

# 100%

ONE HUNDRED PERCENT

COMPREHENSIVE  
AUTHORITATIVE  
WHAT YOU NEED

ONE HUNDRED PERCENT

**Defend** corporate  
systems and wireless  
networks against  
intrusion

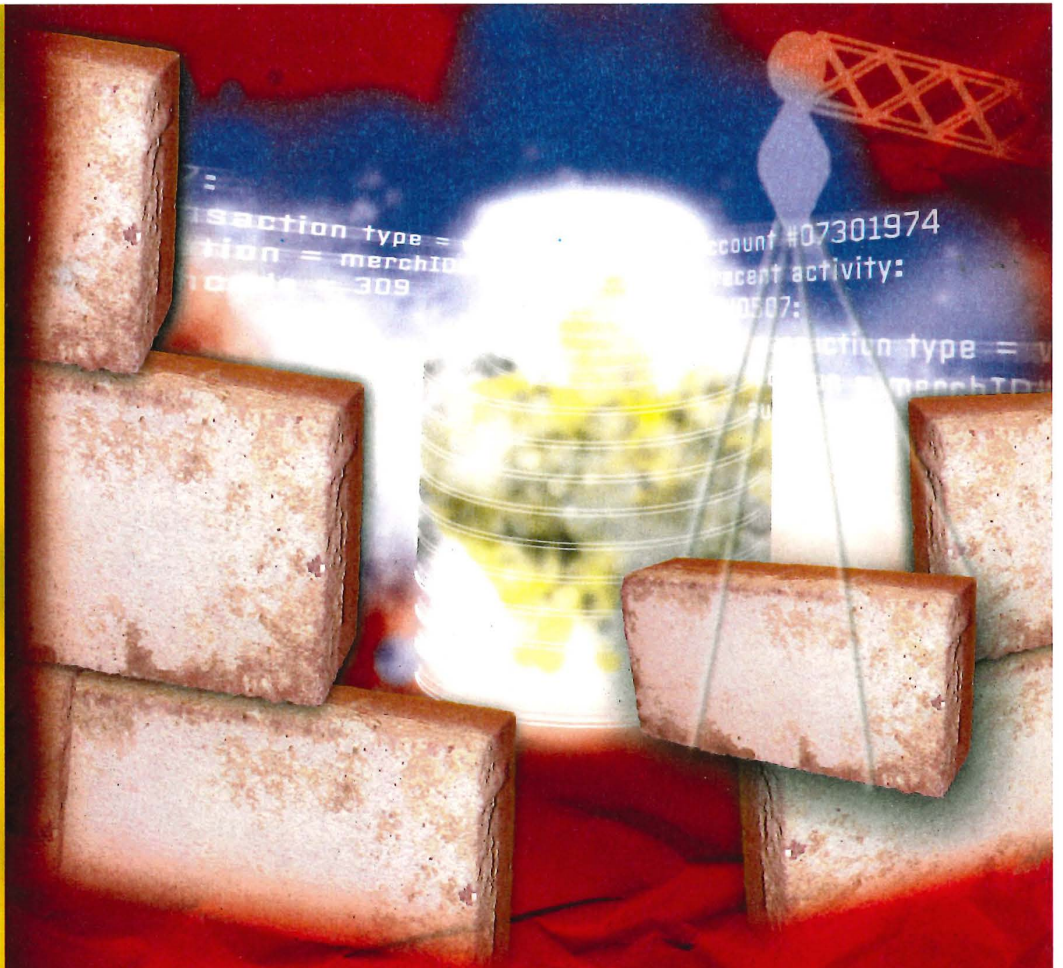
**Apply** the latest  
techniques for pro-  
tecting information,  
testing your system,  
and responding to  
attacks

**Learn** from  
recognized experts  
in government,  
media, academic,  
and corporate  
security

# Network Security Bible

The first comprehensive,  
A-to-Z guide to network  
security, by nationally  
acclaimed authorities in the field

Eric Cole, PhD, CISSP; Ronald Krutz, PhD, PE, CISSP, ISSEP;  
and James W. Conley, CISSP, Security+, CCNA



# **Network Security Bible**

**Dr. Eric Cole, Dr. Ronald Krutz, and James W. Conley**



**WILEY**

**Wiley Publishing, Inc.**

## **Network Security Bible**

Published by  
**Wiley Publishing, Inc.**  
10475 Crosspoint Boulevard  
Indianapolis, IN 46256  
www.wiley.com

Copyright © 2005 by Wiley Publishing, Inc., Indianapolis, Indiana

Published simultaneously in Canada

ISBN: 0-7645-7397-7

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

1B/SZ/RS/QU/IN

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. Requests to the Publisher for permission should be addressed to the Legal Department, Wiley Publishing, Inc., 10475 Crosspoint Blvd., Indianapolis, IN 46256, (317) 572-3447, fax (317) 572-4355, E-Mail: brandreview@wiley.com.

**LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.**

For general information on our other products and services or to obtain technical support, please contact our Customer Care Department within the U.S. at (800) 762-2974, outside the U.S. at (317) 572-3993 or fax (317) 572-4002.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

### **Library of Congress Cataloging-in-Publication Data**

Cole, Eric.

Network security bible / Eric Cole, Ronald Krutz, James W. Conley.

p. cm.

ISBN 0-7645-7397-7 (pbk.)

1. Computer security. 2. Computer networks — Security measures. I. Krutz, Ronald L., 1938- II. Conley, James W. III. Title.

QA76.9.A25C5985 2005

005.8—dc22

2004025696

**Trademarks:** Wiley, the Wiley logo, and related trade dress are registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. All other trademarks are the property of their respective owners. Wiley Publishing, Inc., is not associated with any product or vendor mentioned in this book.

# Contents at a Glance

.....

|                       |       |
|-----------------------|-------|
| Acknowledgments ..... | xxv   |
| Introduction .....    | xxvii |

## **Part I: Security Principles and Practices ..... 1**

|                                                         |    |
|---------------------------------------------------------|----|
| Chapter 1: Information System Security Principles ..... | 3  |
| Chapter 2: Information System Security Management ..... | 43 |
| Chapter 3: Access Control Considerations .....          | 79 |

## **Part II: Operating Systems and Applications ..... 97**

|                                                  |     |
|--------------------------------------------------|-----|
| Chapter 4: Windows Security .....                | 99  |
| Chapter 5: UNIX and Linux Security .....         | 155 |
| Chapter 6: Web Browser and Client Security ..... | 201 |
| Chapter 7: Web Security .....                    | 237 |
| Chapter 8: E-mail Security .....                 | 273 |
| Chapter 9: Domain Name System .....              | 309 |
| Chapter 10: Server Security .....                | 333 |

## **Part III: Network Security Fundamentals ..... 365**

|                                                     |     |
|-----------------------------------------------------|-----|
| Chapter 11: Network Protocols .....                 | 367 |
| Chapter 12: Wireless Security .....                 | 381 |
| Chapter 13: Network Architecture Fundamentals ..... | 417 |

## **Part IV: Communications ..... 445**

|                                                               |     |
|---------------------------------------------------------------|-----|
| Chapter 14: Secret Communication .....                        | 447 |
| Chapter 15: Covert Communication .....                        | 479 |
| Chapter 16: Applications of Secure/Covert Communication ..... | 529 |

## **Part V: The Security Threat and the Response ..... 555**

|                                                                 |     |
|-----------------------------------------------------------------|-----|
| Chapter 17: Intrusion Detection and Response .....              | 557 |
| Chapter 18: Security Assessments, Testing, and Evaluation ..... | 591 |
| Chapter 19: Putting Everything Together .....                   | 613 |

|             |     |
|-------------|-----|
| Index ..... | 625 |
|-------------|-----|

# Contents

|                           |       |
|---------------------------|-------|
| Acknowledgments . . . . . | xxv   |
| Introduction . . . . .    | xxvii |

## Part I: Security Principles and Practices 1

### Chapter 1: Information System Security Principles . . . . . 3

|                                                                |    |
|----------------------------------------------------------------|----|
| Key Principles of Network Security . . . . .                   | 3  |
| Confidentiality . . . . .                                      | 4  |
| Integrity . . . . .                                            | 4  |
| Availability . . . . .                                         | 4  |
| Other important terms . . . . .                                | 4  |
| Formal Processes . . . . .                                     | 5  |
| The systems engineering process . . . . .                      | 5  |
| The Information Assurance Technical Framework . . . . .        | 6  |
| The Information Systems Security Engineering process . . . . . | 11 |
| The Systems Development Life Cycle . . . . .                   | 21 |
| Information systems security and the SDLC . . . . .            | 22 |
| Risk Management . . . . .                                      | 31 |
| Definitions . . . . .                                          | 32 |
| Risk management and the SDLC . . . . .                         | 33 |
| Summary . . . . .                                              | 42 |

### Chapter 2: Information System Security Management . . . . . 43

|                                                            |    |
|------------------------------------------------------------|----|
| Security Policies . . . . .                                | 43 |
| Senior management policy statement . . . . .               | 44 |
| Standards, guidelines, procedures, and baselines . . . . . | 45 |
| Security Awareness . . . . .                               | 46 |
| Training . . . . .                                         | 46 |
| Measuring awareness . . . . .                              | 47 |
| Managing the Technical Effort . . . . .                    | 48 |
| Program manager . . . . .                                  | 48 |
| Program management plan . . . . .                          | 48 |
| Systems engineering management plan . . . . .              | 48 |
| Configuration Management . . . . .                         | 56 |
| Primary functions of configuration management . . . . .    | 56 |
| Definitions and procedures . . . . .                       | 57 |

|                                                              |           |
|--------------------------------------------------------------|-----------|
| Business Continuity and Disaster Recovery Planning . . . . . | 59        |
| Business continuity planning . . . . .                       | 60        |
| Disaster recovery planning . . . . .                         | 64        |
| Physical Security . . . . .                                  | 67        |
| Controls . . . . .                                           | 68        |
| Environmental issues . . . . .                               | 72        |
| Fire suppression . . . . .                                   | 73        |
| Object reuse and data remanence . . . . .                    | 74        |
| Legal and Liability Issues . . . . .                         | 75        |
| Types of computer crime . . . . .                            | 75        |
| Electronic monitoring . . . . .                              | 76        |
| Liability . . . . .                                          | 76        |
| Summary . . . . .                                            | 77        |
| <b>Chapter 3: Access Control Considerations . . . . .</b>    | <b>79</b> |
| Control Models . . . . .                                     | 79        |
| Discretionary access control . . . . .                       | 79        |
| Mandatory access control . . . . .                           | 80        |
| Non-discretionary access control . . . . .                   | 81        |
| Types of Access Control Implementations . . . . .            | 81        |
| Preventive/Administrative . . . . .                          | 81        |
| Preventive/Technical . . . . .                               | 82        |
| Preventive/Physical . . . . .                                | 82        |
| Detective/Administrative . . . . .                           | 82        |
| Detective/Technical . . . . .                                | 83        |
| Detective/Physical . . . . .                                 | 83        |
| Centralized/Decentralized access controls . . . . .          | 84        |
| Identification and Authentication . . . . .                  | 84        |
| Passwords . . . . .                                          | 85        |
| Biometrics . . . . .                                         | 85        |
| Single Sign-On . . . . .                                     | 86        |
| Databases . . . . .                                          | 90        |
| Relational databases . . . . .                               | 90        |
| Other database types . . . . .                               | 92        |
| Remote Access . . . . .                                      | 93        |
| RADIUS . . . . .                                             | 93        |
| TACACS and TACACS+ . . . . .                                 | 93        |
| Password Authentication Protocol . . . . .                   | 94        |
| Challenge Handshake Authentication Protocol . . . . .        | 94        |
| Callback . . . . .                                           | 95        |
| Summary . . . . .                                            | 95        |

