

**4 FREE BOOKLETS**  
YOUR SOLUTIONS MEMBERSHIP



# HOW TO CHEAT AT VoIP Security

## **The Perfect Reference for the Multitasked SysAdmin**

- Discover Why “Measure Twice, Cut Once” Applies to Securing a VoIP Infrastructure
- Learn How to Secure an Entire VoIP Infrastructure and Defend Against Denial-of-Service and Hijacking Attacks
- The Perfect Guide if VoIP Engineering is NOT Your Specialty

**Thomas Porter**  
**Michael Gough**

# VISIT US AT

[www.syngress.com](http://www.syngress.com)

Syngress is committed to publishing high-quality books for IT Professionals and delivering those books in media and formats that fit the demands of our customers. We are also committed to extending the utility of the book you purchase via additional materials available from our Web site.

## **SOLUTIONS WEB SITE**

To register your book, visit [www.syngress.com/solutions](http://www.syngress.com/solutions). Once registered, you can access our [solutions@syngress.com](mailto:solutions@syngress.com) Web pages. There you may find an assortment of value-added features such as free e-books related to the topic of this book, URLs of related Web sites, FAQs from the book, corrections, and any updates from the author(s).

## **ULTIMATE CDs**

Our Ultimate CD product line offers our readers budget-conscious compilations of some of our best-selling backlist titles in Adobe PDF form. These CDs are the perfect way to extend your reference library on key topics pertaining to your area of expertise, including Cisco Engineering, Microsoft Windows System Administration, CyberCrime Investigation, Open Source Security, and Firewall Configuration, to name a few.

## **DOWNLOADABLE E-BOOKS**

For readers who can't wait for hard copy, we offer most of our titles in downloadable Adobe PDF form. These e-books are often available weeks before hard copies, and are priced affordably.

## **SYNGRESS OUTLET**

Our outlet store at [syngress.com](http://syngress.com) features overstocked, out-of-print, or slightly hurt books at significant savings.

## **SITE LICENSING**

Syngress has a well-established program for site licensing our e-books onto servers in corporations, educational institutions, and large organizations. Contact us at [sales@syngress.com](mailto:sales@syngress.com) for more information.

## **CUSTOM PUBLISHING**

Many organizations welcome the ability to combine parts of multiple Syngress books, as well as their own content, into a single volume for their own internal use. Contact us at [sales@syngress.com](mailto:sales@syngress.com) for more information.

This Page Intentionally Left Blank

SYNGRESS®

HOW TO CHEAT AT

# VoIP Security

Thomas Porter  
Michael Gough

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

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

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

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

Syngress Media®, Syngress®, “Career Advancement Through Skill Enhancement®,” “Ask the Author UPDATE®,” and “Hack Proofing®,” are registered trademarks of Syngress Publishing, Inc. “Syngress: The Definition of a Serious Security Library”™, “Mission Critical™” and “The Only Way to Stop a Hacker is to Think Like One™” are trademarks of Syngress Publishing, Inc. Brands and product names mentioned in this book are trademarks or service marks of their respective companies.

**KEY SERIAL NUMBER**

001	HJIRTCV764
002	PO9873D5FG
003	829KM8NJH2
004	VTY45Q9PLA
005	CVPLQ6WQ23
006	VBP965T5T5
007	HJJJ863WD3E
008	2987GVTWMK
009	629MP5SDJT
010	IMWQ295T6T

**PUBLISHED BY**

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

**How to Cheat at VoIP Security**

Copyright © 2007 by Syngress Publishing, Inc. All rights reserved. Printed in the United States of America. Except as permitted under the Copyright Act of 1976, no part of this publication may be reproduced or distributed in any form or by any means, or stored in a database or retrieval system, without the prior written permission of the publisher, with the exception that the program listings may be entered, stored, and executed in a computer system, but they may not be reproduced for publication.

Printed in the United States of America

1 2 3 4 5 6 7 8 9 0

ISBN 10: 1-59749-169-1

ISBN 13: 978-1-59749-169-3

Publisher: Amorette Pedersen  
Acquisitions Editor: Gary Byrnes  
Technical Editor: Thomas Porter  
Cover Designer: Michael Kavish

Page Layout and Art: Patricia Lupien  
Copy Editors: Adrienne Rebello, Mike  
McGee  
Indexer: Nara Wood

Distributed by O’Reilly Media, Inc. in the United States and Canada.

For information on rights, translations, and bulk sales, contact Matt Pedersen, Director of Sales and Rights, at Syngress Publishing; email [matt@syngress.com](mailto:matt@syngress.com) or fax to 781-681-3585.



## Lead Author and Technical Editor

**Thomas Porter, Ph.D.** (CISSP, IAM, CCNP, CCDA, CCNA, ACE, CCSA, CCSE, and MCSE) is the Lead Security Architect in Avaya's Consulting & Systems Integration Practice. He also serves as Director of Network Security for the FIFA World Cup 2006.

Porter has spent over 10 years in the networking and security industry as a consultant, speaker, and developer of security tools. Porter's current technical interests include VoIP security, development of embedded micro-controller and FPGA Ethernet tools, and H.323/SIP vulnerability test environments. He is a member of the IEEE and OASIS (Organization for the Advancement of Structured Information Standards). Porter recently published Foundation articles for SecurityFocus titled "H.323 Mediated Voice over IP: Protocols, Vulnerabilities, and Remediation" and "Perils of Deep Packet Inspection."

Tom lives in Chapel Hill, NC, with his wife, Kinga—an Asst. Professor of Internal Medicine at the University of North Carolina—and two Chesapeake Bay Retrievers.



## Contributors

**Brian Baskin** (MCP, CTT+) is a researcher and developer for Computer Sciences Corporation, on contract to the Defense Cyber Crime Center's (DC3) Computer Investigations Training Program (DCITP). Here, he researches, develops, and instructs computer forensic courses for members of the military and law enforcement. Brian currently specializes in Linux/Solaris intrusion investigations, as well as investigations of various network applications. He has designed and implemented networks to be used in scenarios, and he has also exercised penetration-testing procedures.

Brian has been instructing courses for six years, including presentations at the annual DoD Cyber Crime Conference. He is an avid amateur programmer in many languages, beginning when his father purchased QuickC for him when he was 11, and he has geared much of his life around the implementations of technology. He has also been an avid Linux user since 1994 and enjoys a relaxing terminal screen whenever he can. He has worked in networking environment for over 10 years from small Novell networks to large, mission-critical, Windows-based networks.

Brian lives in the Baltimore, MD, area with his lovely wife and son. He is also the founder, and president, of the Lightning Owners of Maryland car club. Brian is a motor sports enthusiast and spends much of his time building and racing his vehicles. He attributes a great deal of his success to his parents, who relinquished their household 80286 PC to him at a young age and allowed him the freedom to explore technology.

**Joshua Brashars** is a security researcher for the External Threat Assessment Team at Secure Science Corporation. Before that, Joshua spent many years in the telecommunications industry as an implementation consultant for traditional and VoIP PBX systems. Joshua would like to extend heartfelt thanks to his family, friends, Lance James and SSC, Johnny Long and all of johnny.ihackstuff.com, and a special nod to Natas, Strom Carlson, and lucky225 for fueling the fire in his passion for telephone systems.

**Michael Cross** (MCSE, MCP+I, CNA, Network+) is an Internet Specialist/Computer Forensic Analyst with the Niagara Regional Police Service (NRPS). He performs computer forensic examinations on computers involved in criminal investigation. He also has consulted and assisted in cases dealing with computer-related/Internet crimes. In addition to designing and maintaining the NRPS Web site at [www.nrps.com](http://www.nrps.com) and the NRPS intranet, he has provided support in the areas of programming, hardware, and network administration. As part of an information technology team that provides support to a user base of more than 800 civilian and uniform users, he has a theory that when the users carry guns, you tend to be more motivated in solving their problems.

Michael also owns KnightWare ([www.knightware.ca](http://www.knightware.ca)), which provides computer-related services such as Web page design, and Bookworms

([www.bookworms.ca](http://www.bookworms.ca)), where you can purchase collectibles and other interesting items online. He has been a freelance writer for several years, and he has been published more than three dozen times in numerous books and anthologies. He currently resides in St. Catharines, Ontario, Canada, with his lovely wife, Jennifer, his darling daughter, Sara, and charming son, Jason.

**Dan Douglass** (MCSE+I, MCDBA, MCSD, MCT, Brainbench .Net Programmer Job Role) is the Special Projects Manager with a cutting-edge medical software company in Dallas, TX. His latest venture is as President/Owner of a new technology firm, Code Hatchery. He currently provides software development skills and internal training and integration solutions, as well as peer guidance for technical skills development. Dan's specialties include enterprise application integration and design; HL7, XML, XSL, C++, C#, JavaScript, Visual Basic, and Visual Basic.Net; database design and administration; Back Office and .NET Server platforms; Network design, including LAN and WAN solutions; all Microsoft operating systems; and Mac OS X, FreeBSD, and Linux. When he has free time, Dan teaches programming, database design, and database administration at a prominent Dallas university. Dan is a former U.S. Navy Nuclear Submariner and lives in Plano, TX, with his very supportive and understanding wife, Tavish.

Dan wishes to extend special thanks to his mother-in-law, Sue Moffett, for all her love and support through the years.

**Bradley Dunsmore** (CCNP, CCDP, CCSP, INFOSEC, MCSE+I, MCDBA) is a Software/QA engineer for the Voice Technology Group at Cisco Systems Inc. He is part of the Golden Bridge solution test team for IPT based in RTP, NC. His responsibilities include the design, deployment, testing, and troubleshooting of Cisco's enterprise voice portfolio. His focus area is the integration of Cisco's network security product line in an enterprise voice environment. Bradley has been working with Cisco's network security product line for four years, and he is currently working on his CCIE lab for Security. Prior to his six years at Cisco, Bradley worked for Adtran, for Bell Atlantic, and as a network integrator in Virginia Beach, VA.

Bradley has authored, coauthored, or edited several books for Syngress Publishing and Cisco Press for network security, telecommunication, and general networking. He would like to thank his fiancée, Amanda, for her



unwavering support in everything that he does. Her support makes all of this possible.

**Michael Gough** is host and webmaster of [www.SkypeTips.com](http://www.SkypeTips.com), which was launched in January 2005 and receives more than 100,000 hits per month, and [www.VideoCallTips.com](http://www.VideoCallTips.com), which receives more than 30,000 hits per month. Michael writes articles on Skype and related issues. He also explains Skype's options and instructions to users so that they can practically apply Skype at home and in the workplace. Michael also evaluates products used with Skype and provides feedback to the vendors on features and improvements to help drive the direction of Skype-related products. Michael is also the host and webmaster for [www.VideoCallTips.com](http://www.VideoCallTips.com), a Web site focused on helping people understand how to make video calls to family and friends, and maintains ratings of the many video call solutions available.

Michael's full-time employment is as a computer security consultant with 18 years' experience in the computer technology field. Michael works for a Fortune 500 company, where he delivers security consulting services to its clients. Michael also presents for his company at many trade shows and conferences and works with associations and groups, advising agencies like the FBI on Skype security and the Center for Internet Security on wireless security.

**Tony Rosela** (PMP, CTT+) is a Senior Member Technical Staff with Computer Sciences Corporation working in the development and delivery of technical instructional material. He provides leadership through knowledge and experience with the operational fundamentals of PSTN architecture and how the PSTN has evolved to deliver high-quality services, including VoIP. His other specialties include IP enabling voice networks, WAN voice and data network design, implementation and troubleshooting as well as spending a great deal of time in the field of computer forensics and data analysis.

**Choon Shim** is responsible for Qovia's technology direction and development of the Qovia product line.

Choon was previously President at Widearea Data Systems, where he designed and developed collaboration platform software. Prior to joining Widearea Data Systems, he was the Senior Development Manager and Principal Engineer for Merant.

Choon is a successful technology leader with 20+ years' experience architecting, building, and delivering large-scale infrastructure software products. He has extensive hands-on technical development skills and has successfully managed software teams for well-known enterprise software companies, including BMC Software and EMC Corporation.

Choon is the author of *Community Works* and *Express/OS shareware* used widely throughout the world. He is a frequent speaker at VoIP and networking conferences for academic and industry. He recently gave a keynote speech to an SNPD conference and chaired a VoIP Security Panel at Supercomm05. Choon holds a B.S. in Computer Science from Kyoungpook National University and an M.S in Electrical Engineering from the University of Wisconsin.

**Michael Sweeney** (CCNA, CCDA, CCNP, MCSE, SCP) is the owner of the Network Security consulting firm Packetattack.com. Packetattack.com's specialties are network design and troubleshooting, wireless network design, security, and analysis. The Packetattack team uses industry-standard tools such as Airmagnet, AiroPeekNX, and NAI Sniffer. Packetattack.com also provides digital forensic analysis services.

Michael has been a contributing author for Syngress for the books *Cisco Security Specialist's Guide to PIX Firewalls* (ISBN: 1-931836-63-9), *Cisco Security Specialist's Guide to Secure Intrusion Detection Systems* (ISBN: 1-932266-69-0), and *Building DMZs for Enterprise Networks* (ISBN: 1-931836-88-4). Through PacketPress, Michael has also published *Securing Your Network Using Linux* (ISBN: 1-411621-77-8).

Michael has recently joined the ranks of "Switchers" where he is now using two OS X Macs full-time in security work and day-to-day activities. He keeps a running blog on his misadventures and discoveries about Apple, OS X, and Macs in general at [hackamac.packetattack.com](http://hackamac.packetattack.com).

Michael graduated from the University of California, Irvine, extension program with a certificate in communications and network engineering.

Michael currently resides in Orange, CA, with his wife, Jeanne, and his three daughters, Amanda, Sara, and Olivia

**Stephen Watkins** (CISSP) is an Information Security Professional with more than 10 years of relevant technology experience, devoting eight of these years to the security field. He currently serves as Information Assurance Analyst at Regent University in southeastern Virginia. Before coming to Regent, he led a team of security professionals providing in-depth analysis for a global-scale government network. Over the last eight years, he has cultivated his expertise with regard to perimeter security and multilevel security architecture. His Check Point experience dates back to 1998 with FireWall-1 version 3.0b. He has earned his B.S. in Computer Science from Old Dominion University and M.S. in Computer Science, with Concentration in Infosec, from James Madison University. He is nearly a life-long resident of Virginia Beach, where he and his family remain active in their Church and the local Little League.

**Andy Zmolek** is Senior Manager, Security Planning and Strategy at Avaya. In that role, Andy drives product security architecture and strategy across Avaya's voice and data communications products. Previously at Avaya, he helped launch the Avaya Enterprise Security Practice, led several Sarbanes-Oxley-related security projects within Avaya IT, and represented Avaya in standards bodies (IETF, W3C) as part of the Avaya CTO Standards Group. Avaya Inc. designs, builds and manages communications networks for more than one million businesses worldwide, including over 90 percent of the FORTUNE 500®.

Andy has been involved with network security for over a decade, and is an expert on Session Initiation Protocol (SIP) and related VoIP standards, Presence systems, and firewall traversal for VoIP. He holds a degree in Mathematics from Brigham Young University and is NSA IAM certified.

Prior to joining Avaya, he directed network architecture and operations at New Era of Networks, a pioneer of enterprise application integration (EAI) technology, now a division of Sybase. Andy got his start in the industry as a systems architect responsible for the design and operation of secure real-time simulation networks for missile and satellite programs at Raytheon, primarily with the Tomahawk program.

# Contents

<b>Chapter 1 Introduction to VoIP Security</b> . . . . .	<b>1</b>
Introduction . . . . .	2
The Switch Leaves the Basement . . . . .	4
What Is VoIP? . . . . .	6
VoIP Benefits . . . . .	6
VoIP Protocols . . . . .	8
VoIP Isn't Just Another Data Protocol . . . . .	9
Security Issues in Converged Networks . . . . .	11
VoIP Threats . . . . .	14
A New Security Model . . . . .	15
Summary . . . . .	16
<b>Chapter 2 The Hardware Infrastructure</b> . . . . .	<b>19</b>
Introduction . . . . .	20
Traditional PBX Systems . . . . .	21
PBX Lines . . . . .	22
PBX Trunks . . . . .	24
PBX Features . . . . .	25
PBX Adjunct Servers . . . . .	28
Voice Messaging . . . . .	28
Interactive Voice Response Servers . . . . .	29
Wireless PBX Solutions . . . . .	30
Other PBX Solutions . . . . .	30
PBX Alternatives . . . . .	30
VoIP Telephony and Infrastructure . . . . .	31
Media Servers . . . . .	31
Interactive Media Service: Media Servers . . . . .	32
Call or Resource Control: Media Servers . . . . .	32
Media Gateways . . . . .	33
Firewalls and Application-Layer Gateways . . . . .	34
Application Proxies . . . . .	34
Endpoints (User Agents) . . . . .	35
IP Switches and Routers . . . . .	38
Wireless Infrastructure . . . . .	38
Wireless Encryption: WEP . . . . .	38

Wireless Encryption: WPA2	39
Authentication: 802.1x	40
Power-Supply Infrastructure	41
Power-over-Ethernet (IEEE 802.3af)	41
UPS	42
Energy and Heat Budget Considerations	43
Summary	44
<b>Chapter 3 Architectures</b>	<b>45</b>
Introduction	46
PSTN: What Is It, and How Does It Work?	46
PSTN: Outside Plant	46
PSTN: Signal Transmission	49
T1 Transmission: Digital Time Division Multiplexing	49
PSTN: Switching and Signaling	55
The Intelligent Network (IN), Private	
Integrated Services, ISDN, and QSIG	56
ITU-T Signaling System Number 7 (SS7)	57
PSTN: Operational and Regulatory Issues	61
PSTN Call Flow	61
PSTN Protocol Security	64
SS7 and Other ITU-T Signaling Security	64
ISUP and QSIG Security	66
The H.323 Protocol Specification	67
The Primary H.323 VoIP-Related Protocols	68
H.225/Q.931 Call Signaling	71
H.245 Call Control Messages	75
Real-Time Transport Protocol	77
H.235 Security Mechanisms	78
Understanding SIP	82
Overview of SIP	83
RFC 2543 / RFC 3261	84
SIP and Mbone	85
OSI	85
SIP Functions and Features	87
User Location	88
User Availability	88
User Capabilities	88
Session Setup	89

Session Management . . . . .	89
SIP URIs . . . . .	89
SIP Architecture . . . . .	90
SIP Components . . . . .	90
User Agents . . . . .	90
SIP Server . . . . .	91
Stateful versus Stateless . . . . .	92
Location Service . . . . .	92
Client/Server versus Peer-to-Peer Architecture . . . . .	93
Client/Server . . . . .	93
Peer to Peer . . . . .	94
SIP Requests and Responses . . . . .	94
Protocols Used with SIP . . . . .	97
UDP . . . . .	97
Transport Layer Security . . . . .	98
Other Protocols Used by SIP . . . . .	99
Understanding SIP's Architecture . . . . .	102
SIP Registration . . . . .	102
Requests through Proxy Servers . . . . .	103
Requests through Redirect Servers . . . . .	103
Peer to Peer . . . . .	104
Instant Messaging and SIMPLE . . . . .	105
Instant Messaging . . . . .	106
SIMPLE . . . . .	107
Summary . . . . .	109
<b>Chapter 4 Support Protocols . . . . .</b>	<b>111</b>
Introduction . . . . .	112
DNS . . . . .	112
DNS Architecture . . . . .	113
Fully Qualified Domain Name . . . . .	114
DNS Client Operation . . . . .	115
DNS Server Operation . . . . .	116
Security Implications for DNS . . . . .	117
TFTP . . . . .	118
TFTP Security Concerns . . . . .	118
TFTP File Transfer Operation . . . . .	119
Security Implications for TFTP . . . . .	119
HTTP . . . . .	120
HTTP Protocol . . . . .	121

HTTP Client Request .....	121
HTTP Server Response .....	122
Security Implications for HTTP .....	122
SNMP .....	123
SNMP Architecture .....	124
SNMP Operation .....	124
SNMP Architecture .....	125
DHCP .....	126
DHCP Protocol .....	126
DHCP Operation .....	127
Security Implications for DHCP .....	128
RSVP .....	129
RSVP Protocol .....	130
RSVP Operation .....	130
Security Implications for RSVP .....	131
SDP .....	132
SDP Specifications .....	132
SDP Operation .....	133
Security Implications for SDP .....	134
Skinny .....	135
Skinny Specifications .....	135
Skinny Operation .....	135
Security Implications for Skinny .....	136
Summary .....	138
<b>Chapter 5 Threats to VoIP Communications Systems ..</b>	<b>141</b>
Introduction .....	142
Denial-of-Service or VoIP Service Disruption .....	142
Call Hijacking and Interception .....	148
ARP Spoofing .....	151
H.323-Specific Attacks .....	155
SIP-Specific Attacks .....	156
Summary .....	157
<b>Chapter 6 Confirm User Identity .....</b>	<b>159</b>
Introduction .....	160
802.1x and 802.11i (WPA2) .....	163
802.1x/EAP Authentication .....	164
Supplicant (Peer) .....	164
Authenticator .....	164

Authentication Server . . . . .	164
EAP Authentication Types . . . . .	167
EAP-TLS . . . . .	169
EAP-PEAP . . . . .	171
EAP-TTLS . . . . .	171
PEAPv1/EAP-GTC . . . . .	171
EAP-FAST . . . . .	171
LEAP . . . . .	172
EAP-MD-5 . . . . .	172
Inner Authentication Types . . . . .	173
Public Key Infrastructure . . . . .	175
Public Key Cryptography Concepts . . . . .	176
Architectural Model and PKI Entities . . . . .	178
Basic Certificate Fields . . . . .	180
Certificate Revocation List . . . . .	181
Certification Path . . . . .	181
Minor Authentication Methods . . . . .	182
MAC Tools . . . . .	182
MAC Authentication . . . . .	183
ARP Spoofing . . . . .	183
Port Security . . . . .	183
Summary . . . . .	183
<b>Chapter 7 Active Security Monitoring . . . . .</b>	<b>185</b>
Introduction . . . . .	186
Network Intrusion Detection Systems . . . . .	187
NIDS Defined . . . . .	187
Components . . . . .	188
Types . . . . .	189
Placement . . . . .	191
Important NIDS Features . . . . .	194
Maintenance . . . . .	194
Alerting . . . . .	194
Logging . . . . .	194
Extensibility . . . . .	194
Response . . . . .	194
Limitations . . . . .	195
Honeypots and Honeynets . . . . .	195
Host-Based Intrusion Detection Systems . . . . .	196



Logging	197
Syslog	197
SNMP	199
What Is a Penetration/Vulnerability Test?	200
Methodology	201
Discovery	201
Scanning	202
Vulnerability Assessment	203
Exploitation	203
Reporting	203
Summary	205
<b>Chapter 8 Logically Segregate Network Traffic</b>	<b>207</b>
Introduction	208
VLANs	209
VLAN Security	212
VLANs and Softphones	212
QoS and Traffic Shaping	214
NAT and IP Addressing	215
How Does NAT Work?	216
NAT Has Three Common Modes of Operation	218
NAT and Encryption	221
NAT as a Topology Shield	225
Firewalls	225
A Bit of Firewall History	226
Shallow Packet Inspection	226
Stateful Inspection	227
Medium-Depth Packet Inspection	227
Deep Packet Inspection	228
VoIP-Aware Firewalls	229
H.323 Firewall Issues	230
SIP Firewall Issues	231
Bypassing Firewalls and NAT	232
Access Control Lists	235
Summary	237
<b>Chapter 9 IETF Encryption Solutions for VoIP</b>	<b>239</b>
Introduction	240
Suites from the IETF	240
S/MIME: Message Authentication	241

S/MIME Messages	244
Sender Agent	244
Receiver Agent	244
E-mail Address	244
TLS: Key Exchange and Signaling Packet Security	244
Certificate and Key Exchange	245
SRTP: Voice/Video Packet Security	247
Multimedia Internet Keying	248
Session Description Protocol Security Descriptions	248
Providing Confidentiality	248
Message Authentications	249
Replay Protection	250
Summary	251
<b>Chapter 10 Skype Security</b>	<b>253</b>
Security	254
Blocking Skype	257
Firewalls	257
Downloads	257
Software Inventory and Administration	258
Firewalls	258
Proxy Servers	260
Embedded Skype	260
A Word about Security	260
<b>Chapter 11 Skype Firewall and Network Setup</b>	<b>263</b>
A Word about Network Address Translation and Firewalls	264
Home Users	266
Small to Medium-Sized Businesses	266
Large Corporations	267
What You Need to Know	
About Configuring Your Network Devices	269
Home Users or Businesses	
Using a DSL/Cable Router and No Firewall	269
Small to Large Company Firewall Users	269
TCP and UDP Primer	269
NAT vs. a Firewall	270
Ports Required for Skype	271
Home Users or Businesses	
Using a DSL/Cable Router and No Firewall	271

Small to Large Company Firewall Users . . . . .	271
Skype's Shared.xml file . . . . .	273
Microsoft Windows Active Directory . . . . .	273
Using Proxy Servers and Skype . . . . .	276
Wireless Communications . . . . .	277
Display Technical Call Information . . . . .	278
Small to Large Companies . . . . .	282
How to Block Skype in the Enterprise . . . . .	282
Endnote . . . . .	283
<b>Appendix A Validate Existing Security Infrastructure</b>	<b>285</b>
Introduction . . . . .	286
Security Policies and Processes . . . . .	287
Physical Security . . . . .	297
Perimeter Protection . . . . .	300
Closed-Circuit Video Cameras . . . . .	300
Token System . . . . .	300
Wire Closets . . . . .	301
Server Hardening . . . . .	301
Eliminate Unnecessary Services . . . . .	302
Logging . . . . .	303
Permission Tightening . . . . .	304
Additional Linux Security Tweaks . . . . .	306
Activation of Internal Security Controls . . . . .	308
Security Patching and Service Packs . . . . .	312
Supporting Services . . . . .	313
DNS and DHCP Servers . . . . .	313
LDAP and RADIUS Servers . . . . .	315
NTP . . . . .	315
SNMP . . . . .	316
SSH and Telnet . . . . .	317
Unified Network Management . . . . .	317
Sample VoIP Security Policy . . . . .	318
Purpose . . . . .	319
Policy . . . . .	319
Physical Security . . . . .	319
VLANs . . . . .	319
Softphones . . . . .	319

Encryption . . . . . 319  
 Layer 2 Access Controls . . . . . 320  
 Summary . . . . . 321

**Appendix B The IP Multimedia Subsystem:  
 True Converged Communications . . . . . 323**

Introduction . . . . . 324  
 IMS Security Architecture . . . . . 325  
 IMS Security Issues . . . . . 328  
     SIP Security Vulnerabilities . . . . . 329  
         Registration Hijacking . . . . . 329  
         IP Spoofing/Call Fraud . . . . . 329  
         Weakness of Digest Authentication . . . . . 329  
         INVITE Flooding . . . . . 329  
         BYE Denial of Service . . . . . 330  
         RTP Flooding . . . . . 330  
         Spam over Internet Telephony (SPIT) . . . . . 330  
     Early IMS Security Issues . . . . . 330  
     Full IMS Security Issues . . . . . 331  
 Summary . . . . . 332  
 Related Resources . . . . . 332

**Appendix C Regulatory Compliance . . . . . 333**

Introduction . . . . . 334  
 SOX: Sarbanes-Oxley Act . . . . . 336  
     SOX Regulatory Basics . . . . . 336  
         Direct from the Regulations . . . . . 336  
         What a SOX Consultant Will Tell You . . . . . 338  
     SOX Compliance and Enforcement . . . . . 341  
         Certification . . . . . 341  
         Enforcement Process and Penalties . . . . . 342  
 GLBA: Gramm-Leach-Bliley Act . . . . . 342  
     GLBA Regulatory Basics . . . . . 343  
         Direct from the Regulations . . . . . 343  
         What a Financial Regulator or  
         GLBA Consultant Will Tell You . . . . . 347  
     GLBA Compliance and Enforcement . . . . . 349  
         No Certification . . . . . 350  
         Enforcement Process and Penalties . . . . . 350

HIPAA: Health Insurance	
Portability and Accountability Act . . . . .	351
HIPAA Regulatory Basics . . . . .	351
Direct from the Regulations . . . . .	351
What a HIPAA Consultant Will Tell You . . . . .	358
HIPAA Compliance and Enforcement . . . . .	359
No Certification . . . . .	359
Enforcement Process and Penalties . . . . .	359
CALEA: Communications Assistance	
for Law Enforcement Act . . . . .	360
CALEA Regulatory Basics . . . . .	363
Direct from the Regulations . . . . .	364
What a CALEA Consultant Will Tell You . . . . .	375
CALEA Compliance and Enforcement . . . . .	376
Certification . . . . .	376
Enforcement Process and Penalties . . . . .	377
E911: Enhanced 911 and Related Regulations . . . . .	377
E911 Regulatory Basics . . . . .	378
Direct from the Regulations . . . . .	378
What an E911 Consultant Will Tell You . . . . .	382
E911 Compliance and Enforcement . . . . .	383
Self-Certification . . . . .	383
Enforcement Process and Penalties . . . . .	383
EU and EU Member States'	
eCommunications Regulations . . . . .	384
EU Regulatory Basics . . . . .	385
Direct from the Regulations . . . . .	385
What an EU Data Privacy Consultant Will Tell You . . . . .	389
EU Compliance and Enforcement . . . . .	390
No Certification . . . . .	390
Enforcement Process and Penalties . . . . .	390
Summary . . . . .	390

## Introduction to VoIP Security

### Solutions in this chapter:

- The Switch Leaves the Basement
- What Is VoIP?
- VoIP Isn't Just Another Data Protocol
- Security Issues in VoIP Networks
- A New Security Model

# Introduction

The business of securing our private data is becoming more important and more relevant each day. The benefits of electronic communication come with proportionate risks. Critical business systems can be and are compromised regularly, and are used for illegal purposes. There are many instances of this: Seisint (Lexis-Nexis research), Choicepoint, Bank of America, PayMaxx, DSW Shoe Warehouses, Ameriprise, and T-Mobile are all recent examples.

- Seisint (Lexis-Nexis research) was hacked, potentially compromising names, addresses, and social security and driver's license information relating to 310,000 people.
- Choicepoint, one of the nation's largest information aggregators, allowed criminals to buy the private identity and credit information of more than 150,000 customer accounts. Besides the harm done to Choicepoint's reputation, in late January, 2006, Choicepoint was fined \$15 million by the FTC for this breach. This figure does not include the millions of dollars spent by Choicepoint on the cleanup of this debacle. This settlement makes it clear that the FTC is increasingly willing to escalate security-related enforcement actions.



## WARNING

---

Victims of personal data security breaches are showing their displeasure by terminating relationships with the companies that maintained their data, according to a new national survey sponsored by global law firm White & Case. The independent survey of nearly 10,000 adults, conducted by the respected privacy research organization Ponemon Institute, reveals that nearly 20 percent of respondents say they have terminated a relationship with a company after being notified of a security breach.

"Companies lose customers when a breach occurs. Of the people we surveyed who received notifications, 19 percent said that they have ended their relationship with the company after they learned that their personal information had been compromised due to security breach. A whopping 40 percent say that they are thinking about terminating their relationship," said Larry Ponemon, founder and head of the Ponemon Institute.

---

- Bank of America announced that it had "lost" tapes containing information on over 1.2 million federal employee credit cards, exposing the individuals involved and the government to fraud and misuse.

- PayMaxx Inc., a Tennessee payroll management company, suffered a security lapse that may have exposed financial data on as many as 100,000 workers.
- DSW Shoe Warehouses revealed that credit card data from about 100 of its stores had been stolen from a company computer over the past three months.
- A hacker even attacked T-Mobile, the cellular telephone network used by actress Paris Hilton, and stole the information stored on Hilton's phone, including private phone numbers of many other celebrities.

These are just a few examples from one month in 2005. Everyone “knows” that information security is important, but what types of damage are we talking about? Certainly, Paris Hilton's phone book is not critical information (except, perhaps to her). Table 1.1 lists the types of losses resulting from attacks on data networks.

**Table 1.1** Losses Resulting from Attacks on Data Networks

<b>Direct Losses</b>	<b>Indirect Losses</b>
Economic theft	Loss of sales
Theft of trade secrets	Loss of competitive advantage
Theft of digital assets	Brand damage
Theft of consumer data	Loss of goodwill
Theft of computing resources	Failure to meet contract obligations
Productivity loss due to data	Noncompliance with privacy regulations
Productivity loss due to spam	corruption
Recovery expenses	Officer liability
	Reparations

The aforementioned bullet points are based on data network examples. VoIP networks simply haven't existed long enough to provide many real-world examples of information breaches. But they will.

The practice of information security has become more complex than ever. By Gartner's estimates, one in five companies has a wireless LAN that the CIO doesn't know about, and 60 percent of WLANs don't have their basic security functions enabled. Organizations that interconnect with partners are beginning to take into account the security environment of those partners. For the unprepared, security breaches and lapses are beginning to attract lawsuits. “It's going to be the next asbestos,” predicts one observer.

The daily challenges a business faces—new staff, less staff, more networked applications, more business partner connections, and an even more hostile Internet environment—should not be allowed to create more opportunities for intruders. The fact is, all aspects of commerce are perilous, and professional security administrators realize that no significant gain is



possible without accepting significant risk. The goal is to intelligently, and economically, balance these risks.

This book is based on the premise that in order to secure VoIP systems and applications, you must first understand them. In addition, efficient and economical deployment of security controls requires that you understand those controls, their limitations, and their interactions with one another and other components that constitute the VoIP and supporting infrastructure.

## The Switch Leaves the Basement

Telephone networks were designed for voice transmission. Data networks were not. Recently—within the last three to five years—PBX functionality has moved logically (and even physically) from the closet or fenced room in the basement into the data networking space, both from physical connectivity and management standpoints. Additionally, the components of the converged infrastructure (gateways, gatekeepers, media servers, IP PBXes, etc.) are no longer esoteric variants of VxWorks, Oryx-Pecos, or other proprietary UNIXs, whose operating systems are not well enough known or distributed to be common hacking targets; but instead run on well-known, commonly exploited Windows and Linux OSes. SS7, which hardly any data networking people understand, is slowly being replaced by SIGTRAN (which is basically SS7 over IP), H.323 (which no one understands ☺), and SIP (which is many things to many people), running over TCP/IP networks. By the way, hackers understand TCP/IP.

Most people, if they even think about it, consider the traditional public switched telephone network (PSTN) secure. On the PSTN the eavesdropper requires physical access to the telephone line or switch and an appropriate hardware bugging device.

### NOTE

---

“Whenever a telephone line is tapped, the privacy of the persons at both ends of the line is invaded, and all conversations between them upon any subject, and although proper, confidential, and privileged, may be overheard. Moreover, the tapping of one man’s telephone line involves the tapping of the telephone of every other person whom he may call, or who may call him. As a means of espionage, writs of assistance and general warrants are but puny instruments of tyranny and oppression when compared with wire tapping.”

—Justice Louis Brandeis, *Olmstead v. United States*, 1928.

---

Toll fraud occurs more frequently than most people realize (one source estimates damages at \$4 billion per year) primarily due to improperly configured remote access policies (DISA—Direct Inward System Access) and voicemail; however, strong authentication codes and passwords, active call detail record accounting, and physical security controls reduce the risk of damage due to toll fraud to reasonable levels. Although it is theoretically possible to “hack” SS7, only sophisticated techniques and direct access to the signaling channel make this possible.

Unlike most standards in data networking—for example, TCP/IP has been relatively stable for more than 20 years now—there is a high degree of inconsistency in support and implementation of VoIP-related standards, due in part to the rapid evolution in the standards themselves, and due in part to vendors attempting to lock in customers to nonstandard protocol implementations. The consequence of this is that, in some cases, immature (vulnerable) applications reach the market. Vendors are oftentimes only familiar with their specific application’s protocol implementation, and when designing a security solution, aren’t always concerned about interoperability. This is actually quite ironic because these same vendors tout standards to foster interoperability.

An additional difference between VoIP and more common protocols is that both major VoIP protocols separate signaling and media on different channels. These channels run over dynamic IP address/port combinations. This has significant security implications that will be detailed later in this book. If you combine this fact (separate signaling and data channels) with the reality that users naturally expect to be able to simply make both inbound and outbound calls, then you should begin to realize that VoIP is more challenging to secure technically than common protocols that initiate with outbound client requests.

VoIP is difficult to firewall. Additionally, since IP addressing information is cascaded within the signaling stream of H.323 and within SIP control packets, encryption of these streams—an obvious security measure—wreaks havoc with NAT implementations. IPv4 was not invented with real-time communications and NAT in mind.

In addition to the vulnerabilities and difficulties that we have summarized, converged networks offer an array of new vectors for traditional exploits and malware. This is due in part to the unique performance requirements of the voice fraction of converged networks, and in part to the fact that more intelligence (particularly in the case of SIP) is moved from the guarded center to the edge of the network. Increased network points of access equals increased network complexity—and complexity is the bane of security engineers. In addition, SIP may become particularly attractive as hacking target, due to its HTTP based underpinnings, and the ease with which ASCII encoded packets can be manipulated.

Are these new problems? Not really. Information systems have long been at some risk from malicious actions or inadvertent user errors, and from natural and man-made disasters. In recent years, systems have become more susceptible to these threats because computers have become more interconnected and, thus, more interdependent, and these systems have become accessible to a larger number of individuals. In addition, the number of individuals

with computer skills is increasing, more automated tools are available, and intrusion, or hacking, techniques are becoming more widely known via the Internet and other media.

Converged VoIP and data networks inherit all the security weaknesses of the IP protocol—including spoofing, sniffing, denial of service attacks, replay attacks, and message integrity attacks. All the legacy application servers that serve as adjuncts in converged networks (DNS, SNMP, TFTP, etc.) will also be targets of attack as they have been on data networks. Viruses and worms will become a real threat to the entire telecommunication infrastructure.

Hacking will converge as well.

Unfortunately, even though the overwhelming majority of VoIP calls will occur uneventfully between two or more trusted individuals—in much the same way that most data sessions take place securely today—the public will focus on extraordinary examples of “the call that went bad.” Our challenge is to restrict these incidents to the best of our abilities.

## What Is VoIP?

Although VoIP, IP Telephony, and Converged Networks all have slightly different definitions, they often are used interchangeably. In this book, we will do the same. When using any of these terms, we are talking about the structures and processes that result from design and implementation of a common networking infrastructure that accommodates data, voice, and multimedia communications. Today, it is all about voice. There are plenty of examples of streaming video, but the enthusiasm today is to replace circuit-switched voice with packet-switched voice within the enterprise and at home across broadband connections.

Why is this happening now? IP telephony adoption is ramping up dramatically for a number of reasons: traditional PBXs and related telco equipment that was upgraded as organizations prepared for Y2K is beginning to reach end-of-life; IP switches are cheaper and potentially offer more features than traditional PBXs; data system administrators and their networks have become more mature, and thus, can support the quality of service that VoIP services require; and VoIP technology (particularly the products) have gotten better. VoIP is attractive to organizations and to broadband end-users as they attempt to derive more value from an infrastructure that is already paid for.

## VoIP Benefits

What does converging voice and data on the same physical infrastructure promise? First, we may actually lower costs after all, due to the economies of supporting one network instead of two. Organizations also will save money on toll bypass, intralata regional toll (also known as local toll) charges, and all the “extra” services that POTS providers currently bill for.

VoIP, from a management and maintenance point of view, is less expensive than two separate telecommunications infrastructures. Implementation can be expensive and painful, but is repaid in the form of lower operating costs and easier administration. The pace and quality of IP application development is increasing in step with VoIP adoption. Features that were unavailable on traditional systems, such as “click-to-talk” with presence awareness, can rapidly be modified and deployed. Even voice encryption, which in the past was limited to select organizations, can now be used by anyone in a VoIP environment.

An often overlooked benefit of converging data and voice is that organizational directories often are updated and consolidated as part of the VoIP deployment process. This not only enables economies in and of itself but also makes features such as Push Directories possible. Push is the capability of an application using the WML protocol to send content to the telephone. IP transforms the everyday telephone into an applications-enabled appliance. The addition of push enables phone displays and/or audio to support a variety of applications (Web browsing, time reporting, emergency alerts, travel reservations, account code entry, announcements, branding via screensaver, inventory lookups, scheduling, etc.).

## NOTE

---

**Presence:** Oftentimes, when discussing VoIP, the term “presence” is thrown around. What is presence? Presence is a system for determining whether or not an individual is available to communicate. In its simplest form, presence has nothing to do with location. In traditional telephony, presence can be determined to some extent by the status of the remote handset after a call is attempted. If the remote handset fails to go off-hook after eight to 10 rings, then the callee is probably not present. A busy tone indicates that the callee is probably present but unavailable. A better example of presence is instant messaging (IM). Instant messaging brought presence—the ability to tell when others are available to chat—to the masses. The next logical step was to incorporate location information into the context of presence. Presence as a source of users’ state information has been maturing over the past few years. In the enterprise the notion of presence is broader. Presence can refer to the type of position a person has (for example, management or call center operator), their physical and organizational location, and a constellation of other personal information.

---

Convergence should simplify telecommunications management. For example, a single management station or cluster can be used to monitor both data and voice components and performance via SNMP. As mentioned earlier in this chapter, directory management will be simplified as well.

## VoIP Protocols

Two major VoIP and multimedia suites dominate today: SIP and H.323. Others (like H.248) exist, and we will discuss some of them in this book, but these are the two major players. For simplicity, I will define SIP and H.323 as signaling protocols. However, whereas H.323 explicitly defines lower level signaling protocols, SIP is really more of an application-layer control framework. The SIP Request line and header field define the character of the call in terms of services, addresses, and protocol features.

Voice media transport is almost always handled by RTP and RTCP, although SCTP (Stream Control Transmission Protocol) has also been proposed and ratified by the IETF (and is used for the IP version of SS7, known as SIGTRAN). The transport of voice over IP also requires a large number of supporting protocols that are used to ensure quality of service, provide name resolution, allow firmware and software upgrades, synchronize network clocks, efficiently route calls, monitor performance, and allow firewall traversal. We talk about these and others in more detail in Chapter 4.

SIP is a signaling protocol for Internet conferencing, telephony, presence, events notification, and instant messaging. SIP is an IETF-ratified response-request protocol whose message flow closely resembles that of HTTP. SIP is a framework in that its sole purpose is to establish sessions. It doesn't focus on other call details. SIP messages are ASCII encoded. A number of open source SIP stacks exist.

H.323, on the other hand, is an ITU protocol suite similar in philosophy to SS7. The H.323 standard provides a foundation for audio, video, and data communications across IP-based networks, including the Internet. The H.323 protocols are compiled using ASN.1 PER. PER (Packed Encoding Rules)—a subset of BER—is a compact binary encoding that is used on limited-bandwidth networks. Also, unlike SIP, H.323 explicitly defines almost every aspect of call flow. The only open source H.323 stack I am aware of is the OpenH323 suite.

Both protocol suites rely upon supplementary protocols in order to provide ancillary services. Both protocols utilize TCP and UDP, and both open a minimum of five ports per VoIP session (Call signaling, two RTP, and two RTCP.) Both protocols offer comparable features, but they are not directly interoperable. Carriers tend to prefer H323 because the methods defined by H.323 make translation from ISDN or SS7 signaling to VoIP more straightforward than for SIP. SIP, on the other hand, is text-based, works better with IM, and typically is implemented on less expensive hardware. H.323 has been the market leader, but SIP rapidly is displacing H.323.

In Table 1.2, many of the more recent protocols that you will find in a VoIP environment are listed. We will talk about these and others in more detail in Chapters 4 and 8.

**Table 1.2** VoIP-Related Protocols

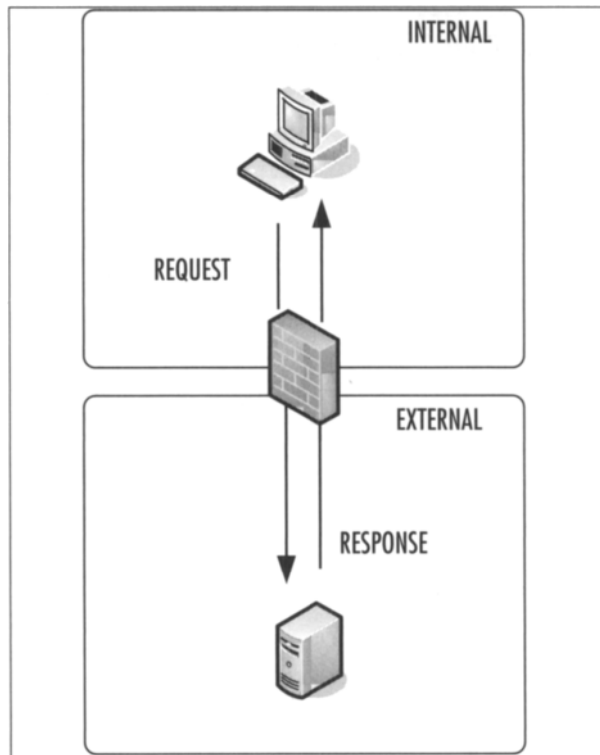
Acronym	Support VoIP Protocol
RTSP	Real Time Streaming Protocol for media play-out control
RSVP	Resource Reservation Protocol
STUN	Simple Traversal of UDP through NAT
TURN	Traversal Using Relay NAT
ICE	Interactive Connectivity Establishment
SDP	Session Discovery Protocol
TLS	Transport Layer Security

## VoIP Isn't Just Another Data Protocol

IP Telephony utilizes the Internet architecture, similar to any other data application. However—particularly from a security administrator's point-of-view—VoIP is different. There are three significant reasons for this:

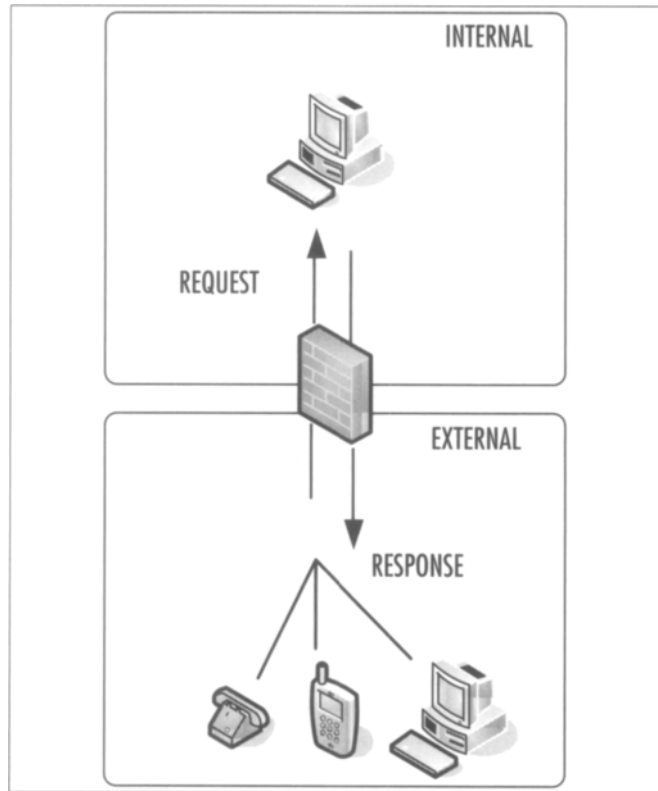
- Voice conversations can be initiated from outside the firewall. Most client-driven protocols initiate requests from inside the firewall. Figure 1.1 shows the basic message flow of a typical Web browsing, e-mail, or SSH session.
- The real-time nature of VoIP—get there a second too late, and the packet is worthless.
- Separation of data and signaling. Sessions, particularly unknown inbound sessions, that define addressing information for the data (media) channel in a discrete signaling channel do not interact well with NAT and encryption.

In Figure 1.1, a request is initiated by a client on the internal side of the firewall to a server daemon residing on a host external to the firewall. Firewalls that are capable of stateful inspection will monitor the connection and open inbound ports if that port is associated with an established session. Application Layer Gateways (ALGs) will behave in a similar manner, proxying outbound and inbound connections for the requesting internal host. For the firewall administrator and the user, the session completes normally, and is as secure as the firewall's permissions allow.

**Figure 1.1** Normal Message Flow

In Figure 1.2, the request-response topology is different from the message flow shown in Figure 1.1. In this figure, an external host (IP Phone, PC softphone, etc.) attempts to place a call to an internal host. Since no session is established, stateful inspection or ALG firewalls will not allow this connection to complete. We talk about this in much more detail in Chapter 8.

There are other differences. VoIP's sensitivity to adverse network conditions is different enough quantitatively from that of most types of data traffic that the difference is qualitative. Real-time applications, including VoIP, place requirements on the network infrastructure that go far beyond the needs of simple best-effort IP transport. Each VoIP packet represents about 20 ms of voice on average. A single lost packet may not be noticeable, but the loss of multiple packets is interpreted by the user as bad voice quality. The simple math indicates that even a short IP telephone call represents the transport of large numbers of packets. Network latency, jitter (interpacket latency variation), and packet loss critically affect the perceived quality of voice communications. If VoIP is going to work, then the network has to perform well—period.

**Figure 1.2** Inbound VoIP Message Flow

Network engineers are accustomed to data network outages. Users, for the most part, don't suffer outages well, but they tolerate them. Users will not be as forgiving with their phone service. Even though cellular telephones seem to have the extraordinary characteristic of dropping connections at the least appropriate or convenient time, enterprise IP telephony users expect their phones to work all the time. Availability is a key VoIP performance metric.

## Security Issues in Converged Networks

Convergence creates a new set of security concerns, as evidenced by the following comment by Winn Schwartau in *Network World's* November 14, 2005 edition:

**The communications world is moving toward VoIP but does not have the security expertise it needs in-house to meet the real-world stress it will encounter.**

In a traditional PSTN network, the PBX or switch encompasses virtually all the intelligence in the system. It is responsible for basic call management including:



- Establishing connections (circuits) between the telephone sets of two or more users
- Maintaining such connections as long as the users require them
- Providing information for management and billing purposes.

Additionally, the PBX usually supports dozens or hundreds of ancillary call functions such as call transfer, call forwarding, voicemail, and so on.

The contemporary IP PBX functions in a similar fashion, although more functionality and intelligence is distributed to the endpoints depending upon the underlying protocols and architecture.

## NOTE

---

**Confidentiality, Integrity, and Availability:** A simple but widely applicable security model is the CIA triad—standing for Confidentiality, Integrity, and Availability—three key principles that should be guaranteed in any kind of secure system. This principle is applicable across the whole security spectrum. Confidentiality refers to mechanisms that ensure that only authorized individuals may access secure information. Cryptography and Encryption are examples of methods used to ensure confidentiality of data. Integrity means that information is unchanged as it moves between endpoints. Availability characterizes the operational state of the network, and usually is expressed as “nines,” or the number of nines on both sides of the decimal point (i.e., 99.999% reliability equals “5 nines”). It is critical to ensure that information is readily accessible to the authorized sender and receiver at all times. The Availability component of this triad is particularly important when securing converged networks.

---

One of the first security issues voiced by organizations implementing VoIP is the issue of the confidentiality of voice conversations. Unlike traditional telephone networks, which are circuit switched and relatively difficult to tap, voice traffic on converged networks is packet switched and vulnerable to interception with the same techniques used to sniff other traffic on a LAN or WAN. Even an unsophisticated attacker can intercept and decode voice conversations.

## Tools & Traps...

### VoIP Call Sniffers

VOIPong is a SIP, H.323, SCCP sniffer utility that can be used to detect and capture VoIP calls. With the appropriate tools, VOIPong can dump the conversation to a separate wave file. The unfortunately named vomit (Voice over Misconfigured Internet Telephones) is an unrelated precursor to VOIPong that can translate tcpdump files of Cisco IP telephone sessions. Other tools exist such as VoIPcrack, but these are not in the public domain.

Although this concern is real, in my view, it is not the most important security threat VoIP faces. Denial of Service (DoS) attacks, whether they are intentional or unintended, are the most difficult VoIP-related threat to defend against. Amplitude Research ([www.amplituderesearch.com](http://www.amplituderesearch.com)) reported in 2005 that:

**Companies had their share of network security problems. Virus and worm attacks led the list of intrusions as 63 percent of companies reported that they've had such problems. Trojan attacks occurred at 58 percent of companies. Backdoor viruses hit 45 percent of companies, while 35 percent say they suffered attacks from viruses or worms that were introduced internally.**

Viruses and worms account for more security-related financial damage than all other security threats combined. The network traffic generated by these agents as they replicate and seek out other hosts to infect has been shown to wreck havoc with even relatively well-secured data networks. Although these data were derived from reports on data networks, VoIP networks, by their nature, are exquisitely sensitive to these types of attacks and should be expected to be affected similarly.

Security administrators can ensure confidentiality using one or several familiar tools. Conversations can be encrypted between endpoints or indirectly by tunneling conversations over VPNs. A PKI or certificate infrastructure, when implemented correctly, guarantees the identities of the two parties involved in a conversation and validates message integrity. But how does this same administrator guarantee availability when the network is under assault from the next incarnation of the Slammer worm? The answer, as it turns out, is that through careful planning and judicious use of networked controls, the physically converged network can be logically separated into compartments much like the bulkheads in a submarine, so that damage to one network compartment is limited to only that compartment. Data network problems can be segregated from the VoIP network and vice versa. We will talk about this approach in much more detail later in the book.

## VoIP Threats

There are a number of ways to classify threats. The most comprehensive list of VoIP threats is maintained by VOIPSA at [www.voipsa.com/Activities/taxonomy.php](http://www.voipsa.com/Activities/taxonomy.php). The threat taxonomy is an excellent introduction to related terminology as well as the technical and social security issues surrounding VoIP. Rather than repeat their results, I've listed VoIP-specific threats based upon a simplified classification: VoIP Data and Service Disruption and VoIP Data and Service Theft. Table 1.3 lists those threats. Some of the more critical threats are explained in more detail in Chapter 5.

**Table 1.3** VoIP-Specific Threats

Type of Risk	Threats
<b>VoIP Data and Service Disruption</b>	VoIP Control Packet Flood
	VoIP Call Data Flood
	TCP/UDP/ICMP Packet Flood
	VoIP Implementation DoS Exploit
	OS/Protocol Implementation DoS Exploit
	VoIP Protocol DoS Exploit
	Wireless DoS Attack
	Network Service DoS Attacks
	VoIP Application Dos Attacks
	VoIP Endpoint PIN Change
	VoIP Packet Replay
	VoIP Packet Injection
	VoIP Packet Modification
	QoS Modification
	VLAN Modification
<b>VoIP Data and Service Theft</b>	VoIP Social Engineering
	Rogue VoIP Device Connection
	ARP Cache Poisoning
	VoIP Call Hijacking
	Network Eavesdropping
	VoIP Application Data Theft
	Address Spoofing
VoIP Call Eavesdropping	

Continued

**Table 1.3 continued** VoIP-Specific Threats

Type of Risk	Threats
	VoIP Control Eavesdropping
	VoIP Toll Fraud
	VoIP Voicemail Hacks

## A New Security Model

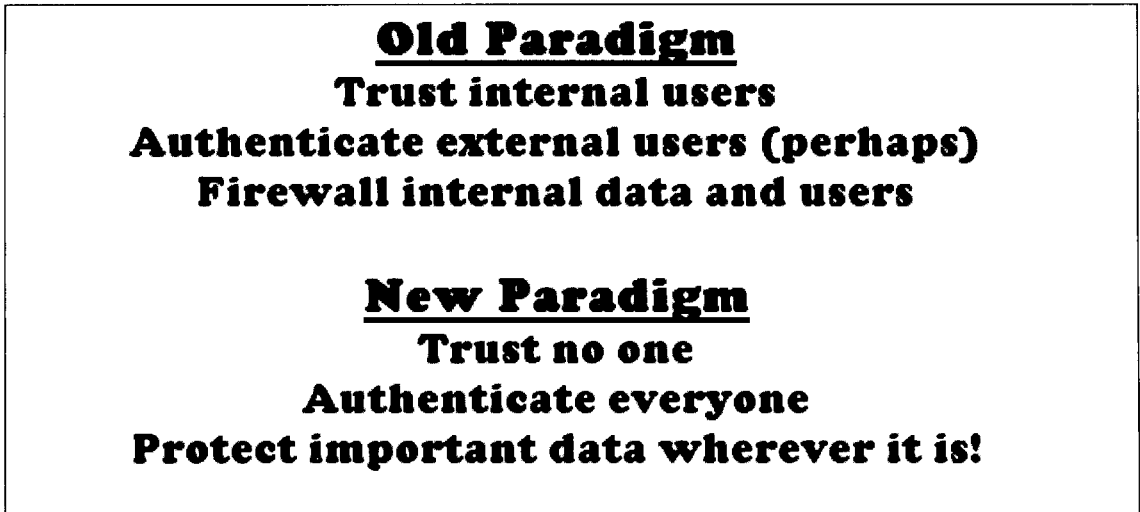
Access to network services is now more important than ever. The growing availability and maturity of Web services combined with advanced directory integration makes it easier to integrate information systems between business partners. Companies are moving their applications out from behind the firewall and onto the edges of their networks, where they can participate in dynamic, Internet-based transactions with customers and business partners. The network perimeter is becoming impossible to define as Intranets, extranets, business partner connections, VPN (Virtual Private Networks), and other RAS (Remote Access Services) services blur the definition of a trusted internal user; and critical corporate data may be located on handhelds, laptops, phones—anywhere.

VoIP distributes applications and services throughout the network. In a VoIP environment, IP phones (obviously) are distributed throughout the infrastructure as well. These devices incorporate microcontrollers and digital signal processors in order to perform voice compression and decompression, line and acoustic echo cancellation, DTMF (Dual Tone, Multi-Frequency—Tone Dial) detection, and network management and signaling. IP phones are smart, and depending upon the vendor, IP phones act as clients for a number of network protocols. This means that the number of network ingress/egress points will increase, and that processor cycles and memory—intelligence—are shifted to the logical edge of the network. This is a reversal of the traditional security model, where critical data is centralized, bounded, and protected.

This means that from a strategic viewpoint, converged networks, regardless of whether they are based upon H.323, SIP, or some other protocol, require a new way of thinking about information security (see Figure 1.3).

“Trust no one” is an obvious bit of overstatement since every functioning system has to trust someone at some point or it won’t work at all. A more concise (but not as catchy) axiom might be: “Don’t assume you can trust anyone.” The point here is this—Any system administrator, user, or device must be authenticated and authorized, regardless of its location, before it is able to access any network resources. Period.

Figure 1.3 The New Security Paradigm



## Summary

We have all heard “Consultant-speak.” Many of us practice it as well. I have done my best in this book to stay away from empty, jargon-laden speech, but I am sure that it creeps in at times. Here is my favorite example:

**Consultant-speak:** VoIP Security is dependent on management of Process.

**What this really means:** Processes define how individuals perform their duties within an organization. For securing VoIP networks, the processes include proactive ones such as formulation of security policies, identity verification management, hardening of operating systems, firewall deployment and configuration, system backup procedures, and penetration testing; and reactive processes such as log analyses, network monitoring, forensics, and incident response. If a process doesn’t exist (e.g., if a task is performed in an ad hoc fashion), then one should be created. The security policies, processes, and standard operating procedures (SOPs) that have already proven successful in securing your data networks need to be reused and extended. The ones that don’t work should be discarded.

Organizations that deploy or plan to deploy VoIP networks will have to work harder at security than before. Security will cost more and it will require better trained administrators. We are getting to the point in networking where naïve system administration is not just bad practice, it may also be criminal. Regulations such as Sarbanes-Oxley (SOX), GLBA, and CALEA in the United States, as well as DPEC in Europe, have been interpreted to mean that privacy violations will be treated as criminal acts.

I've said earlier that the purpose of converging voice and data is to save money by running both types of traffic over the same physical infrastructure and to expand the spectrum of applications that can run over this infrastructure. In this architecture, packetized voice is subject to the same networking and security issues that exist on data-only networks. It seems to me that as organizations transition to this contemporary architecture there exists an unvoiced assumption: Users who have come to expect and accept short outages and sometimes erratic data network performance will *not* accept this same type of performance when it comes to voice communications. Perhaps this is true, or perhaps not. Cellular telephony come to mind here.

Traditional telephone systems have an excellent track record for reliability, and most people never question whether they will receive a dial tone when they pick up the receiver on their handsets. Contrast this with the reliability of most traditional IP networks. These same people who would never question the reliability of their telephone systems are accustomed to IP network outages and outages of systems that connect to the IP network. In a converged network, the loss of availability of the underlying IP network or the loss of availability of the IP telephony equipment (call management and adjunct servers) means the loss of availability of the telephone system.

Many organizations have reasonably well-secured logical perimeters (in so far as they can define those perimeters); however, their overall security environment offers no real defense in depth. Ideally, an enterprise network should offer multiple layers of defense—an intruder who manages to bypass one layer should then be confronted with additional layers, thereby denying the intruder quick access. On most of these networks, an unauthorized user who manages to bypass the logical (and/or physical) perimeter security controls has essentially unlimited access to all of internal assets on the internal IP network.

Authorized users are also assumed trustworthy; they have essentially unlimited access to all assets on the network as well. The lack of network-level security controls on the internal IP network exacerbates the risk of either malicious or accidental network activity, including propagation of worms and viruses.

Most people associate security attacks with the image of the lone hacker, a highly intelligent and motivated individual who attempts to penetrate an organization's IT infrastructure using a public network such as the Internet. Although remote unauthorized users do pose some risk to an organization's IT assets, the most significant IT-related risk to most enterprise organizations is potential financial loss due to direct or collateral damage from a worm or virus.

This point cannot be emphasized enough. The introduction of VoIP into an organization's IP network exacerbates the potential financial losses from a virus or worm outbreak.

The key to securing these networks—as we will see throughout this book—is to:

1. Communicate and enforce security policies.
2. Practice rigorous physical security.

3. Verify user identities.
4. Actively monitor logs, firewalls, and IDSES (Intrusion Detection Systems).
5. Logically segregate data and voice traffic.
6. Harden operating systems.
7. Encrypt whenever and wherever you can.

## The Hardware Infrastructure

### Solutions in this chapter:

- Traditional PBX Systems
- PBX Alternatives
- VoIP Telephony and Infrastructure



# Introduction

Even after the introduction of VoIP, business telephony equipment has remained focused on two areas: (1) reducing the cost of Public Switched Telephone Network (PSTN) connectivity overall and (2) adding business communications feature-functionality. Since the first private branch exchange (PBX) was introduced in 1879, business customers have sought cost savings by reducing the number of physical lines or trunks that interconnect with the PSTN. Because most calls in a large organization remain within it, cost and security benefits accrue immediately by placing a telephone switch inside the organization. And with the introduction of digital switching nearly a century later, a new wave of feature-functionality became possible. For the first time in history, the enterprise telephony capabilities would surpass that offered directly by PSTN carriers. In some respects, the latest developments in VoIP are an extension of this pattern.

## NOTE

---

The basic architecture of the PBX over the past 100 years has evolved similarly to that of the PSTN and its switches overall. If you're interested in that evolution and how it has influenced today's PBX designs, you may want to read Chapter 3 before reading this chapter. Otherwise, consider this chapter to be a discussion of PBX architecture during the past decade and the interaction between the digital PBX and its VoIP equivalents.

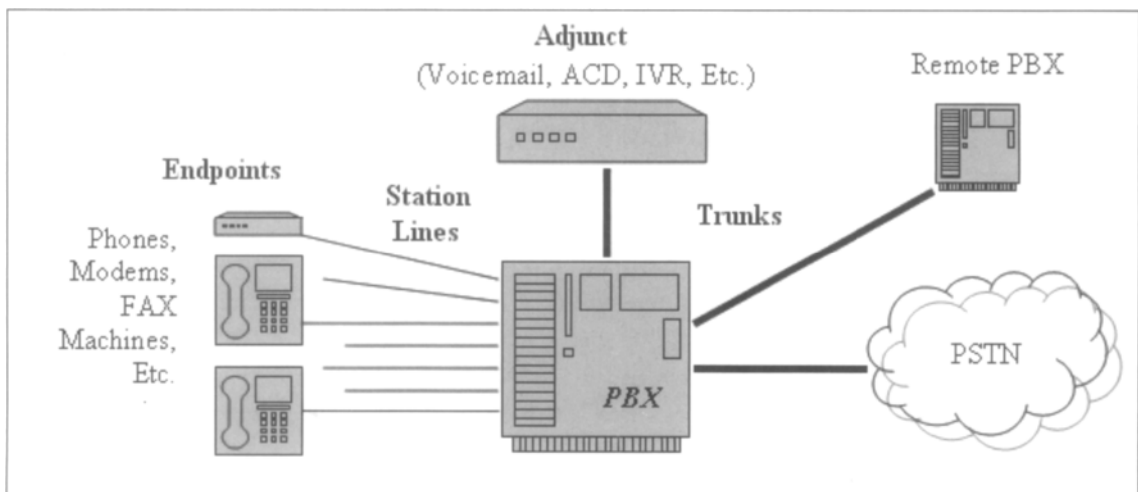
---

From a security perspective, it's important to distinguish between several different architecture models for business and consumer telephony. On the low end, Key Telephone Systems (KTS) for up to 50 users provide a very basic means of sharing outside lines and using dedicated "intercom" lines to talk between stations, but don't provide actual switching services or advanced features (though the latest generation of such systems has blurred that distinction). The traditional PBX is likely to have an Ethernet interface for administration even if it does not support VoIP, and IP-enabled (or *hybrid*) PBX systems can support VoIP interfaces in addition to classic analog or digital stations and trunks. IP-PBX systems dispense with most analog or digital support entirely and focus exclusively on VoIP. In addition, Centrex, IP Centrex, and Hosted IP-telephony services are carrier-based alternatives that provide many of the same switching features as an on-site PBX system but place the switching equipment back into the carrier's infrastructure. Each of these alternatives has a different security profile and may interface with VoIP solutions at different levels, so let's review the critical differences between them and how they may affect your security strategy.

# Traditional PBX Systems

Business telephony in large organizations has revolved around the private branch exchange (PBX) for over a century, and given that length of time, it's easy to see why VoIP often is positioned as a modern alternative to the PBX. However, this comparison is the wrong one to make, as the PBX concept itself is transport-neutral. It would be just as wrong to say “analog vs. PBX” or “digital vs. PSTN,” so let's make sure we've got this basic principle down first. A PBX—or *PABX* internationally (the “A” stands for “Automated”) is a communications switch that (1) replaces PSTN switching functionality for a set of associated extensions, (2) provides access trunks to carriers for routing PSTN calls, and (3) may provide additional communications feature-functionality based on configuration settings and equipment capabilities (see Figure 2.1).

**Figure 2.1** A Basic PBX Diagram\*



\* All PBX systems provide PSTN-like switching services between endpoints and adjuncts, the PSTN, and other private PBX switches (and associated private networks). Only a few of the possible adjunct systems are mentioned here. An ACD is an Automatic Call Distribution server (for use in call centers to direct calls to groups of agents), and an IVR is an Interactive Voice Response server (also commonly used in call centers to let callers use touch tones and voice prompts to select services).

So a PBX could be all IP or all analog or anything in the middle as long as it switches calls between extensions and the PSTN as needed. In the end you will find that despite the marketing hype, most VoIP systems are just PBX systems with different combinations of support for IP lines and trunks. In some cases, the call control part of the system is split out from the gateway that handles the non-IP electrical interfaces. Or it's pushed out to a service provider. But the basic switching concept is preserved somewhere across the system as a

whole. Regardless, understanding basic PBX terminology will help you understand the underlying architecture of the VoIP systems you may encounter, so let's start there.

## PBX Lines

In telephony, a *line* (or *station line*) connects endpoint equipment (digital terminals, analog phones, fax machines, modems, or even an IP phone through an IP network) to the PBX (or central office) for switching. An analog line is the private equivalent of a local loop or loop transmission facility.

### NOTE

---

A PBX is more likely than your phone company to support ground start phones and trunks on analog interfaces. Your phone at home seizes control of the line by using loop start, which involves shorting the two ends of the line together to activate the circuit. Ground start sends one of the leads to ground (typically ring) to seize the line, which is much less likely to cause glare (a condition that arises when both sides on a line or trunk simultaneously seize control of the line).

---

Typically, a PBX supports analog lines (and trunks) through a line card with 8, 12, 16, 24, or more lines per card, which are then wired to a patch panel for interconnection through a structured cabling system to the analog phone or device. Most of the security concerns around analog lines center on how well protected the equipment and cabling systems are from eavesdropping and tampering. Ground start loops will make theft of service less likely because a special phone is required, but otherwise the same basic rules for protecting a PSTN line from tampering apply.

Of course, *line* is also a generic term that may apply to power lines providing electricity to homes and businesses. But when we talk about an analog telephone line, we are talking specifically about the two wires involved: the tip (the first wire in a pair of phone wires, connected to the + side of the battery at the central office or PBX; it is named tip because it was the at the tip of an operator's plug) and the ring (connected to the – side of the switch battery and named because it was connected to the slip ring around the jack). Any equipment that works with Plain Old Telephone Service (POTS) lines will work with a PBX analog line configured for loop start. From a PBX, an analog line will nearly always be 2-wire although 4-wire lines with Earth & Magnet (E&M, sometimes also called Ear and Mouth) interfaces are supported from the same card for analog trunks.



---

If you've ever taken a peek behind the phone jacks that litter the walls of your home, you are likely to see two (or three) pairs of wires, one Green/Red, the next Yellow /Black, then White/Blue, but for our purposes only the first pair is important. The Green wire, referred to as the Tip, is the positively charged terminal. The Red terminal, the Ring, is the neutral, which completes the circuit, enabling electrical signals to flow freely. Note that newer homes may use a more recent color scheme that is also used for Ethernet cabling. The first pair is White/Blue, then White/Orange, then White/Green and finally White/Brown. This scheme is what you're most likely to see in structured cabling systems within buildings.

---

Analog PBX systems supported only analog lines, but with the introduction of digital switching, a new class of line was developed: the digital line. In most PBX systems, a proprietary format for digital line signaling (and media) was created that requires the use of digital phones manufactured by that vendor. Some vendors, however, also support Integrated Services Digital Network (ISDN) standard phones directly (or through the PSTN) via the ITU-standardized ISDN BRI. Most proprietary digital formats use a 2-wire system with 8-wire plugs and jacks, although some are 4-wire systems. ISDN uses a 2-wire system from the CO switch, but is 8-wire to the interface used by a phone terminal, so the actual number of wires used will depend on several factors (such as whether the phone has a built-in NT-1 interface). Also, many proprietary switch features will not be supported on ISDN phones, particularly when the phone is manufactured by a different vendor. And even within a vendor product line, you may discover that newer features are supported only on newer phones or phone firmware. In any case, digital lines for proprietary digital terminals typically are supported by digital line cards with 8, 12, 16, 24, or more lines per card, and ISDN lines for ISDN phones are supported by either ISDN trunk cards or special ISDN BRI line cards, which may come in several flavors depending on the ISDN BRI type.

In the case of the modern hybrid PBX or IP-PBX, there is an equivalent concept for IP lines to IP phones, but unlike analog or digital lines the IP line isn't necessarily tied down to a single electrical interface on the PBX. In fact, the PBX can use multiple Ethernet ports to support an IP line, and IP phones can fail over to multiple IP-enabled PBX systems. The first IP line support built into most PBX systems leveraged the H.323 suite of protocols or proprietary protocols like Cisco "skinny," but almost all new development on PBX systems today uses Session Initiation Protocol (SIP). The bottom line is that the concept of an IP line exists in virtually every VoIP system out there, and understanding how the line concept is expressed in a specific VoIP system will give you an important handle with which to analyze its architecture and security.

This flexibility and versatility are a huge advantage to VoIP, but it does come at a price. Because the phones are now sharing infrastructure and bandwidth with other devices (and perhaps the entire data network), quality-of-service (QoS) guarantees for packet loss, latency (how long each packet takes to arrive from the phone to the PBX), and jitter (variability of latency across packets in a stream) now become the responsibility of the party providing the network infrastructure. Additional vectors for Denial-of-Service attacks on IP lines (either to the phone or the PBX) and Man-In-The-Middle (MITM) attacks must be considered. In my experience, the resulting loss of accountability from a single organization or vendor to multiple entities rarely is included in planning (or ROI calculations) for VoIP deployments.

## PBX Trunks

A trunk is a special kind of line that connects two telephone switches. If one of the two switches is the PBX, the other could be a local or long-distance switch for PSTN access, in which case we would call these local trunks or long-distance trunks, respectively (though it's worth pointing out that even if you don't have dedicated long-distance trunks you likely are able to get long distance services through local trunks). On the other hand, if the other end of the trunk is another privately owned PBX, we would call these private trunks or tie lines, even if they happen to be routed through the PSTN (since the telephone numbers they can reach can only be dialed from within the private network). There are also trunks that can act like both types through the use of Centrex or something called a Virtual Private Network (VPN—but it's not the remote access VPN you may be familiar with from the data world—this VPN is created by a carrier to let you keep a private dial plan across many sites on the same trunks that you use for regular PSTN access).

Some say trunks are so named because in the old days, Ma Bell saw fit to use thick, lead-covered cables to connect the switches. These cables resembled an elephant's trunk. Others claim the word's origin is derived from the way the local loop network resembles the branches of a tree, with the trunks having similarity to... well, a tree trunk. Regardless, trunks are the main lines of the communications system, and the only case where a trunk is not connecting to a switch is when an adjunct server is involved (like a voice messaging server, an Automatic Call Distribution (ACD) server, an Interactive Voice Response (IVR) system, or similar system). In some cases, these servers may use station emulation instead of trunking, so you'll need to verify what actually is being used.

Trunks can be analog, digital, or VoIP-based, just like station lines. Analog trunks can be as simple as a regular 2-wire POTS line to the local CO switch, or a 4-wire analog E&M trunk that provides improved signaling response (less glare). Channelized digital T1 trunks come in two main flavors. The first and oldest type of T1 can have 24 channels of 64 kilobit per second voice with robbed-bit signaling (signaling bits are stolen from the voice stream in a way that's not noticeable to the ear). This type of T1 sends much less signaling data but cannot be used with 64 kbps switched data because of the robbed bits used for signaling, but

can pass 56kbps switched data. ISDN T1 trunks have 23 channels of voice (bearer, or B channels) and a separate 64 kbps channel for signaling (the data, or D channel) that can support ISDN User Part (ISUP) messages, including Automatic Number Identification, which allows calling and called number information to be sent (although it can be spoofed; this is discussed in Chapter 3). In Europe and internationally, the E1 is the typical digital interface, with an ISDN BRI carrying 30 bearer channels (30B+D) as opposed to the 23 channels supported by ISDN over T1 (23B+D).

VoIP trunks also come in various flavors, including H.323, SIP, and proprietary protocols like Inter-Asterisk eXchange (IAX). In some cases, IP-enabled PBX systems also use gateway control protocols with VoIP trunks, such as Simple Gateway Control Protocol (SGCP), H.248/Megaco/Media Gateway Control Protocol (MGCP), Skinny Gateway Control Protocol. One of the difficult problems with VoIP trunks, however, is feature transparency between vendors. ISUP/Q.931 or its private line equivalent (QSIG) has the most complete feature interworking capability, and standards for mapping these onto H.323 and SIP exist, but these are not evenly supported by PBX vendors at this point. Robust, reliable interworking between different PBX vendors over VoIP is not easy to find today (and is still a challenge over private tie lines).

## PBX Features

PBX systems provide a plethora of features typically offered by a telephone provider, such as call waiting, three-way calling, conference calling, voicemail, additional call appearances, and many other routing features. Some vendors count 600 or more separate features among their capabilities, far more than is offered by any carrier on a central office switch as subscriber services. But often overlooked in this list are those used for access control. The PBX is effectively the firewall to the PSTN and because voice access has per-minute and geographic costs associated with each call, this aspect of PBX capability should be a critical consideration for product selection, configuration, and ongoing operations. Yet at the same time, the data security community is rarely concerned with this characteristic because it's not a pure data security issue, yet even in a VoIP system there will be PSTN connectivity; why gamble with this?

Say a company has 200 employees, each with a phone on their desk. Without a PBX, each employee would require their own pair of copper wires from the CO, each with their own phone number that routes to their desk. However, it's a safe bet that not all 200 employees will be on the phone all the time, and it's likely that most of those calls will be to other employees. This is where a PBX really pays off. A business or campus will need many fewer lines from the Local Exchange Carrier (LEC); in the previous example, the company might require only 40 outside lines, routing those calls onto the PSTN trunk lines as necessary on a per call basis. They also could rent 200 Direct Inward Dial (DID) numbers from the LEC, which terminate through those trunk lines. The PBX will then route the inbound call based upon which DID number was dialed to reach it.

## Tools & Traps...

### Asterisk: The Open-Source PBX

PBX servers were notoriously expensive to justify when an organization wasn't ready for a major capital outlay, plus they tended to rely on closed or proprietary architecture, which made PBX systems more expensive than they might otherwise have been. Then along came Asterisk, from the mind of Mark Spencer. Asterisk is an open-source PBX software package that runs on many operating systems, including Linux, BSD, Mac, and even Windows. Asterisk requires very little in the way of hardware, with old Pentium 100MHz boxes with 64MB of RAM still ample enough to power a small business. Aside from the relatively low hardware horsepower requirements, Asterisk doesn't necessarily need any additional hardware, aside from what's already in your computer. Utilizing the popular Session Initiation Protocol (SIP) and the Inter-Asterisk Exchange Protocol (IAX), two increasingly ubiquitous VoIP technologies, Asterisk can make and take calls completely over the Internet or operate with special hardware like PCI T1/E1 cards for PSTN connectivity. Users may purchase DIDs from the VoIP provider to dial in to their PBX from their normal phones, or they may dial in using a special software phone. We discuss softphones later in this chapter.

The appeal of a PBX system is obvious to not only businesses and campuses but also attackers, who have taken an increased interest in them as well, since most PBX systems can support trunk-to-trunk transfer (i.e., dial-out again from the PBX after coming in on another line). PBX security often is overlooked by enterprises until a big phone bill arrives, and oftentimes the hackers have no challenge at all when settings are never changed from the manufacturer's default. Try a Google search for "default password" and a PBX vendor and you'll see just how easy this information can be to obtain. It is important to note that because PBX vendors typically have provided detailed instructions on how to secure the PBX, the remaining security responsibility lies completely on the operator of the PBX system, and any toll charges that may be obtained by fraud are left to be paid by the PBX owner. Attackers who have compromised a PBX system may set up their own private conference room, a "party-line" where they may hang out and exchange illicit information on your dime.

Other features can be a double-edged sword as well. Many PBX systems also provide a call-monitoring feature for managers to supervise their agents (or to record calls). You know those recordings that go, "Your call may be monitored for quality assurance and training purposes"? Well, if you're not careful, they might also be monitored for humorous or larcenous purposes. And it may not be just calls to your call center that get monitored; if your monitoring system wasn't properly designed or an intruder gets access to PBX administration at a high enough level, any call can be monitored.

The bottom line when it comes to PBX features is that you need to read the associated security recommendations carefully. Some vendors have assembled detailed security guides for addressing toll fraud and feature access that are well over 100 pages, and you would be wise to find out what kind of documentation exists. And don't forget to back up your PBX regularly so that you don't lose the security policy you create! More critically, if a VoIP vendor does not have these kinds of capabilities, you would be wise to find out what can be done to reduce exposure to toll fraud. In some cases, the lack of feature-functionality in many VoIP solutions is a blessing because it reduces the opportunities for security-affecting misconfiguration. Yet at best this is a temporary benefit since VoIP solutions are becoming more sophisticated each and every year.

## Notes from the Underground...

### Toll Fraud

Attackers have discovered a myriad of ways to make all the long distance calls they want from your PBX system, leaving you with the hefty collect-call charges. Here are a few:

- Even with good security elsewhere, a caller can ask to transfer to extension to 9011 on a system where dialing 9 goes to an outside line and 011 is the international direct dial access code. Make sure your employees (particularly those that answer many external calls) know about this ruse and consider using your PBX's trace feature to track down the source of such calls (you can even have the call transferred to your security department as part of the trace feature).
- Attackers can read the same manuals online that your systems administrators can, and the smart ones will figure out how to get around the obvious restrictions. For instance, if trunk access codes aren't restricted, it really won't matter how well you've locked out other dial restrictions. And just because you don't use your local trunks for long distance doesn't mean an attacker won't.
- Adding support for IP softphones or WiFi phones to a PBX means that a softphone or wireless phone could be used by a remote attacker who can get onto your IP network (by wire or wireless) for toll fraud or other nefarious purposes. In this case, defense of your IP network overall is what will minimize exposure to the PBX, but it's important that the PBX not weaken overall IP security (by allowing WEP-based security on wireless networks shared by voice and data, for instance).



## PBX Adjunct Servers

Most PBX systems have an adjunct server or two, providing voice messaging or call center functionality that isn't part of the core PBX switching capabilities. The larger and more complex a network gets, the more demanding traffic becomes to the underlying hardware. Given the modularity of voice networks, we can off load some of this functionality to other hardware that can be set to handle a specific task, rather than attempt to do everything itself. Of course, this also complicates the overall security model, so make sure you know how this offloading impacts security.

## Voice Messaging

It's hard to remember that voicemail was once a completely optional capability for PBX systems, but it's still implemented as a separate server by most vendors using analog, digital, or IP trunks to integrate with the PBX. Some settings on that voice messaging server can open the door to fraud and abuse, so be sure to follow manufacturer recommendations for security—especially when it comes to changing default administrator passwords! Are mailboxes using strong enough PINs? Are old mailboxes closed down? Make sure you can answer these questions.

### Notes from the Underground...

#### Voice Messaging: Swiss Army Knife for Hackers?

Voice messaging is not without its share of security considerations, though. Many vendors ship voice mail systems with default passwords installed, which some users opt to never change. These passwords are often as simple as the number of the voice mailbox itself, or a simple string of numbers like 12345. Hackers love it when it's this easy to get in. But that's only the beginning when it comes to security attacks you may need to protect against within your voice messaging systems. Here are a few other scenarios:

- When attackers gain control over a compromised PBX system that supports DID and voice-mail, they might change the outbound greeting to something like "Hello? Yes, yes, that's fine." Or just "Yes (pause) yes (pause) yes..." They then call that number collect and the operator hears what appears to be someone more than willing to accept charges! Some PBX and voice-mail systems send a special tone when a line is forwarded to voice-mail that may discourage this tactic since a savvy operator would recognize the tone. Does your organization know what's happening with old or unused mailboxes?

Continued

- Another security issue can arise when mobile phone providers offer voicemail to their subscribers, but don't require a password to access messages when the voicemail server receives the subscriber ANI (indicating that subscriber is calling from the mobile phone associated with that extension). But by offering their users the "convenience" of quick access to their messages, these carriers may be opening the door to eavesdropping through ANI spoofing (which is discussed in more detail in Chapter 3) unless they have other means of verifying the origin of a given call.
- Eavesdropping on potentially confidential messages is certainly a threat, but an attacker may potentially hijack phone calls intended for a victim as well. This can be done by changing their outbound message greeting to say "Hi, this is Corey. Please call me at my new number at..." and leave a number that they control, performing a man-in-the-middle attack on the intended recipient.
- Another successful social engineering technique involves leaving messages within a voicemail system requesting passwords (for "testing" or "administrative purposes") on another internal extension, lulling the victim into believing that the attacker is a legitimate employee at the target company.
- The latest voice-messaging systems can be used to read e-mail using text-to-speech. Attackers know that a PIN for the voice messaging system is easy to guess, and this may be the easiest way for them to get to an e-mail system.
- And don't forget toll fraud that can happen through out-dial capabilities on voicemail systems. Consider turning off this feature if it isn't needed in your organization. Associated risks can also be mitigated through carefully crafted PBX dial policy.

## Interactive Voice Response Servers

Perhaps you first ran into an IVR when you noticed an incorrect charge on your phone bill, and you decide to speak with a customer service representative to clear things up. But when you dial the toll-free number on the bill, you're greeted with a labyrinth of options allegedly to help you self-navigate to the appropriate agent. This maze of menus is brought to you through an Interactive Voice Response (IVR) system. An IVR is a series of recorded greetings and logic flows that provide a caller with a way to route through the phone system as a means of convenience. Personal feelings about speaking with a recorded voice aside, IVRs are actually a pretty clever way of providing a caller with speedy call placement, taking much of the burden away from agents or operators.

Today's latest-generation IVR systems are built on VoiceXML interpreters, and may have sophisticated development environments. IVR security is a largely unexplored topic since each IVR system is like a unique application, but we occasionally hear about poorly written IVR applications that are insecure or not sufficiently robust.

## Wireless PBX Solutions

Several solutions for adding wireless extensions to PBX systems have been commercialized. Most PBX vendors have implemented proprietary 900 MHz-band solutions in the United States as well as the 1900 MHz Digital Enhanced Cordless Telecommunications (DECT) ETSI standard in Europe, which has driven widespread adoptions of vendor-neutral wireless there. More recently, a number of WiFi solutions have become available, as well as combination WiFi/GSM solutions that let a single device work with both Cellular and Enterprise PBX infrastructure. See the warnings about WEP later in this chapter.

## Other PBX Solutions

Two other PBX solutions with security considerations bear some discussion: Call Detail Recording (CDR) systems and Voice Firewalls. CDR systems enable every call on a PBX to be recorded after it is complete using a standardized format. This allows special reporting software to analyze this data for forensic or diagnostic purposes. It is worth noting, however, that a CDR system will not allow you to stop a fraudulent call still in progress. For this, you would need a voice firewall such as that sold by SecureLogix. Such a firewall allows you to see current calls in real-time, apply policy based on type of call (voice, fax, or data), and set notifications, authentication requirements, or other policy based on rules very similar to those you might set for data traffic on a data firewall.

## PBX Alternatives

Long before the appearance of VoIP, nonswitched alternatives to the PBX have been available. For systems of less than 50 users, Key Telephone Systems (KTS) share outside lines directly and have dedicated intercom lines to talk between stations. Current generation key systems are more PBX-like than ever, so it may be hard to find that distinction anymore. But older key systems won't support advanced switching features like trunk-to-trunk transfer that can lead to toll fraud. Still, so-called hybrid key systems should be treated like a regular PBX when it comes to security.

Centrex, IP Centrex, and Hosted IP-telephony services are carrier-based PBX alternatives that provide a private dial plan plus the more popular switching features that an on-site PBX system might. However, the switching equipment stays in the carrier's infrastructure and is managed by the carrier. This is a mixed blessing since it's likely to reduce the overall functionality and access policy tailoring available to you if your organization uses such a service, but it does mean that the carrier shoulders a larger share of the responsibility for any toll fraud that may result (and consequently won't provide high-risk services like trunk-to-trunk dialing without extra security measures).

More recently, the appearance of IP telephony has provided an opportunity for some manufacturers like Avaya to rearchitect their overall PBX approach and separate the functionality

once provided in a single device into multiple devices. In particular, call control and signaling can be separated from media processing and gateway services; this approach makes possible an architecture where a few call control servers can provide redundant services across an entire organization with media gateways located in every geographic location that contains their physical presence. We'll treat this approach along with other similar VoIP architectures in the next section.

## VoIP Telephony and Infrastructure

With the introduction of VoIP came a new architectural flexibility that in theory can completely distribute PBX functionality across an entire infrastructure. We'll review those concepts in this section and discuss examples of this in action, but keep in mind that few VoIP solutions take full advantage of every aspect described here (and it wouldn't surprise me to discover that none of them did, but today's VoIP market is moving so fast that it's difficult if not impossible to prove that kind of negative). Regardless, these concepts each have significant security implications.

### Media Servers

The term *media server* is totally overloaded in the VoIP world (and even more so within the IT industry as a whole). If we restrict ourselves to VoIP-related definitions only, a server so named still could be any of the following:

- Interactive voice response (IVR) server or media slave, possibly running VoiceXML or MRCP
- Signaling Media Server (Media Gateway Controller) to handle call control in Voice/VoIP network
- Call distribution (ACD) for receiving and distributing calls in a contact center
- Conferencing Media Server for voice, video, and other applications
- Text-to-speech server (TTS) for listening to e-mail, for instance
- Automated voice-to-e-mail response system
- Voice or video applications server
- Streaming content server
- Fax-on-demand server

Sure, some of these are similar and can roughly be grouped together, but at best you'll get this down to semi-overlapping groups that center on two general areas: interactive media services and call or resource control. The point here is that in the VoIP world, we haven't standardized architectures and naming conventions yet so we are left with technically vague

terms like *media server*, *media gateway*, and the worst offender, *softswitch* (a marketing term we will not spend more time on in this chapter except to note that it was intended to conjure up the image of a class 5 switch being displaced by a software blob that runs these media servers and media gateways but has become so overloaded that it has completely lost any technical meaning it once may have enjoyed).

## Interactive Media Service: Media Servers

On the other hand, there is another kind of media server that actually contains DSP resources that it uses to process speech or video (and perhaps one or more additional form of media). These may be involved with generating and receiving DTMF tones, executing the logic of an IVR system, converting text-to-speech or handling streaming or document content in response to speech or DTMF input. Or it may orchestrate multiway call traffic, conference calls, handle translation between codecs, or even fax processing. Media servers of this class may provide VoiceXML interpretation for interactive, dynamic voice applications.

## Call or Resource Control: Media Servers

This class of media server is responsible for managing communications resources at a higher level, such as handling call control while managing media gateways that have DSP and other gateway resources for the actual media manipulation. Most Media Servers support VoIP protocols but are likely also to support others as well, such as digital voice or video trunks, or even analog voice through media gateways. Examples of this kind of media server include call control servers from PBX vendors that control separate gateways, voice processing servers that manage and redirect DSP resources located elsewhere, and call distribution systems that manage off-board call handling resources such as switches and IVR systems.

### *The H.323 Gatekeeper*

This gatekeeper is the manager of one or more gateways, and is responsible for providing address translation (alias to IP address) and access control to VoIP terminals and gateways. A gatekeeper acts as the central authority for other gateways, allowing an administrator to quickly and authoritatively roll out changes across a voice network. Gatekeepers limit the number of calls at a given time on a network by implementing control over a proxy. A gatekeeper works something like this: A user wants to make a call to another user at a different physical location, and his phone registers with a local gateway. The gateway then passes on his call information to the gatekeeper, which acts as a central hub to other gateways and users. The gatekeeper then passes call setup information to the gatekeeper at the other office, which in turn hands it to the appropriate destination gateway, and finally to the desktop of the called party. Many call control media servers include an H.323 gatekeeper.

## *Registration Servers*

In a traditional PSTN or PBX switching system, where each user is at a fixed location, usually tied in place by copper wires, routing calls is (relatively speaking) simple. So-called find-me/follow-me services on PSTN or PBX switches can add PSTN mobility. Forwarding or extension-to-cellular features can increase this sense of mobility, but all these solutions require active user programming or rely on fixed forwarding algorithms and are rooted in the PSTN.

But with VoIP, a user can be geographically located virtually anywhere on the planet (as long as minimum QoS conditions are present). A registration server acts as a point of connection for mobile users. Johnny can log in to the registration server from his hotel room in Amsterdam with an unknown IP address and the registration server will let the gateways know where to route his traffic. That way, Johnny can keep the same phone number no matter where he is physically located. A similar example can be seen with instant messaging networks. A user can log in using his screen name from home and be reachable to the same users as if he had logged in from work. In the H.323 world, registration is a function of a gatekeeper; however, this can be a separate function in the SIP realm.

## *Redirect Servers*

A SIP redirect server acts as the traffic light at the VoIP intersection. Very much like a web page with a redirect tag built in, a redirect server will inform a client if the destination the caller is trying to reach had changed. Armed with the updated information from the redirect server, the client will then rerequest the call using the new destination information. This takes some of the load off proxy servers and improves call routing robustness. In this way, a call can quickly be diverted from a proxy, rather than require the proxy to complete the connection itself.

## *Media Gateways*

A gateway is a device that translates between protocols in general by providing logic and translation between otherwise incompatible interfaces. A voice or media gateway in particular tends to translate between PSTN (trunking) protocols and interfaces and local line protocols and interfaces (though that's not universally true). In addition, the potential protocols and interfaces that a voice gateway now might support include Ethernet and VoIP protocols as well. The voice gateway could have H.323 phones on one side and an ISDN trunk on the other (both digital) or a VoIP phone on one side and an analog loop to the carrier, or even VoIP on both sides (say, H.323 to the station and SIP trunking to the carrier). The point is that there are literally hundreds of different equipment classes that all fall under the voice gateway moniker and thousands of classes that fall under gateway to begin with.

One class of VoIP media gateway connects traditional analog or digital phone equipment or networks to VoIP equipment or networks. A simple home-user implementation of a VoIP

gateway like this is an ATA, or Analog Telephone Adaptor. At a minimum a VoIP media gateway will have both a phone interface (analog or digital) and an Ethernet interface. For an ATA, a regular analog phone is connected to the adaptor, which then translates the signal to digital and passes it back over the Ethernet. Of course, media gateways can get much more complex than this. PBX vendors have split out the line-card cabinet portion of their product and recast it as a media gateway, with the gateway under the control of a media server. IP routing companies have added analog and digital voice/video interfaces to routers and recast them as media gateways. And in many respects these products do contain overlapping functionality even though they may not be equivalent.

## Firewalls and Application-Layer Gateways

Within a firewall, special code for handling specific protocols (like ftp, which uses separate control and data paths just like VoIP) provides the logic required for the IP address filtering and translation that must take place for the protocol to pass safely through the firewall. One name for this is the Application Layer Gateway (ALG). Each protocol that passes embedded IP addresses or that operates with separate data (or media) and control streams will require ALG code to successfully pass through a deep-packet-inspection and filtering device. Due to the constantly changing nature of VoIP protocols, ALGs provided by firewall vendors are constantly playing a game of catch-up. And tests of real-time performance under load for ALG solutions may reveal that QoS standards cannot be met with a given ALG solution. This can cause VoIP systems to fail under load across the perimeter and has forced consideration of VoIP application proxies as an alternative.

## Application Proxies

A Proxy server acts as a translator for transactions or calls of different types. If Johnny's phone speaks IAX and Jen's phone speaks only SIP, the proxy sits between them and translates the message as necessary. Even if both sides speak the same protocol, be it HTTP or SIP, there are security or NAT or other boundaries that call for either a proxy or packet manipulation in an Application Layer Gateway (ALG) within a firewall. The benefit of an application proxy is that it can be designed specifically for a protocol (or even a manufacturer's implementation of a protocol). In addition to allowing boundary traversal, a proxy can also be used as a means of access control, ensuring that a user has the rights to place a call before allowing it to proceed. And the best proxies can even guard against malformed packets and certain types of DoS attacks. Depending on the complexity of your call requirements, a proxy may be integrated into a PBX or Media Server, or it may be an entirely different piece of hardware.

## Endpoints (User Agents)

In a phone system, an endpoint on the network was known as a terminal, reflecting the fact that it was a slave to the switch or call-control server. But today's endpoints may possess much more intelligence, thus in the SIP world the term User Agent is preferred. This could be a hardware IP telephone, a softphone, or any other device or service capable of originating or terminating a communication session directly or as a proxy for the end user.

### *Softphones*

With the advent of VoIP technology, users are able to break free of classical physical restrictions of communication, namely the special-purpose telephone terminal. A softphone is a piece of software that handles voice traffic through a computer using a standard computer speaker and microphone (or improved audio equipment that is connected through an audio or multimedia card). Softphones can emulate the look and feel of a traditional phone, using the familiar key layout of a traditional phone and often even emulating the DTMF sounds you hear when you dial a call. Or it may look more like an instant messaging (IM) client, and act like audio chat added to IM.

In fact, a softphone doesn't even need a computer microphone or speaker: my favorite doesn't need to send media through the computer at all in telecommuter mode—it just uses H.323 signaling to tell my media server which PSTN number (or extension) to dial for sending and receiving the audio. This lets me turn any phone into a fully featured clone of my work extension without regard to QoS available to me on my Internet connection.

Because a soft phone resides on a PC, the principle of logically separating voice and data networks is defeated as the PC must reside in both domains. You will need to consider this trade-off as you design appropriate security policy for your VoIP network, although the long-term trends favor voice-data integration, so at best maintaining physical separation can be only a temporary strategy.

Consumer softphones have exploded over the past few years and nothing is hotter than Skype in that space. Skype is the brainchild of the people who brought us the Kazaa file sharing framework. Utilizing peer-to-peer technology and an encrypted signaling and media channel, Skype has proven to be both easy to set up and use securely by end users, while simultaneously being a thorn in the side of network administrators. Because it aggressively jumps past firewalls to create call traffic, it is considered to be a threat by many enterprise security groups.

One of Skype's major enhancements over instant-messaging-based voice is its superb codec, which is actually better than that used within traditional telephone infrastructure. This provides superior call quality when contacting other Skype users. Another major benefit of Skype is the ability to reach any phone in the PSTN by way of SkypeOut gateways. With its PSTN gateway, Skype has become an attractive alternative for small overseas call centers and other Internet businesses.



## Are You Owned?

### Consumer Softphone Gotchas

Many consumer-oriented softphones contain advertising software that “phones home” with private user information. Several popular softphones (such as X-Lite) store credentials unencrypted in the Window’s registry even after uninstallation of the program. Softphones require that PC-based firewalls open a number of high UDP ports as part of the media stream transaction. Additionally, any special permissions that the VoIP application has within the host-based firewall rule set will apply to all applications on that desktop (e.g., peer-to-peer software may use SIP for bypassing security policy prohibitions).

Also consider that malware affecting any other application software on the PC can also interfere with voice communications. The flip-side is also true—malware that affects the VoIP software will affect all other applications on the PC and the data services available to that PC (a separate VoIP phone would not require access to file services, databases, etc.).

### *IM Clients*

Instant messaging is perhaps the dominant means of real-time communication on the Internet today. IM’s roots can be traced back to the Internet Relay Chat (IRC) networks, which introduced the chat room concept but did not track online presence and never reached the popularity of IM. Just as IM is the next logical step from IRC, voice chat is the next leap from text-based chat. Most of today’s most popular IM clients have included voice functionality, including AOL’s Instant Messenger, Yahoo! Messenger, and MSN Messenger. Skype took the opposite approach and created a chat client that focuses on voice as the star and text chat as an afterthought. Even Google jumped aboard the IM bandwagon, releasing Google Talk. Let’s take a look at these clients to see what makes them similar, and what makes them different.

AIM, AOL’s IM service, surely wasn’t the first on the scene, but it has the largest base of users. Initially AIM was limited to users of the AOL Internet service, but eventually it was opened up to the Internet as a whole. With the addition of a proprietary voice capability in late 1999, AOL was a VoIP pioneer of sorts. (although voice chat was first available through Mirabilis’s ICQ). Yahoo! Chat jumped aboard the voice bandwagon soon after, and Google’s more recent client has included voice from the beginning. In 2005, Yahoo announced interoperability with Google and MSN (who also has a voice chat plug-in for messenger that is also used with its Live Communication Server product). In addition, Microsoft’s popular

Outlook e-mail client (and entire Office suite in the case of LCS) can be linked to Microsoft Messenger. Also worth mentioning is the Lotus Domino IM client that competes with Microsoft LCS in the enterprise instant messaging (and presence) space, as well as Jabber, which can be used to tie together both public and private IM services using the XMPP protocol.

Google Talk is the newest comer to the IM game. Though Google Talk is still in its infancy, it stands to succeed due largely to a philosophical stand point, embracing open standards over proprietary voice chat. Google Talk aims to connect many different voice networks over a series of peering arrangements, allowing users to minimize their need to run several IM clients. Like Skype, Google seeks to bridge traditional phone calls with Internet telephony, promising to federate with SIP networks that provide access to an ordinary telephone dial tone. Google recently released a library called libjingle to programmers, allowing them to hack new functionality into Google Talk. It will be interesting to see where Google takes Google Talk in the future.

### *Video Clients*

Most of us can probably think back and recall seeing episodes of *The Jetsons* when we were younger. Or pictures of the AT&T PicturePhone from the 1964 World's Fair. Movies have all but promised these devices to be a staple of every day life in the future. And for decades, the video conference has been pushed by enterprises seeking to save money on travel (though investments in video conferencing equipment tend to sit around gathering dust). Live video on the Internet has its adherents, and today we see yet another wave of marketing aimed at the business use of video. So, will video finally take off around VoIP just like audio, or is there something different going on here?

The video phone has been tomorrow's next big technology for 50 years but the issue has been more sociological than technological. Certainly, popular instant messaging clients have included video chat capabilities for some time now, although each client typically supports only video between other users of the same client or messaging network. And although it always gives me a kick to see someone else announcing that they've solved the gap with technology, the point is well taken that video is here to stay in VoIP systems—even if it doesn't get as much use as VoIP.

The latest on the video bandwagon is the Skype 2.0 release. At only 15 frames per second and 40 to 75 kbps upload and download, Skype Video works well on a standard home DSL line or better. Other popular IM clients with video include Microsoft's Messenger and Yahoo Instant Messenger. AIM now offers video as well.

H.323-based IP videoconferencing systems have been available in hardware and software from many sources for almost a decade at this point, so there's no shortage of vendors in this space. And SIP video phones are available from many of these same vendors and from startup companies in the SIP space.

## *Wireless VoIP Clients*

Over the past few years, an explosion of wireless VoIP solutions has hit the marketplace. Most of these solutions are immature and if broadly deployed can completely overrun the available bandwidth on 802.11b (or g) networks that were not engineered for high-density voice, even with QoS prioritization. And although 802.11a networks can handle higher wireless VoIP densities, they present other backward-compatibility issues of their own. And we haven't even gotten to the security issues yet! Still, the promise of WiFi VoIP is tantalizing, and most enterprises that have deployed VoIP solutions seem to have experimented with it. The idea of a combined cellphone/WiFi phone (and maybe PDA too) seems just too compelling to ignore, even if power consumption issues sideline keep the concept sidelined in the short term.

## IP Switches and Routers

Although their position is defined by a standard data network rather than VoIP, a router's purpose in life is to connect two or more IP subnetworks at layer 3. An IP switch performs a similar function at layer 2. Routers and switches operate on the network and data-link layers, respectively, investigating the IP address or MAC address for each packet to determine its final destination and then forwarding that packet to its recipient. For VoIP, the biggest consideration at these levels are QoS markings and treatment such as DiffServ and RSVP, which should be supported by this infrastructure in a way that allows legitimate voice packets through with high priority and shuts out malicious packets, particularly those aimed at causing DoS attacks. This may be easier said than done in some cases. If an attacker can inject QoS-marked packets into your network, will your QoS scheme create a DoS condition for both voice and data?

## Wireless Infrastructure

Wireless access points and associated infrastructure are similarly considered an extension of the data network. However, the increasing use of VoIP clients within this infrastructure creates several unique security considerations (particularly DoS given that wireless is a shared medium). In addition, wireless VoIP devices in the marketplace have lagged in implementation of the most current wireless encryption recommendations. All this should be taken into consideration in the design and operation of wireless VoIP.

## Wireless Encryption: WEP

When wireless networking was first designed, its primary focus was ease of implementation, and certainly not security. As any security expert will tell you, it's extremely difficult to secure a system after the fact. WEP, the Wired Equivalent Privacy encryption scheme, initially was targeted at preventing theft-of-service and eavesdropping attacks. WEP comes in two major varieties, standard 64-bit and 128-bit encryption. 256-bit and 512-bit implemen-

tations exist, but they are not nearly as supported by most vendors. 64-bit WEP uses a 24-bit initialization vector that is added to the 40-bit key itself; combined, they form an RC4 key. 128-bit WEP uses a 104-bit key, added to the 24 bit initialization vector. 128-bit WEP was implemented by vendors once a U.S. government restriction limiting cryptographic technology was lifted.

In August of 2001, Fluhrer, Mantin, and Shamir released a paper dissecting cryptographic weaknesses in WEP's RC4 algorithm. They had discovered that WEP's 24-bit initialization vectors were not long enough, and repetition in the cipher text existed on busy networks. These so-called weak IVs leaked information about the private key. An attacker monitoring encrypted traffic long enough was able to recreate the private key, provided enough packets were gathered. Access Point Vendors responded by releasing hardware that filtered out the weak IVs.

However, in 2004 a hacker named Korek released a new statistical-analysis attack on WEP, which led the way to a whole new series of tools. These new wireless weapons broke WEP using merely IVs, and no longer just IVs were considered weak. On a 64-bit WEP encrypted network, an attacker need gather only around 100,000 IVs to crack in (although more certainly increases the chance of penetration) and only 500,000 to 700,000 for 128-bit WEP. On a home network, it can take days, even weeks to see enough traffic to make cracking the key possible. However, clever attackers discovered a way to stimulate network traffic by replaying encrypted network level packets at the target. By mimicking legitimate network traffic, the target network would respond over and over, causing a flood of network traffic and creating IVs at an accelerated rate. With this new attack, a 128-bit WEP network can be broken in as little as 10 minutes.

## Wireless Encryption: WPA2

WPA, WiFi Protected Access, was created to address overwhelming concerns with WEP's inadequacy. WPA uses RC4; however, it uses a 128-bit key appended to a 48-bit initialization vector. This longer key defeats the key recovery attacks made popular against WEP using the Temporal Key Integrity Protocol (TKIP), which changes keys mid-session, on the fly. Additionally, the Message Integrity Code (MIC) includes a frame counter in the packet, which prevents the replay attacks that cripple WEP.

WPA2 was the child of the IEEE group, their certified form of 802.11i. RC4 was replaced by the favorable AES encryption scheme, which is still considered secure. WPA's MIC is replaced by CCMP, the Counter Mode with Cipher Block Chaining Message Authentication Code Protocol. CCMP checks to see if the MIC sum has been altered, and if it has, will not allow the message through.

Perhaps the most beneficial attribute of WPA2 is its ease of implementation. In most cases, hardware vendors needed only reflash the firmware of their Access Points to allow for WPA2 compatibility.

Although considerably stronger than its older brother, WEP, WPA2 is not without guilt. WPA2 encrypted traffic is still susceptible to dictionary attacks since WPA2 uses a hashing algorithm that can be reproduced. Joshua Wright released a tool called coWPAtty, which is a brute-force cracking tool that takes a list of dictionary words and encrypts them using WPA2's algorithms, one at a time. The encrypted value of each word then is compared against the encrypted value of captured traffic, and if the right password is found, POOF! The packet becomes intelligible.

