translated to this single IP address on the outside interface of the router, we can just add it by placing a check in the check box next to that network address range.

When we click **Next** to continue, the wizard presents a summary of the configuration it is about to implement, as shown in Figure 13-16.

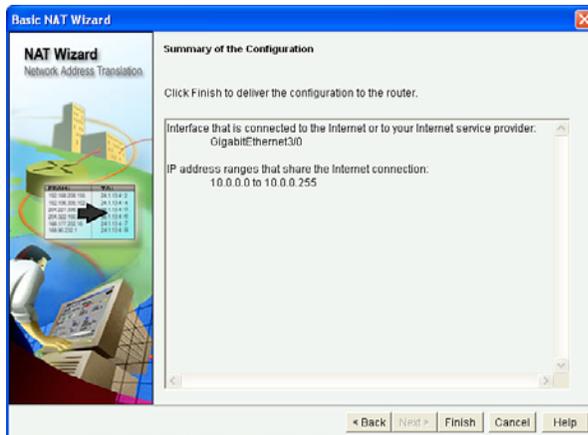


Figure 13-16 *Summary Screen for the Basic NAT Wizard*

Click the **Finish** button to continue. You can also implement NAT from the CLI. The configuration that we just implemented via CCP is implemented with the commands shown in Example 13-4.



Example 13-4 *Implementing NAT*

```

! The ACL is a classifier, that is simply identifying (matching) on packets
! that have a source IP address from the 10.0.0.0/24 network.
R3(config)# access-list 2 permit 10.0.0.0 0.0.0.255

! The next two commands tell the router which of its interfaces from a NAT
! perspective are connected to the inside versus the outside.
! The normal configuration is to translate packets coming from the inside
! that are going to be outside. This is also referred to as inside NAT
! notice also that the term NAT is often used, even when in reality
! we are using port address translation (PAT), as is the case in this
! example
R3(config)# interface GigabitEthernet3/0
R3(config-if)# ip nat outside
R3(config-if)# exit
R3(config)# interface GigabitEthernet1/0
R3(config-if)# ip nat inside
R3(config-if)# exit

! this next rule says that if traffic matches access list number 2
! and the traffic is going to be routed out of the outside interface
! then the source IP address should be translated to the current IP address
! that is configured on the outside interface of the router.
! The overload keyword indicates PAT, and can support thousands of
! internal clients
R3(config)# ip nat inside source list 2 interface GigabitEthernet3/0 overload

```

Verifying Whether NAT Is Working

To verify the configuration, we could show the running configuration from the command line or view it inside CCP. To do the latter, navigate to **Configure > Router > NAT** and select the **Edit NAT Configuration** table. Figure 13-17 shows our current configuration.

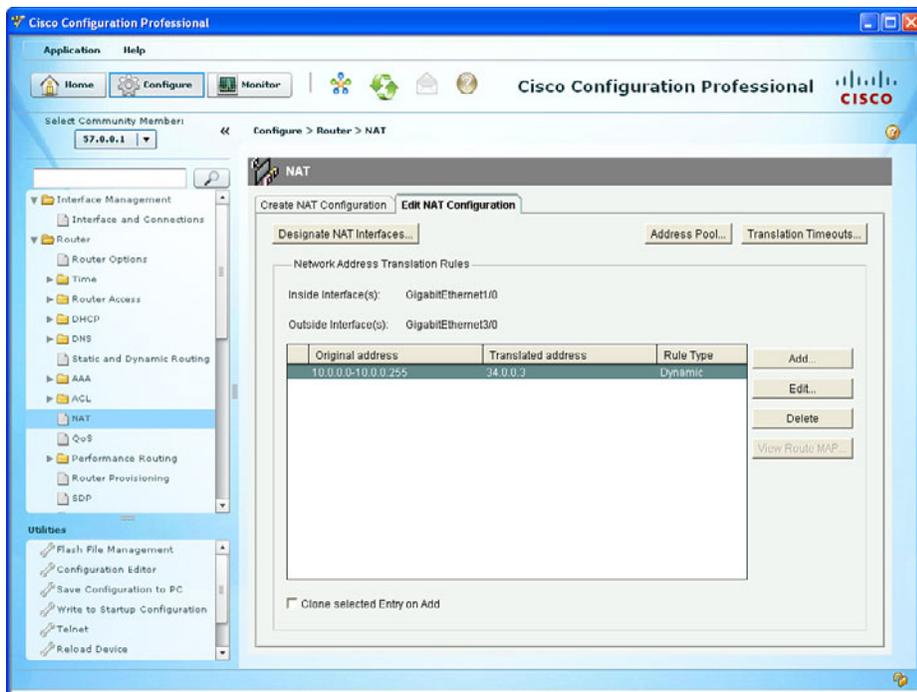


Figure 13-17 Verifying Our NAT Configuration from Within CCP

If you actually have traffic flowing through the router from the inside to the Internet, you should be able to see those translations, as shown in Example 13-5.

Example 13-5 Viewing Existing Translations

```
! Current translations in use include a ping from PC1, a telnet session
! from PC1 and a TFTP file transfer requested by PC2

! All of these sessions have been translated to the routers outside
! address (34.0.0.3) using PAT.

R3# show ip nat translations
Pro Inside global      Inside local      Outside local     Outside global
icmp 34.0.0.3:9        10.0.0.11:9      34.0.0.4:9       34.0.0.4:9
tcp 34.0.0.3:36906    10.0.0.11:36906  34.0.0.4:23      34.0.0.4:23
udp 34.0.0.3:49213    10.0.0.22:49213  34.0.0.4:69      34.0.0.4:69
```



Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 13-5 lists these key topics.



Table 13-5 *Key Topics*

Key Topic Element	Description	Page Number
Text	Overview of how the ZBF operates	294
List	Specific features of the ZBF	294
List	Putting the pieces together	296
Table 13-2	Policy map actions	297
Table 13-3	Traffic interaction between zones	298
Example 13-1	ZBF components	299
Table 13-4	Self zone traffic behavior	300
List	ZBF Wizard configurable security levels	304
Example 13-4	CLI commands to implement NAT	322
Example 13-5	Viewing existing translations	323

Review the Video Bonus Material

Review and practice the CCP video content related to ZBFs. Being able to correctly interpret a policy from the CCP is a significant skill that the video and hands-on will help you master.

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

zones, zone pairs, class map type inspect, policy map type inspect, service policy, stateful inspection, PAT

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 13-6 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 13-6 *Command Reference*

Command	Description
show class-map type inspect	Show ZBF-related class maps
show policy-map type inspect	Show ZBF related policy maps
class-map type inspect match-any MY-CLASS-MAP	Create a ZBF-related class map that will be a match if <i>any</i> of its entries is a match
policy-map type inspect MY-POLICY-MAP	Create a ZBF-related policy map
class type inspect MY-CLASS-MAP	Used inside of a ZBF policy map to call on the classification services of a zone-based class map
zone-pair security in-to-out source inside destination outside	Create a zone pair that identifies an initial unidirectional flow of traffic
show ip nat translations *	Show current active address translations occurring on the router



This chapter covers the following subjects:

- The ASA appliance family and features
- ASA Firewall fundamentals
- Configuring the ASA

Configuring Basic Firewall Policies on Cisco ASA

Cisco over the years has had a dedicated firewall appliance. Many years ago, it was a device named the PIX. As technology improved, a new device was created that leveraged all the features of the PIX and added some new ones. This new device is the *Adaptive Security Appliance (ASA)*. It is pronounced as if you are saying the three individual letters of A, S, A. It is important to understand the basics of a device before you configure it, so that's what we do here in this chapter. You first learn the logic of what the ASA goes through as it makes its determination about forwarding or filtering, and then once you understand that, we configure and implement a security policy and verify the security policy with the built-in tool called Packet Tracer.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter's topics before you begin. Table 14-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 14-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
The ASA Appliance Family and Features	1–3
ASA Firewall fundamentals	4–7
Configuring the ASA	8–10

1. Which of the following features does the Cisco ASA provide? (Choose all that apply.)
 - a. Simple packet filtering using standard or extended access lists
 - b. Layer 2 transparent implementation
 - c. Support for remote-access SSL VPN connections
 - d. Support for site-to-site SSL VPN connections
2. Which of the following has an option slot that can support a hardware module? (Choose all that apply.)
 - a. 5505
 - b. 5510
 - c. 5540
 - d. FWSM
3. When used in an access policy, which component could identify multiple servers?
 - a. Stateful filtering
 - b. Application awareness
 - c. Object groups
 - d. DHCP services
4. Which of the following is an accurate description of the word *inbound* as it relates to an ASA? (Choose all that apply.)
 - a. Traffic from a device that is located on a high-security interface
 - b. Traffic from a device that is located on a low-security interface
 - c. Traffic that is entering any interface
 - d. Traffic that is exiting any interface
5. When is traffic allowed to be routed and forwarded if the source of the traffic is from a device located off of a low-security interface if the destination device is located off of a high-security interface? (Choose all that apply.)
 - a. This traffic is never allowed.
 - b. This traffic is allowed if the initial traffic was inspected and this traffic is the return traffic.
 - c. If there is an access list that is permitting this traffic.
 - d. This traffic is always allowed by default.

6. Which of the following tools could be used to configure or manage an ASA? (Choose all that apply.)
 - a. Cisco Security Manager (CSM)
 - b. ASA Security Device Manager (ASDM)
 - c. Cisco Configuration Professional (CCP)
 - d. The command-line interface (CLI)
7. Which of the following elements, which are part of the Modular Policy Framework on the ASA, are used to classify traffic?
 - a. Class maps
 - b. Policy maps
 - c. Service policies
 - d. Stateful filtering
8. When configuring the ASA as a DHCP server for a small office, what default gateway will be assigned for the DHCP clients to use?
 - a. The service provider's next-hop IP address.
 - b. The ASA's outside IP address.
 - c. The ASA's inside IP address.
 - d. Clients need to locally configure a default gateway value.
9. When configuring network address translation for a small office, devices on the Internet will see the ASA inside users as coming from which IP address?
 - a. The inside address of the ASA.
 - b. The outside address of the ASA.
 - c. The DMZ address of the ASA.
 - d. Clients will each be assigned a unique global address, one for each user.
10. You are interested in verifying whether the security policy you implemented is having the desired effect. How can you verify this policy without involving end users or their computers?
 - a. Run the policy check tool which is built in to the ASA.
 - b. The ASA automatically verifies that policy matches intended rules.
 - c. Use the Packet Tracer tool.
 - d. You must manually generate the traffic from an end-user device to verify that the firewall will forward it or deny it based on policy.

Foundation Topics

The ASA Appliance Family and Features

This section examines the various models and offerings for the ASA and many of the features provided by these firewalls.

Meet the ASA Family



The ASA family comes in many shapes and sizes, but they all provide a similar set of features. Typically, the smaller the number of the model represents a smaller capacity for throughput. The main standalone appliance model number begins with a 55, but there are also devices in the ASA family that go into a switch such as a 6500. Table 14-2 describes the various models of the ASA.

Table 14-2 ASA Models

Model	Description
ASA 5505	This is the entry-level device. It is relatively small compared to the other appliances, and is not large enough (that is, not wide enough) to be rack mounted in a 19-inch-wide rack. It comes with a built-in switch that has 8 ports, and 2 of those provide support for Power over Ethernet. By default, all the interfaces on the switch port belong to VLAN 1, and the method used to connect this device to multiple networks is to assign the switch ports to at least 2 separate VLANs and then create <i>switched virtual interfaces (SVI)</i> , which are logical Layer 3 interfaces just like on a management interface for a switch, for each logical Layer 3 interface you want the ASA to use. This is the only ASA 55xx series appliance with a built-in switch and with this behavior. This device has a single slot allowing the addition of a compatible module.
ASA 5510	This firewall has 4 built-in routable interfaces, and a management Ethernet interface that can be used as a dedicated interface for management only or can be converted to be a fifth routable interface on the ASA. This firewall has an option slot that supports a compatible module, such as an <i>intrusion prevention system (IPS)</i> module, which is like having an IPS appliance (if installed) that lives inside the ASA.
ASA 5520, 5540, 5550	These firewalls are like the 5510, with the exception that they have more capacity.

Model	Description
ASA 5585	High-performance, high-capacity firewall devices that support multiple add-ons, such as modules compatible with these appliances. These appliances take a more vertical space in a rack compared the 5510 to 5550.
<i>Firewall Services Module (FWSM)</i> and the ASA Services Module	These are blade firewalls that fit into a compatible switch, such as a 6500. They support many of the same features of the standalone ASA appliances in the 55xx family.

ASA Features and Services

Summing up the exact features of an ASA could take quite a while because most of the features discussed in the previous chapter related to firewalls and different implementations are included in ASA. ASA provides the following features:



- **Packet filtering:** Simple packet filtering normally represents an access list. It is also true with regard to this feature that the ASA provides. The ASA supports both standard and extended access lists. The most significant difference between an access list on an ASA versus an access list on a router is that the ASA never ever uses a wildcard mask. Instead, if it needs to represent a mask related to a **permit** or **deny** statement in an access list, it just use the real mask in the *access control list (ACL)*.
- **Stateful filtering:** By default, the ASA enters stateful tracking information about packets that have been initially allowed through the firewall. Therefore, if you have an access list applied inbound on the outside interface of the firewall that says deny everything, but a user from the inside makes a request to a server on the outside, the return traffic is allowed back in through the firewall (in spite of the access lists that stops initial traffic from the outside) because of the stateful inspection that is done by default on the initial traffic from the client out to the server, which is now dynamically allowing the return traffic to come back in. This is probably the most significant and most used feature on the ASA. One way of thinking about stateful filtering is to imagine that the ASA is going to build a dynamic permit entry in a virtual ACL that will permit the return traffic. Suppose that you are sending a packet to a web server. Your source address is 4.4.4.4, and your source TCP port is 4444. The destination IP address of the server is 5.5.5.5, and the destination port is TCP 80 (web/HTTP). The ASA will (virtually, as this is just a way to consider it) remember this outbound session and expect to see a return packet from 5.5.5.5 destined to 4.4.4.4 (the client), and the source port is TCP:80 (for the return packet), and the destination port is TCP:4444 (again going back to the client). The “virtual” ACL, or state table, that is dynamically created by the ASA would say please permit this packet (the return one) from the outside network to the inside network where the client is waiting for this reply.
- **Application inspection/awareness:** The ASA can listen in on conversations between devices on one side and devices on the other side of the firewall. The

benefit of listening in is so that the firewall can pay attention to application layer information. An example of this is a client on the inside of our network going to an FTP server on the outside. The client may open a connection from a source port of 6783 to the well-known FTP port of TCP:21. During the conversation between the client and the server, stateful inspection is inspecting traffic (and allowing reply traffic inbound from the outside networks) as long as the source IP address is the server and the source port is 21 (coming from the server back to the client) and the destination port is 6783. That is how stateful inspection works. Unfortunately, some applications, such as FTP, dynamically use additional ports. In the case of standard FTP, the client and the server negotiate the data connection, which is sourced from ports 20 at the server and destined for whatever port number was agreed to by the client. The challenge with this is that the initial packets for this data connection are initiated from the server on the outside. As a result, normal stateful filtering denies it (either by default rules or an access list that is denying initial traffic from the outside). With application layer inspection, the ASA learns about the dynamic ports that were agreed to and dynamically allows the data connection to be initiated from the server who is on the outside going to the client on the inside.

- **Network Address Translation (NAT):** You learned about the benefits of NAT and *Port Address Translation (PAT)* earlier in this book, and it comes as no surprise that the ASA supports both of these. It supports inside and outside NAT, and both static and dynamic NAT and PAT, including Policy NAT, which is only triggered based on specific matches of IP addresses or ports. There is also the ability to perform NAT exemption (for example, specifying that certain traffic should not be translated). This comes in handy if you have NAT rules that say everybody who is going from the inside networks out to the Internet should be translated, but at the same time you have a *virtual private network (VPN)* tunnel to either a remote user or a remote network. Any traffic from the inside network going over the VPN tunnel in most cases should not be translated, so you set up an exemption rule that says traffic from the inside networks to the destinations that are reachable via the VPN tunnels should not be translated. The policy that indicates that traffic should not be translated is often referred to as *NAT zero*.
- **DHCP:** The ASA can act as a *Dynamic Host Configuration Protocol (DHCP)* server or client or both. This is a handy feature when implementing a firewall at a smaller office that might require getting a globally routable address from our service provider through DHCP and at the same time the ability to hand out addresses to the internal DHCP clients at that location.
- **Routing:** The ASA supports most of the interior gateway routing protocols, including RIP, EIGRP, and OSPF. It also supports static routing.
- **Layer 3 or Layer 2 implementation:** The ASA can be implemented as a traditional Layer 3 firewall, which has IP addresses assigned to each of its routable interfaces. The other option is to implement a firewall as a transparent firewall, in which the actual physical interfaces receive individual IP addresses, but a pair of interfaces operate like a bridge. Traffic that is going across this two-port bridge is still subject to the rules and inspection that can be implemented by the ASA. The ASA can still perform application layer inspection and stateful filtering.

- **VPN support:** The ASA can operate as either the head-end or remote-end device for VPN tunnels. When using IPsec, the ASA can support remote-access VPN users and site-to-site VPN tunnels. When supporting *Secure Sockets Layer (SSL)*, it can support the clientless SSL VPN and the full AnyConnect SSL VPN tunnels (which hand out IP addresses to remote VPN users, similar to the IPsec remote VPN users). SSL is a very upcoming and popular option for VPNs and is only used for remote access, not for site-to-site VPNs.
- **Object groups:** An object group is a configuration item on the ASA that refers to one or more items. In the case of a network object group, it refers to one or more IP addresses or network address ranges. The benefit of an object group is that a single entry in an access list could refer to an object group as the source IP or destination IP address in an individual access control entry (a single line of an access list), and the ASA logically applies that entry against all the IP addresses that are currently in the object group. If an object group has four IP addresses in it, and we use that object group in a single entry of an access list that permits TCP traffic to the object group, in effect we are allowing TCP traffic to each of those four IP addresses that are in the group. If we change the contents of the group, the dynamics of what that access list permits or denies also change.
- **Botnet traffic filtering:** A botnet is a collection of computers that have been compromised and are willing to follow the instructions of someone who is attempting to centrally control them (for example, 10,000 machines all willing [or so commanded] to send a flood of ping requests to the IP address dictated by the person controlling these devices). Often, users of these computers have no idea that their computers are participating in this coordinated attack. The ASA works with an external system at Cisco that provides information about the Botnet Traffic Filter Database and so can protect against this.
- **High availability:** By using two firewalls in a high-availability failover combination, you can implement protection against a single system failure.
- **AAA support:** The use of *authentication, authorization, and accounting (AAA)* services, either locally or from an external server such as *Access Control Server (ACS)*, is supported.

ASA Firewall Fundamentals

This section covers the logic that is used by the ASA to provide firewall services, the various ways to manage the firewall, and the components used to implement policy.

ASA Security Levels

With the IOS *Zoned-Based Firewalls (ZBF)* discussed in a previous chapter, we placed interfaces into zones, and no traffic was allowed between zones until we specified a policy.

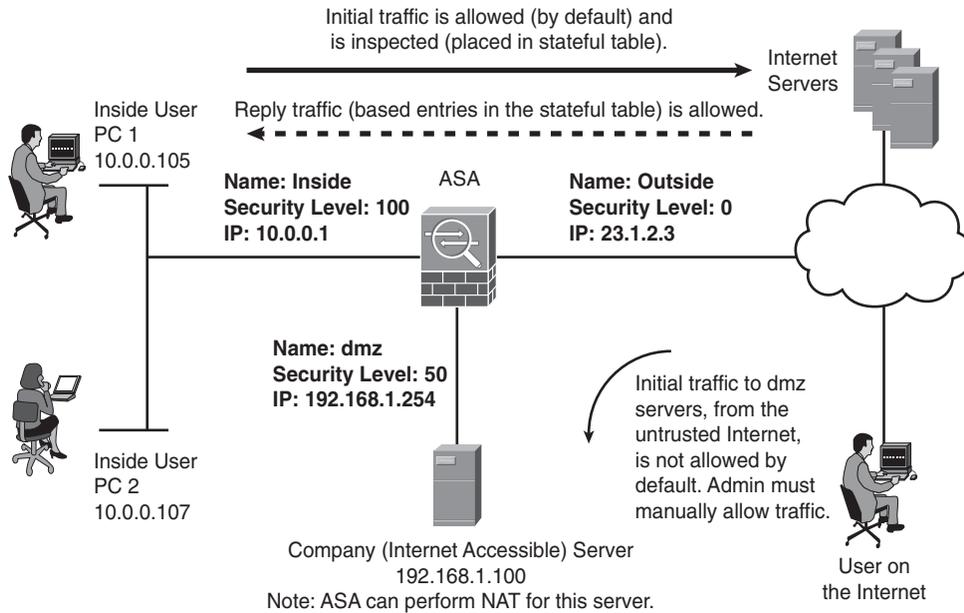


With the ASA, it works a bit differently. The ASA uses security levels associated with each routable interface. The security level is a number between 0 and 100. The bigger the number, the more trust you have for the network that the interface is connected to. For example, I would very likely give a value of 100 to the interface that is connected to my inside network because it is the most trusted network I am connected to. Be aware, though, that we are not just talking about the directly connected network, but also any packets that might come in to the firewall via this interface. So, if I have a firewall with an inside interface, and I have assigned the interface a security level of 100, and if there are 57 additional networks that all forward traffic to me through that single interface, they are all considered as coming in through a security 100 level interface. If your firewall is connected to an untrusted network, such as the Internet, which is not under your control, you very likely will assign for that interface minimum security level of 0.

In addition to assigning security levels to interfaces, you also assign a name to the interface, such as *inside* on the interface that connects to your trusted inside network, or *outside* to label the interface that connects to the Internet. You do not have to use these names, but if you do not, your co-workers might laugh, and the rest of the world may wonder. In all seriousness, it makes sense to label them this way even though you are free to choose. Each of these interfaces is assigned an IP address, and the syntax for assigning an IP address is just the same as on a Cisco router. So, you do three things to make interfaces on the ASA operational:

- Assign a security level to the interface
- Assign a name to the interface
- Bring up the interface with the **no shutdown** command

Besides the inside and outside interfaces, it is typical to have at least one interface that is somewhere in between. The reason for this is you may have a web server that you want to make available out to the Internet, and the best practice for allowing users to access a resource on your network is to avoid placing your server on your internal private network, instead placing it on a separate network off of the firewall and allowing users from the outside the limited access they need to that server that lives off of this third interface of the firewall. A common name assigned to this third interface is the *demilitarized zone (DMZ)*. Besides an IP address for this interface and the name of DMZ, you also have to assign a security level. The security level for a DMZ type of interface off of the firewall is usually set to somewhere between 1 and 99. Let's choose 50 for the DMZ interface, as shown in Figure 14-1.



On the ASA 5505, there is a built-in 8-port switch used for the physical connectivity to the users, the server on the dmz, and the physical connection to the service provider/Internet.

Figure 14-1 Typical Topology for a Firewall with DMZ

The Default Flow of Traffic

By default, and this is important, the ASA forwards traffic (assuming it has a route to know where to forward it) if the initial traffic is sourced from a device that lives off of its high-security interface (such as the inside at security 100, which is the highest) and if the destination of the packet is being routed out of an interface that has a lower security level. That's it in a nutshell. So, a user on the inside can initiate traffic to devices off of the DMZ because that is going from higher to lower security levels (100 to 50). The user on the inside can initiate traffic to a server that lives off of the outside interface because that initial traffic is also going from a higher security level to a lower security level. This is where the default stateful inspection happens, as initial traffic goes through the firewall. As a result of this inspection, when the server on the outside replies back to our inside client, the ASA dynamically allows that return traffic only because it is in the stateful table of the ASA that is expecting that return traffic. The same thing happens for return traffic from the DMZ back to the inside client, again because of the initial inspection.

So, you might ask, how is that a firewall? What is it stopping? The firewall, by default, is stopping all initial traffic that is trying to go from lower security levels to higher security levels. For example, a server on the outside is trying to start a conversation with the server on the DMZ network. Because the initial traffic came in on an interface at a security level of 0 and is trying to go uphill through an interface on the DMZ with the security



level of 50, that traffic is denied by default. If we play this all the way through, servers from the outside cannot initialize traffic to the DMZ or to the inside. The DMZ devices could initialize traffic out to the outside (from high to low, 50 going to 0), but that same DMZ device could not initialize a conversation going to the inside (from low to high, 50 trying to go to 100). Think of it like water: The default policy is like a waterfall, in that it allows higher to go to lower. The stateful inspection determines whether the reply traffic (in response to the initial session) is allowed to make it back through. Figure 14-2 shows the initial traffic that will be permitted and the reply traffic that is being permitted because of the stateful filtering. Without any additional controls in place, this is the full extent of what the firewall allows.

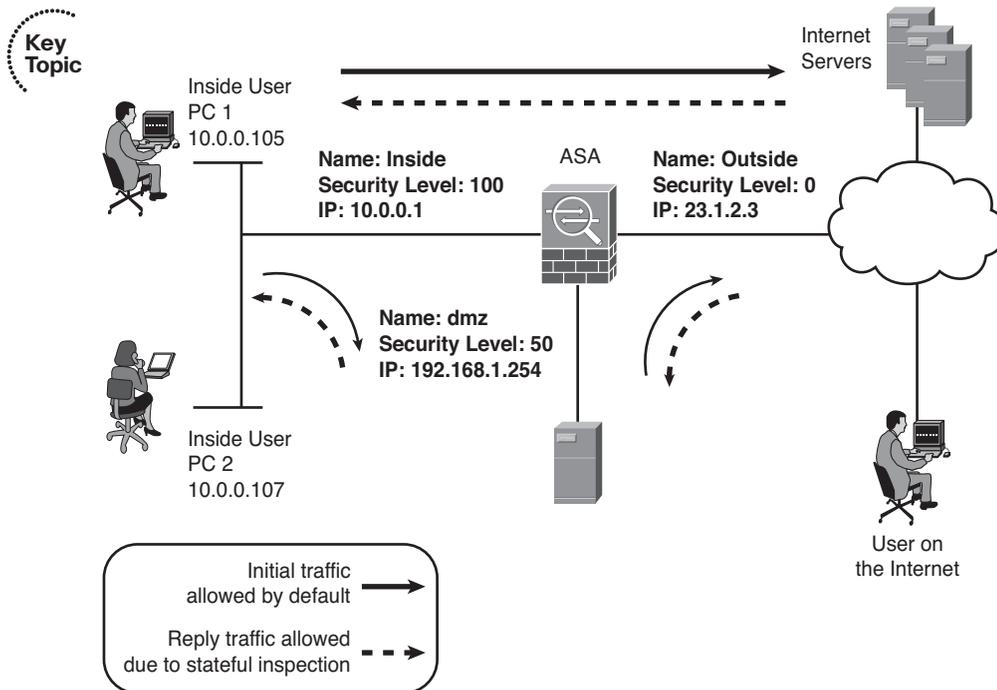


Figure 14-2 *Default Permissions and Return Traffic Allowed Because of Stateful Filtering*

By default, if two interfaces are both at the exact same security level, traffic is not allowed between those two interfaces. Also by default the ASA does not like to (meaning it will not) receive a packet on a specific interface and route the same packet out of the exact same interface (hairpin routing). You can change both of these default behaviors.

Tools to Manage the ASA

You can use several tools to manage an ASA, including the following:

- **Command-line interface (CLI):** Functions similar to IOS
- **ASA Security Device Manager (ASDM):** GUI included with the ASA

- **Cisco Security Manager (CSM):** An enterprise (commercial grade) GUI tool that can manage most of your network devices, including routers, switches, and security appliances such as the ASA

Initial Access

On a brand-new ASA without an IP address configured, you connect to the console port of the ASA. In the old days, most computers had an RS-232 type serial port, and we connected from that serial port on the computer using the blue rollover cable to the console port. In current times, we have moved to USB on the computers. Part of your equipment that you should carry with you is a USB-to-serial adapter so that you can connect from your USB port to the serial blue cable, which then connects to the console port on your network device (in this case, the ASA).

At the CLI, we have access to ROM Monitor, EXEC mode (both user and privileged), configuration mode, interface configuration mode, and several others. Context-sensitive help and navigation at the CLI is similar to the router, including the functions of enable, exit, Ctrl+Z, and so on. One difference worth noting is that if you scroll through multiple pages of output, you use the letter Q (for quit) to stop scrolling through the pages of output one page at a time.

ASDM is a configuration tool that is built in to (included with) the ASA Firewall family. We use ASDM to implement and verify our security policy later in this chapter. It uses SSL to ensure secure communications and runs on a variety of Windows platforms. You can connect to up to five separate firewalls and switch between them conveniently from ASDM.

Packet Filtering on the ASA

We discussed the default flow of traffic, which allows initial traffic to flow from higher to lower security levels, and we also discussed why the reply traffic is allowed (because of the stateful filtering and the database created from the initial traffic flow). But what about individuals on the outside who need access to your web server on the DMZ? You need to allow that initial traffic if you want the customer to have access. To provide access, you can implement packet filtering access lists on the interfaces. These work just like IOS router access lists; there are both standard and extended lists, and they can be applied inbound and outbound to a given interface.

The word *inbound*, from a firewall perspective, could mean traffic that is trying to go uphill (from a low-security interface to a higher-security interface). By default, initial traffic in this direction is denied by the firewall. Reply traffic matching the database is allowed. Another use of the word *inbound* is as traffic is going into an interface. In this context, traffic from users on the inside of the network as they send traffic to the outside world is inbound to the inside interface. If there is a user or device on the outside network that is sending traffic to the inside network, those packets, as they reach the firewall's outside interface, are also inbound to that interface. Here's a quick summary:

- **Inbound to an interface:** Traffic that is going into an interface (any interface). This is also referred to as ingress traffic (from an interface perspective).



- **Inbound from a security level perspective:** Traffic that is being routed by the ASA from a lower-security interface to a higher-security interface, such as from the outside to the DMZ, from the outside to the inside, from the DMZ to the inside. This is from a high-level perspective of the firewall as a whole device.
- **Outbound to an interface:** Traffic that is exiting an interface (any interface) is also referred to as egress traffic (from an interface perspective).
- **Outbound from a security level perspective:** Traffic that is being routed by the ASA from a high-security interface to a lower-security interface, such as inside to DMZ, inside to outside, or DMZ to outside. This is from a high-level perspective of the firewall as a whole device.

Implementing a Packet-Filtering ACL

Now that we have that cleared up, let's return to the issue of a device on the outside needing to initiate a connection to a server on the DMZ. To make that happen, you use an access list that specifically permits the traffic to the server from the outside. If the server will be accessed by the general public, the access list specifies that any device has access through the firewall to the DMZ server as long as the destination IP address and port numbers match the server's address and services offered by the DMZ server. If the access list is applied inbound on the outside interface, any permit entries inside the access list allow traffic to be sourced on the lower-security interface and go to the higher-security interface such as the DMZ. One thing that a lot of people learned the hard way is that just like a router, there is an implicit **deny** at the end of an access list. For outside users, this is no big deal, because previously they had no access to the DMZ, and after the access list they are simply being allowed access to only that one server. The big challenge with access lists comes into play when you apply them inbound on a high-security interface such as the inside interface. When you do that, the inside users can initiate connections through the firewall if there is an explicit **permit** statement allowing it. So, if you are using access lists on each interface of the ASA, the security levels no longer control what the initial traffic flows may be. With access lists, the initial traffic flow is completely controlled by the entries in that access list, which are processed from top to bottom; and the stateful inspection, which is still being done dynamically, allows the return traffic to come back through the firewall regardless of any access lists in place (related to the return traffic).

In short, everything you learned about access lists in the previous chapter on that topic applies to access lists on the ASA. The biggest implementation difference is that no wildcard masks are used on the ASA access lists, but rather just normal masks.

Modular Policy Framework



For IOS ZBFs, class maps are used to identify traffic, policy maps are used to implement actions on that traffic, and the application of those policies is done with the service policy commands. On the IOS router, all of these features included the keywords **inspect** to differentiate them from normal class maps and policy maps and service policies.

On the ASA, you also use class maps to identify traffic, policy maps to identify the actions you are going to take on that traffic, and service policy commands to implement the policy. The service policies can attach the policy to a specific interface or can be applied globally, which would affect all interfaces on the ASA. One way to use the *Modular Policy Framework (MPF)* is to allow the ASA to perform application layer inspection on FTP traffic, to listen in and dynamically allow the data connection to commence from the server (as discussed earlier in this chapter). Another option is that you want to take the traffic destined for your servers and forward it to the IPS module that installs as a hardware add-on in your ASA in its option slot. Another example is that you want to prioritize the forwarding of voice traffic so that once again the class map looks for the voice traffic, the policy map says to give priority queuing to the voice traffic, and the service policy implements where that policy applies (either to a specific interface or all interfaces).

Class maps can identify traffic based on Layer 3 and Layer 4. (There are also application-specific class maps for Layers 5 to 7 that identify traffic based on application layer information, a discussion best saved for your CCNP Security training. For now, just stick with normal class maps at Layer 3 and Layer 4). These Layer 3 and Layer 4 class maps can identify traffic using several different methods, including the following:

- Referring to an access list
- Looking at the *differentiated services codepoint (DSCP)* and/or IP Precedence fields of the packet
- TCP or UDP ports
- IP Precedence
- *Real-time Transport Protocol (RTP)* port numbers
- VPN tunnel groups

The policy maps use the services of the class maps to identify traffic, and then specify the actions to take on each class of traffic, which may include the following:

- Reroute the traffic to a hardware module such as the IPS module that is inside the ASA
- Perform inspection on that traffic (related to stateful filtering or application layer inspection/filtering)
- Give priority treatment to the forwarding of that traffic
- Rate-limit or police that traffic
- Perform advanced handling of the traffic

Where to Apply a Policy

You can apply a policy to a specific interface, and any given interface can have only one policy applied to it. You can also apply a policy globally, which means that all interfaces

implement that policy. It is possible that an interface has a manually configured policy and an inherited global policy, at which point both policies are implemented (so long as no conflict of policy exists between the two).

Configuring the ASA

In this section, you use the ASDM GUI to implement and verify a security policy on an ASA Firewall.

Beginning the Configuration

Now that you know the features and functions of the ASA and the core concepts of what it can do (for example, stateful filtering, packet filtering, NAT), it is time to put those concepts into practice.

Most of the time, you will be dealing with a firewall that is already configured and in a production network. If so, you just use the CLI, ASDM, or CSM (if your company owns CSM) to manage the device. However, if it is a brand-new firewall and has no configuration, you want to establish a console port connection to it, power up the firewall, and set your terminal emulation program to connect through the serial cable using 9600 bits per second, no parity, 8 data bits, and 1 stop bit. In your terminal emulation program, you press the **Enter** key on your keyboard to initialize the CLI EXEC session to the ASA. With the console connection, if you watch the ASA power up, a 5505 boot looks similar to what is shown in Example 14-1.

Example 14-1 Initial Boot of the 5505 ASA

```
CISCO SYSTEMS
Embedded BIOS Version 1.0(12)13 08/28/08 15:50:37.45

Low Memory: 632 KB
High Memory: 507 MB
PCI Device Table.
Bus Dev Func VendID DevID Class                Irq
 00 01 00  1022  2080 Host Bridge
 00 01 02  1022  2082 Chipset En/Decrypt 11
 00 0C 00  1148  4320 Ethernet          11
 00 0D 00  177D  0003 Network En/Decrypt 10
 00 0F 00  1022  2090 ISA Bridge
 00 0F 02  1022  2092 IDE Controller
 00 0F 03  1022  2093 Audio            10
 00 0F 04  1022  2094 Serial Bus       9
 00 0F 05  1022  2095 Serial Bus       9

Evaluating BIOS Options ...
Launch BIOS Extension to setup ROMMON
```

```
Cisco Systems ROMMON Version (1.0(12)13) #0: Thu Aug 28 15:55:27 PDT 2008

Platform ASA5505

Use BREAK or ESC to interrupt boot.
Use SPACE to begin boot immediately.

Launching BootLoader...
Boot configuration file contains 1 entry.

Loading disk0:/asa842-k8.bin... Booting...
Platform ASA5505

Loading...
IO memory blocks requested from bigphys 32bit: 9672
¿dosfsck 2.11, 12 Mar 2005, FAT32, LFN
Starting check/repair pass.
Starting verification pass.
/dev/hda1: 178 files, 49617/62844 clusters
dosfsck(/dev/hda1) returned 0
Processor memory 348127232, Reserved memory: 62914560

Total SSMs found: 0

Total NICs found: 10
88E6095 rev 2 Gigabit Ethernet @ index 09 MAC: 0000.0003.0002
88E6095 rev 2 Ethernet @ index 08 MAC: 4055.39ee.d4d7
88E6095 rev 2 Ethernet @ index 07 MAC: 4055.39ee.d4d6
88E6095 rev 2 Ethernet @ index 06 MAC: 4055.39ee.d4d5
88E6095 rev 2 Ethernet @ index 05 MAC: 4055.39ee.d4d4
88E6095 rev 2 Ethernet @ index 04 MAC: 4055.39ee.d4d3
88E6095 rev 2 Ethernet @ index 03 MAC: 4055.39ee.d4d2
88E6095 rev 2 Ethernet @ index 02 MAC: 4055.39ee.d4d1
88E6095 rev 2 Ethernet @ index 01 MAC: 4055.39ee.d4d0
y88acs06 rev16 Gigabit Ethernet @ index 00 MAC: 4055.39ee.d4d8
Encryption hardware device : Cisco ASA-5505 on-board accelerator (revision
0x0)
                Boot microcode           : CN1000-MC-BOOT-2.00
                SSL/IKE microcode         : CNLite-MC-SSLm-PLUS-2.03
                IPSec microcode           : CNlite-MC-IPSECm-MAIN-2.06
Verify the activation-key, it might take a while...
Running Permanent Activation Key: 0xc72beef 0xd033f00d 0xf482cafe 0x8296783
0xc4004713
```

Licensed features for this platform:

Maximum Physical Interfaces	: 8	perpetual
VLANs	: 3	DMZ Restricted
Dual ISPs	: Disabled	perpetual
VLAN Trunk Ports	: 0	perpetual
Inside Hosts	: 10	perpetual
Failover	: Disabled	perpetual
VPN-DES	: Enabled	perpetual
VPN-3DES-AES	: Enabled	perpetual
AnyConnect Premium Peers	: 2	perpetual
AnyConnect Essentials	: Disabled	perpetual
Other VPN Peers	: 10	perpetual
Total VPN Peers	: 25	perpetual
Shared License	: Disabled	perpetual
AnyConnect for Mobile	: Disabled	perpetual
AnyConnect for Cisco VPN Phone	: Disabled	perpetual
Advanced Endpoint Assessment	: Disabled	perpetual
UC Phone Proxy Sessions	: 2	perpetual
Total UC Proxy Sessions	: 2	perpetual
Botnet Traffic Filter	: Disabled	perpetual
Intercompany Media Engine	: Disabled	perpetual

This platform has a Base license.

Cisco Adaptive Security Appliance Software Version 8.4(2)

***** Warning *****

This product contains cryptographic features and is subject to United States and local country laws governing, import, export, transfer, and use. Delivery of Cisco cryptographic products does not imply third-party authority to import, export, distribute, or use encryption. Importers, exporters, distributors and users are responsible for compliance with U.S. and local country laws. By using this product you agree to comply with applicable laws and regulations. If you are unable to comply with U.S. and local laws, return the enclosed items immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
<http://www.cisco.com/wvl/export/crypto/tool/stqrg.html>

If you require further assistance please contact us by sending email to export@cisco.com.

***** Warning *****

Copyright (c) 1996-2011 by Cisco Systems, Inc.

Restricted Rights Legend

Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) of the Commercial Computer Software - Restricted Rights clause at FAR sec. 52.227-19 and subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS sec. 252.227-7013.

Cisco Systems, Inc.
170 West Tasman Drive
San Jose, California 95134-1706

Reading from flash...

Flash read failed

ERROR: MIGRATION - Could not get the startup configuration.
Configuration has non-ASCII characters and will be ignored.

Pre-configure Firewall now through interactive prompts [yes]?

At this point, to initially bootstrap the ASA, you can press **Enter**, to tell the ASA that you want to use the interactive prompts for the initial setup. If you answer **no** to this, you can later run the **setup** command to return to this script. The objective here is to give the ASA enough basic information so that you can connect to it via ASDM, and then use ASDM to configure the rest of it. As we work through this example, we look at both the configuration done through the ASDM and from the CLI. If you answer **yes**, we can supply the basic information needed for connectivity on the ASA by the ASDM, as shown in Example 14-2.

Example 14-2 *Running the Initial Setup Script on the ASA*

Pre-configure Firewall now through interactive prompts [yes]?

! By pressing the Enter key, the value in the brackets, such as the [yes]
! above will be accepted.

! The other option would be transparent mode (non-routed)
Firewall Mode [Routed]:

! Please use a password that is more secure than this example
Enable password [<use current password>]: **cisco123**

```
! pressing enter will accept the option presented in the brackets
Allow password recovery [yes]?
Clock (UTC):
  Year [2013]:
  Month [Mar]:
  Day [2]:
  Time [17:34:41]:

! this will be the IP address on the logical interface VLAN1
! remember all eight switch ports belong to this VLAN by default
! we could use any of the eight ports to connect ASA to our network
! it will name this interface "management", and give it a security level
! of zero you will want to plan ahead of time regarding which IP address
! to use
Management IP address: 192.168.1.254
Management network mask: 255.255.255.0
Host name: Keith-IINS-ASA
Domain name: test.com

! the ASA doesn't allow any ASDM/HTTPS connections to it by default
! it will ask for the address of your computer that you will be using
! to access ASDM, and allow that connection
IP address of host running Device Manager: 192.168.1.7

! a summary is provided before asking you to confirm
The following configuration will be used:
Enable password: cisco123
Allow password recovery: yes
Clock (UTC): 17:34:41 Mar 2 2012
Firewall Mode: Routed
Management IP address: 192.168.1.254
Management network mask: 255.255.255.0
Host name: Keith-IINS-ASA
Domain name: test.com
IP address of host running Device Manager: 192.168.1.7

! if everything looks right you can type yes and press enter to implement
! the changes
Use this configuration and write to flash? yes
INFO: Security level for "management" set to 0 by default.

! it takes a few moments for the self signed certificate to be generated
! by the ASA for use with SSL, so the warning below is only relevant
! for the first few seconds, and then will be ok.
```

```

WARNING: http server is not yet enabled to allow ASDM access.
Cryptochecksum: 3001087b 2c98260b a4ed70b8 06b690d6

2061 bytes copied in 0.930 secs

Type help or '?' for a list of available commands.
Keith-IINS-ASA>

```

As a good initial check to verify that connectivity is at least working from an IP perspective, you can ping a device on the local network (be sure to verify it is a device that is willing to respond to a ping request), as shown in Example 14-3.

Example 14-3 *Issuing an ICMP Echo Request (Ping) from the ASA*

```

Keith-IINS-ASA# ping 192.168.1.100
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.1.100, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/28/130 ms
Keith-IINS-ASA#

```

Getting to the ASDM GUI

With that in place, the next thing to attempt is opening an HTTPS connection from a Windows PC (whose IP address you identified in the setup script). On the initial connection, your PC is given a digital certificate of the ASA, and unfortunately the certificate is self-signed by the ASA, and your browser will not by default trust that certificate. On a brand-new ASA, you need to accept the certificate to get ASDM functionality. Later you can implement a *public key infrastructure (PKI)* signed certificate for the ASA. (For more information on digital certificates, see the chapter on PKI in this book.) After you accept the certificate, you are given the option of running ASDM as an applet directly from the ASA, or you can install the program on your local PC and launch it from there. Either way, when it is launched, ASDM prompts you for a username and password, which is quite interesting because you did not configure any usernames in the setup script. At this point, you just leave the username blank and supply the enable secret that was configured in the setup script. Once you have authenticated, and the configuration is then downloaded from the ASA to ASDM, you are provided with the dashboard for the ASA, as shown in Figure 14-3.

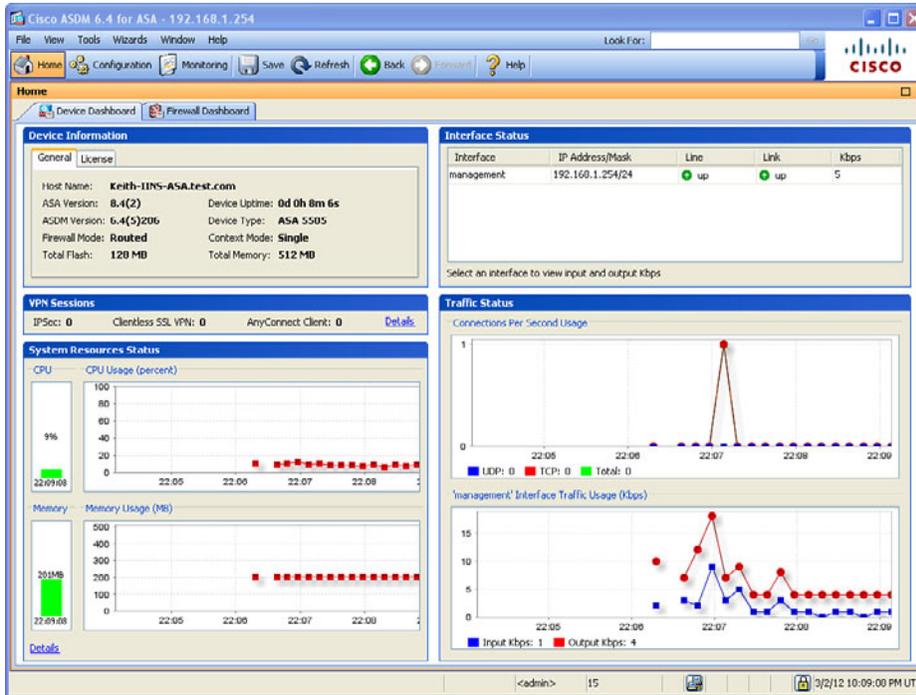


Figure 14-3 Initial Dashboard Presented by the ASDM

As shown in the figure, the dashboard shows the general information about the firewall, including the version of software, the model, the mode it is running in, and the memory size of flash and RAM. A tab shows current licensing information, as well. The dashboard also graphically represents information about VPN sessions, system resources, and traffic status. When you initially connect to the ASA, one of the options is to run the Startup Wizard. In our example, we chose to go directly to ASDM, but it is not too late, on the menu bar is an option labeled Wizards. By choosing that menu option and from the drop-down selecting **Startup Wizard**, you can launch the Startup Wizard, to help you configure more of the basics to get your firewall up and running. Figure 14-4 shows the welcome screen for the wizard.

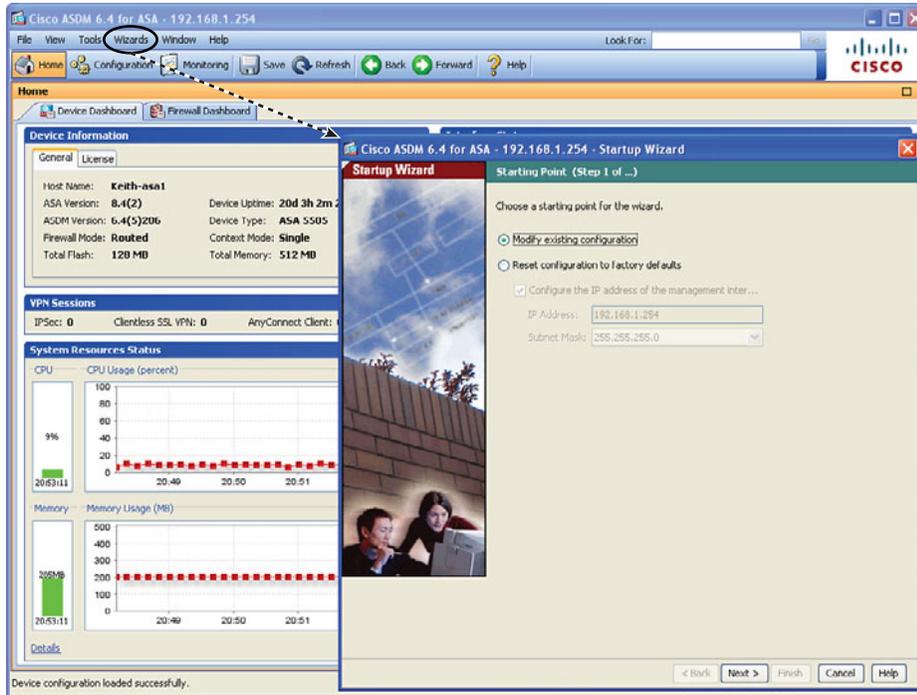


Figure 14-4 Welcome Screen for the Startup Wizard

When you click the **Next** button to continue, you are presented with the option to configure many of the components required for a functional firewall, including the IP addresses to use, the names of the interfaces, NAT configuration, and so forth. From an instructional perspective and certification-relevancy issue, I want to walk you through the configuration manually using the GUI (instead of the wizard) so that you will both know where to go in the interface and how to configure each item.

Configuring the Interfaces

The first order of business is to configure the interfaces. To do this, you click the **Configuration** button on the menu bar in the upper left, and then navigate to **Configuration > Device Setup > Interfaces**, as shown in Figure 14-5.



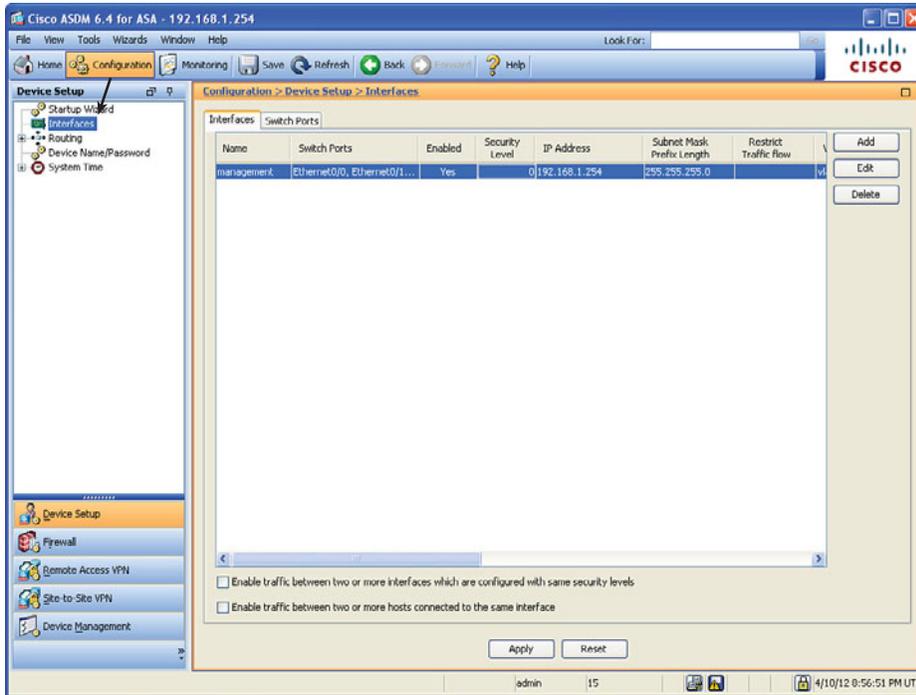


Figure 14-5 Interface Configuration in ASDM

Currently, we have one logical Layer 3 interface. It is interface VLAN 1, and it has been given the name of **management** with a security level of **0** and all of the eight switch ports belong to this VLAN as access ports by default. (This was done from the CLI setup script.) To create new switched virtual interfaces (the Layer 3 interfaces), you just click **Add** and specify the information for each of the interfaces one at a time. Figure 14-6 shows an example of this.

Figure 14-6 shows creating a logical Layer 3 interface named **inside**, with a security level of **100** (which is pretty standard for your interface connected to your internal network). I also associated five of the ports that I intend to use for connecting internal users and devices as belonging to this new VLAN. By default, this creates a new VLAN using a VLAN number that does not currently exist on this firewall. If you click the **Advanced** tab, you can specify the exact number of the VLAN you want, as shown in Figure 14-7.

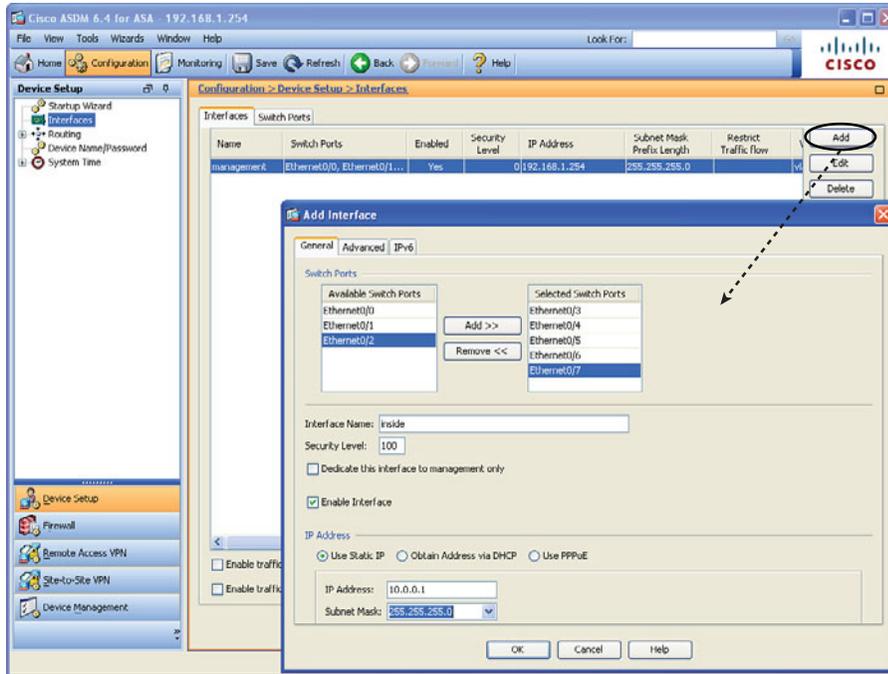


Figure 14-6 *Configuring Interfaces on the ASA*

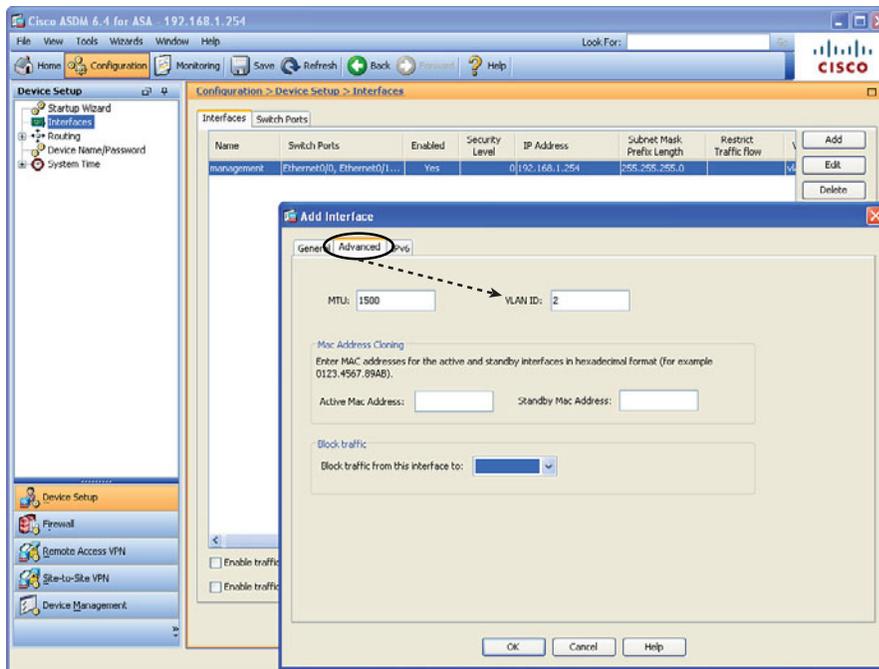


Figure 14-7 *Specifying the VLAN Number to Associate with the New Layer 3 Interface*

In this example, I use the VLAN number 2 for the inside. From this window, you can also specify the *maximum transmission unit (MTU)*, and some information that could be used for high-availability failover. You can also specify that traffic from this VLAN should never be forwarded to another interface, which can be selected from the drop-down list in the Block Traffic section.

The number of named interfaces you are allowed to use is governed by the license on the ASA. The 5505 with a Base license can support at the most three named interfaces. After you have configured your interfaces, you can apply the changes by clicking the **Apply** button (see Figure 14-8).

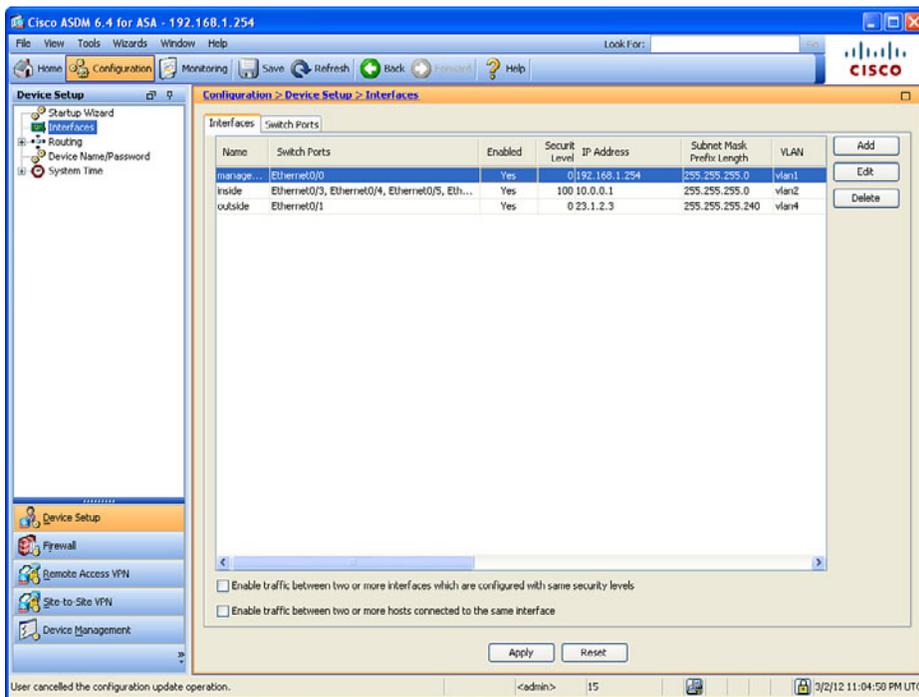


Figure 14-8 Summary Screen for Adding New Interfaces

The Summary page includes the name of the interface, which switch ports are associated with that logical interface, the security level, IP address, VLAN number, and more. If the information does not fit on one page, scrolling to the right will reveal the rest of the columns.

Earlier, we talked about the defaults related to an ASA not being willing to forward traffic between two interfaces if those two interfaces are both at the same security level. We also discussed that the ASA does not like to route a packet out the exact same interface that the packet came in on. To modify both these behaviors, you can place a check mark in each of the check boxes near the bottom of the screen shown in Figure 14-8.

The number of named interfaces that you can create is controlled by the license on the ASA. A 5505 with the Base license supports only three named logical SVI interfaces. In

our scenario, I want to have an inside interface, an outside interface, and a DMZ interface, so I reconfigure the name and security level of the current interface named management and use it as a DMZ interface, as shown in Figure 14-9.

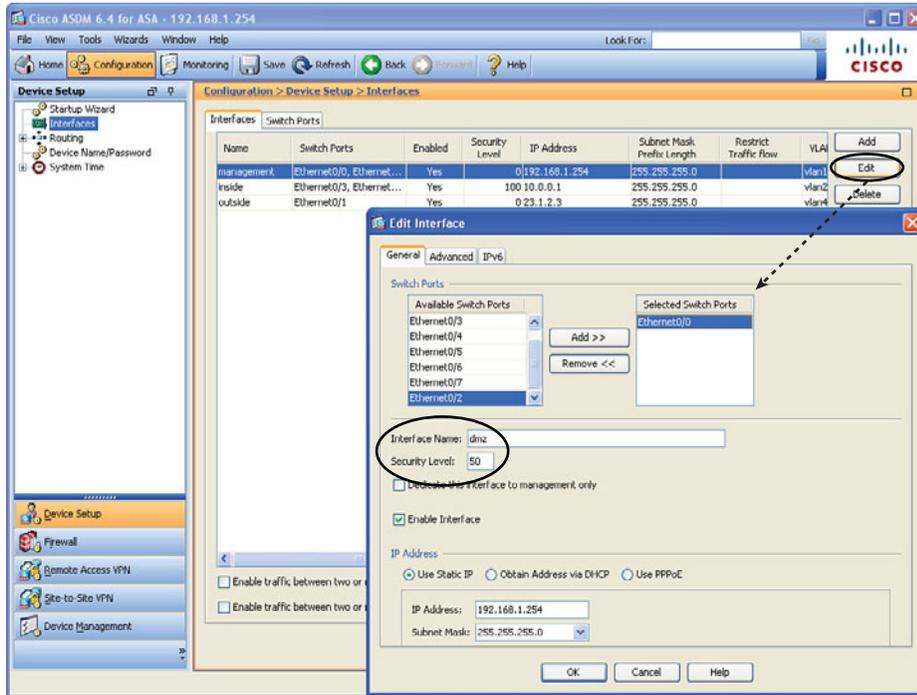


Figure 14-9 *Editing an Interface*

Changing the name of the interface does not change the security posture, but it makes sense to name the interfaces so that when you see them you will know where they connect. In addition, changing the security level (in our example to 50) is important because this interface has a higher security level than the outside but a lower security level than the inside. We could have changed the security level to any number between 1 and 99, inclusive, and achieved the same results. To implement these changes, you click the **OK** button to dismiss the pop-up windows, and click the **Apply** button to implement the changes on the ASA. In the process of making these changes, you may receive warnings from ASDM indicating that changing interface parameters, including security levels, may cause loss of connectivity (especially if you are disabling an interface that you are currently using to communicate with the ASA).

Because you are likely to run into this issue, you need to be aware of one more tweak for this configuration. On the Base license for the 5505, the ASA does not allow full traffic forwarding between all three interfaces. (An upgraded Plus license would do just fine.) So, to implement three named interfaces with the Base license, you must limit functionality: On the Advanced tab shown in Figure 14-7, indicate that you are willing to block traffic from one of your interfaces so that it cannot be forwarded to another specific interface. An upgraded Plus license allows for three full-functioning interfaces.

Figure 14-10 shows some restrictions in place in the Restrict Traffic Flow column. This figure shows restricting traffic from the outside to the inside. (This is an example only; in production, this configuration would not work too well for your users who are trying to get replies from servers on the outside.) I will not be leaving the restriction in place.

Figure 14-10 shows the final configuration.

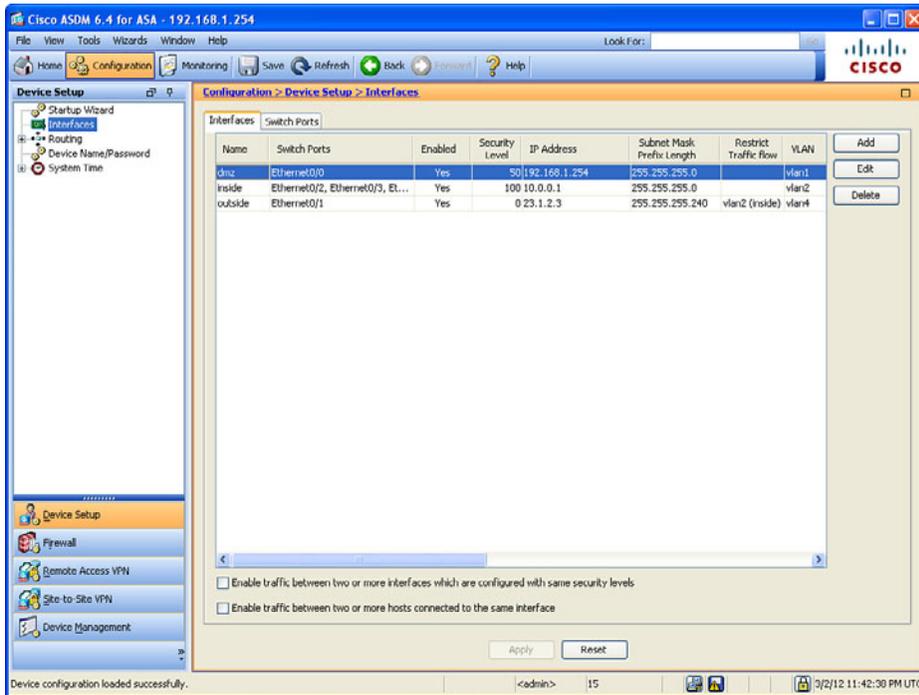


Figure 14-10 Final Configuration of the Interfaces

From the CLI, you can implement these same changes using the commands shown in Example 14-4.

Example 14-4 Implementing Additional Firewall Interfaces

**Key
Topic**

```
Keith-IINS-ASA(config)# configure terminal
Keith-IINS-ASA(config)#

! Configure the logical Switched Virtual Interface (SVI, it is the Layer 3
! interface)
Keith-IINS-ASA(config)# interface Vlan1

! Bring it out of shutdown state
Keith-IINS-ASA(config-if)# no shutdown

! Add an optional description
Keith-IINS-ASA(config-if)# description Connect to the dmz
```

```
! Give the interface a name
Keith-IINS-ASA(config-if)# nameif dmz

! Give the interface a security level
Keith-IINS-ASA(config-if)# security-level 50

! Give the interface an IP address
Keith-IINS-ASA(config-if)# ip address 192.168.1.254 255.255.255.0
Keith-IINS-ASA(config-if)# exit

! Repeat this process for the other interfaces
Keith-IINS-ASA(config)# interface Vlan2
Keith-IINS-ASA(config-if)# no shutdown
Keith-IINS-ASA(config-if)# description Connects to my private network
Keith-IINS-ASA(config-if)# nameif inside
Keith-IINS-ASA(config-if)# security-level 100
Keith-IINS-ASA(config-if)# ip address 10.0.0.1 255.255.255.0
Keith-IINS-ASA(config-if)# exit
Keith-IINS-ASA(config)#
Keith-IINS-ASA(config)# interface Vlan4
Keith-IINS-ASA(config-if)# no shutdown
Keith-IINS-ASA(config-if)# description Connects to the Internet
Keith-IINS-ASA(config-if)# no forward interface Vlan2
Keith-IINS-ASA(config-if)# nameif outside
Keith-IINS-ASA(config-if)# security-level 0
Keith-IINS-ASA(config-if)# ip address 23.1.2.3 255.255.255.240
Keith-IINS-ASA(config-if)# exit
Keith-IINS-ASA(config)#

! Assign the access ports of the built in switch to the VLANs that you
! want them to belong to, repeat for all switch ports you intend to use.
Keith-IINS-ASA(config)# interface Ethernet0/1
Keith-IINS-ASA(config-if)# switchport access vlan 4
Keith-IINS-ASA(config-if)# exit
Keith-IINS-ASA(config)#
Keith-IINS-ASA(config)# interface Ethernet0/2
Keith-IINS-ASA(config-if)# switchport access vlan 2
Keith-IINS-ASA(config-if)# exit
Keith-IINS-ASA(config)#
Keith-IINS-ASA(config)# interface Ethernet0/3
Keith-IINS-ASA(config-if)# switchport access vlan 2
Keith-IINS-ASA(config-if)# exit
Keith-IINS-ASA(config)#
Keith-IINS-ASA(config)# interface Ethernet0/4
Keith-IINS-ASA(config-if)# switchport access vlan 2
```

```
Keith-IINS-ASA(config-if)# exit
Keith-IINS-ASA(config)#
Keith-IINS-ASA(config)# interface Ethernet0/5
Keith-IINS-ASA(config-if)# switchport access vlan 2
Keith-IINS-ASA(config-if)# exit
Keith-IINS-ASA(config)#
Keith-IINS-ASA(config)# interface Ethernet0/6
Keith-IINS-ASA(config-if)# switchport access vlan 2
Keith-IINS-ASA(config-if)# exit
Keith-IINS-ASA(config)#
Keith-IINS-ASA(config)# interface Ethernet0/7
Keith-IINS-ASA(config-if)# switchport access vlan 2
Keith-IINS-ASA(config-if)# exit

! To verify your work:
Keith-IINS-ASA(config)# show run interface

! Note the E0/0 is assigned to VLAN 1 (the dmz interface) and because the
! default is for a port to be assigned to VLAN 1, there is no specific
! configuration that shows up in the interface belonging to VLAN 1
interface Ethernet0/0
!

! The rest of the ports are assigned to non-default VLANs, so they show up
! with the VLAN assignment in their configuration.
interface Ethernet0/1
  switchport access vlan 4
!
interface Ethernet0/2
  switchport access vlan 2
!
interface Ethernet0/3
  switchport access vlan 2
!
interface Ethernet0/4
  switchport access vlan 2
!
interface Ethernet0/5
  switchport access vlan 2
!
interface Ethernet0/6
  switchport access vlan 2
!
interface Ethernet0/7
```

```

switchport access vlan 2
!
interface Vlan1
description Connect to the dmz portion of my network
nameif dmz
security-level 50
ip address 192.168.1.254 255.255.255.0
!
interface Vlan2
description Connects to my private network
nameif inside
security-level 100
ip address 10.0.0.1 255.255.255.0
!
interface Vlan4
description Connects to the Internet
no forward interface Vlan2
nameif outside
security-level 0
ip address 23.1.2.3 255.255.255.240
Keith-IINS-ASA(config)#

```

For the physical connectivity, you would use a patch cable from the appropriate port on the built-in eight-port switch to either a DMZ device, an inside device, or to the outside network device. For example, our server on the dmz is connected with a patch cable to port 0/0 on the ASA. An inside host could be connected to any of the ports between 0/2 through 0/7 because all of those ports have been assigned to VLAN2 (where VLAN 2 is where the inside logical interface (the SVI) of the ASA is configured).

IP Addresses for Clients

Now that the ASA has its own IP addresses, you can configure it to hand out addresses to clients using DHCP by acting as a DHCP server. To do that, navigate to **Configuration > Device Management > DHCP > DHCP Server**, as shown in Figure 14-11.



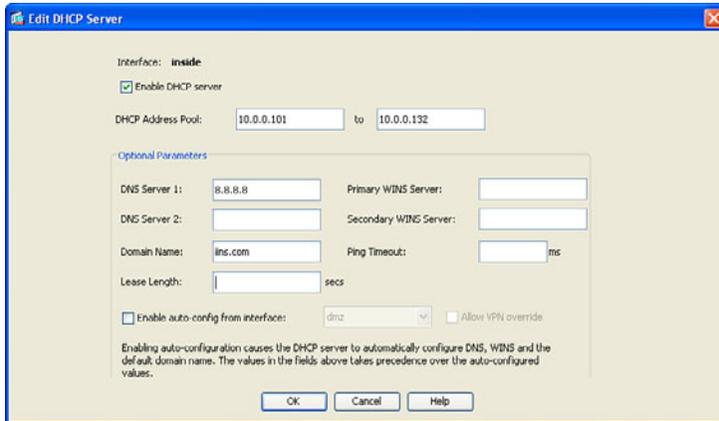


Figure 14-11 Configuring the ASA to Be a DHCP Server

By editing the properties of the inside interface and checking the check box that says you want to enable the DHCP service, you then also apply the pool of addresses that you want to hand out. The maximum size of the pool is 32 on the configuration shown for the 5505. You can also supply DNS and other related information. If we do not specify timeouts and lengths, it assumes the defaults for those values.

Example 14-5 shows the CLI equivalent result for the work we just did in ASDM.

Example 14-5 Configuring the ASA as a DHCP Server for Inside Clients

```
! specifies the pool range, enables the feature and specifies the
! interface
Keith-IINS-ASA(config)# dhcpd address 10.0.0.101-10.0.0.132 inside
Keith-IINS-ASA(config)# dhcpd enable inside
Keith-IINS-ASA(config)# dhcpd dns 8.8.8.8 interface inside
Keith-IINS-ASA(config)# dhcpd domain iins.com interface inside
```

The ASA, by default, assigns itself as the default gateway for the DHCP clients to use.

Basic Routing to the Internet



The ASA needs to know how to forward traffic. Just like a router, ASAs can learn routes via dynamic routing protocols (*interior gateway protocols [IGP]* not *Border Gateway Protocol [BGP]*) from directly connected networks (which an ASA knows how to reach because it is directly connected) or default routes. If you want to look at or modify the routing table on the ASA, navigate to **Configuration > Device setup > Routing**. From this location, you can view or manage static routes and dynamic routing protocols. If you want to add a static route such as a default route, you do that by clicking the **Static Routes** link and then clicking the **Add** button. From there, you use the drop-down menu to choose the interface where you are going to add this route. (This means the interface closest to the next hop where traffic will flow out of this interface to reach the destination network.) Figure 14-12 shows adding a static default route.

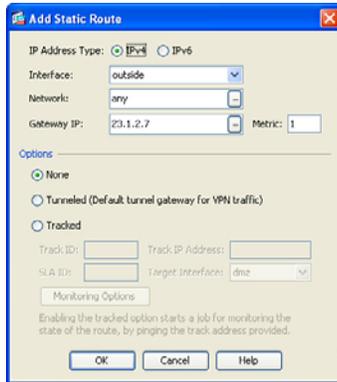


Figure 14-12 Adding a Static Default Route

The default gateway IP address (for use by the ASA) is the IP address of your service provider that is giving you access to the Internet. After you click **OK** and then **Apply**, the changes are sent to the ASA. Example 14-6 shows the CLI equivalent for these commands.

Example 14-6 CLI Equivalent for Adding a Static Route

```
! this tells the ASA that the default route will use the next hop of
! 23.1.2.7
! which is located off of the outside interface (on that same subnet)
Keith-IINS-ASA(config)# route outside 0.0.0.0 0.0.0.0 23.1.2.7
```

NAT and PAT

Now that the ASA knows how to forward to the Internet and the DHCP clients on the inside know to use the ASA as their default gateway, we have a problem with the IP addresses the clients are using on the inside: They are all using private IP address space. Those packets will not be allowed on the Internet. Have no fear, because we know that the ASA can do *Network/Port Address Translation (NAT/PAT)*. To implement this, navigate to **Configuration > Firewall > NAT Rules** and click **Add** (see Figure 14-13).

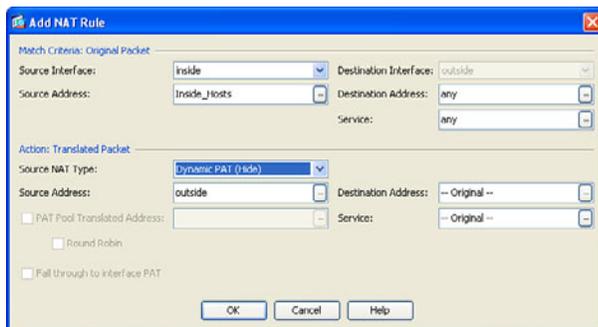


Figure 14-13 Adding NAT Rules

In the top half of the dialog box, you specify the source traffic (where the traffic will be coming from). I specified that the traffic will be coming into the inside interface from an object group that I created named `inside_hosts`. This interface allows you to dynamically create object groups by using the ellipsis button (...). It also asks about the exit interface that packet will have to be taking for the address translation to be used. The bottom half of the dialog box asks what you want to translate the IP address to. If you want to translate the addresses to the outside interface (meaning you are going to use PAT and the global address that is on the outside interface of the ASA), you can specify that you want to use **Dynamic PAT (Hide)** mode and then select the **outside** interface. The ASA translates the user's source IP addresses, and to the Internet it will appear that all transmitted packets (from your clients) are coming from the source address of the outside interface of the ASA. When you click **OK** and **Apply**, this configuration change is sent to the ASA. Example 14-7 shows the CLI equivalent.

Example 14-7 *CLI Equivalent for Implementing Dynamic PAT*

```
! creates a network object that refers to the 10.0.0.0/24 network
Keith-asal(config)# object network Inside_Hosts
Keith-asal(config-network-object)# subnet 10.0.0.0 255.255.255.0
Keith-asal(config-network-object)# description Inside_Hosts
Keith-asal(config-network-object)# exit
! creates a NAT rule that says any traffic sourced from devices
! from the Inside_Hosts object group (network the 10.0.0.0/24 network),
! and coming in on the inside interface, as well as exiting (being routed
! through) the outside interface (based on the routing table of the ASA),
! it would then translate the source address of these packets, and
! substitute the source address of the outside interface of the ASA.
! Additionally it would track this in a NAT/PAT table, that is separate
! from the stateful database, and the ASA would manage both of these
! tables.
Keith-asal(config)# nat (inside,outside) 1 source dynamic Inside_Hosts
interface
! With the NAT on version 8.3 and newer, there are multiple options of
! configuring the NAT, including a NAT command done within object group
! configuration mode. These additional options, including advanced ASA NAT
! configuration are covered in the CCNP Security curriculum.
```

To verify the NAT configuration, navigate to the same location shown in Figure 14-13 and look at the NAT rules, as shown in Figure 14-14.

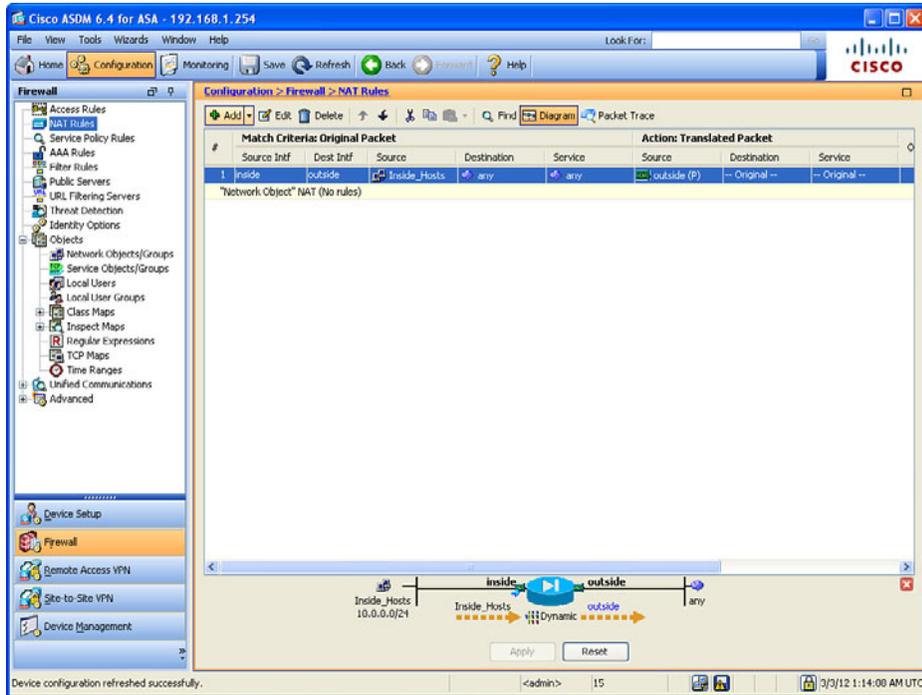


Figure 14-14 NAT/PAT Rules Verification on the ASA

Figure 14-14 shows that devices from the `inside_hosts` group are identified, but we do not know exactly what's in that group. On a live interface, you can hover over that group until a pop-up shows you the details of that group. Another option is to click the **Network Objects/Groups** link in the left navigation pane in Figure 14-14 to look at the details there. One other challenge that might need to be addressed is that we can see the NAT will happen using the outside interfaces IP address. But how do we know what that IP address is (on the outside interface)? Referring to Figure 14-10, we could look at the details of that interface, and even if it is DHCP-assigned for the outside interface (assigned to us by the service provider), the IP address will be revealed there, next to the interface name.