## Part II: Operating Systems and Applications 97

### Chapter 4: Windows Security . . . . . 99

|                                                              |     |
|--------------------------------------------------------------|-----|
| Windows Security at the Heart of the Defense . . . . .       | 101 |
| Who would target me? . . . . .                               | 101 |
| Be afraid . . . . .                                          | 102 |
| Microsoft recommendations . . . . .                          | 103 |
| Out-of-the-Box Operating System Hardening . . . . .          | 105 |
| Prior to system hardening . . . . .                          | 105 |
| The general process of system hardening . . . . .            | 105 |
| Windows 2003 new installation example . . . . .              | 107 |
| Specifics of system hardening . . . . .                      | 110 |
| Securing the typical Windows business workstation . . . . .  | 114 |
| Securing the typical Windows gaming system . . . . .         | 114 |
| Installing Applications . . . . .                            | 115 |
| Antivirus protection . . . . .                               | 116 |
| Personal firewalls . . . . .                                 | 118 |
| Secure Shell . . . . .                                       | 118 |
| Secure FTP . . . . .                                         | 119 |
| Pretty Good Privacy . . . . .                                | 119 |
| Putting the Workstation on the Network . . . . .             | 120 |
| Test the hardened workstation . . . . .                      | 120 |
| Physical security . . . . .                                  | 120 |
| Architecture . . . . .                                       | 120 |
| Firewall . . . . .                                           | 121 |
| Intrusion detection systems . . . . .                        | 122 |
| Operating Windows Safely . . . . .                           | 122 |
| Separate risky behavior . . . . .                            | 122 |
| Physical security issues . . . . .                           | 124 |
| Configuration issues . . . . .                               | 125 |
| Configuration control . . . . .                              | 127 |
| Operating issues . . . . .                                   | 130 |
| Upgrades and Patches . . . . .                               | 138 |
| Keep current with Microsoft upgrades and patches . . . . .   | 138 |
| Keep current with application upgrades and patches . . . . . | 139 |
| Keep current with antivirus signatures . . . . .             | 139 |
| Use the most modern Windows version . . . . .                | 140 |
| Maintain and Test the Security . . . . .                     | 140 |
| Scan for vulnerabilities . . . . .                           | 141 |
| Test questionable applications . . . . .                     | 141 |
| Be sensitive to the performance of the system . . . . .      | 141 |
| Replace old Windows systems . . . . .                        | 142 |
| Periodically re-evaluate and rebuild . . . . .               | 142 |
| Monitoring . . . . .                                         | 143 |
| Logging and auditing . . . . .                               | 144 |

|                                                             |            |
|-------------------------------------------------------------|------------|
| Clean up the system . . . . .                               | 144        |
| Prepare for the eventual attack . . . . .                   | 145        |
| Attacks Against the Windows Workstation . . . . .           | 145        |
| Viruses . . . . .                                           | 145        |
| Worms . . . . .                                             | 146        |
| Trojan horses . . . . .                                     | 147        |
| Spyware and ad support . . . . .                            | 148        |
| Spyware and "Big Brother" . . . . .                         | 149        |
| Physical attacks . . . . .                                  | 149        |
| TEMPEST attacks . . . . .                                   | 150        |
| Backdoors . . . . .                                         | 150        |
| Denial-of-service attacks . . . . .                         | 151        |
| File extensions . . . . .                                   | 151        |
| Packet sniffing . . . . .                                   | 152        |
| Hijacking and session replay . . . . .                      | 152        |
| Social engineering . . . . .                                | 152        |
| Summary . . . . .                                           | 153        |
| <b>Chapter 5: UNIX and Linux Security . . . . .</b>         | <b>155</b> |
| The Focus of UNIX/Linux Security . . . . .                  | 155        |
| UNIX as a target . . . . .                                  | 155        |
| UNIX/Linux as a poor target . . . . .                       | 157        |
| Open source issues . . . . .                                | 158        |
| Physical Security . . . . .                                 | 160        |
| Limiting access . . . . .                                   | 161        |
| Detecting hardware changes . . . . .                        | 162        |
| Disk partitioning . . . . .                                 | 163        |
| Prepare for the eventual attack . . . . .                   | 164        |
| Controlling the Configuration . . . . .                     | 166        |
| Installed packages . . . . .                                | 166        |
| Kernel configurations . . . . .                             | 167        |
| Operating UNIX Safely . . . . .                             | 174        |
| Controlling processes . . . . .                             | 174        |
| Controlling users . . . . .                                 | 187        |
| Encryption and certificates . . . . .                       | 194        |
| Hardening UNIX . . . . .                                    | 196        |
| Configuration items . . . . .                               | 196        |
| TCP wrapper . . . . .                                       | 198        |
| Checking strong passwords . . . . .                         | 198        |
| Packet filtering with iptables . . . . .                    | 199        |
| Summary . . . . .                                           | 200        |
| <b>Chapter 6: Web Browser and Client Security . . . . .</b> | <b>201</b> |
| Web Browser and Client Risk . . . . .                       | 201        |
| Privacy versus security . . . . .                           | 202        |
| Web browser convenience . . . . .                           | 202        |

|                                                   |            |
|---------------------------------------------------|------------|
| Web browser productivity and popularity . . . . . | 202        |
| Web browser evolution . . . . .                   | 203        |
| Web browser risks . . . . .                       | 204        |
| Issues working against the attacker . . . . .     | 205        |
| How a Web Browser Works . . . . .                 | 205        |
| HTTP, the browser protocol . . . . .              | 205        |
| Cookies . . . . .                                 | 208        |
| Maintaining state . . . . .                       | 210        |
| Caching . . . . .                                 | 212        |
| Secure Socket Layer . . . . .                     | 212        |
| Web Browser Attacks . . . . .                     | 216        |
| Hijacking attack . . . . .                        | 216        |
| Replay attack . . . . .                           | 217        |
| Browser parasites . . . . .                       | 218        |
| Operating Safely . . . . .                        | 219        |
| Keeping current with patches . . . . .            | 220        |
| Avoiding viruses . . . . .                        | 220        |
| Using secure sites . . . . .                      | 220        |
| Securing the network environment . . . . .        | 222        |
| Using a secure proxy . . . . .                    | 223        |
| Avoid using private data . . . . .                | 223        |
| General recommendations . . . . .                 | 224        |
| Web Browser Configurations . . . . .              | 225        |
| Cookies . . . . .                                 | 225        |
| Plugins . . . . .                                 | 226        |
| Netscape-specific issues . . . . .                | 230        |
| Internet Explorer-specific issues . . . . .       | 231        |
| Summary . . . . .                                 | 236        |
| <b>Chapter 7: Web Security . . . . .</b>          | <b>237</b> |
| What Is HTTP? . . . . .                           | 237        |
| How Does HTTP Work? . . . . .                     | 239        |
| HTTP implementation . . . . .                     | 242        |
| Persistent connections . . . . .                  | 244        |
| The client/server model . . . . .                 | 248        |
| Put . . . . .                                     | 249        |
| Get . . . . .                                     | 250        |
| Burstable TCP . . . . .                           | 250        |
| HTML . . . . .                                    | 251        |
| Server Content . . . . .                          | 252        |
| CGI scripts . . . . .                             | 252        |
| PHP pages . . . . .                               | 253        |
| Client Content . . . . .                          | 254        |
| JavaScript . . . . .                              | 254        |
| Java . . . . .                                    | 255        |
| ActiveX . . . . .                                 | 257        |

|                                              |            |
|----------------------------------------------|------------|
| State . . . . .                              | 260        |
| What is state? . . . . .                     | 260        |
| How does it relate to HTTP? . . . . .        | 260        |
| What applications need state? . . . . .      | 260        |
| Tracking state . . . . .                     | 261        |
| Cookies . . . . .                            | 261        |
| Web bugs . . . . .                           | 264        |
| URL tracking . . . . .                       | 265        |
| Hidden frames . . . . .                      | 265        |
| Hidden fields . . . . .                      | 266        |
| Attacking Web Servers . . . . .              | 266        |
| Account harvesting . . . . .                 | 266        |
| SQL injection . . . . .                      | 267        |
| E-commerce Design . . . . .                  | 269        |
| Physical location . . . . .                  | 269        |
| Summary . . . . .                            | 271        |
| <b>Chapter 8: E-mail Security . . . . .</b>  | <b>273</b> |
| The E-mail Risk . . . . .                    | 273        |
| Data vulnerabilities . . . . .               | 273        |
| Simple e-mail versus collaboration . . . . . | 274        |
| Spam . . . . .                               | 285        |
| Maintaining e-mail confidentiality . . . . . | 288        |
| Maintaining e-mail integrity . . . . .       | 289        |
| E-mail availability issues . . . . .         | 290        |
| The E-mail Protocols . . . . .               | 290        |
| SMTP . . . . .                               | 290        |
| POP . . . . .                                | 294        |
| IMAP . . . . .                               | 295        |
| E-mail Authentication . . . . .              | 296        |
| Plain login . . . . .                        | 296        |
| Login authentication . . . . .               | 297        |
| APOP . . . . .                               | 297        |
| NTLM/SPA . . . . .                           | 298        |
| +OK logged onPOP before SMTP . . . . .       | 299        |
| Kerberos and GSSAPI . . . . .                | 299        |
| Operating Safely When Using E-mail . . . . . | 300        |
| Be paranoid . . . . .                        | 300        |
| Mail client configurations . . . . .         | 301        |
| Application versions . . . . .               | 302        |
| Architectural considerations . . . . .       | 302        |
| SSH tunnel . . . . .                         | 303        |
| PGP and GPG . . . . .                        | 307        |
| Summary . . . . .                            | 308        |