Although brute-force cracking is not guaranteed to yield results, it leverages a weakness found in almost all security mechanisms—the user. If a user chooses a password that is not strong enough, or uses semipredictable modifications (the use of the number 3 instead of “e”), the network will fall. It is recommended that users install a pass-phrase instead of a traditional password. A pass-phrase longer than eight characters, which includes nonalphanumeric characters, is much less likely to be discovered by brute-forcing methods. And never, ever, use a dictionary word as a password, as these will often be discovered within minutes using freely available software from the Internet.

When implementing wireless VoIP, always use WPA2 or use an alternative means for protecting the VoIP stream (i.e., media and signaling encryption or IPSEC tunneling). Given the speed with which WEP can be cracked, it's almost pointless to use it since it adds encryption latency and creates a false sense of security.

## Authentication: 802.1x

802.1x is an authentication (and to a lesser extent, authorization) protocol, whereas WEP/WPA are encryption protocols. And although 802.1x can be used on wired networks as well, it is most common today on wireless networks. It acts as an added layer of protection for existing wireless security implementations like WEP or WPA2 by requiring additional authentication to join a network beyond the shared secret associated with the encryption key.

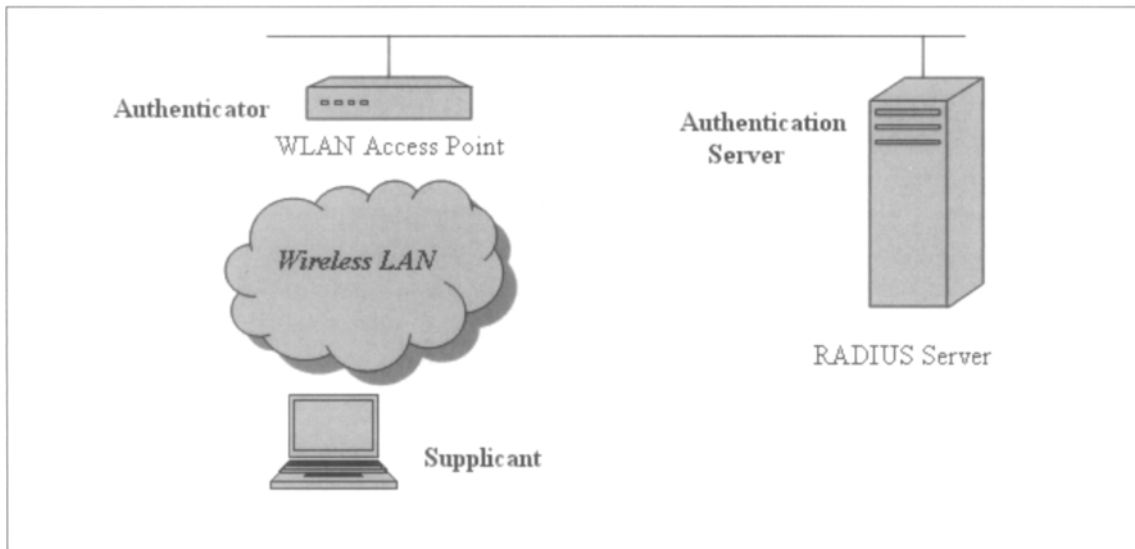
802.1x works by forcing users (or devices) to identify themselves before their traffic is ever allowed onto the network. This happens through the use of the Extensible Authentication Protocol (EAP) framework. EAP orchestrates password negotiation and challenge-response tokens, coordinating the user with the authentication server. 802.1x sticks the EAP traffic inside of Ethernet, instead of over PPP, a much older authentication protocol used all over the Internet. Keep in mind that there are a lot of different EAP methods available, so when you are comparing vendor support for 802.1x in infrastructure and VoIP devices you need to pay careful attention to the specific methods supported.

As soon as the access point, called an *authenticator*, detects that the link is active, it sends an EAP Request Identity packet to the user requesting access, known as the *supplicant*. The user then responds with an EAP Response Identity packet, which the authenticator passes to the authentication server, who grants or denies access (see Figure 2.2).

Think of the supplicant as the guy trying to get into “Club WLAN” who asks the guy at the door if he's on the list. The authenticator then flags down the bouncer (authentication

server) to see if he's "on the list." If he is, the bouncer lets him in to party with the rest of the party-packets. If not, it's to the curb he goes!

**Figure 2.2** A Basic 802.1x Implementation for a Wireless Network\*



\* If this were a wired 802.1x solution, the supplicant would be connected directly to the authenticator (typically a LAN switch).

Because of its moderately complex nature, 802.1x is not as quick to catch on with home users. The involvement of an authentication server (such as a RADIUS server) puts this technology just out of reach for most. However, 802.1x is ideal for businesses and public hot spots looking for more security than WEP or WPA2 alone provide.

## Power-Supply Infrastructure

Often overlooked as part of the infrastructure required for secure VoIP is how power issues will be addressed. PBX and PSTN phones run on a common battery system that provides availability for free in the face of a power outage, but VoIP phones and the infrastructure that powers them must be carefully designed to meet equivalent requirements.

## Power-over-Ethernet (IEEE 802.3af)

As the name implies, Power-over-Ethernet (POE) eliminates the need to run a separate power supply to common networking appliances. POE works by injecting power using a switch or special power injector that pushes Direct Current (DC) voltage into the CAT5 cable. POE can be used directly with devices specifically designed for POE or with other

DC-powered devices with a converter installed. This converter, called a picker or a tap, diverts the extra voltage from the CAT5 cable and redirects it to a regular power jack.

The major advantage of POE is that it allows greater flexibility in installing networking equipment. Access points can be set up in remote locations that normally would be limited to its proximity to a power outlet. It's often easier to route cat5 cable outdoors (on an antenna or in a tree, for instance) when only network cable is required. POE is also very popular with supplementary low-power devices, such as IP telephones and webcams, even computers!

POE is regulated by the IEEE 802.3af standard. This standard dictates the device must provide 48 volts of direct current, split over two pairs of a four-pair cable. The maximum current is limited at 350 mA and a maximum load of 16.8 watts. Several vendors have created proprietary (prestandard) implementations of POE, however in most cases newer equipment from these vendors is now available that is compliant with the IEEE standard (although at least one of these vendors now advertises an ability for the client to request a lower or higher amount of current through a proprietary process of negotiation above and beyond specifications within the standard).

To properly address VoIP phone availability concerns using POE, be sure that the power injector, network equipment, and voice servers (and gateways) can all operate on battery power for a sufficient length of time, and consider use of a generator when appropriate.

POE in action is pretty simple. The power source checks to see if the device on the other end of the wire is capable of receiving POE. If it is, the source then checks to see on which pairs of wires the device will accept power. If the device is capable, it will operate in one of two modes, A or B. In mode A, power is sent one way over pins 1 and 2, and is received over pins 3 and 6. In mode B, power is sent over pins 4 and 5 and is received over pins 7 and 8. Although only one mode will be used at a time, a device must be able to use both A or B to be IEEE 802.3af compliant.

## UPS

No availability strategy can be considered complete without appropriate use of Uninterruptible Power Supply (UPS) technology. Mission critical equipment such as PBX systems and servers need to be protected from unscheduled power outages and other electrical maladies. Because of the sensitive nature of electronic equipment, safeguards need to be put in place to ensure the safety of this equipment. A UPS protects against several availability threats:

- **Power surges** When the power on the line is greater than it should be, the UPS acts as a buffer, ensuring that no more power reaches the machine than is supposed to. If a power surge were to occur without a UPS inline, sensitive electronics literally could be zapped out of life.

- **Partial loss of power** A brownout occurs when the power on the line is less than is required to run an appliance. In many cases a brown out is considered to be more dangerous than a total power failure, as electrical circuitry is very sensitive to power requirements.
- **Complete loss of power** A blackout occurs when power is completely lost to an area. This is very common during natural disasters, where severe weather may topple the electrical infrastructure of an area. Gas or battery powered UPS systems allow for equipment to continue functioning for a set period of time after the lights have gone out. This is ideal for finicky gear that needs to be completely shut down before going dark, lest system integrity be compromised.

In a call-center environment, downtime to the phone system can be fatal to business. With a properly implemented disaster recovery plan including a network of UPS devices, the phones can continue to work when standard computer systems might not be able to. This may mean the difference between success and doom for some companies.

## Energy and Heat Budget Considerations

Given the heat and energy crisis being faced in many data centers due to the rapid increase in equipment densities (without a corresponding decrease in energy efficiency), planning for VoIP availability must include consideration for heat and power capacities in the room where VoIP servers and gateways will be housed. Don't omit this step only to discover after you've deployed that you have no power or cooling headroom for the additional equipment!



## Summary

VoIP hardware infrastructure reflects the hybridization of two worlds that are colliding:

- A specialized voice infrastructure based on the PBX and central office circuit-switching paradigm
- A general-purpose data infrastructure based on large-scale proliferation of software-based communication solutions running over packet data networks

In order to address VoIP security, a detailed knowledge of both models is essential. As more people and organizations deploy VoIP solutions, securing that infrastructure will become more crucial than ever before. Security must be considered from the design phase in every component.

## Architectures

### Solutions in this chapter:

- PSTN: What Is It, and How Does It Work?
- PSTN Call Flow
- PSTN Protocol Security
- The H.323 Protocol Specification
- The Primary H.323 VoIP-Related Protocols
- H.235 Security Mechanisms
- Understanding SIP
- SIP Functions and Features
- SIP Architecture
- Instant Messaging and SIMPLE

# Introduction

In this chapter we discuss the architecture of the Public Switched Telephone Network (PSTN), the architecture of networks based on the H.323 protocol, and the architecture of IP networks based on the Session Initiation Protocol (SIP). It's essential to include the PSTN and its associated risks when examining VoIP security. The PSTN has evolved considerably in recent years, but the addition of VoIP services also has created new and novel vulnerabilities for both data and voice. H.323 and SIP are signaling protocols—that is, they are involved in call setup, teardown, and modification.

## PSTN: What Is It, and How Does It Work?

Today, the PSTN is the most broadly interconnected communications system in the world, and is likely to remain so for at least another decade or more. For voice, it has no equal. VoIP services like Skype have banked on this fact; their business model depends on a steady flow of PSTN interconnect charges. But the PSTN provides FAX, data, telex, video, and hundreds of other multimedia services as well. And for many decades, the PSTN has enjoyed a universal numbering scheme called E.164. When you see a number that begins with “+” and a country code, you are seeing an E.164 number. In most of the world, connectivity to the PSTN is considered as essential as electricity or running water. Even the Internet itself depends on the PSTN to deliver dedicated access circuits as well as dial-up.

In the early days, wired communications at its most advanced meant two (or more) devices sharing a single iron wire, whether you were using a telegraph or telephone. A grounded wire to earth completed the circuit running between phones, each with its own battery to generate the current necessary to transmit. It was noisy and lines couldn't run very far, and it would be many decades before it could truly be called a global network, much less a national one.

To fully define today's PSTN, we'll need to focus on several areas in turn. First, the physical “cable plant” required for signal distribution, from twisted-pair copper and coaxial electric to the latest fiber-optic cabling. Second, its signal transmission models, combining analog and digital signal processing and transmission over electrical, optical, and radio interfaces. This directly affects the kinds of content it can carry. Third, the increasing sophistication of associated signaling (control) protocols and “intelligent network” design introduced with the Integrated Services Digital Network (ISDN). And finally, its associated operational and regulatory infrastructure on international, national, state, and local levels.

## PSTN: Outside Plant

The original premise behind the telephone exchange or Central Office (CO) was to run only one wire or set of wires into each house and have a centrally located facility for switching connections via operator (or automated equipment). Even though new homes

today may see six or more wire pairs, plus a coaxial cable for broadband cable television, the basic principle remains the same: each line to the customer forms a loop that passes through to the CO.

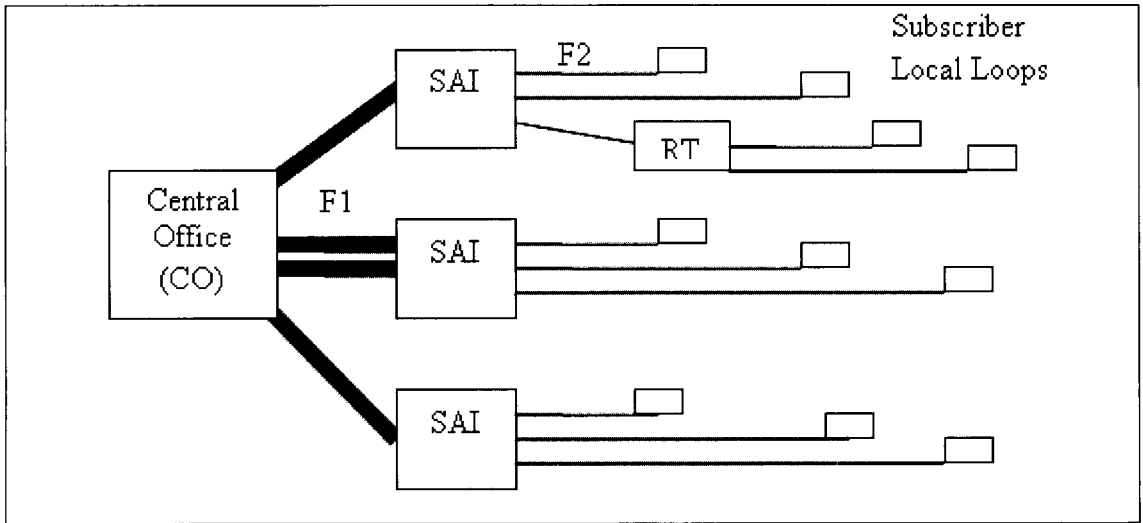
The collection of cabling and facilities that support all local loops outside the CO (or “wire center”) is known as the “loop distribution plant” and is owned by the Local Exchange Carrier (LEC). It starts out from the CO in a large underground cable vault with primary feeder cable (F1) to reach out over copper (or fiber) to the Serving Area Interface (SAI) for that area (look for a large grey or green box with doors mounted on a concrete pedestal in most areas of the United States). F1 cable is typically 600 to 2000 or more pairs and usually must be buried because of its weight (although fiber-optic F1 cable can be aerial if needed). It often is armored or pressurized and generally is enclosed in a concrete trench all the way to the CO, with manholes or other access points at least every 750 feet to allow for installation of repeaters (for digital trunks like the T1), loading coils, and other necessary equipment. In most of the world, the LEC is able to keep F1 and SAI fairly secure through physical locks, alarms, and so on.

At the SAI, F1 feeds are cross-connected to secondary feeder cable (F2) that goes out over copper underground to pedestal boxes where the distribution cable is split out or on poles to aerial drop splitters. Subscriber drop wires are then cross-connected to the F2 at that point. In rural areas, even lower-level cable facilities (F3, F4, F5) may exist before a drop wire is terminated. A box is installed where the drop wire is terminated outside the subscriber’s premises and this box is considered the demarcation point for the LEC. All wiring from there to the CO is the responsibility of the LEC, and from there to the phone devices themselves is the subscriber’s responsibility (or that of the landlord). Physical security of that inside wiring—particularly in shared facilities—can be an issue in some cases. And F2 or lower feeds and pedestals are not well secured in general (and present the biggest opportunity to an eavesdropper).

Where growth or other planning challenges have exhausted the supply of F1 or F2 pairs, it’s sometimes necessary for the LEC to install Remote Terminal (RT) equipment (sometimes called “pair gain” systems) that can multiplex multiple local loops on to a digital T-carrier (using Time-Division Multiplexing (TDM) over a 4-wire copper or pair of fiber-optic cables), or via older Frequency-Division Multiplexing (FDM) systems. RT units generally are locked and alarmed, however. And it is much more difficult to eavesdrop on a digital trunk (such as a T-carrier) or FDM system because of the costly equipment required. Figure 3.1 shows a diagram of a central office equipped with outside distribution plant (ODP).

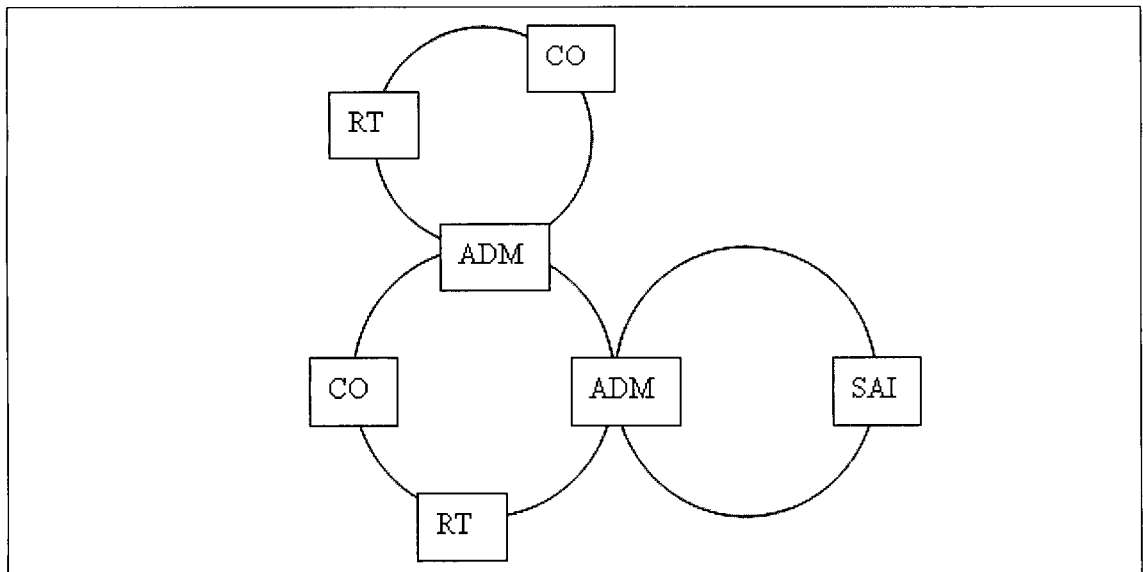
In addition to the loop distribution plant, the LEC will have outside plant for trunking between central offices, and the LEC and other Inter-exchange Carriers (IXCs) will have outside plant for long distance connections between COs and other switching centers such as toll centers. And the LEC or other Competitive Local Exchange Carriers (CLECs) may run fiber for SONET (or SDH) rings (see Figure 3.2).

**Figure 3.1** The Central Office with ODP\*



\* This classic example assumes no fiber is in use to these SAIs within the CO (see SONET example in Figure 3.2).

**Figure 3.2** A Modern SONET Ring Example



The diagram in Figure 3.2 shows that by using path diversity for fiber-optic routes along with SONET rings with Add-Drop Multiplexers, several self-healing SONET rings provide F1 and some F2 subscriber loop feeds as well as trunking between two central offices. Large

business customers can also connect to this SONET ring for high-capacity voice and data services if they are located close enough to the buried fiber.

## PSTN: Signal Transmission

In the old days, the path an analog voice signal took from your phone to the CO switch (or switchboard) was simple. With the appropriate cross-connects, each local loop was half of the analog circuit required for a phone conversation, and the switch (or operator) simply connected you with a calling or called party that represented the other half of that circuit. Although loading coils might have been used to reduce signal attenuation on the circuit, no amplification or signal processing was used.

Since Bell's original invention, several improvements had been added. Common battery from the CO with a separate return path instead of the earth eliminated the need for a battery in each phone and made the phone less noisy. Ringing was accomplished through magnetos, first added to the phones themselves and later pulled in to the CO and standardized as 90 Volts of Alternating Current (AC)—all other phone/PSTN functions on the line use Direct Current (DC). And eventually, automated electromechanical switching eliminated much of the need for an operator within the PSTN.

Still, analog transmission and switching had their limits. Until 1915, it wasn't possible to go much further than 1,500 miles on an analog long-distance circuit. And even when that limit was broken thanks to the vacuum-tube amplifier, these long-distance calls were very noisy. Radio telephony overseas and to ships further expanded the reach of analog telephony in 1927. And Frequency Division Multiplexing techniques were developed in the late 1930s that allowed many calls to pass over a single voice circuit by using frequency shifting techniques equivalent to those used by FM radio. Each 4 kHz band of voice conversation would be shifted up or down to a specific slot, allowing many calls to be carried simultaneously over a single coaxial cable or radio interface. By the 1950s, 79% of the inner-city CO trunks in the United States were using FDM. But even the microwave systems in use since the 1950s were analog systems.

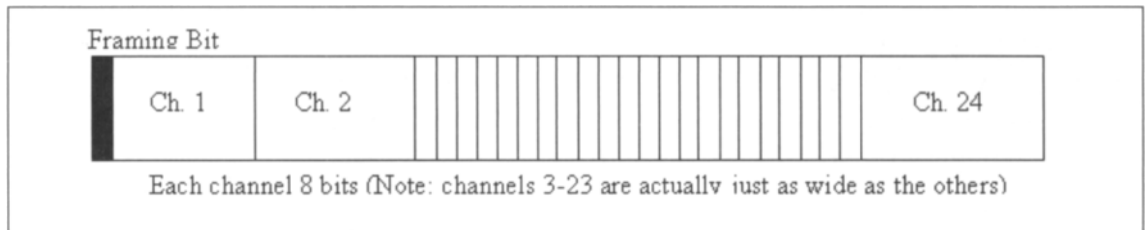
## T1 Transmission: Digital Time Division Multiplexing

Even though Alec Reeves of Britain had developed Pulse Code Modulation (PCM) techniques in 1937 for digitizing audio signals, and Bell labs had invented the transistor in 1948, which was required for the large-scale implementation of digital techniques, it would take more than a decade to make digital transmission a reality (and longer still before the advent of digital switching could make the full signal path digital outside the local loop). 1963 brought the introduction of the T1 or Transmission One digital carrier using revolutionary signal manipulation techniques that would forever change telephony.

Unlike all previous carriers, the T1 started in an all-digital format, meaning that it was structured as a series of bits (193 per frame to be exact, 8 bits per channel, 24 channels, plus

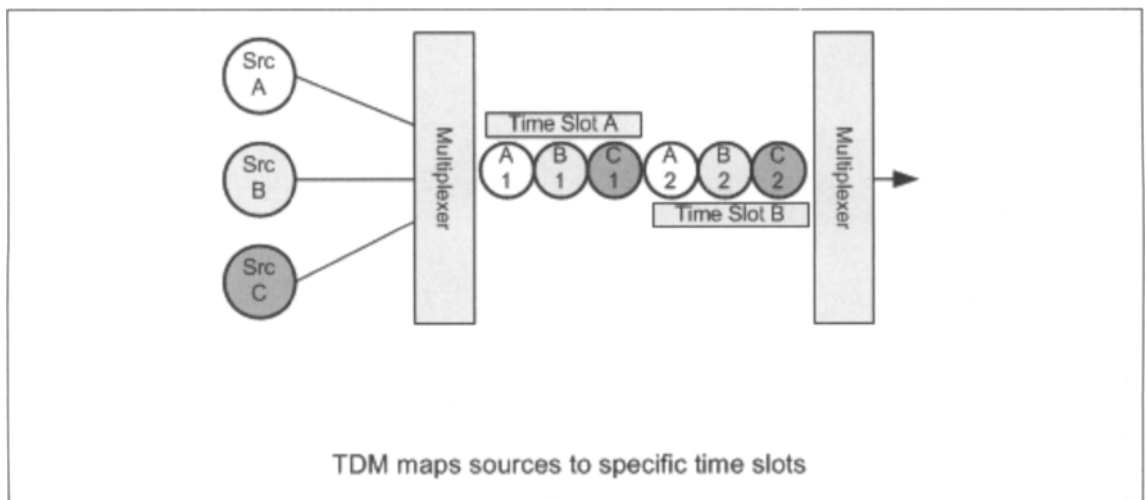
the framing bit—moving at the rate of 8,000 frames, or 1,544 Megabits per second) that by design could be completely regenerated again without data loss over long distances (see Figures 3.3 and 3.4). This provides a 64-kilobit-per-second digital bitstream for each of the 24 channels, using Time Division Multiplexing (TDM).

**Figure 3.3 A T1 Frame\***



\* Eight bits in each channel capture a 125 $\mu$ s slice of each associated analog audio signal.

**Figure 3.4 Time Division Multiplexing**



TDM as introduced in the T1 is the multiplexing workhorse of the telecommunications world and will be the base multiplexing environment for the rest of our discussion of the PSTN. Yet for the T1 to be successful, it is just as important to have a foolproof way of converting an analog signal to digital bits that would make or break the new form of digital transmission. This is the job of a codec. Although today in the era of digital media we take for granted the engineering required to create the first effective PCM codec—now commonly known as G.711—it was no small feat in its day. Yet, even today as debate rages over what codec is best to use for VoIP, G.711 is still considered the “toll quality” standard that

others must beat, and is especially good at preserving modem and FAX signals that low-bandwidth codecs can break.

## NOTE

Although we're not going to do a deep dive on digital/analog conversion here, it is worth pointing out that slight differences between U.S. and European standards will mean that some conversion needs to take place even within a standard G.711-encoded channel in order for that channel to move from a T1 to an E1 or vice versa. Specifically, slight differences in PCM encoding algorithm ( $\mu$ -law vs. A-law) may require conversion when voice or VoIP streams cross international boundaries. Of course, on a data circuit, that conversion is not going to happen automatically (if it did, it would scramble the data). But it can cause problems across a VoIP if you're not careful.

Similarly, when using a T1 circuit for data, it's important to make sure the circuit is properly configured since some signaling modes can use what's called "robbed-bit" signaling, which is fine for circuit-based voice but will corrupt data running on it. For this reason, only 56K of the 64K channel could be used for data on early data circuits. Today, clear channel data can be provisioned that uses a full 64K channel.

Back to the codec issue, however. It's worth pointing out that very complex trade-offs exist in codec selection and they're not as simple as quality vs. bandwidth. Some codecs require much more processing, others work poorly with modems, faxes, and other nonvoice applications (particularly low bandwidth codecs: it's not hard to imagine the problems inherent with sending a 56 Kbps modem signal through a 4Kbps voice-optimized codec. Even the best compression algorithms would struggle to represent that much information in so few bits, not to mention the inherent distortion present in D/A-A/D conversion.

Starting with the introduction of the T1, timing became an important consideration for the PSTN. Digital circuits like the T1 must be plesiochronous, meaning that their bit rate must vary only within a fairly limited range or other problems can be created within the PSTN. In comparison, analog circuits are completely asynchronous. This requirement has forced a hierarchy of master clocks to be incorporated into its infrastructure.

With the advent of SONET, a fully synchronous solution to the timing problem has arrived, along with massive bandwidth that can be further enhanced with Wavelength Division Multiplexing (WDM—basically the use of different colored light on a single optical fiber to increase capacity). Pointers and bit-stuffing in SONET and SDH are used to minimize the impact of clock drift between digital circuits, though the advent of VoIP has



created some challenges because VoIP is asynchronous. VoIP is also a packet technology (since it runs on packet networks), so it is subject to variations in latency and jitter and packet loss that are simply not significant issues in circuit networks because timeslots are guaranteed. On the other hand, the PSTN's circuit network is far less efficient overall than any packet network because of the excess capacity it reserves.

As T1 and other digital trunks were deployed in the PSTN, digitized voice services in 64Kbps increments, each called a *Digital Signal 0* (DS0) —became the basic switchable unit of the PSTN. A single DS0 is a 64Kbps channel equivalent to an analog line converted to digital via G.711. With the advent of TDM-based digital switching, the DS0s were aggregated by digital access and cross-connect systems (DACS) for transport or presentation to the switch via DS1 (1.5 Mbps) or DS3 (45 Mbps) interfaces. These digital switches communicate over T1 and other digital trunks to access and toll tandem switches, sending calls across the telephone network to destination switches. The DS0 voice channels are then split back out to their original 64Kbps state and converted back to analog signals sent onward to the destination local loop.

In fact, there is now a full hierarchy to the T carrier system in North American and the E carrier system in Europe (as well as the more recent SONET/SDH optical carrier system). Aggregation of voice and data channels at many levels can take place, and knowing how these systems can interact is essential. Table 3.1 roughly defines the capacity and equivalency of the various North American, Japanese, and European digital signal hierarchies in a single chart. I've never been able to find this information in one place, so I created a single chart to cover the whole range of PSTN transport solutions in use today.

In Table 3.1, dark bands are for the circuits most commonly provisioned for business customers. Bolded items are used most commonly in wide area networks overall. Note: Although SONET and SDH are directly equivalent to each other, the process of mapping between them and their T or E-carrier counterparts requires the use of SONET Virtual Tributaries (VTs) and Virtual Tributary Groups (VTGs) or SDH Virtual Containers (VCs).

As you can see from Table 3.1, 24 DS0 channels make up a T1 circuit, 28 T1 circuits make up a T3 or OC-1 link, and so forth. An OC-12 link can support up to 7936 DS0 channels if it's broken out into E4 circuits or 8064 if it's broken out by T3 circuits through a DACS or Add Drop Multiplexer (ADM). 10 Gigabit Ethernet can run over an OC-192 SONET ring, and so on. These mappings are essential to understanding capacities for Internet access circuits as well when sizing for VoIP, since upper limits on Speed (left column) cannot be physically exceeded (note that actual throughput will be at least 10% lower because of overhead).

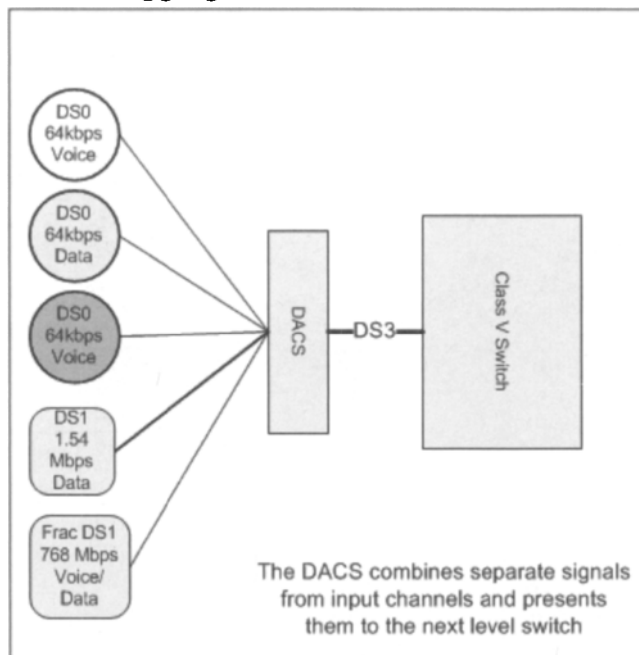
**Table 3.1** Digital Signal Hierarchy (North America and Europe)

Speed Mbps	Max 64k Channels	SONET	North America	Europe	SDH	Equivalency
64 kbps	1		DS0	E0		DS0 = E0
144 kbps	2B+D		ISDN BRI	ISDN BRI		2 DS0 or E0
1.544 1.728*	23B+D 24	VT 1.5*	ISDN PRI (NA) DS1 (T1)		VC-11*	23 DS0 24 DS0 [Japan: J1]
2.048 2.304*	30 B+D 31	VT 2.0*		ISDN PRI E1	VC-12*	30 E0 31 E0
3.152 3.456*	48	VT 3.0*	DS1C (T1C)			2 T1
6.312 6.912*	96	VT 6.0* [VTG*]	DS2 (T2)		VC-21* [TUG-2*]	4 T1 or 3 E1 VTG = TUG-2 [J2]
8.448	124			E2	VC-22	4 E1
32.06* 34.37	480* 496			E3	VC-3* [TUG-3*]	[J3* = 5x J2] 16 E1 via 4 E2 [VC-31]
44.74 48.96* 51.84**	672	STS-1** (OC-1)	DS3 (T3)			28 T1 or 7 T2 or 21 E1 TUG-3 = 7 TUG-2 7 VTG [VC-32]
97.73† 100.0‡ 135.0* 139.3 155.5**	1440**** 1488 1984 2016	STS-3** (OC-3)		E4	VC-4** STM-1**	[J4† = 3x J3] 48 E1 via 3 E3 via 3 VC-3 64 E1 via 1 E4 via VC-4 84 T1 or 63 E1 via 3 T3 Fast Ethernet ‡, FDDI ‡
274.2 311.0**	4032	STS-6** (OC-6)	DS4 (T4)		STM-2**	168 T1 (or 126 E1) via 6 T3 128 E1 via 2 E4 or 2 VC-4
400.4 466.6**	5760 6048	STS-9** (OC-9)	DS5 (T5)		STM-3**	T5 = 240 T1 via 60 T2 [J5] 252 T1 (or 189 E1) via 9 T3
565.6 622.1**	7936 8064	STS-12** (OC-12)		E5	STM-4**	4 E4 via 4 VC-4 or 4 OC-3 336 T1 via 12 T3
1 Gbps ‡ 2.5 Gbps	32,256	STS-48 (OC-48)			STM-16	48 T3 or 16 E4 or 16 VC-4 2x Gigabit Ethernet ‡
10 Gbps	129,024	STS-192 (OC-192)			STM-64	192 T3 or 64 E4 or 64 VC-4 10 Gigabit Ethernet
40 Gbps	516,096	STS-768 (OC-768)			STM-256	21,504 T1 or 768 T3 or 256 E4 or 256 VC-4
160 Gbps	2,064,384	STS-3072 (OC-3072)			STM-1024	86,016 T1 or 3072 T3 or 1024 E4 or 1024 VC-4

Perhaps you have ordered and provisioned a voice or data T1 for your company or clients. Have you ever thought why only one voice T1 is needed for a company of 100 employees with a PBX, knowing that only 24 channels can be used at any one time? The answer is that not everyone will be on the phone, receiving a fax, or otherwise using an available channel at once. Normally you can count on a six-to-ten ratio when calculating how many DS0s are needed. Those in the sales and service industry may go as low as four-to-one because they are on the phone more and need higher channel availability. Even with VoIP, sizing access circuits is important, since there are hard limits on the amount of data that can be pushed through that circuit network, even if the number of channels isn't so important. Less bandwidth might be required if G.729 was used in place of G.711, but more would be required if the link also supported Internet access, especially if Quality of Service (QoS) limitations weren't set up on the corresponding routers.

In Figure 3.5 we see that the DACS can be used to combine a wide variety of digital signal inputs and present them through a single interface to the next hop, which might be a switch, SONET multiplexing equipment, enterprise routing equipment, or something else. Keep in mind that although both voice and data traffic of any flavor can run over SONET, timing requirements won't allow something like a T1 to run over something asynchronous like Gigabit Ethernet.

**Figure 3.5** DACS Channel Aggregation



# PSTN: Switching and Signaling

As the PSTN's global reach and capabilities become more extensive, signaling became the most significant security concern within the PSTN. In its early days, signaling was no more complicated than taking the phone off-hook to let an operator know you wanted to make a call. Dialing gradually became more automatic, first for operators, then later for subscribers. Today's direct-dial networks, VoIP gateways, and myriad protocols only serve to increase the complexities and risks when it comes to signaling.

Electromechanical automated switching equipment first appeared in 1891 following Almon Strowger's patented Step by Step (SXS) system, although Bell System resistance to it would postpone its adoption for decades. The classic rotary dial phone was another Strowger invention that was finally adopted by the Bell System in 1919 along with SXS switches. Yet it would take until 1938 for Western Electric (the equipment R&D arm of the Bell system) to develop a superior automatic switching system, namely the crossbar switch. And not until the 1950s did Bell Labs embark on a computer-controlled switch project, but the 101 ESS PBX that resulted in 1963 was only partially digital. Also introduced that year was the T1 circuit and Touch Tones, the Dual-Tone Multi-Frequency (DTMF) dialing scheme that is still with us today. Despite the fact that switching itself was analog, digital T1 circuits quickly replaced analog backbone toll circuits and most analog CO interconnect trunks. By 1965 Bell had released the first central office switch with computerized stored program control, the 1ESS that offered new features like speed dialing and call forwarding. Yet the 1ESS was still an analog switch at its core. Thanks to T1 "robbed bit" signaling, however, all signaling was out of band, at least from the phone phreaker's perspective.

Insiders suggest that AT&T was prepared to postpone true digital switching until the 1990s, but Northern Telecom changed their plans with the DMS-10 all-digital switch, introduced in the late 1970s. The need for an all-digital AT&T alternative drove development of the 5ESS and accelerated implementation of ISDN. Today, the most common Class 5 (central office) switches in North America are the Nortel DMS-100 and Lucent 5ESS, running ITU-T Signaling System Number 7 (SS7) with full ISDN support.

The Class 5 switch is the first point where we can find the full suite of telephone services being handled in one place as part of the Intelligent Network model. A typical Class 5 can handle operator services, call waiting, long distance, ISDN, and other data services. The Class 5 will have tables that are queried for every service and will send the appropriate request to the right place. For instance, when you pick up the phone in your house to make a long distance phone call, the Class 5 switch detects the line is open and provides a timeslot in the switch for your call (this is when you hear the dial tone), then based on the buttons pushed (dialed) the switch will send the call either to the local carrier or to the long distance provider. If you dial a long distance call from a provider who is not your local provider, the switch will deliver the request to the closest switch that handles calls for that particular carrier. Class 5 switches act on demand (i.e., they set up, sustain, and tear down

connections as needed). This helps to reduce the amount of traffic over the lines when not needed, thus expanding the overall capacity of the system. These switches are a real workhorse for telephone companies (LECs, CLECs, and even IXC, though they can use a Class 4 switch in most cases). A Class 5 switch can handle thousands of connections per minute.

## The Intelligent Network (IN), Private Integrated Services, ISDN, and QSIG

The model drawn up in the 1980s and 1990s for advanced network functionality is called the Intelligent Network (IN). Services such as 8XX-number lookups as well as Calling Cards, Private Integrated Services Network (PISNs), and many other advanced services are all made possible through SS7, ISDN, and IN capabilities. PISNs are geographically disparate networks that are connected via leased lines that allow for enhanced services such as multi-vendor PBX deployments, Voice VPNs (don't get these confused with data VPNs, they are a true private network for voice, just like that provided by a PBX), and even certain kinds of VoIP. A Private Integrated service Network Exchange (PINX) lives within a PISN. Another application is integration with the QSIG protocol, which allows PBX products from other vendors be able to be used transparently to integrate all voice networks.

QSIG (a Q.931 ISDN extension) as a protocol has been around since the early to mid 1990s. We will talk about ISDN in the next section, but QSIG can be used to integrate systems even without ISDN. QSIG also leverages DPNSS, which was developed prior to when the final QSIG protocol was agreed upon. Not used much in U.S. networks, DPNSS had much of its life in the United Kingdom. Modern networks are using QSIG as the means to interconnect voice channels between PBXs while preserving critical information about caller and call state in the process.

ISDN is a common-channel signaling (CCS) solution that works with media or data traveling down one pair of wires while signaling control is handled over another. Remembering back to our earlier discussions of the channels of 64 kbps in size, a typical ISDN will hold 23 bearer (B) channels that carry voice and data and one data (D) channel that carries signaling information. All channels are 64kbps, so we have 24, 64-kbps channels totaling 1536 Mbps, or equivalent to a T1 and 30 B channels plus a D channel on an E-1, but in each case we lose one channel for signaling. Not only was distance from the central office a new issue with ISDN trunks, but the customer also had to implement new equipment. This Customer Premise Equipment (CPE) required ISDN terminators in order to access the network. Today the use of ISDN in the provisioning and delivery of broadband Internet access via DSL and cable services keep pricing competitive and affordable. Besides its use in the DSL services, ISDN still has an active share in providing redundant and emergency data network access to critical servers and services when higher speed lines or primary access has been disrupted.

Over the last 100 years, signaling has moved from operator-assisted modes to loop and disconnect modes, from single frequency to multifrequency signaling, and now to common channel signaling using the ISDN signaling channel.

## ITU-T Signaling System Number 7 (SS7)

SS7 (or C7) is an ITU-T (formerly CCITT) standard that defines how equipment in the PSTN digitally exchange data regarding call setup and routing. Other ITU-T signaling systems are still in use throughout the world, particularly:

- ITU-T 4, Channel-Associated Signaling (CAS) with a 2VF (voice frequency) code in the voice band and a 2040/2400 Hz supervisory tone
- ITU-T 5 CAS with 2VF and a 2400/2600 Hz supervisory tone, plus inter-register codes with Multi-Frequency (MF) tones
- ITU-T [5] R2 is a revision of ITU-T 5 but uses different frequencies

What sets SS7 apart above all is the fact that it is Common Channel Signaling (CCS), not CAS like its predecessors. Throughout the telecommunications industry the SS7 can be used for call session setup, management and tear down, call forwarding, caller identification information, toll free, LNP, and other service as implemented by carriers. Information passed through SS7 networks are communicated completely out of band meaning that signaling and media do not travel down the same path. The SS7 was loosely designed around the OSI 7-layer model. Figure 3.6 illustrates their basic similarities.

### *Message Transfer Parts 1, 2, and 3 (MTP)*

MTP level 1 is much the same as the Physical layer (1) of the OSI. Here the electrical and physical characteristics of the digital signaling are addressed. The physical interfaces defined here are those such as our previously discussed DS0 and T1. MTP level 2 aligns with the Data Link layer of the OSI. MTP level 2 takes care of making sure transmissions are accurate from end to end, just like the Data Link layer issues such as flow control and error checking are handled in the MTP level 2 area. MTP level 3 aligns itself with the Network layer of the OSI. MTP level 3 reroutes calls away from failed links and controls signaling when congestion is present.

### *Telephone User Part (TUP)*

This is an analog system component. Prior to digital signaling the TUP was used to set up and tear down calls. Today most countries are using the ISDN User Part (ISUP) to handle this requirement.

## *ISDN User Part (ISUP)*

Most countries are using ISUP to handle basic call components. ISUP works by defining the protocols used to manage calls between calling and called parties.

Automatic Number Identification (ANI), or—when it's passed on to a subscriber, known as Calling Party Identification Presentation (CLIP)—caller ID is passed to the PSTN (or back again) through ISDN trunks and displays the calling party's telephone number at the called party's telephone set during the ring cycle. ANI is used for all Custom Local Area Signaling Services (CLASS) such as custom ringing, selective call forwarding, call blocking, and so on.

### Notes from the Underground...

#### **ANI Spoofing Services: Think You Can Trust Caller ID?...Think Again!**

A number of services aimed at private investigators, collections agencies, or law enforcement have sprung up since 2000 to provide pay-per-call ANI spoofing. The service works like this: After setting up payment, you choose the 10-digit ANI you want Caller ID to show (the LEC will typically add the business or individual associated with the ANI number), plus the target number you want to call, then the service calls you and initiates the spoofed ANI call to the target number you've selected. Your target thinks you're Pizza Hut calling back, or their mother, or whoever you're spoofing and you've just fooled them into picking up.

What you may not know is that this can be done from any PBX with ISDN trunks that can support ANI. Most LECs have no way of validating the ANI you present to them and happily pass that information along via CallerID, whether it's accurate or not. Note that this is different from the "Caller ID spoofing" that can be done after a caller picks up on some CallerID equipment (fun with friends, but not very useful if the caller decides not to answer). Effectively, ANI spoofing "poisons the well" from which Caller ID gets its data.

Some carriers have suggested that they will crack down on this practice, but since no comprehensive DID ownership database is kept across all LECs and CLECs there is no current method to verify an ANI in real-time when it's been presented.

## *Signaling Connection Control Part (SCCP)*

The SCCP is used mainly for translating 800, calling card, and mobile telephone numbers into a set single point destination code.

## Transaction Capabilities Applications Part (TCAP)

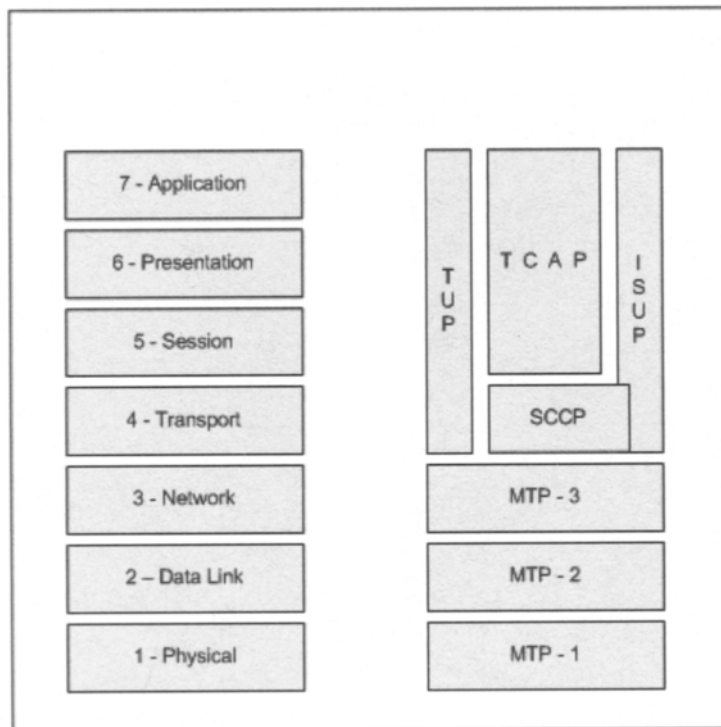
TCAP supports the passing and exchange of data within noncircuit-related communications. An example of noncircuit-related data is authentication of a user to a calling card plan.

Communication within an SS7 network and its equipment are called signaling points, of which there are three; Service Switching Points (SSP), Service Transfer Points (STP), and Service Control Points (SCP).

Service Switching Points (SSPs) are the primary calling switches; they set up, manage, and terminate calls. When calls need to be routed outside of the SSP's trunk group a request may be sent to a Service Control Point (SCP), which is a database that responds to queries and sends routing information to requesting switches that delivery the appropriate route for the type of call placed. A Service Transport Point (STP) is a packet switch that forwards messages down the appropriate link depending on the information contained within the packet.

Figure 3.6 shows basic OSI and SS7 stacks. Links between the SS7 network are broken down into six different types, lettered A through F. Figure 3.7 illustrates a typical SS7 network topology with specific link type labeled. Table 3.2 describes each link.

**Figure 3.6** Basic OSI and SS7 Stacks

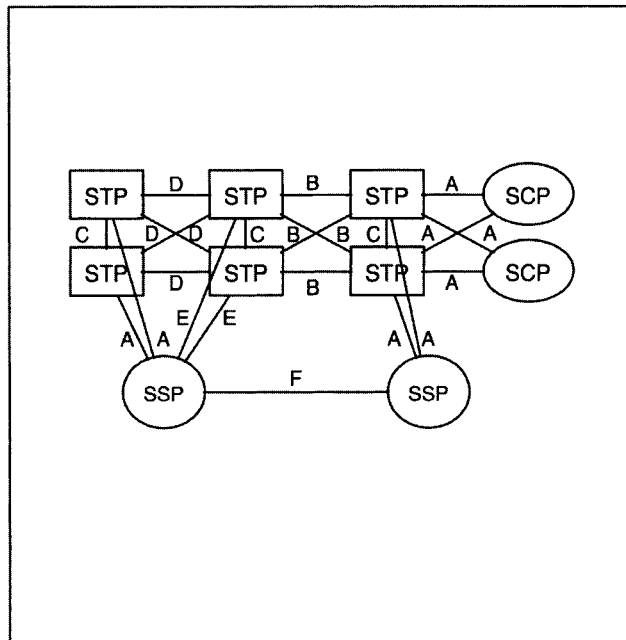


SS7 can also be run on IP networks using SCTP, using a slightly different stack that includes SCTP transport (instead of TCP or UDP).



SS7 has important security considerations, particularly between carriers where misconfigured implementations with unverified data can open the door to large scale fraud and other risks. This will be discussed in detail in this chapter's final section on PSTN Protocol Security, but the bottom line is that SS7 is a peer-to-peer protocol that may be out-of-band for phone phreaks, but carries significant risk from other sources, especially if it's running unencrypted over IP through SIGTRAN (SCTP).

**Figure 3.7** An SS7 Network Topology and Link Types



**Table 3.2** SS7 Network Links

Link Name	Function	Description
A	Access	Connects signal endpoints to an STP
B	Bridge	Connects peering STPs
C	Cross	Connects STPs into pairs to improve reliability
D	Diagonal	Essentially same as B
E	Extended	Used if A links are not available
F	Fully Associated	Direct connection of two endpoints (SSPs)

## PSTN: Operational and Regulatory Issues

Public Telephone and Telegraph (PTT) organizations are the highest-level monopoly (or ex-monopoly) in each country, and generally are expected to comply with ITU-T standards for interoperability. Each PTT is regulated by its country of origin. In the United States, AT&T was broken up in 1982 into a long distance unit (AT&T as the Inter-exchange carrier (IXC) was authorized only to carry long distance traffic), and reorganized groups of regional Bell Operating Companies were given a limited Local Exchange Carrier (LEC) role that until recently prevented them from selling interstate (or interLATA) long distance services. Competitive LECs (CLECs), in spite of regulatory advantages, hold less than 10% of local lines.



### WARNING

---

VoIP used for toll bypass is illegal in certain countries. Be sure you understand associated laws before implementing a VoIP system internationally.

---

As part of the AT&T breakup, 160 local access and transport areas (LATAs) were created around area code boundaries. Initially, LECs could not provide long distance service across and long distance companies could not provide local service, and some states have not removed these restrictions. Similar attempts to promote competitive services within specific countries are underway in various parts of the world.

## PSTN Call Flow

Now that we have discussed what makes up the PSTN, let's put it all together and walk through a messaging sequence. Here we will start from a caller picking up the phone attempting to make a call. The flow will be broken down into off-hook, digit receipt, ring down, conversation, and on-hook sections. We will start by imagining someone (Party B) picking up the phone to make the call (to Party A, on the same CO switch). The following list outlines, in order, the actions performed by the network:

Party B picks up the phone, and the off-hook sequence begins:

1. The off-hook state is detected by the switch (loop or ground start).
2. The switch establishes the time slot and sends a dial tone on the voice path.
3. The switch awaits digits pressed by Party B.

The digit receipt sequence is as follows:

1. Party B dials digits on the touch pad.
2. Each digit is received by the switch and sends a silence tone and starts Inter Digit Timer (IDT).
3. IDT starts when the switch is awaiting a dialed digit and stops when the digit is pressed.

After Party B dials the last number, the ring down sequence begins:

1. When the digit receipt stops (or when the maximum dialed digits are pressed), the switch sends the request to the called number to allocate a time slot.
2. When the called switch allocates a time slot the path is switched to the call handler.
3. Party A's phone rings (unless it is already off-hook).

Parties A and B can begin their conversation after the following sequence of steps is completed:

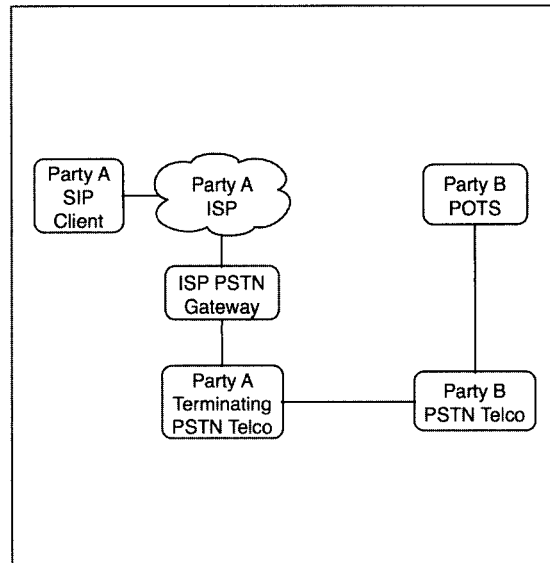
1. Party A picks up the phone.
2. The switch receives an answered call indication (off-hook).
3. The ring-down signals stop.
4. Parties A and B are able to speak on the established voice path.

After the two parties finish their conversation, the on-hook sequence of steps begins:

1. The conversation ends with either party hanging up the phone.
2. The on-hook indication is received by switches on access networks.
3. The switches release established paths (termination).
4. The call is ended.

During each of these sections there is traffic traveling in both directions to keep the signal alive. There are numerous acknowledgement requests between the caller and their access network, and the two access networks and the called party and their network, to keep this communication path alive. Most of this traffic is happening along the voice path.

This book is about securing voice over Internet networks, so later in the book you will be introduced to a protocol called Session Initiation Protocol (SIP). Though it is early on in the text we will now walk through a SIP to PSTN call. Remember that PSTN is a voice network and the SIP is originating from a data-only network. We will follow the sections of off-hook, digit receipt, ring down, conversation, and on-hook. To better visualize this call sequence we will use the following illustration (see Figure 3.8) to help us. Party A will be the SIP user and Party B will be the PSTN user.

**Figure 3.8** SIP-to-PSTN Call Flow

Party A picks up the phone, and the off-hook sequence begins:

1. Party A picks up the phone and dials the number.
2. An off-hook state is noticed by the SIP client.
3. The SIP client sends a request to the SIP proxy (at ISP).
4. The SIP client sends the SIP tel URL with the request.
5. ISUP message is prepared by the ISP PSTN Gateway.
6. The ISP Proxy finds the local terminating PSTN to send the call through (Network PSTN Gateway NGW).

The digit receipt sequence of steps begins:

1. Since Party A already sent the entire dialed number through the SIP phone prior to the call being sent through the Network PSTN Gateway, all the dial information is already there, so when the call is sent to the PSTN the switches already have all the information they need to process and route the call (i.e., no overlap sending is required).
2. This is sent through ISUP Messaging by the ISP PSTN Gateway.

Now, the ring down sequence begins:

1. Party A's switch establishes a one-way voice path.

2. Party A's switch sends a ringing tone.
3. At the same time, Party B's switch is establishing its voice path.
4. Party B's switch completes the set up.
5. Party B's phone rings.

Parties A and B can begin their conversation after the following sequence of steps is completed:

1. Party B picks up the phone.
2. Switches receive an answered call indication.
3. Party A's switch sets communication to bidirectional.
4. Parties A and B are able to speak on the established voice path.

When the two parties end their conversation, the on-hook sequence of steps begins:

1. The conversation ends with Party A hanging up the phone.
2. The SIP client sends a BYE message to Proxy at ISP.
3. The ISP Proxy sends a BYE signal to NGW.
4. Switches release established paths (termination).
5. The call is ended.

## PSTN Protocol Security

If you thought that PSTN protocols are more secure than the IP protocols riding on PSTN access circuits, then prepare to be shocked. In some respects, one of the greatest threats to the Internet is the PSTN itself.

## SS7 and Other ITU-T Signaling Security

Despite the fact that ITU-T signaling protocols prior to SS7 are notoriously insecure (see the sidebar on Blueboxing and the Phone Phreaking community earlier in the chapter), they continue to be deployed around the world along with older switching equipment that is vulnerable to toll fraud, eavesdropping, and other risks. If your VoIP system will be interfacing with such equipment, take countermeasures to reduce potential exposure and liability, set alarms, and review logs.

That is not to suggest that SS7 is particularly secure, but it is much harder for a subscriber to inject signaling into an SS7 network. That being said, the primary threat for SS7 networks are the peering arrangements (particularly among CLEC partners) for injection of false and/or fraudulent signaling and other messaging information. SS7 as currently defined does

not have policy controls built in to address this issue. The risks and countermeasures were summarized quite well by the 3GPP SA WG3 Technical Specification Group in January 2000 for 3G TR 33.900 V1.2.0:

The security of the global SS7 network as a transport system for signaling messages e.g. authentication and supplementary services such as call forwarding is open to major compromise.

The problem with the current SS7 system is that messages can be altered, injected or deleted into the global SS7 networks in an uncontrolled manner. In the past, SS7 traffic was passed between major PTOs covered under treaty organization and the number of operators was relatively small and the risk of compromise was low

Networks are getting smaller and more numerous. Opportunities for unintentional mishaps will increase, as will the opportunities for hackers and other abusers of networks. With the increase in different types of operators and the increase in the number of interconnection circuits there is an ever-growing loss of control of security of the signaling networks.

There is also exponential growth in the use of interconnection between the telecommunication networks and the Internet. The IT community now has many protocol converters for conversion of SS7 data to IP, primarily for the transportation of voice and data over the IP networks. In addition new services such as those based on IN will lead to a growing use of the SS7 network for general data transfers.

There have been a number of incidents from accidental action, which have damaged a network. To date, there have been very few deliberate actions. The availability of cheap PC based equipment that can be used to access networks and the ready availability of access gateways on the Internet will lead to compromise of SS7 signaling and this will affect mobile operators.

The risk of attack has been recognized in the USA at the highest level of the President's office indicating concern on SS7. It is understood that the T1, an American group is seriously considering the issue. For the network operator there is some policing of incoming signaling on most switches already, but this is dependent on the make of switch as well as on the way the switch is configured by operators.

Some engineering equipment is not substantially different from other advanced protocol analyzers in terms of its fraud potential, but is more intelligent and can be programmed more easily. The SS7 network as presently engineered is insecure. It is vitally important that network operators ensure that signaling screening of SS7 incoming messages takes place at the entry points to their networks and that operations and maintenance systems alert against unusual SS7 messages. There are a number of messages that can have a significant effect on the operation of the network and inappropriate messages should be controlled at entry point.

Network operators or network security engineers should on a regular basis carry out monitoring of signaling links for these inappropriate messages. In signing agreements with roaming partners and carrying out roaming testing, review of messages and also to seek appropriate confirmation that network operators are also screening incoming SS7 messages their networks to ensure that no rogue messages appear.

In summary there is no adequate security left in SS7. Mobile operators need to protect themselves from attack from hackers and inadvertent action that could stop a network or networks operating correctly.

Bottom line: Just because SS7 is harder for subscribers to crack doesn't mean it is secure overall. SS7 peering in the PSTN is not nearly as robust as its BGP equivalent on the Internet, and this has the potential for dire consequences if it were to be exploited maliciously. It's not yet clear if or how the ITU-T plans to address these concerns directly in a revision to SS7, although a T1S1 SS7 Security Standard was proposed at one time as part of an overall Study Group 17 (SG-17) effort. RFC 3788, Security Considerations for SIGTRAN protocols, was published by the Internet Engineering Task Force (IETF) in June 2004, and suggests the use of specific TLS and IPSEC profiles when using SS7 over IP, though it also notes that the "Peer To Peer" challenge still exists with SS7. The Network Interconnection Interoperability Forum (NIIF) within the Alliance for Telecommunications Industry Solutions (ATIS) has published many guidelines on the topic of secure interconnections (available to members or to the public for a fee). The good news is that unlike the Internet's in-band signaling model, which is vulnerable to direct attack, the SS7 signaling network is out of band to the voice and data communication it carries.

## ISUP and QSIG Security

Automatic Number Identification (ANI)-based security mechanisms can be spoofed in both directions, although some carriers claim to have clamped down on this practice (I'm not convinced this can be done). This can be used to create false Caller-ID data to subscribers. If

your organization uses ANI to verify identity (as a very large credit card user has been known to do), you are asking for trouble. It's only slightly more difficult than spoofing an e-mail address if you know what you're doing, so tread carefully here.

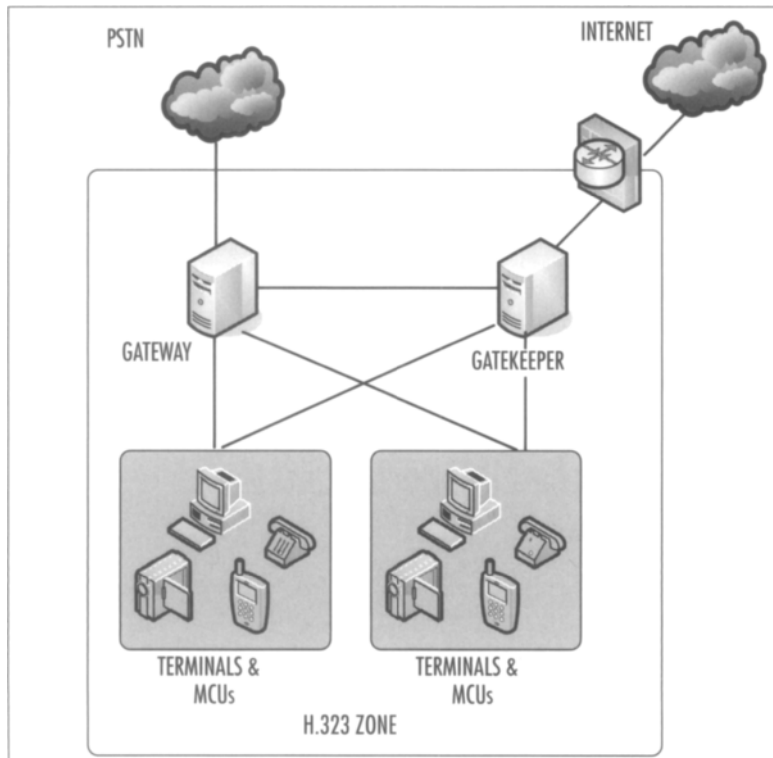
Other ISUP and QSIG fields have similar problems, so be very careful with any trust assumptions you make with these protocols. Always assume that CLASS services like distinctive ringing, selective call acceptance, selective call forward, and so on will be fooled by ANI spoofing and similar ISUP or SSIG attacks.

## The H.323 Protocol Specification

The H.323 protocol suite allows dissimilar communication devices to communicate with each other. H.323 (which is implemented primarily at versions 4 and 5 as of the time of this writing) is a sometimes Byzantine international protocol published by the ITU that supports interoperability between differing vendor implementations of telephony and multimedia products across IP-based networks. H.323 entities provide for real-time audio, video, and/or data communications. Support for audio is mandatory; support for data and video is optional.

The H.323 specification defines four different H.323 entities as the functional units of a complete H.323 network (see Figure 3.9). These components of an H.323 system include endpoints (terminals), gateways, gatekeepers, and multipoint control units (MCUs).

**Figure 3.9** H.323 Entities





Endpoints (telephones, softphones, IVRs, voice mail, video cameras, etc.) are typically devices that end-users interact with. MS Netmeeting is an example of an H.323 endpoint. Endpoints provide voice-only and/or multimedia such as video and real-time application collaboration.

Gateways handle signaling and media transport, and are optional components. Gateways typically serve as the interface to other types of networks such as ISDN, PSTN, or other H.323 systems. You can think of a gateway as providing “translation” functions. For example, an H.323 gateway will handle conversion of H.323 to SIP or H.323 to ISUP (ISUP (ISDN User Part) defines the interexchange signaling procedures for the trunk call control). Another way to think of this is that a gateway provides the interface between a packet-based network (e.g., a VoIP network) and a circuit-switched network (e.g., the PSTN). If a gatekeeper exists, VoIP gateways register with the gatekeeper and the gatekeeper finds the “best” gateway for a particular session.

Gatekeepers, which are also optional, handle address resolution and admission to the H.323 network. Its most important function is address translation between symbolic alias addresses and IP addresses. For example, in the presence of a gatekeeper, it is possible to call “Tom,” rather than 192.168.10.10. Gatekeepers also manage endpoints’ access to services, network resources, and optionally can provide additional services. They also monitor service usage and provide limited network bandwidth management. A gatekeeper is not required in an H.323 system. However, if a gatekeeper is present, terminals must make use of the services offered by gatekeepers. RAS defines these as address translation, admissions control, bandwidth control, and zone management. The gatekeeper and gateway functionalities are often present on a single physical device.

MCUs support multiparty conferencing between three or more endpoints. The H.323 standard allows for a variety of ad hoc conferencing scenarios, either centralized or decentralized.

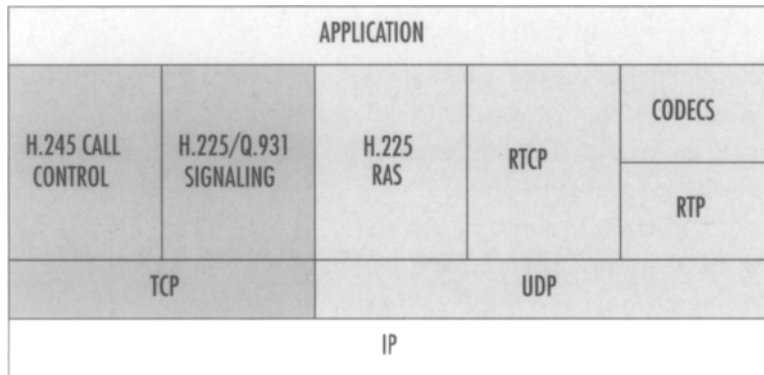
Back-end servers (BES) are an important supplementary function in an H.323-based environment. BES may provide services for user authentication, service authorization, accounting, charging and billing, and other services. In a simple network, the gatekeeper or gateway provides such services.

## The Primary H.323 VoIP-Related Protocols

H.323 is an umbrella-like specification that encompasses a large number of state machines that interact in different ways depending upon the presence, absence, and topological relationship of participating entities and the type of session (for example, audio or video). There are many subprotocols within the H.323 specification. In order to understand the overall message flows within an H.323 VoIP transaction, we will concern ourselves with the most

common ones that relate to VoIP. Figure 3.10 shows the relevant protocols and their relationships.

**Figure 3.10** VoIP-Related H.323 Protocol Stack



H.323 defines a general set of call setup and negotiating procedures—the most important in VoIP applications being H.225, H.235, H.245, and members of the Q.900 signaling series. Basic data-transport methods are defined by the real-time protocols RTP and RTCP. H.323 also specifies a group of audio codecs for VoIP communications, the G.700 series:

- **H.225/Q.931** Defines signaling for call setup and teardown, including source and destination IP addresses, ports, country code, and H.245 port information.
- **H.225.0/RAS** Specifies messages that describe signaling, Registration Admission and Status (RAS), and media stream information.
- **H.245** Specifies messages that negotiate the terminal capabilities set, the master/slave relationship, and logical channel information for the media streams.
- **Real Time Protocol (RTP)** Describes the end-to-end transport of real-time data.
- **Real Time Control Protocol (RTCP)** Describes the end-to-end monitoring of data delivery and QoS by providing information such as jitter and average packet loss.
- **Codecs** **The G.700 series of codecs used for VoIP includes:**
  1. **G.711** One of the oldest codecs, G.711 does not use compression, so voice quality is excellent. This codec consumes the most bandwidth. This is the same codec used by PSTN and ISDN.

2. **G.723.1** This codec was designed for videoconferencing/telephony over standard phone lines and is optimized for fast encode and decode. It has medium voice quality.
3. **G.729** This codec is used primarily in VoIP applications because of its low bandwidth requirements.

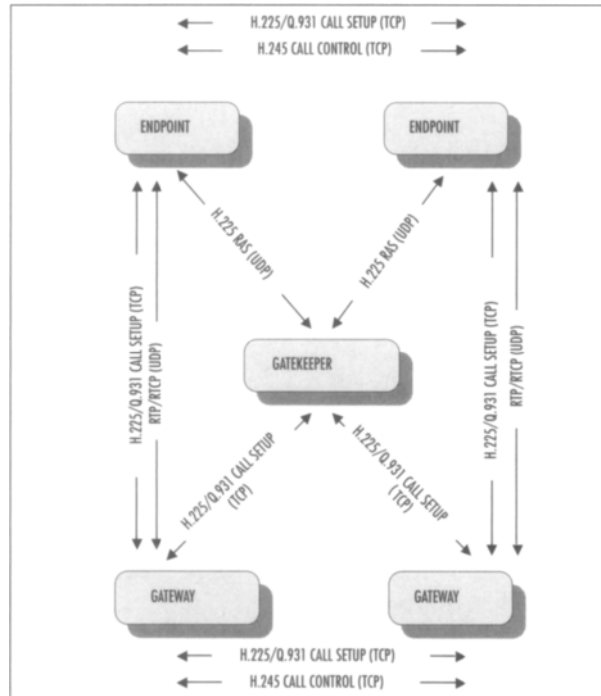
H.323 signaling exchanges typically are routed via gatekeeper or directly between the participants as chosen by the gatekeeper. Media exchanges normally are routed directly between the participants of a call. H.323 data communications utilizes both TCP and UDP. TCP ensures reliable transport for control signals and data, because these signals must be received in proper order and cannot be lost. UDP is used for audio and video streams, which are time-sensitive but are not as sensitive to an occasional dropped packet. Consequently, the H.225 call signaling channel and the H.245 call control channel typically run over TCP, whereas audio, video, and RAS channel exchanges rely on UDP for transport. Table 3.3 shows H.323 VoIP ports and protocols.

**Table 3.3** H.323 VoIP Ports and Protocols

Protocol	Function	Port(s)	Layer 4
H.225	(Q.931) Call Setup	1720	TCP
H.225	RAS	1719	UDP
H.245	Call Capabilities Negotiation	DYNAMIC	TCP
RTP/RTCP	Media Transport	DYNAMIC	UDP

In addition, H.235 recommends an assortment of messages, procedures, structures, and algorithms for securing signaling, control, and multimedia communications under the H.323 architecture. We will now look at each of these major VoIP-related protocols in more detail. Figure 3.11 shows the major signaling paths in an H.323 VoIP environment, and illustrates the several paths that signaling can take. In order to simplify the messaging sequence discussion we will ignore Fast Connect and Extended Fast Connect. There are two types of gatekeeper call signaling methods: Direct Endpoint signaling, where the terminating gateways or endpoints transfer call signaling information directly between themselves; and Gatekeeper-Routed call signaling, where setup signaling information is mediated by a gatekeeper.

Figure 3.11 Typical H.323 Channels



## H.225/Q.931 Call Signaling

Assuming a slow start connection procedure, the H.225 protocol defines the two important stages of call setup: Call signaling and RAS. Call signaling describes standards for call setup, maintenance and control, and teardown. A subset of Q.931 call signaling messages are used to initiate connections between H.323 endpoints, over which real-time data can be transported. The signaling channel is opened between an endpoint-gateway, a gateway-gateway, or gateway-gatekeeper prior to the establishment of any other channels. If no gateway or gatekeeper is present, H.225 messages are exchanged directly between the endpoints.

### Tools & Traps...

#### What Is the Difference Between QSIG and Q.931?

Initially, PBXs only connected via trunk lines to the PSTN. Then, people begin connecting PBXs with other PBXs over private leased lines (tie-lines) in order to save toll charges. This worked so well that these same people decided to form a single logical

Continued

[www.syngress.com](http://www.syngress.com)

PBX out of a number of smaller switches. In order to provide all the extra features that callers had come to expect, supplementary signaling functionality was added to the protocols used to connect the switches. DPNSS describes this signaling. DPNSS is an industry standard interface defined between PBXes.

Q.931, also a standard, is designed to work between the PSTN and a PBX. It does not support the same features and services as DPNSS. QSIG (Q Signaling) is the European Computer Manufacturers' Association standard for PBX-to-PBX connections based on ISDN PRI. It's largely Q.931, with extensions for additional PBX features. QSIG is based on, and also supports many of the same features and services as DPNSS. QSIG is used to tunnel PSTN signaling messages over H.323 to another PSTN network transparently, as if the two PSTN networks were one and the same.

H.225 messages are encoded in binary ASN.1 PER (Packed Encoding Rules) format. Although the H.225.0 signaling channel may be implemented on top of UDP, all entities must support signaling over TCP port 1720.

## NOTE

Signaling traffic is binary encoded using ASN.1 (Abstract Syntax Notation One) syntax and per encoding rules. ASN.1 is not a programming language. It is a flexible notation that allows one to define a variety of data types. ASN.1 theoretically allows two or more dissimilar systems to communicate in an unambiguous manner. Frankly, this aim is more difficult than it might seem at first.

ASN.1 encoding rules are sets of rules used to transform data specified in the ASN.1 language into a standard format that can be decoded on any system that has a decoder based on the same set of rules. The H.323 family of protocols is compiled into a wire-line protocol using PER. PER (Packed Encoding Rules), a subset of BER, is a compact binary encoding that is used on limited-bandwidth networks. PER is designed to optimize the use of bandwidth, but the tradeoff is complexity—decoding PER PDUs has led to problems due to a number of factors including issues with octet alignment (PER encoding can be aligned or unaligned), integer precision (at times, a PER value may not contain a length field), and unconstrained character strings.

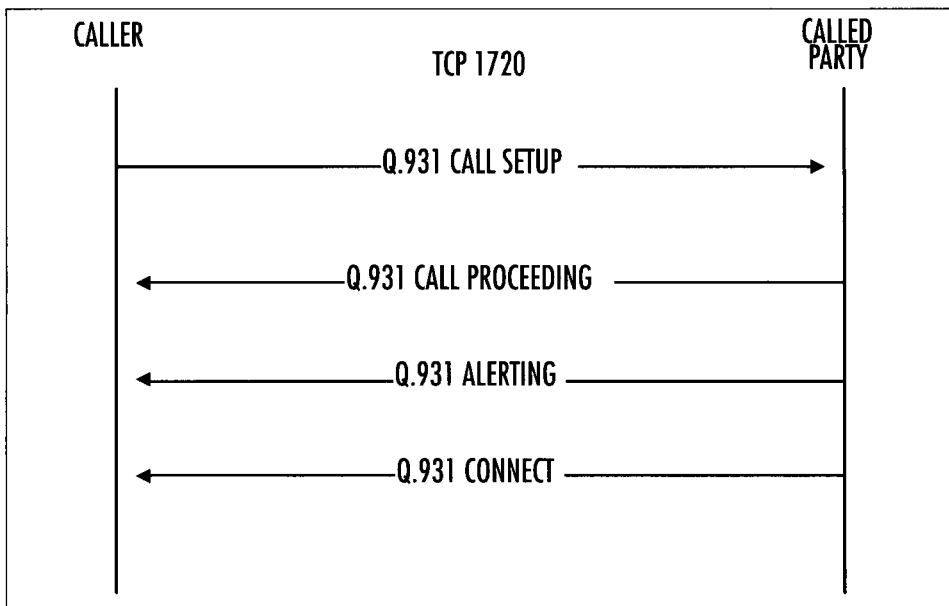
The H.225 protocol also defines messages used for endpoint-gatekeeper and gatekeeper-gatekeeper communication—this part of H.225 is known as RAS (Registration, Admission, Status), and unlike call signaling, runs over UDP. RAS is used to perform registration, admission control, bandwidth status changes, and teardown procedures between endpoints and gatekeepers. A RAS channel, separate from the call setup signaling channel, is used to

exchange RAS messages. This second signaling channel is opened between an endpoint and a gatekeeper prior to the establishment of additional channels.

Establishing a call between two endpoints requires a different connection schedule depending upon what entities are involved in the session. For direct connections between endpoints, two TCP channels are set up between the endpoints: one for call setup (Q.931/H.225 messages) and one for capabilities exchange and call control (H.245 messages). First, an endpoint initiates an H.225/Q931 exchange on a TCP well-known port (TCP 1720) with another endpoint. Several H.225/Q.931 messages are exchanged, during which time the called phone rings. Successful completion of the call results in an end-to-end reliable channel that supports the first of a number of H.245 messages. At the end of this exchange the called party picks up the receiver.

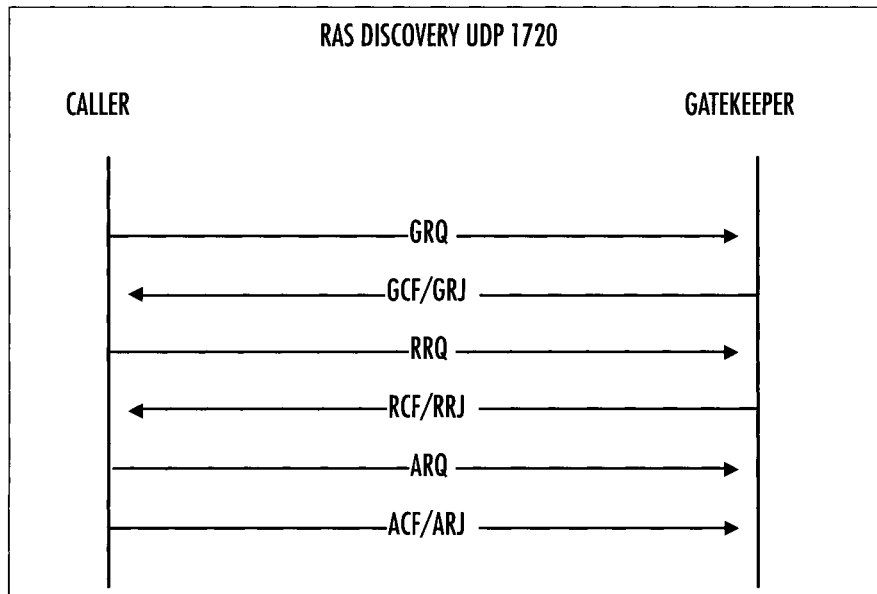
Note that the first of these signaling messages, the H.225.Q.931 Call Setup message (see Figure 3.12), has been the focus of extensive security vulnerability studies by the Oulu Secure Programming Group.

**Figure 3.12** H.225/Q.931 Signaling



If a gatekeeper is present between the endpoints (a more common scenario), then H.225 RAS signaling precedes the Q.931 signaling and abides by the sequence diagram shown in Figure 3.13.

Figure 3.13 H.225/Q.931 RAS



These messages are used to register with a gatekeeper and to request permission to initiate the call:

- **Gatekeeper Request (GRQ)** The GRQ packet is unicast in order to discover whether any gatekeepers exist. This requires that the gatekeepers IP address is configured on the endpoint. If this is not configured, the endpoint can fall back to multicast discovery of the gatekeeper.
- **Gatekeeper Confirm or Reject (GCF/GRJ)** Reply from the gatekeeper to endpoint that rejects the endpoint's registration request. Often due to configuration problems.
- **Registration Request (RRQ)** Request from a terminal or gateway to register with a gatekeeper.
- **Registration Confirm or Reject (RCF/RRJ)** Gatekeeper either confirms or rejects.
- **Admission Request (ARQ)** Request for access to packet network from terminal to gatekeeper.
- **Admission Confirm or Reject (ACF/ARJ)** Gatekeeper either confirms or rejects. If confirmed, the transport address and port to use for call signaling are included in the reply.

There are supplementary messages defined in the H.225/RAS specification that are used to request changes in bandwidth allocation, to reset timers, and for informational purposes. After the gatekeeper confirms the admission request, call signaling can begin. Signaling proceeds in the same manner as in Figure 3.11.

### NOTE

We have found privately that flooding multiple, malformed GRQ (Gatekeeper Request) packets to the gatekeeper results in the disconnection of a number of vendor's IP phones.

## H.245 Call Control Messages

After a connection has been set up via the call signaling procedure, H.245 messages (there are many of these) are used to resolve the call media type, to exchange terminal capabilities, and to establish the media flow before the call can be established. H.245 also manages call parameters after call establishment. H.245 messages also are encoded in ASN.1 PER syntax. The messages carried include notification of terminal capabilities, and commands to open and close logical channels. The H.245 control channel is permanently open, unlike the media channels.

### NOTE

Table 3.4 lists various types of messages and the H.323 ports used to transport them.

**Table 3.4** H.323 Ports

Message	Protocol/Port
H.245 messages	Dynamically assigned ports
RTP messages	Dynamically assigned ports
Gatekeeper	UDP Discovery Port 1718
Gatekeeper	UDP Registration and Status Port 1719
Endpoint	TCP Call Signaling Port 1720
Gatekeeper	Multicast 224.0.1.41

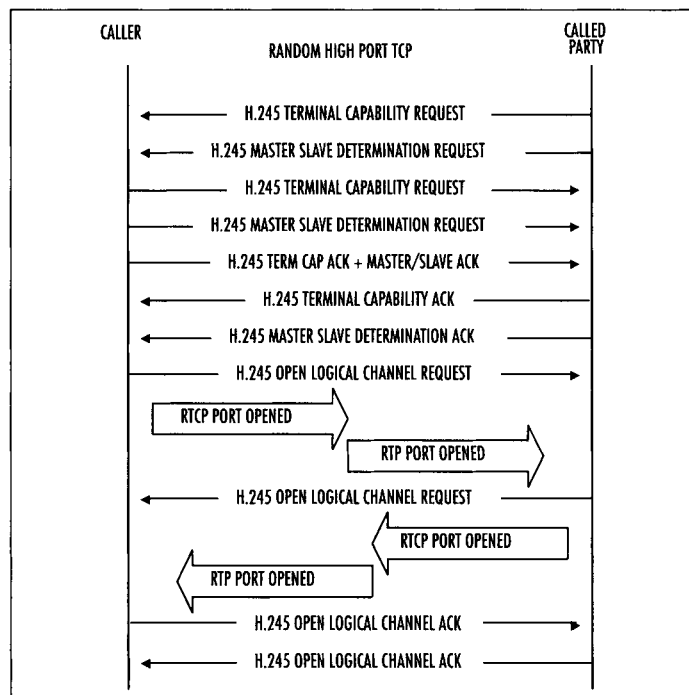
Continued



**Table 3.4 continued** H.323 Ports

Message	Protocol/Port
DNS	UDP 53
TFTP	UDP 69
SNMP	UDP 161, 162

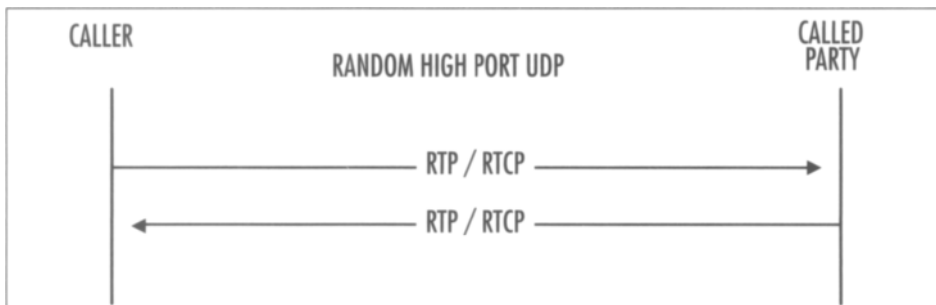
H.245 negotiations usually take place on a separate channel from the one used for H.225 exchanges, but newer applications support tunneling of H.245 PDUs within the H.225 signaling channel. There is no well-known port for H.245. The H.245 transport address always is passed in the call-signaling message. In other words, port information is passed within the payload of the preceding H.225/Q.931 signaling packets. The media channels (those used to transport voice and video) are similarly dynamically allocated. Figure 3.14 is an example of H.245 call control.

**Figure 3.14** H.245 Call Control

The called party opens the TCP port for establishing the control channel after extracting the port information from the H.225/Q.931 signaling packet. During this exchange, terminal capabilities such as codec choice and master/slave determination are negotiated. Media channel negotiations begin with the *OpenLogicalChannel* Request packet. When the called

party is ready to talk, it responds with an *OpenLogicalChannel Ack*, which contains the dynamic port information in the payload. As an aside, this use of dynamic ports makes it difficult to implement security policy on firewalls, NAT, and traffic shaping. In some cases, a special H.323-aware firewall or firewall component called an Application Layer Gateway (ALG) is required to reliably pass H.323 signaling and associated media. Once both RTP/RTCP channels are opened, communications proceeds (see Figure 3.15).

**Figure 3.15** RTP/RTCP Media Streams



## Real-Time Transport Protocol

Real-time transport protocol (RTP) is an application layer protocol that provides end-to-end delivery services of real-time audio and video. RTP provides payload identification, sequencing, time-stamping, and delivery monitoring. UDP provides multiplexing and checksum services. RTP can also be used with other transport protocols like TCP, and in conjunction with other signaling protocols like SIP or H.248.

The actual media (e.g., the voice packets) first is encoded by using an appropriate codec. The encoded audio stream is then passed via RTP, which is used to transfer the real-time audio/video streams over the Internet. Real-time transport control protocol (RTCP) is a required counterpart of RTP that provides control services for RTP streams. The primary function of RTCP is to provide feedback on the quality of the data distribution. Other RTCP functions include carrying a transport-level identifier for an RTP source, called a canonical name, which can be used by receivers to synchronize audio and video.

### NOTE

---

RTP runs on dynamic, even-numbered, high ports (ports > 1024), whereas RTCP runs on the next corresponding odd numbered, high port.

---

## H.235 Security Mechanisms

H.235 is expected to operate in conjunction with other H-series protocols that utilize H.245 as their control protocol and/or use the H.225.0 RAS and/or Call Signaling Protocol. H.235's major premise is that the principal security threat to communications is assumed to be eavesdropping on the network, or some other method of diverting media streams. The security issues related to DoS attacks are not addressed.

This family of threats relies on the absence of cryptographic assurance of a request's originator. Attacks in this category seek to compromise the message integrity of a conversation. This threat demonstrates the need for security services that enable entities to authenticate the originators of requests and to verify that the contents of the message and control streams have not been altered in transit.

Authentication is, in general, based either on using a shared secret (you are authenticated properly if you know the secret) or on public key-based methods with certifications (you prove your identity by possessing the correct private key). The basis for authentication (trust) and privacy is defined by the endpoints of the communications channel. For a connection establishment channel, this may be between the caller (such as a gateway or IP telephone endpoint) and a hosting network component (a gateway or gatekeeper). For example, a telephone "trusts" that the gatekeeper will connect it with the telephone whose number has been dialed. The result of trusting an element is the confidence to reveal the privacy mechanism (algorithm and key) to that element. Given the aforementioned information, all participants in the communications path should authenticate any and all trusted elements.

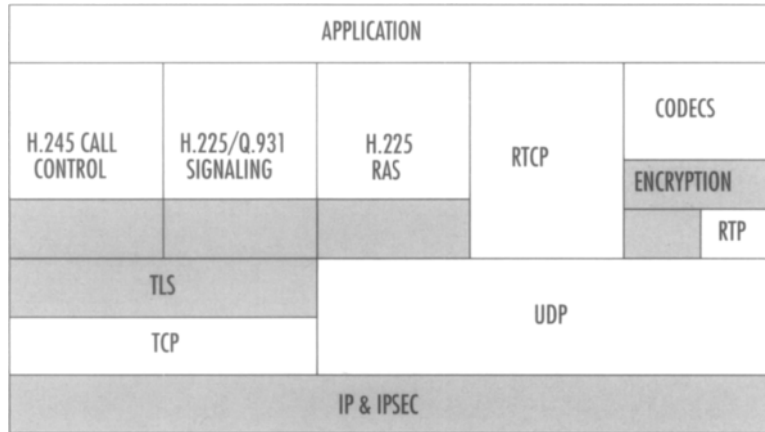
Encryption methods are defined as DES, 3DES, and AES. TLS (Transport Layer Security) and IPsec (IP Security) are recommended to secure layer 4 and layer 3 protocol messages, respectively. IPsec and TLS provide solutions at different levels of the ISO model—IPsec in the Network Layer, and TLS in the Transport Layer. Both use the same type of negotiation to set up tunnels, but IPsec often encrypts crucial header information, and TLS encrypts only the application payload of packet, thus TLS encryption retains IP addressing.

The scope of the H.235 specification is shown in Figure 3.16. H.235 addresses the protocols that are shaded in gray.

Let's look at how the H.235 specification interacts with each protocol.

- **H.245** The call signaling channel may be secured using TLS. Users may be authenticated either during the initial call connection, in the process of securing the H.245 channel, and/or by exchanging certificates on the H.245 channel. Media encryption details often are negotiated in private control channels determined by information carried in the OpenLogicalChannel connection.

Figure 3.16 H.235 Scope



- **H.225.0/Q.931** Q.931 can be secured via transport-layer security (TLS) or IPsec prior to any H.225.0 message exchange.
- **H.225.0/RAS** During the RAS phase of registering, the endpoint and the gate-keeper can exchange security policies and capabilities to define the security methods to be used in the initiated call session.
- **RTP/RTCP** H.245 signaling messages are used to provide confidentiality for a secured RTP channel. The method uses H.245 capability exchange for opening secured logical channels as part of the H.245 capability exchange phase, DES, 3DES or AES. The security capability is exchanged per media stream (RTP channel). The security mechanisms protect media streams and any control channels to operate in a completely independent manner.

H.235 specifies a number of security profiles. You can think of each security profile as a module consisting of a set of terms, definitions, requirements, procedures, and a profile overview that describe a particular instantiation of security methods. Security profiles, which are optional, may be implemented either selectively or in almost any combination. Endpoints may initially offer multiple security profiles simultaneously using the aforementioned RRQ/GRQ messages. H.235 also explicitly defines particular combinations of profiles that are useful or possible. For example, H.323 shows that the baseline security profile can be combined with SP4–Direct and selective routed call security, SP6–Voice encryption profile with native H.235/H.245 key management, and SP9–Security gateway support for H.323.

Profiles can be differentiated by the spectrum of security services each particular profile supports. The following security services are defined: Authentication, Nonrepudiation, Integrity, Confidentiality, Access Control, and Key Management. For example, the baseline security profile supports the security services shown in Figure 3.17.

**Figure 3.17** Baseline Security Profile Security Services (H.235.1)

SECURITY SERVICES	CALL FUNCTIONS			
	H.245 CALL CONTROL	H.225/Q.931 SIGNALING	H.225 RAS	RTP
AUTHENTICATION	PASSWORD HMAC-SHA1-96	PASSWORD HMAC-SHA1-96	PASSWORD HMAC-SHA1-96	
NONREPUDIATION				
INTEGRITY	PASSWORD HMAC-SHA1-96	PASSWORD HMAC-SHA1-96	PASSWORD HMAC-SHA1-96	
CONFIDENTIALITY				
ACCESS CONTROL				
KEY MANAGEMENT		SUBSCRIPTION BASED	SUBSCRIPTION BASED	

You can see that this profile provides for authentication and integrity of the signaling streams but does not provide support for encryption, nonrepudiation, or access control of these streams. The baseline security profile (H.235.1) specifies the following: Authentication and integrity protection, or authentication-only for H.225/RAS, H.225/Q.931 messages, and tunneled H.245 messages using password-based protection. The security profile is applicable to communications between H.323 terminal to gatekeeper, gatekeeper to gatekeeper, and H.323 gateway to gatekeeper.

The following Security Profiles are defined:

- **235.1** Baseline security profile
- **235.2** Signature security profile
- **235.3** Hybrid security profile
- **235.4** Direct and selective routed call security
- **235.5** Framework for secure authentication in RAS using weak shared secrets
- **235.6** Voice encryption profile with native H.235/H.245 key management
- **235.7** Usage of the MIKEY key management protocol for the Secure Real Time Transport Protocol
- **235.8** Key exchange for SRTP using secure signaling channels

- **235.9** Security gateway support for H.323

Each security profile defines security services in the context of the generic classes of attacks that can be prevented by implementing that particular profile. In the case of the baseline security profile, the following attacks are thwarted.

- **Man-in-the-middle attacks** Application level hop-by-hop message authentication and integrity protects against such attacks when the man in the middle is between an application level hop.
- **Replay attacks** Use of time stamps and sequence numbers prevent such attacks.
- **Spoofing** User authentication prevents such attacks.
- **Connection hijacking** Use of authentication/integrity for each signaling message prevents such attacks.

Other threats are not addressed in this profile. For example, the issue of confidentiality via encryption is left to other security profiles. Thus, any H.323 system that uses only this profile will be subject to attacks that rely upon data interception by sniffing traffic. If however, the endpoints that specify the security profiles available to the system indicate that they support SP6–Voice encryption profile with native H.235/H.245 key management, as well as the baseline security profile, then the threat posed by eavesdropping attacks will be minimized.

The matrix describing the security services provided by security profile H.235.6 is shown in Figure 3.18.

**Figure 3.18** Voice Encryption Profile with Native H.235/H.245 Key Management

SECURITY SERVICES	CALL FUNCTIONS			
	H.245 CALL CONTROL	H.225/Q.931 SIGNALING	H.225 RAS	RTP
AUTHENTICATION				
NONREPUDIATION				
INTEGRITY				
CONFIDENTIALITY				AES, 3DES, DES, RC-2
ACCESS CONTROL				
KEY MANAGEMENT	AUTHENTICATED DIFFIE-HELLMAN	AUTHENTICATED DIFFIE-HELLMAN		

In Figure 3.18 you can see that the addition of security profile H.235.6 to the baseline security profile adds methods for Diffie-Hellman key management and encryption of the media streams. In this fashion, security profiles can be added to the H.323 entities within your environment so as to provide only the security controls dictated by your security requirements. This approach allows some customization of the H.323 security controls so that, for example, they can be configured to work with your particular existing firewall infrastructure. We'll discuss H.323 firewall issues in Chapter 8.

## Understanding SIP

As the Internet became more popular in the 1990s, network programs that allowed communication with other Internet users also became more common. Over the years, a need was seen for a standard protocol that could allow participants in a chat, videoconference, interactive gaming, or other media to initiate user sessions with one another. In other words, a standard set of rules and services was needed that defined how computers would connect to one another so that they could share media and communicate. The Session Initiation Protocol (SIP) was developed to set up, maintain, and tear down these sessions between computers.

By working in conjunction with a variety of other protocols and specialized servers, SIP provides a number of important functions that are necessary in allowing communications between participants. SIP provides methods of sharing the location and availability of users and explains the capabilities of the software or device being used. SIP then makes it possible to set up and manage the session between the parties. Without these tasks being performed, communication over a large network like the Internet would be impossible. It would be like a message in a bottle being thrown in the ocean; you would have no way of knowing how to reach someone directly or whether the person even could receive the message.

Beyond communicating with voice and video, SIP has also been extended to support instant messaging and is becoming a popular choice that's incorporated in many of the instant messaging applications being produced. This extension, called SIMPLE, provides the means of setting up a session in much the same way as SIP. SIMPLE also provides information on the status of users, showing whether they are online, busy, or in some other state of presence. Because SIP is being used in these various methods of communications, it has become a widely used and important component of today's communications.

SIP was designed to initiate interactive sessions on an IP network. Programs that provide real-time communication between participants can use SIP to set up, modify, and terminate a connection between two or more computers, allowing them to interact and exchange data. The programs that can use SIP include instant messaging, voice over IP (VoIP), video teleconferencing, virtual reality, multiplayer games, and other applications that employ single-media or multimedia. SIP doesn't provide all the functions that enable these programs to communicate, but it is an important component that facilitates communication between two or more endpoints.

You could compare SIP to a telephone switchboard operator, who uses other technology to connect you to another party, set up conference calls or other operations on your behalf, and disconnect you when you're done. SIP is a type of signaling protocol that is responsible for sending commands to start and stop transmissions or other operations used by a program. The commands sent between computers are codes that do such things as open a connection to make a phone call over the Internet or disconnect that call later on. SIP supports additional functions, such as call waiting, call transfer, and conference calling, by sending out the necessary signals to enable and disable these functions. Just as the telephone operator isn't concerned with how communication occurs, SIP works with a number of components and can run on top of several different transport protocols to transfer media between the participants.

## Overview of SIP

One of the major reasons that SIP is necessary is found in the nature of programs that involve messaging, voice communication, and exchange of other media. The people who use these programs may change locations and use different computers, have several usernames or accounts, or communicate using a combination of voice, text, or other media (requiring different protocols). This creates a situation that's similar to trying to mail a letter to someone who has several aliases, speaks different languages, and could change addresses at any particular moment.

SIP works with various network components to identify and locate these endpoints. Information is passed through proxy servers, which are used to register and route requests to the user's location, invite another user(s) into a session, and make other requests to connect these endpoints. Because there are a number of different protocols available that may be used to transfer voice, text, or other media, SIP runs on top of other protocols that transport data and perform other functions. By working with other components of the network, data can be exchanged between these user agents regardless of where they are at any given point.

It is the simplicity of SIP that makes it so versatile. SIP is an ASCII- or text-based protocol, similar to HTTP or SMTP, which makes it more lightweight and flexible than other signaling protocols (such as H.323). Like HTTP and SMTP, SIP is a request-response protocol, meaning that it makes a request of a server, and awaits a response. Once it has established a session, other protocols handle such tasks as negotiating the type of media to be exchanged, and transporting it between the endpoints. The reusing of existing protocols and their functions means that fewer resources are used, and minimizes the complexity of SIP. By keeping the functionality of SIP simple, it allows SIP to work with a wider variety of applications.

The similarities to HTTP and SMTP are no accident. SIP was modeled after these text-based protocols, which work in conjunction with other protocols to perform specific tasks. As we'll see later in this chapter, SIP is also similar to these other protocols in that it uses Universal Resource Identifiers (URIs) for identifying users. A URI identifies resources on



the Internet, just as a Uniform Resource Locator (URL) is used to identify Web sites. The URI used by SIP incorporates a phone number or name, such as SIP: user@syngress.com, which makes reading SIP addresses easier. Rather than reinventing the wheel, the development of SIP incorporated familiar aspects of existing protocols that have long been used on IP networks. The modular design allows SIP to be easily incorporated into Internet and network applications, and its similarities to other protocols make it easier to use.

## RFC 2543 / RFC 3261

The Session Initiation Protocol is a standard that was developed by the Internet Engineering Task Force (IETF). The IETF is a body of network designers, researchers, and vendors that are members of the Internet Society Architecture Board for the purpose of developing Internet communication standards. The standards they create are important because they establish consistent methods and functionality. Unlike proprietary technology, which may or may not work outside of a specific program, standardization allows a protocol or other technology to function the same way in any application or environment. In other words, because SIP is a standard, it can work on any system, regardless of the communication program, operating system, or infrastructure of the IP network.

The way that IETF develops a standard is through recommendations for rules that are made through Request for Comments (RFCs). The RFC starts as a draft that is examined by members of a Working Group, and during the review process, it is developed into a finalized document. The first proposed standard for SIP was produced in 1999 as RFC 2543, but in 2002, the standard was further defined in RFC 3261. Additional documents outlining extensions and specific issues related to the SIP standard have also been released, which make RFC 2543 obsolete and update RFC 3261. The reason for these changes is that as technology changes, the development of SIP also evolves. The IETF continues developing SIP and its extensions as new products are introduced and its applications expand.



---

Reviewing RFCs can provide you with additional insight and information, answering specific questions you may have about SIP. The RFCs related to SIP can be reviewed by visiting the IETF Web site at [www.ietf.org](http://www.ietf.org). Additional materials related to the Session Initiation Protocol Working Group also can be found at [www.softarmor.com/sipwgf/](http://www.softarmor.com/sipwgf/).

---

## SIP and Mbone

Although RFC 2543 and RFC 3261 define SIP as a protocol for setting up, managing, and tearing down sessions, the original version of SIP had no mechanism for tearing down sessions and was designed for the Multicast Backbone (Mbone). Mbone originated as a method of broadcasting audio and video over the Internet. The Mbone is a broadcast channel that is overlaid on the Internet, and allowed a method of providing Internet broadcasts of things like IETF meetings, space shuttle launches, live concerts, and other meetings, seminars, and events. The ability to communicate with several hosts simultaneously needed a way of inviting users into sessions; the Session Invitation Protocol (as it was originally called) was developed in 1996.

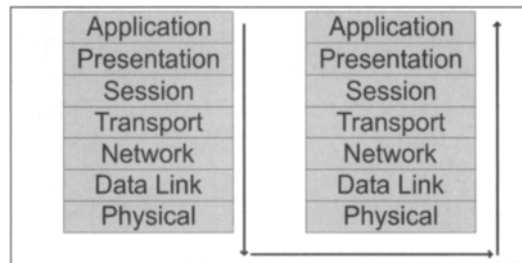
The Session Invitation Protocol was a precursor to SIP that was defined by the IETF MMUSIC Working group, and a primitive version of the Session Initiation Protocol used today. However, as VoIP and other methods of communications became more popular, SIP evolved into the Session Initiation Protocol. With added features like the ability to tear down a session, it was a still more lightweight than more complex protocols like H.323. In 1999, the Session Initiation Protocol was defined as RFC 2543, and has become a vital part of multi-media applications used today.

## OSI

In designing the SIP standard, the IETF mapped the protocol to the OSI (Open Systems Interconnect) reference model. The OSI reference model is used to associate protocols to different layers, showing their function in transferring and receiving data across a network, and their relation to other existing protocols. A protocol at one layer uses only the functions of the layer below it, while exporting the information it processes to the layer above it. It is a conceptual model that originated to promote interoperability, so that a protocol or element of a network developed by one vendor would work with others.

As seen in Figure 3.19, the OSI model contains seven layers: Application, Presentation, Session, Transport, Network, Data Link, and Physical. As seen in this figure, network communication starts at the Application layer and works its way down through the layers step by step to the Physical layer. The information then passes along the cable to the receiving computer, which starts the information at the Physical layer. From there it steps back up the OSI layers to the Application layer where the receiving computer finalizes the processing and sends back an acknowledgement if needed. Then the whole process starts over.

**Figure 3.19** In the OSI Reference Model, Data is Transmitted down through the Layers, across the Medium, and Back up through the Layers



The layers of the OSI reference model have different functions that are necessary in transferring data across a network, and mapping protocols to these layers make it easier to understand how they interrelate to the network as a whole. Table 3.5 shows the seven layers of the OSI model, and briefly explains their functions.

**Table 3.5** Layers of the OSI Model

Layer	Description
7: Application	The Application layer is used to identify communication partners, facilitate authentication (if necessary), and allows a program to communicate with lower layer protocols, so that in turn it can communicate across the network. Protocols that map to this layer include SIP, HTTP, and SMTP.
6: Presentation	The Presentation layer converts data from one format to another, such as converting a stream of text into a pop-up window, and handles encoding and encryption.
5: Session	The Session layer is responsible for coordinating sessions and connections.
4: Transport	The Transport layer is used to transparently transfer data between computers. Protocols that map to this layer include TCP, UDP, and RTP.
3: Network	The Network Layer is used to route and forward data so that it goes to the proper destination. The most common protocol that maps to this layer is IP.
2: Data Link	The Data Link layer is used to provide error correction that may occur at the physical level, and provide physical addressing through the use of MAC addresses that are hard-coded into network cards.

Continued

**Table 3.5 continued** Layers of the OSI Model

Layer	Description
1: Physical	The Physical layer defines electrical and physical specifications of network devices, and provides the means of allowing hardware to send and receive data on a particular type of media. At this level, data is passed as a bit stream across the network.

### *SIP and the Application Layer*

Because SIP is the Session Initiation Protocol, and its purpose is to establish, modify, and terminate sessions, it would seem at face-value that this protocol maps to the Session layer of the OSI reference model. However, it is important to remember that the protocols at each layer interact only with the layers above and below it. Programs directly access the functions and supported features available through SIP, disassociating it from this layer. SIP is used to invite a user into an interactive session, and can also invite additional participants into existing sessions, such as conference calls or chats. It allows media to be added to or removed from a session, provides the ability to identify and locate a user, and also supports name mapping, redirection, and other services. When comparing these features to the OSI model, it becomes apparent that SIP is actually an Application-layer protocol.

The Application layer is used to identify communication partners, facilitate authentication (if necessary), and allows a program to communicate with lower layer protocols, so that in turn it can communicate across the network. In the case of SIP, it is setting up, maintaining, and ending interactive sessions, and providing a method of locating and inviting participants into these sessions. The software being used communicates through SIP, which passes the data down to lower layer protocols and sends it across the network.

## **SIP Functions and Features**

When SIP was developed, it was designed to support five specific elements of setting up and tearing down communication sessions. These supported facets of the protocol are:

- User location, where the endpoint of a session can be identified and found, so that a session can be established
- User availability, where the participant that's being called has the opportunity and ability to indicate whether he or she wishes to engage in the communication
- User capabilities, where the media that will be used in the communication is established, and the parameters of that media are agreed upon

- Session setup, where the parameters of the session are negotiated and established
- Session management, where the parameters of the session are modified, data is transferred, services are invoked, and the session is terminated

Although these are only a few of the issues needed to connect parties together so they can communicate, they are important ones that SIP is designed to address. However, beyond these functions, SIP uses other protocols to perform tasks necessary that allow participants to communicate with each other, which we'll discuss later in this chapter.

## User Location

The ability to find the location of a user requires being able to translate a participant's username to their current IP address of the computer being used. The reason this is so important is because the user may be using different computers, or (if DHCP is used) may have different IP addresses to identify the computer on the network. The program can use SIP to register the user with a server, providing a username and IP address to the server. Because a server now knows the current location of the user, other users can now find that user on the network. Requests are redirected through the proxy server to the user's current location. By going through the server, other potential participants in a communication can find the user, and establish a session after acquiring their IP address.

## User Availability

The user availability function of SIP allows a user to control whether he or she can be contacted. Users can set themselves as being away or busy, or available for certain types of communication. If available, other users can then invite the user to join in a type of communication (e.g., voice or videoconference), depending on the capabilities of the program being used.

## User Capabilities

Determining the user's capabilities involves determining what features are available on the programs being used by each of the parties, and then negotiating which can be used during the session. Because SIP can be used with different programs on different platforms, and can be used to establish a variety of single-media and multimedia communications, the type of communication and its parameters needs to be determined. For example, if you were to call a particular user, your computer might support video conferencing, but the person you're calling doesn't have a camera installed. Determining the user capabilities allows the participants to agree on which features, media types, and parameters will be used during a session.

## Session Setup

Session setup is where the participants of the communication connect together. The user who is contacted to participate in a conversation will have their program “ring” or produce some other notification, and has the option of accepting or rejecting the communication. If accepted, the parameters of the session are agreed upon and established, and the two endpoints will have a session started, allowing them to communicate.

## Session Management

Session management is the final function of SIP, and is used for modifying the session as it is in use. During the session, data will be transferred between the participants, and the types of media used may change. For example, during a voice conversation, the participants may decide to invoke other services available through the program, and change to a video conferencing. During communication, they may also decide to add or drop other participants, place a call on hold, have the call transferred, and finally terminate the session by ending their conversation. These are all aspects of session management, which are performed through SIP.

## SIP URIs

Because SIP was based on existing standards that had already been proven on the Internet, it uses established methods for identifying and connecting endpoints together. This is particularly seen in the addressing scheme that it uses to identify different SIP accounts. SIP uses addresses that are similar to e-mail addresses. The hierarchical URI shows the domain where a user’s account is located, and a host name or phone number that serves as the user’s account. For example, SIP: myaccount@madeupsip.com shows that the account *myaccount* is located at the domain *madeupsip.com*. Using this method makes it simple to connect someone to a particular phone number or username.

Because the addresses of those using SIP follow a *username@domainname* format, the usernames created for accounts must be unique within the namespace. Usernames and phone numbers must be unique as they identify which account belongs to a specific person, and used when someone attempts sending a message or placing a call to someone else. Because the usernames are stored on centralized servers, the server can determine whether a particular username is available or not when a person initially sets up an account.

URIs also can contain other information that allows it to connect to a particular user, such as a port number, password, or other parameters. In addition to this, although SIP URIs will generally begin with SIP:, others will begin with SIPs:, which indicates that the information must be sent over a secure transmission. In such cases, the data and messages transmitted are transported using the Transport Layer Security (TLS) protocol, which we’ll discuss later in this chapter.