Permitting Additional Access Through the Firewall

The permissions allowing traffic sourced on higher-security interfaces and being routed through egress interfaces with lower security levels is allowed by default, and the stateful nature of the ASA dynamically allows the return traffic. If you want to apply access lists either to filter what the inside users can initiate or to permit access that allows users on the outside to reach our DMZ resources, you can use a packet-filtering access list. To apply an access list, navigate to **Configuration > Firewall > Access Rules**. By default, the policy on the inside and DMZ interfaces (because they are not at security level 0) is to allow traffic sourced by devices on those interfaces to be forwarded to less-secure networks. The default policy on the outside interface is to deny everything (because the



security level is 0). If you click the **Diagram** button, a diagram displays (near the bottom of the screen), which is handy to remind you which interfaces and directions you are working on within the GUI. Also down by the diagram is the option to show IPv4 and or IPv6 access rule types, which are both supported on the ASA, as shown in Figure 14-15.

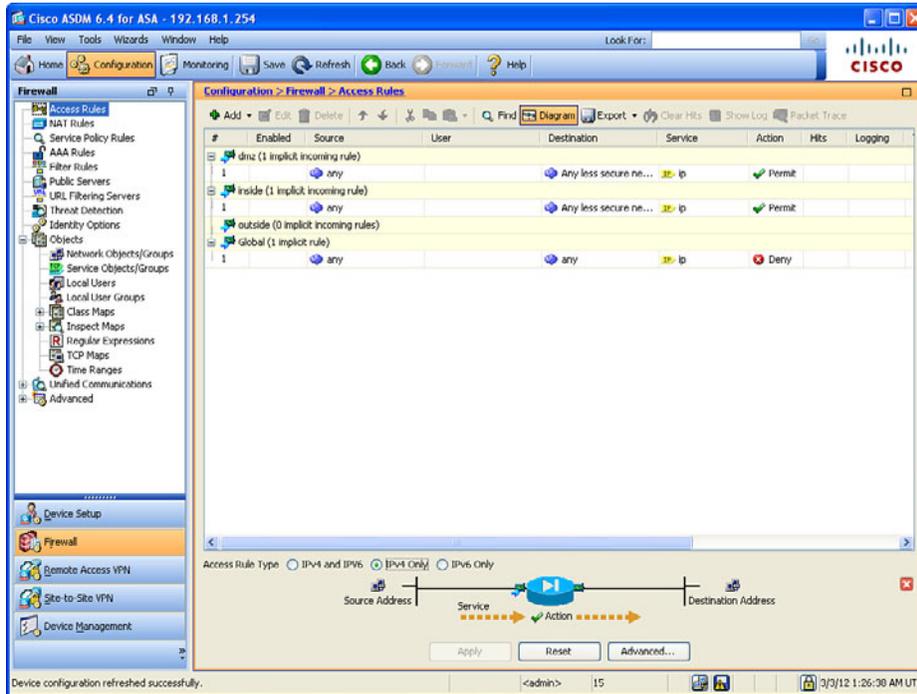


Figure 14-15 Access Rules Configuration Page

To create a new rule that denies Telnet traffic outbound from the inside network to the outside (regardless of source or destination IP address), you can create the rule that specifies TCP port 23 is denied, as shown in Figure 14-16.

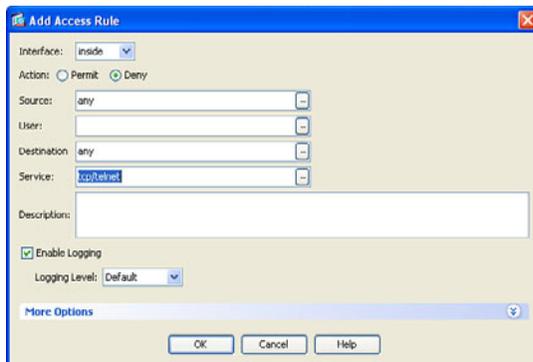


Figure 14-16 Adding an Access Rule

To create a new rule, click the **Add** button. Use the drop-down list to select the interface that will be using this new rule. In the Add Access Rule dialog box, complete the form to indicate the addresses, ports, and protocols involved, and whether to permit or deny this traffic. When the configuration is as you want it, click **OK**.

You now have an access list with one entry. Fortunately, none of this gets applied to the ASA until you click the **Apply** button. So, before you apply the changes, you also want to add a permit entry to this ACL to allow everything else, besides the Telnet traffic, through. Remember that at the end of an access list there is an implied **deny all**.

Click **Add** once more and specify the same interface, this time specifying that you are going to permit all traffic. This then becomes the second entry in your final access list.

Figure 14-17 shows a summary.

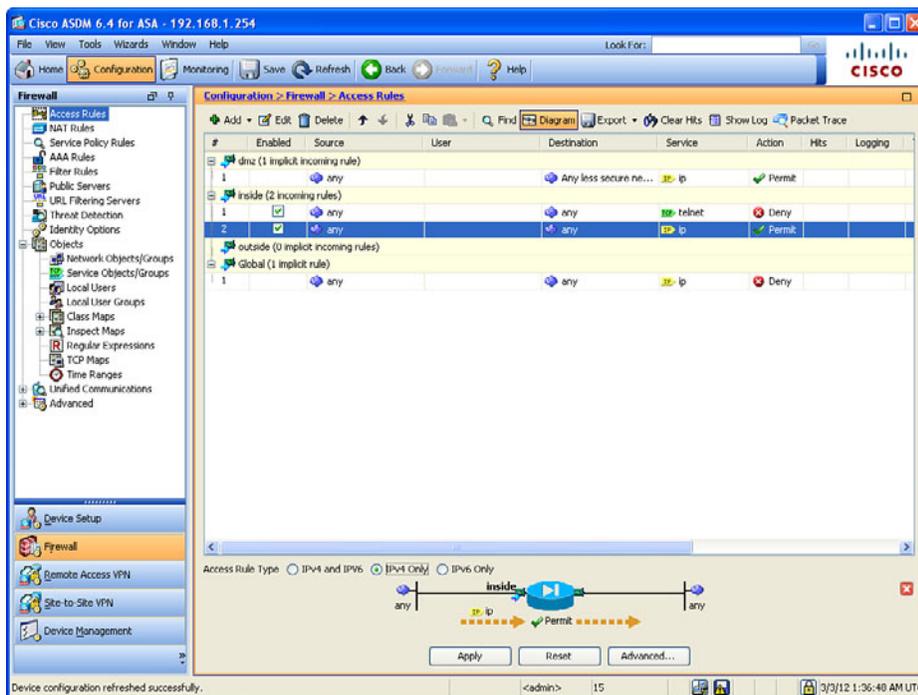


Figure 14-17 Summary Page of Proposed Changes for the Access Rule

The ASDM interface assumes that you will be applying the new ACL in the inbound direction to the specified interface.

After you have all the entries that you want to apply, just click the **Apply** button. Example 14-8 shows the CLI equivalent for the ACL entry (creating and applying it to the interface).

Example 14-8 *Creating and Applying an ACL at the CLI*

```
Keith-asal(config)# access-list inside_access_in deny tcp any any eq telnet
Keith-asal(config)# access-list inside_access_in permit ip any any
Keith-asal(config)# access-group inside_access_in in interface inside
! Note: the optional elements of line number, and extended are optional.
! The ASA assumes the ACL as an extended (if the keyword "standard" isn't
! used)
! In the absence of a "line" command, the ASA adds new entries to the end
! of the ACL
! To apply the ACL, the ASA uses a global access-group command, which is
! different than on an IOS router, where applying an ACL is done in
! interface configuration mode.
```

Using Packet Tracer to Verify Which Packets Are Allowed

Now that the firewall with interfaces has been configured and a default route has been set up and is providing NAT for the benefit of our clients, we should probably make sure that the rules that we have configured, including NAT, are performing as we want. Being able to troubleshoot a problem before it even occurs is a wonderful thing. ASA has a built-in tool called *Packet Tracer* that enables you to identify whether the ASA will forward or drop a packet, before the user even powers on their computer. Packet Tracer even indicates the reason why a packet would be dropped by the ASA.

You can launch Packet Tracer from the Tools menu, and there is an icon for Packet Tracer located on many of the configuration windows as well, including the Access Rules window shown previously in Figure 14-17. Also note that Cisco Academy has a simulator program called Packet Tracer; this is not the same tool as the Packet Tracer tool integrated into the ASAs.

After launching Packet Tracer from either the Tools menu or from an icon in the current window, you are presented with a dialog box in which you enter the specific traffic flow that you want to test or verify. Because we just placed an access list on the inside interface that should permit everything except for Telnet traffic, we can test to see whether web traffic will be forwarded from our inside users as they make requests to web servers out on the Internet. To simulate this, we enter the source IP and port information and destination port and IP address and the interface and packet type that the packet will be using as it leaves the user and enters the firewall. Figure 14-18 shows the input for this test.

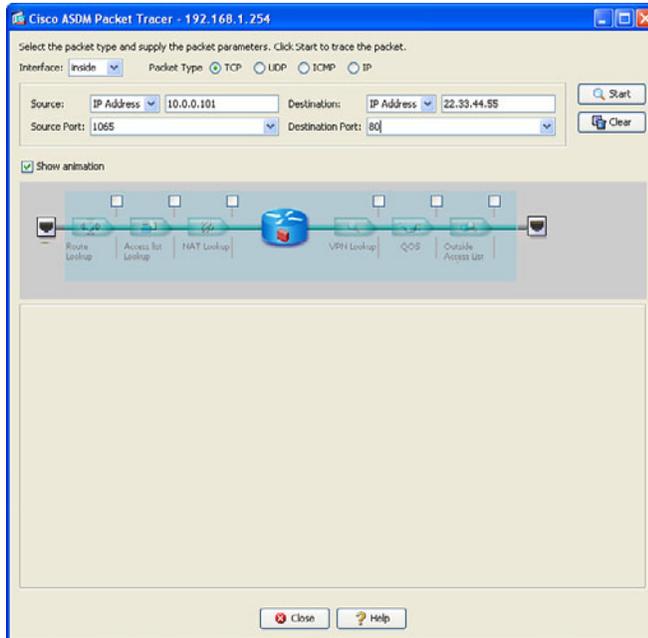


Figure 14-18 *Configuring Input for the Packet Tracer Test*

The literal IP of the source and destination do not have to be valid hosts, as the ASA is only determining, from its current configuration, whether the firewall would allow the packet. In Figure 14-18, we use a random source port, higher than 1023, and the destination well-known TCP port of 80. The IP addresses represent a device from the inside and a server on the outside (based on the default route on the ASA). Also in the Packet Tracer, we identified that the traffic is entering the ASA on the inside interface. When we click **Start**, ASDM sends the commands to the ASA to simulate this packet.

The output in the GUI shows the final result of the packet being either allowed or denied, along with each of the individual checks that the ASA did along the way (each of which can be expanded), as shown in Figure 14-19.

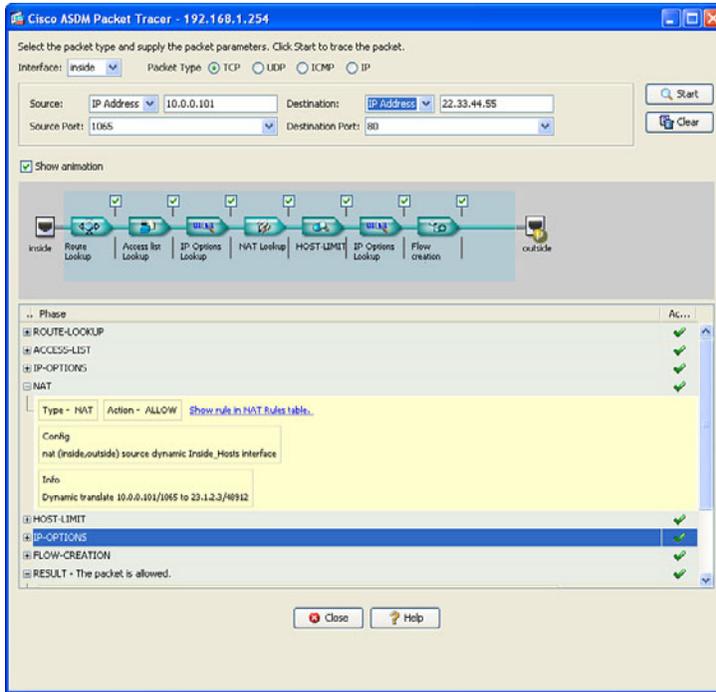


Figure 14-19 Packet Tracer Results Screen

From the output, we can see that the ASA performed several checks, and ultimately would have allowed this packet through the firewall. If there was any issue with ACLs, interfaces that are down, Modular Policy Framework, or any other policies or issues that would cause this ASA not to forward the packet, the result would show as a deny, including which part of the processing on the ASA caused the packet to be denied.

Behind the scenes, the ASA is really processing a CLI command and feeding the information back to ASDM. Example 14-9 shows the actual CLI for this command.

Example 14-9 Using the Packet Tracer Utility at the CLI

```
! Checks to see if a packet, inbound on the inside interface,
! that is coming from host 10.0.0.101 and going to 22.33.44.55, and is
! TCP based and from port 1065 going to 80, and tell us if it would make
! it through the firewall
Keith-IINS-ASA# packet-tracer input inside tcp 10.0.0.101 1065 22.33.44.55
80

! Here are the results of each of the tests it internally checks (based on
! the current, configured and default rules in place)
Phase: 1
Type: ROUTE-LOOKUP
Subtype: input
```

```
Result: ALLOW
Config:
Additional Information:
in 0.0.0.0 0.0.0.0 outside

Phase: 2
Type: ACCESS-LIST
Subtype: log
Result: ALLOW
Config:
access-group inside_access_in in interface inside
access-list inside_access_in extended permit ip any any
Additional Information:

Phase: 3
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 4
Type: NAT
Subtype:
Result: ALLOW
Config:
nat (inside,outside) source dynamic Inside_Hosts interface
Additional Information:
Dynamic translate 10.0.0.101/1065 to 23.1.2.3/5069

Phase: 5
Type: HOST-LIMIT
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 6
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 7
Type: FLOW-CREATION
```

```
Subtype:
Result: ALLOW
Config:
Additional Information:
New flow created with id 1427, packet dispatched to next module

Result:
input-interface: inside
input-status: up
input-line-status: up
output-interface: outside
output-status: up
output-line-status: up
Action: allow

Keith-IINS-ASA#
```

So, from this, we can safely say that the firewall will allow a host on the inside to initiate an HTTP session with a web server on the outside.

Verifying the Policy of No Telnet

Let's run a second test to see whether Telnet is denied (which it should be because of our ACL rule). In Packet Tracer, we input the details the same as before, but change the port to 23, which is the well-known destination port for Telnet, and run the test. Figure 14-20 shows the results.

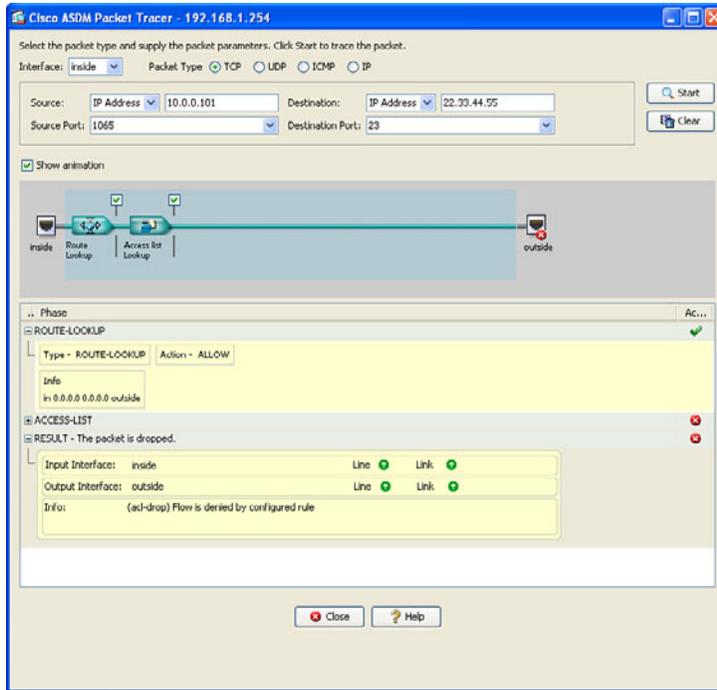


Figure 14-20 *Verifying the ACL Is Preventing Telnet Through the ASA*

This time we see that the initial route lookup took place, but when the ACL was checked, it failed and told us the result. The nice part of this is that it can assist in isolating not only that it did not work, but also the exact component (the reason) that caused it to fail.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 14-3 lists these key topics.



Table 14-3 *Key Topics*

Key Topic Element	Description	Page Number
Text	Meet the ASA family	330
Text	ASA features and services	331
Text	ASA security levels	333
Text	The default flow of traffic	335
Figure 14-2	Default permissions and return traffic due to stateful filtering	336
Text	Packet filtering on the ASA	337
Text	Modular Policy Framework	338
Text	Configuring the interfaces	347
Example 14-4	Using the CLI to implement additional firewall interfaces	352
Text	IP addresses for clients	355
Text	Basic routing to the Internet	356
Text	NAT and PAT	357
Text	Permitting additional access through the firewall	359
Text	Verifying which packets are allowed with ASA Packet Tracer	362

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

stateful filtering, security levels, SVI, Modular Policy Framework, class map, policy map, service policy

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 14-4 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 14-4 *Command Reference*

Command	Description
<code>nameif <i>bubba</i></code>	Assign a name <i>bubba</i> to a Layer 3 interface, done from interface configuration mode
<code>security-level <i>50</i></code>	Assign a security level to an interface, done from interface configuration mode
<code>no shutdown</code>	Bring an interface up out of shutdown mode



This chapter covers the following subjects:

- IPS versus IDS
- Identifying malicious traffic on the network
- Managing signatures
- Monitoring and managing alarms and alerts

Cisco IPS/IDS Fundamentals

Cisco *intrusion detection systems (IDS)* and *intrusion prevention systems (IPS)* are some of many systems used as part of a defense-in-depth approach to protecting the network against malicious traffic. Cisco has many different platforms and options for implementing an IPS/IDS system, but the basic concepts apply across all of these platforms. This chapter focuses on the concepts of IPS/IDS in general, and then the next chapter examines the implementation of IPS/IDS as a software-based IOS solution.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 15-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 15-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
IPS Versus IDS	1–3
Identifying Malicious Traffic on the Network	4–6
Managing Signatures	7–8
Monitoring and Managing Alarms and Alerts	9–10

1. Which method should you implement when it is not acceptable for an attack to reach its intended victim?
 - a. IDS
 - b. IPS
 - c. Out of band
 - d. Hardware appliance

- 2.** A company has hired you to determine whether attacks are happening against the server farm, and they do not want any additional delay added to the network. Which deployment method should be used?

 - a.** Appliance based inline
 - b.** IOS software based inline
 - c.** Appliance based IPS
 - d.** IDS
- 3.** Why does IPS have the ability to prevent an ICMP-based attack from reaching the intended victim?

 - a.** Policy-based routing.
 - b.** TCP resets are used.
 - c.** The IPS is inline with the traffic.
 - d.** The IPS is in promiscuous mode.
- 4.** Which method of IPS uses a baseline of normal network behavior and looks for deviations from that baseline?

 - a.** Reputation-based IPS
 - b.** Policy-based IPS
 - c.** Signature-based IPS
 - d.** Anomaly-based IPS
- 5.** Which type of implementation requires custom signatures to be created by the administrator?

 - a.** Reputation-based IPS
 - b.** Policy-based IPS
 - c.** Engine-based IPS
 - d.** Anomaly-based IPS
- 6.** Which method requires participation in global correlation involving groups outside your own enterprise?

 - a.** Reputation-based IPS
 - b.** Policy-based IPS
 - c.** Signature-based IPS
 - d.** Anomaly-based IPS

7. Which of the micro-engines contains signatures that can only match on a single packet, as opposed to a flow of packets?
 - a. Atomic
 - b. String
 - c. Flood
 - d. Other
8. Which of the following are properties directly associated with a signature? (Choose all that apply.)
 - a. ASR
 - b. SVR
 - c. TVR
 - d. RR
9. Which of the following is not a best practice?
 - a. Assign aggressive IPS responses to specific signatures
 - b. Assign aggressive IPS responses based on the resulting risk rating generated by the attack
 - c. Tune the IPS and revisit the tuning process periodically
 - d. Use correlation within the enterprise and globally, for an improved security posture
10. What is the name of Cisco cloud-based services for IPS correlation?
 - a. SIO
 - b. EBAY
 - c. ISO
 - d. OSI

Foundation Topics

IPS Versus IDS

This section examines the platforms you can use for intrusion detection/prevention and explains the differences between IPS and IDS.

What Sensors Do

A sensor is a device that looks at traffic on the network and then makes a decision based on a set of rules to indicate whether that traffic is okay or whether it is malicious in some way. Because these are systems acting based on configured rules, no single system is ever 100 percent perfect. However, the objective is the same: to reduce the risk of malicious traffic, even though it cannot be completely eliminated.

Difference Between IPS and IDS

You can place a sensor in the network to analyze network traffic in one of two ways. The first option is to put a sensor inline with the traffic, which just means that any traffic going through your network is forced to go in one physical or logical port on the sensor. At the sensor, the traffic is analyzed. Then the sensor forwards out another logical or physical interface if the packet continues its journey toward its destination. If the traffic (while on its short layover at the sensor) is identified as being malicious by the sensor, the sensor (based on the rules configured) could decide that it will not forward the packet any further and drop it. Because the sensor is inline with the network, and because it can drop a packet and deny that packet from ever reaching its final destination (because it might cause harm to that destination), the sensor has in fact just prevented that attack from being carried out. That is the concept behind *intrusion prevention systems (IPS)*. Whenever you hear IPS mentioned, you immediately know that the sensor is inline with the traffic, which makes it possible to prevent the attack from making it further into the network. One negatives about IPS is that because it is inline, if the sensor fails and you do not have an alternate path in your network, the entire network could fail as a result of the sensor having a problem. In addition, a slight additional delay occurs as traffic is analyzed and then forwarded through the inline IPS.

So, then, what is an *intrusion detection system (IDS)*? To understand IDS, let's use the same sensor as we did previously, but instead of placing it inline in the network, we just send copies of the packets that are going through a network to the IDS sensor. When the packets arrive at the sensor in what is called promiscuous mode (because it is willing to look at anything that you send it), it can still analyze the traffic, and it can still generate alerts. However, because the original packet (that we have a copy of) is probably already on its way toward the destination, the sensor all by itself cannot deny the original packet from making its way further into the network. So, we could say that the IDS is *detecting* the attack (hence the term *intrusion detection system*) but is not *preventing* the attack. In a nutshell, that is the difference between IPS, which is inline, and IDS, which operates

in promiscuous mode and is not inline but simply is analyzing copies of packets that were sent over to it (often by a switch configured to send it there). One benefit of IDS is that no delay is added to the original packet, and if the IDS fails, it does not hinder the network throughput because the IDS is not inline with the production network traffic.

Figure 15-1 shows a sensor implementation as an IDS versus an IPS. Also be aware that one physical sensor appliance with multiple interfaces could be configured as an IDS in one part of the network and as an IPS in a different part of the network.

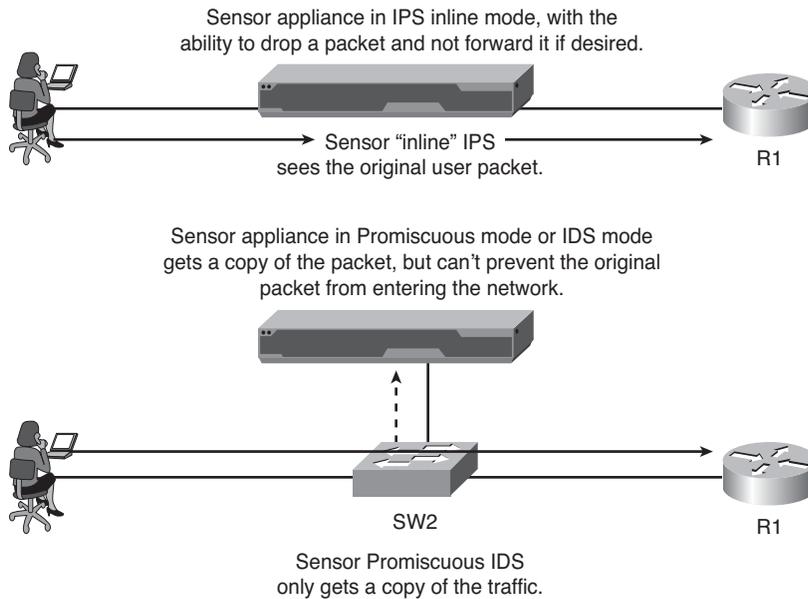


Figure 15-1 *IPS Versus IDS*

The ability to compare and contrast the two implementation methods is important for both certification and the real world. Table 15-2 provides a side-by-side comparison.

Table 15-2 *IDS Versus IPS*

	IDS	IPS
Position in the network flow	Off to the side, the IDS is sent copies of the original packets.	Directly inline with the flow of network traffic and touches every packet on its way through the network.
Also known as	Promiscuous mode, out of band.	Inline mode.



	IDS	IPS
Latency or delay	Does not add delay to the original traffic because it is not inline.	Adds a small amount of delay before forwarding it through the network.
Impact caused by the sensor failing to forward packets	There is no negative impact if the sensor goes down.	If the sensor goes down, traffic that would normally flow through the sensor could be impacted.
Ability to prevent malicious traffic from going into the network	By itself, a promiscuous mode IDS cannot stop the original packet. Options do exist for a sensor in promiscuous mode to request assistance from another device that is inline which may block future packets.	The IPS can drop the packet on its own because it is inline. The IPS can also request assistance from another device to block future packets just as the IDS does.
Normalization ability	Because the IDS does not see the original packet, it cannot manipulate any original inline traffic,	Because the IPS is inline, it can normalize (manipulate or modify) traffic inline based on a current set of rules.

Sensor Platforms



Adding an IPS/IDS sensor to your network makes sense for the enhanced protection it provides, but this might cause a dilemma: For which of the many different areas of your network that need protection do you add a sensor? To help counter that concern, Cisco has several platforms that enable you to implement the IPS/IDS sensor functionality. Options include the following:

- A dedicated IPS appliance, such as the 4200 series
- Software running on the router in versions of IOS that support it
- A module in an IOS router, such as the AIM-IPS or NME-IPS modules
- A module on an ASA firewall in the form of the AIP module for IPS
- A blade that works in a 6500 series multilayer switch

True/False Negatives/Positives

Generally speaking, receiving accurate information (or true information) from a computer system is the desired outcome, including from an IPS/IDS looking for malicious traffic. Obviously, then, information from a computer system that is false (or inaccurate) is not a desired outcome, including from an IPS/IDS.

Positive/Negative Terminology

When working with an IPS/IDS, you will likely come across the following terms:

- False positive
- False negative
- True positive
- True negative



A *false positive* is when the sensor generates an alert about traffic and that traffic is not malicious or important as related to the safety of the network. False positives are easy to identify because alerts are generated, and easily viewed. What is tricky are the false negatives. A *false negative* is when there is malicious traffic on the network, and for whatever reason the IPS/IDS did not trigger an alert, so there is no visual indicator (at least from the IPS/IDS system) that anything negative is going on. In the case of a false negative, you must use some third-party or external system to alert you to the problem at hand, such as syslog messages from a network device.

The true positives and true negatives are fantastic. A *true positive* means that there was malicious traffic and that the sensor saw it and reported on it; if the sensor was an IPS, it may have dropped the malicious traffic based on your current set of rules in place. A *true negative* is also a wonderful thing in that there was normal non-malicious traffic, and the sensor did not generate any type of alert, which is normal sensor behavior regarding non-malicious traffic.

Identifying Malicious Traffic on the Network

Sensors can identify malicious traffic in many different ways. This section examines some of the techniques used by IPS and IDS sensors.

When the sensor is analyzing traffic, it looks for malicious traffic based on the rules that are currently in place on that sensor. There are several different methods that sensors can be configured to use to identify malicious traffic, including the following:

- Signature-based IPS/IDS
- Policy-based IPS/IDS
- Anomaly-based IPS/IDS
- Reputation-based IPS/IDS



Let's take a look at each of these options now.

Signature-Based IPS/IDS

A signature is just a set of rules looking for some specific pattern or characteristic in either a single packet or a stream of packets. A new sensor may have thousands of

default signatures provided by Cisco. Not all the signatures are enabled, but the user (that is you and I, as administrators) can enable, disable, customize, and create new signatures to meet the needs of the current network where the sensor is operating. An example is a known attack, such as a cross-site scripting attack, where an HTTP variable contains a quote or bracket character to terminate some HTML syntax, followed by a `<SCRIPT>` tag. Cisco has written a signature (set of rules) looking for exactly that. Cisco creates additional signatures as new and significant attacks are discovered, which the administrator can implement by updating the signatures on the sensor. Signature-based IPS/IDS is the most significant method used on sensors today. For tuning purposes, if a default signature keeps triggering based on traffic that is normal for your network, that is considered a false positive for your specific network, and you could tune the sensor by either disabling that signature or setting the filter so that that signature does not generate an alert when it sees a match from specific IP addresses.

Policy-Based IPS/IDS

This type of traffic matching can be implemented based on the policy for your network. For example, if your company has a policy that states that no Telnet traffic should be used (for security reasons) on specific areas of your network, you can create a custom rule that states that if TCP traffic is seen destined to port 23 (which is the well-known port for Telnet) the IPS can generate an alert and drop the packet. If this is configured as IDS, it could simply generate an alert (but cannot drop the packet on its own because IDS is in promiscuous mode, and not inline). The sneaky part of this one is that policy-based IPS/IDS is most likely implemented by creating a custom signature that causes this policy to be enforced (which would also make this signature based).

Anomaly-Based IPS/IDS

An example of anomaly-based IPS/IDS is creating a baseline of how many TCP sender requests are generated on average each minute that do not get a response. This is an example of a half-formed session. If a system creates a baseline of this (and for this discussion, let's pretend the baseline is an average of 30 half-formed sessions per minute), and then notices the half-formed sessions have increased to more than 100 per minute, and then acts based on that and generates an alert or begins to deny packets, this is an example of anomaly-based IPS/IDS. The Cisco IPS/IDS appliances have this ability (called *anomaly detection*), and it is used to identify worms that may be propagating through the network. This feature is on by default on the 4200 series appliances.

Reputation-Based IPS/IDS

If some type of global attack is creeping across the networks of the world but has not hit your network yet, wouldn't it be nice to know about it so that you can filter that traffic before it enters your network? The answer is yes, obviously. Reputation-based IPS collects input from systems all over the planet that are participating in global correlation; so what other sensors have learned collectively, your local sensor can use locally. Reputation-based IPS/IDS may include descriptors such as blocks of IP addresses, URLs,

DNS domains, and so on as indicators of the sources for these attacks. Global correlation services are managed by Cisco as a cloud service.

Table 15-3 describes the advantages and disadvantages of these four categories of IPS/IDS detection technologies.

Table 15-3 *IPS/IDS Method Advantages and Disadvantages*

	Advantages	Disadvantages
Signature based	Easy to configure, simple to implement.	Does not detect attacks outside of the rules. May need to disable signatures that are creating false positives. Signatures must be updated periodically to be current.
Policy based	Simple and reliable, very customizable, only allows policy-based traffic that could deny unknown attacks, which by default are outside of the policy being allowed.	Policy must be manually created. Implementation of the policy is only as good as the signatures you manually create.
Anomaly based	Self-configuring baselines, detect worms based on anomalies, even if specific signatures have not been created yet for that type of traffic.	Difficult to accurately profile extremely large networks. May cause false positives based on significant changes in valid network traffic.
Reputation based	Leverages enterprise and global correlation, providing information based on the experience of other systems. Early-warning system.	Requires timely updates, and requires participation in the correlation process.



When Sensors Detect Malicious Traffic

Based on how the sensor is configured, and which mode it is in (IPS or IDS), the sensor can implement the actions described in Table 15-4.

Table 15-4 *Possible Sensor Responses to Detected Attacks*

Response	What It Means
Deny attacker inline	Available only if the sensor is configured as an IPS. This action denies packets from the source IP address of the attacker for a configurable duration of time, after which the deny action can be dynamically removed.

Response	What It Means
Deny connection inline	Available only if the sensor is configured as an IPS. This action terminates the packet that triggered the action, and future packets on the same TCP flow. The attacker could open up a new TCP session (using different port numbers), which could still be permitted through the inline IPS.
Deny packet inline	Available only if the sensor is configured as an IPS. All the “deny” options only apply to IPS mode. Deny packet terminates the packet that triggered the alert.
Log attacker packets	This action begins to log future packets based on the attacker’s source IP address. This is done usually for a short duration, such as 30 seconds, after the initial alert. Log files are stored in a format that is readable by most protocol analyzers
Log victim packets	This logging action begins to log all IP packets that have a destination IP address of the victim (the destination address from the packet or packets that triggered the alert).
Log pair packets	This logging action begins to log IP packets if the source and destination addresses indicate that the packets from the source IP address that triggered the alert and the destination address match the destination address of the packet that triggered the alert. In essence, it is future packets between the attacker and the victim (the attacked device address).
Produce alert	An alert is the basic mechanism that is used by the IDS/IPS to identify that an event has occurred, such as a signature match indicating malicious traffic. This is the default behavior for most of the signatures.
Produce verbose alert	Produce verbose alert has the same behavior as produce alert, with the added bonus that it includes a copy of the entire packet that triggered the alert. If both produce alert and produce verbose alert are enabled, it will still only generate a single alert and will include a copy of the triggering packet.
Request block connection	Some sensor devices can ask for help to block the attacker’s traffic at some point in the network. The device that connects to implement the blocking is called a blocking device, and could be an IOS router, a switch that supports <i>VLAN access control lists (VACL)</i> , or an <i>Adaptive Security Appliance (ASA)</i> Firewall. This action causes the sensor to request a blocking device to block based on the source IP address of the attacker, the destination IP address of the victim, and the ports involved in the packet that triggered the alert. The difference between this option and the one that follows is that request blocked connection gives an opportunity for the attacker to send traffic on different ports or different destination IPs and still allows connectivity for new sessions.
Request block host	This causes the sensor to requests its blocking devices (see the preceding paragraph) to implement blocks based on the source IP address of the attacker regardless of the ports in use or destination IP addresses for future packets.

Response	What It Means
Request SNMP trap	This generates an <i>Simple Network Management Protocol (SNMP)</i> trap message that is sent to the configured management address for SNMP.
Reset TCP connection	This causes a sensor to send a proxy TCP reset to the attacker, with the intention of fooling the attacker into believing it is the victim sending the TCP reset. This action has an effect only on TCP-based traffic.

Controlling Which Actions the Sensors Should Take

Many years ago, as the number of signatures kept increasing, it became very tedious to manage individual actions on each and every signature. A solution to this is allow all the IPS/IDS sensors (after generating an alert), to consider how significant the risk is (related to that alert), and if the risk is high enough, then let the sensor go ahead and take appropriate countermeasure actions.

This is implemented using a calculated result called a *risk rating*. The maximum value for risk rating is 100. As the administrator, you can choose which countermeasure to take based on the risk rating that triggers an alert. An important thing to know are the factors that make up the final risk rating. There are three primary influencers of the final risk rating value. The first is the accuracy of the signature (meaning how likely it is to make a mistake), and this accuracy rating is known as the *signature fidelity rating (SFR)*, and it is configured as a property of a signature. The second major component that goes into calculating the risk rating is the *attack severity rating (ASR)* of the signature that triggered the alert. This property is also configured as part of the signature. The third major component that is used to calculate the risk rating is in the hands and control of the administrator of the sensor. It is called the *target value rating (TVR)*. To do this, you tell the sensor which of your destination IP addresses or subnets are the most critical. When attacks are seen going to these IP addresses, the end risk rating is higher than if that same attack is going to a less-important device. The TVR is not a property of any specific signature, but rather is a configured general parameter in the IPS. Some additional minor factors go into the risk rating, and Table 15-5 provides a summary of most relevant factors that influence the risk rating.

Table 15-5 *Risk Rating (RR) Calculation Factors*

Factor That Influences Risk Rating	Description
Target value rating (TVR)	The value that you as an administrator have assigned to specific destination IP addresses or subnets where the critical servers/devices live.
Signature fidelity rating (SFR)	The accuracy of the signature as determined by the person who created that signature.



Factor That Influences Risk Rating	Description
Attack severity rating (ASR)	How critical the attack is as determined by the person who created that signature.
Attack relevancy (AR)	This is a minor contributor to the risk rating. A signature match that is destined to a host where the attack is relevant, such as a Windows server-based attack, which is going to the destination address of a known Windows server, is considered a relevant attack, and the risk rating increases slightly as a result.
Global correlation	If the sensor is participating in global correlation and receives information about specific source addresses that are being used to implement large-scale attacks, attacks coming from the source IP addresses are also given a slightly increased risk rating value.

Implementing Actions Based on the Risk Rating

Although it is true that you can implement actions as properties of individual signatures, it makes the most sense to configure actions based on the risk rating that is created as a result of the signature matches. For example, you can specify severe countermeasures if a risk rating is generated that is 90 or higher. (The max is 100, and if the risk rating calculation ends up with a value larger than 100, it rounds it down to that number.) A risk rating of 50 or lower may simply be configured to generate an alert but not cause a severe countermeasure, such as deny attacker, to be implemented. All of this is under administrator control.

IPv6 and IPS

There is a new protocol (actually an old protocol, more than a decade old) that is making its way more and more into our production networks: IPv6. As a result, more and more signatures and rules related to IPv6 are being built in to the IPS/IDS sensors.

Circumventing an IPS/IDS

An attacker has an objective, and it is likely that he does not want to be stopped or seen. If you have IPS/IDS systems in place, an attacker may try to evade the IPS/IDS. Table 15-6 describes evasion methods an attacker may try to use and Cisco options to counter these evasion techniques.

Table 15-6 *IPS/IDS Evasion Techniques*

Evasion Method	Description	Cisco Anti-Evasion Techniques
Traffic fragmentation	The attacker splits malicious traffic into multiple parts with the intent that any detection system will not see the attack for what it really is.	Complete session reassembly so that the IPS/IDS can see the big picture.
Traffic substitution and insertion	The attacker substitutes characters in the data using different formats that have the same final meaning. An example is Unicode strings, which an end station could interpret but perhaps a lesser IPS/IDS might not.	Data normalization and de-obfuscation techniques. Cisco's implementation is looking for Unicode, case sensitivity, substitution of spaces with tabs, and other similar anti-evasion techniques.
Protocol level misinterpretation	An attacker may attempt to cause a sensor to misinterpret the end-to-end meaning of a network protocol and so perhaps not catch an attack in progress.	IP Time-To-Live (TTL) analysis, TCP checksum validation.
Timing attacks	By sending packets at a rate low enough so as to not trigger a signature (for example, a flood signature that triggers at 1000 packets per second, and the attacker sending packets at 900 packets per second).	Configurable intervals and use of third-party correlation.
Encryption and tunneling	Encrypted payloads are called encrypted for a reason. If an IPS/IDS sees only encrypted traffic, the attacker can build a Secure Sockets Layer (SSL) or IPsec session between himself and the victim and could then send private data over that virtual private network (VPN).	If traffic is encrypted and passing through the sensor as encrypted data, the encrypted payload cannot be inspected. For generic routing encapsulation (GRE) tunnels, there is support for inspection if the data is not encrypted.
Resource exhaustion	If thousands of alerts are being generated by distractor attacks, an attacker may just be trying to disguise the single attack that they are trying to accomplish. The resource exhaustion could be overwhelming the sensor and overwhelming the administration team who has to view the events.	Dynamic and configurable event summarization. Here is an example: 20,000 devices are all under the control of the attacker. All those devices begin to send the same attack. The sensor summarizes those by showing a few of the attacks as alerts, and then summaries at regular intervals that indicate the attack is still in play and how many thousands of times it occurred over the last interval. This is much better than trying to wade through thousands of individual alerts.

Managing Signatures

The most prominent way to identify malicious traffic in the Cisco IPS/IDS systems is signature-based matching. This section covers how signatures are manipulated and managed to meet a specific network requirement.

Dealing with signatures is one of the tasks that you will perform as you tune, implement, and maintain a sensor appliance (or an IOS router running IPS in software). Cisco organizes their signatures into groups that have similar characteristics. For each of their groups, a signature micro-engine is used to govern that set of signatures. When a packet comes through the sensor, all the signatures in a specific group or micro-engine are compared simultaneously to the packet looking for matches. If you modify a signature, the micro-engine responsible for that signature is responsible for updating and implementing the changes behind the scenes. There are several signature micro-engines, and even inside of the micro-engine there are further subdivisions for the organization of the signatures. Fortunately, as an administrator, we do not really have to worry too much about the specific micro-engines, but for certification you definitely want to be aware that they exist. Table 15-7 describes a few of the micro-engines.

Table 15-7 *Micro-Engines (Groupings of Signatures)*



Signature Micro-Engine	Signatures in This Grouping
Atomic	Signatures that can match on a single packet, as compared to a string of packets
Service	Signatures that examine application layer services, regardless of the operating system
String or Multistring	Supports flexible pattern matching, and can be identified in a single packet or group of packets, such as a session
Other	Miscellaneous signatures that may not specifically fit into other categories

Note that this is only a subset of the micro-engines and is presented here to introduce the concept.

Signature or Severity Levels

One of the properties of each signature is signature severity (also called *attack severity rating [ASR]*). This is a rating between 0 and 100 that indicates (in the eyes of the individual who created the signature) how severe the attack is that the signature is looking for. We discussed earlier the three primary factors that go into calculating the risk rating, and this severity level, which is a property of the signature, is one of those three elements. Instead of having to set a numeric value for the severity, the interface for IPS/IDS prompts us for one of four levels. Those four options are informational, low, medium,

or high. The higher the severity, the greater the number in the background that goes into the calculation for this factor into the risk rating.

The other property that is part of the signature and is a significant portion of the overall risk rating calculation is the signature fidelity rating, and this value literally is a numeric value between 0 and 100 set by the person who created the signature. Both the signature fidelity and the signature severity can be tuned by the administrator regardless of what the initial value was set to by default.

Monitoring and Managing Alarms and Alerts

Cisco sensors can identify a wide range of attacks. Being aware that the attacks are happening is a big part of the IPS/IDS solution, and this section examines the options for working with these sensor-generated alarms and alerts.

Once your sensor is in place and doing its job, if it generates an alert you need to be able to see that somehow. As the sensor generates alerts, you can feed those alerts real time to a monitoring system, which can show you the information in beautiful color-coded formats, or you could go to the database of stored alerts, extract them, and analyze them that way, as well. Three main protocols are used in delivering alerts. They are *Security Device Event Exchange (SDEE)*, syslog, and SNMP. You can use one or all of these methods to get the alerts off of the sensor and sent to the device that you choose to use to view what is happening in the world of alerts.

SDEE is used for real-time delivery of alerts, and is the most secure method for delivering alerts. These can be sent to an application running on a server. One example is the software named *IPS Manager Express (IME)*, which can run on a workstation and be a central point of event viewing that can support up to 10 sensors simultaneously. Other management consoles, such as *Cisco Security Manager (CSM)*, can also be used and can support greater numbers of simultaneous sensors. The upper limit of what is reasonable is about 25 sensors reporting to a single manager machine.

Security Intelligence

One thing I've noticed is that the more I work on a specific network, the more familiar I am with its behavior. This also is true with an IPS sensor. The sensor with multiple interfaces can operate in many different parts of the network at the same time, and the more visibility you have to those areas of your network, the more intelligence or information you will receive about what is going on in those parts of the network. If you have multiple sensors in your enterprise environment, you can correlate all the events on a management station to get a better overall picture of what is happening and where the attacks are. So, in short, the more sensors you have reporting, the better the information is going to be about the attacks that are going on.

If we take this one step further and involve multiple organizations who are all reporting threats that are on global networks, such as the Internet, and we can correlate those events, we can use that information to defend our network borders against an attack that might not have reached us yet. In essence, a single sensor can give this device intelligence

about that area of the network. Multiple sensors can give this enterprise intelligence about all the networks in your enterprise. The final step is global intelligence, where multiple organizations that are running sensors participate in global correlation and share information about external threats that may affect other companies, as well. With global correlation, we can increase the risk rating for specific attacks if they are from source addresses that we identified as suspect in information learned from external sensors through the global correlation process. Global correlation is available on the sensor appliances, but does not have to be enabled.

Cisco offers the *Security Intelligence Operations (SIO)* service, which facilitates global threat information, reputation-based services, and sophisticated analysis for the benefit of Cisco security devices to better protect the networks they serve.

IPS/IDS Best Practices



Here are some of the recommended IPS/IDS best practices:

- Implement an IPS so that you can analyze traffic going to your critical servers and other mission-critical devices.
- If you cannot afford dedicated appliances, use modules or IOS software-based IPS/IDS. Appliances have better performance than modules, and modules have better performance than adding on the feature to existing IOS routers in software only.
- Take advantage of global correlation to improve your resistance against attacks that may be moving toward your organization, and use correlation internally across all your sensors to get the best visibility of the network attacks that are being attempted.
- Use a risk-based approach, where countermeasures occur based on the calculated risk rating as opposed to manually assigning countermeasures to individual signatures.
- Use automated signature updates when possible instead of manually installing updates; this will assist in keeping the signatures current.
- Continue to tune the IPS/IDS infrastructure as traffic flows and network devices and topologies change. IPS tuning is mostly done on a brand-new implementation, but is never truly 100 percent complete.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 15-8 lists these key topics.

Table 15-8 *Key Topics*

Key Topic Element	Description	Page Number
Table 15-2	IDS versus IPS	375
Text	Sensor platforms	376
List	Positive/negative terminology	377
List	The core methods for matching malicious traffic	377
Table 15-3	IPS methods advantages and disadvantages	379
Table 15-5	Risk rating calculation factors	381
Table 15-6	IPS evasion techniques	383
Table 15-7	Micro-engines (groupings of signatures)	384
List	IPS/IDS best practices	386



Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

IPS, IDS, risk rating, attack severity rating, target value rating, signature fidelity rating



This chapter covers the following subjects:

- Understanding and installing an IOS-based IPS
- Working with signatures in an IOS-based IPS
- Managing and monitoring IPS alarms

Implementing IOS-Based IPS

An IOS-based IPS is a software-implemented *intrusion prevention system (IPS)*. Many companies purchase appliances that are dedicated to IPS. For companies with limited budgets, however, an IPS implemented on the perimeter router can provide attack visibility and help thwart attacks that it recognizes. A company may have appliances at their corporate headquarters and can still leverage the IOS-based IPS at branch offices.

This chapter focuses on taking a router that has an IOS feature set that can support IOS-based IPS and configuring it from scratch. After configuring it, we put it to work by generating alerts and take a look at those alerts.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 16-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 16-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Understanding and Installing an IOS-Based IPS	1–4
Working with Signatures in an IOS-Based IPS	5–6
Managing and Monitoring IPS Alarms	7–8

1. Because of how a router operates, which IPS/IDS mode does it operate in?
 - a. Promiscuous
 - b. Out-of-band
 - c. IPS
 - d. IDS

- 2.** Which of the following enable you to manage an IOS router's IPS configuration? (Choose all that apply.)
 - a.** CLI
 - b.** CSM
 - c.** CCP
 - d.** IME
- 3.** Why is the public key for Cisco required as part of the IPS installation?
 - a.** It is used to verify the routers self-signed certificate.
 - b.** It is used to log in to CCO for automated signature updates.
 - c.** It is used to validate the signature that Cisco has placed on the signature package.
 - d.** No public key is used as part of an IOS-based IPS.
- 4.** Where are the specific signature-related configuration files kept related to an IOS-based IPS?
 - a.** RADIUS server
 - b.** TACACS+ server
 - c.** On Flash or FTP or other reachable destination that the router is configured to use for that purpose
 - d.** NVRAM
- 5.** Which of the following are examples of tuning a signature? (Choose all that apply.)
 - a.** Changing the default severity level
 - b.** Enabling it if it was disabled by default
 - c.** Disabling it if it was enabled by default
 - d.** Changing the default action
- 6.** Which of the following is used to indicate that a signature is enabled? (Choose all that apply.)
 - a.** A green icon in the Enabled column
 - b.** A check mark in the Enabled column
 - c.** A missing check mark in the Disabled column
 - d.** A missing checkmark in the Unretired column

- 7.** Which of the following are methods that enable you to see alerts that have been triggered by IOS based IPS?
 - a.** CLI
 - b.** CCP
 - c.** IME
 - d.** CSM

- 8.** Why is it considered a best practice to avoid compiling, enabling, and running all available signatures? (Choose all that apply.)
 - a.** CPU utilization
 - b.** Memory utilization
 - c.** The size of NVRAM
 - d.** Not a best practice

Foundation Topics

Understanding and Installing an IOS-Based IPS

This section discusses which features of Cisco IPS are included in the IOS software implementation of IPS.

What Can IOS IPS Do?

The router operates in IPS mode because it is already inline with the customer's traffic; it is a router after all. You do not need to redesign the topology of the network to put this IPS in place. You just need to have a license that supports the feature, and configure the software. IOS-based IPS supports the following detection technologies:

- Profile based
- Signature based
- Protocol analysis based

One option that is missing from this list, that a sensor appliance does have but the IOS IPS does not, is an anomaly-based detection.

The IOS IPS provides the following benefits:

- The ability to do dynamic updates of signatures
- Integrates easily into an existing network
- Compatible to work alongside of other security features, such as *Zone-Based Firewalls (ZBF)*, *virtual private network (VPN)* termination, *access control lists (ACL)*, *authentication, authorization, and accounting (AAA)*, and many others on the same router as long as there is enough memory and CPU to support all the features
- Can be managed by *Cisco Configuration Professional (CCP)*, *IPS Manager Express (IME)* and *Cisco Security Manager (CSM)*, and via the *command-line interface (CLI)*
- Supports attack signatures from the same signature database that is used by the IPS appliances

Table 16-2 lists several of the features commonly found in Cisco IPS products and which are included in the IOS IPS implementation.

Table 16-2 *IOS IPS Features*

Cisco IOS IPS Signature Features	Description
Regular expression string pattern matching	Enables the creation of string patterns using variables. An example of a regular expression is <i>hotcold</i> , which is part of the signature that would look for a match on the word <i>hot</i> or <i>cold</i> . Using regular expressions can allow a single string to be used to match several possible combinations of that string inside of a packet.
Response actions	This enables the sensor to take action in response to a triggered event, such as denying a packet, creating an alert, resetting the TCP connection, or denying the attacker's future packets for a period of time.
Alarm summarization	This helps prevent resource exhaustion by summarizing events that are all the same or at least the same signature. Under heavy attack, a summary may show the attack happened 15,000 times as opposed to producing 15,000 individual alerts. Summarization is tunable on both the IOS IPS and the appliances.
Threshold configuration	Threshold configuration identifies thresholds, which if exceeded may trigger events. For example, a specific string of text can be identified in a signature. That same signature can specify that an alert will be generated only after that string of text has been seen five times within a 60-second window.
Anti-evasive techniques	Similar to the appliances, the intelligence in the IOS IPS is designed to correctly interpret the actual data regardless if it is fragmented or using a combination of character sets, such as Unicode.
Risk ratings	A calculated number between 0–100 associated with an alert. The higher the number, the more risk is presumed. Identifying your critical servers/hosts enables the system to generate higher risk ratings when signatures to those devices are matched. Those higher risk ratings can then trigger countermeasures against the attack(s).

Installing the IOS IPS Feature

The first step to getting IOS-based IPS working is to make sure that you are running a version of IOS that supports it. You can confirm this by using the Feature Navigator at Cisco.com (<http://www.cisco.com/go/fn>). From there, you can specify the type of hardware that you are running and the feature you are looking for; it can then confirm which versions of the IOS support it.

Another important task before installing the IOS-based IPS is obtaining the actual signature files from Cisco for your router. These come in a couple of different formats. One is a ZIP format and the other has an extension of .pkg. The ZIP file is used by CCP if you are going to have the file locally on your computer and move it from your computer to the router via CCP. The PKG file is used at the command line of the router for installing or updating the signatures, and is also the format that is used if you told the

router to obtain the file from a web or from an FTP server. One of the nice features about IOS-based IPS is that the router can be configured to automatically obtain updates directly from Cisco.com after the installation is in place. (This must be configured, and a valid CCO account that has the access rights to the update files is required to be configured into the router.) The benefit of that is that you do not need to worry about manually keeping your updates current. The negative side of that is that when a Cisco router gets a new signature update package it has to compile all the updates in their respective signature micro-engines. This takes excessive CPU and really should be scheduled in a maintenance window where there is a possibility of very slow performance while the compile happens. Avoid the surprise of learning this fact on the production network.

We are going to install the IPS via CCP; but have no fear, I also show you the configuration that we can use straight from the command line. Both ways work.

Getting to the IPS Wizard

You start by going to CCP and navigating to **Configure > Security > Intrusion Prevention**, as shown in Figure 16-1. Note that based on the platform and version of CCP, the navigation may be slightly different, and may appear as **Configure > Security > Advanced Security > Intrusion Prevention** (with the extra step labeled Advanced).

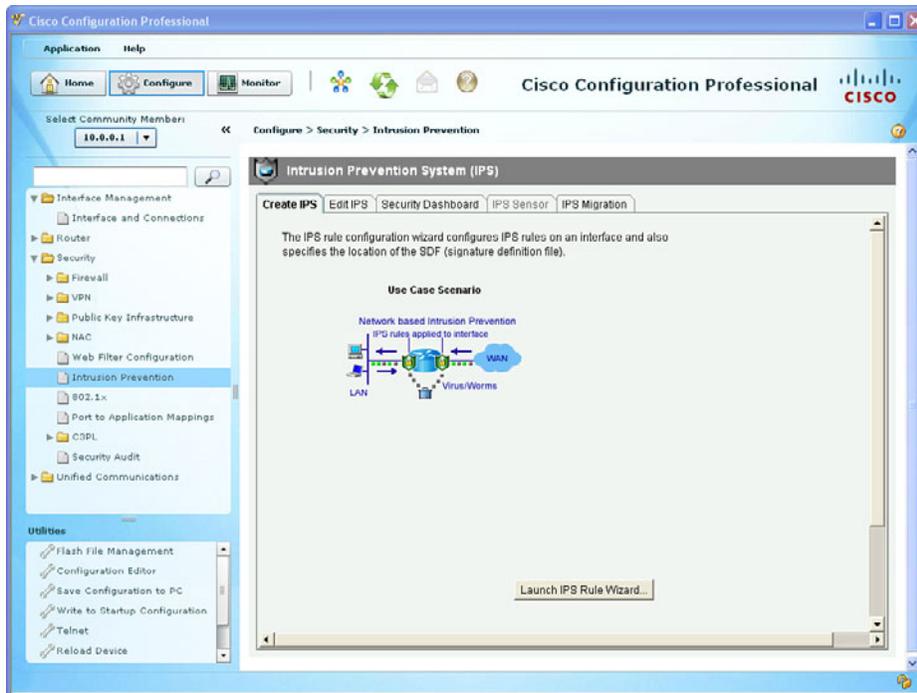


Figure 16-1 Navigating to the IOS IPS Configuration Screen in CCP

From here, click the **Launch IPS Rule Wizard** button to begin the setup. After you click the button, if *IPS Security Device Event Exchange (SDEE)*, the protocol used to deliver

alert information, has not yet been enabled, you are prompted to enable it, as shown in Figure 16-2.



Figure 16-2 *Enabling SDEE*

Click OK to enable the SDEE function. A pop-up window announces that CCP is going to open up a subscription to the router. A subscription is how an IPS managing devices requests any new updated alerts from that IPS device. CCP can receive these alerts after the configuration is done, this is why SDEE is set up along with the subscription to the router. Figure 16-3 shows an example of this.



Figure 16-3 *CCP Opening a Subscription to Obtain Events from the Router*

When you click OK to confirm the pop-up window, the IPS policy welcome page appears, as shown in Figure 16-4.

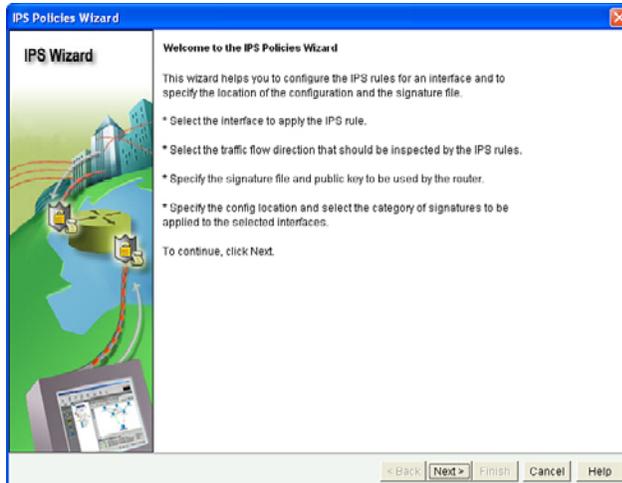


Figure 16-4 *Welcome Page for the IPS Policies Wizard*

When you click **Next** to continue, the wizard asks which interfaces you want to apply the IPS policy to. The interfaces that are currently on the router are presented, and you check the check box to indicate which interfaces and which directions you want the inspection of traffic (the analysis of that traffic to see whether it matches the signature)

to happen on. Normally, the inspection should happen inbound on the interface or interfaces connected to the untrusted networks, or less-trusted networks (such as the Internet). Figure 16-5 shows an example of selecting the interface or interfaces and the direction for the inspection of traffic.

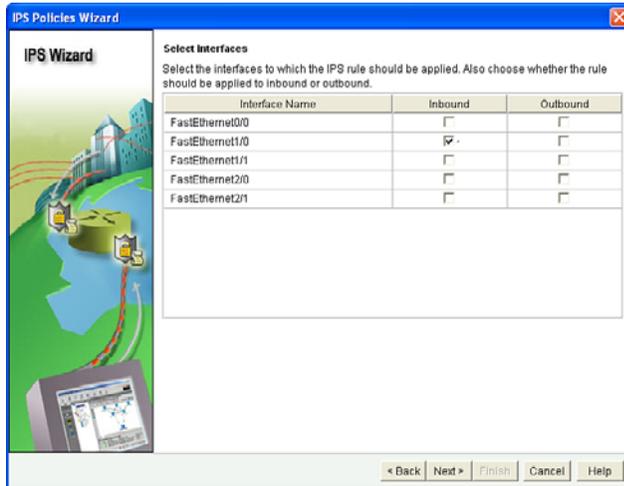


Figure 16-5 *Selecting the Interfaces and Direction for IPS Inspection of Traffic*

After you click **Next**, you are presented with a dialog box asking for the signature file. This is the ZIP file if it is stored on your PC, or the PKG file if you are going to have the router get it from an FTP or other type of server. If the signature file is already sitting on the flash of the router, you can indicate that, as well, from this dialog box. In this demonstration, I chose the option of specifying a location where the signature file is, by choosing that radio button, and then on the **Browse** button I indicated that the file is on an FTP server and have included the full URL to the file on a server, as shown in Figure 16-6.

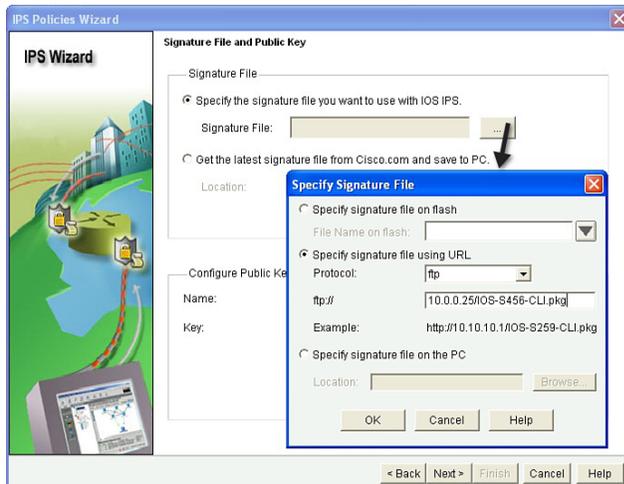


Figure 16-6 *Defining Where the Signature File Is*

After clicking **OK**, you then need to put in the public key that can be used to verify the signature of Cisco. Here's why: If an attacker distributes a bogus set of signatures and you unknowingly installed them, this could create a security hole in your defenses. To protect against that, Cisco signs the signature files, using its private key to do so, and the only way to verify that signature is to have the public key of the entity that signed it. That is why we are installing the public key from Cisco on the router that needs to verify the signature on the IPS signature file. The public key information can also be downloaded from Cisco.com. The download for this is in the same area as the download for the IPS signature files. Figure 16-7 shows an example of providing this information.

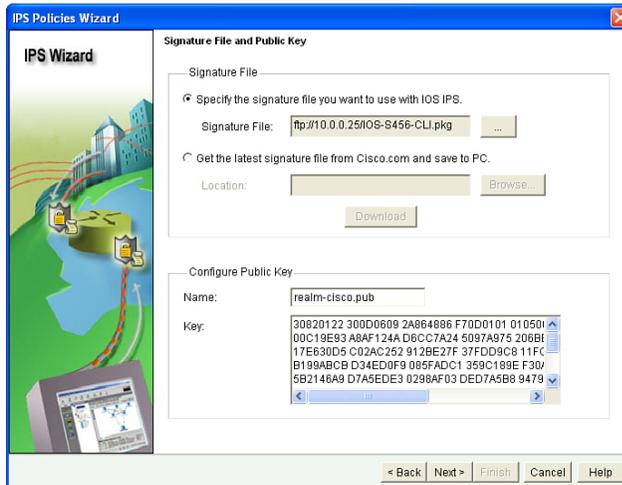


Figure 16-7 Adding Cisco Public Key Information

After you click **Next**, you are prompted to specify the location of the configuration files that the router will use to maintain any configurations related to signatures. This is an exception to how the router normally works. Normally, everything goes into the running configuration. With IPS, the running configuration contains information about which interfaces are involved in IPS and where the configuration files are, but the actual signature configuration files are maintained in a file system that could be on the flash of the router or on an external server such as an FTP server. For this demonstration, I store the IPS signature configuration file information on an FTP server, which can be selected by clicking the **Browse** button and then clicking the radio button specifying that you are going to use a URL, selecting the type of file server based on the drop-down menu, and then specifying the full URL to the folder where these files will be saved. Figure 16-8 shows an example of this.



Figure 16-8 Specifying the Location for the IPS Signature Configuration Files

After clicking **OK** to continue, you are also prompted to specify a category of signatures. It really boils down to this: The Advanced category has more than 1000 signatures enabled, and the Basic category has fewer than 500 signatures enabled. If your router has plenty of memory and CPU resources available, you could use the Advanced category (with more signature coverage, and it would use more resources than the Basic category). In this demonstration, I chose the Basic category, which still supplies hundreds of signatures that are available.

When you click the **Next** button, CCP gives you a summary screen of the commands that are about to be deployed to the router, as shown in Figure 16-9.

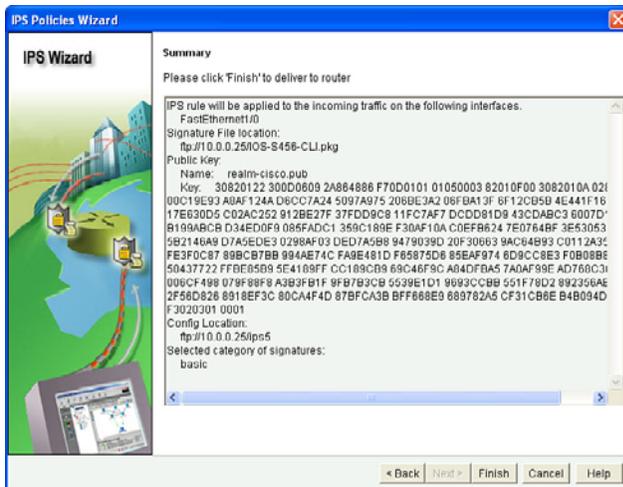


Figure 16-9 Summary Screen for the IPS Wizard

Be aware that as this is deployed there will be a very heavy hit on CPU resources while the router compiles all the signatures in the micro-engines that are associated with the Basic category. This could be up to 5 minutes on a low-end router, with CPU utilization near 100 percent during that time. To finish the wizard, you click the **Finish** button and any other confirmation windows that appear.

At the command line of the router, we can see the activity of the compiling of the signatures specific to each of the micro-engines. This also provides a deeper insight into what many of the micro-engines are in the IOS-based IPS. Example 16-1 shows such output.

Example 16-1 *Output from the Console While Signatures Are Compiled*



```

R1#
Applying Category configuration to signatures ...
%IPS-6-ENGINE_BUILDS_STARTED: 18:31:20 UTC Feb 21 2012

! Notice the Micro Engine names, such as atomic-ip, service, etc.
%IPS-6-ENGINE_BUILDING: atomic-ip - 3 signatures - 1 of 13 engines
%IPS-6-ENGINE_READY: atomic-ip - build time 12 ms - packets for this engine
will be scanned
%IPS-6-ALL_ENGINE_BUILDS_COMPLETE: elapsed time 12 ms
R1#
%IPS-6-ENGINE_BUILDS_STARTED: 18:35:55 UTC Feb 21 2012
%IPS-6-ENGINE_BUILDING: multi-string - 40 signatures - 1 of 13 engines
%IPS-6-ENGINE_READY: multi-string - build time 120 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: service-http - 801 signatures - 2 of 13 engines
%IPS-6-ENGINE_READY: service-http - build time 2044 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: string-tcp - 2058 signatures - 3 of 13 engines
%IPS-6-ENGINE_READY: string-tcp - build time 7108 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: string-udp - 79 signatures - 4 of 13 engines
%IPS-6-ENGINE_READY: string-udp - build time 92 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: state - 37 signatures - 5 of 13 engines
%IPS-6-ENGINE_BUILDS_STARTED: 18:38:13 UTC Feb 21 2012
%IPS-6-ENGINE_BUILDING: multi-string - 40 signatures - 1 of 13 engines
%IPS-6-ENGINE_READY: multi-string - build time 56 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: service-http - 801 signatures - 2 of 13 engines
%IPS-6-ENGINE_READY: service-http - build time 452 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: string-tcp - 2058 signatures - 3 of 13 engines
%IPS-6-ENGINE_READY: string-tcp - build time 780 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: string-udp - 79 signatures - 4 of 13 engines
%IPS-6-ENGINE_READY: string-udp - build time 24 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: state - 37 signatures - 5 of 13 engines
%IPS-6-ENGINE_READY: state - build time 16 ms - packets for this engine
will be scanned
%IPS-6-ENGINE_BUILDING: atomic-ip - 373 signatures - 6 of 13 engines
%IPS-6-ENGINE_READY: atomic-ip - build time 824
R1# ms - packets for this engine will be scanned

```

```

%IPS-6-ENGINE_BUILDING: string-icmp - 3 signatures - 7 of 13 engines
%IPS-6-ENGINE_READY: string-icmp - build time 0 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: service-ftp - 3 signatures - 8 of 13 engines
%IPS-6-ENGINE_READY: service-ftp - build time 0 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: service-rpc - 76 signatures - 9 of 13 engines
%IPS-6-ENGINE_READY: service-rpc - build time 28 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: service-dns - 39 signatures - 10 of 13 engines
%IPS-6-ENGINE_READY: service-dns - build time 16 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: normalizer - 9 signatures - 11 of 13 engines
%IPS-6-ENGINE_READY: normalizer - build time 0 ms - packets for this engine
will be scanned
%IPS-6-ENGINE_BUILDING: service-smb-advanced - 49 signatures - 12 of 13
engines

! Note that CPU is extremely high over the last several minutes while the
! compiling was taking place

R1# show process cpu sorted | include seconds
CPU utilization for five seconds: 80%/100%; one minute: 85%; five minutes: 73%

```

Working with Signatures in an IOS-Based IPS

In this section, we enable and tune a signature and cause it to trigger using CCP.

So, now that we have IPS configured, and it is looking at traffic coming inbound on its Fast Ethernet 1/0 interface, how do we manage or tune it? You can use several tools to manage and configure the IPS on this router. You can use CCP, the CLI, CSM, or IME (as mentioned previously) as tools to manage IOS-based IPS. In this case, we use CCP and the CLI for configuring and verifying.

To view/modify the signatures, navigate to **Configure > Security > Intrusion Prevention** and click the **Edit IPS** tab. Then, click the **Signatures** option to view all of the signatures. (Just as before, the navigation may vary slightly and have the extra keyword **Advanced** to go to the IPS Management section.) Figure 16-10 shows the signatures.

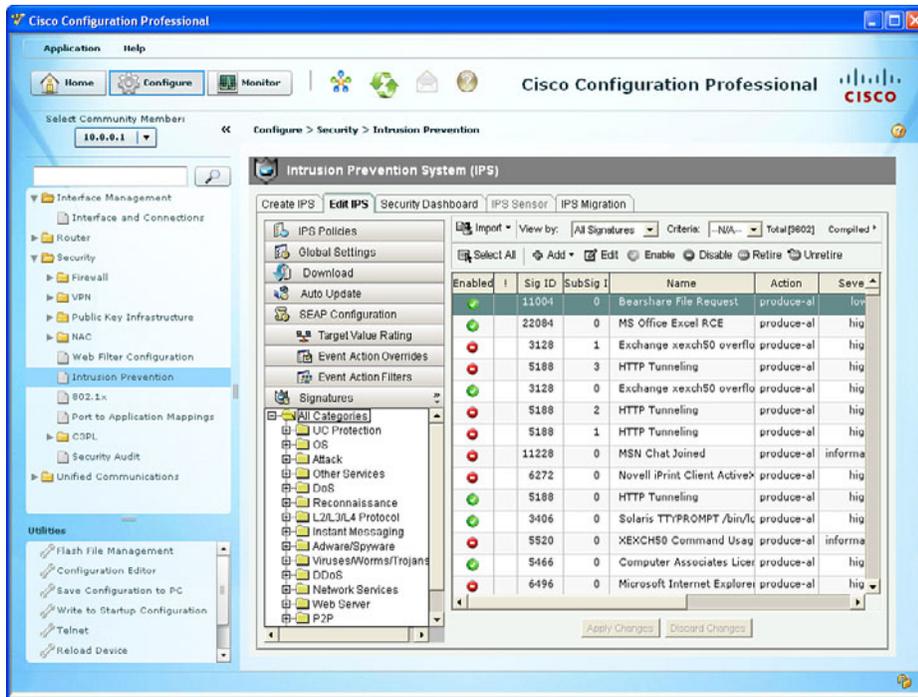


Figure 16-10 Viewing the IPS Signatures

To modify or view a specific signature, you can simply scroll through the list of the thousands of signatures, or you can use the filter options by selecting the **View By** drop-down menu, and then specifying that you want to view only specific signatures based on their name, their number, or some other attribute so that you can narrow the output to just the signatures that you want to see on the screen. Once you can see the signature you want to modify or view, you can click it to select it and then click the **Edit** button. Another option is to highlight the signature you want to edit, and then use one of the four modification buttons that are located just above the signatures. These four modification buttons are as follows:

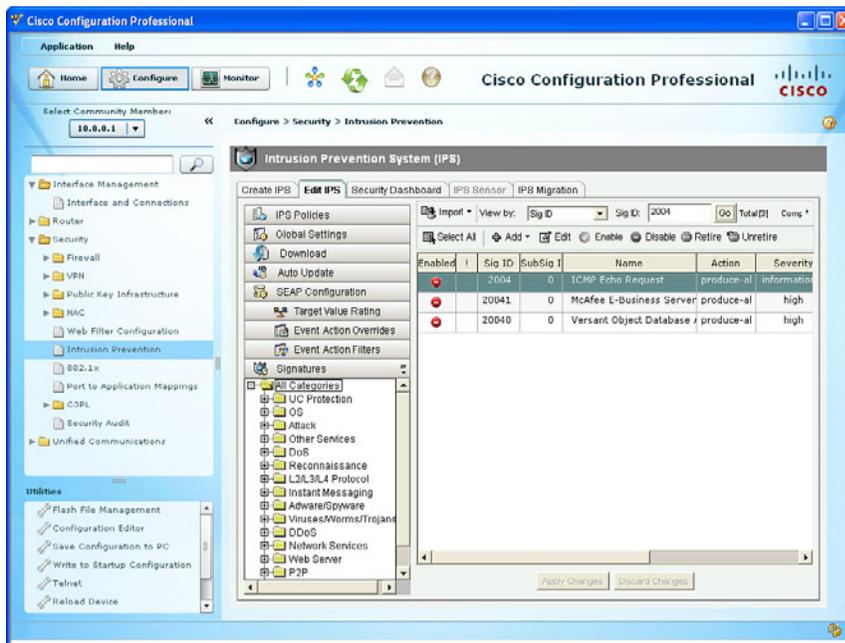
- **Enable:** Which makes the signature active so long as it is not retired. An enabled and nonretired signature is used as the router scans packets looking for malicious traffic.
- **Disable:** Retired or not, a signature that is disabled is not used to compare traffic against when performing IPS.
- **Retire:** If a signature is retired, it is not compiled as part of the available list of signatures that can be enabled. Long story short: Retiring signatures reduces the amount of RAM needed on the router.
- **Unretire:** This option causes the signature specified to be part of the compiled signatures that if also enabled will be actively used by IPS when scanning traffic.

Table 16-3 is a good overview of these four options.

**Table 16-3** Matrix for Retired/Unretired/Enabled/Disabled

Compiling/ Allowing Action	Enabled	Disabled
Retired	No memory consumption, and no action related to the signature during packet analysis	No memory consumption, and no action related to the signature during packet analysis
Unretired	Consumes memory, and the signature is considered during packet analysis	Consumes memory, but no action related to the signature during packet analysis

Figure 16-11 shows filtering based on the signature ID and including the string 2004 in the ID.

**Figure 16-11** Sorting Based on Signature ID

The red symbol in the Enabled column indicates that the corresponding signature is currently disabled. For testing purposes, if you want a signature to trigger every time an *Internet Control Message Protocol (ICMP)* echo request (ping request) is seen, you can highlight signature **2004**, click the **Enable** button, and then click the **Unretire** button (just to confirm that it is not retired).

When changes are made to a signature, a little orange/amber/yellow (take your pick) indicator shows up to let you know that changes have been made but have not been committed, as shown in Figure 16-12.

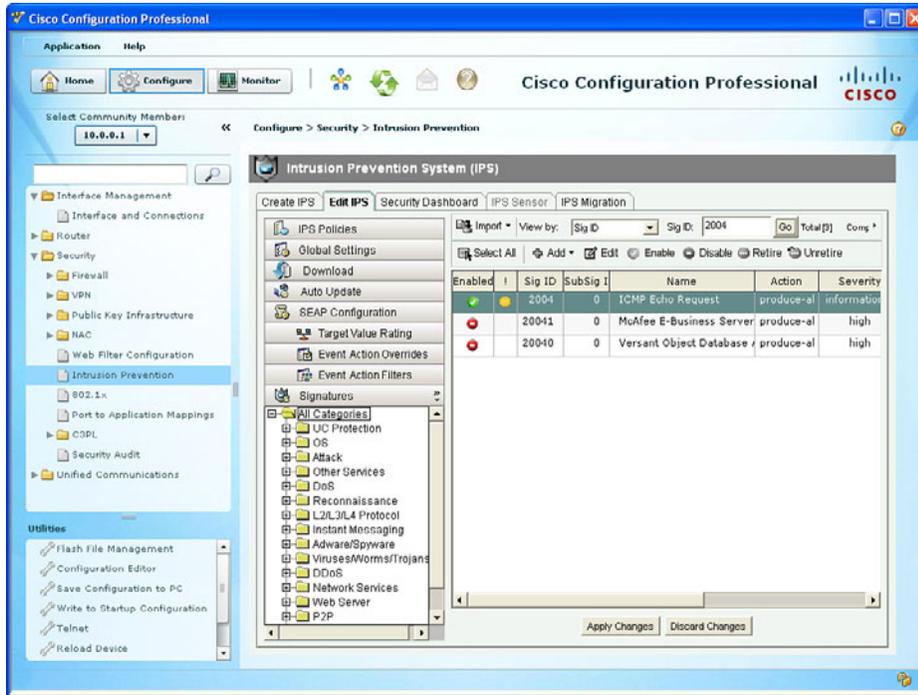


Figure 16-12 *Modifying the Properties of the Signature*

When you click the **Apply Changes** button, CCP applies the changes to the router, which requests the router to recompile the changes, and when it is done, if we send a ping request, it will trigger an alert.

Example 16-2 shows the CLI as the router is recompiling to implement the changes we made in the signature configuration.

Example 16-2 *CLI Output Indicating the Background Process*

```
R1#
%IPS-6-ENGINE_BUILDS_STARTED: 19:27:30 UTC Feb 21 2012
%IPS-6-ENGINE_BUILDING: multi-string - 40 signatures - 1 of 13 engines
%IPS-6-ENGINE_READY: multi-string - build time 56 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: service-http - 801 signatures - 2 of 13 engines
%IPS-6-ENGINE_READY: service-http - build time 380 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: string-tcp - 2058 signatures - 3 of 13 engines
%IPS-6-ENGINE_READY: string-tcp - build time 692 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: string-udp - 79 signatures - 4 of 13 engines
%IPS-6-ENGINE_READY: string-udp - build time 20 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: state - 37 signatures - 5 of 13 engines
%IPS-6-ENGINE_READY: state - build time 12 ms - packets for this engine
```

```

will be scanned
%IPS-6-ENGINE_BUILDING: atomic-ip - 373 signatures - 6 of 13 engines
%IPS-6-ENGINE_READY: atomic-ip - build time 600
R1# ms - packets for this engine will be scanned
%IPS-6-ENGINE_BUILDING: string-icmp - 3 signatures - 7 of 13 engines
%IPS-6-ENGINE_READY: string-icmp - build time 0 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: service-ftp - 3 signatures - 8 of 13 engines
%IPS-6-ENGINE_READY: service-ftp - build time 0 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: service-rpc - 76 signatures - 9 of 13 engines
%IPS-6-ENGINE_READY: service-rpc - build time 24 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: service-dns - 39 signatures - 10 of 13 engines
%IPS-6-ENGINE_READY: service-dns - build time 16 ms - packets for this
engine will be scanned
%IPS-6-ENGINE_BUILDING: normalizer - 9 signatures - 11 of 13 engines
%IPS-6-ENGINE_READY: normalizer - build time 0 ms - packets for this engine
will be scanned
R1#

```

Now that the signature is active and enabled, CCP indicates it with a green symbol (including a check mark, if you are looking at this in black and white) in the Enabled column, as shown in Figure 16-13.

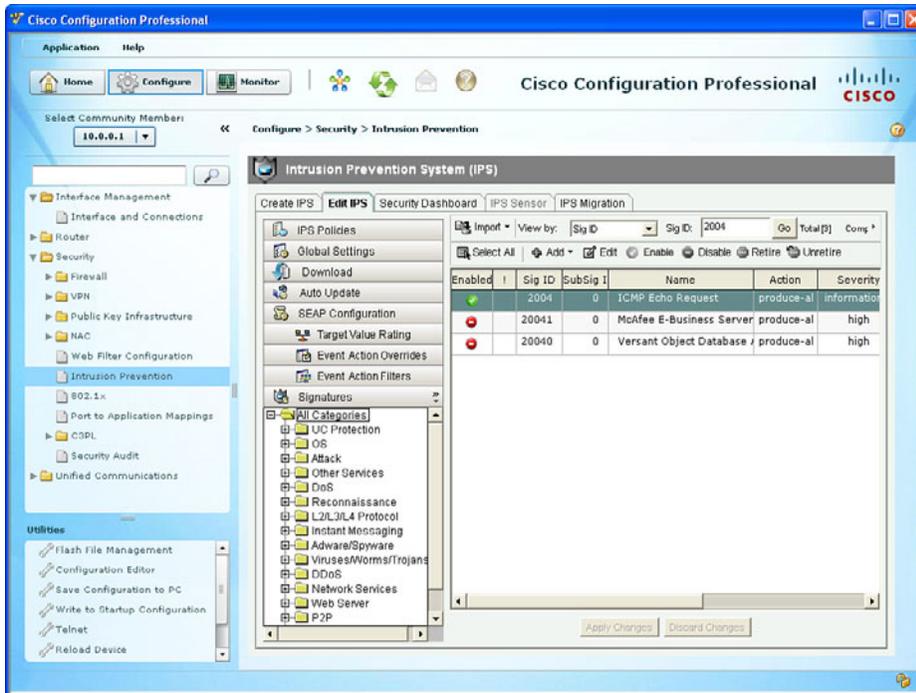


Figure 16-13 Enabled Signature 2004

Another reason for enabling this signature during the testing phase is because generating the attack (an ICMP echo request) is very easy to do and can be done to make sure the IPS function is actually responding and operating correctly. You can also see from this figure that the action (from the Action column) is to generate an alert.