**Chapter 9: Domain Name System . . . . . 309**

|                                                     |     |
|-----------------------------------------------------|-----|
| Purpose of DNS . . . . .                            | 310 |
| Forward lookups . . . . .                           | 315 |
| Reverse lookups . . . . .                           | 316 |
| Alternative Approaches to Name Resolution . . . . . | 318 |
| Security Issues with DNS . . . . .                  | 319 |
| Misconfigurations . . . . .                         | 321 |
| Zone transfers . . . . .                            | 322 |
| Predictable query IDs . . . . .                     | 325 |
| Recursion and iterative queries . . . . .           | 325 |
| DNS Attacks . . . . .                               | 326 |
| Simple DNS attack . . . . .                         | 327 |
| Cache poisoning . . . . .                           | 327 |
| Designing DNS . . . . .                             | 329 |
| Split DNS . . . . .                                 | 329 |
| Split-split DNS . . . . .                           | 329 |
| Master Slave DNS . . . . .                          | 331 |
| Detailed DNS Architecture . . . . .                 | 331 |
| Summary . . . . .                                   | 332 |

**Chapter 10: Server Security . . . . . 333**

|                                                         |     |
|---------------------------------------------------------|-----|
| General Server Risks . . . . .                          | 333 |
| Security by Design . . . . .                            | 334 |
| Maintain a security mindset . . . . .                   | 335 |
| Establishing a secure development environment . . . . . | 340 |
| Secure development practices . . . . .                  | 344 |
| Test, test, test . . . . .                              | 351 |
| Operating Servers Safely . . . . .                      | 354 |
| Controlling the server configuration . . . . .          | 354 |
| Controlling users and access . . . . .                  | 356 |
| Passwords . . . . .                                     | 357 |
| Monitoring, auditing, and logging . . . . .             | 357 |
| Server Applications . . . . .                           | 358 |
| Data sharing . . . . .                                  | 358 |
| Peer to peer . . . . .                                  | 362 |
| Instant messaging and chat . . . . .                    | 363 |
| Summary . . . . .                                       | 364 |

**Part III: Network Security Fundamentals 365****Chapter 11: Network Protocols . . . . . 367**

|                                               |     |
|-----------------------------------------------|-----|
| Protocols . . . . .                           | 367 |
| The Open Systems Interconnect Model . . . . . | 368 |

|                                                                          |            |
|--------------------------------------------------------------------------|------------|
| The OSI Layers . . . . .                                                 | 369        |
| The Application layer . . . . .                                          | 369        |
| The Presentation layer . . . . .                                         | 370        |
| The Session Layer . . . . .                                              | 370        |
| The Transport layer . . . . .                                            | 371        |
| The Network layer . . . . .                                              | 372        |
| The Data Link layer . . . . .                                            | 373        |
| The Physical layer . . . . .                                             | 374        |
| The TCP/IP Model . . . . .                                               | 375        |
| TCP/IP Model Layers . . . . .                                            | 377        |
| Network Address Translation . . . . .                                    | 379        |
| Summary . . . . .                                                        | 379        |
| <b>Chapter 12: Wireless Security . . . . .</b>                           | <b>381</b> |
| Electromagnetic Spectrum . . . . .                                       | 381        |
| The Cellular Phone Network . . . . .                                     | 383        |
| Placing a Cellular Telephone Call . . . . .                              | 385        |
| Wireless Transmission Systems . . . . .                                  | 386        |
| Time Division Multiple Access . . . . .                                  | 386        |
| Frequency Division Multiple Access . . . . .                             | 386        |
| Code Division Multiple Access . . . . .                                  | 387        |
| Wireless transmission system types . . . . .                             | 388        |
| Pervasive Wireless Data Network Technologies . . . . .                   | 393        |
| Spread spectrum . . . . .                                                | 393        |
| Spread spectrum basics . . . . .                                         | 393        |
| IEEE Wireless LAN Specifications . . . . .                               | 397        |
| The PHY layer . . . . .                                                  | 398        |
| The MAC layer . . . . .                                                  | 398        |
| IEEE 802.11 Wireless Security . . . . .                                  | 400        |
| WEP . . . . .                                                            | 400        |
| WEP security upgrades . . . . .                                          | 402        |
| 802.11i . . . . .                                                        | 408        |
| Bluetooth . . . . .                                                      | 413        |
| Wireless Application Protocol . . . . .                                  | 414        |
| Summary . . . . .                                                        | 416        |
| <b>Chapter 13: Network Architecture Fundamentals . . . . .</b>           | <b>417</b> |
| Network Segments . . . . .                                               | 418        |
| Public networks . . . . .                                                | 418        |
| Semi-private networks . . . . .                                          | 418        |
| Private networks . . . . .                                               | 419        |
| Perimeter Defense . . . . .                                              | 419        |
| Network Address Translation . . . . .                                    | 420        |
| Basic Architecture Issues . . . . .                                      | 422        |
| Subnetting, Switching, and VLANs . . . . .                               | 424        |
| Address Resolution Protocol and Media Access Control Addresses . . . . . | 426        |

| <i>Class</i>                                                  | <i>Network Address</i>                                                                                       | <i>Host Address</i>                      | <i>Example Address</i> |
|---------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|------------------------------------------|------------------------|
| Class D<br>Address range =<br>224.0.0.0 to<br>239.255.255.255 | Binary address always<br>begins with 1110,<br>therefore, the decimal<br>address ranges from<br>224 to 239.   | Reserved for<br>multicasting             |                        |
| Class E<br>Address range =<br>240.0.0.0 to<br>254.255.255.254 | Binary addresses start<br>with 1111, therefore,<br>the decimal number<br>can be anywhere from<br>240 to 255. | Reserved for<br>experimental<br>purposes |                        |

## TCP/IP Model Layers

The TCP/IP model comprises four layers: the Application layer, the Host-to-Host layer or Transport layer, the Internet layer, and the Network Access layer. These layers and their corresponding functions and protocols are summarized in Table 11-3.

**Table 11-3**  
**TCP/IP Model Layers**

| <i>Layer</i>                                   | <i>Function</i>                                                                                                                                                                                                                                                                              | <i>Protocols or Standards</i> |
|------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------|
| Layer 4:<br>Application                        | Equivalent to Application, Presentation, and Session layers of the OSI model. In TCP/IP, an application is a process that is above the Transport layer. Applications communicate through sockets and ports.                                                                                  | SMTP, POP, HTTP, FTP          |
| Layer 3:<br>Host-to-Host or<br>Transport Layer | Similar to the OSI Transport layer; performs packet sequencing, supports reliable end-to-end communications, ensures data integrity, and provides for error-free data delivery.                                                                                                              | TCP, UDP                      |
| Layer 2:<br>Internet Layer                     | Isolates the upper-layer protocols from the details of the underlying network and manages the connections across the network. Uses protocols that provide for logical transmission of packets over a network and controls communications among hosts; assigns IP addresses to network nodes. | IP, ARP, RARP, ICMP           |

*Continued*

Table 11-3 (continued)

| Layer                               | Function                                                                                                                                                                                                                                                                                                                                                    | Protocols or Standards                 |
|-------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------|
| Layer 1:<br>Network<br>Access Layer | Combines the Data Link layer and Physical layer functions of the OSI model. These functions include mapping IP addresses to MAC addresses, using software drivers, and encapsulation of IP datagrams into frames to be transmitted by the network. It is also concerned with communications hardware and software, connectors, voltage levels, and cabling. | EIA RS-232,<br>EIA RS-449,<br>IEEE 802 |

As with the OSI model, encapsulation occurs as data traverses the layers from the Application layer to the Network Access layer at the transmitting node. This process is reversed in the receiving node. Encapsulation in TCP/IP is illustrated in Figure 11-1.

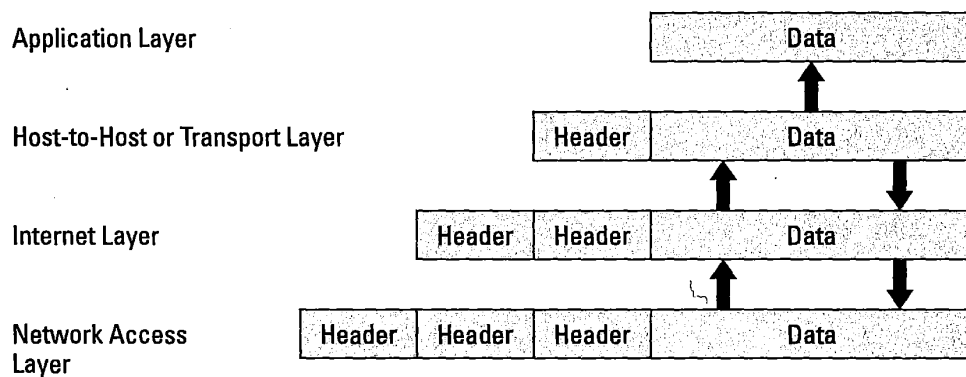


Figure 11-1: TCP/IP Encapsulation.

The example protocols listed in Table 11-3 have been discussed under the OSI model, except for the Post Office Protocol (POP). Using POP, an e-mail client can retrieve e-mail from a mail server. The latest version of POP is POP3 and can be used with or without SMTP. A security issue with POP3 is that the password used for authentication is transmitted in the clear.

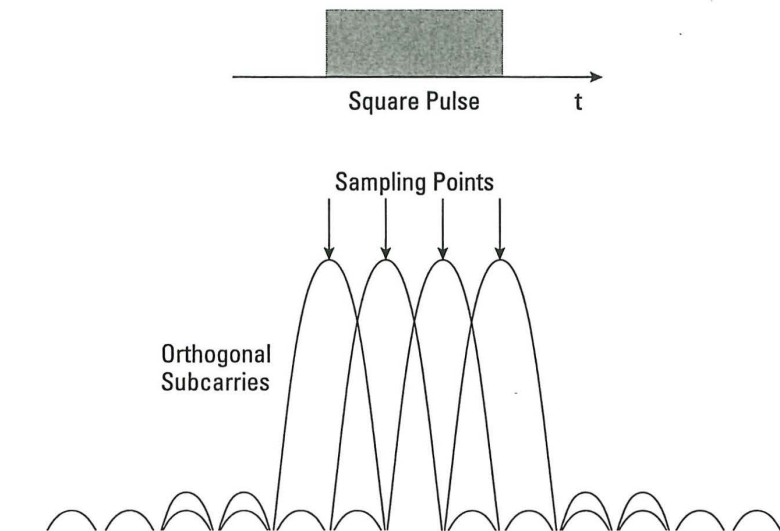


Figure 12-11: OFDM subcarriers

The orthogonality in OFDM comes from the subcarrier spacings, as shown in Figure 12-11. The IFFT modulation of the square pulse produces subcarrier spacings, at which, when sampled at the points shown in Figure 12-11, all other signals are at zero value. Thus, the subcarrier is orthogonal to all the other signals. At the receiving end, a Fast Fourier Transform is applied to recover the original data. OFDM is used in the IEEE 802.11 Wireless LAN standard, Asymmetric DSL, and for digital television in Australia, Europe, and Japan.

## IEEE Wireless LAN Specifications

The IEEE 802.11 family of wireless LAN standards specifies an interface between a wireless client and a base station or access point, as well as among wireless clients. Work on the first standard, 802.11, began in 1990 and evolved from various draft versions; approval of the final of 802.11 draft occurred on June 26, 1997.