# SIP Architecture

Though we've discussed a number of the elements of SIP, there are still a number of essential components that make up SIP's architecture that we need to address. SIP would not be able to function on a network without the use of various devices and protocols. The essential devices are those that you and other participants would use in a conversation, allowing you to communicate with one another, and various servers may also be required to allow the participants to connect together. In addition to this, there are a number of protocols that carry your voice and other data between these computers and devices. Together, they make up the overall architecture of SIP.

## SIP Components

Although SIP works in conjunction with other technologies and protocols, there are two fundamental components that are used by the Session Initiation Protocol:

- User agents, which are endpoints of a call (i.e., each of the participants in a call)
- SIP servers, which are computers on the network that service requests from clients, and send back responses

## User Agents

User agents are both the computer that is being used to make a call, and the target computer that is being called. These make the two endpoints of the communication session. There are two components to a user agent: a client and a server. When a user agent makes a request (such as initiating a session), it is the User Agent Client (UAC), and the user agent responding to the request is the User Agent Server (UAS). Because the user agent will send a message, and then respond to another, it will switch back and forth between these roles throughout a session.

Even though other devices that we'll discuss are optional to various degrees, User Agents must exist for a SIP session to be established. Without them, it would be like trying to make a phone call without having another person to call. One UA will invite the other into a session, and SIP can then be used to manage and tear down the session when it is complete. During this time, the UAC will use SIP to send requests to the UAS, which will acknowledge the request and respond to it. Just as a conversation between two people on the phone consists of conveying a message or asking a question and then waiting for a response, the UAC and UAS will exchange messages and swap roles in a similar manner throughout the session. Without this interaction, communication couldn't exist.

Although a user agent is often a software application installed on a computer, it can also be a PDA, USB phone that connects to a computer, or a gateway that connects the network

to the Public Switched Telephone Network. In any of these situations however, the user agent will continue to act as both a client and a server, as it sends and responds to messages.

## SIP Server

The SIP server is used to resolve usernames to IP addresses, so that requests sent from one user agent to another can be directed properly. A user agent registers with the SIP server, providing it with their username and current IP address, thereby establishing their current location on the network. This also verifies that they are online, so that other user agents can see whether they're available and invite them into a session. Because the user agent probably wouldn't know the IP address of another user agent, a request is made to the SIP server to invite another user into a session. The SIP server then identifies whether the person is currently online, and if so, compares the username to their IP address to determine their location. If the user isn't part of that domain, and thereby uses a different SIP server, it will also pass on requests to other servers.

In performing these various tasks of serving client requests, the SIP server will act in any of several different roles:

- Registrar server
- Proxy server
- Redirect server

### *Registrar Server*

Registrar servers are used to register the location of a user agent who has logged onto the network. It obtains the IP address of the user and associates it with their username on the system. This creates a directory of all those who are currently logged onto the network, and where they are located. When someone wishes to establish a session with one of these users, the Registrar server's information is referred to, thereby identifying the IP addresses of those involved in the session.

### *Proxy Server*

Proxy servers are computers that are used to forward requests on behalf of other computers. If a SIP server receives a request from a client, it can forward the request onto another SIP server on the network. While functioning as a proxy server, the SIP server can provide such functions as network access control, security, authentication, and authorization.

### *Redirect Server*

The Redirect servers are used by SIP to redirect clients to the user agent they are attempting to contact. If a user agent makes a request, the Redirect server can respond with



the IP address of the user agent being contacted. This is different from a Proxy server, which forwards the request on your behalf, as the Redirect server essentially tells you to contact them yourself.

The Redirect server also has the ability to “fork” a call, by splitting the call to several locations. If a call was made to a particular user, it could be split to a number of different locations, so that it rang at all of them at the same time. The first of these locations to answer the call would receive it, and the other locations would stop ringing.

## NOTE

---

RFC 3261 defines the different types of SIP servers as logical devices, meaning that they can be implemented as separate servers or as part of a single application that resides on a single physical server. In other words, a single physical server may act in all or one of these roles.

In addition to this, the SIP servers can interact with other servers and applications on your network to provide additional services, such as authentication or billing. The SIP servers could access Lightweight Directory Access Protocol (LDAP) servers, database applications, or other applications to access back-end services.

---

## Stateful versus Stateless

The servers used by SIP can run in one of two modes: stateful or stateless. When a server runs in stateful mode, it will keep track of all requests and responses it sends and receives. A server that operates in a stateless mode won't remember this information, but will instead forget about what it has done once it has processed a request. A server running in stateful mode generally is found in a domain where the user agents resides, whereas stateless servers are often found as part of the backbone, receiving so many requests that it would be difficult to keep track of them.

## Location Service

The location service is used to keep a database of those who have registered through a SIP server, and where they are located. When a user agent registers with a Registrar server, a REGISTER request is made (which we'll discuss in the later section). If the Registrar accepts the request, it will obtain the SIP-address and IP address of the user agent, and add it to the location service for its domain. This database provides an up-to-date catalog of everyone who is online, and where they are located, which Redirect servers and Proxy servers can then use to acquire information about user agents. This allows the servers to connect user agents together or forward requests to the proper location.

# Client/Server versus Peer-to-Peer Architecture

In looking at the components of SIP, you can see that requests are processed in different ways. When user agents communicate with one another, they send requests and responses to one another. In doing so, one acts as a User Agent Client, and the other fulfills the request acts as a User Agent Server. When dealing with SIP servers however, they simply send requests that are processed by a specific server. This reflects two different types of architectures used in network communications:

- Client/Server
- Peer-to-peer

## Client/Server

In a client/server architecture, the relationship of the computers are separated into two roles:

- The client, which requests specific services or resources
- The server, which is dedicated to fulfilling requests by responding (or attempting to respond) with requested services or resources

An easy-to-understand example of a client/server relationship is seen when using the Internet. When using an Internet browser to access a Web site, the client would be the computer running the browser software, which would request a Web page from a Web server. The Web server receives this request and then responds to it by sending the Web page to the client computer. In VoIP, this same relationship can be seen when a client sends a request to register with a Registrar server, or makes a request to a Proxy Server or Redirect Server that allows it to connect with another user agent. In all these cases, the client's role is to request services and resources, and the server's role is to listen to the network and await requests that it can process or pass onto other servers.

The servers that are used on a network acquire their abilities to service requests by the programs installed on it. Because a server may run a number of services or have multiple server applications installed on it, a computer dedicated to the role of being a server may provide several functions on a network. For example, a Web server might also act as an e-mail server. In the same way, SIP servers also may provide different services. A Registrar can register clients and also run the location service that allows clients and other servers to locate other users who have registered on the network. In this way, a single server may provide diverse functionality to a network that would otherwise be unavailable.

Another important function of the server is that, unlike clients that may be disconnected from the Internet or shutdown on a network when the person using it is done, a server is generally active and awaiting client requests. Problems and maintenance aside, a dedicated server is up and running, so that it is accessible. The IP address of the server generally doesn't

change, meaning that clients can always find it on a network, making it important for such functions as finding other computers on the network.

## Peer to Peer

A peer-to-peer (P2P) architecture is different from the client/server model, as the computers involved have similar capabilities, and can initiate sessions with one another to make and service requests from one another. Each computer provides services and resources, so if one becomes unavailable, another can be contacted to exchange messages or access resources. In this way, the user agents act as both client and server, and are considered peers.

Once a user agent is able to establish a communication session with another user agent, a P2P architecture is established where each machine makes requests and responds to the other. One machine acting as the User Agent client will make a request, while the other acting as the User Agent server will respond to it. Each machine can then swap roles, allowing them to interact as equals on the network. For example, if the applications being used allowed file sharing, a UAC could request a specific file from the UAS and download it. During this time, the peers could also be exchanging messages or talking using VoIP, and once these activities are completed, one could send a request to terminate the session to end the communications between them. As seen by this, the computers act in the roles of both client and server, but are always peers by having the same functionality of making and responding to requests.

## SIP Requests and Responses

Because SIP is a text-based protocol like HTTP, it is used to send information between clients and servers, and User Agent clients and User Agent servers, as a series of requests and responses. When requests are made, there are a number of possible signaling commands that might be used:

- **REGISTER** Used when a user agent first goes online and registers their SIP address and IP address with a Registrar server.
- **INVITE** Used to invite another User agent to communicate, and then establish a SIP session between them.
- **ACK** Used to accept a session and confirm reliable message exchanges.
- **OPTIONS** Used to obtain information on the capabilities of another user agent, so that a session can be established between them. When this information is provided a session isn't automatically created as a result.
- **SUBSCRIBE** Used to request updated presence information on another user agent's status. This is used to acquire updated information on whether a User agent is online, busy, offline, and so on.

- **NOTIFY** Used to send updated information on a User agent's current status. This sends presence information on whether a User agent is online, busy, offline, and so on.
- **CANCEL** Used to cancel a pending request without terminating the session.
- **BYE** Used to terminate the session. Either the user agent who initiated the session, or the one being called can use the BYE command at any time to terminate the session.

When a request is made to a SIP server or another user agent, one of a number of possible responses may be sent back. These responses are grouped into six different categories, with a three-digit numerical response code that begins with a number relating to one of these categories. The various categories and their response code prefixes are as follows:

- **Informational (1xx)** The request has been received and is being processed.
- **Success (2xx)** The request was acknowledged and accepted.
- **Redirection (3xx)** The request can't be completed and additional steps are required (such as redirecting the user agent to another IP address).
- **Client error (4xx)** The request contained errors, so the server can't process the request
- **Server error (5xx)** The request was received, but the server can't process it. Errors of this type refer to the server itself, and they don't indicate that another server won't be able to process the request.
- **Global failure (6xx)** The request was received and the server is unable to process it. Errors of this type refer to errors that would occur on any server, so the request wouldn't be forwarded to another server for processing.

There are a wide variety of responses that apply to each of the categories. The different responses, their categories, and codes are shown in Table 3.6.

**Table 3.6** Listing of Responses, Response Codes, and Their Meanings

Response Code	Response Category	Response Description
100	Informational	Trying
180	Informational	Ringing
181	Informational	Call is being forwarded
182	Informational	Queued
200	Success	OK
300	Redirection	Multiple choices

Continued

**Table 3.6 continued** Listing of Responses, Response Codes, and Their Meanings

<b>Response Code</b>	<b>Response Category</b>	<b>Response Description</b>
301	Redirection	Moved permanently
302	Redirection	Moved temporarily
303	Redirection	See other
305	Redirection	Use proxy
380	Redirection	Alternative service
400	Client Error	Bad request
401	Client Error	Unauthorized
402	Client Error	Payment required
403	Client Error	Forbidden
404	Client Error	Not found
405	Client Error	Method not allowed
406	Client Error	Not acceptable
407	Client Error	Proxy authentication required
408	Client Error	Request timeout
409	Client Error	Conflict
410	Client Error	Gone
411	Client Error	Length required
413	Client Error	Request entity too large
414	Client Error	Request-URI too large
415	Client Error	Unsupported media type
420	Client Error	Bad extension
480	Client Error	Temporarily not available
481	Client Error	Call leg/transaction does not exist
482	Client Error	Loop detected
483	Client Error	Too many hops
484	Client Error	Address incomplete
485	Client Error	Ambiguous
486	Client Error	Busy here
500	Server Error	Internal server error
501	Server Error	Not implemented
502	Server Error	Bad gateway
503	Server Error	Service unavailable

Continued

**Table 3.6 continued** Listing of Responses, Response Codes, and Their Meanings

Response Code	Response Category	Response Description
504	Server Error	Gateway time-out
505	Server Error	SIP version not supported
600	Global Failures	Busy everywhere
603	Global Failures	Decline
604	Global Failures	Does not exist anywhere
606	Global Failures	Not acceptable

## Protocols Used with SIP

Although SIP is a protocol in itself, it still needs to work with different protocols at different stages of communication to pass data between servers, devices, and participants. Without the use of these protocols, communication and the transport of certain types of media would either be impossible or insecure. In the sections that follow, we'll discuss a number of the common protocols that are used with SIP, and the functions they provide during a session.

### UDP

The User Datagram Protocol (UDP) is part of the TCP/IP suite of protocols, and is used to transport units of data called *datagrams* over an IP network. It is similar to the Transmission Control Protocol (TCP), except that it doesn't divide messages into packets and reassembles them at the end. Because the datagrams don't support sequencing of the packets as the data arrives at the endpoint, it is up to the application to ensure that the data has arrived in the right order and has arrived completely. This may sound less beneficial than using TCP for transporting data, but it makes UDP faster because there is less processing of data. It often is used when messages with small amounts of data (which requires less reassembling) are being sent across the network, or with data that will be unaffected overall by a few units of missing data.

Although an application may have features that ensure that datagrams haven't gone missing or arrived out of order, many simply accept the potential of data loss, duplication, or errors. In the case of Voice over IP, streaming video, or interactive games, a minor loss of data or error will be a minor glitch that generally won't affect the overall quality or performance. In these cases, it is more important that the data is passed quickly from one endpoint to another. If reliability were a major issue, then the use of TCP as a transport protocol would be a better choice over hindering the application with features that check for the reliability of the data it receives.

## Notes from the Underground...

### UDP Denial-of-Service Attacks

Although denial-of-service (DoS) attacks are less common using UDP, data sent over this protocol can be used to bog down or even shut down a system that's victim to it. Because UDP is a connectionless protocol, it doesn't need to have a connection with another system before it transfers data. In a UDP Flood Attack, the attacker will send UDP packets to random ports on another system. When the remote host receives the UDP packets, it will do the following:

1. Determine which application is listening to the port.
2. Find that no application is waiting on that port.
3. Reply to the sender of the data (which may be a forged source address) with an ICMP packet of DESTINATION UNREACHABLE.

Although this may be a minor issue if the remote host has to send only a few of these ICMP packets, it will cause major problems if enough UDP packets are sent to the host's ports. A large number of UDP packets sent to the victim will cause the remote host to repeat these steps over and over. The victim's ports are monopolized by receiving data that isn't used by any application on the system, and ICMP packets are sent out to relay this fact to the attacker. Although other clients will find the remote host unreachable, eventually the system could even go down if enough UDP packets are sent.

To reduce the chances of falling victim to this type of attack, a number of measures can be taken. Proxy servers and firewalls can be implemented on a network to prevent UDP from being used maliciously and filter unwanted traffic. For example, if an attack appeared to come from one source previously, you could set up a rule on the firewall that blocks UDP traffic from that IP address. In addition to this, chargen and echo services, as well as other unused UDP services, could be either disabled or filtered. Once these measures are taken, however, you should determine which applications on your network are using UDP, and monitor for signs of a UDP Flood Attack or other signs of misuse.

## Transport Layer Security

Transport Layer Security (TLS) is a protocol that can be used with other protocols like UDP to provide security between applications communicating over an IP network. TLS uses encryption to ensure privacy, so that other parties can't eavesdrop or tamper with the messages being sent. Using TLS, a secure connection is established by authenticating the client

and server, or User Agent Client and User Agent Server, and then encrypting the connection between them.

Transport Layer Security is a successor to Secure Sockets Layer (SSL), which was developed by Netscape. Even though it is based on SSL 3.0, TLS is a standard that has been defined in RFC 2246, and is designed to be its replacement. In this standard, TLS is designed as a multilayer protocol that consists of:

- TLS Handshake Protocol
- TLS Record Protocol

The TLS Handshake Protocol is used to authenticate the participants of the communication and negotiate an encryption algorithm. This allows the client and server to agree upon an encryption method and prove who they are using cryptographic keys before any data is sent between them. Once this has been done successfully, a secure channel is established between them.

After the TLS Handshake Protocol is used, the TLS Record Protocol ensures that the data exchanged between the parties isn't altered en route. This protocol can be used with or without encryption, but TLS Record Protocol provides enhanced security using encryption methods like the Data Encryption Standard (DES). In doing so, it provides the security of ensuring data isn't modified, and others can't access the data while in transit.



---

The Transport Layer Security Protocol isn't a requirement for using SIP, and generally isn't needed for standard communications. For example, if you're using VoIP or other communication software to trade recipes or talk about movies with a friend, then using encryption might be overkill. However, in the case of companies that use VoIP for business calls or to exchange information that requires privacy, then using TLS is a viable solution for ensuring that information and data files exchanged over the Internet are secure.

---

## Other Protocols Used by SIP

As mentioned, SIP does not provide the functionality required for sending single-media or multimedia across a network, or many of the services that are found in communications programs. Instead, it is a component that works with other protocols to transport data, control streaming media, and access various services like caller-ID or connecting to the Public Switched Telephone Network (PSTN). These protocols include:

- Session Description Protocol, which sends information to effectively transmit data



- Real-Time Transport Protocol, which is used to transport data
- Media Gateway Control Protocol, which is used to connect to the PSTN
- Real-time Streaming Protocol, which controls the delivery of streaming media

The Session Description Protocol (SDP) and Real-time Transport Protocol (RTP) are protocols that commonly are used by SIP during a session. SDP is required to send information needed during a session where multimedia is exchanged between user agents, and RTP is to transport this data. The Media Gateway Control Protocol (MGCP) and Real-time Streaming Protocol (RTSP) commonly are used by systems that support SIP, and are discussed later for that reason.

### *Session Description Protocol*

The Session Description Protocol (SDP) is used to send description information that is necessary when sending multimedia data across the network. During the initiation of a session, SDP provides information on what multimedia a user agent is requesting to be used, and other information that is necessary in setting up the transfer of this data.

SDP is a text-based protocol that provides information in messages that are sent in UDP packets. The text information sent in these packets is the session description, and contains such information as:

- The name and purpose of the session
- The time that the session is active
- A description of the media exchanged during the session
- Connection information (such as addresses, phone number, etc.) required to receive media

#### **NOTE**

---

SDP is a standard that was designed by the IETF under RFC 2327.

---

### *Real-Time Transport Protocol*

The Real-Time Transport Protocol (RTP) is used to transport real-time data across a network. It manages the transmission of multimedia over an IP network, such as when it is used for audio communication or videoconferencing with SIP. Information in the header of the packets sent over RTP tells the receiving user agent how the data should be reconstructed and also provides information on the codec bit streams.

Although RTP runs on top of UDP, which doesn't ensure reliability of data, RTP does provide some reliability in the data sent between user agents. The protocol uses the Real-time Control Protocol to monitor the delivery of data that's sent between participants. This allows the user agent receiving the data to detect if there is packet loss, and allows it to compensate for any delays that might occur as data is transported across the network.

## NOTE

---

RTP was designed by the IETF Audio-Video Transport Working Group, and originally was specified as a standard under RFC 1889. Since then, this RFC has become obsolete, but RTP remains a standard and is defined under RFC 3550. In RFC 2509, Compressed Real-time Transport Protocol (CRTP) was specified as a standard, allowing the data sent between participants to be compressed, so that the size was smaller and data could be transferred quicker. However, since CRTP doesn't function well in situations without reliable, fast connections, RTP is still commonly used for communications like VoIP applications.

---

## *Media Gateway Control Protocol*

The Media Gateway Control Protocol (MGCP) is used to control gateways that provide access to the Public Switched Telephone Network (PSTN), and vice versa. In doing so, this protocol provides a method for communication on a network to go out onto a normal telephone system, and for communications from the PSTN to reach computers and other devices on IP networks. A media gateway is used to convert the data from a format that's used on PSTN to one that's used by IP networks that use packets to transport data; MGCP is used to set up, manage, and tear down the calls between these endpoints.

## NOTE

---

MGCP was defined in RFC 2705 as an Internet standard by the IETF. However, the Media Gateway Control Protocol is also known as H.248 and Megaco. The IETF defined Megaco as a standard in RFC 3015, and the Telecommunication Standardization Sector of the International Telecommunications Union endorsed the standard as Recommendation H.248.

---

## *Real-Time Streaming Protocol*

The Real-Time Streaming Protocol (RTSP) is used to control the delivery of streaming media across the network. RTSP provides the ability to control streaming media much as you would control video running on a VCR or DVD player. Through this protocol, an application can issue commands to play, pause, or perform other actions that effect the playing of media being transferred to the application.

### NOTE

---

IETF defined RTSP as a standard in RFC 2326, allowing clients to control streaming media sent to them over protocols like RTP.

---

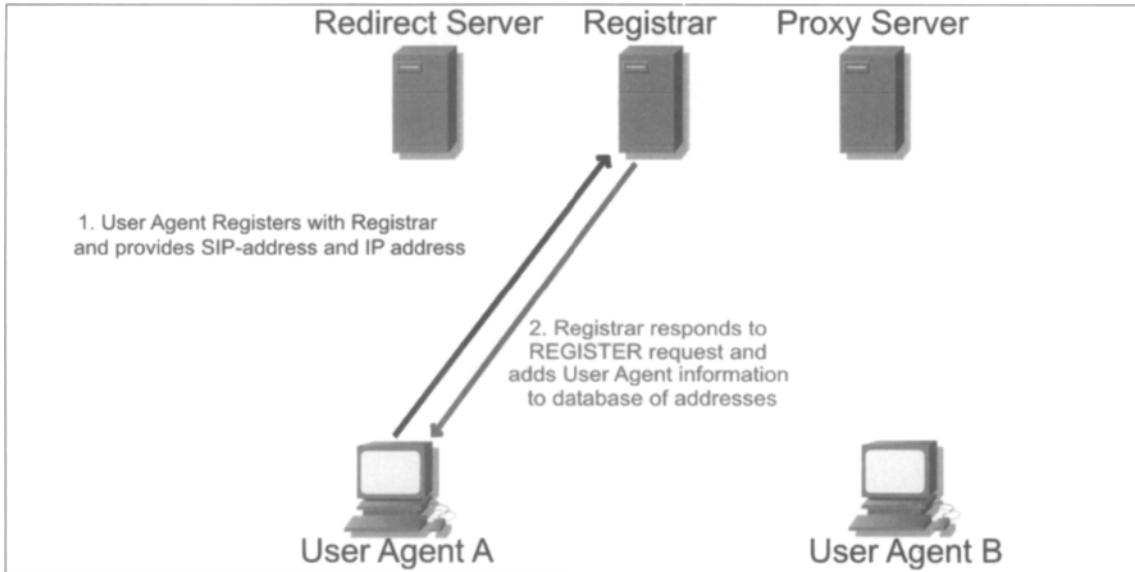
## Understanding SIP's Architecture

Now that we've looked at the various components that allow SIP to function on an IP network, let's look at how they work together to provide communication between two endpoints on a system. In doing so, we can see how the various elements come together to allow single and multimedia to be exchanged over a local network or the Internet.

The User agents begin by communicating with various servers to find other User agents to exchange data with. Until they can establish a session with one another, they must work in a client/server architecture, and make requests of servers and wait for these requests to be serviced. Once a session is established between the User agents, the architecture changes. Because a User agent can act as either a client or a server in a session with another User agent, these components are part of what is called a peer-to-peer (P2P) architecture. In this architecture, the computers are equal to one another, and both make and service requests made by other machines. To understand how this occurs, let's look at several actions that a User agent may make to establish such a session with another machine.

## SIP Registration

Before a User agent can even make a request to start communication with another client, each participant must register with a Registrar server. As seen in Figure 3.20, the User agent sends a REGISTER request to the SIP server in the Registrar role. Once the request is accepted, the Registrar adds the SIP-address and IP address that the User agent provides to the location service. The location service can then use this information to provide SIP-address to IP-address mappings for name resolution.

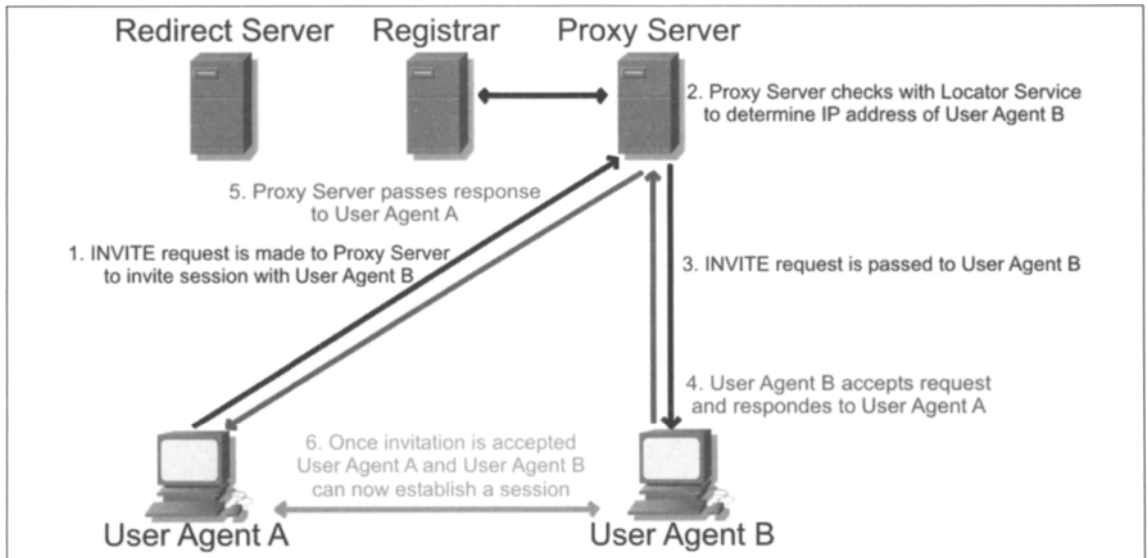
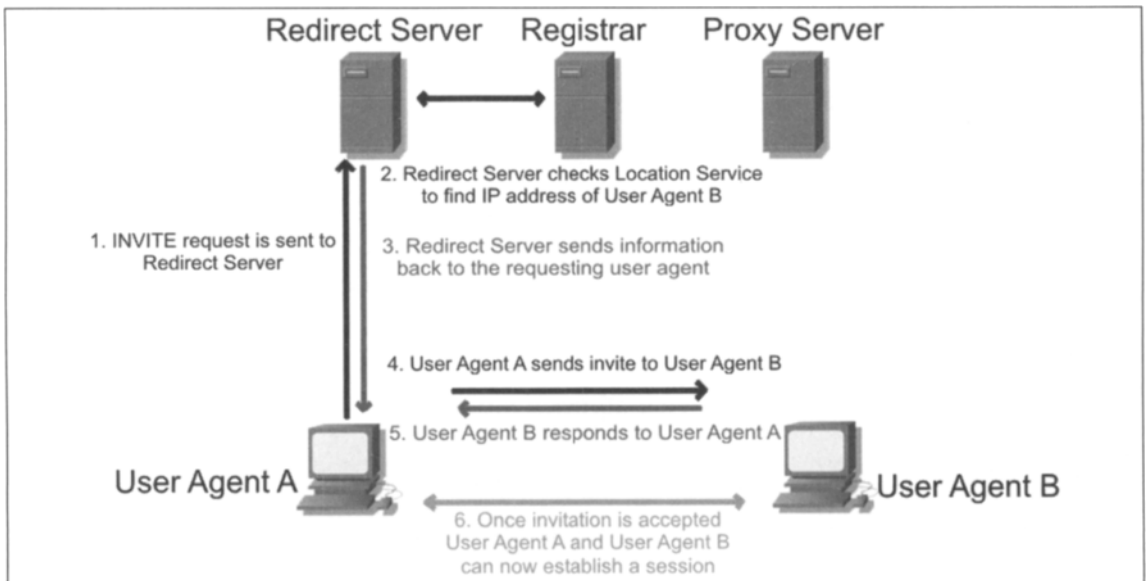
**Figure 3.20** Registering with a SIP Registrar

## Requests through Proxy Servers

When a Proxy Server is used, requests and responses from user agents initially are made through the Proxy server. As seen in Figure 3.21, User Agent A is attempting to invite User Agent B into a session. User Agent A begins by sending an INVITE request to User Agent B through a Proxy server, which checks with the location service to determine the IP address of the client being invited. The Proxy server then passes this request to User Agent B, who answers the request by sending its response back to the Proxy server, who in turn passes this response back to User Agent A. During this time, the two User agents and the Proxy server exchange these requests and responses using SDP. However, once these steps have been completed and the Proxy server sends acknowledgements to both clients, a session can be created between the two User agents. At this point, the two User agents can use RTP to transfer media between them and communicate directly.

## Requests through Redirect Servers

When a Redirect server is used, a request is made to the Redirect server, which returns the IP address of the User agent being contacted. As seen in Figure 3.22, User Agent A sends an INVITE request for User Agent B to the Redirect server, which checks the location service for the IP address of the client being invited. The Redirect server then returns this information to User Agent A. Now that User Agent A has this information, it can now contact User Agent B directly. The INVITE request is now sent to User Agent B, which responds directly to User Agent A. Until this point, SDP is used to exchange information. If the invitation is accepted, then the two User agents would begin communicating and exchanging media using RTP.

**Figure 3.21** Request and Response Made through Proxy Server**Figure 3.22** Request Made through Redirect Server

## Peer to Peer

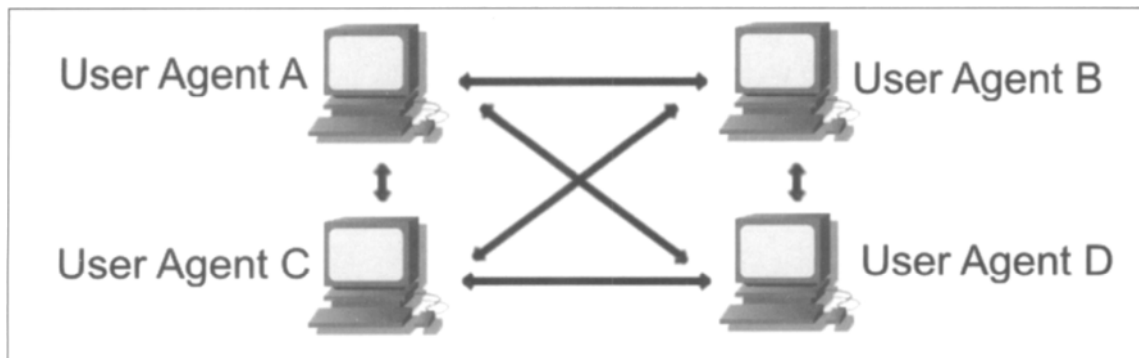
Once the user agents have completed registering themselves, and making requests and receiving responses on the location of the user agent they wish to contact, the architecture changes from one of client/server to that of peer-to-peer (P2P). In a P2P architecture, user

agents act as both clients who request resources, and servers that respond to those requests and provide resources. Because resources aren't located on a single machine or a small group of machines acting as network servers, this type of network is also referred to as being *decentralized*.

When a network is decentralized P2P, it doesn't rely on costly servers to provide resources. Each computer in the network is used to provide resources, meaning that if one becomes unavailable, the ability to access files or send messages to others in the network is unaffected. For example, if one person's computer at an advertising firm crashed, you could use SIP to communicate with another person at that company, and talk to them and have files transferred to you. If one computer goes down, there are always others that can be accessed and the network remains stable.

In the same way, when user agents have initiated a session with one another, they become User agent clients and User agent servers to one another, and have the ability to invite additional participants into the session. As seen in Figure 3.23 each of these User agents can communicate with one another in an audio or videoconference. If one of these participants ends the session, or is using a device that fails during the communication, the other participants can continue as if nothing happened. This architecture makes communication between User agents stable, without having to worry about the network failing if one computer or device suddenly becomes unavailable.

**Figure 3.23** Once SIP Has Initiated a Session, a Peer-to-Peer Architecture Is Used



## Instant Messaging and SIMPLE

Instant messaging (IM) has long been one of the most common and popular methods of communicating over IP networks. Whereas VoIP uses voice communication and videoconferencing uses live images and sound, IM simply uses text messages to allow participants to converse. These text messages are sent in real-time between the users who use the same IM application, and allows an individual to essentially create a private chat room with another individual where they can send text messages to one another. Many applications will even provide the

ability to add additional participants to the chat, creating a text-based conference room of multiple users.

To manage the messages and identify whether specific users are online, an extension of SIP for instant messaging has been developed. SIMPLE is an acronym that stands for the *Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions*. Although the name is ironically less than simple to remember, it is being developed as an open standard for how individuals can determine the status of a person (i.e., whether they are online, busy, etc.), and for managing the messages that go back and forth between the participants in a chat.

## Instant Messaging

In different variations, instant messaging has been around longer than the Internet has been popular. In the 1970s, the TALK command was implemented on UNIX machines, which invoked a split screen that allowed users of the system to see the messages they typed in individual screens. In the 1980s, Bulletin Board Systems (BBSes) became popular, where people would use a modem to dial into another person's computer to access various resources, such as message boards, games, and file downloads. On BBSes, the system operator (SYSOP) could invoke a chat feature that allowed the SYSOP to send messages back and forth with the caller on a similar split-screen. If the BBS had multiple phone lines, then the callers could Instant message with each other while they were online. As the Internet gained popularity, the ability to exchange messages with other users became a feature that was desired and expected.

Today there are a large number of IM applications that can be used to exchange text messages over the Internet and other IP networks. Although this is nowhere near a complete list, some of the more popular ones include:

- AIM, America Online Instant Messenger
- ICQ
- Yahoo Messenger
- MSN Messenger

In addition to these, there are also applications that allow communication using VoIP or other multimedia that also provide the ability to communicate using text messages. Skype provides a chat feature that allows two or more users to communicate in a private chat room (see Chapter 10).

One of the important features of any IM application is the ability to keep a contact list of those with whom you routinely communicate. In many programs the contact list is also known as a *Buddy List*. However, even with this listing, it would be impossible to contact anyone if you didn't know when each contact was available. If a person had a high-speed

connection and was always connected to the Internet, then they might always appear online. As such, they would need a way of indicating that they were online but not available, or whether the person was available for one form of communication but not another. The ability to display each contact's availability in a Buddy List when someone opens an IM application is called *presence*.

## SIMPLE

SIMPLE is an extension of SIP, which is used for maintaining presence information and managing the messages that are exchanged between the participants using instant messaging. Just as SIP registers users with a SIP server before they can begin a session, SIMPLE registers presence information. When a user registers through SIMPLE, those with this user in their Buddy List can access information that the user is online. When the people who have the user in their lists are alerted that the user is online, they can initiate a chat. If the user needs to do some work and changes their status to busy, or goes away from their desk and changes their status to being away, then this information is updated in the IM applications that have this person as a contact. Generally, the presence of a user is indicated in these programs through icons that change based on the user's status.

Because SIMPLE is an extension of SIP, it has the same features and methods of routing messages. The users are registered, and then send text-based requests to initiate a session. The messages are sent between user agents as individual requests between User agent clients and User agent servers. Because the messages are small, they can move between the two User agents quickly with minimal time lag even during peak Internet hours.

Although the IETF IM and Presence Protocol Working Group are still developing SIMPLE as a standard, it has been implemented by a number of IM applications. Windows XP was the first operating system to include SIMPLE, and is used by Microsoft Windows Messenger, and numerous other IM applications also are using SIMPLE as a standardized method for instant messaging.

### Are You Owned?

#### Compromising Security with Instant Messaging

Instant messaging has become a tool that not only is used by the public for pleasure, but also one that is used by companies for business. IM software can be used as an alternative method of communicating with salespeople, customers, suppliers, and others who need to be contacted quickly. Because it is an effective communication tool, businesses have found benefits implementing it as part of their communications systems.

Continued



Unfortunately, a drawback of IM applications is that it provides a potential gap in security. Although companies will monitor outgoing e-mail for illegal or inappropriate content, IM applications available to the public don't provide a centralized method of logging conversations that can be locked down. IM applications routinely offer a method of logging conversations, but these settings can be toggled on and off by the person using the program. This means that someone could inadvertently or maliciously provide sensitive information in Instant messages without anyone at the company every realizing it.

Added to this problem is the fact that IM applications provide the ability to transfer other forms of media between participants. IM applications can be used for file sharing, where one person sends a file to another through the program. This can result in activities like sharing music files at work, which albeit illegal is relatively harmless, but it could also cause major issues if sensitive corporate files were being sent. Imagine an employee at a hospital or doctor's office sending patient files, or a disgruntled employee sending out a secret formula to the public or competition, and its impact becomes more apparent.

Because files may contain more than you bargained for, the possibility of spyware or viruses being disseminated through instant messaging must also be considered. Some applications that have supported instant messaging include additional software that is spyware, which can obtain information about your system or track activities on your system. Even if the IM software used on a machine doesn't include spyware, the files sent between participants of a communication session can contain viruses or other malicious code. By opening these files, the person puts their computer and possibly their local network at risk.

If a company wishes to allow IM software installed on their machines, and doesn't want to block IM communications to the Internet, they need to educate users and install additional software on the computers. Just as employees should know what information should not be discussed on a telephone or sent by mail, they should know these same facts, and files should be off-limits in other communications. In addition to this, anti-virus software should be installed, and regularly updated and run. To determine if spyware is installed on the machines, they should either invest in anti-virus software that also looks for these programs or install additional software that searches for and removes them from the computer. In performing these steps, the risks associated with IM applications in a business can be decreased, making it safer for both the user and the company.

## Summary

Today's PSTN is more powerful than ever; it is now capable of delivering services that Alexander Bell could not have ever imagined (like dedicated Internet access and SONET-based Internet backbone links). The telecommunications industry that cares for the PSTN affects our everyday lives from our traditional telephone lines, cell phones, Internet access, wireless solutions, and even cable television. The act of making a single phone call requires instantaneous network performance. The networks that make up the PSTN always are responding to a fast-changing environment that continues to demand increased reliability and capability.

Digital multiplexing started with time division, but now includes wavelength division, having come nearly full circle with old analog frequency division multiplexing. In all these cases, increased capacity from the outside cable plant was created in response to increased demand for telecommunications bandwidth.

The design of the PSTN has changed from one centered on a human operator to one leveraging large-scale automated switches that handle thousands of calls at once. Located within each central office are the thousands of individual local loops coming in, such as the voice DS0s, plus DSL and many digital circuits from subscribers that are then collected via a DACS and presented up the network on high-speed digital interfaces to the switch. Adherence to industry-standard signaling and technological protocols, such as the SS7 and SIP, is necessary, but it may not be sufficient as the number of interconnected carriers continues to multiply.

H.323 is a complex protocol suite. A number of H.323 VoIP-related protocols create channels made up of dynamic IP address/port combinations. Each terminal-terminal conversation requires, at a minimum, four channels to be opened—two control channels per endpoint (one H.225 and one H.245), and two unidirectional voice channels. Three of these (excepting the H.225 signaling traffic) will be on dynamically allocated ports. In addition, users naturally expect to be able to make both inbound and outbound calls. Because H.323 relies heavily on dynamic ports, traditional packet-filtering or stateful inspection firewalls are not a viable solution, as every port greater than 1024 would have to be opened to everyone on the Internet. Additionally, H.323 contains embedded addressing information (port numbers) that is not rewritten by most NAT implementations.

Therefore, most firewall solutions supporting H.323 must at least disassemble the control stream packets (H.245, H.225.0) and dynamically open up the firewall as needed. All these features make the implementation of H.323 security complex. As if this is not enough complexity, signaling and control messages are binary encoded according to ASN.1 rules. ASN.1 parsers have been exploited in a variety of implementations, and parsing takes time—adding latency to an already latency-sensitive application. H.323-aware firewalls, ALGs, and session border controllers (SBCs) have proven to be up to the task of effectively securing H.323 traffic without exposing internal networks to external attack.

SIP works in conjunction with a variety of other protocols and specialized servers to provide communication between participants. Through SIP, a User agent is able to find the location and availability of other users, the capabilities of the software or device they're using, and then provides the functions necessary to set up, manage, and tear down sessions between participants. This allows participants to communicate directly with one another, so that data can be exchanged effectively and (if necessary) securely.

SIP is a standard of the Internet Engineering Task Force (IETF) under RFC 3261, and maps to the application layer of the OSI reference model. Because it isn't a proprietary technology, implementations of it can be used on any platform or device, and can be used on any IP network. In addition to this, SIP also makes use of other standards, such as URIs, which are used to identify the accounts used in SIP.

SIP's architecture is made up of a number of different protocols and components that allow it to function. Its architecture begins as a client/server architecture, in which requests are made to SIP servers. As the servers service these requests, they allow the participants to eventually communicate directly with one another, changing the architecture to a distributed peer-to-peer. As information is passed between these machines, a variety of different protocols are used, allowing data to be passed quickly between the computers, and securely if needed.

Instant messaging is another technology where SIP is being used. An extension of SIP called SIMPLE is used to maintain presence information and manage messages that are exchanged between the participants. Because SIMPLE provides the same features as SIP and is also an open standard, it is being used increasingly in IM software, making SIP and SIMPLE a staple in communications on IP networks.

## Support Protocols

### Solutions in this chapter:

- DNS
- TFTP
- HTTP
- SNMP
- DHCP
- RSVP
- SDP
- SKINNY

# Introduction

Protocols such as MGCP and SIP, or protocol umbrella groups like H.323, are usually the first things that come to mind when discussing VoIP technology. Although they are all great protocols in their own right, they depend on, and interoperate heavily with, support protocols. Many of the support protocols that are used by VoIP architectures enable services and features required for proper network operation.

This chapter will cover several of the support protocols typically found in VoIP environments and some of the security implications that they bring with them. This chapter is not intended to be an all-inclusive tutorial on these protocols. Instead, the intent is to review both their use and any security implications involving your network.

## NOTE

---

It is important to keep in mind that most of these support protocols do not include any encryption or authentication mechanisms by default. For this reason, most of this traffic is susceptible to interception and/or modification. Proper network planning and configuration is thus essential.

---

## DNS

The Domain Name System (DNS) is a static hierarchical name resolution architecture that relies on client/server communication for operation. DNS is a protocol that many use every day and may not know it. Whenever someone browses the Internet, DNS is used in the background to translate host names into IP addresses so that the proper network destinations can be found. DNS is equally important in VoIP networks for its ability to resolve destination endpoint addresses or allow gateway registration to call servers and gatekeepers by host name.

DNS was created so that no one would be required to memorize the IP addresses of every host on a private network or the Internet. Most people have a hard enough time remembering one or two passwords, let alone several billion IP addresses. With the development of DNS, the only requirement is knowledge of the target Web page name that you wish to go to. DNS resolves the target Web page name entered into one or more server IP addresses. It has also been designed to allow the reverse or “inverse” resolution of IP addresses to host names.

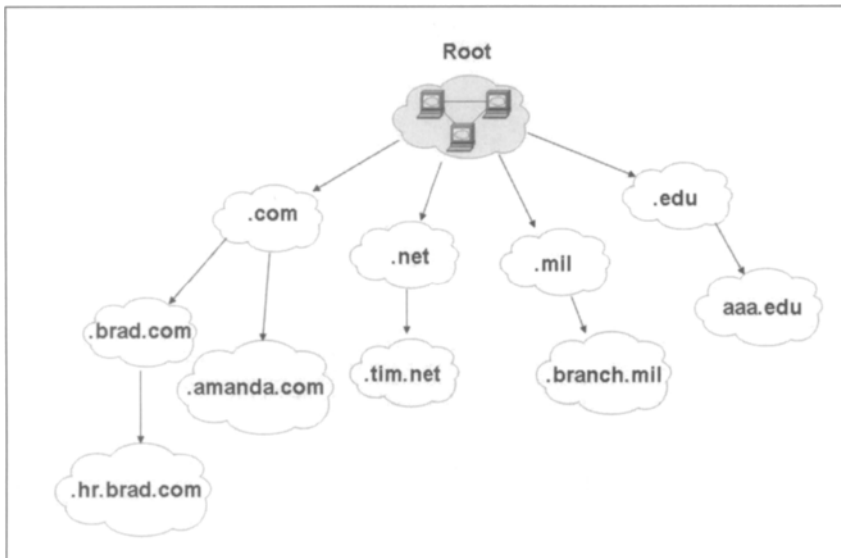
The DNS architecture was first discussed in detail in RFCs 881 through 883, and later updated in RFCs 1034 and 1035. Several of the newer RFCs include recommendations for how to secure the DNS architecture, including the addition of DNS security extensions

(DNSSEC) beginning with RFC 4033. The next few sections detail a high-level overview of the DNS architecture and several security threats associated with DNS systems.

## DNS Architecture

In order to better understand and be able to address the security concerns associated with DNS properly, it is important to have at least a high-level understanding of how DNS works. The hierarchy previously mentioned for DNS exists as a pyramid, with the highest level of the DNS architecture at the top. DNS is organized into myriad logical groupings called domains, which are further segmented into an endless number of subdomains. Figure 4.1 illustrates a sample hierarchy of the DNS system and is by no means exhaustive. The intent is to show the structure of the hierarchy.

**Figure 4.1** Sample DNS Architecture

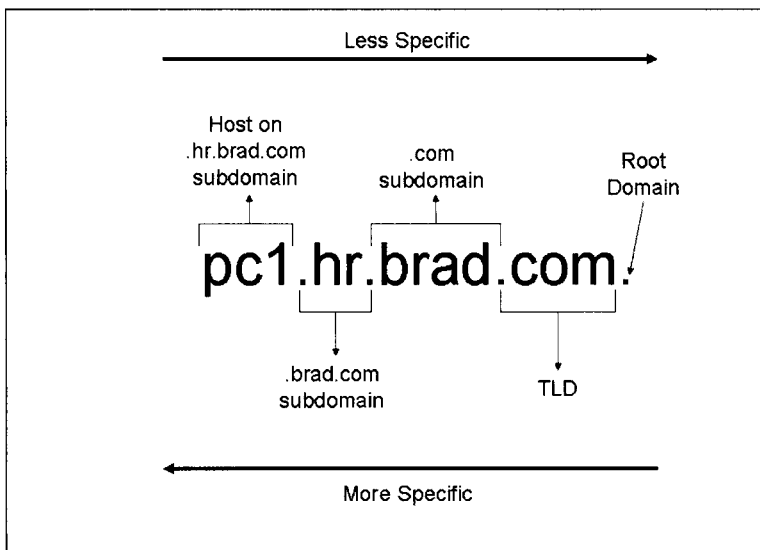


Located at the top of the DNS hierarchy are the root DNS servers. The root DNS servers are located in the root DNS zone, annotated by a single “.”, and are responsible for maintaining the location of the top-level domain servers (TLD). A TLD DNS server is one that is responsible for the management of one of the commonly associated address suffix identifiers, such as .com, .net, .edu, or .org. The TLD DNS servers are assigned or “delegated” the responsibility by the root DNS servers. They are known as the authoritative server for that TLD. Likewise, the TLD DNS servers delegate the management of one of their many subdomains. The subdomain DNS servers for .brad.com would be responsible for any resource records (RR) for that subdomain as well as the location of any related subdomains (.hr.brad.com). The resource records are the entries for the host systems. This process of delegation distributes the load of the DNS system across many different servers.

## Fully Qualified Domain Name

Each host has its own pointer for DNS, known as a fully qualified domain name (FQDN). The FQDN is used to identify the path taken through the DNS architecture to find the requested host. Figure 4.2 illustrates what path is taken through the previously discussed DNS hierarchy from Figure 4.1 to reach host pc1.

**Figure 4.2** Fully Qualified Domain Names



There are a couple of things to keep in mind about FQDNs. First of all, the explicit FQDN path from the top of the hierarchy (root) is read from right to left. Secondly, even though most FQDN illustrations do not include the final dot to represent the root domain, it is an implied part of the complete FQDN. Most applications, like IE, will not append a trailing "." to the end of a requested Web resource. Followed from right to left, the host pc1 follows a path out of the root domain, through the TLD .com, to the .com subdomain .brad.com, and then finally into the .brad.com subdomain of .hr.brad.com.

FQDNs are entered into the DNS tables as one of several types of RRs:

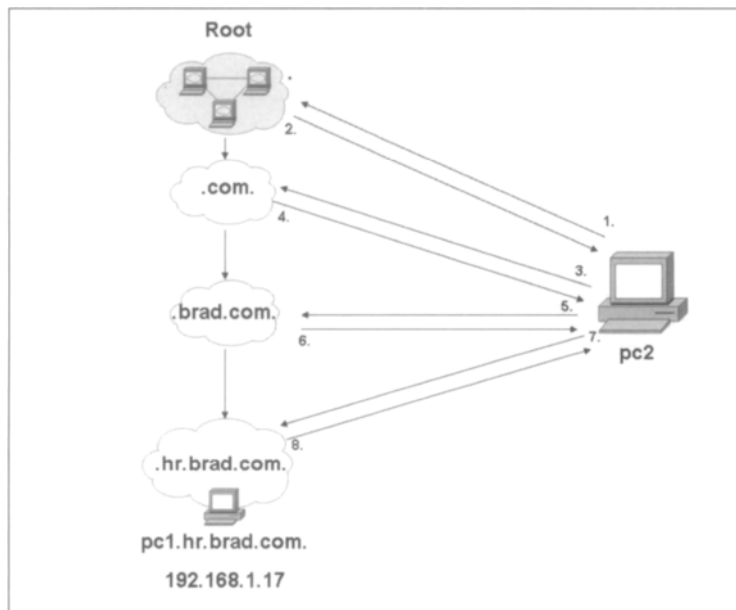
- **A** An A record is an address record, denoting a standard host entry in the DNS table. The key here is that it is used to resolve an FQDN to an IP address.
- **PTR** PTR records are used by the inverse lookup zones in DNS. The PTR record resolves an IP address to an FQDN.
- **SOA** The SOA record identifies zone information such as the zone name and serial number.
- **MX** MX records identify mail servers for the zone.

- **NS** NS records are used for name servers for the zone.
- **CNAME** CNAME records act as alias records to allow for the translation of one host name into another.
- **INFO** Provides information about hosts listed in the DNS table.
- **SRV** SRV records identify SIP servers for the zone.

## DNS Client Operation

In order to locate the IP address for a host, the client's application will send a request to a resolver on the same client system. The resolver will then formulate and send out the DNS query. From a high level, the query will typically follow a path of trial and error known as a recursive lookup. Figure 4.3 illustrates what a recursive lookup from a host, pc2, would look like to find the IP address for host pc1.

**Figure 4.3** Recursive Lookups Using DNS



1. The client's resolver sends its DNS query, which will be sent to the root domain.
2. The root domain server does not have the RR for the host pc1, so the response is sent to redirect the resolver on pc2 to the TLD DNS server for .com since it knows where .com is.
3. The resolver, in turn, sends a query to the TLD DNS server for .com.



4. The TLD DNS server does not have the RR for the host pc1, so the response is sent to redirect the resolver on pc2 to the .brad.com. DNS server since it knows where .brad.com. is.
5. The resolver, in turn, sends a query to the DNS server for .brad.com.
6. The .brad.com. DNS server does not have the RR for the host pc1, so the response is sent to redirect the resolver on pc2 to the .hr.brad.com. DNS server since it knows where .hr.brad.com. is.
7. The resolver, in turn, sends a query to the DNS server for .hr.brad.com.
8. The authoritative DNS server for .hr.brad.com. has the RR for the host pc1 and sends back the information to pc2. pc2 now has the IP address information for pc1, and may use it accordingly.

## NOTE

---

It is not required to have a separate DNS server for each subdomain. A single DNS server may be the authoritative server for many, or all, of the subdomains in a corporation, although there are usually backup DNS servers configured for each primary DNS server.

---

## DNS Server Operation

The DNS server is responsible for cataloging all of the RRs that belong to any of the zones that it is the authoritative DNS server for. It is also responsible for keeping track of any of the DNS servers that it has delegated subdomain responsibility to. By keeping track of the subdomains, the DNS server is able to redirect client queries to the proper location in the event that the requested host RR does not reside on that server.

DNS servers may also be configured to maintain a cache of domain names, as well as their respective IP addresses, as they are requested by clients. This configuration allows a DNS server to retrieve an IP address only once and then store the value for any subsequent queries by the same client or any other client. These entries are cached for only a short period of time, equal to the Time To Live (ttl) value applied to the record. When a client requests a particular domain name resolution, the DNS server will first attempt to find the records in its local database. If this search fails, the DNS server will attempt to contact a root name server, if it's been configured to do so, to request the value.

Another important function that the DNS servers provide is the replication of the DNS table, also known as a zone transfer. The zone transfer insures that all entries for a given zone will be available on all DNS servers in that zone. This is necessary so that DNS can provide

a resilient operating architecture. Two types of zone transfers can be found between DNS servers: full and incremental. A full zone transfer is exactly as it sounds, a complete transfer of zone information between DNS servers. An incremental zone transfer, on the other hand, is one where only changed zone information is exchanged between DNS servers. Incremental zone transfers make more efficient use of bandwidth and network resources, but not all DNS server vendors support the newer implementation.

Zone transfers are based on several items, including serial numbers and refresh intervals. The secondary DNS server will request a zone transfer from the primary DNS server and there is a serial number embedded in the response. If the secondary server receives the response and the serial number is lower than or equal to the serial number of its current table version, the response will not be used to update the server's table. However, if the serial number is higher, the DNS table will be updated to what is enclosed in the response.

The refresh interval is used to identify how often the secondary server should request a zone transfer from the primary server. It is used as a polling mechanism to help ensure that the secondary server remains up-to-date with the current DNS information. NOTIFY messages may also be used by the primary DNS server to tell the secondary DNS servers when changes have been made to the DNS table. When the secondary DNS server receives the NOTIFY, they can request a zone transfer to ensure table synchronization.

## Security Implications for DNS

DNS is a core component of modern networking, and as such, is a rather attractive target for many attackers. When the DNS architecture was developed, security was not included as part of the design. There was nothing designed into the architecture for peer authentication, origin authentication, or data encryption. Some recent advancements in DNS have helped to alleviate some of the current security concerns, but they have not been able to remove them altogether.

The dangers of DNS are well publicized and well documented, owing to its long life on the Internet. More information on these security threats, how they are performed, and how to protect your DNS servers can be found at [www.dnssec.net/dns-threats.php](http://www.dnssec.net/dns-threats.php). There is also an RFC on DNS Threats, published as RFC 3833. Several types of attacks should be kept in mind regarding your DNS deployment, and some best practices can be employed to help lessen your exposure:

- DNS footprinting (using DNS zone data to learn host names, subdomains, and subnets)
- Denial of Service (DoS)
  1. SYN flooding of DNS server
  2. Transfer of blank DNS table
- DNS cache poisoning

## TFTP

The Trivial File Transfer Protocol (TFTP) is a simplified protocol used to transfer files from a server to a client. Unlike more evolved file protocols, such as FTP, TFTP was designed to work in pure simplicity, requiring less overhead and interaction. Its primary usage today is in computers and devices that do not have storage devices, commonly known as “thin client PCs.” Without offline storage, especially one that can be updated, it is difficult to maintain how such devices can operate. Instead of booting off of a hard drive or flash ROM, these devices use TFTP to request data from a central server to boot from. Or, such devices can boot from internal ROM memory and use TFTP to request configuration data to use during their operation. Also, devices can use TFTP to request firmware updates which they can then flash to their ROM chips to update the built-in software code. This is especially useful since customized sets of data can be stored for individual user devices within a corporate environment.

The role of TFTP in transferring data is well used throughout the computer industry. Virtually all modern computers support the ability to boot from the network. In this mode, the computer will attempt to locate a TFTP server on its network segment once it boots. In finding one, the client requests a bootable image from the server, usually in the form of a floppy disk image. Once it has received the data, the client will then proceed to boot from the image, as if it was an actual floppy disk or CD-ROM.

In the VoIP community, TFTP has a critical role in allowing VoIP devices and telephones to obtain configuration data from centralized servers. These devices are built with internal Flash ROM memory chips that contain simplified hardware architecture that does not allow for continual write access to memory. Instead, data is only written once to the device’s memory and read continuously by the internal operating system.

The TFTP protocol was first described in 1980 as IEN (Internet Experiment Note) 133. Its first formal RFC was RFC 783, which was later updated in RFC 1350. However, there are various RFCs that also describe individual actions and abilities that TFTP could be used for. These include Bootstrap loading (RFC 906) and TFTP multicasting (RFC 2090). The next few sections of the chapter detail a high-level overview of the TFTP architecture and several related security threats associated with the protocol.

## TFTP Security Concerns

In order to better understand and be able to address the security concerns associated with TFTP properly, it is important to have at least a high-level understanding of how TFTP works. Unlike most other file transfer protocols, TFTP operates by transmitting UDP packets. While connection-less UDP packets are generally frowned upon for reliable data transmissions, they allow for a simpler implementation into the protocol, as well as faster transfer speeds. The abilities of the protocol are also very limited, allowing only for the ability to read and write data. The protocol does not have any mechanism displaying information about

available files and directories on a server. The client must know the name of the file that they wish to download when connecting.

There are very strict regulations on how data is sent between computers, which allows for client applications to be written easier. Similar to the FTP protocol, TFTP allows for data to be sent as either ASCII or binary. This data is sent in individual UDP packets between the two devices. Of these packets, five types can be transmitted, each one identified by an operation code in the header of the data.

- Read Request (RRQ)
- Write Request (WRQ)
- Data
- Acknowledgement (ACK)
- Error

## TFTP File Transfer Operation

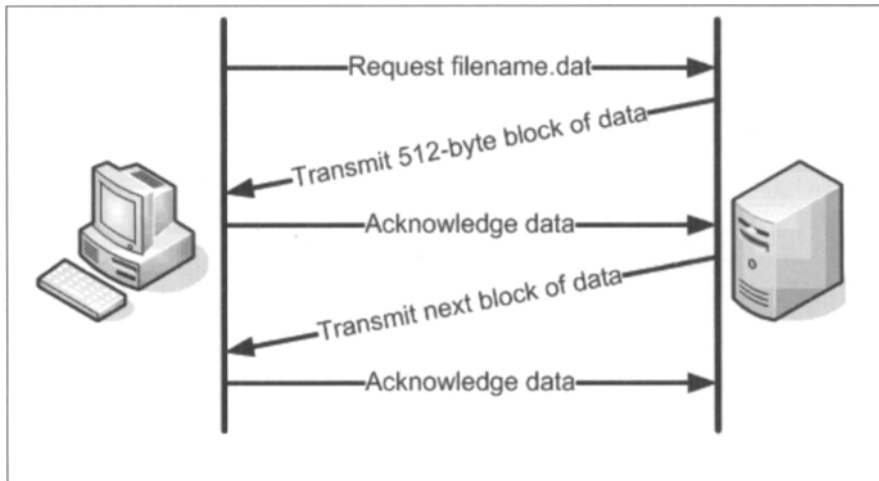
When a client wishes to download a file from a TFTP server, it first sends a Read Request (RRQ) packet to the TFTP server. This packet identifies itself as an RRQ packet, and also specifies both the name of the file the client wishes to download and the data mode (binary or ASCII). Likewise, if the client wishes to upload a file to a TFTP server, it sends an identical Write Request (WRQ) packet, which also contains the file name and data mode. The sending computer then immediately starts sending data packets to the recipient computer. If the data is greater than 512 bytes in size, multiple packets will be sent. A packet that contains a data portion smaller than 512 bytes is seen as the last packet in the transfer. Following the receipt of each data packet, the receiving computer sends an acknowledgement (ACK) packet to the sender, notifying it that the transfer was successful. Figure 4.4 details this transfer of data between two computers.

## Security Implications for TFTP

Insomuch that TFTP was designed for simplicity and ease of use, any mechanisms normally used to secure data were not implemented into its protocol. It was originally planned by engineers that usernames and passwords should not ever be required for TFTP access, but this has led to many security issues. This concern is also greater because all TFTP packets are sent in the clear across a network, with no data encryption. Given there is no authentication, and no encryption, TFTP is generally not recommended for the transfer of sensitive data. However, its role as a “bootstrap protocol” could allow usernames and passwords to be transferred in the clear across a network when these aren’t protected by higher-level mechanisms. Since TFTP is often used to download boot images from a remote server, and these images often contain sensitive data required to connect into various servers on the network, it is possible to retrieve

stored account information from within these boot images. Any person who is capturing network traffic on the same network segment as the TFTP session could easily gather the transferred data and re-create the original file. If the file contains sensitive data, such as usernames and passwords, it would then be readily available to anyone capturing the traffic.

**Figure 4.4** TFTP Data Transferral



### WARNING

The TFTP protocol sends all data in clear text across the network. As it is commonly used to transfer configuration data to devices and clients, it is important to verify that there is no sensitive data contained within transferred data. Otherwise, anyone sniffing the wire could have access to various usernames and passwords used by such devices.

## HTTP

The HyperText Transfer Protocol (HTTP) is one of the most well known, and well used, protocols on the Internet. It is the protocol by which Web pages are transmitted from Web servers to clients, but it is also used by many other applications to send data between computers. For example, many peer-to-peer clients make use of the solid structure of HTTP to transfer data segments of shared files between peers. HTTP can be used to transmit both ASCII and binary data between computers.

HTTP is commonly used in the VoIP community as a way for administrators to remotely administer and configure devices. Many network management devices offer a Web-based administration panel by which the device can be altered and configured for a

particular environment. Many such devices also require user authentication to be able to fully access the configuration data.

HTTP was first described in RFC 1945 at HTTP 1.0 by its founder, Tim Berners-Lee. Currently, RFC 2616 is used to describe the HTTP 1.1 protocol; however, various other RFCs describe additional extensions and uses for the HTTP protocol. These include HTTP Authentication (RFC 2617), Secure HTTP (RFC 2660), and CGI (RFC 3875).

## HTTP Protocol

The function of HTTP and its protocol was designed to be very straightforward and usable by many applications. When a client wishes to request a file from an HTTP server, it simply creates a TCP session with the server and transmits a GET command with the name of the requested file and the HTTP protocol version (for example, GET /index.html HTTP/1.1). The HTTP server then responds back with the appropriate data. The response from the server will be either the data requested by the client, or an error message describing why it cannot send the data. All of the commands within the HTTP protocol are sent in regular ASCII text, with each line followed by a carriage return/line feed (CR/LF). In network logs, the CR/LF appear as hexadecimal 0x0D0A.

## HTTP Client Request

For a client to retrieve data from an HTTP server, it must know the exact filename and location to construct an appropriate file request. For most purposes, this information is supplied in the form of a uniform resource locator (URL), which specifies a particular HTTP server, directory path, and file name (for example, www.digg.com/faq/index.php). When a client wishes to view this specific page, index.php, it must first make a connection to www.digg.com. This is performed by resolving the domain name to an IP through DNS, which results in the IP address of 64.191.203.30. The client then initiates a TCP connection to 64.191.203.30 and makes a request of GET /faq/index.php HTTP/1.1. This request also includes other information about the client, some of which may be required for HTTP 1.1, such as the host value. An example of a full HTTP GET request is shown next:

```
GET /download.html HTTP/1.1
Host: www.ethereal.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.6) Gecko/20040113
Accept: \
text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain; \
q=0.8,image/png,image/jpeg,image/gif;q=0.2,*/*;q=0.1
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
```

```
Connection: keep-alive
Referer: http://www.ethereal.com/development.html
```

## HTTP Server Response

Upon receiving a GET request from a client, a server first ensures that the file requested does exist. If it does, the data is then sent back to the requesting client. If not, an error message is sent. Regardless of the action, a specific server response is sent back to the client that includes a status code. This status code informs the client of the response type. The most common is a 200 code, which informs the client that the file was found and will be sent. It is transmitted in the form of HTTP/1.1 200 OK, which specifies the HTTP protocol version, the status code, and a brief description of the code. Other common status codes include “404 Not Found,” which indicates that the requested file could not be located by the server, and “500 Internet Server Error,” which indicates that there is a problem with the HTTP server. The following is an example of an HTTP response:

```
HTTP/1.1 200 OK
Date: Thu, 13 May 2004 10:17:12 GMT
Server: Apache
Last-Modified: Tue, 20 Apr 2004 13:17:00 GMT
Accept-Ranges: bytes
Content-Length: 18070
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=ISO-8859-1
```

## Security Implications for HTTP

Due to the simple design of HTTP, and the early state of the Internet when it was unveiled, security wasn't a high priority in the protocol. All data sent through HTTP was sent as clear text, which allowed any person to be able to sniff the traffic flowing across the wire and parse out sensitive data, such as usernames, passwords, and network configuration data. This is particularly dangerous since many VoIP and network management devices use HTTP as a means to allow administrators to check the status of the device and to configure additional settings. A person with malicious intent on the same network segment as the device could pick out various usernames and passwords that may work on additional computers or devices.

HTTP also supports multiple forms of authentication, which is a means by which the HTTP server can verify a user's identity. The two authentication forms currently used are basic and digest authentications. When a server supports authentication, it sends a 401 “Authentication Required” response to clients that request sensitive data. This response will

also include a “realm” (a name associated with the Web site) that notifies the user what they are accessing. When a client receives such a response, it will provide a log-in window to the user to input a valid user name and password. These values will then be transmitted back to the requesting server for verification. Because of HTTP’s design, though, these credentials will have to be constantly transmitted to the server for every further data transmission. Each of these transactions will transmit the user name and password in the clear.

Another form of authentication supported by modern HTTP clients and servers is digest authentication, which is described in depth in RFC 2617. Digest authentication has an advantage over basic authentication in that it does not send a clear password over the network. Instead, an MD5 (Message Digest) value of the password is transmitted to the requesting server. The server then uses this digest value for password comparisons. However, digest authentication is not fully supported in many older Web browsers. It also does not fully protect a user’s credentials. The user name and other information about the user are still transmitted in the clear. And, even though the password is obfuscated, a skilled, malicious user can still capture the MD5 value and use it for future transactions with that particular server to use another person’s account.

Many devices have recently provided support for HTTPS to overcome the openness of the HTTP protocol. HTTPS is a modification of HTTP wherein all data between a client and server are encrypted using the Secure Sockets Layers (SSL). In order for HTTPS to function, both the server and the client must be able to support it, and it must be specifically chosen as the form of communication in the URL. For example, instead of `http://www.foo.com`, a secure connection would use `https://www.foo.com`.

## SNMP

SNMP, short for Simple Network Management Protocol, is a high-level protocol and architecture that allows for the monitoring and maintenance of network devices to detect problems, and to fine-tune the network for performance. There are two key versions of SNMP in use today, SNMPv1 and SNMPv2. While the two share many commonalities, there are some very beneficial additions made to SNMPv2. However, as many people disagreed with the security profiles implemented into SNMPv2, it has remained less popular and less used than SNMPv1. Since that time, a newer version of SNMP was released: the Community-Based SNMP, or SNMPv2c. However, the current standard, adopted in 2004, is SNMPv3. SNMP plays a useful role in maintaining and administering VoIP networks by allowing a person the ability to easily monitor the bandwidth and performance of all the major components of a network.

The SNMP protocol is defined under RFC 1157 as SNMPv1, and the characteristics of its immediate successor, SNMPv2, are defined in RFC 1902. SNMPv2c is officially detailed in RFC 1901 and in RFC 1908. SNMPv3 is defined in RFC 3411 and RFC 3418.

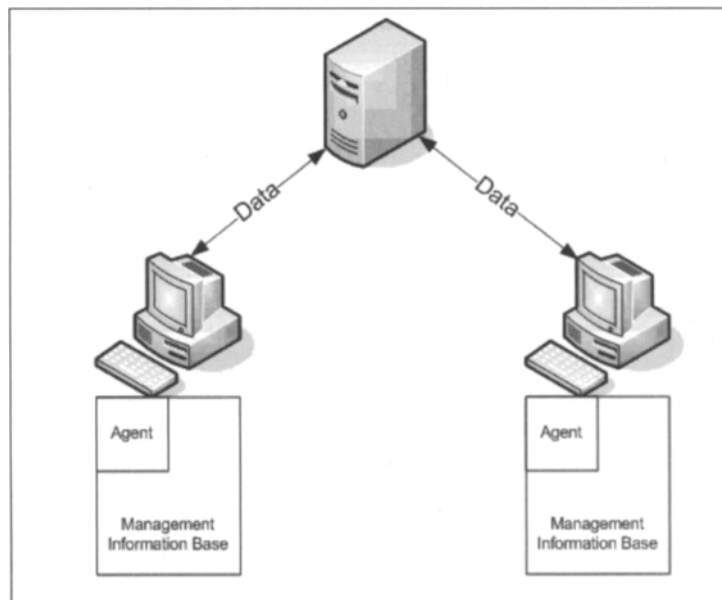


## SNMP Architecture

An SNMP implementation on a network involves three components to be integrated: the devices to be managed, agents, and Network Management Systems (NMSes). The devices to be managed are simply computers or devices on the network that reside on the network. These are the devices that an administrator would like to monitor on the network. Each device must have an agent installed on them, which is a software application that continually monitors the device for predefined events or errors and transmits them to a centralized management server, an NMS. The NMS collects all of the data that is routinely transferred from the various network devices and correlates it into useful information for an administrator to read and evaluate.

However, even with all of these components working together on a network, there still must be a structure to all of the individual data that can be gathered across a network by an NMS. This is implemented by the use of a Management Information Base (MIB). See Figure 4.5 for a diagram on how these components work together.

**Figure 4.5** SNMP Network Components



## SNMP Operation

The SNMP protocol works under a very simplified model of data collection and control of the managed devices. Only a few basic commands are used in the SNMP protocol, such as GETREQUEST, GETNEXTREQUEST, SETREQUEST, and TRAP. An NMS invokes GETREQUEST to collect data from a device, and GETNEXTREQUEST to retrieve the

next value in a set. An NMS can also invoke the SETREQUEST command to save data to a managed device. The TRAP command is the only one not initiated by the NMS; it is sent out by the client to report any unusual activity it has detected.

On the client side, the Management Information Base (MIB) acts as a tree that catalogs all of the various data components of the system or device. Each of these data components are known by their object identifiers (OIDs). The OID is made up of multiple sets of numbers, each separated by a period, in a structured order similar to that of an IP address. As a general rule, all OIDs begin with .1.3.6.1.2.1, except on many Cisco devices which use .1.3.6.1.4.1.9. To request a data value, an established OID must be specified. For example, to request the system up time, OID .1.3.6.1.2.1.1.2 is read.

## SNMP Architecture

The SNMP protocol has many areas that require careful attention and configuration simply due to the amount of information that could be leaked out to malicious users. Since all of this data is retrievable by anyone requesting it, there must be some safeguards put in place to prevent unauthorized users from being able to read data, or modify it. This is performed by the use of a community string. A community string acts as a password to group data into either read-only or read-write areas. By default, most software is setup to use a default community string of “public” for their read-only data. Likewise, many implementations use a default community string of “private” for their read-write data. It is particularly dangerous to leave such community strings in place, as they are well known to malicious users, and an unchanged read-write community string allows an attacker the ability to modify critical data on a device.

### Are You Owned?

#### Are You Allowing Sensitive Data to Be Leaked?

Due to the open nature of SNMP, allowing any person to easily request data, unique community strings should be defined for network components that you can administer. Proper care must also be taken in evaluating IP telephones to ensure that they do not have unsecured SNMP access available. Otherwise, all of your SNMP-enabled components, such as workstations, servers, routers, and phones, can disclose sensitive information to anyone who asks. Unless you are constantly monitoring network traffic, you may not even know that this information is being gathered by malicious people within your network environment—or, even worse, being modified to cover unauthorized actions.

Continued

This issue came to light recently when it was discovered that the Cisco 7920 Wireless IP Phone contained a fixed community string that allowed malicious users to gather and modify data on the devices. The vulnerability and its fix were given a Cisco bug ID of CSCsb75186. They can also be reviewed at <http://securitytracker.com/id?1015232>.

Likewise, similar SNMP vulnerabilities surfaced with the Hitachi IP5000 phone. These devices did not have a protected community string, which meant that any person could have full SNMP access to all of the data on the device, including the ability to alter and erase it.

On a lesser scale, the UTstarcom F1000 IP phone featured the default public community string, which allowed anyone to view data stored on the phone, some of which could be considered sensitive. Additionally, when using SNMP scanning software, the phone suffered from numerous SNMP issues that required a full reboot to fix.

## DHCP

The Dynamic Host Configuration Protocol (DHCP) is a protocol that was designed to allow network configuration of clients and workstations. Every workstation and device that is making use of a network must be assigned a unique IP address, as well as assigned a subnet mask and gateway IP address. In a network environment where there are hundreds, or thousands, of workstations, this could become an administrative nightmare. DHCP is a popular answer to this problem, automatically assigning IP addresses and other relevant configuration information to each individual device as it comes online.

DHCP is a critical support protocol in the VoIP world because it allows VoIP phones and devices to be portable from one network to another. Instead of manually configuring the device after plugging it into each network, the device simply “pings” the network to find an existing DHCP server. The device then automatically receives an IP address and network details from the server and is then immediately useable on the network, without any interaction with the user.

The DHCP protocol was first discussed in RFC 1531 and RFC 1541 in 1993. Currently, RFC 2131 describes DHCP, and has made the previous RFCs obsolete. There are many RFCs that describe additional extensions and uses for DHCP, though—for example, DHCP for IEEE 1394 (RFC 2855) and DHCP for SIP servers (RFC 3361).

## DHCP Protocol

The primary function of DHCP is to supply critical network information to clients automatically, to reduce the effort of a network administrator in manually configuring various devices on a network. For DHCP to work, there must be a DHCP server (or relay) running on the network segment where clients will be connecting. The DHCP server listens con-

stantly for incoming UDP packets on port 67, a port reserved for DHCP usage. When a new, DHCP-enabled device is connected to the network, it sends a broadcast packet to detect any running DHCP servers. The DHCP server then responds with a DHCP offer, which contains an assigned IP address.

Eight types of packets are used within the DHCP protocol:

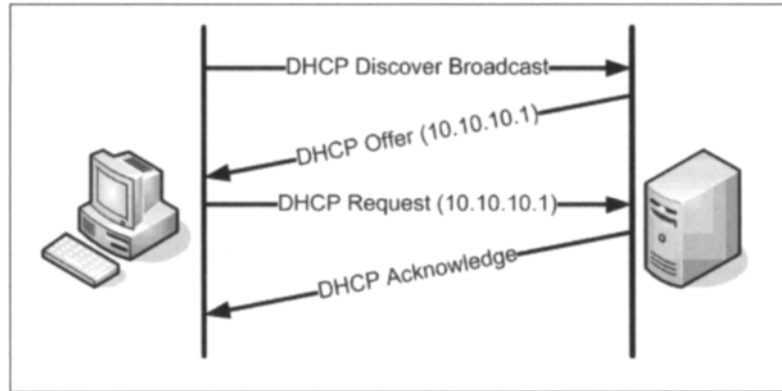
- Discover
- Offer
- Request
- Decline
- ACK
- NAK
- Release
- Inform

## DHCP Operation

When a client first joins a network, either by being plugged into the network segment or by being powered on, it does not have an IP address assigned to it. In order to request one, it sends a DHCP Discover packet across the network. It does so by sending a packet from IP address 0.0.0.0 to the broadcast IP address 255.255.255.255, which allows the packet to reach every single device on the network segment. This packet may include information about the client itself, such as the network interface's MAC address and the computer's designated host name.

Once a server has received a DHCP Discover packet, it immediately checks its preset range (scope) of IP addresses to determine the next available number. Optionally, the DHCP server will also compare the requestor's MAC address against a local table to determine if the client is allowed to receive an IP address. After an address has been chosen, a DHCP Offer packet is transmitted back to the requesting client, targeted by its MAC address. This packet includes the assigned IP address, the lease time of the IP address, subnet mask, gateway address, and chosen DNS servers, as well as other network information that is to be implemented into the client.

Once the client has received a DHCP Offer packet, it responds with a DHCP Request packet. This packet is similar to the original DHCP Discover packet in that it is sent from 0.0.0.0 to 255.255.255.255. This packet serves to notify the server that the client has accepted the assigned IP address, and also notifies all other clients on the network segment that the assigned IP address has been taken. Finally, the server responds back to the client with a DHCP Acknowledgement (ACK) to confirm the address has Request has been received. This communication between the client and DHCP server is detailed in Figure 4.6.

**Figure 4.6** The DHCP Process

## Security Implications for DHCP

A variety of security concerns come into play whenever DHCP is enabled on a network segment. These security issues don't deal so much with leaked data such as passwords. Instead, they focus more on access into a network from unauthorized clients. A basic DHCP server runs under the assumption that any DHCP Discover and Request should be honored as an authorized client. In this setup, any device that requests network information will be able to receive it, no questions asked. However, this opens the door for any person with physical access to the network to be able to plug in unauthorized devices and receive network access.

A number of ways exist to reduce this network exposure, from modifying the network switches to modifying the DHCP configuration. Most of these security implementations involve verifying the MAC address of the client device before allowing it to receive an IP address. One of the more extensive fixes is to enable port security on the implemented network switches. With port security in place, the physical connection port can be locked to allow only a single MAC address access through it. This can help prevent employees, or contractors, from installing a small network hub or wireless router, and giving multiple devices access to the network.

However, an easier method is to provide DHCP addresses just to devices that have a particular MAC address assigned to them. All network devices have a MAC address coded into them, and these addresses follow a set structure. The first six bytes of the MAC address specifies the vendor ID, or the company that manufactured the device. If you wish to restrict DHCP to just particular VoIP phones or devices on your network, this is possible by identifying the vendor ID on the devices and configuring the DHCP server to provide addresses only to devices that have the same vendor ID. For example, Grandstream Networks VoIP phones all have a vendor ID of 00:0B:82.

Another security issue that can arise with DHCP is coupled with TFTP, and the security risks associated with it. If a network uses a TFTP server to transmit bootable disk images to computers, much of the configuration material to specify where these particular disk images are located is located within the DHCP responses. When clients receive a DHCP offer, they can choose to take advantage of this information, depending on their boot states. However, a malicious user could monitor these packets to determine the location of any TFTP servers, as well as the particular files used on these servers.



---

To ease the installation of IP telephones, create a separate scope of IP addresses with a MAC filter to only allow IP telephones to lease an address. Collect the unique vendor IDs from the authorized telephones to create this filter.

---

## RSVP

RSVP, short for the **R**esource **R**e**S**er**V**ation **P**rotocol, is a protocol designed to allow clients on networks to negotiate bandwidth to provide and maintain a high Quality of Service (QoS) for a specific connection. Normally, TCP/IP will make a best effort to route packets from one machine to another as quickly as possible. However, due to the dynamic routing of internetworking, where packets take completely different routes each time they are transmitted, this cannot be guaranteed. This creates a special issue for VoIP communication, which requires a high QoS to maintain seamless and non-interruptive communication between two people. VoIP can be an especially demanding protocol that requires long periods of high bandwidth and low latency, and without RSVP, these conditions may fall below acceptable levels which could result in a loss of quality or disconnections. RSVP allows a dedicated path across a network between each client so that packets are routed randomly around, which retains a high level of bandwidth, and less latency. RSVP is especially useful for WAN connections within a global organization to maintain these set paths inside a network, as many Internet routers do not support the protocol.

The RSVP protocol was first described in RFC 2205 in late 1997. Further modifications were made to this RFC, and the best current practices for the RSVP protocol are now discussed in RFC 3936, created in late 2004. There are also other RFCs that describe additional extensions and uses for the RSVP protocol. These include RSVP for LSP Tunnels (RFC 3209) and RSVP security properties (RFC 4230).

## RSVP Protocol

The RSVP protocol works by transferring UDP packets from the recipient of the data transfer to its sender. This allows the data recipient to control whether to use regular TCP/IP or to use a dedicated path of travel between the two clients. The connection recipient initiates this path by sending a constructed RSVP packet to the connection initiator. This packet will contain a specific Message Type that indicates the action that should be acted upon. The common Message Types for an RSVP protocol are

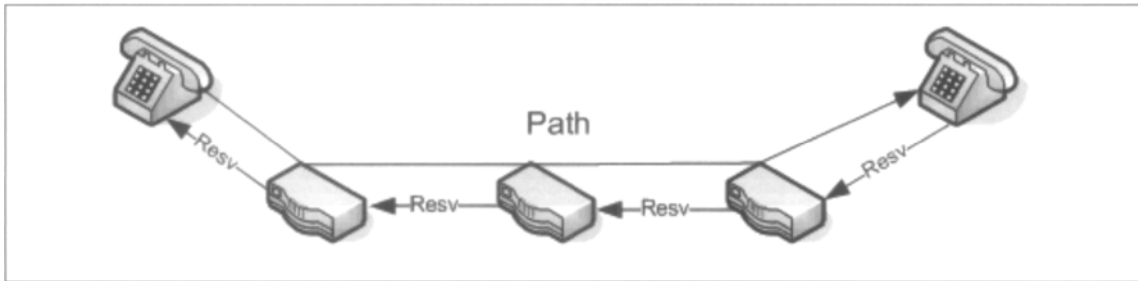
- Path
- Resv (Reservation Request)
- PathErr (Path Error)
- ResvErr (Reservation Error)
- PathTear (Path Teardown)
- ResvTear (Reservation Teardown)
- ResvConf (Reservation Confirmation)

The RSVP packet also carries a data payload containing specific information on how the path should be constructed. The payload contains information such as:

- Session (Destination IP, Tunnel ID, Extended Tunnel ID)
- Hop (the neighboring router's IP)
- Time Values (the refresh interval)
- Explicit Route (a list of routers between the two devices that creates the data path)
- Adspec (specifies the minimum path latency, MTU, and bandwidth requirements)

## RSVP Operation

To create a dedicated path of travel, the RSVP protocol relies heavily on its Path and Resv messages. The Path message packet is used to define the path of routers to be used for communication between the two clients. This packet is sent from the receiving end of the communication towards the sender. As it passes through each individual router, the router examines the packet to determine its neighboring IP addresses, to which it must route packets to. The Resv message, or Reservation request, is equally important. The Resv message is sent from each router to its neighboring router, one hop at a time. The Resv packet helps create the reservation on each router involved in the path. The transfer of Path and Resv packets is detailed in Figure 4.7.

**Figure 4.7** Creating an RSVP Path

Once a path has been created, with each router maintaining a reservation for the data, it must be updated routinely to remain open. If a router has not received a Resv and Path packet before the refresh interval on the path has been exhausted, then the router will remove the reservation from itself. As Resv and Path packets arrive to maintain the reservation, they may also make changes to it. If the path between the clients is to change to substitute routers, the recipient just sends a new Path message with the updated path and it will become effective. Each router will continually update its stored information based on the packets it continually receives during the transmission.

Once the communication between the two devices has ended, they initiate a teardown of the path. Although, realistically they could just stop transmitting RSVP packets and eventually the reservations on the routers would expire, it is recommended that they formally tear down the path immediately after finishing the connection. The teardown may be initiated by either side of the communication, or from any of the routers within the communication. A PathTear packet may be sent downstream from the sender, or a ResvTear may be sent upstream from the receiver. As each router in the path receives a teardown packet, they will immediately remove the path reservation and forward the packet onto the next hop in the path.

## Security Implications for RSVP

Many of the security issues with the RSVP protocol involve actions that a person with malicious intentions could take to either disrupt traffic or capture it. For one, as the Path and Resc packets are transmitted across the network, they each include a session ID that can be used to uniquely identify a particular RSVP session. This data is also sent as clear text, where anyone who is armed with a network sniffer can capture the data. Knowing the session ID, a person could then use the same session ID and send a Path message to one of the routers in the path. This new Path could alter the path of the network, leading the network transmission to a completely different client than intended. Or, it could be used to disrupt the communication completely, preventing an RSVP connection to take place between the two devices.



There are various solutions that have come about to resolve issues like this. For one, the Session ID could be encoded into a public key that will be included in each packet, as well as a timestamp that acts as a digital signature. If the two devices are within the same localized network, a third-party server could be used to establish the identities of each device. Many such security implications and solutions were drafted by various authors, including Hannes Tschofenig, in an Internet Draft located at [www.tschofenig.com/drafts/draft-ietf-nsis-rsvp-sec-properties-06.txt](http://www.tschofenig.com/drafts/draft-ietf-nsis-rsvp-sec-properties-06.txt).

## SDP

SDP, short for Session Description Protocol, is a simple protocol that allows clients to share information about a multimedia stream to clients wishing to connect. Further extensions on the protocol also allow clients to share their multimedia abilities with other devices. As its name denotes, it is used primarily to describe a client's session abilities. It plays an integral part in VoIP communications to share the fact that a communication session is taking place, and to provide information to other clients so that they have the ability to join and interact with the session, such as with a group teleconference.

SDP was first described in RFC 2327 in April 1998, and the original RFC still defines the protocol's basic abilities today. There are updates, though, to the RFC, such as RFC 3266, which adds IPv6 support to SDP. Other associated RFCs include the RTCP attribute in SDP (RFC 3605), TCP-Based Media Transport in SDP (RFC 4145) and PSTN/Internet Interworking (PINT), a set of extensions to SIP and SDP for IP Access to Telephone Call Services (RFC 2848). A fairly recent RFC, RFC 3407, allowed the clients the ability to share their multimedia abilities to other devices.

## SDP Specifications

SDP is used as a specification protocol, not as an actual transport protocol (or even a session negotiation protocol, although higher-level protocols like SIP may add that capability above it). In other words, SDP does not actually transfer data between clients, it just establishes a structure for communicating the attributes for those data streams. The data must be transferred using another transport protocol, such as SAP, SIP, RTSP, or HTTP. The information contained within an SDP packet is in ASCII text, and although it was not designed for human readability, it is easy to decipher. An SDP packet is broken into multiple lines of text, where each line represents a single field and its corresponding value. Common data fields include

- **v** (Protocol Version)
- **o** (Owner of session, Session ID, Session Version, Network Type, Address type, and Owner's IP Address)

- **s** (Session name)
- **i** (Session description)
- **u** (URI of subject material)
- **e** (E-mail address of Session Point of Contact)
- **p** (Phone number of Session Point of Contact)
- **c** (Connection information: IP version and CIDR IP address)
- **k** (Encryption key as clear text, base64, uri, or prompt)
- **m** (Media type, connection port, transport method, and format list)
- **t** (Session begin and end times)
- **a** (Attribute)

The following is an example of SDP data for supplying capabilities:

```
v=0
o=bsmith 2208988800 2208988800 IN IP4 68.33.152.147
s=-
e=bsmith@foo.com
c=IN IP4 20.1.25.50
t=0 0
a=recvonly
m=audio 0 RTP/AVP 0 1 101
a=rtpmap:0 PCMU/8000
a=rtpmap:1 GSM/8000
a=rtpmap:101 telephone-event/8000
```

## SDP Operation

Once a device has been queried, usually by a client sending an SIP request, it forms an SDP packet to send back. This SDP packet supplies all of the critical information about the session capabilities that the device offers. In its simplest form, this data contains the owner information, the audio and video codecs supported, and which ports connections are accepted on. In queries for particular sessions, the reply contains the session name, the session description, connection ports, and the range of time when the session will be active. All time stamps in SDP data are formed using Network Time Protocol (NTP) values.

Additionally, the session ID and session version, which must be unique values, are generally created using NTP values to signify the current date and time.

Much of the current SDP usage is documented in RFC 4317, which describes the SDP Offer/Answer model. In this model, when a client wishes to communicate with another, it

transmits an SDP offer packet. This packet is arranged in a structure similar to the following example, provided by RFC 4317:

```
v=0
o=alice 2890844526 2890844526 IN IP4 host.atlanta.example.com
s=
c=IN IP4 host.atlanta.example.com
t=0 0
m=audio 49170 RTP/AVP 0 8 97
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/8000
m=video 51372 RTP/AVP 31 32
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
```

Reading through this packet, you can see that the owner line describes that the packet sender is “alice,” who is listening for connections on `host.atlanta.example.com`. This data is sent to the person with whom she wishes to communicate. Once the other person has received the data and wishes to continue the connection, an answer packet is returned. Here is an example of this answer:

```
v=0
o=bob 2808844564 2808844564 IN IP4 host.biloxi.example.com
s=
c=IN IP4 host.biloxi.example.com
t=0 0
m=audio 49174 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 49170 RTP/AVP 32
a=rtpmap:32 MPV/90000
```

In this example, Alice is initiating a connection with Bob. Alice’s Offer packet identifies that she supports three types of audio connections (PCMU, PCMA, and iLBC), as well as two types of video connections (H.261 and MPV). Once Bob’s client has received the invitation and parsed the values, it chooses a compatible audio and video format and responds back. In the answer packet shown earlier in this chapter, Bob’s client responds back wishing to communicate with PCMU audio and MPV video.

## Security Implications for SDP

Similar to the security issues of RSVP, much of the security implications for SDP arise due to the fact that a person can easily read session IDs and connection information off of a net-

work segment and then tamper with existing communications. In seeing existing connection offers, and their corresponding SDP replies, an eavesdropper could use the information to determine devices that are allowing VoIP communications, and also spoof his way into an existing communication. An attacker may also be able to collect SDP offers and replay them at a later time, overriding values for ongoing communications, with the potential to disable audio feeds. However, nearly all security issues with SDP can be solved by using protocols to handle user authentication, such as SIP.

## Skinny

The Skinny protocol is the casual name for a complex, lightweight VoIP protocol signaling scheme owned by Cisco Systems, Inc., and is in use for all VoIP telephones that Cisco produces. The formal name is SCCP, for Skinny Client Control Protocol, and was originally designed by the Selsius Corporation, which Cisco acquired. Skinny is a proprietary protocol that allows “skinny clients”, such as Cisco IP telephones, to communicate with each other via Cisco CallManager (CCM). The Skinny clients are small, user-friendly devices that work in conjunction with a CCM. The CCM also acts as a proxy to relay communications to H.323 clients and the PSTN.

## Skinny Specifications

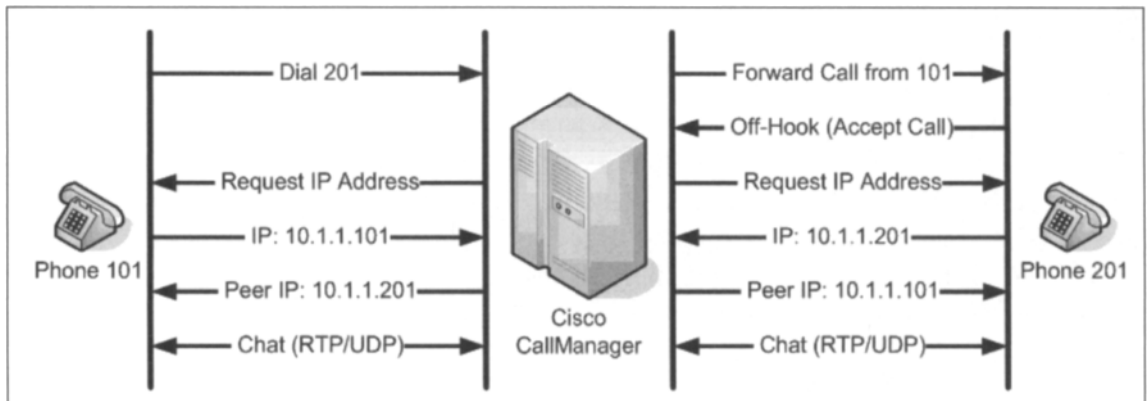
Skinny (SCCP) is the exclusive protocol used by Cisco brand IP telephones, as well as some phones developed by other manufacturers. Using the Skinny protocol, an IP phone will use normal TCP/IP to communicate with the Cisco CallManager. If the Cisco phone needs to communicate with a non-Skinny client, then the CCM acts as a proxy gateway, allowing the two to communicate, at which time the phones will start using UDP. However, when a Skinny phone wishes to communicate with another Skinny phone, the two will use RTP/UDP packets for communication.

## Skinny Operation

The ability for Skinny clients to communicate with each other is governed by the Cisco CallManager (CCM) on the same network. When an IP phone wishes to dial another on the same network, the user takes the phone off-hook and begins dialing the necessary numbers. As the numbers are entered, they are transmitted to the CCM over TCP packets. The CCM performs a “digit analysis” to determine if they match another phone number in the database. If so, the CCM communicates with the receiving phone, causing it to start ringing and to send a ring back to the calling phone. Once the second phone goes off-hook, the CCM sends packets to both phones requesting their IP address and open UDP port on which to accept the RTP media. The CCM also checks the media capabilities of each phone to determine if they can directly communicate with each other, or if a transcoder is

required to allow the communication. Once the CCM has received the connection information from each phone, it proceeds to transmit the information to the other phone, so that each phone has the connection information of its peer. At this point, the CCM creates an RTP/UDP channel for the phones to pass data through for communication. Once either of the phones goes on-hook and disconnects the line, the CCM terminates the channel. An example of this connection process is shown in Figure 4.8.

**Figure 4.8** The Skinny Client Communication Process



## Security Implications for Skinny

Similar in implications to the other protocols discussed previously, the largest problem with the Skinny (SCCP) protocol is the fact that all traffic that uses it is sent in the clear, with no encryption taking place unless the device is capable and configured to support Transport Layer Security (TLS). Ultimately, this means that people with malicious intent on the same network segment are able to capture the traffic using a network sniffer. This allows such people to store recorded conversations, or to even capture the numbers that a particular phone dials during a time period.

### NOTE

While the SCCP/Skinny protocol was not designed for the transfer of secure data, some protocols are. Cisco CallManager 4.0 introduced Secure SCCP, or simply "Secure Skinny" to add beefier security to a Cisco VoIP network. Secure SCCP encrypts all data between IP telephones and the Cisco CallManager using TLS.

Certain Cisco CallManager versions also suffer from a known vulnerability. This vulnerability takes advantage of malformed SCCP packets sent to a vulnerable Cisco IOS (internal operating system). If successful, the exploit is able to cause devices, or the entire CCM, to reboot. The issue is documented as Cisco bug ID CSCee08584, and can be fixed by upgrading or migrating the IOS of the affected hardware.

## Summary

While there are more popular and interesting protocols in place to handle much of the VoIP traffic on networks and the Internet, there is also a very important set of support protocols that doesn't share as much of the limelight. These protocols are crucial in making sure VoIP networks can operate, and that individual clients can communicate with each other quickly and efficiently. However, they also all have their own specific security risks and implications when implemented.

DNS is one such protocol which is required for most usage on the Internet. As the means by which domain names are resolved to IP addresses, it has ultimate control over where to send clients that are asking for directions to a particular machine. Proper care must be taken to ensure that network clients are using appropriate DNS servers that can be trusted to direct devices properly. TFTP is mentioned as one of the primary protocols used to transfer small data files between a server and a device. Though its primary usage is in transferring bootable images to thin clients, TFTP is also critical in supplying configuration information to devices that do not have the means to store data. However, this configuration information could be sensitive in nature, if it contains authentication information, and due to the protocol design, it will be sent in the clear on the network, allowing anyone listening to gather it. HTTP is one of the most popular, and well-used, protocols in use today and is the primary means for users to download data from Web servers. It is also commonly used in other applications and areas as a way to transfer data between computers. However, if SSL is not used, the information is also sent in the clear and is thus visible to network sniffers.

SNMP is one of the more useful protocols for network administrators since it allows applications to create a central repository of data involving all networked devices on a network segment. This data can then be used to monitor network activity, improve performance, or locate and resolve issues as they occur. It is also a protocol implemented into many VoIP telephones in use today. However, as shown earlier, many implementations of SNMP were not done correctly in some IP phone models, allowing malicious users to gather, modify, or erase data contained within these devices. DHCP is another useful protocol for many network administrators across the world. DHCP allows IP addresses to be leased out to computers as they come online, abolishing the practice of manually configuring each and every network device with a unique IP address. The use of DHCP allows for a greater number of devices to use a network during a day since many components are not running continuously. However, it dangerously supplies IP addresses and network information to unauthorized clients. Various methods of protecting your network from this are available, however, as discussed in this chapter.

RSVP is an important protocol in the VoIP world since it allows for static pathways to be constructed between two VoIP telephones across a network, or the Internet. This pathway uses Quality of Service controls to maintain a high-bandwidth connection between the two devices to avoid static and dropped connections. However, due to its unencrypted design, it

is also possible for unauthorized users to track the pathways, and even change them in mid-stream, severing communications between devices. SDP was also mentioned as a data format protocol used to provide information about an ongoing telephony session, or to just provide information about what protocols a particular device is capable of communicating with. Its open design allows unauthorized users to detect and track ongoing communication sessions, and to even disrupt them. Finally, the Skinny (SCCP) protocol was discussed, being Cisco Systems proprietary protocol used for their internal VoIP network implementations. The Skinny protocol uses the Cisco CallManager system to make connections to other telephones within the network segment, or to devices on other network segments.



This Page Intentionally Left Blank

## Threats to VoIP Communications Systems

### Solutions in this chapter:

- Denial-of-Service or VoIP Service Disruption
- Call Hijacking and Interception
- H.323-Specific Attacks
- SIP-Specific Attacks

## Introduction

Converging voice and data on the same wire, regardless of the protocols used, ups the ante for network security engineers and managers. One consequence of this convergence is that in the event of a major network attack, the organization's entire telecommunications infrastructure can be at risk. Securing the whole VoIP infrastructure requires planning, analysis, and detailed knowledge about the specifics of the implementation you choose to use.

Table 5.1 describes the general levels that can be attacked in a VoIP infrastructure.

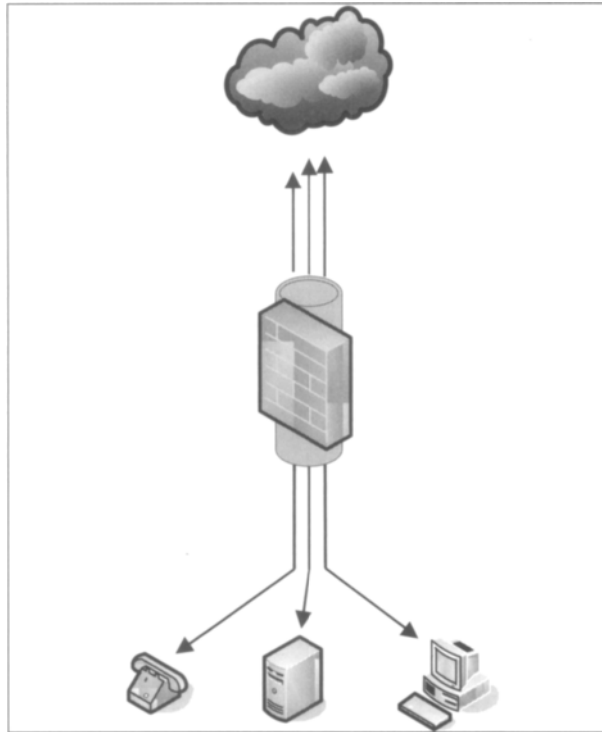
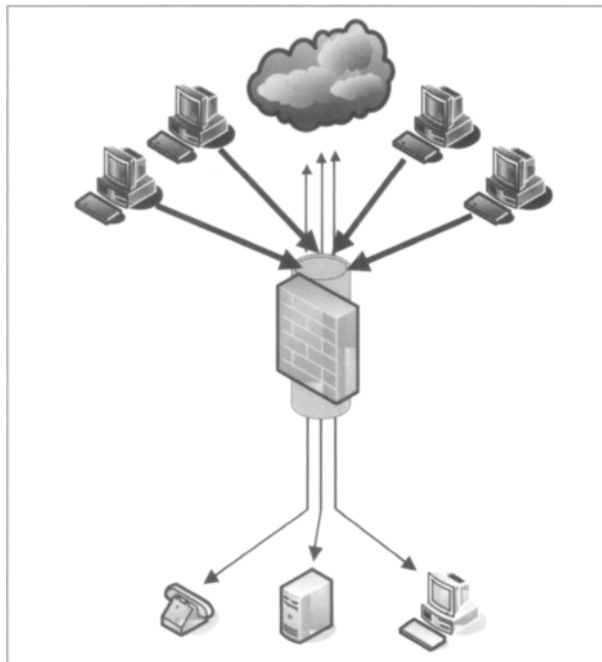
**Table 5.1** VoIP Vulnerabilities

Vulnerability	Description
IP infrastructure	Vulnerabilities on related non-VoIP systems can lead to compromise of VoIP infrastructure.
Underlying operating system	VoIP devices inherit the same vulnerabilities as the operating system or firmware they run on. Operating systems are Windows and Linux.
Configuration	In their default configuration most VoIP devices ship with a surfeit of open services. The default services running on the open ports may be vulnerable to DoS attacks, buffer overflows, or authentication bypass.
Application level	Immature technologies can be attacked to disrupt or manipulate service. Legacy applications (DNS, for example) have known problems.

## Denial-of-Service or VoIP Service Disruption

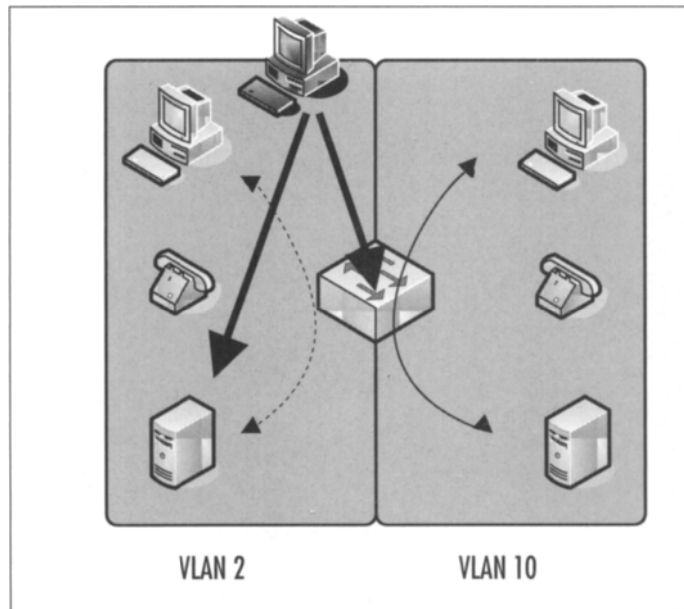
Denial-of-service (DoS) attacks can affect any IP-based network service. The impact of a DoS attack can range from mild service degradation to complete loss of service. There are several classes of DoS attacks. One type of attack in which packets can simply be flooded into or at the target network from multiple external sources is called a distributed denial-of-service (DDoS) attack (see Figures 5.1 and 5.2).

In this figure, traffic flows normally between internal and external hosts and servers. In Figure 5.2, a network of computers (e.g., a botnet) directs IP traffic at the interface of the firewall.

**Figure 5.1** Typical Internet Access**Figure 5.2** A Distributed Denial-of-Service Attack

The second large class of Denial of Service (DoS) conditions occurs when devices within the internal network are targeted by a flood of packets so that they fail—taking out related parts of the infrastructure with them. As in the DDoS scenarios described earlier in this chapter, service disruption occurs to resource depletion—primarily bandwidth and CPU resource starvation (see Figure 5.3). For example, some IP telephones will stop working if they receive a UDP packet larger than 65534 bytes on port 5060.

**Figure 5.3** An Internal Denial-of-Service Attack



Neither integrity checks nor encryption can prevent these attacks. DoS or DDoS attacks are characterized simply by the volume of packets sent toward the victim computer; whether those packets are signed by a server, contain real or spoofed source IP addresses, or are encrypted with a fictitious key—none of these are relevant to the attack.

DoS attacks are difficult to defend against, and because VoIP is just another IP network service, it is just as susceptible to DoS attack as any other IP network services. Additionally, DoS attacks are particularly effective against services such as VoIP and other real-time services, because these services are most sensitive to adverse network status. Viruses and worms are included in this category as they often cause DoS or DDoS due to the increased network traffic that they generate as part of their efforts to replicate and propagate.

How do we defend against these DoS conditions (we won't use the term attack here because some DoS conditions are simply the unintended result of other unrelated actions)? Let's begin with internal DoS. Note in Figure 5.3 that VLAN 10 on the right is not affected by the service disruption on the left in VLAN 2. This illustrates one critical weapon the security administrator has in thwarting DoS conditions—logical segregation of network domains in separate compartments. Each compartment can be configured to be relatively immune to the results of DoS in the others. This is described in more detail in Chapter 8.

Point solutions will also be effective in limiting the consequences of DoS conditions. For example, because strong authentication is seldom used in VoIP environments, the message processing components must trust and process messages from possible attackers. The additional processing of bogus messages exhausts server resources and leads to a DoS. SIP or H.323 Registration Flooding is an example of this, described in the list of DoS threats, later. In that case, message processing servers can mitigate this specific threat by limiting the number of registrations it will accept per minute for a particular address (and/or from a specific IP address). An intrusion prevention system (IPS) may be useful in fending off certain types of DoS attacks. These devices sit on the datapath and monitor passing traffic. When anomalous traffic is detected (either by matching against a database of attack signatures or by matching the results of an anomaly-detection algorithm) the IPS blocks the suspicious traffic. One problem I have seen with these devices—particularly in environments with high availability requirements—is that they sometimes block normal traffic, thus creating their own type of DoS.

Additionally, security administrators can minimize the chances of DoS by ensuring that IP telephones and servers are updated to the latest stable version and release. Typically, when a DoS warning is announced by bugtraq, the vendor quickly responds by fixing the offending software.

## NOTE

VoIP endpoints can be infected with new VoIP device or protocol-specific viruses. WinCE, PalmOS, SymbianOS, and POSIX-based softphones are especially vulnerable because they typically do not run antivirus software and have less robust operating systems. Several Symbian worms already have been detected in the wild. Infected VoIP devices then create a new “weak link” vector for attacking other network resources.

Compromised devices can be used to launch attacks against other systems in the same network, particularly if the compromised device is trusted (i.e., inside the firewall). Malicious programs installed by an attacker on compromised devices can capture user input, capture traffic, and relay user data over a “back channel” to the attacker. This is especially worrisome for softphone users.

VoIP systems must meet stringent service availability requirements. Following are some example DoS threats can cause the VoIP service to be partially or entirely unavailable by preventing successful call placement (including emergency/911), disconnecting existing calls, or preventing use of related services like voicemail. Note that this list is not exhaustive but illustrates some attack scenarios.