Actions That May Be Taken

The possible actions that we can choose in IOS IPS are as follows:

- Deny attacker inline
- Deny connection inline
- Deny packet inline
- Produce alert (the default action for most signatures)
- Reset TCP connection (effective only against TCP-based attacks)

To modify the actions, right-click the signature, select **Actions**, and simply put a check mark in the boxes next to the actions you want to take against the attacker, as shown in Figure 16-14.

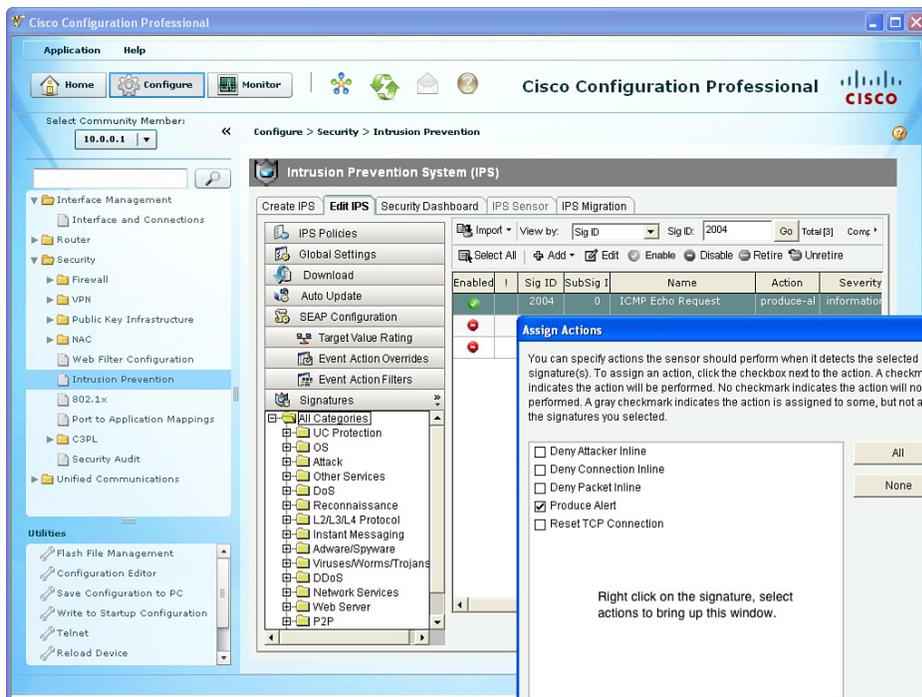


Figure 16-14 Assigning Actions to a Signature

Click **OK** and then **Apply Changes** to implement any changes you made. Once the router has compiled these changes, they are implemented on the router. In this example,

a ping request should now trigger an alert. Let's try it by sending a ping request from a workstation to the router on the interface that has the IPS rule applied.

Example 16-3 shows this in action.

Example 16-3 *Testing the IPS Signature 2004*

```
! From the PC at 10.0.0.25, a ping to the router at 10.0.0.1
C:\> ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time=38ms TTL=255
Reply from 10.0.0.1: bytes=32 time=9ms TTL=255
Reply from 10.0.0.1: bytes=32 time=10ms TTL=255
Reply from 10.0.0.1: bytes=32 time=9ms TTL=255
Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 9ms, Maximum = 38ms, Average = 16ms
C:\>

! On the Router console
R1#
%IPS-4-SIGNATURE: Sig:2004 Subsig:0 Sev:25 ICMP Echo Request [10.0.0.25:8
-> 10.0.0.1:0] VRF:NONE RiskRating:25
R1#
%IPS-4-SIGNATURE: Sig:2004 Subsig:0 Sev:25 ICMP Echo Request [10.0.0.25:8
-> 10.0.0.1:0] VRF:NONE RiskRating:25
R1#
%IPS-4-SIGNATURE: Sig:2004 Subsig:0 Sev:25 ICMP Echo Request [10.0.0.25:8
-> 10.0.0.1:0] VRF:NONE RiskRating:25
R1#
%IPS-4-SIGNATURE: Sig:2004 Subsig:0 Sev:25 ICMP Echo Request [10.0.0.25:8
-> 10.0.0.1:0] VRF:NONE RiskRating:25
R1#
```

Now we know that the basic functionality of the IPS is working. We also addressed how to take a look at signatures. If you want to modify additional properties of any signature, highlight that signature and click the edit icon. In the pop-up window, the green square box (just to the left of the item) indicates the value is at its default. If you want to modify one of these properties, click the green box, which changes it to a red diamond (in black and white, this looks like a box that has been rotated 45 degrees, with a corner facing up), and then this allows you to change the value of that property. For example, if you want to modify a signature property of Alert Severity from a high severity level to informational, the previously explained process is how you do it and is shown in Figure 16-15.

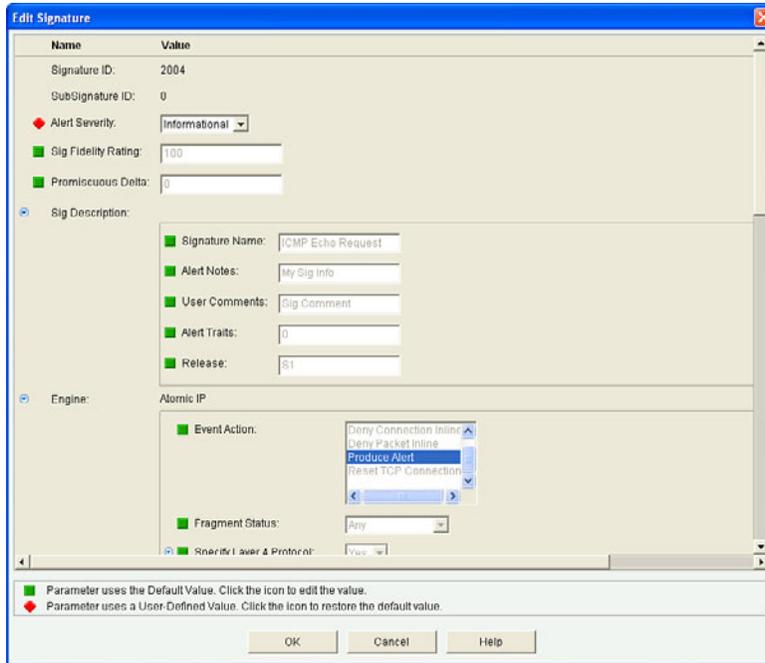


Figure 16-15 *Modifying the Properties of a Signature*

All the configuration that we have done via the GUI of CCP we could have also done at the command line. Example 16-4 shows the CLI equivalent for what we have configured.

Example 16-4 *CLI Commands for Configuring IPS*

```

! Enables SDEE
R1(config)# ip ips notify SDEE

! Creates an IPS rule
! Note: many of the CCP scripts actually are using the routines of its
! predecessor Security Device Manager (SDM). Many of the entries it
! creates will use the term "sdm" in its naming conventions for that reason.
R1(config)# ip ips name sdm_ips_rule

! Disables the advanced, and basic categories included in "all"
R1(config)# ip ips signature-category
R1(config-ips-category)# category all
R1(config-ips-category-action)# retired true
R1(config-ips-category-action)# exit

! Enables the basic signature category (which compiles less signatures)
! we are still in IPS configuration mode, the exit took us 1 step back but
! not out

```

```

R1(config-ips-category)# category ios_ips basic
R1(config-ips-category-action)# retired false
R1(config-ips-category-action)# exit
R1(config-ips-category)# exit
Do you want to accept these changes? [confirm]

! apply the IPS rule inbound on the interface
R1(config)# interface FastEthernet1/0
R1(config-if)# ip ips sdm_ips_rule in
R1(config-if)# exit

! Specify where the router should keep the configuration of any custom or tuned
! signatures or rules that we configure
R1(config)# ip ips config location ftp://10.0.0.2/ips5
Writing ips5/R1-sigdef-default.xml
Writing ips5/R1-sigdef-delta.xml
Writing ips5/R1-sigdef-typedef.xml
Writing ips5/R1-sigdef-category.xml
Writing ips5/R1-seap-delta.xml
Writing ips5/R1-seap-typedef.xml

! enable signature 2004 to ensure it is both enabled and not retired
R1(config)# ip ips signature-definition
R1(config-sigdef)# signature 2004
R1(config-sigdef-sig)# status
R1(config-sigdef-sig-status)# enabled true
R1(config-sigdef-sig-status)# retired false
R1(config-sigdef-sig-status)# exit
R1(config-sigdef-sig)# exit
R1(config-sigdef)# exit
Do you want to accept these changes? [confirm]
Writing ips5/R1-sigdef-delta.xml
%IPS-6-ENGINE_BUILDS_STARTED: 20:06:53 UTC Feb 21 2012
%IPS-6-ENGINE_BUILDING: atomic-ip - 373 signatures - 1 of 13 engines
%IPS-6-ENGINE_READY: atomic-ip - build time 576 ms - packets for this
engine will be scanned
%IPS-6-ALL_ENGINE_BUILDS_COMPLETE: elapsed time 1200 ms

! To verify our configuration

R1# show ip ips configuration

IPS Signature File Configuration Status
Configured Config Locations: ftp://10.0.0.25/ips5/
Last signature default load time: 18:36:08 UTC Feb 21 2012
Last signature delta load time: 20:06:55 UTC Feb 21 2012
Last event action (SEAP) load time: -none-

```

```
General SEAP Config:
Global Deny Timeout: 3600 seconds
Global Overrides Status: Enabled
Global Filters Status: Enabled

IPS Auto Update is not currently configured

IPS Syslog and SDEE Notification Status
Event notification through syslog is enabled
Event notification through SDEE is enabled

IPS Signature Status
Total Active Signatures: 299
Total Inactive Signatures: 3303

IPS Packet Scanning and Interface Status
IPS Rule Configuration
  IPS name sdm_ips_rule
  IPS fail closed is disabled
  IPS deny-action ips-interface is false
Interface Configuration
  Interface FastEthernet1/0
    Inbound IPS rule is sdm_ips_rule
    Outgoing IPS rule is not set

IPS Category CLI Configuration:
Category all:
  Retire: True
Category ios_ips basic:
  Retire: False

R1#
! To check on a specific signature, such as 2004
! note: the subsignature ID is 0 for many signatures, including 2004

R1# show ip ips signatures sigid 2004 subid 0

En - possible values are Y, Y*, N, or N*
  Y: signature is enabled
  N: enabled=false in the signature definition file
  *: retired=true in the signature definition file
Cmp - possible values are Y, Ni, Nr, Nf, or No
  Y: signature is compiled
  Ni: signature not compiled due to invalid or missing parameters
```

```

    Nr: signature not compiled because it is retired
    Nf: signature compile failed
    No: signature is obsoleted
Action=(A)lert, (D)eny, (R)eset, Deny-(H)ost, Deny-(F)low
Trait=alert-traits          EC=event-count          AI=alert-interval
GST=global-summary-threshold  SI=summary-interval  SM=summary-mode
SW=swap-attacker-victim      SFR=sig-fidelity-rating Rel=release

SigID:SubID En  Cmp  Action Sev  Trait  EC  AI  GST  SI  SM SW SFR Rel
-----
2004:0      Y  Y   A     INFO 0    1  0   200  30  FA N 100 S1
  sig-name: ICMP Echo Request
  sig-string-info: My Sig Info
  sig-comment: Sig Comment
  Engine atomic-ip params:
    fragment-status :
    icmp-type : 8
    l4-protocol : icmp
R1#

! A way to view the number of active signatures, as well as the micro-engines
R1# show ip ips signatures count

Cisco SDF release version S456.0
Trend SDF release version V0.0

Signature Micro-Engine: multi-string: Total Signatures 40
  multi-string enabled signatures: 34
  multi-string retired signatures: 36
  multi-string compiled signatures: 4

Signature Micro-Engine: service-http: Total Signatures 801
  service-http enabled signatures: 133
  service-http retired signatures: 746
  service-http compiled signatures: 55
  service-http obsoleted signatures: 3

Signature Micro-Engine: string-tcp: Total Signatures 2058
  string-tcp enabled signatures: 675
  string-tcp retired signatures: 1932
  string-tcp compiled signatures: 126
  string-tcp obsoleted signatures: 22

Signature Micro-Engine: string-udp: Total Signatures 79
  string-udp enabled signatures: 0

```

```
string-udp retired signatures: 73
string-udp compiled signatures: 6
string-udp obsoleted signatures: 2
```

```
Signature Micro-Engine: state: Total Signatures 37
state enabled signatures: 16
state retired signatures: 26
state compiled signatures: 11
```

```
Signature Micro-Engine: atomic-ip: Total Signatures 373
atomic-ip enabled signatures: 91
atomic-ip retired signatures: 351
atomic-ip compiled signatures: 22
```

```
Signature Micro-Engine: string-icmp: Total Signatures 3
string-icmp enabled signatures: 0
string-icmp retired signatures: 3
```

```
Signature Micro-Engine: service-ftp: Total Signatures 3
service-ftp enabled signatures: 1
service-ftp retired signatures: 2
service-ftp compiled signatures: 1
```

```
Signature Micro-Engine: service-rpc: Total Signatures 76
service-rpc enabled signatures: 44
service-rpc retired signatures: 52
service-rpc compiled signatures: 24
```

```
Signature Micro-Engine: service-dns: Total Signatures 39
service-dns enabled signatures: 27
service-dns retired signatures: 16
service-dns compiled signatures: 23
service-dns obsoleted signatures: 1
```

```
Signature Micro-Engine: normalizer: Total Signatures 9
normalizer enabled signatures: 8
normalizer retired signatures: 1
normalizer compiled signatures: 8
```

```
Signature Micro-Engine: service-smb-advanced: Total Signatures 49
service-smb-advanced enabled signatures: 40
service-smb-advanced retired signatures: 35
service-smb-advanced compiled signatures: 14
```

```
Signature Micro-Engine: service-msrpc: Total Signatures 35
service-msrpc enabled signatures: 17
```

```
service-msrpc retired signatures: 30
service-msrpc compiled signatures: 5
service-msrpc obsoleted signatures: 1

Total Signatures: 3602
Total Enabled Signatures: 1086
Total Retired Signatures: 3303
Total Compiled Signatures: 299
Total Obsoleted Signatures: 29

R1#
```

Best Practices When Tuning IPS

The following are best practices when tuning IPS:

- Begin with the basic signature category, and see how much memory and CPU utilization this takes in the production network, before moving to the advanced signature category which will take significantly more CPU and resources from the router.
- Schedule downtime for the installation and updates.
- Retire signatures that are irrelevant to your network to save resources on the router.
- Monitor free memory to ensure that you do not cause harm to your router by loading too many additional services.
- There are options available that can tell the IOS router to not forward any traffic through an IPS-protected interface if some type of problem causes the signature not to compile. The term for this is *fail closed*. The other option, which indicates that if a problem with the IPS signatures not compiling occurs the router should still forward traffic, is called *fail open*. Based on the security policy, you want to choose the option that meets the needs for the company. A fail close could cause a failure of the network due to a failure of IPS, but it is more secure than fail open.
- For performance reasons, be very careful before unretiring and enabling the All category of signatures.

Managing and Monitoring IPS Alarms

This section examines the options for viewing alerts and alarms and demonstrates how to do it via CCP and the CLI.

You have already seen how to view alerts right at the CLI of the router, through console messages. Let's take a more in-depth look at managing and working with alerts being generated from the IPS function on the router.

In CCP, navigate to **Monitor > Security > IPS Status**. By default, it displays the information on the IPS Signature Statistics tab. You can scroll down this list until you find the signature you are interested in, or simply scroll down to see how many times the various signatures have been triggered. In Figure 16-16, you can see that signature 2004 signature has been triggered.

The screenshot shows the Cisco Configuration Professional (CCP) interface. The main window is titled "Cisco Configuration Professional" and has a menu bar with "Application" and "Help". Below the menu bar are navigation buttons: "Home", "Configure", and "Monitor". The "Monitor" button is selected. The breadcrumb navigation shows "Monitor > Security > IPS Status".

The "IPS Status" window is open, showing the "IPS Signature Statistics" tab. At the top, it displays "Total Signatures: 3602" with "Update", "Clear", and "SDEE Log" buttons. Below this, it shows "Total Enabled Signatures: 1006", "Total Retired Signatures: 3303", and "Total Compiled Signatures: 299".

Signature ID	Description	Source IP Address	Destination IP Address	Hits	0
1107.0	RFC 1918 Addresses Set			0	0
2150.0	Fragmented ICMP Traffic			0	0
1005.0	IP options- SATNET ID			0	0
1003.0	IP options-Provide s,c,h,l,c			0	0
1002.0	IP options-Timestamp			0	0
1001.0	IP options-Record Packet			0	0
2151.0	Large ICMP Traffic			0	0
2011.0	ICMP Address Mask Reql			0	0
2010.0	ICMP Information Reply			0	0
2009.0	ICMP Information Request			0	0
2008.0	ICMP Timestamp Reply			0	0
2007.0	ICMP Timestamp Request			0	0
2006.0	ICMP Parameter Problem			0	0
2005.0	ICMP Time Exceeded for e			0	0
2004.0	ICMP Echo Request	10.0.0.25.8	10.0.0.1.0	4	0
2003.0	ICMP Redirect			0	0
2002.0	ICMP Source Quench			0	0
2000.0	ICMP Echo Reply			0	0
5485.1	ISS PAM.dll ICO Parser Bt			0	0
5684.2	Malformed SIP Packet			0	0
20363.0	Firewall Services Module			0	0
17401.0	TFTPDWNN Long Message			0	0
17397.0	Asterisk Channel Driver R			0	0

Figure 16-16 Viewing Alerts from Within CCP

This screen also provides information on the total number of signatures, including information about how many of those are enabled, retired, and compiled.

Another way to view all alerts, including additional details about those alerts, is by clicking the **SDEE Log** hyperlink in the upper-right corner of Figure 16-16. The SDEE log is shown in Figure 16-17.

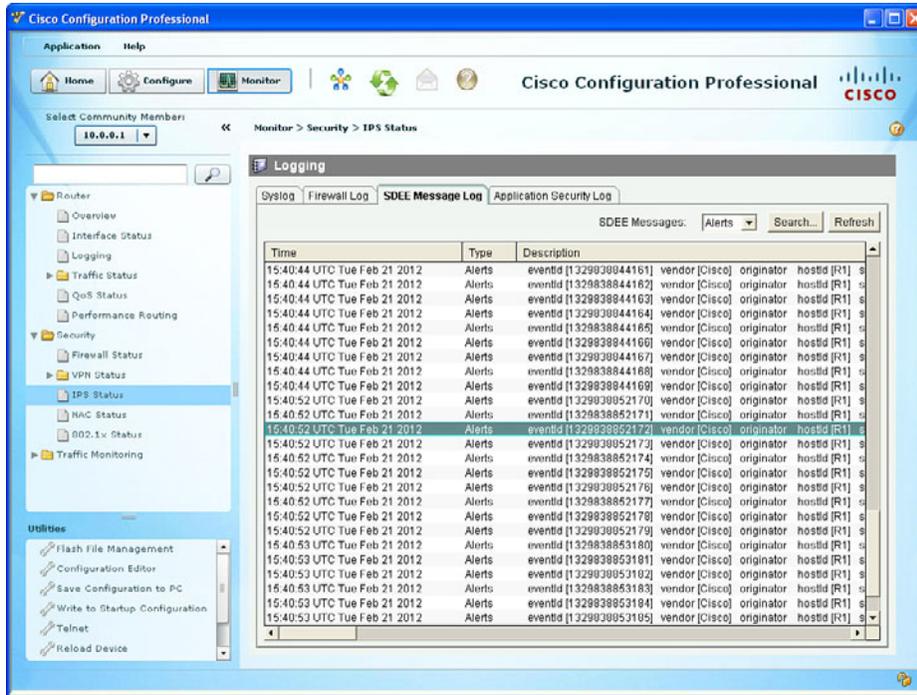


Figure 16-17 SDEE Log File

Another way to get to the same log screen is to navigate to **Monitor > Router > Logging** and click the **SDEE Message Log** tab (the same one shown earlier). Regardless of how you navigate to get to this log information, it includes the time stamp for the alerts, the type (an alert or a status entry), and the event ID, signature ID, the risk rating, and other information related to the alarm.

You can also filter what is shown in this log window by using the drop-down menu next to SDEE Messages and selecting one of the four filtering options:

- All
- Error
- Status
- Alerts

Another handy feature is the ability to search for specific entries using the Search button, which enables you to search based on source or destination IP address and specific text that you want to find in one of the messages.

Another way to view individual messages inside of CCP is to navigate to **Monitor > Security > IPS Status** and click the **IPS Alert Statistics** tab, as shown in Figure 16-18.

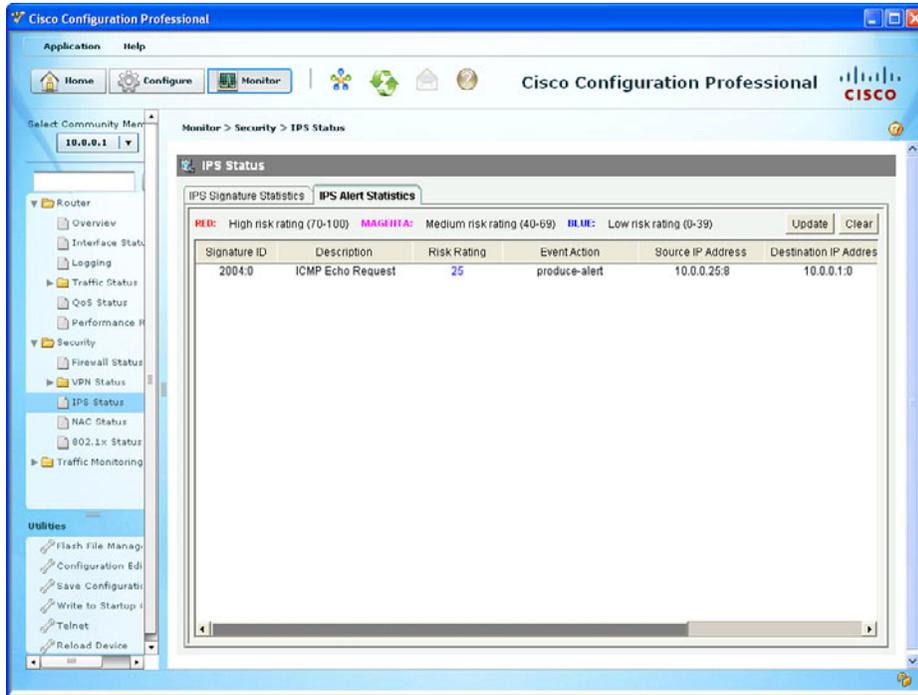


Figure 16-18 Viewing the Event Details

In the output, although it is not obvious in black and white, the live interface color codes the risk rating value that is generated/calculated for this alert. It also shows the risk rating categories and their ranges, which makes the color just eye candy. The reason the risk rating is important is because you can configure something called an event action override, which automatically initiates countermeasures based on the resulting risk rating. Many of the additional features in IPS are not covered directly in the CCNA Security material, but have no fear, they will be waiting for you when you move on to the CCNP Security, where those and other topics are covered in greater depth.

Another way to look at this information is from the command line, as shown in Example 16-5.

Example 16-5 Viewing Alerts from the CLI

```
R1# show ip sdee alerts
Alert storage: 200 alerts using 96000 bytes of memory

                                SDEE Alerts
  SigID  Sig Name                SrcIP:SrcPort  DstIP:DstPort
  ---  ---
  1:  2004:0  ICMP Echo Request  10.0.0.25:8   10.0.0.1:0
  2:  2004:0  ICMP Echo Request  10.0.0.25:8   10.0.0.1:0
  3:  2004:0  ICMP Echo Request  10.0.0.25:8   10.0.0.1:0
  4:  2004:0  ICMP Echo Request  10.0.0.25:8   10.0.0.1:0
```



```
R1#
```

```
R1# show ip ips statistics
```

```
Signature statistics [process switch:fast switch]
```

```
signature 2004:0: packets checked [0:4] alarmed [0:4] dropped [0:0]
```

```
Interfaces configured for ips 1
```

```
Session creations since subsystem startup or last reset 264
```

```
Current session counts (estab/half-open/terminating) [4:0:0]
```

```
Maxever session counts (estab/half-open/terminating) [6:2:0]
```

```
Last session created 00:00:45
```

```
Last statistic reset never
```

```
TCP reassembly statistics
```

```
received 38 packets out-of-order; dropped 6
```

```
peak memory usage 37 KB; current usage: 0 KB
```

```
peak queue length 16
```

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 16-4 lists these key topics.

Table 16-4 *Key Topics*

Key Topic Element	Description	Page Number
Table 16-2	IOS IPS features	393
Figure 16-7	Adding Cisco public key information	397
Example 16-1	Output from console while signatures are compiled	399
Figure 16-10	Viewing the IPS signatures	401
Table 16-3	Matrix for retired/unretired/enabled/disabled	402
Figure 16-12	Modifying the properties of the signature	403
Figure 16-14	Assigning actions to a signature	405
Figure 16-18	Viewing the event details	415
Example 16-5	Viewing alerts from the CLI	415



Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

IPS, signature files, Cisco public key, signature micro-engines, enabled, disabled, retired, unretired

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side Table 16-5 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 16-5 *Command Reference*

Command	Description
<code>show ip sdee alerts</code>	Allow viewing of alert events from the CLI
<code>show ip ips configuration</code>	Allow viewing of the IPS configuration from the CLI
<code>ip ips <i>sdm_ips_rule</i> in</code>	Apply a rule named <code>sdm_ips_rule</code> inbound on the current interface it is being configured under

This page intentionally left blank

CCNA Security exam topics covered in this part:

- Describe the different methods used in cryptography
- Describe VPN technologies
- Describe the building blocks of IPsec
- Implement an IOS IPsec site-to-site VPN with pre-shared key authentication
- Verify VPN operations
- Implement Secure Sockets Layer (SSL) VPN using ASA Device Manager

Part IV: Using VPNs for Secure Connectivity

Chapter 17: Fundamentals of VPN Technology

Chapter 18: Fundamentals of the Public Key Infrastructure

Chapter 19: Fundamentals of IP Security

Chapter 20: Implementing IPsec Site-to-Site VPNs

Chapter 21: Implementing SSL VPNs Using Cisco ASA

Chapter 22: Final Preparation



This chapter covers the following subjects:

- Understanding VPNs and why we use them
- Cryptography basic components

Fundamentals of VPN Technology

Virtual private networks (VPN) are a huge hit, and they are not just a fad. Understanding why they are important and the underlying building blocks that make them work so well is the focus of this chapter.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 17-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 17-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Understanding VPNs and Why We Use Them	1–5
Cryptography Basic Components	6–10

1. What element in a VPN provides the *P* portion?
 - a. Data integrity
 - b. Confidentiality
 - c. Antireplay
 - d. Authentication
2. What algorithms in a VPN provide the confidentiality? (Choose all that apply.)
 - a. MD5
 - b. SHA-1
 - c. AES
 - d. 3DES

- 3.** A remote user needs to access the corporate network from a hotel room from a laptop. What type of VPN is used for this?
 - a.** Site-to-site VPN
 - b.** Dial-up VPN
 - c.** PPP VPN
 - d.** Remote-access VPN
- 4.** Which type of VPN technology is likely to be used in a site-to-site VPN?
 - a.** SSL
 - b.** TLS
 - c.** HTTPS
 - d.** IPsec
- 5.** Which two of the following are benefits of VPNs?
 - a.** Hashing
 - b.** Confidentiality
 - c.** Diffie-Hellman
 - d.** Data integrity
- 6.** How can a publicly available and well-known cipher be used to securely encrypt data between two endpoints of a VPN tunnel?
 - a.** MD5
 - b.** Keys
 - c.** Authentication
 - d.** Antireplay
- 7.** Which of the of following are symmetrical encryption ciphers? (Choose all that apply.)
 - a.** SHA1
 - b.** AES
 - c.** RSA
 - d.** 3DES

- 8.** What is the primary difference between a hash and Hashed Message Authentication Code (HMAC)?
 - a.** Keys
 - b.** MD5
 - c.** SHA1
 - d.** AES

- 9.** What is used to encrypt the hash in a digital signature?
 - a.** Sender's public key
 - b.** Sender's private key
 - c.** Receiver's public key
 - d.** Receiver's private key

- 10.** What are valid options to protect data in motion with or without a full VPN? (Choose all that apply.)
 - a.** TLS
 - b.** SSL
 - c.** HTTPS
 - d.** IPsec

Foundation Topics

Understanding VPNs and Why We Use Them

This section examines the reasons why VPNs are so important and what types of VPNs are available to deploy and why a specific type of VPN is appropriate for a given business need.

What Is a VPN?



If we break down the term *virtual private network* into its individual components, we could say that a network allows connectivity between two devices. Those two devices could be computers on the same local-area network or could be connected over a wide-area network. In either case, a network is providing the basic connectivity between the two. The word *virtual* in VPN refers to a logical connection between the two devices. For example, one user may be connected to the Internet in Las Vegas, Nevada, and another user may be connected to the Internet in Vienna, Austria, and we could build a logical network, or virtual network, between the two devices using the Internet as our transport mechanism. The letter *P* in VPN refers to *private*. The virtual network we could create between our two users in Las Vegas and Vienna would be private between those two parties. So, there are the basics for VPN, a virtual private network.

Unfortunately, if we did have a VPN established between two devices over the Internet, what would prevent an individual who had access to the packets from eavesdropping on the conversation? The answer is not much, by default. So, in addition to most VPNs, we add the ingredients of confidentiality and data integrity so that anyone who is eavesdropping cannot make sense of the data because it is encrypted and they do not have the keys required to decrypt or unlock the data to see what the data actually is. The confidentiality provided by the encryption could also represent the *P* in VPNs. We also use integrity checking to make sure that our VPN is correctly seeing the packets as they were sent from the other side of the VPN and that they are not being altered or manipulated maliciously along the path.

Using the example of the user in Vienna and Las Vegas, why would we ever want to use a VPN between the two? We do have other options for connectivity. We could purchase each user a dedicated WAN connection from Vienna to Las Vegas. Each user could connect to his local side and communicate with each other over the dedicated link. One of the obvious problems with this is cost. It is much cheaper to connect the user to the Internet through a local service provider than to purchase a dedicated circuit that goes to only one other destination.

Another benefit of using a VPN is scalability. If 10 or 20 more new users need to connect to the corporate headquarters, we can provide users access to the Internet via their local service providers (*digital subscriber line [DSL]*, cable modem, and so on). Leveraging the single Internet connection from the headquarters site, we could then simply build logical VPNs using the Internet for the connectivity.

Types of VPNs

Based on the definition of a virtual private network, the following could be considered VPN technologies:

- **IPsec:** Implements security of IP packets at Layer 3 of the OSI model, and can be used for site-to-site VPNs and remote-access VPNs.
- **SSL:** Secure Sockets Layer implements security of TCP sessions at Layer 4 of the OSI model, and can be used for remote-access VPNs (as well as being used to securely visit a web server that supports it via HTTPS).
- **MPLS:** Multiprotocol Label Switching and MPLS Layer 3 VPNs are provided by a service provider to allow a company with two or more sites to have logical connectivity between the sites using the service provider network for transport. This is also a type of VPN (called *MPLS L3VPN*), but there is no encryption by default. IPsec could be used on top of the MPLS VPN to add confidentiality (through encryption) and the other benefits of IPsec to protect the Layer 3 packets. MPLS L3VPNs are not the primary type of VPNs we focus on for the rest of this chapter and book. The primary VPNs that provide encryption, data integrity, authentication of who the peer is on the other end of the VPN, and so on use IPsec or SSL.



Two Main Types of VPNs

There are two major categories into which VPNs could be placed: remote-access and site-to-site. The following are details about each, including when they might be used:

- **Remote-access VPNs:** Some users might need to build a VPN connection from their individual computer to the corporate headquarters (or to the destination they want to connect to). This is referred to as a *remote-access VPN connection*. Remote-access VPNs can use IPsec or SSL technologies for their VPN.
- **Site-to-site VPNs:** The other main VPN implementation is by companies that may have two or more sites that they want to connect securely together (likely using the Internet) so that each site can communicate with the other site or sites. This implementation is called a *site-to-site VPN*. Site-to-site VPNs traditionally use a collection of VPN technologies called *IPsec*.

Main Benefits of VPNs

The main benefits of using a VPN for either remote access or site to site, include the following:

- Confidentiality
- Data integrity
- Authentication
- Antireplay



We take a closer look at each of these four items right now.

Confidentiality

Confidentiality means that only the intended parties can understand the data that is sent. Any party that eavesdrops may see the actual packets, but the contents of the packet or the payload are scrambled (also called *cipher text*) and meaningless to anyone who cannot unlock or decrypt the data.

Consider the practical illustration shown in Example 17-1.

Example 17-1 A Secret Message, Encrypted

Tp uijt jt uif tfdsfu nfttbhf. Ju jt fbtz up ef-fodszqu jg zpv lopx uif lfz.

Take a moment and see whether you can figure out what the message means, and imagine that this is the payload of a packet being sent over a VPN and that you have intercepted the packet and are trying to understand or make sense of it. The major goal of a VPN is confidentiality, and it is accomplished by the sender encrypting the data or the packet that needs to be protected and then sending it over the VPN. The receiver of the packet or data then faces the same challenge as the eavesdropper, in that the data must be decrypted to make sense out of it. The algorithms and formulas for encrypting data are publicly available and are well known. The part that makes the message secret is the key or “secret” that is used to encrypt the data. If the sender and the receiver both know the key that is used, they can encrypt and decrypt information back and forth using the same key or keys, and anyone in the middle who does not know the key or keys that were used cannot decrypt.

Going back to the encrypted message, if a “symmetric” algorithm was used, then if I gave you the key that I used to encrypt it, you could use that same key to decrypt it. So, here’s the key (pun intended) for the encrypted example: Every letter in the encrypted message is one off of the real letter in the alphabet. So, take each letter, and replace it with the letter before it in the alphabet and you will have decrypted the message. The purpose of this exercise is to demonstrate the concept of how confidentiality is implemented through some type of an encryption process, and how having the keys come in handy for the person who is decrypting. An encryption algorithm that uses the same key or keys for encryption and those exact same key or keys for decryption is an example of a *symmetrical* encryption algorithm. Example 17-2 shows the decrypted message from Example 17-1.

Example 17-2 Results of Secret Message When Using the Correct “Key”

"So this is the secret message. It is easy to de-encrypt if you know the key."

Data Integrity

If two devices are communicating over a VPN, another important factor about the data that is being sent is to make sure it is accurate from end to end. If an attacker injects bits

or data into the packets of a VPN session, data integrity could suffer if the modification of the data goes undetected.

An example of how to verify data is to look at a Cisco IOS image. When you download an IOS image and place it on the flash of a router, how do you know the data file is exactly the same as the file you tried to download from Cisco? For example, there could have been an error during the download, and perhaps the file is not sound. To verify the integrity of the downloaded image, you can use the **verify** command on a Cisco router, as shown in Example 17-3. The download page for the IOS software provides a *message digest algorithm 5 (MD5)* value associated with the file. If you download the file and run your own test against the file and it generates that same number, you know that the file you downloaded matches the file that Cisco had on their site. Later in this chapter, you take a closer look at how hashing works.

Example 17-3 Verifying Data Integrity with a Hash

```
dev-1# dir Directory
of flash:/

   1  -rw-      57762740   c2800nm-advipservicesk9-mz.124-24.T4.bin

64225276 bytes total (6456576 bytes free)

! Run the verify command to see what the md5 hash result is
dev-1# verify /md5 flash:/c2800nm-advipservicesk9-mz.124-24.T4.bin
.....
.....
.....
.....
.....
.....
.....
.....
.....Done!
verify /md5 (flash:/c2800nm-advipservicesk9-mz.124-24.T4.bin) =
 28518159ba5f75ef0eeb9617fd35e2ba
dev-1#
! Now we would compare the md5 hash generated by the router to the hash
! shown on the download page from Cisco.com for that file.  If they match,
! we have data integrity.
```

Now that we know the concept of being able to verify the integrity of a file, what if we could use that same process to verify the integrity of each and every packet? That is exactly what the hashing protocols inside of our VPNs can do for us to verify the integrity of each packet using a *hash*.

Authentication

A VPN tunnel is fantastic in that you can encrypt data and verify that data has not been modified while in transit. But what if you have established a VPN connection, also called a VPN tunnel, directly to the attacker's computer? Being able to validate or authenticate the device that you are connected to is an important aspect of a good VPN. You can authenticate the peer at the other end of the VPN tunnel in several different ways, including the following:

- Pre-shared keys used for authentication only
- Public and private key pairs used for authentication only
- User authentication (in combination with remote-access VPNs)

Antireplay

If an attacker watches your VPN traffic and captures it with the intent to replay it back and fool one of the VPN peers into believing that the peer trying to connect is a legitimate peer, an attacker might be able to build a VPN pretending to be a different device. To solve that, most implementations of VPNs have an antireplay functionality built in. This just means that once a VPN packet has been sent and accounted for, that exact same VPN packet is not valid the second time in the VPN session.

Cryptography Basic Components

You now know that confidentiality is a function of encryption, data integrity is a function of hashing, and authentication is the process of proving the identity of the other side of the tunnel. Now it is time to take a look at how those methods are implemented and the choices you have for each.

This section discusses the basic components cryptography, including algorithms for hashing, encryption, and key management, which may be used by VPNs.

Ciphers and Keys



Understanding the terminology is a large part of understanding any technology, so let's begin with some fundamentals.

Ciphers

A *cipher* is a set of rules, which can also be called an *algorithm*, about how to perform encryption or decryption. Literally hundreds of encryption algorithms are available, and

there are likely many more that are proprietary and used for special purposes such as government and national security.

Common methods that ciphers use include the following:

- **Substitution:** This type of cipher substitutes one character for another. The example earlier used a simple cipher that substituted each letter from the alphabet with the previous letter of the alphabet. To make it more challenging, we could have shifted more than just a single character and only chose certain letters to substitute. The exact method of substitution could be referred to as the *key*. If both parties involved in the VPN understand the key, they can both encrypt and decrypt data.
- **Polyalphabetic:** This is similar to substitution, but instead of using a single alphabet, it could use multiple alphabets and switch between them by some trigger character in the encoded message.
- **Transposition:** This uses many different options, including the rearrangement of letters. For example, if we have the message “This is secret,” we could write it out (top to bottom, left to right) as shown in Example 17-4.

Example 17-4 *Transposition Example*

T	S	S	R
H	I	E	E
I	S	C	T

We then encrypt it as RETCSIHTSSEI, which is starting at top right and going around like a clock, spiraling inward. To know how to encrypt/decrypt this correctly, we need the correct key.

Keys

The key in Example 17-4 refers to the instructions for how to reassemble the characters. In this case, it begins at the top-right corner and moves clockwise and spirals inward.

A *one-time pad (OTP)* is a good example of a key that is only used once. Using this method, if we want to encrypt a 32-bit message, we use a 32-bit key, also called the *pad*, which is used one time only. Each bit from the pad is mathematically computed with a corresponding bit from our message, and the results are our cipher text, or encrypted content. The key in this case is the one-time use pad. The pad must also be known by the receiver if he wants to decrypt the message. (Another use of the acronym OTP is for a user’s one-time password, which is a different topic than the one-time pad.)

Block and Stream Ciphers

Encryption algorithms can operate on blocks of data at a time, or bits and bytes of data, based on the type of cipher. Let’s compare the two methods.

Block Ciphers

A block cipher is a *symmetric* key (same key to encrypt and decrypt) cipher that operates on a group of bits called a *block*. A block cipher encryption algorithm may take a 64-bit block of plain text and generates a 64-bit block of cipher text. With this type of encryption, the same key to encrypt is also used to decrypt. Examples of symmetrical block cipher algorithms include the following:

- *Advanced Encryption Standard (AES)*
- *Triple Digital Encryption Standard (3DES)*
- Blowfish
- *Digital Encryption Standard (DES)*
- *International Data Encryption Algorithm (IDEA)*

Block ciphers may add padding in cases where there is not enough data to encrypt to make a full block size. This might result in a very small amount of wasted overhead, as the small padding would be processed by the cipher along with the real data.

Stream Ciphers

A *stream cipher* is a *symmetric* key cipher (same key to encrypt as decrypt), where each bit of plaintext data to be encrypted is done 1 bit at a time against the bits of the key stream, also called a *cipher digit stream*. The resulting output is a ciphertext stream. Because a cipher stream does not have to fit in a given block size, there may be slightly less overhead than a block cipher that is requiring padding to complete a block size.

Symmetric and Asymmetric Algorithms



As you build your vocabulary, the words *symmetric* and *asymmetric* are important ones to differentiate. Let's look at the options of each and identify which of these requires the most CPU overhead and which one is used for bulk data encryption.

Symmetric

As mentioned previously, a *symmetric* encryption algorithm, also known as a *symmetrical cipher*, uses the same key to encrypt the data and decrypt the data. Two devices connected via a VPN both need the key or keys to successfully encrypt and decrypt the data that is protected using a symmetric encryption algorithm. Common examples of symmetric encryption algorithms include the following:

- DES
- 3DES
- AES
- IDEA

- RC2, RC4, RC5, RC6
- Blowfish

Symmetrical encryption algorithms are used for most of the data that we protect in VPNs today. The reason we use symmetrical to encrypt the bulk of our data is because it is much faster to use a symmetrical encryption algorithm and takes less CPU for the same symmetrical encryption algorithm than it would for an asymmetrical algorithm. As with all encryption, the more difficult the key, the more difficult it is for someone who does not have the key to intercept and understand the data. We usually refer to keys with VPNs by their length. A longer key means better security. A typical key length is 40 bits to 256 bits. The minimum key length should be at least 80 bits for symmetrical encryption algorithms to be considered fairly safe. Again, bigger is better.

Asymmetric

An example of an *asymmetric* algorithm is public key algorithms. There is something magical about them. Instead of using the same key for encrypting and decrypting, we use two different keys that mathematically work together as a pair. Let's call these keys the *public key* and *private key*. Together they make a *key pair*. Let's put these keys to use with an analogy.

Imagine a huge shipping container that has a special lock with two keyholes (one large keyhole, and one smaller keyhole). With this magical shipping container, if we use the small keyhole with its respective key to lock the container, the only way to unlock it is to use the big keyhole with its larger key. Another option is to initially lock the container using the big key in the big keyhole, and then the only way to unlock it is to use the small key in the small keyhole. (I told you it was magic). This analogy explains the interrelationship between the public key and its corresponding private key. (I'll let you decide which one you want to call the big key and which one you want to call the little key.) There is a very high CPU cost when using key pairs to lock and unlock data. For that reason, we use asymmetric algorithms sparingly. Instead of using them to encrypt our bulk data, we use asymmetric algorithms for things such as authenticating a VPN peer or generating keying material that we could use for our symmetrical algorithms. Both of these tasks are infrequent compared to encrypting all the user packets (which happens consistently).

One reason this is called *public key cryptography* is that we allow one of these keys to be published and available to anyone who wants to use it (the public key). The other key in the key pair is the private key, and this private key is known only to the device that owns the public-private key pair. An example of using a public-private key pair is visiting a secure website. In the background, the public-private key pair of the server is being used for security of the session. Your PC has access to the public key, and the server is the only one that knows its private key. You learn more about *Secure Sockets Layer (SSL)* later in this chapter, and again in more detail in the *public key infrastructure (PKI)* chapter.

Hashes



Hashing is a method used to verify data integrity. Earlier in this chapter, we looked at a method for verifying the integrity of a downloaded IOS file from Cisco, and the method that was used was a hash.

A cryptographic hash function is a process that takes a block of data and creates a small fixed-sized hash value. It is a one-way function, meaning that if two different computers take the same data and run the same hash function they should get the same fixed-sized hash value (for example, perhaps a 12-bit long hash). (MD5 is an example.) It is a one-way function in that it can be generated from the data over and over again, but it is not possible (at least not realistically) to generate the same hash from a different block of data. The result of the hash is a fixed-length small string of data, and is sometimes referred to as the *digest*, *message digest*, or simply the *hash*.

An example of using a hash to verify integrity is the sender running a hash algorithm on each packet and attaching that hash to the packet. The receiver runs the same hash against the packet and compares his results against the results the sender had (which were attached to the packet, as well). If the hash generated matches the hash that was sent, we know that the entire packet is intact. If a single bit of the hashed portion of the packet is modified, the hash calculated by the receiver will not match, and the receiver will know that the packet had a problem, specifically with the integrity of the packet.

The three most popular types of hashes are as follows:

- **Message digest 5 (MD5):** This creates a 128-bit digest.
- **Secure Hash Algorithm 1 (SHA-1):** This creates a 160-bit digest.
- **Secure Hash Algorithm 2 (SHA-2):** Options include a digest between 224 bits and 512 bits.

With encryption and cryptography, and now hashing, bigger is better, and more bits equals better security.

Hashed Message Authentication Code



Hashed Message Authentication Code (HMAC) uses the mechanism of hashing, but it kicks it up a notch. Instead of using a hash that anyone can calculate, it includes in its calculation a secret key of some type. Then only the other party who also knows the secret key and can calculate the resulting hash can correctly verify the hash. When this mechanism is used, an attacker who is eavesdropping and intercepting packets cannot inject or remove data from those packets without being noticed because he cannot recalculate the correct hash for the modified packet because he does not have the key or keys used for the calculation.

Digital Signatures

When you sign something, it often represents a commitment to follow through, or at least prove that you are who you say you are. In the world of cryptography a digital signature provides three core benefits:

- Authentication
- Data integrity
- Nonrepudiation



Digital Signatures in Action

One of the best ways to understand how a digital signature operates is to remember what you learned in the previous sections about public and private key pairs, hashing, and encryption. Digital signatures involve each of these elements. Here's the play by play. Bob and Lois are two devices that want to establish a VPN connection to each other, and to do so they want to use digital signatures to verify each other to make sure they are talking to the right device. Both the devices want to verify each other, but for simplicity will focus on one device: Bob wanting to prove its identity to the other device Lois. (This could also be phrased as Lois asking Bob to prove Bob's identity.)

As a little set up beforehand, you should know that both Bob and Lois have generated public-private key pairs, and they both have been given digital certificates from a common *certificate authority (CA)*. A CA is a trusted entity that hands out digital certificates (more on that later). If you and I were to open a digital certificate, we would find the name of the entity (for example, Bob). We would find Bob's public key (which Bob gave to the CA when he applied for his digital certificate). There would also be a digital signature of the CA. Both Bob and Lois trust the CA and have both received their certificates. Okay, now back to the story.

Bob takes a packet and generates a hash. Bob then takes this small hash and encrypts it using Bob's private key. (Think of this as a shipping container, and we are using the small key in the small keyhole to lock the data.) We attach this encrypted hash to the packet and send it to Lois. There is a fancy name for this encrypted hash: a *digital signature*.

Lois when she receives this packet looks at the encrypted hash that was sent and she decrypts it using Bob's public key. (Think of this as a big keyhole and the big key being used to unlock the data.) She then sets the decrypted hash off to the side for one moment and she runs the same hash algorithm on the packet she just received. If the hash she just calculated matches the hash she received (after she decrypted it using the sender's public key), she knows two things. She knows the only person who could have encrypted that was Bob with Bob's private key, and that data integrity on the packet is solid, because if 1 bit had changed the hash would not have matched. This process is called *authentication*, using digital signatures, and normally happens in both directions with an IPsec VPN tunnel if the peers are using digital signatures for authentication, referred to as **rsa-signatures** in the configuration.

One might ask, okay so how did Lois get Bob's key (Bob's public key) to begin with? The answer is that Bob and Lois also exchanged digital certificates, which contained each other's public keys. Bob and Lois do not just trust any certificates, but they do trust certificates that are digitally signed by a CA that they trust. This also implies that to verify digital signatures from the CA, both Bob and Lois would also need the CA's public key. Most browsers today have the built-in certificates and public keys for the mainstream CAs on the Internet today.

Key Management

Key management is huge in the world of cryptography. We have symmetric keys that can be used with symmetric algorithms such as hashing and encryption. We have asymmetric keys such as public-private key pairs that can be used with asymmetric algorithms such as digital signatures, among other things. We could say that the key to security with all of these algorithms that we have taken a look at are the keys themselves.

Key management deals with generating keys, verifying keys, exchanging keys, storing keys, and at the end of their lifetime, destroying keys. An example of why this is critical is if two devices that want to establish a VPN session send the encryption keys over at the beginning of their session in plaintext. If that happens, an eavesdropper who sees the keys could go ahead and use them to change cipher text into understandable data, which would result in a lack of confidentiality within the VPN.

Keyspace refers to all the possible key values for a key. The bigger the key, the more secure the algorithm will be. The only negative of having an extremely long key is that the longer the key, the more the CPU is used for the decryption and encryption of data.

IPsec and SSL



IPsec is a suite of protocols used to protect IP packets and has been around for decades. It is in use today for both remote-access VPNs and site-to-site VPNs. SSL is the new kid on the block in its application with remote-access VPNs. Let's take a closer look at both of these options.

IPsec

IPsec is a collection of protocols and algorithms used to protect IP packets at Layer 3 (hence the name of *IP Security [IPsec]*). IPsec provides the core benefits of confidentiality through encryption, data integrity through hashing and HMAC, and authentication using digital signatures or using a *pre-shared key (PSK)* that is just for the authentication, similar to a password. IPsec also provides antireplay support. We take a closer look at IPsec in a later chapter, but here's a good preview of the coming attractions:

- **ESP and AH:** The two primary methods for implementing IPsec. The acronyms stand for *Encapsulating Security Payload (ESP)*, which can do all the features of IPsec, and *Authentication Header (AH)*, which can do many parts of the IPsec objectives, except for the important one of encryption of the data. For that reason, we do not frequently see AH being used.

- **Encryption algorithms for confidentiality:** DES, 3DES, AES.
- **Hashing algorithms for integrity:** MD5, SHA.
- **Authentication algorithms:** Pre-shared keys (PSK), RSA digital signatures.
- **Key management:** An example would be *Diffie-Hellman (DH)*, which can be used to dynamically generate symmetrical keys to be used by symmetrical algorithms. PKI, which supports the function of digital certificates issued by trusted CAs. *Internet Key Exchange (IKE)*, which does a lot of the negotiating and management for us for IPsec to operate.

SSL

Transmitting information over a public network needs to be secured through encryption to prevent unauthorized access to that data. An example is going online to do banking. Not only do you want to avoid an attacker seeing your usernames, passwords, and codes, you also do not want an attacker to be able to modify the packets in transit during a transaction with the bank. It would seem that this would be a perfect opportunity for IPsec to be used to encrypt the data and perform integrity checking and authentication of the server you are connected to. Although it is true that IPsec could do all of this, there is not an IPsec client or software currently running on everybody's computer. Even if there were, not everyone has a digital certificate or a PSK that they could successfully use for authentication.

You can still benefit from the concept of encryption and authentication by using a different type of technology. This additional option is called *Secure Sockets Layer (SSL)*. The convenient thing about SSL is that almost every web browser on every computer supports it, so almost anyone who has a computer can use it.

To use SSL, the user connects to an SSL server, which is a fancy way of saying a web server that supports SSL, by using HTTPS rather than HTTP. An easy way to remember is that the *S* means *secure*. Depending on whom you talk to, SSL may also be labeled as *Transport Layer Security* or *TLS*. To the end user such as you or I, it represents a secure connection to the server, and to the correct server.

Even if the customer does not type in HTTPS, the website could redirect the user behind the scenes to the correct URL. Once there, the browser requests that the web server identifies itself. (Be aware that all of this that is about to happen is occurring in the background and does not require user intervention.) The server sends the browser a copy of its digital certificate, which may also be called an SSL certificate. When the browser receives the certificate, it checks whether it trusts the certificate. The browser decides whether it is trusted by looking at the digital signature of the CA that is on the certificate; using the method for verifying a digital signature discussed earlier, the browser determines the certificate is valid based on the signature of the CA (or is not valid). (If the signature is not valid, or at least if our browser does not think the certificate is valid, a pop-up is usually presented to the user asking whether the user wants to proceed.) This is where user training is important and customers should be trained never to continue or accept a certificate that the browser does not trust.) Assuming the certificate is trusted, the browser now has access to the server's public key contained in the certificate.

Most of the time, the server does not require the browser to prove who it is. Instead, the web server uses some type of user authentication, such as a username or password as required, to verify who the user is.

After the authentication has been done, several additional exchanges occur between the browser and the server as they establish the encryption algorithm they will use and the keys that they will use to encrypt and decrypt the data. You learn more about that exact process in the chapter on PKI.

As mentioned previously, understanding the terminology is important for you in mastering VPN technologies. Table 17-2 describes VPN components, their functions, and examples of their implementation. Some of these terms are a review, whereas others are new. These concepts and their functions are repeated throughout the chapters on these topics to assist you in learning and applying these concepts.

Table 17-2 *VPN Components*

Component	Function	Examples of Use
Symmetrical encryption algorithms	Uses the same key for encrypting and decrypting data.	DES, 3DES, AES, IDEA
Asymmetrical encryption	Uses a public and private key. One key encrypts the data, and the other key in the pair is used to decrypt.	RSA, Diffie-Hellman
Digital signature	Encryption of hash using private key, and decryption of hash with the sender's public key.	RSA signatures
Diffie-Hellman key exchange	Uses a public-private key pair asymmetrical algorithm, but creates final shared secrets (keys) that are then used by symmetrical algorithms.	Used as one of the many services of IPsec
Confidentiality	Encryption algorithms provide this by turning clear text into cipher text.	DES, 3DES, AES, RSA, IDEA
Data integrity	Validates data by comparing hash values.	MD5, SHA-1
Authentication	Verifies the peer's identity to the other peer.	PSKs, RSA signatures

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 17-3 lists these key topics.

Table 17-3 *Key Topics*

Key Topic Element	Description	Page Number
Text	What is a VPN?	426
List	VPN technologies	427
List	Main benefits of VPNs	427
Text	Ciphers and keys	430
Text	Symmetric and asymmetric algorithms	432
Text	Hashes	434
Text	Hashed Message Authentication Code	434
Text	Digital signatures	435
Text	IPsec and SSL	436



Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

VPN, SSL, IPsec, 3DES, AES, MD5, SHA1, hash, HMAC, digital signature, symmetrical, asymmetrical, key



This chapter covers the following subjects:

- Public key infrastructure
- Putting the pieces of PKI to work

Fundamentals of the Public Key Infrastructure

The *public key infrastructure (PKI)* is a combination of policies, procedures, hardware, software, and people that are required to create, manage, and revoke (when necessary) digital certificates. Similar to the concept of how a driver of a car can receive a license to drive, and that license because it is issued from a trusted agency can be used by others to validate the identity of the person who has the license, an identity digital certificate, issued by a trusted *certificate authority (CA)*, can provide similar functionality in the digital world of networking.

On a bigger scale, digital certificates can be used for authenticating a specific user, system, or server, and going beyond that, the certificate can also be used as part of the protocols to establish an encrypted private session.

We have been using the services of the PKI for years on the Internet, and in this chapter, we delve into the details of the components that make it up and how they interoperate.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 18-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 18-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Public Key Infrastructure	1–8
Putting the Pieces of PKI to Work	9–10

1. Why is the public key in a typical public-private key pair referred to as public?
 - a. Because the public already has it.
 - b. Because it is shared publicly.
 - c. Because it is a well-known algorithm that is published.
 - d. The last name of the creator was publica, which is Latin for public.

2. What is the key component used to create a digital signature?
 - a. Ink
 - b. Public key
 - c. Private key
 - d. AES
3. What is the key component used to verify a digital signature?
 - a. Sender's public key
 - b. Receiver's public key
 - c. AES
 - d. One-time PAD
4. What is another name for a hash that has been encrypted with a private key?
 - a. MD5
 - b. SHA-1
 - c. AES
 - d. Digital signature
5. What are the primary responsibilities for a certificate authority (CA)? (Choose all that apply.)
 - a. Verification of certificates
 - b. Issuing identity certificates
 - c. Maintaining client's private keys
 - d. Tracking identity certificates
6. Which of the following is *not* a way for a client to check to see whether a certificate has been revoked?
 - a. Look at the lifetime of the certificate itself
 - b. CRL
 - c. OSCP
 - d. LDAP

7. Which of the following could be found in a typical identity certificate? (Choose all that apply.)
 - a. CRL locations
 - b. Validity date
 - c. Public key of the certificate owner
 - d. Serial number
8. Which standard format is used to request a digital certificate from a CA?
 - a. PKCS#7
 - b. PKCS#10
 - c. LDAP
 - d. TLS/SSL/HTTPS
9. When obtaining the initial root certificate, what method should be used for validation of the certificate?
 - a. Sender's public key
 - b. Telephone
 - c. HTTPS/TLS/SSL
 - d. Receiver's private key
10. Which method, when supported by both the client and the CA, is the simplest to use when implementing identity certificates on the client?
 - a. PKCS#7
 - b. PKCS#10
 - c. SCEP
 - d. LDAP

Foundation Topics

Public Key Infrastructure

This section covers the moving parts and pieces involved with the public key infrastructure. This section presumes that you've read the previous chapter regarding *virtual private network (VPN)* technologies.

Public and Private Key Pairs



A *key pair* is a set of two keys that work in combination with each other as a team. In a typical key pair, you have one public key and one private key. The public key may be shared with everyone, and the private key is not shared with anyone. For example, the private key for a web server is known only to that specific web server. If you use the public key to encrypt data using an asymmetric encryption algorithm, the corresponding private key is used to decrypt the data. The inverse is also true. If you encrypt with the private key, you then decrypt with the corresponding public key. Another name for this asymmetric encryption is *public key cryptography* or *asymmetric key cryptography*. The uses for asymmetric algorithms are not limited to only authentication as in the case of the digital signature discussed in the previous chapter, but that is one example of an asymmetrical algorithm. Examples of asymmetrical algorithms include the following:

- **RSA:** Named after Rivest, Shamir, and Adleman, who created the algorithm. The primary use of this asymmetrical algorithm today is for authentication. It is also known as *public key cryptography standard (PKCS) #1*. The key length may be from 512 to 2048, and a minimum size for good security is at least 1024. Regarding security, bigger is better.
- **DH:** Diffie-Hellman key exchange protocol. DH is an asymmetrical algorithm that allows two devices to negotiate and establish shared secret keying material (keys) over an untrusted network. The interesting thing about DH is that although the algorithm itself is asymmetrical, the keys generated by the exchange are symmetrical keys that can then be used with symmetrical algorithms such as *Triple Digital Encryption Standard (3DES)* and *Advanced Encryption Standard (AES)*.
- **ElGamal (second character is an L):** This asymmetrical encryption system is based on the DH exchange.
- **DSA:** Digital Signature Algorithm was developed by the U.S. National Security Agency.
- **ECC:** Elliptic Curve Cryptography.

Asymmetrical algorithms require more CPU processing power than a symmetrical algorithm. Asymmetrical algorithms, however, are more secure. A typical key length used in asymmetrical algorithms can be anywhere between 512 to 4096. A key length that is shorter than 1024 is considered unreliable or not as secure as a longer key.

A commonly used asymmetrical algorithm used for authentication is RSA (as in RSA digital signatures).

RSA Algorithm, the Keys, and Digital Certificates

Keys are the secrets that allow cryptography to provide confidentiality. Let's take a closer look at the keys involved with RSA and how they are used.



Who Has Keys and a Digital Certificate?

With RSA digital signatures, with both parties intending on authenticating the other side, each party has a public-private key pair. Going back to the analogy in the previous chapter let's use two computers named Bob and Lois. They both generated their own public-private key pair, and they both enrolled with a *certificate authority (CA)*. That certificate authority took each of their public keys and their names and IP addresses and created individual digital certificates, and the CA issued these certificates back to Bob and Lois, respectively. The CA also digitally signed each certificate.

How Two Parties Exchange Public Keys

When Bob and Lois want to authenticate each other, they send each other their digital certificates (or least a copy of them). Upon receiving the other party's digital certificate, they both verify the authenticity of the certificate by checking the signature of a certificate authority that they currently trust. (When you talk about trusting a certificate authority, it really means that you know who the CA is and can verify that certificate authority's digital signature, by knowing the public key of that CA.)

Now that Bob and Lois both have each other's public keys they can authenticate each other. This normally happens inside of a *virtual private network (VPN)* tunnel in both directions (when RSA signatures are used for authentication). For the purpose of clarity, we focus on just one of these parties (for example, the computer Bob) proving its identity to computer Lois.

Creating a Digital Signature

Bob takes some data, generates a hash, and then encrypts the hash with Bob's private key. (Note that the private key has not been shared with anyone else; not even Bob's closest friends have it.) This encrypted hash is attached to the packet and sent to Lois. This encrypted hash is Bob's digital signature.

Lois, having received the packet with the digital signature attached, first decodes or decrypts the encrypted hash using Bob's public key. She sets the decrypted hash to the side for a moment and runs a hash against the same data that Bob did previously. If the hash that Lois generates matches the decrypted hash, which was sent as a digital signature from Bob, she has just authenticated Bob. The reason is because only Bob has the private key used for the creation of his digital signature.

Certificate Authorities



A certificate authority is a computer or entity that creates and issues digital certificates. Inside of a digital certificate is information about the identity of a device, such as its IP address, *fully qualified domain name (FQDN)*, and the public key of that device. The CA takes requests from devices who supply all of that information (including the public key generated by the computer who is making the request) and generates a digital certificate, which the CA assigns a serial number to and signs the certificate with its own digital signature (the CA's signature). Also included in the final certificate is a URL that other devices can check to see whether this certificate has been revoked and the validity dates for the certificate (which is similar to the expiration date of food products). Also in the certificate is the information about the CA that issued the certificate and several other parameters used by PKI.

By using a third-party trusted certificate authority, the computers Bob and Lois can receive and verify identity certificates from each other (and thousands of others), as long as the certificates are signed by a CA that is trusted by Bob and Lois. Commercial CAs charge a fee to issue and maintain digital certificates. One benefit of using a commercial CA server to obtain digital certificates for your devices is that most web browsers maintain a list of the more common trusted public CA servers, and as a result anyone using a browser can verify the identity of your web server by default without having to modify the web browser at all. If a company wants to set up their own internal CA and then configure each of the end devices to trust the certificates issued by their internal CA, no commercial certificate authority is required, but the scope of that CA is limited to the company and its managed devices, because any devices outside of the company would not trust the company's internal CA by default.

Root and Identity Certificates



A digital certificate can be thought of as an electronic document that identifies a device or person. It includes information such as the name of a person or organization, their address, and the public key of that person or device. There are different types of certificates, including root certificates (which identify the CA), and identity certificates, which identify devices such as servers and other devices that want to participate in PKI.

Root Certificate

A *root certificate* contains the public key of the CA server and the other details about the CA server. Figure 18-1 shows an example of one.

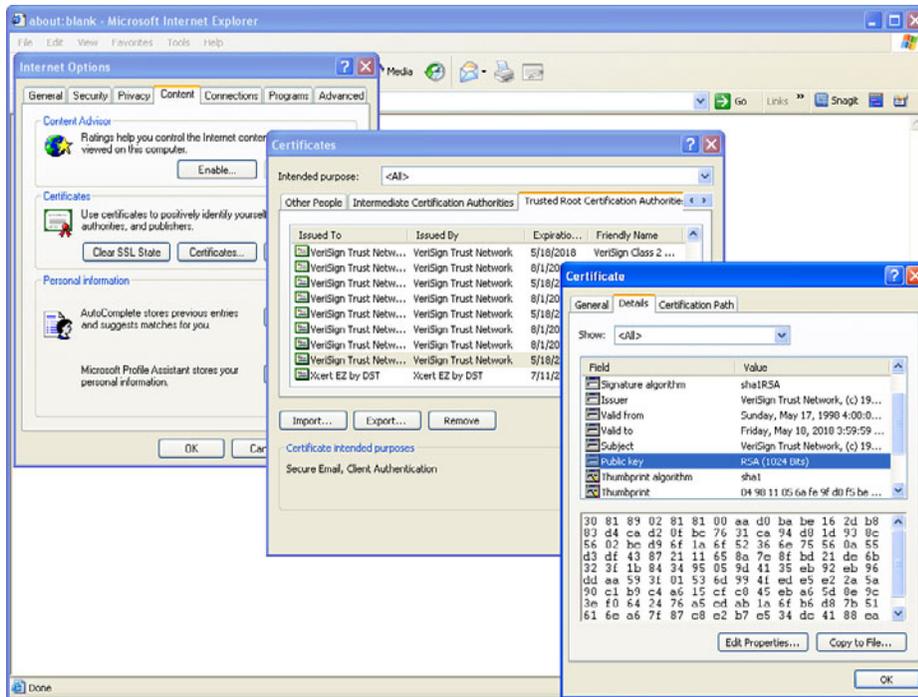


Figure 18-1 Root Certificate Details

The output in Figure 18-1 can be seen on most browsers, although the location might differ a bit depending on your browser vendor and version. In Figure 18-1, I used Internet Explorer 6.x, and the navigation to see the root certificates is **Tools > Internet Options > Content > Certificates > Trusted Root Certification Authorities**. After highlighting a CA, click the **View** button, then the **Details** button to see the details. I recommend taking a moment to do this the next time you are at a computer.

I recommend knowing the relevant parts of the certificate, including the following:

- **Serial number:** Issued and tracked by the CA that issued the certificate.
- **Issuer:** The CA that issued this certificate (Even root certificates need to have their certificates issued from someone (perhaps even themselves).
- **Validity dates:** The time window during which the certificate may be considered valid. If a local computer believes the date to be off by a few years, that same PC may consider the certificate invalid due to his own error about the time. Using *Network Time Protocol (NTP)* is a good idea to avoid this problem.
- **Subject of the certificate:** This includes the *Organizational Unit (OU)*, *Organization (O)*, *Country (C)*, and other details commonly found in an X.500 structured directory (more on that later in the chapter). The subject of the root certificate is the CA itself. The subject for a client's identity certificate is the client.



- **Public key:** The contents of the public key and the length of the key are often both shown. After all, the public key is public.
- **Thumbprint algorithm and thumbprint:** This is the hash for the certificate. On a new root certificate, you could use a phone to call and ask for the hash value and compare it to the hash value you see on the certificate. If it matches, you have just performed out-of-band (using the telephone) verification of the digital certificate.

Identity Certificate

An *identity certificate* is similar to a root certificate, but it describes the client and contains the public key of an individual host (the client). An example of a client is a web server that wants to support *Secure Sockets Layer (SSL)* or a router that wants to use digital signatures for authentication of a VPN tunnel. Figure 18-2 shows an example of an identity certificate. We walk through the process for installing an identity certificate on a client later in this chapter. (In this figure, it is an ASA Firewall.)

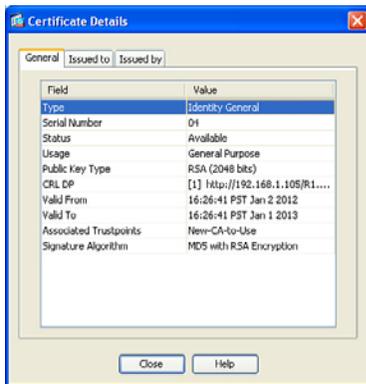


Figure 18-2 Identity Certificate

Using the Digital Certificates to get the Peer's Public Key

In its basic components, any device that wants to verify a digital signature must have the public key of the sender. So, let's use an example of you and I. If we want to authenticate each other, and we both trust a common CA and have previously requested and received digital certificates (identity certificates) from the CA server, we exchange our identity certificates, which contain our public keys. We both verify the CA's signature on the digital certificate we just received from each other using the public key of the CA. In practice, this public key for the CA is built in to most of our browsers today for public CA servers. Once we verify each other's certificates, we can then trust the contents of those certificates (and most important, the public key). Now that you and I both have each other's public key, we can use those public keys to verify each other's digital signatures.

X.500 and X.509v3 Certificates

X.500 is a series of standards focused on directory services and how those directories are organized. Many popular network operating systems have been based on X.500, including Microsoft Active Directory. This X.500 structure is the foundation from which you see common directory elements such as CN=Bob (Common Name = CN), OU=engineering (Organizational Unit = OU), O=cisco.com (Organization = O), and so on in an “org-chart” way, shaped like a pyramid. X.509 Version 3 is a standard for digital certificates that is widely accepted and incorporates many of the same directory and naming standards. A common protocol that is used to do lookups from a directory is called *Lightweight Directory Access Protocol (LDAP)*. A common use for this is having a digital certificate being used for authentication, and then based on the details for that certificate (for example, the OU=sales in the certificate itself), the user could be dynamically assigned the access rights that are associated with that group in Active Directory or some other LDAP accessible database. The concept is to define the rights in one place, and then leverage that over and over again. An example is setting up Active Directory for the network and then using that to control what access is provided to each user after he or she authenticates.



As a review, most digital certificates contain the following information:

- **Serial number:** Assigned by the CA and used to uniquely identify the certificate
- **Subject:** The person or entity that is being identified
- **Signature algorithm:** The specific algorithm that was used for signing the digital certificate
- **Signature:** The digital signature from the certificate authority, which is used by devices that want to verify the authenticity of the certificate issued by that CA
- **Issuer:** The entity or CA that created and issued the digital certificate
- **Valid from:** The date the certificate became valid
- **Valid to:** The expiration date of the certificate
- **Key usage:** The functions for which the public key in the certificate may be used
- **Public key:** The public portion of the public and private key pair generated by the host whose certificate is being looked at
- **Thumbprint algorithm:** The hash algorithm used for data integrity
- **Thumbprint:** The actual hash
- **Certificate revocation list location:** The URL that can be checked to see whether the serial number of any certificates issued by the CA have been revoked



Authenticating and Enrolling with the CA



If you want to use a new CA as a trusted entity, and want to request and receive your own identity certificate from this CA, it is really a two-step process:

Step 1. The first step is to authenticate the CA server, or in other words trust the CA server. Unfortunately, if you do not have the public key for a CA server, you cannot verify the digital signature of the CA server. This is sort of like the chicken and the egg story, because you need the public key, which can be found in the root's CA certificate, but you cannot verify the signature on a certificate until you have the public key.

To get the ball rolling, you could download the root certificate, and then use an out-of-band method, such as making a telephone call, to validate the root certificate. This can be done after downloading the root certificate and looking at the hash value, calling the administrators for the root CA and asking them to verbally tell you what the hash is. If the hash that they tell you over the phone matches the hash that you see on the digital certificate (and assuming that you called the right phone number and talked with the right people), you then know that the certificate is valid, and you can then use the public key contained in a certificate to verify future certificates which are signed by that CA. This process of getting the root CA certificate installed is often referred to as authenticating the CA.

Step 2. After you have authenticated the root CA and have a known good root certificate for that CA, you can then request your own identity certificate. This involves generating a public-private key pair and including the public key portion in any requests for your own identity certificate. An identity certificate could be for a device or person. Once you make this request, the CA can take all of your information and generate an identity certificate for you, which includes your public key, and then send this certificate back to you. If this is done electronically, how do you verify the identity certificate you got is really from the CA server that you trust? The answer is simple because the CA has not only issued the certificate but it also signed the certificate. Because you authenticated the CA server earlier and you have a copy of its digital certificate with its public key, you can now verify the digital signature it has put on your own identity certificate. If the signature from the CA is valid, you also know that your certificate is valid and so you can install it and use it.

Public Key Cryptography Standards



There are lots of standards in use for the PKI. Many of them have *Public Key Cryptography Standards (PKCS)* numbers. Some of these standards control the format and use of certificates, including requests to a CA for new certificates, the format for a file that is going to be the new identity certificate, and the file format and usage access for certificates. Having the standards in place helps with interoperability between different CA servers and many different CA clients.

Here are a few standards you should become familiar with, which include protocols by themselves and protocols used for working with digital certificates:

- **PKCS #10:** This is a format of a certificate request sent to a CA who wants to receive their identity certificate. This type of request would include the public key for the entity desiring a certificate.
- **PKCS #7:** This is a format that can be used by a CA as a response to a PKCS#10 request. The response itself will very likely be the identity certificate (or certificates) that had been previously requested.
- **PKCS#1:** RSA Cryptography Standard.
- **PKCS#12:** A format for storing both public and private keys using a symmetric password-based key to “unlock” the data whenever the key needs to be used or accessed.
- **PKCS#3:** Diffie-Hellman key exchange.

Simple Certificate Enrollment Protocol

The process of authenticating a CA server, generating a public-private key pair, requesting an identity certificate, and then verifying and implementing the identity certificate can be a several-step process. Cisco, in association with a few other vendors, developed the *Simple Certificate Enrollment Protocol (SCEP)*, which can automate most of the process for requesting and installing an identity certificate. Although it is not an open standard, it is supported by most Cisco devices and makes it convenient to get and install both root and identity certificates, as you see in action later in this chapter.



Revoked Certificates

If you decommission a device that has been assigned an identity certificate, or if the device assigned a digital certificate has been compromised and you believe that the private key information is no longer “private,” you could request from the CA that the previously issued certificate be revoked. This poses a unique problem. Normally when two devices authenticate with each other, they do not need to contact a CA to verify the identity of the other party. This is because the two devices already have the public key of the CA and can validate the signature on a peer’s certificate without direct contact with the CA. So here’s the challenge: If a certificate has been revoked by the CA, and the peers are not checking with the CA each time they try to authenticate the peers, how does a peer know whether the certificate it just received has been revoked? The answer is simple: It is to check and see. A digital certificate contains information on where an updated list of revoked certificates can be obtained. This URL could point to the CA server itself or to some other publicly available resource on the Internet. The revoked certificates are listed based on the serial number of the certificates, and if a peer has been configured to check for revoked certificates, it adds this check before completing the authentication with a peer. If a *certificate revocation list (CRL)* is checked, and the certificate from the peer is on that list, the authentication stops at that moment. The three basic ways to check whether certificates have been revoked are as follows, in order of popularity:



- **Certificate Revocation List (CRL):** This is a list of certificates, based on their serial numbers, that had initially been issued by a CA but have since been revoked and as a result should not be trusted. A CRL could be very large, and the client would have to process the entire list to verify the certificate is not on the list. A CRL can be thought of as the naughty list. This is the primary protocol used for this purpose, compared to OSCP and AAA. A CRL could be accessed by several protocols, including LDAP and HTTP. A CRL could also be obtained via SCEP.
- **Online Certificate Status Protocol (OCSP):** This is an alternative to CRLs. Using this method, a client simply sends a request to find the status of a certificate and gets a response without having to know the complete list of revoked certificates.
- **Authentication, authorization, and accounting (AAA):** Cisco AAA services also provide support for validating digital certificates, including a check to see whether a certificate has been revoked. Because this is a proprietary solution, this is not often used in PKI.

Uses for Digital Certificates

Digital certificates aren't just for breakfast anymore. They can be used for clients who want to authenticate a web server to verify they are connected to the correct server using *HTTP Secure (HTTPS)*, *Transport Layer Security (TLS)*, or *Secure Sockets Layer (SSL)*. For the average user who does not have to write these protocols, but simply benefits from using them, they are all effectively the same, which is HTTP combined with TLS/SSL for the security benefits. This means that digital certificates can be used when you do online banking from your PC to the bank's website. It also means that if you use SSL technology for your remote-access VPNs that you can also use digital certificates for authenticating the peers (at each end) of the VPN.

You can also use digital certificates with the protocol family of IPsec, which can also use digital certificates for the authentication portion.

Digital certificates can also be used with protocols such as 802.1X, which involves authentication at the edge of the network before allowing the user's packets and frames to progress through the network. An example is a wireless network, controlling access and requiring authentication, using digital certificates for the PCs/users, before allowing them in on the network.

PKI Topologies



There is not a one-size-fits-all solution for PKI. In small networks, a single CA server may be enough, but in a network with 30,000 devices, a single server may not provide the availability and fault tolerance required. To answer these issues, let's investigate the options available to us for implementation of the PKI, using various topologies, including single and hierarchical. Let's start off with the single CA and expand from there.

Single Root CA

If you have one trusted CA, and you have tens of thousands of customers who want to authenticate that CA and request their own identity certificates, there might be too large of a demand on a single server even though a single CA does not have to be directly involved in the day-to-day authentication that happens between peers. To offload some of the workload from a single server, you could publish CRLs on other servers. At the end of the day, it still makes sense to have at least some fault tolerance for your PKI, which means more than just a single root CA server.

Hierarchical CA with Subordinate CAs

One of our options to support fault tolerance and increased capacity is to use intermediate or subordinate CAs to assist the root CA. The root CA is the king of the hill. The root CA delegates the authority (to the subordinate CAs) to create and assign identity certificates to clients. This is called a *hierarchical PKI topology*. The root CA signs the digital certificates of its subordinate or intermediate CAs, and the subordinate CAs are the ones to issue certificates to clients. For a client to verify the “chain” of authority, a client needs both the subordinate CAs certificate and the root certificate. The root certificate (and its public key) is required to verify the digital signature of the subordinate CA, and the subordinate CA’s certificate (and its public key) is required to verify the signature of the subordinate CA. If there are multiple levels of subordinate CAs, a client needs the certificates of all the devices in the chain from the root all the way to the CA that issued the client’s certificate.

Cross-Certifying CAs

Another approach to hierarchical PKIs is called *cross-certifying*. With cross-certification, you could have a CA with a horizontal trust relationship over to a second CA so that clients of either CA could trust the signatures of the other CA.

Putting the Pieces of PKI to Work

This section covers how to implement these components in an actual production network.

We have taken a look at the ingredients for the recipe called PKI, the public key infrastructure. What I want to do now is walk you through an example of applying these concepts to some devices you are already familiar with if you have read the previous portions of this book. Both the *Adaptive Security Appliance (ASA)* and Cisco routers can use digital certificates. Let’s take a look at installing digital certificates on the ASA, using the *Adaptive Security Device Manager (ASDM)*, shown in Figure 18-3.



Figure 18-3 Using the ASDM to Manage the ASA Firewall

Default of the ASA

The ASA is going to use a self-signed digital certificate by default. It needs this to support an administrator connecting to the firewall to support the ASDM, and for the ability to support any SSL VPN clients that you will be configuring in a later chapter. The problem with a self-signed certificate is that no browsers or other devices will have the ASA listed as a trusted CA, and HTTPS connections to the ASA, such as an administrator who wants to run ASDM, will receive a warning message that the certificate is not trusted.

If you do not want to use a self-signed certificate, but instead want to use a certificate from a CA server on the Internet, you must to install a root certificate (of the CA you are going to trust), and then request an identity certificate.

Viewing the Certificates in ASDM

In the Device Management section of ASDM, you have options for configuring and viewing both identity certificates and root certificates. You could also find these options under the VPN sections of the Configuration menu, as well. Figure 18-4 shows this section of the ASDM.