The 802.11 specification identifies an over-the-air interface between a mobile device wireless client and a base station or between two mobile device wireless clients.

The IEEE 802.11 standard specifies parameters of both the physical (PHY) and medium access control (MAC) layers of the network.

## The PHY layer

The PHY layer is responsible for the transmission of data among nodes. It can use DSSS, FHSS, OFDM, or infrared (IR) pulse position modulation. The family of standards supports data rates ranging from 2 Mbps to 54 Mbps, with the early draft of a proposed new standard, 802.11n, targeting 100 Mbps.

## The MAC layer

The MAC layer is a set of protocols responsible for maintaining order in the use of a shared medium. The 802.11 standard specifies a carrier sense multiple access with collision avoidance (CSMA/CA) protocol for the wireless LANs. The MAC layer provides the following services:

- ♦ **Data transfer** — CSMA/CA media access.
- ♦ **Association** — Establishment of wireless links between wireless clients and access points in infrastructure networks.
- ♦ **Reassociation** — An action that occurs in addition to association when a wireless client moves from one Basic Service Set (BSS) network to another, such as in roaming. A BSS is a group of 802.11-compliant stations that comprise a fully connected wireless network.
- ♦ **Authentication** — The process of proving a client identity through the use of an 802.11 authentication mechanism.
- ♦ **Privacy** — In the 802.11 family of standards, there are options for different levels of protection of the transmitted data. These options are discussed in detail in a later section of this chapter.

802.11 offers two operational modes: ad hoc and infrastructure.

The *ad hoc mode* refers to Independent Basic Service Set (IBSS) networks, which do not have a backbone infrastructure and involve at least two wireless stations. IBSS is not intended for long-range use, but is normally implemented in rooms or sections of a building. Ad hoc mode is a peer-to-peer networking paradigm in which each mobile client communicates directly with other mobile clients in the network and only clients within the same cell can communicate. Communication outside of the cell can take place if a cell member operates as a routing service.

The *infrastructure mode* refers to BSS networks that incorporate access points to communicate between mobile devices and the wired network or other wireless networks.

As of this writing, there are five 802.11 standards: 802.11, 802.11a, 802.11b, 802.11g, and 802.11e. 802.11e differs from the others in that it focuses on providing for quality of service (QoS) in a wireless LAN. A proposed standard that is in development at this time is standard 802.11n. Table 12-2 summarizes the characteristics of the 802.11 family.

**Table 12-2**  
**IEEE 802.11 Wireless LAN Standards**

| <i>Standard</i>                                       | <i>Band</i> | <i>Technology</i> | <i>Transmission Speed</i>                                                                          | <i>Comments</i>                                                                                                                                                                                                                                                                                                                                                                                                    |
|-------------------------------------------------------|-------------|-------------------|----------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 802.11                                                | 2.4 GHz     | FHSS or<br>DHSS   | 1 or 2 Mbps                                                                                        | Original wireless LAN<br>standard                                                                                                                                                                                                                                                                                                                                                                                  |
| 802.11b                                               | 2.4 GHz     | DSS               | 11 Mbps, but<br>decreases to 5.5<br>Mbps, 2 Mbps, or 1<br>Mbps as a function<br>of signal strength | Extension to 802.11;<br>known as Wi-Fi                                                                                                                                                                                                                                                                                                                                                                             |
| 802.11a                                               | 5 GHz       | OFDM              | 54 Mbps                                                                                            | Extension to 802.11                                                                                                                                                                                                                                                                                                                                                                                                |
| 802.11g                                               | 2.4 GHz     | OFDM              | 54 Mbps                                                                                            | Extension to 802.11                                                                                                                                                                                                                                                                                                                                                                                                |
| 802.11e<br>Quality of<br>Service<br>(QoS)<br>Standard |             |                   |                                                                                                    | Guaranteed timely<br>delivery of application<br>data to specified<br>destinations; provides<br>guaranteed bandwidth;<br>supports streaming<br>multimedia files;<br>incorporates error<br>correction.                                                                                                                                                                                                               |
| 802.11n<br>proposed<br>standard                       | 5 GHz       | OFDM<br>proposed  | Targeting 100 Mbps                                                                                 | Currently under<br>development; probability<br>of using Multiple-In,<br>Multiple-Out (MIMO)<br>scheme. MIMO uses<br>multiple transmitting<br>antennas, tuned to the<br>same channel, with each<br>antenna transmitting a<br>different signal. The<br>receiver has the same<br>number of antennas that<br>listen to all transmitters.<br>The received signals are<br>recombined to recover<br>the transmitted data. |

## IEEE 802.11 Wireless Security

The original 802.11 wireless LAN specifications defined a security option, Wired Equivalent Privacy (WEP). In WEP, a shared key is configured into the access point and its wireless clients. In the 802.11 standard, data is transferred in the clear by default. If confidentiality is desired, the WEP option encrypts data before it is sent. The WEP algorithm is the RC4 symmetric cipher. The algorithm employs a secret key that is shared between a mobile station (for example, a laptop with a wireless Ethernet card) and a base station access point to protect the confidentiality of information being transmitted on the LAN. The transmitted packets are encrypted with a secret key and an Integrity Check (IC) field composed of a CRC-32 check sum attached to the message.

The WEP implementation is weak in both confidentiality and authentication, and, with tools readily available on the Internet, WEP is easily broken and messages compromised. As a result, stronger privacy systems have been implemented for 802.11. The evolution of these mechanisms is detailed in the following sections.

### WEP

As previously noted, WEP uses the RC4 symmetric key stream cipher for protecting the confidentiality of the transmitted messages. WEP also provides for a weak authentication of the user station to the access point, but not vice versa. This authentication is usually accomplished using a shared secret key to encrypt the data frames. The WEP symmetric key is comprised of two components, a variable, 24-bit Initialization Vector (IV) and a fixed 40- or 104-bit secret key. Because the secret key is seldom changed, the purpose of the IV is to thwart cryptanalysis against WEP by having the client use a different IV when encrypting message packets.

Because of the limited processing power in commodity-produced access points, the RC4 stream encryption algorithm is off-loaded to custom hardware. The hardware functions encrypt each message packet with a key that is comprised of the concatenation of a base secret key with an IV. The packet construction and the key composition are illustrated in Figure 12-12.

Note that the IV is transmitted as plaintext in the packet. When the packet is received at the access point, the hardware function retrieves the base secret key that it knows, concatenates it with the IV in the transmitted message packet, and uses this key to decrypt the packet.

However, because the IV is relatively short, packet monitoring will show repetitions of the IV and, thus, enable attackers to obtain the base secret key. One approach is to use the plaintext IV and discover WEP RC4 weak keys to mount a known plaintext attack. Researchers at UC Berkeley ([www.isaac.cs.berkeley.edu/isaac/wep-faq.html](http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html)) have shown that WEP security can be easily broken. Subsequently, a freely available program called AirSnort was widely distributed on the Internet that can be used to break WEP encryption and read transmitted messages. WEP is also vulnerable to forgery and replay attacks, wherein an attacker can modify packets and retransmit them or capture packets and retransmit them at a later time.

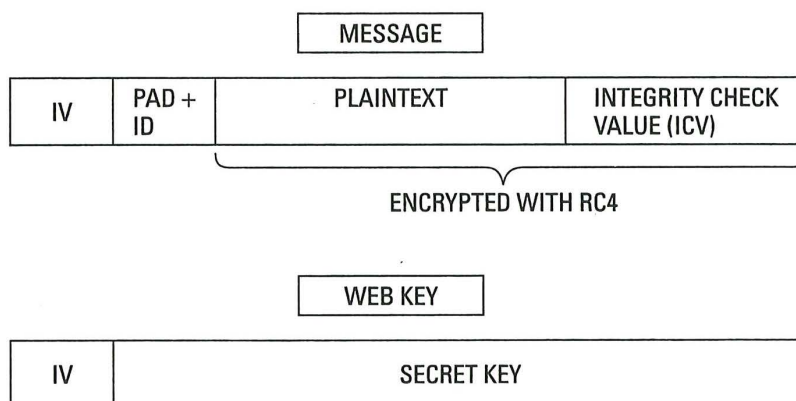


**Figure 12-12:** A WEP message and key

WEP provides for open and shared key authentication. The following sections describe each option and their associated vulnerabilities.

### WEP open authentication

In WEP *open authentication* a client station provides a *Service Set Identity (SSID)* that is common to the stations on its network segment and its access point. This SSID authorizes and associates a client station to the access point. A vulnerability exists with this approach in that the access point transmits the SSID in the clear at intervals in management frames. Thus, the SSID is easily available to attackers to establish an association with the access point.

### WEP shared key authentication

The WEP *shared key architecture* was intended to implement secure authentication of the client by the access point through the following steps:

1. The client station transmits an authorization request.
2. The access point returns a challenge string in the clear.
3. The client chooses an IV.
4. Using the IV and secret base key, the client encrypts the challenge string.
5. The client station sends the IV and encrypted challenge string to the access point.
6. The access point also encrypts the challenge string using the transmitted IV and the same secret base key.
7. If the client's encrypted challenge string is identical to the challenge string sent by the client station, the association occurs.

The vulnerability in this process is that cryptanalysis can use the intercepted plain text/cipher text pair and IV to determine the RC4 key. This attack is possible when

all the IVs have been exhausted for a session and the IVs have to be reused. In this situation, when IV1 in a message is equal to IV2 in another message, the cryptanalysis proceeds as follows:

1. Ciphertext C1 = Plaintext P1 XOR [Stream Cipher RC4 with key generated through the use of K, IV1]
2. Ciphertext C2 = Plaintext P2 XOR [Stream Cipher RC4 with key generated through the use of K, IV2]
3. If IV1 = IV2, then proceed to Step 4.
4.  $C1 \text{ XOR } C2 = \{ \text{Plaintext P1 XOR [Stream Cipher RC4 with key generated through the use of K, IV1]} \} \text{ XOR } \{ \text{Plaintext P2 XOR [Stream Cipher RC4 with key generated through the use of K, IV2]} \} = P1 \text{ XOR } P2$ , the XOR of the two plaintexts.

With the Exclusive Or of the two plain text items known corresponding to the transmitted cipher text items, dictionary attacks can be applied to determine the plain text items.

## WEP security upgrades

Because of the weaknesses in WEP security, IEEE 802.11 established Task Group i (TGi) to develop approaches to address WEP problems. TGi had to consider a number of issues and constraints. One path was to redesign 802.11 security so as not to include any legacy WEP functions. Another path was to upgrade WEP security while keeping the same WEP architecture. Both approaches were chosen, resulting in the completely new 802.11i standard and the upgraded WEP encryption and integrity method called the *Temporal Key Integrity Protocol* (TKIP). The latter approach was necessary to accommodate the huge base of existing wireless WEP devices already deployed and to have improved security in place because of the anticipated delay in developing and finalizing the 802.11i standard. The installed WEP implementations have hardware-based WEP functions that cannot be easily modified, so the TKIP solution was chosen because it can be installed as a software upgrade to the legacy systems. In addition, because of the limited additional computing capability remaining on extant access points, the TKIP upgrade could not be computing resource intensive. TKIP uses the *802.1X authentication architecture* as a basis for secure key exchange, so the next section briefly describes 802.1X as a precursor to an overview of the TKIP algorithms.