- **TLS Connection Reset** It's not hard to force a connection reset on a TLS connection (often used for signaling security between phones and gateways)—just send the right kind of junk packet and the TLS connection will be reset, interrupting the signaling channel between the phone and call server.
- **VoIP Packet Replay Attack** Capture and resend out-of-sequence VoIP packets (e.g., RTP SSRC—SSRC is an RTP header field that stands for Synchronization Source) to endpoints, adding delay to call in progress and degrading call quality.
- **Data Tunneling** Not exactly an attack; rather tunneling data through voice calls creates, essentially, a new form of unauthorized modem. By transporting modem signals through a packet network by using pulse code modulation (PCM) encoded packets or by residing within header information, VoIP can be used to support a modem call over an IP network. This technique may be used to bypass or undermine a desktop modem policy and hide the existence of unauthorized data connections. This is similar in concept to the so-called “IP over HTTP” threat (i.e., “Firewall Enhancement Protocol” RFC 3093)—a classic problem for any ports opened on a firewall from internal sources.
- **QoS Modification Attack** Modify non-VoIP-specific protocol control information fields in VoIP data packets to and from endpoints to degrade or deny voice service. For example, if an attacker were to change 802.1Q VLAN tag or IP packet ToS bits, either as a man-in-the-middle or by compromising endpoint device configuration, the attacker could disrupt the quality of service “engineered” for a VoIP network. By subordinating voice traffic to data traffic, for example, the attacker might substantially delay delivery of voice packets.
- **VoIP Packet Injection** Send forged VoIP packets to endpoints, injecting speech or noise or gaps into active call. For example, when RTP is used without authentication of RTCP packets (and without SSRC sampling), an attacker can inject RTCP packets into a multicast group, each with a different SSRC, which can grow the group size exponentially.
- **DoS against Supplementary Services** Initiate a DoS attack against other network services upon which the VoIP service depends (e.g., DHCP, DNS, BOOTP). For example, in networks where VoIP endpoints rely on DHCP-assigned addresses, disabling the DHCP server prevents endpoints (soft- and hardphones) from

acquiring addressing and routing information they need to make use of the VoIP service.

- **Control Packet Flood** Flood VoIP servers or endpoints with unauthenticated call control packets, (e.g., H.323 GRQ, RRQ, URQ packets sent to UDP/1719). The attacker's intent is to deplete/exhaust device, system, or network resources to the extent that VoIP service is unusable. Any open administrative and maintenance port on call processing and VoIP-related servers can be a target for this DoS attack.
- **Wireless DoS** Initiate a DoS attack against wireless VoIP endpoints by sending 802.11 or 802.1X frames that cause network disconnection (e.g., 802.11 Deauthenticate flood, 802.1X EAP-Failure, WPA MIC attack, radio spectrum jamming). For example, a Message Integrity Code attack exploits a standard countermeasure whereby a wireless access point disassociates stations when it receives two invalid frames within 60 seconds, causing loss of network connectivity for 60 seconds. In a VoIP environment, a 60-second service interruption is rather extreme.
- **Bogus Message DoS** Send VoIP servers or endpoints valid-but-forged VoIP protocol packets to cause call disconnection or busy condition (e.g., RTP SSRC collision, forged RTCP BYE, forged CCMS, spoofed endpoint button push). Such attacks cause the phone to process a bogus message and incorrectly terminate a call, or mislead a calling party into believing the called party's line is busy.
- **Invalid Packet DoS** Send VoIP servers or endpoints invalid packets that exploit device OS and TCP/IP implementation denial-of-service CVEs. For example, the exploit described in CAN-2002-0880 crashes Cisco IP phones using jolt, jolt2, and other common fragmentation-based DoS attack methods. CAN-2002-0835 crashes certain VoIP phones by exploiting DHCP DoS CVEs. Avaya IP phones may be vulnerable to port zero attacks.
- **Immature Software DoS** PDA/handheld softphones and first generation VoIP hardphones are especially vulnerable because they are not as mature or intensely scrutinized. VoIP call servers and IP PBXs also run on OS platforms with many known CVEs. Any open administrative/maintenance port (e.g., HTTP, SNMP, Telnet) or vulnerable interface (e.g., XML, Java) can become an attack vector.
- **VoIP Protocol Implementation DoS** Send VoIP servers or endpoints invalid packets to exploit a VoIP protocol implementation vulnerability to a DoS attack. Several such exploits are identified in the MITRE CVE database (<http://cve.mitre.org>). For example, CVE-2001-00546 uses malformed H.323 packets to exploit Windows ISA memory leak and exhaust resources. CAN-2004-0056 uses malformed H.323 packets to exploit Nortel BCM DoS vulnerabilities. Lax software update practices (failure to install CVE patches) exacerbate risk.



- **Packet of Death DoS** Flood VoIP servers or endpoints with random TCP, UDP, or ICMP packets or fragments to exhaust device CPU, bandwidth, TCP sessions, and so on. For example, an attacker can initiate a TCP Out of Band DoS attack by sending a large volume of TCP packets marked “priority delivery” (the TCP Urgent flag). During any flood, increased processing load interferes with the receiving system’s ability to process real traffic, initially delaying voice traffic processing but ultimately disrupting service entirely.
- **IP Phone Flood DoS** Send a very large volume of call data toward a single VoIP endpoint to exhaust that device’s CPU, bandwidth, TCP sessions, and so on. Interactive voice response systems, telephony gateways, conferencing servers, and voicemail systems are able to generate more call data than a single endpoint can handle and so could be leveraged to flood an endpoint.

## Call Hijacking and Interception

Call interception and eavesdropping are other major concerns on VoIP networks. The VOIPSA threat taxonomy ([www.voipsa.org/Activities/taxonomy-wiki.php](http://www.voipsa.org/Activities/taxonomy-wiki.php)) defines eavesdropping as “a method by which an attacker is able to monitor the entire signaling and/or data stream between two or more VoIP endpoints, but cannot or does not alter the data itself.” Successful call interception is akin to wiretapping in that conversations of others can be stolen, recorded, and replayed without their knowledge. Obviously, an attacker who can intercept and store these data can make use of the data in other ways as well.

### Tools & Traps...

#### DNS Poisoning

A DNS A (or address) record is used for storing a domain or hostname mapping to an IP address. SIP makes extensive use of SRV records to locate SIP services such as SIP proxies and registrars. SRV (service) records normally begin with an underscore (`_sip.tcpserver.udp.domain.com`) and consist of information describing service, transport, host, and other information. SRV records allow administrators to use several servers for a single domain, to move services from host to host with little fuss, and to designate some hosts as primary servers for a service and others as backups.

An attacker’s goal, when attempting a DNS Poisoning or spoofing attack, is to replace valid cached DNS A, SRV, or NS records with records that point to the attacker’s server(s). This can be accomplished in a number of fairly trivial ways—the easiest being to initiate a zone transfer from the attacker’s DNS server to the victim’s misconfigured

Continued

DNS server, by asking the victim's DNS server to resolve a networked device within the attacker's domain. The victim's DNS server accepts not only the requested record from the attacker's server, but it also accepts and caches any other records that the attacker's server includes.

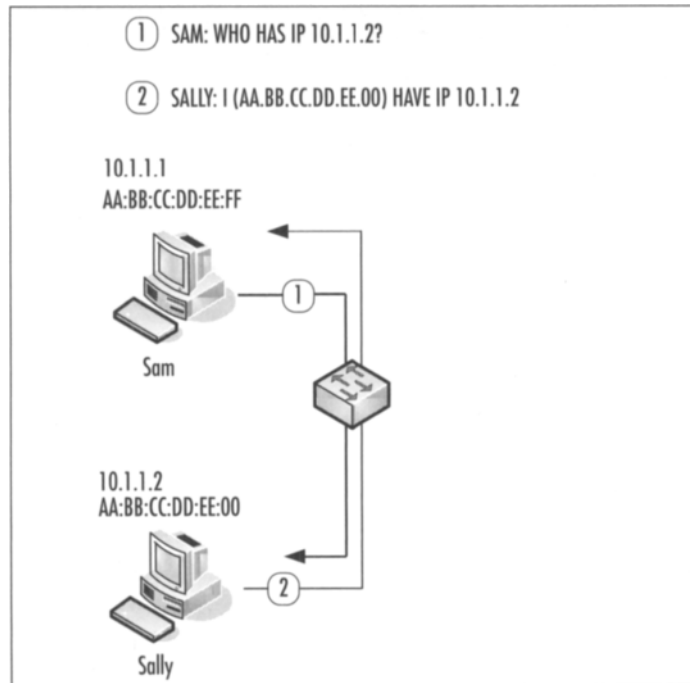
Thus, in addition to the A record for `www.attacker.com`, the victim DNS server may receive a bogus record for `www.yourbank.com`. The innocent victim will then be redirected to the `attacker.com` Web site anytime he or she attempts to browse to the `yourbank.com` Web site, as long as the bogus records are cached. Substitute a SIP URL for a Web site address, and the same scenario can be repeated in a VoIP environment.

This family of threats relies on the absence of cryptographic assurance of a request's originator. Attacks in this category seek to compromise the message integrity of a conversation. This threat demonstrates the need for security services that enable entities to authenticate the originators of requests and to verify that the contents of the message and control streams have not been altered in transit.

In the past several years, as host PCs have improved their processing power and their ability to process networked information, network administrators have instituted a hierarchical access structure that consists of a single, dedicated switched link for each host PC to distribution or backbone devices. Each networked user benefits from a more reliable, secure connection with guaranteed bandwidth. The use of a switched infrastructure limits the effectiveness of packet capture tools or protocol analyzers as a means to collect VoIP traffic streams. Networks that are switched to the desktop allow normal users' computers to monitor only broadcast and unicast traffic that is destined to their particular MAC address. A user's NIC (network interface card) literally does not see unicast traffic destined for other computers on the network.

The address resolution protocol (ARP) is a method used on IPv4 Ethernet networks to map the IP address (layer 3) to the hardware or MAC (Media Access Control) layer 2 address. (Note that ARP has been replaced in IPv6 by Neighbor Discovery [ND] protocol. The ND protocol is a hybrid of ARP and ICMP.) Two classes of hardware addresses exist: the broadcast address of all ones, and a unique 6 byte identifier that is burned into the PROM of every NIC (Network Interface Card).

Figure 5.4 illustrates a typical ARP address resolution scheme. A host PC (10.1.1.1) that wishes to contact another host (10.1.1.2) on the same subnet issues an ARP broadcast packet (ARPs for the host) containing its own hardware and IP addresses. NICs contain filters that allow them to drop all packets not destined for their unique hardware address or the broadcast address, so all NICs but the query target silently discard the ARP broadcast. The target NIC responds to the query request by unicasting its IP and hardware address, completing the physical to logical mapping, and allowing communications to proceed at layer 3.

**Figure 5.4** Typical ARP Request/Reply

To minimize broadcast traffic, many devices cache ARP addresses for a varying amount of time: The default ARP cache timeout for Linux is one minute; for Windows NT, two minutes, and for Cisco routers, four hours. This value can be trivially modified in most systems. The ARP cache is a table structure that contains IP address, hardware address, and oftentimes, the name of the interface the MAC address is discovered on, the type of media, and the type of ARP response. Depending upon the operating system, the ARP cache may or may not contain an entry for its own addresses.

In Figure 5.4, Sam's ARP cache contains one entry prior to the ARP request/response:

Internet Address	Physical Address	
10.1.1.1	AA:BB:CC:DD:EE:FF	int0

After the ARP request/response completes, Sam's ARP cache now contains two entries:

Internet Address	Physical Address	
10.1.1.1	AA:BB:CC:DD:EE:FF	int0
10.1.1.2	AA:BB:CC:DD:EE:00	int0

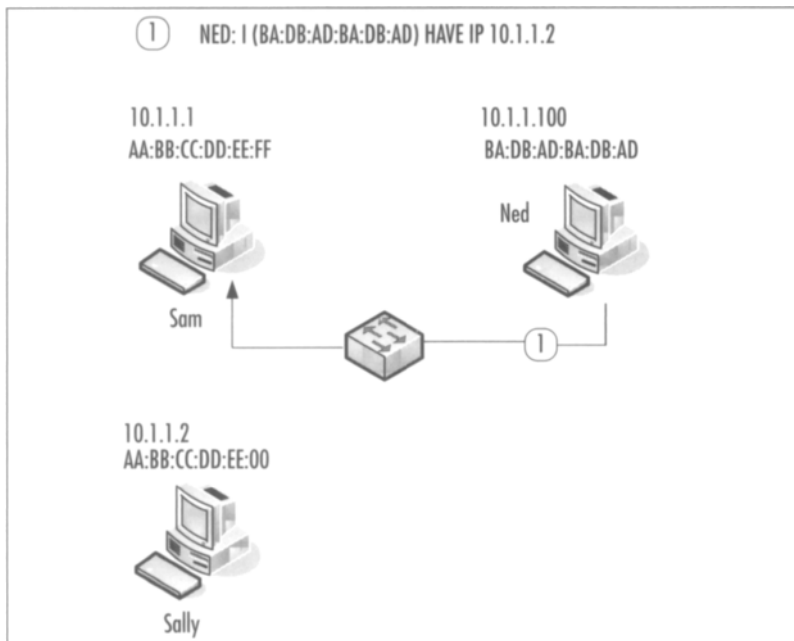
Note that Sally's ARP cache, as a result of the request/response communications, is updated with the hardware:IP mappings for both workstations as well.

# ARP Spoofing

ARP is a fundamental Ethernet protocol. Perhaps for this reason, manipulation of ARP packets is a potent and frequent attack mechanism on VoIP networks. Most network administrators assume that deploying a fully switched network to the desktop prevents the ability of network users to sniff network traffic and potentially capture sensitive information traversing the network. Unfortunately, several techniques and tools exist that allow any user to sniff traffic on a switched network because ARP has no provision for authenticating queries or query replies. Additionally, because ARP is a stateless protocol, most operating systems (Solaris is an exception) update their cache when receiving ARP reply, regardless of whether they have sent out an actual request.

Among these techniques, ARP redirection, ARP spoofing, ARP hijacking, and ARP cache poisoning are related methods for disrupting the normal ARP process. These terms frequently are interchanged and confused. For the purpose of this section, we'll refer to ARP cache poisoning and ARP spoofing as the same process. Using freely available tools such as ettercap, Cain, and dsniff, an evil IP device can spoof a normal IP device by sending unsolicited ARP replies to a target host. The bogus ARP reply contains the hardware address of the normal device and the IP address of the malicious device. This "poisons" the host's ARP cache (see Figure 5.5).

**Figure 5.5** ARP Spoofing (Cache Poisoning)



In Figure 5.5, Ned is the attacking computer. When SAM broadcasts an ARP query for Sally's IP address, Ned, the attacker, responds to the query stating that the IP address

(10.1.1.2) belongs to Ned's MAC address, BA:DB:AD:BA:DB:AD. Packets sent from Sam supposedly to Sally will be sent to Ned instead. Sam will mistakenly assume that Ned's MAC address corresponds to Sally's IP address and will direct all traffic destined for that IP address to Ned's MAC. In fact, Ned can poison Sam's ARP cache without waiting for an ARP query since on Windows systems (9x/NT/2K), static ARP entries are overwritten whenever a query response is received regardless of whether or not a query was issued.

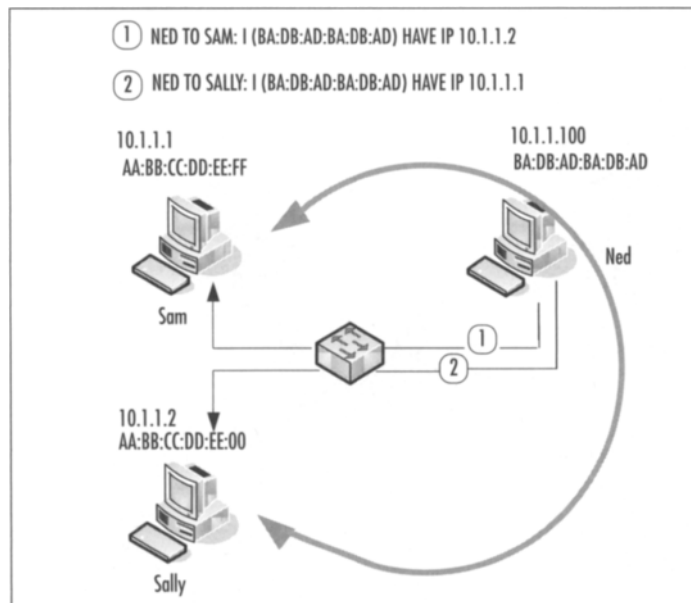
Sam's ARP cache now looks like this:

Internet Address	Physical Address	
10.1.1.1	AA:BB:CC:DD:EE:FF	int0
10.1.1.2	BA:DB:AD:BA:DB:AD	int0

This entry will remain until it ages out or a new entry replaces it.

ARP redirection can work bidirectionally, and a spoofing device can insert itself in the middle of a conversation between two IP devices on a switched network (see Figure 5.6). This is probably the most insidious ARP-related attack. By routing packets on to the devices that should truly be receiving the packets, this insertion (known as a Man/Monkey/Moron in the Middle attack) can remain undetected for some time. An attacker can route packets to /dev/null (nowhere) as well, resulting in a DoS attack.

**Figure 5.6** An ARP MITM Attack



Sam's ARP cache:

Internet Address	Physical Address	
10.1.1.1	AA:BB:CC:DD:EE:FF	int0
10.1.1.2	BA:DB:AD:BA:DB:AD	int0

Sally's ARP cache:

Internet Address	Physical Address	
10.1.1.1	BA:DB:AD:BA:DB:AD	int0
10.1.1.2	AA:BB:CC:DD:EE:00	int0

As all IP traffic between the true sender and receiver now passes through the attacker's device, it is trivial for the attacker to sniff that traffic using freely available tools such as Ethereal or tcpdump. Any unencrypted information (including e-mails, usernames and passwords, and web traffic) can be intercepted and viewed.

This interception has potentially drastic implications for VoIP traffic. Freely available tools such as vomit and rtpsniff, as well as private tools such as VoipCrack, allow for the interception and decoding of VoIP traffic. Captured content can include speech, signaling and billing information, multimedia, and PIN numbers. Voice conversations traversing the internal IP network can be intercepted and recorded using this technique.

There are a number of variations of the aforementioned techniques. Instead of imitating a host, the attacker can emulate a gateway. This enables the attacker to intercept numerous packet streams. However, most ARP redirection techniques rely on stealth. The attacker in these scenarios hopes to remain undetected by the users being impersonated. Posing as a gateway may result in alerting users to the attacker's presence due to unanticipated glitches in the network, because frequently switches behave in unexpected ways when attackers manipulate ARP processes. One unintended (much of the time) consequence of these attacks, particularly when switches are heavily loaded, is that the switch CAM (Content-Addressable Memory) table—a finite-sized IP address to MAC address lookup table—becomes disrupted. This leads to the switch forwarding unicast packets out many ports in unpredictable fashion. Penetration testers may want to keep this in mind when using these techniques on production networks.

In order to limit damage due to ARP manipulation, administrators should implement software tools that monitor MAC to IP address mappings. The freeware tool, Arpwatch, monitors these pairings. At the network level, MAC/IP address mappings can be statically coded on the switch; however, this is often administratively untenable. Dynamic ARP Inspection (DAI) is available on newer Cisco Catalyst 6500 switches. DAI is part of Cisco's Integrated Security (CIS) functionality and is designed to prevent several layer two and layer

three spoofing attacks, including ARP redirection attacks. Note that DAI and CIS are available only on Catalyst switches using native mode (Cisco IOS).

The potential risks of decoding intercepted VoIP traffic can be eliminated by implementing encryption. Avaya's Media Encryption feature is an example of this. Using Media Encryption, VoIP conversations between two IP endpoints are encrypted using AES encryption. In highly secure environments, organizations should ensure that Media Encryption is enabled on all IP codec sets in use.

DAI enforces authorized MAC-to-IP address mappings. Media Encryption renders traffic, even if intercepted, unintelligible to an attacker.

The following are some additional examples of call or signal interception and hijacking. This class of threats, though typically more difficult to accomplish than DoS, can result in significant loss or alteration of data. DoS attacks, whether caused by active methods or inadvertently, although important in terms of quality of service, are more often than not irritating to users and administrators. Interception and hijacking attacks, on the other hand, are almost always active attacks with theft of service, information, or money as the goal. Note that this list is not exhaustive but illustrates some attack scenarios.

- **Rogue VoIP Endpoint Attack** Rogue IP endpoint contacts VoIP server by leveraging stolen or guessed identities, credentials, and network access. For example, a rogue endpoint can use an unprotected wall jack and auto-registration of VOIP phones to get onto the network. RAS password guessing can be used to masquerade as a legitimate endpoint. Lax account maintenance (expired user accounts left active) increases risk of exploitation.
- **Registration Hijacking** Registration hijacking occurs when an attacker impersonates a valid UA to a registrar and replaces the registration with its own address. This attack causes all incoming calls to be sent to the attacker.
- **Proxy Impersonation** Proxy impersonation occurs when an attacker tricks a SIP UA or proxy into communicating with a rogue proxy. If an attacker successfully impersonates a proxy, he or she has access to all SIP messages.
- **Toll Fraud** Rogue or legitimate VoIP endpoint uses a VoIP server to place unauthorized toll calls over the PSTN. For example, inadequate access controls can let rogue devices place toll calls by sending VoIP requests to call processing applications. VoIP servers can be hacked into in order to make free calls to outside destinations. Social engineering can be used to obtain outside line prefixes.
- **Message Tampering** Capture, modify, and relay unauthenticated VoIP packets to/from endpoints. For example, a rogue 802.11 AP can exchange frames sent or received by wireless endpoints if no payload integrity check (e.g., WPA MIC, SRTP) is used. Alternatively, these attacks can occur through registration hijacking, proxy impersonation, or an attack on any component trusted to process SIP or

H.323 messages, such as the proxy, registration servers, media gateways, or firewalls. These represent non-ARP-based MITM attacks.

- **VoIP Protocol Implementation Attacks** Send VoIP servers or endpoints invalid packets to exploit VoIP protocol implementation CVEs. Such attacks can lead to escalation of privileges, installation and operation of malicious programs, and system compromise. For example, CAN-2004-0054 exploits Cisco IOS H.323 implementation CVEs to execute arbitrary code. CSCed33037 uses unsecured IBM Director agent ports to gain administrative control over IBM servers running Cisco VoIP products.

## Notes from the Underground...

### ANI/Caller-ID Spoofing

Caller ID is a service provided by most telephone companies (for a monthly cost) that will tell you the name and number of an incoming call. Automatic Number Identification (ANI) is a system used by the telephone company to determine the number of the calling party. To spoof Caller-ID, an attacker sends modem tones over a POTS lines between rings 1 and 2. ANI spoofing is setting the ANI so as to send incorrect ANI information to the PSTN so that the resulting Caller-ID is misleading. Traditionally this has been a complicated process either requiring the assistance of a cooperative phone company operator or an expensive company PBX system.

In ANI/Caller-ID spoofing, an evildoer hijacks phone number and the identity of a trusted party, such as a bank or a government office. The identity appears on the caller ID box of an unsuspecting victim, with the caller hoping to co-opt valuable information, such as account numbers, or otherwise engage in malicious mischief. This is not a VoIP issue, per se. In fact, one of the big drawbacks about VoIP trunks is their inability to send ANI properly because of incomplete standards.

## H.323-Specific Attacks

The only existing vulnerabilities that we are aware of at this time take advantage of ASN.1 parsing defects in the first phase of H.225 data exchange. More vulnerabilities can be expected for several reasons: the large number of differing vendor implementations, the complex nature of this collection of protocols, problems with the various implementations of ASN.1/PER encoding/decoding, and the fact that these protocols—alone and in concert—have not endured the same level of scrutiny that other more common protocols have been subjected to. For example, we have unpublished data that shows that flooding a



gateway or media server with GRQ request packets (RAS registration request packets) results in a DoS against certain vendor gateway implementations—basically the phones deregister.

## SIP-Specific Attacks

Multiple vendors have confirmed vulnerabilities in their respective SIP (Session Initiation Protocol) implementations. The vulnerabilities have been identified in the INVITE message used by two SIP endpoints during the initial call setup. The impact of successful exploitation of the vulnerabilities has not been disclosed but potentially could result in a compromise of a vulnerable device. (CERT: CA-2003-06.) In addition, many recent examples of SIP Denial of Service attacks have been reported.

Recent issues that affect Cisco SIP Proxy Server (SPS) [Bug ID CSCec31901] demonstrate the problems SIP implementers may experience due to the highly modular architecture of this protocol. The SSL implementation in SPS (used to secure SIP sessions) is vulnerable to an ASN.1 BER decoding error similar to the one described for H.323 and other protocols. This example illustrates a general concern with SIP: As the SIP protocol links existing protocols and services together, all the classic vulnerabilities in services such as SSL, HTTP, and SMTP may resurface in the VoIP environment.

## Summary

DoS attacks, whether they are intentional or unintended, are the most difficult VoIP-related threat to defend against. The packet switching nature of data networks allows multiple connections to share the same transport medium. Therefore, unlike telephones in circuit-switched networks, an IP terminal endpoint can receive and potentially participate in multiple calls at once. Thus, an endpoint can be used to amplify attacks. On VoIP networks, resources such as bandwidth must be allocated efficiently and fairly to accommodate the maximum number of callers. This property can be violated by attackers who aggressively and abusively obtain an unnecessarily large amount of resources. Alternatively, the attacker simply can flood the network with large number of packets so that resources are unavailable to all other callers.

In addition, viruses and worms create DoS conditions due to the network traffic generated by these agents as they replicate and seek out other hosts to infect. These agents are proven to wreak havoc with even relatively well-secured data networks. VoIP networks, by their nature, are exquisitely sensitive to these types of attacks. Remedies for DoS include logical network partitioning at layers 2 and 3, stateful firewalls with application inspection capabilities, policy enforcement to limit flooded packets, and out-of-band management. Out-of-band management is required so that in the event of a DoS event, system administrators are still able to monitor the network and respond to additional events.

Theft of services and information is also problematic on VoIP networks. These threats are almost always due to active attack. Many of these attacks can be thwarted by implementing additional security controls at layer 2. This includes layer 2 security features such as DHCP Snooping, Dynamic ARP Inspection, IP Source Guard, Port Security, and VLAN ACLs. The fundamental basis for this class of attacks is that the identity of one or more of the devices that participate is not legitimate.

Endpoints must be authenticated, and end users must be validated in order to ensure legitimacy. Hijacking and call interception revolves around the concept of fooling and manipulating weak or nonexistent authentication measures. We are all familiar with different forms of authentication, from the password used to login to your computer to the key that unlocks the front door. The conceptual framework for authentication is made up of three factors: “something you have” (a key or token), “something you know” (a password or secret handshake), or “something you are” (fingerprint or iris pattern). Authentication mechanisms validate users by one or a combination of these. Any type of unauthenticated access, particularly to key infrastructure components such as the IP PBX or DNS server, for example, can result in disagreeable consequences for both users and administrators.

VoIP relies upon a number of ancillary services as part of the configuration process, as a means to locate users, manage servers and phones, and to ensure favorable transport, among others. DNS, DHCP, HTTP, HTTPS, SNMP, SSH, RSVP, and TFTP services all have been the subject of successful exploitation by attackers. Potential VoIP users may defer transi-

tioning to IP Telephony if they believe it will reduce overall network security by creating new vulnerabilities that could be used to compromise non-VoIP systems and services within the same network. Effective mitigation of these threats to common data networks and services could be considered a security baseline upon which a successful VoIP deployment depends. Firewalls, network and system intrusion detection, authentication systems, anti-virus scanners, and other security controls, which should already be in place, are required to counter attacks that might debilitate any or all IP-based services (including VoIP services).

H.323 and SIP suffer security vulnerabilities based simply upon their encoding schemes, albeit for different reasons. Because SIP is an unstructured text-based protocol, it is impossible to test all permutations of SIP messages during development for security vulnerabilities. It's fairly straightforward to construct a malformed SIP message or message sequence that results in a DoS for a particular SIP device. This may not be significant for a single UA endpoint, but if this "packet of death" can render all the carrier-class media gateway controllers in a network useless, then this becomes a significant problem. H.323 on the other hand is encoded according to ASN.1 PER encoding rules. The implementation of H.323 message parsers, rather than the encoding rules themselves, results in security vulnerabilities in the H.323 suite.

## Confirm User Identity

### Solutions in this chapter:

- 802.1x and 802.11i
- Public Key Infrastructure
- Minor Authentication Methods

# Introduction

Authentication is a measure of trust. The point of this chapter is to illustrate trust complexities and to cover authentication of both user identity and device identity. These two identities are not equal. Authentication in the networking world, in general, is based either on using a shared secret (you are authenticated if you know the secret) or on public key-based methods with certificates (you prove your identity by possessing the correct private key).

Authentication establishes the identities of devices and users to a degree that is in accord with your security policies. Authorization, on the other hand, establishes the amount and type of network and application resources authorized individuals and devices are able to access.

Device authentication can be automated and made transparent to the user based upon assigning and verifying a unique profile for the device. This profile may include attributes such as model, serial number, MAC address, IP address, physical location, time-of-day, and so on, and may include a shared secret or a certificate. Device authentication literally blocks rogue endpoints from accessing any network resources. In a VoIP environment, this prevents malicious endpoints from placing unauthorized calls or causing other mischief. Some of the 802.1x and 802.11i standards described later in this chapter can be used as part of an automated device authentication process.

Everyone who has logged on to a computer is familiar with user authentication. Users identify themselves to an authenticator by presenting credentials. The most common of these is a username/password combination, although user authentication can also be accomplished using other means including biometric or token-based methods. Common network-based authentication methods include Windows domain authentication, NIS+, and Kerberos. Windows 2000 and later platforms offer two default authentication mechanisms: MS Kerberos and NTLM. Most users believe that logging on to an account in a Windows domain gives them access to the network. That is not true. When the Kerberos protocol (the default) is used for network authentication, the user's first access is to the domain's authentication service, which ultimately provides access to network resources.

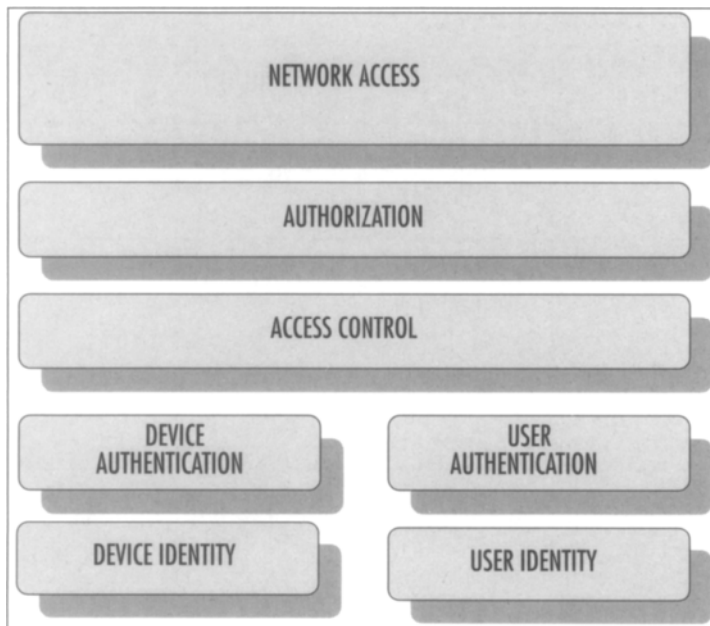
In order to secure VoIP networks, the identity of both the user and the device must be verified. This can be accomplished in a number of ways. Network-based authentication methods such as those mentioned earlier in this chapter often are used, and in many environments, this user authentication is considered sufficient for virtually unrestricted access to network resources. However, as we argue in Chapter 1, network boundaries are disappearing, network users are increasingly mobile, more types and quantities of devices are registering with the network, and devices no longer even require a physical link to access network resources. The addition of VoIP resources to the existing infrastructure only adds to this complexity. The aforementioned mechanisms are not sufficient to cope with these new sophisticated technologies.

Some simple fixes are available. User identity can be confirmed using a method as simple as HTTP Digest authentication, and devices can simply be filtered by MAC address lists.

These point solutions have their drawbacks. Both can be circumvented by attackers with minimal skills, and neither scale well. In order to confirm user and device identity on enterprise VoIP networks, system administrators will ultimately turn to 802.1x/EAP, a certificate infrastructure, or a combination of these. The remainder of this chapter discusses these two technologies.

Figure 6.1 shows the generic components involved in a model authentication scheme. The static beginning and end states are the device and user identities, and internal network access, respectively. The processes are access control and authorization. Much of this chapter is devoted to exploring these mechanisms.

**Figure 6.1** General Authentication—Authorization Framework



In H.323 environments the basis for authentication (trust) is defined by the endpoints of the communications channel. For a connection establishment channel, this may be between the caller (such as a gateway or IP telephone endpoint) and a hosting network component (a gateway or gatekeeper). For example, a telephone “trusts” that the gatekeeper will connect it with the telephone whose number has been dialed. The result of trusting an element is the confidence to reveal the privacy mechanism (algorithm and key) to that element. Given the aforementioned information, all participants in the communications path should authenticate any and all trusted elements. This is described in more detail in Chapter 3.

The SIP draft does not explicitly define authentication mechanisms. In contrast, SIP developers chose a modular approach—reusing the same headers, error codes, and encoding rules as HTTP. From RFC 3261:

The fundamental security services required for the SIP protocol are: preserving the confidentiality and integrity of messaging, preventing replay attacks or message spoofing, providing for the authentication and privacy of the participants in a session, and preventing denial-of-service attacks. Bodies within SIP messages separately require the security services of confidentiality, integrity, and authentication. Rather than defining new security mechanisms specific to SIP, SIP reuses wherever possible existing security models derived from the HTTP and SMTP space.

SIP defines a set of security mechanisms that can be used by any SIP client or server to share authentication data (see Table 6.1).

**Table 6.1** SIP Security Mechanisms

AUTHENTICATION METHOD	AUTHENTICATION TYPE	CONFIDENTIALITY	INTEGRITY
S/MIME	PUBLIC KEY INFRASTRUCTURE	YES	YES
TLS	PUBLIC KEY INFRASTRUCTURE	YES	YES
IPSEC	PUBLIC KEY INFRASTRUCTURE & PSK	YES	YES
HTTP 1.1 DIGEST	PRE-SHARED KEY	NO	NO

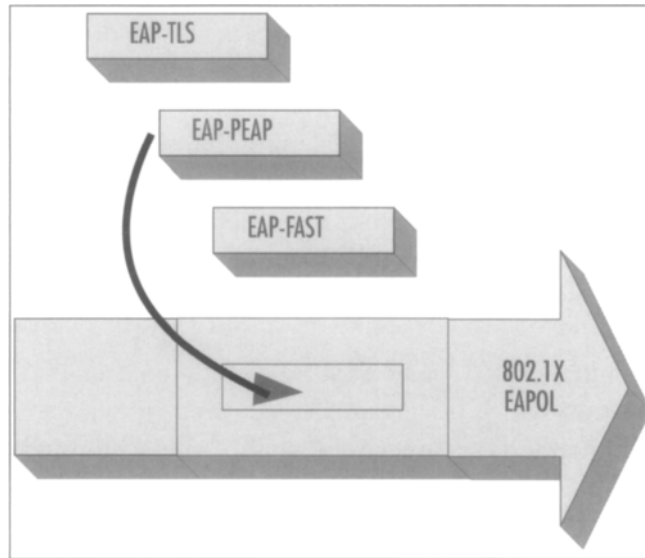
Since SIP's syntax is based on HTTP, it reuses HTTP Digest Authentication to authenticate endpoints. S/MIME, TLS, and IPSec can also be used to protect components of the SIP infrastructure. SIP can use TLS for signaling security between routing elements (hop by hop), as well as S/MIME for security of signaling end to end. TLS security is visible to users and other elements by using the "sips:" URI scheme, similar to "https:".

The threats in this category rely on the absence of cryptographic assurance of a request's originator. Attacks in this category seek to compromise the message integrity of a conversation and interfere with nonrepudiation. Oftentimes the goal of these attacks is economic or data theft. These threats demonstrate the need for security services that enable entities to authenticate the originators of requests and to verify that the contents of the message and control streams have not been altered in transit.

## 802.1x and 802.11i (WPA2)

The 802.1x protocol defines port-based, network access control that is used to provide authenticated network access (see Figure 6.2). Although this standard is designed for wired Ethernet networks, it has been adapted for use on 802.11 WLANs. It is simply a standard for passing EAP over a wired or wireless LAN.

**Figure 6.2** EAPOL



802.1x restricts unauthorized clients from connecting to a LAN. The client must first authenticate with an Authentication server, typically a RADIUS server, before the switch port is made available and the network can be accessed. EAP (Extensible Authentication Protocol) is a general authentication protocol that provides a framework for multiple authentication methods, including traditional passwords, token cards, Kerberos, Digital Certificates, and public-key authentication.

WEP (Wireless Equivalent Privacy) has famously been shown to be insecure (Anton Rager's wepcrack was the first publicly available tool for this—<http://wepcrack.sourceforge.net/>); however WEP protection of wireless connections is still better than no encryption at all. The Wi-Fi Alliance (a consortium of major vendors—<http://wi-fi.org/>) is responsible for drafting both the WPA (Wi-Fi Protected Access) and WPA2 standards. The Wi-Fi alliance also formed a VoWLAN (Voice over Wireless LAN) working group tasked with developing WMM (Wi-Fi Multimedia) QoS standards for VoIP and other multimedia over wireless networks.

WPA implements a subset of IEEE802.11i, and differs from WEP mainly in that it utilizes TKIP (Temporal Key Integrity protocol) and the EAP framework for authentication. 802.11i is a draft IEEE standard for 802.11 wireless network security. 802.11i, also known as



WPA2, uses 802.1x as the authentication mechanism and the Advanced Encryption Standard (AES) block cipher for encryption. WEP and WPA use the RC4 stream cipher. Table 6.2 shows some of the key features of these three security standards.

**Table 6.2** Security Standard Features

Protocol	Authentication	Cipher	Key Length	Key Management
WEP	None	RC-4	40/104	None
WPA	802.1x/EAP	RC-4	128	802.1x/EAP
WPA2	802.1x/EAP	AES	128	802.1x/EAP

It is helpful to think of 802.1x not as a single protocol but rather as a security framework using existing, and proven security standards that serves two critical security functions—authentication (PSK or PKI, for example) and encryption (TLS or AES, for example). Note that 802.1x does not define either authentication or encryption methods (in fact 802.1x can be used without encryption); rather these are defined largely through this choice of an EAP type.

Until the client is authenticated via 802.1x/EAP access control, the only protocol allowed through the port to which the client is connected is Extensible Authentication Protocol traffic. After authentication is successful, traffic can pass through the port.

## 802.1x/EAP Authentication

Now we'll define the terms associated with 802.1x/EAP authentication.

### Supplicant (Peer)

This is the other end of the point-to-point link; the end that is being authenticated by the authenticator. Generally this term refers to the client in an EAP exchange.

### Authenticator

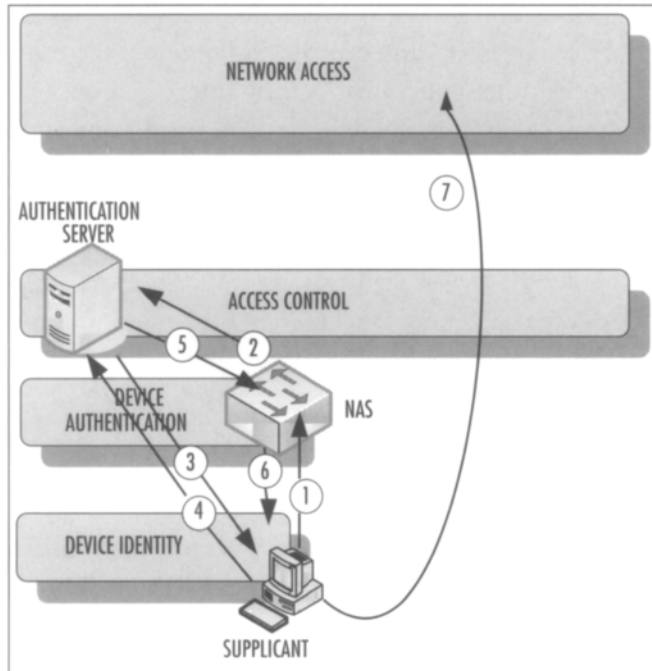
Authenticator is a wireless access point (AP) or switch (NAS—Network Access Server). The authenticator maintains the network (WLAN or LAN) in closed state to all unauthenticated traffic. It does not do authentication directly, but instead tunnels the extensible authentication protocol (EAP) to an authentication server.

### Authentication Server

The authentication server performs the actual client authentication and instructs the authenticator to allow or reject the supplicant's traffic. The authentication server is typically a RADIUS server.

Figure 6.3 illustrates the basic message flow in an 802.1x/EAP authentication scenario. This is an example of the most common 802.1x/EAP model—a Full/Pass-Through state machine, which allows an NAS (network access server) or edge device to pass EAP Response messages to an Authentication Server where the authentication method resides. The NAS does not have to understand the request type and must be able to simply act as a passthrough agent for a back-end server. The NAS need look only for the success/failure code from the Authentication Server to terminate the authentication phase.

**Figure 6.3** Generic EAP Authentication



## Tools & Traps...

### AAA, RADIUS, and DIAMETER

RADIUS (Remote Authentication Dial In User Service) is an AAA (authentication, authorization, and accounting) protocol for applications such as network access or IP mobility. AAA is a term for a framework that allows methods to intelligently control access to computer resources, enforce policies, audit usage, and provide information necessary to bill for services. Because AAA services often are used to authenticate remote system administrators, availability is critical, and should always be provided by

Continued

at least a pair of physically separated, dedicated AAA servers that serve as master and backup.

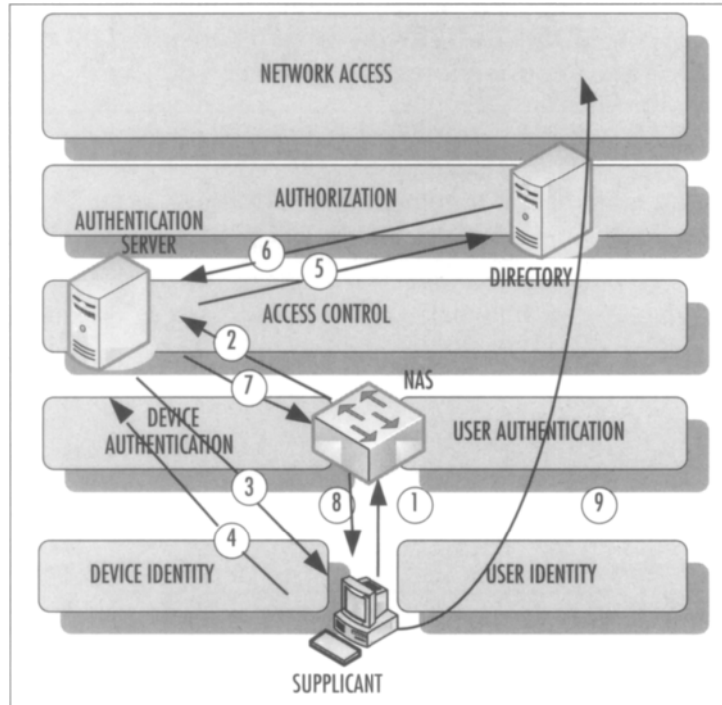
DIAMETER is a new extended AAA protocol that is designed to replace RADIUS. DIAMETER (a play on words, since diameter is twice the radius of a circle) is designed to enhance RADIUS functions and to fix several security problems (such as unencrypted CHAP response) that have plagued RADIUS in recent years.

In step 1, the supplicant (a workstation, wireless access point, IP phone, etc.) sends one or more requests to the NAS petitioning for access to the network. The NAS (step 2) passes the EAP message to the Authentication Server, which is almost always a RADIUS server. In step 3, the Authentication Server requests the credentials of the supplicant and specifies the type of credentials required to confirm the supplicant's identity. (Note here that the arrows between the RADIUS server and the client indicate logical, not physical, connectivity. All traffic between the two passes through the NAS.) The Authentication Server makes its decision to grant or deny access based upon Native RADIUS credentials. In step 4, the supplicant sends its credentials to the RADIUS server. Upon validating the supplicant's credentials, the Authentication Server transmits a success/failure message to the NAS (step 5). In step 6, if access is granted, the NAS opens the port to all traffic (as opposed to just EAPOL traffic) and data exchange between the authenticated LAN device and the LAN is allowed. If access is granted, then (step 7) the supplicant is able to access network resources.

You will notice that after access is approved, the supplicant has unrestricted access to network resources. Only the device identity has been authenticated. No authorization has been performed, nor has the user of the device been authenticated.

Figure 6.4 illustrates a more typical generic 802.1x transaction. The first several steps in this scenario are similar to the scenario we just described. In step 1, the supplicant (a workstation, wireless access point, IP phone, etc) sends one or more requests to the NAS petitioning for access to the network. The NAS (Step 2) passes the EAP message to the Authentication Server, which is almost always a RADIUS server. In step 3, The Authentication Server requests the credentials of the supplicant and specifies the type of credentials required to confirm the supplicant's identity. (Note here that the arrows between the RADIUS server and the client indicate logical, not physical, connectivity. All traffic between the two passes through the NAS.)

In step 5 the Authentication Server (RADIUS) forwards the access request to the AD server. The AD server responds with a success or failure message, and if successful, also forwards the client's AD domain credentials in step 6. Upon validating the supplicant's credentials, the Authentication Server transmits a success/failure message to the NAS (step 7). In step 8, if access is granted, the NAS opens the port to all traffic. If access is granted, then (step 9) the supplicant is able to access authorized network resources.

**Figure 6.4** EAP Authentication with Authorization

In this scenario, administrators can limit user access to specific VLANs, and via Windows permissions, to most network resources. The specifics of authentication and authorization depend upon the type of EAP policy chosen. There are a variety of them, and we'll look at those most widely deployed.

## EAP Authentication Types

Most of the more recent EAP types are made up of two components: an outer and an inner authentication type, separated by a forward slash—such as PEAPv0/EAP-MSCHAPv2. The outer type defines the method used to establish an encrypted channel between the client (peer) and the Authentication Server.

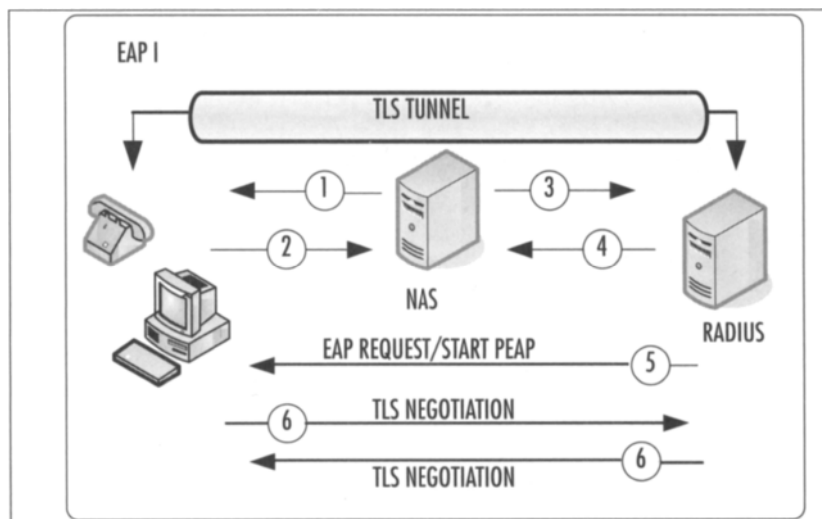
### NOTE

The primary goal of the Transport Level Security (TLS) Protocol is to provide privacy and data integrity between two communicating applications. TLS is based on the Netscape SSL 3.0 Protocol Specification, although they are not interoperable. The protocol is composed of two layers: the TLS Record Protocol and the TLS Handshake Protocol, and is situated between ISO layers 3 and 4. Symmetric cryptography is used for data encryption (e.g., DES, RC4,

AES, etc.). The keys for this symmetric encryption are generated uniquely for each connection. Message transport includes a message integrity check using a keyed MAC (SHA, MD5). These two elements ensure data confidentiality and integrity for each connection.

In Figure 6.5 an outer authentication method, PEAP, is negotiated between a client such as an IP phone or a workstation and a RADIUS authentication server. The intermediate NAS proxies the first several exchanges and then serves to passively mediate traffic in both directions. The NAS does not have knowledge of the keys used to instantiate the TLS tunnel, and thus, cannot be used to snoop on the encrypted traffic passing through it.

**Figure 6.5** EAP Part I Outer Tunnel

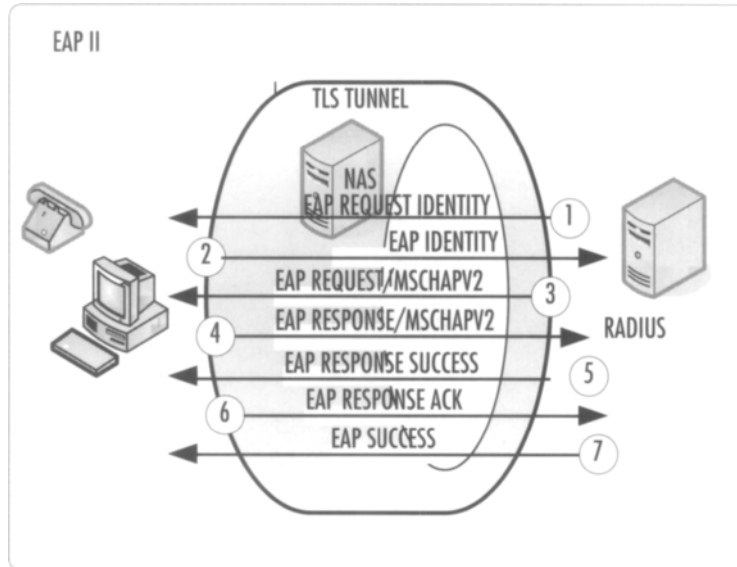


This outer tunnel verifies the server to the client using digital certificates.

Once the outer channel is established, the inner authentication type passes the user's credentials to the Authentication Server over this TLS encrypted tunnel for additional authentication of, typically, user credentials. Passing user credentials through the TLS encrypted tunnel protects them from exposure (see Figure 6.6).

One of EAP's potential security vulnerabilities is that data exchanged as part of some of the outer authentication types, such as identity data, and the results of parameter negotiations are sent in the clear. This can result in a Denial-of-Service (DoS) condition since an attacker, for example, can flood the connection with different types of EAP notification messages.

Figure 6.6 EAP Part II Inner Tunnel



In Table 6.3 some of the characteristics for the different types are summarized. In the last two fields more plus signs (+) equals greater difficulty and more strength, respectively.

Most of the newer EAP types defined by the Wi-Fi Alliance (those with the forward slash and EAP-SIM) are derived from this EAP type. EAP-PEAP and PEAPv0/EAP-MSCHAPv2 are the same thing. PEAPv1/EAP-GTC is a Cisco invention.

## EAP-TLS

EAP-TLS (Extensible Authentication Protocol–Transport Layer Security) provides for certificate-based and mutual authentication of the client and the network. EAP-TLS is the most secure of the common EAP types, but requires a PKI (public key infrastructure) to manage and distribute client certificates. The TLS protocol has its roots in the Netscape SSL protocol, which was originally intended to secure HTTP. It provides either one-way or mutual authentication of client and server based on certificates. In its most typical use in HTTP, the client authenticates the server based on the server's certificate and establishes a tunnel through which HTTP traffic is passed. Username and password management in this scheme is irrelevant as identity is based upon possession of the appropriate private key. The obligatory overhead of a certificate management infrastructure normally precludes use of this EAP type.

**Table 6.3** EAP Types Summary

EAP Type	Server Authentication	Client Authentication	Native Windows 2003 Support	Confidentiality	Integrity	Deployment Difficulty	Security Strength
EAP-TLS	Certificate	Certificate	Yes	TLS	+	+++++	+++++
EAP-PEAP	Certificate	Certificate, Smartcard, MS-CHAP-V2	Yes	TLS	+	++	++++
PEAPv0/ EAP-MS CHAPv2	Certificate	Certificate, Smartcard, MS-CHAP-V2	Yes	TLS	+	++	++++
EAP-TTLS	Certificate	PAP, CHAP, EAP, MS-CHAP-V2, Certificate	No	TLS	+	+++	++++
PEAPv1/ EAP-GTC	Password hash	Password hash (Token)	No	No	+	???	+++
EAP-SIM	128-bit secret	SIM secret	No	+/-	+/-	+++	++
EAP-FAST	Optional (PAC) password	Password (PAC)	No	+	+	+++	+++
LEAP	Password	Password	No	+	+	+++	+
MD5	None	None	Yes	-	-	+	+

## EAP-PEAP

EAP-PEAP (Extensible Authentication Protocol–Protected Extensible Authentication Protocol) provides a method to transport secure authentication data, including legacy password-based protocols. PEAP accomplishes this by tunneling user credentials over a TLS tunnel between PEAP clients and an authentication server. EAP-PEAP is the best combination of security and ease of deployment in Windows environments today. EAP-PEAP requires only a server certificate (which is simple enough to create for testing using the native MS Certification Authority) and client side username/password combinations. EAP-PEAP is natively supported on Windows XP and Windows 2000 SP4 and above client platforms and IAS (Internet Authentication server). PEAPv0/EAP-MSCHAPv2 is the same thing as EAP-PEAP.

## EAP-TTLS

EAP-TTLS (Extensible Authentication Protocol–Tunneled Transport Layer Security) is supported primarily by the Funk RADIUS people. EAP-TTLS, like PEAP, is also relatively easy to deploy (it requires only a server-side certificate) and quite secure since it tunnels user credentials inside of a TLS tunnel; however, this Funk Software invention has not been supported by Microsoft on clients or IAS server. Thus, EAP-TTLS requires the use of an additional software. TTLS and PEAP are similar in other ways, but there are differences: TTLS supports other EAP authentication methods and also supports inner authentication methods, PAP, CHAP, MS-CHAP, and MS-CHAPv2; whereas PEAP can tunnel only EAP-type protocols such as EAP-TLS, EAP-MS-CHAPv2, and EAP-SIM.

## PEAPv1/EAP-GTC

PEAPv1/EAP-GTC (Extensible Authentication Protocol–Generic Token Card) was defined in RFC2284 along with one-time passwords, and MD5 was one of the initial set of EAP Types used in Request/Response exchanges. Cisco supports this type of PEAP (v1 vs. v0) and Microsoft supports only PEAPv0.

## EAP-FAST

EAP-FAST (Extensible Authentication Protocol–Flexible Authentication via Secure Tunneling) was developed by Cisco. EAP-FAST authenticates both the client and the authentication server using a preshared secret known as the Protected Access Credential (PAC). EAP-FAST is a certificate-free replacement for LEAP. EAP-FAST is easy to implement in Windows/Cisco mixed environments, but this method is vulnerable to MITM (man in the middle) attacks in which an attacker can acquire the MS-CHAPv2 hash of the user's passwords, which can then be subjected to off-line dictionary attacks.



## LEAP

LEAP (Lightweight Extensible Authentication Protocol) is an EAP authentication type used primarily in Cisco Aironet WLANs. LEAP supports strong mutual authentication, based upon a modified MS-CHAPv2 challenge/response, between the client and a RADIUS server using a logon password as the shared secret. It provides dynamic per-user, per-session WEP encryption keys. LEAP has been superseded by EAP-FAST due to the public availability of LEAP hash cracking tools such as ASLEAP. There is some disagreement regarding the value of complex password enforcement when using LEAP. When in doubt, use the longest, most complicated passwords that your userbase will agree to.

## EAP-MD-5

EAP-MD-5 (Extensible Authentication Protocol–Message Digest) is an EAP authentication type that provides base-level EAP support. EAP-MD5-Tunneled is an EAP protocol designed for use as an inner authentication protocol within a tunneling protocol such as EAP-TTLS or EAP-PEAP. This has additional security features, but has not been widely deployed.

### Notes from the Underground...

#### RainbowCrack

Passwords are the most common form of computer authentication today. Password encryption is done using a one-way hashing algorithm such as MD5 or SHA-1. A one-way hash function, also known as a message digest, is a mathematical function that takes a variable-length input string and converts it into a fixed-length binary sequence that is computationally difficult to invert—that is, generate the original string from the hash. Conventional password crackers grab a word or string of wordlike tokens and run it though the hash algorithm. It then compares its generated hash with the target password hash. If they match, then the password has been discovered. The computationally expensive part of this process is the hash generation preceding the hash comparison, not the actual comparison process itself.

RainbowCrack is a general-purpose implementation of Philippe Oechslin's faster time-memory trade-off technique. In short, the RainbowCrack tool is an extremely fast and effective hash cracker. The simple but brilliant idea of time-memory trade-off is to do all the hash generation computation in advance and store the result in chains of files called "rainbow tables." It does take a long time to precompute the tables (it takes 2–3 days to generate the rainbow tables necessary to crack a lowercase-letters-only Windows (LM hash) password that's between 1 and 7 characters in length), but

Continued

after this one-time computation is finished, a time-memory trade-off cracker can crack passwords hundreds or thousands of times faster than a brute force cracker.

## Inner Authentication Types

A number of inner authentication methods exist. The most commonly used is MS-CHAP-V2 because it is relatively secure and it is supported natively on all recent Microsoft clients. Additionally, PAP, CHAP, MD5, GTC, and other inner authentication methods exist but are not nearly as commonly used. Interestingly, even EAP itself can be tunneled within EAP.

### *MS-CHAP v2*

MS-CHAP v2 is a one-way encrypted password, two-way authentication process that provides mutual authentication between peers (see Figure 6.7). It differs from MS-CHAP-V1 because it piggybacks an additional peer challenge (PCS) on the Response packet and an additional authenticator response on the Success packet. Both the authenticating server and the client challenge and authenticate each other. The message flow is as follows:

**Figure 6.7** MS-CHAP-V2



1. Authenticator sends a challenge consisting of a Session ID and random authenticator challenge string (ACS).
2. Client (peer) sends a response containing an encrypted one-way hash of the session ID, username, a peer challenge string (PCS), the peer response (PR), and the user password (secret).
3. Authenticator responds with another one-way hash (based on the client response) of a success/failure code, the authenticator response (AR), and the user's password (secret).

4. The peer verifies the authenticator response and begins communications if the response is successful. It disconnects on failure.

This authentication method depends upon a secret (password) known only to the authenticator and the peer. The secret is not sent over the link. A one-way hash function, also known as a message digest, is a mathematical function that takes a variable-length input string and converts it into a fixed-length binary sequence that is computationally difficult to invert—that is, generate the original string from the hash.

### *CHAP and MS-CHAP*

CHAP was defined in RFC1994: PPP Challenge Handshake Authentication Protocol. CHAP (Challenge-Handshake Authentication Protocol) was initially used to verify client identity on PPP links using a three-way handshake. The handshake begins with the authenticator issuing a challenge to the client. The client responds with a digest calculated using a hashing function. The authenticator then verifies the response and acknowledges the connection if the match is successful, otherwise it terminates the connection. CHAP depends upon a secret known only to the authenticator and the client. The secret is not sent over the link.

MS-CHAP differs from CHAP in that MS-CHAP does not require that the shared secret be stored in cleartext at both ends of the link. The Microsoft client knows the hash method used by the server so it can reproduce it, effectively creating a “matching” password on both ends. The client proves its identity based on the fact that it can reproduce the hashed value of the password.

### *PAP*

PAP (Password Authentication Protocol) is described in RFC1334. PAP provides a simple method for the peer to establish its identity using a two-way handshake. PAP is not a strong authentication method. Passwords are sent over the connection in cleartext and there is no protection from playback or repeated trial and error attacks.

### *MD5*

MD5 (Message-Digest algorithm 5) is a widely used cryptographic hash function that results in a 128-bit hash value. The 128-bit (16-byte) MD5 hashes (also termed message digests) typically are represented as 32-digit hexadecimal numbers (for example, `ec55d3e698d289f2afd663725127bace`). EAP-MD-5 typically is not recommended for wireless LAN implementations because it may expose the user’s password, and because several collision-based weaknesses have been demonstrated. It provides for only one way authentication - there is no mutual authentication of wireless client and the network. And very importantly it does not provide a means to derive dynamic, per-session wired equivalent privacy (WEP) keys.

## GTC

Typically, password (PIN) information is read by a user from a token card device and entered as ASCII text into the client. GTC is similar to PAP in that passwords are sent in the clear.

### Notes from the Underground...

#### Dictionary Attacks

Passwords can be broken in real-time (active) and offline (passive) modes. The premise of a dictionary attack is that by trying every possible combination of words (or tokens), an attacker ultimately will succeed in discovering user secret passwords. A dictionary attack relies on the fact that a password is often a common word, name, or concatenation of words or names with a minor modification such as a trailing digit or two. Longer passwords with a variety of characters (such as `^Y2o4uEA16r3-2e64A12EFing!`) offer the greatest protection against dictionary attacks.

During an online dictionary attack, an attacker tries to actively gain network access by trying many possible combinations of passwords for a specific user. Online dictionary attacks can be prevented using password lockout mechanisms that lock out the user account after a certain number of invalid login attempts. Online attacks also generally show up in logs, which can indicate that this type of “loud” hacking activity occurred or is occurring. Offline attacks rely on the attacker’s ability to capture and record data from the datastream usually by using a sniffer such as `tcpdump` or `ethereal`. These captured data can then be compared at leisure against tables of hashes until a password is discovered or the attacker gives up. The offline attacks can be thwarted by changing passwords regularly and limiting attackers’ access to the datastream.

## Public Key Infrastructure

The very starting point of Internet or VoIP security is to correctly identify the user or servicing nodes, called subjects, without leaving any room for impersonation or spoofing.

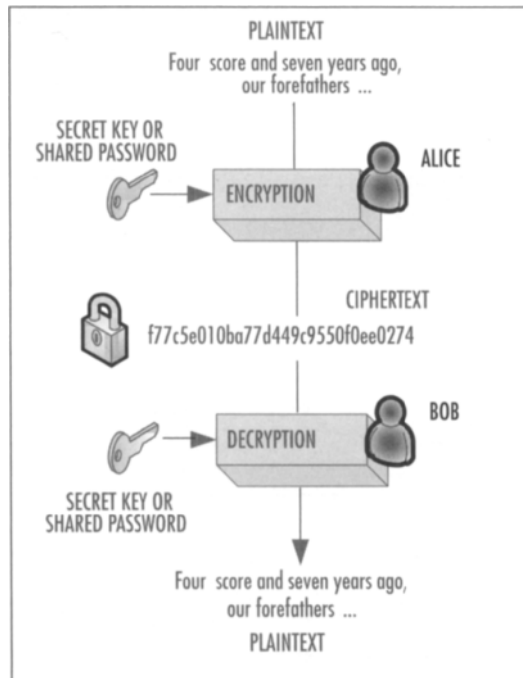
Subjects include all the entities that hold or issue certificates such as an end entity or CA. IETF adopted Public-Key Infrastructure (PKI) as its basis for subject identification. PKI is known to satisfactorily meet the needs of deterministic, automated identification, authentication, access control, and authorization functions.

The IETF RFC 3280 specification profiles the format and semantics of certificates and Certificate Revocation Lists (CRL) for the Internet PKI. The goal of this specification is to develop a profile to facilitate the use of X.509 certificates within Internet applications. Such applications could include WWW, electronic mail, user authentication, and VoIP.

## Public Key Cryptography Concepts

Within the PKI framework, who you are is defined by the private keys you possess. From the point-of-view of PKI authentication authorities, you are your private key. In order to understand PKI, you will first have to understand some basic cryptological concepts. In Figure 6.8 the concept of a secret key is presented. Alice and Bob often are used as examples of the two parties engaged in a secure communications channel, and we will use them here. In this case, Alice and Bob both possess the same secret key. This can be a password, a token, or some other form of secret. Alice encrypts the plaintext that she wishes to send to Bob using her secret key. After Bob receives the ciphertext, he decrypts it using the same secret. The fact that *the same key* is used for both encryption and decryption determines that this is a symmetric exchange.

**Figure 6.8** Symmetric Key Cryptography

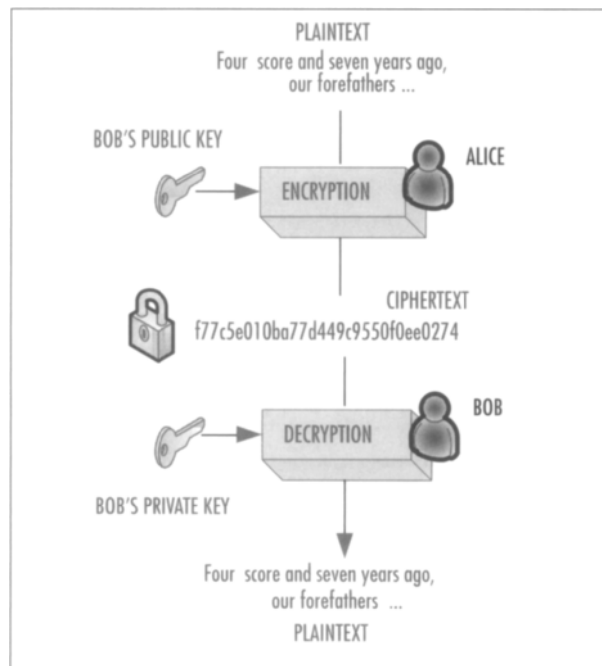


PKI relies on a public/private key combination. The public and private keys are mathematical entities that are related. One key is used to encrypt information and only the related key can decrypt that same information; however, if you know one of the keys, it is computationally unfeasible to calculate the other. Your public key is something that you make public. It is freely distributed and can be accessed by everyone. A corresponding (and unique) private key is something that you keep secret. It is not shared with anyone. Your private key enables you to prove, unequivocally, that you are who you claim to be.

In Figure 6.9, Alice uses public key cryptography to send a ciphertext to Bob. She first locates Bob's public key (normally from some type of directory service or from a previous secured document that Bob has sent to her) and encrypts the plaintext with Bob's public key. She sends the encrypted text to Bob. Only Bob has the corresponding private key that can be used to decode the ciphertext.

Note that in normal practice, for performance reasons, the actual ciphertext is encrypted using a secret key algorithm as shown in Figure 6.8. Symmetric algorithms are much faster than public/private key algorithms (asymmetric cryptography). A random key (the session key) is generated, and it is used with the symmetric algorithm to encrypt the information. The public key is then used to encrypt that key and both are sent to the recipient. The private key is then used to decrypt the session key, and the resulting session key is used to decrypt the actual data.

**Figure 6.9** Public Key Cryptography



The developers of public key cryptography were economical with keys. Both the public and private key are used for more than just encrypting and decrypting data or session keys. The private key also is used to digitally sign the sent message so that the sender's identity is guaranteed. If the sender wishes to prove to a recipient that they are the source of the information (perhaps they accept legal responsibility for it), the sender uses his or her (or its) private key to digitally sign a message (a digital signature). Unlike a handwritten signature, a digital signature is different every time it is created. To create the digital signature, a hash of

the message is signed (encrypted) with the sender's private key. The encrypted value either is attached to the end of the message or is sent as a separate file together with the message. The sender's public key that corresponds to this private key may also be sent with the message, either on its own or as part of a certificate.

The receiver uses the sender's public key to verify that the message hash calculated by the receiver (when certificates are used, the type of hashing algorithm will be included in the public key certificate sent with the message) is the same as the original hash. If the values match, the receiver is reasonably assured that the sender (the individual or device that owns the private key that corresponds with the public key) sent the information. The receiver also is reasonably assured that the information has not been altered since it was signed. This exchange forms the basis for two key security principles: nonrepudiation (the identity of the sender is verified) and message integrity (the contents of the message have not been altered in transit). Table 6.4 summarizes the intended use and owner of both public and private keys in public key cryptography.

**Table 6.4** Key Usage in Public Key Cryptography

Function	Key Type	Key Owner
Encrypt Data	Public Key	Bob (Receiver)
Sign Data	Private Key	Alice (Sender)
Decrypt Data	Private Key	Bob (Receiver)
Verify Data Integrity	Public Key	Alice (Sender)

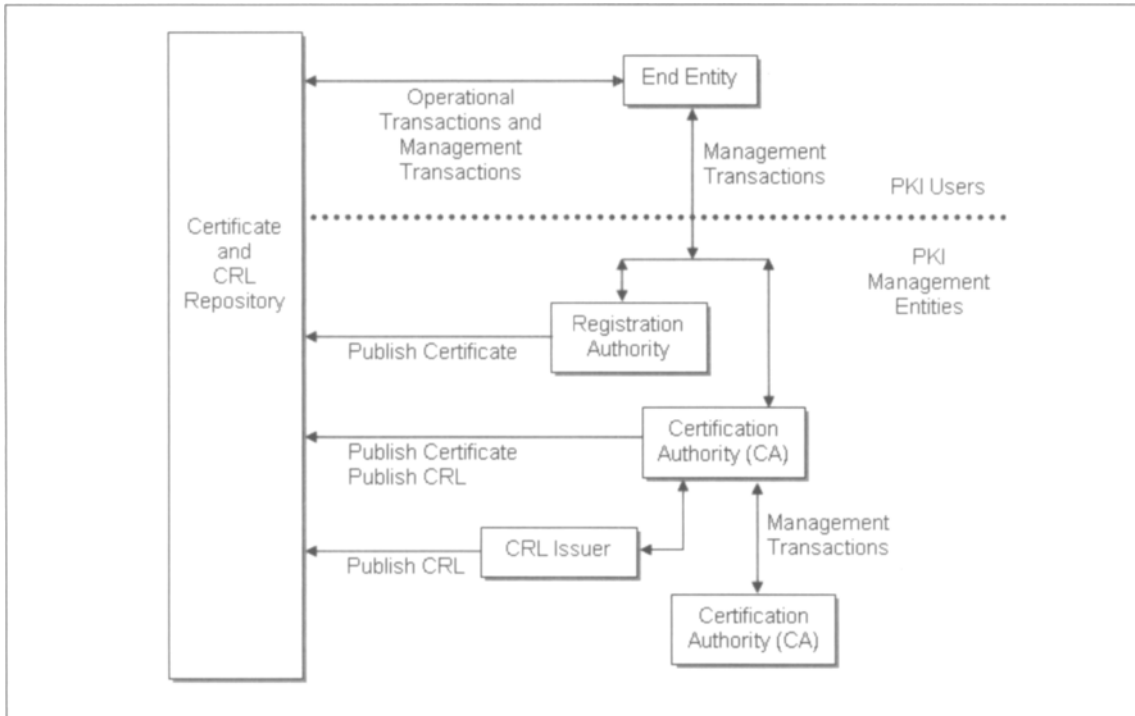
## Architectural Model and PKI Entities

Figure 6.10 shows a simplified view of the architectural model assumed by the PKI specification. This model is analogous to the credit card infrastructure. Even though the data is encrypted differently, the ways in which the entities in the two structures interact with each other are conceptually similar. Each PKI entity is like an entity in the credit card infrastructure.

We'll now define the following PKI entities:

- **End Entity** User of PKI certificates and/or end-user system that is the subject of a certificate. Like a credit card reader in a retail store or restaurant, it reads a user certificate (credit card number) and queries the credit card company for the card holder's legitimacy and credit limits.
- **Certification Authority (CA)** A system that issues PKI certificates. Think of credit card application processing, which checks an applicant's credit history and issues a credit card.

Figure 6.10 PKI Entities and Their Relationships



- **Registration Authority (RA)** An optional system to which a CA delegates certain management functions.
- **CRL issuer** An optional system to which a CA delegates the publication of certificate revocation lists. This entity manages the equivalent of a stolen or lost credit card report and distributes certificate revocation information.
- **Repository** A system or collection of distributed systems that stores certificates and CRLs and that serves as a means of distributing these certificates and CRLs to end entities. An analogy would be a credit card holder database.

Operational protocols deliver certificates and CRLs (or status information) to client systems that use certificates. A variety of different ways to deliver certificates and CRLs are needed, including distribution procedures based on Lightweight Directory Access Protocol (LDAP), HTTP, File Transfer Protocol (FTP), and X.500.

Management protocols support online interactions between PKI user and management entities. For example, a management protocol might be used between a CA and a client system with which a key pair is associated, or between two CAs that cross-certify each other. The set of functions potentially needing to be supported by management protocols include user registration, client initialization, user certification, periodic key pair update, revocation request, and cross-certification.



## Basic Certificate Fields

Basic certificate fields for X.509 version 3 are shown in Table 6.5. The To Be Signed (TBS) certificate field contains the names of the subject and issuer, a public key associated with the subject, a validity period, and other associated information. It usually includes extensions which hold additional optional information. The subject field identifies the entity associated with the public key stored in the subject public key field. It also distinguishes if a certificate is for an end entity, a CA, or a CRL. The Subject Public Key Info (SPKI) field is used to carry the public key and to identify the algorithm by which the key is used (e.g., RSA, DSA, or Diffie-Hellman).

The signature algorithm field contains the identifier for the cryptographic algorithm used by the CA to sign the certificate.

The signature value field contains a signature digitally added to the encoded TBS certificate. By generating this signature, a CA certifies the validity of the information in the TBS certificate. To be more specific, the CA certifies the binding between the public key material and the subject of the certificate.

**Table 6.5** Basic Certificate Fields for X.509

Certificate Fields	Attribute	Type	
TBS Certificate	Version	V1, v2, v3	
	Certificate Serial Number	Integer	
	Algorithm Id	Algorithm Object Id.	
	Issuer	Name	
	Validity		Not before time
			Not after time
	Subject	Name	
	Subject Public Key Info		Algorithm Id
			Bit string
	Issuer Unique Id	Bit string	
Subject Unique Id	Bit string		
Extensions			
Signature Algorithm		Algorithm Id	
Signature Value		Bit string	

## Certificate Revocation List

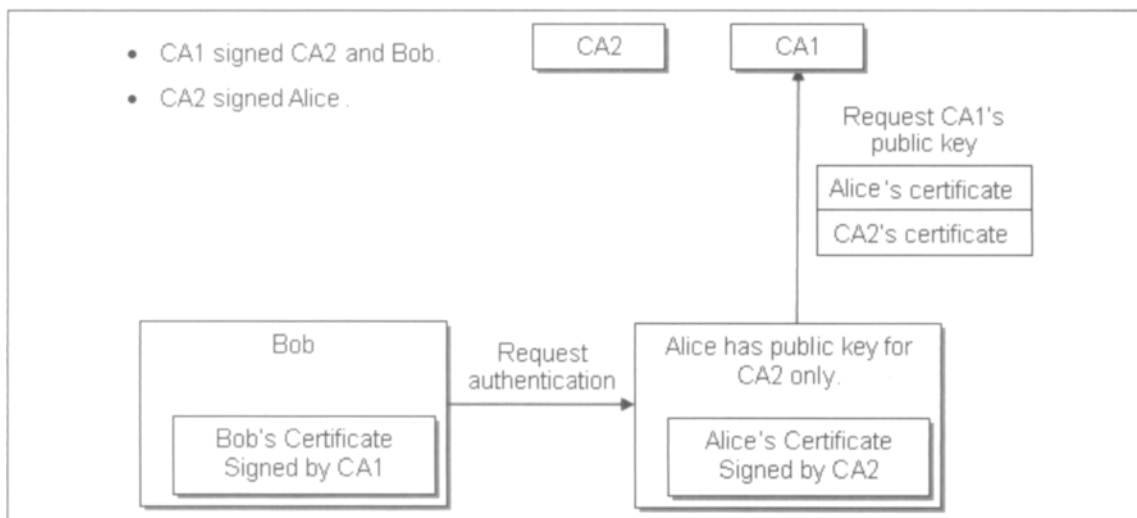
When a certificate is issued, it is expected to be in use for its entire validity period. However, various circumstances may cause a certificate to become invalid prior to the expiration of the validity period. Such circumstances include change of name, change of association between subject and CA (e.g., an employee terminates employment with an organization), and compromise or suspected compromise of the corresponding private key. Under such circumstances, the CA needs to revoke the certificate.

CRL is similar to notices of stolen or lost credit cards reported to other credit companies. The CA periodically issues a signed data structure called a CRL. A CRL is a time-stamped list identifying revoked certificates. The list is signed by a CA or CRL issuer and made freely available in a public certificate and CRL repository. Each revoked certificate is identified in a CRL by its certificate serial number. When a system employing certificates uses a certificate for verifying a remote user's digital signature, that system not only checks the certificate signature and validity, but also acquires a recent CRL and checks that the certificate serial number is not on that CRL.

## Certification Path

If a public key user does not already hold a copy of the CA that signed the certificate including the CA's name, then it might need an additional certificate to obtain that public key. A sample scenario appears in Figure 6.11. Let's assume that Bob requested authentication from Alice with his certificate signed by CA1. But Alice, whose certificate was signed by CA2, does not have the public key for CA1, which is required to validate Bob's certificate. Then, Alice forms a certificate chain that contains both CA2's and her certificate and requests that CA1 provide a public key for CA1.

**Figure 6.11** A Sample Certification Path



In general, a chain of multiple certificates might be needed that would make up a certificate containing the public key owner (the end entity) signed by one CA, and zero or more additional certificates originating from CAs signed by other CAs. Such chains, called certification paths, are required because a public key user is initialized with only a limited number of assured CA public keys. Certification path processing verifies the binding between the subject name and subject public key. This requires obtaining a sequence of certificates that support that binding.

Many organizations elect to create self-signed certificates for their public key infrastructure rather than purchase one or more from a Certificate Authority. In most cases, this is fine. However there are two differences between self-signed certificates and CA-signed certificates. SSL-enabled Web browsers normally recognize a CA-generated certificate and automatically allow a secure connection to be made, without prompting the user. Self-signed certificates usually generate an annoying (and sometimes to nontechnical users, frightening) pop-up. CAs also guarantee the identity of the organization that is providing services to the browser or other certificate-enabled device.

Before signing a certificate, a CA verifies the identity of the requesting organization. Thus, if your PKI is accessed by the public at large, you should provide a certificate signed by a CA so that people who visit or call know that your infrastructure is owned by the organization who claims to own it.

## Minor Authentication Methods

Information security often is defined as a number of layers. The basis for this is the idea that every time and place a logical or physical impediment can be created that might reasonably stop an attacker (without hindering normal users' access to network resources) it should be done. 802.1x/EAP and PKI are large, complex layers, that when implemented and maintained correctly, result in highly secured access. There are a number of less expensive, less labor-intensive measures that administrators can take that also result in restricting network access to authorized devices.

## MAC Tools

A basic security rule is that endpoints cannot be trusted until the identity of the endpoint is confirmed, or authenticated. In the case of VoIP, a method for authentication of IP phones is the hardware or MAC address. The MAC (Media Access Control) address is a six-byte address that usually is represented as hex numbers in the form AA-BB-CC-DD-EE-FF or AA:BB:CC:DD:EE:FF. The first three bytes represent the vendor ID and the remaining three bytes form a unique address for any network connected device. There are potentially  $2^{48}$  or 281,474,976,710,656 possible MAC addresses. The Web site [http://coffer.com/mac\\_find/](http://coffer.com/mac_find/) is useful for doing MAC/Vendor lookups.

## MAC Authentication

If an IP phone with an unknown MAC address attempts to download a configuration from a registration server, then that device should not receive a configuration assuming automatic registration has been disabled. This setup prevents someone from placing a rogue phone or sniffer into the network, unless of course the person spoofs the MAC address in hopes of intercepting calls.

## ARP Spoofing

ARP spoofing is an essential part of call interception. If an attacker cannot successfully meddle with the switch's ARP table then eavesdropping is virtually eliminated. Of course, unrestrained console access to a switch also offers the chance for call interception; however, appropriate physical security controls and good passwords will minimize this threat. This topic and countermeasures are discussed in detail in Chapter 8.

## Port Security

Since 802.1x is still an emerging technology, not all devices support it. Devices that do not support 802.1x can be controlled by Media Access Control (MAC) address authentication. Devices with static IP addresses that do not support 802.1x (such as printers and some IP phones) can be accommodated by utilizing various port security commands without the use of 802.1x (different switch vendors have different names for these commands). These devices should also be placed into their own VLAN.

## Summary

As VoIP evolves, the requirements for user and device authentication and authorization will evolve as well. VoIP and other contemporary network services necessitate increased requirements for identity management both within and between organizational domains. Users often maintain multiple identities. IP endpoints proliferate. Individuals employ different usernames, passwords, and other identifying attributes in various online contexts, and then they have trouble remembering all these multiple usernames and passwords. The foundation of identity management is authentication services.

Authorized access begins with authentication. We are all familiar with different forms of authentication—from the password used to login to your computer to the key that unlocks your front door. The conceptual framework for authentication is made up of three factors: “something you have” (a key or certificate), “something you know” (a password or secret handshake), or “something you are” (a fingerprint or iris pattern). Authentication mechanisms validate users by one or a combination of these.

User and device identities are not the same and need to be verified independently. This can be accomplished in a number of ways. Microsoft Kerberos and NTLM authentication

are the most widely used authentication schemes due to the large installed Windows 2000 and XP user base. These authentication schemes—particularly Kerberos—provide reasonable security, but Windows authentication is primarily a user authentication scheme, and many VoIP infrastructure components do not run on Windows. In addition, Windows authentication cannot be used to restrict access to the layer 2 network.

The two most commonly used, general-purpose, user and device authentication methods are 802.1x/EAP and PKI. Though they are functionally unrelated, both define umbrella-like suites that provide frameworks for positively identifying users and devices based upon a spectrum of credentials. In addition, both of these approaches are extensible.

802.1x and 802.11i/WPA2 rely on an Authentication Server (usually a RADIUS server) and an Authenticator (usually a switch or wireless access point) to authenticate users and to proxy user credentials, respectively. 802.1x relies on EAP to carry out the authentication process. In the spirit of protocol isolation that has been successfully pulled off in the TCP/IP suite, 802.1x provides support for EAP, which provides a framework for multiple authentication methods, including traditional passwords, token cards, Kerberos, digital certificates and public-key authentication.

These EAP types normally are composed of an inner and outer type, and in many situations, inner and outer types can be mixed to correspond with an organization's specific security requirements.

PKI techniques, methods, and infrastructure components were developed principally to support secure information exchange over insecure networks such as the Internet where such features cannot otherwise be readily provided. PKI, however, can be used just as easily for information exchanged over private networks, including corporate internal networks. PKI can also be used to deliver cryptographic keys between users and devices (including IP phones and servers) securely.

Other point solutions can be used to limit network access to only authorized devices. These are normally vendor-dependent, and typically involve some type of MAC address filtering or access lists.

## Active Security Monitoring

### Solutions in this chapter:

- Network Intrusion Detection Systems
- Host-Based Intrusion Detection Systems
- Logging
- Penetration and Vulnerability Testing

# Introduction

At this point, we have examined and hardened the working components of the existing security infrastructure, established procedures to confirm user and device identities, and logically separated voice and data traffic, thus allowing the network to now carry them. The next step in maintaining the security of this infrastructure is to monitor traffic and the state of key devices. This is accomplished by active monitoring.

Plenty of commercial and open-source tools exist to help with this, and in this chapter we will look at several categories of them. We won't, however, discuss in any detail the large commercial network monitoring suites like NetIQ, SMARTS, BMC Patrol, HP OpenView Operations, HP Network Node Manager NNM, IBM Tivoli, Nortel Optivity NMS, Cisco Ciscoverks, Sun Solstice SunNet Enterprise Manager, Micromuse, Computer Associates CA Unicenter, and Microsoft Operations Manager 2000 (MOM). While we recommend that organizations employ one or more of these enterprise tool suites (particularly to monitor network jitter, packet loss, and latency), the configuration, use, or integration of any one of these tool suites with VoIP network monitoring components is complex, dependent upon both the suite chosen for monitoring, and the peculiarities of each particular network. For these reason we will have to leave this discussion to another time.

A related class of tools for both monitoring and performance testing of VoIP networks include tools like Empirix Hammer, Brix Network Verifier, and Shunra's Virtual Enterprise. These tools use different techniques and metrics to monitor the functionality, performance, scalability, and robustness of VoIP networks to provide signaling and media quality data on every call. Administrators can monitor high-level network metrics via integration with their existing Network Management Systems or can drill into the details of any call down to individual protocol and network messages.

We will start off by discussing in more detail two intrusion detection (ID) technologies: NIDS (network-based) and HIDS (host-based). NIDS inspects all inbound and outbound network activity and identifies patterns of packet data that may indicate a network or system attack. NIDSs are normally arranged in a multiple-sensor-to-one-console configuration, where the sensors reside on dedicated appliances distributed at key network junctions, and report back to a central management console. HIDSs, on the other hand, normally reside on the server that they monitor. HIDS can also report back to a central management console. A third class of intrusion detection is exemplified by DShield or Symantec—distributed intrusion detection—where global system attacks are reported to, and consolidated by, a central management server. Intrusion detection is a requirement in contemporary networks since it is not possible to stay abreast of existing and potential threats to modern computing systems.

Next, we will take a look at logging, primarily focusing on syslog and SNMP. Syslog (system logger) provides a means to allow a machine to send event notification messages across IP networks to event message collectors (also known as syslog servers). The decision regarding how much and what types of data should be logged is a critical responsibility of

the system administrator. However, in most modern systems the sheer amount of logging data generated by system loggers can easily overwhelm most system administrators. We have witnessed organizations that react to log events, not based upon the data contained in the logs, but rather according to the number of logs generated per some unit of time. In order to deal with this mass of data, many system administrators develop scripts or tools to examine the log files and extract the important information. These tools are important because, without them, log data is often ignored. SNMP (Simple Network Management Protocol) is the primary transport for most of the aforementioned large tool suites. There are, however, simple point solution SNMP tools available, and we'll offer suggestions regarding general SNMP usage.

Finally, in this chapter, we will close with a section on penetration testing. Penetration testing is a means of monitoring the state of security controls on your VoIP network. The primary reason for testing systems or networks is to identify potential vulnerabilities and subsequently repair them. Penetration Testing (Pen Testing) is an intelligent combination of automated and manual examinations that are launched from either inside or outside the perimeter of a private network. This testing emulates the threat from hackers and other parties, and their attempts to enumerate and compromise visible services.

Although we are not aware of production ready VoIP-specific NIDS, several are rumored to be in development. As a note: Based upon data gathered from historical analysis of call flows, anomaly detection, particularly in a call center setting where traffic is more defined than in an entire converged network, may prove to be an effective NIDS strategy.

## Network Intrusion Detection Systems

Network Intrusion Detection Systems (NIDSs) are designed to alert administrators when malicious or illegitimate traffic is detected. Malicious traffic can consist of worm or exploit-based code, while illegitimate traffic (often termed “misuse”) consists of traffic that deviates from established security policy such as surfing porn sites or peer-to-peer connections. Network-based IDSs can monitor an entire, large network with only a few well-situated nodes or devices and impose little overhead on a network. NIDSs are found in most networked computing environments today because, no matter how well security controls are implemented, it is impractical to maintain defenses against all known and potential threats to networked systems and applications. In VoIP environments, NIDSs provide an additional layer of defense.

### NIDS Defined

NIDSs detect suspicious activity in three ways. First, the security community maintains an extremely large database of specific attack signatures. These signatures are programmed into the NIDS sensor, and are updated on a regular basis. Examples of attack signatures include Code Red, NIMDA, DoS attacks, buffer overflows, ASP, and CGI vulnerabilities. Second, the



NIDS sensors contain preprocessors that continuously monitor the network for anomalous behavior. Though not as specific as attack signatures, these anomalies are still highly effective for the detection of port scans, distributed network probes, new forms of buffer overflows, and Denial-of-Service attacks. Third, all NIDS appliances can apply and detect security policy deviations. These policy deviations include the detection of unauthorized network services, applications running on unusual ports, and backdoor/Trojan activity.

Signature-based NIDSs are essentially network sniffers combined with a database of attack signatures. One of the most difficult (and necessary) tasks when initially configuring the NIDS is the job of de-tuning it. It is important that the number of false positives be reduced; otherwise, they will make meaningful analysis of the data impossible.

## Components

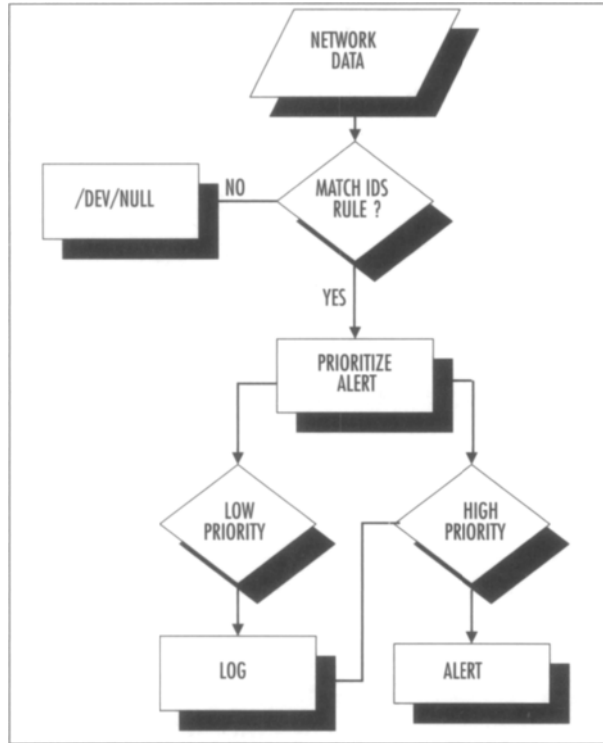
Most NIDSs are configured in a client- (sensor) to-server (management console) configuration. Many sensors normally report to one or several management consoles. Sensors can be dedicated appliances, can run as an application on a host running other applications, or can run independently in a virtual subsystem such as VMware or Xen. Note that if the sensor does not reside on a dedicated appliance, then the OS of the host computer should be hardened.

Because NIDSs do not reside in the datapath (normally one NIC is used as a sensor and a second NIC is used for management traffic), the sensor Ethernet interface can be configured in a number of ways as receive only. Sensor hardware requirements are not particularly strict since the sensor application normally inspects packets, and upon finding a signature or pattern match, sends the subsequent data upstream to the management console for processing and visualization.

The term “signature” refers to a set of conditions that, when met, indicate some type of intrusion event. Typical modern sensors contain a signature database consisting of 1000 to 2000 entries. Often, sensors inspect traffic based upon a mixture of signature matching as well as pattern matching. Pattern matching is based on looking for a fixed sequence of bytes in a single packet. A more sophisticated method is stateful pattern matching. Stateful pattern matching is useful when the intrusion signature spans more than a single packet. Similar to antivirus software, a signature-based IDS requires regular access to an up-to-date database of attack signatures so recent exploits are not missed.

Figure 7.1 is a simple illustration of the basic logic used by NIDS management stations when resolving an event reported by a remote sensor. The “Match IDS Rule” logic normally resides on the sensor. When a rule is matched (for example: “packet from outside to inside contains illegal SIP rerouting headers”), the data is forwarded to the management console where it is prioritized, logged, and visualized.

Figure 7.1 NIDS Logic



The management console (MC) hardware requirement is normally stricter than that of the sensor since the MC is responsible for data correlation from multiple sensors, as well as storage, alerting, and visualization. Often, the MC also includes an integrated sensor.

## Types

NIDSs are normally classified according to the methods they use for attack detection; either as signature-based, or anomaly detection. Note, though, that almost all current NIDSs use a mixture of these approaches. Signature-based approaches, as mentioned earlier in this chapter, rely on some type of pattern matching. NIDS sensors parse the entire IP packet, and make decisions by means of a simple rule-based logic that is based upon signatures or regular expression matching. In other words, they compare the data within a packet payload to a database of predefined attack signatures (a string of bytes). Additionally, statistical or historical algorithms may supplement static pattern matching. Attack signatures usually consist of one or more of the following fields:

- Source and destination IP addresses, or an address or range
- TCP/UDP source and destination ports and ICMP type/code
- IP header flags and options

- TCP header flags and options
- A definition of the payload data to search (hex or ASCII)
- A starting point for the payload search (offset) and the search depth

Analysis of packet headers can be done economically since the locations of packet header fields are restricted by protocol standards. However, the payload contents are, for the most part, unconstrained. Therefore, searching through the payload for multiple string patterns within the datastream can be a computationally expensive task. The requirement that these searches be performed at wirespeed only adds to the cost.

Anomaly detection NIDSs are based on the assumption that normal traffic can be defined, and that attack or misuse patterns will differ from “normal” traffic. Heuristic-based signatures, on the other hand, use some type of algorithmic logic on which to determine their alarm decisions.

#### NOTE

---

Heuristic is the art and science of discovery and invention. The word comes from the same Greek root as “eureka,” which means “I find.” Heuristics defines a problem-solving technique in which the most appropriate solution is selected at successive stages of a program for use in the next step of the program. Heuristic approaches utilize simplification or an educated guess to reduce or limit the search for solutions. A heuristic can be a single algorithmic solution to a problem, but unlike an algorithm, heuristics does not guarantee optimal, or even feasible, solutions.

---

These algorithms are often statistical evaluations of the type of traffic being inspected. An example of a heuristic signature is a signature used to detect a port scan. This signature defines a particular threshold number of external probes against unique ports or a specific combination of ports. The signature may be further restricted by specification of the types of packets (for instance, SYN only) it reacts to. Interesting trends can be learned from these data, and it is possible to detect ongoing attacks based on these algorithms; however, the information that these systems provide is generally very nonspecific and requires extensive human investigation before actionable intelligence is gathered.

By creating baselines of normal behavior, anomaly-based NIDSs are able to detect when current network behavior deviates statistically from the norm. This capability theoretically gives an anomaly-based NIDS the capacity to detect new attacks that are either unknown or to detect attacks for which no signatures exist.

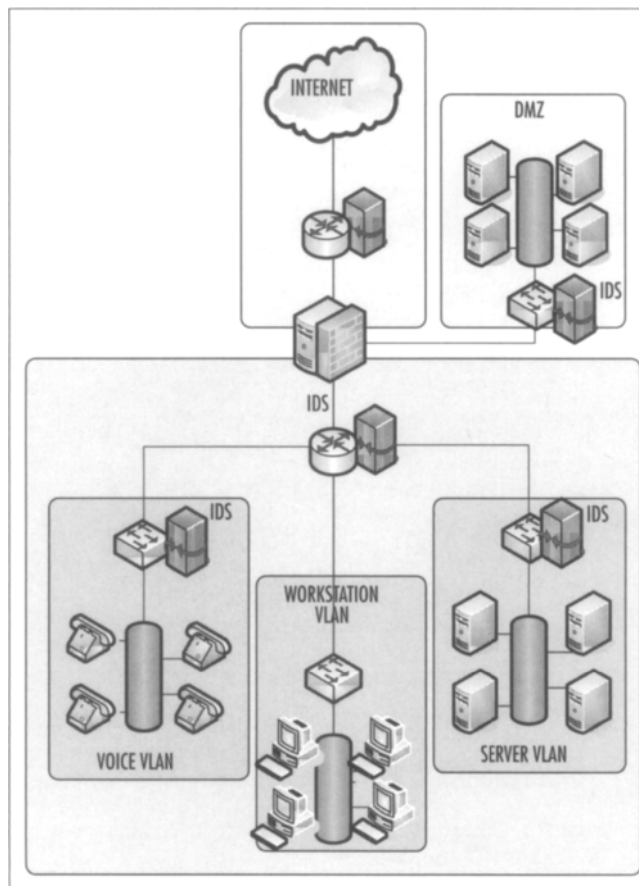
The major problem with this type of approach is that normal network traffic is difficult or impossible to define. Since normal network behavior can change easily and readily,

anomaly-based NIDSs are prone to false positives. Additionally, inconsistency of detector performance, training issues (for example, how often an anomaly-based detection system should be retrained to ensure acceptable performance), and inadvertent incorporation of intrusive behavior into an NIDS concept of normal behavior during training negatively affect performance.

## Placement

NIDSs should be located where they can most effectively monitor critical traffic. This doesn't necessarily mean that NIDSs should be placed where they can monitor the *most* traffic. In Figure 7.2, an example network is diagrammed. This network consists of a single Internet connection, a DMZ (demilitarized zone), and three internal VLANs, configured for voice users, workstations, and servers. The circular network symbols signify routers or layer 3 switches, while the square network symbols signify layer 2 switches. The five NIDSs are shown as arrowed rectangular boxes.

**Figure 7.2** NIDS Locations



In this figure, an NIDS is located on the external side of the firewall to monitor all inbound and outbound Internet traffic. NIDSs are located on internal layer 2 switches in the voice and server VLANs, and on the layer 3 switch that is used to truck these connections. An additional NIDS is situated in the DMZ. In this architecture, the NIDSs will have access to all the network traffic, but are they all really necessary?

Frankly, no. Several of the NIDSs are either redundant or will report so many events as to be meaningless. The external NIDS is unnecessary since it is exposed to the Internet. Even those of you with broadband connections realize that your single IP address is constantly bombarded with exploit probes and port scans. The following is a sample from 30 minutes of scans against a typical home system.

```
~ # tail -f /var/log/messages | egrep -v "repeated"
Oct  3 00:27:33 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.193.208.77:2258 flags:0x02
Oct  3 00:30:26 ns1 /kernel: Connection attempt to TCP 192.168.20.20:901 from
211.172.40.72:4896 flags:0x02swat
Oct  3 00:30:27 ns1 /kernel: Connection attempt to TCP 192.168.20.20:901 from
211.172.40.72:4896 flags:0x02swat
Oct  3 00:30:58 ns1 /kernel: Connection attempt to TCP 192.168.20.20:445 from
83.37.160.160:3400 flags:0x02
Oct  3 00:31:00 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.199.122.40:3829 flags:0x02
Oct  3 00:31:01 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.199.122.40:3829 flags:0x02.
Oct  3 00:31:01 ns1 /kernel: Connection attempt to TCP 192.168.20.20:445 from
83.37.160.160:3400 flags:0x02
Oct  3 00:31:01 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.199.122.40:3829 flags:0x02
Oct  3 00:31:43 ns1 /kernel: Connection attempt to UDP 192.168.20.20:137 from
66.63.173.19:1316          netbios-ns
Oct  3 00:31:47 ns1 /kernel: Connection attempt to UDP 192.168.20.20:137 from
66.63.173.19:1316
Oct  3 00:31:54 ns1 /kernel: Connection attempt to TCP 192.168.20.20:1433 from
212.33.102.36:2784 flags:0x02mssql/slammer
Oct  3 00:32:03 ns1 /kernel: Connection attempt to UDP 192.168.20.20:137 from
66.63.173.19:1316
Oct  3 00:32:27 ns1 /kernel: Connection attempt to UDP 192.168.20.20:137 from
66.63.173.19:1316
Oct  3 00:33:01 ns1 /kernel: Connection attempt to UDP 192.168.20.20:137 from
66.63.173.19:1316
Oct  3 00:43:18 ns1 /kernel: Connection attempt to TCP 192.168.20.20:445 from
24.199.80.94:1831 flags:0x02
Oct  3 00:47:55 ns1 /kernel: Connection attempt to TCP 192.168.20.20:5000 from
24.84.67.76:4593 flags:0x02  UPnP backdoor
```

```
Oct  3 00:47:58 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.84.67.76:3254 flags:0x02
Oct  3 00:48:50 ns1 /kernel: Connection attempt to TCP 192.168.20.20:4899 from
69.60.111.98:1361 flags:0x02 Radmin exploit
Oct  3 00:49:26 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.199.105.227:3192 flags:0x02
Oct  3 00:52:19 ns1 /kernel: Connection attempt to TCP 192.168.20.20:5000 from
24.199.230.130:4456 flags:0x02
Oct  3 00:52:22 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.199.230.130:4276 flags:0x02
Oct  3 00:52:22 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.199.230.130:4276 flags:0x02
Oct  3 00:55:37 ns1 /kernel: Connection attempt to UDP 192.168.20.20:1029 from
203.21.20.30:30065          ICQNUke98
Oct  3 00:55:42 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.199.105.227:4103 flags:0x02
Oct  3 00:56:17 ns1 /kernel: Connection attempt to TCP 192.168.20.20:135 from
24.167.27.37:2995 flags:0x02
```

As you can see, most of this traffic is the result of automated scanning by worms and viruses, or by simple automated scanning tools. In an enterprise environment, where IDS and log data accumulates in copious amounts, these external data can be ignored. It is more important to focus on the events that occur within the firewall perimeter.

The NIDS situated on the layer 2 switches can also be eliminated since this traffic can be monitored at the central layer 3 switch. In addition, the management connection passes through the firewall and may allow an attacker to piggyback into the network if the sensor is compromised. Although there are no hard and fast rules for deploying NIDS, most system administrators deploy them on uplinks and at devices where many VLANs are trunked so that the fewest number of NIDSs can monitor the most traffic. In our sample network, two NIDSs are suitable to monitor most of the network traffic—one NIDS in the DMZ, and one on the central layer 3 switch.

Note that on the layer 3 switch, there are two interesting and separate traffic flows—one on the uplink between the switch and the firewall, and one port that trunks inter VLAN traffic. The choice of how to monitor both traffic flows depends on how the switch-NIDS connection is configured. Two common methods for allowing NIDS access to network traffic are port mirroring (spanning) and the insertion of a tap (we recommend NetOptics taps because they have two power connections and a wide choice of physical interfaces; check them out at [www.netoptics.com](http://www.netoptics.com)). Port mirroring, depending upon its configuration, can enable the NIDS to inspect all of the traffic traversing the switch, and is an inexpensive option. However, some vendor's switches or OS revisions break when port mirroring is enabled. Be sure to check this with your vendor before connecting an NIDS. The second option, a network tap, is more expensive but offloads the mirroring to a separate device. In

this simple example network setting, two network taps would have to be used to visualize all of the traffic—one on the uplink, and one on an inter VLAN port.

## Important NIDS Features

Let's now discuss the important features of an NIDS.

### Maintenance

Most NIDS systems support centralized installation, configuration, and updating since in an enterprise network a security administrator cannot physically access each sensor. In addition, most vendors support the automated download of signatures and software updates.

Distribution and customization of the signature libraries and policies should be possible on a per-sensor basis and on a per-group basis (these groups should be defined by the security administrator) so the group signatures and policies do not have to be pushed to each sensor individually.

Communication between the IDS components (sensors and management console) should be encrypted using strong authentication (via key exchange or challenge). And as mentioned earlier, NIDS Ethernet interfaces should be stealthy. Transmission of data via the sensing interface is prohibited, unless it is configured intentionally (TCP resets, which we discuss later in this chapter, may be an exception).

### Alerting

The management console should be configurable to support alerting via a variety of mechanisms, including SNMP traps, e-mail alerts, pager messages, syslog messages, SMS (short message service), IM, and console alerts.

### Logging

All alerts and header and payload data should be automatically stored in a central event database that is backed up regularly via SCP or other secure means.

### Extensibility

The NIDS should support simple integration of additional vulnerability assessment tools such as Nmap or Nessus, and should provide support the correlation of data from other IDSs (for example, NIDS and HIDS).

### Response

Some NIDS are able to actively respond to attacks or misuse by interfering with the particular message stream that generated the alert. This is normally accomplished via targeted TCP resets that eventually tear down the connection, or by dynamically altering firewall rules or

Access Control Lists to block the connection. These active-response NIDSs are often referred to as intrusion prevention systems (IPSs).

Most administrators do not activate these features because of the risk of blocking normal traffic. Imagine that this functionality was enabled on a system directly connected to the Internet. A clever attacker could send traffic to an IPS with the source address spoofed to that of an upstream router, and designed to trigger the IDS. The resultant blocking of the upstream router could effectively remove the organization from the Internet. In a VoIP environment where availability is a key metric, IPSs are not recommended because of this potential to obstruct voice traffic.

## Limitations

NIDSs that rely upon signatures must constantly update the signature database. Obviously, pure signature matching NIDSs will not alert on attacks for which they have no signature. If signature definitions are too specific, signature-based IDSs may miss variations on known attacks. (A common technique for creating new attacks is to modify existing attacks.) Signature-based NIDSs can also impose noticeable performance problems on systems when numerous attack signatures are matched concurrently. Additionally, signature-based NIDS inspection can be evaded. Secure Networks showed in 1998 that attacks which exploit fundamental TCP/IP problems—insertion, evasion, and Denial-of-Service attacks—are able to elude NIDS detection. Dan Kaminsky recently showed he could send a series of fragmented packets to a NIDS that, based on the time and the operating system platform that they arrive at, reassemble into an attack for that platform that is not recognized by the NIDS.

## Honeypots and Honeynets

A honeypot is a computer system that is shielded from the Internet by a router or firewall that is transparent to an attacker. The honeypot masquerades as a normal undefended system, yet it logs every action taken against it and every operation that is performed on it. The goal of a honeypot operator is to lure an attacker into hacking the system in hopes of learning all of the details of the attack. A honeypot is a system designed to illustrate the methods used by black-hats to probe for, and exploit, a system. Honeynets are networks that contain at least one honeypot. Typically, honeynets present a virtual network complete with virtual services and applications that look to an attacker like a real network.

Honeypots and honeynets are learning tools, and can also be useful as canaries (canaries were used in mines to provide an early warning to miners if air conditions turned sour). Unlike NIDSs and HIDSs, where false positives are a common nuisance, honeypots and honeynets, if configured correctly, do not have a measurable false positive rate. Honeynets are often configured so that their IP space resides within unoccupied IP space in an organization's internal network. In this configuration, anything that hits the honeynet is either an attack or a precursor to an attack since this IP space is supposedly unused. In its canary role, a honeynet can provide an early warning of a virus or worm attack.



# Host-Based Intrusion Detection Systems

Host-based intrusion detection systems (HIDSs) are applications that operate on information collected from individual computer systems. This vantage point allows an HIDS to analyze activities on the host it monitors at a high level of detail; it can often determine which processes and/or users are involved in malicious activities. Furthermore, unlike NIDSs, HIDSs are privy to the outcome of an attempted attack since they can directly access and monitor the data files and system processes targeted by these attacks.

Tripwire (the reference model for many of the follow-on HIDSs) is described in more detail in the “Server Hardening” section of Appendix A. Tripwire operates on MD5 hashes of critical system files, as defined by the system administrator. It is one model for host-based intrusion detection—like the secret agent trick of putting a hair on the doorknob, it lets you know if somebody’s been changing things inside your system—but only *after* this occurs.

Alternatively, HIDSs can utilize information sources of two types, operating system audit trails, and system logs. Operating system audit trails are usually generated at the innermost (kernel) level of the operating system, and are therefore more detailed and better protected than system logs. System logs are much less obtuse and much smaller than audit trails, and are normally far easier to comprehend.

Most HIDS software, like Tripwire, establishes a “digital inventory” of files and their attributes in a known state, and uses that inventory as a baseline for monitoring any system changes. The “inventory” is usually a file containing MD5 checksums for individual files and directories. This must be stored offline on a secured, read-only medium that is not available to an attacker. On a server with no read-only media (a blade server, for example), one method to accomplish this is to store the statically compiled intrusion detection application and its data files on a remote computer. When you wish to run an HIDS report, SCP (secure copy) the remote files to /tmp (or its equivalent) on the target server and run them from there. When you modify any files on the server, rerun the application, and make a new data set, which should be stored on the remote computer.