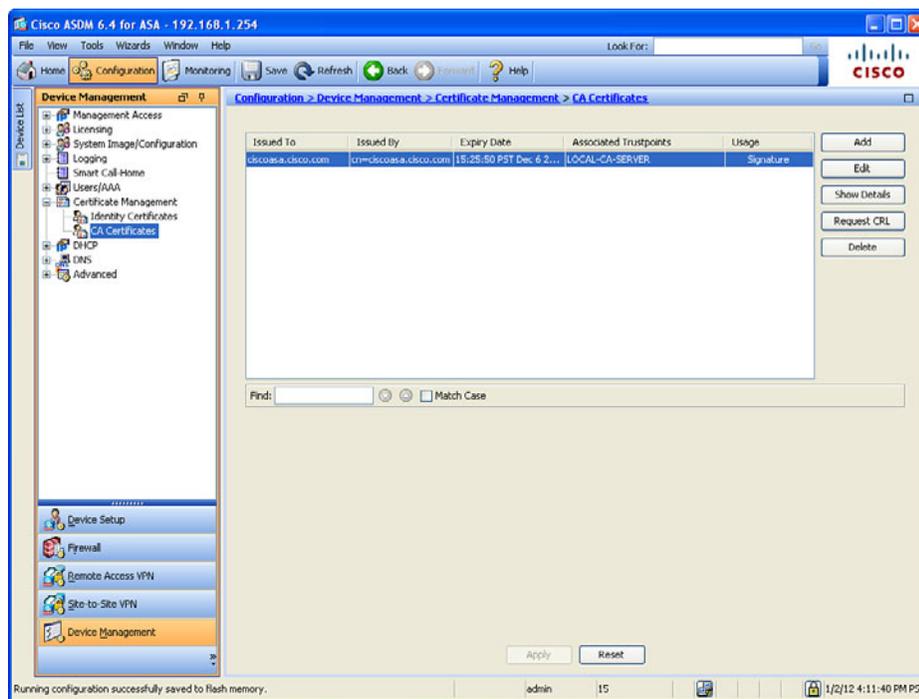


Figure 18-4 Viewing Current CA Certificates

Adding a New Root Certificate

If you want to add a new root certificate, click **Add**, and then you have options to install a root certificate from a file, or you could paste in the information or use SCEP. If you want to use the manual method from a file or through cut and paste, your CA vendor provides the file or instructions for obtaining the file for its root CA certificate. In this example, I have a CA that supports SCEP, so that is the option I choose, as shown in Figure 18-5.

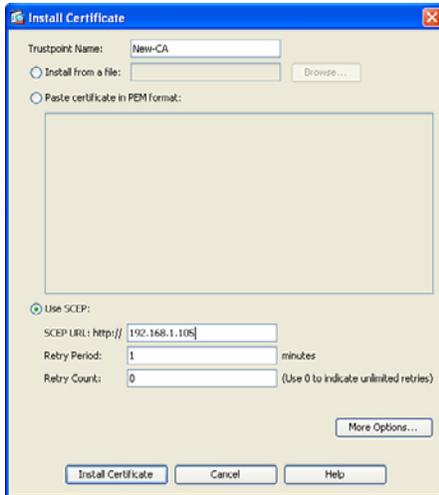


Figure 18-5 Adding a New Root Certificate

When you add a new root certificate, you are also adding details about how you are going to work with that CA. By clicking the **More Options** button, you can answer questions about the CRL and specify other details about which protocols to be used for certificate verification for this firewall to use when dealing with certificates issued by this CA, as shown in Figure 18-6.

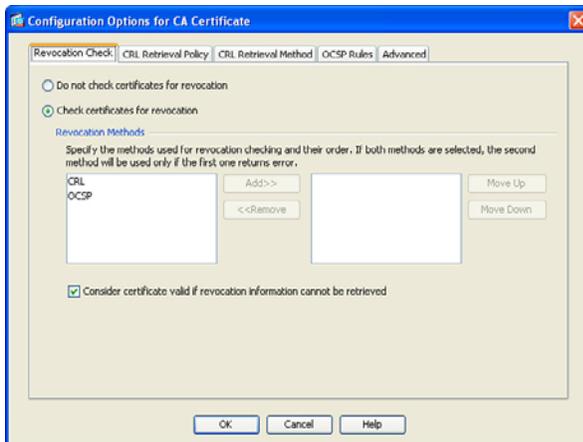


Figure 18-6 Additional Options Related to the CA

After you install a root certificate, and verify it is valid by calling the CA and comparing the hash they give you against the hash for the certificate installed, you then can request your own identity certificate and follow a similar process to install it.

Easier Method for Installing Both Root and Identity certificates

An easier option than manually installing the root certificate file is to use SCEP and install the root certificate, generate a new key pair, and request your identity certificate all using SCEP.

For this, you could begin in the Identity Certificate area in ASDM. Click **Add**, assign a name you want to associate with the new CA, and then click the **Add a New Identity Certificate** radio button. From here, if you want to use a brand new key pair in conjunction with this CA (that will be putting your public key into your digital certificate), click **New** (next to the key pair option), assign the key pair a name and the size of the key to use, and then click the **Generate Now** button, as shown in Figure 18-7.



Figure 18-7 *Generating a New Key Pair*

After you click **Generate Now**, a public private key pair is generated and the public key portion of it is sent to the CA as part of the SCEP certificate request process. The equivalent CLI command that could be used to generate the new key pair is shown in Example 18-1.

Example 18-1 *Generating a New Key Pair*

```
Keith-asal(config)# crypto key generate rsa label My-Key-Pair modulus 2048
noconfirm
```



After the key pair is generated, and before clicking the **Add Certificate** button, you can specify the details of the CA server and how to reach that CA server by clicking the **Advanced** button, as shown in Figure 18-8.



Figure 18-8 *Specifying What Key Pair to Use and the Option of Advanced*

Using the options presented by the Advanced button, you can specify the enrollment mode of SCEP and the IP address of the CA server that supports SCEP, as shown in Figure 18-9.

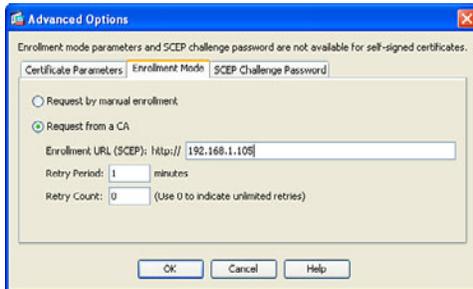


Figure 18-9 *Specifying the Enrollment Mode of SCEP*

Once the enrollment method and IP address are configured, click the **OK** button, and then click the **Add Certificate** button.

Example 18-2 shows the equivalent CLI commands to authenticate and enroll with a new CA via SCEP.



Example 18-2 *Authenticating and Enrolling with a New CA via SCEP*

```
! Create the name that you want the ASA to reference the CA by
Keith-asal(config)# crypto ca trustpoint New-CA-to-Use

! Specify which key-pair will be used for the public portion that will go
! into the digital certificate. Below the new key pair we created is specified.
Keith-asal(config-ca-trustpoint)# keypair New-Key-Pair

! Specify what the certificate may be used for. (Both SSL and IPsec)
Keith-asal(config-ca-trustpoint)# id-usage ssl-ipsec
```

```

! Specify whether or not the fully qualified domain name (fqdn) will be
! required
Keith-asal(config-ca-trustpoint)# no fqdn

! Specify the x.500 common name (CN)
Keith-asal(config-ca-trustpoint)# subject-name CN=ciscoasa

! Specify where the CA server can be reached.  HTTP must be running on the
! CA server.
Keith-asal(config-ca-trustpoint)# enrollment url http://192.168.1.105
Keith-asal(config-ca-trustpoint)# exit

! Retrieve and install the root certificate.  The "nointeractive" won't
! prompt the user for additional information.
Keith-asal(config)# crypto ca authenticate New-CA-to-Use nointeractive

! Request and install the identity certificate from the CA.  The "noconfirm"
! will avoid prompting the user for additional confirmation messages.
Keith-asal(config)# crypto ca enroll New-CA-to-Use noconfirm

```

If the SCEP-capable CA server is reachable, and configured correctly, a success message appears, as shown in Figure 18-10.

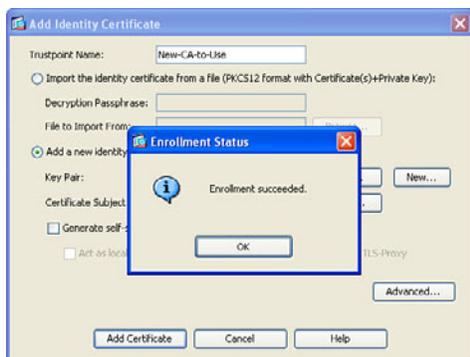


Figure 18-10 *Enrollment Succeeded Message*

To see the details of the new certificate, highlight the certificate and click **Show Details**, as shown in Figure 18-11.

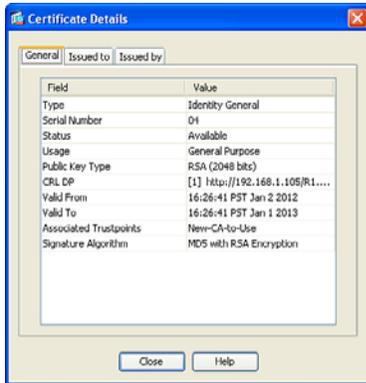


Figure 18-11 *Details of the Identity Certificate*

Notice the details that are found in most digital certificates, including serial number, CRL, and the validity dates.

Note The serial number on a certificate from a public PKI server is much longer than the one issued by my internal CA server used in my lab.

We have discussed the concepts of using PKI, and also taken a look at an example using an ASA. To make sure you understand all components of PKI, review Table 18-2.

Table 18-2 *Key PKI Components*

Component	Description
RSA digital signatures	Using its private key to encrypt a generated hash, a digital signature is created. The receiver uses the public key of the sender to validate the digital signature and verify the identity of the peer.
Digital certificate	File that contains the public key of the entity, a serial number, and the signature of the CA that issued the certificate
Public and private keys	Used as a pair to encrypt and decrypt data in an asymmetrical fashion.
Certificate authority	The CA's job is to fulfill certificate requests and generate the digital certificates for its clients to use. It also maintains a list of valid certificates that have been issued, and maintains a CRL listing any revoked certificates.
X.509v3	A common certificate format used today.
Subordinate CA/RA	Assistant to the CA, which can issue certificates to clients. Clients need both the certificates from the root and the subordinate to verify signatures all the way to the root. Used in a hierarchal PKI topology.
PKCS	Public Key Cryptography Standards, agreed to and implemented by vendors who want the ability to have compatibility with other devices in the PKI.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 18-3 lists these key topics.



Table 18-3 *Key Topics*

Key Topic Element	Description	Page Number
Text	Public and private key pairs	444
Text	RSA algorithm, the keys, and digital certificates	445
Text	Certificate authorities	446
Text	Root and identity certificates	446
List	Certificate components	447
Text	X.500 and X.509v3 certificates	449
List	What goes in to a digital certificate	449
Text	Authenticating and enrolling with the CA	450
Text	PKCS standards	450
Text	SCEP	451
Text	Revoked certificates	451
Text	PKI topologies	452
Example 18-1	Commands to generate key pairs	457
Example 18-2	Commands to authenticate and enroll with CA using SCEP	458
Table 18-2	Key components for PKI	461

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

PKI, CA, subordinate CA, root certificate, identity certificate, PKCS#7, PKCS#12, RSA, digital signature, public key, X.509v3, CRL, SCEP, LDAP

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 18-4 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 18-4 *Command Reference*

Command	Description
<code>crypto key generate rsa</code>	Generate a public/private key pair on the ASA
<code>crypto ca authenticate</code>	Retrieve and installs the root certificate via SCEP
<code>crypto ca enroll</code>	Request and installs an identity certificate via SCEP



This chapter covers the following subjects:

- IPsec concepts, components, and operations
- Configuring and verifying IPsec

Fundamentals of IP Security

IP Security (IPsec) has been around for a long time, and that is quite a good trick for any technology in today's fast-moving world. The secret of IPsec is that it is not locked in to one specific protocol or even one set of protocols. As technology advances, so can the protocols that are being used by IPsec. The goal of IPsec is quite simple: to provide confidentiality, data integrity, and authentication of the *virtual private network (VPN)* peer and provide antireplay support. It implements all of these to Layer 3 packets individually, protecting each one as they are sent from one end of the VPN tunnel until they reach the other end.

This chapter presumes that you have read the previous two chapters, and we build based on that.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter's topics before you begin. Table 19-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 19-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
IPsec Concepts, Components, and Operations	1–10
Configuring and Verifying IPsec	11

1. Which technology is a primary method that IPsec uses to implement data integrity?
 - a. MD5
 - b. AES
 - c. RSA
 - d. DH
2. What are the source and destination addresses used for an encrypted IPsec packet?
 - a. Original sender and receiver IP addresses
 - b. Original sender's and outbound VPN gateway's addresses

- c. Sending and receiving VPN gateways
 - d. Sending VPN gateway and original destination address in the packet
- 3. Which tunnel is used for private management traffic between the two VPN peers?
 - a. IPsec
 - b. IKE Phase 1
 - c. IKE Phase 2
 - d. IKE Phase 3
- 4. Which of the following are negotiated during IKE Phase 1?
 - a. Hashing
 - b. DH group
 - c. Encryption
 - d. Authentication method
- 5. What method is used to allow two VPN peers to establish shared secret keys and to establish those keys over an untrusted network?
 - a. AES
 - b. SHA
 - c. RSA
 - d. DH
- 6. Which of the following is *not* part of the IKE Phase 1 process?
 - a. Negotiation of the IKE phase 1 protocols
 - b. Running DH
 - c. Authenticating the peer
 - d. Negotiating the transform set to use
- 7. How is the negotiation of the IPsec (IKE Phase 2) tunnel done securely?
 - a. Uses the IKE Phase 1 tunnel
 - b. Uses the IPsec tunnel
 - c. Uses the IKE Phase 2 tunnel
 - d. Uses RSA
- 8. What are the two main methods for authenticating a peer as the last step of IKE Phase 1? (Choose all that apply.)
 - a. RSA signatures, using digital certificates to exchange public keys
 - b. PSK (pre-shared key)

- c. DH Group 2
 - d. TCP three-way handshake
9. Which component acts as an if-then statement, looking for packets that should be encrypted before they leave the interface?
- a. `crypto isakmp policy`
 - b. `crypto map`
 - c. `crypto ipsec transform-set`
 - d. `crypto access-list` (access list used for cryptography)
10. What is true about symmetrical algorithms and symmetrical crypto access lists used on VPN peers?
- a. Symmetrical algorithms use the same secret (key) to lock and unlock the data. Symmetrical ACLs between two VPN peers should symmetrically swap the source and destination portions of the ACL.
 - b. Symmetrical algorithms like RSA use the same secret (key) to lock and unlock the data. Symmetrical ACLs between two VPN peers should symmetrically swap the source and destination portions of the ACL.
 - c. Symmetrical algorithms use the same secret (key) to lock and unlock the data. Symmetrical ACLs between two VPN peers should be identical.
 - d. Symmetrical algorithms use the same secret (key) to lock and unlock the data. Symmetrical ACLs between two VPN peers require that only symmetrical algorithms be used for all aspects of IPsec.
11. Which one of the following commands reveal the ACLs, transform sets, and peer information and indicate which interface is being used to connect to the remote IPsec VPN peer?
- a. `show crypto map`
 - b. `show crypto isakmp policy`
 - c. `show crypto config`
 - d. `show crypto ipsec sa`

Foundation Topics

IPsec Concepts, Components, and Operations

This section examines the moving parts and pieces involved with IPsec. This section presumes that you've read the previous two chapters on VPN technologies and *public key infrastructure (PKI)*.

The Goal of IPsec



To best understand how IPsec operates, let's take a look at a simple topology that we can use as a framework for this entire chapter, shown here in Figure 19-1.

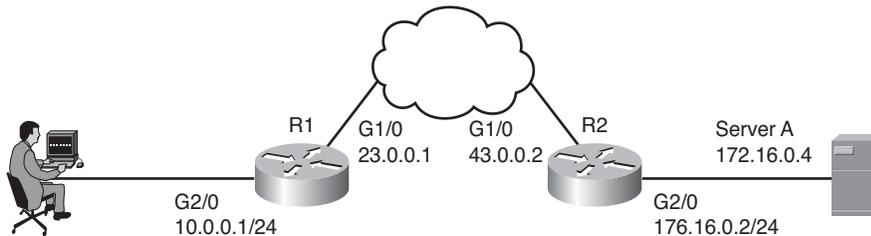


Figure 19-1 IPsec Topology Used for This Discussion

IPsec has four fundamental goals, as shown in Table 19-2.

Table 19-2 IPsec Goals and the Methods Used to Implement Them

Goal	Method That Provides the Feature
Confidentiality	Encryption
Data integrity	Hashing
Peer authentication	Pre-shared keys, RSA digital signatures
Antireplay	Integrated into IPsec, basically applying serial numbers to packets

The goals can be described as follows:

- **Confidentiality:** Provided through encryption changing clear text into cipher text.
- **Data integrity:** Provided through hashing and or through *Hashed Message Authentication Code (HMAC)* to verify that data has not been manipulated during its transit across the network.
- **Authentication:** Provided through authenticating the VPN peers near the beginning of a VPN session using *pre-shared keys (PSK)* or digital signatures (leveraging digi-

tal certificates). Authentication can also be done continuously through the use of an HMAC, which includes a secret known only to two ends of the VPN.

- **Antireplay support:** When VPNs are established, the peers can sequentially number the packets, and if a packet is attempted to be replayed again (perhaps by an attacker), the packet will not be accepted because the VPN device believes it has already processed that packet.

From our topology, we could decide that any traffic from the 10.0.0.0 network on the far left that needs to go to the 172.16.0.0 network on the far right should first be encrypted by R1, who would then send the protected packets over the Internet until they reach R2. The cloud between R1 and R2 represents an untrusted network, such as the Internet. R2 then decrypts each packet and sends the traffic on to its final destination, which may be PC or server on the 172.16.0.0 network. The protected packets could be encrypted, hashed, and kept track of to provide the four major benefits previously listed.

The Play by Play for IPsec

Let's start the play-by-play discussion assuming that both routers have been correctly configured to be VPN peers and that they have default routes pointing to the Internet and that they were both just powered up. With a site-to-site VPN, as shown in our topology, each of the peers could also be called a *VPN gateway*, which is serving the customers on the 10.0.0.0/24 and 172.16.0.0/24 networks. The two routers will become IPsec *peers* with each other to form the IPsec tunnel over the Internet.



The first thing the router on the left (R1) is going to do, if it is been told to encrypt and protect traffic that is sourced from the 10.0.0.0 network and destined for the 172.16.0.0 network, is wait for that traffic to show up. Let's say a user on the 10.0.0.0 network sends a packet to a server on the 172.16.0.0 network, and now R1 sees this packet and needs to encrypt it and protect it before sending it on its way. Unfortunately, the router has not yet established any VPN tunnels between itself and the router on the far right. So, if traffic did show up at R1 and it needed to be encrypted based on the policy, R1 would initiate negotiations with the router on the right. In this case, R1 would be the initiator of the VPN.

Step 1: Negotiate the IKE Phase 1 Tunnel

What these two routers first negotiate is something called an *Internet Key Exchange (IKE) Phase 1* tunnel. This can be done in one of two modes. Main mode or Aggressive mode. Main mode uses more packets for the process than Aggressive mode, but Main mode is considered more secure. Most-current VPN implementations default to using Main mode. This first tunnel (the IKE Phase 1 tunnel) is used between the two routers to speak directly to each other. This tunnel (once established) is not going to be used to forward user packets, but rather only to protect management traffic related to the VPN between the two routers. Packets such as a keepalive message to verify that the VPN tunnel is still working are an example of traffic that these two routers send across the IKE Phase 1 tunnel directly to each other.

Because the router on the left (R1) first received traffic that needed to be encrypted and there was no IKE Phase 1 tunnel in place, the router on the left becomes the initiator for the negotiations. The initiator sends over all of its configured/default parameters that it is willing to use for the IKE Phase 1 tunnel. Five basic items need to be agreed upon between the two VPN devices/gateways (in this case, the two routers) for the IKE Phase 1 tunnel to succeed, as follows:

- **Hash algorithm:** This could be *message digest 5 algorithm (MD5)* or *Secure Hash (SHA)* on most devices.
- **Encryption algorithm:** This could be *Digital Encryption Standard (DES)* (bad idea, too weak), *Triple DES (3DES)* (better) or *Advanced Encryption Standard (AES)* (best) with various key lengths. (Longer is better for keys.)
- **Diffie-Hellman (DH) group to use:** The DH “group” refers to the modulus size (length of the key) to use for the DH key exchange. Group 1 uses 768 bits, group 2 uses 1024, and group 5 uses 1536. The purpose of DH is to generate shared secret keying material (symmetric keys) that may be used by the two VPN peers for symmetrical algorithms, such as AES. It is important to note that the DH exchange itself is asymmetrical (and is CPU intensive), and the resulting keys that are generated are symmetrical.
- **Authentication method:** Used for verifying the identity of the VPN peer on the other side of the tunnel. Options include a *pre-shared key (PSK)* used only for the authentication or RSA signatures (which leverage the public keys contained in digital certificates).
- **Lifetime:** How long until this IKE Phase 1 tunnel should be torn down. (The default is 1 day, listed in seconds.) This is the only parameter that does not have to exactly match with the other peer to be accepted. If all other parameters match and the lifetime is different, they agree to use the smallest lifetime between the two peers. A shorter lifetime is considered more secure because it gives an attacker less time to calculate keys used for a current tunnel.



How to Remember the Five Items Negotiated in IKE Phase 1

As a handy way to recall the five pieces involved in the negotiation of the IKE Phase 1 tunnel, you might want to remember that the two devices HAGLE over IKE Phase 1:

H: Hash

A: Authentication method

G: DH group (a stretch, but it works)

L: Lifetime of the IKE Phase 1 tunnel

E: Encryption algorithm to use for the IKE Phase 1 tunnel

Who Begins the Negotiation?

The initiator sends over all of its IKE Phase 1 policies, and the other VPN peer looks at all of those policies to see whether any of its own policies match the ones it just received. If there is a matching policy, the recipient of the negotiations sends back information about which received policy matches, and they use that matching policy for the IKE Phase 1 tunnel.

Step 2: Run the DH Key Exchange

Now having agreed to the IKE Phase 1 policy of the peer, the two devices run the DH key exchange. They use the DH group (DH key size for the exchange) they agreed to during the negotiations, and at the end of this key exchange they both have symmetrical keying material (which is a fancy way of saying they both have the same secret keys that they can use with symmetrical algorithms).

DH, as you learned in a previous chapter, allows two devices that do not yet have a secure connection to establish shared secret keying material (keys that can be used with symmetrical algorithms, such as AES).

Step 3: Authenticate the Peer

The last piece of IKE phase 1 is to validate or authenticate the peer on the other side. For authentication, they use whatever they agreed to in the initial HAGLE, and if they successfully authenticate with each other, we now have an IKE Phase 1 tunnel in place between the two VPN gateways. This tunnel is bidirectional, meaning that either device can send or receive on that IKE Phase 1 tunnel. The authentication could be done either using a PSK or using RSA digital signatures (depending on what they agreed to use in Step 1).

What About the User's Original Packet?

Now here is the challenge: After all the work that went in to building the IKE Phase 1 tunnel, this tunnel is used only as a management tunnel so that the two routers can securely communicate with each other directly. This IKE Phase 1 tunnel is not used to encrypt or protect the end user's packets. To protect the end user's packets, (which is the entire goal for IPsec), the two VPN devices build a second tunnel for the sole purpose of encrypting the end-user packets. This second tunnel is called the IKE Phase 2 tunnel, it is also commonly referred to as (drum roll please) the IPsec tunnel. This IKE Phase 2 tunnel is the tunnel used to protect the end-user packets as those packets cross untrusted networks between the VPN peers.

Leveraging What They Have Already Built

The two routers, with a beautiful IKE Phase 1 tunnel in place, can use that IKE Phase 1 tunnel to securely negotiate and establish the IPsec or IKE Phase 2 tunnel. In my years of working with students, this is where the confusion sometimes creeps in, because during the configuration the students say to themselves, "Didn't I already specify the details

for encryption and hashing? Why is it asking for them again in the configuration?” The answer is that we have to set up specific commands to specify the IKE Phase 1 policies, and we set up a different set of similar commands for the IKE Phase 2 policy (including the component called a *transform set*).

Immediately after the IKE Phase 1 tunnel is established (the two different modes to set up the IKE Phase 1 tunnel are Main mode, which takes more packets, or Aggressive mode, which takes fewer packets and is considered less secure), the routers immediately begin to establish the IKE Phase 2 tunnel.

The IKE Phase 1 tunnel is their management tunnel. The entire conversation and negotiation of the IKE Phase 2 tunnel are completely done in private because of the IKE Phase 1 tunnel protection the negotiated traffic. The IKE Phase 2 tunnel includes the hashing and encryption algorithms. The name of the mode for building the IKE Phase 2 tunnel is called Quick mode.

Now IPsec Can Protect the User’s Packets



Once the IKE Phase 2 tunnel is built, the routers can then begin to encrypt the user’s traffic and send those encrypted packets directly to the peer on the far side. From the Internet’s perspective, it looks like packets sourced from the IP address of R1 are being sent to the IP address of R2. The encrypted payload of these packets contains the original IP addresses and contents of the user who is forwarding a packet to a server or vice versa. If these packets are eavesdropped upon, the eavesdropper sees the IP addresses involved between the two routers; the payload (the original packets) have been encrypted and encapsulated inside, and are cipher text and unreadable to the individual who does not have the symmetric keys to decrypt the contents.

Traffic Before IPsec

If the packet is captured before IPsec is used, as it crosses the cloud (untrusted network), the eavesdropper sees the packet as it appears in Figure 19-2.

```

Internet Protocol, Src: 10.0.0.25 (10.0.0.25), Dst: 172.16.0.4 (172.16.0.4)
  Version: 4
  Header length: 20 bytes
  Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
  Total Length: 58
  Identification: 0x3aaed (15085)
  Flags: 0x02 (Don't Fragment)
  Fragment offset: 0
  Time to live: 127
  Protocol: TCP (0x06)
  Header checksum: 0x0aa4 [correct]
  Source: 10.0.0.25 (10.0.0.25)
  Destination: 172.16.0.4 (172.16.0.4)
Transmission Control Protocol, Src Port: ssslog_mgr (1204), Dst Port: telnet (23), Seq: 4, Ack: 18, Len: 18
  Source port: ssslog_mgr (1204)
  Destination port: telnet (23)
  [Stream index: 0]
  Sequence number: 4 (relative sequence number)
  [Next sequence number: 22 (relative sequence number)]
  Acknowledgement number: 18 (relative ack number)
  Header length: 20 bytes
  Flags: 0x18 (PSH, ACK)
  Window size: 64223
  Checksum: 0x0247 [validation disabled]
  [SEQ/ACK analysis]
Telnet
  Command: Do Suppress Go Ahead
  Command: Will Terminal Type
  Command: Will Negotiate About Window Size
  Suboption Begin: Negotiate About Window Size
  Command: Suboption End

```

Before IPsec, all the original IP, TCP, and Application Header information and payload are in plain text.

An eavesdropper would be able to see an entire telnet conversation between client and server.

Figure 19-2 Plain Text, Before IPsec

In this figure, any eavesdropper can see and determine the entire conversation between the client and the server. Because Telnet offers no encryption capabilities on its own, the attacker could learn the username and password used for initiation of this Telnet session and each command the Telnet user issued and their results.

Traffic After IPsec

After you configure R1 and R2 to become VPN peers/gateways, and telling them that all packets between the two networks of 10.0.0.0/24 and 172.16.0.0/24 should be protected by IPsec, R1 and R2 negotiate and build their VPN tunnels (IKE phase 1 and IKE Phase 2), and then any traffic from either network and destined for the other is protected. Let's consider the packet shown in the earlier figure. When R1 sees this same packet heading out to 172.16.0.4, and because its source IP address is on the 10.0.0.0/24 network, R1 uses the IKE Phase 2 tunnel and encrypts the packet and encapsulates the encrypted packet with a new IP header that shows the source IP address as R1 and the destination address as R2. The Layer 4 protocol would show as being *Encapsulating Security Payload (ESP)*, which is reflected in the IP header as protocol #50, which is in plain text, but the content after that is the encrypted original packet. When R2 receives this, R2 de-encapsulate the packet, sees that it is ESP, and then proceeds to decrypt the original packet. Once decrypted, R2 forwards the plaintext packet to the server at 172.16.0.4. The encrypted packet as it crossed over the untrusted network between R1 and R2 appears, as shown Figure 19-3.



```

Ethernet II, Src: ca:00:60:e0:00:1c (ca:00:60:e0:00:1c), Dst: ca:02:60:e0:00:1c (ca:02:60:e0:00:1c)
Internet Protocol, Src: 23.0.0.1 (23.0.0.1), Dst: 43.0.0.2 (43.0.0.2)
Encapsulating Security Payload

```

Only the new Layer 3 IP header and Layer 4 ESP header can be seen in plain text. The content of the original packet from the PC to the server is now protected as cipher text as the payload of this IPsec packet.

```

30 ca 02 68 e8 00 1c ca 00 68 e8 00 1c 08 00 45 00  .h....h....E.
10 00 60 05 f8 40 00 ff 32 33 71 17 00 00 01 2b 00  .2...t...t...
20 00 02 80 8a 39 d0 00 00 00 21 43 7a 88 53 02 02  ....t...t...
30 97 09 99 8d 39 ab a9 39 d1 74 1a 9e 29 a3 ba f9  .g.e.t...t...
40 83 d6 17 84 e0 17 13 de 53 b5 3c 8b b9 36 a7  ....s...t...
50 20 84 19 3a 18 06 c8 35 09 ab 8c a8 b9 a9 04  ....s...t...
50 13 75 e1 0e ea 40 7d b2 8c 59 ab 56 e8 c3  .(.....).....

```

Most of this represents the (now encrypted) original packet from the PC to the server.

Figure 19-3 Encrypted Packet Crossing the Internet

Summary of the IPsec Story



In summary, the VPN peers/gateways negotiate the IKE phase 1 tunnel using Aggressive or Main mode, and then use Quick mode to establish the IKE Phase 2 tunnel. They use the IKE Phase 2 tunnel to encrypt and decrypt user packets. Behind the scenes, the IKE Phase 2 tunnel really creates two one-way tunnels: one from R1 to R2, and one from R2 to R1. The end user does not see the process in any detail, and end users do not know the encryption is even being applied to their packets. So, we could say we have one IKE Phase 1 bidirectional tunnel used for management between the two VPN peers and two IKE Phase 2 unidirectional tunnels used for encrypting and decrypting end-user packets. These tunnels are often referred to as the security agreements between the two VPN peers. Many times, these agreements are called *security associations (SA)*. Each SA is assigned a unique number for tracking.

Configuring and Verifying IPsec

Now that we have taken a look at the building blocks for IPsec, let's apply what you have learned to the topology introduced at the beginning of the chapter in Figure 19-1.

Tools to Configure the Tunnels

In the CCNA Security courseware, *Cisco Configuration Professional (CCP)* is used to configure the VPN tunnels, including both IKE Phase 1 and IKE Phase 2. We use CCP here, but you also learn the *command-line interface (CLI)* equivalent for each of the commands, which are annotated to let you know what each command does.

Start with a Plan

The first thing to plan is what protocols to use for IKE Phase 1 and IKE Phase 2 and to identify which traffic should be encrypted.

From the earlier topology, let's agree to encrypt any traffic from the 10.0.0.0/24 network behind R1 if those packets are going to 172.16.0.0/24 behind R2 and packets in the other direction from 172.16.0.0/24 to 10.0.0.0/24.

For IKE Phase 1, let's use the following:

H: For hashing, we can use MD5 (128 bits) or SHA-1 (160 bits). Let's go for MD5 for IKE Phase 1.

A: Authentication. We can use PSKs or digital certificates. Let's start off with PSKs (a password really) for authentication.

G: For DH group, we can use 1, 2, or 5 on most routers. Let's use group 2.

L: Lifetime is default to 1 day. Let's set the lifetime for the IKE Phase 1 to 600 seconds.

E: Encryption of the IKE Phase 1 can be DES, 3DES, or some flavor of AES. Let's use 128-bit AES.

Now for Phase 2, we also need to decide on hashing and encryption at a minimum. We can use the defaults for lifetime. For hashing, let's use SHA (just to see the difference between the hashing here and the hashing protocol in IKE Phase 1). Let's also use AES-256 in IKE Phase 2. The policies used for IKE Phase 2 are called transform sets.

Applying the Configuration

With all of that in mind, let's start the configuration on R1. Using CCP, select R1 from the drop-down menu and navigate to **Configure > Security > VPN > Site-to-Site VPN**. From there, you verify that the **Create a Site-to-Site VPN** option is selected, and then click the **Launch the Selected Task** button, as shown in Figure 19-4.



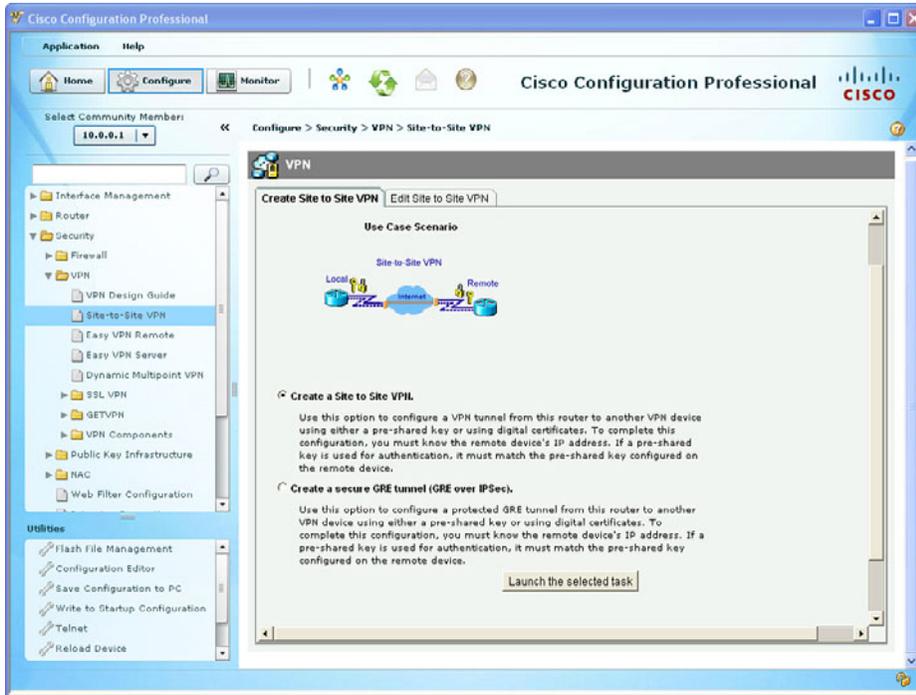


Figure 19-4 Using CCP to Configure IPsec VPN Site-To-Site Tunnels

Next, you are prompted to either use Quick Setup or the Step by Step Wizard. Quick Setup uses the defaults for the IKE Phase 1 and IKE Phase 2 that are built in to CCP. If you want to customize the policies, choose the **Step by Step Wizard**, as shown in Figure 19-5, and then click **Next**.



Figure 19-5 Selecting the Step by Step Wizard

From the interface drop-down list, select the interface on R1 that will be facing the Internet (this is also the interface facing toward its peer, R2), and configure the IP address of the peer (the reachable address over the Internet). In this case, R2's outside address is 43.0.0.2. Select the option for authenticating using a PSK and configure the key. (This needs to be the same key on both sides. For this example, we use the PSK of cisco123 for the IKE Phase 1 authentication.) After entering the data, review it to make sure it is accurate, as shown in Figure 19-6, and then click **Next** to continue.

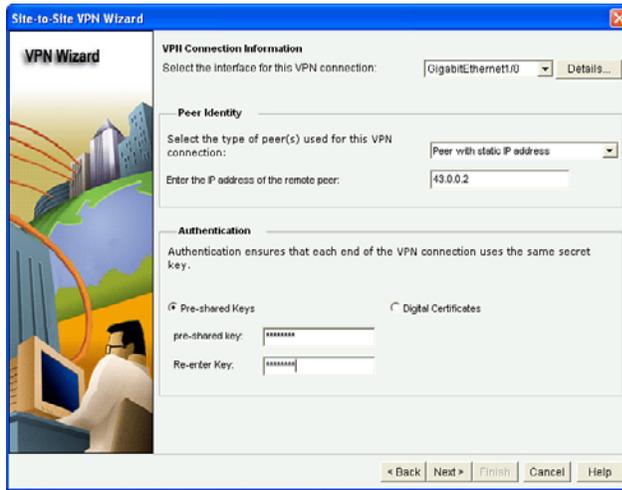


Figure 19-6 *Entering the Local Ethernet and Remote Peer Information, Including the PSK*

You are next asked for the IKE Phase 1 proposals you want to use. If you want to use the default, that is fine as long as you use it on both sides (both routers use the same policy) and it matches what you want to use for the IKE Phase 1 policy. We decided (earlier, you and I) that we would use MD5 for hashing, PSK for authentication, DH group 2, a 600-second lifetime, and AES 128-bit key for encryption. After looking at the defaults, shown in Figure 19-7, you click **Add** to create a new IKE Phase 1 policy.

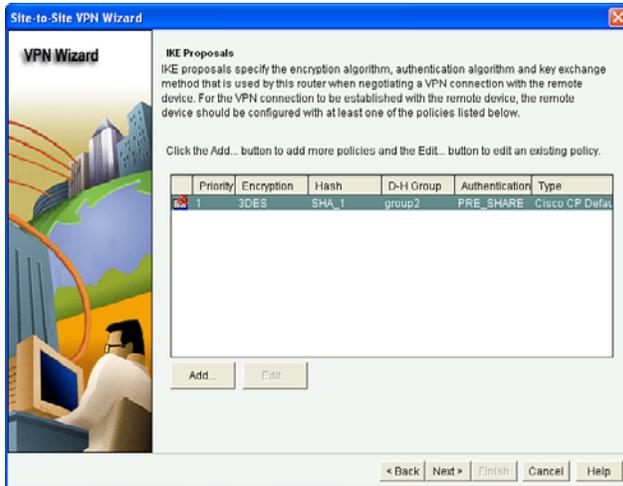


Figure 19-7 Default IKE Phase 1 Policy Within CCP

After clicking the **Add** button, you put in your desired IKE Phase 1 policies, as shown in Figure 19-8, and click **OK**.

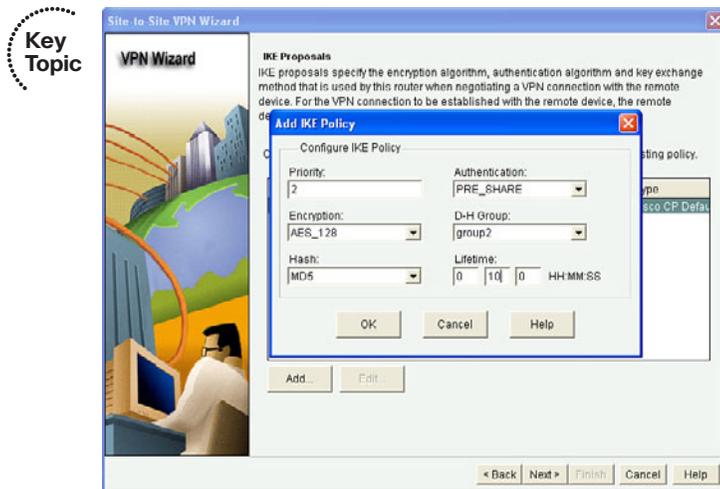


Figure 19-8 Entering Custom IKE Phase 1 Policies

After creating your new IKE Phase 1 policy, you still need to select it (highlight it) before clicking **Next**. The CCP creates its default policy by default, along with your new policy. After highlighting the new policy (priority 2 in the example, as shown in Figure 19-9), click **Next** to continue.



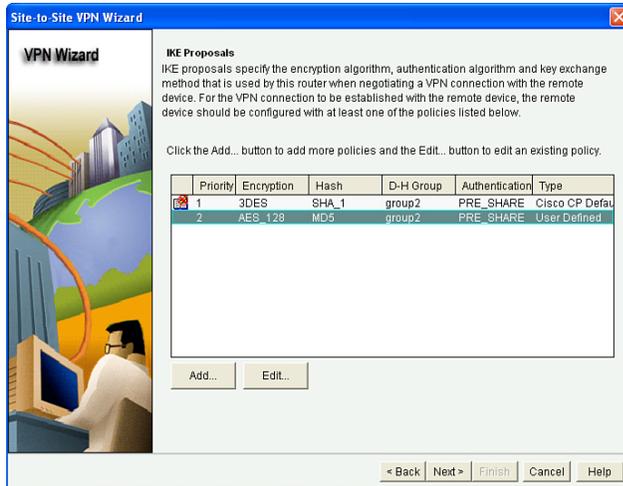


Figure 19-9 Highlight the New Policy You Created before Clicking Next

The next screen that appears looks similar to the first, but this box has the title Transform Set near the top. A transform set refers to the methods of encryption and hashing that you want to use for the IKE Phase 2 tunnels. We do not want to use the defaults, but rather we want to follow our plan of using AES-256 and SHA for the IKE Phase 2 tunnels. (We could have used the same exact protocols, but I wanted you to see the distinction between the options we have for either tunnel independently.) Figure 19-10 shows the default transform set.

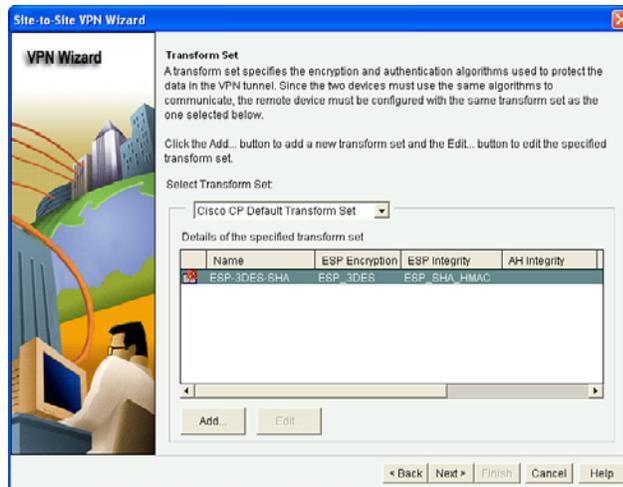


Figure 19-10 Default Transform Set Used by CCP

By clicking Add, you can specify the IKE Phase 2 policies of your choice. Remember that whatever you choose here, you also need to configure on the other router, as well. Figure 19-11 shows an example of creating a new transform set.

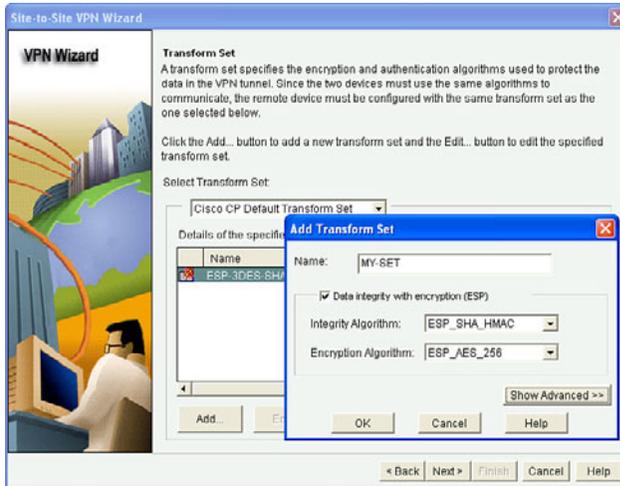


Figure 19-11 *Creating a New Transform Set (IKE Phase 2 Policy)*

After entering the new information for your transform set, click **OK**, and then verify your new transform set is selected before clicking **Next** to continue, as shown in Figure 19-12.

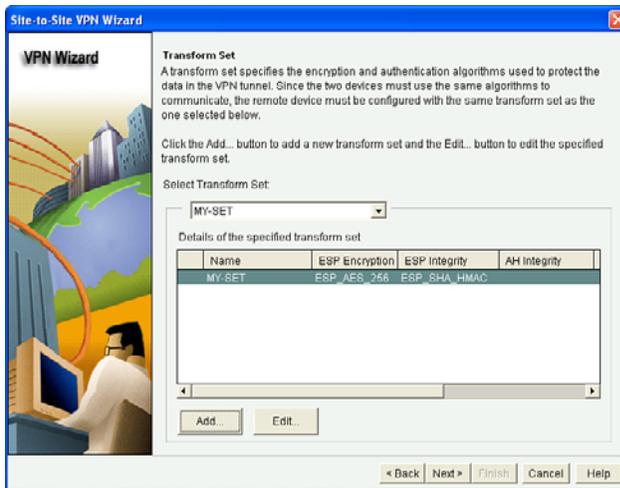


Figure 19-12 *Selecting the New Transform Set*

The wizard then asks what traffic should be encrypted. Because we are on R1, we should focus only on outbound traffic that should be encrypted. (It is R2's responsibility to make sure that the correct inbound traffic to R1 from R2 is encrypted). To do this, we use an access list as the classifier or identifier of what traffic should be encrypted. R1's and R2's classifying *access control lists (ACL)* should be symmetrical, in that if R1 says to encrypt all packets that are from 10.0.0.0/24 and going to 172.16.0.0/24, R2 should say that it will encrypt all packets from 172.16.0.0/24 that are destined for 10.0.0.0/24.

That is what is meant by having *symmetrical* access lists on the VPN peers in a site-to-site VPN. (It is just a bad coincidence that we also use the word *symmetrical* to describe algorithms like AES that these peers will be using.) An ACL that has been created to identify which traffic should be encrypted is called a *crypto ACL*. Note that a crypto ACL is not applied directly to any interface, but instead it is referenced by a policy called a *crypto map* (discussed soon). The crypto map is directly applied to an interface.

From R1's perspective, we should “protect,” which means use IPsec on packets with a source address from the 10.0.0.0/24 network and that also have a destination address in the 172.16.0.0/24 network. So, we fill in the wizard as shown in Figure 19-13 and click **Next** to continue.

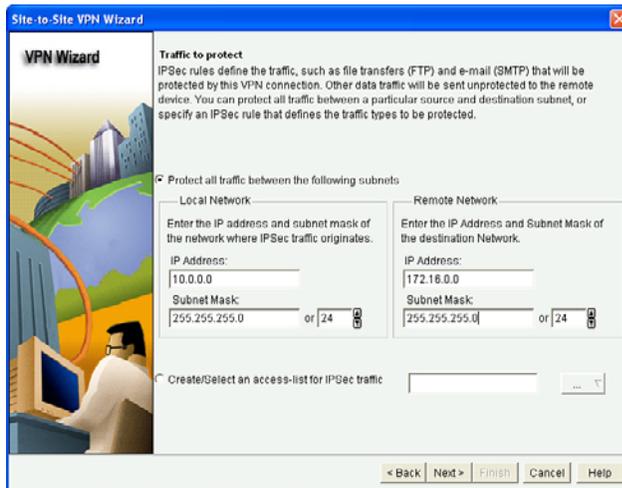


Figure 19-13 Configuring the ACL Used to Classify Whether Traffic Should Be Protected by IPsec

Packets that are not matched for IPsec protection will be forwarded as normal packets, without any IPsec encapsulation or encryption applied. When you click the **Next** button, a summary displays of the IKE policies (IKE Phase 1) and transform sets (IKE Phase 2) that it will implement on the router. Note that CCP likes to implement the default IKE Phase 1, along with the custom IKE Phase 1 policy, so both will end up in the configuration. The policy also specifies the authentication method we selected earlier in the wizard (PSK), and which network's traffic should be protected. (The traffic to protect is from the outbound perspective. In this case, R1's outbound traffic is from the 10.0.0.0/24 to the 172.16.0.0/24 network.) Figure 19-14 shows this summary table. If everything is correct, click **Finish** to deliver the configuration to the router.

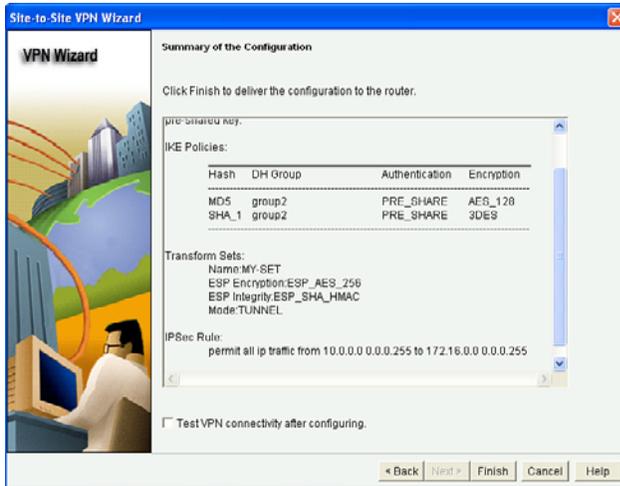


Figure 19-14 Summary of What the VPN Wizard Is About to Implement

Based on your preference settings, CCP may show you the CLI equivalent of the configuration it is about to deploy or may just deploy it when you click **Finish**. You can control these settings in the preference settings for CCP.

Viewing the CLI Equivalent at the Router

Example 19-1 shows the CLI equivalent that is implemented on R1 from the configuration we did in CCP on R1.



Example 19-1 The CLI Equivalent Commands to Implement IPsec VPNs

```
! This implements our IKE phase 1 policy.      The default policy that CPP
! implements is its policy #1, (which has higher priority than a higher
! numbered policy, including our policy #2.)
R1(config-isakmp)# crypto isakmp policy 2
R1(config-isakmp)# authentication pre-share
R1(config-isakmp)# encr aes 128
R1(config-isakmp)# hash md5
R1(config-isakmp)# group 2
R1(config-isakmp)# lifetime 600
R1(config-isakmp)# exit
! Note: I like to remove the default policy from CCP for IKE phase 1, and
! for that reason, I have not replicated it here.

! This specifies that the PSK of cisco123 should be used as a key for the
! authentication of IKE phase 1 with peer 43.0.0.2.
R1(config)# crypto isakmp key cisco123 address 43.0.0.2
```

```

! Access list that identifies any traffic from the 10.0.0.0/24 network
! and destined for the 172.16.0.0/24 network.  An ACL used for cryptography
! is often referred to as a "crypto ACL".  This ACL will not be directly
! applied to an interface, but rather it will be called on or "referenced"
! within the crypto map, later in this configuration.
R1(config)# access-list 100 permit ip 10.0.0.0 0.0.0.255 172.16.0.0
0.0.0.255

! The IKE Phase 2 transform set that says SHA and AES 256 should be used.
R1(config)# crypto ipsec transform-set MY-SET esp-sha-hmac esp-aes 256

! Tunnel mode is the default, and means that R1 will take any outbound
! packets matching the access list, encrypt them and then re-encapsulate
! them inside of an IPsec packet, which is then forwarded to the peer (R2)
! on the other side of the VPN tunnel.  Whenever customer traffic is going
! through a VPN router, it will need to be in tunnel mode to work.
! Transport mode is the other option, and it is used only when the transit
! traffic is directly from and to the endpoints of the VPN tunnel (such as
! R1 and R2 talking amongst themselves).  Because we are encrypting traffic
! for the end users, tunnel mode (the default) will be used.
R1(cfg-crypto-trans)# mode tunnel

R1(cfg-crypto-trans)# exit

! The crypto map is a big "if-then" statement.  It is applied to the outside
! (Internet facing) interface, and then it watches for traffic.
! If outbound traffic matches the ACL, then the router knows the packet
! should be encrypted, encapsulated into an IPsec header (usually protocol
! 50, which is ESP and stands for Encapsulating Security Payload), and then
! sent to the IP address of the peer on the other side (R2) who would
! decrypt and forward the plain text packet to the device on network
! 172.16.0.0/24  "ipsec-isakmp" means that we want the router to automatically
! negotiate the IKE Phase 2 tunnel, using isakmp, which stands for Internet
! Security Association Key Management Protocol.  In short, it means automate
! the process, so the administrator doesn't manually have to configure all
! keys for encryption.  The "1" represents sequence number 1.  If we had
! 5 different IPsec peers, we could use 5 different sequence numbers in the
! same crypto map to organize our policies based on the sequence number and
! corresponding peer we would be using IPsec with.

```

```
R1(config)# crypto map SDM_CMAP_1 1 ipsec-isakmp

! This tells the crypto map to pay attention to ACL 100 to see if traffic
! should be encrypted or not
R1(config-crypto-map)# match address 100
! If the traffic matches the ACL, then R1 should use the transform-set
! named MY-SET to negotiate the IKE Phase 2 tunnel, with the peer at
! 43.0.0.2
! If the IKE Phase 1 tunnel isn't present, it will trigger the negotiation
! of that first. If the IKE Phase 2 is already in place, the router will
! use the existing tunnel for the encryption and transmission of the
! customer's packet
R1(config-crypto-map)# set transform-set MY-SET
R1(config-crypto-map)# set peer 43.0.0.2
R1(config-crypto-map)# exit

! Applying the crypto map to the interface, is what activates our policy,
! and tells the router to start paying attention in looking for interesting
! traffic (which is the traffic that matches the ACLs referenced in the
! crypto map).
R1(config)# interface GigabitEthernet1/0
R1(config-if)# crypto map SDM_CMAP_1
R1(config-if)# exit
%CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
```

Completing and Verifying IPsec

When you click **Finish**, the CCP shows you the status of the VPN tunnel, as shown in Figure 19-15. The reason the tunnel is down is because the other side is not yet configured.

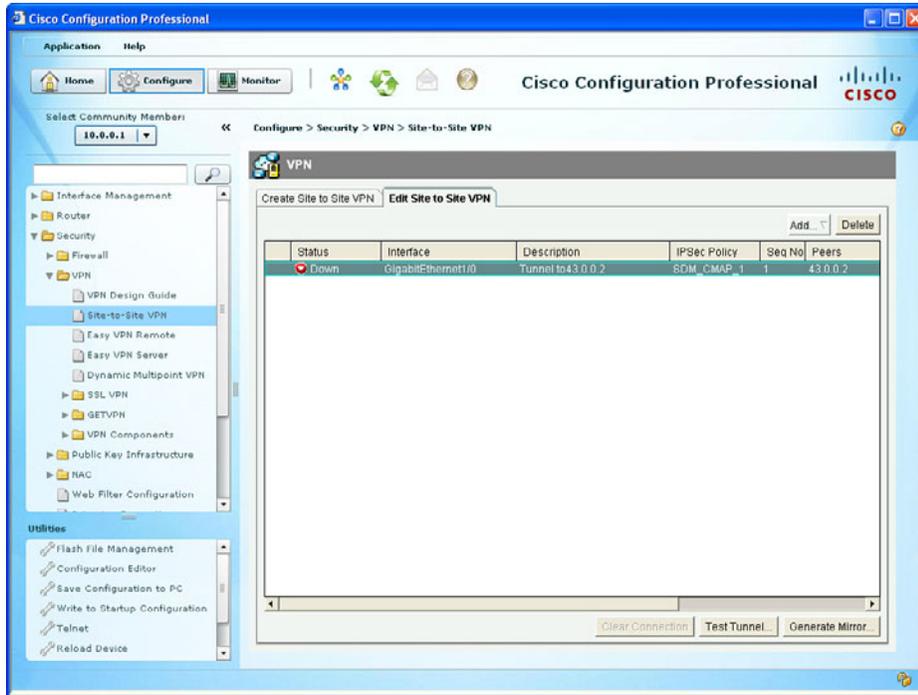


Figure 19-15 Results of Finishing the VPN Wizard

To configure R2, we could select R2 from within CCP and follow the same process. A shortcut that CCP has provided is the ability to use the Generate Mirror button from R1's CCP, and then modify and apply that mirror image of the VPN-related configuration to R2. Figure 19-16 shows the result of clicking the **Generate Mirror** button.

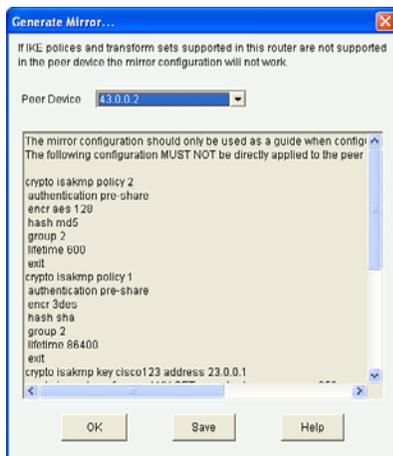


Figure 19-16 Generating a Mirror of the VPN as a Guideline for the Remote Peer

We could then take this file, edit it, and apply it to R2. Example 19-2 shows an edited file that is appropriate for R2.



Example 19-2 *Edited Mirrored VPN Configuration Appropriate for R2*

```
crypto isakmp policy 2
 authentication pre-share
 encr aes 128
 hash md5
 group 2
 lifetime 600
 exit

crypto isakmp key cisco123 address 23.0.0.1
crypto ipsec transform-set MY-SET esp-sha-hmac esp-aes 256
 mode tunnel
 exit

ip access-list extended SDM_1
 permit ip 172.16.0.0 0.0.0.255 10.0.0.0 0.0.0.255
 exit

crypto map SDM_CMAP_1 1 ipsec-isakmp
 match address SDM_1
 set transform-set MY-SET
 set peer 23.0.0.1
 exit

interface g1/0
 crypto map SDM_CMAP_1
 end
```

If you are using CCP and apply this directly to the CLI of R2, be sure to refresh or rediscover R2 via CCP to reflect the changes. Saving the changes to NVRAM is also recommended after a working solution has been implemented. After generating some traffic from a device on the 10.0.0.0/24 network that is destined for 172.16.0.0/24, the outbound traffic on R1 should trigger encryption, which triggers IKE Phase 1 and Phase 2 to build their tunnels and then begin forwarding the traffic. The successful ping of a device in the 10.0.0.0/24 network to a device in the 172.16.0.0 network, along with the status being shown from CCP, confirms that the VPN tunnel is working, as shown in Figure 19-17.

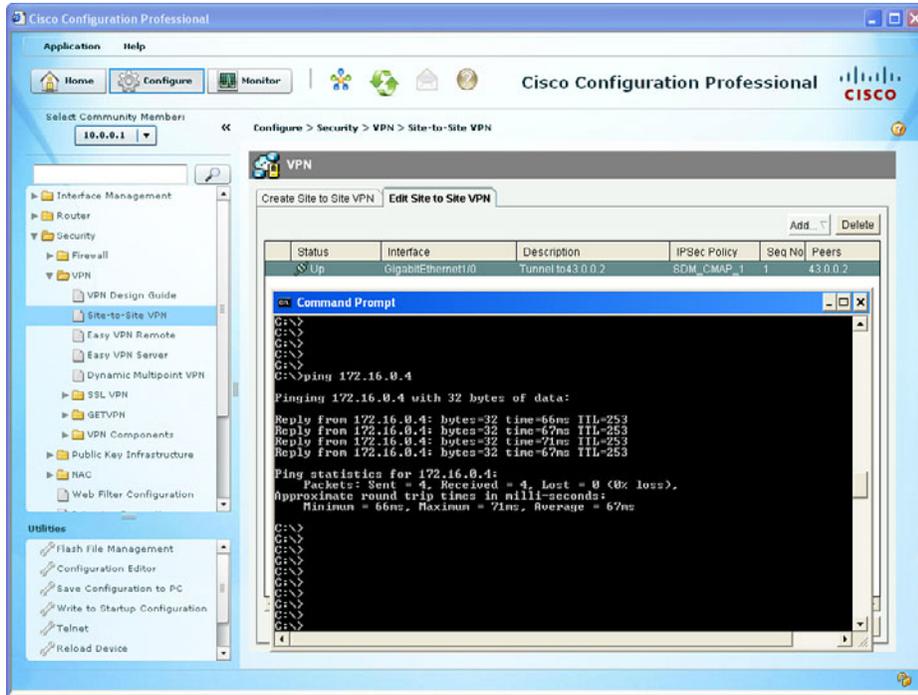


Figure 19-17 Verifying the Tunnel Is Working

Subsequent user packets use the newly formed IKE Phase 2 (IPsec) tunnel for the lifetime of that tunnel. From the command line, you could use the following to verify the IPsec, as well, as shown in Example 19-3.

Example 19-3 Verifying the IPsec VPN from the CLI

```

! Verify the IKE Phase 1 policies in place on the router
R1# show crypto isakmp policy

Global IKE policy
Protection suite of priority 2
  encryption algorithm: AES - Advanced Encryption Standard (128 bit
    keys).
  hash algorithm: Message Digest 5
  authentication method: Pre-Shared Key
  Diffie-Hellman group: #2 (1024 bit)
  lifetime: 600 seconds, no volume limit

! Show the details of the crypto map, and where it is applied, showing
! the contents of the IKE Phase 2 transform sets, learning the ACLs
! involved for the VPN, who the current peer is, and more.
R1# show crypto map

```



```

Crypto Map "SDM_CMAP_1" 1 ipsec-isakmp
  Description: Tunnel to43.0.0.2
  Peer = 43.0.0.2
  Extended IP access list 100
    access-list 100 permit ip 10.0.0.0 0.0.0.255 172.16.0.0
    0.0.0.255
  Current peer: 43.0.0.2
  Security association lifetime: 4608000 kilobytes/3600 seconds
  Responder-Only (Y/N): N
  PFS (Y/N): N
  Transform sets={
    MY-SET: { esp-256-aes esp-sha-hmac } ,
  }
  Interfaces using crypto map SDM_CMAP_1:
    GigabitEthernet1/0

! See the details for the IKE Phase 1 tunnel that is in place

R1# show crypto isakmp sa detail
Codes: C - IKE configuration mode, D - Dead Peer Detection
       K - Keepalives, N - NAT-traversal
       T - cTCP encapsulation, X - IKE Extended Authentication
       psk - Preshared key, rsig - RSA signature
       renc - RSA encryption
IPv4 Crypto ISAKMP SA

C-id  Local      Remote      I-VRF      Status Encr Hash Auth DH Lifetime Cap.
1001  23.0.0.1    43.0.0.2
      Engine-id:Conn-id = SW:1

! See the details for the IKE Phase 2 tunnels that are in place. There is
! one inbound Security Association (SA) and one outbound. They both have
! different SA numbers used for tracking these sessions.
! ESP is used, and it provides all the services desirable from IPsec.
! The other option is Authentication Header (AH) which isn't used because
! it doesn't support any encryption algorithms.

R1# show crypto ipsec sa
<Note: less relevant content removed>
interface: GigabitEthernet1/0
  Crypto map tag: SDM_CMAP_1, local addr 23.0.0.1
! Shows what traffic is being encrypted. All IP traffic between
! 10.0.0.0/24 and 172.16.0.0/24

```

```

local ident (addr/mask/prot/port): (10.0.0.0/255.255.255.0/0/0)
remote ident (addr/mask/prot/port): (172.16.0.0/255.255.255.0/0/0)

! IKE phase 1 uses UDP port 500 to negotiate and set up the IKE phase 1
! tunnel
  current_peer 43.0.0.2 port 500
    #pkts encaps: 29, #pkts encrypt: 29, #pkts digest: 29
    #pkts decaps: 29, #pkts decrypt: 29, #pkts verify: 29

! From R1's perspective, the local side is its G1/0, and R2 is at 43.0.0.2
  local crypto endpt.: 23.0.0.1, remote crypto endpt.: 43.0.0.2
  path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet1/0

! An SPI is a Security Parameter Index. It is a fancy way of tracking
! a specific Security Association (SA) between itself and a peer.
! Think of it as a serial number (unique) for each SA.
  current outbound spi: 0x48A3CF57(1218694999)

! PFS stands for Perfect Forward Secrecy, and it is the ability for IKE
! Phase 2 to run the DH algorithm again, instead of using the keys
! generated during the DH from IKE phase 1. This feature is off by
! default for most platforms.
  PFS (Y/N): N, DH group: none

! The IPsec or IKE Phase 2 is really two tunnels. There is one for
! traffic from R1 to R2. There is another from R2 to R1. They have
! different SPIs, but together, these two unidirectional tunnels make up
! the "IPsec" tunnel.
! Encapsulating Security Payload (ESP) is the primary method used by IPsec.
! The other option is to use Authentication Header (AH), but it doesn't
! have the ability to encrypt, and isn't often used for that reason. AH
! also breaks when going through Network Address Translation (NAT).
! Here is the inbound SA used by R1 to receive encrypted user packets from
! R2.
  inbound esp sas:
    spi: 0xE732E3A0(3878871968)
      transform: esp-256-aes esp-sha-hmac ,
      in use settings ={Tunnel, }
      conn id: 1, flow_id: SW:1, sibling_flags 80000046, crypto map:
        SDM_CMAP_1
      sa timing: remaining key lifetime (k/sec): (4388080/3230)
      IV size: 16 bytes

! Here is the built in anti-replay support
  replay detection support: Y

```

```

        Status: ACTIVE
! We aren't using AH, so there are no Security Associations (SAs) for AH.
  inbound ah sas:

! Here is the Outbound SA used by R1 to send encrypted user packets to R2.
  outbound esp sas:
    spi: 0x48A3CF57(1218694999)
    transform: esp-256-aes esp-sha-hmac ,
    in use settings ={Tunnel, }
    conn id: 2, flow_id: SW:2, sibling_flags 80000046, crypto map: SDM_
      CMAP_1
    sa timing: remaining key lifetime (k/sec): (4388079/3230)
    IV size: 16 bytes
    replay detection support: Y
    Status: ACTIVE

  outbound ah sas:

! Another way of seeing that the encryption and decryption is working.
R1# show crypto engine connections active
Crypto Engine Connections

   ID  Type      Algorithm          Encrypt  Decrypt IP-Address
   --  ---      -
    1  IPsec     AES256+SHA        0        29 23.0.0.1
    2  IPsec     AES256+SHA        29        0 23.0.0.1
  1001 IKE       MD5+AES           0         0 23.0.0.1

```

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 19-3 lists these key topics.

Table 19-3 *Key Topics*

Key Topic Element	Description	Page Number
Text	IPsec goals	468
Text	The play by play for IPsec	469
Text	How to remember the five items negotiated in IKE Phase 1	470
Text	Now IPsec can protect user packets	472
Text	A look at the traffic after IPsec	473
Text	The IPsec story in a nutshell	474
Text	Start with a plan	475
Figure 19-8	Entering custom IKE Phase 1 policies	478
Figure 19-11	Creating a new transform set (IKE Phase 2 policy)	480
Figure 19-13	Configuring the access list used to classify traffic should be protected by IPsec	481
Example 19-1	The CLI equivalent commands to implement IPsec VPNs	482
Example 19-2	Edited mirrored VPN configuration appropriate for R2	486
Example 19-3	Verifying the IPsec VPN from the CLI	487



Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

IKE Phase 1, IKE Phase 2, transform set, DH group, lifetime, authentication, encryption, hashing, DH key exchange

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 19-4 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 19-4 *Command Reference*

Command	Description
<code>crypto map mymap 1 ipsec-isakmp</code>	Generate or edit a crypto map named MYMAP, sequence number 1, and request the services of ISAKMP.
<code>crypto isakmp policy 3</code>	Enter IKE Phase 1 configuration mode for policy number 3.
<code>show crypto map</code>	Verify which components are included in the crypto map, including the ACL, the peer address, the transform set, and where the crypto map is applied.
<code>crypto ipsec transform set myset</code>	This is the beginning sequence to creating an IKE Phase 2 transform set named MYSET. This is followed by the HMAC (hashing with authentication) and encryption method (3DES, or AES preferably) that you want to use.

This page intentionally left blank



This chapter covers the following subjects:

- Planning and preparing an IPsec site-to-site VPN
- Implementing and verifying an IPsec site-to-site VPN

Implementing IPsec Site-to-Site VPNs

In the previous chapters, you learned about the benefits of *virtual private networks* (VPN) and the protocols and methods used to implement those benefits, such as encryption for confidentiality, hashing for data integrity, and authentication for peer verification. You have also seen examples of these protocols, such as *Triple Digital Encryption Standard (3DES)* and *Advanced Encryption Standard (AES)* for encryption, *message digest 5 algorithm (MD5)* and *Secure Hash (SHA)* for data integrity, and *pre-shared keys (PSK)* or RSA signatures (also known as digital signatures) used for authentication.

In this chapter, we look at a case study and implement a VPN site-to-site tunnel using IOS routers as the VPN peers to provide the security the customer is looking for.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter’s topics before you begin. Table 20-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 20-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Planning and Preparing an IPsec Site-to-Site VPN	1–6
Implementing and Verifying an IPsec Site-to-Site VPN	7–10

1. Which of the following could be part of both an IKE Phase 1 and IKE Phase 2 policy? (Choose all that apply.)
 - a. MD5
 - b. AES
 - c. RSA
 - d. DH

2. How is it possible that a packet with a private Layer 3 destination address is forwarded over the Internet?
 - a. It is encapsulated into another packet, and the Internet only sees the outside valid IP destination address.
 - b. It cannot be sent. It will always be dropped.
 - c. The Internet does not filter private addresses, only some public addresses, based on policy.
 - d. NAT is used to change the destination IP address before the packet is sent.
3. What is the method for specifying the IKE Phase 2 encryption method?
 - a. Crypto ACLs
 - b. `crypto isakmp policy`
 - c. `crypto ipsec transform-set`
 - d. RSA signatures
4. Which of the following potentially could be negotiated during IKE Phase 2? (Choose all that apply.)
 - a. Hashing
 - b. DH group
 - c. Encryption
 - d. Authentication method
5. Which of the DH groups is the most prudent to use when security is of the utmost importance?
 - a. 1
 - b. 2
 - c. 5
 - d. 6
6. Which of the following is never part of an IKE Phase 2 process?
 - a. Main mode
 - b. Specifying a hash (HMAC)
 - c. Running DH (PFS)
 - d. Negotiating the transform set to use

7. Which encryption method will be used to protect the negotiation of the IPsec (IKE Phase 2) tunnel?
 - a. The one negotiated in the transform set.
 - b. The one negotiated for the IKE phase 2 tunnel.
 - c. The one negotiated in the ISAKMP policy.
 - d. There is not encryption during this time; that is why DH is used.
8. Which is the most secure method for authentication of IKE Phase 1?
 - a. RSA signatures, using digital certificates to exchange public keys
 - b. PSK
 - c. DH group 5
 - d. Symmetrical AES-256
9. Which component is not placed directly in a crypto map?
 - a. Authentication policy
 - b. ACL
 - c. Transform set
 - d. PFS
10. Which of the following would cause a VPN tunnel using IPsec to never initialize or work correctly? (Choose all that apply.)
 - a. Incompatible IKE Phase 2 transform sets
 - b. Incorrect pre-shared keys or missing digital certificates
 - c. Lack of interesting traffic
 - d. Incorrect routing

Foundation Topics

Planning and Preparing an IPsec Site-to-Site VPN

In this section, we use a case study to identify a customer's needs for VPN services and plan out the details to implement the VPN. This section builds on the information learned in the previous chapters about VPNs.

Customer Needs

For this scenario, let's say you and I have a customer with a local area network and a single router, R1, that connects to the Internet. We refer to this as site number 1. The customer recently opened up a new branch office in another state, and is calling that location site number 2, and is using router R2 at that location to provide Internet access for site 2. Figure 20-1 is a topology diagram of this network.

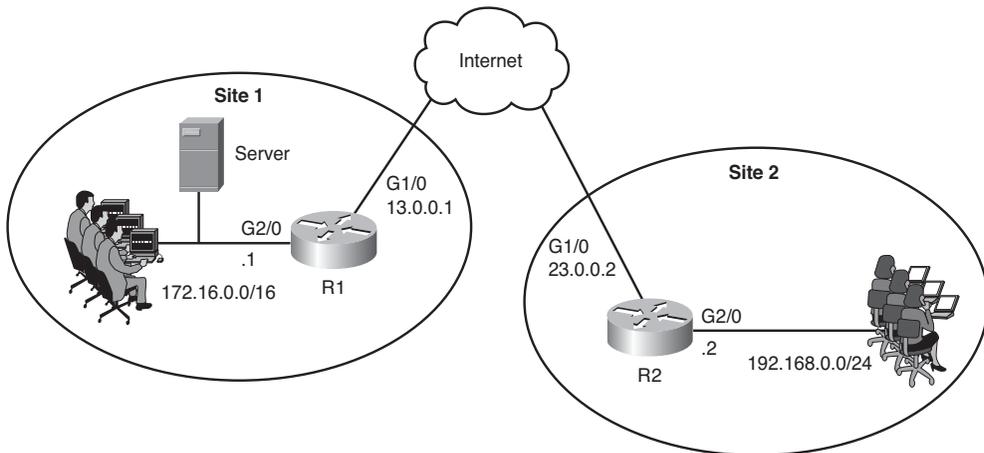


Figure 20-1 *Company Network Topology with Two Sites*

Site 1 has file servers that contain sensitive customer data, and the users at site 2 will need access to that data. In addition, users in site 1 need the ability to securely access some of the computers in site 2 that have file sharing services enabled. Both sites are using private IP addresses for the LAN that cannot be forwarded directly over the Internet.

The customer has asked us for a recommendation to allow file services between the two offices that can be done securely. The customer also wants to ensure that the data as it is being sent over the networks does not become altered or corrupted in transit. The customer is also concerned about a possible attacker, who is on the Internet at some location other than at site 1 or site 2 being able to fool one of the routers by pretending to be the other router and connecting to the network. At the current time, the company does not need additional remote access to the networks other than directly between the two sites.

You and I go back to our office and consider the customer's network and requirements. As we consider the VPN options that provide security, we remember that IPsec can perform the following:

- **Confidentiality:** Using symmetrical encryption algorithms such as 3DES, IDEA, AES, and so on to encrypt clear text into cipher text.
- **Data integrity:** Using hashing algorithms such as MD5 or SHA and *Hashed Message Authentication Code (HMAC)* to verify that data has not been manipulated during its transit across the network.
- **Authentication:** Done by authenticating the *virtual private network (VPN)* peers near the beginning of a VPN session, using PSKs or digital signatures (leveraging digital certificates).
- **Hiding the private address space from the Internet:** Because IPsec's *Encapsulation Security Protocol (ESP)* in tunnel mode encrypts and encapsulates the original packet, and then places a new IP header before forwarding the packet, the Internet sees only the packet as being from the global IP address of one router and destined to the global address of the second router.

IPsec technologies and methods look like a perfect fit for the customer. Before we go too much further, you want to verify that the Internet connection for R1 and R2 are working, and that R1 and R2 have reachability to each other. You can do so with a simple ping to the global address of R2 from R1. If there is filtering of *Internet Control Message Protocol (ICMP)*, which is used by the ping utility, it does not necessarily mean that the IPsec will not work, as the protocols for IPsec may still be allowed between the routers. Table 20-2 shows the critical protocols that we may need between R1 and R2.

Table 20-2 *Protocols That May Be Required for IPsec*

Protocol/Port	Who Uses It	How It Is Used
UDP port 500	IKE Phase 1	IKE Phase 1 uses UDP:500 for its negotiation.
UDP port 4500	<i>NAT-T (NAT Traversal)</i>	If both peers support NAT-T, and if they detect that they are connecting to each other through a <i>Network Address Translation (NAT)</i> device (translation is happening), they may negotiate that they want to put a fake UDP port 4500 header on each IPsec packet (before the ESP header) to survive a NAT device that otherwise may have a problem tracking an ESP session (Layer 4 protocol 50).
Layer 4 Protocol 50	ESP	IPsec packets have the Layer 4 protocol of ESP (IP Protocol #50), which is encapsulated by the sender and de-encapsulated by the receiver for each IPsec. packet. ESP is normally used instead of <i>Authentication Header (AH)</i> . The ESP header is hidden behind a UDP header if NAT-T is in use.



Protocol/Port	Who Uses It	How It Is Used
Layer 4 protocol #51	AH	AH packets have the Layer 4 protocol of AH (IP Protocol #51). We do not normally use AH (as opposed to ESP) because AH lacks any encryption capability for user data.

If R1 and R2 have access lists applied inbound on their outside interfaces (G1/0), we would want to ensure that we are allowing the required protocols between the global (Internet) IP addresses of the two routers. Each router needs to believe it could reach the remote networks through specific routes, or at a minimum, a default route. If the router does not have a route, it will not try to forward a packet, and will not trigger any crypto maps that are looking for the interesting traffic. The routing decision happens before IPsec is implemented.

Planning IKE Phase 1

With the connectivity verified, our first planning step is to choose the components to use for the IKE Phase 1 tunnel. (Remember our HAGLE options from a previous chapter.) Table 20-3 lists some of our choices for IKE Phase 1.



Table 20-3 *IKE Phase 1 Policy Options*

Function	Strong Method	Stronger Method
Hashing	MD5, 128-bit	SHA1, 160-bit
Authentication	Pre-shared Key (PSK)	RSA-Sigs (digital signatures)
Group # for DH key exchange	1,2	5
Lifetime	86400 seconds (1 day, default)	Shorter than 1 day, 3600
Encryption	3DES	AES-128 (or 192, or 256)

For the customer, we decide that we will use the stronger options, and will use the following for the IKE phase 1 policy:

- **Hashing:** SHA
- **Authentication:** RSA-Sigs (which require PKI to be used)
- **DH group:** 5
- **Lifetime:** 3600 seconds
- **Encryption:** AES-256

We also note that all of these parameters are to be used for the IKE Phase 1 policy, which we specify using the command `crypto isakmp policy`.

Planning IKE Phase 2

For IKE Phase 2, which is the actual tunnel that will be used to protect the user's packets, we have the elements listed in Table 20-4 to plan for.

Table 20-4 *IKE Phase 2 Policy Options*

Item to Plan	Implemented By	Notes
Peer IP addresses	Crypto map	Having a known reachable IP address for the VPN peer is critical for the traditional IPsec site-to-site tunnel to negotiate and establish the VPN (both phases).
Traffic to encrypt	Crypto ACL, which is referred to in the crypto map	Extended ACL that is not applied to an interface but is referenced in the crypto map. This should <i>only</i> reference outbound (egress) traffic, which should be protected by IPsec. Traffic not matching the crypto ACL will not be encrypted, but will be sent as a normal packet.
Encryption method	Transform set, which is referred to in the crypto map	DES, 3DES, AES are all options. IKE phase 2 does not need to be the same method as Phase 1. The method does need to match the peer's policy (transform sets) for Phase 2.
Hashing (HMAC) method	Transform set, which is referred to in the crypto map	MD5 and SHA HMACs may be used, and need to match the Phase 2 policy of the peer.
Lifetime (time, or data)	Global configuration command: crypto ipsec security-association lifetime ...	Lifetime for Phase 2 should match between the peers. If both use the default lifetime (by not specifying a lifetime), both peers would have compatible lifetime policies. The lifetime can be specified as number of seconds or number of kilobytes.
<i>Perfect Forward Secrecy (PFS)</i> (run DH again or not)	Crypto map	DH is run during IKE Phase 1, and Phase 2 reuses that same keying material that was generated. If you want Phase 2 to rerun the DH, it is called Perfect Forward Secrecy (PFS), and you must choose a DH group number 1,2 or 5 for Phase 2 to use.
Which interface used to peer with the other VPN device	Crypto map applied to the outbound interface	From a routing perspective, this is the interface of a VPN peer that is closest to the other peer, where outbound IPsec packets are leaving the router and inbound IPsec packets are coming into the router.



For our customer, we document and decide to implement the following for IKE Phase 2:

- **VPN Peer global IP addresses:** R1=13.0.0.1 R2=23.0.0.2
- **Traffic to protect:** Bidirectional traffic between 172.16.0.0/16 (R1's local network), and 192.168.0.0/24 (R2's local network).
- **Encryption:** AES-192 (just to mix it up a bit, default for AES is 128)
- **HMAC:** SHA
- **Lifetime:** Default
- **Outside interfaces of routers:** G1/0 (on both)
- **PFS:** Group 2

With our plan in place, our next step is to implement the IPsec tunnel. If you have not done so already, I would like you to write out, on a piece of paper, our plans for both IKE Phase 1 and IKE Phase 2. By doing this, you have a resource you can look at, without turning the pages back and forth, as we implement the IPsec together.

Implementing and Verifying an IPsec Site-to-Site VPN

In this section, we take the information from our planning in the previous section to implement, verify, and troubleshoot the VPN using a combination of *Cisco Configuration Professional (CCP)* and the *command-line interface (CLI)*.