### 802.1X authentication

802.1X is a port-based authentication mechanism that operates under the *Extensible Authentication Protocol* (EAP) transport protocol (RFC 2284). For wireless LANs, the EAP protocol is known as EAP over LAN (EAPOL). EAPOL is applied to the exchange

of challenges and responses between client stations, or *supplicants*, as they are called in the protocol, and an authentication server. The third entity in 802.1X is the *authenticator*, a dual access control port, similar to the access point. The authentication server is usually a RADIUS server, but other authentication servers can be employed. In this discussion, a RADIUS server is used. EAPOL supports a number of protocols, including Transport Layer Security (TLS), RFC 2246. A typical authentication process employing EAPOL proceeds as follows:

1. The supplicant sends credentials to the RADIUS server.
2. The RADIUS server provides credentials to the supplicant.
3. Upon mutual authentication, the protocol is used to establish session keys.
4. The session keys are used to encrypt the client station message.

In detail, the sequence occurs in the following steps:

1. A conventional 802.11 association is established.
2. At this point, all non-802.1X traffic is blocked.
3. The RADIUS server sends a challenge to the supplicant (client station).
4. The client hashes the user-provided password as a response to the RADIUS server. This hash is sent to the RADIUS authentication server through the authenticator.
5. The RADIUS server uses the same process to compute the hash based on its database of user passwords.
6. If a match of the hashes is obtained, the RADIUS server generates a dynamic WEP secret key and sends it to the authenticator.
7. The WEP secret key is sent to the client via EAPOL key frames.
8. The secret keys are updated at specified intervals.

Because employing 802.1X for WEP encryption does not eliminate weak IV and IV collision vulnerabilities, TKIP was developed to address these and other WEP security weaknesses.

### Temporal Key Integrity Protocol

TKIP is built around the existing WEP security algorithm because of the necessity of not adding complex cryptographic algorithms whose execution would far exceed the spare CPU cycles available on most of today's deployed access points. Table 12-3 lists the upgrades provided by TKIP in terms of the security weaknesses addressed.

**Table 12-3**  
**TKIP Upgrades for WEP Weaknesses**

| <i>Weakness</i>                   | <i>TKIP Upgrade</i>                         |
|-----------------------------------|---------------------------------------------|
| Correlation of IVs with weak keys | Per-packet key mixing function              |
| Replay                            | IV sequencing discipline                    |
| Key reuse                         | Rekeying approach                           |
| Susceptibility to forgery         | Message Integrity code (MIC) called Michael |

#### Per-packet mixing function

The TKIP *per-packet key mixing function* addresses the problem of correlating IVs with weak keys by using a key that varies with time, or temporal key, as the WEP secret base key. It then uses the packet sequence counter and temporal key to construct the per-packet key and IV. These operations hide the relationship between the IV and the per-packet key and are illustrated in Figure 12-13.

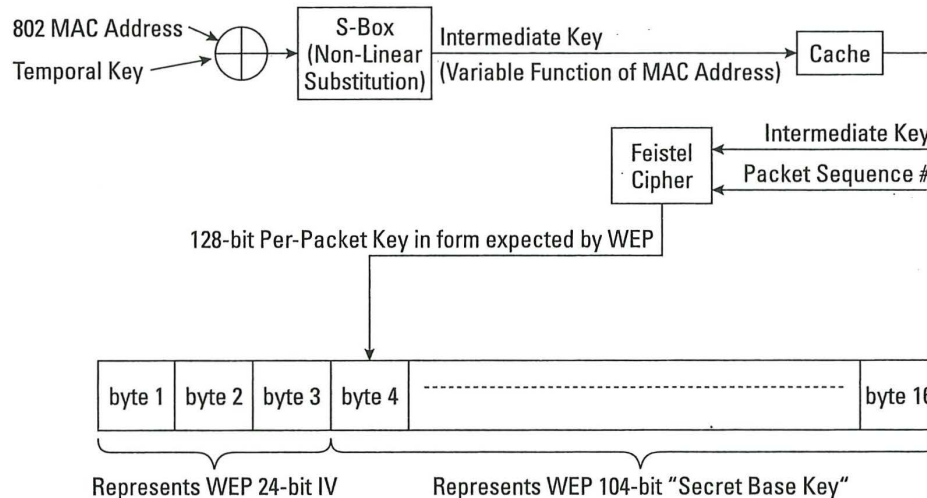


**Figure 12-13:** TKIP per-packet mixing function

The process in Figure 12-13 shows that using the Exclusive Or function for the local MAC address with the temporal key results in different client stations and access points generating correspondingly different intermediate keys. Thus, the per-packet encryption keys are different at every client station. The result of the total process is a 16-byte packet that corresponds to the input that is expected by existing WEP hardware.

#### IV sequencing discipline

As a control against replay attacks, TKIP applies an IV sequencing discipline in which a receiver determines if a packet is out of sequence. If that condition is true, the receiver assumes it is a replay and discards the packet. A packet is defined as out of sequence if its IV is less than or equal to that of a previously correctly received packet. By using the WEP IV field as a packet sequence number, the procedure for detecting and countering replays is summarized as follows:

1. New TKIP keys are used.
2. Receiver and transmitter initialize the packet sequence number to zero.
3. As each packet is transmitted, the packet sequence number is incremented by the transmitter.
4. The IV sequencing discipline is applied to determine if a packet is out of sequence and a replay has occurred.

This procedure is illustrated in Figure 12-14.

#### Message Integrity Codes against forgery

An ideal *Message Integrity Code* (MIC) is a unique, unambiguous representation of the transmitted message that will change if the message bits change. Thus, if an MIC is calculated using an authentication key by a transmitting entity and sent with the message, the receiver can similarly calculate another MIC based on the message and compare it to the MIC that accompanied the message. If the two MICs are identical, in theory, the message was not modified during transmission.

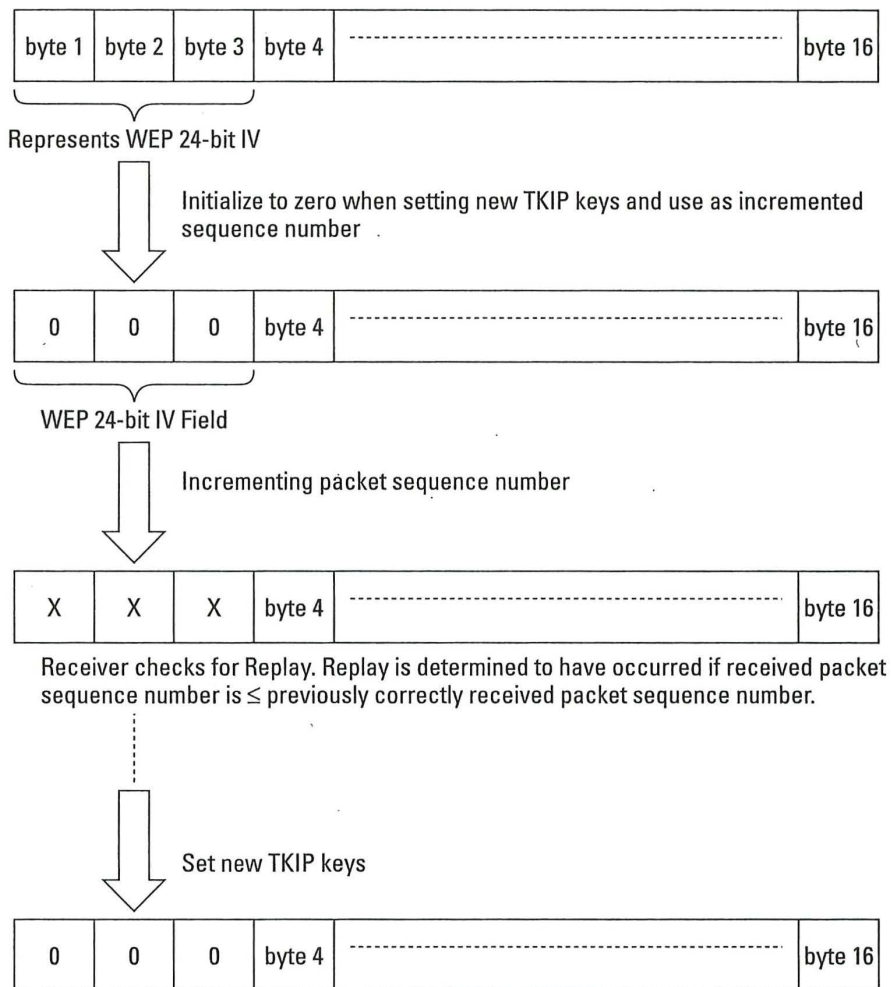


**Figure 12-14:** TKIP replay sequence checking

In TKIP, the 64-bit MIC is called *Michael* and was developed by Niels Ferguson, an independent cryptography consultant based in Amsterdam, Holland. His Web site is located at [www.macfergus.com/niels/](http://www.macfergus.com/niels/). The TKIP MIC process is illustrated in Figure 12-15.

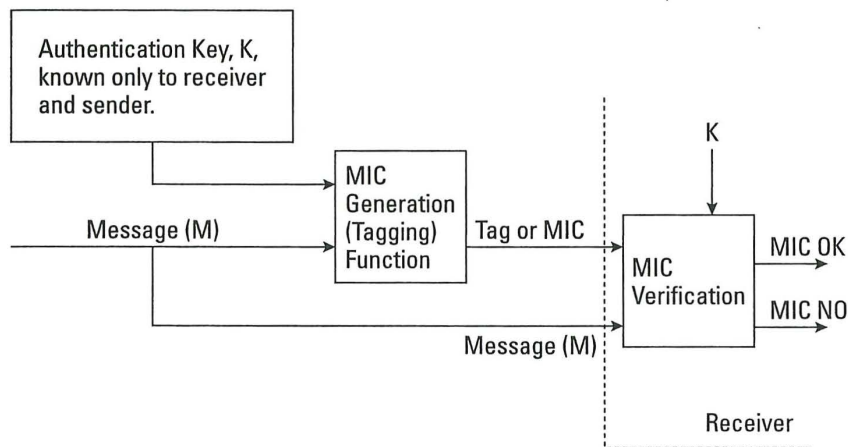


**Figure 12-15:** TKIP MIC generation and verification

### Rekeying against key reuse

To protect against key reuse, 802.1X uses a hierarchy of master keys, key encryption keys, and temporal keys. The 802.1X temporal keys are used in the TKIP authentication and confidentiality processes. A temporal key set comprises a 64-bit key for the MIC process, as described in the previous section, and a 128-bit encryption key. A different set of temporal keys is used in each direction when an association is established. The material used to generate the temporal keys must be protected from compromise and this protection is accomplished by use of key encryption keys. The master key is needed to set up the key encryption keys. This process is summarized as follows:

- ♦ 802.1X defines that the authentication server and client station share a secret key, the master key.
- ♦ 802.1X defines that the authentication server and access point share a secret key, derived by the authentication server and client station from the master key and distributed by the authentication server to the access point.
- ♦ A new master key is used with each session (a session covers the time from authentication to when the key expires, is revoked, or when a client station no longer communicates).
- ♦ The master key is used to protect the communication of key encryption keys between a client station and the access point.
- ♦ The key encryption keys are employed to protect the transmitted keying material used by the access point and client to generate sets of temporal keys.
- ♦ The pairs of temporal keys are used for integrity protection and confidentiality of the data.