HIDS surveillance is especially important on VoIP media, proxy, and registration servers and should be considered as part of the initial install package. Indeed, vendors such as Cisco are even making this part of the default installation. For instance, the Cisco Security Agent (CSA) comes with every Call Manager license, and Avaya Media Servers ship with a Web-enabled version of Tripwire installed and preconfigured.

The downside to HIDS use is that clever attackers who compromise a host can attack and subvert host-based HIDSs as well. HIDS can not prevent DoS attacks. Most significantly, a host-based IDS consumes processing time, storage, memory, and other resources on the hosts where such systems operate. HIDSs that operate in a client-server mode (most of them) can also add to network traffic congestion.

# Logging

Interestingly, when discussing system logging, tired metaphors seem most apt. System log information “is a goldmine” of useful information, but searching through these data is “like trying to find a needle in a haystack.” Tired metaphor or not, the preceding statement is true. Time-stamped logs generated by servers, gateways, firewalls, proxies, routers, and switches often contain invaluable security-related information, but system administrators are normally so overwhelmed with other maintenance and configuration chores that analyses of these logs is disregarded. The key to successful log analysis is to adopt the proper tools for your environment to automatically parse, visualize, and report summarized log data. For example, many organizations utilize MRTG (Multi Router Traffic Grapher—<http://people.ee.ethz.ch/~oetiker/webtools/mrtg/>) to visualize router and switch SNMP network data.

## Syslog

In its most simplistic terms, the syslog protocol provides a transport to allow a machine to send event notification messages across IP networks to event message collectors—also known as syslog servers. Syslog is an odd protocol in that it was implemented on many platforms before the protocol was ratified by the IEEE in RFC3164. Rather than begin by defining the protocol, RFC3164 starts with “*This document describes the observed behavior of the syslog protocol.*”

Syslog messages use UDP/514 for transport, increasing the possibility of losing packets and never noticing, and also making it easy for anybody to forge fake packets, either to insert log events, or to flood the server. Syslog, at this time, does not provide for encryption, so the messages are sent in the clear and can be sniffed by anyone on the wire. Recently, a proposed draft has been submitted that describes a mechanism to add origin authentication, message integrity, replay-resistance, message sequencing, and detection of missing syslog messages, but this is not commonly implemented. Several of the popular syslogd replacements (including syslog-ng) can use TCP for reliable delivery, and some add a checksum and/or cryptographic signature to each log event.

Syslog is native on most UNIX platforms, but is not available natively on Microsoft Windows. The most common Windows syslog daemon is Kiwi Syslog ([www.kiwisyslog.com](http://www.kiwisyslog.com)).

Syslog messages (ASCII-based) may be sent to local logs, a local console, a remote syslog server, or a remote syslog relay. The syslog facility collects messages and records them normally in log files in `/var/log`. A facility is defined as a subsystem which generates messages. What is recorded and where it is placed depends on the facility configuration file (`syslog.conf`). Syslog also uses severity (or priority to some) to classify log messages by importance. The severity levels, from least to most important, are:

- 0 Emergency: system is unusable
- 1 Alert: action must be taken immediately
- 2 Critical: critical conditions
- 3 Error: error conditions
- 4 Warning: warning conditions
- 5 Notice: normal but significant condition
- 6 Informational: informational messages
- 7 Debug: debug-level messages

There is also a severity of none. A severity of none indicates that all messages should be discarded. Entries in `syslog.conf` indicate how messages from each facility at the various severity levels should be handled.

Here is a small section of a BSD `syslog` file:

```
Jan  9 14:46:50 ns1 /kernel: usb1: <VIA 83C572 USB controller> on
uhcil
Jan  9 14:46:50 ns1 /kernel: usb1: USB revision 1.0
Jan  9 14:46:50 ns1 /kernel: uhub1: VIA UHCI root hub, class 9/0, rev
1.00/1.00, addr 1
Jan  9 14:46:50 ns1 /kernel: uhub1: 2 ports with 2 removable, self
powered
Jan  9 14:47:00 ns1 login: ROOT LOGIN (root) ON ttyv0
Jan  9 14:47:18 ns1 /kernel: Connection attempt to UDP
192.168.20.20:162 from 192.168.20.1:24343
Jan  9 14:47:19 ns1 /kernel: Connection attempt to TCP
127.0.0.1:16001 from 127.0.0.1:1024
```

Note that the messages are ASCII based and are composed of three major space-delimited fields: the time and date stamp, the hostname and facility, and a text-based message.

Configuration of `syslog` and `syslog` remote logging is trivial. Much more difficult than generating appropriate `syslog` messages is defining processes that determine how the logs are parsed, who is responsible for parsing, and what type of log entries result in actionable alerts.

In a VoIP environment, IP phones may generate `syslog` messages and servers almost certainly will. These messages should be sent to a centralized server where they are automatically parsed, and where reports are generated at least on a daily basis. These logs are a valuable and often ignored source of both intrusion detection events and system performance messages. For example, `syslog` can be configured to report failed logon attempts, `sudo` (a command that attempts to change a restricted user's permissions to system level or root privilege) attempts, or any action that interacts with the PAM subsystem (Pluggable Authentication Module—an authentication framework). Any message that refers to one of these events may indicate that an intrusion has occurred.

# SNMP

The Simple Network Management Protocol (SNMP) is an application layer protocol that facilitates the exchange of management information between network devices. SNMP messages are encoded as ASN.1 binary using BER encoding, and run over UDP/161 and UDP/162. SNMP enables network administrators to manage network performance and to find and solve network problems. Three versions of SNMP exist: SNMP version 1 (SNMPv1), SNMP version 2 (SNMPv2), and SNMP version 3 (SNMPv3). SNMPv1 and SNMPv2 have a number of features in common, but SNMPv2 offers enhancements, such as additional protocol operations. Neither version provides for any authentication or encryption. SNMPv3 includes, among other things, a model for access control and security as well as for a new architecture. SNMPv3 has yet to attain wide acceptance; thus, SNMPv1 and SNMPv2 still predominate.

An SNMP network normally consists of three key components: managed devices, agents, and network-management systems (NMSs). A managed device is a network node that contains an SNMP agent. Almost every networked device functions as a managed device. An agent is a network-management software module that resides in a managed device. An agent has local knowledge of management information and translates that information into a form compatible with SNMP. An NMS executes applications that monitor and control managed devices. NMSs provide the bulk of the processing and memory resources required for network management. Applications such as HP Openview or Tivoli are examples of NMSs.

Managed devices are monitored and controlled using three basic SNMP commands: *read*, *write*, and *trap*. These commands are defined as follows:

- The *read* command is used by an NMS to monitor managed devices.
- The *write* command is used by an NMS to control managed devices.
- The *trap* command is used by managed devices to asynchronously report events to the NMS.

Additionally, NMS and other applications (such as GetIF; see [www.wtcs.org/snmp4tpc/getif.htm](http://www.wtcs.org/snmp4tpc/getif.htm)) can read and display the Management Information Base (MIB). A MIB is a (sometimes vendor-supplied) collection of information about the managed device that is organized hierarchically. The MIB contains fields that list all of the data the managed device can make available to the NMS.

SNMP community strings and some device configuration data are often among the first findings in penetration tests or vulnerability assessments. Most administrators forget about this threat or simply ignore it.

The best method for securing SNMP today is to turn it off. In VoIP networks, most IP-enabled telephones use SNMPv1 and SNMPv2 for configuration and performance moni-

toring. Thus, it is often impossible to disable this service. If you must run SNMP over your internal networks, then adopt the following practices:

- Immediately change the default read/write community strings
  1. Do not use the default “public” or “private” string.
  2. Do not use a string that would be easy to guess, such as the company’s name or phone number.
  3. Do not use a text-only string; use an alphanumeric string (both text and numerals).
  4. Use both uppercase and lowercase letters (community strings are case-sensitive).
  5. Use a community string that is at least eight characters long.
- Employ ingress and egress filtering at the nearest network border, or limit SNMP to specific management and configuration VLANs.
- Allow SNMP traffic to only a few authorized internal hosts. Only a few network management systems need to initiate SNMP request messages. Thus, administrators can configure SNMP agents to prohibit request messages from unauthorized hosts.

## What Is a Penetration/Vulnerability Test?

These tests or pseudo-attacks are conducted by an objective evaluation team and emulate an attack on one or more computer systems to discover ways to breach the system’s security controls, to obtain sensitive information, to obtain unauthorized services, or to simulate damage to the system by denying service to legitimate users. Security testing comprises a detailed inventory of network assets and a set of controlled attacks intended to find vulnerabilities in those network assets. The words attack and test are used to mean the same in the context of a security assessment.

Penetration tests (pen-tests) usually refer to tests against perimeter defenses, while vulnerability testing refers to tests against specific systems (host, applications, or networks). External assessments can be loosely defined as testing that is launched from outside the perimeter of the private network. This kind of testing emulates the threat from hackers and other external parties and is often concerned with breaching firewalls and other forms of perimeter security. On the other hand, for vulnerability testing the analyst is located somewhere within the perimeter of the private network and emulates the threat experienced from internal staff, consultants, disgruntled employees, or, in the event of unauthorized physical access or a compromise of the perimeter security, a hacker. The general rule of thumb is that internal threats comprise more than 60 percent of the total threat portfolio.

Testing can consist of something as simple as an Nmap or Nessus scan, or it can be as detailed as tests against a multitiered business application architecture requiring months of

code review and application testing. The ground rules for testing define successful completion. Testing is successfully concluded when:

- A defined number of flaws are found.
- A set level of penetration time has transpired.
- A dummy target object is accessed by unauthorized means.
- The security policy is violated sufficiently.
- Money and resources are exhausted.
- Internal resources are accessed.
- Transaction data is captured.
- A particular program or transaction is executed.
- Access is gained to any user account.
- Access is gained to a root/administrative account.
- Network management systems are subverted.
- The ability to remotely control resources is demonstrated.

## Methodology

The team should thoroughly investigate target systems and networks in a structured manner, documenting their findings as they proceed. The goal is to attempt to identify all the *significant* vulnerabilities on the network—including their location and implications—and provide recommendations for securing the affected systems. Testing results in a comprehensive, operational review or “snapshot” of the state of the network. Testing should include an analysis of the external network from the perspective of an outside hacker, and/or a review of the internal network from the perspective of a disgruntled employee or contractor.

## Discovery

The discovery process takes advantage of publicly available information that relates to your organization. Internet search engines, Whois databases, network registrars, DNS servers, and company Web sites are all sources of information. This phase can yield data that your organization might wish to protect. Table 7.1 lists a number of recommended tools used during the discovery phase. All of these are either native UNIX tools or are freeware, with the exception of WSPingPro.

**Table 7.1** Common Security Testing Tools

Discovery	Scanning	Vulnerability Assessment
Whois	Hping	tcpdump
SamSpade	Nmap	Voipong
WSPingPro	LDAPMiner	Wepcrack
SuperScan	scanrand	Getlf
dig	NetStumbler	Nessus
nslookup	Kismet	Retina
ping	Nikto	Brute
tracert	PSTools	WinFingerprint
TCPTraceroute	WSPingPro	Lophtrcrack5
	SQLPing 2	ISS Internet Scanner
	ToneLoc	SnagIT
	Dsniff	@stake Proxy
	SuperScan	Ethereal
		Ettercap
		Amap
		John the Ripper
		Netcat

## Scanning

Scanning or fingerprinting utilizes a variety of automated, non-intrusive scans. Nmap is a recommended tool for this step. Foundstone's SuperScan is another useful tool at this stage. Results of these scans should be constantly monitored in order to minimize bandwidth issues and to ensure that the scanning process does not result in loss of network connectivity for any networked devices. If any device fails under this type of scanning, that is a finding in itself.

It may be useful to emulate specific IP phones when testing VoIP gateways. For testing H.323 gateways or gatekeepers, the OpenH323 project offers OpenPhone, which has a GUI for Windows clients and command-line options for Linux distributions.

For testing SIP proxies, registrars, and gateways, many sites (such as sipXphone and YATE) have open-source SIP clients that are quite configurable. SJ Labs' SJphone softphone ([www.softjoys.com](http://www.softjoys.com)) is also useful for testing in a VoIP environment, and is free for 30 days. SIPsak and SIPbomber are also useful SIP proxy testing tools. Callflow (<http://callflow.sourceforge.net/>) can be very useful for examining and understanding the alterations in calling message sequences that can result when performing SIP testing.

As an indication of the maturity of this field, SiVuS ([www.vopsecurity.org](http://www.vopsecurity.org)) has been released. SiVuS is the first publicly available vulnerability scanner for VoIP networks that use the SIP protocol.

## Vulnerability Assessment

Vulnerability assessment, one of the most important phases of penetration testing, occurs when your team maps the profile of the environment to publicly known or, in some cases, unknown vulnerabilities. Tools such as Nessus, Retina, and ISS Internet Scanner are all good choices at this stage. An excellent listing of the top 75 security tools can be found at [www.insecure.org/tools.html](http://www.insecure.org/tools.html).

When you are vulnerability testing VoIP networks, it is not necessary to test every IP phone. Because of the oftentimes, sheer number of IP phones, vulnerability testing has the potential to generate enough network traffic that voice quality is negatively affected. Testing one particular IP phone per vendor is often adequate since configurations should be functionally identical.

In most VoIP environments, it is possible to identify IP phones by their SNMP signature. Calling the IP phone directly—thus, bypassing any gateways or gatekeepers—can sometimes yield interesting information.

## Exploitation

The exploitation phase begins once the target system's vulnerabilities are mapped. The testers will attempt to gain privileged access to a target system by exploiting the identified vulnerabilities. This may take the form of running an exploit tool such as `scalp.c` or `iis5hack.c`, or launching a password guessing attack using THC-Hydra, a network authentication cracker. (An excellent resource of known/default accounts and associated passwords is located at [www.phenoelit.de/dpl/dpl.html](http://www.phenoelit.de/dpl/dpl.html).)

## Reporting

Throughout the testing, the team should maintain a detailed journal of activities to account for effects and results of the testing procedures. This record will serve to distinguish the test team's activities from any other anomalies that occur during the course of the penetration test. Some techniques for capturing these data include the use of echo and logging. When appropriate, the use of screen captures may be an option.

- Detailed results of the testing performed
- What the results indicate
- Recommendations on types of corrective actions



One internal measure that can be used to quantify a particular vulnerability is a “Threat Index.” This index is based upon two independent metrics: perceived risk (Table 7.2) and an estimated frequency (Table 7.3). The subsequent two-part identifier is formed by combining these two results, and is placed in the 3X3 matrix. The Threat Index (TI) has several purposes: First, it is used to rapidly prioritize a discovered vulnerability. Severe or high TIs (see Table 7.4) require immediate attention, and may also require more in-depth analysis by testers. Second, the TI can be used to rapidly code particular vulnerabilities. For example, if a newly discovered vulnerability is ranked with a TI of H1, all members of the team immediately understand that this is a severe problem that requires immediate action, while a TI of L3 indicates an insignificant issue.

**Table 7.2 Risk Categories**

High Risk (H)	Loss of critical proprietary information, system disruption, or severe environmental damage
Medium Risk (M)	Loss of proprietary information, severe occupational illness, or major system or environmental damage
Low Risk (L)	Minor system or environmental damage

**Table 7.3 Modified Department of Defense Frequency Categories**

Frequent (1)	Likely repeated occurrences
Occasional (2)	Possibility of repeated occurrences
Improbable (3)	Practically impossible

**Table 7.4 Threat Index**

	High Risk (H)	Medium Risk (M)	Low Risk (L)
Frequent (1)	H1	M1	L1
Occasional (2)	H2	M2	L2
Improbable (3)	H3	M3	L3

Your organization can apply these criteria in any way you see fit. The point is to determine as objectively as possible a method to prioritize threats against your infrastructure. You may even use different rankings based upon different portions of the network infrastructure—for example, when testing data services, threats to data integrity may be particularly important, compared to voice services, where threats that negatively impact availability may be critical.

In Table 7.4, any vulnerability with a threat index of H1, H2, M1, M2, and L1 requires immediate attention.

## Summary

An appropriate firewall policy can minimize the exposure of your internal networks. However, attackers are evolving their attacks and network subversion methods. These techniques include e-mail-based Trojan horses, stealth scanning techniques, and attacks which bypass firewall policies by tunneling access over allowed protocols such as ICMP, HTTP, or DNS. Attackers are also getting better at using the ever-growing list of application vulnerabilities to compromise the few services that are allowed through a firewall.

Firewalls and Access Control Lists are requisite security controls in any enterprise, but they are not sufficient in contemporary networks. Active monitoring of the network and attached devices provides not only one or more additional layers of defense, but also supplies data that may have a forensic utility. Active monitoring consists of the following types of activities: network monitoring, network intrusion detection, host-based intrusion detection, syslog, and SNMP logging. Penetration and vulnerability testing monitors and validates existing security controls.

On enterprise networks, network monitoring is typically managed by a comprehensive tool suite such as OpenView. Traffic patterns and quantities, and device state are common measurements. These tools supply data that can be useful to security administrators, particularly when combined with the results of recent penetration/vulnerability tests or with NIDS/HIDS data. Unfortunately, the correlation of these data is difficult even when using tools such as SMARTS (a root-cause correlation engine), because of the overwhelming amount of data that must be organized.

NIDS and HIDS are complementary intrusion detection technologies. NIDS monitors the network for malicious or unauthorized traffic and HIDS monitors critical servers for changes to significant files and directories. Both relay event data to a central management console for logging and visualization. Most current NIDSs use a combination of signature (pattern or regex) and anomaly-based detection. Both of these methods have benefits and drawbacks. Signature-based detection is quick, effective, and popular, but it won't catch attacks that don't have signatures. Anomaly detection is theoretically a better method for detecting attacks, but suffers from the basic problem that it is difficult to define "normal" traffic on a network.

Although functionally dissimilar, SNMP and syslog both provide transport for event messages over the network from agents or endpoints to a centralized information repository. SNMP is a highly structured, binary-formatted message type, while syslog messages are ASCII-based and relatively arbitrary within the confines of three defined fields. Neither protocol is encrypted. Thus, SNMP and syslog messages should always be limited to a constrained management network.

Penetration and vulnerability testing is both art and science. These assessments are only as good as the people and tools used to perform them. In today's environment most types of penetration/vulnerability assessment have been commoditized due to the ready availability of scanning and vulnerability assessment tools.

Some tools, such as Nessus (which until recently was open source), make it possible for naïve administrators to perform at least baseline vulnerability scans on their networks. In this case, we recommend that an experienced security analyst be brought in to analyze the data since all of the vulnerability scanners report various false alarms. One important note is that the results of a test only reflect the security status during the testing period. Even minor administrative and architectural changes to the environment performed only moments after a penetration test can alter the system's security profile.

## Logically Segregate Network Traffic

### Solutions in this chapter:

- VLANs
- QoS and Traffic Shaping
- NAT and IP Addressing
- Firewalls
- Access Control Lists

# Introduction

One of the principal advantages of converging voice and data is to save money and to simplify administration and management by running both types of traffic over the same physical infrastructure. With this in mind, it is ironic that most of the engineering effort expended during the VoIP architecture design phase focuses on logically separating this same voice and data traffic.

Packetized voice is indistinguishable from any other packet data at Layers 2 and 3, and thus is subject to the same networking and security risks that plague data-only networks. The general idea that motivates the logical separation of data from voice is the expectation that network events such as broadcast storms and congestion, and security-related phenomena such as worms and DoS attacks, that affect one network will not impact the other. This is the principal consequence of compartmentalization.

In practice, system and security administrators have a number of options to realize this logical division. Packet headers can be manipulated in order to separate datagrams and datagrams at Layer 2, to provide certain classes of packets with preferential treatment or more bandwidth; and to alter source and destination IP addresses. Firewalls (particularly VoIP-aware firewalls), application layer gateways (ALGs), routers, and switches are inserted in the datapath to monitor and control traffic streams. Many devices now support robust access control lists (ACLs) that are used to fine-tune network and application access. Encryption is used often to ensure data and signal channel authentication, integrity, and privacy, but the encryption process results in subtle and not-so-subtle interactions with the methods that manipulate packet headers.

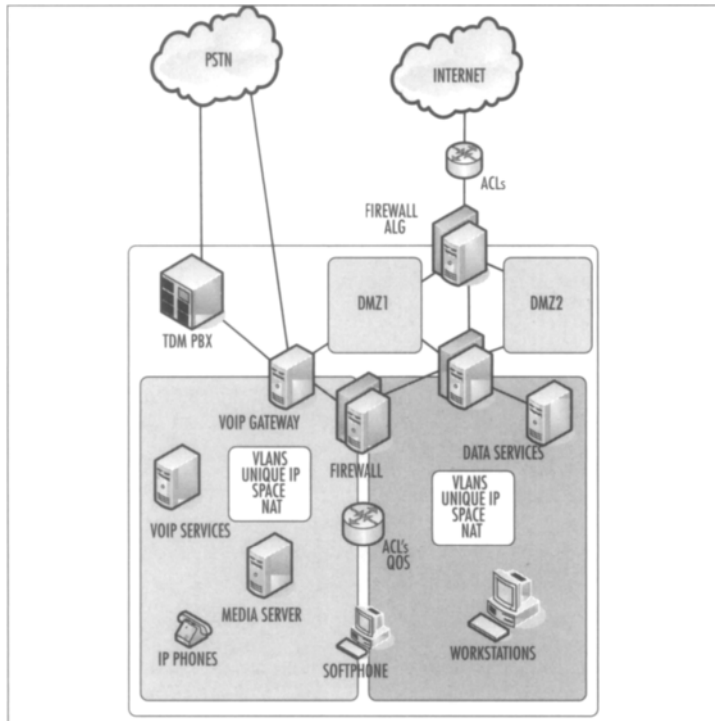
Maintaining and securing contemporary data and voice networks is complex stuff—something not recommended for naïve system administrators. Gone are the days when networks could be pieced together in an ad hoc fashion in order to support gopher, e-mail, and ftp. Modern VoIP/data networks must be designed to support a sometimes bewildering array of applications—all with their own unique service requirements and SLAs—in an open, yet secure environment.

To this end, in this chapter we look at the methods used to segregate voice and data into logically isolated networks that run over the same physical infrastructure. Figure 8.1 shows the components of this architecture. The primary elements of the security architecture are VLANs, QoS scheduling, firewalls, NAT and intelligent IP address space management, and ACLs. Encryption also plays a role in this. We will look at each of these technologies in more detail in the following sections.

Figure 8.1 is a diagram of a VoIP/data reference network that illustrates the major security components involved in logical segregation of network traffic types. At the border between the Internet and the internal network, firewalls, ALGs, and router-based ACLs provide the first line of defense or security layer against illicit traffic and attackers. Within the internal domains, VLANs, QoS, private IP addresses, and NAT segregate VoIP traffic from

other data network traffic, and VoIP-aware firewalls and router-based ACLs manage traffic between the two domains. Softphones may or may not span both domains depending upon an organization's sensitivity to risk.

**Figure 8.1** Converged Reference Network



## VLANs

Logical separation of voice and data traffic via VLANs is recommended in order to prevent data network problems from affecting voice traffic and vice versa. In a switched network environment, VLANs create a logical segmentation of broadcast or collision domains that can span multiple physical network segments. VLANs remove the need to organize and manage PCs or softphones based upon physical location, and can be used to arrange endpoints based upon function, class of service, class of user, connection speed, or other criteria. The separation of broadcast domains reduces traffic to the balance of the network. Effective bandwidth is increased due to the elimination of latency from router links. Additional security is realized if access to VLAN hosts is limited to only hosts on specific VLANs and not those that originate from other subnets beyond the router.

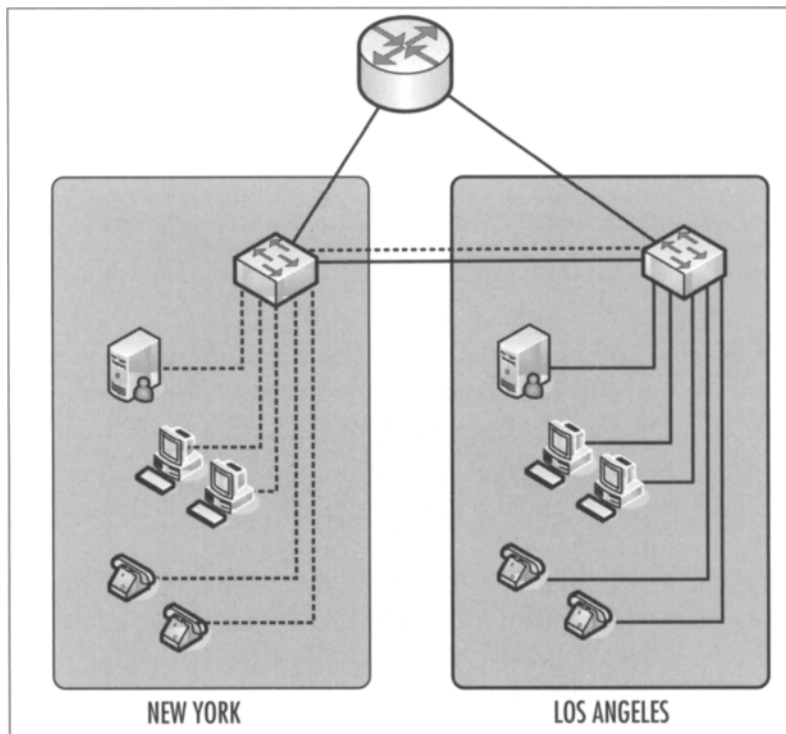
VLANs, or virtual LANs, can be thought of as logically segmented networks mapped onto physical hardware. One or more VLANs can coexist on a single physical switch. The predominant VLAN flavor is IEEE 802.1Q, as defined by the IEEE. Prior to the introduc-

tion of 802.1q, Cisco's ISL (Inter-Switch Link) was one of several proprietary VLAN protocols. ISL is now deprecated in favor of 802.1q. VLANs operate at layer 2 of the OSI model. However, a VLAN often is configured to map directly to an IP network or subnet, which gives the appearance that it is involved at layer 3.

VLANs can be configured in various ways—by protocol (IP or IPX, for example) or based on MAC address, subnet, or physical port. They can be static, dynamic, or port-centric. Mechanistically, VLANs are formed by either frame-tagging or frame-filtering. Frame-tagging, the more common mechanism, requires adding and removing a unique, 2-byte L2 frame identifier so that switches may appropriately send and receive their cognate VLAN traffic. Frame-filtering relies upon the participating switches building and communicating a filtering database in order to forward traffic to its correct VLAN.

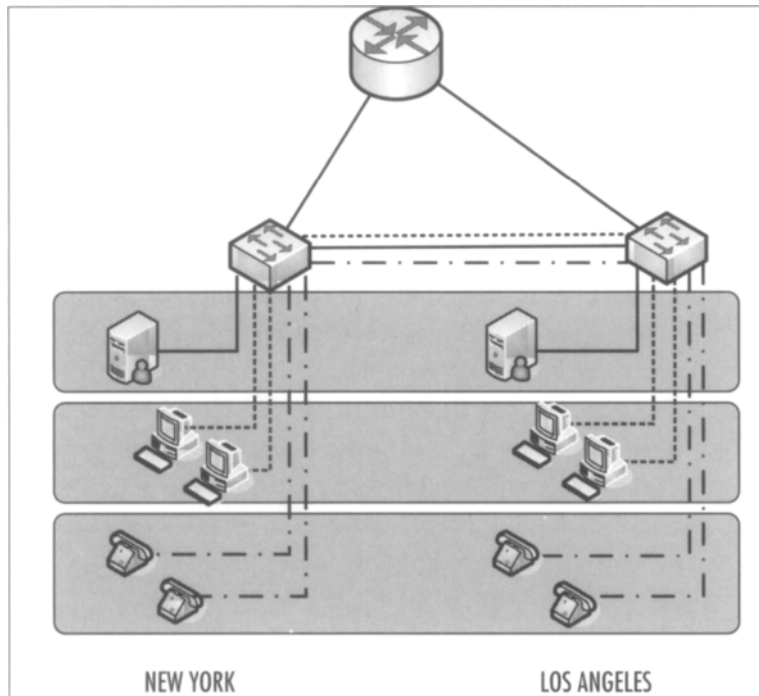
In Figure 8.2, dotted lines represent VLAN 2 and solid lines represent VLAN 10. The presence of the two lines that form a trunk between the top level switches should not be taken to indicate that there are two physical connections. Servers and workstations are logically isolated based upon their physical location. If a New York workstation requires the services of a Los Angeles server, then those data are routed between the top level switches.

**Figure 8.2** Location-Based VLANs



In Figure 8.3, dotted lines represent VLAN 2, solid lines represent VLAN 10, and dash-dot lines represent VLAN 100. The presence of the three lines that form a trunk between the top level switches should not be taken to indicate that there are three physical connections. In the network shown in Figure 8.3, broadcast traffic in the telephone subnet will not be seen by hosts in the workstation subnet.

**Figure 8.3** Function-Based VLANs



VLANs provide some security and create smaller broadcast domains by creating logically separated subnets. Broadcasts are a common, sometimes noisy phenomenon in data networks. Creating a separate VLAN for voice reduces the amount of broadcast traffic (and unicast traffic on a shared LAN) the telephone will receive. Separate VLANs can result in more effective bandwidth utilization, and reduce the processor burden on IP telephones and PCs by freeing them from having to analyze irrelevant broadcast packets. Management traffic can be segregated on a management VLAN so that SNMP and syslog traffic do not interfere with data traffic. This also has the benefit of adding a layer of security to the management network. Additionally, VLANs can be used in conjunction with various quality of service mechanisms (see next section) to further isolate and prioritize voice traffic.

The consequences of DoS attacks can be mitigated by logically separating voice and data segments into discrete VLANs. Segregation of network traffic requires that IP traffic pass through a Layer 3 device, thereby enabling the traffic to be inspected at the ACL level. VLAN



segregation forces any DoS packets through the ACLs on the layer 3 device. The use of packet filtering or stateful firewall inspection at these junctions also is recommended. As a side note, user authentication prior to the user's accessing the telephony device also will reduce the possibility of internal DoS attacks.

## VLAN Security

VLAN and layer 2 security is a complex topic, partially because of the uneven support by switch vendors for appropriate datalink safeguards and because many of the exploitable vulnerabilities arise due to misconfiguration of available safeguards. The single most important rule with regard to this topic is to absolutely ensure that unauthorized individuals do not have access to the switch console. Additionally, terminal access to the console should either require strong authentication (RADIUS or AAA) and be restricted to a small set of management PCs, or should be eliminated altogether.

VLAN function depends upon the presence or absence of tag information. If the integrity of the tag information is assured, then the logical security afforded by VLANs is as legitimate as physical security. The key is to certify that tag information originates from the appropriate hosts and is unchanged in transit. A number of controls exist to verify this information such as ARP inspection, DHCP spoofing, VACLs (VLAN ACLs), private and dynamic VLANs, port security, and 802.1X admission controls, but implementation of these is vendor specific and beyond the scope of this section. Additionally, the IEEE 802.1 Working Group has established drafts, particularly, 802.1aj, that decompose security when two related MACs are in a relay configuration.

## VLANs and Softphones

Softphones present a security challenge in a VoIP environment, particularly if VLANs are employed as a major security control. Several popular softphones (such as X-Lite) store credentials unencrypted in the Window's registry even after uninstallation of the program. Many softphones contain advertising software that attempts to "phone home" with private user information. Host-based IDS or firewall applications have limited use in this situation because softphones require that PC-based firewalls open a number of high UDP ports as part of the media stream transaction. Additionally, any special permissions that the VoIP application has within the host-based firewall rule set will apply to all applications on that desktop (e.g., peer-to-peer software may use SIP for bypassing security policy prohibitions).

## Tools & Traps...

### Watch What You Plug into That Phone

We recently observed a situation where plugging in a single phone brought down a substantial piece of the network. The IP phone had two RJ45 plugs in the back and a technician mistakenly plugged both of these into a nearby access switch running STP. No one realized (until later) that the IP phone failed to bridge spanning tree BPDUs. The resulting broadcast storm took out several core switches. This problem was solved temporarily with a bit of glue and some RJ45 plugs.

The most important rule for securing softphones is to harden the underlying operating system. Malware that affects any other application software on the PC can also interfere with voice communications. The flip-side is also true—malware that affects the VoIP software will affect all other applications on the PC and the data services available to that PC (a separate VoIP phone would not require access to file services, databases, etc.). Softphones that contain any type of advertising software must be banned in a secure environment. Softphone installation targets should be tested before deployment and those that do not encrypt user credentials should be prohibited.

Because PC workstations are necessarily on the data network, using a softphone system conflicts with the requirement to separate voice and data networks since the principle of logically separating voice and data networks is defeated because the PC must reside in both domains. One solution to this is dual home workstations—dedicate one NIC to the data domain and one NIC to the voice domain. This arrangement still allows for possible routing of information between domains via a workstation. Cisco recently has introduced a Certificate Trust List (CTL) that contains among other information, the IP addresses of trusted VoIP peers. However, this feature is available only in selected IP phones and requires, for the most part, setup and maintenance of a complex certificate infrastructure. Additionally, unless complex host firewall rules are implemented, non-VoIP related data can enter the voice domain from workstations. Frankly, there is no single good security solution to the issue of softphones on workstations in split voice/data environments. In a highly secure environment, your best choice is to ban them via policy and monitor for illicit usage via IDS or IPS.

## QoS and Traffic Shaping

VoIP has strict performance requirements. The factors that affect the quality of data transmission are different from those affecting the quality of voice transmission. For example, data generally is not affected by small delays. The quality of voice transmissions, on the other hand, is lowered by relatively small amounts of delay. VoIP call quality depends on three network factors, as mentioned earlier:

- **Latency** The time it takes for a voice transmission (or any transmission) to travel from source to destination is increased as packets traverse each security node. Primary latency-producing processes are firewall/NAT traversal, negotiation of long ACLs, and traffic encryption/decryption.
- **Jitter (erratic packet delays)** Jitter may be increased, because in many circumstances, jitter is a function of hop count.
- **Packet loss** The number of non-QoS-aware routers and firewalls that ignore or fail to properly process Type of Service (ToS) fields in the IP header can influence packet loss.

In the absence of QoS or Traffic shaping, data networks operate on a best-effort delivery basis, which means that all data traffic has equal priority and an equal chance of being delivered in a prompt manner. However, when network congestion occurs, all data traffic has an equal chance of being dropped and/or delayed. When voice data is introduced into a network, it becomes critical that priority is given to the voice packets to insure the expected quality of voice calls. The mechanisms used to accomplish this are generically referred to as traffic shaping.

Traffic shaping is an attempt to organize network traffic in order to optimize or guarantee performance and/or bandwidth. Traffic shaping relies upon concepts such as classification, queue disciplines, scheduling, congestion management, quality of service (QoS), class of service (CoS), and fairness.

Common CoS models include the Differentiated Services Code Point (DiffServ or DSCP, defined in RFC 2474 and others) and IEEE 802.1Q/p. DSCP specifies that each packet is classified upon entry into the network. The classification is carried in the IP packet header, using 6 bits from the deprecated IP type-of-service (ToS) field to carry the classification (code point) information, which ranges from 0 through 63. Generally, the higher number equates to higher priority.

802.1Q defines the open standards for VLAN tagging. Twelve of the 16 bits within the two Tag Control Information bytes are used to tag each frame with a VLAN identification number. 802.1p uses three of the remaining bits (the User Priority bits) in the 802.1Q header to assign one of eight different classes of service (0 = low priority; 8 = high priority).

Quality of Service involves giving preferential treatment of particular classes or flows of traffic primarily by manipulating queues and scheduling. A service quality is then negotiated.

Examples of QoS are CBWFQ (Class Based Weighted Fair Queuing), RSVP (RESERVATION Protocol—RFC 2205), MPLS, (Multi Protocol Label Switching—RFC 1117 and others). CoS, or tagging, is ineffective in the absence of QoS because it can only mark data. QoS relies on those tags or filters to give priority to data streams.

Networks with periods of congestion can still provide excellent voice quality when using an appropriate QoS/CoS policy. The recommendation for switched networks is to use IEEE 802.1p/Q. The recommendation for routed networks is to use DiffServ Code Points (DSCP). The recommendation for mixed networks is to use both.

The main purpose of these technologies is to ensure that application performance remains satisfactory regardless of network conditions. In general, they all work by categorizing traffic into discrete subsets that are processed with different priorities. For this reason, QoS techniques may be useful in protecting VoIP networks from a significant security threat—Denial of Service. A number of authors have shown that some VoIP architecture components including IP telephones, SIP proxies, and H.323 gateways may freeze and crash when attempting to process a high rate of packet traffic. QoS can provide some security for these devices during DoS attack either by prioritizing unauthorized data low and/or by prioritizing VoIP high. This measure (security layer) will mitigate the consequences of a DoS attack on applications that share the same physical bandwidth.

The downside of all this is that traffic shaping is, at times, a stew of poorly interoperable technologies and techniques. This ad hoc nature makes a true end-to-end QoS strategy sometimes difficult to implement. If possible, provide enough bandwidth resources to meet the expected peak demands with a substantial safety margin. Note also that the implementation of some security measures can degrade quality of service.

These security-related complications are bulleted at the beginning of this section, and range from interruption or prevention of call setup by misconfigured firewall rules to encryption-produced latency and delay variation (jitter). There is no single best method at present to optimize traffic shaping on VoIP networks without taking into account the relationship of these technologies with the security measures implemented within your environment.

## NAT and IP Addressing

Network Address Translation (NAT) is a method for rewriting the source and/or destination addresses of IP packets as they pass through a NAT device, which is often a router or firewall that separates two realms or domains on the Internet. NAT was first officially proposed (RFC1631) in 1994 as a temporary solution to the problems of IP address space depletion and the rapidly increasing size of route tables. Addresses, at that time, were divided into two classes: local and global addresses. Today we normally refer to these addresses as either private or public, and the private IP space often is referred to as RFC1918 addresses. Per RFC1918,

the Internet Assigned Numbers Authority (IANA) reserved three blocks of the IP address space for private internets:

- 10.0.0.0–10.255.255.255 (10/8 prefix)
- 172.16.0.0–172.31.255.255 (172.16/12 prefix)
- 192.168.0.0–192.168.255.255 (192.168/16 prefix)

NAT commonly is used to enable multiple hosts on private networks to access the Internet using a single public (Internet routable) IP address. Note that although NAT most commonly is used to map IP addresses from internal private IP space to the public IP space, NAT can be used to map between any two IP address domains. Additionally, NAT provides a security function by segregating (hiding) private hosts from the publicly routed Internet. This short-term kludge has had an enormous impact on the day-to-day functioning of the Internet, and has special relevance to system administrators who are charged with securely transporting VoIP packet data across network boundaries.

## How Does NAT Work?

To a system on the Internet, a NAT device appears to be the source/destination for all traffic originating from behind the NAT device. Hosts behind a NAT device do not have true end-to-end Internet connectivity and cannot directly participate in Internet protocols that require initiation of TCP connections from outside the NAT device, or protocols that split signaling and media into separate channels.

A NAT device examines and records certain IP header information from each packet within an active IP connection. It uses these connection data to multiplex or demultiplex traffic depending upon the direction of the traffic flows. Multiplexing, in this case, means that two or more traffic streams are combined into a single outbound channel; demultiplexing refers to the process of separating a complex inbound traffic stream into single traffic streams (see Figure 8.4).

NAT devices manipulate a subset of the IP header information. In order to comprehend the sometimes complex interaction of NAT, encryption, and VoIP protocols, you will have to understand the IP header fields and how they are altered during the NAT and encryption processes. It is not necessary for you to understand these concepts if you are concerned only with a NAT device's ability to hide internal network topology from the Internet, but as part of the process of securing VoIP communications, this information is critical. Get to know the header diagrams shown in Figure 8.5. You'll be seeing them frequently.

Figure 8.4 Multiplexing and Demultiplexing

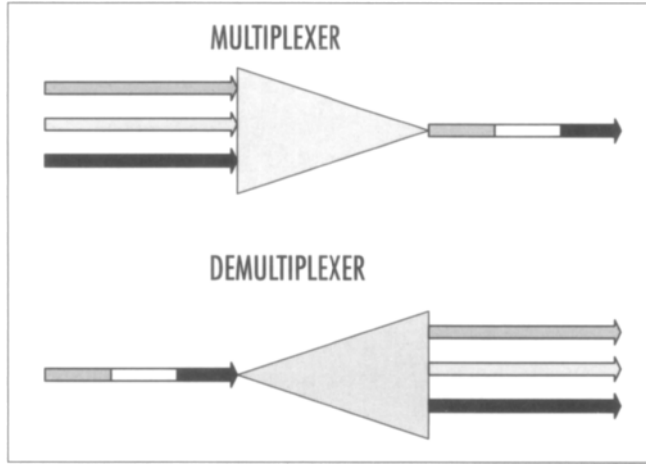


Figure 8.5 IP, TCP, and UDP Headers



Note that the rest of this section applies only to IPv4 packets. IPv6 resolves most of the following issues, but it just hasn't caught on yet. The IP header normally consists of 20 bytes of data. The TCP header also normally consists of 20 bytes of data. An options field exists within each header that allows further bytes to be added, but normally this is not used. The UDP header is 8 bytes in length. Both the TCP and UDP headers reside in the data field of an IP packet. In Figure 8.5, the data field is to the right of the options field for IP and TCP headers and to the right of the CHKSUM field in the case of the UDP header.

NAT devices monitor, record, and alter the source IP address (SIP), destination IP address (DIP), and checksum (CHKSUM) fields within IP headers. NAT also modifies the checksum fields of both TCP and UDP packets since these checksums are computed over a pseudo-header that conceptually consists of the source and destination IP addresses, and the protocol and length fields for TCP. The UDP checksum is calculated over a pseudo-header that consists of the source and destination IP addresses, the UDP header and data. As for ICMP Query packets, no further changes in the ICMP header are required as the checksum in the ICMP header does not include the IP addresses. These checksum fields will prove particularly troubling as we modify VoIP packets by encryption over NAT.

In response to the pseudo-header complexities, RFC1631 suggests that:

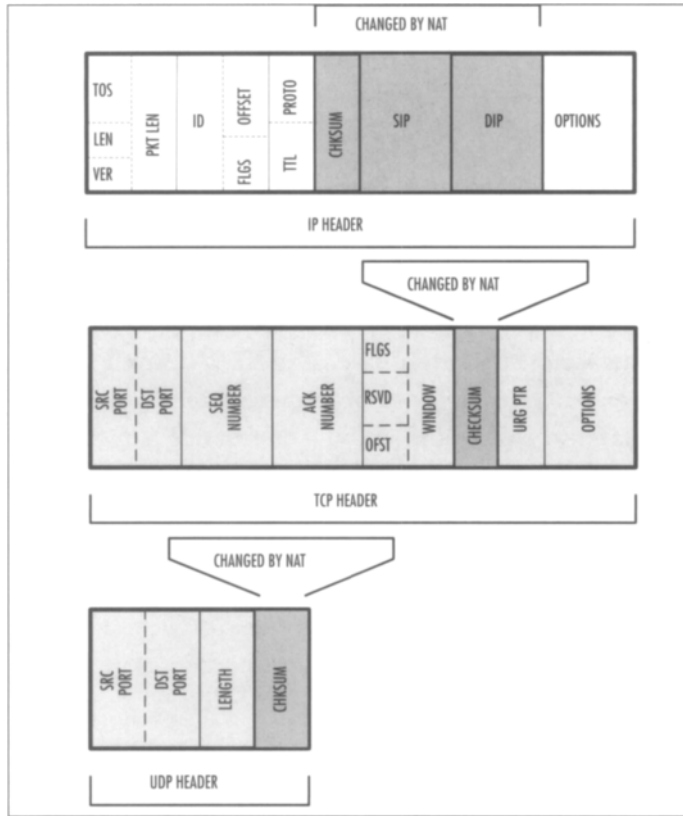
**NAT must also look out for ICMP and FTP and modify the places where the IP address appears. There are undoubtedly other places, where modifications must be done. Hopefully, most such applications will be discovered during experimentation with NAT.**

Though these were bright individuals it seems to me unlikely that they would have imagined that their complex solution would prove to be a major complication to end-to-end application availability on today's contemporary internetworks. Figure 8.6 shows how NAT alters four header fields.

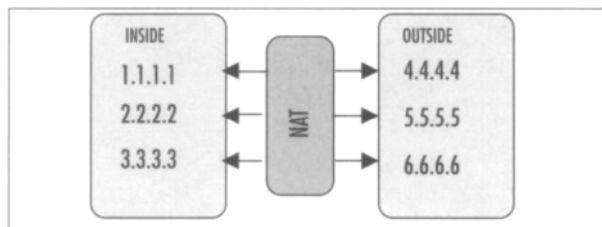
## NAT Has Three Common Modes of Operation

Depending upon networking requirement and topology requirements, NAT is manifested in one of three related modes. Static NAT refers to a one-to-one mapping or correspondence between internal and external IP addresses. In this case, the number of internal IP addresses equals the number of external addresses (see Figure 8.7).

**Figure 8.6** NAT Alters Four Header Fields



**Figure 8.7** Static NAT



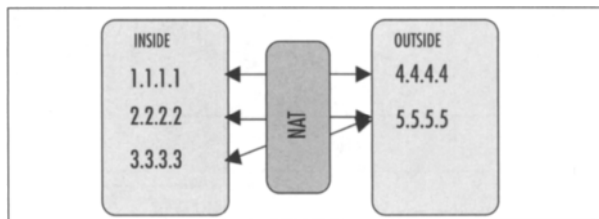
The NAT device maintains a lookup table of internal and external addresses in order to manage translations in a stateless manner. Static NAT has utility in mapping the private internal IP addresses of critical infrastructure servers and network appliances to a unique globally available IP address.

Dynamic NAT in its original form consisted of an outside pool or collection of public IP addresses that were used on a first-come, first-served strategy (see Figure 8.8). Each unique single internal address could be used by any member of the outside pool to communicate with external Internet hosts. Consequently, the size of the outside pool member set limited



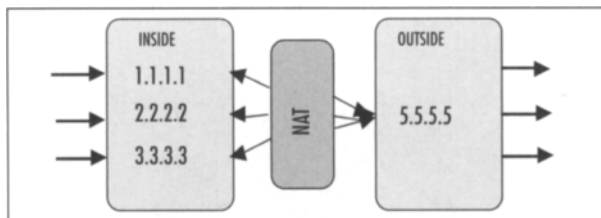
the number of inside users that could connect externally. A built-in timeout mechanism allowed external pool members to be reused.

**Figure 8.8** Dynamic NAT



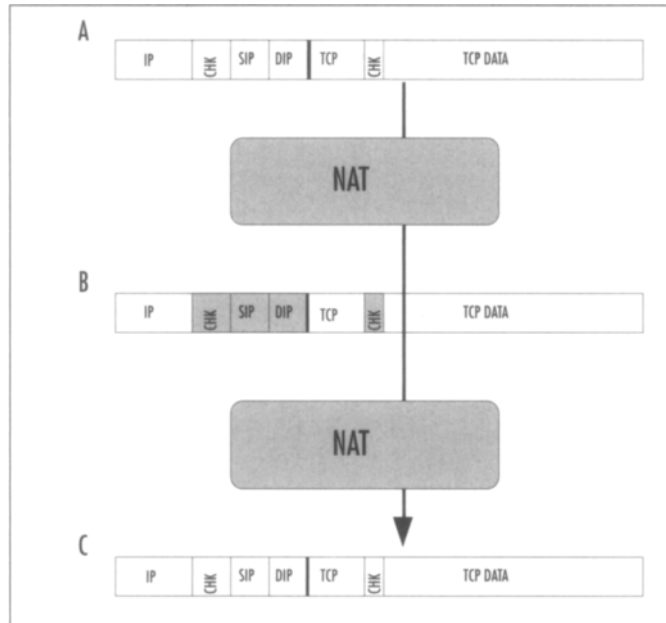
The third and probably most common style of NAT is derived functionally from Dynamic NAT since it reuses a smaller pool or a single external IP address to proxy for all the internal IP addresses. This NAT is known by a number of names, including Network Address Port Translation (NAPT), Port Address Translation (PAT), Full Cone NAT (From the STUN RFC3489), hiding NAT, and masquerading NAT. This type of NAT (we'll call it NAPT to keep things organized) works to preserve state by maintaining a lookup table of source IP, destination IP, source port, and destination port. This 4-tuple is almost always guaranteed to be unique within a given conversation stream. You'll find NAPT operating in almost all home broadband and in most large enterprise networking scenarios. Figure 8.9 shows an example of NAPT.

**Figure 8.9** Network Address Port Translation



So a normal scenario that occurs when moving TCP traffic between two domains running NAT at each edge is shown in Figure 8.10.

In addition to these three NAT modes, STUN (we'll see this later) has defined a three types of NAT that map more or less to these three modes. These are cone NAT, restricted NAT, and symmetric NAT. We'll talk more about these in the section on STUN and TURN.

**Figure 8.10** Normal NAT Process with TCP

Section A of Figure 8.10 shows the TCP/IP packet header prior to NAT. After passing through the first NAT edge device (section B), the four header fields are modified: the three IP header fields—source address, destination address, and checksum—and the TCP header checksum. After passing through the second NAT edge device, the original header fields are regenerated (section C). The same is true for UDP in this situation, except that if the UDP checksum is zero, it will not be altered.

You may naturally ask by now, why is NAT such an issue for VoIP? Well, when we begin to combine NAT and protocols such as H.323 and SIP that partition the signaling and media channels; and, to make things even more interesting, embed IP addresses in the signaling channel, it will be important to understand how, when, and where NAT manipulates these fields. When we add encryption into the mix, NAT adds further complexity to these systems. Additionally, note that NAT stores its address mapping information in binding tables, and that these bindings are only initiated by outbound traffic. NAT breaks the choreography of SIP session flow. Encryption adds further complexity to these systems.

## NAT and Encryption

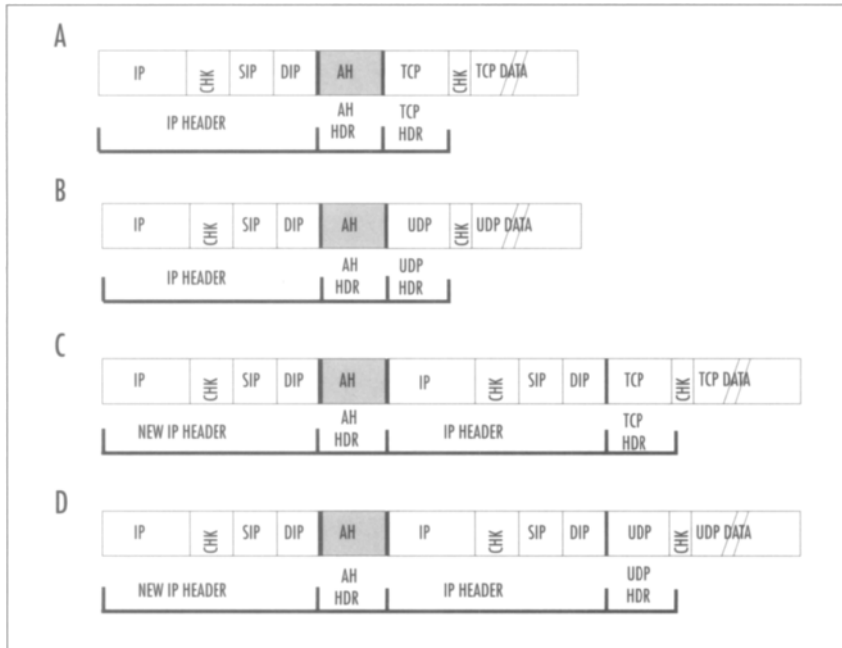
As IPsec VPNs became popular, NAT became an impediment to their initial widespread implementation. I'll use the IPsec model to develop a description of the interactions between NAT and encryption since it is one of the more popular Internet encryption systems and has potential value in VoIP networks. The IP security (IPsec) protocol was defined by the Internet Engineering Task Force (IETF) to provide security for IP networks. IPsec is

a large protocol suite designed to provide the following security services for IP networks: Data Integrity, Authentication, Confidentiality, and Application-transparent Security. IPsec secures packet flows and key transmission. Since we are interested in NAT and encryption, we'll ignore most of the protocol suite including key exchange (IKE), and the various hash and encryption algorithms, and focus instead on the protocols that are used to secure packet flows.

The AH and ESP protocols can operate in two modes: Transport Mode can be visualized simply as a secure connection between two concurring hosts. In Tunnel Mode—more of a “VPN-like” mode—IPsec completely encapsulates the original IP datagram, including the original IP header, within a second IP datagram. ESP and AH normally are implemented independently, though it's possible (but uncommon) to use them both together.

The Authentication Header (AH) and the Encapsulating Security Payload (ESP) are the two main network protocols used by IPsec. The AH provides data origin authentication, message integrity, and protection against replay attacks, but has no provision for privacy—data is not encrypted. The key to the AH authentication process is the inclusion in the AH header of an Integrity Check Value (ICV) —a hash based upon a secret key that is calculated over a subset of the original IP header fields, *including the source and destination IP addresses*. AH guarantees (if implemented correctly) that the data received is identical to the data sent, and asserts the identity of the true sender. AH provides authentication for as much of the IP header as possible, as well as for upper level protocol data. However, some IP header fields (SIP, DIP, TTL, CHKSUM, and optionally, TOS, FLAGS, and OPTIONS) change in transit. The values of such fields usually are not protected by AH. In transport mode, AH is inserted after the IP header and before the upper layer protocol (TCP, UDP, ICMP, etc.) header. In tunnel mode, the AH header precedes the encapsulated IP header. Figure 8.11 shows the AH transport and tunnel modes.

In Figure 8.11, sections A and B show the location of the AH header in transport mode. Sections C and D show the location of the AH header in tunnel mode. The data field in all packets is not to scale (indicated by the double slanted lines). You can see from this figure that tunnel mode AH adds an additional 20 bytes to the length of each packet. None of the fields in this figure are encrypted.

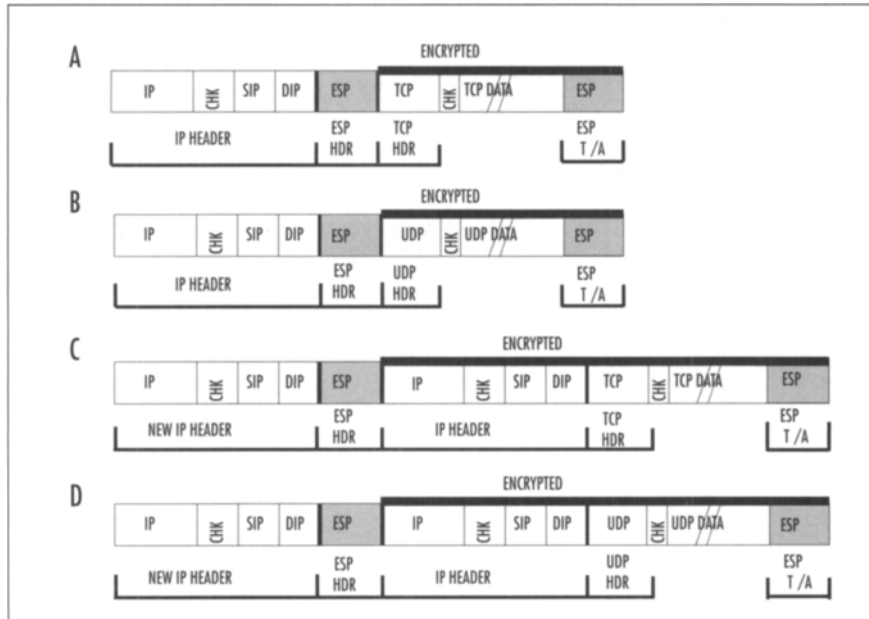
**Figure 8.11** Authentication Header: Transport and Tunnel Modes**NOTE**

The key to the incompatibility of NAT and IPsec AH is the presence of the ICV, whose value depends partially on the values of the source and destination IP addresses, the IP header checksum, and either the TCP or UDP header checksum. The AH ICV calculation takes into account the mutable and predictable header fields that change as the packet moves from hop to hop through the network, but because intermediate devices do not share the secret key, they cannot recalculate the correct ICV after NAT has altered the aforementioned original header fields.

ESP, on the other hand, was used initially only for encryption; authentication functionality was subsequently added. The ESP header is inserted after the IP header and before the upper layer protocol header (transport mode) or before an encapsulated IP header (tunnel mode).

Figure 8.12 shows the location of the ESP header in both transport mode (sections A and B) and tunnel mode (sections C and D) for TCP (sections A and C) and UDP (sections B and D). In transport mode, the original IP header is followed by the ESP header. The rightmost field contains the ESP trailer and optionally, the ESP authorization field. Only the upper-layer protocol header, data, and the ESP trailer (also, optionally, the ESP authorization field) is encrypted. The IP header is not encrypted.

Figure 8.12 ESP Header: Transport Mode and Tunnel Mode



In transport mode, ESP encrypts the entire packet. This means that the entire original IP datagram, including the original IP and protocol header, is encrypted. In this mode, when IP traffic moves between gateways, the outer, unencrypted IP header contains the IP addresses of the penultimate source and destination gateways, and the inner, encrypted IP header contains the IP source and destination addresses of the true endpoints. However, even though ESP encrypts most of the IP datagram in either transport or tunnel mode, ESP is relatively compatible with NAT, since ESP does not incorporate the IP source and destination addresses in its keyed message integrity check. Still, ESP has a dependency on TCP and UDP checksum integrity through inclusion of the pseudo-header in the calculation. As a result, when checksums are calculated, they will be invalidated by passage through a NAT device (except in some cases where the UDP checksum is set to zero).

NAT traversal using ESP leads to a catch-22. NAT must recalculate the TCP header checksums used to verify packet integrity, because as was shown earlier, NAT modifies those headers. If NAT updates the header checksum, ESP authentication will fail. If NAT does not update the checksum, TCP verification will fail. One way around this, if the transport endpoint is under your control, is to turn off checksum verification, but I'm not aware of anyone who has done this in production environments. A second, more common means to do this is to NAT before IPsec; don't perform IPsec before NAT. This can be accomplished by locating the NAT device logically behind the IPsec device. The most common form of NAT traversal used today relies on encapsulating IPsec packets in UDP in order to bypass NAT devices. The IPsec packet is encapsulated in a meta-UDP packet and the meta-

UDP packet is stripped off after it passes through the NAT device. This enables NAT and IPsec to function together but none of these are hardly elegant solutions.

## NAT as a Topology Shield

NAT provides a security function by segregating private hosts from the publicly routed Internet. Depending upon your addressing requirements, NAT can isolate, to some extent, your VoIP network IP space from the balance of your internal network IP space. The large number of private RFC1918 IP addresses allows system architects to intelligently address hosts and other network elements based upon location, function, or other criteria during the design phase of the VoIP network.

External hosts cannot directly access a particular internal host if a NAT intervenes since the external host has no way of targeting its payload to a chosen IP address. Of course, when addresses are assigned dynamically, it becomes even more problematic for an attacker to point to a specific host within the NAT domain. This may help protect internal hosts from external malicious content. At worst, NAT is an additional layer of security controls that you implement as part of your overall security architecture.

The IPsec model is instructive in that it illustrates a complex interaction between encryption and NAT. However, IPsec is not the only functional or proposed security mechanism for VoIP environments. SSL/TLS, S/MIME, HTTP 1.1 digest, and ZRTP have also been proposed as security instruments. Nor are all environments as simple as the symmetric examples we have seen where one or more devices reside on opposite sides of a NAT device. Asymmetric or hairpin call routing (a call from one phone behind a NAT to another phone behind the same NAT), in an environment where basic NAT and encryption issues have been resolved, can cause communications to fail. The point here is to introduce some of the concepts that you will come across as you design and troubleshoot in this area. We'll see in the next section how encryption, NAT, and VoIP protocols work (or don't work) together.

## Firewalls

Firewalls are a key component of virtually any network security architecture. Firewalls demarcate inside from outside, trusted from nontrusted networks, and they are used to separate VoIP from data on internal networks. Two significant issues affect firewall performance with regard to VoIP: The first is that the boundary between inside and outside or trusted and nontrusted networks gradually is becoming less clear; the second is that most firewalls fail to adequately process VoIP packets and sessions, particularly (as you were forewarned) if those session and packets are encrypted.

## A Bit of Firewall History

Traditionally, firewalls have provided a physical and logical demarcation between the inside and the outside of a network. The first firewalls were basically just gateways between two networks with IP forwarding disabled. Most contemporary firewalls share a common set of characteristics:

1. They are single points between two or more networks where all traffic must pass (choke point).
2. They can be configured to allow or deny IP (and other protocol) traffic.
3. They provide a logging function for audit purposes.
4. They provide a NAT function.
5. Their operating systems are hardened.
6. They often serve as a VPN endpoint.
7. They fail closed—that is, if the firewall crashes in some way, no traffic is forwarded between interfaces.

## Shallow Packet Inspection

Steven Bellovin classically stated, “Firewalls are barriers between ‘us’ and ‘them’ for arbitrary values of ‘them.’”

Shallow packet inspection, in contrast to deep packet inspection, inspects only a few header fields in order to make processing decisions. IP packet filtering firewalls all share this same basic mechanism: As an IP packet traverses the firewall, the headers are parsed, and the results are compared to a rule set defined by a system administrator. The rule set, commonly based upon source and/or destination IP address, source and/or destination port, or a combination of the two, defines what type of traffic is subsequently allowed or denied. Packet filtering (and the code that performs these tasks) based upon parsing of IP headers has been common for many years.

## Tools & Traps...

### Thank Goodness for GUIs

Interestingly, some early (and not particularly popular) packet-filtering implementations required that the system administrator define specific byte fields with the packet headers, and the specific byte patterns to match against. Imagine setting up and troubleshooting a 100-field rule set on one of these systems!

## Stateful Inspection

Stateful Inspection Firewall Technology, a term coined by Check Point Software Technologies, described a method for the analysis and tracking of sessions based upon source/destination IP address and source/destination ports. A stateful inspection firewall registers connection data and compiles this information in a kernel-based state table. A stateful firewall examines packet headers and, essentially, remembers something about them (generally source/destination IP address/ports). The firewall then uses this information when processing later packets.

Interestingly, Lance Spitzner ([www.spitzner.net/](http://www.spitzner.net/)) showed that, contrary to what we would expect, sequence numbers and other header information is not utilized by Check Point in order to maintain connection state tracking. Stateful packet inspection firewalls, like packet filtering firewalls, have very little impact on network performance, can be implemented transparently, and are application independent.

## Medium-Depth Packet Inspection

Application layer proxies or gateways (ALG) are a second common type of firewall mechanism. ALGs peer more deeply into the packet than packet filtering firewalls but normally do not scan the entire payload. Unlike packet filtering or stateful inspection firewalls, ALGs do not route packets; rather the ALG accepts a connection on one network interface and establishes the cognate connection on another network interface. An ALG provides intermediary services for hosts that reside on different networks, while maintaining complete details of the TCP connection state and sequencing. In practice, a client host (running, for example, a Web browser application) negotiates a service request with the AP, which acts as a surrogate for the host that provides services (Web server). Two connections are required for a session to be completed—one between the client and the ALG, and one between the AP and the server. No direct connection exists between hosts.



Additionally, ALGs typically possess the ability to do a limited amount of packet filtering based upon rudimentary application-level data parsing. ALGs are considered by most people to be more secure than packet filtering firewalls, but performance and scalability factors have limited their distribution. An adaptive (coined by Gauntlet), dynamic, or filtering proxy is a hybrid of packet filtering firewall and application layer gateway. Typically, the adaptive proxy monitors traffic streams and checks for the start of a TCP connection (ACK, SYN-ACK, ACK). The packet information from these first few packets is passed up the OSI stack and if the connection is approved by the proxy security intelligence, then a packet filtering rule is created on the fly to allow this session. Although this is a clever solution, UDP packets, which are stateless, cannot be controlled using this approach.

Although current stateful firewall technologies and ALGs provide for tracking the state of a connection, most provide only limited analysis of the application data. Several firewall vendors, including Check Point, Cisco, Symantec, Netscreen, and NAI have integrated additional application-level data analysis into the firewall. Check Point, for example, initially added application proxies for Telnet, FTP, and HTTP to the FW-1 product, but have since replaced the Telnet proxy with an SMTP proxy. Cisco's PIX fix-up protocol initially provided for limited application parsing of FTP, HTTP, H.323, RSH, SMTP, and SQLNET. Both vendors since have added support for additional applications. To sum up, the advantages of ALGs is that they do not allow any direct connections between internal and external hosts; they often support user and group-level authentication; and they are able to analyze specific application commands inside the payload portion of data packets. Their drawbacks are that ALGs tend to be slower than packet filtering firewalls, they are not transparent to users, and each application requires its own dedicated ALG policy/processing module.

## Deep Packet Inspection

To address the limitations of Packet Filtering, Application Proxies, and Stateful Inspection, a technology known as Deep Packet Inspection (DPI) was developed (or marketed). DPI analyzes the entire packet, and may buffer, assemble, and inspect several related packets as part of a session. DPI operates at L3-L7 of the OSI stack.

DPI engines parse the entire IP packet, and make forwarding decisions by means of a rule-based logic that is based upon signature or regular expression matching. That is, they compare the data within a packet payload to a database of predefined attack signatures (a string of bytes). Additionally, statistical or historical algorithms may supplement static pattern matching.

The issue with DPI is that packet data contents are virtually unstructured compared with the highly structured packet headers (review the previous section on NAT for more details). Analysis of packet headers can be done economically since the locations of packet header fields are restricted by protocol standards. However, the payload contents are, for the most part, unconstrained. Searching through the payload for multiple string patterns within the datastream is a computationally expensive task. And as wire speeds increase, the require-

ment that these searches be performed at wire speed adds to the cost. Additionally, because the threat signature database is dynamic, it must be easily updateable—this rules out the use of normal ASICs. Promising approaches to these problems include a software-based approach (Snort implementing the Boyer-Moore algorithm) and a hardware-based approach (FPGAs running a Bloom filter algorithm).

## Tools & Traps...

### FPGAs

FPGAs (Field Programmable Gate Arrays) are a class of general-purpose digital logic chips. Some of the larger FPGA vendors are Xilinx and Altera. FPGAs are dynamically programmable, support a wide range of signal processing, and offer true parallel processing. They may provide the hardware solution for processing entire packet streams at multigigabit wire speeds.

Deep Packet Inspection is a promising technology in that it may help to solve these problems. DPI engines are situated at network boundaries where bandwidth and security controls are logically implemented. New, programmable ASICs coupled with efficient algorithms can realistically parse the entire contents of each packet at gigabit speeds. Also, combining Firewall and IDS within a single device should simplify device configuration and management. But there are concerns as well.

One of the primary benefits of the traditional firewall/IDS deployment is that the failure of one component does not leave the network completely unprotected. Deploying devices with separate functionality also prevents being locked into a single solution and vendor.

Particular attention must be paid to firewall and deep packet inspection configurations to make sure they don't introduce unacceptable latency. Implementation of some security measures can degrade QoS. These complications range from interruption or prevention of call setup by firewalls to encryption-produced latency and delay variation (jitter).

## VoIP-Aware Firewalls

With a basic understanding of NAT, encryption, and firewall technologies under our belts, it is now possible to appreciate the challenges of securing VoIP traffic without either throwing away your firewalls or obstructing call flow. The basic problem is twofold: firewall administrators are loath to open up a range of high ports (> 1024) that will allow uncontrolled connections between external and internal hosts, and firewalls often rewrite information that is necessary for VoIP signaling traffic to succeed. In the first case, call parameter traffic, media traffic, and media control traffic travel on arbitrary high ports. In the second case, the general

rule in this fraction of the H.323 protocol suite is that IP address information and port numbers are exchanged within the data stream of the preceding connection. Obviously, since SIP and H.323 are separate protocols, they have different firewall requirements. First, we'll look at H.323.

## H.323 Firewall Issues

For basic voice call setup H.323 requires at least the ports shown in Table 8.1 to be opened.

**Table 8.1** Basic VoIP Call Setup

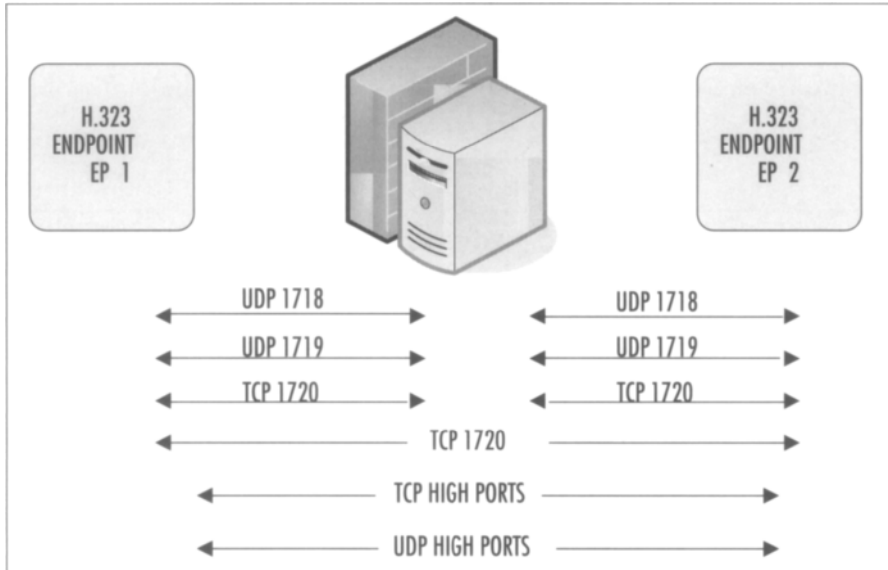
FUNCTION	PORT	PROTOCOL
Gatekeeper discovery	1718	UDP
Gatekeeper RAS	1719	UDP
Q.931 Call Signaling (Setup)	1720	TCP
H.245 Signaling (Call parameters)	1024–65535	TCP
RTP/RTCP (Media)	1024–65535	UDP
H.235 Secure Signaling	1300	TCP

The sequence of H.323 call setup and control depends upon the presence or absence of a gatekeeper (see Chapter 3 for more details). In the example shown in Figure 8.13, we will assume that the network consists of a single gatekeeper and two H.323 endpoints (endpoints can be telephones, gateways, MCUs, etc.) using direct signaling. A generic H.323 call sequence begins with one endpoint (EP1) initiating a gatekeeper discovery process that opens ports UDP/1718. If successful, RAS messages are sent over port UDP/1719 as part of the registration and admission process. EP1 then sends a call signaling setup message to EP2 over TCP/1720. After EP2 registers with the gatekeeper, it sends several H.225 messages to EP1 over port TCP/1720 and the call is established. At this point in the exchange, three static bidirectional ports have been opened—two between EP1 and the gatekeeper and one between EP1 and EP2 (we can ignore the ports opened between EP2 and the gatekeeper for this discussion).

After call establishment, an H.245 call control channel is established over TCP. A subset of RAS messages include IP addressing information in the payload, typically meant to register an endpoint with a gatekeeper or learn about another registered endpoint. The ephemeral port numbers for this connection are established by the preceding H.225/Q.931 signaling traffic (are contained within the data portion of the Q.931 message). After capabilities exchange over the H.245 control channel, media (RTP and RTCP) port and real (rather than private) IP addresses are exchanged. This information again is transported within the data portion of the H.245 message. Q.931 tunneling of H.245 messages or Q.931 multiplexing can reduce the number of ports opened, but the problem with H.323 and firewalls that do NAT should now

be apparent—in order to properly route messages to the real, rather than public, address, a NAT firewall or proxy must inspect each signaling and control channel for the correct ports and IP addresses, and rewrite them appropriately.

**Figure 8.13** H.323 Communications Ports



Since H.323 relies greatly on dynamic ports, packet filtering firewalls are not a particularly favorable solution, as every port greater than 1024 has to be opened bidirectionally for a call to take place. Thus, firewall solutions supporting H.323 must at least dismantle and inspect signaling packets (H.245, H.225.0) and statefully open the firewall ports for both H.245 control packets and bidirectional UDP media packets as well. As if this is not enough complexity, the signaling and control messages are binary encoded according to ASN.1 rules. ASN.1 parsers have been exploited in a variety of implementations, and parsing takes time, adding latency to an already latency-sensitive application.

## SIP Firewall Issues

Unlike H.323, SIP's syntax is based on HTTP. ASCII is more economically parsed than PER encoded PDUs. Like H.323 though, the topology of SIP sessions differs from that of an HTTP, SMTP session in that connections can and will be initiated from parties outside of the firewall. This would be akin to a Web server requesting that you browse its site. The SIP connection topology is similar to IM (Instant Messaging) topologies where callers (session initiators) can exist on either side of the firewall.

Typically, SIP infrastructure consists of User Agents (UAs—normally IP phones or soft-phones), SIP Proxies (SP), and SIP Registrars (SR). For a careful and thorough analysis of

the attacks that can be promulgated against SIP infrastructure see Ofir Arkin's excellent treatment at [www.sys-security.com/index.php?page=voip](http://www.sys-security.com/index.php?page=voip).

SIP sessions can be broken down into three constituents: locating the called person, session setup, and media transport. In the context of traversing firewalls and NAT, SIP's primary problem relates to determination of the "real" IP addresses of end users or UAs, which are often located in private IP address space. Unlike H.323, SIP does not cascade IP address and port numbers within control packets. However, as is the case with H.323, SIP, when used as a VoIP application, opens bidirectional UDP media channels over random high ports.

### WARNING

---

Recent issues that affect Cisco SIP Proxy Server (SPS) [Bug ID CSCec31901] demonstrate the problems SIP implementers may experience due to the highly modular architecture of this protocol. The SSL implementation in SPS (used to secure SIP sessions) is vulnerable to an ASN.1 BER decoding error similar to the one described for H.323. This example illustrates a general concern with SIP: As the protocol links existing protocols and services together, all the classic vulnerabilities in services such as SSL, HTTP, SMTP, and IM may resurface in the VoIP environment.

---

SIP-aware firewalls will need to address these two issues. A helper proxy and registrar, closely associated with the firewall, can allow SIP location services to function in the presence of NAT. The Ingate firewall is one example of this approach. The high ports for RTP media channels are negotiated during the session setup phase, remain open for the duration of the call, and should be closed immediately after the call's termination. A SIP-aware firewall will have to manage these channels by opening a "pinhole" in the firewall rule set that temporarily allows these channels.

## Bypassing Firewalls and NAT

H.323 and SIP have proven so difficult to manage with modern firewalls that some system administrators have given up, and instead, have implemented VoIP controls at other points: on the network perimeter, outside the perimeter, or in specially designated VoIP-DMZs. To secure calls from remote systems, NIST, in its excellent document, *SP 800-58: Security Considerations for Voice Over IP Systems*, suggests the use of VPNs to eliminate all the processing issues associated with NAT, firewalls, and encryption; however, as NIST points out, VPNs don't scale well.

There are literally dozens of proposals and hundreds of acronyms for managing VoIP sessions. My personal favorite is AYIYA (anything in anything). One of the examples mentioned in this proposal is tunneling IPv6-in-UDP-in-IPv4! My sense is that if the protocol

requires this much convolution, perhaps we need to revisit the protocol itself.

Unsurprisingly, no one of these approaches that follow has become dominant to date.

One successful solution to these issues is the development of Session Border Controllers (SBCs). SBCs are a class of dedicated network devices, generally located at the network perimeter, that offload VoIP security, NAT traversal, and media and signaling processing. SBCs are high-powered, complicated network devices. The primary function of most SBCs is to serve as a VoIP-aware NATing firewall. As long as packet latencies remain low and scale uniformly on both media and signaling channels, there is no need to split these functionalities.

However, for more complex operations on the media stream, such as transcoding and silence detection and/or suppression, one or more additional DSP (digital signal processing) farms, controlled by the SBC, can be added. Offloading DSP resources to a separate device will help lower SBC prices by providing additional transcoding capability only when the enterprise requires additional capabilities.

SBCs are often purpose-built to enable a spectrum of services, including real-time IP, support for H.323, SIP, and MGCP, deep-packet processing, traffic management, classification, reporting, and billing. SBCs also provide for lawful intercept. For more information about SBCs, you can check out the following vendors: Acme Packet, Ditech (Jasomi), Juniper (Kagoor), Netrake, Newport Networks, and Tekelec.

## Tools & Traps...

### CALEA

In 1994, the Communications Assistance for Law Enforcement Act (CALEA) was signed. In August 2005, in response to a request from the DEA and FBI, the FCC ruled that VoIP must comply with CALEA—that is, that VoIP must be capable of lawful intercept. This seems to mean that common carriers, facilities-based broadband Internet access providers, and providers of interconnected VoIP services must accommodate law enforcement wiretaps at any time.

A number of experts have commented that that it's one thing to demand that VoIP applications comply with CALEA, and it is quite another to require that the Internet be reengineered at the protocol level to provide wiretapping services. This area of law affects you if you are involved in design or maintenance of a VoIP network.

Because of their complexity, SBCs are expensive and management intensive; thus, in the near future, SBCs will be available to only carriers and large organizations.

Midcom (Middlebox protocol) is an interesting concept that may yet organize all the additional components proposed as adjuncts to firewalls. Essentially, midcom promises to allow applications, using a common language, to signal their requirements to trusted third parties such as firewalls, SBCs, IPSes (intrusion prevention systems), and NATs. Additionally, midcom supports abstraction of various VoIP processing components (for example, ASN.1 parsing or stateful inspection). Asterisk reportedly uses midcom to enable an IP PBX to indicate to a firewall which ports the PBX requires open. Although promising, the midcom protocol has yet to be finalized by the IETF.

### *STUN, TURN, and ICE*

The following protocols and frameworks are methods for enabling SIP, but not H.323, to work in the presence of NATs. STUN (Simple Traversal of UDP through NATs) is a client-server protocol designed to enable an endpoint to discover its public IP address and the type of NATs between the endpoint and its peer. The STUN protocol describes a STUN-enabled client in private IP space and its means of communication with a public STUN server. The public STUN server informs the private client of the client's public IP presence (IP address and port) within a SIP session. The following list of public STUN servers was active at the time of this writing:

- `stun.fwdnet.netn`      69.90.168.14
- `stun.fwd.orgn`      64.186.56.73
- `stun01.sipphone.comn` 69.0.208.27
- `stun.softjoys.comn`    69.3.254.11
- `stun.voxgratia.orgn`    83.103.82.85
- `stun1.vovida.orn`      128.107.250.38
- `xtunnels1.xten.nen`    64.69.76.23

STUN is relatively successful in residential VoIP deployments, but it is not an enterprise solution for a number of reasons. Key among these are STUN does not support TCP (TCP conformance is mandated by the SIP draft), and STUN does not work in the presence of symmetric NATs (the binding table entry for a symmetric NAT is based on source IP and port, and destination IP and port), which are the most common type of NATs in the enterprise.

STUN has been enhanced by the addition of the TURN protocol (Traversal Using Relay NAT). TURN is identical in syntax and general operation to STUN, but differs in behavior. In a simplified STUN exchange, the private STUN client sends a UDP packet to a public STUN server. The STUN server copies the last (closest to the STUN server) public IP address and mirrors this information to the private STUN client. No resources or band-

width are allocated, but the private STUN client is now aware of its public IP presence. The first TURN message, on the other hand, is part of an authentication exchange.

Authentication is required because a TURN server allocates its own resources (processing time, NIC, etc.) as part of its role as a proxy/relay for the private TURN client. TURN complements STUN in that TURN works with symmetric NATs and relays both TCP and UDP. This performance comes at a price, however: TURN requires multiple relays, which adds to latency. Because both of these protocols (as well as other related NAT traversal strategies such as UPNP, RSIP, and UDP hole punching) have strengths and weaknesses, a framework called Interactive Connectivity Establishment (ICE) has been proposed to coordinate these protocols. ICE explains how to use the other protocols for NAT traversal.

Skype has been fabulously successful as a peer-to-peer voice application. Many feel that Skype's success has pushed leading vendors such as Microsoft to support ICE. ICE (see IETF draft-rosenberg-sipping-ice-00 for more information, 2003) is not a protocol. It is best understood as a method, determined by additional SDP attributes, for enabling SIP traffic through multiple independent NATs by utilizing STUN, TURN, or other servers and protocols. ICE basically allows a privately addressed IP device to interrogate trusted external partners about public IP and NAT environment. In the author's own words:

**ICE always works, independent of the types or number of NATs. It always represents the cheapest solution for a carrier. It always results in the minimum voice latency. It can be done with no increase in call setup delays. It is far less brittle than STUN. ICE also facilitates the transition of the Internet from IPv4 to IPv6 ...**

Quite a list, but it is not clear, for all of its promises, that ICE will catch on. Interestingly, like the ICE framework, Skype attempts to connect peers directly. Failing that, it apparently uses a modified STUN/TURN-like mechanism to bypass NAT firewalls. In the event, that this fails, Skype circumvents firewalls by emulating HTTP traffic on TCP ports 80 and 443.

The general issue with most of these approaches is that admission/egress control between the external and internal networks becomes more decentralized and less manageable as additional security and security adjunct components are added. Multiple chokepoints require more resources to defend them. Management and change control will become more difficult since firewall administrators will have to learn to configure and maintain additional devices that are likely from different vendors.

## Access Control Lists

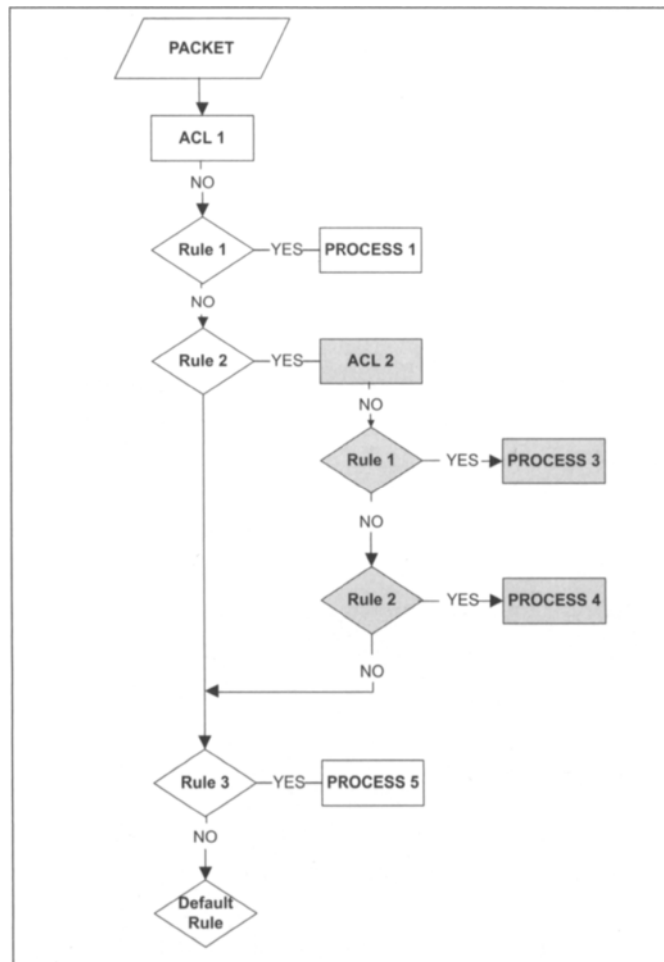
Network access control lists (ACLs) are table-like data structures that normally consist of a single line divided into three parts: a reference number that defines the ACL; a rule (usually permit or deny); and a data pattern, which may consist of source and/or destination IP addresses, source and/or destination port numbers, masks, and Boolean operators. Other pat-



terns are used, but the ones listed are most common. ACLs generally are applied to the ingress or egress side of an interface.

As a packet traverses the interface, the ACL is scanned from top to bottom—in the exact order that it was entered—for a pattern that matches the incoming packet. Figure 8.14 shows the process flow for an access control list. In this case, a packet enters at the top and as it negotiates the ACL structure, some portion or portions of the packet are tested for a match at each rule-node. If the match succeeds, then related processing takes place; if there is no match, then the packet data is tested by the next lower node. A default rule should always be added to process any packets that traverse the entire ACL structure. Note that in this figure, an ACL rule has called an additional ACL. This type of ACL organization leads to exceptionally fine filtering granularity, but these complex rule sets, unless carefully designed, can be computationally expensive, slowing traffic unacceptably.

**Figure 8.14** ACL Flow Diagram—Decision Based upon Match/No Match



A general rule-of-thumb is that outbound ACLs are more efficient than inbound ACLs since the inbound logic must be applied to every packet, but the outbound logic is applied only to those packets exiting a particular interface. ACLs normally are applied at layers 3 and 4 of the OSI model, but some vendors (Cisco and Extreme, for example) offer layer 2 ACLs, and others (Alteon/Nortel, for example) offer ACLs at layers 5 and above.

ACLs, in coordination with VLANs, QoS, and firewalls, are powerful tools for segregating VoIP traffic from other traffic. Additional services may be permitted or denied based upon the client's infrastructure requirements.

## Summary

Logically separate data from voice traffic. Plan on establishing at least two VLANs and put your VoIP system components on a separate dedicated VLAN with 802.1p/q QoS (Quality of Service) and priority VLAN tagging. Limit physical and terminal access to your switch consoles to only authorized personnel.

Traffic shaping normally is associated with ensuring performance, but it also plays a role in security. Voice and data on separate logical VLANs share the same physical bandwidth. If hosts on the data VLAN become infected with viruses or worms that flood the network with traffic, VoIP traffic may remain unaffected if traffic shaping has been configured correctly to ensure that VoIP traffic has priority. The reverse is also true.

Access control lists find new utility at layer 3 of the internal networks, acting to fine-tune and control traffic. Keep ACLs simple and apply them only to egress ports in order to minimize their processing requirements.

Network Address Translation (NAT) will continue to be a major obstacle in VoIP migrations until Ipv6 becomes commonly adopted. Encryption across a NAT device is particularly problematic as both H.323 and SIP embed layer-3 routing and signaling information inside the IP datagram payload.

There is still no simple solution for securely handling calls that originate externally. Packet filtering and stateful inspection firewalls can open a "pinhole" through which outbound replies can pass. However, particularly in the case of SIP-based solutions, private translated internal IP addresses prevent incoming calls from reaching the correct recipient.

One promising approach is to combine an application layer gateway with a stateful packet filtering firewall. In this approach, an ALG software module running in close logical proximity to a NAT firewall device updates payload and header data made invalid by address translation. Complicating this solution is that the ALG software must be configured to be aware of the internal network architecture; and it requires that the ALG software understand the higher-layer protocol that it needs to "patch," thus each protocol requires a separate ALG module.

One particular technology that looks promising with regard to making firewalls intelligent and VoIP-aware is Deep Packet Inspection (DPI). Deep Packet Inspection may enhance

firewall capabilities by adding the ability to dynamically open and close ports for VoIP application traffic—essentially collapsing Intrusion Detection (IDS) functionality into the firewall appliance so that both a firewall and an in-line IDS are implemented on the same device.

Unfortunately, some of these products have been shown to be vulnerable to exploitation of software defects in their DPI inspection engines. These data suggest that the addition of these enhanced functions to firewalls may weaken, rather than strengthen network perimeter security.

The bottom line is that organizations must be able to differentiate and control traffic types based upon the contents of the application payload as networked application traffic and threats to that traffic evolve.

## IETF Encryption Solutions for VoIP

### Solutions in this chapter:

- Suites from the IETF
- S/MIME: Message Authentication
- TLS: Key Exchange and Signaling Packet Security
- SRTP: Voice/Video Packet Security

# Introduction

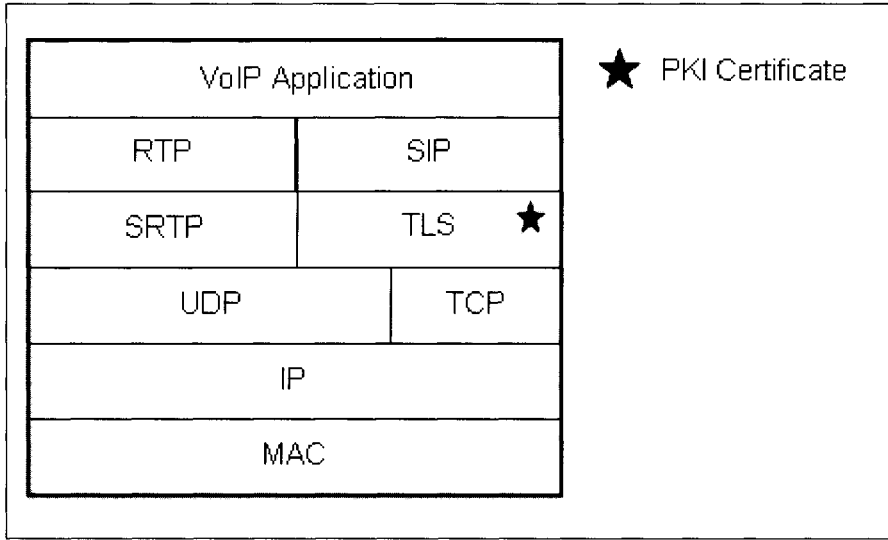
There are two competing breeds of VoIP signaling protocols, H.323 from the ITU and SIP from the IETF. Accordingly, there are also two groups of VoIP security protocols accompanying each of them. One for H.323 is a group of protocols named H.235.x, and the other for SIP includes TLS, S/MIME, and SRTP. They are not completely exclusive to each other. Some components are overlapped, such as X.509 digital certificate, TLS secured transport, and SRTP encryption. In this chapter, we will put our main focus on protocol suites for SIP from the IETF, and then a brief introduction to ITU suites (H.235 group); pointers to individual components are presented for the investigative reader.

## Suites from the IETF

Realizing the security issues present in VoIP, the IETF picked up three landmark security protocols in the SIP standard—Transport Layer Security (TLS), Secure/Multipurpose Internet Mail Extensions (S/MIME), and Secure Real-Time Transfer Protocol (SRTP)—to be used for securing SIP service. The basic approach consisted of adding a security layer below the existing VoIP protocol rather than crafting a new security protocol. The layered architecture is shown in Figure 9.1. The advantage of this approach is that existing protocol implementation can be reused for secured communication by adding security layers.

In general, TLS, which was chosen to protect SIP signaling messages, provides an upper layer secured tunnel to its peer entity. It is basically a successor of Secure Sockets Layer (SSL) version 3. The Service Data Unit (SDU) from the upper layer is encrypted before transmission. At the other end, the received Protocol Data Unit (PDU) is decrypted and passed to the upper layer. Each entity at both ends must have a legitimate certificate issued from a Certificate Authority (CA), which is mandatory for the TLS handshake operation. SIP signaling is passed through the secured tunnel.

**Figure 9.1** Layered Architecture of VoIP Security Protocol



SRTP is used to secure voice/video media from possible eavesdropping and tampering. It secures the confidentiality of RTP payloads and the integrity of all RTP packets by adopting the Advanced Encryption Standard (AES) as a default encryption/decryption algorithm using a symmetric cryptographic key. It also protects against replayed packet attack. The most sensitive issue in SRTP use is how the secret key can be shared between two communicating nodes. Embedding the key manually in all the phones is too cumbersome and error prone. For efficiency, RTP and SRTP can be implemented as one layer, rather than two separate layers. TLS and SRTP are the key components that play a major role in securing VoIP service.

However, there must be supporting protocols or an infrastructure that can authenticate users, validate node/user certificates, and exchange cryptographic keys. Each of these elements should work together in harmony to provide secured VoIP service.

## S/MIME: Message Authentication

In order for you to secure Internet mail, the message must be protected from tapping or tampering, and the sender and receiver must also be correctly identified. The reason why Spam e-mail is thriving these days is that the e-mail sender can be easily faked or spoofed.

Secure/Multipurpose Internet Mail Extensions (S/MIME), specified in the Certificate Handling (RFC 3850) and Message Specification (RFC 3851) RFCs, provide a standard for public key encryption and for signing e-mail encapsulated in the popular MIME format. S/MIME provides the following cryptographic security services for electronic messaging applications: authentication, message integrity, nonrepudiation of origin (using digital signa-

tures), and data confidentiality (using encryption). S/MIME is not restricted to mail. It can be used with any transport mechanism that transports MIME data, such as HTTP or SIP message bodies (and certain SIP headers).

S/MIME applies to the message body overall, but the SIP standard also provides a mechanism to apply S/MIME to protect sensitive headers. Message bodies like SDP are encrypted with S/MIME to keep integrity and remain confidential. However, the header information such as To, From, Call-ID, CSeq, and Contact cannot remain confidential end to end. They are indispensable information for intermediaries, such as SIP proxy servers, firewalls, or UAS, to establish the requested call. To overcome this issue, the information is provided in both plaintext and an S/MIME encrypted format in a SIP message. So the intermediaries may have access to the information without being bothered to decrypt them. And the final recipient with a proper key to decrypt the information can compare the decrypted ones with plaintext to check message integrity and the sender's identity.

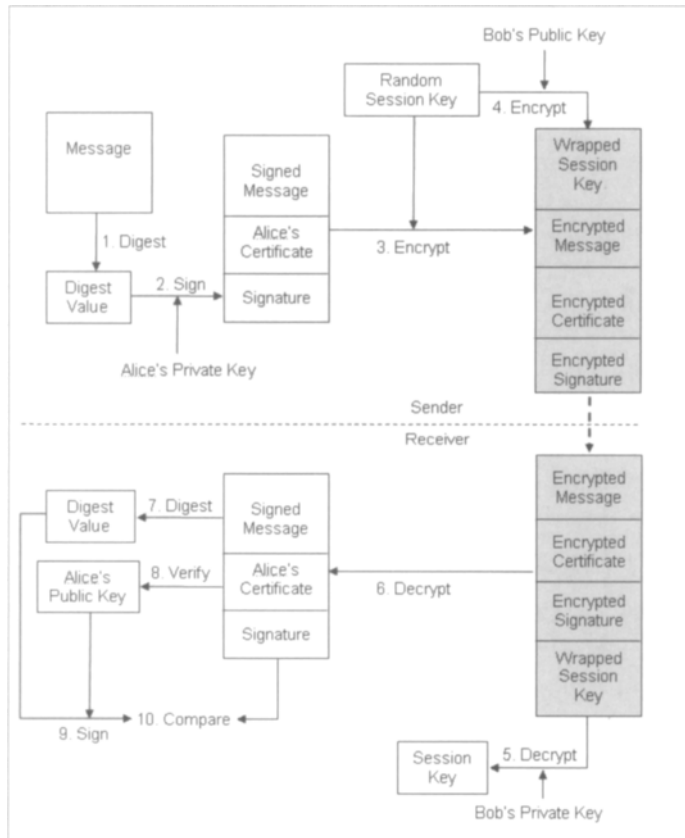
To understand how an S/MIME-based system delivers messages secured to its destination, knowledge of the fundamentals of PKI-based messaging system is necessary. Figure 9.2 shows the overall operation and flow of keys and messages within and between the systems.

1. The raw message is digested using a digestion algorithm. Without digestion, it takes a much longer time to process raw messages with digital signatures. The digestion, or hashing, reduces the message size to one adequate for signing.
2. Alice signs the digested message using a digital signature algorithm and appends the signature to the original message and to her certificate.
3. A session key is randomly generated and used to encrypt the message, certificate, and signature using an encryption algorithm.
4. The random session key is encrypted by Bob's public key using a public key encryption algorithm and wrapped into the encrypted message. The resulting message, which is shown as a shaded box, is transmitted to the receiver.
5. On the receiver side, a random session key is retrieved first by decrypting it with Bob's private key using the same algorithm that appears in step 4.
6. With the recovered session key, the encrypted messages, certificate, and signature are all decrypted using the same algorithm as that in step 3. In this way, data confidentiality is achieved. Now, Bob needs to check to see if the message is really signed by Alice and has to make sure that it has not been tampered with while being transmitted.
7. Using the same algorithm that appears in step 1, Bob digests the message.
8. Bob verifies that Alice's certificate is legitimate. If it is, Alice's public key is retrieved from the certificate.

- 9 Bob uses the same algorithm that appears in step 2, and the digested value is signed with Alice’s public key.
10. The computed signature is compared with the received one. If it does not match, the message has been tampered with. The tampering occurred from outside of the network. Authentication, message integrity, and nonrepudiation therefore are achieved.

During the previous operation, four PKI security primitives were used: digestion, encryption, public key encryption, and digital signature. S/MIME basically specifies which algorithm to use to carry out the four primitives, to format the message, and how to handle the message after the security primitives are applied.

**Figure 9.2 S/MIME Message-Sending Process**





## S/MIME Messages

To certify the sender or receiver, X.509 PKIX (RFC 3280) is adopted. S/MIME messages are a combination of MIME bodies and Cryptographic Message Syntax (CMS) content types.

### Sender Agent

Before using a public key to provide security services, the S/MIME agent verifies that the public key is valid. Sending agents should include any certificates for the user's public key(s) and associated issuer certificates. This increases the likelihood that the intended recipient can establish trust in the originator's public key(s).

It should include at least one chain of certificates up to, but not including, a CA that it believes the recipient can trust as authoritative.

### Receiver Agent

Receiving agents handle an arbitrary number of certificates of arbitrary relationship to the message sender and to each other in an arbitrary order. These agents do not simply trust any self-signed certificates as valid CAs, but use another mechanism, not discussed here, to determine if this is a CA that should be trusted.

### E-mail Address

Sending agents force the e-mail address in the From or Sender header in a mail message to match an Internet mail address in the signer's certificate. Receiving agents check to see that the address in the From or Sender header of a mail message matches an Internet mail address, if present, in the signer's certificate.

## TLS: Key Exchange and Signaling Packet Security

TLS is based on SSL protocol version 3. The IETF standardized TLS published as RFC 2246 in January of 1999. SSLv3 is incompatible with TLS by design. TLS is a protocol that provides a secure channel between two machines. It has facilities for protecting data in transit and for identifying its peer by checking the peer's X.509 certificate.

The secure channel is transparent, meaning that the data passed through the channel is unchanged. The data is encrypted between client and server, but the data that one end writes is exactly what the other end reads. Transparency allows nearly any protocol that can be run over TCP to be run over SSL/TLS with only minimal modification, which is very convenient. As depicted in Figure 9.1, TLS sits right above the TCP layer and below the SIP layer,

meaning that a message at the SIP layer is encrypted by TLS and transmitted through a TCP connection.

Each entity at both ends must have a legitimate certificate issued from a CA. Think of TLS as a transport layer like TCP on which you send SIP messages. There are open source *OpenSSL* APIs that can be used to set up TLS connections programmatically. Once the SSL connection is established, you basically write to the SSL socket, just as you would write to a TCP socket. The SIP message is transferred through the secured channel to its peer.

## WARNING

---

Many enterprises, rather than buying certificates from assured CAs, create their own CAs and issue certificates to their internal users. It may work well between the internal users. But when they want to establish secured communication with another enterprise, their certificates cannot be certified by a common root because the two enterprises do not have a publicly verifiable CA in common.

---

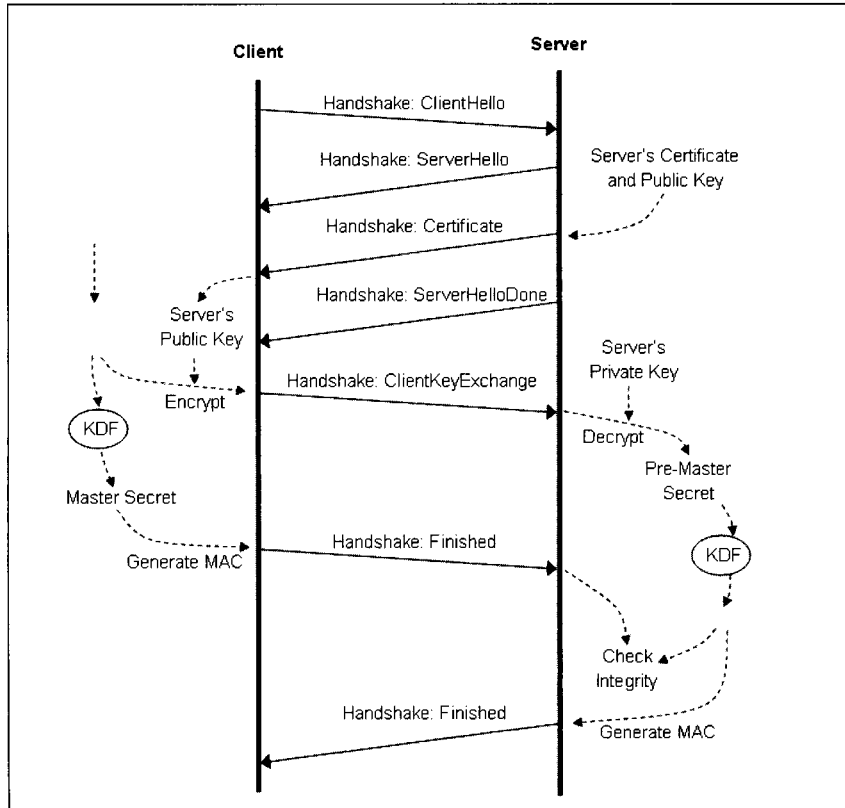
## Certificate and Key Exchange

Figure 9.3 shows the handshake between client and server. The purpose of the handshake is first so that the server and client can agree on a set of algorithms that will be used to protect the data. Second, they need to establish a set of cryptographic keys that will be used by those algorithms.

Figure 9.3 depicts the case where the client challenges the server's authentication. A detailed explanation of Figure 9.3 follows:

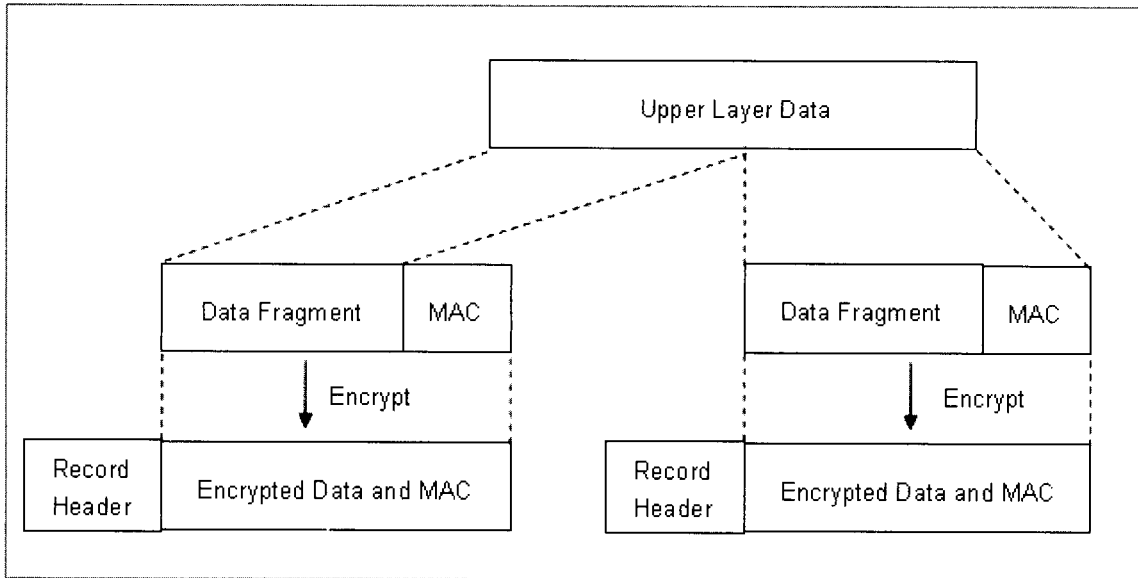
1. With the ClientHello and ServerHello message, the client and server agree on a list of algorithms they will use.
2. The server's certificate and public key are contained in the Certificate Message.
3. The client generates a random number, called a Pre-Master Secret key. Upon receiving a Certificate message, it checks authentication of the server's certificate and extracts its public key. The Pre-Master Secret key is encrypted by the server's public key and sent via the ClientKeyExchange message to the server. Meanwhile, the Key Derivation Function (KDF) generates a master key derived from the Pre-Secret Master key.
4. On the server side, the ClientKeyExchange message is decrypted by the server's private key, resulting in the Pre-Master Secret key. Using the same KDF as the client, the master key is derived from the Pre-Master Secret key.

Figure 9.3 SSL Handshakes for Certificate and Key Exchange



5. With the master key, the client generates the Message Authentication Code (MAC) of the entire previous message it received from the server and sends it via a Finished message to the server.
6. With the master key, the server generates a MAC of the entire previous message it received from the client and sends it via a Finished message to the client.
7. Both the server and the client check the integrity of the received MAC with all the messages they have sent so far.
8. If the check is successful, both server and client share the same Master Secret key.

Figure 9.4 shows how data from the upper layer is encapsulated by the TLS/SSL layer. After data is fragmented, MAC is appended before being encrypted. Then the SSL/TLS record header, containing content type, length, and SSL version, is attached to the encrypted text. There are four types of content: application, alert, handshake, and change cipher specification. The packets described in Figure 9.4 fall into application type and the messages for certification and key exchange in Figure 9.3 are grouped into handshake type.

**Figure 9.4** SSL/TLS Record Protocol

## SRTP: Voice/Video Packet Security

SRTP, specified in RFC 3711, describes how to protect telephony media for encryption of the RTP packet payload, for authentication of the entire RTP packet, and for packet replay protection:

1. Confidentiality of RTP packets protects packet payloads from being read by entities without the secret encryption key.
2. Message authentication of RTP packets protects the integrity of a packet against forgery, alteration, or replacement.
3. Replay protection ensures that the session address (IP address, User Datagram Protocol [UDP] port, and Synchronization Source RC [SSRC]) do not experience a DoS attack.

The protocol is located between the RTP application and RTP transport layers, sitting like a “bump in a stack.” It secures the confidentiality of RTP payloads and the integrity of all RTP packets by adopting the AES using a symmetric cryptographic key. The payloads from the RTP application are encrypted and encapsulated into an SRTP packet.

The most sensitive issue in using SRTP is how the secret key is shared between two nodes communicating in secret. The keys for these services are associated with the stream triple <IP address, UDP port, SSRC> and are called *SRTP cryptographic context*.