In earlier chapters, we discussed important resources such as *Network Time Protocol (NTP)* and *certificate authorities (CA)*. Because we chose to implement RSA-Signatures for this customer, we want to implement NTP as one of our first steps. This is because when exchanging certificates during IKE Phase 1, if R1 thinks the year is 2040, and the certificate it just received from R2 is listed as being valid from 2012 through 2016, R1 will reject the certificate as not being valid, and IKE Phase 1 will not end well. (If IKE Phase 1 does not work, IKE Phase 2 does not have a chance either.) So for this implementation, we use a service provider on the simulated Internet (in our customer's topology) that will provide both NTP and CA services at the address of 3.3.3.3 (in the simulated Internet portion of the topology diagram).

Let's synchronize the time on both R1 and R2 with the service provider's NTP server, as shown in Figure 20-2.

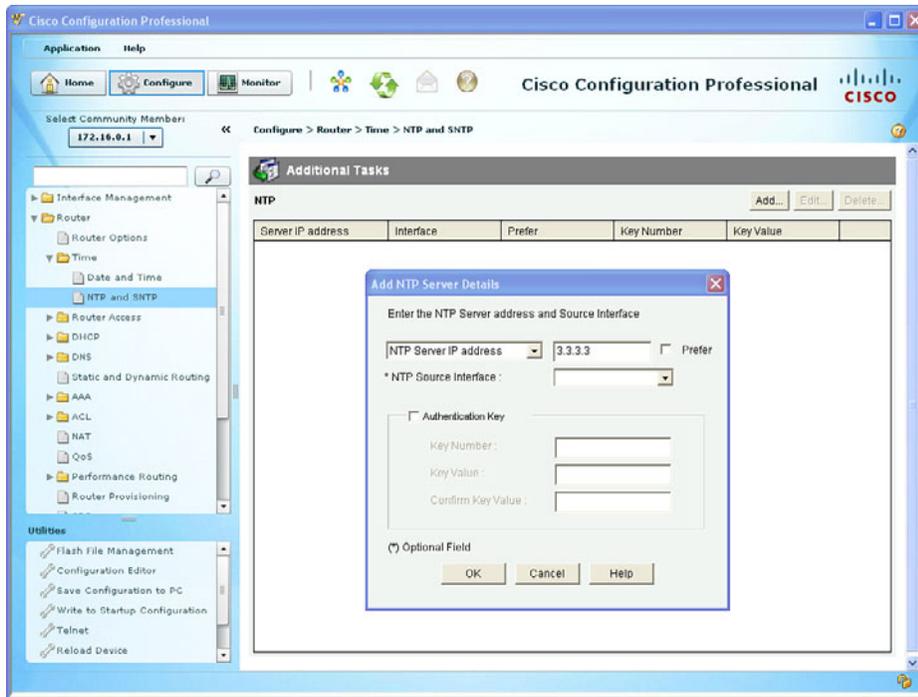


Figure 20-2 *Configuring the Router to Use NTP Services*

The same process would be repeated for the other router as well. NTP can take up to 15 minutes to synchronize. Another item to be aware of is that NTP servers deliver time based on coordinated universal time (UTC), and setting the local time zone on your router is important so that correct offset from UTC is reflected on the local router. To verify that the time is synchronized with the time server, at the CLI of the router we can use the commands shown in Example 20-1.

Example 20-1 *Verifying NTP Status*

```
R1# show ntp status
Clock is unsynchronized, stratum 16, no reference clock
nominal freq is 250.0000 Hz, actual freq is 249.9999 Hz, precision is 2**24
reference time is D2C15194.71E5E637 (14:11:32.444 UTC Wed Jan 18 2012)
clock offset is 0.0000 msec, root delay is 0.00 msec
root dispersion is 0.00 msec, peer dispersion is 0.00 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000085 s/s
system poll interval is 64, last update was 1518 sec ago.
```

```
! Note the above indicates the time isn't synchronized.
! We can check to see if the router has the NTP server configured with the
! following:
```

```

R1# show ntp association

address          ref clock      st  when  poll reach  delay  offset  disp
~3.3.3.3         .INIT.        16   -    64    0  0.000  0.000 16000.
* sys.peer, # selected, + candidate, - outlyer, x falseticker, ~ configured

! Based on this output, we know that it has information to use the 3.3.3.3
! server
! It may take anywhere from 5 to 15 minutes for
! the synchronization to happen.    After verifying the configuration, and
! waiting about 5 minutes, we can then issue the verification commands
! again and see that the synchronization is complete.

R1# show ntp status
Clock is synchronized, stratum 3, reference is 3.3.3.3
nominal freq is 250.0000 Hz, actual freq is 249.9999 Hz, precision is 2**24
reference time is D2C15854.6F453DAE (14:40:20.434 UTC Wed Jan 18 2012)
clock offset is 0.0029 msec, root delay is 0.01 msec
root dispersion is 0.95 msec, peer dispersion is 0.06 msec
loopfilter state is 'CTRL' (Normal Controlled Loop), drift is 0.000000097 s/s
system poll interval is 64, last update was 251 sec ago.

```

Our next task, in preparation for the IPsec, is to generate key pairs on R1 and R2, configure them to use a CA, have them authenticate the CA (get the root certificate), and then enroll with the CA (request their own identity certificates). The CA is at 3.3.3.3 and supports *Simple Certificate Enrollment Protocol (SCEP)*. From R1 and R2, the process is the same, and the commands used for R1 are shown in Example 20-2.

Example 20-2 *Preparing for and Obtaining Digital Certificates*

```

! Specify the domain-name that will be included with the key pair you
! are about to generate
! Note: if you have already created a key-pair to be used with SSH
! you don't need to create a separate key-pair.  You can use the same
! key pair for both purposes if desired.
R1(config)# ip domain name cisco.com
R1(config)# crypto key generate rsa
The name for the keys will be: R1.cisco.com
Choose the size of the key modulus in the range of 360 to 2048 for your
  General Purpose Keys. Choosing a key modulus greater than 512 may take
  a few minutes.
! The larger the key the better.  Using a minimum length of 1024 is a best
! practice
How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable... [OK]

```

```

! Specify the CA that you would like to use, and the URL to be used to
! reach that CA
R1(config)# crypto pki trustpoint CA
R1(ca-trustpoint)# enrollment URL http://3.3.3.3
R1(ca-trustpoint)# exit

! Request the root certificate through "authenticating" the CA
R1(config)# crypto pki authenticate CA
Certificate has the following attributes:
    Fingerprint MD5: B1AF5247 21F35FE3 0200F345 7C20FBA0
    Fingerprint SHA1: F5BB33E3 1CB5D633 0DF720DF 8C72CD48 E744CF5B

% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.

! Request an Identity certificate for this router, via SCEP and the
! "enroll" option
R1(config)# crypto pki enroll CA
%
% Start certificate enrollment ..
% Create a challenge password. You will need to verbally provide this
    password to the CA Administrator in order to revoke your certificate.
    For security reasons your password will not be saved in the configuration.
    Please make a note of it.

! Specifying the challenge password that can be used in the event you need to
! ask the CA to revoke this certificate in the future
Password: SuperSecret!23
Re-enter password: SuperSecret!23

% The subject name in the certificate will include: R1.cisco.com

! The next 2 items are optional elements that may be included in the certificate
% Include the router serial number in the subject name? [yes/no]: no
% Include an IP address in the subject name? [no]: no
Request certificate from CA? [yes/no]: yes
% Certificate request sent to Certificate Authority
% The 'show crypto pki certificate verbose CA' command will show the fin-
    gerprint.

CRYPTO_PKI: Certificate Request Fingerprint MD5: E8E01D26 862C811C
    32CB3FCF 858BAF5F
CRYPTO_PKI: Certificate Request Fingerprint SHA1: E3133B07 07DEA5FD
    BC6A1D64 DBC9F71F 3CACA767
%PKI-6-CERTRET: Certificate received from Certificate Authority

! We would repeat this process on the other router, R2

```

After we have digital certificates on both routers, we can configure the IKE Phase 1 policy. This can be done in CCP by navigating to **Configure > Security > VPN > Site-to-Site VPN**, as shown in Figure 20-3.

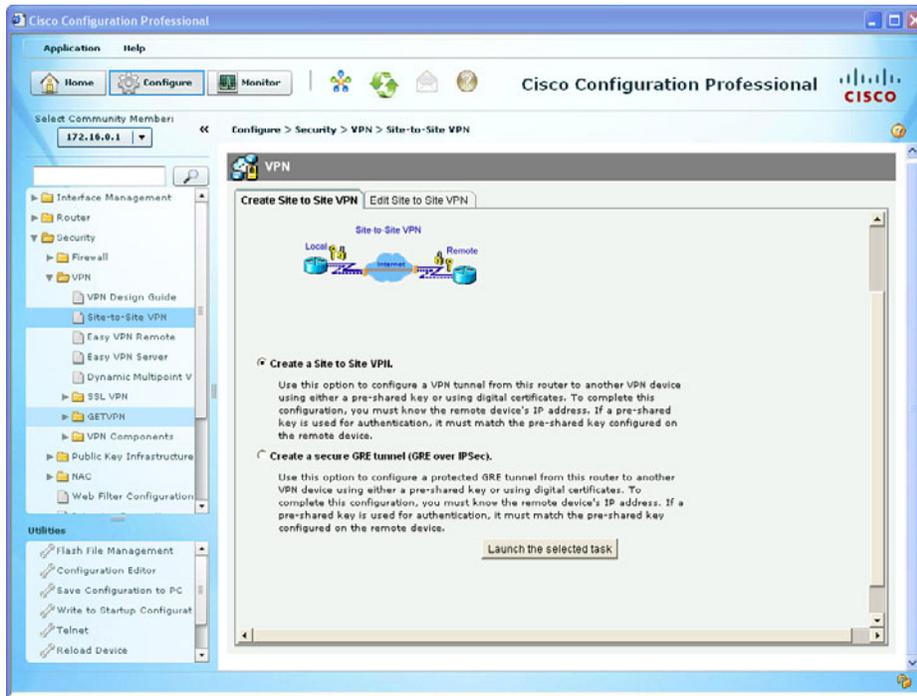


Figure 20-3 Using the Site-to-Site VPN Wizard in CCP

Using the **Launch the Selected Task** button to continue, we choose the **Step-by-Step Wizard** (in contrast to Quick Setup), and then click the **Next** button. We have the opportunity to begin entering the information we collected earlier about the interfaces to use and the policies to implement, as shown in Figure 20-4.

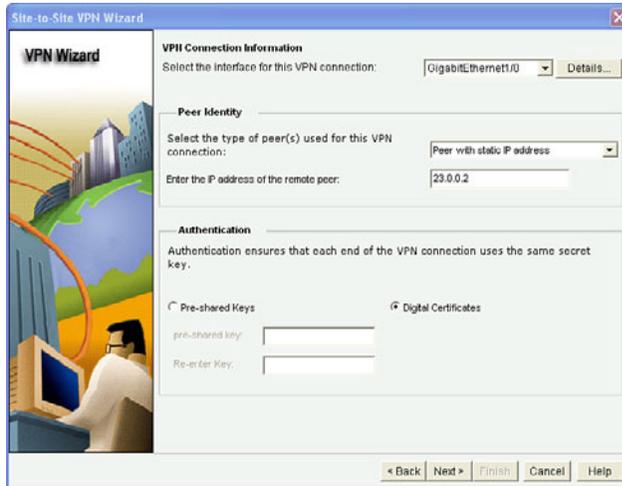


Figure 20-4 *Supplying the Wizard with the Information from Our Design for This Customer*

Notice that this time, because of our policy, we select the **Digital Certificates** radio button instead of using Pre-Shared Keys for the authentication. After clicking **Next**, we are presented with the default IKE Phase 1 policy, which is built in to CCP. To add a new policy, click the **Add** button and enter the details for the IKE Phase 1 policy per the plan, as shown in Figure 20-5.

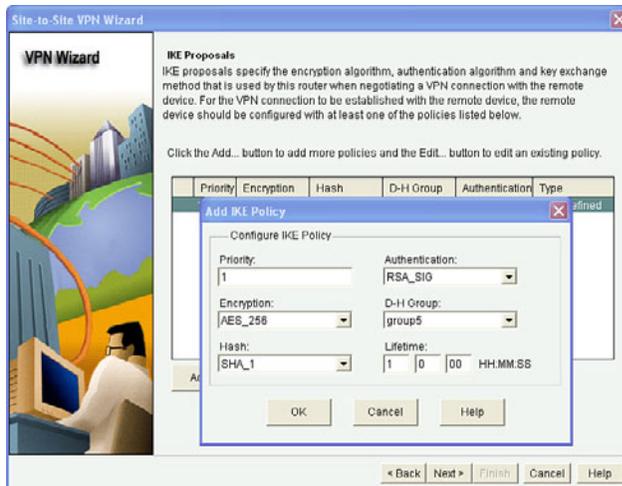


Figure 20-5 *Entering the IKE Phase 1 Policy Details*

After you click **OK**, highlight the new policy, and then click **Next** to advance to the IKE Phase 2 policy information. Once there, click **Add** and enter the IPsec/IKE Phase 2 options that we planned earlier, as shown in Figure 20-6.

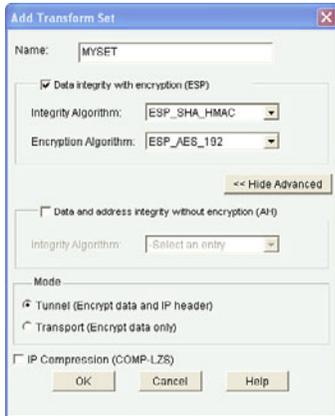


Figure 20-6 *Configuring the IKE Phase 2/Transform Set Details*

Unfortunately, the options for setting up PFS for IKE Phase 2 are not integrated in the wizard. Let's finish the wizard, to confirm the ACL information, by clicking **OK** for the transform set, clicking the transform set to highlight it, and clicking **Next** to continue. In the next window, we specify which traffic to encrypt. Remember that this is from the local router's perspective of outbound traffic that it should apply IPsec to. Figure 20-7 shows an example of implementing our crypto ACLs via the wizard.

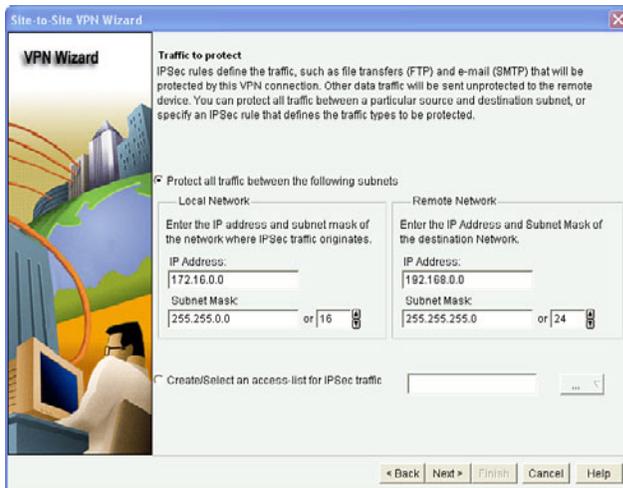


Figure 20-7 *Configuring the Crypto ACL Information*

On R1, the ACL matches on traffic from 172.16.0.0/16 that has a destination in the 192.168.0.0/24 network. The ACL on R2 must be a mirror image of the source and destination networks. When you click **Next**, a summary is provided, and then based on your preference settings in CCP, the actual commands about to be applied may be presented in a final window before you approve them to be delivered.

Example 20-3 shows the CLI implementation of the crypto policy for R1, including the PFS (which the wizard did not offer).

Example 20-3 *CLI Implementation of the Crypto Policy for R1*



```
R1(config)# crypto isakmp policy 1
R1(config-isakmp)# encr aes 256
R1(config-isakmp)# group 5
R1(config-isakmp)# lifetime 3600
R1(config-isakmp)# authentication rsa-sig
R1(config-isakmp)# hash sha

! To verify the configuration:
R1# show crypto isakmp policy

Global IKE policy
Protection suite of priority 1
  encryption algorithm:  AES - Advanced Encryption Standard (256 bit
                        keys) .
  hash algorithm:       Secure Hash Standard
  authentication method: Rivest-Shamir-Adleman Signature
  Diffie-Hellman group:  #5 (1536 bit)
  lifetime:             3600 seconds, no volume limit
! Note, that a show running-config, would only show configured items in the
! policy if they were different from the default.  Here is a snippet from
! the show run:

crypto isakmp policy 1
  encr aes 256
  group 5
  lifetime 3600

! Because the authentication and hash are using the defaults, they are not
! shown even though we put them in the configuration.  (interesting to know)

! We won't need a pre-shared key, because we are using digital signatures/
! certificates for the IKE phase 1 authentication.

! Next we can create our transform-set, and crypto ACL, which will be
! placed inside the crypto map.  The crypto map will be applied to the
! interface of the router.
```

```
! Transform set details the encryption and HMAC to use
R1(config)# crypto ipsec transform-set MYSET esp-aes esp-sha-hmac
R1(cfg-crypto-trans)# exit

! Crypto ACL identifies which traffic (outbound) should be encrypted
R1(config)# access-list 100 permit ip 172.16.0.0 0.0.255.255 192.168.0.0
0.0.0.255

! The crypto map contains the if/then statement to decide to encrypt or
! not to encrypt a packet on its way out
R1(config)# crypto map MYMAP 1 ipsec-isakmp
R1(config-crypto-map)# match address 100
R1(config-crypto-map)# set peer 23.0.0.2
R1(config-crypto-map)# set transform-set MYSET

! Here is the PFS part that we are adding manually, as the wizard didn't
! support this feature
R1(config-crypto-map)# set pfs group2
R1(config-crypto-map)# exit

! Applying the crypto map to the interface is what allows the entire IPsec
! function to be triggered. That is why it is important that the router
! has at least a default route (if not a more specific route) out of this
! interface to reach the remote network for which the router is providing
! IPsec support.
! When the router considers forwarding traffic out the interface, that is
! what triggers the decision to encrypt or not. If the traffic matches
! the crypto ACL in the crypto map, the router will encrypt the original
! packet, encapsulate the encrypted packet into a new packet with ESP as
! the L4 header, and the peer's global IP address as the new L3 header.
! If no IPsec SA (tunnel) is in place yet, this will also trigger the
! negotiations to build the tunnel, including the IKE phase 1 if it is not
! already in place.
R1(config)# interface GigabitEthernet1/0
R1(config-if)# crypto map MYMAP
R1(config-if)# exit
```

After the appropriate compatible configuration has been placed on R2, we should be able to encrypt traffic between the two networks using IPsec.

Troubleshooting IPsec Site-to-Site VPNs

When implementing a new VPN, there may be a problem or two. So, let's walk through the verification of the tunnel, and as we discover it not working, I will show you some of my favorite commands to assist in the troubleshooting process.

Let's first of all verify our configuration so that we can confirm that what we have running on the router is what we configured, starting with Example 20-4.

Example 20-4 *Verifying the IPsec Configuration*

```
! This verifies the IKE phase 1 policy or policies in place. The lower
! the number of the policy, the higher it's priority.

R1# show crypto isakmp policy

Global IKE policy
Protection suite of priority 1
  encryption algorithm: AES - Advanced Encryption Standard (256 bit
    keys).
  hash algorithm: Secure Hash Standard
  authentication method: Rivest-Shamir-Adleman Signature
  Diffie-Hellman group: #5 (1536 bit)
  lifetime: 3600 seconds, no volume limit

! Next is my favorite command, as it shows virtually all of the rest of the
! config including the transform set and crypto ACLs involved, and where
! the crypto map is applied

R1# show crypto map

Crypto Map "MYMAP" 1 ipsec-isakmp
  Peer = 23.0.0.2
  Extended IP access list 100
    access-list 100 permit ip 172.16.0.0 0.0.255.255 192.168.0.0
      0.0.0.255
  Current peer: 23.0.0.2
  Security association lifetime: 4608000 kilobytes/3600 seconds
  Responder-Only (Y/N): N
  PFS (Y/N): Y
  DH group: group2
  Transform sets={
    MYSET: { esp-aes esp-sha-hmac } ,
  }
  Interfaces using crypto map MYMAP:
    GigabitEthernet1/0
```

Armed with this information, a packet from the 172.16.0.0 network destined to the 192.168.0.0 network should trigger the IPsec process. We can test this without even leaving the router console. Because R1 is connected to the 172.16.0.0 network, we can craft a ping request sourced from that network (interface g2/0) and destined for the 192.168.0.2 address of R2, as shown in Example 20-5.

Example 20-5 *Interesting Traffic to Trigger IPsec*

```
R1# ping 192.168.0.2 source g1/0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.0.2, timeout is 2 seconds:
Packet sent with a source address of 13.0.0.1
U.U.U
Success rate is 0 percent (0/5)
R1#
```

The ping is being replied to with U.U.U messages. The *U* represents an ICMP unreachable message being sent to us from one of the Internet routers (probably our directly connected one). If our policy was not applied correctly, it may be possible that we are trying to send packets to 192.168.0.2, and when the service providers see these packets, they are denying them because they addresses in the RFC 1918 address space (which are not being allowed on the Internet). If the packet had been successfully encapsulated into IPsec's Layer 4 protocol 50 (ESP), the Internet would have seen a packet destined to the Layer 3 global address of 23.0.0.2, sourced from the global address of R1.

Let's begin the troubleshooting! We may want to go to R2, and do the same **show** commands we did on R1. If R2 is not accessible via the CLI at the moment, we could also do some additional testing at R1 using **debug** commands specifically for IPsec Phase 1 and Phase 2. For IKE Phase 1 debugging, we could use the commands shown in Example 20-6.

Example 20-6 *Debug Used for Troubleshooting IKE Phase 1*

```
! debug the IKE phase 1 process
R1# debug crypto isakmp
Crypto ISAKMP debugging is on

! Generate interesting traffic that should match the ACLs used in the
  crypto map
R1# ping 192.168.0.2 source g1/0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.0.2, timeout is 2 seconds:
Packet sent with a source address of 13.0.0.1
U.U.U
Success rate is 0 percent (0/5)
```

With the debugging of IKE Phase 1 on, and then using the ping again, we see no output from the debug. This implies that either IKE Phase 1 is already up and does not need to be negotiated, or if it is currently not up, that no interesting traffic is triggering it. This could be due to a down interface, a misapplied crypto map, or routing that is not trying to forward traffic out the interface that has the crypto map applied. Let's take a closer look, as shown in Example 20-7.

Example 20-7 *Troubleshooting by Verifying Configuration*

```
R1# show crypto map
Crypto Map "MYMAP" 1 ipsec-isakmp
  Peer = 23.0.0.2
  Extended IP access list 100
    access-list 100 permit ip 172.16.0.0 0.0.255.255 192.168.0.0
    0.0.0.255
  Current peer: 23.0.0.2
  Security association lifetime: 4608000 kilobytes/3600 seconds
  Responder-Only (Y/N): N
  PFS (Y/N): Y
  DH group: group2
  Transform sets={
    MYSET: { esp-aes esp-sha-hmac } ,
  }
  Interfaces using crypto map MYMAP:
    GigabitEthernet1/0

R1# show ip int brief
Interface          IP-Address  OK? Method  Status          Protocol
FastEthernet0/0    unassigned  YES unset      administratively down  down
GigabitEthernet1/0 13.0.0.1   YES manual  up              up
GigabitEthernet2/0 172.16.0.1 YES manual  up              up

! Just as before, the crypto map appears to be applied to the correct
! interface, and the interfaces are both up.  Let's next check to see if
! there is an IKE phase 1 tunnel already in place

R1# show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst                src                state              conn-id status

! The output of the above command indicates that there is no current IKE
! phase 1 tunnel in place.  Let's check routing, to see if R1 would even
! try to forward packets to 192.168.0.2 through its G1/0 interface.
```

```
R1# show ip route
Codes: C - connected, S - static, 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
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS
         level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static
         route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is 13.0.0.3 to network 0.0.0.0

C    172.16.0.0/16 is directly connected, GigabitEthernet2/0
    13.0.0.0/24 is subnetted, 1 subnets
C      13.0.0.0 is directly connected, GigabitEthernet1/0
S*   0.0.0.0/0 [1/0] via 13.0.0.3

! The show IP route output indicates that R1 would use its G1/0 interface
! to use the default gateway of 13.0.0.3. We know that the internet router
! is there, because we received the U messages earlier from him.

! Sometimes, when making configuration changes regarding IPsec,
! the VPN device may become confused. If there is no VPN traffic working,
! I have found that removing and re-applying the crypto map on the
! interface is often helpful in re-starting the IPsec process on the
! router. Let's try that next on R1.

R1# conf t
Enter configuration commands, one per line. End with CNTL/Z.
R1(config)# int g1/0
R1(config-if)# no crypto map MYMAP
R1(config-if)# crypto map MYMAP
%CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is OFF
R1(config-if)# crypto map MYMAP
R1(config-if)#
%CRYPTO-6-ISAKMP_ON_OFF: ISAKMP is ON
R1(config-if)#

! I see that ISAKMP turned itself off, when the map was removed and was
! turned back on when the map was replaced, based on the console messages.
! ISAKMP stands for Internet Security Association Key Management Protocol
! Now let's try the ping again.
```

```

! Debugging is still on.

R1# show debug

Cryptographic Subsystem:
  Crypto ISAKMP debugging is on

R1# ping 192.168.0.2 source g1/0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.0.2, timeout is 2 seconds:
Packet sent with a source address of 13.0.0.1
U.U.U
Success rate is 0 percent (0/5)
R1#

```

Same result as before. So, why are we going through the pain of troubleshooting? Because this is a skill that you deserve to have, and by going through it together it will make your skills better for the real world. Before we go too much further, let's pause and examine our test ping. A common mistake people make is assuming the VPN should come up, even if there is no interesting traffic (matching the crypto ACLs).

In our ping, we are sourcing the ping from the G1/0 interface (this interface is not the one in the 172.16.0.0/16 network), and therefore as a result the packet is not matching the crypto ACL. The router does not think it should apply IPsec to it, so it sends the ping out as a plain text. The original source and destination IP addresses have not changed, which would cause the service provider to deny that traffic.

Armed with the knowledge that our test ping had an issue, let's leave the debug on and try the ping, as shown in Example 20-8.

Example 20-8 *Test Ping Using the Correct Source Interface and Associated IP Address*

```

R1# ping 192.168.0.2 source g2/0

! Although there is lots of interesting output, I will point out the
! more relevant information regarding our troubleshooting

Sending 5, 100-byte ICMP Echos to 192.168.0.2, timeout is 2 seconds:

! Note the correct source address, important if we want to match the crypto
! acls
Packet sent with a source address of 172.16.0.1

```

```

ISAKMP:(0): SA request profile is (NULL)
ISAKMP: Created a peer struct for 23.0.0.2, peer port 500
ISAKMP: New peer created peer = 0x6A76F7A0 peer_handle = 0x80000005
ISAKMP: Locking peer struct 0x6A76F7A0, refcount 1 for isakmp_initiator
ISAKMP: local port 500, remote port 500
ISAKMP: set new node 0 to QM_IDLE
ISAKMP:(0):insert sa successfully sa = 66570618
ISAKMP:(0):Can not start Aggressive mode, trying Main mode. ISAKMP:
(0):No pre-shared key with 23.0.0.2!
ISAKMP:(0): constructed NAT-T vendor-rfc3947 ID
ISAKMP:(0): constructed NAT-T vendor-07 ID
ISAKMP:(0): constructed NAT-T vendor-03 ID
ISAKMP:(0): constructed NAT-T vendor-02 ID
ISAKMP:(0):Input = IKE_MESG_FROM_IPSEC, IKE_SA_REQ_MM
ISAKMP:(0):Old State = IKE_READY New State = IKE_I_MM1

ISAKMP:(0): beginning Main Mode exchange

! R1 is the initiator, and so he is sending the first packet, trying
! to negotiate a compatible IKE phase 1 policy with R2
ISAKMP:(0): sending packet to 23.0.0.2 my_port 500 peer_port 500 (I) MM_NO_
STATE
ISAKMP:(0):Sending an IKE IPv4 Packet.
ISAKMP (0): received packet from 23.0.0.2 dport 500 sport 500 Global (I)
MM_NO_STATE
ISAKMP:(0):Notify has no hash. Rejected.
ISAKMP (0): Unknown Input IKE_MESG_FROM_PEER, IKE_INFO_NOTIFY: state =
IKE_I_MM1
ISAKMP:(0):Input = IKE_MESG_FROM_PEER, IKE_INFO_NOTIFY
ISAKMP:(0):Old State = IKE_I_MM1 New State = IKE_I_MM1

! This line below is bad news. IKE phase 1 failed.%CRYPTO-6-
IKMP_MODE_FAILURE: Processing of Informational mode failed with
peer at 23.0.0.2

! And our pings didn't make it either (no VPN tunnel working yet).
.....
Success rate is 0 percent (0/5)

! The IKE phase 1 tunnel has a state of MM_NO_STATE which is not good
! We want to see a state of QM_IDLE, meaning the IKE phase 1 is up, in the
! output of the following command, that shows the state of the IKE Phase 1
! tunnel

R1# show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst          src          state          conn-id status
23.0.0.2     13.0.0.1     MM NO STATE    0 ACTIVE

```

Perhaps R2 is not configured correctly. If IKE Phase 1 had completed, we could investigate IKE Phase 2, but due to Phase 1 failing, that is the first thing to check on R2. If the IKE Phase 1 policy matches on R2, we would also want to verify that R2 has a digital certificate to use with the RSA-Signatures. Let's look at R2's policy, as shown in Example 20-9.

Example 20-9 *Verifying the Configuration on R2*

```
R2# show crypto isakmp policy
Global IKE policy
Protection suite of priority 1
    encryption algorithm:  Three key triple DES
    hash algorithm:        Secure Hash Standard
    authentication method: Rivest-Shamir-Adleman Signature
    Diffie-Hellman group:  #5 (1536 bit)
    lifetime:              3600 seconds, no volume limit

! Based on the output, it appears the encryption algorithm for R2's IKE
! Phase 1 is set for 3DES, and R1 was set for AES 256. That is a problem.
! Let's make the change on R2, enable debugging, and see if we get a better
! result.

! Change the policy on R2
R2(config)# crypto isakmp policy 1
R2(config-isakmp)# encryption aes 256
R2(config-isakmp)# end

! Enable debug of IKE phase 1 and issue the ping from R2 to trigger the
! crypto ACLs (which are in the crypto map, which is applied to the
! interface)
R2# debug crypto isakmp
Crypto ISAKMP debugging is on

R2# ping 172.16.0.1 source g2/0

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.1, timeout is 2 seconds:
Packet sent with a source address of 192.168.0.2

ISAKMP:(0): SA request profile is (NULL)
ISAKMP: Created a peer struct for 13.0.0.1, peer port 500
ISAKMP: New peer created peer = 0x6816E21C peer_handle = 0x80000006
ISAKMP: Locking peer struct 0x6816E21C, refcount 1 for isakmp_initiator
ISAKMP: local port 500, remote port 500
ISAKMP: set new node 0 to QM_IDLE
ISAKMP:(0):insert sa successfully sa = 671E34DC
```

```

! The two modes for IKE phase 1 are aggressive, or main.  R2 is going to
! use main mode.
ISAKMP:(0):Can not start Aggressive mode, trying Main mode.

! R2 won't be needing a pre-shared key with R1 for authentication, due to
! it using digital signatures
ISAKMP:(0):No pre-shared key with 13.0.0.1!
ISAKMP:(0): constructed NAT-T vendor-rfc3947 ID
ISAKMP:(0): constructed NAT-T vendor-07 ID
ISAKMP:(0): constructed NAT-T vendor-03 ID
ISAKMP:(0): constructed NAT-T vendor-02 ID
ISAKMP:(0):Input = IKE_MESG_FROM_IPSEC, IKE_SA_REQ_MM
ISAKMP:(0):Old State = IKE_READY  New State = IKE_I_MM1

ISAKMP:(0): beginning Main Mode exchange
ISAKMP:(0): sending packet to 13.0.0.1 my_port 500 peer_port 500 (I) MM_NO_
STATE
ISAKMP:(0):Sending an IKE IPv4 Packet.
ISAKMP (0): received packet from 13.0.0.1 dport 500 sport 500 Global (I)
MM_NO_STATE
ISAKMP:(0):Input = IKE_MESG_FROM_PEER, IKE_MM_EXCH
ISAKMP:(0):Old State = IKE_I_MM1  New State = IKE_I_MM2

ISAKMP:(0): processing SA payload. message ID = 0
ISAKMP:(0): processing vendor id payload
ISAKMP:(0): vendor ID seems Unity/DPD but major 69 mismatch
ISAKMP (0): vendor ID is NAT-T RFC 3947
ISAKMP : Scanning profiles for xauth ...

! Looks like these two peers have found a compatible policy.  It is the
! contents of the policy that need to be compatible, not the literal policy
! priority number  The debug shows the word "transform" but should not be
! confused with IKE Phase 2 which occurs only after IKE Phase 1 (whose
! policy is shown below) is complete.
ISAKMP:(0):Checking ISAKMP transform 1 against priority 1 policy
ISAKMP:      encryption AES-CBC
ISAKMP:      keylength of 256
ISAKMP:      hash SHA
ISAKMP:      default group 5
ISAKMP:      auth RSA sig
ISAKMP:      life type in seconds
ISAKMP:      life duration (basic) of 3600
! The peers have agreed on the IKE Phase 1 policy.
ISAKMP:(0):atts are acceptable. Next payload is 0

```

```

ISAKMP:(0):Acceptable atts:actual life: 0
ISAKMP:(0):Acceptable atts:life: 0
ISAKMP:(0):Basic life_in_seconds:3600
ISAKMP:(0):Returning Actual lifetime: 3600
ISAKMP:(0)::Started lifetime timer: 3600.

ISAKMP:(0): processing vendor id payload
ISAKMP:(0): vendor ID seems Unit.y/DPD but major 69 mismatch
ISAKMP (0): vendor ID is NAT-T RFC 3947
ISAKMP:(0):Input = IKE_MESG_INTERNAL, IKE_PROCESS_MAIN_MODE
ISAKMP:(0):Old State = IKE_I_MM2 New State = IKE_I_MM2
! Next 40 or so lines include getting the certificate from the other side,
! so that this router will have a copy of the peer's public key, and
! performing RSA authentication with the peer.
ISAKMP (0): constructing CERT_REQ for issuer cn=CA
ISAKMP:(0): sending packet to 13.0.0.1 my_port 500 peer_port 500 (I) MM_SA_
SETUP
ISAKMP:(0):Sending an IKE IPv4 Packet.
ISAKMP:(0):Input = IKE_MESG_INTERNAL, IKE_PROCESS_COMPLETE
ISAKMP:(0):Old State = IKE_I_MM2 New State = IKE_I_MM3

ISAKMP (0): received packet from 13.0.0.1 dport 500 sport 500 Global (I)
MM_SA_SETUP
ISAKMP:(0):Input = IKE_MESG_FROM_PEER, IKE_MM_EXCH
ISAKMP:(0):Old State = IKE_I_MM3 New State = IKE_I_MM4

ISAKMP:(0): processing KE payload. message ID = 0
ISAKMP:(0): processing NONCE payload. message ID = 0
ISAKMP:(1004): processing CERT_REQ payload. message ID = 0
ISAKMP:(1004): peer wants a CT_X509_SIGNATURE cert
ISAKMP:(1004): peer wants cert issued by cn=CA
    Choosing trustpoint CA as issuer
ISAKMP:(1004): processing vendor id payload
ISAKMP:(1004): vendor ID is Unity
ISAKMP:(1004): processing vendor id payload
ISAKMP:(1004): vendor ID is DPD
ISAKMP:(1004): processing vendor id payload
ISAKMP:(1004): speaking to another IOS box!
ISAKMP:received payload type 20
ISAKMP (1004): His hash no match - this node outside NAT
ISAKMP:received payload type 20
ISAKMP (1004): No NAT Found for self or peer
ISAKMP:(1004):Input = IKE_MESG_INTERNAL, IKE_PROCESS_MAIN_MODE
ISAKMP:(1004):Old State = IKE_I_MM4 New State = IKE_I_MM4

```

```

ISAKMP:(1004):Send initial contact
ISAKMP:(1004):My ID configured as IPv4 Addr, but Addr not in Cert!
ISAKMP:(1004):Using FQDN as My ID
ISAKMP:(1004):SA is doing RSA signature authentication using id type ID_
FQDN
ISAKMP (1004): ID payload
    next-payload : 6
    type         : 2
    FQDN name    : R2.cisco.com
    protocol     : 17
    port        : 500
    length      : 20
ISAKMP:(1004):Total payload length: 20
ISAKMP (1004): constructing CERT payload for hostnam.e=R2.cisco.com
ISAKMP:(1004): using the CA trustpoint's keypair to sign
ISAKMP:(1004): sending packet to 13.0.0.1 my_port 500 peer_port 500 (I) MM_
KEY_EXCH
ISAKMP:(1004):Sending an IKE IPv4 Packet.
ISAKMP:(1004):Input = IKE_MSG_INTERNAL, IKE_PROCESS_COMPLETE
ISAKMP:(1004):Old State = IKE_I_MM4  New State = IKE_I_MM5

ISAKMP (1004): received packet from 13.0.0.1 dport 500 sport 500 Global (I)
MM_KEY_EXCH
ISAKMP:(1004): processing ID payload. message ID = 0
ISAKMP (1004): ID payload
    next-payload : 6
    type         : 2
    FQDN name    : R1.cisco.com
    protocol     : 17
    port        : 500
    length      : 20
ISAKMP:(0):: peer matches *none* of the profiles
ISAKMP:(1004): processing CERT payload. message ID = 0
ISAKMP:(1004): processing a CT_X509_SIGNATURE cert
ISAKMP:(1004): peer's pubkey is cached
ISAKMP:(1004): Unable to get DN from certificate!
ISAKMP:(1004): Cert presented by peer contains no OU field.
ISAKMP:(0):: peer matches *none* of the profiles
ISAKMP:(1004): processing SIG payload. message ID = 0
ISAKMP:(1004):SA authentication status:
    authenticated

! The IKE phase 1 authentication completed successfully
ISAKMP:(1004):SA has been authenticated with 13.0.0.1
ISAKMP: Trying to insert a peer 23.0.0.2/13.0.0.1/500/, and inserted suc-
cessfully 6816E21C.

```

```

ISAKMP:(1004):Input = IKE_MSG_FROM_PEER, IKE_MM_EXCH
ISAKMP:(1004):Old State = IKE_I_MM5 New State = IKE_I_MM6

ISAKMP:(1004):Input = IKE_MSG_INTERNAL, IKE_PROCESS_MAIN_MODE
ISAKMP:(1004):Old State = IKE_I_MM6 New State = IKE_I_MM6

ISAKMP:(1004):Input = IKE_MSG_INTERNAL, IKE_PROCESS_COMPLETE
ISAKMP:(1004):Old State = IKE_I_MM6 New State = IKE_P1_COMPLETE

! Now that IKE Phase 1 is complete, IKE Phase 2 (quick mode) can begin.
ISAKMP:(1004):beginning Quick Mode exchange, M-ID of -534639709
ISAKMP:(1004):QM Initiator gets spi
ISAKMP:(1004): sending packet to 13.0.0.1 my_port 500 peer_port 500 (I)
QM_IDLE
ISAKMP:(1004):Sending an IKE IPv4 Packet.
ISAKMP:(1004):Node -534639709, Input = IKE_MSG_INTERNAL, IKE_INIT_QM
ISAKMP:(1004):Old State = IKE_QM_READY New State = IKE_QM_I_QM1
ISAKMP:(1004):Input = IKE_MSG_INTERNAL, IKE_PHASE1_COMPLETE

ISAKMP:(1004):Old State = IKE_P1_COMPLETE New State = IKE_P1_COMPLETE

ISAKMP (1004): received packet from 13.0.0.1 dport 500 sport 500 Global (I)
QM_IDLE
ISAKMP: set new node -325744431 to QM_IDLE
ISAKMP:(1004): processing HASH payload. message ID = -325744431
ISAKMP:(1004): processing NOTIFY PROPOSAL_NOT_CHOSEN protocol 3
spi 3138923289, message ID = -325744431, sa = 671E34DC
ISAKMP:(1004): deleting spi 3138923289 message ID = -534639709
ISAKMP:(1004):deleting node -534639709 error TRUE reason "Delete Larval"
ISAKMP:(1004):deleting node -325744431 error FALSE reason "Informational
(in) state 1"
ISAKMP:(1004):Input = IKE_MSG_FROM_PEER, IKE_INFO_NOTIFY
ISAKMP:(1004):Old State = IKE_P1_COMPLETE New State = IKE_P1_COMPLETE

! Bad news, the IKE phase 2 (the IPsec tunnel) didn't come up and allow the
! pings to work. All 5 pings are lost. The missing 3 periods are
! embedded in the debug messages above.
..
Success rate is 0 percent (0/5)
R2#

```

With IKE Phase 1 working, let's focus on IKE Phase 2 and see whether we can resolve the problem (because the pings did not make it through) by comparing the IKE Phase 2 components on both R1 and R2, as shown in Example 20-10.

Example 20-10 Troubleshooting IKE Phase 2, the IPsec Tunnel

**Key
Topic**

```
R1# show crypto map
Crypto Map "MYMAP" 1 ipsec-isakmp
  Peer = 23.0.0.2
  Extended IP access list 100
    access-list 100 permit ip 172.16.0.0 0.0.255.255 192.168.0.0
    0.0.0.255
  Current peer: 23.0.0.2
  Security association lifetime: 4608000 kilobytes/3600 seconds
  Responder-Only (Y/N): N
  PFS (Y/N): Y
  DH group: group2
  Transform sets={
    MYSET: { esp-aes esp-sha-hmac } ,
  }
  Interfaces using crypto map MYMAP:
    GigabitEthernet1/0

! Lets check the other router
R2# show crypto map
Crypto Map "MYMAP" 1 ipsec-isakmp
  Peer = 13.0.0.1
  Extended IP access list 100
    access-list 100 permit ip 192.168.0.0 0.0.0.255 172.16.0.0
    0.0.255.255
  Current peer: 13.0.0.1
  Security association lifetime: 4608000 kilobytes/3600 seconds
  Responder-Only (Y/N): N
  PFS (Y/N): N
  Transform sets={
    MYSET: { esp-aes esp-sha-hmac } ,
  }
  Interfaces using crypto map MYMAP:
    GigabitEthernet1/0

! Based on the output, it looks like R1 is configured to use PFS group 2,
! and R2 isn't. Let's correct this on R2, and retry the ping.

R2(config)# crypto map MYMAP 1 ipsec-isakmp
R2(config-crypto-map)# set pfs group2
R2(config-crypto-map)# end
```



```

    renc - RSA encryption
IPv4 Crypto ISAKMP SA

C-id  Local      Remote      I-VRF      Status Encr Hash Auth DH Lifetime Cap.

1004  23.0.0.2    13.0.0.1          ACTIVE aes  sha  rsig 5  00:55:54

! This verifies a functioning IKE phase 1 (QM_IDLE,ACTIVE) and the detail
! option reveals that the IKE Phase 1 used RSA signatures for
! authentication, AES for encryption, SHA for hashing and DH group 5, with
! the remaining lifetime from what was initially agreed to by the peers.

! To verify the IPsec (IKE phase 2) tunnel, we can do so with the following
! command:

R2# show crypto ipsec sa

interface: GigabitEthernet1/0
    Crypto map tag: MYMAP, local addr 23.0.0.2

protected vrf: (none)
local  ident (addr/mask/prot/port): (192.168.0.0/255.255.255.0/0/0)
remote ident (addr/mask/prot/port): (172.16.0.0/255.255.0.0/0/0)
current_peer 13.0.0.1 port 500
    PERMIT, flags={origin_is_acl,}
    #pkts encaps: 504, #pkts encrypt: 504, #pkts digest: 504
    #pkts decaps: 504, #pkts decrypt: 504, #pkts verify: 504
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 11, #recv errors 0

    local crypto endpt.: 23.0.0.2, remote crypto endpt.: 13.0.0.1
    path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet1/0
    current outbound spi: 0x3BE5B517(1004909847)
    PFS (Y/N): Y, DH group: group2
! Inbound Security Association (SA/tunnel) from traffic coming from the
! other peer
    inbound esp sas:
        spi: 0x87F1D10A(2280771850)
        transform: esp-aes esp-sha-hmac ,
        in use settings = {Tunnel, }

```

```

conn id: 9, flow_id: SW:9, sibling_flags 80000046, crypto map:
  MYMAP
sa timing: remaining key lifetime (k/sec): (4558182/3257)
IV size: 16 bytes
replay detection support: Y
Status: ACTIVE
! Not using AH, so no inbound AH SA
  inbound ah sas:

! Outbound Security Association (SA/tunnel) for traffic going to the other
! peer
  outbound esp sas:
    spi: 0x3BE5B517(1004909847)
      transform: esp-aes esp-sha-hmac ,
      in use settings ={Tunnel, }
      conn id: 10, flow_id: SW:10, sibling_flags 80000046, crypto map:
        MYMAP
      sa timing: remaining key lifetime (k/sec): (4558182/3257)
      IV size: 16 bytes
      replay detection support: Y
      Status: ACTIVE
! Not using AH, so no outbound AH SA
  outbound ah sas:

! These outputs have been detailed in earlier chapters, but it is relevant
! to know that there are 2 SA (security associations) with the IKE phase 2
! (IPsec). One SA for outbound to the other peer, and another for the
! inbound from that peer. We can also see the encrypt and decrypt count for
! each of the SAs.

! One more command that is useful in seeing a bird's eye view of the
! cryptography is this:
R2# show crypto engine connections active
Crypto Engine Connections

   ID  Type      Algorithm          Encrypt  Decrypt IP-Address
   --  ---      -
   9   IPsec     AES+SHA            0        504    23.0.0.2
  10   IPsec     AES+SHA            504      0      23.0.0.2
 1004  IKE       SHA+AES256        0         0      23.0.0.2

```

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 20-5 lists these key topics.



Table 20-5 *Key Topics*

Key Topic Element	Description	Page Number
Table 20-2	Protocols that may be required for IPsec	499
Table 20-3	IKE Phase 1 policy options	500
Table 20-4	IKE Phase 2 policy options	501
Example 20-3	CLI implementation of crypto policy for R1	509
Example 20-10	Troubleshooting IKE phase 2, the IPsec tunnel	522

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

IKE Phase 1, IKE Phase 2, transform set, DH group, lifetime, authentication, encryption, hashing, DH key exchange, PFS

Command Reference to Check Your Memory

This section includes the most important configuration and EXEC commands covered in this chapter. To see how well you have memorized the commands as a side effect of your other studies, cover the left side of Table 20-6 with a piece of paper, read the descriptions on the right side, and see whether you remember the commands.

Table 20-6 *Command Reference*

Command	Description
<code>crypto map MYMAP 1 ipsec-isakmp</code>	Generate or edit a crypto map named MYMAP, sequence number 1, and request the services of ISAKMP.
<code>crypto isakmp policy 3</code>	Enter IKE Phase 1 configuration mode for policy number 3.
<code>show crypto map</code>	Verify what components are included in the crypto map, including the ACL, the peer address, the transform set, and where the crypto map is applied.
<code>set peer 1.2.3.4</code>	Used inside a crypto map to indicate who the VPN peer should be.
<code>match address 100</code>	Used inside a crypto map to indicate which ACL should be used to indicate interesting outbound traffic for the purpose of encryption.
<code>crypto map MYMAP</code>	Applies a crypto map to an interface.
<code>crypto ipsec transform set MYSET</code>	This is the beginning sequence to creating an IKE Phase 2 transform set named MYSET. This is followed by the HMAC (hashing with authentication) and encryption method (3DES, or AES preferably) that you want to use.



This chapter covers the following subjects:

- Functions and use of SSL for VPNs
- Configuring SSL clientless VPNs on ASA
- Configuring the full SSL AnyConnect VPN on the ASA

Implementing SSL VPNs Using Cisco ASA

Almost everybody is going mobile! This makes it convenient for users to be close to their data, but it also increases the challenge for the network administrator because the users are going to want access to their data and access to corporate data and resources from their mobile devices such as smartphones, laptops, and so on.

In the past, for secure remote access we traditionally installed IPsec *virtual private network (VPN)* client software on the end user's device, configured it, and then allowed them to use it to build a tunnel and after authenticating have access to corporate resources over that tunnel. The challenge with this is that the client software has to be preinstalled and configured on each machine.

Someone came up with the brilliant idea that using *Secure Sockets Layer (SSL)*, which is built in to nearly every browser and can communicate securely with the server on the other side, could be used to implement tunnels between the end users and corporate servers or even corporate networks. In addition, because browsers support SSL by default, we may not even need to install a client in every case, or if we do, we can bootstrap the entire process by initially communicating over SSL for security while we install a client. Long story short, bringing up thousands of users who need remote access can be done extremely quickly by leveraging the SSL that is built in to all of those customer devices.

This chapter explores the details of SSL and *Transport Layer Security (TLS)* and shows how to configure both the clientless flavor of SSL VPN and the full AnyConnect client.

“Do I Know This Already?” Quiz

The “Do I Know This Already?” quiz helps you determine your level of knowledge of this chapter's topics before you begin. Table 21-1 details the major topics discussed in this chapter and their corresponding quiz questions.

Table 21-1 “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Functions and Use of SSL for VPNs	1–4
Configuring SSL Clientless VPNs on ASA	5–6
Configuring the Full SSL AnyConnect VPN on the ASA	7–8

- 1.** Which SSL solution is most appropriate for a remote user who is at a borrowed computer and needs access to a single server at the central office?
 - a.** SSL thin client
 - b.** SSL clientless VPN
 - c.** AnyConnect SSL VPN client
 - d.** IPsec VPN client

- 2.** Which of the following solutions assigns a virtual IP address to the remote user to use for traffic sent over the SSL VPN to the server?
 - a.** SSL thin client
 - b.** SSL clientless VPN
 - c.** AnyConnect SSL VPN client
 - d.** IPsec VPN client

- 3.** What is the immediate cost savings when implementing SSL VPNs?
 - a.** No licensing is required on the server.
 - b.** No licensing is required on the clients.
 - c.** Easy deployment.
 - d.** SSL VPN licenses are significantly less expensive on the server than IPsec licenses.

- 4.** How does an SSL client send the desired shared secret to the server?
 - a.** AES.
 - b.** Encrypts it with the server's public key.
 - c.** Encrypts it with the sender's public key.
 - d.** They use DH to negotiate the shared secret.

- 5.** Which of the following is *not* part of configuring the clientless SSL VPN on the ASA?
 - a.** Launching the wizard
 - b.** Specifying the URL
 - c.** Configuring bookmarks
 - d.** Configuring a pool of IP addresses for the remote users to use

6. What may be the potential problem when enabling SSL VPNs on an interface on the ASA?
 - a. ASDM is now disabled on that interface.
 - b. ASDM must be additionally configured with a custom port.
 - c. ASDM must be used with a different URL.
 - d. ASDM is not affected because it does not connect on port TCP:443.
7. Which of the following steps is configured when setting up AnyConnect on the ASA that would not be configured for clientless SSL VPN? (Choose all that apply.)
 - a. NAT exemption
 - b. Pool of addresses
 - c. Connection profile
 - d. Authentication method
8. Where does the ASA keep the copy of the AnyConnect client that may be deployed down to the client?
 - a. On an HTTPS server only
 - b. On flash
 - c. On an SFTP server only
 - d. On NVRAM

Foundation Topics

Functions and Use of SSL for VPNs

This section covers the alternative to IPsec for implementing secure VPN tunnels.

Is IPsec Out of the Picture?

Key
Topic

SSL *virtual private networks* (VPN) and IPsec VPNs both have their pros and cons. The major benefit of using SSL for VPNs is that it is so darn easy to deploy because most popular browsers support SSL by default. IPsec, however, has a better security footprint than SSL, although they both do a terrific job. If a company has thousands of current clients deployed using IPsec, and it is working, there probably is not a compelling urgency to swap it out. The two technologies can both be configured on the same server, and clients, depending on the situation, can use either service. For example, if a user is at a kiosk or a borrowed computer and only needs access to one specific server, that user can open up a browser using the clientless VPN functionality and after authenticating have specific access to that that one specific server. (This is where the clientless SSL VPN feature excels, when connections to only one or a few servers are needed and the full-tunneled AnyConnect client cannot be installed on the local computer.) When a user is done, she logs out, and the PC that she was using does not have a client installed or any software installed remnants related to it. That same exact user, the next day on her own PC in a different city, may connect to the corporate network using the AnyConnect full-blown SSL VPN client and gain full access to all the resources as a typical remote-access VPN user would. That same user could launch her IPsec VPN client (if it was installed and if the server was supporting IPsec) and build a tunnel to the corporate headquarters and have effectively the same features and feel that the AnyConnect SSL VPN provided. Table 21-2 shows a comparison of IPsec versus SSL.

Key
Topic

Table 21-2 *Comparison of IPsec Versus SSL*

	SSL	IPsec
Applications	Web-based applications, file sharing, email (if not using full client). With the full AnyConnect client, all IP-based applications, similar to IPsec, are available.	All IP-based applications are available to the user. The experience is like being on the local network.
Encryption	Moderate range of key lengths.	Stronger range of longer key lengths.
Authentication	Moderate, one-way or two-way authentication.	Strong, two-way authentication using shared secrets or digital certificates.

	SSL	IPsec
Ease of use	Very high.	Moderate. Can be challenging for nontechnical users, and deployment is more time-consuming.
Overall security	Moderate. Any device can initially connect.	Strong. Only specific devices with specific configurations, such as a VPN client, can connect.

SSL and TLS Protocol Framework

TLS and its predecessor SSL are cryptographic protocols that provide secure transactions on the Internet for things such as email, web browsing, instant messaging, and so on. SSL as a protocol was originally developed by Netscape. Most online transactions that are browser based are secured by SSL or TLS. Both of these protocols provide confidentiality, integrity, and authentication services. These protocols are considered to be operating at the session layer and higher in the OSI logical model. They both can use the *public key infrastructure (PKI)* and digital certificates for authentication of the VPN endpoints and for establishing encryption keys that may be used. Similar to IPsec, these protocols use symmetric algorithms for bulk encryption, and asymmetric algorithms are used for the authentication and for the exchange of keys.

SSL 3.0 served as the basis of TLS 1.0. Both terms are casually, but perhaps incorrectly, referred to interchangeably by the average citizen on the street. Some implementations include the ability to switch to the other protocol if necessary, especially in the case of TLS, which can switch over to SSL if the client connection requires it. Fortunately for us, all of this is done behind the scenes and is transparent to the end user.

Table 21-3 compares these two protocols.

Table 21-3 *Comparison Between SSL and TLS*

SSL	TLS
Developed by Netscape in the 1990s	Standard developed by the Internet <i>Engineering Task Force (IETF)</i>
Starts with a secured channel and continues directly to security negotiations on a dedicated port	Can start with unsecured communications and dynamically switch to a secured channel based on the negotiation with the other side
Widely supported on client-side applications	Supported and implemented more on servers, compared to end-user devices
More weaknesses identified in older SSL versions	Stronger implementation because of the standards process



From this point on, we use the term *SSL* to represent the concepts supported by either *SSL* or *TLS* in this chapter. Cisco *SSL VPNs* are really using *TLS* behind the scenes.

The Play by Play of SSL for VPNs



Security is important, and *SSL VPNs* can provide that security. We also know that *SSL* is used for most online transactions that require security. Before we jump in to the *VPN* portion, it is also important to understand the basics of how *SSL* works. If a customer was opening up a browser and going to connect to a banking server, or some other type of *SSL* device, here is what we would expect:

- The client initiates a connection to the server using the destination IP address of the server and the destination TCP port 443. The source IP address is the IP address of the client, and the source port is some random unused port number on the client machine greater than 1023.
- There is the standard three-way handshake, which is the normal process for TCP in establishing sessions.
- After the client initiates its request for the connection, the server responds, providing its digital certificate, which contains the server's public key.
- The client, upon receiving this digital certificate, has a big decision to make. That decision is whether to believe the credibility of the digital certificate that it just received from the *SSL VPN* server. This is where *PKI* comes into play. If the digital certificate is signed by a *certificate authority (CA)* that the client's browser trusts, and the validity dates for that certificate causes the client to believe that the time has not run out on that certificate, and if the client is checking a *certificate revocation list (CRL)* (and the serial number for the certificate is not on the *CRL*), the client can trust the certificate and extract the public key of the server out of the certificate.
- The client then generates a shared secret that it would like to use for encryption back and forth between itself and the server. The problem is now how to get this shared secret that the client wants to use sent securely over to the server? The answer is the client uses the public key of the server to encrypt the shared secret and send the encrypted secret to the server.
- The server decrypts the sent symmetric key using the server's own private key, and now both devices in the session know and can use the shared secret key.
- The key is then used to encrypt the *SSL* session.

SSL VPN Flavors

There are three different types of *SSL VPN* access methods. They are listed in Table 21-4, along with a description of each.

Table 21-4 Options for SSL VPN Implementation

	Clientless SSL VPN	Clientless SSL VPN with Plug-Ins for Some Port Forwarding	Full AnyConnect SSL VPN Client
Other names	Web VPN.	Thin client.	Full SSL client.
Installed software on client	No client required.	Small applets and/or configuration required.	Full install of AnyConnect required, but may be installed by initially connecting via the clientless option, and securely installing it that way.
User experience	Feels like accessing resources (that are on the corporate network) through a specific browser window or hyperlink.	Some applications can be run locally with output redirected through the VPN. Includes the features of the clientless VPN to the left.	Full access to the corporate network. The local computer acts and feels like it is a full participant on the corporate network.
Servers that can be used	IOS with the correct software, and ASA with the correct licenses.	IOS with the correct software, and ASA with the correct licenses.	IOS with the correct software, and ASA with the correct licenses.
How the user looks from the corporate network	Traffic is proxied (<i>Port Address Translation [PAT]</i>) by the SSL server, as the users packets enter the corporate network.	Traffic is proxied (<i>Port Address Translation [PAT]</i>) by the SSL server as the users packets enter the corporate network.	Clients are assigned their own virtual IP address to use while accessing the corporate network. Traffic is forwarded from the given IP address of the client into the corporate network.
Clients supported	Most SSL-capable computers.	Computers that support SSL and Java.	Most computers that support SSL.



SSL VPN support is provided for Windows, Macintosh, Linux, Apple's iOS, Android, and Windows Mobile with the appropriate licenses on the server, where the licenses are managed.

Configuring SSL Clientless VPNs on ASA

Using the *Adaptive Security Device Manager (ASDM)* on the *Adaptive Security Appliance (ASA)*, we walk through how to configure the clientless SSL VPN.

Using the SSL VPN Wizard

Wizards are prevalent in the ASA GUI management tool called the ASA Security Device Manager. Wizards come in handy when implementing configurations that have lots of little steps, which is the case with VPNs. So, as we've done in most of our chapters that include wizards and GUIs, we take a look at the configuration in ASDM, and then see the configuration from the *command-line interface (CLI)* before we finish the section.

Let's start by taking a look at the high-level tasks that may be used to implement the SSL clientless VPN:

- Find and launch the wizard for the SSL VPN inside the ASDM utility for the ASA
- Configure the SSL VPN URL and interface
- Configure user authentication
- Configure user group policy
- Configure bookmark lists
- Verify that the configuration is what you intended, and verify it works

Within ASDM, to launch the wizard, click the **Wizards** menu bar option, and from the drop-down list select **VPN Wizards**. Then from the VPN Wizards drop-down list, select **Clientless SSL VPN Wizard**. This brings up the welcome page of the SSL VPN Wizard, as shown in Figure 21-1.

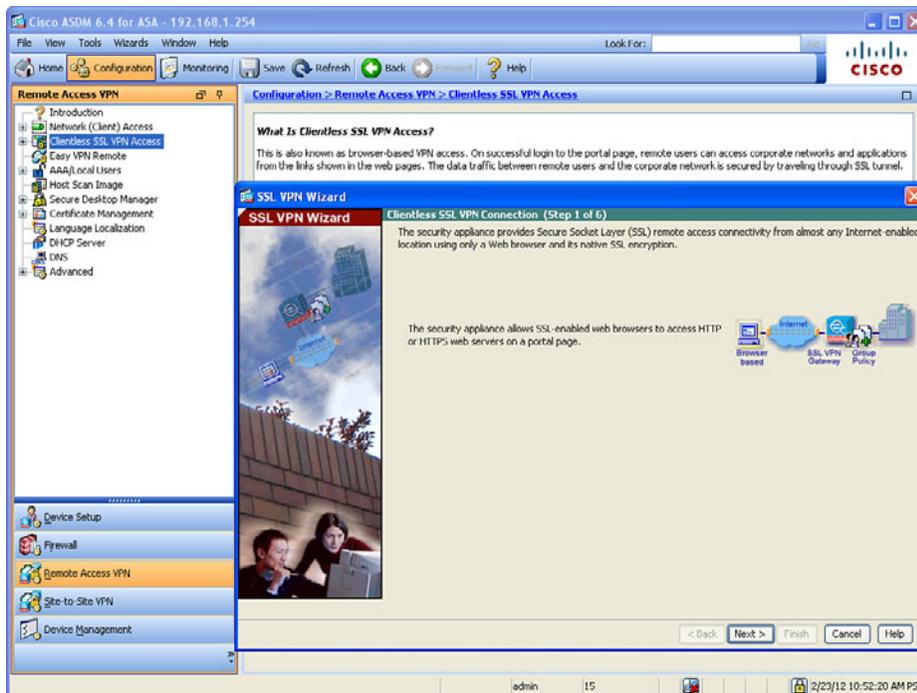


Figure 21-1 SSL VPN Wizard Welcome Page

When you click **Next** to continue, you are presented with a dialog box where you specify a connection profile to be associated with these users who are using clientless SSL VPNs, and the interface these users will be initially connecting to, which is normally the outside interface or a low-security interface on the ASA. Figure 21-2 shows an example of this.

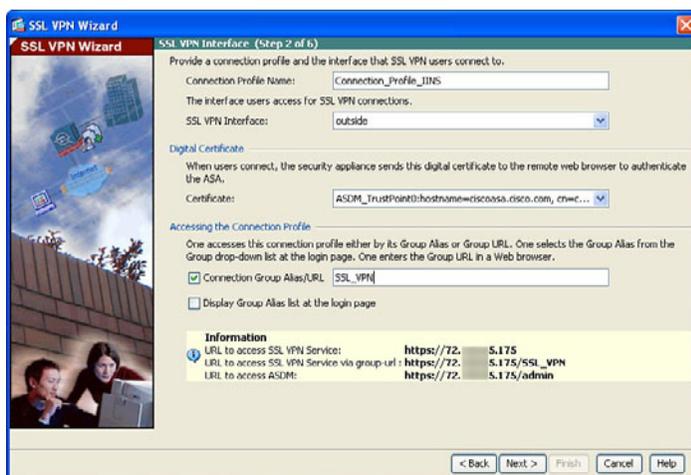


Figure 21-2 Interface Configuration Page

Digital Certificates

By default, a digital certificate is required to be used by the ASA acting as an SSL VPN server. It uses a self-signed digital certificate by default. In most production environments, the company applies for and implements a digital certificate signed by a well-known *public key infrastructure (PKI)* server so that clients connecting will also trust that common *certificate authority (CA)* server and not receive a warning about an unknown certificate. Also on the page shown in Figure 21-2, you indicate the URL that customers could use that would associate them with the correct group. For example, you may have many different SSL VPN groups, with different rights and different users as members of those groups, and handing out the correct URLs to use could make it easier for the initial connection. Another option that is available is to display all the groups from a drop-down list, from which the user could choose which group to connect to. From this page, click **Next** to continue.

If you are using ASDM to manage the ASA on the outside interface, pay special attention to the URL on this page; it should be used in the future for administration purposes for this outside interface, because there is more than one type of SSL connection that will now be coming into the outside interface. Both ASDM and SSL VPN connections are possible.

Authenticating Users

We specify how we are going to go about authenticating the individual users who are trying to connect. We have two general options. The first is that we could use an *authentication, authorization, and accounting (AAA)* server. Very likely, in a Cisco environment, this is an *Access Control Server (ACS)* (or *Identity Services Engine [ISE]*) server. The AAA server could be reached via RADIUS or TACACS+. In the case of authenticating users (end users, specifically SSL VPN users), if the ASA is using the AAA server it will very likely use RADIUS for this purpose. The other option is to use the local database, which just like on the router means the running configuration on the local device (in this case, the ASA's running config). At this point in the wizard, if you want to, you can add additional users to the local database, as shown in Figure 21-3.

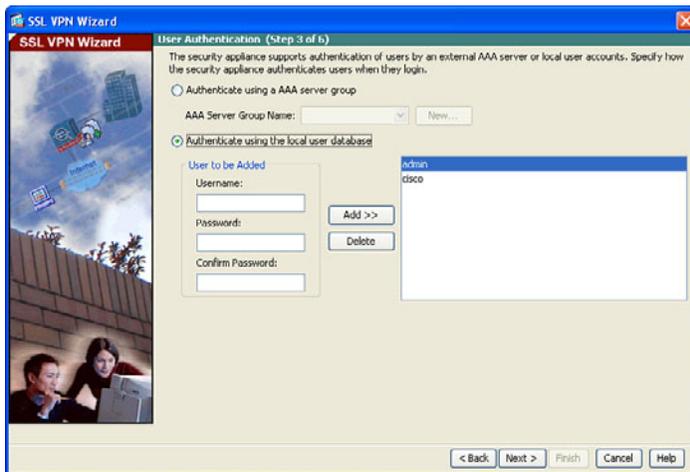


Figure 21-3 Authentication Methods for SSL VPN Users

When you click **Next** to continue, you are asked what group profile you want to use for these users. It is a lot easier to specify attributes and parameters based on a group and then put users into that group instead of individually assigning each user those attributes.

By default, all users belong to a default group, and that default group could be used by the SSL VPN users, as well. If you create a specific group for these users, any parameters assigned to the specific group override the default group policies and apply to those users. The pecking order is that the users inherit properties from their specific group, and a specific group inherits properties from the default group. If a conflict exists between these policies, any attributes assigned to the user win. If a conflict exists between a specific group and a default group, the attributes in the specific group win (or in other words, take precedence). Figure 21-4 shows an example of creating a specific group.

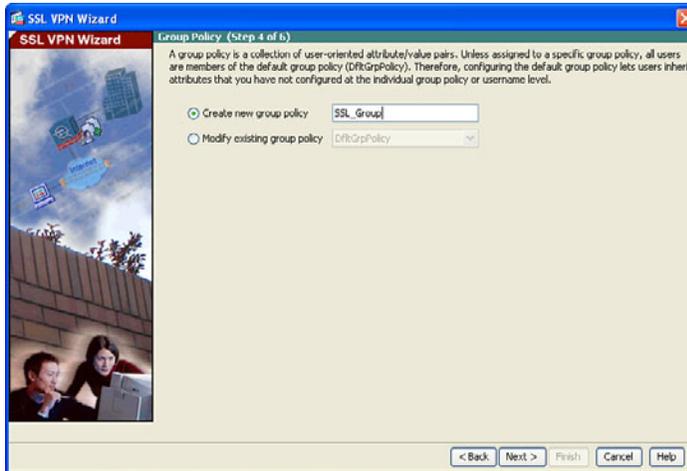


Figure 21-4 Assigning a Specific Group for the SSL VPN Users

When you click **Next**, you are prompted as to whether you want to provide these authenticated SSL VPN users with a convenient list of links/URLs that go to specific services on the corporate network (behind the ASA). Bookmarks can be created and reused for multiple groups. Figure 21-5 shows an example of managing the existing bookmarks and editing them.

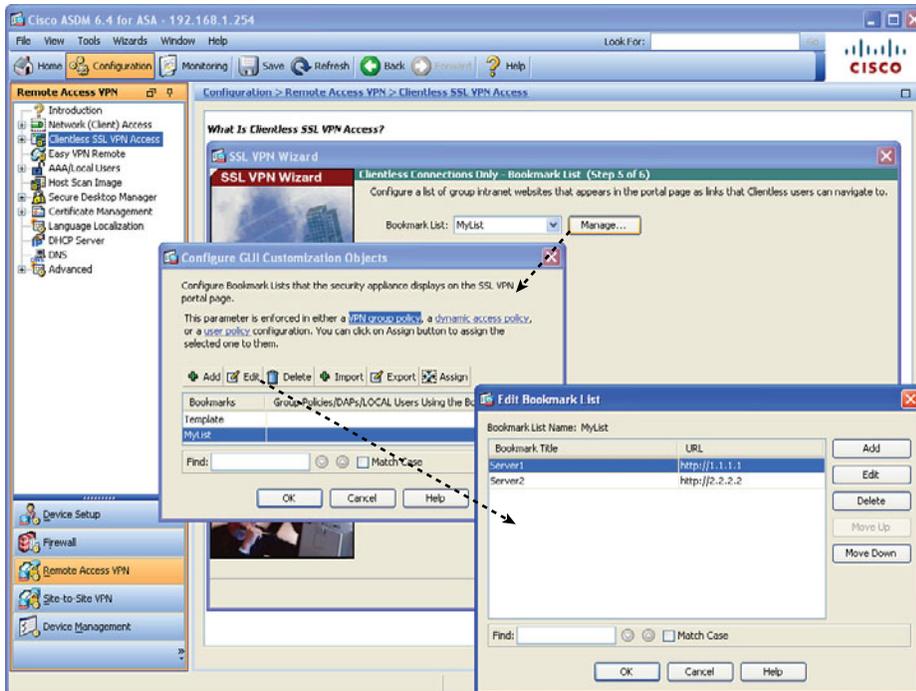


Figure 21-5 Selecting and Editing Bookmarks to Be Provided for Your Users

After you have confirmed using the **Add**, **OK**, and or **Edit** buttons the bookmarks that you want to provide for your users, and click **Next** to continue. A summary of what is about to be deployed is displayed, as shown in Figure 21-6.

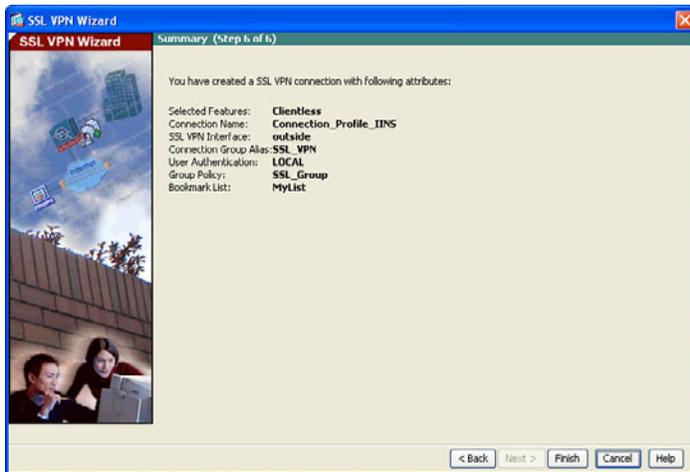


Figure 21-6 Summary of Configuration Elements About to Be Deployed to the ASA Firewall

When you click **Finish**, depending on how you have ASDM configured, it may prompt you for one more confirmation or display the CLI equivalent of what it is about to send.

Speaking of the CLI equivalent, Example 21-1 shows the CLI commands to implement the same policy we just used ASDM for.

Example 21-1 *Implementing a Clientless SSL VPN*

```
! specifies the creation of a local group
asa1(config)# group-policy SSL_Group internal

! Specifies that it's using it own self signed certificate
! and enabling SSL VPN on the outside interface
asa1(config)# ssl trust-point ASDM_TrustPoint0 outside
asa1(config)# webvpn
asa1(config-webvpn)# enable outside

! specifies the attributes for this local group, including the bookmarks
asa1(config-webvpn)# group-policy SSL_Group attributes
asa1(config-group-policy)# vpn-tunnel-protocol ssl-clientless
asa1(config-group-policy)# webvpn
asa1(config-group-webvpn)# url-list value MyList
asa1(config-group-webvpn)# exit
asa1(config-group-policy)# exit
```

```

! specifies a tunnel group for remote access, compared to site to site
asa1(config)# tunnel-group Connection_Profile_IINS type remote-access

! defines the attributes for this connection profile, including the group
! policy to be used
asa1(config)# tunnel-group Connection_Profile_IINS general-attributes
asa1(config-tunnel-general)# default-group-policy SSL_Group

! defines the URL that when connected will trigger what profile to use,
! and that in turn controls what group profile should be applied

asa1(config-tunnel-general)# tunnel-group Connection_Profile_IINS webvpn-
attributes
asa1(config-tunnel-webvpn)# group-alias SSL_VPN enable
asa1(config-tunnel-webvpn)# group-url https://73.143.61.175/SSL_VPN enable
! The asa uses the outside IP address.

```

Logging In

For users to connect, they just point their browser to the HTTPS URL, which includes the IP address (or *Domain Name System [DNS]* resolvable name) and the SSL_VPN portion (based on our configuration). They are prompted for their username and password, and if successful, they are authenticated and provided the bookmarks you configured for that group. Those bookmarks could include not only HTTPS resources but also servers that are reachable via the *Common Internet File System (CIFS)* method. Users, in addition to the links provided by you the administrator, could manually specify resources that they want to reach that are on the corporate networks behind the ASA. All of this can be controlled with the access controls that exist for VPN users on the ASA. Advanced topics, including granular control for the ASA, are in the CCNP Security curriculum, and what you learn here about VPNs and ASAs will assist in preparing you for that.

Figure 21-7 shows what users see when they initially connect and are prompted for their credentials.

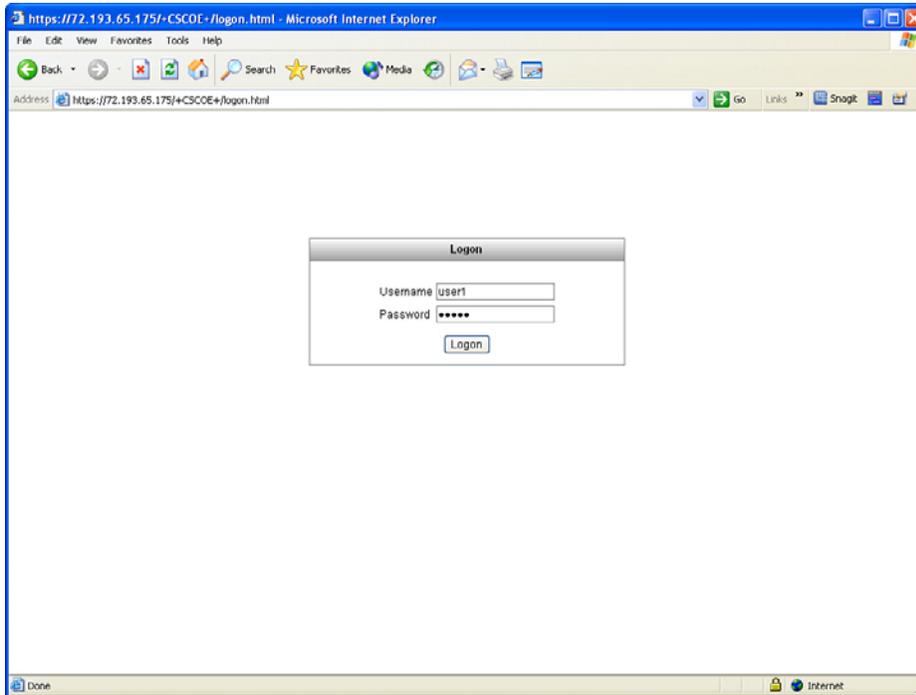


Figure 21-7 *User Interface for Logging In*

The user is presented with the bookmarks configured and the ability to enter an address manually at the top, using the drop-down menu to select the protocol to use, such as HTTP or CIFS (with the latter being used to reach file services being shared on the internal network). Figure 21-8 shows an example of what the user sees (based on your configuration) after authenticating.

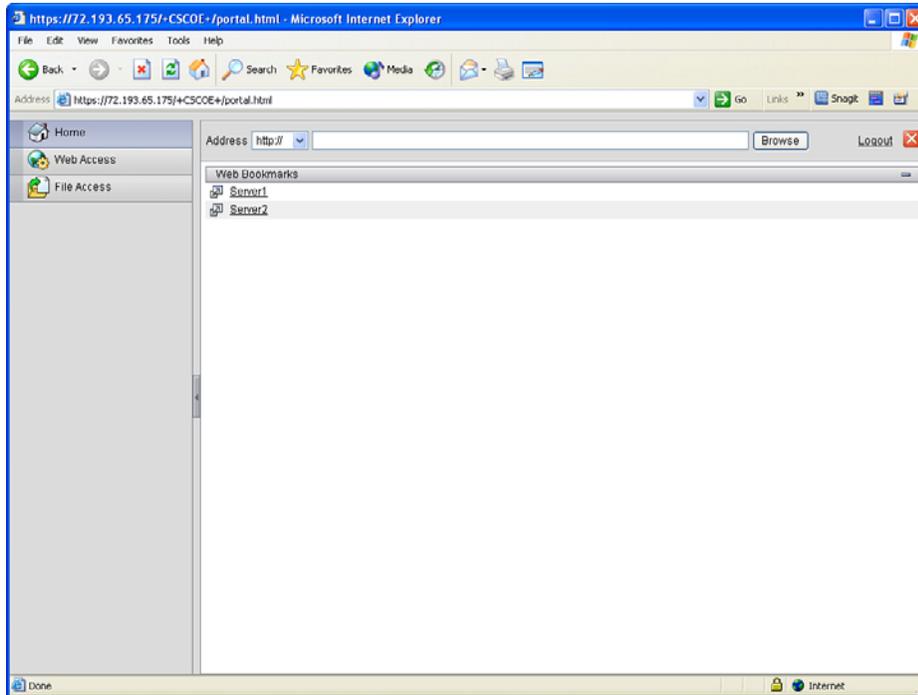


Figure 21-8 *User Interface After Authenticating*

Seeing the VPN Activity from the Server

On the ASA ASDM, to verify the details of your current SSL VPN connections that are in place, navigate to **Monitoring > VPN > VPN Statistics > Sessions**, to see the current VPN sessions. If there are many sessions, you can use the drop-down menu labeled **Filter by:** to narrow down the output to just your VPN SSL remote-access clients. To see the details of a specific session, simply highlight it and click the **Details** button, as shown in Figure 21-9.



The screenshot shows the Cisco ASDM 6.4 interface for an ASA device. The main window displays 'Monitoring > VPN > VPN Statistics > Sessions'. A table shows the following data:

Type	Active	Cumulative	Peak Concurrent	Inactive
Clientless VPN	1	1	3	2
Browser	1	1	3	2

Below this table is a filter section and a table of active sessions:

Username	Group Policy / Connection Profile	Public IP Address / Assigned IP Address	Protocol / Encryption	Login Time / Duration	By / By
user1	SSL_Group / Connection_Profile_IINS	166.147.79.131	Clientless / RC4	11:37:01 PST Thu .. 479664 / 0h:20m:16s	119948

An inset window titled 'Session Details' provides more information for the selected session:

Username	Group Policy / Connection Profile	Public IP Address / Assigned IP Address	Protocol / Encryption	Login Time / Duration	Bytes Tx / Bytes Rx
user1	SSL_Group / Connection_Profile_IINS	166.147.79.131	Clientless / RC4	11:37:01 PST Thu .. 479664 / 0h:21m:57s	119848

The 'Details' tab in the Session Details window shows the following information:

ID	Type	Local Addr. / Subnet_Mask / Protocol / Port	Remote Addr. / Subnet_Mask / Protocol / Port	Encryption	Other	Bytes Tx / Bytes Rx
	Clientless			RC4	Tunnel ID: 14.1 Public IP: 166.147.79.131 Hashing: SHA1 Encapsulation: TLSv1.0 TCP Dst Port: 443 Authentication Mode: userPassword Idle Time Out: 30 Minutes Idle TO Left: 15 Minutes Client Type: Web Browser Client Ver: Mozilla/5.0 (iPhone; CPU iPhone OS 5...	479664 / 119848

Figure 21-9 Monitoring the Details of the SSL VPN Clients in ASDM

As you can see from the output, it gives information about the user who authenticated (which is a user I just created and named user1), that user's source IP address on the Internet, what time the user logged in, how long the user has been logged on, the encryption method that is in use, and the type of client. In this case, the client happens to be an iPhone. (I mentioned that it is an iPhone because earlier I showed an example of connecting to the SSL server from a Windows machine with Internet Explorer, and when I snapped Figure 21-9, it happened to be while a mobile client was connected.) It is also worth noting that the encapsulation shown is TLS, but as I mentioned earlier, the common reference to this type of VPN is still called an *SSL clientless VPN*.