Figure 12-16 shows the relationships and locations of the three types of keys.

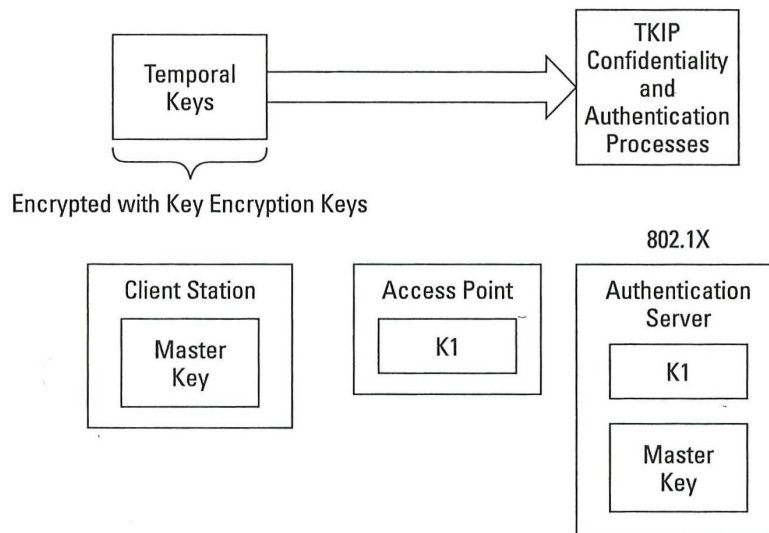


**Figure 12-16:** Key hierarchy for rekeying

## 802.11i

The 802.11i wireless security standard was ratified in June of 2004. The IEEE 802.11 committee considers this specification a long-term solution to wireless security. It incorporates TKIP, 802.1X, and the Advanced Encryption Standard (AES). AES is a block cipher and, in 802.11i, processes plain text in 128-bit blocks. It uses the following set of keys:

- ♦ **A symmetric master key** — Possessed by the authentication server and client station for the positive access decision
- ♦ **A pairwise master key (PMK)** — A fresh symmetric key possessed by the access point and client station and used for authorization to access the 802.11 medium
- ♦ **A pairwise transient key (PTK)** — A collection of the following operational keys:
  - **Key encryption key (KEK)** — Used to distribute the group transient key (GTK), which is an operational temporal key used to protect multicast and broadcast data

- **Key confirmation key (KCK)**— Binds the PMK to the client station and access point
- **Temporal key (TK)**— Protects transmitted data

Thus, 802.11i employs a 128-bit key, combines encryption and authentication, uses temporal keys for both functions, and protects the entire 802.11i packet. Relative to the authentication server and EAP, RADIUS and EAP-TLS are not officially a part of 802.11i, but are de facto standards for use in 802.11i.

The next sections explore the AES and its employment in 802.11i because it is the major component of and provides the increased security capabilities in the new standard.

### **AES Counter and Cipher-Block Chaining modes**

The two modes of operation of AES relative to 802.11i are Counter (CTR) mode and Cipher-Block Chaining (CBC) mode.

In the CTR mode of operation, AES employs a monotonically increasing counter. The encryption process in the CTR mode is summarized as follows and is shown in Figure 12-17:

1. The Message,  $M$ , is broken into 128-bit blocks  $M_1, M_2, \dots, M_n$
2. The key is determined.
3. The counter is initialized to zero.
4. For each block processed, increment the counter by one.
5. For each block, the counter value is encrypted.
6. The encrypted counter value is XORed with the plain text block,  $M_i$ , to generate the cipher text block,  $C_i$ .
7. When all the plain text blocks have been encrypted, the initial counter value is prepended to the cipher text blocks to generate the message (counter)<sub>0</sub>  $C = (\text{counter}_0)\{C_1, C_2, \dots, C_n\}$ .
8. The message is transmitted.
9. The receiver decrypts the message by reversing the process. It uses the prepended initial counter value as a starting point.

For security, the CTR mode requires a new, different key for every session.

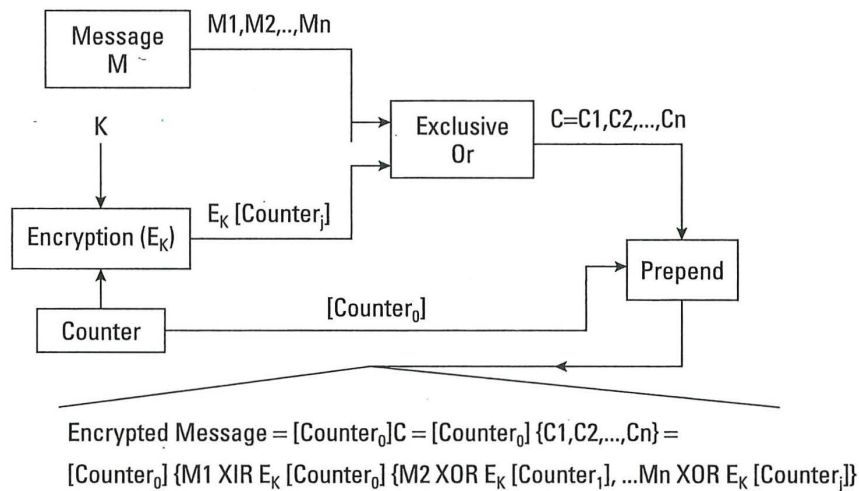


**Figure 12-17:** AES CTR mode

The AES CBC mode employs an initialization vector for enhanced security and operates in the following steps:

1. The Message,  $M$ , is broken into 128-bit blocks  $M1, M2, \dots, Mn$ .
2. A random initial IV value is chosen.
3. This first IV value is XORed with plain text block,  $M1$ .
4. Encrypted block,  $C1$ , is generated by encrypting the result of the XOR in the previous step with the encryption key,  $K$ .  $C1$  also becomes the next IV to be used in the XOR function with  $M2$ .
5. This process iterates until all plain text blocks are encrypted.
6. The message to be transmitted is assembled by prepending the initial IV to the cipher text  $C = C1, C2, \dots, Cn$ .
7. The receiver performs decryption by using the prepended initial IV value and reversing the process.

A different, initial IV must be used for each new message to maintain security.

The steps in the CBC mode are shown in Figure 12-18.

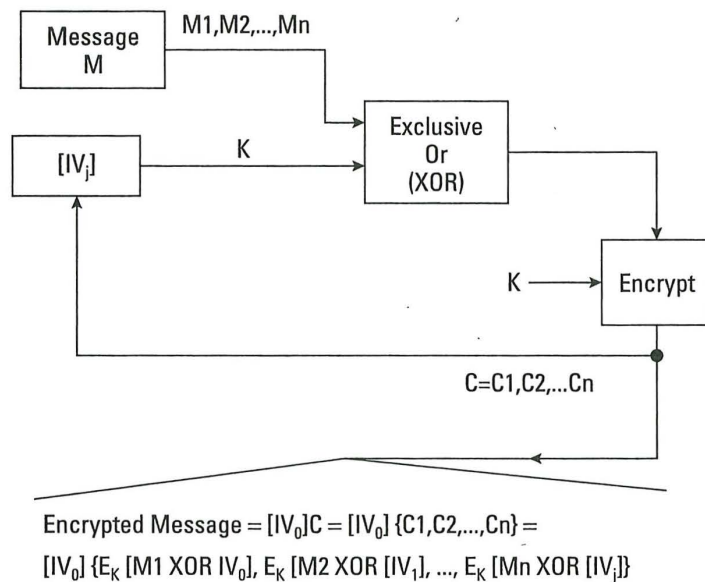


**Figure 12-18:** AES CBC mode

The AES CBC mode can also be employed to generate an MIC and ensure that a message has not been modified during transmission. The MIC is generated as follows:

1. The Message,  $M$ , is broken into 128-bit blocks  $M_1, M_2, \dots, M_n$ .
2. An initial IV value that is known to the transmitter and receiver is chosen.
3. This first IV value is XORed with plain text block,  $M_1$ .
4. A Tag block, MIC1, is generated by encrypting the result of the XOR in the previous step with the encryption key,  $K$ . MIC1 also becomes the next IV to be used in the XOR function with  $M_2$ .
5. This process iterates until the last Tag block, MICn, is generated.
6. The Tag block, MICn, is appended to the transmitted message as an integrity check.

The receiver generates an MICn using the same algorithm and initial IV as the transmitter and compares it to the MICn received with the message. If the values match, the message is assumed to have been transmitted without modification.

Figure 12-19 illustrates AES MIC generation.

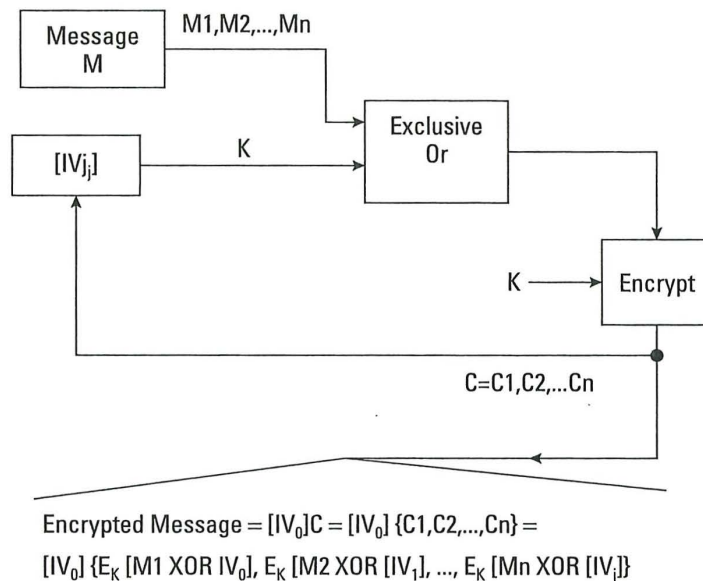


**Figure 12-19:** CBC mode for MIC generation

### Application of AES in 802.11i

The AES is applied in 802.11i in the form of the AES—Counter with CBC-MAC (AES-CCM) protocol. AES-CCM applies the AES CTR mode for confidentiality of data and combination CBC-MAC mode for data integrity.