Unfortunately, key management for SRTP is a huge issue with the associated IETF standards since there have been multiple proposals, MIKEY and SDP Security Description (sdescription), on the table for years. Many implementation options exist within those schemes and a lot of unresolved implementation details caused early SRTP solutions in the market to use improper negotiation vehicles like the SIP INFO message or proprietary headers. As of the writing of this book (February 2006), all the interoperable SRTP implementations on the market are using proprietary negotiation or key management techniques that are nonstandard, although several vendors indicate that their sdescription-based solutions will be released shortly.

## Multimedia Internet Keying

Multimedia Internet Keying (MIKEY) is a simple key management solution intended to be used for one-to-one, simple one-to-many, and small size groups. It provides three different ways to transport or establish traffic encryption key (TEK): with the use of a preshared key, public-key encryption, and Diffie-Hellman (DH) key exchange.

The preshared key method and the public-key method are both based on key transport mechanisms, where the actual TGK (TEK Generation Key) is pushed securely to the recipient(s). In the Diffie-Hellman method, the actual TGK is derived instead from the Diffie-Hellman values exchanged between the peers.

## Session Description Protocol Security Descriptions

SDP Security Descriptions specify a new SDP attribute called *crypto*, which is used to signal and negotiate cryptographic parameters for SRTP media streams. The definition of the *crypto* attribute is limited to one-to-one unicast media streams. It assumes that the underlying service of secured data transport protocol, IPSec, TLS, or SIP S/MIME, protects the SDP message containing the *crypto* attribute. The attribute describes the cryptographic suite, key parameters, and session parameters for the preceding unicast media line.

```
a=crypto:<tag> <crypto-suite> <key-params> [<session-params>]
```

where *tag* is a decimal number used as an identifier for a particular *crypto* attribute; *crypto-suite* is an identifier that describes the encryption and authentication algorithms like AES\_CM\_128\_HMAC\_SHA1\_80; *key-params* consist of method and actual keying information; and *session-params* are specific to a given transport, and use of them is OPTIONAL.

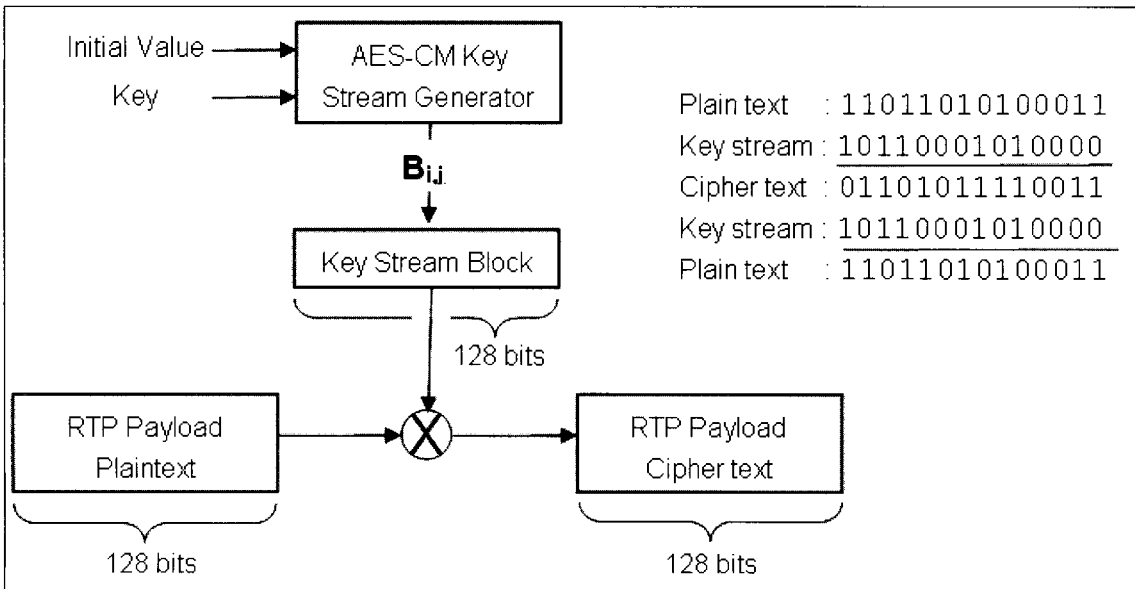
## Providing Confidentiality

A confidentiality service is obtained by encrypting the payload so that only the sender and receiver in possession of the keys can read it. Figure 9.5 shows one key stream block,  $\mathbf{B}_{i,j}$ , which is the AES encryption of the initial value (IV) with key. The IV is computed from the

48-bit packet index, the 32-bit SSRC, and the 112-bit salting key. All these parameters are left-shifted and exclusive-or'ed.

Each IV is encrypted along with the key to produce a pseudorandom block of 128 bits, shown as  $B_{i,j}$ . Each 128-bit block is exclusive-or'ed with an associated block of RTP payload plaintext to produce a block of cipher text, which covers either part of or the entire payload. Both the encryption and decryption processors run the key stream generator with the packet index, SSRC, and salting key; each processor synchronously produces the key stream  $B_{i,j}$ —a stream of concatenated AES blocks.

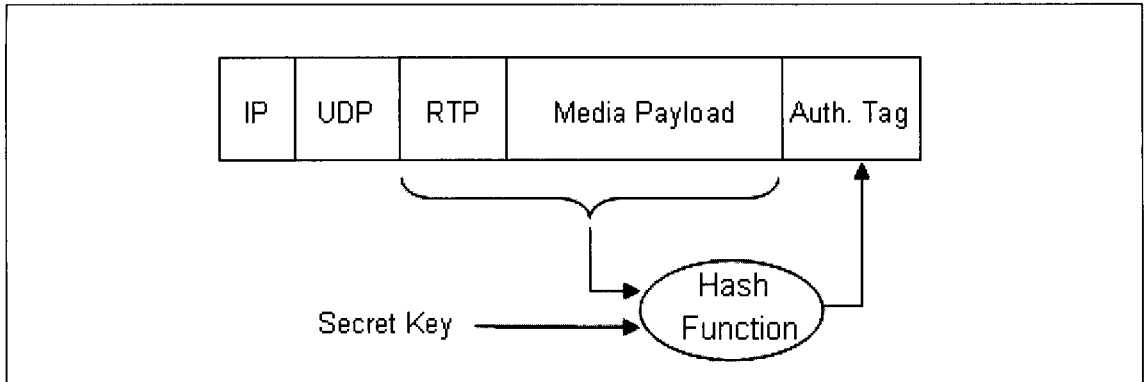
**Figure 9.5** SRTP Packet Encryption



## Message Authentications

An integrity service is obtained by running a one-way hash function on the message using a cryptographic key so that the receiver can ensure that the sender of the message possessed a secret key and that no party lacking that cryptographic key modified the message while in transit.

Figure 9.6 shows how the message authentication works overall. The one-way function, Hash-Based Message Authentication Code with Secure Hashing Algorithm 1 (HMAC-SHA1), is run over the header and payload with a secret key. The sender writes the HMAC-SHA1 hash into the authentication tag, and the receiver runs the same computation and checks its result against the tag. If the two do not match, the message authentication is said to fail and the packet is discarded.

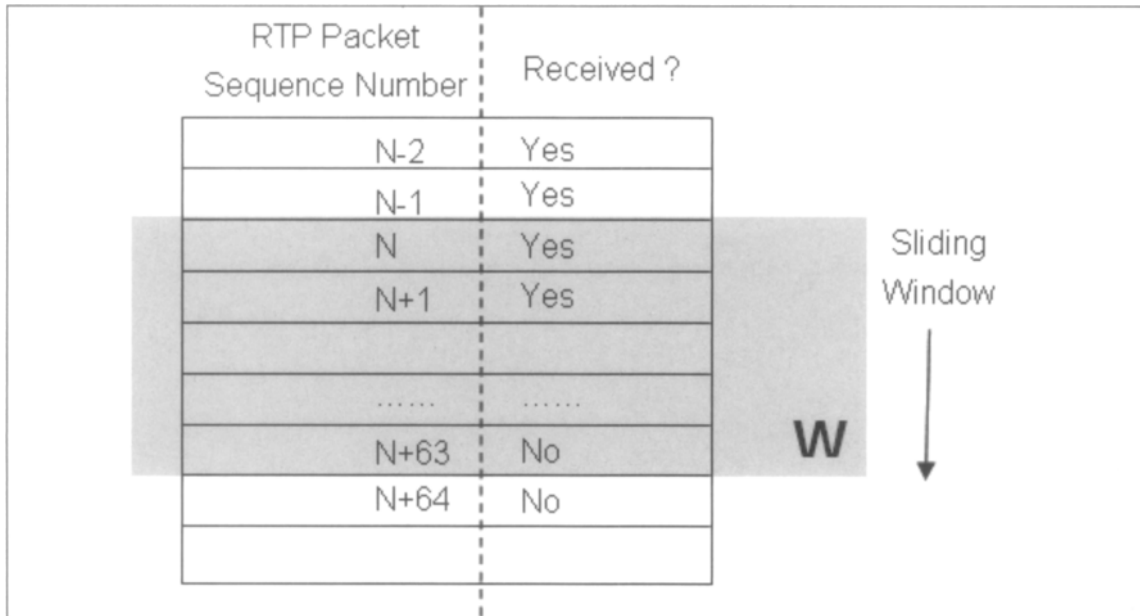
**Figure 9.6** SRTP Packet Authentications

## Replay Protection

SRTP packet-index determination deciphers the index of an invalid as well as a valid packet. There can be no integrity check until the authentication key is determined. SRTP replay protection is the first line of defense against packets sent by an attacker.

To counter replay attack, Rollover Counter (ROC) and sliding window are used. The 16-bit sequence number from the RTP header is added to the 32-bit SRTP ROC that is stored in the cryptographic context to get the 48-bit sequence number, which is the SRTP packet index for the particular packet. The packet index is encrypted with other parameters to generate key stream segments.

As Figure 9.7 depicts, a received packet index must fall within range of the sliding window, and its corresponding “Received?” bit must not be checked in order for the packet to be passed to the next processing step. If the packet does not meet the criteria, it is discarded. If an attacker chooses a sequence number at random, and the window size is 64, there is a 99.9 percent likelihood ( $1-64/2^{16}$ ) that the packet will be discarded before more computationally intense message authentication is applied.

**Figure 9.7** Sliding Window for Packet Replay Protection

## Summary

A brief tutorial on VoIP-related security standards was given, which focused on IETF standards. TLS and SRTP were presented as mainstream protocols to protect VoIP signaling and voice media, respectively. However, these protocols cannot operate alone. The supporting infrastructure, X.509 certificate profile, and S/MIME secured message format were introduced.



This Page Intentionally Left Blank

## Skype Security

Solutions in this chapter:

- Security

# Security

Security on any VoIP network is of considerable importance, given the forensic importance of a phone call. On a conventional VoIP network, as well as on a traditional telephone network, the following information is logged:

- The phone number that was dialed
- When the number was dialed
- When the call was connected
- The duration of the call
- When the call was disconnected

Skype does log some of the preceding information, but only the last 10 records, and a history is not kept as you might see in other VoIP solutions. This raises legal questions if business is conducted over a Skype connection. You need to decide what your security policy is on and whether logging call information is required. Some situations may require logging; others may not. Unauthorized use of Skype on a network can bring the following problems to the network administrator:

- Skype file transfers can cut both ways: unauthorized flow of company data out or the download of files that could be compromised with worms, viruses, and the like that have bypassed your firewalls and scanners.
- Skype file transfers will be caught by an antivirus solution that has an “auto-protect” capability.
- Skype users could consume a considerable amount of bandwidth if unchecked on the network. A large company with a T3 would not notice it right away, but a smaller company with a single T1 or a slower DSL circuit could easily have its WAN link overloaded by excessive VoIP traffic from Skype if all the users performed Skype calls.
- Skype may take over private resources to act as a supernode, even if the user is not actively using the Skype client. This is mitigated by a corporate firewall or DSL/cable router or other NAT device.
- The encryption of instant messaging can lead to exposure of private company data or other legal issues that cannot be monitored by a proactive staff.

Skype is up front about using your computer, or as up front as an end-user license agreement (EULA) can be. What is buried in the fine print of the EULA is the following article:

## Article 4 Permission to Utilize

**4.1 Permission to utilize Your computer.** In order to receive the benefits provided by the Skype Software, You hereby grant permission for the Skype Software to utilize the processor and bandwidth of Your computer for the limited purpose of facilitating the communication between Skype Software users.

**4.2 Protection of Your computer (resources).** You understand that the Skype Software will use its commercially reasonable efforts to protect the privacy and integrity of Your computer resources and Your communication, however, You acknowledge and agree that Skype cannot give any warranties in this respect.

(© Copyright Skype ELUA August 2005)

Please pay attention to these two sections of the EULA. The first one, Section 4.1 of the EULA, says that to use Skype, you give Skype right to use *your* computer, processor, and bandwidth to help facilitate communication between Skype users. In other words, you give approval to be one of those supernodes that we discussed earlier in this chapter. Your computer can be a supernode only if you are an open client on the Internet and do not have NAT protection.

The next section is also very important to administrators or anyone else with an interest in security. Section 4.2 of the EULA basically says that Skype will use *reasonable* efforts to protect your privacy and the integrity of your computer. This might be acceptable to the average home user, but most chief technology officers (CTOs) or other company management will not be very happy to see an application like Skype sitting on their networks with this type of license in play.

Several key properties are important to any discussion of security with respect to Skype. These properties are:

- **Privacy** How secure is your conversation using Skype?
- **Authenticity** Are you are reaching the person you think is at the other end?
- **Availability** Are Skype users always available when they are listed?
- **Survivability** If the Skype network takes a hit, can Skype keep working.
- **Resilience** Can the Skype user reconnect quickly when there is an outage?
- **Conversation integrity** Does Skype lose bits of the conversation?
- **System integrity** Does Skype work well with other applications?

These points are covered in detail in a security analysis paper written by Simson Garfinkel and available at [www.tacticaltech.org/files/Skype\\_Security.pdf](http://www.tacticaltech.org/files/Skype_Security.pdf). This paper is highly recommended for anyone with concerns about Skype's VoIP security model and methods.

The privacy of Skype is due to the encryption method Skype uses. Both voice calls using Skype and any instant messaging are encrypted, so there is a high level of privacy. This may change with government agencies looking to have the ability of monitoring traffic in a solution like Skype.

Most of the time, Skype is available when it should be. But Skype and many other VoIP vendors have ongoing issues with availability compared with the “old” telephone service. The telephone routinely has an uptime of 99.999 percent; people have become very used to this reliability, and they depend on this kind of uptime. Even under very adverse conditions, your POTS has a good chance of being up and working.

This is very unlike VoIP, where the connection can fail in a multitude of places. The gateway can fail, servers can fail, the ISP can fail—the list goes on. The telephone companies have had many years to work out how to build a redundant network, and the technology, although old style, is very robust. The VoIP companies are still working out standards, bugs, and billing issues as well as building a robust infrastructure. Being mostly decentralized, Skype has some advantages in terms of robustness, but Skype still has some weaknesses that administrators and users need to be aware of. The primary weakness at this point is the use of Skype servers for the username and password authentication. Without those, the Skype system fails.

VoIP systems such as Skype and others have a distinct advantage in the category of resilience. If the building or location in which you are using Skype loses its Internet connectivity, just go somewhere else with an Internet access point, and you are back in business. No mess, no fuss, Skype will simply work again. This is in contrast to the traditional phone system, where the numbers are generally not portable, so if the building phone system fails, you lose your phone connectivity on that number until the phone company can reprogram switches and their network or you can forward the phone number to a new number somewhere else. Larger companies and corporations may have multiple and redundant Internet connections and allow for rerouting adding to the reliability of a Skype type of solution.

We now know through the analysis by Tom Berson of Anagram Labs that Skype ensures the integrity of the voice call. Administrators or engineers can now compare Skype to products from other vendors to see who provides the best solution.

Skype is a closed protocol, but there has been some documentation on a few parts of the process by which Skype makes connections. Skype's supernodes carry media stream traffic at times, which has the possibility of being a security risk, since the call is traversing an unsecured server. Remember, the supernode is any computer with sufficient RAM, CPU, and a public IP address not protected behind a NAT device.

## Understanding the Basics...

### Avoid Skype Call Relays and File Transfer Relays

If you do not want to relay your Skype calls or Skype file transfers, configure your network to avoid this.

## Blocking Skype

To block Skype on their networks, administrators will, at best, find it difficult, since Skype, like Kazaa, was designed to intentionally get around the normal network security blocks. One of the few ways is to look at HTTP traffic and make sure that the headers and information are really HTTP traffic and not something like Skype just using port 80 to take advantage of the open port on most networks. Some vendors, such as BlueCoat and Verso, claim they can block Skype traffic. BlueCoat and Verso are enterprise-level solutions and therefore very expensive security appliances that are designed for large networks. BlueCoat recommends blocking Skype by preventing download of the Skype application and using protocol filters on the BlueCoat proxy appliance. BlueCoat provides a free white paper titled “Best Practices for Controlling Skype within the Enterprise” available for download from its Web site, [www.bluecoat.com/resources/resourcedocs/whitepapers.html](http://www.bluecoat.com/resources/resourcedocs/whitepapers.html).

Verso attempts to block Skype by matching Skype communication patterns referred to as signatures. The Verso appliance has an active client that can receive updates to the appliance’s “black list” and algorithms used to block internet traffic, such as Skype. Additional technical information on blocking Skype will be discussed in Chapter 11.

## Firewalls

A security best practice to start with is to block the use of the high-numbered ports on your firewall. Also, taking the approach of blocking everything outbound and allowing only what you need is highly recommended, with the understanding that it will mean more work for the security or network administrator. This approach is becoming much more common on firewalls, so if you have problems with your Skype connection, check your firewall to make sure it is not configured to block all traffic unless explicitly allowed on the outbound side.

## Downloads

If you have the capability to block certain downloads, you can block the Skype executables (`skype.exe`) from being downloaded. Using group policies can help prevent the installation of

the Skype application on an Active Directory domain or prevent the execution of the executable. Of course, on a non-Microsoft client such as Apple's OS X, Active Directory control is pretty much a nonstarter, so you would need to find another way to lock down the OS X operating system.

We cannot suggest strongly enough that you have a policy in place at the company detailing acceptable software use, spelling out definitions of "good" software and "bad" software. Such a policy provides some cover for the company in case legal issues arise with the unauthorized use of Skype.

## Software Inventory and Administration

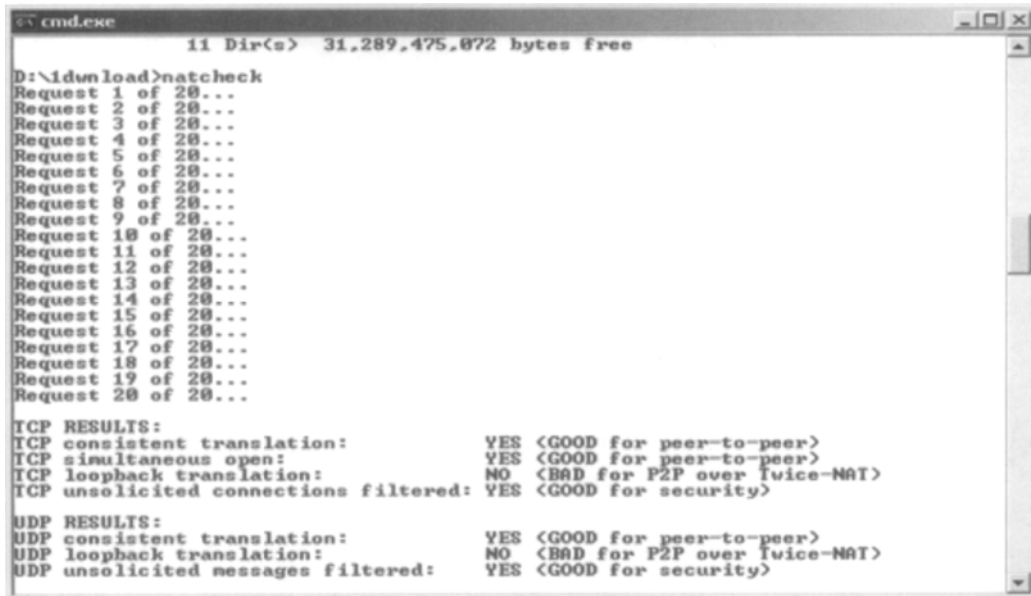
Another method to block the use of Skype is to use a software inventory solution that is typically found in larger organizations. Companies often scan the systems looking for specific software packages like port scanners and other potentially malicious tools and delete them as a part of a good security plan. You could do the same to control the use of Skype inside a corporation with a software inventory and distribution solution like SMS, Radia, and others.

You could also use a script that attaches to all your remote machines, log on as an administrator, scan for unapproved applications, and delete or disable those applications. This practice would work for smaller companies and those with non-Windows operating systems that may not have a software distribution solution.

## Firewalls

Skype uses a modified form of the STUN protocol to deal with security like NAT on a firewall. Restricted ports are dealt with using random ports during the installation and use of HTTP and HTTPS. Between the use of ports 80 and 443 and the random ports, Skype can work around restricted firewalls.

To see if you can pass through your firewall with Skype, you can use a free program called NAT Check, which can be found at <http://midcom-p2p.sourceforge.net/>. This NAT checking program is not from Skype, but Skype suggests its use to verify your network capability with Skype. The NAT Check screen shown in Figure 10.1 is from a typical home network showing a good result:

**Figure 10.1** Results of a NAT Check of a Typical Home Network


```

cmd.exe
11 Dir(s) 31,289,475,872 bytes free

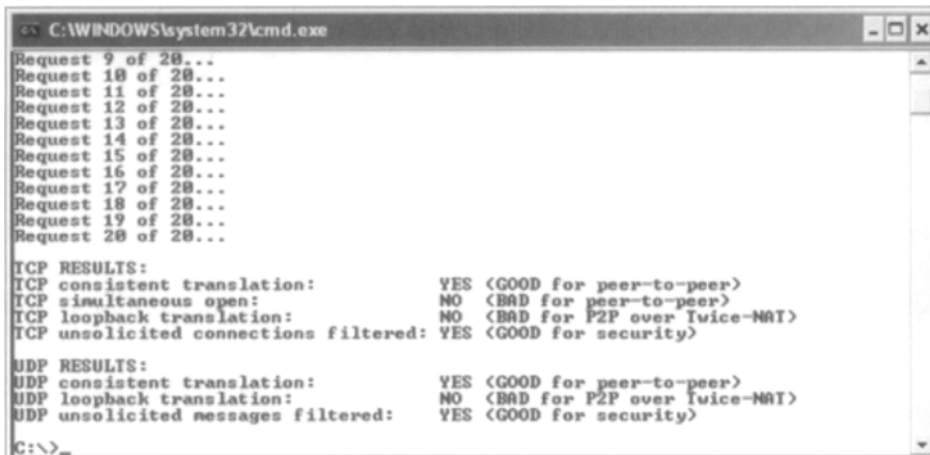
D:\download>natcheck
Request 1 of 20...
Request 2 of 20...
Request 3 of 20...
Request 4 of 20...
Request 5 of 20...
Request 6 of 20...
Request 7 of 20...
Request 8 of 20...
Request 9 of 20...
Request 10 of 20...
Request 11 of 20...
Request 12 of 20...
Request 13 of 20...
Request 14 of 20...
Request 15 of 20...
Request 16 of 20...
Request 17 of 20...
Request 18 of 20...
Request 19 of 20...
Request 20 of 20...

TCP RESULTS:
TCP consistent translation:      YES <GOOD for peer-to-peer>
TCP simultaneous open:         YES <GOOD for peer-to-peer>
TCP loopback translation:      NO <BAD for P2P over Twice-NAT>
TCP unsolicited connections filtered: YES <GOOD for security>

UDP RESULTS:
UDP consistent translation:     YES <GOOD for peer-to-peer>
UDP loopback translation:      NO <BAD for P2P over Twice-NAT>
UDP unsolicited messages filtered: YES <GOOD for security>

```

The NAT Check test shown in Figure 10.2 was over a public wireless access point and using a Cisco VPN SSL client showing a good result:

**Figure 10.2** Results of a NAT Check Test over a Public Wireless Access Point


```

C:\WINDOWS\system32\cmd.exe

Request 9 of 20...
Request 10 of 20...
Request 11 of 20...
Request 12 of 20...
Request 13 of 20...
Request 14 of 20...
Request 15 of 20...
Request 16 of 20...
Request 17 of 20...
Request 18 of 20...
Request 19 of 20...
Request 20 of 20...

TCP RESULTS:
TCP consistent translation:      YES <GOOD for peer-to-peer>
TCP simultaneous open:         NO <BAD for peer-to-peer>
TCP loopback translation:      NO <BAD for P2P over Twice-NAT>
TCP unsolicited connections filtered: YES <GOOD for security>

UDP RESULTS:
UDP consistent translation:     YES <GOOD for peer-to-peer>
UDP loopback translation:      NO <BAD for P2P over Twice-NAT>
UDP unsolicited messages filtered: YES <GOOD for security>

C:\>_

```

You can also use the Display technical call info option to help troubleshoot your Skype connection to see if it is being relayed or not. For more information on how to Display technical call info, see Chapter 11.



## Proxy Servers

You can configure Skype to use the proxy server to gain access to the Internet. The address of the proxy server may not be easily determined by setting Internet Explorer's **Connection LAN Settings** option to **Automatically detect settings**. If you would like to determine the actual proxy server name to enter in the Skype configuration dialog, simply type **netstat - v** at a command prompt. You will notice many Internet connections through the same port, most likely 8080 or 8088, from a machine on your network. That machine is most likely to be your proxy server. If you have more than one proxy server on your network, every time you log in, you may get a different proxy server with the automatic configuration. Manually defining the proxy server allows a network administrator to configure a single proxy server for Skype traffic and have more control over Skype traffic.

## Embedded Skype

Skype has signed contracts with some vendors to embed its client software into various products. These products are phone sets, small office/home office (SOHO) routers, and other network devices. Skype-enabled devices can have adverse effects on the security of your network if you do not know that “hidden” clients are in various pieces of hardware. So the administrator must not only manage the network side but also be aware of the hardware being brought into the network. Be sure to understand the devices on your network and what they do to prevent unauthorized devices being used on your network.

## A Word about Security

Many companies with an IT staff are or will ask, “Is Skype secure?” While writing this book, Skype released a Security White Paper by Tom Berson of Anagram Laboratories, a well-known cryptographer who outlines how Skype uses encryption. This paper may be found at <http://www.skype.com/security/>.

Companies that are security aware and have a good security posture will question whether any voice software, or softphone as this technology is often referred to, is secure enough to discuss anything from human resource issues, mergers and acquisitions to participating in conference calls. Berson's paper should calm the fears of many security practitioners. However, many corporations will still wonder about the security of a softphone on a computer that is mobile or in a remote location, such as an employee's home, hotel, or hotspot. For example, corporations will be concerned whether these mobile devices could get compromised with a recording device and record the voice calls and upload them to a Web site to be listened to by the masses or sold to the competition.

These concerns are valid for any company and might be even for the typical home user, but one of the guiding rules of security is to “classify your data.” Companies must (1.) decide what data, or in this case voice transmissions, needs to be protected; (2.) determine

which devices to use and when to use them; and (3.) set rules or policies that the employees will follow. If you are one of these companies, you should consider developing a policy that sets the classification of voice calls and what the proper device or location should be used when making any voice calls that are “sensitive” in nature and educate your employees to strictly follow this communication policy. Points to consider in this policy include:

- The security of any mobile device capable of voice (cell, Pocket PC, laptop)
- What communication should occur on a landline (i.e., mergers)
- What communication can occur on a soft phone or Skype (i.e., conference calls)
- Where you should or should not be when having conversations on a cell phone or soft phone (airport)

I am amazed at how many people I have encountered in my travels whose companies still do not deploy a personal firewall or encryption on a laptop in today’s world of worms, theft, and malicious activity targeted at Microsoft Windows users. It is not Microsoft’s fault that these users are compromised; it is the fault of companies that are not practicing defense in depth and securing their assets. Many companies have a “Deny all unless explicitly allowed” policy. Under this policy, these companies do not allow Skype because they have not specifically approved Skype, and yet, they deploy thousands of laptops with very sensitive data without encryption. If I were interested in data about a company, I sure would not be trying to capture Skype traffic. I would go after a laptop, since many companies put asset labels on their laptops indicating the company name or users have business card tags on their laptop bags. Social Engineering 101: Users tell you a lot without saying a word.

Regarding sniffing or capturing a Skype call, only a few governments and companies could afford the amount of storage required to capture the sheer quantity of Skype traffic going through an ISP, for example. The weak point is the client that is the laptop or desktop. If proper security measures were taken to protect these devices and if these measures were assessed frequently, this risk should be minimal. Companies as a regular practice install and monitor antivirus and personal firewall software. Many companies also use an intrusion detection solution on the laptops they issue to employees to protect users from open networks they may connect to (e.g., a hotel or their homes with a DSL or cable broadband connection).

Personally, if I were an IT security manager who had read Tom Berson’s Security White Paper and was confident that my laptops were properly secured, I would approve the use of Skype in my corporation as long as I had a “communication policy” covering what I could and could not discuss over Skype or other communication devices. I have overheard some rather interesting cell phone conversations in airports, coffee shops, and airline lounges during which businesspeople discuss very sensitive information. To me, this practice is a greater risk than using Skype.

Besides, if an issue occurred or was discovered with Skype, I could always issue a “Stop using Skype immediately” memo or e-mail to all employees while the issue was under investigation and until the security was reconfirmed. I could also use a software distribution tool or script to disable Skype fairly quickly on all computers that had Skype loaded. A company with valid security concerns could use Skype to reduce its communication costs and enable its employees by adding Skype versus replacing any existing solution like cell phones or landlines. The goals of using Skype in these situations are to reduce the amount of costly cell minutes and lower home office or hotel telephone bill expenses. As long as a company does not replace the traditional way of communicating, Skype could be added with little risk when deployed with a good communication policy.

## Skype Firewall and Network Setup

### Solutions in this chapter:

- A Word about Network Address Translation (NAT) and Firewalls
- What You Need to Know about Configuring Your Network Devices
- Ports Required for Skype
- Using Proxy Servers and Skype
- How to Block Skype in the Enterprise

# A Word about Network Address Translation and Firewalls

When the Internet began, the creators didn't envision the type of growth that we are experiencing today. During the last 10 years, the number of hosts on the Internet increased by more than a factor of 50.<sup>1</sup> In order for each Internet device, or host, to communicate on the Internet, it must have a unique internet protocol (IP) address. The addressing scheme for the Internet allowed for billions of IP addresses, but now most of them are allocated.

The Internet's popularity results in a maximum number of available IP addresses. Homes and offices around the world are now connecting many hosts at a single location and it is not possible for every single device to have its own public IP address. To increase the number of addresses available, a new standard called IPv6 has been developed. Until IPv6 is finalized, other methods are needed to allow for the sharing of public addresses among more systems. The most effective solution is called network address translation (NAT), defined in the request for comments 1631 (RFC 1631).

NAT is a special type of router that has several different implementations. One popular method of implementation allows for the use of special, unroutable IP addresses on private or internal networks. The private addresses are translated to a public host address, which allows communication over the Internet. Three blocks of the unroutable, or private, IP addresses are defined in RFC 1597 and RFC 1918. The private addresses are reserved by the Internet Assigned Numbers Authority (IANA), the organization that is responsible for all IP addresses. The private addresses are represented in Classless Inter-Domain Routing (CIDR) notation as:

- 10.0.0.0/8
- 172.16.0.0/12
- 192.168.0.0/16

These address blocks cannot communicate directly with public addresses on the Internet and must be translated.

NAT utilizes a mechanism in the Transmission Control Protocol/Internet Protocol (TCP/IP) stack called multiplexing to enable these private addresses to establish communication over the Internet. Multiplexing makes it possible for a single device to establish and maintain several simultaneous connections with one or more hosts using different TCP and User Datagram Protocol (UDP) ports. This architecture allows an implementation where a single public IP address can service the needs of an entire network of hosts, a many-to-one relationship.

NAT routers keep a table of internal address and port combinations, as well as the public (global) IP address and port used to establish the remote connection. External hosts do not see the internal address, but instead use the public IP address to respond to requests. When responses

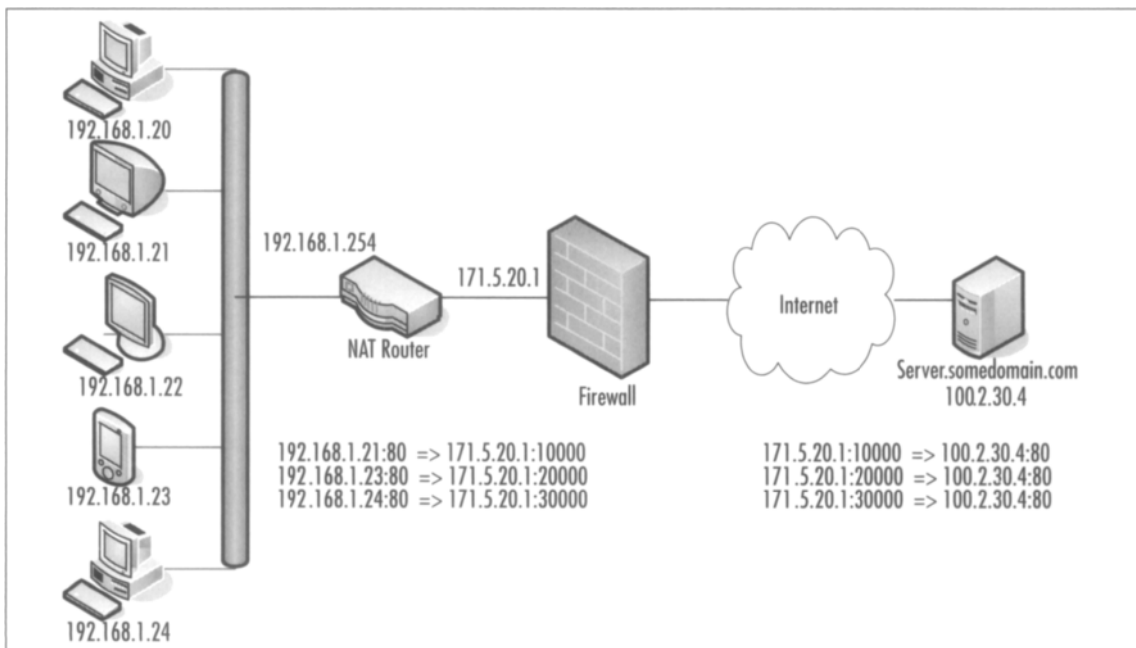
are sent back to the external IP address and port of the NAT router, it translates the response and relays it back to the internal address and port that originated the request.

Firewalls are Protocol layer rules engines. A firewall can be hardware or software based, and many routers include basic firewall functionality as an additional feature. A typical firewall provides a list of rules that are evaluated sequentially against the header data in the packet being processed. As each rule is examined against the packet header, the packet will be blocked, or the next rule will be evaluated. This process continues until the packet is blocked or all rules have been examined, in which case the packet is forwarded.

A proxy server is similar to a firewall, but it works at the Application layer. Proxy servers have packet-filtering features. Packet filtering allows examination of the actual data being transmitted within the packet itself. Packet filters are available on Windows XP, Windows 2000, and Windows Server 2003 products as part of the advanced features of the TCP/IP configuration. However, because Skype encrypts the data it transmits, packet filtering is an ineffective means of managing Skype traffic. Proxy servers handle the requests for each protocol, whereas firewalls merely forward the traffic. If the proxy server is disabled, no traffic is allowed to pass. If you disable a firewall, you are turning off all rules processing and allowing all traffic to pass, which is not a recommended practice.

In Figure 11.1, a single external IP address is exposed to the Internet. When hosts on the private network make a request, the following occurs:

**Figure 11.1** A Single External IP Address Exposed to the Internet



1. The host initiates a request for the remote destination address and port.
2. Since the address is remote, the router handles the request.
3. The NAT router adds the entry for the internal host IP address and port to the translation table.
4. The NAT router assigns a new port on the external interface IP address for the internal client and adds it to the translation table.
5. The NAT router then initiates a connection to the remote host on the external network, through the firewall, substituting a new source port and IP address in the IP packet header.
6. The remote host responds to the request to the external address and port.
7. The firewall compares the IP address and port with the list of firewall rules. If the IP address passes the IP address test, the port is checked. For Skype, this would be a UDP port, or if UDP is blocked, TCP port 443 or TCP port 80.
8. The router uses the translation table to translate the response from the remote host from the external address and port to the original internal address and port of the host that initiated the request.

## Home Users

We strongly recommend that home users obtain a basic peer-to-peer-friendly, broadband router with firewall capabilities. In addition to a hardware-based router/firewall, you should always use a software-based firewall on each client machine. Windows XP has built-in firewall software that is enabled by default after you install Service Pack 2. Other options for software-based firewalls include products by McAfee, Symantec, and Zone Alarm. Skype should work right out of the gate on most home networks without requiring any further configuration. For home users, no modification is needed.

Later in this chapter, we discuss how to improve the quality of the communication, which could require minor configuration settings on your firewall.

## Small to Medium-Sized Businesses

Small to medium-sized businesses must use discretion to determine whether to use a simple implementation, as discussed for home users, or to provide a more robust firewall solution, such as the Symantec Firewall/VPN Appliance, Cisco Pix, or other SOHO solution. Regardless, we suggest that small and medium-sized businesses use software-based firewalls on each network client to provide an additional layer of security.

# Large Corporations

Larger corporations must ensure that the many routers used on the LAN allow Skype traffic over UDP to pass to other clients on the LAN if they want to use Skype effectively.

To better understand how Skype communicates, you need to get a picture of how the Skype network is organized. There are three basic roles in the Skype communication infrastructure. The roles consist of the following:

- Skype client or peer
- Supernodes
- Login servers

A Skype client is your computer running the Skype software. Supernodes are just Skype peer nodes that are not behind a restrictive firewall or a NAT router, and which therefore have unrestricted access to the Internet. Supernodes come and go depending on the needs of the overall network. Any Skype client node can become a supernode if it is not behind a NAT router or blocking firewall and has sufficient CPU and bandwidth capacity.

## Understanding the Basics...

### Avoid Becoming a Supernode

To prevent a Skype client from becoming a supernode, all that is required is for the client to be behind a NAT router or a restrictive firewall (hardware or software).

If a Skype client is behind a NAT router or firewall, the Skype client cannot establish a direct connection to another peer. In these situations, the supernode peers act as relaying agents to help Skype peers behind firewalls or NAT routers establish connections to other peers that are behind firewalls or NAT routers. Skype peers tend to connect to supernodes that are in relative proximity to their locations on the Internet. By connecting to nearby supernodes, Skype reduces utilization and decreases the latency in response times, thus providing a fast and scalable communication network.



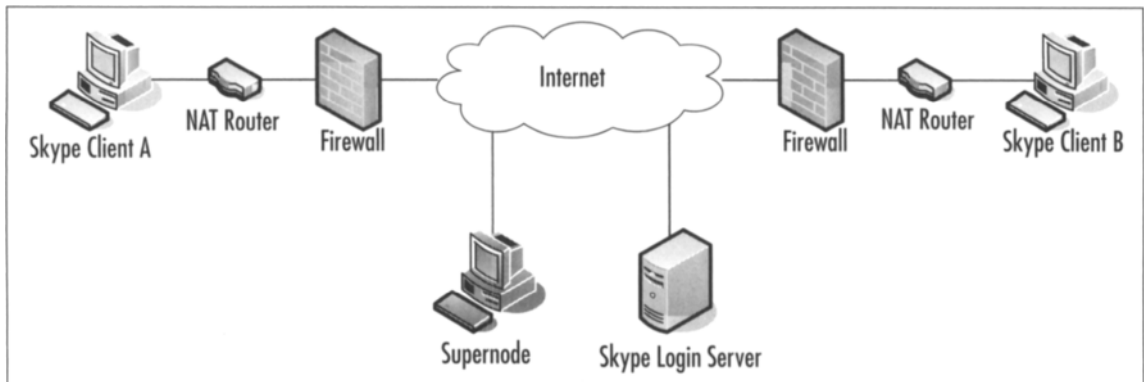
## Understanding the Basics...

### Avoid Relayed Calls or File Transfers

To prevent a Skype call or file transfer from being relayed, the firewall or NAT router must allow a P2P connection.

When Skype starts, it determines whether the client is behind a firewall or NAT router. If there is a firewall or NAT router, Skype determines the best method for communication via the firewall or NAT router using various UDP mechanisms. If no UDP ports are open, Skype will attempt to use TCP port 80, then TCP Port 443. Refer to the basic topology to get a picture of what happens next. Figure 11.2 diagrams communication between Skype Client A and Skype Client B.

**Figure 11.2** Communication between Skype Client A and Skype Client B



After Skype Client A determines how to navigate the firewall or NAT router, Skype contacts a supernode peer from its supernode list to attempt to log in. If for some reason there are no supernodes listed for the client, the client attempts to log in to the Skype login server. Once the client logs in, the supernode list may be updated with the current active list of supernodes.

Once the connection is established, you can place a call, begin to instant message, or transfer a file. The call starts with a search of the Skype Global Index to locate the target Skype user. Skype Client B will follow the same process to log in. If the target user, Skype Client B, is behind a firewall or non-P2P-friendly device, the supernode acts as the liaison to direct traffic

from Client A to Client B and vice versa, thus allowing Skype Clients A and B to find and communicate with each other using Skype Clients as relay nodes.

## What You Need to Know about Configuring Your Network Devices

We'll now discuss configuring network devices in various environments.

### Home Users or Businesses Using a DSL/Cable Router And No Firewall

To use Skype typical home users will not need to configure anything on their DSL/Cable routers with or without wireless unless they have an older DSL/Cable router that is not P2P friendly. Running NAT Check, discussed later in this chapter, and enabling the Technical Information in Skype's Advanced options will help you determine if your router is capable of a Skype P2P connection.

### Small to Large Company Firewall Users

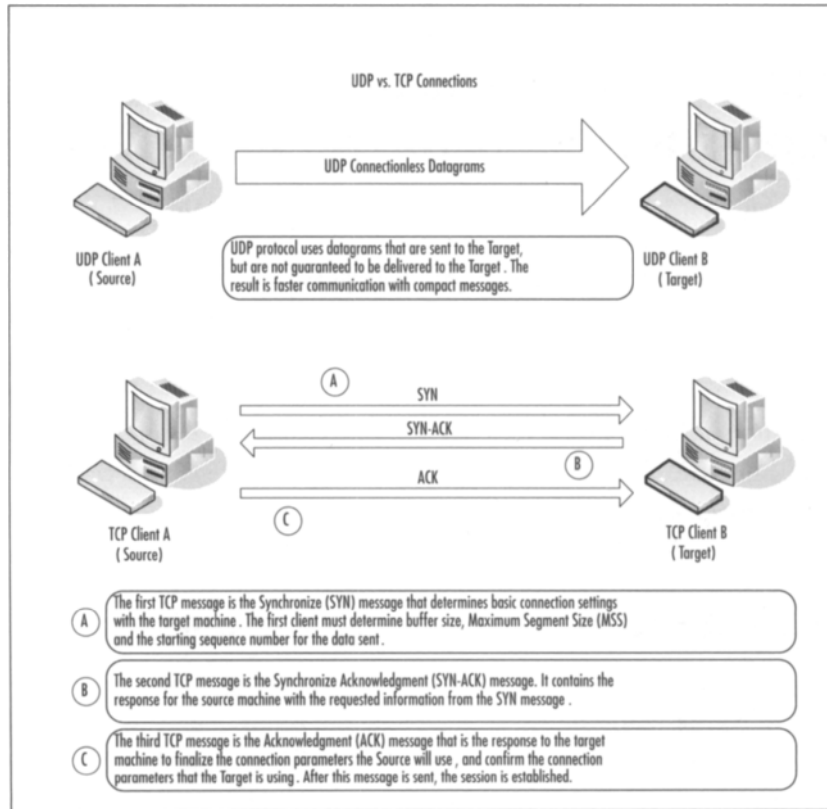
To provide the best performance on your network, you will need to tune your network to optimize handling of the Skype traffic. Skype leverages the use of UDP extensively to provide the best possible connection quality with its peers. The NAT translation table is a volatile table that ages old connections to free up room in the routing device's buffer for new connections.

It is important that the NAT routers hold the definition for UDP datagrams sent from the internal network for at least 30 seconds. The delay ensures that there is ample time provided for a response to the original request initiated from the client. The translation table should consistently map the internal host address and port number for UDP traffic in order to be reliably translated from the external address and port used to establish the communication. UDP has very little overhead, but it is prone to loss because it is not guaranteed to be delivered to the destination. Because it has little overhead, UDP is a faster method for communications.

### TCP and UDP Primer

TCP requires a three-way handshake to verify that data reaches its destination, whereas UDP just sends that data, and does not require acknowledgment of delivery (see Figure 11.3). Because UDP does not require all of the overhead in the message structure, the messages are smaller, and UDP headers are always the same size. The UDP message structure makes the delivery much faster. Establishing communication sessions over TCP takes three trips instead of the one trip UDP requires. The TCP headers are much larger and vary in size, so there is more overhead to process each TCP message as well.

Figure 11.3 TCP and UDP Connections



## NAT vs. a Firewall

Remember, a NAT device just translates many internal IP addresses to one or more external routable Internet addresses. A firewall can also provide NAT functionality and includes additional intelligence to apply rules to the traffic that passes through the firewall. NAT devices such as a DSL/cable router may or may not have firewall functionality.

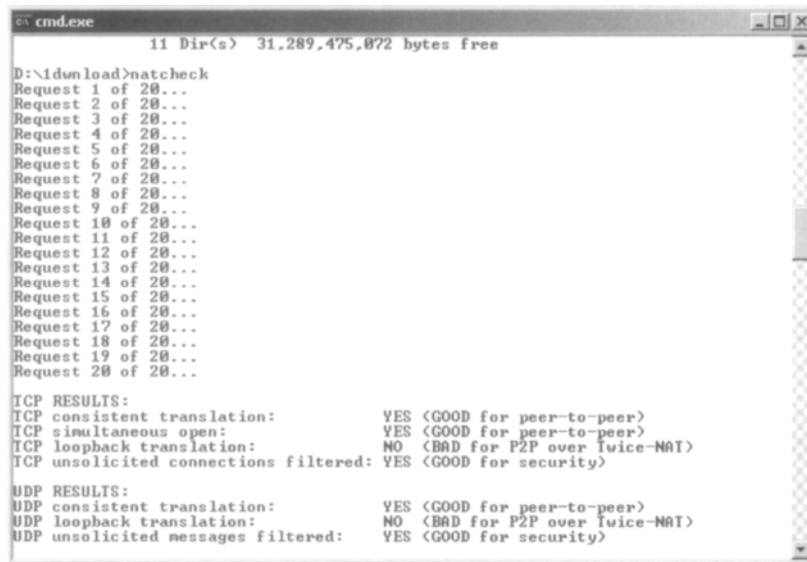
Skype also recommends that the firewall or Internet gateway support IP packet fragmentation and reassembly. Fragmenting the packets allows the stream of data to be broken into smaller packets that can be sent simultaneously over multiple ports to the destination. This packet fragmentation can dramatically improve quality and performance by allowing higher throughput, which in turn allows for more effective bandwidth. Some firewalls detect this type of parallel UDP communication incorrectly as port scanning and will block the host traffic. The result could be a degradation of Skype performance.

Skype references a tool called NAT Check by Bryan Ford. The tool can be located at <http://midcom-p2p.sourceforge.net>.

The tool can be used to determine how P2P friendly your network is. Ford has described the details on UDP communications over the Internet using NAT in an Internet draft. The paper is located at <http://mirrors.isc.org/pub/www.watersprings.org/pub/id/draft-ford-natp2p-00.txt>.

Figure 11.4 shows the output from NAT Check for a relayed call.

**Figure 11.4** Output from a NAT Check for a Relayed Call



```
cmd.exe
11 Dir(s) 31,289,475,872 bytes free

D:\download>natcheck
Request 1 of 20...
Request 2 of 20...
Request 3 of 20...
Request 4 of 20...
Request 5 of 20...
Request 6 of 20...
Request 7 of 20...
Request 8 of 20...
Request 9 of 20...
Request 10 of 20...
Request 11 of 20...
Request 12 of 20...
Request 13 of 20...
Request 14 of 20...
Request 15 of 20...
Request 16 of 20...
Request 17 of 20...
Request 18 of 20...
Request 19 of 20...
Request 20 of 20...

TCP RESULTS:
TCP consistent translation:      YES (GOOD for peer-to-peer)
TCP simultaneous open:         YES (GOOD for peer-to-peer)
TCP loopback translation:      NO (BAD for P2P over Twice-NAT)
TCP unsolicited connections filtered: YES (GOOD for security)

UDP RESULTS:
UDP consistent translation:      YES (GOOD for peer-to-peer)
UDP loopback translation:      NO (BAD for P2P over Twice-NAT)
UDP unsolicited messages filtered: YES (GOOD for security)
```

## Ports Required for Skype

We'll now discuss the ports that are required to use Skype.

### Home Users or Businesses Using a DSL/Cable Router and No Firewall

To use Skype, typical home users will not need to configure anything on their DSL/cable routers or within the Skype software.

### Small to Large Company Firewall Users

Skype uses UDP and TCP to communicate with other Skype clients. UDP is primarily used to establish connectivity and perform global directory searches. If the UDP ports above 1024 are open outbound, and you allow UDP replies to return through the firewall, you can improve Skype's voice quality and performance. Opening UDP ports could allow peers on your network to connect more efficiently by providing closer neighbors on the P2P network, thus reducing

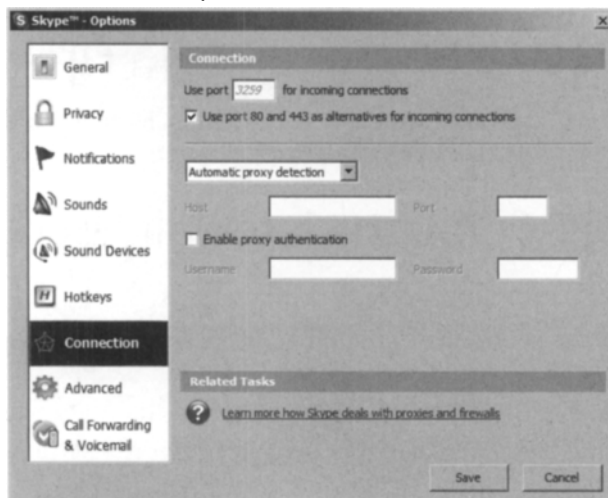
latency and improving call quality. Allowing more UDP ports also prevents internal contention of port translation in the NAT translation table.

In a perfect world, all outgoing TCP ports would be open through the firewall or Internet gateway. If it is not possible to open all outgoing ports, TCP port 80 should be opened. Using port 80 is a standard practice. When Skype attempts to log on, it first tries to connect using random ports. If Skype cannot connect, it attempts to connect via port 80. If port 80 cannot be opened, Skype attempts to use port 443. There is no guarantee that Skype will work through port 80 if the firewall or proxy server is restricting traffic to the HTTP. By restricting traffic to HTTP, the proxy server or firewall can scan the packets to ensure that the data is actually HTTP data. Skype does not use HTTP and will not function correctly through port 80 if traffic is restricted to HTTP traffic. If you receive errors #1101, #1102, or #1103 the firewall may be blocking port 80.

When Skype installs, it will select a random UDP port to communicate. This port setting is found in the Connection tab under Options (see Figure 11.5) and is an adjustable setting and stored in the shared.xml file on each computer and could be set the same for all users of Skype. If you want to avoid relayed Skype calls and relayed file transfers, you can open up the UDP port on your firewall that is specified in Skype to allow for better voice call quality and faster file transfers.

Understand that opening these UDP ports changes the normal corporate security policy, and proper approval and risks associated with opening anything on your firewall should be weighed prior to opening these settings. Discuss this issue thoroughly with your information security team on the impacts and what additional layers of security could be implemented to mitigate any risks, such as enabling a client-side personal firewall solution discussed earlier in this chapter. You could allow TCP and/or UDP inbound on the ports listed in Skype options for all clients internal to the firewall. If necessary, Skype will use TCP ports 80 and 443, respectively, to communicate with other Skype peers, and this will create relayed Skype calls and slow file transfers.

**Figure 11.5** Skype Connection Options



## Skype's Shared.xml file

In a larger network, you can control the port for incoming connections by modifying Skype's shared.xml file in the following location:

- <Drive>\Documents and Settings\<UserName>\Application Data\Skype folder

Using a text editor, find the **<ListeningPort>nnnn</ListeningPort>** entry of the shared.xml file, where 'nnnn' is the random port number that Skype chose when it was initially installed. By configuring all users to use the same UDP port, you can improve the quality of Skype conversations by opening a single inbound UDP port, if your network security policy permits this. If the traffic inbound on that port is high, you could logically segment the traffic by setting different groups of users to use a specific UDP port and opening multiple UDP ports inbound, while still maintaining some control over what ports are opened and to whom. Visit Dan Douglass's Web site at the following URL for scripts and utilities to help modify the shared.xml setting in a business environment: [www.codehatchery.com/skype.html](http://www.codehatchery.com/skype.html).

## Microsoft Windows Active Directory

In a typical Windows Active Directory-based enterprise, with clients running Windows XP Service Pack 2, you can set a Group Policy that allows you to enable the Skype traffic through the Windows Firewall on all client machines with little effort. This can be achieved via the following steps:

1. Open the **Group Policy Object Editor** console on the Active Directory Domain controller.
2. Locate the **Group Policy** setting found in **Computer Configuration\Administrative Templates\Network\Network Connections\Windows Firewall\Standard Profile** (see Figure 11.6).
3. Select the Policy Setting for **Windows Firewall** to enable the **Define program exceptions** policy (see Figure 11.7).

Figure 11.6 The Group Policy Setting

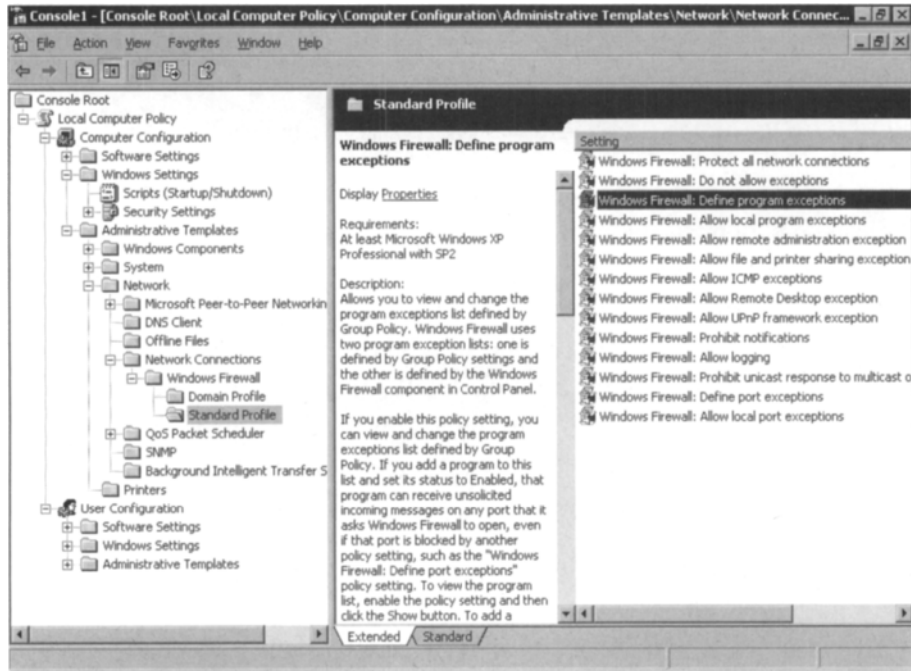
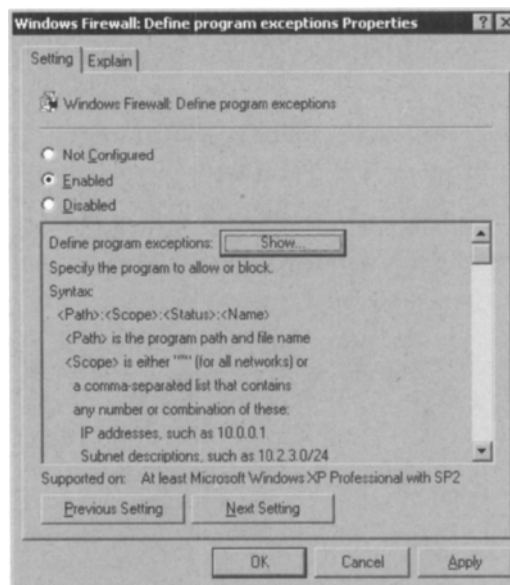


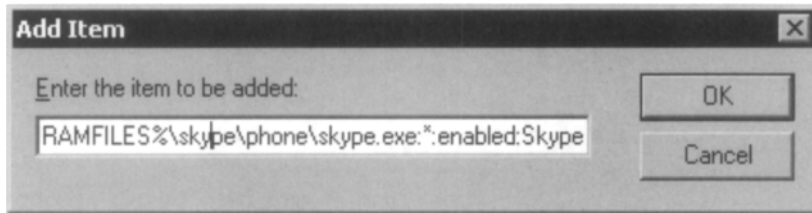
Figure 11.7 Enabling the Define Program Exceptions Policy



- Next, click the **Show** button that was enabled by the previous step.

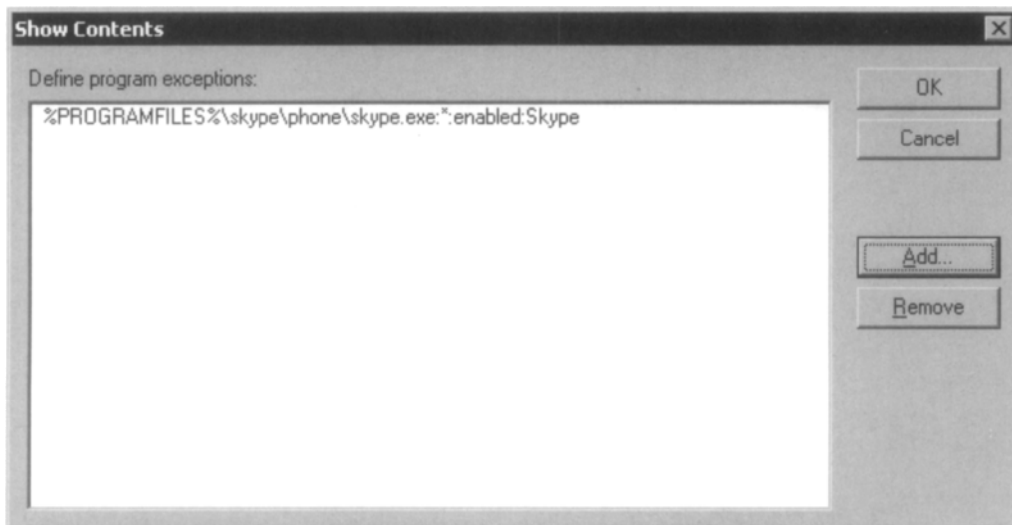
5. Add a definition for a Program Exception as `%PROGRAMFILES%\skype\phone\skype.exe:*:enabled:Skype` and then click **OK** (see Figure 11.8).

**Figure 11.8** Adding a Definition for a Program Exception



6. Click **OK** to close the Show Contents dialog box, then click the **OK** button to close the Windows Firewall: Define program exceptions Properties dialog box (see Figure 11.9).

**Figure 11.9** The Define Program Exceptions Dialog Box



7. Allow time for the Group Policy to be refreshed. The time varies depending on the network settings. Allowing exceptions for Skype and opening up the recommended ports make it easier for Skype to establish reliable communications outside of your network. Other products, such as Norton Internet Security, McAfee Firewall Pro, and Zone Alarm Pro, have similar functionality. Visit Skype's Web site at [http://web.skype.com/help\\_firewalls.html](http://web.skype.com/help_firewalls.html) for the specific configuration of your product.



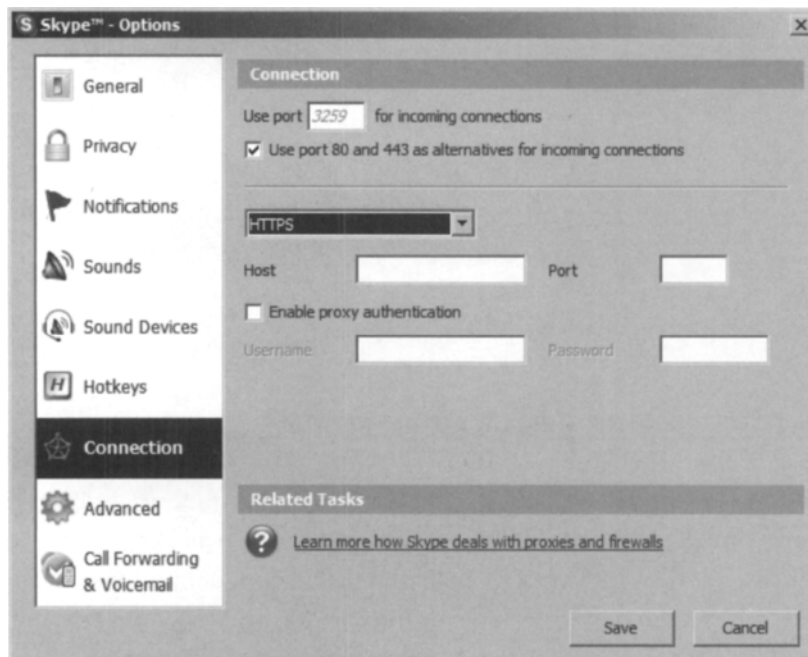
The same option can also be manually configured on each workstation in the enterprise by using the Windows Firewall applet in Control Panel.

1. Open **Control Panel** and double-click the **Windows Firewall** icon.
2. Click the **Exceptions** tab.
3. Tick the box next to **Skype**.

## Using Proxy Servers and Skype

Many popular proxy servers are available on the market today. Skype supports HTTPS, SSL, and SOCKS5 proxy standards. Skype can optionally include authentication over proxies if the proxy server requires it (see Figure 11.10). On Windows clients, Skype automatically uses the connection settings in Internet Explorer to identify the proxy settings that may be defined for that user on that computer. It is possible for the user to set Skype to use a manual configuration in the **Tools** menu, **Options**, and **Connection** tab settings. See Chapter 10 “Skype Security” for tips on identifying your proxy server information using the netstat utility.

**Figure 11.10** Skype Proxy Server Options



If you are using a SOCKS5 proxy server, it must allow unrestricted connections to the ports discussed in the “Ports Required for Skype” section of this chapter. Most proxy server solutions

provide packet-filtering features. As previously mentioned, enabling packet filtering and restricting traffic over port 80 to only HTTP could cause communication problems for Skype.

## Wireless Communications

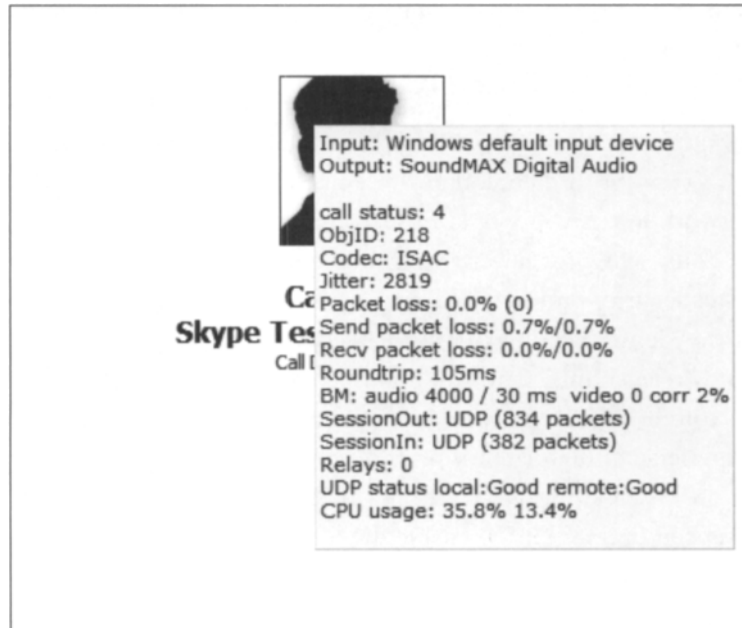
Many companies implement a wireless network, preferably using 802.11G, that directly connects to the Internet. If you want to then connect to company resources, you would VPN back into the corporate network just as you would from home or a hotel over the wireless network. The wireless network could allow for fewer restrictions on traffic for wireless clients while still allowing for stricter security on the wired devices. You should also read the benchmark documents located at the Center For Internet Wireless Benchmarks at the following url: [http://cise-security.org/bench\\_wireless.html](http://cise-security.org/bench_wireless.html). There you will find valuable information on implementing a wireless infrastructure in a secure network enterprise.

If you are experiencing high latency or poor voice quality with Skype, you can troubleshoot your connection quality by using NAT Check or Skype's Display Technical Call info feature found in the Advanced options tab. To enable the tech support feature or edit the Config.xml file manually:

1. Exit Skype.
2. Locate the **Config.xml** file located in the **<Drive>\Documents and Settings\<User Name>\Application Data\Skype\<Skype user name>** folder and open it with Notepad.exe or a similar text editor.
3. Use the 'find' capability to locate the setting **<DisplayCallInfo>0</DisplayCallInfo>**
4. Change the value from 0 to 1 and save the file.
5. Launch Skype.

Visit Dan Douglass's Web site at the following URL for scripts and utilities to modify the config.xml file setting in a business environment: [www.codehatchery.com/skype.html](http://www.codehatchery.com/skype.html).

Once you have enabled the **Display Technical call info** feature, you can make a test call to the Skype Test Call user. Once you have established the call, simply hover the mouse cursor over the user's avatar (picture), and you will see a tooltip-style popup with connection information (see Figure 11.11).

**Figure 11.11** Skype Connection Information

Note that in this scenario, the relays count is 0 and the roundtrip time is 105ms (1000ms = 1 second). Since the Skype answering machine is open, the connection is very clean, and there is very little latency.

## Display Technical Call Information

The following is detailed information about the Technical Call Information popup items shown in the preceding and following examples.

### *Call Status*

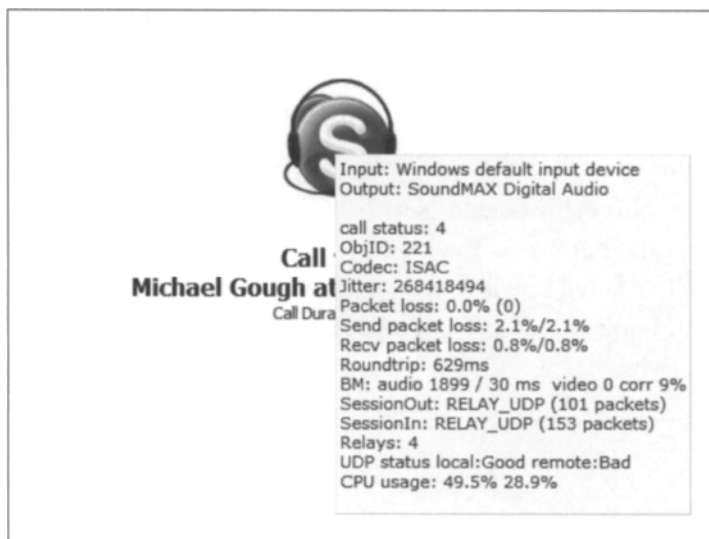
- 0 = Hosting conference.
- 1 = ROUTING - call is currently being routed.
- 2 EARLYMEDIA - with the pstn there is possibility that before the call is actually established, the early media is being played. For example, it can be a calling tone, or it can be some waiting message (all operators are busy, hold on for a sec) etc.
- ?? FAILED - call failed. Try to get FAILUREREASON for more information.
- 3 = RINGING - currently ringing.
- 4 = INPROGRESS - call is in progress.

- 5 = ONHOLD - call is placed on hold by you.
- ?? FINISHED - call is finished.
- ?? MISSED - call was missed.
- 8 = REFUSED - call was refused.
- 8 = BUSY - destination was busy i.e. pressed hang up button.
- 10 = ONHOLD - call is placed on hold by other party.
- 13 = CANCELED (Protocol 2)
- ObjID: Ignore this information as it is not important.
- Codec: ISAC is most commonly used (G729 and iLBC are also possible)
- Jitter: Network administrators need to look at jitter. Jitter is the variation in the time between each of the delivered packets of data arriving from the source to the destination. This could indicate a bandwidth bottleneck or heavy traffic from the source to destination causing some packets to arrive sooner than others. The common method for reducing jitter is to buffer data at the destination.
- Packet Loss: Network administrators need to be aware of packet loss. This is the total percentage of the packets of data that don't make it to or from each party in the conversation. This should be low, but will be something if you are using UDP, since delivery is not guaranteed.
- Send packet loss: Network administrators should pay attention to this setting. This indicates how much data is not making it to the destination party in the call. If the Send packet loss is high, it means that something is causing the packets from getting to the remote client.
- Recv packet loss: Network administrators should pay attention to this setting. This indicates how much data is not making it from the other party in the call. If the Receive packet loss is high, it means that something is preventing the packets from getting to you from the remote client.
- Roundtrip: Normal users and Network administrators can get information from this. The higher the number is, the longer it takes for your voice to get to the other party and back. This should be low, and anything about 300ms starts to get choppy, reducing call quality. Look at SessionOut and SessionIN, or run NAT Check to determine why you are relaying.
- BM: This is related to the bandwidth and quality of the audio and is not important.

- **SessionOut:** Network administrators should look at this if roundtrip values are high. This should say UDP. If it says TCP or RELAY\_UDP, then you are not operating at the best performance. In this case look at UDP status remote. If it says remote:Bad, then the remote party is behind a firewall and cannot receive UDP traffic.
- **SessionIn:** Network administrators should look at this if roundtrip values are high. This should say UDP. If it says TCP, RELAY\_TCP, or RELAY\_UDP you are not operating at the best performance. In this case look at UDP status local. If it says local:Bad, you could, at your discretion, open up the UDP port as discussed earlier in this chapter to allow inbound UDP traffic.
- **Relays:** Ideally the relay count is zero (0), and will be when checking Skype voice-mail. When relaying is in effect the count will almost always be four (4), but you may see a lower number during the time that the relay connections are being established.
- **UDP status:** should always be local:Good remote:Good. If either are Bad, look at SessionIn/SessionOut to remedy.
- **CPU usage:** 35.8% 13.4% Total CPU usage of each processor by all running applications on the local machine. If this is too high, then the machine may be too overloaded to allow Skype to operate efficiently. Other applications will most likely be suffering as well.

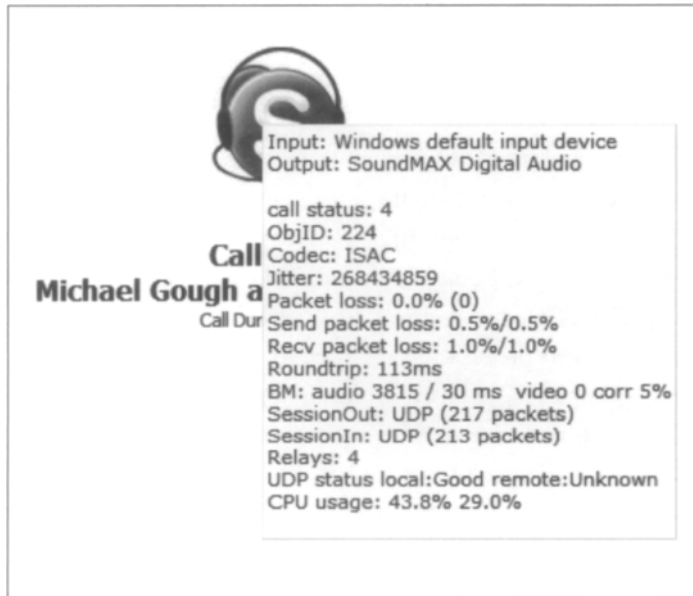
The next example is a call to a user on large corporate network where no inbound UDP is allowed back in through the firewall, and there is a very complex network infrastructure (see Figure 11.12).

**Figure 11.12** A Skype Call to a User on a Large Corporate Network with Firewall Restrictions



Note the difference in the SessionOut and SessionIn results. *RELAY\_UDP*, and the UDP status *remote:Bad* show us that the remote location is the problem, and that a relay node is being used to carry UDP traffic for each of the clients. The result of the relays is the long roundtrip time of 629ms, and therefore, there is a delay in transmitting the voice data to the remote client. Basically, it takes more than half a second for everything you say to get to the remote client, so the conversation is choppy and degraded. To improve this connection, the callers can use NAT Check to see if they are able to use UDP and troubleshoot the connection. If it is possible to open the UDP port inbound to the remote client in this scenario, the sessions can use a direct UDP or peer-to-peer connection, and the communication will be improve almost tenfold. See Figure 11.13, a connection to the same caller, without the firewall restrictions.

**Figure 11.13** A Skype Call to a User on a Large Corporate Network without Firewall Restrictions



To summarize, if you have a bad connection, each client can run NAT Check and the Display Technical info to see who is having difficulty communicating. The findings can be confirmed with the configuration demonstrated in the previous section. To correct the issue, determine the UDP port the trouble client is listening on. Open that port inbound by defining a firewall rule. The rule should be specific to the client, so it might be something like *Allow: WAN \* to LAN 192.169.1.21 UDP: 3259*, which allows all WAN IP addresses to communicate inbound to the private LAN address 192.168.1.21 over UDP port 3259.

## Small to Large Companies

In most large companies, this will not be feasible and may possibly be against the corporate security policy and allowable network practices, but this does remain an option for small to medium-sized businesses that desire better communication quality and have the flexibility to modify their firewall rules. Some firewalls allow rules to be enabled during a specific time frame, and outside of that time window, the rule is disabled. If you wish to limit the use of Skype to only off-business hours, this type of feature would provide better security than leaving the port open all of the time. With any modification to your firewall rules, be sure to check your corporate security policy and with corporate security and your network team to gain approval and to understand the potential risks that are associated with opening any ports on a firewall to an internal client. Additional layers of security should be implemented if this configuration is to be used. If any peer-to-peer communication is allowed, it is recommended that the clients have a personal firewall solution to further protect the systems from malicious activity.

## How to Block Skype in the Enterprise

From a security or network administrator's point of view, the very same features that make Skype connect reliably through a restrictive firewall present a challenge to preventing or blocking Skype traffic on a network. Skype is very robust and can function with access to only port 80. Most corporations allow outbound Web traffic, so port 80 (HTTP) must remain open. Port 443 is the SSL port (HTTPS), and secure Web sites require this port to remain open. It is not as simple as blocking ports to prevent Skype from functioning.

Several tasks must be completed to block Skype in your enterprise. The first step is to block access to the Skype downloads to prevent the executable from even being installed on your client machines. This practice is referred to as *black listing*. This step is not entirely effective by itself, since some users might already have the Skype client installed or could bring the installation package from home on a CD or thumb/flash drive.

It is good practice to prevent unnecessary applications from accessing the Internet. The best way to achieve that is by blocking all ports on the firewall and then selectively allowing known traffic to pass, the "deny all unless explicitly allowed" mentality. In addition, you may choose to restrict access to all Internet sites except those that have been approved by your organization. This is referred to as *white listing*, and although it requires more maintenance, it is much more secure.

Another method used to prevent communication over the Internet is to use packet filters. Packet filters examine the data inside the headers of transmitted packets. This information can be used to create rules to dump messages that contain headers that meet the filter criteria. Unfortunately, Skype data is encrypted, so packet filters are unable to examine the information in the data packets; therefore, packet filtering is useless. However, a new hardware device is pur-

ported to identify the signature of Skype communication and block Skype traffic based on that identification.

In a corporate enterprise environment, you may have other software solutions that allow the use of application filters on the desktops. This is another effective way to block Skype. The method of policies depends on the platform, but essentially, the concept is the same. When a user attempts to execute a program that is defined as disallowed, the process that monitors the client will prevent the program from executing. An example of this would be to use Microsoft Systems Management Server and define a restriction on the Skype.exe executable. Network Associates and Symantec have similar features built in to their groupware products.

Skype is very effective at finding ways to communicate with other Skype peers. There is no straightforward way to block Skype in the enterprise. The most effective method is to prevent the program from running at all or scan for it on all systems that are not approved and delete it from each system.

## Endnote

1. "Number of Hosts Advertised in the DNS." *Internet Domain Survey*, July 2005, [www.isc.org](http://www.isc.org) (accessed October 4, 2005)



This Page Intentionally Left Blank

## Validate Existing Security Infrastructure

### Solutions in this chapter:

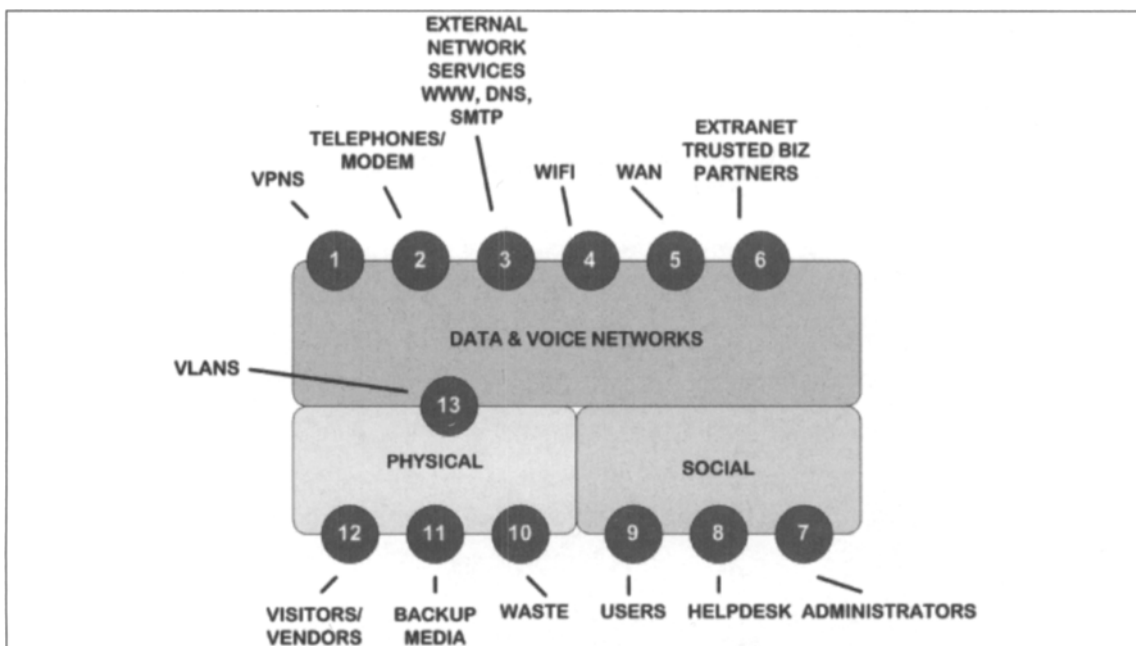
- Security Policies and Processes
- Physical Security
- Server Hardening
- Supporting Services
- Unified Network Management

# Introduction

We begin the process of securing the VoIP infrastructure by reviewing and validating the existing security infrastructure. Addition of VoIP components to a preexisting data network is the ideal opportunity to review and bolster existing security policy, architecture, and processes.

One way of visualizing the components of a given security architecture is to use Figure A.1, which graphically shows a number of network security interfaces.

**Figure A.1** Security Interfaces



The interfaces between data and voice networks and the external world are represented by the red circles numbered 1 through 6. Additionally, data and voice networks share interfaces with the physical and social realms. Interfaces to data and networks include VPNs, telephones and modems (modems that are used to control or monitor servers or other critical systems are particularly interesting to miscreants), typical web browsing and e-mail services, intracompany WAN connections, and intranet or external connections with vendors and business partners. Technical security controls such as firewalls, IDS, and ACLs are useful at these interfaces.

Interfaces 7 through 9 portray the users, administrators, and help desk personnel that connect with the data and voice networks. In some situations, a call center for example, an additional class of users—operators—could be defined. I believe, based upon personal and

anecdotal evidence, that most criminal information security incidents occur via these social interfaces. Unfortunately, technological security controls are difficult to implement and manage at these interfaces.

Interfaces 10 through 12 represent the interfaces between the physical domain and the data and voice network. Recently, problems in this area have resulted in the loss of critical data. In January 2006, a laptop stolen from an Ameriprise Financial worker resulted in the loss of personal information from more than 230,000 customers, and in the same month, an unnamed Toronto health clinic found its private patient data literally “blowing in the wind,” as the clinic’s waste disposal operator improperly recycled rather than shredded the clinic’s data. Numerous other examples exist where discarded laptops or hard drives have been found to contain private information; and “dumpster-diving” is recognized in the security industry as a valid and often lucrative source of information.

Lastly, interface 13 describes the VLAN (Virtual LAN) interface.

This listing is not necessarily complete, but it suggests where security controls can be most effectively implemented. Traffic can oftentimes be monitored, dropped, or approved, or throttled at these synapse-like junctions.

The purpose of this chapter is to reinforce the concept that many of the components that you will require to secure a VoIP/Data network are likely to exist within your current infrastructure.

The first portion of this chapter is not designed as a “how-to” on writing security policies because there a large number of these resources available. In this section, we will argue that information security is critical to an organization, and that security policy underpins all other security efforts. Then we will review the processes required to implement a functional security policy, and we’ll look at some of the critical factors that determine the value of a security policy. We have provided a worksheet that will allow you to perform a gap analysis on your existing security policies. A commented sample VoIP Security Policy module is provided for you as a template at the end of this chapter.

## Security Policies and Processes

In order to reap the benefits of modern communications, we are required to secure the systems and networks that comprise the communications infrastructure.

The process of securing a converged VoIP + Data network begins with the formulation, implementation, and communication of *effective* security policies. This is true for pure data networks as well. Security policy provides metrics against which costs can be justified, drives security awareness, and provides the framework for technology and process. Once policy is in writing, less time will be spent debating security issues. Policy provides a vantage point that can be built into an organization’s reporting systems in order to reassure management about the quality, reliability, and comprehensiveness of its security infrastructure. When

approached in this fashion, information security becomes less an administrative and technical burden, and more of a competitive advantage.

## NOTE

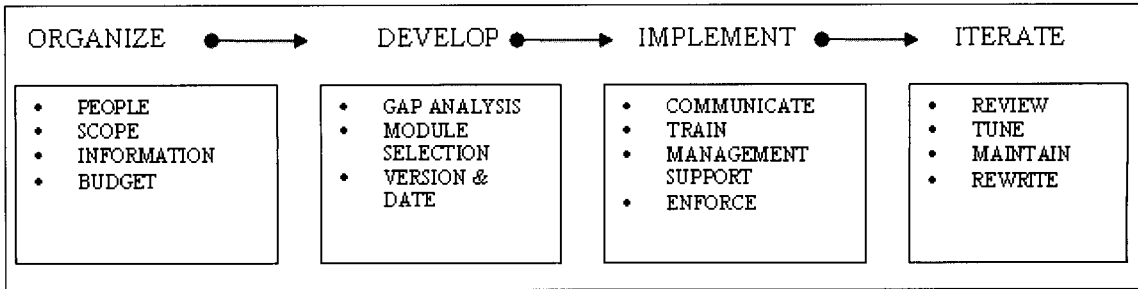
---

A competitive advantage within a vertical can be gained either by providing products or services that provide more benefits at a fixed price, or by providing the same benefits at a lower price. An organization can gain a competitive advantage by utilizing its resources (things like people, knowledge, reputation, brand) or its capabilities (processes, procedures, routines, etc.) more effectively than its competitors. Basically, a competitive advantage allows an organization to sustain profits that exceed the average for other organizations within its industry. In the context of information security, competitive advantage can be affected positively by implementing and maintaining a workable information security methodology. These processes can and should be regularly disseminated to clients and vendors, thus creating a reputation for honest and professional treatment of information. Any types of mishandling of client or vendor information—whether from hackers or from simple misuse—leads to reputation, brand, or knowledge damage, and consequently, loss of competitive advantage.

---

Policy formulation is an important step toward standardization of enterprise security activities. The organization's policy is management's vehicle for emphasizing its commitment to IT security and making clear the expectations for associate involvement and accountability. Policy formulation establishes standards for all information resource protection by assigning program management responsibilities and providing basic rules, guidelines, definitions, and processes for everyone within the organization. One major aim of the security policy is to prevent behavioral inconsistencies that can introduce risks. Ideally, policy will be sufficiently clear and comprehensive to be accepted and followed throughout the organization yet flexible enough to accommodate a wide range of data, activities, and resources.

There is no single best process for developing a security policy. Much of the process is dependent upon variables such as the size, age, and location of an organization, the vertical that the organization occupies, the impact of regulation on the organization, and the organization's sensitivity toward risk. Figure A.2 shows how an approach to policy development and implementation can be organized.

**Figure A.2** Policy Development and Implementation

In general, the first step in policy formulation is convincing management that these policies are necessary. In today's environment, this task is simplified by regulatory requirements and by the sheer number of security-related incidents reported in the popular press (see the previous section of this chapter for recent examples). Once management commits to security policy development, the individuals responsible for policy formulation are selected to form a security steering committee.

One of the most common reasons policy efforts fail is that policy too often is developed in a vacuum or by decree, and as a result, does not reflect the security needs of the entire organization. Being inclusive from the start will make it easier to market the policy within the organization later on; in order for security policy to be appropriate and effective, it needs to have the acceptance and support of all levels of employees.

The following is a list of individuals who should be involved in the creation and review of security policy documents:

- Information or Site security officer (see the CSO discussion in the next section of this chapter)
- Information technology technical staff (network managers, system administrators, etc.)
- Help desk staff
- Business unit heads or authorized representatives
- Security emergency response team
- Representatives of the user groups affected by the security policy
- Management
- Legal counsel
- Human Resources

The previous list is not necessarily comprehensive. The idea is to bring in representation from key stakeholders, management who have budget and policy authority, technical staff

who know what can and cannot be supported, and legal counsel who know the legal ramifications of various policy choices. It may be appropriate to include audit personnel. Involving this group is important if resulting policy statements are to reach the broadest possible acceptance. The role of legal counsel will vary from country to country.

After the security steering committee is formed, the next step is to write policy. These can be written from scratch although I don't recommend this as it is difficult to be comprehensive with this approach. A better method relies on modifying existing security policies or policy modules that can be found on the web (Googling "security polices" garners over 306 million hits). Policies are available for free or can be purchased, oftentimes as templates.

One approach to modifying either new or existing security policies is to perform a gap analysis—contrasting the proposed policies with existing conditions or perceptions. Using the worksheet shown in Table A.1, you can compare an organization's inventory of policies, procedures, standards, and guidelines to a checklist that identifies the security industry's best practices.

This worksheet should be sent to a set of individuals within the organization that represent each business unit. The individuals are asked to determine in their experience, whether or not a particular policy exists as a formal document, an informal document, a draft; or does not exist, is not applicable, or is unknown. In addition, they are asked to rate, on a scale of 1–5 (with 5 equaling the highest priority), how important they felt each policy area was. They are limited to answering 5 (high priority) to only six of the 24 categories.

The questionnaires are returned, and the results are averaged. This gap analysis identifies any important security policies, procedures, standards, and guidelines that are absent, and gives some indication of the strengths and weaknesses of existing security policies.

**Table A.1** A Gap Analysis Worksheet

---

EXISTENCE (1–6): 1=FORMAL; 2 =INFORMAL; 3 =DRAFT; 4 =NO; 5 =NA;  
6=UNKNOWN

PRIORITY (1–5): 1=NOT IMPORTANT; 5=CRITICAL

---

NAME	EXISTENCE	PRIORITY	DESCRIPTION
Acceptable Use Policy			Establishes computer resource usage guidelines for staff during the course of their job duties in a responsible and ethical manner. It also specifies behaviors and practices that are prohibited.

---

Continued

**Table A.1 continued** A Gap Analysis Worksheet

NAME	EXISTENCE	PRIORITY	DESCRIPTION
Access Control Policy			This policy defines the access rights and level of authority of each user or group of users based on their business need. Ensures that only authorized users are given access to certain data or resources.
Account Management Policies			Defines who has authority to make account modifications, and how accounts are created or disabled.
Privacy Policies			Defines reasonable expectations of privacy regarding such issues as monitoring of electronic mail, logging of keystrokes, and access to users' files.
Availability Policies			Statement that sets users' expectations for the availability of resources. It should address redundancy and recovery issues, as well as specify operating hours and maintenance downtime periods. It should also include contact information for reporting system and network failures.
Technology Purchasing Guidelines			Specifies required, or preferred, security features. These typically supplement existing purchasing policies and guidelines.
Configuration Management Policies & Procedures			Defines how new hardware and software are tested and installed, defines how changes are documented.

Continued



**Table A.1 continued** A Gap Analysis Worksheet

NAME	EXISTENCE	PRIORITY	DESCRIPTION
Control of proprietary information and intellectual property			Defines policies to handle proprietary information, trade secrets, and intellectual property. It includes procedures to protect and safeguard information that is considered sensitive and proprietary.
Data Backup Procedures			Defines what gets backed up, when, how often, and how. Also covers how tapes are stored (to prevent theft).
Firewall Management Policy			Describes how the firewall hardware and software is managed and configured; how changes are requested and approved; and auditing requirements and procedures.
Internet Access Control Policy			Defines the services (inbound and outbound) that will be supported when traffic travels between the Internet and company systems.
General Encryption Policy			To assure interoperability and consistency across the organization, this policy would mandate standards to which encryption systems must comply, possibly specifying algorithms and parameters to be used.
Internet Security Awareness & Education Policy			Outlines the educational and training measures that will be taken to make computer users aware of their security responsibilities.

Continued

**Table A.1 continued** A Gap Analysis Worksheet

NAME	EXISTENCE	PRIORITY	DESCRIPTION
Intrusion Detection Policy/Procedures			Defines responsibilities and scope for tools that provide for the timely detection of malicious behavior by users on the network or individual hosts. (Excludes antiviral measures.)
Network Connection Policy			Describes the requirements and constraints for attaching devices to the corporate network.
Password Management Policy/Procedures			Guidelines to support operations for password management such as password assignment, reset, recovery, protection, and strength. These guidelines support privileged and nonprivileged account password assignment.
Remote Access Policy			Outlines and defines acceptable methods of remotely connecting to the internal corporate network (including Internet and VPN access).
Security Incident Handling Policies & Procedures			Procedures describing the steps to be taken in response to computer security incidents that occur within facilities or networks. This includes interfacing with law enforcement agencies, logging and documenting incidents, evidence preservation, and forensic analysis.

Continued

**Table A.1 continued** A Gap Analysis Worksheet

NAME	EXISTENCE	PRIORITY	DESCRIPTION
System Security Standards (for specific OSES)			Procedures for securing specific operating systems (e.g., NT/Win2K, MVS, Linux) that are used within the organization. This document explains how a specific OS needs to be configured for corporate use.
Privileged Access Policy			Establishes requirements for the regulation and use of special access (e.g., root or Administrator) on corporate systems in a responsible and ethical manner. It also specifies behaviors and practices that are prohibited.
Remote Partner Acceptable Use & Connectivity Policy /Procedures			Provides guidelines for the use of network and computing resources associated with third-party networks. Provides a formalized method for the request, approval, and tracking of such connections.
User Account Policies			Outlines the requirements for requesting and maintaining accounts on corporate systems.
Virus Prevention Policy/Procedures			Defines actions that will be taken to detect and remove computer viruses.
IM Policy/Procedures			Defines architecture and deployment guidelines for Instant Messaging.
Wireless Policy/Procedures			Defines architecture, and deployment guidelines for 802.11a/b wireless networks.
VoIP Policy/Procedures			Defines architecture, and deployment guidelines for Voice-over IP networks.

Regardless of the starting point, my experience has been that policy development is an iterative process—policy first is broken down into modules (see sidebar for an example listing of high-level modules), modules are assigned to the appropriate individuals, and each module then is edited by steering committee members. After several cycles through this process, a draft version 1.0 document is produced.

The draft security policy document should be evaluated by the security steering committee based upon a number of characteristics:

- Is the scope of the document appropriate?
- To whom does the policy apply (i.e., all employees, full-time employees only, contractors, consultants, customers)?
- Are the organization's information assets comprehensively defined and are the appropriate controls implemented?
- Is the policy consistent with existing corporate directives and guidelines, and with applicable legislation and regulations?
- Is the document concise? Can it be understood and remembered by all affected parties? I've seen several security policies that numbered over 100 pages. I believe that, in the case of security policy development, shorter is always better. Any policy longer than 40 to 50 pages will not be read or remembered by most users.
- Are the policy guidelines reasonable? That is, can the normal person follow the policy directives and still perform their regular duties? Are the guidelines consistent with current technology, organizational culture, and mission?
- Does the document leave room for good judgment? All relevant personnel should be responsible for exercising good judgment regarding the reasonableness of personal use of company resources. Employees should understand that effective security is a team effort involving the participation and support of all those who deal with information and/or information systems.
- Is the document extensible?

## Policies & Procedures...

### Sample Policies, Procedures, and Guidelines Summary

The following guidelines, policies, and procedures are necessary to effectively secure your systems and network:

1. Acceptable Use Policy
2. Access Control Policy
3. Account Management Policies
4. Availability Policies
5. Configuration Management Policies & Procedures
6. Control of Proprietary Information and Intellectual Property
7. Data Backup Procedures
8. Firewall Management Policy
9. General Encryption Policy
10. IM Security Policy/Procedures
11. Internet Access Control Policy
12. Internet Security Awareness & Education Policy
13. Intrusion Detection Policy/Procedures
14. Network Connection Policy
15. Partner Connection Acceptable Use & Connectivity Policy/Procedures
16. Password Management Policy/Procedures
17. Privacy Policies
18. Privileged Access Policy
19. Remote Access Policy
20. Security Incident Handling Policies & Procedures
21. System Security Standards (for specific OSes)
22. Technology Purchasing Guidelines
23. User Account Policies
24. Virus Prevention Policy/Procedures
25. VoIP Security Policy/Procedures
26. Wireless Policy/Procedures

Implementation of the resulting security policies is also a process. Policy cannot merely be pronounced by upper management in a one-time directive with high expectations of its being readily accepted and acted upon. Rather, just as formulating and drafting policy involves a process, implementation similarly involves a process, which begins with the formal issuance of policy, and continues via user awareness training, intracompany communications utilizing an intranet or other company communications vehicles, review, and update of policy and policy definitions at regular intervals.

Often there exists a lack of awareness of an organization's IT security policies, among both the general user population and the IT staff. It is imperative that an organization undertake some form of education campaign among the general user population to raise awareness of both the existence of IT security policies and their contents.

All employees should be required to read and acknowledge their understanding of parts of the IT security policy relevant to the general user population during the on-boarding process. As updates are made to the policies that affect the general user population, notices should be sent to the users so that they can acquaint themselves with the changes. It is not enough for these notices to be sent out by e-mail; the notification procedure must include some mechanism for the user to acknowledge receipt of the notice and understanding as to the changes to the policy.

The IT security staff should also consider conducting brief, in-person group trainings regarding the provisions of the IT security policy and physical security in general. These trainings are often more effective than impersonal mechanisms such as e-mail, which are often ignored or acknowledged without a full understanding of the contents of the message or notification. In-person trainings also allow the general user population to gain a fuller understanding of IT security issues, as it allows them to ask questions and voice concerns regarding the policy.

In the process of raising awareness of IT security policies, it is important that the general user population understands the sanctions associated with violating these policies. A security policy that is not enforced, or that is enforced on an arbitrary basis, will be honored more in the breach than in the practice. The policies should include mechanisms for measuring compliance, detecting noncompliance, and responding to policy violations. The general user population must be made aware of these mechanisms. These processes are necessary to make sure that users are held accountable for their actions, as well as to guard against the consequences of inappropriate actions.

A sample VoIP security policy module is included at the end of this chapter. You can use this as a starting point for your own customized VoIP security policy module.

## Physical Security

Physical security is an essential part of any security plan. Physical security refers to the protection of building sites and equipment (and all other information and software contained

therein) from theft, intrusion, vandalism, natural disaster, man-made catastrophes, and accidental damage (e.g., from electrical surges, extreme temperatures, and spilled coffee). It requires suitable emergency preparedness, reliable power supplies, adequate climate control, and appropriate protection from intruders.

Statistics show that 70 percent of data theft is physical theft (Computer Associates/Pinkerton, 2004). Physical security safeguards provide a first line of defense for information resources against physical damage, physical theft, and unauthorized disclosure of information.

Safeguards can be broken down into two categories: human and environmental. Human safeguard recommendations are:

- Console access should be restricted or eliminated.
- Logon, boot loader, and other passwords must be a minimum of eight characters including at least one each of alpha, numeric, and ctl characters.
- VoIP components must be located in a secure location that is locked and restricted to authorized personnel only.
- Access to these components, wiring, displays, and networks must be controlled by rules of least privilege.
- System configurations (i.e., hardware, wiring, displays, networks) must be documented. Installations and changes to those physical configurations must be governed by a formal change management process.
- A system of monitoring and auditing physical access to VoIP components, wiring, displays, and networks must be implemented (e.g., badges, cameras, access logs). From the point at which an employee enters the building, it is recommended that there be a digital record of their presence.
- The server room should be arranged in a way that people outside the room cannot see the keyboard (thus seeing users/admin passwords).
- Any unused modems must be disabled/removed.
- No password evidence (notes, sticky notes, etc.) is allowed around the system.

Environmental safeguard recommendations are:

- The CPU case should be locked and the key must be accounted for and protected. A backup key should be made and kept securely offsite (e.g., in a safety deposit box).
- USB, CD-ROM, monitor port, and floppy disks drives should be removed, disabled, or glued shut.

- Adequate temperature and humidity controls must be implemented to avoid equipment damage.
- Adequate surge protectors and UPS must be implemented, maintained, and tested.
- Cleaning and maintenance people should be prohibited from the area surrounding any electronics.
- Food, drink, or smoking is prohibited in the same areas.

Frequently, IT and security staff only considers IT security through the prism of logical (IT-related technical) security controls. However, it is often the case that lapses in physical perimeter security controls can contribute to weaknesses in IT security. Methodical testing and anecdotal evidence indicate that the physical perimeter security is insufficient to prevent unauthorized users from entering secured areas, resulting in easy access to the internal network.

## NOTE

**Choke Points:** Often, the largest failing of physical security is the lack of a single choke point for the authentication and admittance of authorized visitors. Following is a real-world example.

Normally visitors are authenticated in a visitor registration area and are then admitted to an elevator area in a separate part of the building using a badge issued in the visitor registration area. The badge is designed to provide a visual indication once it has expired, and is valid only for that specific day or week. However, still-valid daily badges often can be found discarded in trash receptacles. Since the elevator area is watched by a different group of people from those who authenticate the visitor, the guards at the elevator area have no idea whether a visitor is authorized or not other than the possession of a valid visitor badge.

In this example, multiple choke points result in virtually unrestricted physical access to the internal infrastructure.

IP-PBX equipment should be located in a locked room with limited access. This type of access should be provided as a user authentication system with either a key-card or biometric device. The use of a keypad alone to gain access is not permitted. All methods of gaining entry into the room must provide for a list of users that have accessed the room along with a date/time stamp.



## Perimeter Protection

Perimeter protection is designed as a deterrent to trespassing and to route employees, visitors, and the guests to selected entrances. Here are two useful examples.

### Closed-Circuit Video Cameras

CCTV cameras are relatively inexpensive to deploy and provide a large return on investment. The typical camera should be on a pan/tilt mounting and have a zoom lens, both of which should be controllable by the operator. These features permit the monitoring of wide areas for general activity or the ability to zero in on a particular location.

It is unrealistic to expect an operator to alertly monitor for long periods of time. Therefore, the system should be programmed for periodic sweeps or augmented with intrusion devices triggered by unusual events. All video output should be recorded for future replay if necessary. The videotapes should be archived for a minimum of 30 days. A videotape should be retired and physically destroyed after three complete usage cycles.

### Token System

A token is an object physically carried by the user used for authentication purposes. There are several different types of token identification methods including token cards, readers, and biometric devices. The most widely used method is a token card. The following is a sample of the different types of access cards.

#### *Challenge/Response Tokens*

This device generates a random passcode, based upon a built-in algorithm that is combined with a user pin number. This resulting number is used, in combination with the standard username and password, for user verification method. Passcode sniffing and brute force attacks are futile since the result is good only for one specific period of time.

#### *Dumb Cards*

An example of a dumb card is a photo identification badge. The photo and individual statistics supply enough information to complete the authentication process. Generally, the authentication process is a visual comparison of the ID and the face of the individual.

#### *Smart Cards*

The classic example of a smart card is an ATM card. This device combines an individual PIN with information encoded on the card itself.

## Biometric Devices

All biometric devices rely upon some type of input device, such as a video camera, retinal scanner, thumb pad, or microphone. The data is then digitized and compared to a stored record. If the match is within defined parameters access is granted.

## Wire Closets

Wire closets form a very important piece of the actual network as well as the data that travels on it. Many wire closets contain both network and telephone connections. Oftentimes cases exist where the wire closet is shared by many of the building occupants. The wiring closet can be a very effective launch pad for internal attacks. It is also well suited to the unobserved monitoring of a network. We recommend securing these sensitive locations. When available, they could be added to the already existing card key systems. This would automate the logging of who accessed the location and when. A recommended course of correction would also include the requirement that your organization's representative be physically present during the entire period a collocated wire closet is accessed.

What if the landlord controls access to the closet in a shared-tenant space (a common scenario)? One answer is to use the closet only for external PSTN connectivity and home-run all other wiring to a dedicated closet.

### Security Elements...

#### Passwords: The Single Most Important Security Control

You will see this axiom repeated several times in this text. **Well-chosen passwords are the single most important element of any computer security policy.** They are the front line of protection, and often the only line of protection, for user and administrative accounts. A single poorly chosen password may result in the compromise of an entire enterprise network. The first step in protecting against unauthorized access is to define, communicate, and enforce strong password policies.

## Server Hardening

From a high-level point of view, all devices that participate in network communications should follow the principle of "Least Privilege." This concept is simple to understand and difficult to put into practice as it often interferes with or interrupts an individual's (particularly administrators)

ability to perform routine functions. This means that anything not required should be disabled. Turn off all unneeded services. Disable any features that are not in use. Remove unnecessary applications. This maxim is particularly important when applied to critical infrastructure including servers, routers, firewalls, and so on. Adhering to this principle will reduce the number of potential attack vectors on these systems.

The potential for attack against components of the PBX system is real, and failure to secure a PBX and voice mail system can expose an organization to toll fraud, theft of proprietary information, loss of revenue, and loss of reputation. Hardening the PBX system components limits unauthorized access and use of system resources. The hardening process is OS-specific, but regardless of the OS, consists of: patching, removal of extraneous services, extending logging, removal of unnecessary administrative and user accounts, permission tightening, activation of internal security controls, and various other security tweaks.

## Eliminate Unnecessary Services

Most VoIP server platforms ship today on either the Windows or Linux operating systems. Typically, these systems are delivered with many unneeded services activated. These extra services are potential security risks. There are a large number of online and hardcopy references that explain the details of hardening with Windows and Linux operating systems, so in this section we'll survey the high points.

On the Linux platform, examine the `/etc/inetd.conf` file. This file specifies the services for which the `inetd` daemon will listen. By default, `/etc/inetd.conf` is configured to activate a number of listening daemons. You can see these by typing:

```
grep -v "^#" /etc/inetd.conf
```

Determine the services that you require, and then comment out the unneeded services by placing a “#” sign in front of them. This is important, as several of the services run by `inetd` can pose security threats, such as `popd`, `imapd`, and `rsh`.

Next check your running services by typing:

```
ps aux | wc -l
```

This command will show you the services that normally are started by the `.rc` scripts. These scripts determine the services started by the `init` process. Under Red Hat Linux, these scripts reside in `/etc/rc.d/rc3.d` (or `/etc/rc.d/rc5.d` if you automatically boot to a GUI, such as Gnome or KDE). To stop a script from starting, replace the uppercase `S` with a lowercase `s`. You can easily start the script again just by replacing the lowercase `s` with an uppercase `S`. There are other ways to do this, such as `chkconfig`. The numbers in the names of the startup scripts determine the sequence of initialization. This may vary depending upon the version and Linux distribution that you are using. Scripts that start with an uppercase `K` instead of an uppercase `S` are used to kill services that are already running.

On most Windows Server platforms, the active services are listed in the Services window. This can be reached by typing:

```
services.msc
```

At a command prompt, Services simply can be stopped or started by clicking the appropriate stop/start buttons in the toolbar. Alternatively, services can be permanently stopped or started by double-clicking the particular service that you are interested in, and setting its startup type to either manual (the service may still be activated) or disabled. The choice of running services depends upon your environment, but the adage still remains—turn off any service that you don't explicitly require.

Additionally, Microsoft offers two tools that should be run on any server that is a component of critical infrastructure. These are Microsoft Baseline Security Analyzer (MBSA v.2.0) and the IIS lockdown tool. MBSA is a software tool that scans local and remote Windows machines and generates a report that lists both security vulnerabilities (missing patches, incorrect permission settings, etc) and the means to remediate those vulnerabilities. You can find it at [www.microsoft.com/technet/security/tools/mbsahome.msp](http://www.microsoft.com/technet/security/tools/mbsahome.msp). The IIS Lockdown Tool functions by turning off unnecessary features and removing particular directories. It also incorporates URLScan, which adds additional protection based upon predefined templates. All the default security-related configuration settings in IIS 6.0 (Windows 2003) meet or exceed the security configuration settings made by the IIS Lockdown tool, so it isn't necessary to run this tool on those servers. Currently, you can find the IIS lockdown tool at [www.microsoft.com/technet/security/tools/locktool.msp](http://www.microsoft.com/technet/security/tools/locktool.msp).

## NOTE

---

Bastille Linux is one of the more popular tools for hardening Linux. You can find it at [www.bastille-linux.org/](http://www.bastille-linux.org/).

---

## Logging

Once you have turned off as many services as are consistent with proper server function, enable extended logging. On Linux platforms the system logger (syslog) is controlled by the configuration file, `/etc/syslog.conf`. Syslog is a system utility for tracking and logging all types of system messages from informational to critical. Each message sent to the syslog server is formatted as ASCII text, and has two descriptive labels associated with it. The first describes the function (facility) of the application that generated it. For example, applications such as kernel and cron generate messages with easily identifiable facilities named kernel and cron. The second describes the degree of severity of the message. There are eight levels of

criticality ranging from emergencies to debugging with emergencies signifying the most critical messages. All system logs reside in `/var/log`. `/etc/syslog.conf` can be configured to store messages of differing severities and facilities in different files, and on different remote computers. Many references exist on the Web that describe configuring syslog on Linux. A good one is [www.siliconvalleyccie.com/linux-hn/logging.htm](http://www.siliconvalleyccie.com/linux-hn/logging.htm).