This client, while connected, can access a resource based on a reachable URL that is in a bookmark provided them that goes to a resource on the internal network behind the ASA (or a by using a manually entered URL).

Configuring the Full SSL AnyConnect VPN on the ASA

This section shows you how to implement a full-tunnel VPN using AnyConnect and the SSL functionality.

Types of SSL VPNs

You might recall that we discussed three types of SSL VPN clients. In truth, the thin client, the one that may be using port redirection or have some plug-ins, is really just the SSL clientless VPN with a few extras features. So, really it boils down to two major choices for SSL, and that is SSL using the full-blown installed AnyConnect client software or using the clientless flavor, like the one we just configured in the previous section. Truth be told, if we were going to roll out the AnyConnect client to 1000 users, we would first probably set up the SSL clientless VPN for those users. We would configure the server so that when they initially connect it prompts them to authenticate, and after successful authentication it can either dynamically download and install the client or present the option to install the client on the user's computer. Once the client is installed, we just use the client going forward when they need full access to the corporate network behind the ASA, but still have the option of the clientless connectivity if they need only limited access, instead of using the full AnyConnect VPN client.

Configuring Server to Support the AnyConnect Client

We are going to configure the server to support remote users who are (or will be) using the AnyConnect client. This type of implementation can also be referred to as a *full-tunnel SSL VPN*. To begin the configuration, we click the **Wizards** option on the menu bar, select **VPN Wizards** from the drop-down list, and then select **AnyConnect Wizard** from the next drop-down list.

Once we invoke the VPN AnyConnect Wizard, we are presented with the welcome screen shown in Figure 21-10.

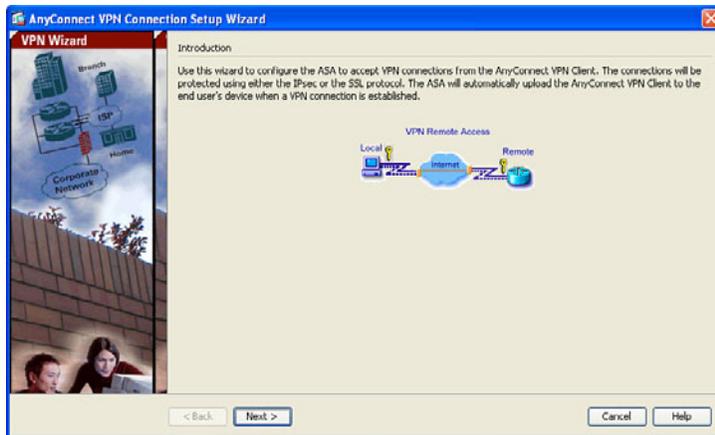


Figure 21-10 The AnyConnect VPN Wizard Welcome Screen

The wizard asks what you want to name the connection profile that will support the AnyConnect users and which interface you are expecting these users to connect on, as shown in Figure 21-11.

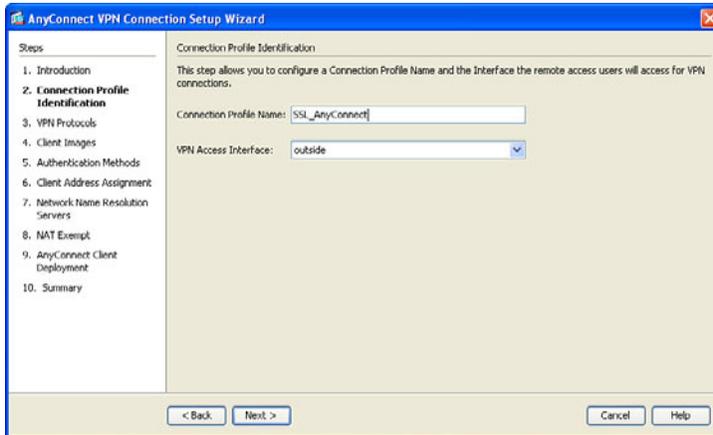


Figure 21-11 *Creating a Connection Profile for the AnyConnect Users*

When you click **Next**, a dialog box asks what protocols you want to support and which digital certificate you want to use on the server. Again, it is best to have a digital certificate that has been issued by a CA that is part of PKI so that when clients connect they do not get a warning message about the certificate. AnyConnect can support both SSL and IPsec. If you do not intend to use IPsec through the AnyConnect client, you can just uncheck the check box and it will not be supported. Figure 21-12 shows an example of this dialog box.

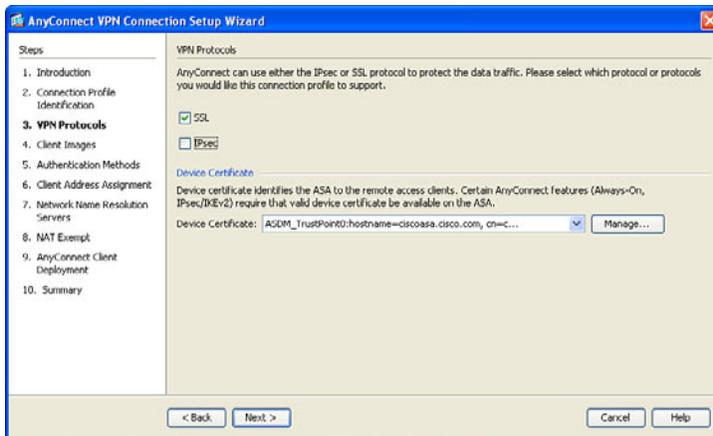


Figure 21-12 *Selecting the Protocols to Support and the Certificate to Use on the Server*

When you click **Next** to continue, you get to identify the AnyConnect software packages that you want this server to be able to deploy to users. A new ASA may come with some of them on flash, but you can also download them from Cisco.com and then copy those files to the flash of the ASA. Figure 21-13 shows the options for adding an AnyConnect software package to the ASA.

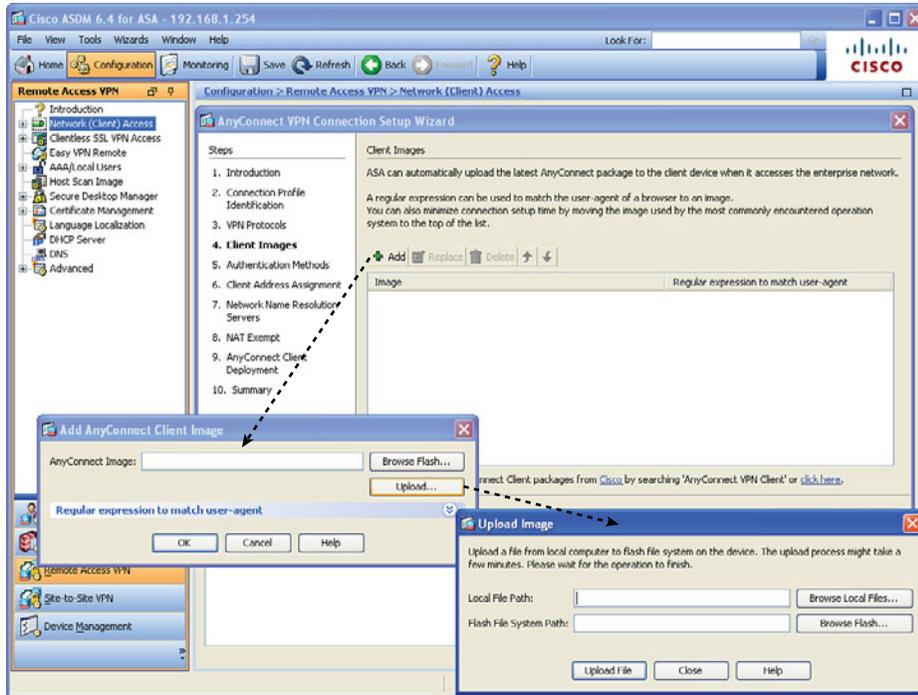


Figure 21-13 *Selecting the AnyConnect Files to Place on the Flash of the ASA*

If the package file is already on the flash, the menus presented in Figure 21-13 allow you to tell the ASA which package (that already exists on the flash) to use.

After specifying the image files to use and clicking the **Next** button, you are presented with a dialog box asking how you want to authenticate users. Similar to the previous configuration, you can use a AAA server such as an ACS server, or you could use the local database on the ASA. Figure 21-14 shows an example of this dialog box.



Figure 21-14 *Specifying the Authentication Method for SSL VPN Users*

The keyword **LOCAL** (in uppercase on the ASA) is the keyword from a AAA perspective on the ASA that represents the local database (the running config).

After adding any additional users to the local database, click **Next**. You are then presented with the questions about what IP address pool you want to use to assign internal addresses to the VPN clients. We didn't have this need before because SSL clientless VPN did not get their own IP addresses but were instead just proxied off the ASA. The pool of addresses that you specify may be a legitimate subnet of one of your networks or a completely made up pool of addresses. Be aware that whatever IP addresses you hand out to clients here should be reachable by the devices inside your corporate network. This could be accomplished with static routing for the network devices, policy-based routing by the network devices, or by creating a loopback interface (on a router) representing a subnet and including it in a dynamic routing protocol. Figure 21-15 shows an example of a pool being created.

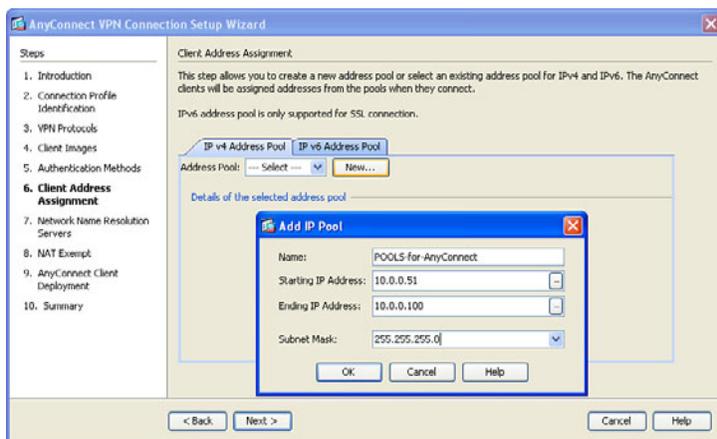


Figure 21-15 Assigning a Pool for the AnyConnect Full-Tunnel Clients

Click **OK** to confirm your pool information. Then click **Next** to continue. At this point, you specify which DNS entries should be handed out to your clients and any NetBIOS name resolution servers (WINS) and a domain name that will primarily play a part in name resolution when the client uses DNSI (see Figure 21-16).

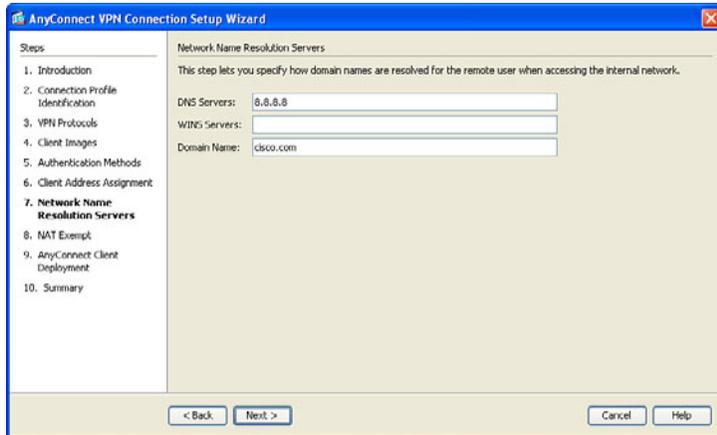


Figure 21-16 DNS, WINS, and Domain Name Configuration to Be Given to AnyConnect Clients

When you click **Next** to continue, you are prompted to confirm that you want to avoid *Network Address Translation (NAT)* between subnets directly connected to the inside interface of the ASA for traffic going to your VPN clients. The reason this is so critical is because your VPN clients, when getting replies back from servers, must not run through the normal NAT process on the ASA. A rule may be in place on the ASA that says all traffic coming from the inside and going to the outside world should be translated into a global address. If the traffic from the inside network that is going back to the VPN clients is translated, the source address is incorrect for what the VPN client is expecting. It also creates a couple of other challenges, and as a result you do not want to NAT from the inside devices back to your VPN devices. If there are additional networks on the inside, or behind the firewall, you could also include those to tell the ASA not to do NAT when packets are going from these internal networks out to this VPN group. To specify the NAT exemption, check the **Exempt VPN Traffic from Network Address Translation** check box and specify the inside network, as shown in Figure 21-17.

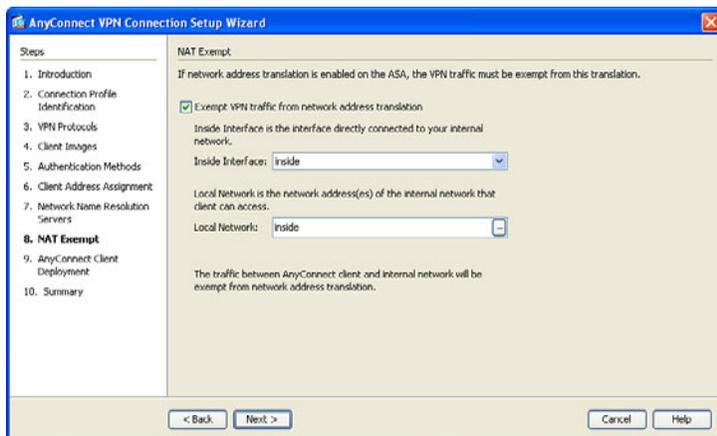


Figure 21-17 Exemptions from NAT for Internal Traffic Going Back to the VPN Clients

When you click **Next** to continue, you are provided with the window indicating that the AnyConnect client can either be preinstalled on a computer or the user can connect using SSL basic connectivity and then install the client from the server, as shown in Figure 21-18.

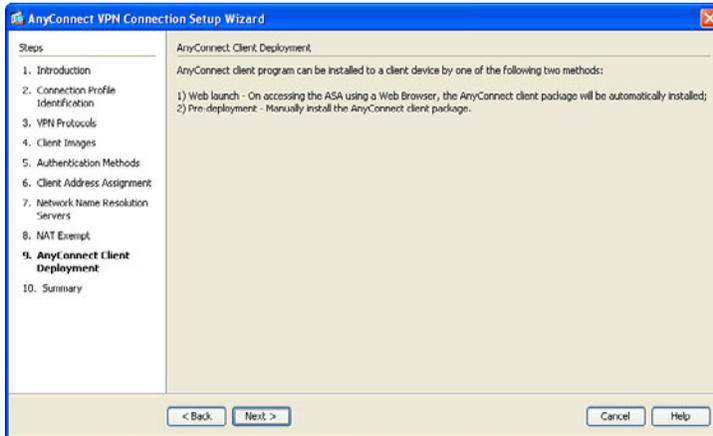


Figure 21-18 Info Screen Indicating How the AnyConnect Client May Be Installed

Clicking **Next** one more time brings up the summary of what is about to be deployed to this ASA, as shown in Figure 21-19.



Figure 21-19 Summary Screen for the AnyConnect Wizard

After clicking **Finish**, depending on how ASDM is configured, you may have one or more dialog boxes to confirm the delivery of the configuration out to the ASA. Example 21-2 shows the CLI equivalent for the same configuration.

Example 21-2 Configuring an SSL AnyConnect Client VPN

```
! For this example, to avoid the wrapping of some of the longer commands
! the firewall prompt has been omitted from the output below
```

```

! For use with the nat exemption, at the end of the config
object network NETWORK_OBJ_10.0.0.0_25
subnet 10.0.0.0 255.255.255.128

! Create the pool for the IP addresses it will be handing out
ip local pool POOLS-for-AnyConnect 10.0.0.51-10.0.0.100 mask 255.255.255.0

! Creates an internal group based on the name below
group-policy GroupPolicy_SSL_AnyConnect internal

! Specifies the attributes of this group, that the protocol for transport
! will be SSL, specifies the DNS and Domain name to hand out.
group-policy GroupPolicy_SSL_AnyConnect attributes
  vpn-tunnel-protocol ssl-client
  dns-server value 8.8.8.8
  wins-server none
  default-domain value cisco.com
exit

! Specifies that SSL is enabled, and which packages from flash are available
! for client images
webvpn
  enable outside
  anyconnect image disk0:/anyconnect-macosx-i386-2.5.2014-k9.pkg 1
  anyconnect image disk0:/anyconnect-win-2.5.2014-k9.pkg 2

! Enables AnyConnect, and provides the group list (connection profile list)
! to users who are logging on, so they can initially select their "group"
  anyconnect enable
  tunnel-group-list enable

! Creates a tunnel group, and specifies the type of tunnel-group it is
tunnel-group SSL_AnyConnect type remote-access

! Specifies what group policy should be used by this tunnel group,
! and what pool of IP addresses should be used for the users
tunnel-group SSL_AnyConnect general-attributes
  default-group-policy GroupPolicy_SSL_AnyConnect
  address-pool POOLS-for-AnyConnect

! Enables this URL (Alias) to be used to access the server
tunnel-group SSL_AnyConnect webvpn-attributes
  group-alias SSL_AnyConnect enable

! provides the exception for NAT (if present) for VPN traffic from the inside
! network if it is going to the address range used by the AnyConnect clients.

```

```
! Note: the following is a single line that is shown as wrapped because
! is longer than the width of this page.
nat (inside,outside) 3 source static inside interface destination static
NETWORK_OBJ_10.0.0.0_25 NETWORK_OBJ_10.0.0.0_25 no-proxy-arp route-lookup
```

Groups, Connection Profiles, and Defaults

A great question that comes up quite a bit is this: Why do we have all these group connection profiles and default groups that seem to be interrelated? The answer is for flexibility. The connection profiles are responsible for the initial connection of the user (the end users only see these as groups, and they do not know all the details behind the scene), and two different connection profiles with two different URLs could use two different authentication methods. After you have authenticated users, you know who they are, and then you can associate group attributes with users' group memberships. In the CCNP Security curriculum, you have a greater opportunity to delve deeper into the workings of these components.

To log in initially to install the AnyConnect client, the customer opens a browser using HTTPS to the IP address of your ASA. The ASA prompts the user for which connection profile to use (displayed as Group to the user), as shown in Figure 21-20.

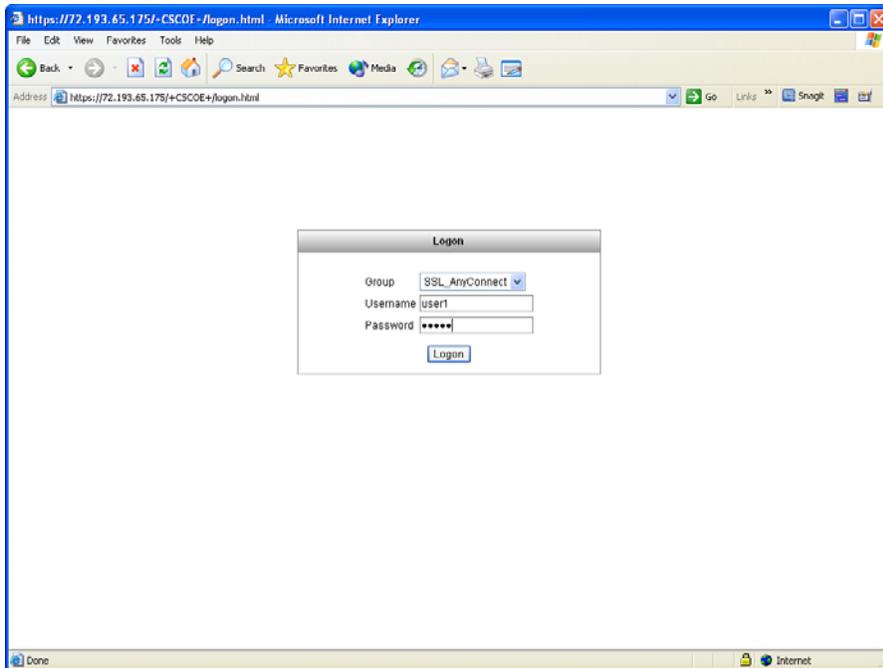


Figure 21-20 Connecting to the ASA, with the Option to select the Connection Profile (Group)

From the drop-down list, users can select the group, supply their credentials, and then click **Logon**.

One Item with Three Different Names

Now here is the interesting part. In the wizard, we created a connection profile called `SSL_AnyConnect`. At the CLI, this is referred to as a tunnel group. From the user's perspective, the drop-down list is called `Group`. It is all referring to the same thing. It is important for you to understand the correlation between all of these, and also to realize that it will be transparent to your users.



After the user authenticates, the ASA attempts to deliver and install the AnyConnect client software on the user's computer. A variety of methods can be used for the install, including ActiveX or Java. There are some minimum requirements at the workstation that is about to receive the installation. For full details of the current minimum requirements, visit Cisco.com for details about the latest version of AnyConnect. Figure 21-21 shows the installation window.

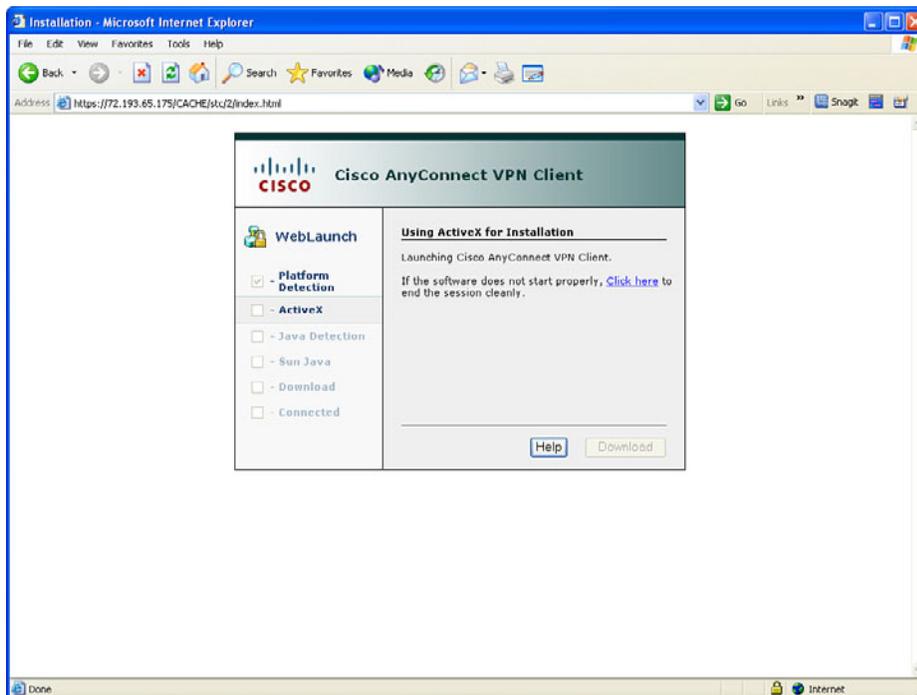


Figure 21-21 *The AnyConnect Software Being Installed on the Client*

Note that the installation looks slightly different on a Macintosh, but the concepts are the same in that the secure tunnel using SSL technology, along with an IP address being assigned to the customer, can allow them full tunnel access to the corporate network over the VPN.

Split Tunneling

One other option that applies to full-tunnel solutions for both AnyConnect and IPsec remote-access clients is the ability to tell that remote device to send traffic over the IPsec or SSL tunnel only if the packets are destined to a specific subnetwork or subnetworks at the headquarter's site. By doing this, the customer at a remote location can directly send out to the Internet to get responses from public servers, and at same time reach remote servers behind the ASA using the VPN. Without split tunneling, all IP traffic leaving the client's machine goes through the tunnel to the ASA (regardless of the destination), and if those resources being sought are not behind the ASA, the ASA also needs to be configured to NAT and redirect those requests out to the Internet. This causes double traffic, and the return path from the Internet server would come back to the ASA and then back through the tunnel to the client. A split tunnel addresses this issue by sending traffic down the VPN only if it is destined for specific networks located at the headquarter site. All other traffic is sent normally, outside the VPN. On the downside, a split tunnel is not considered to be as secure, because an attacker on the Internet may potentially have access to the remote machine, which in turn has access to the internal network through the VPN. By not allowing split tunneling, a corporation could perform additional security features on all the clients traffic, such as IPS and application inspection, before the client's traffic is sent in plain text out to the Internet.

To enable split tunneling on the ASA, navigate to **Configuration > Remote Access VPN > Network(Client) Access > Group Policies**. From there, edit the group policy by going to **Advanced > Split Tunneling**. From there, specify the networks for which you want to tunnel traffic. Anything not identified as traffic that should be tunneled is simply sent by the user through its natural path not inside the tunnel. An example is going to Google or some other web server, with the packets going in plain text to the user's default gateway and then onto the Internet toward that resource. Figure 21-22 shows an example of configuring split tunneling.

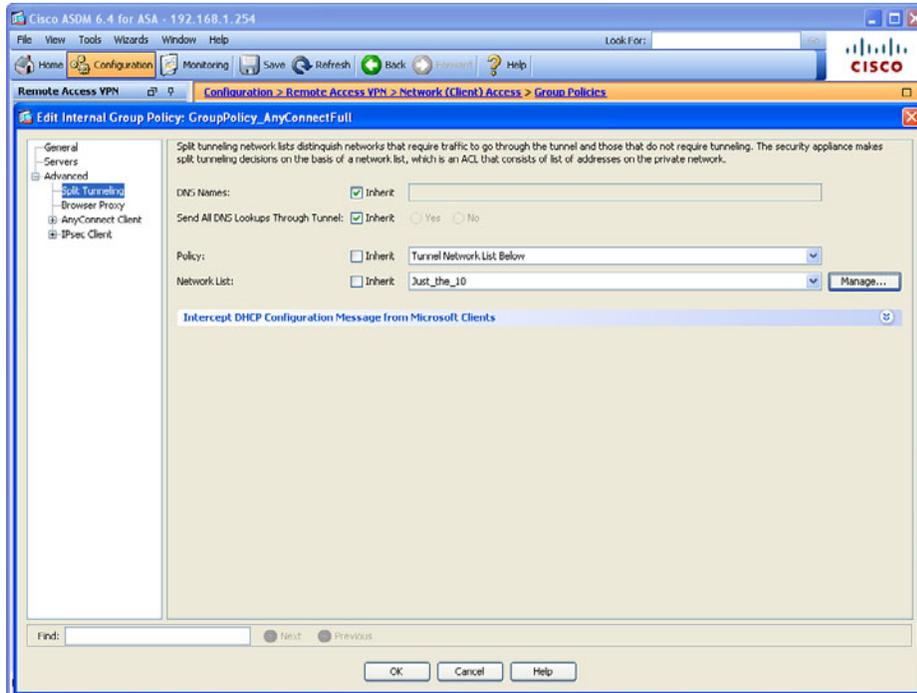


Figure 21-22 *Configuring Split Tunneling*

As clients establish VPN sessions, you are going to want the ability to identify who is connected. You can see this information from the Monitoring section of ASDM, in the same way you looked at the SSL clientless VPN connection information. Navigate to **Monitoring > VPN > VPN Statistics > Sessions**. From there, filter out which types of VPN connections you want to view. You can select an existing connection by clicking it, and then click the **Details** button to see more information.

Exam Preparation Tasks

Review All the Key Topics

Review the most important topics from this chapter, denoted with a Key Topic icon. Table 21-5 lists these key topics.



Table 21-5 *Key Topics*

Key Topic Element	Description	Page Number
Text	Is IPsec out of the picture?	532
Table 21-2	Comparison of IPsec versus SSL	532
Table 21-3	Comparison between SSL and TLS	533
Text	The play by play	534
Table 21-4	SSL VPN implementation options	535
Figure 21-2	Interface configuration page	537
Figure 21-9	Monitoring the details of the SSL VPN clients in ASDM	544
Figure 21-13	Selecting the AnyConnect files to place on the flash of the ASA	547
Text	One item with three different names	553

Complete the Tables and Lists from Memory

Print a copy of Appendix C, “Memory Tables,” (found on the CD) or at least the section for this chapter, and complete the tables and lists from memory. Appendix D, “Memory Tables Answer Key,” also on the CD, includes completed tables and lists so that you can check your work.

Define Key Terms

Define the following key terms from this chapter, and check your answers in the glossary:

SSL, TLS, clientless SSL VPN, AnyConnect full-tunnel VPN, PKI, digital certificate

This page intentionally left blank



Final Preparation

The first 21 chapters of this book cover the technologies, protocols, and considerations required to be prepared to pass the 540-554 IINS (CCNA Security) exam. While these chapters supply the detailed information, most people need more preparation than simply reading the first 21 chapters of this book. This chapter details a set of tools and a study plan to help you complete your preparation for the exams.

This short chapter has two main sections. The first section lists the exam preparation tools useful at this point in the study process. The second section lists a suggested study plan now that you have completed all the earlier chapters in this book.

Note Note that Appendixes C and D exist as appendixes on the CD included in the back of this book.

Tools for Final Preparation

This section lists some information about the available tools and how to access the tools.

Pearson IT Certification Practice Test Engine and Questions on the CD

The CD in the back of the book includes the Pearson IT Certification Practice Test engine—software that displays and grades a set of exam-realistic multiple-choice questions. Using the Pearson IT Certification Practice Test engine, you can either study by going through the questions in Study Mode or take a simulated (timed) CCNA Security exam.

The installation process requires two major steps. The CD in the back of this book has a recent copy of the Pearson IT Certification Practice Test engine. The practice exam—the database of CCNA Security exam questions—is not on the CD.

Note The cardboard CD case in the back of this book includes the CD and a piece of paper. The paper lists the activation key for the practice exam associated with this book. *Do not lose the activation key.* On the opposite side of the paper from the activation code is a unique, one-time use coupon code for the purchase of the *CCNA Security Official Cert Guide, Premium Edition*.

Installing the Software from the CD

The software installation process is routine as compared to other software installation processes. To be complete, the following steps outline the installation process:

- Step 1.** Insert the CD into your PC.
- Step 2.** The software that automatically runs is the Cisco Press software to access and use all CD-based features, including the exam engine and the CD-only appendixes. From the main menu, click the option to **Install the Exam Engine**.
- Step 3.** Respond to windows prompts as with any typical software installation process.

The installation process gives you the option to activate your exam with the activation code supplied on the paper in the CD sleeve. This process requires that you establish a Pearson website login. You need this login to activate the exam, so please do register when prompted. If you already have a Pearson website login, there is no need to register again. Just use your existing login.

Activating and Downloading the Practice Exam

After installing the exam engine, you should then activate the exam associated with this book (if you did not do so during the installation process) as follows:

- Step 1.** Start the Pearson IT Certification Practice Test software from the Windows Start menu or from your desktop shortcut icon.
- Step 2.** To activate and download the exam associated with this book, from the My Products or Tools tab, click the **Activate** button.
- Step 3.** At the next screen, enter the activation key from paper inside the cardboard CD holder in the back of the book. Once entered, click the **Activate** button.
- Step 4.** The activation process downloads the practice exam. Click **Next**, and then click **Finish**.

Once the activation process is completed, the My Products tab should list your new exam. If you do not see the exam, make sure you have selected the **My Products** tab on the menu. At this point, the software and practice exam are ready to use. Just select the exam and click the **Use** button.

To update a particular exam you have already activated and downloaded, on the Tools tab click the **Update Products** button. Updating your exams ensures that you have the latest changes and updates to the exam data.

If you want to check for updates to the Pearson IT Certification Practice Test exam engine software, on the Tools tab click the **Update Application** button. This ensures that you are running the latest version of the software engine.

Activating Other Exams

The exam software installation process, and the registration process, only has to happen once. Then, for each new exam, only a few steps are required. For instance, if you buy another new Cisco Press Official Cert Guide or Pearson IT Certification Cert Guide, extract the activation code from the CD sleeve in the back of that book—you do not even need to CD at this point. From there, all you have to do is start the exam engine (if not still up and running), and perform Steps 2 through 4 from the previous list.

Premium Edition

In addition to the free practice exam provided on the CD-ROM, you can purchase additional exams with expanded functionality directly from Pearson IT Certification. The Premium Edition of this title contains an additional two full practice exams and an eBook (in both PDF and ePub format). In addition, the Premium Edition title has remediation for each question to the specific part of the eBook that relates to that question.

Because you have purchased the print version of this title, you can purchase the Premium Edition at a deep discount. A coupon in the CD sleeve contains a one-time use code and instructions for where you can purchase the Premium Edition.

To view the Premium Edition product page, go to www.pearsonitcertification.com/title/9780132966085.

The Cisco Learning Network

Cisco provides a wide variety of CCNA Security preparation tools at a Cisco Systems website called the Cisco Learning Network. This site includes a large variety of exam preparation tools, including sample questions, forums on each Cisco exam, learning video games, and information about each exam.

To reach the Cisco Learning Network, go to <https://learningnetwork.cisco.com/>, or just search for “Cisco Learning Network.” You must use the login you created at Cisco.com. If you do not have such a login, you can register for free. To register, just go to Cisco.com, click **Register** at the top of the page.

Memory Tables

Like most Official Cert Guides from Cisco Press, this book purposefully organizes information into tables and lists for easier study and review. Rereading these tables can prove very useful before the exam.

Instead of simply reading the tables in the various chapters, this book’s Appendix C, “Memory Tables,” and Appendix D, “Memory Tables Answer Key,” give you another review tool. Appendix C lists partially completed versions of many of the tables from the book. You can open Appendix C (a PDF on the CD that comes with this book) and print it. For review, you can attempt to complete the tables. This exercise can help you focus on the review. It also exercises the memory connectors in your brain, plus it makes you think about the details without being given as much information, which forces a little more contemplation about the facts.

Appendix D, also a PDF located on the CD, lists the completed tables to check yourself. You can also just refer to the tables as printed in the book.

Chapter-Ending Review Tools

Chapters 1 through 21 have several features in the “Exam Preparation Tasks” sections at the end of the chapter. You may have already worked through these in each chapter. It can also be useful to use these tools again as you make your final preparations for the exam.

Videos

Included with the CD are three videos that every CCNA Security candidate will want to watch. The content in these three videos includes the following:

- Bootstrapping a router to work with *Cisco Configuration Professional (CCP)*
- Configuring *Dynamic Host Configuration Protocol (DHCP)* (both client and server)
- Configuring and testing *Network/Port Address Translation (NAT/PAT)* using CCP
- Creating and applying object groups as part of access list filtering using CCP
- Implementing and verifying port security on a switch
- Using CCP to interpret the current implementation of *Zone-Based Firewall*

In addition to these three videos, which are included on the CD that comes with the book, you may also purchase the Video Mentor for the new CCNA Security, which provides a video walkthrough of most of the technologies in the CCNA Security, including *Adaptive Security Appliance (ASA)*, *virtual private networks (VPN)*, and more.

Suggested Plan for Final Review/Study

This section lists a suggested study plan from the point at which you finish reading through Chapter 21 until you take the 540-554 IINS (CCNA Security) exam. Certainly, you can use this plan as is, or just take suggestions from it.

The plan uses four steps:

- Step 1.** **Review key topics and DIKTA questions:** You can use the table that lists the key topics in each chapter or just flip the pages looking for key topics. Also, reviewing the “*Do I Know This Already*” (DIKTA) questions from the beginning of the chapter can be helpful for review.
- Step 2.** **Complete the memory tables:** Open Appendix C on the CD, and print the entire appendix, or print the tables by major part. Then complete the tables.
- Step 3.** **Use the Pearson IT Certification Practice Test engine to practice:** The Pearson IT Certification Practice Test engine on the CD enables you to study using a bank of unique exam-realistic questions available only with this book.
- Step 4.** **Review the three videos on the CD:** Review the three videos on the CD, and if possible, practice the techniques shown in the video, to confirm your understanding of navigating and working with both CCP and the CLI for Zone-Based Firewalls, NAT, and port security.

Using the Exam Engine

The Pearson IT Certification Practice Test engine on the CD includes a database of questions created specifically for this book. The Pearson IT Certification Practice Test engine can be used either in Study mode or Practice Exam mode, as follows:

- **Study mode:** Study mode is most useful when you want to use the questions for learning and practicing. In Study mode, you can select options like randomizing the order of the questions and answers, automatically viewing answers to the questions as you go, testing on specific topics, and many other options.
- **Practice Exam mode:** This mode presents questions in a timed environment, providing you with a more exam-realistic experience. It also restricts your ability to see your score as you progress through the exam and view answers to questions as you are taking the exam. These timed exams not only allow you to study for the actual 540-554 IINS (CCNA Security) exam, they also help you simulate the time pressure that can occur on the actual exam.

When doing your final preparation, you can use Study mode, Practice Exam mode, or both. However, after you have seen each question a couple of times, you will likely start to remember the questions, and the usefulness of the exam database may go down. So, consider the following options when using the exam engine:

- Use this question database for review. Use Study mode to study the questions by chapter, just as with the other final review steps listed in this chapter. Plan on getting another exam (possibly from the Premium Edition) if you want to take additional simulated exams.
- Save the question database, not using it for review during your review of each book part. Save it until the end so that you will not have seen the questions before. Then, use Practice Exam mode to simulate the exam.

Picking the correct mode from the exam engine's user interface is pretty obvious. The following steps show how to move to the screen from which to select Study or Practice Exam mode:

- Step 1.** Click the **My Products** tab if you are not already in that screen.
- Step 2.** Select the exam you want to use from the list of available exams.
- Step 3.** Click the **Use** button.

By taking these actions, the engine should display a window from which you can choose Study Mode or Practice Exam Mode. When in Study mode, you can further choose the book chapters, limiting the questions to those explained in the specified chapters of the book.

Summary

The tools and suggestions listed in this chapter have been designed with one goal in mind: to help you develop the skills required to pass the 640-554 IINS (CCNA Security) exam. This book has been developed from the beginning to not just tell you the facts, but to also help you learn how to apply the facts. No matter what your experience level leading up when you take the exams, it is our hope that the broad range of preparation tools, and even the structure of the book, will help you pass the exam with ease. I hope you do well on the exam.

This page intentionally left blank

Part V: Appendixes

Appendix A: Answers to the “Do I Know This Already?” Quizzes

Appendix B: CCNA Security 640-554 (IINSv2) Exam Updates

Glossary

Appendix C: Memory Tables (on the CD)

Appendix D: Memory Tables Answer Key (on the CD)



Answers to the “Do I Know This Already?” Quizzes

Chapter 1

1. B
2. D
3. B, C, and D
4. B
5. C
6. B
7. D
8. B and D
9. A, B, C, and D
10. B, C, and D

Chapter 2

1. C and D
2. A and B
3. A, B, C, and D
4. B
5. A
6. C
7. A, B, and C
8. C

Chapter 3

1. A
2. B and C
3. B
4. A, B, C, and D
5. A
6. D
7. A
8. A

Chapter 4

1. D
2. B
3. A
4. A and B
5. A and B
6. C and D
7. A, B, and C
8. B
9. A, B, C, and D
10. C

Chapter 5

1. B
2. C
3. A and C
4. B
5. A
6. B and D
7. B

Chapter 6

1. B
2. B
3. B
4. A, B, C, and D
5. C
6. A, B, and D
7. D
8. A and D
9. B and D
10. D

Chapter 7

1. A and D
2. C
3. A and B
4. D
5. A
6. B and C
7. B
8. A
9. C
10. A, B, C, and D

Chapter 8

1. C
2. B
3. D
4. A
5. A
6. A and C
7. C
8. A and D
9. A and E
10. C

Chapter 9

1. B and C
2. A
3. A
4. B
5. A, B, and C
6. A, B, C, and D
7. B and D
8. A, B, and C
9. A and B
10. D

Chapter 10

1. A, B, and D
2. A, B, C, and D
3. B and D
4. C
5. D
6. B

7. A
8. A and C
9. A and C
10. A

Chapter 11

1. B
2. C
3. A
4. B
5. A and B
6. A
7. D
8. C
9. D
10. A

Chapter 12

1. B
2. D
3. B
4. B
5. C
6. A
7. A
8. A

Chapter 13

1. D
2. A
3. A
4. B
5. C
6. A
7. B
8. D

Chapter 14

1. A, B, and C
2. A, B, and C
3. C
4. B and C
5. B and C
6. A, B, and D
7. A
8. C
9. B
10. C

Chapter 15

1. B
2. D
3. C
4. D
5. B
6. A
7. A

8. A and B

9. A

10. A

Chapter 16

1. C

2. A, B, C, and D

3. C

4. C

5. A, B, C, and D

6. A and B

7. A, B, C, and D

8. A and B

Chapter 17

1. B

2. C and D

3. D

4. D

5. B and D

6. B

7. B and D

8. A

9. B

10. A, B, C, and D

Chapter 18

1. B

2. C

3. A

4. D

5. B and D

6. A

7. A, B, C, and D

8. B

9. B

10. C

Chapter 19

1. A

2. C

3. B

4. A, B, C, and D

5. D

6. D

7. A

8. A and B

9. B

10. A

11. A

Chapter 20

1. A, B, and D
2. A
3. C
4. A, B, and C
5. C
6. A
7. C
8. A
9. A
10. A, B, C, and D

Chapter 21

1. B
2. C
3. C
4. B
5. D
6. C
7. A and B
8. B



CCNA Security 640-554 (IINSv2) Exam Updates

Over time, reader feedback allows Cisco Press to gauge which topics give our readers the most problems when taking the exams. To assist readers, authors may create new materials clarifying and expanding on those troublesome exam topics. This additional content about the exam will be posted as a PDF document on this book's companion website, at <http://www.ciscopress.com/title/1587204460>.

This appendix provides you with updated information if Cisco makes minor modifications to the exam on which this book is based. When Cisco releases an entirely new exam, the changes are usually too extensive to provide in a simple update appendix. In those cases, you need to consult the new edition of the book for the updated content. The new exam, 640-554, is a complete rewrite from the previous version. This book is also brand new and addresses the topics you need to know for the exam. Any older study material related to the previous exam should not be relied upon as a complete reference as you prepare for the new certification exam.

This appendix fills the void that occurs with any print book. In particular, this appendix does the following:

- Mentions technical items that might not have been mentioned elsewhere in the book
- Covers new topics if Cisco adds new content to the exam over time
- Provides a way to get up-to-the-minute current information about content for the exam

Always Get the Latest at the Companion Website

You are reading the version of this appendix that was available when your book was printed. However, given that the main purpose of this appendix is to be a living, changing document, it is important that you look for the latest version online at the book's companion website. To do so:

- Step 1.** Browse to <http://www.ciscopress.com/title/1587204460>.
- Step 2.** Select the **Updates** option under the **More Information** box.
- Step 3.** Download the latest "Appendix B" document.

Note The downloaded document has a version number. Comparing the version of this print Appendix B (Version 1.0) with the latest online version of this appendix, you should do the following:

- **Same version:** Ignore the PDF that you downloaded from the companion website.
- **Website has a later version:** Ignore this Appendix B in your book and read only the latest version that you downloaded from the companion website. If there is no appendix posted on the book's website, that simply means there have been no updates to post and Version 1.0 is still the latest version.

Technical Content

The current version of this appendix does not contain any additional technical coverage.

This page intentionally left blank



Glossary

3DES Triple DES, a 168-bit (3 x 56-bit encryption process). DES, or Data Encryption Standard, is a symmetric key encryption algorithm using a block-cipher method.

AAA Authentication, authorization, and accounting.

AAA server The server/host responsible for running RADIUS or TACACS services.

ACS Access Control Server, the RADIUS and TACACS system sold by Cisco.

AES Advanced Encryption Standard, is a symmetric key encryption algorithm using a block-cipher method developed by Joan Daemen and Vincent Rijmen. Available in key sizes of 128-bit, 192-bit, or 256-bit,

amplification attack A method of attack that starts with relatively few packets and amplifies its responses (like Smurf or Fraggle attacks).

AnyConnect Cisco's secure mobility client solution, supporting full-tunnel VPN. Requires a small client on the workstation, but then tunnels all traffic through the SSL or IPsec tunnel, allowing other nonsecure protocols to be transported and secured

ASA Adaptive Security Appliance firewall, such as the ASA 5510 Firewall.

asset Property (tangible or intangible) that has value to a company, something worth protecting.

asymmetrical Meaning both sides are not the same (not symmetrical). An asymmetrical encryption algorithm uses one key to encrypt data and a second (and different) key to decrypt the data.

attack severity rating The amount of damage an attack can cause. It is used as one property of a signature inside of an IPS/IDS system.

audit A detailed review of a network, system or collection of processes. Accounting is another word that has a similar function: collecting information about the network.

authentication method list The list of methods to be used for authentication (RADIUS, TACACS, enable password, Kerberos, vty line, or local database).

authorization method list The list of methods to be used for authorization (RADIUS, TACACS, Kerberos, local database, or to pass if already authenticated). Used to specify what the authenticated user is authorized to do.

C3PL Cisco Common Classification Policy Language. This promotes the concept of using class maps and policy maps to identify and provide specific treatment for traffic.

CA Certificate authority. A system that generates and issues digital certificates. This is usually a device that is trusted by both parties using certificates.

CCP Cisco Configuration Professional. A web-based router administration tool with a GUI.

CCP communities Groups of routers presented together in CCP as a community of devices. A way to organize the devices being managed within CCP.

CCP templates Sections of configurations that can be reapplied to multiple devices in CCP, substituting variables (such as a hostname) that are unique to each router.

CCP user profiles Method to restrict what CCP displays to the administrator, thus limiting what the administrator can see and change through CCP.

Cisco public key The Cisco public key is needed for the IOS-based IPS to verify Cisco's digital signature of the IPS signature package provided by Cisco.

Cisco SIO Security Intelligence Operations. Early warning intelligence, threat and vulnerability analysis, and proven Cisco mitigation solutions to help protect networks.

class map The portion of modular policy framework (MPF) in the ASA, or C3PL on routers and switches, that defines what types of traffic belong to a certain class. Policy maps rely on class maps for the classification of traffic.

class map type inspect This special type of class map defines specific classes and types of traffic to be used for further inspection in Zone-Based Firewalls on IOS routers.

clientless SSL VPN Allows for limited VPN resource access within some protocols that can natively support TLS, such as HTTPS and CIFS shared over HTTPS.

context-aware security Security enforcement that involves the observation of users and roles in addition to things like interface-based controls. An example is an ACS server providing full access to an administrator who is logged in from his local computer, but restricted access when that same user is logged in through a remote device or through a smart phone.

control plane The logic systems in a device that are responsible for the routing or switching decisions (control). Routing protocols are a prime example.

CRL Certificate revocation list. Used in a PKI environment to inform clients about certificates that have been revoked by the CA.

custom privilege level Level 0 (user) and level 15 (enable) are predefined; anything in between (1–14) would be custom privilege level.

data plane The logic systems in a device that are responsible for the actual movement (post-decision) of information. End users sending traffic to their servers is one example of traffic on the data plane.

DH group The Diffie-Hellman exchange, refers to the security algorithm used to exchange keys securely, even over an unsecured network connection. Groups refer to the lengths of the keys involved in the exchange. Group 1 is a 768-bit key exchange, Group 2 is a 1024-bit key exchange, and Group 5 is a 1536-bit key exchange. The purpose of this algorithm is to establish shared symmetrical secret keys on both peers. The symmetric keys are used by symmetric algorithms such as AES. DH itself is an asymmetrical algorithm.

digital signature An encrypted hash that uniquely identifies the sender of a message and authenticates the validity and integrity of the data received. Signing is done with the private key of the sender, and validation of that signature (done by the receiver) is done using the public key of the sender.

disabled signature A signature that is disabled. A signature needs to be both enabled and nonretired to be used by an IPS/IDS system.

eavesdropping Any method of listening in on other conversations, whether voice or data (sniffer).

enabled signature A signature that is enabled. A signature needs to be both enabled and nonretired to be used by an IPS/IDS system.

EUI-64 Extended Unique Identifier-64, an IEEE standard for converting a 48-bit MAC address into a 64-bit host address in IPv6 networks. Used for stateless autoconfiguration.

hash A unidirectional process rather than a reversible algorithm, it takes a variable-sized input and creates a fixed-size output. Common examples include MD5 and SHA.

HMAC Hash Message Authentication Code, used to verify data integrity and authenticity of a message.

identity certificate A digital certificate assigned to a device, host, person, or email in a PKI infrastructure offering a concept of validated identity.

IDS Intrusion detection system. Intrusion detection systems, primarily using signature matching, can alert administrators about an attack on the network, but cannot prevent the initial packet from entering the network.

IKE Phase 1 Internet Key Exchange Phase 1 negotiates the parameters for the IKE Phase 1 tunnel, including hash, DH group, encryption, and lifetime.

IKE phase 2 Internet Key Exchange Phase 2 builds the actual IPsec tunnel. This includes negotiating the transform set for the IPsec SA.

IPS Intrusion prevention system. Intrusion prevention systems, primarily using signature matching, can alert administrators about an attack on the network and can prevent the initial packet from entering the network.

IPsec IPsec is the suite of protocols used to protect the contents of Layer 3 IP packets. ESP is the primary protocol used to encapsulate the Layer 3 packets.

key A password or set of information used to seed other mathematical algorithms.

LDAP Lightweight Directory Access Protocol. This protocol can be used for gathering/managing information from an LDAP-accessible directory/database. An example of its use is having a AAA server use an LDAP request to Active Directory to verify the credentials of a user.

lifetime The amount of time, in seconds or amount of data that has gone by, that a key or security association is considered valid.

man-in-the-middle attack A form of eavesdropping where the attacker inserts himself in the middle of a conversation, masquerading as a wireless access point, router, proxy server, or so on.

management plane The management plane refers to traffic and technologies involved in being able to manage the network and its devices. This could include management sessions with SSH, HTTPS, and so on, and could also include information-gathering tools such as SNMP or NetFlow.

MD5 Message digest algorithm 5. A cryptographic function with a 128-bit hash. Hashing algorithms are unidirectional. The enable secret on an IOS router is stored using an MD5 hash.

method list List of available methods for AAA to use in order (local, RADIUS, TACACS, and so on).

MPF Modular policy framework. A newer technique using the class map and policy map framework to bring about all sorts of manipulations or additional functions to a router. This is what the ASA refers to when using class maps, policy maps, and the service policy commands. On an IOS router, these are referred to as C3PL components.

NA IPv6 neighbor advertisement. Used to communicate information from an IPv6 host to another on the same locally connected network.

NAT Network Address Translation. The process of swapping out an IP address of a packet in transit, with an alternative address. An example of its use would be workstations on the inside of a network using private IP addresses and having those source addresses modified by the NAT router, before packets from those workstations are sent out to the Internet.

NFP Network foundation protection. The concept of breaking down the network into functional components, such as control plane, management plane, and data plane, and then providing protection for each of those components.

NS IPv6 neighbor solicitation. Used by an IPv6 speaker to make a request of one or more local IPv6 devices on the same network.

NTP Network Time Protocol. Used to synchronize time on the network, which is important for log messages and for and IPS/IDS event time stamps to correlate messages across multiple devices.

named access control list (ACL) Configured with ip access-list rather than just access-list commands, and can be defined as either standard or extended, but by name. Named ACLs are easier to edit than numbered ACLs due to the access-list configuration mode provided by the named ACL.

packet filtering Packet filtering is a static check on known information such as source/destination address and source/destination port information.

parser view Commands are available only within particular contexts (views). This is a way to implement role-based management, by creating views and associating specific administrators with those views.

PAT Port Address Translation. This is a subset of NAT, with multiple devices being mapped to a single address. It is also referred to as a many-to-one translation.

PFS Perfect Forward Secrecy. New keys within DH are not based on seeds from previous keys when PFS is enabled, further increasing security. PFS is associated only with IKE Phase 2.

PKCS#10 Public Key Cryptography Standards #10 is a file format used when sending certificate requests to a CA.

PKCS#12 Public Key Cryptography Standards #12 is a file format used to store private keys with accompanying public key certificates.

PKCS#7 Public Key Cryptography Standards #7 is used by a CA to distribute digital certificates.

PKI Public key infrastructure. A scalable architecture that includes software, hardware, people, and procedures to facilitate the management of digital certificates.

policy map The portion of MPF or C3PL that defines what actions occur to traffic belonging to each class.

policy map type inspect The policy map type is associated with Zoned-Based Firewalls on the IOS. The ASA also has specific purpose policy maps for deep packet inspection.

public key The part of a key pair that is shared with other people in a PKI exchange

qualitative A method of risk assessment that uses a scenario model, including expert opinion.

quantitative A method of risk assessment that uses a mathematical model based on data.

RA IPv6 router announcement. Used by a router to inform other IPv6 devices about the local network address to which they are connected.

RADIUS Remote Authentication Dial-In User Service. This is one method for a router or switch to communicate with a AAA server, such as ACS.

regulatory compliance Security policy created because of local/national laws or regulations (SOX, HIPAA, and so on).

retired signature If a particular signature is deemed old and no longer a common threat, it can be retired, which reduces memory used by the IOS IPS.

risk A measurement of the likelihood of a successful attack by measuring the level of threat against a particular vulnerability.

risk rating A quantitative rating of your network before security measures are put in place. The IOS IPS also uses a risk rating to calculate the potential danger of an attack.

root certificate The certificate at the top of a certificate hierarchy in PKI.

RS IPv6 router solicitation request. Used by an IPv6 device to obtain information from an IPv6 router on the local network.

RSA In 1977, Rivest, Shamir, and Adleman developed a public key algorithm still used by most browsers today. This is an asymmetrical algorithm used for authentication.

SCEP Simple Certificate Enrollment Protocol. SCEP was created to facilitate large-scale deployments of PKI, by automating the process of authenticating and enrolling with a CA that supports SCEP. This is a Cisco-sponsored protocol and is supported by some, but not all, other vendors.

secure bootset Part of the Cisco IOS Resilient Configuration feature, preventing the erasure of IOS files from a storage device, such as flash or NVRAM.

SecureX Cisco's security framework to establish and enforce security policies across a distributed network.

security levels Numeric levels used in the ASA to define a relationship of more secure or less secure.

service policy Just like in MQC for quality of service (QoS), this is the device that ties a policy to an interface (QoS) or to a zone pair (ZBF). On an ASA, this is the command element that links a policy to one or more interfaces.

SFR Signature fidelity rating. An IPS measurement of the degree of attack certainty related to that signature correctly indicating the attack on which it is supposed to match.

SHA1 Secure Hash Algorithm 1. A successor to MD5, developed by the National Security Agency (NSA).

signature files Package of signatures that update an IDS/IPS against new attack methods. IOS IPS signature packages are similar to the signatures used on the IPS/IDS appliances.

signature micro-engines Part of IDS/IPS that supports a group of signatures in a common category.

SNMP Simple Network Management Protocol is used for device management, including requesting information and receiving updates from network devices.

spoofed address The source address of an IP packet that has been changed to something not actually assigned or belonging to the location from which it came. Like identity theft for an IP address.

spoofing An attack where the source pretends to be another host or user (MAC, IP, email).

SSH Secure Shell. An encrypted alternative to Telnet. For remote CLI management access to a network device.

SSL Secure Sockets Layer. The original security method for HTTPS, although succeeded by TLS, this term is still widely used and assumed. This is a secure alternative to HTTP.

standard/extended ACL Access control list for packet filtering, set up by number. ACLs 1–100 are standard (source IP only), and 100–199 are extended (source and destination IP as well as port information). ACLs 1300–1999 are also standard ACLs, and 2000–2699 are also extended ACLs.

stateful filtering More than just a simple packet filter check, stateful inspection can determine whether a network flow exists and can look at information up to the application layer. A stateful filtering firewall dynamically allows the return traffic to the user, from the server they were accessing on the other side of the firewall. This is implemented in the ASA firewall and in the Zone-Based Firewall feature on an IOS router.

subordinate CA A certificate authority at a level below the root CA. Large PKI infrastructures use multiple subordinate CAs to offload the work from a single root CA.

SVI Switched virtual interface, or “interface VLAN,” on a switch.

symmetrical Literally meaning both sides are the same, such as with pre-shared keys, where both ends have the exact same information used to encode/decode data. DH produces symmetrical keys. Symmetrical keys would be used by symmetrical algorithms, such as AES, where one key encrypts the data and the same key is used to decrypt the data.

SYN-flood attack An exploit against TCP's three-way handshake opening lots of sessions via the initial SYN packet with no intent of replying to the SYN-ACK and completing the session. This leaves half-open, or embryonic, connections and can overflow a server's session table.

syslog Logging messages can be sent to a syslog server that gathers all incoming messages into text files. Syslog server programs can sort by incoming device IP and by severity/facility levels to make security monitoring simpler.

TACACS+ Terminal Access Controller Access Control System. This is one of the protocols that can be used to communicate between an AAA server and its client (such as between an ACS server and a router).

threat The potential for a vulnerability to be exploited.

TLS Transport Layer Security. Based on SSL, but more widely adopted as an IETF standard in RFC 5246.

transform set A set of secure protocol parameters to be used by IPsec in IKE Phase 2. To properly peer, both sides must agree on a common set.

transparent firewall Firewall implemented at Layer 2 of the OSI model, but still including the ability to analyze traffic at Layer 3 and higher.

TVR Target value rating. User-defined variable in IPS/IDS of the criticality of a particular target if attacked.

unretired In IPS, if a new variant would cause old signatures to become valid again, the signature can be assigned as “unretired” which will make the signature available for use, and will consume memory on the IOS router.

uRPF Unicast Reverse Path Forwarding. Comparing the entry point of a packet's source address against the routing table and making sure the ingress interface matches what the egress interface would be to reach the source of the packet. If the interface does not match, the router assumes the source address is bogus (spoofed) and can drop the packet.

VPN Virtual private network. Used to provide encryption, authentication, data integrity, and antireplay for network traffic.

vulnerability A flaw or weakness in a system's design or implementation that could be exploited.

X.509v3 The ITU standard for PKI. Version 3 typically refers more to the IETF standard (RFC 3280), which includes CRL usage.

zone pairs The traffic flow, for initial traffic, unidirectionally between two zones. An example is a zone pair that begins in the inside zone and goes to the outside zone. Policies can then be applied to initial traffic that is moving the direction of the zone pair (in our case, from inside to outside).

zones The grouping of multiple interfaces under a similar security policy together, such as inside or outside.

This page intentionally left blank

Index

A

AAA (Authentication, Authorization, Accounting), 55

accounting/auditing, 98

ACS

benefits, 140

configuring, 154-164

ISE, compared, 141

platforms supported, 141

router communication protocols, 141-143

routers, configuring, 142-154

troubleshooting, 164-170

AnyConnect SSL VPNs, 547-548

ASA support, 230, 333

authentication, 98

authorization, 98

best practices, 97-98

connectivity, testing, 115

enabling, 87

implementing

CCP, 116-118

command line, 113-115

IPv6, 211

management plane, 55

method lists, creating, 101-102

revoked certificates, 452

routers, 229

access authentication, 100

router-to-ACS, testing, 164-165

self-contained, 99

user authentication

best practices, 95

implementing, 108-113

usernames/passwords/access rules

storage, 98-99

verifying, 146-147

VPN users, 99-100

access

AAA, 97-98

accounting/auditing, 98

authentication, 98

authorization, 98

method lists, creating, 101-102

router access authentication, 100

usernames/passwords/access rules storage, 98-99

VPN users, 99-100

ASA rules, 359-362

CBAC, 229

classes, HTTP service/vty lines, 87

controlling, 55-56

AAA services, 55

encrypted/authenticated SNMP, 56

IP addresses, 56

password policies, 55

- RBAC, 55
 - syslog lockdown*, 56
 - time accuracy*, 56
- firewall rules, 284
- IPv6, controlling, 211
- ports
 - assigning to VLANs*, 178-179
 - negotiations, not allowing*, 190
- reflexive access lists, 229
- remote-access VPNs, 427
- role-based. *See* RBAC
- rules, storing, 98-99
- unauthorized, mitigating, 212
- Access Control Entries (ACE)**, 243
- access control lists. *See* ACLs
- Access Control Server. *See* ACS
- accounting (AAA), 98
- accounts (user)
 - ACS, creating, 160
 - parser views, assigning, 122
- ACE (Access Control Entries)**, 243
- ACLs (Access Control Lists)**, 58
 - applying to interfaces, 249
 - ASA firewalls, 239
 - ASDM, 359-361
 - command line*, 362
 - crypto, 481
 - data plane protection, 58
 - empty, 242
 - extended
 - defined*, 242
 - identifying*, 242
 - standard ACLs, compared*, 243
 - IOS class maps, 239
 - IPv4 packet filtering
 - ACLs, *creating*, 246
 - applying ACLs to interfaces*, 249
 - CLI implementation*, 248
 - lines, adding*, 246
 - object groups*, 251-254
 - ordering*, 247
 - policies*, 244
 - rules, applying*, 251
 - summary page (CCP)*, 245
 - verifying*, 254
 - IPv6 packet filtering, 259-262
 - creating and applying*, 261-262
 - ICMP, 262
 - objectives*, 260
 - topology*, 260
 - lines
 - adding*, 246
 - numbers*, 243
 - logging
 - firewall log details*, 259
 - logs, viewing*, 258
 - summary syslog messages*, 257
 - syslog destinations*, 258

- malicious traffic
 - general vulnerabilities, 241*
 - IP address spoofing, 240*
 - reconnaissance attacks, 240-241*
 - stopping, 239-240*
 - TCP SYN-flood attacks, 240*
- monitoring, 255-257
- NAT/PAT, 239
- object groups, 244
- ordering, 247
- outbound traffic, 242
- packet-filtering, 239
 - ASA firewalls, 230*
 - creating policies, 241*
 - enforcing policies, 241-242*
 - firewalls, 285*
 - routers, 229*
- QoS, 239
- routing protocols, 239
- standard
 - defined, 242*
 - extended ACLs, compared, 243*
 - identifying, 242*
- traffic protection, classifying, 480-481
- VPNs, 239
- wildcard masks, 244
- ACS (Access Control Server), 99**
 - benefits, 140
 - configuring, 154-164
 - adding network drives to device groups, 157-158*
 - authorization policies, 161-163*
 - device groups, creating, 156*
 - group summary, viewing, 159*
 - licensing, 156*
 - login screen, 156*
 - user accounts, creating, 160*
 - user groups, creating, 158*
 - functionality, 99
 - ISE (Identity Service Engine), 141
 - platforms supported, 141
 - routers, configuring, 142-154
 - CCP, 148-154*
 - CLI, 144-147*
 - communication protocols, 141-143*
 - objectives, 142-144*
 - self-contained AAA, 99
 - Solution Engine, 99
 - troubleshooting, 164-170
 - AAA, 164-165*
 - connections, 164*
 - method lists, 166-170*
 - reports, 165-166*
 - user authentication, 14
 - Windows, 99
- actions**
 - IOS-based IPS response, 392
 - policy maps, 296-297
 - risk rating-based, implementing, 381
 - signatures, 405
- activating practice exams, 560**
- Adaptive Security Appliance (ASA)**
 - family models. *See also* ASA firewalls
- Adaptive Security Device Manager.**
 - See* ASDM
- Address Resolution Protocol. *See* ARP**
- addresses**
 - bogus, filtering, 214
 - IP
 - AnyConnect VPN clients, assigning, 548*
 - hosts, assigning, 203*
 - IKE Phase 2, planning, 501*
 - IPv6 versus IPv4, 203*
 - management sessions, controlling, 56*
 - source interfaces, testing, 515-516*

- source NAT, 278-279*
 - spoofing attacks, preventing, 240*
- IPv6
 - all-nodes multicast, 206*
 - all-routers multicast addresses, 206*
 - decimal/binary/hexadecimal conversions, 204*
 - formatting, 202-204*
 - hexadecimal hard way example, 204-205*
 - IPv4, compared, 203*
 - link local, 205-206*
 - loopback, 206*
 - multicast, 207*
 - remote device communication, 205*
 - solicited-node multicast, 207*
 - unicast/anycast, 206-207*
 - zero shortcuts, 205*
- link local, 205-206
- loopback, 206
- MAC
 - flooding, 59*
 - port security, 192-194*
- multicast, 207
 - all-nodes, 206*
 - all-routers, 206*
 - non-local, filtering, 215*
 - solicited-node, 207*
- administrators**
 - access/protocols, protecting, 55-56
 - AAA services, 55*
 - encrypted/authenticated SNMP, 56*
 - IP addresses, controlling, 56*
 - password policies, 55*
 - RBAC, 55*
 - syslog lockdown, 56*
 - time accuracy, 56*
 - control countermeasures, 12
 - alarm summarization (IOS-based IPS), 392**
 - alerts**
 - details, viewing, 414
 - IPS/IDS
 - delivering, 385*
 - types, 380*
 - signatures, viewing, 413-414
 - viewing
 - command line, 415-416*
 - SDEE log file screen, 413-414*
 - all-nodes multicast addresses, 206**
 - all-routers multicast addresses, 206**
 - analysis**
 - cost-benefit, 9-10
 - risks, 25-26
 - current posture assessment, 26-27*
 - qualitative, 26*
 - quantitative, 26*
 - anomaly-based IPS/IDS, 378**
 - antireplay functionality**
 - IPsec support, 468-469
 - VPN component, 430
 - anycast addresses, 206-207**
 - AnyConnect Client, 42**
 - installing, 550
 - software packages, choosing, 546-547
 - SSL_AnyConnect connection profile/
tunnel group/Group correlation, 553
 - AnyConnect SSL VPNs**
 - AnyConnect client
 - installing, 550*
 - software packages, choosing, 546-547*
 - authentication, 547-548
 - clientless SSL VPNs, compared, 545
 - command line configuration, 550-552

- connection profiles, creating, 545
- digital certificates, 546
- DNS, configuring, 548
- domain name configurations, 548
- groups, 552-553
- IP address pool, assigning, 548
- NAT exemptions, 549
- overview, 534
- protocols, choosing, 546
- split tunneling, 554-555
- SSL AnyConnect connection profile/
tunnel group/Group correlation, 553
- summary page, 550
- VPN AnyConnect Wizard, starting, 545
- WINS, configuring, 548
- application inspection firewalls, 276**
- application layer**
 - attacks, 212
 - gateways
 - firewalls, 275*
 - inspections/awareness, 331-332
 - IPv6 versus IPv4, 203
 - visibility, 226
- application polices, 30**
- applying**
 - ACLs
 - rules, 251*
 - interfaces, 249*
 - ASA policies, 339-340
 - IPv6 ACLs as filters, 261-262
 - method lists (AAA), creating, 152
 - object groups, 253-254
 - templates (CCP), 76-77
 - user profiles (CCP), 80
- AR (attack relevancy), 382**
- ARP (Address Resolution Protocol), 85**
 - dynamic, 228
 - gratuitous, 85
 - proxy, 86
- ASA family models, 330-331**
- ASA firewalls, 42**
 - AAA support, 333
 - access rules, 359-362
 - ACLs, 239
 - AnyConnect software packages,
choosing, 546-547
 - application inspection/awareness,
331-332
 - ASDM. *See* ASDM
 - availability, 333
 - botnets, filtering, 333
 - client IP addresses, 355
 - clientless SSL VPNs
 - authentication, 538-540*
 - CLI implementation, 540-541*
 - configuring, 535-544*
 - digital certificates, 537*
 - interfaces, 537*
 - logging in, 541-542*
 - session details, viewing, 543-544*
 - SSL VPN Wizard, 535-544*
 - configuring, 340-345
 - initial boot, 340-343*
 - setup script, running, 343-345*
 - connections
 - console ports, 337*
 - verifying, 345*
 - default traffic flow, 335-336
 - DHCP, 332
 - DMZ, 334
 - group objects, 333
 - interfaces
 - configuring, 347-355*
 - editing, 351*

- final configuration*, 352
- implementing*, 352-355
- maximum allowed*, 350
- summary page*, 350
- VLAN number associations*, 349-350
- Layer 2/Layer 3 implementations, 332
- managing, 336-337
- NAT, 332, 357-359
 - implementing*, 357
 - verifying*, 358
- packet filtering, 331, 337-338
 - implementing*, 338
 - inbound traffic*, 337-338
 - outbound traffic*, 338
- Packet Tracer, 362-367
 - command line*, 364-366
 - input, configuring*, 332-362
 - launching*, 362
 - results*, 363-364
 - Telnet denial, verifying*, 366-367
- PAT, 357-359
 - dynamic, implementing*, 358
 - rules verification*, 358
- policies
 - applying*, 339-340
 - MPF*, 338-339
- routing, 332, 356-357
- security features, 230
 - AAA*, 230
 - ACLs (packet-filtering)*, 230
 - IPS*, 230
 - management protocols*, 230
 - MPF*, 230
 - routing protocol authentication*, 230
 - stateful filtering*, 230
 - URL filtering*, 230
 - VPNs*, 230
- security levels, 333-334
- self-signed certificates, 454
- split tunneling, 554-555
- stateful filtering, 331
- VPN support, 333
- ASDM (Adaptive Security Device Manager)**
 - ACLs, implementing, 359-361
 - certificates, viewing, 455
 - clientless SSL VPNs. *See* clientless SSL VPNs, configuring on ASA
 - dashboard, 345
 - interfaces
 - configuring*, 347-355
 - editing*, 351
 - final configuration*, 352
 - implementing*, 352-355
 - maximum allowed*, 350
 - summary page*, 350
 - VLAN number associations*, 349-350
 - overview, 337
 - Packet Tracer, 362-367
 - input, configuring*, 362
 - launching*, 362
 - results*, 363-364
 - running, 345-347
 - Startup wizard, 346-347
 - usernames/passwords/access rules storage, 345
- ASR (attack severity rating)**, 382, 384-385
- assets**
 - classifying, 10-11
 - criteria*, 11
 - governmental*, 11
 - private sector*, 11
 - roles*, 11

- defined, 9-10
- risk management, 27-28
- asymmetric algorithms, 438**
 - examples, 444
 - key length, 444
 - overview, 433
- atomic micro-engine, 384**
- attack relevancy (AR), 382**
- attack severity rating (ASR), 382**
- attacks, 14-15**
 - application layer, 212
 - back doors, 15
 - botnets, 17
 - CAM overflow, 59
 - covert channels, 17
 - dictionary, 85
 - DoS/DDoS, 17
 - IPv6, 211-212*
 - preventing, 59*
 - TCP SYN-flood attacks, 240*
 - evidence, collecting, 32
 - incident response policies, 32
 - malicious traffic
 - general vulnerabilities, 241*
 - IP address spoofing, 240*
 - reconnaissance, 240-241*
 - sensor responses, 379-380*
 - stopping, 239-240*
 - TCP SYN-flood attacks, 240*
 - man-in-the-middle, 14-16, 212
 - packet amplification, 214
 - password, 17
 - potential attackers, 13-14
 - motivations/interests, understanding, 14*
 - types, 13*
 - privilege escalation, 15
 - reconnaissance, 15
 - routers, 213
 - social engineering, 15
 - spoofing, preventing, 59
 - timing, 381
 - trust exploitation, 17
 - vectors, 14
- auditing, 16**
 - AAA, 98
 - CCP Security Audit, 81
 - AAA, enabling, 87*
 - authentication failure rates, 85*
 - banners, setting, 85*
 - BOOTP service, disabling, 84*
 - CDP, 84*
 - CEF, enabling, 85*
 - enable secret password, setting, 86*
 - Finger service, disabling, 84*
 - firewalls, enabling, 87*
 - fixing identified potential problems, 82-83*
 - gratuitous ARPs, 85*
 - HTTP service/vty lines access class, setting, 87*
 - ICMP redirects, disabling, 86*
 - identification service, disabling, 84*
 - identifying potential problems, 82*
 - interface connections, 82*
 - IP directed broadcasts, disabling, 87*
 - IP mask reply messages, disabling, 87*
 - IP source route, disabling, 85*
 - IP unreachable, disabling, 87*
 - logging, enabling, 85*
 - minimum password lengths, 85*
 - MOP, disabling, 87*
 - One-Step Lockdown, 84*

- options*, 81
 - password encryption, enabling*, 85
 - proxy ARPs, disabling*, 86
 - RPF, enabling*, 87
 - scheduler allocation*, 86
 - scheduler interval, setting*, 86
 - SNMP, disabling*, 86
 - SSH*, 87
 - starting*, 81
 - summary*, 83
 - TCP keepalives, enabling*, 85
 - TCP small servers service, disabling*, 84
 - TCP SYN-Wait times, setting*, 85
 - Telnet settings, enabling*, 86
 - UDP small servers service, disabling*, 84
 - users, configuring*, 86
- authentication**
- AAA, 98
 - ACS method lists
 - routers, configuring*, 144
 - testing*, 166-170
 - AnyConnect SSL VPNs, 547-548
 - CAs (certificate authorities), 450
 - failure rates, setting, 85
 - IKE Phase 1
 - peer*, 471
 - planning*, 499
 - tunnel negotiations*, 470
 - IPsec, 468-469, 499
 - method lists, 149-150
 - NTP, 132
 - routing protocols
 - ASA firewalls*, 230
 - control plane*, 56
 - IPv6*, 211
 - routers*, 229
 - SNMPv3, 130
 - SSL VPN users, 538-540
 - bookmarks provided, editing*, 539
 - groups, assigning*, 538
 - methods*, 538
 - summary page*, 540
 - users
 - best practices*, 95
 - implementing*, 108-113
 - requiring*, 14
 - VPNs, 99-100, 430, 438
- Authentication, Authorization, Accounting. See AAA**
- authNoPriv security level (SNMP), 129**
- authorization**
- AAA, 98
 - ACS method lists
 - routers, configuring*, 144, 150-151
 - testing*, 166-170
 - ACS policies
 - creating*, 161-163
 - customizing*, 163
 - profiles*, 162
 - profiles, 162
 - VPN users, 99-100
- authPriv security level (SNMP), 129**
- auto secure utility, 53**
- availability**
- ASA, 333
 - defined, 9
-
- B**
- back doors, 15**
 - bandwidth management, 59**
 - banners, configuring, 85**

Basic Firewall wizard

CME warning message, 303

DNS, choosing, 305

interfaces

connecting, 302*not belonging warning message*,
303*untrusted warning message*, 303

security levels, choosing, 304

summary page, 305

welcome screen, 302

**binary/decimal/hexadecimal
conversions**, 204**block ciphers**, 432**BOOTP service, disabling**, 84**borderless networks**

changing nature of networks, 40

data centers, 41

defined, 36

end zones, 41

Internet, 41

logical boundaries, 40-41

policy management points, 41

prevention strategies, 42-43

ASA firewalls, 42*IPS (Intrusion Prevention
System)*, 43*IronPort Email Security/Web
Security Appliances*, 43*ISR (Integrated Services Routers)*,
42*ScanSafe*, 43

secured management protocols, 43

SecureX architecture, 42

AnyConnect Client, 42*context awareness*, 42*SIO (Security Intelligence
Operations)*, 42*TrustSec*, 42

single-console management tools, 43

VPN connectivity, 43

botnets, 17, 333**BPDU (bridge protocol data units)**, 184**BPDU guards**

implementing, 190-191

switches, 228

broadcasts (IP)

directed, disabling, 87

IPv6 versus IPv4, 203

buffer logs, receiving, 104**bugs (IPv6)**, 214**business continuity planning**, 33**buttons (CCP toolbar)**, 68**C****C3PL (Cisco Common Classification
Policy Language)**, 296**Call Manager Express (CME)**, 303**CAM (content-addressable memory)**
overflow attacks, 59**CAs (certificate authorities)**, 446

authenticating, 450

certificate information, 446

commercial, 446

cross-certifying, 453

enrolling, 450

hierarchical with subordinate CAs, 453

IPsec site-to-site VPNs, 504-505

overview, 460

single root, 453

subordinate CAs, 460

CBAC (Context-Based Access Control),
229**CCP (Cisco Configuration Professional)**,
63

AAA, implementing, 116-118

ACLs

- applying to interfaces, 249*
- creating, 246*
- lines, adding, 246*
- object groups, creating, 251-252*
- ordering, 247*
- summary page, 245*

alerts, viewing

- IPS Alert Statistics tab, 414*
- IPS Signature Statistics tab, 413*
- SDEE log file screen, 413-414*

benefits, 63

commands, previewing, 83

communities, 70-73

- adding devices, 72-73*
- creating, 71*
- defined, 71*
- discovering devices, 73*
- maximum devices, 71*

configuring routers for ACS servers, 148-154

- ACS servers, adding, 148*
- applying method lists, 152*
- authentication method lists, 149-150*
- authorization method lists, 150-151*
- local users, adding, 153-154*
- method lists, creating, 149*

Express, 65

IKE Phase 1, configuring, 506-507

IKE Phase 2, configuring, 507-508

interface

- content pane, 69*
- left navigation pane, 67*
- menu bar, 66*
- status bar, 69*
- toolbar, 67-68*

IOS-based IPS installation, 394-400

- configuration screen navigation, 394*
- deployment bit on CPU resources, 398*
- interfaces, choosing, 396*
- IPS policy welcome page, 395*
- public key, adding, 397*
- router subscriptions, opening, 395*
- SDEE, enabling, 395*
- signature file locations, defining, 396-397*
- signatures, compiling, 399-400*
- summary page, 398*
- traffic inspection direction, 396*

IPS signatures

- configuration changes output, 403-404*
- editing, 401*
- enabling, 404-405*
- filtering based on signature IDs, 402*
- modification buttons, 401*
- properties, editing, 402*
- viewing, 400*

IPsec, configuring, 475-484

- IKE Phase 1 policy, 477-478*
- local Ethernet information, entering, 477*
- remote peer information, entering, 477*
- Step by Step wizard, 476*
- summary, 481*
- traffic encryption, 480-481*
- transform sets, 479-480*

layout, 65

licensing, 65

logging

*configuring, 126**editing, 126-127*

NAT

*configuring, 319-321**verifying, 322*

NTP configuration, 131

overview, 65

router communication, configuring,

69-70

Security Audit, 81

*AAA, enabling, 87**authentication failure rates, 85**banners, setting, 85**BOOTP service, disabling, 84**CDP, disabling, 84**CEF, enabling, 85**enable secret password, setting, 86**Finger service, disabling, 84**firewalls, enabling, 87**fixing identified potential problems, 82-83**gratuitous ARPs, disabling, 85**HTTP service/vty lines access class, setting, 87**ICMP redirects, disabling, 86**identification service, disabling, 84**identifying potential problems, 82**interface connections, 82**IP directed broadcasts, disabling, 87**IP mask reply messages, disabling, 87**IP source route, disabling, 85**IP unreachable, disabling, 87**logging, enabling, 85**minimum password lengths, 85**MOP, disabling, 87**One-Step Lockdown, 84**options, 81**password encryption, enabling, 85**proxy ARPs, disabling, 86**RPF, enabling, 87**scheduler allocation, setting, 86**scheduler interval, setting, 86**SNMP, disabling, 86**SSH, enabling, 87**starting, 81**summary, 83**TCP keepalives, enabling, 85**TCP small servers service, disabling, 84**TCP SYN-Wait times, setting, 85**Telnet settings, enabling, 86**UDP small servers service, disabling, 84**users, configuring, 86*

SNMP, configuring, 130-131

templates, 74-78

*applying, 76-77**creating, 75-76**merging/overriding options, 77-78*

toolbar properties, 67

user profiles, 78-80

*applying, 80**creating, 79**restrictions, 78**saving, 80**verifying, 80*

ZBFs, configuring, 300-313

*Basic Firewall wizard welcome screen, 302**CME warning message, 303**DNS, choosing, 305**Firewall wizard page, 301-302*