AES-CCM uses the same AES key for encryption and for generating an MIC. In addition, AES-CCM employs a 48-bit packet sequence counter. This counter is then applied in the CTR mode and in the generation of the CBC-MAC mode initialization vector. The following steps describe this process:

1. Concatenate the source MAC address, the packet sequence counter, a 16-bit per-packet block counter, and a 16-bit string to form the CTR mode counter and CBC MAC-IV. The 16-bit string differentiates the two concatenation results as being the CTR mode counter or the CBC-MAC IV.
2. Increment the packet sequence counter.
3. The CCM-MAC IV and secret AES key are used to compute an MIC over the message packet, including the source and destination addresses.
4. Truncate the MIC to 64 bits.
5. Encrypt the packet and append MIC, applying the CTR mode counter and secret AES key.

6. Insert the packet sequence counter number in between the 802.11 header field and the encrypted message data.
7. Transmit the packet.

On the receiving end, the packet sequence counter is obtained from the message packet and checked for replay. If the message is valid, the packet sequence counter is used to generate the CTR mode counter and the CBC-MAC IV. Then, the process steps used in the transmission process are reversed.

The AES-CCM mode protects against forgeries through the use of an MIC, protects against replays by checking the packet sequence counter, encrypts the source and destination addresses, and does not use an initialization vector or counter value with the same AES secret key.

### **Additional 802.11i capabilities**

802.11i provides for pre-authentication for roaming and, also, a Pre-Shared Key (PSK) mode. In this mode, there is no authentication exchange and a single private key can be assigned to the entire network or on a per-client station pair. PSK is amenable for use in ad hoc and home networks. The PSK mode uses the PKCS#5v2.0PBKDF2 key derivation function to produce a 256-bit PSK from an ASCII string password. RFC 2898, PKCS #5: Password-Based Cryptography Specification Version 2.0" describes this operation, which applies a pseudorandom function to derive keys. The PSK mode is vulnerable to password/passphrase guessing using dictionary attacks.

## **Bluetooth**

Bluetooth is a peer-to-peer, short-range protocol named after Haral Bluetooth, the king of Denmark in the late 900s. It is used to connect cellular phones, laptops, handheld computers, digital cameras, printers, and so on. It is defined in IEEE standard, IEEE 802.15 and has the following characteristics:

- ♦ **FHSS**—Hops 1,600 times per second among 79 RF channels
- ♦ **Transmission rate**—1 Mbps
- ♦ **Transmission distance**—About 30 feet
- ♦ **Frequency band**—2.4 Ghz to 2.5Ghz
- ♦ **Transmitting power**—1 milliwatt, which minimizes interference with other networks (cell phones can transmit up to 3 watts of power)
- ♦ **Transmission range extension**—Range can be extended to 300 feet by increasing transmitting power to 100 milliwatts
- ♦ **Number of devices on the network**—8

Because FHSS is used, other Bluetooth networks can exist in the same area without any mutual interference. Bluetooth devices operate by setting up a personal area network (PAN) called a *piconet* based on the devices' assigned addresses. A Bluetooth piconet operates in the following manner:

- ♦ As an ad hoc network.
- ♦ All Bluetooth devices are peer units.
- ♦ Different piconets have different frequency hopping sequences to prevent interference.
- ♦ All devices on the same piconet are synchronized to the frequency hopping sequence for that piconet.
- ♦ One device operates as a master and the other devices operate as slaves (point-to-multipoint topology).
- ♦ A maximum of 7 active slaves can exist on a piconet, each assigned a 3-bit active member address.
- ♦ Up to 256 inactive (*parked*) slaves that are synchronized to the frequency-hopping sequence can be assigned to the piconet. They can activate rapidly because they are synchronized.

Bluetooth security uses challenge response protocols for authentication, a stream cipher for encryption, and dynamic session keys.

## Wireless Application Protocol

The Wireless Application Protocol (WAP) is widely used by mobile devices to access the Internet. Because it is aimed at small displays and systems with limited bandwidth, it is not designed to display large volumes of data. In addition to cellular phones and PDAs, WAP is applied to network browsing through TV and in automotive displays. It has analogies to TCP/IP, IP, and HTML in wired Internet connections and is actually a set of protocols that covers Layer 7 to Layer 3 of the OSI model. Due to the memory and processor limitations on mobile devices, WAP requires less overhead than TCP/IP.

WAP has evolved through a number of versions, the latest being version 2.0. WAP 2.0 includes support for the transmission and reception of sound and moving pictures over telephones and other devices, as well as providing a toolkit for development and deployment of new services, such as Extensible Hypertext Markup Language (XHTML).

The WAP architecture comprises the following levels:

♦ **Application layer**—Contains the wireless application environment (WAE) and is the direct interface to the user. The Application layer includes the following:

- The Wireless Markup Language (WML)
- A microbrowser specification for Internet access
- WMLScript (development language)

The Handheld Device Markup Language (HDML) is a simpler alternative to and actually preceded WML. HDML contains minimal security features, however. Another alternative is Compact HTML (C-HTML). Used primarily in Japan through NTT DoCoMo's i-mode service, C-HTML is essentially a stripped-down version of HTML. Due to this approach, C-HTML can be displayed on a standard Internet browser.

♦ **Session layer**—Contains the Wireless Session Protocol (WSP), which facilitates the transfer of content between WAP clients and WAP. This layer provides an interface to the WAE through the following activities:

- Connection creation and release between the client and server
- Data exchange between the client and server
- Session suspend and release between the client and server

♦ **Transaction layer**—Provides functionality similar to TCP/IP through the Wireless Transactional Protocol (WTP). WTP provides transaction services to WAP, including acknowledgment of transmissions, retransmissions, and removal of duplicate transactions.

♦ **Security layer**—Contains Wireless Transport Layer Security (WTLS). WTLS is based on Transport Layer Security (TLS) and can be invoked similar to HTTPS in conventional Web browsers. WTLS supports privacy, data integrity, DoS protection services, and authentication. WTLS provides the following three types of authentication:

- **Class 1 (anonymous authentication)**—The client logs on to the server, but in this mode, neither the client nor the server can be certain of the identity of the other.
- **Class 2 (server authentication)**—The server is authenticated to the client, but the client is not authenticated to the server.
- **Class 3 (two-way client and server authentication)**—The server is authenticated to the client and the client is authenticated to the server.

Authentication and authorization can be performed on the mobile device using smart cards to execute PKI-enabled transactions. A specific security issue that is associated with WAP is the WAP GAP. A WAP GAP results from the requirement to change security protocols at the carrier's WAP gateway from

the wireless WTLS to Secure Sockets Layer (SSL) for use over the wired network. At the WAP gateway, the transmission, which is protected by WTLS, is decrypted and then re-encrypted for transmission using SSL. Thus, the data is temporarily in the clear on the gateway and can be compromised if the gateway is not adequately protected. To address this issue, the WAP Forum has put forth specifications that will reduce this vulnerability and support e-commerce applications. These specifications include WMLScript Crypto Library and the WAP Identity Module (WIM). The WMLScript Crypto Library supports end-to-end security by providing for cryptographic functions to be initiated on the WAP client from the Internet content server. These functions include digital signatures originating with the WAP client and the encryption and decryption of data. The WIM is a tamper-resistant device, such as a smart card, that cooperates with WTLS and provides cryptographic operations during the handshake phase. A third alternative is to use a client proxy server that communicates authentication and authorization information to the wireless network server.

- ♦ **Transport layer**—Supports the Wireless Datagram Protocol (WDP), which provides an interface to the wireless networks. It supports network protocols such as GSM, CDMA, and TDMA. It also performs error correction.

The Public Key Infrastructure (PKI) for mobile applications provides for the encryption of communications and mutual authentication of the user and application provider. One concern associated with the mobile PKI relates to the possible time lapse between the expiration of a public key certificate and the reissuing of a new valid certificate and associated public key. This “dead time” may be critical in disasters or in time-sensitive situations. One solution to this problem is to generate one-time keys for use in each transaction.

## Summary

This chapter reviews the electromagnetic spectrum and focuses on the UHF band for cellular phone communications. The major components of the cellular phone network are described, including the mobile station, cell tower, subscriber identity module, base transceiver station, and mobile switching center. TDMA, FDMA, and CDMA technologies are explained along with a subset of CDMA, spread spectrum technology. In particular, DSS, FSS, and OFDM spread spectrum implementations are discussed. The different generations of cellular systems development, including AMPS, TACS, NMT, GSM, UMTS, and IMT-2000 are reviewed. The 802.11 wireless LAN standard, including its various upgrades and instantiations are explained and summarized, including 802.11, 802.11a, 802.11b, 802.11g, and 802.11i. The related 802.11 wireless security issues are explored and the various solutions to the original 802.11 WEP security deficiencies are developed. In conclusion, Bluetooth piconets are introduced and the WAP protocols are detailed.



# Network Architecture Fundamentals

---

**N**etwork communication has been a very significant development over the past 25 years. In particular, the 1990s saw a huge expansion of public access and public-oriented communication networks that had the ability to bind the entire world into a single network. Current networking technology has its roots in military and academic research projects initiated in the 1970s. Thus, networks are no longer a prerogative for the exclusive, but an essential tool and object in the routine life of everybody. The architecture of public networks and, more important, the Internet is very complex and sophisticated. Some of the vital applications and components incorporated into present day networks include the following:

- ◆ Web browsing
- ◆ File transfers
- ◆ E-mail
- ◆ Remote logins
- ◆ Multimedia
- ◆ Telephony
- ◆ Security services

Organizations such as the IETF, IEEE, and so on continually endeavor to enhance these vital components of public networks. This chapter focuses on some of the basic network components used in the present Internet and other network technology (most important, network security services).

## CHAPTER 13



### In This Chapter

Understanding the different types of networks

Reviewing Network Address Translation

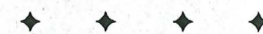
Discussing basic network architecture issues

Understanding subnets and switching

Analyzing firewalls

Reviewing intrusion detection

Understanding common attacks



## Network Segments

Over the past few years, there has been a heavy integration of network technologies, which has created highly unified and global network architectures. Yet business, commercial, and military requirements demand segregation of network segments into authorized domains or network segments. The boundaries of such network segments are established by devices capable of regulating and controlling the flow of packets into and out of the segment, including the following:

- ♦ Routers
- ♦ Switches
- ♦ Hubs
- ♦ Bridges
- ♦ Multi-homed gateways

These segments can be theoretically classified into the following:

- ♦ Public networks
- ♦ Semi-private networks
- ♦ Private networks

### Public networks

Public networks allow accessibility to everyone. The common Internet is a perfect example of a public network. On public networks there is a huge amount of trivial and unsecured data. Users normally pay for public network services and security controls on these networks are weak. Most of the networks you find at Internet cafés, airports, hospitals, shopping malls, and so on are examples of public access networks. Typically, security measures for public access networks are quite restricted. A one-time password would be all that is required to log into publicly available machines and public access networks. Despite the lack of security, large volumes of unprotected data are transmitted worldwide over public networks, because of their convenience and the variety of services they provide.