Note that remote syslog messages are encapsulated as UDP packets, and until RFC3411 is updated, remote syslog messages are not encrypted. Thus, anyone on the LAN can sniff the syslog traffic. This may be an issue if extended debug messages are generated by a critical server and sent across the LAN.

Windows does not ship with a native syslog daemon; instead, Windows relies upon the System Event Notification manager to track system events such as Windows logon, network, and power events. The System Event Notification manager also notifies COM+ Event System subscribers of these events. A number of syslog addons for Windows exist—I recommend the Kiwi Syslog Daemon. The KIWI product is a full-featured syslog daemon that is free in its basic edition. The extended version can be very useful in that it allows logging to a number of ODBC-compliant databases. Additionally, Kiwi offers a free syslog generator that simplifies testing of syslog functions and connections.

Additionally, under Windows, you'll want to enable extended logging via the Domain Security Policy and Local Security Policy snap-ins. These determine which security events are logged into the Security log on the computer (successful attempts, failed attempts, or both). (The Security log is part of Event Viewer.) Under the Audit Policy tab, logging can be enabled for nine particular security-related events. You should at least enable auditing of failed logon events, successful or failed policy change events, successful or failed account management, and successful or failed privilege use. Note that if the server is in a domain, domain security policies will override local security policies.

## Permission Tightening

Under Windows, permission tightening is an art. In addition, the process is significantly different depending upon whether the server version is Windows 2000 or Windows 2003. In these operating systems, Microsoft created a complex and powerful set of interrelating file, folder, and user permission controls that are, frankly, too complex for most system administrators to understand and configure. In my view, the complexity of configuring permissions leads to more security-related events than bad coding on Microsoft platforms, because most administrators rely on default permissions. I will note that with Windows 2003, Microsoft has created a more secure platform with regard to default permissions. Unfortunately, we don't have the space to cover the intricacies securing Windows permissions here. Suffice to say that if you are given the option, choose Windows 2003 as the base OS rather than Windows 2000.

Linux provides a number of accounts that likely are not required for use as a media server or PBX. The rule of thumb is: If you do not require an account, remove it. Each additional account is one more possible avenue of access to the system.

Create the “wheel” group if it doesn’t already exist, and populate that group with administrators. The wheel group is a group of select individuals that can execute powerful commands, such as `/bin/su`. By limiting the people that can access these commands, you enhance system security.

If they exist on your system, lock down the files `.rhosts`, `.netrc`, and `/etc/hosts.equiv`. The `r` commands, which are deprecated for remote access nowadays, use these files to configure access to systems. To lock them down, touch the files, and then change the permissions to zero. This way no one but root can create or alter the files. For example:

```
/bin/touch /root/.rhosts /root/.netrc /etc/hosts.equiv
/bin/chmod 0 /root/.rhosts /root/.netrc /etc/hosts.equiv
```

This step disables any `rhost`-based authentication.

Change the following files (if they exist) permissions to the following more secure mode:

<code>/bin/</code>	<code>root.root</code>	<code>711</code>
<code>/boot/</code>	<code>root.root</code>	<code>700</code>
<code>/dev/</code>	<code>root.root</code>	<code>711</code>
<code>/etc/</code>	<code>root.wheel</code>	<code>711</code>
<code>/etc/modules.conf</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/cron.daily/</code>	<code>root.wheel</code>	<code>750</code>
<code>/etc/cron.hourly/</code>	<code>root.wheel</code>	<code>750</code>
<code>/etc/cron.monthly/</code>	<code>root.wheel</code>	<code>750</code>
<code>/etc/cron.weekly/</code>	<code>root.wheel</code>	<code>750</code>
<code>/etc/crontab</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/ftpaccess</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/hosts.allow</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/hosts.deny</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/hosts.equiv</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/inetd.conf</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/rc.d/init.d/</code>	<code>root.wheel</code>	<code>750</code>
<code>/etc/rc.d/init.d/syslog</code>	<code>root.wheel</code>	<code>740</code>
<code>/etc/inittab</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/ld.so.conf</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/modules.conf</code>	<code>root.wheel</code>	<code>640</code>
<code>/etc/motd</code>	<code>root.wheel</code>	<code>644</code>
<code>/etc/printcap</code>	<code>root.lp</code>	<code>640</code>
<code>/etc/profile</code>	<code>root.root</code>	<code>644</code>

/etc/rc.d/	root.wheel	640	
/etc/securetty	root.wheel	640	
/etc/shutdown.allow	root.root		600
/etc/ssh/ssh_config	root.root	644	
/etc/ssh/ssh_host_key	root.wheel	640	
/etc/ssh/ssh_host_key.pub	root.wheel	644	
/etc/ssh/sshd_config	root.wheel	640	
/etc/syslog.conf	root.wheel	640	
/etc/updatedb.conf	root.wheel	640	
/home/	root.wheel	751	
/home/*	current	700	
/lib/	root.wheel	751	
/mnt/	root.wheel	750	
/root/	root.root	700	
/sbin/	root.wheel	751	
/tmp/	root.root	1777	
/usr/	root.wheel	751	
/usr/*	root.wheel	751	
/usr/bin/	root.wheel	751	
/usr/sbin/	root.wheel	751	
/var/	root.root	755	
/var/log/	root.root	711	
/var/log/*	root.root	600	
/var/spool/mail/	root.mail	771	
/var/tmp	root.root	1777	

## Additional Linux Security Tweaks

Now we'll discuss additional security tweaks for securing Linux systems.

1. Remove any files related to: audio (esp), and DHCP (dhcpcd). For example:
  - a. `rm -rf /etc/dhcpcd`
  - b. `rm -rf /etc/dhpcpd`
2. Disable cron use for anyone but root and wheel. This limits the possibility of someone running an unauthorized program periodically
3. Disable Set User ID (SUID) status from dump/restore, cardctl, dosemu, news server programs, rsh, rlogin, mount, umount, ping, ping6, at, usernetctl, traceroute, traceroute6, if possible. The SUID bit is set when a particular program needs to access resources at a higher privilege level than it is normally allowed. For example, traceroute sets the TTL field directly rather than through the sockets interface on the

packets it sends. Normally, only a program with root permissions is able to use this low-level interface; thus, traceroute normally is installed with the SUID bit enabled. Unless a pressing need exists in your environment for normal users to access the aforementioned utility programs, disable SUID on all these programs. Failure to remove this bit opens your systems to a number of exploits that result in privilege escalation to root level.

To find suid programs, issue the following command:

```
find / -type f -perm -2000 -o -perm -4000 -print
```

Then remove the SUID bit as follows:

```
chmod -s /bin/ping
chmod -s /sbin/ping6
chmod -s /bin/mount
chmod -s /bin/umount
chmod -s /usr/sbin/traceroute
chmod -s /usr/sbin/traceroute6
chmod -s /usr/sbin/usernetctl
chmod -s /usr/bin/at
chmod -s /usr/bin/newgrp
```

## Are You Owned?

### Protect Yourself from Root Kits

Install chkrootkit for monitoring of root kits. Chkrootkit is a tool to check a local machine for signs of a root kit. It does this in a number of ways: it checks critical system binaries for signs of root kit modification; it checks to see if a network interface is in promiscuous mode; it checks for wtmp, wtmpx, lastlog, and utmp deletions; and it checks for LKM Trojan modifications. Make sure to add chkrootkit to daily crontab and monitor its results regularly.

#### 4. Clean up mail:

```
cd /var/mail
cat /dev/null > *
chmod 000 *
```

#### 5. Clean up /usr:

```
cd /usr
```



```
rm -rf rpms
rm -rf games
rm -rf dict
rm -rf X11R6
cd /usr/local
rm -rf games
```

#### 6. Clean up /etc:

```
rm -rf /etc/X11
rm -rf /etc/yp.conf
```

A number of OS- and version-specific security tweaks exist. The following list is not exhaustive since many of these are environment-specific; however, these will give you some areas to focus on.

1. Enforce password aging.
2. Enforce limits on resources to prevent a DoS attack.
3. Password-protect boot loader.
4. Password-protect single user mode.
5. Add additional logging.
6. Disable apmd, NFS, Samba, PCMCIA, DHCP server, NNTP server, routing daemons, NIS, SNMPD, and GPM.
7. Disable printing and files related to lpd.
8. Activate TMPDIR protection.
9. Set umask to 077.
10. Restrict “.” from the PATH variable.
11. Activate Internal security controls.
12. Apply security patches (see last section of this chapter).

## Activation of Internal Security Controls

1. Configure TCP Wrappers by editing /etc/hosts.allow and /etc/hosts.deny. Put this first in /etc/hosts.allow. Then edit /etc/hosts.deny so that it reads ALL : ALL : DENY. Don't enter this until all the daemons are activated in /etc/hosts.allow.

```
sshd : ALL \
: spawn /bin/echo SSH Connection on `bin/date` from
%h>>/var/log/messages \
: allow
```

```
in.ftpd : ALL : spawn /bin/echo FTP access from %h on
        `/bin/date`>>/var/log/messages : allow
sshd : ALL : spawn /bin/echo SSH access from %h on
        `/bin/date`>>/var/log/messages : allow
in.telnetd : ALL : spawn /bin/echo TELNET access from %h on
        `/bin/date`>>/var/log/messages : allow
in.tftpd : ALL : spawn /bin/echo TFTP access from %h on
        `/bin/date`>>/var/log/messages : allow
```

2. Install Tripwire, a file system integrity-checking program for Windows and UNIX operating systems. The core of any computer system is the disk drive, whether the underlying objects are UNIX file systems, Windows NTFS, or the Registry. In general, making harmful changes to a computer system requires some type of modification to the data on disk, such as planting Trojan horse programs, back doors, root kits (a compressed group of files that allows a user to obtain system level privileges by exploiting a security hole in the operating system), or by modifying critical system files such as `/etc/passwd`.

From a security perspective, one of the most important responsibilities of modern operating systems is to authenticate users and preserve privilege levels. In computer security, root (superuser or admin) privilege level is all powerful: Root kits allow attackers to steal these privileges and to cover their tracks. Trojan horses masquerade as common harmless programs but may carry programs that facilitate remote superuser access. Backdoors allow unrestricted, unauthorized hacker access to network assets.

Tripwire is one form of intrusion detection. Much like the secret agent trick of putting a hair on the doorknob to validate that no one has entered a room, Tripwire validates that critical system files have not been altered. Tripwire creates a secure database of file and directory attributes (including, if desired, complex cryptographic file hashes), which are then used to compare against to monitor if a file or directory has been altered. For example, if an attacker has broken in and added a bogus entry to the `/etc/passwd` file, Tripwire will alert.

Tripwire software is used for host-based intrusion detection (HIDS), file integrity assessment, damage discovery, change/configuration management, system auditing, forensics, and policy compliance. Host-based IDS software is able to monitor a system or application log file for unauthorized changes. Tripwire's integrity assessment detects external and internal attacks and misuse. Ultimately, the role of Tripwire is to notify system administrators of changed, added, and deleted files in some meaningful and useful manner. These reports can then be used for the purposes of intrusion detection, recovery, and forensic analysis.

To use Tripwire, you first must specify a configuration file that designates the directories and files that you want to protect. You then run Tripwire (with the ini-

tialize option) to create a database of cryptographic checksums that correspond with the files and directories specified in the configuration file. Tripwire then is run periodically via cron, and the current checksums are compared with the originals. If a file is altered, then the checksums will not match. To protect the Tripwire program, configuration file, and initialized database against corruption, be sure to transfer them to a medium that can be designated as physically write-protected, such as a CD-ROM.

- a. Edit `/etc/tripwire/twcfg.txt`. Here is a sample configuration.

```

ROOT                =/usr/sbin
POLFILE             =/etc/tripwire/tw.pol
DBFILE              =/var/lib/tripwire/$(HOSTNAME).twd
REPORTFILE          =/var/lib/tripwire/report/$(HOSTNAME)-
$(DATE).twr
SITEKEYFILE         =/etc/tripwire/site.key
LOCALKEYFILE        =/etc/tripwire/$(HOSTNAME)-local.key
EDITOR              =/bin/vi
LATEPROMPTING       =false
LOOSEDIRECTORYCHECKING =false
MAILNOVIOLATIONS    =true <- Change to false
EMAILREPORTLEVEL    =3
REPORTLEVEL         =3
MAILMETHOD          =SENDMAIL
SYSLOGREPORTING     =false <- Change to true
MAILPROGRAM         =/usr/sbin/sendmail -oi -t

```

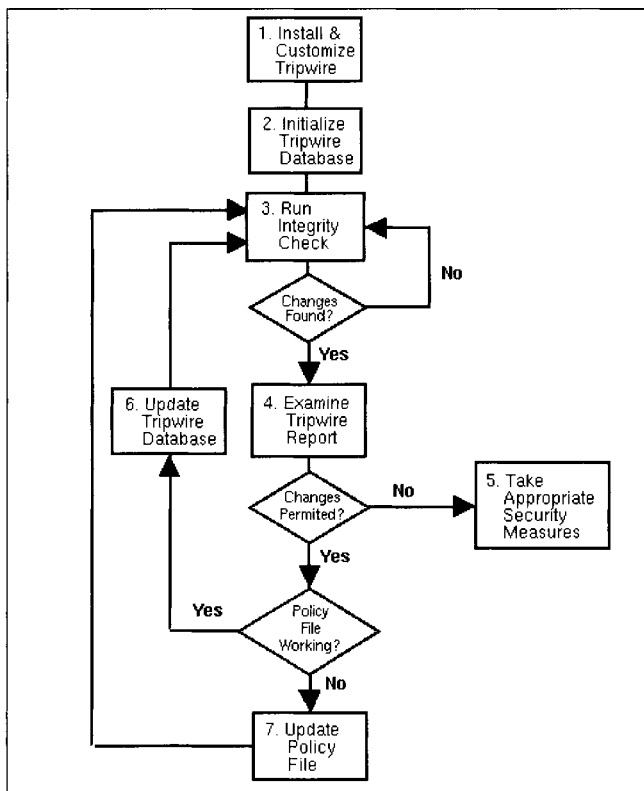
- b. As the root user, type `/etc/tripwire/twinstall.sh` at the shell prompt to run the configuration script. The `twinstall.sh` script will ask you for site and local passwords. These passwords are used to generate cryptographic keys for protecting Tripwire files. The script then creates and signs these files. When selecting the site and local passwords, you should consider the following guidelines:
  - c. Make the Tripwire passwords completely different from the root or any other password for the system.
  - d. Use unique passwords for both the site key and the local key.
  - e. The site key password protects the Tripwire configuration and policy files. The local key password protects the Tripwire database and report files. Warning: There is no way to decrypt a signed file if you forget your password. If you forget the passwords, the files are unusable and you will have to run the configuration script again.

- f. Run `/usr/sbin/tripwire --init` in order to initialize the tripwire database. This may take a while. Once you finish these steps successfully, Tripwire has the baseline snapshot of your file system necessary to check for changes in critical files. After initializing the Tripwire database, you should run an initial integrity check.

```
/usr/sbin/tripwire --check
```

This check should be done prior to connecting the computer to the network and putting it into production. Figure A.3 outlines the Tripwire processes.

**Figure A.3** A Diagram of Tripwire Processes



- g. By default, the Tripwire RPM adds a shell script called `tripwire-check` to the `/etc/cron.daily/` directory. This script automatically runs an integrity check once per day. You can, however, run a Tripwire integrity check at any time by typing the following command: `/usr/sbin/tripwire --check`
- h. To view a Tripwire report, type:

```
/usr/sbin/twprint -m r --twrfile \  
/var/lib/tripwire/report/<report_name>. twr
```

- i. Remove Tripwire install files: `twcfg.txt`, `twinstall.sh`, `twpol.txt`. and ftp the remaining files in `tripwire` directory to a secure server or burn them to disk.
- j. Be sure to check the Tripwire reports regularly. Much like other types of forensic logging, if the reports are not viewed by humans at regular intervals, then they serve little purpose.

*Activate iptables firewall*

*ftp rc.firewall.sh* script to `/etc/init.d`

Start script by running: `sh rc.firewall.sh`

Firewall services can be checked by: `service iptables status`

## Security Patching and Service Packs

In this section we'll put down some of our thoughts on Best Practices for the application and determination of appropriate service packs and security patches for VoIP-related client and server computers.

Service packs correct known problems and provide tools, drivers, and updates that extend product functionality, including updates, system administration tools, drivers, security updates, and additional components developed after the product was released. Service packs often contain many files, and are normally cumulative, but not always. Check this before you apply the service pack. Normally, service packs are packaged for easy downloading and installation. Patches, on the other hand, are usually specific to a particular file. Security patches eliminate (hopefully) security vulnerabilities. Oftentimes, security patches are released in response to the public circulation of exploit code. Service packs and patches often are interrelated, and it is important to check that the patch is workable for a particular service pack.

Before applying any service pack or patch, read all relevant documentation. Schedule server outages and be sure to have a complete set of backups available, in case a restoration is required. If possible, test the update(s) on noncritical infrastructure first. Develop and follow change control procedures. A good change control procedure has an identified owner, an audit trail for any changes, a defined announcement and review period, testing procedures, and a well-understood back-out plan. A good rule of thumb is: If you don't have a back-out plan, don't patch.

Only patch or update when you have to. It is likely that you have been part of a situation where a router or server function failed mysteriously. Typically, the vendor response is that you upgrade to a new operating system revision. The consequent upgrade then results in a number of new, unrelated problems. Murphy's Law dictates that this occurs only on the most critical infrastructure components at the most sensitive times. Alternatively, there are examples of a patch for one file that damages the functionality of another unrelated file.

Test before patching. Test after patching. Then, test again. If possible, monitor the updated production servers carefully for the first few days after the update.

## Supporting Services

VoIP relies upon a number of ancillary services as part of the configuration process, as a means to locate users, for management, and to ensure favorable transport, among others. These include DNS, DHCP, LDAP, RADIUS, HTTP, HTTPS, SNMP, SSH, TELNET, NTP, and TFTP. Other services that modify QoS are also required. We recommend that those services that support the VoIP infrastructure be dedicated to that infrastructure. The following sections assume that the support infrastructure is protected from direct Internet traffic by a firewall, firewalls, IDS, IPS, or a combination of these.

### DNS and DHCP Servers

DHCP is used in VoIP environments to provide an IP address and other relevant information such as the default gateway location, the subnet mask, the IP address of local DNS servers, the name and location of firmware and configuration servers, and other options. DHCP relies upon a broadcast mechanism to query for an IP address, so be sure to locate DHCP servers in separate broadcast domains in order to eliminate confusing addressing results.

DHCP services may be susceptible to a Rogue DHCP server attack. During boot-up, the IP phone sends a DHCP request for its own IP address and the address of a RAS server. Because DHCP replies are not authenticated, a rogue DHCP server can reply with erroneous information resulting in, at best, a Denial of Service, and at worst, routing to a server under the control of the attacker. One solution to this is to install an IDS on the VoIP-related subnets that could detect repeated DHCP requests (these are broadcast packets) and determine that an IP phone is having trouble booting. Alternatively, methods have been suggested (RFC3118) for authentication of DHCP messages. Unfortunately, few devices support these methods.

#### Tech Terms...

#### Acronym Soup

An Analog Telephony Adapter (ATA) is a device used to connect one or more analog telephones to a VoIP-based network. The ATA usually takes the form of a small box with a power adapter, one or more Ethernet ports, and one or more FXS telephone

Continued

ports. Another way to think about an ATA is that it functions as an FXS to Ethernet gateway.

A Foreign eXchange Subscriber (FXS) port is a legacy term for an interface that connects to subscriber equipment (telephone, modem, or fax). An FXS interface points to the terminal endpoint equipment, and additionally, provides the following primary services to the subscriber device: Dial Tone, Battery Current, and Ring Voltage. You plug your phone into an FXS port.

The complementary member to an FXS is the Foreign eXchange Office (FXO) port. This interface receives POTS service, typically from a Central Office (CO). In other words an FXO interface points to the Telco office. If your ATA contains an FXO interface, then you connect this interface to the jack in the wall.

In May, 2005, a DoS exploit was announced that relied upon sending specially crafted DNS packets to Cisco IP phones, ACNS, Unity Express, and ATAs. The only fix for this was to upgrade to a fixed software revision. This illustrates the requirement to stay informed of current software vulnerabilities, and to maintain some type of regular patching/update cycle.

DNS services have a number of uses within a VoIP environment, the most important being IP address name resolution. In a simple configuration, DNS services may be used simply to map a URI (Uniform Resource Locator) to one or more IP addresses. As VoIP technology and infrastructure arrangements mature, DNS will play a more central role in converting E164 defined telephone numbers to IP addresses via the ENUM framework. One caveat in this arrangement is that synchronization and delegation of DNS servers must be planned and managed carefully in order for the system to function properly.

DHCP and DNS servers should be secured by hardening their respective operating systems, and in the case of DNS, by ensuring that the BIND daemon is patched and up-to-date. Running a recent version of BIND generally means that you are running the most secure version of BIND. Additionally, you should disallow queries from unauthorized nameservers, ensure that only your slave nameservers are allowed to update by requesting zone transfers, and BIND should be run with least privilege—jailing or chrooting the BIND daemon is always good practice. In high security environments it is worthwhile to run TSIG (transaction signatures) between nameservers in order to authenticate DNS messages (see *DNS & BIND*, Albitz & Liu, O'Reilly, 2001 for more detailed information).

DNS traffic also can be difficult to correctly firewall. DNS traffic runs over port 53 via TCP or UDP depending upon the transaction. The problem is that in recent major versions of BIND (8 and 9), nameservers, by default, send queries from random high-numbered ports to port `udp/53` of the resolver (client). Resolvers also send their queries from random high-numbered ports to port `udp/53` of the nameserver. One way to resolve (sorry for the pun) this issue is to *allow from any to port udp/53* in both directions on the firewall. However, this is not a particularly elegant solution in that the control is not very granular. A better solution is to use the *query-source* option to force BIND to send queries from port 53. This enables more stringent control of DNS traffic on the firewall.

## LDAP and RADIUS Servers

LDAP (Lightweight Directory Access Protocol) is a protocol for accessing X.500 directory services. LDAP is the de facto standard for directory-based application, authentication, authorization, and search requests. An LDAP server is essentially a database optimized for read rather than read/write operations. LDAP services provide call routing and subscriber information within a VoIP environment. RADIUS (Remote Authentication Dial In User Service) is an AAA (authentication, authorization, and accounting) protocol for many different types of applications ranging from router and switch access to subscriber AAA in a VoIP environment.

The LDAP directory stores information about objects on a network and makes this information available to applications, users, and network administrators. Using LDAP, authorized network users can access resources anywhere on the network using a single login process. Within the enterprise, LDAP directories often comprise the corporate directory. Much of the data in these types of directories is considered security-critical data because it includes personal information including usernames, passwords, contact information, and, of course, telephone numbers and SIP URIs.

This leads to a conundrum: The location services provided by the LDAP directory server (or more typically, a cluster of these servers) must be quickly and easily accessible by anyone or any machine with the appropriate login credentials. On the other hand, these services must be completely inaccessible by any nonauthorized user. Complicating this scenario is that properly authenticated users must be given enough, and only enough, authorization so that they can access their cognate data and no other.

LDAP and RADIUS security tasks include hardening the operating system that the services reside upon and restricting access to port tcp/389 (LDAP) and ports tcp/1812 and tcp/1813 (RADIUS) to only those agents that require access. Additionally, most LDAP implementations provide for native (though complex) access control in the form of Access Control Lists (ACLs). Proper configuration of these ACLs is critical to securing your LDAP directory server; however, this task must be designed and implemented carefully.

Lastly, LDAP natively provides no protection against sniffing or active attackers, whereas RADIUS provides some protection based upon shared secrets. SSL v3 or TLS are recommended for securing LDAP data while in transmission. Normally these data are received on port tcp/636.

## NTP

Time synchronization often is overlooked during the design of network infrastructure. On a stand-alone computer or network device such as a router or a switch, the time, which usually is based on inexpensive oscillator circuits, can drift by seconds each day. Over time, this drift leads to significant variation in the times of different network clocks. Why is this important for VoIP infrastructure and security?



To begin with, any servers or other networked devices that participate in clusters for load balancing or high availability will act inconsistently if their clocks are not synchronized. Network monitoring services (see the next section) rely upon an accurate clock for determining the root-cause of network outages or delays. In forensic analysis, DHCP leases can be tied to specific workstations if the clocks on all machines are accurately synchronized. Directory services require accurate clocks. Windows 2000 and Windows 2003 are significant examples of this since the default authentication protocol (Kerberos v5) for many domain functions uses the workstation time as part of the ticketing process. Most importantly, from a security point of view, any type of logging, particularly if logs from different hosts are stored on a remote server, relies upon accurate timestamps to correlate specific data with specific events.

For these reasons, it is recommended to create a time synchronization hierarchy as part of the foundation VoIP architecture.

## SNMP

SNMP is vital in VoIP networks, particularly for monitoring discrete systems and for traffic supervision. In addition, many vendors use SNMP as part of the IP telephone configuration process. SNMP traffic, at least for versions 1 and 2, is encoded using ASN.1 syntax and BER encoding; however, it is not encrypted. SNMP v3 traffic can be encrypted.

Unfortunately, the default community strings associated with the most common versions of SNMP (v1 and v2) are well-known and easily guessed. These community strings act as passwords that allow access to the SNMP-managed device. The default read-only community string (public) allows a user to browse configuration information regarding the device or server. Information gathered in this manner can potentially be used to gain further access to the device.

SNMP messages, like syslog messages, can be stolen by eavesdroppers, and these data can be used to determine the state and configuration of networked devices. Routers and switches can be reconfigured as well by the appropriate SNMP commands. Thus, it is recommended to use SNMP v3 for monitoring and configuration of VoIP networks. If the use of SNMP v3 is not a valid option, due to network constraints or a lack of support by networked devices, then it is essential to restrict SNMP to subnets that are segregated from the Internet and from the balance of the network.

This can be accomplished in a number of ways including VLANs, firewalls, and access control lists. These methods are described in more detail in Chapter 8. Note that a number of different vendors' (UTstarcom, Cisco, and Hitachi, for example) IP phones have shipped in the past 18 months with default SNMP read/write strings. This allows any remote user to read, write, and erase the configuration of an affected device. Before you deploy your IP phones, check that the default community strings have been replaced by complex passwords. This highlights a key concept in securing SNMP on any type of network. Always check for the presence of default community strings and if they exist, change them to complex strings.

## SSH and Telnet

SSH and Telnet are real-time protocols that often are used by VoIP system administrators for normal maintenance and troubleshooting. Telnet is a protocol commonly used for remote administration of servers and network devices. A major failing of Telnet is that it passes data in the clear; it uses no encryption. Usernames and passwords used to log into remote devices traverse the IP network unencrypted and are susceptible to interception. Although many network administrators believe that this risk is mitigated by the use of a switched network, techniques and tools exist that allow interception of switched traffic.

In the mid 1990s, as sniffer software became more readily available (i.e., free), system administrators began to search for a secure encrypting replacement for Telnet, rsh, rcp, and so on. SSLTelnet and SNP (Secure Network protocol) are two examples that have faded into history. SSH (Secure Shell) became the de facto choice for secure communication between networked devices. SSH allows an individual to log into another computer over a network, to execute commands on the remote machine, and to move files from one machine to another (SCP). It provides strong authentication and secure communications over insecure channels. A number of free SSH clients exist for both Windows and LINUX operating systems, and almost all servers support the SSH protocol.

Recently, several versions have been vulnerable to the CRC32 Compensation Attack exploit. If you plan to use a version of SSH based upon OpenSSH, be sure to install the most up-to-date version available, run SSH protocol 2, and be sure to disable the option to drop back to SSH protocol 1.

The message in this section is clear: There is no longer a place in any contemporary VoIP network for nonsecure, nonencrypted administrative maintenance or troubleshooting traffic.

## Unified Network Management

Network management tools that are used on the data network can be used to monitor the entire converged infrastructure. This is one of the major advantages of a converged network. Existing network management tools may need to be updated to reflect the enhanced requirements of a VoIP network. If possible, management traffic should be segregated to an out-of-band, dedicated management network.

Proactive management of this complex environment ensures that the quality of voice calls will fall within acceptable limits. Voice quality is made up of both objective and subjective factors. The objective factors in assessing VoIP quality are delay, jitter, and packet loss. Delay is defined as the time it takes a packet to traverse the network from the sending node to the receiving node. It usually is estimated as the round-trip-time (RTT) divided by 2. Jitter is defined as the variance or change in delay times. If RTT are greater than 250 to 300 msec, then voice quality will suffer. All three measurements are interrelated. Studies have shown that

the greater the jitter in a VoIP environment, the greater the packet loss. VoIP does not tolerate packet loss (dropped media packets are not resent), thus the greater the packet loss, the lower the voice quality. Active monitoring and management of voice quality in a VoIP environment is a must to help identify and reduce such undesirable occurrences.

If you are responsible for network monitoring in your own VoIP environment, then a number of tools—in a range from freeware to expensive commercial—are available to you. At the low end (price-wise, but not feature-wise) are tools like MRTG, NTOP, Nagios, and a host of other SNMP-based agent-managers. At the high end, tools like HP OpenView, Tivoli, and SMARTS not only discover and manage network objects, but in some cases, attempt to determine the root cause of network problems. The key security issue in rolling your own security monitoring infrastructure is that you segregate management traffic to a dedicated, secure, management network. The other key point is that managing your own network monitoring professionally requires that you dedicate human beings to the task of reading, analyzing, and acting upon the resultant data.

Many clients rely upon third-party remote management of VoIP infrastructure components. How do you choose between differing vendor offerings? What are the criteria you should use when making this decision? Hopefully, the next several paragraphs will give you some insight into this process.

First, you will require a secure and auditable path between your managed sites and the vendor sites that support remote delivery of services. One of the most challenging problems in remote management of large networks is the complexity of security administration. This can be a difficult issue to solve technically as mutual trust, at some point, becomes an issue. Technical workarounds for this include multiple layers of firewalls—some of which are managed by each party; coincident visualization of all encrypted traffic that spans the two networks; and strongly typed, enforced, and audited role-based access controls (RBAC).

You should specify that the remote management services incorporate a standards-based approach that enables secure maintenance access and monitoring for multivendor services support. Standards will enable visibility into the processes that are used to monitor your network. Check that all regulatory requirements that are relevant for your particular industry are met, including a strong audit trail for all transactions. Ensure that the remote management vendor provides a single point of alarm consolidation, ticketing, and inbound/outbound access to the corporate network; and that a customer self-service maintenance portal with unrestricted access to audit trail information and reports is available. Last, be certain that you retain access and control of the devices within your own infrastructure.

## Sample VoIP Security Policy

In this section we'll discuss the components of a sample VoIP security policy.

## Purpose

VoIP is a highly critical data application and as such, is subject to all the policies detailed in other data security policy sections (this assumes that the VoIP Security Policy module is part of a larger set of security policy modules). The purpose of this section is to provide an additional checklist to ensure that VoIP systems sharing the data network as a converged technology are implemented in a secure fashion.

## Policy

Security in an IP telephony environment includes all the security features of traditional telephony and adds all the security concerns of the data network. IP telephony converts voice to data and places these data into IP packets. As such, these packets can be “sniffed” just like any other data packet on the network, thereby raising serious issues of confidentiality. The operating systems underlying IP-PBXs and other gateway devices are susceptible to the same attacks that regularly disrupt other types of servers.

## Physical Security

IP-PBX equipment must be located in a locked room with limited access. This type of access must be provided as a user authentication system with either a key-card or biometric device. The use of a keypad alone to gain access is not permitted. All methods of gaining entry into the room must provide for a list of users that have accessed the room along with a date/time-stamp.

## VLANs

Logical separation of voice and data traffic via VLANs is required to prevent the VoIP streams from broadcast collisions, and to protect data network problems from affecting voice traffic.

## Softphones

Softphones that contain any type of advertising software must be banned in a highly secure environment. Softphone installation targets should be tested before deployment and those that do not encrypt user credentials should be prohibited.

Because a softphone is an application running on an operating system, its security depends principally upon the status of the underlying OS, and is subject to the same security concerns as any other communications program including e-mail, browsing, and IM.

## Encryption

All VoIP systems should use a form of Media (RTP channel) Encryption in order to avoid the sniffing of VoIP data. All communications between network elements should be

encrypted. Complete end-to-end IP voice encryption is recommended to mitigate the threat of eavesdropping attempts. Additionally, all administrative access to critical server and network components must use encrypted protocols such as SSL and/or SSH. All access to remote administrative functions should be restricted to connections to the switch itself or to a designated management PC.

## Layer 2 Access Controls

The most comprehensive solution is to require all devices to authenticate on layer two using 802.1X before receiving layer three (IP) configuration settings.

Additionally, consider enabling port security as well as MAC address filtering on distribution switches. The port security feature of these devices provides the ability to restrict the use of a port to a specific MAC address or set of MAC addresses. It is generally considered that this is difficult to implement and maintain, but with proper planning, port security does not have to be difficult. Several third-party tools are available to help manage and maintain port security in enterprise environments.

## Summary

In this chapter, we have discussed many of the ways that you can reuse portions of your existing security infrastructure as you prepare to add voice traffic to the mix. After you or your management has made the decision to move to a converged network, and before the new architecture is completed, it is important that one or more representatives of the security group participate in the architectural discussions. “Bolting on” security components and processes after the network and application architecture is finalized just doesn’t work. Security as an afterthought usually results in a network that is insecure, as well as users that are frustrated because they now have to “do things differently.”

Adding VoIP to your network may introduce additional risks, so your first step is to review your existing security policies. Do they exist at all? If so, are they current? Do most associates know where to find them? Do people understand their responsibilities?

In the section on Security Policies, we discussed the steps involved in formulation of policy. We talked about implementation and communication of the policy guidelines, as well as who should be involved in the process. A sample VoIP Security Policy module is located near the end of the chapter. Feel free to use this as a template for your own policies.

In the section on Physical Security, we discussed some of the measures and physical controls that are needed in a VoIP environment. A truly dedicated attacker, finding little means of accessing an organization’s internal IP network over a public network such as the Internet, often will turn to physical penetration to bypass the organization’s logical perimeter security controls. This is not just a theoretical vulnerability; numerous incidences of attackers using physical penetration to bypass logical perimeter security controls have been reported in the mainstream media. A comprehensive security strategy must consider the efficacy of physical perimeter security as well as its logical or technical perimeter security.

The section on Server Hardening went into some detail regarding hardening of specific platforms and the rationale for doing so. All hosts attached to the VoIP network should follow a standard build procedure and be subjected to hardening before they are connected to the network. One group within the organization should bear the responsibility for maintaining standard build and hardening guidelines for Windows, Linux, AIX, and other UNIX and UNIX-like operating systems. This group should define these guidelines, ensure that these hosts are hardened and patched before deployment, and ensure that patches are updated periodically as appropriate. This group should also maintain a central registry of individuals and groups running these operating systems so that periodic audits can be conducted to guarantee that the systems do not deviate from the established security baselines.

The section on Supporting Services described the functions and security characteristics of VoIP supplementary services. The servers that host these services should be hardened and patched per security policy guidelines. Hardening of these servers, as mentioned earlier, should follow the principle of “Least Privilege.” This means that anything not required

should be disabled. Turn off all unneeded services. Disable any features that are not in use. Remove unnecessary applications.

Last, the section on Unified Management detailed some of your responsibilities when designing the monitoring network for your VoIP infrastructure. Many open source network-monitoring tools exist that work as well as more expensive commercial packages. The trade-off is that the open source tools are usually more difficult to set up and maintain than their commercial counterparts. If you decide to outsource your network management tasks, make certain that you have defined in detail the SLAs, network topology, trust relationships, and reporting requirements.

This design period is an excellent time to inventory, unravel, and review your existing security infrastructure. It makes good business sense to reuse and recycle devices and processes that have worked in the past, and to eliminate those that don't work or those that do not provide a reasonable ROI.

At this point, you have updated your security policies to reflect the addition of voice to your data networks. You have physically secured the VoIP and data infrastructure components so that it is impossible (or at least unlikely) that unauthorized individuals have direct access to these components. You have hardened and patched servers, routers, switches, and other supporting devices so that they are resistant to common exploits. And you have determined how you will monitor your infrastructure.

## The IP Multimedia Subsystem: True Converged Communications

Solutions in this chapter:

- IMS Architecture
- Communication Flow in IMS
- IMS Security Architecture
- IMS Security Issues



# Introduction

The IP Multimedia Subsystem (IMS) is a next-generation multimedia communication framework that encompasses mobile, fixed, packet-switching, and traditional circuit-switching communication systems. It has been proposed by the Third Generation Partnership Project (3GPP) and uses the Voice over Internet Protocol (VoIP) framework, especially the Session Initiation Protocol (SIP) standard. The 3GPP is a standards organization driven by wireless carriers and equipment manufacturers and aims at producing globally applicable specification and reports for Third Generation (3G) Mobile Systems based upon GSM (Global System for Mobile Communication). The 3GPP also is working with the Internet Engineering Task Force (IETF) on SIP-related standards. The goal of the IMS is to provide a wide spectrum of services with ease and consistency. These services include videoconferencing, Push-to-Talk (PTT), Text-to-Speech (TTS), instant messaging (IM), content sharing, and multipart gaming. To achieve this goal, IMS uses an open standard IP protocol and extension of SIP.

SIP is standardized by the IETF as multimedia signaling protocol. Its architecture is highlighted by a User Agent (UA), which is a terminal, and a set of servers, including proxy server, registration server, redirection server, and so on. From its inception, SIP has been gaining momentum due to its open architecture and extensibility to mobile device and multimedia communication. Because SIP signaling protocol is based on Hypertext Transfer Protocol (HTTP) and ASCII-based encoding, it is easy to understand. SIP also sits side by side with a standard media protocol named Real-time Transport Protocol (RTP), also standardized by the IETF. RTP is a vehicle with which any kind of multimedia can be transmitted as long as the media is digitized with a proper codec. For instance, one of the popular voice codecs in VoIP systems is G.711. This codec samples voice streams at 8,000 times/sec, also digitizing and transmitting them at 64K bit/sec. With a G.711 codec, each packet has a 160-byte payload (214 packet size with header information), and each VoIP phone sends 50 packets in one second. If there is a need for compression, a VoIP phone can transmit the digital voice with a compressed format (for instance 8K in G.729 format), thereby saving bandwidth. Video can be easily transmitted with SIP and RTP protocols if it is digitized with an appropriate format such as H264 or any of the MPEG series formats such as MPEG1, MPEG2, MPEG4, or H.264.

IMS architecture extends to the SIP and 3GPP architecture and adds several components suitable for mobile communication. It was driven originally by the 3GPP to boost the packet-switched services and attract more users to the packet-switched domain. To do so, it adds three important features in the GSM-based packet-switched network that already offers Internet services such as surfing the Web and accessing e-mail:

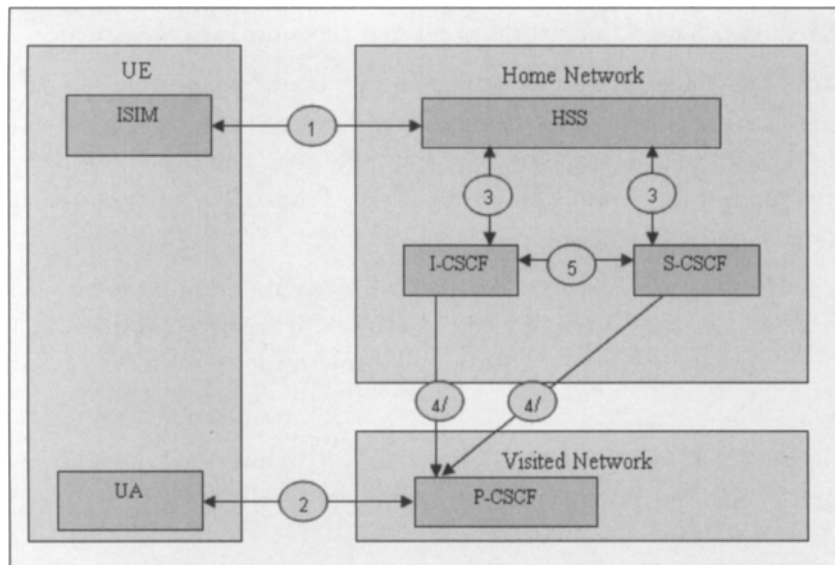
1. It requires QoS (Quality of Service) in a session, so that the quality of service can be met. Existing packet-switched network does not guarantee the quality of service and users may have a fluctuation of service depending on the situation.
2. It adds a flexible charging mechanism so that operators are able to charge appropriately for multimedia services. For example, operators can charge the video conferencing based upon the time used or any other reasonable measure rather than the bandwidth that the service consumes, which may end up being unacceptable to a customer.
3. It provides integrated services to users and offers ample room for third parties to provide services. Operators don't need to stick to the services that the large equipment vendors offer, but have flexibility to offer a variety of services developed by third parties.

The architecture of IMS extends the SIP and 3GPP architecture and adds several components that are suitable for mobile communication.

## IMS Security Architecture

IMS has its own security architecture in addition to the general 3GPP security architecture named Network Domain Security. Network Domain Security addresses security issues at the IP layer and recommends IPSec as the basic security mechanism among Security Gateways (SEGs). SEG is a security component that sits in each network domain and communicates with SEGs of the destination domain. As shown in Figure B.1, the IMS-specific security covers the security issues between the IP Multimedia Services Identity Module (ISIM) and HSS (path 1), UA and P-CSCF (path 2); the Network Domain Security covers the other paths that are implementing IP protocol (paths 3, 4, 5). ISIM and UA collectively are called UE (User Equipment). Suppose that P-CSCF and S-CSCF are located in different networks, each of which implements its own security policy. In this case, all the traffic between P-CSCF and S-CSCF traverse two SEGs in such a way that packets are encrypted in a SEG and decrypted in the other end of SEG. The two SEGs have their own security binding (Key exchange and encryption/decryption algorithm), implementing IPSec.

Figure B.1 IMS-Specific Security



The Security mechanisms of paths 1 and 2 are specified in the security document as follows (see the fifth reference in the Related Resources section):

1. Provides mutual authentication. The HSS delegates the performance of subscriber authentication to the S-CSCF. However the HSS is responsible for generating keys and challenges. The long-term key in the ISIM and the HSS is associated with the user private identity (IMPI). The subscriber will have one network internal IMPI and at least one external user public identity (IMPU).
2. Provides a secure link and a security association between the User Equipment (UE) and a P-CSCF for protection of the Gm reference point.

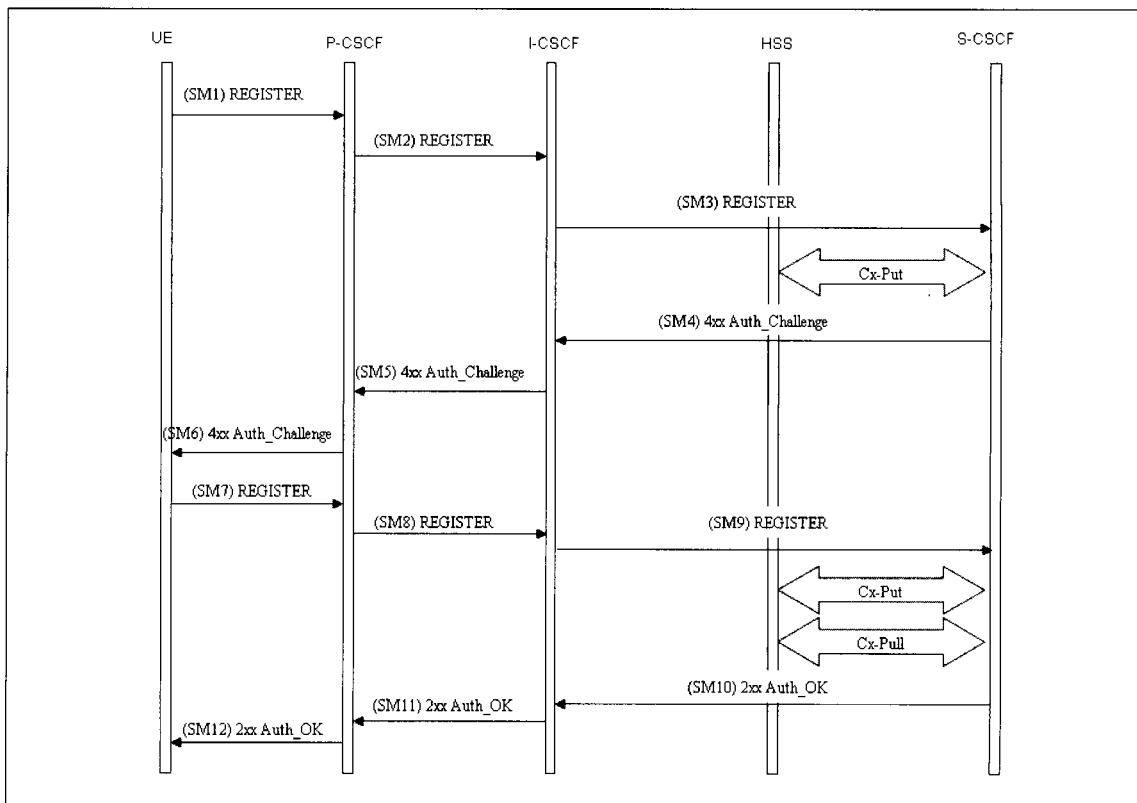
IMS security covers the initial secure authentication that occurs between the ISIM and the HSS in which user devices are authenticated through a secure link. Once the user device is authenticated and allowed to use IMS, the IMS security provides a secure communication mechanism through which all the information can be transmitted.

To summarize IMS security, all the UAs are authenticated before they are allowed to get into the system. The HSS is the central component for the security policy. The HSS gives commands for what kinds of security algorithm is used and provides correct authentication information for all the users. Each UA or ISIM, which is a term indicating the collection of IMS security data and functions on a Universal Integrated Circuit Card (UICC), has built-in authentication information in the UICC.

IM-subscribers have their profile information stored in HSS in their home network. At registration, an S-CSCF is assigned to the subscriber by the I-CSCF. The subscriber profile will be downloaded to the S-CSCF over the Cx-reference point from the HSS (Cx-Pull). When a subscriber requests access to the Network, the assigned S-CSCF decides if the subscriber is allowed to continue with the request or not; that is, Home Control (Authorization of IM-services). These procedures are carried out with a security association, so that all the information can be transmitted encrypted, and therefore, not tampered with.

The authentication message flow is described in Figure B.2. (See the fifth reference in the Related Resources section.) If a user is permitted to get into the IMS service, at least an IMPU has to be registered in HSS in advance and the corresponding IMPI has to be authenticated. When a UE wants to be registered, it sends a SIP REGISTER message to the SIP registrar (in this case the corresponding S-CSCF).

**Figure B.2 Authentication Message Flow in IMS**



The SIP REGISTER is passed to I-CSCF and to the corresponding S-CSCF that refers to HSS, and retrieves all the authentication information of the user from HSS with the Cx-Pull method. S-CSCF uses an Authentication Vector (AV) to conduct authentication and

key agreement with the user, where the AV consists of five elements: a random number RAND, an expected response XRES, a Cipher Key (CK), an Integrity Key (IK), and an authentication token AUTN. If S-CSCF has no valid AV, S-CSCF sends an AV request to HSS together with the number of AVs (for instance,  $n$ ) wanted.

Upon receipt of the AV request from S-CSCF, HSS sends an ordered array of  $n$  AVs. Once S-CSCF receives them, it uses them on a first-in/first-out basis and sends them for authentication challenge to users. The S-CSCF sends a SIP 4xx Auth\_Challenge, that is, an authentication challenge to the UE including the challenge RAND, the authentication token AUTN in SM4. The challenge also includes the IK and the CK for the P-CSCF. The S-CSCF also stores the RAND sent to the UE for use in case a synchronization failure should occur.

When the P-CSCF receives SM5 it stores the key(s) and forwards the rest of the message to the UE, so that the key is not revealed in the clear text. When the UE gets the challenge, SM6, it takes the AUTN, which includes the MAC and a sequence number SQN and checks if the MAC is the same as XMAC and the SQN is correct. If both checks are successful, UE computes the authentication information, response RES using the random number, and authentication token. Then the UE puts it into the Authorization header and sends it back to the P-CSCF that forwards it to the S-CSCF.

When S-CSCF receives the SM9, it retrieves the active XRES for that user and checks the validity of the authentication information sent by the UE. If the user is successfully authenticated, the S-CSCF sends a SIP 2xxx Auth\_OK message to I-CSCF that forwards the same message to the UE and completes the authentication procedure.

## IMS Security Issues

IMS was from its inception designed to be secure to eliminate many of the vulnerability issues that plague existing packet-based communication systems. The security of IMS has been especially fortified with the built-in security functions of IPv6. For instance, the use of IPsec would eliminate the vulnerabilities such as eavesdropping, tampering, and IPspoofing. However, it is expected to take a substantial amount of time to fully migrate from the existing IPv4-based network to IPv6. Hence 3GPP came up with a compromise solution called early IMS. Early IMS uses IPv4 and it is expected that this model will be a popular implementation in the early stages of IMS. Some early IMS may not be fully compliant with the security features defined in TS 33.203 because of the potential lack of support ISIM interface and inability to support the IPsec on some UE platforms. Because IMS implementation is based on SIP, it also carries as many security vulnerabilities as SIP. With the full IMS implementation based on IPv6 and when the security is put in place, it is inevitable that IMS will have Denial of Service (DoS) attacks.

# SIP Security Vulnerabilities

First let's review some of the security vulnerabilities of SIP, so that we can better understand the security issues faced by early IMS. SIP was designed to make the communication system standard and open to any other system compliant with SIP standards. SIP has many security vulnerabilities and is susceptible to being breached by hackers. The following list presents several well known SIP vulnerabilities.

## Registration Hijacking

SIP has a registration hijacking vulnerability that is similar to man-in-the-middle attacks. An attacker sniffs a REGISTER message from a legitimate user and modifies it with its own address as the contact address. Getting this fake message, the SIP registrar updates the contact address belonging to the legitimate address with the fake address. When incoming calls are received for the legitimate users, the proxy server refers to the registrar and redirects all the incoming calls to the fake address, which makes the man-in-the-middle attack successful.

## IP Spoofing/Call Fraud

An attacker impersonates another legitimate user with spoofed ID and sends an INVITE or REGISTER message. In IPv4, there is no way to block IP spoofing when SIP messages are sent in clear text and an attacker is able to use an arbitrary IP address easily. Hence, when an arbitrary IP address is sent to the registrar with the legitimate user account, the incoming calls that follow are transferred to the wrong address and are never sent to the correct user. When an INVITE message is sent to a user with an arbitrary destination IP address, the call is never sent through or connected to the hacker's terminal. If a hacker can use a legitimate IP address and make a call with that IP, he or she can execute call fraud and make free calls.

## Weakness of Digest Authentication

SIP recommends the use of the HTTP digest authentication that is based on the MD5 digest algorithm. However, the MD5 digest algorithm is weak and cannot be used in an authentication system requiring high security. At the same time, SIP digest authentication algorithm does not include all header fields, which can be forged as well.

## INVITE Flooding

A hacker keeps sending INVITE messages with a fake address and paralyzes the user terminal or SIP proxy server. This attack is quite similar to SYN Flood attacks in TCP connections.

## BYE Denial of Service

A SIP signaling packet by default is sent in clear text and can be tampered with. If a hacker sniffs legitimate INVITE messages, he can counterfeit a legitimate BYE message and can send it to one of the communicating parties, resulting in tear-down of the ongoing conversation.

## RTP Flooding

RTP Flooding is related to media transmission. Most media transmissions are based on RTP once the communication is set up with SIP signaling. With RTP Flooding, a hacker makes fake RTP packets and bombards either of the ends with the fake RTP packets, resulting in quality degradation or terminal reboot.

## Spam over Internet Telephony (SPIT)

A SPIT threat sends unsolicited calls to legitimate users that contain mostly prerecorded messages and that annoy people or congest a voicemail system to overflowing.

## Early IMS Security Issues

Early IMS refers to IMS systems that are not compliant with full IMS security. One of the key characteristics of early IMS is that it does not use the security binding between UE and P-CSCF, and thus it provides neither the integrity nor the message confidentiality that should be afforded to messages passed among UE, HSS, and P-CSCF. Therefore, IPsec is not used. Yet, the lower level Network Domain Security might provide security among lower level components such as SEGs.

Early IMS security addresses the threat of IP spoofing and presents a way to avoid this threat if full IMS security is not in place. As exists in SIP vulnerability, IP spoofing enables hackers to use an IMS account or IP address freely. To prevent IP spoofing, early IMS security recommends the use of a RADIUS server in connection with HSS to check the IP address of the IMPU. Early IMS security features combined with the use of a RADIUS server and HSS restricts use to only one IMPU at a time and registers it with the legitimate IP address. Hence, multiple IMPUs cannot be used at the same time for individual users. Because of this restriction, users are not allowed the use of multiple devices such as mobile phones, VoIP phones, and Personal Digital Assistant (PDA)s. Once the IMPU is registered with the legitimate IP address, the hacker is not able to spoof his IP address with a legitimate user account, since it does not match in the RADIUS server.

However, there are still several security vulnerabilities left in early IMS security. Because early IMS does not use passwords or secure keys, it might be easy for the hacker to sniff the legitimate REGISTER message and counterfeit it. Once the legitimate user deregisters from the HSS, the hacker is able to reregister with his or her own IP address bound to the legiti-

mate user account and get into the system. When the hacker gets into the system with his or her own IP address, the legitimate user is no longer able to get into the system. This vulnerability is just one aspect of early IMS security issues. There are still DoS attacks and SPIT issues that leave early IMS systems unprotected.

One thing to note is that early IMS security does not adopt HTTP digest authentication as in SIP, which requires a user account with its password and then encrypts them with a digest algorithm. One of the main reasons for this is that HTTP digest can allow multiple users to get into the system at the same time if they share the same account and password. This makes appropriate billing difficult and thus, can have an impact on the service provider's revenue.

## Full IMS Security Issues

Full IMS security includes the security architecture that implements IPv6 and IPSec among IMS components. All user terminals (collectively called UE) have security keys and can encrypt messages as well as include digital signatures for secure authentication. These characteristics protect from eavesdropping, tampering with messages, and IP spoofing. Full IMS security also is designed to block potential replay attacks since the encryption is based on the random numbers generated by HSS that are valid for a certain period of time.

Full IMS security tends to eliminate many of the vulnerabilities posed by SIP. Yet, full IMS contains a certain degree of DoS vulnerabilities and SPIT problems. An attacker can capture the encrypted packets and figure out the IP addresses of P-CSCF or S-CSCF if transport mode is used with IPSec. Attackers can also bombard servers with massive DoS attacks with SYN Flooding or other kinds of attacks. If the network has a firewall and implements appropriate security policies such as rate limiting, the hacker at least might be able to use up all the network bandwidth and disrupt IMS services. Full IMS security is also vulnerable to SPIT attacks. A legitimate UE is able to get into the system and send SPIT attacks to target users easily. It can also compromise many servers and mastermind distributed DoS SPIT attacks using the compromised servers. It may be quite difficult to detect DoS SPIT attacks since each compromised server acts like normal users, occasionally sending stealth calls. Therefore, a proper protection mechanism is still to be designed.



## Summary

IMS promises a nice integration of IP and cellular networks. It allows both users and operators to take advantage of benefits of both sides. A mobile phone user gets comprehensive services available in both cellular network and Internet. At the same time, the embedment of QoS (Quality of Service) in the communication session improves the preservation of service quality. It prevents the users from suffering the quality degradation. Operators can also benefit from it since they have flexible control over charging mechanism. They can select the appropriate charging method based upon the bandwidth or duration of time, so that they have higher revenue. Also the operator can select third-party services on top of the given services freely, which allows the operator to provide a variety of service to users.

IMS enables many feature sets of convergence services but opens the IP network to security vulnerabilities. IMS addresses some security issues like unauthorized use, privacy, and denial of services. A built-in IPsec makes it hard to do packet forgery, eavesdropping, and IP spoofing and session hijacking. Nevertheless, there is still room for the hackers to disrupt the service by layer 2 and 3 DoS attacks and Voice Spam attacks. Additional security mechanisms like spam blockers and IPS are needed to prevent these attacks.

## Related Resources

1. *SIP: Session Initiation Protocol*, IETF RFC 3261, [www.ietf.org/rfc/rfc3261.txt](http://www.ietf.org/rfc/rfc3261.txt), June 2002.
2. ETSI TS 123 002 V6.10.0, Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); Network architecture, December 2005.
3. Peter Howard, *Sipping IETF51 3GPP Security and Authentication*, September 2001, [www3.ietf.org/proceedings/01aug/slides/sipping-7/](http://www3.ietf.org/proceedings/01aug/slides/sipping-7/).
4. *ETSI TS 133 210 V7.0*, Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); 3G security; Network Domain Security (NDS); IP network layer security (3GPP TS 33.210 version 7.0.0 Release 7), December 2005.
5. *ETSI, ETSI TS 133 203 V7.0.0*, Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); 3G security; Access security for IP-based services, 3GPP, December 2005.
6. *ETSI TR 133 978 V6.3.0*, Universal Mobile Telecommunications System (UMTS); Security aspects of early IP Multimedia Subsystem (IMS) (3GPP TR 33.978 version 6.3.0 Release 6), December 2005.
7. *RTP: A Transport Protocol for Real Time Application*, IETF RFC 3550, July 2003, [www.ietf.org/rfc/rfc3550.txt](http://www.ietf.org/rfc/rfc3550.txt).
8. Dongwook Shin and Choon Shim, *Voice Spam Control with Gray Leveling, Second VoIP Security Workshop*, June 2006, Washington D.C.

## Regulatory Compliance

### Solutions in this chapter:

- SOX: Sarbanes-Oxley Act
- GLBA: Gramm-Leach-Bliley Act
- HIPAA: Health Insurance Portability and Accountability Act
- CALEA: Communications Assistance for Law Enforcement Act
- E911: Enhanced 911 and Related Regulations
- EU and EU Member States' eCommunications Regulations

# Introduction

The past decade has seen an explosion of government regulation that will directly or indirectly affect VoIP implementation security. Some of these regulations can be addressed by selecting and implementing compliant equipment, but the vast majority of these are *operational* in nature, meaning that to ensure compliance you'll need to pay more attention to (1) how your IP communications systems are designed and (2) how your organization's business and IT operations groups are using the equipment once it's live.

For this chapter, each applicable set of regulations will be discussed separately. What you'll want to ask yourself in each section is:

- Does this regulation apply to me and my organization (or my client's organization)?
- Who in my organization has responsibility for overall compliance with this regulation? In some cases, the answer may be you if there isn't already someone designated, but for many of these regulations your organization is likely to have a person or group specifically designated as the lead for addressing compliance, particularly with regulations for which security is only an ancillary component of the overall regulation.
- Is it likely that my systems and/or operations are not compliant with this regulation today? If you suspect that remediation is necessary, it's important to raise the concern to the appropriate level of management in a way that allows the issue to be corrected and reduce the risk of fines, negative publicity, or worse.



## WARNING

---

Always consult experienced legal counsel (or your organization's audit or compliance department) for legal advice with regulatory issues that could materially affect your organization. Although this chapter highlights the most common regulatory concerns surrounding VoIP, it cannot provide complete guidance for every situation or jurisdiction. For instance, VoIP itself is considered illegal in certain countries when it bypasses national carriers (sometimes known as PTTs) who may have a telecommunications monopoly. And new data privacy laws around the world seem to appear monthly.

---

**NOTE**

Despite the aforementioned caveat, you may find that the compliance experts available to you are not familiar with VoIP and how to apply broad regulations like GLBA or HIPAA to voice and other real-time communications systems. To help with these situations, pay special attention to the “Tools & Traps” sidebars in this chapter. They will provide specific guidance for you to share with a specialized compliance expert in that area of regulation.

Don't be surprised, however if your expert chooses to ignore the additional information. Many of the experts I've met with over the years prefer to apply these regulations narrowly and don't want to open the door to unanticipated compliance costs (common concern for internal experts) or expand the scope of compliance work without having the billable expertise to address it (typical for external experts). If that happens to you, just make sure to complete your due diligence by advising your organization's responsible executive (corporate counsel or chief compliance office) of your concerns in writing and leaving the matter in their hands.

In the next six sections, we'll review regulations that may affect you or your organization. You may safely skip some of them, so here's a quick way to tell which sections won't apply to you and your organization:

- If your organization is not public (listed on any U.S. stock exchange), then you can skip SOX.
- If your organization isn't involved with banking, consumer finance, securities, or insurance, then you can skip GLBA.
- If your organization doesn't handle any medical records (don't forget your HR department and any health insurance-related records when considering this question), you can skip HIPAA.
- If you're not a telecommunications carrier (or effectively replace one, like a university does for on-campus students, for example), then you can skip CALEA.
- If you don't have any physical locations in the United States or provide phone service there, you can skip E911.
- If you don't have any customers, suppliers, or operations in an EU country, then you can skip the EU section, though if you operate in a state or country with data privacy regulations then this section might still be relevant.

# SOX: Sarbanes-Oxley Act

Enacted in response to corporate scandals at Enron, Tyco, and Worldcom during 2001, the Sarbanes–Oxley Act of 2002 was designed to bolster confidence in the financial reporting of publicly traded corporations in the United States. When he signed the Act into law, President Bush hailed it as “the most far reaching reforms of American business practices since the time of Franklin Delano Roosevelt.” Since that time, an estimated \$5 billion has been spent by U.S.-listed corporations to comply with the act.

## SOX Regulatory Basics

Let’s take a few minutes to go through the Sarbanes–Oxley Act and what it requires, starting with what the regulations themselves explicitly require. Then we’ll look at related recommendations that SOX consultants and auditors are likely to recommend above and beyond the explicit legal requirements.

### Direct from the Regulations

When it comes to VoIP or any other IP application, Section 404 is the only part of SOX that even remotely applies. Section 404 isn’t long but since it’s been the basis for hundreds (perhaps thousands) of costly IT reporting and process changes ultimately attributed to Sarbanes–Oxley over the past few years, I’m going to reproduce it in its entirety—but first here’s the simple version:

- 404(a) requires an annual report from management regarding the effectiveness of internal controls.
- 404(b) requires an independent auditor to report on (and attest to) management’s annual report.

So we’re really just talking about two reports here: one that’s signed by the officers of a company, and another that’s signed by their independent auditor (typically from a large accounting and consulting firm). However, since a negative report could have huge consequences in the stock market, being able to produce an acceptable report supported by your auditor is a big deal

Here’s the actual text of Section 404 of the Sarbanes–Oxley Act of 2002:

#### **Section 404 Management Assessment Of Internal Controls**

(a) **RULES REQUIRED-** The Commission shall prescribe rules requiring each annual report required by section 13 of the Securities Exchange Act of 1934 (15 U.S.C. 78m) to contain an internal control report, which shall—

(1) state the responsibility of management for establishing and maintaining an adequate internal control structure and procedures for financial reporting; and

(2) contain an assessment, as of the end of the most recent fiscal year of the issuer, of the effectiveness of the internal control structure and procedures of the issuer for financial reporting.

**(b) INTERNAL CONTROL EVALUATION AND REPORTING-** With respect to the internal control assessment required by subsection (a), each registered public accounting firm that prepares or issues the audit report for the issuer shall attest to, and report on, the assessment made by the management of the issuer. An attestation made under this subsection shall be made in accordance with standards for attestation engagements issued or adopted by the Board. Any such attestation shall not be the subject of a separate engagement.

Now, if you've been part of an internal "SOX audit" you may be saying to yourself, "So where does it say I need to have complex passwords and encrypted links and quarterly user reviews and vulnerability testing and so forth?" And that's an excellent question because, of course, it doesn't say that at all. In fact, even the new internal controls audit standard ("Auditing Standard No. 2" or AS2) created by the Public Company Accounting Oversight Board (an organization created by the Act) addresses information technology only in terms of internal controls.

However, since Section 404 clearly states that the independent auditor must validate management's internal controls report, this gives management a strong incentive to defer to the auditor. As many large public companies found out in 2004 and 2005, a "disclaimer opinion" from an auditor suggesting that a company's internal controls are inadequate tends to push down its stock price. Thus, the security best-practices advice given by an auditor or SOX consultant is very likely to be driven down through an organization as if the law itself required it when that's not strictly true.

Nevertheless, since Section 404 speaks in terms of "internal controls," it only makes sense to ask what an internal control really is. The commonly accepted definition comes from the Committee of Sponsoring Organizations of the Treadway Commission (COSO):

**Internal control is broadly defined as a process, effected by an entity's board of directors, management and other personnel, designed to provide reasonable assurance regarding the achievement of objectives in the following categories:**

- Effectiveness and efficiency of operations.
- Reliability of financial reporting.

- **Compliance with applicable laws and regulations.**

What's most important to note about this definition is that it's *not* made in terms of technology (although organizations routinely use information technology as a *part* of the implementation of many internal controls). It's not just a report, or a policy, or a line of code by itself; rather it's an entire operational process. Given that definition, it's easy to see that SOX really doesn't care if you're using VoIP or telepathy for your business communications so long as any associated internal controls (such as those for billing) are adequate. The critical standard to be met in designing a control is "reasonable assurance"—not absolute assurance. According to COSO, adequate controls should provide visibility and focus but cannot be expected to take the place of effective management:

**The likelihood of achievement is affected by limitations inherent in all internal control systems. These include the realities that judgments in decision-making can be faulty, and that breakdowns can occur because of simple error or mistake. Additionally, controls can be circumvented by the collusion of two or more people, and management has the ability to override the system. Another limiting factor is that the design of an internal control system must reflect the fact that there are resource constraints, and the benefits of controls must be considered relative to their costs.**

In other words, design with the assumption that management can make appropriate executive decisions given the necessary background and context. If your control provides that level of input to decision-makers, it is adequate.

## What a SOX Consultant Will Tell You

External auditors and other SOX consultants hired by your company have many incentives to provide broad, conservative guidance regarding SOX best practices. Why? First, given Arthur Andersen's collapse in the wake of the Enron debacle, one lesson learned by the large audit firms was the importance of giving conservative guidance even if management might take issue with the cost/benefit ratio. Keep in mind, however, that your company's independent auditor is prevented by SOX from offering nonaudit (consulting) services, so these recommendations may force another consulting firm to join the process.

For these additional consultants, comprehensive recommendations on their part tend to increase the length and scope of their billable engagements. And they don't have to worry about jeopardizing a long-term audit relationship through a failed project. So with your management more concerned about passing the next SOX audit than the business value being derived from SOX-related work, a SOX consultant is much more likely to recommend embarking on a comprehensive security strategy in the name of SOX compliance. And the independent auditor has no good reason to suggest to management that the extra work is unnecessary, as that could only increase their liability in the post-Enron world.

If you're involved with security, that dynamic is a double-edged sword. On the plus side, some security best practices that you may have unsuccessfully lobbied for in the past are suddenly now the new law of the land in your company, with the full support of your CIO and CFO arriving in the name of SOX compliance. On the other hand, all sense of perspective when it comes to risk management seems to have been lost in the process. Millions of dollars are spent to implement solutions like enterprise role definition (ERD), single sign-on (SSO), and identity and access management (IAM). At the same time, labor-intensive tasks like a quarterly user review that cannot be outsourced to consultants are taking large chunks of time from the operational resources that you need in order to address risks not tied to SOX at all. And good security practices not tied to SOX may fall off the management's radar screen entirely.

So what specific recommendations are you likely to get from a SOX consultant for a VOIP system? Primarily, these are security best practices you may already be familiar with. Here's what you might expect in a thorough SOX examination of a VoIP system that is deemed to have internal financial controls (because of external billing or internal charge-backs, for example):

- **Logging and audit trails** Does your VoIP system log administrative changes and provide basic usage logs (in this case, Call Detail Records (CDRs) or something equivalent)? If a billing process requires those logs then what is protecting them? More broadly, are the associated internal controls around that billing system adequate? Are lists of authorized administrators and users reviewed for accuracy on a periodic basis (at least annually)?
- **Password complexity** Does your organization enforce consistent requirements for password complexity across applications, including the VoIP system? For example, a password must be at least eight characters with at least one uppercase letter and one non-alpha character. Also, are default administrative passwords changed to comply (or default users removed)?
- **Password expiration** Does your organization enforce consistent expiration timeframes (example: 90 day expiration, 10 day warning) for passwords across all applications, including the VoIP system? Also, are accounts with expired passwords removed after a set timeframe?
- **Database user management** Do associated databases enforce password complexity and expiration rules? Are default database users removed or assigned new passwords that comply?
- **Server (and database) vulnerability management** Do associated servers/databases receive regular vulnerability scans, virus scans with regular updates, and security patches as part of a vulnerability and patch management system?



- **Server hardening** Are unnecessary services, packages, and tools removed from the VoIP system? Are all VoIP processes running as a nonprivileged user?
- **Encrypted IP communications** Do all administrative and operational links prevent user data, passwords, and any other sensitive information from being seen in the clear? This means that Telnet and ftp have been replaced with their TLS-based equivalents (like ssh, sftp), external database connectivity runs over TLS, and (on a VoIP system) that signaling and media encryption are used.
- **Role-Based Access Control (RBAC)** Do you have a fine-grained authorization scheme that allows you to grant access to each administrative and functional capability independently? For VoIP systems, that means that there are separately granted administrative permissions for each major area of configuration (such as networking, PSTN integration, user administration, etc.) and user-level permissions for different classes of features, calling restrictions, and so on.
- **Segregation of Duties (SoD)** Have you separated administrative, operational, and audit roles within your VoIP system so that, for instance, an auditor can gain access to system logs without having the ability to change settings? To properly implement SoD, you will need to support RBAC.
- **Identity and Access Management (IAM) with Provisioning** Have you tied the VoIP system's user and administrative identities back to enterprisewide directories and authentication schemes? In other words, do users and administrators accessing the VoIP system use the same IDs and passwords on the VoIP system as they would on other enterprise applications? Do directory attributes like groups enable automatic assignment of roles in the VoIP system's RBAC scheme? Does VoIP system deprovisioning (or disablement) happen automatically for a user that has been removed from the enterprisewide directory upon termination? Optional: Are new employees able to be provisioned automatically to the VoIP system as part of the on-boarding process?
- **Enterprise Role Definition (ERD)** Has your organization identified across its business applications the employee roles and access required by those roles to be able to map the VoIP system's roles into that enterprise scheme? Have those roles been screened for Segregation of Duties conflicts with the VoIP system included? Note that RBAC and IAM with Provisioning typically are required for an ERD system to work smoothly in practice.

## Tools & Traps...

### Core SOX Compliance Issues for IP Communications Systems

The only direct SOX impacts to VoIP and other communications systems are likely to be billing related if your VoIP system is part of a service billed to others or if your SOX controls team considers it to be part of an internal control around PSTN usage costs being billed back to your company. Of course, indirect impacts through IT security policies around user, password, logging, systems, and database management are all likely since the VoIP system is a part of the overall IT infrastructure of your organization.

The SOX issue most likely to be ignored by your SOX team: internal controls for controlling VoIP calls that route through the PSTN create financial obligations (i.e., long-distance charges) so long as your long distance isn't fixed-cost (or free), since abuse of IP communications systems could have a material financial impact on your organization. In SOX terms, that means that the same controls used with critical financial systems should be evaluated for applicability to IP communications systems as well.

## SOX Compliance and Enforcement

It may surprise you to know that most of the Act itself is focused on new practices and penalties for independent auditors, not public companies. The Sarbanes-Oxley Act created the Public Company Accounting Oversight Board (PCAOB) to address the audit processes used for public companies. The Act gives the PCAOB the authority to register, investigate, and discipline public accounting firms and auditors. Oversight of the PCAOB falls to the Securities and Exchange Commission (SEC). Penalties for certain white-collar crime were increased and the SEC has some additional civil enforcement tools as part of the Act, but in general all nonaudit compliance and enforcement for SOX remains within the enforcement frameworks previously established at the SEC.

### Certification

Compliance is evaluated on an annual basis by two groups: the management of the public company itself (typically through your internal audit or compliance group) for the management report asserting that internal controls are adequate (i.e., compliant with Sarbanes-Oxley requirements); and the company's independent auditor for their attestation—either unqualified support of management's report or a "disclaimer opinion" that raises concerns about the adequacy of internal controls. Just to complete the attestation process each year,

large companies can be charged up to \$1 million or more by their independent auditor—over and above the fees paid for basic corporate audit work. These costs (and potential conflicts the process can create with EU Data Protection directives) have prompted a number of European firms to de-list from American stock exchanges.

SOX has no notion of “product certification” like some of the other regulations in this chapter.

## Enforcement Process and Penalties

Auditors and auditing organizations are investigated and sanctioned by the Public Company Accounting Oversight Board (PCAOB), and corporate officers and corporations are investigated and sanctioned by the SEC. For the PCAOB, the maximum penalty for “violations committed in the preparation and issuance of audit reports,” was \$110,000 in 2005 for an individual and \$2.1 million for an entity. And the SEC maximum penalty in 2005 for “intentional or knowing conduct, including reckless conduct, or repeated instances of negligent conduct” was \$800,000 for an individual and \$15.825 million for an entity.

The Act itself increased the maximum penalty for mail, securities, and wire fraud to up to 25 years imprisonment, and established maximum penalties for CEOs and CFOs that made willful and knowing violations of financial statement and disclosure rules punishable by a fine of not more than \$500,000 and/or imprisonment of up to five years. The latter garnered a lot of press at the time and resulted in increased attention to SOX by corporate chiefs.

Both the SEC and PCAOB have processes in place to accept both anonymous tips and formal complaints. For the SEC, tips can be sent to [enforcement@sec.gov](mailto:enforcement@sec.gov) and online forms can be found at [www.sec.gov](http://www.sec.gov). The PCAOB can accept tips at [tips@pcaobus.org](mailto:tips@pcaobus.org) or online at [www.pcaobus.org](http://www.pcaobus.org).

## GLBA: Gramm-Leach-Bliley Act

The US Gramm-Leach-Bliley Act of 1999—commonly referred to as GLBA—is landmark legislation that completely reorganized the statutory and legislative framework in place since the 1930s for the banking and financial services market. Of particular note is Title V, Subtitle A, Section 501, which requires that banking, consumer finance, securities, and insurance companies develop and meet new standards for protection of consumer privacy and safeguarding of financial institution infrastructure. Although VoIP systems were not specifically called out in the Act itself, the Federal Deposit Insurance Corporation (FDIC) and other financial regulatory agencies subsequently have issued VoIP-specific guidance to be used by regulated entities.

# GLBA Regulatory Basics

Because the regulatory scope of GLBA is extensive and we really are interested only in the privacy and security effects of the legislation (and specifically, how they interact with VoIP systems), we will limit our discussion to Title V—PRIVACY. For those in the security community, every security reference to GLBA that you've seen is tied back to Title V, and we will review its contents later in this chapter. In addition, we will discuss supplementary guidance from consultants and regulators (including the FDIC VoIP recommendation) to help you understand what your organization will need for your VoIP system to operate in compliance with GLBA.

## Direct from the Regulations

Title V is broken out into two subtitles. Subtitle A, “Disclosure of Nonpublic Personal Information,” is where we will center most of our attention, particularly in Section 501. Subtitle B, “Fraudulent Access to Financial Information” criminalizes the act of using false pretenses to obtain financial information from an institution except under certain law-enforcement and investigative exclusions. We won't spend any more time with Subtitle B, but if you ever find yourself investigating someone's financial information you would be wise to familiarize yourself with its contents.

Of the 10 sections in subtitle A, I am only going to reproduce section 501 in its entirety, since it is the basis for all of the GLBA security recommendations I encounter. The other nine talk through privacy definitions, enforcement, and the creation of detailed regulations from the GLBA. In any case, Section 501 is what we want to be most familiar with, and it is fairly straightforward:

### **SEC. 501. PROTECTION OF NONPUBLIC PERSONAL INFORMATION.**

(a) **PRIVACY OBLIGATION POLICY-** It is the policy of the Congress that each financial institution has an affirmative and continuing obligation to respect the privacy of its customers and to protect the security and confidentiality of those customers' nonpublic personal information.

(b) **FINANCIAL INSTITUTIONS SAFEGUARDS-** In furtherance of the policy in subsection (a), each agency or authority described in section 505(a) shall establish appropriate standards for the financial institutions subject to their jurisdiction relating to administrative, technical, and physical safeguards—

(1) to insure the security and confidentiality of customer records and information;

(2) to protect against any anticipated threats or hazards to the security or integrity of such records; and

(3) to protect against unauthorized access to or use of such records or information which could result in substantial harm or inconvenience to any customer.

From this point onward, I'll use the commonly accepted terminology for the rules created by this section. 501(a) and subsequent joint regulations are collectively known as the *privacy rule* and 501(b) with its joint regulations is called the *safeguarding rule*. Later in this chapter, you'll notice that HIPAA regulations follow a similar model, except the latter is called "security" instead of "safeguarding." (That's the way I think about GLBA as well: privacy + security.)

After the GLBA was signed, the Secretary of the Treasury, the National Credit Union Administration (NCUA), the Federal Trade Commission (FTC), and the Securities and Exchange Commission (SEC) were required to create appropriate regulations as part of Title V. The resulting documents can be found at the FTC at [www.ftc.gov/os/2000/05/glb000512.pdf](http://www.ftc.gov/os/2000/05/glb000512.pdf) and the Office of the Comptroller of the Currency (OCC) at [www.occ.treas.gov/ftp/release/0509fin.pdf](http://www.occ.treas.gov/ftp/release/0509fin.pdf). Detailed requirements for the privacy disclosures and opt-out procedures are spelled out in detail within these two documents (and if you're like me, you receive the annual privacy disclosures they require in droves from financial institutions). In general, there are no VoIP considerations within the privacy rule that aren't more directly addressed by the safeguarding rule, so we're going to spend the rest of this section on the safeguarding rule.

Detailed regulations for the safeguarding were finalized in 2001 as the "Interagency Guidelines Establishing Information Security Standards" (see [www.fdic.gov/regulations/laws/rules/2000-8660.html](http://www.fdic.gov/regulations/laws/rules/2000-8660.html) or [www.ots.treas.gov/docs/2/25231.pdf](http://www.ots.treas.gov/docs/2/25231.pdf) for a typical copy), and it is these rules that you will want to become most familiar with, in particular, part III:

### **III. Development and Implementation of Information Security Program**

**A. Involve the Board of Directors.** The board of directors or an appropriate committee of the board of each bank holding company shall:

1. Approve the bank holding company's written information security program; and
2. Oversee the development, implementation, and maintenance of the bank holding company's information security program, including assigning specific responsibility for its implementation and reviewing reports from management.

**B. Assess Risk.** Each bank holding company shall:

1. Identify reasonably foreseeable internal and external threats that could result in unauthorized disclosure, misuse, alteration, or destruction of customer information or customer information systems.
2. Assess the likelihood and potential damage of these threats, taking into consideration the sensitivity of customer information.
3. Assess the sufficiency of policies, procedures, customer information systems, and other arrangements in place to control risks.

**C. Manage and Control Risk.** Each bank holding company shall:

1. Design its information security program to control the identified risks, commensurate with the sensitivity of the information as well as the complexity and scope of the bank holding company's activities. Each bank holding company must consider whether the following security measures are appropriate for the bank holding company and, if so, adopt those measures the bank holding company concludes are appropriate:
  - a. Access controls on customer information systems, including controls to authenticate and permit access only to authorized individuals and controls to prevent employees from providing customer information to unauthorized individuals who may seek to obtain this information through fraudulent means.
  - b. Access restrictions at physical locations containing customer information, such as buildings, computer facilities, and records storage facilities to permit access only to authorized individuals;
  - c. Encryption of electronic customer information, including while in transit or in storage on networks or systems to which unauthorized individuals may have access;
  - d. Procedures designed to ensure that customer information system modifications are consistent with the bank holding company's information security program;
  - e. Dual control procedures, segregation of duties, and employee background checks for employees with responsibilities for or access to customer information;
  - f. Monitoring systems and procedures to detect actual and attempted attacks on or intrusions into customer information systems;
  - g. Response programs that specify actions to be taken when the bank holding company suspects or detects that unauthorized individuals have gained access to customer information systems, including appropriate reports to regulatory and law enforcement agencies; and
  - h. Measures to protect against destruction, loss, or damage of customer information due to potential environmental hazards, such as fire

and water damage or technological failures.

2. Train staff to implement the bank holding company's information security program.

3. Regularly test the key controls, systems and procedures of the information security program. The frequency and nature of such tests should be determined by the bank holding company's risk assessment. Tests should be conducted or reviewed by independent third parties or staff independent of those that develop or maintain the security programs.

**D. Oversee Service Provider Arrangements.** Each bank holding company shall:

1. Exercise appropriate due diligence in selecting its service providers;
2. Require its service providers by contract to implement appropriate measures designed to meet the objectives of these Guidelines; and
3. Where indicated by the bank holding company's risk assessment, monitor its service providers to confirm that they have satisfied their obligations as required by paragraph D.2. As part of this monitoring, a bank holding company should review audits, summaries of test results, or other equivalent evaluations of its service providers.

**E. Adjust the Program.** Each bank holding company shall monitor, evaluate, and adjust, as appropriate, the information security program in light of any relevant changes in technology, the sensitivity of its customer information, internal or external threats to information, and the bank holding company's own changing business arrangements, such as mergers and acquisitions, alliances and joint ventures, outsourcing arrangements, and changes to customer information systems.

**F. Report to the Board.** Each bank holding company shall report to its board or an appropriate committee of the board at least annually. This report should describe the overall status of the information security program and the bank holding company's compliance with these Guidelines. The reports should discuss material matters related to its program, addressing issues such as: risk assessment; risk management and control decisions; service provider arrangements; results of testing; security breaches or violations and management's responses; and recommendations for changes in the information security program.

**G. Implement the Standards.**

1. Effective date. Each bank holding company must implement an information security program pursuant to these Guidelines by July 1, 2001.
2. Two-year grandfathering of agreements with service providers.

Until July 1, 2003, a contract that a bank holding company has entered into with a service provider to perform services for it or functions on its behalf satisfies the provisions of section III.D., even if the contract does not include a requirement that the servicer maintain the security and confidentiality of customer information, as long as the bank holding company entered into the contract on or before March 5, 2001.

These are the standards against which financial regulators will evaluate your organization if it falls under the GLBA. For VoIP systems, the primary concern will be to ensure that risk management and security processes for compliance include the VoIP infrastructure and that your organization's security standards developed for GLBA compliance will be applied to your IP communications systems as well.

## What a Financial Regulator or GLBA Consultant Will Tell You

Until July 2005, when the FDIC provided very specific and detailed VoIP guidance, it was not uncommon for GLBA experts to consider voice communications systems to be outside the scope of GLBA's safeguarding rule. In what's known as a Financial Institution Letter or FIL (in this case FIL-69-2005—see [www.fdic.gov/news/news/financial/2005/fil6905.html](http://www.fdic.gov/news/news/financial/2005/fil6905.html) for a complete copy); the FDIC made it clear that VoIP systems must be included in GLBA risk assessment reports and processes. In their highlights, the FDIC noted:

- VoIP is susceptible to the same security risks as data networks if security policies and configurations are inadequate.
- The risks associated with VoIP should be evaluated as part of a financial institution's periodic risk assessment, with status reports submitted to the board of directors as mandated by section 501(b) of the Gramm-Leach—Bliley Act (GLBA). Any identified weaknesses should be corrected during the normal course of business.

This effectively told regulators and institutions that they will be expected to include IP communications systems in their GLBA compliance planning and reporting going forward. The FDIC VoIP security recommendation follows:

Financial institutions can access various publicly available sources to develop VoIP security policies and practices. Widely accepted best practices are published by the National Institute of Standards and Technology (NIST), the agency responsible for developing information security standards for federal agencies (Special NIST Publication 800-58, Security Considerations for Voice over IP Systems, can be found at <http://csrc.nist.gov/publications/nistpubs/800-58/SP800-58-final.pdf>.)



Financial institutions contemplating the use of VoIP technology should consider the following best practices. Details of these best practices are further discussed in the attached "Voice over Internet Protocol Informational Supplement."

- Ensure that the institution has examined and can acceptably manage and mitigate the risks to information, systems operations and continuity of essential operations when implementing VoIP systems.
- Assess the level of concern about security and privacy. If warranted and practical, do not use "softphone" systems, which implement VoIP using an ordinary PC with a headset and special software.
- Carefully review statutory requirements for privacy and record retention with competent legal advisors.
- Develop appropriate network architecture.
- Use VoIP-ready firewalls and other appropriate protection mechanisms. Financial institutions should enable, use and routinely test security features included in VoIP systems.
- Properly implement physical controls in a VoIP environment.
- Evaluate costs for additional backup systems that may be required to ensure continued operation during power outages.
- Consider the need to integrate mobile telephone units with the VoIP system. If the need exists, consider using products implementing WiFi Protected Access (WPA), rather than Wired Equivalent Privacy (WEP).
- Give special consideration to emergency service communications. Automatic location services are not always as available with VoIP as they are with phone calls made through the PSTN.

When a financial institution decides to invest in VoIP technology, the associated risks should be evaluated as part of a financial institution's periodic risk assessment and discussed in status reports submitted to the board of directors as mandated by section 501(b) of the Gramm-Leach-

Bliley Act. Any identified weaknesses should be corrected during the normal course of business.

The aforementioned FDIC VoIP Informational Supplement can be downloaded at [www.fdic.gov/news/news/financial/2005/fil6905a.html](http://www.fdic.gov/news/news/financial/2005/fil6905a.html) if you'd like to get more detail on the points covered in the previous section of this chapter. Since it rehashes points covered in detail elsewhere in this book, I will leave this as an exercise for you, dear reader.

## Tools & Traps...

### Core GLBA-Compliance Issues for IP Communications Systems

Although GLBA does not have specific rules for VoIP, its integration with the rest of your organization's data network clearly puts it in scope of GLBA safeguarding provisions. This was reinforced by FDIC FIL-69-2005, which suggests nine specific GLBA risk management activities for VoIP systems:

- Include VoIP systems into general risk management and continuity planning
- Avoid softphone systems (where warranted and practical)
- Review privacy and records retention approach within VoIP system
- Review VoIP network architecture as part of overall network architecture
- Enable and test VoIP security features; use VoIP-ready firewalls
- Implement appropriate physical controls on VoIP systems
- Consider costs of additional backup systems required during power outages
- Avoid WEP on wireless VoIP; use WPA instead
- Consider E911 location service implications

In addition to the items highlighted by the FDIC, the same user, password, log, and database management policies used for data applications should also be applied to IP communications systems.

## GLBA Compliance and Enforcement

Enforcement of Title V of the GLBA falls to 57 different regulators in three classes: federal functional regulators, state insurance authorities, and the Federal Trade Commission as follows:

- **State insurance authorities in each state** Insurance providers
- **Securities and Exchange Commission (SEC)** Brokers, dealers, investment advisors and investment companies
- **Office of the Comptroller of the Currency (OCC)** National banks
- **National Credit Union Administration (NCUA)** Federally insured credit unions
- **Board of Governors of the Federal Reserve System (FRB)** Member banks of the Federal Reserve System (other than national banks), branches and agencies of foreign banks (except federal branches, federal agencies, and insured state branches of foreign banks), commercial lending companies owned or controlled by foreign banks, organizations operating under section 25 or 25A of the Federal Reserve Act, and bank holding companies and their nonbank subsidiaries or affiliates not subject to the SEC or state authorities
- **Board of Directors of the Federal Deposit Insurance Corporation (FDIC)** Banks insured by the FDIC (except Federal Reserve System members), insured state branches of foreign banks, and their nonbank subsidiaries or affiliates not subject to the SEC or state authorities
- **Director of the Office of Thrift Supervision (OTS)** Savings associations insured by the FDIC and their nonbank subsidiaries or affiliates not subject to the SEC or state authorities
- **Federal Trade Commission (FTC)** All others

## No Certification

GLBA has no concept of certification, either for institutions, individuals, or products.

## Enforcement Process and Penalties

The FDIC, NCUA, OTS, OCC, and FRB use uniform principles, standards, and report forms created by the Federal Financial Institutions Examination Council (FFIEC). The FFIEC has gathered together a broad set of IT-related presentations, examination booklets, and other resources ([www.ffiec.gov/ffiecinfobase/index.html](http://www.ffiec.gov/ffiecinfobase/index.html)) that provide an excellent guide to what their examiners will be looking for in an information security examination. For the banks and other financial institutions that fall under these agencies, GLBA enforcement is part of the overall enforcement regime that is standardized by the FFIEC.

Each of the 57 possible regulators has discretion over sanctions and penalties for privacy or safeguarding rule violations (for Subtitle B there are criminal penalties but these don't apply to the privacy or safeguarding rules, only to criminal access to financial data under

fraudulent pretenses), so penalties may vary. Also, civil suits can be brought against financial institutions that violate the GLBA privacy rule.

## HIPAA: Health Insurance Portability and Accountability Act

Within the U.S. Health Insurance Portability and Accountability Act of 1996, Congress adopted a broad range of reforms and standards designed to improve healthcare and health insurance and move toward electronic transaction processing and recordkeeping. As part of the 1996 Act, Congress acknowledged the need for privacy standards, but it failed to produce them in time to meet its own deadline; that job fell to the Department of Health and Human Services (HHS), which issued the final rule for privacy in December 2000. The final security rule was issued by HHS in February 2003.

### HIPAA Regulatory Basics

The privacy and security mandates that can affect VoIP systems are found in Title II, Subtitle F, Part C – Administrative Simplification. There are three aspects to Title II: Privacy, Code Sets, and Security. HHS has issued detailed regulations for all three, but the only two that can apply to VoIP systems are Privacy and Security.

Critical to understanding HIPAA is the concept of Protected Health Information (PHI) or Individually Identifiable Health Information (IIHI). Think of IIHI or PHI as any set of information that contains health-related data for an individual that can be traced back to that person. In order to share health-related information with other individuals or groups that participate in a patient's care, a Covered Entity (organization subject to HIPAA) must first receive the patient's consent to share that PHI with those participants (insurance, billing, physicians, hospitals, pharmacies, and so forth). Protection of PHI by a Covered Entity is the objective of the HIPAA Privacy Rule and Security Rule.

### Direct from the Regulations

Privacy in HIPAA is addressed in Section 264 (of Title II, Subtitle F, Part C). The HHS Privacy Rule is based on this text in the Act:

**The recommendations under subsection (a) shall address at least the following: (1) The rights that an individual who is a subject of individually identifiable health information should have. (2) The procedures that should be established for the exercise of such rights. (3) The uses and disclosures of such information that should be authorized or required.**

Three years and over 52,000 comments later, the first HHS Final Rule for Privacy was published, and after four more amendments (the last of which was in April 2003) the

“Standards for Privacy of Individually Identifiable Health Information” had reached its present form (for a copy of the combined Privacy and Security regulations along with enforcement and penalty information, go to [www.hhs.gov/ocr/combinedregtext.pdf](http://www.hhs.gov/ocr/combinedregtext.pdf)). In general, the Policy Rule applies more to an organization’s procedures independent of technology, so it makes more sense to dig into HHS Security Rule, “Security Standards for the Protection of Electronic Protected Health Information,” which is based on this text in Section 1173 of the Act:

**(1) SECURITY STANDARDS.—**The Secretary shall adopt security standards that—

**(A) take into account—**(i) the technical capabilities of record systems used to maintain health information; (ii) the costs of security measures; (iii) the need for training persons who have access to health information; (iv) the value of audit trails in computerized record systems; and (v) the needs and capabilities of small health care providers and rural health care providers (as such providers are defined by the Secretary); and

**(B) ensure that a health care clearinghouse, if it is part of a larger organization, has policies and security procedures which isolate the activities of the health care clearinghouse with respect to processing information in a manner that prevents unauthorized access to such information by such larger organization.**

**(2) SAFEGUARDS.—**Each person described in section 1172(a) who maintains or transmits health information shall maintain reasonable and appropriate administrative, technical, and physical safeguards—

**(A) to ensure the integrity and confidentiality of the information;**

**(B) to protect against any reasonably anticipated—**(i) threats or hazards to the security or integrity of the information; and (ii) unauthorized uses or disclosures of the information; and

**(C) otherwise to ensure compliance with this part by the officers and employees of such person.**

Notice the way that security is broken out in the Act—this structure is carried forward into the HHS Security Rule (and believe me, without that knowledge it’s hard to make sense of the Rule).

## *The Security Rule*

Within the Security Rule, there are general requirements that outline what a covered entity is required to document for compliance overall. Specific requirements then follow in four main categories: Administrative, Physical, and Technical Safeguards, plus Organizational Requirements. Understanding the difference between the first three is crucial to following the Security Rule:

**Administrative safeguards** are administrative actions, and policies and procedures, to manage the selection, development, implementation, and maintenance of security measures to protect electronic protected health information and to manage the conduct of the covered entity's workforce in relation to the protection of that information.

**Physical safeguards** are physical measures, policies, and procedures to protect a covered entity's electronic information systems and related buildings and equipment, from natural and environmental hazards, and unauthorized intrusion

**Technical safeguards** means the technology and the policy and procedures for its use that protect electronic protected health information and control access to it.

With this in mind, let's start with the general requirements and objectives for the security rule, and the flexibility allowed in implementing and documenting standards in each of the four categories:

(a) General requirements. Covered entities must do the following:

(1) Ensure the confidentiality, integrity, and availability of all electronic protected health information the covered entity creates, receives, maintains, or transmits.

(2) Protect against any reasonably anticipated threats or hazards to the security or integrity of such information.

(3) Protect against any reasonably anticipated uses or disclosures of such information that are not permitted or required under subpart E of this part.

(4) Ensure compliance with this subpart by its workforce.

(b) Flexibility of approach.

(1) Covered entities may use any security measures that allow the covered entity to reasonably and appropriately implement the standards and implementation specifications as specified in this subpart.

(2) In deciding which security measures to use, a covered entity must take into account the following factors:

(i) The size, complexity, and capabilities of the covered entity.

(ii) The covered entity's technical infrastructure, hardware, and software security capabilities.

(iii) The costs of security measures.

(iv) The probability and criticality of potential risks to electronic protected health information.

This flexibility is key to making your compliance document less painful to write. When you find that a vendor's equipment or solution does not provide a technical solution to a given standard, you can usually assemble an administrative solution that provides an acceptable workaround. And for those items that are not required (marked as Addressable in the Security Rule), you can still be compliant if you document why implementation of that item isn't reasonable or appropriate. Specifically,

(d) Implementation specifications. In this subpart:

(1) Implementation specifications are required or addressable. If an implementation specification is required, the word "Required" appears in parentheses after the title of the implementation specification. If an implementation specification is addressable, the word "Addressable" appears in parentheses after the title of the implementation specification.

(2) When a standard adopted in § 164.308, § 164.310, § 164.312, § 164.314, or § 164.316 includes required implementation specifications, a covered entity must implement the implementation specifications.

(3) When a standard adopted in § 164.308, § 164.310, § 164.312, § 164.314, or § 164.316 includes addressable implementation specifications, a covered entity must—

(i) Assess whether each implementation specification is a reasonable and appropriate safeguard in its environment, when analyzed with reference

to the likely contribution to protecting the entity's electronic protected health information; and

(ii) As applicable to the entity—

(A) Implement the implementation specification if reasonable and appropriate; or

(B) If implementing the implementation specification is not reasonable and appropriate—

(1) Document why it would not be reasonable and appropriate to implement the implementation specification; and

(2) Implement an equivalent alternative measure if reasonable and appropriate.

With this in mind, I want to skip ahead to the documentation standard so that you understand why documentation is so critical for HIPAA compliance:

(b)(1) Standard: Documentation.

(i) Maintain the policies and procedures implemented to comply with this subpart in written (which may be electronic) form; and

(ii) If an action, activity or assessment is required by this subpart to be documented, maintain a written (which may be electronic) record of the action, activity, or assessment.

(2) Implementation specifications:

(i) Time limit (Required). Retain the documentation required by paragraph (b)(1) of this section for 6 years from the date of its creation or the date when it last was in effect, whichever is later.

(ii) Availability (Required). Make documentation available to those persons responsible for implementing the procedures to which the documentation pertains.

(iii) Updates (Required). Review documentation periodically, and update as needed, in response to environmental or operational changes affecting the security of the electronic protected health information.



You may never need to produce that documentation, but if your organization is subject to an investigation or a compliance review and you don't have it ready, you and your organization could face significant penalties.



## WARNING

---

It's tempting to think of HIPAA documentation as something you can ask the VoIP (or other product) vendor to take care of for you, but there are two reasons why I don't recommend it. First, the vendor is not on the hook for your HIPAA processes; suppose they agreed to document a process for you, but it's one that you can't reasonably implement—it's your organization that will be held responsible by regulators, not the vendor. Second, remember that HIPAA is about your organization's operational processes, not any specific software or hardware. Unless you're hiring a consultant specifically for that purpose, asking an equipment vendor to document that process for you makes about as much sense as asking your local car dealer to pass a driving test for you. Maybe you get a salesman who takes you up on it just to close the sale, but that doesn't really make it appropriate or legal (and it won't make you a safe driver).

---

So what needs to be documented? Each of the items within the four main categories of the security rule: Administrative, Physical, and Technical Safeguards, plus Organizational Requirements. Since these are lengthy sections, I'm going to summarize and highlight specific parts from each that are likely to come into play with VoIP systems. You'll want to consult the Security Rule for specific details if you believe a listed standard will apply to the VoIP system.

### *Administrative Safeguards with VoIP Applicability*

- Documented security management process to prevent, detect, contain, and correct security violations. Required elements: risk analysis, risk management, sanction policy, and logging/activity review.
- Authorization policies and procedures must be established to grant access to PHI only to those who require it. Addressable elements: Authorization and/or supervision, workforce clearance procedure, termination procedure.
- Security awareness and training program. Addressable elements: security reminders, malicious software protection, log-in monitoring, password management.
- Security incident procedures. Required elements: response and reporting.

- Contingency plan. Required elements: data backup plan, disaster recovery plan, emergency mode operation plan. Addressable elements: testing and revision procedures, applications and data criticality analysis.

### *Physical Safeguards with VoIP Applicability*

- Physical access controls implementation. Addressable elements: contingency operations, facility security plan, physical access control and validation procedures, maintenance records.
- Device and media controls. Required elements: disposal, media reuse. Addressable elements: accountability, data backup and storage.

### *Technical Safeguards with VoIP Applicability*

- Access control. Required elements: unique user identification, emergency access procedure. Addressable elements: automatic logoff, encryption and decryption.
- Audit controls (record of activity within systems containing PHI).
- Integrity. Addressable element: authentication mechanism (for PHI).
- Authentication (individual and entity seeking access to PHI).
- Transmission security. Addressable elements: integrity controls, encryption.

### *Organizational Requirements*

These will generally not have any VoIP applicability except in the unusual case where there is a business relationship established with a service provider with access to recorded information containing PHI.

### *Other Considerations*

Don't assume that because VoIP runs over IP it is considered to be "transmission via electronic media" by HIPAA. Within HHS General Administrative Requirements there is an official definition stating that:

**Certain transmissions, including of paper, via facsimile, and of voice, via telephone, are not considered to be transmissions via electronic media, because the information being exchanged did not exist in electronic form before the transmission**

In general this excludes VoIP from HIPAA so long as the transmission is not recorded. Recording is the critical distinction. Within that same section HHS notes:

Health information means any information, whether oral or recorded in any form or medium, that: (1) Is created or received by a health care provider, health plan, public health authority, employer, life insurer, school or university, or health care clearinghouse; and (2) Relates to the past, present, or future physical or mental health or condition of an individual; the provision of health care to an individual; or the past, present, or future payment for the provision of health care to an individual.

From this, we see that a *recorded* VoIP call or voicemail clearly will fall within the scope of the HIPAA Privacy and Security Rules even though a nonrecorded call would not.

## What a HIPAA Consultant Will Tell You

My experience with HIPAA consultants is that few of them have thought much about what happens when you record a VoIP conversation and what documentation is required for the system overall when you do. Nearly all agree that VoIP by itself does not create any HIPAA requirements. The question is how much documentation is required for voicemail and other call recording technologies.

Given the flexibility that the Security Rule allows, my suggestion is to document just that part of the system involved in recording, but even with that limited scope there will be plenty to document. If the VoIP system includes or interfaces with an Interactive Voice Response (IVR) system, that may need to be documented as well if it can be used as a gateway to PHI contained on a database system behind it.

### Tools & Traps...

#### Core HIPAA-Compliance Issues for IP Communications Systems

Although HIPAA regulations only briefly touch on voice communication systems at all, several general principles still apply. First, the use of VoIP by itself does not create any electronic records unless some related system is recording a session containing Protected Health Information (PHI). In that case, the system doing the recording will fall under HIPAA requirements. This means that voice messaging and call recording equipment may require fully documented HIPAA-compliant operational processes. Second, if a VoIP-related system (such as a VoiceXML server) is a gatekeeper to databases or other record-keeping systems that contain PHI, then HIPAA also will apply. Another example of this is an IVR system that front-ends patient records or billing systems.

# HIPAA Compliance and Enforcement

The Department of Health and Human Services (HHS) delegated compliance and enforcement of the HIPAA Privacy Rule to the Office for Civil Rights (OCR) along with authority for allowing exceptions where certain state laws may conflict with HIPAA. The Centers for Medicare and Medicaid services (CMS) received delegated responsibility from HHS for enforcing the security rule, transactions, and code set standards (and identifiers standards when those are published). Through its Office of HIPAA Standards (OHS), CMS will enforce these rules and continue to enforce the insurance portability requirements under Title I of HIPAA.

## No Certification

No official certification process exists for covered entities under HIPAA, although HHS did receive authority to perform compliance reviews as part of the Act. Products are not certified as part of HIPAA (although it's not uncommon to see them promoted as if they were). Regardless, documentation as specified in the Security Rule and Privacy Rule must exist and might be reviewed by a business partner, for example, as part of a due-diligence process. Other than that, the only time you would have to produce it is if you are investigated by HHS or OCR in response to a complaint or as part of a compliance review.

## Enforcement Process and Penalties

In general, OCR acts on Privacy Rule violations in response to complaints that are registered with it. OCR requires written notification but does accept e-mail at [OCRComplaint@hhs.gov](mailto:OCRComplaint@hhs.gov) (see “How to File a Health Information Privacy Complaint” at [www.hhs.gov/ocr/privacyhowtofile.htm](http://www.hhs.gov/ocr/privacyhowtofile.htm) for more details). CMS has stated that the enforcement process for its portion of HIPAA will be primarily complaint-driven, although their primary strategy is to achieve “voluntary compliance through technical assistance.” Penalties would be imposed as a last resort. When a complaint is received (typically through their Web site at [www.cms.hhs.gov/Enforcement](http://www.cms.hhs.gov/Enforcement) or via mail), CMS first allows the provider the opportunity to demonstrate compliance (or submit a plan for corrective action). Only if the provider fails to respond would penalties be considered.

The Administrative Simplification Compliance Act (ASCA) permits the Secretary of HHS to exclude noncompliant covered entities from the Medicare program. In addition, the original HIPAA legislation permits civil monetary penalties of not more than \$100 for each violation, with a cap of \$25,000 per calendar year. In addition, criminal penalties can be imposed for certain wrongful disclosures up to a \$250,000 fine and 10 years imprisonment for willful conduct.

# CALEA: Communications Assistance for Law Enforcement Act

The Communications Assistance for Law Enforcement Act first arrived from the U.S. Congress in 1994 with a simple goal: improving wiretapping effectiveness for law-enforcement in an increasingly digital PSTN. Advances in telecommunications made prior wiretapping methods less effective and CALEA was intended to force all carriers and carrier-grade equipment vendors to provide consistent and accessible electronic monitoring capabilities. For private equipment, including PBX and similar business-class voice equipment, CALEA doesn't apply except when that equipment was deemed a "substantial replacement" for the public telephone service.

Between 1994 and 2004, CALEA eventually progressed to a rough set of technically feasible standards backed by FCC regulations (and deep involvement by the Federal Bureau of Investigation (FBI) and Department of Justice (DOJ)), though packet communications was still a CALEA minefield. These VoIP and broadband issues came to a head in August 2004 when the FCC issued a Notice of Proposed Rulemaking and Declaratory Ruling (NPRM) for public comment, stirring up anew the privacy and civil-liberties debate (see the sidebar, "CALEA and the Xbox?"). Lost to many observers was the fact the new NPRM might now be broad enough to force enterprises, universities, and other previously excluded organizations that deploy VoIP to become subject to the revised regulations. Although several requests for clarification on that topic still are pending at the FCC, it's clear these rules could substantially affect the design and deployment of enterprise VoIP.

If you're a carrier (of Voice, VoIP, or even just broadband IP), CALEA regulation is already a certainty (although in the case of broadband, there is a lot of work remaining even to agree on the technical standards, and the FBI has yet to specify capacity requirements as required by the Act). And in spite of the fact that in November 1994, the FCC had ruled that VoIP was a "data service" for other regulatory purposes, the FCC and DOJ agreed that data services were still within the scope of CALEA. Although predictable, this nevertheless came as a shock to many carriers who had in recent years become comfortable with the FCC hands-off approach to data networks and VoIP despite pressure from the FBI and Department of Justice (DOJ).