- interface not belonging warning message*, 303
- interfaces, connecting*, 302
- literal CLI commands generated*, 306-313
- security levels, choosing*, 304
- summary page*, 305
- untrusted interfaces warning message*, 303
- verifying*, 314-315
- CD (book)**
 - installing, 560
 - videos, 562
- CDP (Cisco Discovery Protocol)**, 84
- CEF (Cisco Express Forwarding)**, 85
- central servers**, 98-99
- centralized authentication servers.**
 - See ACS*
- centralized monitoring**, 226
- Certificate Revocation Lists (CRLs)**, 452
- certificates**, 460
 - AnyConnect SSL VPNs, 546
 - ASA self-signed, 454
 - authorities, 446
 - authenticating*, 450
 - certificate information*, 446
 - commercial*, 446
 - cross-certifying*, 453
 - enrolling*, 450
 - hierarchical with subordinate CAs*, 453
 - IPsec site-to-site VPNs*, 504-505
 - overview*, 460
 - single root*, 453
 - subordinate CAs*, 460
 - clientless SSL VPNs, 537
 - functions, 452
 - identity, 448
 - installing with SCEP*, 457-459
 - manually installing*, 456
 - requesting*, 450
 - IPsec site-to-site VPNs, 504-505
 - issuers, 449
 - peers public keys, obtaining, 448
 - public keys, 449
 - revocation list location, 449
 - revoked, 451-452
 - root, 446-448
 - authenticating*, 450
 - installing with SCEP*, 457-459
 - issuers*, 447
 - manually installing*, 455-456
 - public keys*, 448
 - serial numbers*, 447
 - subjects*, 447
 - thumbprint*, 448
 - validity dates*, 447
 - SCEP (Simple Certificate Enrollment Protocol), 451
 - serial numbers, 449
 - signatures, 449
 - subjects, 449
 - thumbprint, 449
 - validity dates, 449
 - viewing in ASDM, 455
 - X.500/X.509v3, 449, 460
- challenges**, 4
- Change Default Credentials dialog box**, 72
- ciphers**
 - asymmetrical, 433
 - block, 432
 - defined, 431
 - polyalphabetic, 431

- stream, 432
- substitutions, 431
- symmetrical, 432-433
- transposition, 431
- Cisco Configuration Professional.**
 See CCP
- Cisco Discovery Protocol (CDP), 84**
- Cisco Express Forwarding (CEF), 85**
- Cisco Learning Network, 561**
- Cisco Security Manager (CSM), 43, 231**
- class maps**
 - ASAs, 339
 - defined, 296
- classifying**
 - assets, 10-11
 - criteria, 11*
 - governmental, 11*
 - private sector, 11*
 - roles, 11*
 - countermeasure controls, 12
 - administrative, 12*
 - logical, 12*
 - physical, 12*
 - vulnerabilities, 11-12
- clientless SSL VPNs**
 - AnyConnect SSL VPNs, compared, 545
 - configuring on ASA, 535-544
 - authentication, 538-540*
 - CLI implementation, 540-541*
 - digital certificates, 537*
 - interfaces, 537*
 - SSL VPN Wizard, 535-544*
 - logging in, 541
 - overview, 534
 - session details, viewing, 543-544
- CME (Call Manager Express), 303**
- collecting evidence, 32**
- command line**
 - ACLs
 - implementing, 248*
 - monitoring, 255-257*
 - object groups, creating, 253*
 - alerts, viewing, 415-416
 - AnyConnect SSL VPNs, configuring, 550-552
 - ASA access rules, implementing, 362
 - CA authentication/enrollment, 458-459
 - clientless SSL VPNs implementation, 540-541
 - configuring routers for ACS servers, 144-147
 - AAA, verifying, 146-147*
 - authentication method lists, 144*
 - authorization method lists, 144*
 - overview, 147*
 - crypto policies, configuring, 509-510
 - IOS-based IPS
 - installing, 407-412*
 - signature compilation output, 399-400*
 - IPsec
 - configuring, 482-484*
 - verifying, 486-490*
 - logging, configuring, 126-127
 - NAT
 - configuring, 322*
 - verifying, 323*
 - Packet Tracer, 364-366
 - signature configuration changes
 - output, 403-404
 - SNMP, configuring, 131
 - ZBFs
 - configuration commands, 306-313*
 - verifying, 315-319*

commands

- AAA method lists, 102

- CCP, previewing, 83

- debug

- AAA, 113-115

- ACS method lists, 166-170

- IKE Phase 1, 512

- ping

- IPsec traffic triggers, 512

- router-to-ACS connections, 164

- routers, 499

- signatures, 406

- source interfaces with associated IP addresses, 515-516

- test aaa, 115, 164-165

commercial CAs, 446**Common Classification Policy Language (C3PL), 296****Common Vulnerabilities and Exposures (CVE) database, 12****communication. *See also* traffic**

- ACS server to router protocols, 141-143

- choosing, 142-143

- RADIUS, 142

- TACACS+, 141

- CCP/routers, configuring, 69-70

- encryption

- best practices, 95

- HTTPS, implementing, 125

- SSH, implementing, 122-124

communities, 70-73

- creating, 71

- defined, 71

- devices

- adding, 72-73

- discovering, 73

- maximum, 71

companion website, 573**confidentiality**

- defined, 8

- IPsec, 468, 499

- VPNs, 428, 438

Configure button (CCP toolbar), 68**configuring**

- ACS, 154-164

- adding network drives to device groups, 157-158

- authorization policies, 161-163

- device groups, creating, 156

- group summary, viewing, 159

- licensing, 156

- login screen, 156

- user accounts, creating, 160

- user groups, creating, 158

- ASAs, 340-345

- ASDM, 345-347

- initial boot, 340-345

- setup script, running, 343-345

- authentication failure rates, 85

- banners, 85

- CCP/router communication, 69-70

- clientless SSL VPNs on ASA, 535-544

- authentication, 538-540

- CLI implementation, 540-541

- digital certificates, 537

- interfaces, 537

- SSL VPN Wizard, 535-544

- crypto policies, 508-510

- DNS for AnyConnect clients, 548

- domain names for AnyConnect clients, 548

- enable secret password, 86

- firewall interfaces, 347-355

- final configuration, 352

- maximum allowed, 350

- summary page, 350*
 - VLAN number associations, 349-350*
- HTTP service/vty lines access class, 87
- IKE Phase 1, 506-507
- IKE Phase 2, 507-510
- interfaces, 351
- IPsec, 475-484
 - command line, 482-484*
 - IKE Phase 1 policy, 477-478*
 - local Ethernet information, entering, 477*
 - mirrored VPN for remote peers, 485-486*
 - remote peer information, entering, 477*
 - Step by Step wizard, 476*
 - summary, 481*
 - traffic encryption, 480-481*
 - transform sets, 479-480*
 - VPN tunnel status, 484*
- IPv6 routing, 208-210
- logging, 126
- NAT, 281, 319-322
- NTP, 131-132, 502
 - authentication, 132*
 - CCP, 131*
 - synchronization, verifying, 132*
- Packet Tracer input, 332-362
- password lengths, 85
- Rapid Spanning Tree, 187-188
- routers for ACS servers, 142-154
 - CCP, 148-154*
 - CLI, 144-147*
 - objectives, 142-144*
- scheduler
 - allocation, 86*
 - intervals, 86*

- SNMP
 - CCP, 130-131*
 - command line, 131*
- split tunneling, 554
- syslog support, 125-126
- TCP SYN-Wait times, 85
- thresholds, 392
- trunk ports, 180-181
- users, 86
- WINS for AnyConnect clients, 548
- ZBF components, 298-300
- ZBFs, 300-313
 - Basic Firewall wizard welcome screen, 302*
 - CME warning message, 303*
 - DNS, choosing, 305*
 - Firewall wizard page, 301-302*
 - interface not belonging warning message, 303*
 - interfaces, connecting, 302*
 - literal CLI commands, 306-313*
 - security levels, choosing, 304*
 - summary page, 305*
 - untrusted interfaces warning message, 303*

connections

- AAA, testing, 115
- AnyConnect SSL VPNs profiles, creating, 545
- ASAs
 - console ports, 337*
 - verifying, 345*
- clientless SSL VPNs logins, 541
- interfaces (ZBF zones), 302
- management plane, 94
- router-to-ACS, testing, 164
- VPNs, 43

- console logs, receiving, 104
- content-addressable memory (CAM)
 - attacks, 59
- content pane (CCP), 69
- context awareness, 42
- Context-Based Access Control (CBAC), 229
- control plane
 - CoPP, 56
 - CPPr, 56
 - defined, 52
 - nontransit traffic, 56
 - protection/policing, 229
 - routing protocol authentication, 56
 - security measures, 54
- Control plane policing (CoPP), 56
- Control plane protection (CPPr), 56
- controls
 - administrative, 12
 - logical, 12
 - physical, 12
- CoPP (Control plane policing), 56
- cost-benefit analysis, 9-10
- countermeasures
 - classifying, 12
 - administrative controls*, 12
 - logical controls*, 12
 - physical controls*, 12
 - defined, 9-10
 - designing
 - ACLs*. See *ACLs*
 - application layer visibility*, 226
 - ASA firewalls*, 230
 - centralized monitoring*, 226
 - CSM (Cisco Security Manager)*, 231
 - defense in depth*, 226
 - end-user education*, 226
 - end user risks*, 224-225
 - incident responses*, 226
 - IPS (Intrusion Prevention System)*, 231
 - mitigation policies/techniques*, 226
 - opportunities for attacks*, 224
 - policy procedures*, 226
 - potential risks*, 224
 - routers*, 227-229
 - SIO services*, 231
 - switches*, 227
- DoS attacks, 211
- firewall risks
 - exposure of sensitive systems to untrusted individuals*, 271
 - malicious data*, 271
 - protocol flaw exploitation*, 271
 - unauthorized users*, 271
- IPv6 threats
 - application layer attacks*, 212
 - DoS attacks*, 212
 - man-in-the-middle attacks*, 212
 - router attacks*, 213
 - sniffing/eavesdropping*, 212
 - spoofed packets*, 212
 - unauthorized access*, 212
- Layer 2 threats
 - best practices*, 189
 - BPDUs guards*, 190-191
 - err-disabled ports, restoring*, 191-192
 - negotiations, not allowing*, 190
 - port security*, 192-194
 - root guards*, 192
 - switch ports, locking down*, 189-190
 - tools*, 190

- malicious traffic attacks, 379-380
 - deny attacker inline*, 380
 - deny connection inline*, 380
 - deny packet inline*, 380
 - log attacker packets*, 380
 - log pair packets*, 380
 - log victim packets*, 380
 - produce alert*, 380
 - produce verbose alert*, 380
 - request block connection*, 380
 - request block host*, 380
 - request SNMP trap*, 380
- threats
 - mitigation/containment strategies, designing*, 224
- covert channels, 17
- CPPr (Control plane protection), 56
- creating
 - AAA method lists, 101-102
 - ACS authorization policies, 161-163
 - customizing*, 163
 - profiles*, 162
 - AnyConnect SSL VPNs connection profiles, 545
 - device groups, 156
 - digital signatures, 445
 - firewall rules, 285-286
 - IPv6 ACLs, 261-262
 - key pairs, 457
 - object groups, 251-253
 - packet-filtering ACL policies, 241
 - parser views, 103, 121-122
 - passwords, 97
 - policies (security), 28
 - strategies
 - changing nature of networks*, 40
 - logical boundaries*, 40-41
 - prevention*, 42-43
 - secured management protocols*, 43
 - SecureX architecture*, 42
 - single-console management tools*, 43
 - VPN connectivity*, 43
- subinterfaces, 182-183
- templates (CCP), 75-76
- traffic tags, 180-181
- transform sets, 479
- users
 - accounts*, 160
 - groups*, 158
 - profiles*, 79
- CRLs (Certificate Revocation Lists), 452
- cross-certifying CAs, 453
- crypto ACLs, 481
- crypto policies, configuring, 508-510
- cryptography, 430
 - asymmetric, 438
 - examples*, 444
 - key length*, 444
 - overview*, 433
 - ciphers
 - block*, 432
 - defined*, 431
 - polyalphabetic*, 431
 - substitution*, 431
 - transposition*, 431
 - digital signatures, 438
 - creating*, 445
 - DSA*, 444
 - RSA*, 460
 - VPN functions*, 435-436
 - hashes, 434
 - data integrity, verifying*, 434
 - HMAC (Hashed Message Authentication Code)*, 434

- overview*, 434
 - types*, 434
 - keys, 431
 - Diffie-Hellman key exchange*, 438
 - keyspace*, 436
 - lengths*, 433
 - managing*, 436
 - public key cryptography*, 433
 - PKI. *See* PKI
 - stream ciphers, 432
 - symmetric, 432-433, 438
 - CSM (Cisco Security Manager), 43, 231
 - current posture assessment, 26-27
 - external, 27
 - general, 27
 - internal, 27
 - wireless, 27
 - custodians (asset classification), 11
 - customizing
 - ACS authorization policies, 163
 - firewall interfaces, 351
 - logging settings
 - CCP*, 126
 - command line*, 127
 - privilege levels, 103, 118-120
 - signatures, 401, 406
 - CVE (Common Vulnerabilities and Exposures) database, 12
- ## D
-
- DAI (Dynamic ARP inspection), 59
 - dashboard (ASDM), 345
 - data centers, 41
 - data integrity
 - IPsec, 468, 499
 - verifying, 434
 - VPNs, 428-430, 438
 - data plane
 - ACLs, 58
 - bandwidth management, 59
 - CAM overflow attacks, 59
 - DAI, 59
 - defined, 53
 - DHCP snooping, 59
 - DoS attacks, preventing, 59
 - IOS
 - firewall support*, 58
 - IPS*, 58
 - IP source guard, 59
 - IPS (Intrusion Prevention System), 59
 - MAC address flooding, 59
 - security measures, 54
 - spoofing attacks, preventing, 59
 - TCP intercept, 58
 - transit traffic, 56
 - unicast reverse path forwarding, 58
 - unwanted traffic, blocking, 59
 - databases, public domain threats, 12
 - DDoS (Distributed Denial-of-Service) attacks, 17. *See also* DoS
 - debug commands
 - AAA, 113-115
 - ACS method lists, 166-170
 - IKE Phase 1, 512
 - decimal/binary/hexadecimal conversions, 204
 - default command (AAA method lists), 102
 - defense in depth, 16
 - firewalls, 272-273
 - threats, mitigating, 226
 - delivering IPS/IDS alerts, 385
 - Denial-of-Service attacks. *See* DoS
 - deny attacker inline sensor response, 380

deny connection inline sensor response, 380

deny packet inline sensor response, 380

deployment

firewalls, 283-284

NAT options, 281

designing threat mitigation/containment strategies, 224

ASA firewalls, 230

AAA, 230

ACLs (*packet-filtering*), 230

IPS (*Intrusion Prevention System*), 230

management protocols, 230

MPF, 230

routing protocol authentication, 230

stateful filtering, 230

URL filtering, 230

VPNs, 230

components

application layer visibility, 226

centralized monitoring, 226

defense in depth, 226

end-user education, 226

incident responses, 226

mitigation policies/techniques, 226

policy procedures, 226

CSM (Cisco Security Manager), 231

end user risks, 224-225

IPS (Intrusion Prevention System), 231

opportunities for attacks, 224

potential risks, 224

routers, 227-229

AAA, 229

ACLs (*packet-filtering*), 229

CBAC, 229

control plane protection/policing, 229

IPS, 229

management protocols, 229

reflexive access lists, 229

routing protocol authentication, 229

VPNs, 229

Zone-Based Firewalls, 229

SIO services, 231

switches, 227

BPDUs guards, 228

DHCP snooping, 228

dynamic ARP inspections, 228

IP source guards, 228

modules, 228

port security, 228

root guards, 228

storm control, 228

device groups, creating, 156-158

devices, hardening, 211

DHCP (Dynamic Host Configuration Protocol), 59

ASA, 332, 355

IPv6

IPv4, compared, 203

risks, 213

snooping, 59, 228

dialog boxes

Change Default Credentials, 72

Manage Community, 71

dictionary attacks, 85

Diffie-Hellman key exchange

IKE Phase 1

planning, 499

tunnel negotiations, 470

PKI, 444

running, 471

VPNs, 438

digital certificates. *See* certificates

digital signatures, 438

- creating, 445
- DSA, 444
- RSA, 460
- VPN functions, 435-436

directed broadcasts, disabling, 87

disabling

- BOOTP service, 84
- CDP, 84
- Finger service, 84
- gratuitous ARPs, 85
- ICMP redirects, 86
- identification services, 84
- IP directed broadcasts, disabling, 87
- IP mask reply messages, 87
- IP source routing, 85
- IP unreachable, 87
- MOP, 87
- proxy ARPs, 86
- signatures, 401
- SNMP, 86
- TCP small servers service, 84
- UDP small servers service, 84

disaster recovery planning, 33

Distributed Denial-of-Service attacks (DDoS), 17

DMZ (demilitarized zone), 334

DNS (Domain Name Service)

- AnyConnect clients, configuring, 548
- ZBFs, configuring, 305

domain name configurations (AnyConnect client), 548

DoS (Denial-of-Service) attacks, 17

- IPv6, 211-212
- preventing, 59
- TCP SYN-flood attacks, 240

downloading practice exams, 560

DSA (Digital Signature Algorithm), 444

dual stacks (IPv6 risks), 214

dynamic ARP, 228

Dynamic ARP inspection (DAI), 59

Dynamic Host Configuration Protocol.
See DHCP

dynamic NAT, 281

dynamic PAT, 281, 358

E

eavesdropping, 212

ECC (Elliptic Curve Cryptography), 444

editing. *See* customizing

ElGamal, 444

email policies, 30

enabling

- AAA, 87
- CEF, 85
- firewalls, 87
- logging, 85
- password encryption services, 85
- RPF, 87
- signatures, 401, 404-405
- split tunneling, 554
- SSH, 87
- TCP keepalives, 85
- Telnet settings, 86

encryption

- asymmetric algorithms, 438
 - examples, 444*
 - key length, 444*
 - overview, 433*
- communications
 - best practices, 95*
 - HTTPS, implementing, 125*
 - SSH, implementing, 122-124*

- IKE Phase 1
 - planning*, 499
 - tunnel negotiations*, 470
- IKE Phase 2, *planning*, 501
- IPS/IDS, 381
- management protocols, 103-104
- SNMPv3, 130
- symmetric algorithms, 432-433, 438
- traffic
 - after IPsec*, 473
 - before IPsec*, 472-473
 - identifying*, 475
 - IKE Phase 2, planning*, 501
 - IPsec*, 472, 480-481
- end zones (borderless), 41
- enforcement
 - guidelines, 31
 - packet-filtering ACLs, 241-242
 - policies. *See* policies
 - procedures, 31
 - standards, 31
- err-disabled ports, *restoring*, 191-192
- evasion methods (IPS/IDS), 381
 - encryption/tunneling, 381
 - protocol level misinterpretation, 381
 - resource exhaustion, 381
 - timing attacks, 381
 - traffic
 - fragmentation*, 381
 - substitution/insertion*, 381
- evidence, *collecting*, 32
- exam updates, 573-574
 - companion website, 573
 - print version versus online version, 574
- extended ACLs
 - defined, 242
 - identifying, 242

- object groups
 - applying*, 253-254
 - creating*, 251-253
- rules, *applying*, 251
- standard ACLs, *compared*, 243
- verifying*, 254
- external risk assessment, 27

F

- false negatives (IPS/IDS), 377
- false positives (IPS/IDS), 377
- FE80 (link local addresses), 206
- features
 - ASA firewalls, 230
 - AAA, 230, 333
 - ACLs (*packet-filtering*), 230
 - application inspection/awareness*, 331-332
 - availability*, 333
 - botnets, filtering*, 333
 - DHCP, 332
 - IPS (Intrusion Prevention System), 230
 - Layer 2/Layer 3 implementations, 332
 - management protocols*, 230
 - MPF, 230
 - NAT support, 332
 - object groups, 333
 - packet filtering*, 331
 - routing*, 230, 332
 - stateful filtering*, 230, 331
 - URL filtering, 230
 - VPNs, 230, 333
 - IOS router security, 228
 - routers, 227-229
 - AAA, 229
 - ACLs (*packet-filtering*), 229

- CBAC, 229
- control plane protection/policing*, 229
- IPS, 229
 - management protocols*, 229
 - reflexive access lists*, 229
 - routing protocol authentication*, 229
 - VPNs, 229
 - Zone-Based Firewalls*, 229
- SSL, 534
- switches, 227
 - BPDUs guards*, 228
 - DHCP snooping*, 228
 - dynamic ARP inspections*, 228
 - IP source guards*, 228
 - modules*, 228
 - port security*, 228
 - root guards*, 228
 - storm control*, 228
- ZBFs, 294-295
- FF02::1 (multicast address), 206
- files
 - IOS, protecting, 106
 - log, viewing, 258
 - primary bootset, storing, 132
 - signatures
 - configuration files, locating*, 397
 - locations, defining*, 396
 - obtaining*, 393-394
 - public key, adding*, 397
 - system, protecting, 96
- filtering
 - ASA packet, 331, 337-338
 - implementing*, 338
 - inbound traffic*, 337-338
 - outbound traffic*, 338
 - bogus addresses, 214
 - botnets, 333
 - ICMP unused traffic, 215
 - IPv4 packet
 - ACLs, creating*, 246
 - applying ACLs to interfaces*, 249
 - CLI implementation*, 248
 - lines, adding*, 246
 - object groups*, 251-254
 - ordering*, 247
 - policies*, 244
 - rules, applying*, 251
 - summary page (CCP)*, 245
 - verifying*, 254
 - IPv6 packet, 259-262
 - creating and applying*, 261-262
 - ICMP*, 262
 - objectives*, 260
 - topology*, 260
 - non-local multicast addresses, 215
 - packet-filtering ACLs, 239
 - ASA firewalls*, 230
 - creating policies*, 241
 - enforcing policies*, 241-242
 - firewalls*, 285
 - routers*, 229
 - SDEE log file screen, 414
 - signatures, based on signature IDs, 402
 - stateful, 276-277
 - ASA, 331
 - ASA firewalls*, 230
 - static packets, 274-275
 - traffic, 212
 - URLs, 230
- final review/study plan, 562
- Finger service, disabling, 84

firewalls

- access rules, 284
- application inspection, 276
- application layer gateways, 275
- ASA, 42
 - AAA support*, 333
 - access rules*, 359-362
 - ACLs*, 239
 - application inspection/awareness*, 331-332
 - ASDM*, 345-347
 - availability*, 333
 - botnets, filtering*, 333
 - client IP addresses*, 355
 - configuring*, 340-345
 - connectivity, testing*, 345
 - console ports, connecting*, 337
 - default traffic flow*, 335-336
 - DHCP*, 332
 - initial boot*, 340-345
 - interfaces, configuring*, 347-355
 - Layer 2/Layer 3 implementations*, 332
 - managing*, 336-337
 - models*, 330-331
 - MPF*, 338-339
 - NAT*, 332, 357-359
 - object groups*, 333
 - packet filtering*, 331, 337-338
 - Packet Tracer*, 362-367
 - PAT*, 357-359
 - policies, applying*, 339-340
 - routing*, 332, 356-357
 - security features*, 230
 - security levels*, 333-334
 - self-signed certificates*, 454
 - setup script, running*, 343-345
 - stateful filtering*, 331
 - VPN support*, 333
- capacities, 273
- defense in depth, 272-273
- designing, 283-284
- DMZ, 334
- enabling, 87
- implementing, 274
- IOS support, 58
- limitations, 272
- logs viewing, 259
- NAT, 278-281
 - deployment options*, 281
 - inside/outside/local/global terminology*, 279
 - PAT*, 279-281
 - source IP addresses*, 278-279
- objectives, 270-271
- packet-filtering ACLs, 285
- protecting against
 - exposure of sensitive systems to untrusted individuals*, 271
 - malicious data*, 271
 - protocol flaw exploitation*, 271
 - unauthorized users*, 271
- rules
 - access*, 284
 - guidelines*, 285-286
 - implementation consistency*, 286-287
- stateful packet filtering, 276-277
- static packet filtering, 274-275
- technologies, 270, 283
- transparent, 276-278
- ZBFs, 229
 - administrator created zones*, 295
 - class maps*, 296
 - components, configuring*, 298-300

- configuring*, 300-313
- monitoring*, 314-315
- NAT, configuring*, 319-322
- NAT, verifying*, 322-323
- overview*, 294
- policy maps*, 296-297
- self zones*, 297-298
- service policies*, 297
- traffic interaction between zones*, 297-298
- verifying with CCP*, 314-315
- verifying with command line*, 315-319
- zone pairs*, 295

formatting IPv6 addresses, 202-204

fragmenting traffic, 381

frameworks

- MPF, 230, 338-339

- NFP (network foundation protection), 52-53

- control*, 52

- data*, 53

- interdependence*, 53

- management*, 52

full-tunnel SSL VPN. *See* AnyConnect SSL VPNs

G

gateways (application layer)

- firewalls, 275

general security posture assessment, 27

GET messages, 129

global correlation, 382, 386

global NAT, 279

governmental asset classifications, 11

gratuitous ARPs, disabling, 85

groups

- AnyConnect SSL VPNs, 552-553

device

- creating*, 156

- network devices, adding*, 157-158

object

- applying*, 253-254

- creating*, 251-253

- overview*, 244

signatures, 384

SSL VPN users, assigning, 538

user, creating, 158

guards

BPDU

- implementing*, 190-191

- switches*, 228

IP source, 228

root, 192, 228

guidelines, 16

- auditing, 16

- defense in depth, 16

- policies, 29

- rule of least privilege, 16

- separation of duties, 16

H

Hashed Message Authentication Code (HMAC), 434

hashes, 434

- data integrity, verifying, 428-430, 434

- HMAC (Hashed Message Authentication Code), 434

IKE Phase 1

- planning*, 499

- tunnel negotiations*, 470

IKE Phase 2, planning, 501

overview, 434

types, 434

headers (IPv6)

- IPv6 versus IPv4, 203
- risks, 214
- routing header 0s, dropping, 215

Help icon (CCP toolbar), 68

hexadecimal/binary/decimal conversions, 204

hierarchical PKI topology, 453

HIPAA (Health Insurance Portability and Accountability Act), 28

HMAC (Hashed Message Authentication Code), 434

Home button (CCP toolbar), 68

HTTP (Hypertext Transfer Protocol), 87

HTTPS (Hypertext Transfer Protocol Secure), 125

I

ICMP (Internet Control Message Protocol), 86

IPv6

- packet filtering*, 262
- risks*, 214

mask reply messages, disabling, 87

redirects, disabling, 86

unreachables, disabling, 87

unused traffic, filtering, 215

identity certificates, 448

installing with SCEP, 457-459

- CA server details*, 457

- command line*, 458-459

- details, viewing*, 459

- enrollment modes*, 458

- key pairs, creating*, 457

- success message*, 459

manually installing, 456

requesting, 450

Identity Service Engine. *See* ISE

IDS (Intrusion Detection System), 374

advantages/disadvantages, 379

alerts, delivering, 385

best practices, 386

countermeasure actions, 379-380

- deny attacker inline*, 380

- deny connection inline*, 380

- deny packet inline*, 380

- log attacker packets*, 380

- log pair packets*, 380

- log victim packets*, 380

- produce alert*, 380

- produce verbose alert*, 380

- request block connection*, 380

- request block host*, 380

- request SNMP trap*, 380

evasion methods, 381

- encryption/tunneling*, 381

- protocol level misinterpretation*, 381

- resource exhaustion*, 381

- timing attacks*, 381

- traffic fragmentation*, 381

- traffic substitution/insertion*, 381

false positives/negatives, 377

information accuracy, 376

intelligence

- collecting*, 385-386

- global correlation*, 386

IPS, compared, 374-376

malicious traffic, identifying, 377

- anomaly-based*, 378

- method advantages/disadvantages*, 379

- policies*, 378

- reputation-based*, 378-379

- signatures*, 377-378

- risks
 - actions, implementing, 381*
 - ratings, 379-382*
- sensors
 - defined, 374*
 - platforms, 375-376*
- signatures
 - ASR (attack severity rating), 384-385*
 - groups, 384*
 - micro-engines, 384*
 - SFR (signature fidelity rating), 385*
- true positives/negatives, 377
- IKE (Internet Key Exchange)**
 - Phase 1
 - authentication, 471*
 - configuring, 506-507*
 - Diffie-Hellman key exchange, running, 471*
 - planning, 499-500*
 - policy, 477-478*
 - protocols, choosing, 475*
 - summary, 481*
 - troubleshooting, 512*
 - tunnels, negotiating, 469-470*
 - Phase 2, 471-472
 - configuring, 507-510*
 - planning, 501-502*
 - protocols, choosing, 475*
 - summary, 481*
 - transform sets, 479-480*
 - troubleshooting, 522-525*
- traffic encryption
 - before IPsec, 472-473*
 - after IPsec, 473*
- user packets, encrypting, 472
- implementing**
 - AAA
 - CCP, 116-118*
 - command line, 113-115*
 - debug command, 115*
 - actions based on risk ratings, 381
 - ASA packet filtering, 338
 - BPDU guards, 190-191
 - dynamic PAT, 358
 - firewalls, 274
 - application inspection, 276*
 - application layer gateways, 275*
 - best practices, 283-284*
 - interfaces, 352-355*
 - NAT, 278-281*
 - rules, 286-287*
 - stateful packet filtering, 276-277*
 - static packet filtering, 274-275*
 - technologies, 283*
 - transparent, 276-278*
 - HTTPS, 125
 - IPv4 packet filtering
 - ACLs, creating, 246*
 - applying ACLs to interfaces, 249*
 - CLI implementation, 248*
 - lines, adding, 246*
 - object groups, 251-254*
 - ordering, 247*
 - policies, 244*
 - rules, applying, 251*
 - summary page (CCP), 245*
 - verifying, 254*
 - IPv6 packet filtering, 259-262
 - creating and applying, 261-262*
 - ICMP, 262*
 - objectives, 260*
 - topology, 260*

- logging, 125-127
 - CCP configuration*, 126
 - settings, editing*, 126-127
 - syslog output, viewing*, 127
 - syslog support, configuring*, 125-126
- NAT, 357
- NFP (network foundation protection)
 - auto secure utility*, 53
 - plane protection*, 53-54
- NTP, 502-504
- parser views, 120-122
- port security, 192-194
- RBAC, 118-120
 - parser views*, 120-122
 - privilege levels, customizing*, 118-120
- security policies, 231
- SSH, 122-124
- SSL VPNs, 533
- strong passwords, 106-108
- use authentication, 108-113
- in-band management**, 96
- inbound traffic (ASA firewalls)**, 337-338
- incident response policies**, 32, 226
- infrastructure**, 52. *See also* NFP
- inside NAT**, 279
- installing**
 - AnyConnect client, 550
 - CD (book), 560
 - IOS-based IPS from command line, 407-412
 - IOS-based IPS with CCP, 394-400
 - configuration screen navigation*, 394
 - deployment bit on CPU resources*, 398
 - interfaces, choosing*, 396
 - IPS policy welcome page*, 395
 - public key, adding*, 397
 - router subscriptions, opening*, 395
 - SDEE, enabling*, 395
 - signature file locations, defining*, 396-397
 - signatures, compiling*, 399-400
 - summary page*, 398
 - traffic inspection direction*, 396
 - public keys, 397
- Integrated Services Routers (ISR)**, 42
- integrity**
 - data. *See* data integrity
 - defined, 8
 - SNMPv3, 130
- interdependence (NFP planes)**, 53
- interfaces**
 - ACLs, applying, 249
 - CCP
 - content pane*, 69
 - left navigation pane*, 67
 - menu bar*, 66
 - status bar*, 69
 - toolbar*, 67-68
 - clientless SSL VPNs, configuring, 537
 - default traffic flow, 335-336
 - firewalls
 - configuring*, 347-355
 - editing*, 351
 - final configuration*, 352
 - implementing*, 352-355
 - maximum allowed*, 350
 - summary page*, 350
 - VLAN number associations*, 349-350
 - IKE Phase 2, planning, 501
 - IPS policies, applying, 396

- names, 334
- security levels, assigning, 333-334
- source, testing, 515-516
- ZBF zones
 - connections*, 302
 - not belonging warning message*, 303
 - untrusted interfaces warning message*, 303
- internal risk assessment, 27
- Internet Control Message Protocol. *See* ICMP
- Internet Key Exchange. *See* IKE
- inter-VLAN routing, 182
- Intrusion Detection System. *See* IDS
- Intrusion Prevention System. *See* IPS
- IOS (router operating system)
 - class maps, 239
 - files, protecting, 106
 - firewall support, 58
 - Inspect class map, 239
 - IPS (Intrusion Prevention System), 58
 - router security features, 228
- IOS-based IPS
 - alerts, viewing, 412-416
 - command line*, 415-416
 - IPS Alert Statistics tab*, 414
 - SDEE log file screen*, 413-414
 - signatures*, 413
 - benefits, 392
 - detection methods supported, 392
 - features, 392
 - alarm summarization*, 392
 - anti-evasive techniques*, 392
 - regular expression string pattern matching*, 392
 - response actions*, 392
 - threshold configuration*, 392
 - installing from command line, 407-412
 - installing with CCP, 394-400
 - configuration screen navigation*, 394
 - deployment hit on CPU resources*, 398
 - interfaces, choosing*, 396
 - IPS policy welcome page*, 395
 - public key, adding*, 397
 - router subscriptions, opening*, 395
 - SDEE, enabling*, 395
 - signature file locations, defining*, 396-397
 - signatures, compiling*, 399-400
 - summary page*, 398
 - traffic inspection direction*, 396
 - requirements, 393
 - risk ratings, 392
 - signatures
 - actions*, 405
 - configuration changes output*, 403-404
 - disabling*, 401
 - editing*, 401
 - enabling*, 401, 404-405
 - files, obtaining*, 393-394
 - filtering based on signature IDs*, 402
 - modification buttons*, 401
 - properties, editing*, 402, 406
 - retiring*, 401
 - testing*, 406
 - unretiring*, 401
 - viewing*, 400
 - tuning, 412
- IP addresses
 - AnyConnect VPN clients, assigning, 548

- hosts, assigning, 203
- IKE Phase 2, planning, 501
- IPv6 versus IPv4, 203
- management sessions, controlling, 56
- source
 - interfaces, testing, 515-516*
 - NAT, 278-279*
- spoofing attacks, preventing, 240

IP protocol

- BOOTP service, disabling, 84
- CEF, enabling, 85
- directed broadcasts, disabling, 87
- gratuitous ARPs, disabling, 85
- Identification services, disabling, 84
- IPv6. *See* IPv6
- source
 - guards, 59, 228*
 - routing, disabling, 85*

IP Security. *See* IPsec

IPS (Intrusion Prevention System), 43, 58

- advantages/disadvantages, 379
- alerts, delivering, 385
- ASA firewalls, 230
- best practices, 386
- countermeasure actions, 379-380
 - deny attacker inline, 380*
 - deny connection inline, 380*
 - deny packet inline, 380*
 - log attacker packets, 380*
 - log pair packets, 380*
 - log victim packets, 380*
 - produce alert, 380*
 - produce verbose alert, 380*
 - request block connection, 380*
 - request block host, 380*
 - request SNMP trap, 380*
- data plane protection, 59

- evasion methods, 381
 - encryption/tunneling, 381*
 - protocol level misinterpretation, 381*
 - resource exhaustion, 381*
 - timing attacks, 381*
 - traffic fragmentation, 381*
 - traffic substitution/insertion, 381*

false positives/negatives, 377

IDS, compared, 374-376

information accuracy, 376

intelligence, 385-386

IOS-based

- alarm summarization, 392*
- alerts, 412-416*
- anti-evasive techniques, 392*
- benefits, 392*
- detection methods supported, 392*
- features, 392*
- installing from command line, 407-412*
- installing with CCP, 394-400*
- regular expression string pattern matching, 392*
- requirements, 393*
- response actions, 392*
- risk ratings, 392*
- signature files, obtaining, 393-394*
- threshold configuration, 392*
- tuning, 412*

IPv6, 381

malicious traffic, identifying, 377

- anomaly-based, 378*
- method advantages/disadvantages, 379*

policies, 378

reputation-based, 378-379

signatures, 377-378

- risk ratings, 379-382
 - actions, implementing, 381*
 - factors, 379-382*
- routers, 229
- security, implementing, 231
- sensors
 - defined, 374*
 - platforms, 375-376*
- signatures, 384-385
 - ASR (attack severity rating), 384-385*
 - groups, 384*
 - micro-engines, 384*
 - SFR (signature fidelity rating), 385*
- true positives/negatives, 377
- IPS Policies wizard, 395**
- IPsec**
 - configuring, 475-484
 - command line, 482-484*
 - IKE Phase 1 policy, 477-478*
 - local Ethernet information, entering, 477*
 - mirrored VPN for remote peers, 485-486*
 - remote peer information, entering, 477*
 - Step by Step wizard, 476*
 - summary, 481*
 - traffic encryption, 480-481*
 - transform sets, 479-480*
 - VPN tunnel status, 484*
 - goals, 465, 468-469
 - antireplay support, 468-469*
 - authentication, 468-469, 499*
 - confidentiality, 468, 499*
 - data integrity, 468, 499*
 - private addresses, hiding, 499*
 - IKE Phase 1
 - authentication, 471*
 - Diffie-Hellman key exchange, running, 471*
 - tunnels, negotiating, 469-470*
 - IKE Phase 2, 471-472
 - IP Security, 465
 - IPv6 versus IPv4, 203
 - overview, 469
 - protocols, choosing, 475
 - site-to-site VPNs. *See site-to-site VPNs*
 - tools, 475
 - topology, 468
 - traffic
 - encrypting, 472*
 - identifying for encryption, 475*
 - before IPsec, 472-473*
 - after IPsec, 473*
 - verifying, 486-490
 - VPNs, 427, 436-437
- IPv4**
 - IPv6, comparison, 202-203
 - packet filtering
 - ACLs, creating, 246*
 - applying ACLs to interfaces, 249*
 - CLI implementation, 248*
 - lines, adding, 246*
 - object groups, 251-254*
 - ordering, 247*
 - policies, 244*
 - rules, applying, 251*
 - summary page (CCP), 245*
 - verifying, 254*
- IPv6**
 - addresses
 - 128-bit, 203*
 - all-nodes multicast, 206*

- all-routers multicast*, 206
- decimal/binary/hexadecimal conversions*, 204
- formatting*, 202-204
- hexadecimal hard way example*, 204-205
- link local*, 205-206
- loopback*, 206
- multicast*, 207
- remote device communication*, 205
- solicited-node multicast*, 207
- unicast/anycast*, 206-207
- zero shortcuts*, 205
- application layer protocols
 - support, 203
- benefits, 202
- bogus addresses, filtering, 214
- headers, 203
- ICMP unused traffic, filtering, 215
- IP addresses, 203
- IPS, 381
- IPsec support, 203
- IPv4, compared, 202-203
- Layer 2 support, 203
- Layer 4 protocols support, 203
- migration, 210
- NAT, 203
- NDP (Neighbor Discovery Protocol), 203
- network masks, 203
- non-local multicast addresses, filtering, 215
- packet filtering, implementing, 259-262
 - creating and applying*, 261-262
 - ICMP*, 262
 - objectives*, 260
 - topology*, 260
- risks, 213-214
 - autoconfiguration*, 214
 - bugs*, 214
 - DHCP*, 213
 - dual stacks*, 214
 - hop-by-hop extension headers*, 214
 - ICMP*, 214
 - NDP*, 213
 - packet amplification attacks*, 214
 - tunneling*, 214
- rogue devices, 215
- routing
 - configuring*, 208-210
 - header 0s, dropping*, 215
 - router output example*, 207-208
- security
 - advantages*, 213
 - best practices*, 210-211
 - policies*, 211
- threats
 - application layer*, 212
 - DoS attacks*, 212
 - man-in-the-middle attacks*, 212
 - router attacks*, 213
 - sniffing/eavesdropping*, 212
 - spoofed packets*, 212
 - unauthorized access*, 212
- tunneling, 215
- IronPort Email Security/Web Security Appliances**, 43
- ISE (Identity Service Engine)**
 - ACS, compared, 141
 - user authentication, 14
- ISR (Integrated Services Routers)**, 42
- issuers (certificates), 447, 449

J-K

key pairs

- creating, 457
- overview, 460

keys, 431

- asymmetric encryption algorithms, 432-433, 438
- block ciphers, 432
- Diffie-Hellman key exchange, 438
- keyspace, 436
- lengths, 433
- managing, 436
- OTP (one-time pad), 431
- PKI. *See* PKI
- public
 - algorithms*, 433
 - certificates*, 448-449
 - exchanging*, 445
 - installing*, 397
 - peers, obtaining*, 448
- public key cryptography. *See* asymmetric algorithms
- stream ciphers, 432
- symmetric encryption algorithms, 432-433, 438

L

Layer 2

- ASA, 332
- IPv6 versus IPv4, 203
- loops
 - lifecycle*, 184
 - solution*, 184-187
- switch security features
 - DHCP snooping*, 228
 - dynamic ARP inspections*, 228

- IP source guards*, 228
- modules*, 228
- port security*, 228
- root guards*, 228
- storm control*, 228

threats, mitigating

- best practices*, 189
- BPDUs guards*, 190-191
- err-disabled ports, restoring*, 191-192
- negotiations, not allowing*, 190
- port security*, 192-194
- root guards*, 192
- switch ports, locking down*, 189-190
- tools*, 190
- upper-layer disruptions*, 188

toolkit, 190

trunking

- automatic switch negotiation*, 182
- native VLANs*, 181
- negotiations, not allowing*, 190
- topology*, 178
- traffic, tagging*, 180-181

VLANs

- access ports, assigning*, 178-179
- frames, following*, 181
- inter-VLAN routing*, 182
- negotiations, not allowing*, 190
- overview*, 178
- physical interfaces disadvantage*, 182
- PVST+*, 187
- router on a stick*, 182
- STP. *See* STP
- subinterfaces, creating*, 182-183
- switch ports, locking down*, 189-190
- topology*, 178

Layer 3, 332

Layer 4 protocols

50, 500

51, 500

IPv6 versus IPv4, 203

left navigation pane (CCP), 67

lengths

keys

asymmetric, 444

symmetric, 433

passwords, setting, 85

liabilities, 33

licensing

ACS, 156

CCP, 65

lifecycles

loops, 184

security, 25

lifetime

IKE Phase 1

planning, 499

tunnel negotiations, 470

IKE Phase 2, planning, 501

lines (ACLs)

adding, 246

numbers, 243

link local addresses, 205-206

list-name command, 102

local NAT, 279

**local users (ACS routers), adding,
153-154**

logging

ACLs

firewall log details, 259

logs, viewing, 258

summary syslog messages, 257

syslog destinations, 258

attacker packets, 380

best practices, 96

configuring, 126

enabling, 85

implementing, 125-127

output destinations, sending, 104-105

pair packets, 380

SDEE log file screen

filtering, 414

searching, 414

viewing, 413-414

settings, editing

CCP, 126

command line, 127

syslog, 105

destinations, 258

locking down, 56

output, viewing, 127

support, configuring, 125-126

victim packets, 380

viewing, 104

logging in (clientless SSL VPNs), 541

logical boundaries, 40-41

data centers, 41

end zones, 41

Internet, 41

policy management points, 41

logical controls, 12

login screen (ACS), 156

loopback addresses, 206

loops (Layer 2)

lifecycle, 184

solution, 184-187

M

MAC addresses

flooding, 59

port security, 192-194

- Maintenance Operations Protocol, 87**
- malicious data, protecting against, 271**
- malicious traffic**
 - general vulnerabilities, 241
 - identifying, 377
 - anomaly-based, 378*
 - method advantages/disadvantages, 379*
 - policy-based, 378*
 - reputation-based, 378-379*
 - signature-based, 377-378*
 - IP address spoofing, 240
 - reconnaissance attacks, 240-241
 - risks, reducing. *See* IPS/IDS
 - sensor responses, 379-380
 - deny attacker inline, 380*
 - deny connection inline, 380*
 - deny packet inline, 380*
 - log attacker packets, 380*
 - log pair packets, 380*
 - log victim packets, 380*
 - produce alert, 380*
 - produce verbose alert, 380*
 - request block connection, 380*
 - request block host, 380*
 - stopping, 239-240
 - TCP SYN-flood attacks, 240
- man-in-the-middle attacks, 14-16, 212**
- Manage Community dialog box, 71**
- Manage Community icon (CCP toolbar), 68**
- Management Information Base (MIB), 128**
- management plane**
 - AAA, 55
 - accounting/auditing, 98*
 - authentication, 98*
 - authorization, 98*
 - best practices, 97-98*
 - CCP implementation, 116-118*
 - command line implementation, 113-115*
 - method lists, creating, 101-102*
 - router access authentication, 100*
 - usernames/passwords/access rules storage, 98-99*
 - VPN users, 99-100*
 - defined, 52, 94
 - encrypted communications
 - best practices, 95*
 - HTTPS, implementing, 125*
 - management protocols, 103-104*
 - SSH, implementing, 122-124*
 - IOS files, protecting, 106
 - IP addresses, controlling, 56
 - logging, 104-105
 - best practices, 96*
 - configuring, 126*
 - implementing, 125-127*
 - output destinations, sending, 104-105*
 - settings, editing, 126-127*
 - syslog, 105*
 - syslog output, viewing, 127*
 - syslog support, configuring, 125-126*
 - viewing, 104*
 - NTP
 - authentication, 132*
 - CCP configuration, 131*
 - configuring, 131-132*
 - synchronization, verifying, 132*
 - overview, 55
 - passwords
 - policies, 55*
 - recommendations, 97*
 - strong, 95, 106-108*

- primary bootset storage, 132
- RBAC, 55, 101-103
 - best practices*, 95
 - implementing*, 118-122
 - parser views*, 103, 120-122
 - privilege levels, customizing*, 103, 118-120
- remote connections, 94
- security measures, 54
- SNMP, 128-131
 - agent*, 128
 - CCP configuration*, 130-131
 - command line configuration*, 131
 - defined*, 128
 - manager*, 128
 - message types*, 129
 - MIB*, 128
 - security levels*, 129
 - security model*, 129
 - sending/receiving information vulnerability*, 129
 - v1/v2 security weaknesses*, 129
 - v3 enhancements*, 130
 - v3 security levels*, 129
- syslog lockdown, 56
- system files, 96
- time accuracy, 56, 96, 105-106
- user authentication
 - best practices*, 95
 - implementing*, 108-113
- management protocols**
 - ASA firewalls, 230
 - encrypting, 103-104
 - router security, 229
- management traffic**, 94
- managing**
 - ASAs, 336-337
 - bandwidth, 59
 - in-band management, 96
 - keys, 436
 - risks
 - attackers, becoming*, 32-33
 - disaster recovery/business continuity planning*, 33
 - evidence, collecting*, 32
 - guidelines*, 31
 - incident responses*, 32
 - liabilities*, 33
 - new assets*, 27-28
 - policies*, 31
 - procedures*, 31
 - standards*, 31
 - testing security*, 30
 - transferring to someone else*, 13
 - signatures
 - ASR (attack severity rating)*, 384-385
 - groups*, 384
 - micro-engines*, 384
 - SFR (signature fidelity rating)*, 385
- masks**
 - network, 203
 - reply messages, disabling, 87
 - wildcard, 244
- maximum tolerable downtime (MTD)**, 33
- memory (CAM overflow attacks)**, 59
- memory tables**, 561
- menu bar (CCP)**, 66
- merging options (CCP templates)**, 77-78
- messages (SNMP)**, 129
- method command**, 102
- method lists (AAA)**
 - ACS authentication
 - routers, configuring*, 144, 149-150
 - testing*, 166-170

- ACS authorization
 - routers, configuring, 144, 150-151*
 - testing, 166-170*
 - applying, 152
 - creating, 101-102, 144
 - methods of attacks, 14-15**
 - back doors, 15
 - botnets, 17
 - covert channels, 17
 - DoS/DDoS, 17
 - passwords, 17
 - privilege escalation, 15
 - reconnaissance, 15
 - social engineering, 15
 - trust exploitation, 17
 - MIB (Management Information Base), 128**
 - micro-engines, 384**
 - IOS-based IPS, 399-400
 - migrating IPv6, 210**
 - models (ASA family), 330-331**
 - Modular Policy Framework (MPF), 230, 338-339**
 - modules (switches), 228**
 - Monitor button (CCP toolbar), 68**
 - monitoring**
 - ACLs, 255-257
 - SSL VPN sessions, 543-544
 - threats
 - ASA firewalls, 42*
 - centralized, 226*
 - IPS (Intrusion Prevention System), 43*
 - IronPort Email Security/Web Security Appliances, 43*
 - ISR (Integrated Services Routers), 42*
 - prevention tools, 42-43*
 - ScanSafe, 43*
 - ZBFs, 314-315*
 - MOP (Maintenance Operations Protocol), 87**
 - MPF (Modular Policy Framework), 230, 338-339**
 - MPLS (Multiprotocol Label Switching), 427**
 - MTD (maximum tolerable downtime), 33**
 - multicast addresses, 207**
 - all-nodes, 206
 - all-routers, 206
 - non-local, filtering, 215
 - solicited-node, 207
 - multistring micro-engine, 384**
-
- ## N
- NAC (Network Admission Control), 14**
 - names (interfaces), 334**
 - NAT (Network Address Translation), 203**
 - ACLs, 239
 - AnyConnect VPN exemptions, 549
 - ASAs, 357-359
 - implementing, 357*
 - verifying, 358*
 - ASA support, 332
 - configuring
 - CCP, 319-321*
 - command line, 322*
 - dynamic, 281
 - firewalls, 278-281
 - deployment options, 281*
 - inside/outside/local/global terminology, 279*
 - PAT, 279-281*
 - source IP addresses, 278-279*

- IPv6 versus IPv4, 203
- policy-based, 281
- static, 283
- terminology, 279
- verifying, 322-323
- wizard, 319-321
- National Vulnerability Database, 12**
- native VLANs, 181**
- NDP (Neighbor Discovery Protocol), 203, 213**
- Network Address Translation. *See* NAT**
- Network Admission Control (NAC), 14**
- network foundation protection. *See* NFP**
- network masks, 203**
- network policies, 30**
- Network Time Protocol. *See* NTP**
- NFP (network foundation protection), 49**
 - control plane
 - CoPP, 56*
 - CPPr, 56*
 - defined, 52*
 - nontransit traffic, 56*
 - protection/policing, 229*
 - routing protocol authentication, 56*
 - security measures, 54*
 - data plane
 - ACLs, 58*
 - bandwidth management, 59*
 - CAM overflow attacks, 59*
 - DAI, 59*
 - defined, 53*
 - DHCP snooping, 59*
 - DoS attacks, reducing, 59*
 - IOS firewall support, 58*
 - IOS IPS, 58*
 - IP source guard, 59*
 - IPS (Intrusion Prevention System), 59*
 - MAC address flooding, 59*
 - security measures, 54*
 - spoofing attacks, preventing, 59*
 - TCP intercept, 58*
 - transit traffic, 56*
 - unicast reverse path forwarding, 58*
 - unwanted traffic, blocking, 59*
- framework
 - interdependence, 53*
 - planes, 52-53*
- implementing
 - auto secure utility, 53*
 - plane protection, 53-54*
- infrastructure importance, 52
- management plane
 - AAA implementation, 113-118*
 - defined, 52, 94*
 - encrypted/authenticated SNMP, 56*
 - encrypted communications, 95*
 - encrypted management protocols, 103-104*
 - HTTPS, implementing, 125*
 - IOS files, protecting, 106*
 - IP addresses, controlling, 56*
 - logging, 96, 104-105, 125-127*
 - NTP, configuring, 131-132*
 - overview, 55*
 - password policies, 55*
 - password recommendations, 97*
 - primary bootset storage, 133*
 - RBAC, 55, 95, 101-103, 118-122*
 - remote connections, 94*
 - security measures, 54*
 - SNMP, 128-131*
 - SSH, implementing, 122-124*
 - strong passwords, 95, 106-108*
 - syslog lockdown, 56*
 - system files, 96*

- time accuracy*, 56, 96, 105-106
- user authentication*, 95, 108-113
- noAuthNoPriv security level (SNMP)**, 129
- non-local multicast addresses, filtering**, 215
- nontransit traffic protection**, 56
 - CoPP, 56
 - CPPr, 56
 - routing protocol authentication, 56
- NTP (Network Time Protocol)**, 96
 - authentication, 132
 - best practices, 105-106
 - configuring, 131-132
 - site-to-site VPNs, implementing, 502-504
 - synchronization, verifying, 132
- NVD (National Vulnerability Database)**, 12

O

- object groups**
 - applying, 253-254
 - ASA, 333
 - creating, 251-253
 - overview, 244
- objectives**, 8
 - availability, 9
 - confidentiality, 8
 - configuring routers for ACS servers, 142-144
 - integrity, 8
- One-Step Lockdown (CCP Security Audit)**, 84
- one-time pad (OTP)**, 431
- ordering ACLs**, 247
- OSCP (Online Certificate Status Protocol)**, 452
- OTP (one-time pad)**, 431
- outbound traffic**
 - ACLs, 242
 - ASAs, 338
- output (syslog)**, 127
- outside NAT**, 279
- override options (CCP templates)**, 77-78
- owners (asset classification)**, 11

P

- Packet Tracer**, 362-367
 - command line, 364-366
 - input, configuring, 332-362
 - launching, 362
 - results, 363-364
 - Telnet denial, verifying, 366-367
- packets**
 - amplification attacks, 214
 - ASA filtering, 331, 337-338
 - implementing*, 338
 - inbound traffic*, 337-338
 - outbound traffic*, 338
 - encrypting (IPsec), 472
 - filtering (ACLs), 239
 - ASA firewalls*, 230
 - creating policies*, 241
 - enforcing policies*, 241-242
 - firewalls*, 285
 - IPv4*. *See IPv4, packet filtering routers*, 229
- Packet Tracer**, 362-367
 - command line, 364-366
 - input, configuring, 332-362
 - launching, 362
 - results, 363-364
 - Telnet denial, verifying, 366-367

spoofed, mitigating, 212

stateful filtering

ASA firewalls, 230

firewalls, 276-277

static packet filtering, 274-275

parser views

creating, 103, 121-122

implementing, 120-122

user accounts, assigning, 122

passwords

ASDM, 345

attacks, 17

authentication failure rates, 85

enable secret password, setting, 86

encryption services, enabling, 85

management plane, securing, 55

minimum lengths, setting, 85

recommendations, 97

storing, 98-99

strong

best practices, 95

implementing, 106-108

PAT (Port Address Translation), 239

ACLs, 239

ASAs, 332, 357-359

dynamic, 281

firewalls, 279-281

policy-based, 281

rules verification, 358

Pearson IT Certification Practice

Test engine, 559

activating/downloading, 560

CD software, installing, 560

modes, 563

navigating, 563

peer authentication

IKE Phase 1, 471

IPsec, 468-469

Per-VLAN Spanning Tree Plus (PVST+),
187

PFS (Perfect Forward Secrecy), 501

pharming, 15

phases (security lifecycles), 25

phishing, 15

physical controls, countermeasures, 12

physical security (IPv6), 210

ping command

IPsec traffic triggers, 512

routers, 499

router-to-ACS connections, 164

signatures, 406

source interfaces with associated IP
addresses, 515-516

PKCS (Public Key Cryptography
Standards), 450, 460

PKI (Public Key Infrastructure), 441

asymmetric algorithms

examples, 444

key length, 444

overview, 433

certificate authorities, 446, 460

authenticating, 450

certificate information, 446

commercial, 446

enrolling, 450

certificates, 460

ASA self-signed, 454

functions, 452

identity, 448

issuers, 449

peers public keys, obtaining, 448

public keys, 449

revocation list location, 449

revoked, 451-452

root, 446-448

- SCEP root/identity certificates installations*, 457-459
- serial numbers*, 449
- signatures*, 449
- subjects*, 449
- thumbprint*, 449
- validity dates*, 449
- viewing in ASDM*, 455
- X.500/X.509v3*, 449
- X.500/X.509v3 certificates*, 460
- components, 461
- key pairs, 444
- PKCS (Public Key Cryptography Standards), 450, 460
- public-private key pairs, 460
- RSA
 - digital signatures, creating*, 445, 460
 - public keys, exchanging*, 445
 - public-private key pairs*, 445
- SCEP (Simple Certificate Enrollment Protocol), 451
- subordinate CA, 460
- topologies, 453
 - cross-certifying CAs*, 453
 - hierarchical with subordinate CAs*, 453
 - single root CAs*, 453
- planes (NFP)**, 52-53
 - control, 54
 - CoPP*, 56
 - CPPr*, 56
 - defined*, 52
 - nontransit traffic*, 56
 - protection/policing*, 229
 - routing protocol authentication*, 56
 - security measures*, 54
 - data
 - ACLs*, 58
 - bandwidth management*, 59
 - CAM overflow attacks*, 59
 - DAI*, 59
 - defined*, 53
 - DHCP snooping*, 59
 - DoS attacks, reducing*, 59
 - IOS firewall support*, 58
 - IOS IPS*, 58
 - IP source guard*, 59
 - IPS (Intrusion Prevention System)*, 59
 - MAC address flooding*, 59
 - security measures*, 54
 - spoofing attacks, preventing*, 59
 - TCP intercept*, 58
 - transit traffic*, 56
 - unicast reverse path forwarding*, 58
 - unwanted traffic, blocking*, 59
 - interdependence, 53
 - management
 - AAA implementation*, 113-118
 - defined*, 52, 94
 - encrypted/authenticated SNMP*, 56
 - encrypted communications*, 95
 - encrypted management protocols*, 103-104
 - HTTPS, implementing*, 125
 - IOS files, protecting*, 106
 - IP addresses, controlling*, 56
 - logging*, 96, 104-105, 125-127
 - NTP, configuring*, 131-132
 - overview*, 55
 - password policies*, 55
 - password recommendations*, 97
 - primary bootset storage*, 132

- RBAC, 55, 95, 118-122
- remote connections*, 94
- security measures*, 54
- SNMP, 128-131
- SSH, *implementing*, 122-124
- strong passwords*, 95, 106-108
- syslog lockdown*, 56
- system files*, 96
- time accuracy*, 56, 96, 105-106
- user authentication*, 95, 108-113
- platforms**
 - ACS supported, 141
 - sensors, 375-376
- policies**
 - ASA
 - applying*, 339-340
 - MPF, 338-339
 - authorization, 161-163
 - crypto, configuring, 508-510
 - IKE Phase 1
 - configuring*, 506-507
 - creating*, 477-478
 - planning*, 499-500
 - IKE Phase 2, 501-502
 - configuring*, 507-510
 - encryption*, 501
 - hashes*, 501
 - interfaces, selecting*, 501
 - lifetimes*, 501
 - peer IP addresses*, 501
 - PFS (*Perfect Forward Secrecy*), 501
 - traffic encryption*, 501
 - incident responses, 32, 226
 - IPv6, 211
 - management points, 41
 - packet-filtering ACLs
 - creating*, 241
 - enforcing*, 241-242
 - password, 55
 - security
 - application*, 30
 - content*, 28
 - creators*, 28
 - defined*, 31
 - email*, 30
 - formal procedures*, 226
 - functions*, 28
 - guideline*, 29
 - implementing*, 231
 - network*, 30
 - overview*, 28
 - remote-access*, 30
 - telephony*, 30
 - types*, 29-30
 - service
 - defined*, 297
 - traffic interaction between zones*, 297-298
 - threat mitigation, 226
- policy-based**
 - IPS/IDS, 378
 - NAT, 281
 - PAT, 281
- policy maps**
 - actions, 297
 - ASAs, 339
 - defined, 296
- polyalphabetic ciphers**, 431
- Port Address Translation. See PAT**
- ports**
 - access
 - assigning to VLANs*, 178-179
 - negotiations, not allowing*, 190
 - err-disabled, restoring, 191-192
 - root guards, 192

- security, implementing, 192-194, 228
- STP caution towards new, 187
- switch
 - BPDUs guards*, 190-191
 - locking down*, 189-190
- trunk
 - automatic switch negotiation*, 182
 - traffic tags, creating*, 180-181
- potential attackers, 13-14**
 - motivations/interests, understanding, 14
 - not becoming, 32-33
 - types, 13
- practice exams, 559**
 - activating/downloading, 560
 - CD software, installing, 560
 - Premium Edition practice exams, 561
- Premium Edition practice exams, 561**
- prevention strategies (borderless networks), 42-43**
 - ASA firewalls, 42
 - IPS (Intrusion Prevention System), 43
 - IronPort Email Security/Web Security Appliances, 43
 - ISR (Integrated Services Routers), 42
 - ScanSafe, 43
- previewing CCP commands, 83**
- primary bootset, storing, 132**
- private sector asset classifications, 11**
- privileges**
 - escalation, 15
 - levels, customizing, 103, 118-120
- procedures, 31**
- profiles**
 - AnyConnect SSL VPN connection, 545
 - authorization, 162
 - user (CCP), 78-80
 - applying*, 80
 - creating*, 79
 - restrictions*, 78
 - saving*, 80
 - verifying*, 80
- protection**
 - administrator access/protocols, 55-56
 - AAA services*, 55
 - encrypted/authenticated SNMP*, 56
 - IP addresses*, 56
 - password policies*, 55
 - RBAC*, 55
 - syslog lockdown*, 56
 - time accuracy*, 56
 - IOS files, 106
 - network foundation. *See* NFP
 - nontransit traffic, 56
 - CoPP*, 56
 - CPPr*, 56
 - routing protocol authentication*, 56
 - system files, 96
 - traffic, 480-481
 - transit traffic, 56
 - ACLs*, 58
 - bandwidth management*, 59
 - CAM overflow attacks*, 59
 - DAI*, 59
 - DHCP snooping*, 59
 - DoS attacks, preventing*, 59
 - IOS firewall support*, 58
 - IOS IPS*, 58
 - IP source guard*, 59
 - IPS (Intrusion Prevention System)*, 59
 - MAC address flooding*, 59
 - spoofing attacks, preventing*, 59
 - TCP intercept*, 58
 - unicast reverse path forwarding*, 58
 - unwanted traffic, blocking*, 59

protocols

- ACS server/router communication, 141-143
 - choosing*, 142-143
 - RADIUS, 142
 - TACACS+141
- administrator, protecting, 55-56
 - AAA services, 55
 - encrypted/authenticated SNMP*, 56
 - IP addresses, *controlling*, 56
 - password policies*, 55
 - RBAC, 55
 - syslog lockdown*, 56
 - time accuracy*, 56
- AnyConnect SSL VPNs, *choosing*, 546
- application layer, 203
- ARPs
 - dynamic*, 228
 - gratuitous, disabling*, 85
 - proxy, disabling*, 86
- CDP, *disabling*, 84
- DHCP
 - ASA, 332, 355
 - IPv6 risks, 213
 - IPv6 versus IPv4, 203
 - snooping*, 59, 228
- flaws, *exploiting*, 271
- HTTPS, *implementing*, 125
- ICMP
 - IPv6 *packet filtering*, 262
 - IPv6 risks, 214
 - mask reply messages, disabling*, 87
 - redirects, disabling*, 86
 - unreachables, disabling*, 87
 - unused traffic, filtering*, 215
- IKE Phase 1, *choosing*, 475
- IKE Phase 2, *choosing*, 475

IP

- BOOTP service, disabling*, 84
- CEF, *enabling*, 85
- directed broadcasts, disabling*, 87
- gratuitous ARPs*, 85
- identification services, disabling*, 84
- IPv6. *See* IPv6
- source guards*, 59, 228
- source routing, disabling*, 85
- IPsec. *See* IPsec
- IPv6
 - 128-bit addresses, 203
 - all-nodes multicast addresses*, 206
 - all-routers multicast addresses*, 206
 - application layer*, 203, 212
 - benefits*, 202
 - bogus addresses, filtering*, 214
 - decimal/binary/hexadecimal conversions*, 204
 - DoS attacks, reducing*, 212
 - formatting addresses*, 202-204
 - headers*, 203
 - hexadecimal hard way example*, 204-205
 - ICMP unused traffic, filtering*, 215
 - IP addresses, 203
 - IPS, 381
 - IPsec support, 203
 - IPv4, *compared*, 202-203
 - Layer 2 support*, 203
 - Layer 4 protocols support*, 203
 - link local addresses*, 205-206
 - loopback addresses*, 206
 - man-in-the-middle attacks*, 212
 - migration*, 210
 - multicast addresses*, 207
 - NAT, 203

- NDP (*Neighbor Discovery Protocol*), 203
- network masks, 203
- non-local multicast addresses, filtering, 215
- packet filtering, 259-262
- remote device communication, 205
- risks, 213-214
- rogue devices, 215
- router attacks, 213
- router output example, 207-208
- routing, configuring, 208-210
- routing header 0s, dropping, 215
- security advantages, 213
- security best practices, 210-211
- sniffing/eavesdropping, 212
- solicited-node multicast addresses, 207
- spoofed packets, 212
- tunneling, 215
- unauthorized access threats, 212
- unicast/anycast addresses, 206-207
- zero shortcuts, 205
- Layer 4
 - IPv6 versus IPv4, 203
 - protocol 50, 500
 - protocol 51, 500
- level misinterpretations, 381
- management
 - ASA firewalls, 230
 - encrypting, 103-104
 - router security, 229
- MOP, disabling, 87
- NDP, 203, 213
- NTP, 96
 - authentication, 132
 - best practices, 105-106
 - CCP configuration, 131
 - configuring, 131-132
 - site-to-site VPNs, implementing, 502-504
 - synchronization, verifying, 132
- OSCP (*Online Certificate Status Protocol*), 452
- RADIUS
 - overview, 142
 - TACACS+, compared, 142-143
- routing
 - ACLs, 239
 - ASA firewalls, 230
 - authentication, 56, 229-230
 - control plane, 56
 - IPv6, 211
 - routers, 229
- SCEP (*Simple Certificate Enrollment Protocol*), 451, 457-459
- secured management, 43
- SNMP
 - agent, 128
 - CCP configuration, 130-131
 - command line configuration, 131
 - defined, 128
 - disabling, 86
 - logs, receiving, 104
 - management plane, 56
 - manager, 128
 - message types, 129
 - MIB, 128
 - security levels, 129
 - security model, 129
 - sending/receiving information vulnerability, 129
 - v1/v2 security weaknesses, 129
 - v3 enhancements, 130
 - v3 security levels, 129

SSL. *See* SSL

STP, 183

loop lifecycle, 184

new ports, 187

PVST+, 187

Rapid Spanning Tree, 187-188

verification/annotations, 184-187

TACACS+

overview, 141

RADIUS, compared, 142-143

TCP

intercept, 58

keepalives, enabling, 85

SYN-flood attacks, 240

SYN-Wait times, setting, 85

TLS, 532-534

Provide feedback to Cisco icon (CCP toolbar), 68

proxy ARPs, disabling, 86

Public Key Infrastructure. *See* PKI

public keys, 431

algorithms, 433

certificates, 448-449

cryptography. *See* asymmetric algorithms

exchanging, 445

installing, 397

peers, obtaining, 448

PVST+ (Per-VLAN Spanning Tree Plus), 187

Q

QoS (Quality of Service), 239

qualitative risk analysis, 26

quantitative risk analysis, 26

R

RADIUS (Remote Authentication Dial-In User Service)

overview, 142

TACACS+, compared, 142-143

Rapid Spanning Tree, configuring, 187-188

RBAC (role-based access control), 55, 101-103

best practices, 95

implementing, 118-122

management plane, 55

parser views

best practices, 103

creating, 121-122

implementing, 120-122

user accounts, assigning, 122

privilege levels, customizing, 103, 118-120

reconnaissance attacks, 15, 240-241

recovery point objective (RPO), 33

recovery time objective (RTO), 33

redirects (ICMP), disabling, 86

reflexive access lists, 229

Refresh icon (CCP toolbar), 68

regular expressions, string pattern matching, 392

regulatory compliance, as risks, 28

remote-access

policies, 30

VPNs, 427

Remote Authentication Dial-In User Service. *See* RADIUS

reports

ACS, 165-166

Security Audit Report Card, 82

- reputation-based IPS/IDS, 378-379
- request block sensor responses
 - connections, 380
 - hosts, 380
- request SNMP trap sensor response, 380
- restoring err-disabled ports, 191-192
- retiring signatures, 401
- Reverse Path Forwarding (RPF), 87
- revocation list location (certificates), 449
- revoked certificates, 451-452
- risk ratings. *See* RRs
- risks
 - analysis, 25-26
 - cost-benefit analysis, 9-10*
 - current posture assessment, 26-27*
 - qualitative, 26*
 - quantitative, 26*
 - defined, 10
 - end users, 224-225
 - firewall protection against
 - exposure of sensitive systems to untrusted individuals, 271*
 - malicious data, 271*
 - protocol flaw exploitation, 271*
 - unauthorized users, 271*
- IPv6, 213-214
 - autoconfiguration, 214*
 - bugs, 214*
 - DHCP, 213*
 - dual stacks, 214*
 - hop-by-hop extension headers, 214*
 - ICMP, 214*
 - NDP, 213*
 - packet amplification attacks, 214*
 - tunneling, 214*
- managing, 26-28
 - assuming, 13*
 - attackers, becoming, 32-33*
 - disaster recovery/business continuity planning, 33*
 - evidence, collecting, 32*
 - guidelines, 31*
 - incident responses, 32*
 - liabilities, 33*
 - new assets, 27-28*
 - policies, 31*
 - procedures, 31*
 - standards, 31*
 - testing security, 30*
 - transferring to someone else, 13*
- regulatory compliance, 28
- threat mitigation/containment strategies, designing, 224
- Rivest, Shamir, Adleman. *See* RSA algorithm
- rogue routers, 215
- role-based access control. *See* RBAC
- roles
 - asset classification, 11
 - RBAC, 101-103
 - best practices, 95*
 - implementing, 118-122*
 - management plane, 55*
 - parser views, 103, 120-122*
 - privilege levels, customizing, 103, 118-120*
 - separation of duties, 16
- root certificates, 446-448
 - authenticating, 450
 - installing with SCEP, 457-459
 - CA server details, 457*
 - command line, 458-459*
 - details, viewing, 459*

- enrollment modes*, 458
 - key pairs, creating*, 457
 - success message*, 459
- issuers, 447
- manually installing, 455-456
- public keys, 448
- serial numbers, 447
- subjects, 447
- thumbprint, 448
- validity dates, 447
- root guards**, 192, 228
- routers**
 - access authentication, 100
 - ACS
 - communication protocols*, 141-143
 - interactions, troubleshooting*, 164-170
 - interoperation, configuring*, 142-154
 - attacks, 213
 - CCP communication, configuring, 69-70
 - communities, 70-73
 - adding devices*, 72-73
 - creating*, 71
 - defined*, 71
 - discovering devices*, 73
 - maximum devices*, 71
 - firewalls. *See* firewalls
 - IOS-based IPS
 - alarm summarization*, 392
 - alerts*, 412-416
 - anti-evasive techniques*, 392
 - benefits*, 392
 - detection methods supported*, 392
 - features*, 392
 - installing from command line*, 407-412
 - installing with CCP*, 394-400
 - regular expression string pattern matching*, 392
 - requirements*, 393
 - response actions*, 392
 - risk ratings*, 392
 - signature files, obtaining*, 393-394
 - signatures*. *See* *signatures, IOS-based IPS*
 - threshold configuration*, 392
 - tuning*, 412
 - IOS security features, 228
 - IPsec
 - authentication*, 471
 - Diffie-Hellman key exchange, running*, 471
 - encrypting traffic*, 472
 - IKE Phase 1 tunnels, negotiating*, 469-470
 - IKE Phase 2*, 471-472
 - traffic after*, 473
 - traffic before*, 472-473
 - ISR (Integrated Services Routers), 42
 - on a stick, 182
 - operating system. *See* IOS
 - pinging, 499
 - rogue, 215
 - security features, 227-229
 - AAA, 229
 - ACLs (*packet-filtering*), 229
 - CBAC, 229
 - control plane protection/policing*, 229
 - IPS, 229
 - management protocols*, 229
 - reflexive access lists*, 229
 - routing protocol authentication*, 229
 - VPNs, 229
 - Zone-Based Firewalls, 229

- subscriptions, opening, 395
 - traffic. *See* traffic
 - VLANs
 - inter-VLAN routing*, 182
 - router on a stick*, 182
 - subinterfaces, creating*, 182-183
 - routing**
 - ASA, 332, 356-357
 - header 0s, dropping, 215
 - IPv6, configuring, 208-210
 - protocols
 - ACLs*, 239
 - ASA firewalls*, 230
 - control plane*, 56
 - IPv6*, 211
 - routers*, 229
 - RPF (Reverse Path Forwarding), 87
 - RPO (recovery point objective), 33
 - RRs (risk ratings), 379-382
 - calculation factors, 381
 - factors, 379-382
 - IOS-based IPS, 392
 - IPS/IDS actions, 381
 - RSA (Rivest, Shamir, Adleman)
 - algorithm**, 444
 - defined, 444
 - digital signatures, 445, 460
 - public keys, exchanging, 445
 - public-private key pairs, 445
 - RTO (recovery time objective), 33
 - rule of least privilege**, 16
 - rules**
 - access, storing, 98-99
 - ACLs, applying, 251
 - ASA access, 359-362
 - firewalls
 - access*, 284
 - guidelines*, 285-286
 - implementation consistency*, 286-287
 - NAT
 - adding*, 357
 - verifying*, 358
 - PAT, verifying, 358
-
- S**
 - Sarbanes-Oxley (SOX)**, 28
 - saving**
 - primary bootset, 132
 - Security Audit Report Card, 82
 - user profiles, 80
 - ScanSafe**, 43
 - SCEP (Simple Certificate Enrollment Protocol)**, root/identity certificates, installing, 457-459
 - CA server details, 457
 - command line, 458-459
 - details, viewing, 459
 - enrollment mode, 458
 - key pairs, creating, 457
 - success message, 459
 - scheduler**
 - allocation, 86
 - intervals, 86
 - SDEE (Security Device Event Exchange)**, 385
 - alerts, delivering, 385
 - enabling, 395
 - log file screen
 - filtering*, 414
 - searching*, 414
 - viewing*, 413-414
 - Search icon (CCP toolbar)**, 68
 - Secure Shell**. *See* SSH
 - Secure Sockets Layer**. *See* SSL

secured management protocols, 43**SecureX architecture, 42**

- AnyConnect Client, 42
- context awareness, 42
- SIO (Security Intelligence Operations), 42
- TrustSec, 42

Security Audit (CCP), 81

- authentication failure rates, 85
- banners, setting, 85
- disabling
 - BOOTP service, disabling, 84*
 - CDP, 84*
 - Finger service, 84*
 - gratuitous ARPs, 85*
 - ICMP redirects, 86*
 - identification service, disabling, 84*
 - IP directed broadcasts, 87*
 - IP mask reply messages, 87*
 - IP source route, 85*
 - IP unreachable, 87*
 - MOP, 87*
 - proxy ARPs, 86*
 - SNMP, 86*
 - TCP small servers service, 84*
 - UDP small servers service, 84*

enabling

- AAA, 87*
- CEF, 85*
- firewalls, 87*
- logging, 85*
- password encryption, 85*
- RPF, 87*
- secret password, setting, 86*
- SSH, 87*
- TCP keepalives, 85*
- Telnet settings, 86*

- HTTP service/vty lines access class, setting, 87

- interface connections, 82

- minimum password lengths, 85

- One-Step Lockdown, 84

- options, 81

- potential problems

- fixing, 82-83*

- identifying, 82*

- scheduler, setting

- allocation, 86*

- intervals, 86*

- starting, 81

- summary, 83

- TCP SYN-Wait times, setting, 85

- users, configuring, 86

Security Device Event Exchange.