## Notes from the Underground...

### CALEA and the Xbox?

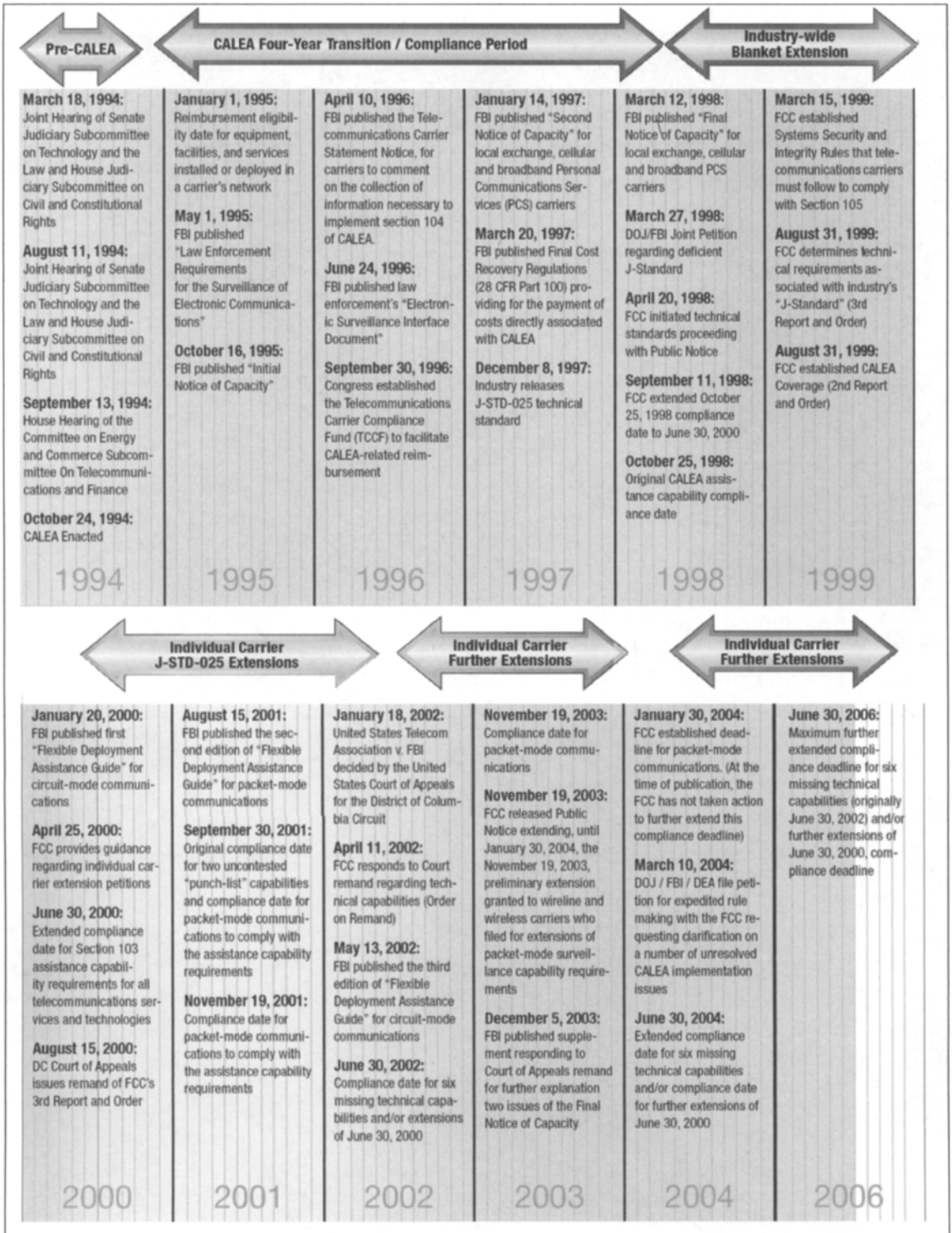
With the latest CALEA guidance for broadband, it's applicability to VoIP and data networks that has become a new privacy battleground. Groups like the Electronic Frontier Foundation have been heavily involved in the debate, and from their perspective, the consequences of the revised CALEA rules could have long-ranging—and possibly dire—consequences:

“If the FBI gets its way, the NPRM’s tentative regulations will only be the tip of the iceberg. Soon, software companies, under threat of an expansive definition of CALEA’s requirements, will face economic incentives to create email and IM programs that are surveillance-ready. Many computer game consoles that people can use to play over the Internet, such as the Xbox, allow gamers to chat with each other while they play. If any communication program running on the Internet has to be CALEA-compliant before being bought and sold, what would stop law enforcement from pushing for a tappable Xbox?”

Although it remains to be seen just how far the FCC and DOJ take enforcement of CALEA, it seems unlikely that serious enforcement will happen outside of the carrier space (with the possible exception of organizations like universities that provide phone service over a large campus).

Figure C.1 shows a timeline for the development of the CALEA.

Figure C.1 CALEA Timeline\*



\* Published in the Communications Assistance for Law Enforcement Act (CALEA) Flexible Deployment Assistance Guide, Fourth Edition

Since the publication of this guide, the following developments have taken place:

- September 23, 2004: FCC rules that all “push-to-talk” services are subject to CALEA
- September 23, 2005: FCC responds to DOJ / FBI / DEA petition and issues Notice of Proposed Rulemaking (NPRM) that will require broadband and VoIP providers to comply with CALEA; compliance deadline will be 18 months after final order.

## CALEA Regulatory Basics

Several critical documents are required reading for those wanting to understand the intent of the original Act, and subsequent VoIP policy from the FCC, FBI, DOJ, and other agencies, particularly in the context of VoIP and its place in the latest CALEA rules. Here’s the short list; we’ll cover each of these in more detail later in the section:

- The 1994 Act itself as passed by Congress (see [www.askcalea.net/calea.html](http://www.askcalea.net/calea.html) for a full copy) broadened wiretap applicability to new telecommunications technologies and added a new requirement to gather “call-identifying information” as part of a legal communications intercept.
- J-STD-025, “Lawfully Authorized Electronic Surveillance” published by the Telecommunications Industry Association (TIA) as a result of work started in 1995 to address CALEA; known initially as TIA/EIA SP 3580. J-STD-025 was first published by TIA in December, 1997. (The current version required for FCC compliance is J-STD-025-A, published by the TIA in December, 2000—available for purchase at [www.tiaonline.org/standards/catalog/](http://www.tiaonline.org/standards/catalog/) for nonmembers.)
- FCC “CALEA Third Report and Order,” August 31, 1999 (for a full copy, see [www.fcc.gov/Bureaus/Engineering\\_Technology/Orders/1999/fcc99230.pdf](http://www.fcc.gov/Bureaus/Engineering_Technology/Orders/1999/fcc99230.pdf) or .txt), defined capability requirements in terms of J-STD-025 for wireline, cellular, and broadband PCS carriers, and specified that six of the nine additional capabilities in the FBI “CALEA punch list” for J-STD-025 would be required for CALEA compliance (subsequently incorporated into J-STD-025-A).
- DOJ, FBI and Drug Enforcement Agency (DEA), “Joint Petition for Expedited Rulemaking” ([www.askcalea.net/docs/20040310.calea.jper.pdf](http://www.askcalea.net/docs/20040310.calea.jper.pdf)) filed before the FCC March 10, 2004 requested clear rules for how CALEA will be implemented on a wide variety of services, including packet technologies generally and VoIP specifically. Although not itself a regulation, this document serves as a roadmap for



FCC rulemaking that will take place in 2006 and beyond, directly affecting VoIP service providers.

- FCC “First Report and Order and Further Notice of Proposed Rulemaking,” FCC 05-153 (get a copy at [www.askcalea.net/docs/20050923-fcc-05-153.pdf](http://www.askcalea.net/docs/20050923-fcc-05-153.pdf) or at [www.fcc.gov](http://www.fcc.gov)), September 23, 2005, issued in response to the March 2004 Joint Petition.

## Direct from the Regulations

The basic technical requirements of the Act can be found in the first part of Section 103. In a nutshell, when a court order is present, the law enforcement requires access to all communications and their surrounding context without letting the target discover the “wiretap” (known in CALEA as a “lawful intercept”):

### **SEC. 103. ASSISTANCE CAPABILITY REQUIREMENTS.**

(a) **CAPABILITY REQUIREMENTS-** Except as provided in subsections (b), (c), and (d) of this section and sections 108(a) and 109(b) and (d), a telecommunications carrier shall ensure that its equipment, facilities, or services that provide a customer or subscriber with the ability to originate, terminate, or direct communications are capable of—

(1) expeditiously isolating and enabling the government, pursuant to a court order or other lawful authorization, to intercept, to the exclusion of any other communications, all wire and electronic communications carried by the carrier within a service area to or from equipment, facilities, or services of a subscriber of such carrier concurrently with their transmission to or from the subscriber’s equipment, facility, or service, or at such later time as may be acceptable to the government;

(2) expeditiously isolating and enabling the government, pursuant to a court order or other lawful authorization, to access call-identifying information that is reasonably available to the carrier—

(A) before, during, or immediately after the transmission of a wire or electronic communication (or at such later time as may be acceptable to the government); and

(B) in a manner that allows it to be associated with the communication to which it pertains, except that, with regard to information acquired solely pursuant to the authority for pen registers and trap and trace devices (as defined in section 3127 of title 18, United States Code), such

call-identifying information shall not include any information that may disclose the physical location of the subscriber (except to the extent that the location may be determined from the telephone number);

(3) delivering intercepted communications and call-identifying information to the government, pursuant to a court order or other lawful authorization, in a format such that they may be transmitted by means of equipment, facilities, or services procured by the government to a location other than the premises of the carrier; and

(4) facilitating authorized communications interceptions and access to call-identifying information unobtrusively and with a minimum of interference with any subscriber's telecommunications service and in a manner that protects—

(A) the privacy and security of communications and call-identifying information not authorized to be intercepted; and

(B) information regarding the government's interception of communications and access to call-identifying information.

Bottom line: CALEA even at this level not only requires the media itself for a VoIP call, but a good deal of signaling information as well (labeled “call-identifying information” in the Act). In addition, you must facilitate the process and provide appropriate equipment to enable the surveillance to take place, although some cost recovery is permitted (this is an open issue, however, as you'll see in the 2004 Joint Petition). If you're a carrier (or substantial replacement for one) and fall under CALEA, every communications service that you provide to your customers must be capable of meeting these requirements.

## NOTE

Although these terms are by no means unique to CALEA, it's useful to review the different types of legal interception available to Law Enforcement Agencies (LEAs) today:

1. Pen Register—what numbers were called by the target?
2. Trap and Trace—what numbers called the target?
3. Interception (Title III)—recorded conversation of the target (plus the other two items in this list). Most of the time, CALEA talks about this type of legal intercept.

The rest of the act lays out specific regulatory mandates and responsibilities, mainly targeted at the FCC. Sections 102, 104, 107, and 109 mandate that the FCC establish regulations for systems security and integrity, associated technical requirements, and determinations for specific equipment, facility, or services. An important compliance concept is also part of Section 107 and are known as “Safe harbor standards.” Section 107(a)(2) of CALEA allows a carrier to be deemed in compliance with CALEA’s capability requirements in Sections 103 and 106 if it complies with an appropriate publicly available technical standard. Also in Section 107 is a provision that allows a carrier to petition for an extension of the CALEA deadline when appropriate standards or technology isn’t available. Here’s the complete text of Sections 106 and 107:

**SEC. 106. COOPERATION OF EQUIPMENT MANUFACTURERS AND PROVIDERS OF TELECOMMUNICATIONS SUPPORT SERVICES.**

(a) CONSULTATION- A telecommunications carrier shall consult, as necessary, in a timely fashion with manufacturers of its telecommunications transmission and switching equipment and its providers of telecommunications support services for the purpose of ensuring that current and planned equipment, facilities, and services comply with the capability requirements of section 103 and the capacity requirements identified by the Attorney General under section 104.

(b) COOPERATION- Subject to sections 104(e), 108(a), and 109 (b) and (d), a manufacturer of telecommunications transmission or switching equipment and a provider of telecommunications support services shall, on a reasonably timely basis and at a reasonable charge, make available to the telecommunications carriers using its equipment, facilities, or services such features or modifications as are necessary to permit such carriers to comply with the capability requirements of section 103 and the capacity requirements identified by the Attorney General under section 104.

**SEC. 107. TECHNICAL REQUIREMENTS AND STANDARDS; EXTENSION OF COMPLIANCE DATE.**

(a) SAFE HARBOR-

(1) CONSULTATION- To ensure the efficient and industry-wide implementation of the assistance capability requirements under section 103, the Attorney General, in coordination with other Federal, State, and local law enforcement agencies, shall consult with appropriate associations and standard-setting organizations of the telecommunications

industry, with representatives of users of telecommunications equipment, facilities, and services, and with State utility commissions.

(2) **COMPLIANCE UNDER ACCEPTED STANDARDS-** A telecommunications carrier shall be found to be in compliance with the assistance capability requirements under section 103, and a manufacturer of telecommunications transmission or switching equipment or a provider of telecommunications support services shall be found to be in compliance with section 106, if the carrier, manufacturer, or support service provider is in compliance with publicly available technical requirements or standards adopted by an industry association or standard-setting organization, or by the Commission under subsection (b), to meet the requirements of section 103.

(3) **ABSENCE OF STANDARDS-** The absence of technical requirements or standards for implementing the assistance capability requirements of section 103 shall not—

(A) preclude a telecommunications carrier, manufacturer, or telecommunications support services provider from deploying a technology or service; or

(B) relieve a carrier, manufacturer, or telecommunications support services provider of the obligations imposed by section 103 or 106, as applicable.

(b) **COMMISSION AUTHORITY-** If industry associations or standard-setting organizations fail to issue technical requirements or standards or if a Government agency or any other person believes that such requirements or standards are deficient, the agency or person may petition the Commission to establish, by rule, technical requirements or standards that—

(1) meet the assistance capability requirements of section 103 by cost-effective methods;

(2) protect the privacy and security of communications not authorized to be intercepted;

(3) minimize the cost of such compliance on residential ratepayers;

(4) serve the policy of the United States to encourage the provision of new technologies and services to the public; and

(5) provide a reasonable time and conditions for compliance with and the transition to any new standard, including defining the obligations of telecommunications carriers under section 103 during any transition period.

**(c) EXTENSION OF COMPLIANCE DATE FOR EQUIPMENT, FACILITIES, AND SERVICES-**

(1) **PETITION-** A telecommunications carrier proposing to install or deploy, or having installed or deployed, any equipment, facility, or service prior to the effective date of section 103 may petition the Commission for 1 or more extensions of the deadline for complying with the assistance capability requirements under section 103.

(2) **GROUNDS FOR EXTENSION-** The Commission may, after consultation with the Attorney General, grant an extension under this subsection, if the Commission determines that compliance with the assistance capability requirements under section 103 is not reasonably achievable through application of technology available within the compliance period.

(3) **LENGTH OF EXTENSION-** An extension under this subsection shall extend for no longer than the earlier of—

- (A) the date determined by the Commission as necessary for the carrier to comply with the assistance capability requirements under section 103;
- or
- (B) the date that is 2 years after the date on which the extension is granted.

(4) **APPLICABILITY OF EXTENSION-** An extension under this subsection shall apply to only that part of the carrier's business on which the new equipment, facility, or service is used.

These extensions, once routine, are now scrutinized much more closely by the FCC, FBI, and DOJ. Even for packet-based solutions like VoIP, the existence of adequate technical standards is forcing equipment manufacturers and carriers to show compliance with CALEA.

### *J-STD-025 and Other Technical Standards*

Shortly after CALEA was enacted, work began in Subcommittee TR-45.2 of the Telecommunications Industry Association (TIA) to create an appropriate technical interface

between Law Enforcement Agencies (LEAs) and carriers. Interim standard J-STD-025 was developed specifically to define services and features required by CALEA for “wireline, cellular, and broadband PCS carriers to support lawfully-authorized electronic surveillance, and specifies interfaces necessary to deliver intercepted communications and call-identifying information to a law enforcement agency.”

## NOTE

---

J-STD-025 and subsequent TIA technical standards referenced by FCC regulations—although available to the public—are not free. They can be purchased on the TIA Web site (see [www.tiaonline.org/standards/CALEA\\_JEM](http://www.tiaonline.org/standards/CALEA_JEM) for more information) or through the Alliance for Telecommunications Industry Solutions (ATIS—see [www.atis.org/atis/docstore/](http://www.atis.org/atis/docstore/) for more information). In general, most of the standards referenced in this section require membership or document fees to be paid in order to access the associated standard.

---

Originally published in December, 1997, J-STD-025, the standard was the subject of a March 27, 1998, Joint petition to the FCC from the DOJ and FBI, which argued that it was deficient in nine specific areas. This list commonly is referred to as the FBI “punch list” of additional capabilities, six of which were subsequently required by the FCC and incorporated into the revised J-STD-025-A specification, published by TR-45.2 in May, 2000.

Since that time, a number of standards have been developed by other industry groups and are recognized by the FBI and FCC as meeting the safe harbor provisions of CALEA. Many of these have been coordinated with ongoing TIA TR45 LAES work on J-STD-025. Among these standards are:

- TIA TR45 LAES J-STD-025B for CDMA2000 packet data intercepts
- T1P1 T1.724 for GPRS packet data intercepts
- T1S1 T1.678 for VoIP and other wire-line data intercepts
- PKT-SP-ESP-I03-40113 for PacketCable data intercepts
- AMTA Electronic Surveillance for ESMR Dispatch Version 1.0 for ESMR Push-To-Talk intercepts
- American Association of Paging Carriers (AAPC) Paging Technical Committee (PTC) CALEA Suite of Standards, Version 1.3 for Traditional Paging, Advanced Messaging, and Ancillary Services (see [www.pagingcarriers.org/ptc.asp](http://www.pagingcarriers.org/ptc.asp) for this freely available standard)

### *FCC CALEA Third Report and Order (August 1999)*

By 1999, the FCC was ready to require all carriers to implement the capabilities of the TIA J-standard and six FBI punch list capabilities by June 30, 2002. Packet-mode communications capability (including VoIP) was to be implemented by September 30, 2002 (though in practice CALEA extensions for packet continued routinely until late 2005). In addition, the FCC reached important conclusions regarding location information (not directly specified by the Act itself) and packet-mode communications capabilities. The FCC press release states:

#### **Actions Regarding the Interim Standard (J-STD-025)**

The FCC concluded the following regarding the location information and packet-mode communications capabilities of the interim standard:

**Location information:** The FCC required that location information be provided to law enforcement agencies (LEAs) under CALEA's assistance capability requirements for "call-identifying information," provided that a LEA has a court order or legal authorization beyond a pen register or trap and trace authorization. The FCC found that location information identifies the "origin" or "destination" of a communication and thus is covered by CALEA. The FCC, however, did not mandate that carriers be able to provide LEAs with the precise physical location of a caller. Rather, it permitted LEAs with the proper legal authorization to receive from wireline, cellular, and broadband PCS carriers only the location of a cell site at the beginning and termination of a mobile call.

**Packet-mode communications:** The FCC required that carriers provide LEAs access to packet-mode communications by September 30, 2001. However, the Commission acknowledged that significant privacy issues had been raised with regard to the J-STD-025 treatment of packet-mode communications. Under the J-STD-025, law enforcement could be provided with access to both call identifying information and call content, even where it may be authorized only to receive call identifying information. Accordingly, the FCC invited TIA to study CALEA solutions for packet-mode technology and report to the FCC by September 30, 2000 on steps that can be taken, including particular amendments to the interim standard, that will better address privacy concerns.

#### **Actions Regarding the Capabilities Requested by DoJ/FBI**

Of the nine items in the DoJ/FBI punch list, the following capabilities were required by the FCC:

**Content of subject-initiated conference calls**— A LEA will be able to access the content of conference calls initiated by the subject under surveillance (including the call content of parties on hold), pursuant to a court order or other legal authorization beyond a pen register order.

**Party hold, join, drop on conference calls**— Messages will be sent to a LEA that identify the active parties of a call. Specifically, on a conference call, these messages will indicate whether a party is on hold, has joined, or has been dropped from the conference call.

**Subject-initiated dialing and signaling information**— Access to dialing and signaling information available from the subject will inform a LEA of a subject's use of features (e.g., call forwarding, call waiting, call hold, and three-way calling).

**In-band and out-of-band signaling (notification message)**— A message will be sent to a LEA whenever a subject's service sends a tone or other network message to the subject or associate (e.g., notification that a line is ringing or busy, call waiting signal).

**Timing information**— Information will be sent to a LEA permitting it to correlate call-identifying information with the call content of a communications interception.

**Dialed digit extraction**—The originating carrier will provide to a LEA on the call data channel any digits dialed by the subject after connecting to another carrier's service., pursuant to a pen register authorization. The FCC found that some such digits fit within CALEA's definition of call-identifying information, and that they are generally reasonably available to carriers

In requiring the six punch list capabilities, the FCC noted that it determined that five of them constitute call-identifying information that is generally reasonably available to carriers and therefore is required under CALEA. The FCC found that although the cost to carriers of providing some of these five capabilities is significant, no automatic exemptions will be provided. Exclusions must be filed and approved on a case-by-case basis.

The following punch list items were not required by the FCC:

**Surveillance status**—Carriers would have been required to send a message to a LEA to verify that a wiretap had been established and was functioning correctly.



**Continuity check tone (C-tone)**— Electronic signal would have alerted a LEA if the facility used for delivery of call content interception failed or lost continuity.

**Feature status**— A LEA would have been notified when, for the facilities under surveillance, specific subscription-based calling services were added or deleted.

The FCC found that these three capabilities, although potentially useful to LEAs, were not required by the plain language of CALEA. However, carriers are free to provide these capabilities if they wish to do so.

### *DOJ-FBI-DEA Joint Petition for Expedited Rulemaking (March 2004)*

Given CALEA's stated purpose, namely to "preserve law enforcement's ability to conduct lawful electronic surveillance despite changing telecommunications technologies," the DOJ, FBI, and DEA felt that key aspects of the law and its original intent were not being addressed by the FCC, carriers, and equipment manufacturers. The petition states:

CALEA applies to all telecommunications carriers, and its application is technology neutral. Despite a clear statutory mandate, full CALEA implementation has not been achieved. Although the Commission has taken steps to implement CALEA, there remain several outstanding issues that are in need of immediate resolution.

To resolve the outstanding issues, law enforcement asks the Commission to:

- (1) formally identify the types of services and entities that are subject to CALEA;
- (2) formally identify the services that are considered "packet-mode services";
- (3) initially issue a Declaratory Ruling or other formal Commission statement, and ultimately adopt final rules, finding that broadband access services and broadband telephony services are subject to CALEA;
- (4) reaffirm, consistent with the Commission's finding in the CALEA Second Report and Order, that push-to-talk "dispatch" service is subject to CALEA;

- (5) adopt rules that provide for the easy and rapid identification of future CALEA-covered services and entities;
- (6) establish benchmarks and deadlines for CALEA packet-mode compliance;
- (7) adopt rules that provide for the establishment of benchmarks and deadlines for CALEA compliance with future CALEA-covered technologies;
- (8) outline the criteria for extensions of any benchmarks and deadlines for compliance with future CALEA-covered technologies established by the Commission;
- (9) establish rules to permit it to request information regarding CALEA compliance generally;
- (10) establish procedures for enforcement action against entities that do not comply with their CALEA obligations;
- (11) confirm that carriers bear sole financial responsibility for CALEA implementation costs for post-January 1, 1995 communications equipment, facilities and services;
- (12) permit carriers to recover their CALEA implementation costs from their customers; and
- (13) clarify the cost methodology and financial responsibility associated with intercept provisioning.

In general, existing FCC rules are incomplete, inconsistent, or otherwise inadequate in these areas and you should expect to see new or clarified regulations from the FCC over the next few years that address the DOJ/FBI/DEA concerns. Many of these will directly impact VoIP systems design and operational practices within carriers.

### *FCC First Report and Order and Further Notice of Proposed Rulemaking, (September, 2005)*

In response to the DOJ-FBI-DEA Joint Petition, the FCC ruled that CALEA does apply to providers of certain broadband and interconnected VoIP services. From the FCC press release:

The Commission found that these services can essentially replace conventional telecommunications services currently subject to wiretap rules, including circuit-switched voice service and dial-up Internet access. As replacements, the new services are covered by the Communications Assistance for Law Enforcement Act, or CALEA, which requires the Commission to preserve the ability of law enforcement agencies to conduct court-ordered wiretaps in the face of technological change.

The Order is limited to facilities-based broadband Internet access service providers and VoIP providers that offer services permitting users to receive calls from, and place calls to, the public switched telephone network. These VoIP providers are called interconnected VoIP providers.

The Commission found that the definition of “telecommunications carrier” in CALEA is broader than the definition of that term in the Communications Act and can encompass providers of services that are not classified as telecommunications services under the Communications Act. CALEA contains a provision that authorizes the Commission to deem an entity a telecommunications carrier if the Commission “finds that such service is a replacement for a substantial portion of the local telephone exchange.”

Because broadband Internet and interconnected VoIP providers need a reasonable amount of time to come into compliance with all relevant CALEA requirements, the Commission established a deadline of 18 months from the effective date of this Order, by which time newly covered entities and providers of newly covered services must be in full compliance. The Commission also adopted a Further Notice of Proposed Rulemaking that will seek more information about whether certain classes or categories of facilities-based broadband Internet access providers – notably small and rural providers and providers of broadband networks for educational and research institutions – should be exempt from CALEA.

The Commission’s action is the first critical step to apply CALEA obligations to new technologies and services that are increasingly used as a substitute for conventional services. The Order strikes an appropriate balance between fostering competitive broadband and advanced services deployment and technological innovation on one hand, and meeting the needs of the law enforcement community on the other.

The potential impact of this ruling is huge and will reverberate within the VoIP and broadband communities over the next few years. What is perhaps most surprising is the

determination that broadband data services will need to support a lawful intercept function. Lawsuits have already been filed (partly over the unfunded mandate the FCC created for higher education: an estimated \$7 billion in CALEA implementation costs are expected for colleges and universities alone, according to EDUCAUSE). Much of the story has yet to be written but the impact of this round of FCC rulemaking on the VoIP community will be hard to overstate. How this will affect Skype and other consumer services in the long run remains to be seen, but in the meantime this FNPR has served as a shot across the bow of the VoIP industry.

### *Telecommunications Carrier Systems Security and Integrity Plan*

The FCC mandates that carriers file this plan as part of their CALEA compliance. From the FCC CALEA page:

CALEA also requires telecommunications carriers to file with the Commission information regarding the policies and procedures used for employee supervision and control, and to maintain secure and accurate records of each communications interception or access to call-identifying information. In particular, all carriers that must comply with CALEA's capacity and capability requirements must also comply with 47 C.F.R. §§64.2100 - 64.2106 of the Commission's rules (available at [www.access.gpo.gov/nara/cfr/waisidx\\_03/47cfr64\\_03.html](http://www.access.gpo.gov/nara/cfr/waisidx_03/47cfr64_03.html)) by filing with the Commission a Telecommunications Carrier Systems Security and Integrity Plan. Resellers of local exchange services, both facilities-based and switchless, must also comply with these rules by filing a Systems Security and Integrity Plan.

## What a CALEA Consultant Will Tell You

First and foremost, it's very important to know for sure if your organization is required to comply with CALEA. At this point, the FCC has issued extensive guidance but it still does not cover all cases. A CALEA expert can help guide you through existing precedent and determine which—if any—of the services offered by your organization must be compliant with CALEA. From there, identifying any safe harbor standards accepted by the FCC and FBI is the next step. If you can implement one or more safe harbor standards, then do it and consider yourself lucky.

If you can't, you'll need some help determining which section to file under (107 or 109) so that the FCC can grant you a little breathing room while you figure out what your long-term solution will be (presumably with the help of your VoIP system vendor(s)). Unfortunately, today's VoIP systems are a little behind the curve on implementing CALEA standards, and if your software or hardware providers don't already have a plan to address CALEA, you may want to consider alternatives since the FCC has signaled that it will no

longer routinely grant deferrals and other exceptions when adequate technical solutions exist and are available to the market.

### Tools & Traps...

#### **Core CALEA-Compliance Issues for IP Communications Systems**

Unlike regulations like HIPAA, GLBA, or SOX, within CALEA there is more focus on equipment capabilities and standards as part of CALEA compliance. Know which standard to apply (start with J-STD-025B or T1.678 for VoIP). Retrofitting a compliant solution over a noncompliant system can be difficult and expensive for VoIP, so if you are required to comply with CALEA, make sure that your equipment (or software) supplier evaluation / procurement process adequately screens for CALEA support, and be sure to stay on top of FCC filings (and the latest FCC orders, since VoIP rules under CALEA are still being worked out). Consider filing a comment with the FCC if you're reading this early enough in the rulemaking process.

## CALEA Compliance and Enforcement

In general, the FCC (with input from the DOJ and FBI) is responsible for compliance (although there are minor aspects of CALEA that the DOJ can enforce directly).

### Certification

Individual LEAs can be CALEA-certified, but in general that term isn't applied to equipment or carriers. Equipment sold to carriers can (and should) be CALEA Section 106-compliant in the sense that if it meets a standard accepted by the FCC (and/or FBI in some cases where a technical standard hasn't been adopted by the FCC regulations directly). Use of CALEA-compliant equipment by a carrier will bring Section 107 Safe Harbor provisions into play to deem that service to be CALEA-compliant. In general, however, it is a carrier and associated service that can be certified as compliant, by meeting Section 103 requirements directly with the agreement of the FBI (these have been phased out as technical standards now fill the gap that once required this FBI consent) or by meeting FCC mandates and Filing directly with the FCC for certification.

## Enforcement Process and Penalties

The FCC requires appropriate CALEA filings by each carrier and can impose fines when those filings are missing, incomplete, or otherwise not in line with CALEA regulations. At this point, there have been no major fines but the threat of fines has kept most carriers on top of all required filings. With the inclusion of VoIP it will be interesting to see if the FCC takes a “get tough” stance on CALEA enforcement once the current set of VoIP and packet-related CALEA lawsuits has been resolved.

Elliott Eichen at Verizon suggests a “Four-Step Process” to describe the regulatory experience surrounding CALEA (and E911) compliance; it rings particularly true for me:

1. Denial: “Not us!”
2. Depression: “We can’t do it technically.”
3. Anger: “This is going to cost a fortune!”
4. Acceptance: “CALEA and E911 are not going away; let’s make it work.”

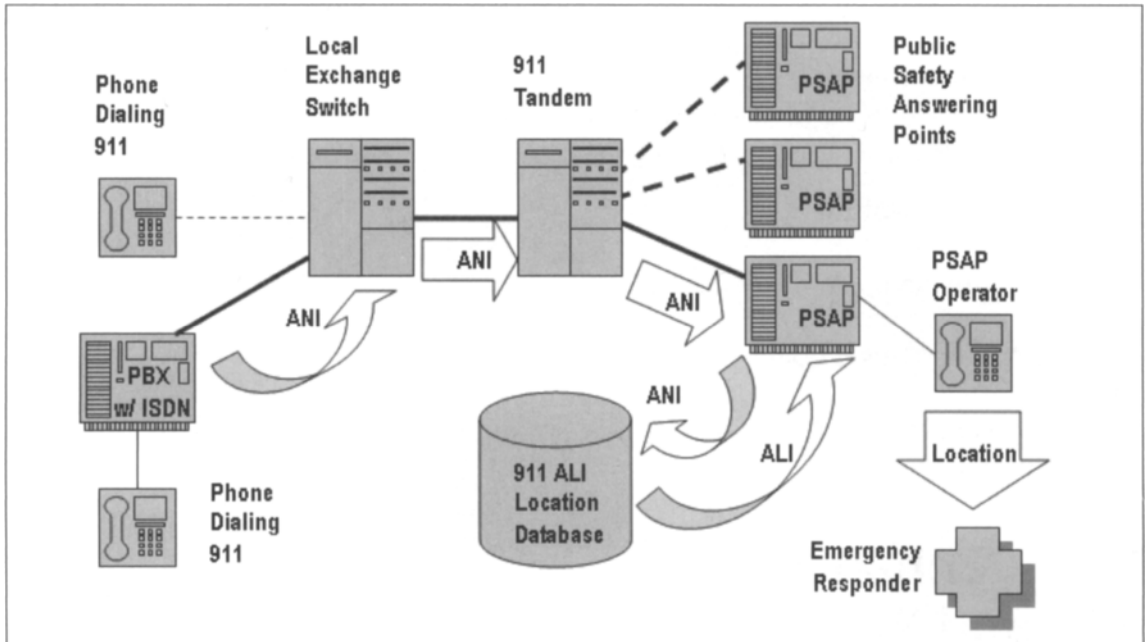
## E911: Enhanced 911 and Related Regulations

Within the United States and Canada, 911 is the official national emergency number; calls to 911 are directed to the most appropriate Public Safety Answering Point (PSAP) dispatcher for local emergency medical, fire, and law enforcement agencies via specialized infrastructure. Enhanced 911 (E911) systems automatically show the PSAP a calling number telephone number and location for wireline phones using the Automatic Location Identifier (ALI) database (maintained specifically for PSAP use, it translates a phone number from Automatic Number Identification (ANI) to a physical location). In 1996 the FCC established the wireless E911 program; which, when fully implemented, will provide a PSAP with a precise location for wireless 911 calls. Figure C.2 is an example of an enhanced 911 system. In this example, the ALI Location Database translates an ANI identifier into a physical location that can be used for emergency dispatch.

Given all the progress around E911 it may come as a surprise to you that 911 failures due to incomplete VoIP E911 design have led to several high-profile, preventable deaths (accompanied by lawsuits and demand for increased regulation). In fact, the rise of VoIP carriers that are interconnected with the PSTN has been accompanied by two massive breakdowns in E911 capability that eventually forced an urgent VoIP E911 order from the FCC in June, 2005. The first involves VoIP carriers not having adequate interconnection arrange-

ments to pass E911 calls. But the second is the more interesting problem. What happens when you can register a VoIP phone over an IP network from any physical location in the world (so long as it can be connected to the Internet)?

**Figure C.2** An Enhanced 911 System



## E911 Regulatory Basics

There are several dimensions to E911, the most important being the distinction between wireline and wireless regulations. But in this section we will focus exclusively on the FCC VoIP E911 rulings in 2005 that have added an important new dimension to FCC rules for E911.

### Direct from the Regulations

On June 3, 2005 the FCC released the *VoIP 911 Order* requiring interconnected VoIP providers to provide their new and existing subscribers with 911 service no later than November 28, 2005. The FCC accompanying press release gives an excellent summary of the resulting regulations:

Specifically, as a condition of providing interconnected VoIP service, each interconnected VoIP provider must, in addition to satisfying the subscriber notification, acknowledgment, and labeling requirements set forth in section 9.5(e) of the Commission's rules.

- Transmit all 911 calls to the public safety answering point (PSAP), designated statewide default answering point, or appropriate local emergency authority that serves the caller's "Registered Location." Such transmissions must include the caller's Automatic Numbering Information (ANI) [ANI is a system that identifies the billing account for a call and, for 911 systems, identifies the calling party and may be used as a call back number] and Registered Location to the extent that the PSAP, designated statewide default answering point, or appropriate local emergency authority is capable of receiving and processing such information;
- Route all 911 calls through the use of ANI and, if necessary, pseudo-ANI [Pseudo-ANI is "a number, consisting of the same number of digits as ANI, that is not a North American Numbering Plan telephone directory number and may be used in place of an ANI to convey special meaning. The special meaning assigned to the pseudo-ANI is determined by agreements, as necessary, between the system originating the call, intermediate systems handling and routing the call, and the destination system], via the Wireline E911 Network, [a "dedicated wireline network that: (1) is interconnected with but largely separate from the public switched telephone network; (2) includes a selective router; and (3) is utilized to route emergency calls and related information to PSAPs, designated statewide default answering points, appropriate local emergency authorities or other emergency answering points."] and make a caller's Registered Location available to the appropriate PSAP, designated statewide default answering point or appropriate local emergency authority from or through the appropriate Automatic Location Identification (ALI) database;
- Obtain from each of its existing and new customers, prior to the initiation of service, a Registered Location; and
- Provide all of their end users one or more methods of updating their Registered Location at will and in a timely manner. At least one method must allow end users to use only the same equipment (such as the Internet telephone) that they use to access their interconnected VoIP service.

### Compliance Letters

Additionally, given the vital public safety interests at stake, the VoIP 911 Order requires each interconnected VoIP provider to file with the



Commission a Compliance Letter on or before November 28, 2005 detailing its compliance with the above 911 requirements. To ensure that interconnected VoIP providers have satisfied the requirements set forth above, we require interconnected VoIP providers to include the following information in their Compliance Letters:

- **911 Solution:** This description should include a quantification, on a percentage basis, of the number of subscribers to whom the provider is able to provide 911 service in compliance with the rules established in the VoIP 911 Order. Further, the detailed description of the technical solution should include the following components:
  1. **911 Routing Information/Connectivity to Wireline E911 Network:** A detailed statement as to whether the provider is transmitting, as specified in Paragraph 42 of the VoIP 911 Order, “all 911 calls to the appropriate PSAP, designated statewide default answering point, or appropriate local emergency authority utilizing the Selective Router, the trunk line(s) between the Selective Router and the PSAP, and such other elements of the Wireline E911 Network as are necessary in those areas where Selective Routers are utilized.” If the provider is not transmitting all 911 calls to the correct answering point in areas where Selective Routers are utilized, this statement should include a detailed explanation why not. In addition, the provider should quantify the number of Selective Routers to which it has interconnected, directly or indirectly, as of November 28, 2005.
  2. **Transmission of ANI and Registered Location Information:** A detailed statement as to whether the provider is transmitting via the Wireline E911 Network the 911 caller’s ANI and Registered Location to all answering points that are capable of receiving and processing this information. This information should include: (i) a quantification, on a percentage basis, of how many answering points within the provider’s service area are capable of receiving and processing ANI and Registered Location information that the provider transmits; (ii) a quantification of the number of subscribers, on a percentage basis, whose ANI and Registered Location are being transmitted to answering points that are capable of receiving and processing this information; and (iii) if the provider is not transmitting the 911 caller’s ANI and Registered Location to all answering points that are capable of receiving and processing this information, a detailed explanation why not.

3. **911 Coverage:** To the extent a provider has not achieved full 911 compliance with the requirements of the VoIP 911 Order in all areas of the country by November 28, 2005, the provider should: 1) describe in detail, either in narrative form or by map, the areas of the country, on a MSA basis, where it is in full compliance and those in which it is not; and 2) describe in detail its plans for coming into full compliance with the requirements of the order, including its anticipated timeframe for such compliance.
- **Obtaining Initial Registered Location Information:** A detailed description of all actions the provider has taken to obtain each existing subscriber's current Registered Location and each new subscriber's initial Registered Location. This information should include, but is not limited to, relevant dates and methods of contact with subscribers and a quantification, on a percentage basis, of the number of subscribers from whom the provider has obtained the Registered Location.
  - **Obtaining Updated Registered Location Information:** A detailed description of the method(s) the provider has offered its subscribers to update their Registered Locations. This information should include a statement as to whether the provider is offering its subscribers at least one option for updating their Registered Location that permits them to use the same equipment that they use to access their interconnected VoIP service.
  - **Technical Solution for Nomadic Subscribers:** A detailed description of any technical solutions the provider is implementing or has implemented to ensure that subscribers have access to 911 service whenever they use their service nomadically.

The Bureau notes that in an October 7, 2005 letter submitted in WC Docket Nos. 04-36 and 05-196, AT&T outlined an innovative compliance plan that it is implementing to address the Commission's 911 provisioning requirements that take effect on November 28, 2005. In letters filed on October 21, 2005 in these dockets, MCI and Verizon each outlined similar compliance plans. Each of these plans includes an automatic detection mechanism that enables the provider to identify when a customer may have moved his or her interconnected VoIP service to a new location and ensure that the customer continues to receive 911 service even when using the interconnected VoIP service nomadically. These plans also include a commitment to not accept new interconnected VoIP customers in areas where the provider cannot provide 911 service and to

adopt a “grandfather” process for existing customers for whom the provider has not yet implemented either full 911 service or the automatic detection capability.

The Bureau applauds the steps undertaken by AT&T, MCI and Verizon and strongly encourages other providers to adopt similar measures. The Bureau will carefully review a provider’s implementation of steps such as these in deciding whether and how to take enforcement action. Providers should include in their November 28, 2005, Compliance Letters a detailed statement as to whether and how they have implemented such measures. To the extent that providers have not implemented these or similar measures, they should describe what measures they have implemented in order to comply with the requirements of the VoIP 911 Order.

Although we do not require providers that have not achieved full 911 compliance by November 28, 2005, to discontinue the provision of interconnected VoIP service to any existing customers, we do expect that such providers will discontinue marketing VoIP service, and accepting new customers for their service, in all areas where they are not transmitting 911 calls to the appropriate PSAP in full compliance with the Commission’s rules.

## What an E911 Consultant Will Tell You

This is a very active and emerging space, particularly around VoIP E911, but the National Emergency Number Association (NENA) has some excellent recommendations in this area. They have published a 9-1-1 System Reference Guide (go to [www.nena.org](http://www.nena.org) for more information) that is “a single-source reference for PSAP and Selective Router administrative data”—invaluable information for a VoIP carrier that needs to comply with the new FCC order. Also underway is a NG E9-1-1 Program, a public-private partnership to improve the nation’s 9-1-1 system and provide necessary VoIP and PSAP standards to make deployable VoIP E911 more achievable.

## Tools & Traps...

### Core E911-Compliance Issues for IP Communications Systems

As with CALEA, there is a bit more focus on equipment capabilities and standards as part of compliance. However, retrofitting a compliant solution over a noncompliant system isn't necessarily difficult and expensive if it's well planned. Regardless, E911 should be a critical part of your vendor-facing solution evaluation / procurement process.

For enterprise VoIP systems, the critical considerations involve local regulations that require accurate information for ALI tables. Many enterprise system vendors have location databases, capabilities for end-user location self-reporting, and partnerships with third-party solutions for maintaining location information even when IP phones are moved.

## E911 Compliance and Enforcement

The FCC and the National Association of Regulatory Utility Commissioners (NARUC) formed the Joint Federal/State VoIP Enhanced 911 Enforcement Task Force to facilitate compliance with FCC VoIP 911 rules as well as any necessary enforcement. The Task Force is made up of FCC staff and representatives from various State PUCs, and operates in conjunction with NENA, the Association of Public Safety Communications Officials, and various state and local emergency authorities. The Task Force's mission is to "develop educational materials to ensure that consumers understand their rights and the requirements of the FCC's VoIP 911 Order; develop appropriate compliance and enforcement strategies; compile data; and share best practices."

### Self-Certification

At this point, the FCC process requires a self-certification by each VoIP carrier that must be filed with the FCC. As standards emerge, some form of product certification for VoIP E911 may eventually take place.

### Enforcement Process and Penalties

Despite the number of extensions granted by the FCC in 2005, a number of fines and other penalties have been levied recently against noncompliant VoIP carriers. State and local agencies also are involved in enforcement and follow their own enforcement regimes.

# EU and EU Member States' eCommunications Regulations

In April 2002, a European Union (EU) regulatory framework for electronic communications was adopted and went into effect in July 2003. In its introduction to the framework, the EU Information Society Directorate-General explains:

The convergence of the telecommunications, media and information technology sectors demands a single regulatory framework that covers all transmission networks and services. The EU regulatory framework addresses all communications infrastructure in a coherent way, but does not cover the content of services delivered over and through those networks and services. There are five different directives: the Framework Directive<sup>6</sup> (2002/21/EC) and four specific directives, being the Authorisation Directive<sup>7</sup> (2002/20/EC), the Access Directive<sup>8</sup> (2002/19/EC), the Universal Service Directive<sup>9</sup> (2002/22/EC) and the Privacy Directive<sup>10</sup> (2002/58/EC). In addition, the Competition Directive (2002/77/EC) applies.

The objectives set out in the EU regulatory framework are:

- To promote competition by fostering innovation, liberalising markets and simplifying market entry;
- To promote the single European market and;
- To promote the interest of citizens.

All Member States are required to implement the EU framework in their national law. The framework lays down the role of Member States and national regulatory authorities, the rights and obligations for market players, and the rights of users of electronic communications networks and services. In addition, Member States may take measures justified on the grounds of public health and public security as set out in the EC Treaty, for example by imposing requirements for legal interception or critical infrastructure protection, and such measures are not covered by the EU regulatory framework.

What many non-EU readers may not realize is the degree to which EU regulations (particularly privacy regulations) will force specific policy and practice outside of the EU. Its effects (particularly with respect to VoIP) will be briefly discussed in this final section. At the present, the EU IS Directorate-General is soliciting public comment on VoIP policy for input into a future regulatory regime for VoIP.

## EU Regulatory Basics

Seven active EU Communications Directives with potential VoIP Implications that your organization may need to consider are listed here. Note that each of these directives is required to be expressed within the law for each EU nation, which may have additional regulatory measures of their own. In some cases (such as with the German Data Privacy Law) national laws are considerably more restrictive than the overall EU directive. Here is the list:

- **Directive 97/66/EC** Processing of personal data and protection of privacy (up to October 30, 2003)
- **Directive 2002/58/EC** Privacy and electronic communications (from October 31, 2003 onward)
- **Directive 2002/19/EC** Access and interconnection
- **Directive 2002/20/EC** Authorization of electronic communications networks and services (i.e., allocation of radio frequencies)
- **Directive 2002/21/EC** Common regulatory framework
- **Directive 2002/22/EC** Universal service and users' rights relating to electronic communications networks and services
- **Directive 2002/77/EC** On competition in the markets for electronic communications services

Although VoIP is directly or indirectly addressed in each of these, this section will focus on the only VoIP security concern addressed in the EU electronic communications regulations, namely the privacy and electronic communications directive.

### Direct from the Regulations

Central to understanding EU privacy laws are the broad definitions used for personal data and its processing. We will focus on Directive 2002/58/EC since it establishes the minimum go-forward privacy framework for EU member states going forward with respect to electronic communications services. Note that despite specific references to ISDN and mobile networks in this directive, subsequent guidance from the EU IS Directorate-General has indicated that VoIP services will be expected to comply with this directive as well. Here is the relevant text within the directive:

#### **Article 3 - Services concerned**

1. This Directive shall apply to the processing of personal data in connection with the provision of publicly available telecommunications services in public telecommunications networks in the Community, in particular

via the Integrated Services Digital Network (ISDN) and public digital mobile networks.

2. Articles 8 ([www.bild.net/dataprEU1.htm#HD\\_NM\\_8](http://www.bild.net/dataprEU1.htm#HD_NM_8)), 9 ([www.bild.net/dataprEU1.htm#HD\\_NM\\_9](http://www.bild.net/dataprEU1.htm#HD_NM_9)) and 10 ([www.bild.net/dataprEU1.htm#HD\\_NM\\_10](http://www.bild.net/dataprEU1.htm#HD_NM_10)) shall apply to subscriber lines connected to digital exchanges and, where technically possible and if it does not require a disproportionate economic effort, to subscriber lines connected to analogue exchanges.

3. Cases where it would be technically impossible or require a disproportionate investment to fulfill the requirements of Articles 8 ([www.bild.net/dataprEU1.htm#HD\\_NM\\_8](http://www.bild.net/dataprEU1.htm#HD_NM_8)), 9 ([www.bild.net/dataprEU1.htm#HD\\_NM\\_9](http://www.bild.net/dataprEU1.htm#HD_NM_9)) and 10 ([www.bild.net/dataprEU1.htm#HD\\_NM\\_10](http://www.bild.net/dataprEU1.htm#HD_NM_10)) shall be notified to the Commission by the Member States.

#### **Article 4 - Security**

1. The provider of a publicly available telecommunications service must take appropriate technical and organizational measures to safeguard security of its services, if necessary in conjunction with the provider of the public telecommunications network with respect to network security. Having regard to the state of the art and the cost of their implementation, these measures shall ensure a level of security appropriate to the risk presented.

2. In case of a particular risk of a breach of the security of the network, the provider of a publicly available telecommunications service must inform the subscribers concerning such risk and any possible remedies, including the costs involved.

#### **Article 5 - Confidentiality of the communications**

Member States shall ensure via national regulations the confidentiality of communications by means of public telecommunications network and publicly available telecommunications services. In particular, they shall prohibit listening, tapping, storage or other kinds of interception or surveillance of communications, by others than users, without the consent of the users concerned, except when legally authorized.

## **Article 8 - Presentation and restriction of calling and connected line identification**

1. Where presentation of calling-line identification is offered, the calling user must have the possibility via a simple means, free of charge, to eliminate the presentation of the calling-line identification on a per-call basis. The calling subscriber must have this possibility on a per-line basis.
2. Where presentation of calling-line identification is offered, the called subscriber must have the possibility via a simple means, free of charge for reasonable use of this function, to prevent the presentation of the calling line identification of incoming calls.
3. Where presentation of calling line identification is offered and where the calling line identification is presented prior to the call being established, the called subscriber must have the possibility via a simple means to reject incoming calls where the presentation of the calling line identification has been eliminated by the calling user or subscriber.
4. Where presentation of connected line identification is offered, the called subscriber must have the possibility via a simple means, free of charge, to eliminate the presentation of the connected line identification to the calling user.
5. The provisions set out in paragraph 1 shall also apply with regard to calls to third countries originating in the Community; the provisions set out in paragraphs 2, 3 and 4 shall also apply to incoming calls originating in third countries.
6. Member States shall ensure that where presentation of calling and/or connected line identification is offered, the providers of publicly available telecommunications services inform the public thereof and of the possibilities set out in paragraphs 1, 2, 3 and 4.

## **Article 9 - Exceptions**

Member States shall ensure that the provider of a public telecommunications network and/or publicly available telecommunications service may override the elimination of presentation of the calling line identification:



(a) on a temporary basis, upon application of a subscriber requesting the tracing of malicious or nuisance calls; in this case, in accordance with national law, the data containing the identification of the calling subscriber will be stored and be made available by the provider of a public telecommunications network and/or publicly available telecommunications service;

(b) on a per-line basis for organizations dealing with emergency calls and recognized as such by a Member State, including law enforcement agencies, ambulance services and fire brigades, for the purpose of answering such calls.

#### **Article 10 - Automatic call forwarding**

Member States shall ensure that any subscriber is provided, free of charge and via a simple means, with the possibility to stop automatic call forwarding by a third party to the subscriber's terminal.

#### **Article 11 - Directories of subscribers**

1. Personal data contained in printed or electronic directories of subscribers available to the public or obtainable through directory enquiry services should be limited to what is necessary to identify a particular subscriber, unless the subscriber has given his unambiguous consent to the publication of additional personal data. The subscriber shall be entitled, free of charge, to be omitted from a printed or electronic directory at his or her request, to indicate that his or her personal data may not be used for the purpose of direct marketing, to have his or her address omitted in part and not to have a reference revealing his or her sex, where this is applicable linguistically.

2. Member States may allow operators to require a payment from subscribers wishing to ensure that their particulars are not entered in a directory, provided that the sum involved is reasonable and does not act as a disincentive to the exercise of this right.

3. Member States may limit the application of this Article to subscribers who are natural persons.

## Article 12 - Unsolicited calls

1. The use of automated calling systems without human intervention (automatic calling machine) or facsimile machines (fax) for the purposes of direct marketing may only be allowed in respect of subscribers who have given their prior consent.
2. Member States shall take appropriate measures to ensure that, free of charge, unsolicited calls for purposes of direct marketing, by means other than those referred to in paragraph 1, are not allowed either without the consent of the subscribers concerned or in respect of subscribers who do not wish to receive these calls, the choice between these options to be determined by national legislation.
3. Member States may limit the application of paragraphs 1 and 2 to subscribers who are natural persons.

## What an EU Data Privacy Consultant Will Tell You

In addition to the EU eCommunications framework, you may need to worry about data contained in corporate directories. Any collection, use, disclosure, or other processing about an individual that resides within the EU requires careful handling that goes far beyond that prescribed by the privacy provisions contained in U.S. law for GLBA or HIPAA with their associated regulations. This can create legal issues within the EU regardless of whether the individuals are employees, consumers, suppliers, or other legal entities. Cross-border data transfer restrictions may prohibit the transfer of such data to a jurisdiction without an equivalent data protection regime. For export to the United States, the FTC provides a Safe Harbor program that can meet this test, but there are significant tradeoffs to taking this route so you should consult with an EU data privacy expert before committing to this route. In many respects, addressing EU data privacy rules is more art than science.

Tools & Traps...

### Core EU-Compliance Issues for IP Communications Systems

EU member countries do have important differences in data privacy rules, so be sure to consult appropriate experts for the countries in which you operate. Rules for VoIP as an eCommunication service may vary somewhat from the data-centric rules used for

Continued

data applications. Unless your organization is a vendor or carrier, most other EU compliance issues will be addressed by purchasing equipment and services approved for sale within the EU.

## EU Compliance and Enforcement

Within the EU and member states, compliance and enforcement happens at several levels. Some member states, such as Germany, have enforcement of privacy and electronic communication laws at a more local level as well as on a national basis. Decisions at the national level can be appealed at the EU level, and critical precedents are often set at this level.

### No Certification

In general, the EU and member countries do not have certification processes for the privacy and eCommunications regulation.

### Enforcement Process and Penalties

Data privacy fines can be stiff within the EU and its member states, though they do vary considerably by jurisdiction.

## Summary

Unfortunately, the trend is clearly heading toward *more* regulation, not less. By the time you read this, another VoIP-affecting regulation will have been enacted in some part of the world. In the United States, regulations like California's SB 1386 (which forces security breach notifications or end-to-end encryption of Social Security and credit card numbers and could impact you if you operate a VoIP call center) are being considered at the U.S. federal level and by other countries around the world.

# Index

## A

- AAA (authentication, authorization, and accounting), 165–166
- access control, 25
- Access Control Lists (ACLs), 235–237
- access points, 42
- ACD (Automatic Call Distribution), 21
- ACF/ARJ (Admission Confirm or Reject), 74
- acknowledgement (ACK) packet
  - DHCP, 127–128
  - in TFTP file transfer, 119, 120
- ACLs (Access Control Lists), 235–237
- active-response NIDSs, 194–195
- Address Resolution Protocol (ARP)
  - address resolution scheme, 149–150
  - spoofing, 151–155, 183
- adjunct servers, 24, 28–29
- Admission Confirm or Reject (ACF/ARJ), 74
- Admission Request (ARQ), 74
- Advanced Encryption Standard (AES)
  - for authentication, 164
  - SRTP packet encryption, 248–249
  - SRTP's use of, 247
- agents, in SNMP architecture, 124–125
- AH (Authentication Header), 222–223
- AIM (AOL's IM service), 36
- alerting, 194
- ALG (Application Layer Gateway), 34, 227–228
- Amplitude Research, 13
- analog line, 22
- Analog Telephone Adaptor (ATA), 33–34
- analog transmission, 49
- analog trunks, 24
- analog/digital conversion, 51
- ANI. *See* Automatic Number Identification
- anomaly detection, 189, 190
- Application Layer Gateway (ALG), 34, 227–228
- Application layer, OSI, 85–87
- application proxies, 34
- architectural model, PKI, 178–179
- architectures
  - of DNS, 113
    - H.235 security mechanisms, 78–82
    - H.323 protocol specification, 67–68
    - H.323 VoIP-related protocols, 68–77
  - instant messaging, 105–108
  - overview of, 109–110
  - PSTN, 46–61
    - PSTN call flow, 61–64
    - PSTN protocol security, 64–67
  - Session Initiation Protocol, 82–105
  - SIMPLE, 106, 107
  - of SNMP, 124
- ARP. *See* Address Resolution Protocol
- ARQ (Admission Request), 74
- ASN.1
  - encoding of signaling traffic, 72
    - H.245 messages encoded in, 75
    - H.323 vulnerabilities, 155–156
- Asterisk, 26
- AT&T, 55, 61
- ATA (Analog Telephone Adaptor), 33–34
- attack signatures, 187–188
- attackers, 26
- attacks
  - dictionary attacks, 175
  - DNS, 117
    - H.235 security mechanisms and, 81
    - H.323-specific attacks, 155–156
  - SIP-specific attacks, 156

- toll fraud, 27
  - voice messaging and, 28–29
  - VoIP threats, 12–13, 14–15
  - VoIP vulnerabilities, 5–6, 142
    - on WEP, 39
    - on WPA2, 40
  - authentication
    - 802.1x, 40–41
    - 802.1x/802.11i (WPA2), 163–164
    - 802.1x/EAP authentication, 164–167
    - basic certificate fields, 180
    - certificate revocation list, 181
    - certification path, 181–182
    - dictionary attacks, 175
    - EAP authentication types, 167–175
    - H.235 security mechanisms, 78
    - in H.323 environments, 161
    - HTTP server authentication, 122–123
    - MAC tools, 182–183
    - model authentication scheme, 161
    - overview of, 183–184
    - PKI entities, architectural model, 178–179
    - point solutions, 160–161
    - Public Key Infrastructure, 175–178
    - Session Initiation Protocol, 161–162
    - S/MIME message authentication, 241–244
    - SRTP message authentication, 249–250
    - types of, 160
    - for VoIP security, 157
  - authentication, authorization, and accounting (AAA), 165–166
  - Authentication Header (AH), 222–223
  - authentication server
    - in authentication process, 166
    - definition of, 164
    - in EAP authentication, 165
  - authenticator
    - definition of, 164
    - in MS-CHAP v2 authentication, 173–174
  - authorization, 160
    - authorized users, 17
  - Automatic Call Distribution (ACD), 21
  - Automatic Number Identification (ANI)
    - caller-ID spoofing, 155
    - description of/spoofing services, 58
    - security of, 66–67
  - availability, 12, 256
  - Avaya, 30–31
  - Avaya Media Encryption, 154
- ## B
- back-end servers (BES), 68
  - bandwidth, 211, 254
  - Bank of America, 2
  - baseline security profile, 79–81
  - basic authentication, of HTTP server, 122–123
  - basic certificate fields, 180
  - BBSes (Bulletin Board Systems), 106
  - Bell Operating Companies, 61
  - Bell System, 55
  - Bellovin, Steven, 226
  - Berners-Lee, Tim, 121
  - Berson, Tom, 256, 260
  - BES (back-end servers), 68
  - “Best Practices for Controlling Skype within the Enterprise” (BlueCoat), 257
  - bits, 49–50, 51
  - black listing, 282
  - blocking, Skype, 257–258, 282–283
  - BlueCoat, 257
  - bogus message DoS, 147
  - boot, 118
  - broadband, 6
  - brute-force attack, 40
  - Buddy List, 106–107
  - bugs, 137
  - Bulletin Board Systems (BBSes), 106
  - businesses
    - firewall for Skype, 266

- network device configuration for Skype, 269
  - ports for Skype, 271–272
  - wireless Skype communication, 282
- bypassing/NAT, 232–235
- ## C
- CA. *See* Certification Authority
  - cabling, 47–48
  - cache poisoning, 151–155
  - CALEA (Communications Assistance for Law Enforcement Act), 233
  - call control
    - H.245 call control messages, 75–77
    - media servers for, 32
    - separation of, 31
  - Call Detail Recording (CDR) systems, 30
  - call flow, PSTN, 61–64
  - call hijacking
    - ARP spoofing, 151–155
    - caller ID spoofing, 155
    - VoIP threats, 148–150
  - call signaling, 70–75
    - . *See also* signaling
  - caller ID, spoofing, 58, 155
  - Calling Party Identification Presentation (CLIP), 58
  - call-monitoring features, of PBX system, 26
  - canaries, 195
  - carrier-based PBX alternatives, 30
  - CAT5 cable, 41–42
  - CCM (Cisco CallManager), 135–137
  - CCS (Common Channel Signaling), 57
  - CDR (Call Detail Recording) systems, 30
  - Center for Internet Wireless Benchmarks, 277
  - Central Office (CO), 46–49
  - Centrex, 20, 30
  - certificate fields, basic, 180
  - Certificate Revocation List (CRL), 179, 181
  - Certificate Trust List (CTL), 213
  - certificates
    - basic certificate fields for X.509, 180
    - Certificate Revocation List, 181
    - certification path, 181–182
    - in SIP standard, 240–241
    - in S/MIME message authentication, 242–244
    - TLS, 244–245
    - TLS certificate, key exchange, 245–247
  - Certification Authority (CA)
    - basic certificate fields and, 180
    - certification path, 181–182
    - definition of, 178–179
    - as PKI entity, 179
    - self-signed certificates and, 245
  - certification path, 181–182
  - Challenge-Handshake Authentication Protocol (CHAP), 174
  - channel aggregation, DACS, 54
  - chat, 107
  - Choicepoint, 2
  - CIA (confidentiality, integrity, and availability), 12
  - CIDR (Classless Inter-Domain Routing), 264
  - ciphertext, 177
  - Cisco 7920 Wireless IP Phone, 126
  - Cisco bug ID CSCee08584, 137
  - Cisco CallManager (CCM), 135–137
  - Cisco Catalyst 6500 switches, 153–154
  - Cisco SIP Proxy Server (SPS), 156, 232
  - Cisco Systems, Inc., 135
  - Class 5 switch, 55–56
  - Class of Service (CoS), 214–215
  - Classless Inter-Domain Routing (CIDR), 264
  - clear text, 122
  - client
    - DHCP, 126–129
    - DNS operation, 115–116
    - HTTP client request, 121–122
  - client/server architecture, 93–94, 102

- CLIP (Calling Party Identification Presentation), 58
  - CO (Central Office), 46–49
  - codec
    - for H.323, 69–70
    - selection of, 51
    - for T1 transmission, 50–51
  - Common Channel Signaling (CCS), 57
  - communication
    - instant messaging, 105–108
    - of Skype, 267–269
  - communication policy, 261–262
  - Communications Assistance for Law Enforcement Act (CALEA), 233
  - community string, 125–126
  - compartmentalization, 208
  - computer (PC)
    - Skype security and, 255, 261
    - softphone and, 35–36
  - confidentiality
    - CIA principle, 12
    - of RTP packets, 247
    - with SRTP, 248–249
    - VoIP security, 12–13
  - confidentiality, integrity, and availability (CIA), 12
  - contact list, 106–107
  - control packet flood, VoIP, 147
  - converged networks
    - security issues in, 11–15
    - security vulnerabilities of, 5–6
    - switch to, 4–6
    - use of term, 6
    - . *See also* IP telephony; VoIP
  - convergence, of data/voice, 6–7
  - conversion, 51
  - corporations
    - blocking Skype, 283
    - router/firewall for Skype, 267–269
    - wireless Skype communication, 282
  - CoS (Class of Service), 214–215
  - cost
    - PSTN cost savings, 20
    - of VoIP, 6–7
  - coWPAtty, 40
  - CRL (Certificate Revocation List), 179, 181
  - crypto attribute, 248
  - CTL (Certificate Trust List), 213
- ## D
- DACS, 52, 54
  - DAI (Dynamic ARP Inspection), 153–154
  - data
    - convergence with voice, 6–7
    - information security breach, 2–3
    - separation of data/signaling, 9
    - Skype security and, 260–261
    - TFTP for data transfer, 118–120
  - Data Link layer, OSI, 85–86
  - data tunneling, 146
  - databases, signature, 188
  - DDoS (distributed denial-of-service) attack, 142–143
  - decentralized network, P2P, 105
  - Deep Packet Inspection (DPI), 228–229
  - defense in depth, 17
  - denial-of-service (DoS) attacks
    - harmfulness of, 154
    - mitigation of via VLANs, 211–212
    - QoS and, 215
    - SIP, 156
    - UDP, 98
    - on VoIP communication systems, 142–148
    - as VoIP threat, 13, 157
  - device authentication, 160
  - devices, in SNMP architecture, 124–125
  - DHCP. *See* Dynamic Host Configuration Protocol
  - DHCP Acknowledgement packet, 127–128
  - DHCP Discover packet, 127, 128
  - DHCP Offer packet, 127

- DHCP Request packet, 127, 128
- DIAMETER, 166
- dictionary attacks, 175
- DID (Direct Inward Dial) numbers, 25, 28
- DiffServ Code Points (DSCP), 215
- digest authentication, 123
- digestion, 242
- digital certificates. *See* certificates
- digital line, 23
- Digital Signal 0 (DS0), 52, 54
- digital signal hierarchy, 52–53
- digital signature
  - in public key cryptography, 177–178
  - in S/MIME message authentication, 242–243
- digital transmission, 49–54
- digital trunks, 24–25
- digital/analog conversion, 51
- Direct Endpoint signaling, 70
- Direct Inward Dial (DID) numbers, 25, 28
- discovery, 201–202
- Display Technical Call Information, Skype, 277–282
- distributed denial-of-service (DDoS) attack, 142–143
- Domain Name System (DNS)
  - architecture, 113
  - client operation, 115–116
  - description of, 112–113
  - FQDN, 114–115
  - overview of, 138
  - poisoning, 148–149
  - security implications for, 117
  - server operation, 116–117
- domains, in DNS architecture, 113
- DoS attacks. *See* denial-of-service (DoS) attacks
- Douglass, Dan, 277
- downloads, 257–258, 282
- DPI (Deep Packet Inspection), 228–229
- DS0 (Digital Signal 0), 52, 54
- DSCP (DiffServ Code Points), 215

- DSP resources, 32
- DSW Shoe Warehouses, 3
- Dual-Tone Multi-Frequency (DTMF)
  - dialing, 35, 55
- Dynamic ARP Inspection (DAI), 153–154
- Dynamic Host Configuration Protocol (DHCP)
  - description of, 126–127
  - operation, 127–128
  - overview of, 138
  - security implications for, 128–129
- dynamic NAT, 219–220

## E

- E.164 numbering scheme, 46
- EAP. *See* Extensible Authentication Protocol
- EAP-FAST (Extensible Authentication Protocol–Flexible Authentication via Secure Tunneling), 170, 171
- EAP-MD-5 (Extensible Authentication Protocol–Message Digest), 170, 172
- EAP-PEAP (Extensible Authentication Protocol–Protected Extensible Authentication Protocol), 170, 171
- EAP-TLS (Extensible Authentication Protocol–Transport Layer Security), 169, 170
- EAP-TTLS (Extensible Authentication Protocol–Tunneled Transport Layer Security), 170, 171
- eavesdropping, 78, 148
- 802.1q, 209–210
- 802.1x
  - authentication, 40–41, 163–164, 184
  - authentication terms, 164–167
  - EAP authentication types, 167–175
- 802.11i, 163–164
- e-mail, 241–244
- embedded Skype, 260
- Encapsulating Security Payload (ESP), 222–225
- encoding, 72



- encryption
    - 802.1x and, 164
    - for ARP spoofing protection, 154
    - H.235 security mechanisms, 78
    - NAT, 221–225
    - in PKI, 177
    - S/MIME message authentication, 242
    - SRTP packet encryption, 248–249
    - support protocols and, 112
    - of VoIP conversations, 13
    - WEP, 38–39
    - WPA2, 39–40
  - encryption solutions, IETF
    - S/MIME message authentication, 241–244
    - SRTP, voice/video packet security, 247–251
    - suites from IETF, 240–241
    - TLS, 244–247
  - end entity, 178, 179
  - end users, 157
  - endpoints
    - authentication of, 157
    - device authentication and, 160
    - H.323 call signaling, 72–73
    - of H.323 network, 67–68
    - IM clients, 36–37
    - rogue VoIP endpoint attack, 154
    - SIP and, 83
    - softphones, 35–36
    - video clients, 37
    - wireless VoIP clients, 38
  - energy budget, 43
  - equipment, telephony. *See* hardware infrastructure
  - ESP (Encapsulating Security Payload), 222–225
  - Ethereal, 153
  - Ethernet, 33, 34
  - exploitation, 203
  - extensibility, NIDSs, 194
  - Extensible Authentication Protocol (EAP)
    - authentication process, 40–41, 165–167
    - authentication terms, 164–165
    - authentication with, 184
    - definition of, 163
  - Extensible Authentication Protocol (EAP), authentication types, 167–175
    - chart of, 169–170
    - EAP-FAST, 171
    - EAP-MD-5, 172
    - EAP-PEAP, 171
    - EAP-TLS, 169
    - EAP-TTLS, 171
    - in general, 167–169
    - LEAP, 172
    - PEAPv1/EAP-GTC, 171
    - RainbowCrack, 172–173
  - Extensible Authentication Protocol-Flexible Authentication via Secure Tunneling (EAP-FAST), 170, 171
  - Extensible Authentication Protocol-Message Digest (EAP-MD-5), 170, 172
  - Extensible Authentication Protocol-Protected Extensible Authentication Protocol (EAP-PEAP), 170, 171
  - Extensible Authentication Protocol-Transport Layer Security (EAP-TLS), 169, 170
  - Extensible Authentication Protocol-Tunneled Transport Layer Security (EAP-TTLS), 170, 171
- ## F
- FDM (Frequency Division Multiplexing), 49
  - Field Programmable Gate Arrays (FPGAs), 229
  - file sharing, 108
  - file transfer
    - HTTP, 120–123
    - RSVP, 129–132
    - Skype, security issues of, 254
    - TFTP, 118–120

## firewall

- application layer gateway and, 34
  - bypassing/NAT, 232–235
  - deep packet inspection, 228–229
  - limits of, 205
  - medium-depth packet inspection, 227–228
  - NAT and, 265–266
  - NIDSs and, 192–193
  - overview of, 225–226
  - ports for Skype and, 272, 273–276
  - shallow packet inspection, 226–227
  - SIP firewall issues, 231–232
  - Skype and, 258–259
  - Skype blocking with, 257
  - for Skype businesses, 266
  - for Skype communication, 267–269
  - Skype configuration, 269, 270–271
  - for Skype home users, 266
  - stateful inspection, 227
  - voice firewall, 30
  - VoIP and, 5
  - VoIP message flow, 9–10
  - VoIP-aware firewalls, 229–231
  - wireless Skype communication, 277–282
- floods
- IP phone flood DoS, 148
  - UDP flood attack, 98
  - VoIP control packet flood, 147
- Ford, Bryan, 270–271
- FPGAs (Field Programmable Gate Arrays), 229
- FQDN (Fully Qualified Domain Name), 114–115
- Frequency Division Multiplexing (FDM), 49
- FTC, 2
- Full Zone NAT, 220
- full zone transfer, 117
- Fully Qualified Domain Name (FQDN), 114–115

**G**

- G.723.1 codec, 70
- G.700 series codecs, 69–70
- G.711 codec, 50–51, 69
- G.729 codec, 70
- Garfinkel, Simson, 256
- gatekeeper
  - H.323, 32, 67, 68
  - H.323 call signaling, 70–71, 72–75
- Gatekeeper Confirm or Reject (GCF/GRJ), 74
- Gatekeeper Request (GRQ), 74, 75
- Gatekeeper-Routed call signaling, 70
- gateway control protocols, 25, 101
- gateways
  - gatekeeper and, 32
  - of H.323 network, 67, 68
  - Media Gateway Control Protocol, 101
  - media gateways, 33–34
- GCF/GRJ (Gatekeeper Confirm or Reject), 74
- GET command, 121–122
- GETEXTRAREQUEST, 124–125
- GETREQUEST, 124
- Google Talk, 36, 37
- ground start, 22
- Group Policy, 273–276
- GRQ (Gatekeeper Request), 74, 75
- GTC, 175

**H**

- H.225.0/RAS, 79
- H.235.x, 240
- H.255.0/RAS, 69
- H.225/Q.931
  - call signaling, 71–75
  - function of, 69
  - H.235 security mechanisms, 79
  - ports, 70
  - vulnerability of, 155

- H.235
    - functions under H.323 architecture, 70
    - security mechanisms, 78–82
  - H.245
    - call control messages, 75–77
    - function of, 69
    - H.235 security mechanisms, 78
    - ports, 70
  - H.323
    - authentication in, 161
    - description of, 8, 67–68
    - firewall issues, 230–231
    - H.323-specific attacks, 155–156
    - overview of, 109
    - security vulnerabilities of, 158
    - VoIP security protocols and, 240
    - VoIP-related protocols, 68–77
  - H.323 gatekeeper, 32
  - hardware infrastructure
    - architecture models, 20
    - PBX alternatives, 30–31
    - PBX systems, traditional, 21–30
    - VoIP telephony, infrastructure, 31–44
  - hashing
    - inner authentication types, 173–174
    - in public key cryptography, 178
    - RainbowCrack, 172–173
    - SRTP message authentication, 249–250
  - heat budget, 43
  - heuristics, 190
  - HIDSs. *See* host-based intrusion detection systems
  - hijacking, 154
    - . *See also* call hijacking
  - Hilton, Paris, 3
  - Hitachi IP5000 phone, 126
  - home users
    - network device configuration for Skype, 269
    - ports for Skype, 271
    - router/firewall for Skype, 266
  - honeypots/honeynets, 195
  - host-based intrusion detection systems (HIDSs)
    - description of, 186
    - overview of, 196, 205
  - hosted IP-telephony services, 20, 30
  - HTTP server, 121–122
  - HTTPS, 123
  - HyperText Transfer Protocol (HTTP)
    - client request, 121–122
    - description of, 120–121
    - Digest Authentication, 162
    - overview of, 138
    - security implications for, 122–123
    - server response, 122
    - SIP and, 83
    - Skype blocks and, 257
- I**
- IANA (Internet Assigned Numbers Authority), 264
  - IAX (Inter-Asterisk Exchange Protocol), 26
  - ICE (Interactive Connectivity Establishment), 9, 234–235
  - ICV, 223
  - IEEE802.3af standard, 42
  - IETF. *See* Internet Engineering Task Force
  - IM. *See* instant messaging
  - immature software DoS, 147
  - implementation attacks, VoIP protocol, 155
  - IN (Intelligent Network), 56
  - incremental zone transfer, 117
  - information security, 2–4
  - inner authentication
    - EAP, 168, 169
    - types of, 173–175
  - instant messaging (IM)
    - clients, 36–37
    - description of, 105–107
    - presence, 7
    - security compromise, 107–108

- SIMPLE for, 107, 110
  - SIP for, 82
  - integrity, 12, 256
  - Intelligent Network (IN), 56
  - Interactive Connectivity Establishment (ICE), 9, 234–235
  - Interactive Voice Response (IVR) server, 21, 29
  - Inter-Asterisk Exchange Protocol (IAX), 26
  - interception
    - ARP spoofing, 151–155
    - caller ID spoofing, 155
    - definition of, 148
    - description of, 149–150
  - internal denial-of-service attack, 144, 145
  - Internet, 264–266
  - Internet Assigned Numbers Authority (IANA), 264
  - Internet Engineering Task Force (IETF)
    - RFC 2543/ RFC 3261, 84
    - RTP development, 101
    - S/MIME message authentication, 241–244
    - SRTP, voice/video packet security, 247–251
    - suites from, 240–241
    - TLS, 244–247
  - Internet Relay Chat (IRC), 36
  - Internet Society Architecture Board, 84
  - interoperability, 5
  - intrusion prevention system (IPS), 145, 194–195
  - invalid packet DoS, 147
  - INVITE message, 156
  - IP address
    - ARP address resolution, 149–150
    - ARP spoofing, 151–155
    - DHCP and, 126, 127
    - DNS and, 112
    - DNS client operation, 115–116
    - DNS server operation, 116–117
    - NAT and, 215–225, 264–266
  - IP Centrex, 20, 30
  - IP line, 23–24
  - IP PBX, 12
  - IP protocol, 6
  - IP router, 38
  - IP Security (IPSec), 162, 221–225
  - IP switch, 38
  - IP telephones
    - DHCP security and, 128, 129
    - DoS attack and, 145
    - MAC tools for authentication, 182–183
    - Skinny protocol and, 135–137
    - SNMP access from, 125–126
    - VoIP security model and, 15
    - . *See also* Skype; softphones
  - IP telephony
    - PBX alternatives, 30–31
    - reliability of, 17
    - Skinny protocol and, 135–137
    - use of term, 6
    - . *See also* converged networks; VoIP; VoIP telephony/infrastructure
  - IP-PBX system, 20
  - IPS (intrusion prevention system), 145, 194–195
  - IPSec (IP Security), 162, 221–225
  - IPv6, 264
  - IRC (Internet Relay Chat), 36
  - ISDN
    - description of, 56–57
    - Signaling System 7, 57–60
  - ISDN User Part (ISUP), 58, 66–67
  - ITU standards, 240
  - ITU-T 4, 57
  - ITU-T 5, 57
  - ITU-T signaling systems, 57–60
  - ITU-T standards, 61
  - IVR (Interactive Voice Response) server, 21, 29
- ## J
- Jabber, 37

**K**

Kaminsky, Dan, 195  
 key exchange, TLS, 245–247  
 Key Telephone Systems (KTS), 20, 30  
 keys  
   certification path, 181–182  
   CRL and, 181  
   Public Key Infrastructure, 175–179  
   S/MIME message authentication, 242–244  
   in SRTP, 247–250  
   TLS key exchange, 245–247  
 Kiwi Syslog, 197  
 Korek (hacker), 39

**L**

LAN, 163  
 laptop, 261  
 latency, 229  
 layer 2, 212  
 LEC (Local Exchange Carrier), 47–49, 61  
 Lightweight Extensible Authentication Protocol (LEAP), 170, 172  
 lines, PBX, 22–24  
 links, SS7, 60  
 Local Exchange Carrier (LEC), 47–49, 61  
 location service, SIP, 92  
 logging  
   description of, 186–187  
   NIDSs, 194  
   overview of, 197  
   Skype, 254  
   SNMP, 199–200  
   syslog, 197–198  
 logical separation. *See* network traffic segregation  
 loop distribution plant, 47  
 loop start, 22  
 losses, 3

Lotus Domino IM client, 37  
 Lucent 5ESS switches, 55

**M**

MAC (Message Authentication Code), 246  
 MAC address  
   ARP spoofing, 149–150, 152, 153–154  
   authentication tools, 182–183  
   in DHCP process, 127  
   DHCP security and, 128  
 magnetos, 49  
 maintenance, NIDSs, 194  
 management console (MC), 188–189  
 Management Information Base (MIB), 124, 125  
 Mbone (Multicast Backbone), 85  
 MCUs (multipoint control units), 67, 68  
 MD5. *See* Message Digest 5  
 Media Encryption, Avaya, 154  
 Media Gateway Control Protocol (MGCP), 99, 101  
 media gateways, 33–34  
 media servers, 31–38  
   application proxies, 34  
   call or resource control, 32–33  
   endpoints, 35–38  
   firewalls, application layer gateways, 34  
   interactive media service, 32  
   media gateways, 33–34  
   VoIP, characteristics of, 31–32  
 medium-depth packet inspection, 227–228  
 message authentication  
   of RTP packets, 247  
   S/MIME, 241–244  
   SRTP, 249–250  
 Message Authentication Code (MAC), 246  
 message digest, 172–173  
 Message Digest 5 (MD5)  
   authentication with, 174  
   HIDSs and, 196

- in HTTP digest authentication, 123
- message flow, VoIP, 9–11
- message tampering, 154–155
- Message Transfer Parts (MTP), 57
- Message Types, 130
- messaging sequence, 61–64
- MGCP (Media Gateway Control Protocol), 99, 101
- MIB (Management Information Base), 124, 125
- Microsoft Windows Active Directory, 273–276
- Microsoft Windows Messenger, 36–37, 107
- Microsoft Windows XP, 107
- MIKEY (Multimedia Internet Keying), 248
- mobile phone, 29
- monitoring. *See* security monitoring, active
- MRTG (Multi Router Traffic Grapher), 197
- MS-CHAP, 174
- MS-CHAP v2, 173–174
- MTP (Message Transfer Parts), 57
- Multi Router Traffic Grapher (MRTG), 197
- Multicast Backbone (Mbone), 85
- multimedia, 132
- Multimedia Internet Keying (MIKEY), 248
- multiplexing, NAT, 216, 264
- multipoint control units (MCUs), 67, 68

## N

- name resolution, 112–113
- NAPT (Network Address Port Translation), 220
- NAS. *See* Network Access Server
- NAT. *See* Network Address Translation
- NAT Check
  - link for, function of, 270–271
  - with Skype, 258–259
  - for wireless Skype communication, 277, 281
- network
  - ACLs, 235–237

- DHCP functions for, 126–129
- SNMP for, 123–126
- user authentication, 160
  - . *See also* converged networks
- Network Access Server (NAS)
  - in authentication process, 165, 166–167
  - authenticator, 164
  - in PEAP authentication, 168
- Network Address Port Translation (NAPT), 220
- Network Address Translation (NAT)
  - encryption and, 221–225
  - firewall and, 265–266
  - function of, 264–265
  - operation modes, 218–221
  - overview of, 215–218
  - router for Skype, 267–269, 271
  - as topology shield, 225
- network devices, configuration for Skype, 269–271
- network domains, 145
- network interface card (NIC), 149–150, 188
- Network Intrusion Detection Systems (NIDSs)
  - components, 188–189
  - description of, 186
  - features of, 194–195
  - honeypots/honeynets, 195
  - limitations of, 195
  - overview of, 187–188, 205
  - placement of, 191–194
  - types of, 189–191
- Network layer, OSI, 85–86
- network links, SS7, 60
- Network Management Systems (NMSes), 124–125
- network security. *See* security monitoring, active
- network sniffing, 136
- network traffic segregation
  - ACLs, 235–237
  - firewalls, 225–235

- NAT/IP addressing, 215–225
    - overview of, 208–209, 237–238
    - QoS/traffic shaping, 214–215
    - VLANs, 209–213
  - Network World*, 11
  - NIC (network interface card), 149–150, 188
  - NIDSs. *See* Network Intrusion Detection Systems
  - NMSes (Network Management Systems), 124–125
  - Nortel DMS-100 switches, 55
  - Northern Telecom, 55
- O**
- object identifiers (OIDs), 125
  - Oechslin, Philippe, 172
  - one-way hash function, 174
  - Open Systems Interconnect (OSI) model
    - SIP and, 85–87
    - SS7 and, 57, 59
  - operation modes, NAT, 218–221
  - outer authentication, EAP, 168
- P**
- P2P. *See* peer-to-peer (P2P) architecture
  - Packed Encoding Rules (PER), 72
  - packet filters, 282–283
  - packet headers, 190
  - packet injection, VoIP, 146
  - packet loss, 214
  - packet of death DoS, 148
  - packets
    - DoS attack and, 142–148
    - packet replay attack, 146
    - RSVP operation, 129–131
    - SDP, 132–134
    - SRTP for packet security, 247–251
  - PAP (Password Authentication Protocol), 174
  - password
    - community string for SNMP, 125
    - dictionary attacks, 175
    - inner authentication types and, 173–175
    - RainbowCrack, 172–173
    - for voice messaging, 28, 29
    - WPA2 and, 40
  - Password Authentication Protocol (PAP), 174
  - PAT (Port Address Translation), 220
  - path, 130–131
  - Path message, 130, 131
  - PathTear packet, 131
  - pattern matching, 188
  - payload, 130
  - PayMaxx Inc., 3
  - PBX (private branch exchange), 21–30
    - adjunct servers, 28–29
    - alternatives to, 30–31
    - description of, 21–22
    - features of, 25–27
    - function of, 20
    - IP telephony adoption and, 6
    - lines, 22–24
    - responsibilities of, 11–12
    - toll fraud, 27
    - trunks, 24–25
    - wireless, 30
  - PC. *See* computer (PC)
  - PCM (Pulse Code Modulation), 49
  - PEAP, 168
  - PEAPv1/EAP-GTC (Extensible Authentication Protocol–Generic Token Card), 170, 171
  - peering arrangements, 64–66
  - peer-to-peer (P2P) architecture
    - SIP, 94
    - SIP communication, 102, 104–105
  - penetration testing
    - description of, 187
    - methodology, 201–205

- overview of, 200–201, 206
- PER (Packed Encoding Rules), 72
- personal data, 2–3
- Physical layer, OSI, 85, 87
- PINX (Private Integrated service Network Exchange), 56
- PISNs (Private Integrated Services Networks), 56
- PKI. *See* Public Key Infrastructure
- POE (Power-over-Ethernet), 41–42
- Ponemon, Larry, 2
- Ponemon Institute, 2
- Port Address Translation (PAT), 220
- port mirroring, 190
- ports
  - blocking Skype, 282
  - H.323, 70, 75–76
  - MAC address authentication, 183
  - for Skype, 271–276
  - in Skype communication, 268
- power loss, 43
- power surges, 42
- Power-over-Ethernet (POE), 41–42
- power-supply infrastructure, 41–43
- presence, 7, 107
- Presentation layer, OSI, 85–86
- privacy
  - regulations, 16
  - Skype and, 255, 256
- private address, 264–266
- private branch exchange. *See* PBX
- Private Integrated service Network Exchange (PINX), 56
- Private Integrated Services Networks (PISNs), 56
- private key, 176–178, 181
- private trunks, 24
- process, 16
- profiles, H.235 security, 79–82
- protocols
  - 802.1x, 40–41
  - application proxies, 34
  - firewall, application layer gateway and, 34
  - H.323 VoIP-related protocols, 68–77
  - IETF suites, 240–241
  - IPsec, NAT and, 221–225
  - media gateways and, 33–34
  - SIP, 97–102
  - VoIP, 5, 8–9
  - with VoIP trunks, 25
  - WEP, 38–39
  - WPA2, 39–40
  - See also* support protocols
- proxies, application, 34
- proxy impersonation, 154
- proxy servers
  - NAT and, 265–266
  - SIP, function of, 91
  - SIP requests through, 103, 104
  - Skype and, 260, 276–277
- PSTN. *See* Public Switched Telephone Network
- public key
  - certification path, 181–182
  - cryptography concepts, 176–178
- Public Key Infrastructure (PKI)
  - authentication, 175–178, 184
  - entities, architectural model, 178–179
  - public key cryptography concepts, 176–178
  - S/MIME message authentication, 242–244
  - for subject identification, 175
- Public Switched Telephone Network (PSTN), 46–61
  - architectures, 109–110
  - call flow, 61–64
  - cost savings, 20
  - description of, 46–49
  - media gateways and, 33
  - PBX functionality and, 21
  - protocol security, 64–67
  - regulations, 61
  - security of, 4–5



signal transmission, 49–54  
 switching/signaling, 54–60  
 Pulse Code Modulation (PCM), 49

## Q

Q.931, 71–75

QSIG

function of, 56  
 Q.931 *vs.*, 71–72  
 security, 66–67

Quality of Service (QoS)

modification attack, 146  
 RSVP for, 129  
 traffic shaping and, 214–215

## R

RA (Registration Authority), 179

RADIUS (Remote Authentication Dial In  
 User Service), 165–166

Rager, Anton, 163

RainbowCrack, 172–173

RAS (Registration, Admission, Status),  
 72–73

RC4 stream cipher, 164

RCF/RRJ (Registration Confirm or  
 Reject), 74

Read Request (RRQ) packet, 119, 120

Real Time Control Protocol (RTCP), 8, 69

Real Time Protocol (RTP), 69

Real-Time Streaming Protocol (RTSP),  
 102

Real-Time Transport Control Protocol  
 (RTCP), 79

Real-Time Transport Protocol (RTP)  
 function of, 77

H.235 security mechanisms, 79

SIP and, 101

SRTP for packet security, 241, 247–251

voice media transport with, 8

receiving agent, 243

recursive lookup, 115–116

redirect server

function of, 33

SIP, function of, 91–92

SIP requests through, 103–104

redirection, ARP, 152

Reeves, Alec, 49

refresh interval, 117

REGISTER request, 92

Registrar server

function of, 91

location service, 92

SIP registration, 102–103

Registration, Admission, Status (RAS),  
 72–73

registration, SIP, 102–103

Registration Authority (RA), 179

Registration Confirm or Reject  
 (RCF/RRJ), 74

registration hijacking, 154

Registration Request (RRQ), 74

registration servers, 33

regulations

privacy, 16

of PSTN, 61

relay call, 257, 268

Remote Authentication Dial In User  
 Service (RADIUS), 165–166

replay protection, 247, 250–251

reporting, 203–204

repository, 179

Request for Comments (RFCs)  
 1631, 218

1918, 225

2543/3261, 84

2705, 101

3261, 92, 161–162

3280, 175

requests, SIP, 94–95, 103–105

resilience, of Skype, 256

resolver, 115–116

resource control, 32  
Resource ReSerVation Protocol (RSVP)  
  description of, 129–130  
  operation, 130–131  
  overview of, 138–139  
  security implications for, 131–132  
  VoIP-related protocol, 9  
response  
  NIDSs, 194–195  
  SIP, 94–97  
  SIP requests/responses, 103–105  
Resv message, 130–131  
RFCs. *See* Request for Comments  
rogue VoIP endpoint attack, 154  
Rollover Counter (ROC), 250  
root DNS servers, 113  
router  
  IP, 38  
  NAT, 264–266  
  NAT router for Skype, 270–271  
  network device configuration for Skype,  
    269  
  RSVP operation, 130–131  
  for Skype communication, 267–269  
  for Skype home users, 266  
RRQ (Read Request) packet, 119, 120  
RRQ (Registration Request), 74  
RRs, 116  
RSVP. *See* Resource ReSerVation Protocol  
RTCP (Real Time Control Protocol), 8, 69  
RTCP (Real-Time Transport Control  
  Protocol), 79  
RTP. *See* Real-Time Transport Protocol  
RTP (Real Time Protocol), 69  
rtpsniff, 153  
RTSP (Real Time Streaming Protocol), 9

## S

SBCs (Session Border Controllers), 233  
scanning, 202–203  
SCCP. *See* Skinny Client Control Protocol

SCCP (Signaling Connection Control Part),  
  58  
Schwartau, Winn, 11  
SCP (Service Control Point), 59  
SCTP (Stream Control Transmission  
  Protocol), 8  
SDP. *See* Session Description Protocol  
SDP (Session Discovery Protocol), 9,  
  132–134  
Secure Networks, 195  
Secure Real-Time Transfer Protocol  
  (SRTP)  
  confidentiality, 248–249  
  message authentication, 249–250  
  Multimedia Internet Keying, 248  
  for packet security, 247–248  
  replay protection, 250–251  
  SDP Security Descriptions, 248  
  VoIP security protocol, 240–241  
Secure SCCP (Secure Skinny), 136  
Secure Sockets Layer (SSL)  
  handshake for certificate/key exchange,  
    245–247  
  TLS and, 99  
  TLS as SIP transport layer, 244–245  
Secure/Multipurpose Internet Mail  
  Extensions (S/MIME), 162, 240–244  
security  
  in converged networks, 11–15  
  H.235 security mechanisms, 78–82  
  IETF encryption solutions, 240–251  
  instant messaging and, 107–108  
  of PBX line, 22  
  of PBX system, 26–27  
  PSTN protocol, 64–67  
  VLAN, 212  
  of voice messaging, 28–29  
  VoIP, 17–18  
  VoIP problems, 5–6  
  VoIP process, 16–17  
  *See also* Skype security; threats, VoIP  
security breach, 2–3

- Security Descriptions, SDP, 248
- security implications
  - for DHCP, 128–129
  - for DNS, 117
  - for HTTP, 122–123
  - for RSVP, 131–132
  - for SDP, 134–135
  - for Skinny, 136–137
  - for SNMP, 125–126
  - for TFTP, 119–120
- security model, new, 15–16
- security monitoring, active
  - HIDSs, 196
  - logging, 197–200
  - NIDSs, 187–195
  - overview of, 186–187
  - penetration/vulnerability testing, 200–205
- security profiles, H.235, 79–82
- security protocols, VoIP, 240–241
- Seisint (Lexis-Nexis research), 2
- self-signed certificates, 182
- Selsius Corporation, 135
- sender agent, 243
- sensors, NIDS, 188
- serial number, 117
- servers
  - DHCP server, 126–129
  - DNS server operations, 116–117
  - HTTP server, 121–123
  - PBX adjunct servers, 28–29
  - SIP, 90, 91–92
  - See also* media servers
- Service Control Point (SCP), 59
- Service Switching Points (SSPs), 59
- Service Transport Point (STP), 59
- Session Border Controllers (SBCs), 233
- Session Description Protocol (SDP)
  - description of, 132
  - operation, 133–134
  - overview of, 139
  - Security Descriptions, 248
  - security implications for, 134–135
  - SIP and, 99–100
  - specifications, 132–133
- Session Discovery Protocol (SDP), 9, 132–134
- session ID
  - RSVP security and, 131–132
  - SDP security and, 134–135
- Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions (SIMPLE)
  - description of, 107
  - function of, 82, 106
  - need for, 106–107
  - use of, 110
- Session Initiation Protocol (SIP), 82–105
  - architecture of, 102–105
  - Asterisk's use of, 26
  - authentication, 161–162
  - client/server *vs.* peer-to-peer architecture, 93–94
  - components, 90–92
  - description of, 8
  - design of, 82–83
  - firewall issues, 231–232
  - functions, features of, 87–89
  - IP line's use of, 23–24
  - NAT and, 221
  - overview of, 83–87, 110
  - protocols used with, 97–102
  - redirect server, 33
  - requests/responses, 94–97, 133
  - security vulnerabilities of, 158
  - servers, 90, 91–92, 93–94
  - SIP layer, TLS and, 244–245
  - SIP to PSTN call flow, 62–64
  - SIP-specific attacks, 156
  - S/MIME and, 241–244
  - VoIP security issue, 5
  - VoIP security protocol, 240–241
  - VoIP security protocols and, 240
- session key, 242–243

- Session layer, OSI, 85–86
- session management, 88, 89
- session setup, 88, 89
- SETREQUEST, 125
- shallow packet inspection, 226–227
- shared.xml file, 273
- signal transmission, PSTN, 49–54
- signaling
  - H.245 call control messages, 75–77
  - H.323 call signaling, 70–75
  - PSTN, 55–60
  - separation of data/signaling, 9
  - SIP requests/responses, 94–97
  - SS7, other ITU-T signaling security, 64–67
- Signaling Connection Control Part (SCCP), 58
- signaling points, 59–60
- signaling protocols
  - H.323/SIP, 247
  - VoIP, 8–9
- Signaling System 7 (SS7)
  - as CCS, 57
  - hacking, 5
  - ISDN User Part, 58
  - MTP/TUP, 57
  - network links, 60
  - SCCP, 58
  - signaling security, 64–66
  - TCAP, 59–60
- signature, digital
  - in public key cryptography, 177–178
  - in S/MIME message authentication, 242–243
- signature algorithm, 180
- signature matching, 188
- signature value field, 180
- signature-based NIDSs, 188
- signatures, attack
  - definition of, 188
  - fields used by, 189–190
  - NIDSs detection of, 187–188
- SIGTRAN, 8
- SIMPLE. *See* Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions
- Simple Network Management Protocol (SNMP)
  - architecture, 124, 125
  - description of, 123
  - for logging events, 199–200, 205
  - operation, 124–125
  - overview of, 138
  - security of, 125–126
- SIP. *See* Session Initiation Protocol
- Skinny Client Control Protocol (SCCP)
  - operation, 135–136
  - overview of, 139
  - security implications for, 136–137
  - specifications, 135
- Skype
  - chat feature, 106
  - description of, 35
  - PSTN and, 46
  - voice chat client, 36
- Skype firewall/network setup
  - blocking Skype, 282–283
  - firewalls, 265–266
  - home users, 266
  - large corporations, 267–269
  - NAT, 264–265
  - network devices, configuration of, 269–271
  - ports required, 271–276
  - proxy servers, 276–277
  - small to medium-sized businesses, 266
  - wireless communications, 277–282
- Skype security
  - blocking Skype, 257–258
  - embedded Skype, 260
  - firewalls, 258–259
  - issues, 260–262
  - overview of, 254–257
  - proxy servers, 260

- Skype Video, 37
- Skype's Technical Call Information, 277–282
- sliding window, 250–251
- S/MIME (Secure/Multipurpose Internet Mail Extensions), 162, 240–244
- sniffing
  - ARP spoofing, 151–155
  - Skype call, 261
  - VoIP, 13
- SNMP. *See* Simple Network Management Protocol
- softphones
  - description of, 35
  - PBX attacks and, 27
  - private information and, 36
  - VLANs and, 212–213
  - . *See also* IP telephones; Skype
- softswitch, 32
- software
  - immature software DoS, 147
  - inventory to block Skype, 258
- SONET, 51–52
- SONET ring, 47–49
- spanning, 190
- specification protocol, SDP, 132–135
- Spencer, Mark, 26
- SPKI (Subject Public Key Info) field, 180
- spoofing
  - ANI spoofing services, 58
  - ARP, 151–155, 183
  - DNS poisoning, 148–149
- SPS (Cisco SIP Proxy Server), 156, 232
- spyware, 108
- SRTP. *See* Secure Real-Time Transfer Protocol
- SRV records, 148–149
- SS7. *See* Signaling System 7
- SSL. *See* Secure Sockets Layer
- SSPs (Service Switching Points), 59
- stacks, 59
- standards, 5
- stateful inspection firewall, 227
- stateful mode, 92
- stateful pattern matching, 188
- stateless mode, 92
- static NAT, 218–219
- Step by Step (SXS) system, 55
- STP (Service Transport Point), 59
- Stream Control Transmission Protocol (SCTP), 8
- Strowger, Almon, 55
- STUN protocol
  - for enabling SIP with NATs, 234–235
  - NAT and, 220
  - Skype's use of, 258
  - VoIP-related protocol, 9
- Subject Public Key Info (SPKI) field, 180
- supernode
  - security risks, 256
  - in Skype communication, 267, 268
  - Skype computer as, 255
- supplementary services, 146–147
- supplicant (peer), 164
- supplication, 166
- support protocols
  - Domain Name System, 112–117
  - Dynamic Host Configuration Protocol, 126–129
  - HyperText Transfer Protocol, 120–123
  - overview of, 138–139
  - Resource ReSerVation Protocol, 129–132
  - Session Description Protocol, 132–135
  - Simple Network Management Protocol, 123–126
  - Skinny, 135–137
  - Trivial File Transfer Protocol, 118–120
- switches
  - DSOs and, 52
  - history of, 55–56
  - IN, ISDN, QSIG, 56–57
  - IP switch, function of, 38
  - PBX trunks and, 24–25
  - See also* PBX

- switching
    - PBX system and, 21–22
    - PSTN, 55–60
    - PSTN signal transmission, 49
  - SXS (Step by Step) system, 55
  - symmetric key cryptography, 176
  - syslog, 197–198
- ## T
- T1 line, 55
  - T1 transmission, 49–54
  - T1 trunks, 24–25
  - TALK command, 106
  - taps, 4, 190
  - TCAP (Transaction Capabilities Applications Part), 59–60
  - TCO session, 121
  - TCP. *See* Transmission Control Protocol
  - TDM (Time Division Multiplexing), 50
  - teardown, 131
  - telephone exchange, 46–49
  - telephone networks, 4–5
  - Telephone User Part (TUP), 57
  - telephones. *See* IP telephones; softphones
  - telephony equipment. *See* hardware infrastructure
  - Temporal Key Integrity Protocol (TKIP), 163
  - testing, 200–205
  - text messages, 105–108
  - TFTP. *See* Trivial File Transfer Protocol
  - Threat Index, 204–205
  - threats, VoIP, 14–15
    - call hijacking/interception, 148–155
    - DNS poisoning, 148–149
    - DoS or VoIP service disruption, 142–148
    - H.323-specific attacks, 155–156
    - overview of, 157–158
    - SIP-specific attacks, 156
    - VoIP vulnerabilities, 142
  - 3GPP SA WG3 Technical Specification Group, 65–66
  - Time Division Multiplexing (TDM), 50
  - timing, 51–52
  - TKIP (Temporal Key Integrity Protocol), 163
  - TLD DNS server
    - client operation, 115–116
    - function of, 113
  - TLS. *See* Transport Layer Security
  - TLS Handshake Protocol, 99
  - TLS Record Protocol, 99
  - T-Mobile, 3
  - To Be Signed (TBS) certificate field, 180
  - toll bypass, 61
  - toll fraud
    - from PBX system, 27
    - reasons for, 5
    - voice mail systems and, 29
    - as VoIP threat, 154
  - tools
    - for network monitoring, 186–187
    - for vulnerability scanning, 206
  - topology shield, 225
  - Touch Tones, 55
  - traffic, 187
    - . *See also* network traffic segregation
  - traffic shaping, 214–215
  - Transaction Capabilities Applications Part (TCAP), 59–60
  - transistor, 49
  - Transmission Control Protocol (TCP)
    - H.323 call signaling, 72–73
    - H.323 signaling and, 70
    - ports for Skype, 271–272
    - Skype communication over, 269–270
  - transparency, 244–245
  - Transport layer, OSI, 85–86
  - Transport Layer Security (TLS)
    - connection reset, 146
    - goal of, layers of, 167–168
    - IETF encryption solutions, 244–247

- key exchange, signaling packet security, 244–247
  - SIP and, 98–99, 162
  - VoIP security protocol, 9, 240–241
  - transport mode, 223–224
  - Traversal Using Relay NAT (TURN), 9, 234–235
  - Tripwire, 196
  - Trivial File Transfer Protocol (TFTP)
    - description of, 118–119
    - DHCP security and, 129
    - file transfer operation, 119
    - overview of, 138
    - security implications for, 119–120
  - trunks, 24–25, 27
  - trust
    - authentication in H.323 environment, 161
    - of authorized users, 17
    - in VoIP security model, 15–16
    - . *See also* authentication
  - Tschofenig, Hanes, 132
  - tunnel mode, 223–224
  - TUP (Telephone User Part), 57
  - TURN (Traversal Using Relay NAT), 9, 234–235
  - 200 code, 122
- ## U
- UDP. *See* User Datagram Protocol
  - Uninterruptible Power Supply (UPS), 42–43
  - Universal Resource Identifiers (URIs), 83–84, 89
  - UPS (Uninterruptible Power Supply), 42–43
  - URIs (Universal Resource Identifiers), 83–84, 89
  - User Agent Client (UAC)
    - function of, 90
    - SIP architecture and, 93–94
    - SIP communication, 105
  - User Agent Server (UAS)
    - function of, 90
    - SIP architecture and, 93–94
    - SIP communication, 105
  - user agents
    - IM clients, 36–37
    - peer-to-peer architecture, 104–105
    - requests and, 103–104
    - SIP, 90–91
    - SIP architecture and, 93–94
    - SIP registration, 102–103
    - SIP servers and, 91–92
    - softphones, 35–36
    - video clients, 37
    - wireless VoIP clients, 38
  - user authentication, 160
  - user availability, 87, 88
  - user capabilities, 87, 88
  - User Datagram Protocol (UDP)
    - H.323 signaling and, 70
    - network device configuration for Skype, 269–270
    - ports for Skype, 271–273
    - SIP and, 97–98
    - wireless Skype communication, 281
  - User Datagram Protocol (UDP) packets
    - in DHCP, 127
    - RSVP transfer of, 130
    - Skinny protocol and, 135–136
    - TFTP and, 118–119
  - user identity, confirmation of
    - 802.1x/802.11i (WPA2), 163–164
    - 802.1x/EAP authentication, 164–167
    - authentication types, 160
    - basic certificate fields, 180
    - certificate revocation list, 181
    - certification path, 181–182
    - dictionary attacks, 175
    - EAP authentication types, 167–175
    - in H.323 environments, 161

- MAC tools, 182–183
- model authentication scheme, 161
- overview of, 183–184
- PKI entities, architectural model, 178–179
- point solutions, 160–161
- Public Key Infrastructure, 175–178
- Session Initiation Protocol, 161–162
- user location, 87, 88
- users
  - IP telephony outage and, 11, 17
  - passwords of, 40
- UTstarcom F1000 IP phone, 126

## V

- Verso, 257
- video clients, 37
- Virtual Local Area Networks (VLANs)
  - security, 212
  - softphones, 212–213
  - voice/data traffic separation via, 209–212
- Virtual Private Networks (VPNs), 221
- viruses
  - instant messaging and, 108
  - VoIP DoS attack, 144
  - VoIP threats, 13, 17, 157
- VLANs. *See* Virtual Local Area Networks
- voice communications, 10
- voice encryption profile, 81–82
- voice firewall, 30
- voice gateway, 33–34
- voice messaging, 28–29
- Voice over Misconfigured Internet Telephones (vomit), 13, 153
- voice/data convergence, 6–7
- voice/video media, 241, 247–251
- VoiceXML interpreters, 32
- VoIP
  - benefits of, 6–7
  - characteristics of, 9–11
  - information security breach, 2–3
  - practice of information security, 3–4
  - protocols, 8–9
  - PSTN and, 46
  - securing, 17–18
  - security issues in converged networks, 11–15
  - security model, new, 15–16
  - security problems, 5–6
  - security process, 16–17
  - telephone to VoIP switch, 4–5
  - use of term, 6
  - See also* converged networks; IP telephony; support protocols
- VoIP, threats to
  - call hijacking/interception, 148–155
  - DNS poisoning, 148–149
  - DoS or VoIP service disruption, 142–148
  - H.323-specific attacks, 155–156
  - overview of, 157–158
  - SIP-specific attacks, 156
  - VoIP vulnerabilities, 142
- VoIP protocol implementation attacks, 147, 155
- VoIP telephony/infrastructure, 31–44
  - as hybrid, 44
  - IP switches/routers, 38
  - media servers, 31–38
  - power-supply infrastructure, 41–43
  - wireless infrastructure, 38–41
- VoIP trunks, 25
- VoIP-aware firewalls, 229–231
- VoipCrack, 153
- VOIPong, 13
- VOIPSA, 14, 148
- vomit (Voice over Misconfigured Internet Telephones), 13, 153
- VoWLAN working group, 163
- VPNs (Virtual Private Networks), 221
- vulnerabilities
  - of Cisco CallManager, 137
  - of VoIP, 5–6, 142
  - . *See also* threats, VoIP
- vulnerability testing. *See* penetration testing



**W**

Wavelength Division Multiplexing (WDM), 51

WEP (Wired Equivalent Privacy), 38–39

WEP (Wireless Equivalent Privacy), 163

wepcrack, 163

Western Electric, 55

white listing, 282

Wi-Fi Alliance, 163

Wi-Fi Protected Access 2 (WPA2), 163–164

Wi-Fi Protected Access (WPA), 39–40, 163–164

WiFi VoIP, 38

wire, 23, 46–49

wire tapping, 4

Wired Equivalent Privacy (WEP), 38–39

wireless, PBX systems, 30

wireless communications  
with Skype, 277–278  
Skype's Technical Call Information, 278–282

wireless DoS, 147

wireless encryption  
WEP, 38–39  
WPA2, 39–40

Wireless Equivalent Privacy (WEP), 163

wireless infrastructure, 38–41

802.1x, 40–41

WEP, 38–39

WPA2, 39–40

wireless PBX solutions, 30

wireless VoIP clients, 38

worms  
attacks on VoIP devices, 145  
VoIP DoS attack, 144  
VoIP threats, 13, 17, 157

WPA (Wi-Fi Protected Access), 39–40, 163–164

WPA2 (Wi-Fi Protected Access 2), 163–164

Wright, Joshua, 40

Write Request (WRQ) packet, 119, 120

**X**

X.509 certificate, 180, 244

X-Lite, 36

**Y**

Yahoo! Chat, 36

**Z**

zone transfer, 116–117