See SDEE

Security Intelligence Operations (SIO), 42, 231, 386**security terms, 10****self zones, 297-298****sensors**

- alerts, delivering, 385

- countermeasure actions, 379-380

- deny attacker inline, 380*

- deny connection inline, 380*

- deny packet inline, 380*

- log attacker packets, 380*

- log pair packets, 380*

- log victim packets, 380*

- produce alert, 380*

- produce verbose alert, 380*

- request block connection, 380*

- request block host, 380*

- request SNMP trap, 380*

- defined, 374

- intelligence
 - collecting*, 385-386
 - global correlation*, 386
- IPS/IDS
 - best practices*, 386
 - comparison*, 375-376
- malicious traffic, identifying, 377
 - anomaly-based IPS/IDS*, 378
 - method advantages/disadvantages*, 379
 - policy-based IPS/IDS*, 378
 - reputation-based IPS/IDS*, 378-379
 - signature-based IPS/IDS*, 377-378
- platforms, 375-376
- risk ratings, 379-382
 - actions, implementing*, 381
 - factors*, 379-382
- separation of duties, 16
- serial numbers (certificates), 447, 449
- servers
 - ACS. *See* ACS
 - central, 98-99
 - DHCP, 355
 - DNS, 305
 - SNMP logs, receiving, 104
 - syslogs, receiving, 104
- services
 - AAA, 55
 - BOOTP, disabling, 84
 - Finger, disabling, 84
 - HTTP access class, configuring, 87
 - identification, disabling, 84
 - micro-engine, 384
 - password encryption, enabling, 85
- policies
 - traffic interaction between zones*, 297-298
 - ZBFs, 297
- SIO (Security Intelligence Operations), 231
- TCP small servers, disabling, 84
- UDP small servers, disabling, 84
- SET messages, 129
- SFR (signature fidelity rating), 382, 385
- signatures
 - alerts, viewing, 413
 - certificates, 449
 - digital, 438
 - creating*, 445
 - DSA (Digital Signature Algorithm)*, 444
 - RSA, 460
 - VPNs, 435-436
 - groupings, 384
 - IOS-based IPS
 - actions*, 405
 - compiling*, 399-400
 - configuration changes output*, 403-404
 - configuration files, locating*, 397
 - disabling*, 401
 - editing*, 401
 - enabling*, 401, 404-405
 - files, obtaining*, 393-394
 - filtering based on signature IDs*, 402
 - locations, defining*, 396
 - modification buttons*, 401
 - properties, editing*, 402, 406
 - public key, adding*, 397
 - retiring*, 401
 - testing*, 406

- unretiring*, 401
- viewing*, 400
- IPS/IDS, 377-378
 - ASR (attack severity rating)*, 384-385
 - groups*, 384
 - micro-engines*, 384
 - SFR (signature fidelity rating)*, 385
- retired/unretired/enabled/disabled matrix, 384
- Simple Network Management Protocol. See SNMP**
- single-console management tools, 43
- single root CAs, 453
- SIO (Security Intelligence Operations), 42, 231, 386
- site-to-site VPNs, 427
 - crypto policies, configuring, 508-510
 - digital certificates, 504-505
 - file sharing needs assessment, 498
 - IKE Phase 1, 499-500
 - authentication*, 499
 - configuring*, 506-507
 - Diffie-Hellman key exchange*, 499
 - encryption*, 499
 - hashes*, 499
 - lifetimes*, 499
 - troubleshooting*, 512
 - IKE Phase 2, 501-502
 - configuring*, 507-510
 - encryption*, 501
 - hashes*, 501
 - interfaces, selecting*, 501
 - lifetimes*, 501
 - peer IP addresses*, 501
 - PFS*, 501
 - traffic encryption*, 501
 - NTP, implementing, 502-504
 - configuring*, 502
 - verifying*, 503-504
 - pinging routers, 499
 - protocols, 499
 - SSL VPNs, compared, 532-533
 - troubleshooting
 - configuration, verifying*, 511
 - IKE Phase 1*, 512
 - IKE Phase 2*, 522-525
 - router 1 configuration*, 513-515
 - router 2 configuration*, 517-521
 - source interfaces with associated IP addresses*, 515-516
 - traffic triggers*, 512
- sniffing (IPv6), 212
- SNMP (Simple Network Management Protocol), 56**
 - agent, 128
 - configuring
 - CCP*, 130-131
 - command line*, 131
 - defined, 128
 - disabling, 86
 - logs, receiving, 104
 - management plane protection, 56
 - manager, 128
 - message types, 129
 - MIB, 128
 - security levels, 129
 - security model, 129
 - sending/receiving information
 - vulnerability, 129
 - v1/v2 security weaknesses, 129
 - v3 security
 - enhancements*, 130
 - security levels*, 129
- social engineering attacks, 15**

- solicited-node multicast addresses, 207
- source IP addresses
 - interfaces, testing, 515-516
 - NAT, 278-279
- SOX (Sarbanes-Oxley), 28
- Spanning Tree Protocol. *See* STP
- split tunneling, 554-555
- spoofing attacks, preventing, 59
- SSH (Secure Shell), 87
 - enabling, 87
 - implementing, 122-124
- SSL (Secure Sockets Layer), 437-438
 - AnyConnect VPNs
 - AnyConnect client installation*, 550
 - AnyConnect software packages, choosing*, 546-547
 - authentication*, 547-548
 - clientless SSL VPNs, compared*, 545
 - command line configuration*, 550-552
 - connection profiles, creating*, 545
 - digital certificates*, 546
 - DNS, configuring*, 548
 - domain name configurations*, 548
 - groups*, 552-553
 - IP address pool, assigning*, 548
 - NAT exemptions*, 549
 - protocols, choosing*, 546
 - split tunneling*, 554-555
 - SSL_AnyConnect connection profile/tunnel group/Group correlation*, 553
 - summary page*, 550
 - VPN AnyConnect Wizard, starting*, 545
 - WINS, configuring*, 548
 - clientless VPNs
 - authentication*, 538-540
 - CLI implementation*, 540-541
 - configuring on ASA*, 535-544
 - digital certificates*, 537
 - interfaces*, 537
 - logging in*, 541
 - session details, viewing*, 543-544
 - SSL VPN Wizard*, 535-544
 - features, 534
 - overview, 427
 - TLS, compared, 532-534
 - VPNs
 - implementing*, 437-438
 - IPsec, compared*, 532-533
 - types*, 534
 - wizard*, 535-544
- standard ACLs
 - defined, 242
 - extended ACLs, compared, 243
 - identifying, 242
 - IPv4 packet filtering. *See* IPv4, packet filtering
- standards
 - defined, 31
 - PKCS (Public Key Cryptography Standards), 450, 460
- Startup wizard (ASDM), 346-347
- stateful filtering, 230, 276-277
 - ASA, 331
- static NAT, 283
- static packet filtering, 274-275
- static routes, 356-357
- status bar (CCP), 69
- Step by Step wizard, 476

storing

- primary bootset, 132
- usernames/passwords/access rules, 98-99

storm control (switches), 228**STP (Spanning Tree Protocol), 183**

- loops lifecycle, 184
- new ports, 187
- PVST+, 187
- Rapid Spanning Tree, 187-188
- verification/annotations, 184-187

strategies

- changing nature of networks, 40
- logical boundaries, 40-41
 - data centers, 41*
 - end zones, 41*
 - Internet, 41*
- policy management points, 41
- prevention, 42-43
 - ASA firewalls, 42*
 - IPS (Intrusion Prevention System), 43*
 - IronPort Email Security/Web Security Appliances, 43*
 - ISR (Integrated Services Routers), 42*
 - ScanSafe, 43*
- secured management protocols, 43
- SecureX architecture, 42
 - AnyConnect Client, 42*
 - context awareness, 42*
 - SIO (Security Intelligence Operations), 42*
 - TrustSec, 42*
- single-console management tools, 43
- threat mitigation/containment, 224
 - ACLs. See ACLs*
 - ASA firewalls, 230*

CSM (Cisco Security Manager), 231

end-user education, 226

end user risks, 224-225

IPS (Intrusion Prevention System), 231

mitigation policies/techniques, 226

opportunities for attacks, 224

policy procedures, 226

potential risks, 224

routers, 227-229

SIO (Security Intelligence Operations), 231

switches, 227

VPN connectivity, 43

stream ciphers, 432**strings**

- micro-engine, 384
- pattern matching (regular expressions), 392

study plan, 562

subinterfaces (VLANs), creating, 182-183

subordinate CAs, 453, 460

subscriptions (routers), opening, 395

substitution ciphers, 431

switches

- access ports, assigning, 178-179
- err-disabled ports, restoring, 191-192
- ports
 - BPDU guards, 190-191*
 - locking down, 189-190*
- root guards, 192
- security features, 227
 - BPDU guards, 228*
 - DHCP snooping, 228*
 - dynamic ARP inspections, 228*
 - IP source guards, 228*

- modules*, 228
- port security*, 228
- root guards*, 228
- storm control*, 228
- trunking
 - automatic switch negotiation*, 182
 - native VLANs*, 181
 - negotiations, not allowing*, 190
 - security best practices*, 189
 - security tools*, 190
 - switch ports, locking down*, 189-190
 - traffic tags, creating*, 180-181
- symmetric algorithms**, 432-433, 438
- syslog**
 - locking down, 56
 - logging, 105
 - output, viewing, 127
 - receiving, 104
 - summary messages, 257
 - support, configuring, 125-126
- system files, protecting**, 96

T

- TACACS+ (Terminal Access Control Access Control Server)**
 - overview, 141
 - RADIUS, compared, 142-143
- target value rating (TVR)**, 382
- TCP (Transmission Control Protocol)**
 - intercept, 58
 - keepalives, enabling, 85
 - small servers service, disabling, 84
 - SYN-flood attacks, 240
 - SYN-Wait times, setting, 85
- telephony policies**, 30

- Telnet**
 - denial, verifying, 366-367
 - settings, enabling, 86
- templates (CCP)**, 74-78
 - applying, 76-77
 - creating, 75-76
 - merging/overriding options, 77-78
- Terminal Access Control Access Control Server.** *See* TACACS+
- test aaa command**, 115, 164-165
- test preparation tools**
 - activating/downloading exams, 560
 - CD software, installing, 560
 - Cisco Learning Network, 561
 - memory tables. *See* memory tables
 - Pearson IT Certification Practice Test engine
 - modes*, 563
 - navigating*, 563
 - practice exams, 559
 - Premium Edition practice exams, 561
 - videos, 562
- testing.** *See also* verifying
 - AAA connections, 115
 - ASA connections, 345
 - IPsec traffic triggers, 512
 - Packet Tracer, 362-367
 - command line*, 364-366
 - input, configuring*, 332-362
 - launching*, 362
 - results*, 363-364
 - Telnet denial, verifying*, 366-367
- router-to-ACS**
 - AAA, 164-165
 - connections*, 164
 - method lists*, 166-170

- security, 30
- source interfaces with associated IP addresses, 515-516
- threats, 14-15**
 - back doors, 15
 - botnets, 17
 - covert channels, 17
 - defined, 9-10
 - DoS/DDoS, 17
 - evidence, collecting, 32
 - incident response policies, 32
 - IPv6
 - application layer*, 212
 - DoS attacks*, 212
 - man-in-the-middle attacks*, 212
 - router attacks*, 213
 - spoofed packets*, 212
 - unauthorized access*, 212
 - Layer 2, mitigating
 - best practices*, 189
 - BPDU guards*, 190-191
 - err-disabled ports, restoring*, 191-192
 - negotiations, not allowing*, 190
 - port security*, 192-194
 - root guards*, 192
 - switch ports, locking down*, 189-190
 - tools*, 190
 - upper-layer disruptions*, 188
 - malicious traffic
 - general vulnerabilities*, 241
 - IP address spoofing*, 240
 - reconnaissance attacks*, 240-241
 - risks, reducing*. See *IPS/IDS*
 - stopping*, 239-240
 - TCP SYN-flood attacks*, 240
 - man-in-the-middle attacks, 14-16
 - mitigation/containment strategies, designing, 224
 - ACLs*. See *ACLs*
 - application layer visibility*, 226
 - ASA firewalls*, 230
 - centralized monitoring*, 226
 - CSM (Cisco Security Manager)*, 231
 - defense in depth*, 226
 - end-user education*, 226
 - end user risks*, 224-225
 - incident responses*, 226
 - IPS (Intrusion Prevention System)*, 231
 - mitigation policies/techniques*, 226
 - opportunities for attacks*, 224
 - policy procedures*, 226
 - potential risks*, 224
 - routers*, 227-229
 - SIO services*, 231
 - switches*, 227
 - monitoring, 42-43
 - ASA firewalls*, 42
 - IPS (Intrusion Prevention System)*, 43
 - IronPort Email Security/Web Security Appliances*, 43
 - ISR (Integrated Services Routers)*, 42
 - ScanSafe*, 43
 - password attacks, 17
 - pharming, 15
 - phishing, 15
 - potential attackers, 13-14
 - motivations/interests, understanding*, 14
 - types*, 13
 - privilege escalation, 15
 - reconnaissance, 15

- social engineering, 15
- trust exploitation, 17
- vectors, 14
- thresholds, configuring, 392**
- thumbprints (certificates), 448-449**
- time accuracy, 56, 96, 105-106.**
 - See also* NTP
- timing attacks (IPS/IDS), 381**
- TLS (Transport Layer Security), 532-534**
- toolbars (CCP), 67-68**
- tools**
 - ASAs, 336-337
 - IPsec, 475
 - Layer 2 security, 190
- traffic**
 - ASA, filtering, 337-338
 - default flow, 335-336*
 - implementing, 338*
 - inbound, 337-338*
 - outbound traffic, 338*
 - routing, 356-357*
 - encrypting
 - identifying, 475*
 - IKE Phase 2, planning, 501*
 - IPsec, 472, 480-481*
 - after IPsec, 473*
 - before IPsec, 472-473*
 - fragmentation, 381
 - inspection direction, choosing, 396
 - IPsec triggering, testing, 512
 - malicious
 - countermeasure actions, 379-380*
 - general vulnerabilities, 241*
 - identifying, 377-379*
 - IP address spoofing, 240*
 - reconnaissance attacks, 240-241*
 - risks, reducing. See IPS/IDS*
 - stopping, 239-240*
 - TCP SYN-flood attacks, 240*
 - management, 94
 - nontransit, 56
 - CoPP, 56*
 - CPPr, 56*
 - routing protocol authentication, 56*
 - outbound, 242
 - sensors, 374
 - spoofed packets, mitigating, 212
 - substitution/insertion, 381
 - transit. *See* transit traffic
 - ZBFs, 295
 - interaction between zones, 297-298*
 - self zones, 297-298*
- transferring risks to someone else, 13**
- transform sets, 479**
 - creating, 479
 - default, 479
 - selecting, 479
- transit traffic, 56**
 - ACLs, 58
 - bandwidth management, 59
 - CAM overflow attacks, 59
 - DAI, 59
 - DHCP snooping, 59
 - DoS attacks, preventing, 59
 - IOS
 - firewall support, 58*
 - IPS, 58*
 - IP source guard, 59
 - IPS (Intrusion Prevention System), 59
 - MAC address flooding, 59
 - spoofing attacks, preventing, 59

- TCP intercept, 58
- unicast reverse path forwarding, 58
- unwanted traffic, blocking, 59
- Transmission Control Protocol. *See* TCP**
- transparent firewalls, 276-278**
- Transport Layer Security (TLS), 532-534**
- transposition ciphers, 431**
- trap messages, 129**
- troubleshooting**
 - ACS, 164-170
 - AAA, 164-165*
 - connections, 164*
 - method lists, 166-170*
 - reports, 165-166*
 - IPsec site-to-site VPNs
 - configuration, verifying, 511*
 - IKE Phase 1, 512*
 - IKE Phase 2, 522-525*
 - router 1 configuration, 513-515*
 - router 2 configuration, 517-521*
 - source interfaces with associated IP addresses, testing, 515-516*
 - traffic triggers, 512*
 - IPv6, 214
- true negatives, 377**
- true positives, 377**
- trunking**
 - automatic switch negotiation, 182
 - native VLANs, 181
 - threats, mitigating
 - best practices, 189*
 - BPDU guards, 190-191*
 - err-disabled ports, restoring, 191-192*
 - negotiations, not allowing, 190*
 - port security, 192-194*
 - root guards, 192*
 - switch ports, locking down, 189-190*
 - tools, 190*
- topology, 178
- traffic, tagging, 180-181
- trust exploitation, 17**
- TrustSec, 42**
- tuning IPS, 412**
- tunneling**
 - IKE Phase 1, 469-470
 - IKE Phase 2, 471-472
 - IPsec, troubleshooting, 522-525
 - IPS/IDS, 381
 - IPv6, 214-215
 - split, 554-555
 - VPN
 - status, 484*
 - verifying, 486-490*
- TVR (target value rating), 382**
- type command, 102**
- types**
 - centralized servers, 98-99
 - hashes, 434
 - IPv6 addresses
 - all-nodes multicast, 206*
 - all-routers multicast addresses, 206*
 - link local, 206*
 - loopback, 206*
 - multicast, 207*
 - solicited-node multicast, 207*
 - unicast/anycast, 206-207*
 - malicious traffic
 - general vulnerabilities, 241*
 - IP address spoofing, 240*
 - reconnaissance attacks, 240-241*
 - TCP SYN-flood attacks, 240*

potential attackers, 13
 security policies, 29-30
 application, 30
 email, 30
 guideline, 29
 network, 30
 remote-access, 30
 telephony, 30
 SNMP messages, 129
 SSL, 534
 VPNs, 427
 IPsec, 427
 MPLS, 427
 SSL, 427

U

UDP port 500, 500
 UDP port 4500, 500
 UDP small servers service, disabling, 84
 unauthorized access threats, 212
 unauthorized users protection, 271
 unicast addresses, 206-207
 unretiring signatures, 401
 unwanted traffic, blocking, 59
 updates (exam), 573-574
 companion website, 573
 print version versus online version, 574
 URLs, filtering, 230
 uRPF (Unicast Reverse Path Forwarding), 58
 users
 accounts
 ACS, 160
 parser views, assigning, 122
 ACS router configuration, adding, 153-154
 asset classification, 11

authentication
 best practices, 95
 implementing, 108-113
 requiring, 14
 SSL VPNs, 538-540
 configuring, 86
 educating, 226
 groups, creating, 158
 names, 345
 storing, 98-99
 packets, encrypting, 472
 profiles, 78-80
 AnyConnect SSL VPN connection, creating, 545
 applying, 80
 creating, 79
 restrictions, 78
 saving, 80
 verifying, 80
 risks, 224-225
 unauthorized, 271
 verifying. *See* AAA
 VPN, 99-100

V

validity dates (certificates), 447, 449
 verifying. *See also* testing
 AAA, 146-147
 ACL configurations, 254
 ASA connections, 345
 data integrity, 428-430, 434
 IPsec, 486-490
 IPsec site-to-site VPNs, 511
 router 1 configuration, 513-515
 router 2 configuration, 517-521
 NAT, 322-323, 358

- NTP, 503-504
- PAT rules, 358
- router-to-ACS
 - AAA, 164-165
 - connections, 164
 - method lists, 166-170
- STP, 184-187
- Telnet denial, 366-367
- user profiles (CCP), 80
- users. *See* AAA
- ZBFs, 314-315, 319
- videos (book CD), 562**
- viewing**
 - ACS groups summary, 159
 - alerts
 - command line*, 415-416
 - IPS Alert Statistics tab*, 414
 - SDEE log file screen*, 413-414
 - signatures*, 413
 - certificates, 455
 - logs, 104, 258
 - SDEE log file screen, 413-414
 - signatures, 400
 - SSL VPN sessions, 543-544
 - syslog output, 127
- views**
 - creating, 103, 121-122
 - implementing, 120-122
 - user accounts, assigning, 122
- virtual private networks. *See* VPNs**
- VLANs (virtual LANs)**
 - access ports, assigning, 178-179
 - frames, following, 181
 - interface number associations, 349-350
 - inter-VLAN routing, 182
 - native, 181
 - overview, 178
 - physical interfaces disadvantage, 182
 - router on a stick, 182
 - STP, 183
 - loop lifecycle*, 184
 - new ports*, 187
 - PVST+*, 187
 - Rapid Spanning Tree*, 187-188
 - verification/annotations*, 184-187
 - subinterfaces, creating, 182-183
 - threats, mitigating
 - best practices*, 189
 - BPDU guards*, 190-191
 - err-disabled ports, restoring*, 191-192
 - negotiations, not allowing*, 190
 - port security*, 192-194
 - root guards*, 192
 - switch ports, locking down*, 189-190
 - tools*, 190
 - topology, 178
 - trunking
 - automatic switch negotiation*, 182
 - native VLANs*, 181
 - traffic, tagging*, 180-181
- VPNs**
 - ACLs, 239
 - antireplay functionality, 430
 - AnyConnect SSL VPNs
 - AnyConnect client installation*, 550
 - AnyConnect software packages, choosing*, 546-547
 - authentication*, 547-548
 - clientless SSL VPNs, compared*, 545
 - command line configuration*, 550-552
 - connection profiles, creating*, 545
 - digital certificates*, 546

- DNS, *configuring*, 548
- domain name configurations, 548
- groups, 552-553
- IP address pool, *assigning*, 548
- NAT exemptions, 549
- protocols, *choosing*, 546
- split tunneling, 554-555
- SSL_AnyConnect connection
 - profile/tunnel group/Group correlation, 553
- summary page, 550
- VPN AnyConnect Wizard,
 - starting*, 545
- WINS, *configuring*, 548
- AnyConnect Wizard, *starting*, 545
- ASA firewalls, 230, 333
- authentication, 430, 438
- benefits, 427-428
- clientless SSL
 - authentication*, 538-540
 - CLI implementation*, 540-541
 - configuring on ASA*, 535-544
 - digital certificates*, 537
 - interfaces*, 537
 - logging in*, 541
 - session details, viewing*, 543-544
 - SSL VPN Wizard*, 536-537
- components, 438
- confidentiality, 428, 438
- connectivity, 43
- cryptology, 430
 - asymmetric*, 433, 438
 - block ciphers*, 432
 - ciphers*, 430-431
 - Diffie-Hellman key exchange*, 438
 - digital signatures*, 435-436, 438
 - hashes*, 434
 - key length*, 433
 - key management*, 436
 - keys*, 431
 - stream ciphers*, 432
 - symmetric*, 432-433, 438
- data integrity, 428-430, 438
- IPsec, *configuring*, 436-437, 475-484
 - command line*, 482-484
 - IKE Phase 1 policy*, 477-478
 - local Ethernet information, entering*, 477
 - mirrored VPN for remote peers*, 485-486
 - remote peer information, entering*, 477
 - status*, 484
 - Step by Step wizard*, 476
 - summary*, 481
 - traffic encryption*, 480-481
 - transform sets*, 479-480
 - verification*, 486-490
- IPsec site-to-site
 - configuration, verifying*, 511
 - crypto policies, configuring*, 508-510
 - digital certificates*, 504-505
 - file sharing needs assessment*, 498
 - IKE Phase 1, configuring*, 506-507
 - IKE Phase 1, planning*, 499-500
 - IKE Phase 1, troubleshooting*, 512
 - IKE Phase 2, configuring*, 507-510
 - IKE Phase 2, planning*, 501-502
 - IKE Phase 2, troubleshooting*, 522-525
 - NTP, implementing*, 502-504
 - pinging routers*, 499
 - protocols*, 499
 - router 1 configuration, verifying*, 513-515

- router 2 configuration, verifying, 517-521*
 - source interfaces with associated IP addresses, testing, 515-516*
 - SSL VPNs, compared, 532-533*
 - traffic triggers, testing, 512*
- overview, 426
- remote-access, 427
- routers, 229
- site-to-site, 427
- SSL
 - implementing, 437-438*
 - IPsec VPNs, compared, 532-533*
 - SSL features, 534*
 - TLS, compared, 532-534*
 - types, 534*
- types, 427
 - IPsec, 427*
 - MPLS, 427*
 - SSL, 427*
- user authentication/authorization, 99-100
- vty lines**
 - access class, setting, 87
 - logs, receiving, 104
- vulnerabilities**
 - classifying, 11-12
 - CVE (Common Vulnerabilities and Exposures) database, 12
 - defined, 9-10
 - malicious traffic, 241
 - NVD (National Vulnerability Database), 12
 - SNMP, 129

W

websites

- Cisco Learning Network, 561
- companion, 573
- Premium Edition, 561
- SIO services, 231
- VLAN routing, 182

wildcard masks, 244

- WINS (AnyConnect clients), configuring, 548**

wireless risk assessment, 27

wizards

- ASDM Startup, 346-347
- Basic Firewall
 - CME warning message, 303*
 - DNS, choosing, 305*
 - interface not belonging warning message, 303*
 - interfaces, connecting, 302*
 - security levels, choosing, 304*
 - summary page, 305*
 - untrusted interfaces warning message, 303*
 - welcome screen, 302*
- IPS Policies, 395
- NAT, 319-321
- Security Audit
 - fixing identified potential problems, 82-83*
 - identifying potential problems, 82*
 - interface connections, 82*
 - summary, 83*
- SSL VPN, 535-544
- Step by Step, 476
- VPN AnyConnect, 545

X - Y

X.500/X.509v3 certificates, 449, 460

Z

ZBFs (Zone-Based Firewalls), 294

class maps, 296

components, configuring, 298-300

configuring, 300-313

Basic Firewall wizard welcome screen, 302

CME warning message, 303

DNS, choosing, 305

Firewall wizard page, 301-302

interface not belonging warning message, 303

interfaces, connecting, 302

literal CLI commands, 306-313

security levels, choosing, 304

summary page, 305

untrusted interfaces warning message, 303

features, 294-295

monitoring, 314-315

NAT

configuring with CCP, 319-321

configuring with command line, 322

verifying, 322-323

overview, 294

policy maps, 297

actions, 297

defined, 296

service policies

defined, 297

traffic interaction between zones, 297-298

verifying

CCP, 314-315

command line, 315-319

zones

administrator created, 295

pairs, 295

self, 297-298

traffic interaction between, 298



Memory Tables

Chapter 1

Table 1-2 *Security Terms*

Vocabulary Term	Explanation
Asset	
Vulnerability	
Threat	
Risk	
Countermeasure	

Table 1-5 *Additional Attack Methods*

Method	Description
Covert channel	
Trust exploitation	
Password attacks	
Botnet	
DoS and DDoS	

Chapter 2

Table 2-3 *The Who, What, and Why of Security Policies*

Security Policies	Explanation
Who creates security policies?	
What is in a security policy?	
Why do we have security policies?	

Chapter 3

Table 3-2 *Borderless Network Components*

Component	Explanation
Borderless end zone	
Borderless data center	
Borderless Internet	
Policy management point	

Chapter 4

Table 4-2 *Components of a Threat Control and Mitigation Strategy*

Plane	Security Measures	Protection Objectives
Management plane	<i>Authentication, authorization, accounting (AAA)</i> <i>Authenticated Network Time Protocol (NTP)</i> <i>Secure Shell (SSH)</i> <i>Secure Sockets Layer/Transport Layer Security (SSL/TLS)</i> Protected syslog <i>Simple Network Management Protocol Version 3 (SNMPv3)</i> Parser views	
Control plane	<i>Control plane policing (CoPP) and control plane protection (CPPr)</i> Authenticated routing protocol updates	
Data plane	<i>Access control lists (ACL)</i> Layer 2 controls, such as private VLANs, <i>Spanning Tree Protocol (STP)</i> guards IOS IPS, Zone-Based Firewall	

Table 4-4 *Protecting the Data Plane*

Feature	Explanation
ACLs used for filtering	
IOS firewall support	
IOS IPS	
TCP Intercept	
Unicast Reverse Path Forwarding	

Chapter 5

Table 5-3 *Properties of the Toolbar*

Tool Name	Description
Home button	
Configure button	
Monitor button	
Manage community icon	
Refresh icon	
Provide feedback to Cisco icon	
Help icon	
Search icon	

Chapter 6

Table 6-2 *AAA Components to Secure Administrative and Remote LAN Access*

Access Type Mode	Mode	Where These Are Likely to Be Used	AAA Command Element
Remote administrative access Usually TACACS+ between the router and the ACS	Character (line or EXEC mode)		login, enable, exec
Remote network access end users Usually RADIUS between the router and the ACS	Packet (interface mode) such as an interface with PPP requiring authentication		ppp, network, vpn groups

Table 6-3 *Method List Options*

Command Element	Description
<i>type</i>	
default	
<i>list-name</i>	

Command Element	Description
------------------------	--------------------

method

Chapter 7

Table 7-2 *TACACS+ Versus RADIUS*

	TACACS+	RADIUS
Functionality		Combines many of the functions of authentication and authorization together. Has detailed accounting capability when accounting is configured for use.
Standard		Open standard, and supported by nearly all vendors' AAA implementation.
L4 protocol		UDP.
Replacement coming		Possibly Diameter (named to imply that RADIUS is only half as much, pun intended).
Confidentiality		Only the password is encrypted with regard to packets sent back and forth between the ACS server and the router.
Granular command by command authorization		No explicit command authorization checking rules can be implemented.
Accounting		Provide accounting support, and generally acknowledged as providing more detailed or extensive accounting capability than TACACS+.

Table 7-4 *Key Components for Configuring ACS*

Component of ACS	How It Is Used
Network device groups	
Network devices (ACS clients/routers/switches)	
Identity groups (user/admin groups)	
User accounts	
Authorization profiles	

Chapter 8

Table 8-2 *Tool Kit for L2 Security*

Tool	Description
Port security	
BPDU guard	
Root guard	
Dynamic ARP inspection	
IP source guard	
802.1x	
DHCP snooping	
Storm control	
Access control lists	

Chapter 9

Table 9-3 *Conversion Charts Between Decimal, Binary and Hexadecimal*

Decimal	Binary	Hexadecimal
0	0000	
1	0001	
2	0010	
3	0011	
4	0100	
5	0101	
6	0110	
7	0111	
8	1000	
9	1001	
10	1010	
11	1011	
12	1100	
13	1101	
14	1110	
15	1111	

Chapter 10

Table 10-3 *Security Features on Cisco Switches*

Feature	Description
Port security	
DHCP snooping	
Dynamic Address Resolution Protocol (ARP) inspection	
IP source guard	
Root guard, BPDUs guard, BPDUs filtering	
Storm control	
Additional modules	

Table 10-4 *Security Features of IOS Routers*

Feature	Description
	Reflexive access lists
	<i>Context-based access control (CBAC)</i>
	Zoned-Based Firewall
	Packet-filtering ACLs
	AAA
	VPNs
	IPS
	Routing protocol authentication
	Control plane protection and control plane policing

Feature	Description
Secure management protocols	

Table 10-5 *Security Features of ASA Firewalls*

Feature	Description
Stateful filtering	
<i>Modular Policy Framework (MPF)</i>	
URL filtering	
Packet-filtering ACLs	
AAA	
VPNs	
IPS	
Routing protocol authentication	
Secure management protocols	

Table 10-6 *Other Appliances and Services Used to Implement a Security Policy*

Device or System	Explanation
IPS	
<i>Cisco Security Manager (CSM)</i>	
<i>Cisco Security Intelligence Operations (SIO) Service</i>	

Chapter 11

Table 11-3 *Standard ACLs Versus Extended ACLs*

	Standard ACL	Extended ACL
Numeric range	1–99, 1300–1999.	
Option for using names for the ACL instead of numbers	Yes.	
What they can match on	Source IP only of the packet being compared to the list.	
Where to place	Unfortunately, these need to be placed relatively close to the destination. Applying these access lists too close to the source may limit that source from reaching other destinations that were not intended to be limited.	

Chapter 12

Table 12-5 *Advantages and Disadvantages of Application Layer Gateways*

Advantages	Disadvantages
Very tight control is possible, due to analyzing the traffic all the way to the application layer.	
It is more difficult to implement an attack against an end device because of the proxy server standing between the attacker and potential victim.	
Can provide very detailed logging.	
May be implemented on common hardware.	

Table 12-6 *Advantages and Disadvantages of Stateful Packet-Filtering Devices*

Advantages	Disadvantages
	Might not be able to identify or prevent an application layer attack.
	Not all protocols contain tightly controlled state information, such as <i>User Datagram Protocol (UDP)</i> and <i>Internet Control Message Protocol (ICMP)</i>
	Some applications may dynamically open up new ports from the server, which if a firewall is not analyzing specific applications or prepared for this server to open up a new port, it could cause a failure of that application for the end user. If a firewall also supports application layer inspection, it may be able to predict and allow this inbound connection.
	Stateful technology, by itself, does not support user authentication. This, however, does not prevent a firewall that implements stateful packet filtering from also implementing authentication as an additional feature.

Table 12-8 *NAT Terminology*

NAT Term	Description
Inside local	
Inside global	
Outside local	
Outside global	

Table 12-10 *Firewall Access Rules*

Rule	Description
Rules based on service control	
Rules based on address control	
Rules based on direction control	
Rules based on user control	
Rules based on behavior control	

Chapter 13

Table 13-2 *Policy Map Actions*

Policy Action	Description	When to Use It
Inspect		This should be used on transit traffic initiated by users who expect to get replies from devices on the other side of the firewall.
Pass		Traffic that does not need a reply. Also in the case of protocols that do not support inspection, this policy could be applied to the zone pair for specific outbound traffic, and be applied to a second zone pair for inbound traffic.
Drop		Traffic you do not want to allow between the zones where this policy map is applied.
Log		If you want to see log information about packets that were dropped because of policy, you can add this option.

Table 13-3 *Traffic Interaction Between Zones*

Ingress Interface Member of Zone	Egress Interface Member of Zone	Zone Pair Exists, with Applied Policy	Result
No	No	Does not matter	
No	Yes (any zone)	Does not matter	
Yes (zone A)	Yes (zone A)	Does not matter	
Yes (zone A)	Yes (zone B)	No	
Yes (zone A)	Yes (zone B)	Yes	

Table 13-4 *Self Zone Traffic Behavior*

Source Traffic Member of Zone	Destination Traffic Member of Zone	Zone Pair Exists, with a Policy Applied	Result
Self	Zone A	No	
Zone A	Self	No	
Self	Zone A	Yes	
Zone A	Self	Yes	

Chapter 14

Table 14-2 *ASA Models*

Model	Description
ASA 5505	
ASA 5510	
ASA 5520, 5540, 5550	
ASA 5585	
<i>Firewall Services Module (FWSM)</i> and the ASA Services Module	

Chapter 15

Table 15-2 *IDS Versus IPS*

	IDS	IPS
Position in the network flow	Off to the side, the IDS is sent copies of the original packets.	
Also known as	Promiscuous mode, out of band.	
Latency or delay	Does not add delay to the original traffic because it is not inline.	
Impact caused by the sensor failing to forward packets	There is no negative impact if the sensor goes down.	
Ability to prevent malicious traffic from going into the network	By itself, a promiscuous mode IDS cannot stop the original packet. Options do exist for a sensor in promiscuous mode to request assistance from another device that is inline which may block future packets.	
Normalization ability	Because the IDS does not see the original packet, it cannot manipulate any original inline traffic,	

Table 15-3 *IPS/IDS Method Advantages and Disadvantages*

	Advantages	Disadvantages
Signature based	Easy to configure, simple to implement.	
Policy based	Simple and reliable, very customizable, only allows policy-based traffic that could deny unknown attacks, which by default are outside of the policy being allowed.	
Anomaly based	Self-configuring baselines, detect worms based on anomalies, even if specific signatures have not been created yet for that type of traffic.	
Reputation based	Leverages enterprise and global correlation, providing information based on the experience of other systems. Early-warning system.	

Table 15-5 *Risk Rating (RR) Calculation Factors*

Factor That Influences Risk Rating	Description
	The value that you as an administrator have assigned to specific destination IP addresses or subnets where the critical servers/devices live.
	The accuracy of the signature as determined by the person who created that signature
	How critical the attack is as determined by the person who created that signature

Factor That Influences Risk Rating	Description
	This is a minor contributor to the risk rating. A signature match that is destined to a host where the attack is relevant, such as a Windows server-based attack, which is going to the destination address of a known Windows server, is considered a relevant attack, and the risk rating increases slightly as a result.
	If the sensor is participating in global correlation and receives information about specific source addresses that are being used to implement large-scale attacks, attacks coming from the source IP addresses are also given a slightly increased risk rating value.

Table 15-6 *IPS/IDS Evasion Techniques*

Evasion Method	Description	Cisco Anti-Evasion Techniques
Traffic fragmentation		Complete session reassembly so that the IPS/IDS can see the big picture.
Traffic substitution and insertion		Data normalization and de-obfuscation techniques. Cisco's implementation is looking for Unicode, case sensitivity, substitution of spaces with tabs, and other similar anti-evasion techniques.
Protocol level misinterpretation		IP <i>Time-To-Live (TTL)</i> analysis, TCP checksum validation.
Timing attacks		Configurable intervals and use of third-party correlation

Evasion Method	Description	Cisco Anti-Evasion Techniques
Encryption and tunneling		If traffic is encrypted and passing through the sensor as encrypted data, the encrypted payload cannot be inspected. For <i>generic routing encapsulation (GRE)</i> tunnels, there is support for inspection if the data is not encrypted.
Resource exhaustion		Dynamic and configurable event summarization. Here is an example: 20,000 devices are all under the control of the attacker. All those devices begin to send the same attack. The sensor summarizes those by showing a few of the attacks as alerts, and then summaries at regular intervals that indicate the attack is still in play and how many thousands of times it occurred over the last interval. This is much better than trying to wade through thousands of individual alerts.

Table 15-7 *Micro-Engines (Groupings of Signatures)*

Signature Micro-Engine	Signatures in This Grouping
Atomic	
Service	
String or Multistring	
Other	

Chapter 16

Table 16-3 *Matrix for Retired/Unretired/Enabled/Disabled*

Compiling/ Allowing Action	Enabled	Disabled
Retired	No memory consumption, and no action related to the signature during packet analysis	
Unretired	Consumes memory, and the signature is considered during packet analysis	

Chapter 17

Table 17-2 *VPN Components*

Component	Function	Examples of Use
Symmetrical encryption algorithms	Uses the same key for encrypting and decrypting data.	
Asymmetrical encryption	Uses a public and private key. One key encrypts the data, and the other key in the pair is used to decrypt.	
Digital signature	Encryption of hash using private key, and decryption of hash with the sender's public key.	
Diffie-Hellman key exchange	Uses a public-private key pair asymmetrical algorithm, but creates final shared secrets (keys) that are then used by symmetrical algorithms.	
Confidentiality	Encryption algorithms provide this by turning clear text into cipher text.	
Data integrity	Validates data by comparing hash values.	
Authentication	Verifies the peer's identity to the other peer.	

Chapter 18

Table 18-2 *Key PKI Components*

Component	Description
RSA digital signatures	
Digital certificate	
Public and private keys	
Certificate authority	
X.509v3	
Subordinate CA/RA	
PKCS	

Chapter 19

Table 19-2 *IPsec Goals and the Methods Used to Implement Them*

Goal	Method That Provides the Feature
Confidentiality	
Data integrity	
Peer authentication	
Antireplay	

Chapter 20

Table 20-3 *IKE Phase 1 Policy Options*

Function	Strong Method	Stronger Method
Hashing	MD5, 128-bit	
Authentication	Pre-shared key (PSK)	
Group # for DH key exchange	1,2	
Lifetime	86400 seconds (1 day, default)	
Encryption	3DES	

Table 20-4 *IKE Phase 2 Policy Options*

Item to Plan	Implemented By	Notes
Peer IP addresses		Having a known reachable IP address for the VPN peer is critical for the traditional IPsec site-to-site tunnel to negotiate and establish the VPN (both phases).
Traffic to encrypt		Extended ACL that is not applied to an interface but is referenced in the crypto map. This should <i>only</i> reference outbound (egress) traffic, which should be protected by IPsec. Traffic not matching the crypto ACL will not be encrypted, but will be sent as a normal packet.
Encryption method		DES, 3DES, AES are all options. IKE phase 2 does not need to be the same method as Phase 1. The method does need to match the peer's policy (transform sets) for Phase 2.
Hashing (HMAC) method		MD5 and SHA HMACs may be used, and need to match the Phase 2 policy of the peer.

Item to Plan	Implemented By	Notes
Lifetime (time, or data)		Lifetime for Phase 2 should match between the peers. If both use the default lifetime (by not specifying a lifetime), both peers would have compatible lifetime policies. The lifetime can be specified as number of seconds or number of kilobytes.
<i>Perfect Forward Secrecy (PFS)</i> (run DH again or not)		DH is run during IKE Phase 1, and Phase 2 reuses that same keying material that was generated. If you want Phase 2 to rerun the DH, it is called Perfect Forward Secrecy (PFS), and you must choose a DH group number 1,2 or 5 for Phase 2 to use.
Which interface used to peer with the other VPN device		From a routing perspective, this is the interface of a VPN peer that is closest to the other peer, where outbound IPsec packets are leaving the router and inbound IPsec packets are coming into the router.

Chapter 21

Table 21-3 *Comparison Between SSL and TLS*

SSL	TLS
Developed by Netscape in the 1990s	
Starts with a secured channel and continues directly to security negotiations on a dedicated port	
Widely supported on client-side applications	
More weaknesses identified in older SSL versions	

Table 21-4 *Options for SSL VPN Implementation*

	Clientless SSL VPN	Clientless SSL VPN with Plug-Ins for Some Port Forwarding	Full AnyConnect SSL VPN Client
Other names	Web VPN.	Thin client.	
Installed software on client	No client required.	Small applets and/or configuration required.	
User experience	Feels like accessing resources (that are on the corporate network) through a specific browser window or hyperlink.	Some applications can be run locally with output redirected through the VPN. Includes the features of the clientless VPN to the left.	
Servers that can be used	IOS with the correct software, and ASA with the correct licenses.	IOS with the correct software, and ASA with the correct licenses.	
How the user looks from the corporate network	Traffic is proxied (<i>Port Address Translation [PAT]</i>) by the SSL server, as the users packets enter the corporate network.	Traffic is proxied (<i>Port Address Translation [PAT]</i>) by the SSL server as the users packets enter the corporate network.	
Clients supported	Most SSL-capable computers.	Computers that support SSL and Java.	



Memory Tables Answer Key

Chapter 1

Table 1-2 *Security Terms*

Vocabulary Term	Explanation
Asset	An asset is an item that is to be protected and can include property, people, and information/data that have value to the company. This includes intangible items such as proprietary information or trade secrets and the reputation of the company. The data could include company records, client information, proprietary software, and so on.
Vulnerability	A vulnerability is an exploitable weakness of some type. That exploitation might result from a malicious attack, or it might be accidentally triggered because of a failure or weakness in the policy, implementation, or software running on the network.
Threat	<p>This is what you are protecting against. A threat is anything that attempts to gain unauthorized access to, compromise, destroy, or damage an asset. Threats are often realized via an attack or exploit that takes advantage of an existing vulnerability.</p> <p>Threats today come in many varieties and spread more rapidly than ever before. Threats can also morph and be modified over time, and so you must be ever diligent to keep up with them.</p>
Risk	Risk is the <i>potential</i> for unauthorized access to, compromise, destruction, or damage to an asset. If a threat exists, but proper countermeasures and protections are in place (it is your goal to provide this protection), the potential for the threat to be successful is reduced (thus reducing the overall risk).
Countermeasure	A countermeasure is a device or process (a safeguard) that is implemented to counteract a potential threat, which thus reduces risk.

Table 1-5 *Additional Attack Methods*

Method	Description
Covert channel	<p>This method uses programs or communications in unintended ways. For example, if the security policy says that web traffic is allowed but peer-to-peer messaging is not, users can attempt to tunnel their peer-to-peer traffic inside of HTTP traffic. An attacker may use a similar technique to hide traffic by tunneling it inside of some other allowed protocol to avoid detection. An example of this is a backdoor application collecting keystroke information from the workstation and then slowly sending it out disguised as <i>Internet Control Message Protocol (ICMP)</i>. This is a covert channel.</p> <p>An overt channel is the legitimate use of a protocol, such as a user with a web browser using HTTP to access a web server.</p>
Trust exploitation	<p>If the firewall has three interfaces, and the outside interface allows all traffic to the <i>demilitarized zone (DMZ)</i>, but not to the inside network, and the DMZ allows access to the inside network from the DMZ, an attacker could leverage that by gaining access to the DMZ and using that location to launch his attacks from there to the inside network. Other trust models, if incorrectly configured, may allow unintentional access to an attacker including active directory and <i>NFS (network file system)</i> in UNIX).</p>
Password attacks	<p>These could be brute force, where the attacker's system attempts thousands of possible passwords looking for the right match. This is best protected against by specifying limits on how many unsuccessful authentication attempts may occur within a specified time frame. Password attacks can also be done through malware, man-in-the-middle attacks using packet sniffers, or by using key loggers.</p>
Botnet	<p>A botnet is a collection of infected computers that are ready to take instructions from the attacker. For example, if the attacker has the malicious backdoor software installed on 10,000 computers, from his central location he could instruct those computers to all send TCP SYN requests or ICMP echo requests repeatedly to the same destination. To add insult to injury, he could also spoof the source IP address of the request so that reply traffic is sent to yet another victim. A covert channel is generally used by the attacker to manage the individual devices that make up the botnet.</p>
DoS and DDoS	<p>Denial-of-service attack and distributed denial-of-service attack. An example is using a botnet to attack a target system. If an attack is launched from a single device with the intent to cause damage to an asset, the attack could be considered a DoS attempt, as opposed to a DDoS. Both types of attacks want the same result, and it just depends on how many source machines are used in the attack as to whether it is called a DoS or DDoS.</p>

Chapter 2

Table 2-3 *The Who, What, and Why of Security Policies*

Security Policies	Explanation
Who creates security policies?	<p>The executive senior management team is ultimately responsible for the data and the networks that carry the data for their company. From a technician's perspective, this might seem a bit odd that the senior management team is creating a security policy, but that is who specifies the overall goals of the policy. The high-level security policy is often referred to as a <i>governing policy</i>.</p> <p>It is up to the management teams and staff who have the skills to implement the appropriate controls (which include physical, logical, and administrative controls). At this level, we often use technical policies to implement the security responsibilities based on the roles the staff are filling.</p> <p>It is up to the end users to agree to and abide by the policies set forth by the company. This is referred to as an <i>end-user policy</i>, which is sometimes called <i>acceptable use policy (AUP)</i>.</p> <p>Policies may also apply to individuals outside of the company, including customers, suppliers, contractors, and so on.</p>
What is in a security policy?	<p>In a security policy, a primary aspect is risk management. In that light, it could include items such as access controls, backups, virus protection, incident handling, encryption, monitoring, password requirements, disposing of resources, inspections and reviews, personal/physical security, system-configured change process, auditing, security awareness and training, documentation, AUP (and the list goes on).</p> <p>A security policy should begin with a general overview about why the policy was written and what it covers and what it does not cover. This is often referred to as the <i>scope of the policy</i>.</p>
Why do we have security policies?	<p>Besides risk management, security policies are also used to educate users, staff, and other workers about what the policy of the company is. It can also be used to establish a baseline for which security measures must be implemented to protect assets. Without a security policy in place, the risk (which is a factor of assets that are vulnerable being attacked and resulting in a loss) is too great.</p>

Chapter 3

Table 3-2 *Borderless Network Components*

Component	Explanation
Borderless end zone	This is where devices connect to the network. It is here that we are concerned with viruses, malware, and other malicious software. Using techniques such as <i>Network Admissions Control (NAC)</i> and <i>Identity Services Engine (ISE)</i> , we can properly interrogate devices before they are allowed onto the network to verify they meet certain minimum requirements (installations of virus scanning tools, service packs, patch revision levels, and so on).
Borderless data center	This represents a cloud-driven business environment that could provide services. It is in this borderless data center where we implement firewalls such as the <i>Adaptive Security Appliance (ASA)</i> and <i>intrusion prevention systems (IPS)</i> to protect network resources there. Virtual tools can also be used inside virtual environments in the data center, such as virtual switches that can enforce policy on virtual devices that are connected to that virtual switch.
Borderless Internet	This represents the biggest IP network on the planet, which we are all familiar with. Service providers and other individuals connected to the Internet use various techniques for security, including IPSs, firewalls, and protocol inspection (all the way from Layer 2 to Layer 7 of the OSI model).
Policy management point	In a perfect environment, we would have a single point of control that could implement appropriate security measures across the entire network. <i>Cisco Security Manager (CSM)</i> is an example of one of these enterprise tools. Another example is <i>Cisco Access Control Server (ACS)</i> , which provides contextual access. For example, if you want to allow administrators full access to a router only if they are logging in from a specific location, you could enforce that with ACS and <i>authentication, authorization, accounting (AAA)</i> rules. Under that same system, administrators could also potentially gain access from other locations.

Chapter 4

Table 4-2 *Components of a Threat Control and Mitigation Strategy*

Plane	Security Measures	Protection Objectives
Management plane	<p><i>Authentication, authorization, accounting (AAA)</i></p> <p><i>Authenticated Network Time Protocol (NTP)</i></p> <p><i>Secure Shell (SSH)</i></p> <p><i>Secure Sockets Layer/Transport Layer Security (SSL/TLS)</i></p> <p>Protected syslog</p> <p><i>Simple Network Management Protocol Version 3 (SNMPv3)</i></p> <p>Parser views</p>	<p>Authenticate and authorize any administrators. Protect time synchronization by using authenticated NTP. Use only encrypted remote-access protocols, such as SSH for CLI and SSL/TLS for GUI tools, and use secure versions of SNMP. If plaintext tools are used (such as syslog or Telnet), they should be protected by encryption protocols such as IPsec or should be used out of band (a separate network just for management traffic). A parser “view” is a way to limit what a specific individual, based on his role, can do on the router.</p>
Control plane	<p><i>Control plane policing (CoPP) and control plane protection (CPPr)</i></p> <p>Authenticated routing protocol updates</p>	<p>The control plane tools can be implemented to limit the damage an attacker can attempt to implement directly at the router’s IP address (traffic addressed directly to the router, which the router must spend CPU resources to process).</p> <p>Routing protocol updates should be authenticated to remove the possibility of an attacker manipulating routing tables by putting a rogue router running the same routing protocol on your network. The attacker could be doing reconnaissance to learn the routes, or the attacker could be attempting to manipulate the resulting data plane by changing the routing on the network.</p>
Data plane	<p><i>Access control lists (ACL)</i></p> <p>Layer 2 controls, such as private VLANs, <i>Spanning Tree Protocol (STP)</i> guards</p> <p>IOS IPS, Zone-Based Firewall</p>	<p>ACLs, when applied as filters on interfaces, can control which traffic (transit traffic) is allowed on the data plane.</p> <p>At Layer 2, by protecting the infrastructure there, you can avoid a rogue switch from becoming the root of your spanning tree, which would affect the data plane at Layer 2.</p> <p>Firewall filtering and services can also control exactly what traffic is flowing through your network. An example is using an IOS Zone-Based Firewall to implement policy about the data plane and what is allowed.</p>

Table 4-4 *Protecting the Data Plane*

Feature	Explanation
ACLs used for filtering	<p>There are many types of ACLs and many ways to apply them for filtering.</p> <p>Note that an ACL can be used as a classification mechanism used in other features, such as an IOS firewall, identifying traffic for control plane protection, identifying who is allowed to connect to a vty line, where SNMP is allowed, and so on. In the discussion of protecting the data plane, we focus primarily on ACLs applied directly to interfaces for the purpose of filtering.</p>
IOS firewall support	<p>The firewall features on an IOS router have grown over the years. The older technology for implementing a firewall on IOS routers was called <i>context-based access control (CBAC)</i>. CBAC has been replaced with the more current <i>Zone-Based Firewall</i> on the IOS.</p>
IOS IPS	<p>IOS IPS is a software implementation of an <i>intrusion prevention system (IPS)</i> that is overlaid on top of the existing routing platform, to provide additional security. IOS IPS uses signature matches to look for malicious traffic. When an alert goes off because of a signature match, the router can prevent the packet from being forwarded, thus preventing the attack from reaching the final destination.</p>
TCP Intercept	<p>This tool allows the router to look at the number of half-formed sessions that are in place and intervene on behalf of the destination device. This can protect against a destination device from a SYN-flood attack that is occurring on your network. The <i>Zone-Based Firewall</i> on an IOS router includes this feature.</p>
Unicast Reverse Path Forwarding	<p><i>Unicast Reverse Path Forwarding (uRPF)</i> can mitigate spoofed IP packets. When this feature is enabled on an interface, as packets enter that interface the router spends an extra moment considering the source address of the packet. It then considers its own routing table, and if the routing table does not agree that the interface that just received this packet is also the best egress interface to use for forwarding to the source address of the packet, it then denies the packet.</p> <p>This is a good way to limit IP spoofing.</p>

Chapter 5

Table 5-3 *Properties of the Toolbar*

Tool Name	Description
Home button	Click this button to display what is called the Community View page. This information summarizes the community information and allows you to add, edit, and even discover new devices. You can also use the Home button to see the device status of each device.
Configure button	If you want to make a change to the configuration or view the existing configuration of the router, you use this Configure button to get to the correct area. From the drop-down list, you can make sure you are configuring the correct router based on its IP address, and then using features selected from the navigation pane on the left, configure the specific elements of the router you want to view or change. Not all features are available for configuration. For example, if a feature such as voice is not supported on a device, that feature is not displayed as a configurable option. Another reason that some of the options may not be configurable is because of the individual who is logged in. With <i>role-based access control (RBAC)</i> , not every user has to be given full access to configure everything. You can restrict what the administrator can see or configure by using user profiles, as covered later in this chapter.
Monitor button	This button displays the router and security features that you can monitor on a specific router. A list of items that can be selected for monitoring is presented in the left navigation pane.
Manage community icon	If you want to view or edit your existing communities, or create a new one, clicking this icon provides those options. From the Manage Community pop-up window, you can also request CCP to “discover” those routers, which means it will log in to them and read the running configuration.
Refresh icon	Clicking the refresh icon instructs CCP to reach out and request the current running configuration from the specified device. This is especially important if changes have been made at the command line of the router after CCP discovered the device. This refresh allows CCP to correctly display the configured settings, including those that were done at the command line, outside of CCP.
Provide feedback to Cisco icon	This icon opens the CCP feedback form, which you can use to send feedback about this product to Cisco Systems.
Help icon	The help icon, which looks like a question mark, opens context-sensitive help that is relevant for the active window.
Search icon	The search feature opens up a new browser window and enables you to search the help documents based on a keyword.

Chapter 6

Table 6-2 AAA Components to Secure Administrative and Remote LAN Access

Access Type Mode	Mode	Where These Are Likely to Be Used	AAA Command Element
Remote administrative access Usually TACACS+ between the router and the ACS	Character (line or EXEC mode)	Lines: vty, AUX console, and tty	login, enable, exec
Remote network access end users Usually RADIUS between the router and the ACS	Packet (interface mode) such as an interface with PPP requiring authentication	Interfaces: async, group-async, BRI, PRI, Other functionality: VPN user authentication	ppp, network, vpn groups

Table 6-3 Method List Options

Command Element	Description
<i>type</i>	Identifies the type of list being created. Relevant options are authentication , authorization , or accounting .
default	Specifies the default list of methods to be used based on the methods that follow this argument. If you use the keyword default , a custom name is not used.
<i>list-name</i>	Used to create a custom method list. This is the name of this list, and is used when this list is applied to a line, such as to vty lines 0–4.

Command Element	Description
<i>method</i>	<p>At least one method must be specified. To use the local user database, use the local keyword. A single list can contain up to 4 methods, which are tried in order, from left to right.</p> <p>In the case of an authentication method list, methods include the following:</p> <p>enable: The enable password is used for authentication. This might be an excellent choice as the last method in a method list. This way, if the previous methods are not available (such as the AAA server, which might be down or not configured), the router times out on the first methods and eventually prompts the user for the enable secret as a last resort.</p> <p>krb5: Kerberos 5 is used for authentication.</p> <p>krb5-telnet: Kerberos 5 Telnet authentication protocol is used when using Telnet to connect to the router.</p> <p>line: The line password (the one configured with the password command, on the individual line) is used for authentication.</p> <p>local: The local username database (running config) is used for authentication.</p> <p>local-case: Requires case-sensitive local username authentication.</p> <p>none: No authentication is used.</p> <p>group radius: A RADIUS server (or servers) is used for authentication.</p> <p>group tacacs+: A TACACS+ server (or servers) is used for authentication.</p> <p>group <i>group-name</i>: Uses either a subset of RADIUS or TACACS+ servers for authentication as defined by the aaa group server radius or aaa group server tacacs+ command.</p>

Chapter 7

Table 7-2 TACACS+ Versus RADIUS

	TACACS+	RADIUS
Functionality	Separates AAA functions into distinct elements. Authentication is separate from authorization, and both of those are separate from accounting.	Combines many of the functions of authentication and authorization together. Has detailed accounting capability when accounting is configured for use.
Standard	Cisco proprietary, but very well known.	Open standard, and supported by nearly all vendors' AAA implementation.
L4 protocol	TCP.	UDP.
Replacement coming	None officially planned.	Possibly Diameter (named to imply that RADIUS is only half as much, pun intended).
Confidentiality	<i>All</i> packets are encrypted between ACS server and the router (which is the client).	Only the password is encrypted with regard to packets sent back and forth between the ACS server and the router.
Granular command by command authorization	This is supported, and the rules are defined on the ACS server about which commands are allowed or disallowed.	No explicit command authorization checking rules can be implemented.
Accounting	Provides accounting support.	Provide accounting support, and generally acknowledged as providing more detailed or extensive accounting capability than TACACS+.

Table 7-4 *Key Components for Configuring ACS*

Component of ACS	How It Is Used
Network device groups	Groups of network devices, normally based on routers or switches with similar functions/devices managed by the same administrators.
Network devices (ACS clients/routers/switches)	The individual network devices that go into the device groups.
Identity groups (user/admin groups)	Groups of administrators, normally based on users who will need similar rights and access to specific groups of network devices.
User accounts	Individual administrator/user accounts that are place in Identity groups.
Authorization profiles	These profiles control what rights are permitted. The profile is associated with a network device group and a user/administrator identity group.

Chapter 8

Table 8-2 *Tool Kit for L2 Security*

Tool	Description
Port security	Limits the number of MAC addresses to be learned on an access switch port.
BPDU guard	If BPDUs show up where they should not, the switch protects itself.
Root guard	Control which ports are not allowed to become root ports to remote root switches.
Dynamic ARP inspection	Prevents spoofing of Layer 2 information by hosts.
IP source guard	Prevents spoofing of Layer 3 information by hosts.
802.1x	Authenticates users before allowing their data frames into the network.
DHCP snooping	Prevents rogue DHCP servers from impacting the network.
Storm control	Limits the amount of broadcast or multicast traffic flowing through the switch.
Access control lists	Traffic control to enforce policy. Access control is covered in another chapter.

Chapter 9

Table 9-3 *Conversion Charts Between Decimal, Binary and Hexadecimal*

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Chapter 10

Table 10-3 *Security Features on Cisco Switches*

Feature	Description
Port security	Limits the number of MAC addresses that a port can learn. This protects against a CAM <i>content-addressable memory</i> (CAM) (also known as the <i>MAC table</i>) overflow. An attacker may attempt to flood bogus source MAC addresses in an attempt to consume all the memory in the table, which would cause the switch to forward unicast frames out all ports in the same VLAN. By launching this attack, the attacker is hoping to see all frames on the VLAN and perform an eavesdropping reconnaissance against the network.
DHCP snooping	An attacker who attempts to place a rogue DHCP server on the network could potentially hand out incorrect <i>Dynamic Host Configuration Protocol</i> (DHCP) information, including the default gateway for the clients to use, which could cause a man-in-the-middle attack and allow eavesdropping by the attacker. DHCP snooping only allows server responses from specifically trusted ports that lead to your authorized DHCP servers. This also protects the DHCP server by rate-limiting how many TCP requests can be sent per interval. This is useful if somewhere an attacker is requesting thousands of IP addresses in an attempt to consume the entire pool on the DHCP server.
Dynamic Address Resolution Protocol (ARP) inspection	Using the information from DHCP snooping or from manually configuring it, a switch can confirm that your traffic includes accurate MAC address information in ARP communications, to protect against an attacker trying to perform Layer 2 spoofing.
IP source guard	This can be used to verify the client on a given port is not doing Layer 3 spoofing (IP address spoofing).
Root guard, BPDU guard, BPDU filtering	These features enable you to control your spanning-tree topology, including resisting a rogue switch's attempt to become root of the spanning tree.
Storm control	This feature allows the switch to begin clamping on traffic at configurable levels. For example, broadcast storm control could tell the switch to stop forwarding broadcast traffic (or limit it) if broadcasts ever reach more than 50 percent utilization (for example) of the switch capacity.
Additional modules	Modules are supported on various networking devices, which add functionality to that device. Examples include IPS modules, VPN modules, firewall modules, anti-malware modules, and so on. You can expand security services on many network devices, such as routers, switches, and even add on to the functionality of firewalls.

Table 10-4 *Security Features of IOS Routers*

Feature	Description
Reflexive access lists	This is mostly for historical purposes, but this was one of the early attempts on Cisco IOS to perform stateful filtering. We discuss stateful filtering in detail in the firewall chapters, but the concept is to not allow any traffic in from the outside world (if it is initiated from the outside). If a user on the inside of your network sends traffic out to a server on the outside network, the reflexive access lists look at that flow of traffic, creates an <i>access control entry (ACE)</i> , which is the mirror image (swapping the source and destination IP addresses and ports), and dynamically applies that so that the return traffic from the server is allowed. Reflexive access lists are not used much anymore.
<i>Context-based access control (CBAC)</i>	This was the evolution of the IOS router to now support stateful filtering, without creating reflexive access lists. This used to be called the IOS Firewall, because CBAC was the primary feature of the IOS Firewall feature set.
Zoned-Based Firewall	This replaced CBAC, and is the current recommended way to implement stateful filtering on IOS routers. An entire chapter in the book covers this topic. Zone-Based Firewalls use class maps to identify traffic, policy maps to specify actions to take on that traffic, and a service policy set of commands to put the policy in place. Among other things, a Zone-Based Firewall can do application layer inspection and URL filtering and has other security-related features.
Packet-filtering ACLs	Using standard and extended ACLs, you can implement your policy of what traffic is allowed or denied through the interfaces of the router.
AAA	AAA stands for <i>authentication, authorization, and accounting</i> . The IOS router has extensive support for each of these features and to work with external servers relevant to these features if desired.
VPNs	IOS supports remote-access VPNs using <i>Secure Sockets Layer (SSL)</i> or IPsec. It also supports VPNs in a site-to-site configuration when using IPsec. (SSL is not generally used for site-to-site VPNs.)
IPS	The IOS router can implement an <i>intrusion prevention system (IPS)</i> in software or by using a hardware module in an available option slot. With an IPS function on the router, you can leverage the added security that the routing function currently provides.
Routing protocol authentication	This provides security that prevents an unauthorized router from being trusted or believed as it sends routing updates with an attempt to influence or learn the routing information from another router.
Control plane protection and control plane policing	This enables you to set thresholds and limits for traffic that is directed to the router. In an attempt to overwhelm the router, an attacker might send thousands of packets directly to the router, which by default would have to be processed by the router itself (as opposed to forwarding the packet somewhere else as in the case of the transit packet). The protection and policing set limits on these packets so that CPU can be preserved.

Feature	Description
Secure management protocols	<i>Secure Shell (SSH)</i> and SSL are supported for managing the router.

Table 10-5 *Security Features of ASA Firewalls*

Feature	Description
Stateful filtering	This allows the ASA to remember the state of a connection (for example, a client going out to a web server) and dynamically allow the return traffic back to the client. The firewall can be implemented as a Layer 2 or Layer 3 device and in either case can analyze traffic all the way up to the application layer.
<i>Modular Policy Framework (MPF)</i>	Used by the ASA (via class maps, policy maps, and service policy rules) to perform simple protocol and application layer inspection and policy enforcement.
URL filtering	Working with statically configured URLs or with a third-party system, the ASA can control which URLs are allowed to be accessed by users through this firewall.
Packet-filtering ACLs	Using standard and extended ACLs, you can implement your policy of what traffic is allowed or denied through the interfaces of the router.
AAA	AAA stands for <i>authentication, authorization, and accounting</i> . The ASA has extensive support for each of these features and can work with external servers related to these features (such as an <i>Access Control Server [ACS]</i> server).
VPNs	ASA supports remote-access VPNs using SSL or IPsec. It also supports VPNs in a site-to-site configuration when using IPsec. (SSL is not generally used for site-to-site VPNs.)
IPS	The ASA can implement an IPS by adding a hardware module to an available option slot on the ASA.
Routing protocol authentication	This provides security that prevents a rogue router from being trusted or believed as it sends routing updates with an attempt to influence or learn the routing information from another router.
Secure management protocols	SSH and SSL are supported for managing the ASA.

Table 10-6 *Other Appliances and Services Used to Implement a Security Policy*

Device or System	Explanation
IPS	An IPS analyzes network traffic, can report on traffic that it deems malicious or harmful, and can take countermeasures against the offending traffic. This can be implemented as an appliance, as a blade in a 6500 switch, or as a module in an ASA or IOS router. The primary method for identifying problem traffic is through signature matching.
<i>Cisco Security Manager (CSM)</i>	This is an enterprise-level configuration tool that you can use to manage most security devices.
<i>Cisco Security Intelligence Operations (SIO) Service</i>	The SIO researches and analyzes threats and provides real-time updates and best practices related to these threats. They can dynamically deliver the latest breaking news right when it happens. There is also an application for smart phones. You can learn more about http://www.cisco.com/go/sio .

Chapter 11

Table 11-3 *Standard ACLs Versus Extended ACLs*

	Standard ACL	Extended ACL
Numeric range	1–99, 1300–1999.	100–199, 2000–2699.
Option for using names for the ACL instead of numbers	Yes.	Yes.
What they can match on	Source IP only of the packet being compared to the list.	Source or destination IP, plus most Layer 4 protocols, including items in the Layer 4 header of the packet being compared.
Where to place	Unfortunately, these need to be placed relatively close to the destination. Applying these access lists too close to the source may limit that source from reaching other destinations that were not intended to be limited.	Because the extended ACL has the granularity of matching on specific source and destination, you can place these very close to the source of the host who is generating the packet, because it will only deny the traffic to the specific destination and will not cause a loss of service to other destinations that are still being permitted.

Chapter 12

Table 12-5 *Advantages and Disadvantages of Application Layer Gateways*

Advantages	Disadvantages
Very tight control is possible, due to analyzing the traffic all the way to the application layer.	Is processor intensive because most of the work is done via software on the proxy server.
It is more difficult to implement an attack against an end device because of the proxy server standing between the attacker and potential victim.	Not all applications are supported, and in practice it might support a specific few applications.
Can provide very detailed logging.	Special client software may be required.
May be implemented on common hardware.	Memory and disk intensive at the proxy server. Could potentially be a single point of failure in the network, unless fault tolerance is also configured.

Table 12-6 *Advantages and Disadvantages of Stateful Packet-Filtering Devices*

Advantages	Disadvantages
Can be used as a primary means of defense by filtering unwanted or unexpected traffic	Might not be able to identify or prevent an application layer attack.
Can be implemented on routers and dedicated firewalls	Not all protocols contain tightly controlled state information, such as <i>User Datagram Protocol (UDP)</i> and <i>Internet Control Message Protocol (ICMP)</i>
Dynamic in nature compared to static packet filtering	Some applications may dynamically open up new ports from the server, which if a firewall is not analyzing specific applications or prepared for this server to open up a new port, it could cause a failure of that application for the end user. If a firewall also supports application layer inspection, it may be able to predict and allow this inbound connection.
Provides a defense against spoofing and <i>denial-of-service (DoS)</i> attacks	Stateful technology, by itself, does not support user authentication. This, however, does not prevent a firewall that implements stateful packet filtering from also implementing authentication as an additional feature.

Table 12-8 *NAT Terminology*

NAT Term	Description
Inside local	The real IP configured on an inside host, such as PC1.
Inside global	The mapped/global address that the router is swapping out for the inside host during NAT. The outside world sees PC1 coming from this mapped/global address.
Outside local	If performing NAT on outside devices (outside NAT), this is the mapped address of the outside device (such as Server A) as it would appear to inside hosts. If not doing outside NAT on the router, this appears as the normal outside device's IP address to the inside devices.
Outside global	The real IP configured on an outside host, such as the IP on Server A.

Table 12-10 *Firewall Access Rules*

Rule	Description
Rules based on service control	These rules are based on the types of services that may be accessed through the firewall, inbound or outbound. An example is that access to web servers, both HTTP or HTTPS, is allowed while all other types of traffic are denied.
Rules based on address control	These rules are based on the source/destination addresses involved, usually with a permit or deny based on specific entries in an access control list.
Rules based on direction control	These rules specify where the initial traffic can flow. For example, a rule might say that traffic from the inside going to the outside (which we could also call outbound traffic) is permitted. Traffic initiated from the outside going to inside resources (which we could call inbound traffic) would be denied. Note that stateful filtering, with its stateful database, could dynamically allow the return traffic back to the inside users. These types of rules could very easily be combined (and usually are) with various protocols/services (such as HTTP, HTTPS, and so on).
Rules based on user control	These rules control access based on knowing who the user is and what that user is authorized to do. This can be implemented via AAA services.
Rules based on behavior control	These rules control how a particular service is used. For example, a firewall may implement an email filter to protect against spam.

Chapter 13

Table 13-2 *Policy Map Actions*

Policy Action	Description	When to Use It
Inspect	Permit and statefully inspect the traffic	This should be used on transit traffic initiated by users who expect to get replies from devices on the other side of the firewall.
Pass	Permits/allows the traffic but does not create an entry in the stateful database	Traffic that does not need a reply. Also in the case of protocols that do not support inspection, this policy could be applied to the zone pair for specific outbound traffic, and be applied to a second zone pair for inbound traffic.
Drop	Deny the packet	Traffic you do not want to allow between the zones where this policy map is applied.
Log	Log the packets	If you want to see log information about packets that were dropped because of policy, you can add this option.

Table 13-3 *Traffic Interaction Between Zones*

Ingress Interface Member of Zone	Egress Interface Member of Zone	Zone Pair Exists, with Applied Policy	Result
No	No	Does not matter	Traffic is forwarded.
No	Yes (any zone)	Does not matter	Traffic is dropped.
Yes (zone A)	Yes (zone A)	Does not matter	Traffic is forwarded.
Yes (zone A)	Yes (zone B)	No	Traffic is dropped.
Yes (zone A)	Yes (zone B)	Yes	Policy is applied. If policy is inspect or pass, the initial traffic is forwarded. If the policy is drop, the initial traffic is dropped.

Table 13-4 *Self Zone Traffic Behavior*

Source Traffic Member of Zone	Destination Traffic Member of Zone	Zone Pair Exists, with a Policy Applied	Result
Self	Zone A	No	Traffic is passed.
Zone A	Self	No	Traffic is passed.
Self	Zone A	Yes	Policy is applied.
Zone A	Self	Yes	Policy is applied.

Chapter 14

Table 14-2 *ASA Models*

Model	Description
ASA 5505	This is the entry-level device. It is relatively small compared to the other appliances, and is not large enough (that is, not wide enough) to be rack mounted in a 19-inch-wide rack. It comes with a built-in switch that has 8 ports, and 2 of those provide support for Power over Ethernet. By default, all the interfaces on the switch port belong to VLAN 1, and the method used to connect this device to multiple networks is to assign the switch ports to at least 2 separate VLANs and then create <i>switched virtual interfaces (SVI)</i> , which are logical Layer 3 interfaces just like on a management interface for a switch, for each logical Layer 3 interface you want the ASA to use. This is the only ASA 55xx series appliance with a built-in switch and with this behavior. This device has a single slot allowing the addition of a compatible module.
ASA 5510	This firewall has 4 built-in routable interfaces, and a management Ethernet interface that can be used as a dedicated interface for management only or can be converted to be a fifth routable interface on the ASA. This firewall has an option slot that supports a compatible module, such as an <i>intrusion prevention system (IPS)</i> module, which is like having an IPS appliance (if installed) that lives inside the ASA.
ASA 5520, 5540, 5550	These firewalls are like the 5510, with the exception that they have more capacity.
ASA 5585	High-performance, high-capacity firewall devices that support multiple add-ons, such as modules compatible with these appliances. These appliances take a more vertical space in a rack compared the 5510 to 5550.
<i>Firewall Services Module (FWSM) and the ASA Services Module</i>	These are blade firewalls that fit into a compatible switch, such as a 6500. They support many of the same features of the standalone ASA appliances in the 55xx family.

Chapter 15

Table 15-2 *IDS Versus IPS*

	IDS	IPS
Position in the network flow	Off to the side, the IDS is sent copies of the original packets.	Directly inline with the flow of network traffic and touches every packet on its way through the network.
Also known as	Promiscuous mode, out of band.	Inline mode.
Latency or delay	Does not add delay to the original traffic because it is not inline.	Adds a small amount of delay before forwarding it through the network.
Impact caused by the sensor failing to forward packets	There is no negative impact if the sensor goes down.	If the sensor goes down, traffic that would normally flow through the sensor could be impacted.
Ability to prevent malicious traffic from going into the network	By itself, a promiscuous mode IDS cannot stop the original packet. Options do exist for a sensor in promiscuous mode to request assistance from another device that is inline which may block future packets.	The IPS can drop the packet on its own because it is inline. The IPS can also request assistance from another device to block future packets just as the IDS does.
Normalization ability	Because the IDS does not see the original packet, it cannot manipulate any original inline traffic.	Because the IPS is inline, it can normalize (manipulate or modify) traffic inline based on a current set of rules.

Table 15-3 *IPS/IDS Method Advantages and Disadvantages*

	Advantages	Disadvantages
Signature based	Easy to configure, simple to implement.	Does not detect attacks outside of the rules. May need to disable signatures that are creating false positives. Signatures must be updated periodically to be current.
Policy based	Simple and reliable, very customizable, only allows policy-based traffic that could deny unknown attacks, which by default are outside of the policy being allowed.	Policy must be manually created. Implementation of the policy is only as good as the signatures you manually create.
Anomaly based	Self-configuring baselines, detect worms based on anomalies, even if specific signatures have not been created yet for that type of traffic.	Difficult to accurately profile extremely large networks. May cause false positives based on significant changes in valid network traffic.
Reputation based	Leverages enterprise and global correlation, providing information based on the experience of other systems. Early-warning system.	Requires timely updates, and requires participation in the correlation process.

Table 15-5 *Risk Rating (RR) Calculation Factors*

Factor That Influences Risk Rating	Description
Target value rating (TVR)	The value that you as an administrator have assigned to specific destination IP addresses or subnets where the critical servers/devices live.
Signature fidelity rating (SFR)	The accuracy of the signature as determined by the person who created that signature.
Attack severity rating (ASR)	How critical the attack is as determined by the person who created that signature.
Attack relevancy (AR)	This is a minor contributor to the risk rating. A signature match that is destined to a host where the attack is relevant, such as a Windows server-based attack, which is going to the destination address of a known Windows server, is considered a relevant attack, and the risk rating increases slightly as a result.
Global correlation	If the sensor is participating in global correlation and receives information about specific source addresses that are being used to implement large-scale attacks, attacks coming from the source IP addresses are also given a slightly increased risk rating value.

Table 15-6 *IPS/IDS Evasion Techniques*

Evasion Method	Description	Cisco Anti-Evasion Techniques
Traffic fragmentation	The attacker splits malicious traffic into multiple parts with the intent that any detection system will not see the attack for what it really is.	Complete session reassembly so that the IPS/IDS can see the big picture.
Traffic substitution and insertion	The attacker substitutes characters in the data using different formats that have the same final meaning. An example is Unicode strings, which an end station could interpret but perhaps a lesser IPS/IDS might not.	Data normalization and de-obfuscation techniques. Cisco's implementation is looking for Unicode, case sensitivity, substitution of spaces with tabs, and other similar anti-evasion techniques.
Protocol level misinterpretation	An attacker may attempt to cause a sensor to misinterpret the end-to-end meaning of a network protocol and so perhaps not catch an attack in progress.	IP <i>Time-To-Live (TTL)</i> analysis, TCP checksum validation.
Timing attacks	By sending packets at a rate low enough so as to not trigger a signature (for example, a flood signature that triggers at 1000 packets per second, and the attacker sending packets at 900 packets per second).	Configurable intervals and use of third-party correlation
Encryption and tunneling	Encrypted payloads are called encrypted for a reason. If an IPS/IDS sees only encrypted traffic, the attacker can build a <i>Secure Sockets Layer (SSL)</i> or IPsec session between himself and the victim and could then send private data over that <i>virtual private network (VPN)</i> .	If traffic is encrypted and passing through the sensor as encrypted data, the encrypted payload cannot be inspected. For <i>generic routing encapsulation (GRE)</i> tunnels, there is support for inspection if the data is not encrypted.

Evasion Method	Description	Cisco Anti-Evasion Techniques
Resource exhaustion	If thousands of alerts are being generated by distractor attacks, an attacker may just be trying to disguise the single attack that they are trying to accomplish. The resource exhaustion could be overwhelming the sensor and overwhelming the administration team who has to view the events.	Dynamic and configurable event summarization. Here is an example: 20,000 devices are all under the control of the attacker. All those devices begin to send the same attack. The sensor summarizes those by showing a few of the attacks as alerts, and then summaries at regular intervals that indicate the attack is still in play and how many thousands of times it occurred over the last interval. This is much better than trying to wade through thousands of individual alerts.

Table 15-7 *Micro-Engines (Groupings of Signatures)*

Signature Micro-Engine	Signatures in This Grouping
Atomic	Signatures that can match on a single packet, as compared to a string of packets
Service	Signatures that examine application layer services, regardless of the operating system
String or Multistring	Supports flexible pattern matching, and can be identified in a single packet or group of packets, such as a session
Other	Miscellaneous signatures that may not specifically fit into other categories

Chapter 16

Table 16-3 *Matrix for Retired/Unretired/Enabled/Disabled*

Compiling/ Allowing Action	Enabled	Disabled
Retired	No memory consumption, and no action related to the signature during packet analysis	No memory consumption, and no action related to the signature during packet analysis
Unretired	Consumes memory, and the signature is considered during packet analysis	Consumes memory, but no action related to the signature during packet analysis

Chapter 17

Table 17-2 *VPN Components*

Component	Function	Examples of Use
Symmetrical encryption algorithms	Uses the same key for encrypting and decrypting data.	DES, 3DES, AES, IDEA
Asymmetrical encryption	Uses a public and private key. One key encrypts the data, and the other key in the pair is used to decrypt.	RSA, Diffie-Hellman
Digital signature	Encryption of hash using private key, and decryption of hash with the sender's public key.	RSA signatures
Diffie-Hellman key exchange	Uses a public-private key pair asymmetrical algorithm, but creates final shared secrets (keys) that are then used by symmetrical algorithms.	Used as one of the many services of IPsec
Confidentiality	Encryption algorithms provide this by turning clear text into cipher text.	DES, 3DES, AES, RSA, IDEA
Data integrity	Validates data by comparing hash values.	MD5, SHA-1
Authentication	Verifies the peer's identity to the other peer.	PSKs, RSA signatures

Chapter 18

Table 18-2 *Key PKI Components*

Component	Description
RSA digital signatures	Using its private key to encrypt a generated hash, a digital signature is created. The receiver uses the public key of the sender to validate the digital signature and verify the identity of the peer.
Digital certificate	File that contains the public key of the entity, a serial number, and the signature of the CA that issued the certificate
Public and private keys	Used as a pair to encrypt and decrypt data in an asymmetrical fashion.
Certificate authority	The CA's job is to fulfill certificate requests and generate the digital certificates for its clients to use. It also maintains a list of valid certificates that have been issued, and maintains a CRL listing any revoked certificates.
X.509v3	A common certificate format used today.
Subordinate CA/RA	Assistant to the CA, which can issue certificates to clients. Clients need both the certificates from the root and the subordinate to verify signatures all the way to the root. Used in a hierarchal PKI topology.
PKCS	Public Key Cryptography Standards, agreed to and implemented by vendors who want the ability to have compatibility with other devices in the PKI.

Chapter 19

Table 19-2 *IPsec Goals and the Methods Used to Implement Them*

Goal	Method That Provides the Feature
Confidentiality	Encryption
Data integrity	Hashing
Peer authentication	Pre-shared keys, RSA digital signatures
Antireplay	Integrated into IPsec, basically applying serial numbers to packets

Chapter 20

Table 20-3 *IKE Phase 1 Policy Options*

Function	Strong Method	Stronger Method
Hashing	MD5, 128-bit	SHA1, 160-bit
Authentication	Pre-shared key (PSK)	RSA-Sigs (digital signatures)
Group # for DH key exchange	1,2	5
Lifetime	86400 seconds (1 day, default)	Shorter than 1 day, 3600
Encryption	3DES	AES-128 (or 192, or 256)

Table 20-4 *IKE Phase 2 Policy Options*

Item to Plan	Implemented By	Notes
Peer IP addresses	Crypto map	Having a known reachable IP address for the VPN peer is critical for the traditional IPsec site-to-site tunnel to negotiate and establish the VPN (both phases).
Traffic to encrypt	Crypto ACL, which is referred to in the crypto map	Extended ACL that is not applied to an interface but is referenced in the crypto map. This should <i>only</i> reference outbound (egress) traffic, which should be protected by IPsec. Traffic not matching the crypto ACL will not be encrypted, but will be sent as a normal packet.
Encryption method	Transform set, which is referred to in the crypto map	DES, 3DES, AES are all options. IKE phase 2 does not need to be the same method as Phase 1. The method does need to match the peer's policy (transform sets) for Phase 2.
Hashing (HMAC) method	Transform set, which is referred to in the crypto map	MD5 and SHA HMACs may be used, and need to match the Phase 2 policy of the peer.

Item to Plan	Implemented By	Notes
Lifetime (time, or data)	Global configuration command: <code>crypto ipsec security-association lifetime ...</code>	Lifetime for Phase 2 should match between the peers. If both use the default lifetime (by not specifying a lifetime), both peers would have compatible lifetime policies. The lifetime can be specified as number of seconds or number of kilobytes.
<i>Perfect Forward Secrecy (PFS)</i> (run DH again or not)	Crypto map	DH is run during IKE Phase 1, and Phase 2 reuses that same keying material that was generated. If you want Phase 2 to rerun the DH, it is called Perfect Forward Secrecy (PFS), and you must choose a DH group number 1,2 or 5 for Phase 2 to use.
Which interface used to peer with the other VPN device	Crypto map applied to the outbound interface	From a routing perspective, this is the interface of a VPN peer that is closest to the other peer, where outbound IPsec packets are leaving the router and inbound IPsec packets are coming into the router.

Chapter 21

Table 21-3 *Comparison Between SSL and TLS*

SSL	TLS
Developed by Netscape in the 1990s	Standard developed by the Internet <i>Engineering Task Force (IETF)</i>
Starts with a secured channel and continues directly to security negotiations on a dedicated port	Can start with unsecured communications and dynamically switch to a secured channel based on the negotiation with the other side
Widely supported on client-side applications	Supported and implemented more on servers, compared to end-user devices
More weaknesses identified in older SSL versions	Stronger implementation because of the standards process

Table 21-4 *Options for SSL VPN Implementation*

	Clientless SSL VPN	Clientless SSL VPN with Plug-Ins for Some Port Forwarding	Full AnyConnect SSL VPN Client
Other names	Web VPN.	Thin client.	Full SSL client.
Installed software on client	No client required.	Small applets and/or configuration required.	Full install of AnyConnect required, but may be installed by initially connecting via the clientless option, and securely installing it that way.
User experience	Feels like accessing resources (that are on the corporate network) through a specific browser window or hyperlink.	Some applications can be run locally with output redirected through the VPN. Includes the features of the clientless VPN to the left.	Full access to the corporate network. The local computer acts and feels like it is a full participant on the corporate network.
Servers that can be used	IOS with the correct software, and ASA with the correct licenses.	IOS with the correct software, and ASA with the correct licenses.	IOS with the correct software, and ASA with the correct licenses.
How the user looks from the corporate network	Traffic is proxied (<i>Port Address Translation [PAT]</i>) by the SSL server, as the users packets enter the corporate network.	Traffic is proxied (<i>Port Address Translation [PAT]</i>) by the SSL server as the users packets enter the corporate network.	Clients are assigned their own virtual IP address to use while accessing the corporate network. Traffic is forwarded from the given IP address of the client into the corporate network.
Clients supported	Most SSL-capable computers.	Computers that support SSL and Java.	Most computers that support SSL.

This page intentionally left blank