### Semi-private networks

Semi-private networks sit between public networks and private networks. From a security standpoint, a semi-private network may carry confidential information but under some regulations. Semi-private networks are most often exclusive subnets of

large public networks such as the Internet. Large peer-to-peer networks that are designed to handle and share exclusive information (usually multimedia) among its users can also be classified under semi-private networks. A virtual private network uses public networks optimized with security features that only privileged users can use successfully.

## Private networks

Private networks are organizational networks that handle confidential and propriety data. Each organization at every geographical location may own a private network. If the organization is spread over vast geographical distances, the private networks present at each location may be interconnected through the common Internet or other public networks. Generally, most commercial organizations prefer not to lay down dedicated lines over vast geographical distances, mainly due to cost factors. Private networks may have exclusive addressing and protocols and do not have to be compatible with the Internet. Address translation schemes and various tunneling protocols could be used to have incompatible private and public networks interoperate.

## Perimeter Defense

In most cases, internal networks are composed of various network component blocks. Following are the most important of these:

- ♦ Application servers
- ♦ Proxy servers
- ♦ Middleware servers
- ♦ Data servers
- ♦ Presentation servers

Securing such enormous processing units often requires security solutions to be highly fortified at the network in addition to using individual server-based security systems. In most common environments, firewalls would be placed at the terminal ends of every network segment. Firewalls (independent or combined with routers) can be ideal choices for securing network perimeters. Demilitarized zones can be defined around the periphery for enhanced security features. Specialized application proxies normally placed at the boundaries of network environments can also function as perimeter defense systems. Figure 13-1 shows a comprehensive view of a network protected by perimeter systems (usually firewalls).

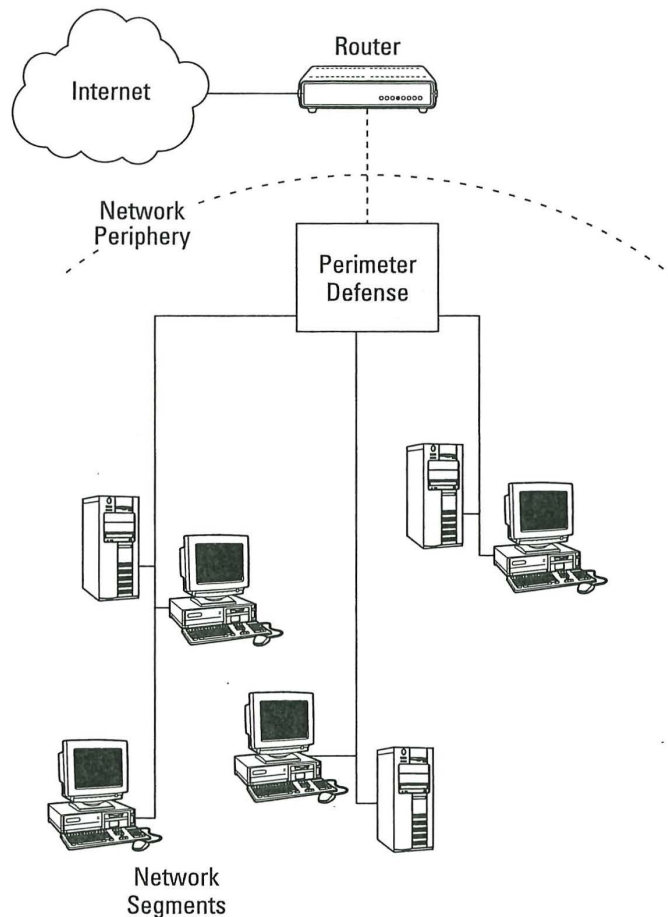


**Figure 13-1:** Perimeter defense strategies employed on various segments for an internal network

## Network Address Translation

Network Address Translation (NAT) is a scheme employed by organizations to defy the address deficiency of IPv4 networking. It basically translates private addresses that are normally internal to a particular organization into routable addresses on public networks such as the Internet. In particular, NAT is a method of connecting multiple computers to the Internet (or any other IP network) using one IP address. Though NAT's main goal is to increase the scope of IP addresses (this necessity is addressed to a great extent by IP-v6 network architectures where there is an abundance of network addresses), security is an essential attribute that can potentially be achieved by NAT.

NAT complements the use of firewalls in providing an extra measure of security for an organization's internal network. Usually, hosts from inside the protected networks (with private address) are able to communicate with the outside world, but systems that are located outside the protected network have to go through the NAT boxes to reach internal networks. Moreover, NAT allows an organization to use fewer IP addresses in making entire networks operational, which aids in confusing attackers as to which particular host they are targeting; in this way security dimensions are increased. Many denial-of-service attacks such as SYN flood and ping of death can be prevented using NAT technology.

The main feature in NAT is the translation table that the NAT box maintains. A NAT box might be implemented with a laptop computer and the appropriate network interface cards. The translation table maps external unique IP addresses to internal private IP addresses. Normally, this mapping is not one-to-one. To conserve address space, a single global IP address may be mapped to more than one private IP address. Typically, port associations (on the NAT boxes) are created to achieve multiple mapping of public and private addresses. Any packets from the outside attempting to reach a particular host on the private network get routed with the NAT-specified global address. It becomes the responsibility of the NAT software to look up the translation table to find out the particular private address to which the packet has to be routed. Figure 13-2 shows the technique involved in NAT. Normally, translation tables are built using three methods:

- ♦ **Static**—In this configuration, the relationships among the global and private IP addresses are fixed.
- ♦ **Dynamic outbound packets**—In this mode, the translation tables get updated automatically as outbound packets are processed from the private network.
- ♦ **Domain name lookups**—When packets from the external Internet make domain name lookups of hosts inside the private network, the domain name lookup software takes the responsibility of updating the NAT tables.

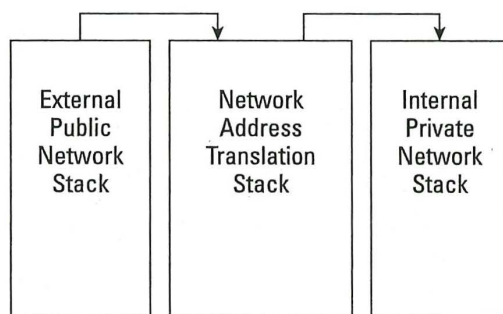


Figure 13-2: The NAT methodology

## Basic Architecture Issues

Network architecture consists of various components. Each component has its own functionalities and responsibilities in effecting the various tasks involved in network communication. Many functions, such as quality of services, remote logins, security, and so on, require specialized components intended for a specific function or combination of functions. This section deals with certain building blocks (or components) that make the realization of these functionalities possible.

- ♦ **Demilitarized zone**—A demilitarized zone (DMZ) is a noncritical yet secure region generally designed at the periphery of the internal and external networks. Normally, the configuration of a DMZ is such that it is either separated by a firewall from the external network or sandwiched between two firewalls, one at the external periphery and the other at the internal. Figure 13-3 shows a demilitarized zone setup for a Web server application.

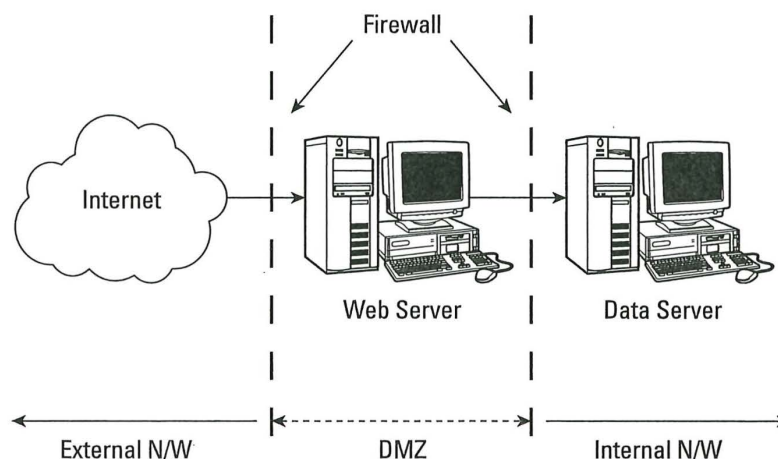


**Figure 13-3:** A Web server in a DMZ

- ♦ **Modems**—Modems (*modulators-demodulators*) are functional end user equipment that are used to transmit digital signals over analog telephone lines. Thus, digital signals are converted by the modem into analog signals of different frequencies and transmitted to a modem at the receiving location. The receiving modem performs the reverse transformation and provides a digital output to a device connected to a modem, usually a computer. The digital data is usually transferred to or from the modem over a serial line through an industry standard interface, RS-232. Many telephone companies (who offer DSL services) and cable operators (offering Internet cables) use modems as end terminals for identification and recognition of home and personal users.

- ♦ **Hubs**—A hub is a device for connecting multiple LAN devices together. It also performs as a repeater in that it amplifies signals that deteriorate after traveling long distances over connecting cables. Hubs do not perform packet filtering or any addressing functions.
- ♦ **Bridges**—Bridges are devices that are used to connect two or more hosts or network segments together. Bridges work only at the physical and link layer level and use the hardware Media Access Control (MAC) addresses for transferring frames. The basic role of bridges in network architecture is storing and forwarding frames between the different segments that it connects. Typically, a single bridge can have more than two ports, which means that more than two networking elements can be combined to communicate with each other using a single bridge.
- ♦ **Switches**—Network switches generally have a more intelligent role than hubs. Strands of local area networks (LANs), normally belonging to the same collision domain, are usually connected using switches. Mainly working on the Layer 2 frames (Data Link layer), they are equipped with the ability to read the packet headers and process appropriately. Generally, switches have the ability to read the incoming packets' hardware addresses to transmit them to the appropriate destination. Frames could be lost if a particular host is either unreachable or disconnected. Switches play an important role in regulating traffic regulations on the segments they interconnect. Because switches directly connect one of many devices connected to its input ports to one of many devices connected to its output ports, a switch necessarily has a larger number of input/output interface cards than bridges.
- ♦ **Routers**—Routers are one of the main components of the chassis of networks. Routers are mainly involved in the transmission of packets to their destinations, routing a path through the sea of interconnected network devices. The packets are removed from the incoming frames and individually analyzed. Routers normally work at the Network layer (Layer 3 of the OSI model), which assigns the much familiar IP addresses. IP addresses are software or logical addresses that point to a logical location or connection on a network. Worldwide IP addresses of any connection are unique unless they are not defined for private use. Routers normally process connectionless network layer packets, otherwise known as *datagrams*. Packets originating from the same source and reaching the same destination as part of the same connection may be routed through different routes. IP packets are equipped with header fields that give the routers knowledge of where it originated from and its intended destination. There is a plethora of work on routing algorithms. Routing algorithms are the knowledge base of the routers. Routers, which hop packets from one point to the other, use the routing algorithms for effecting their decisions. Bellman-Ford, Distance vector, OSPF, and so on are some well-known routing algorithms used on the Internet. Proprietary organizations can have their own implementations of these routing algorithms.

♦ **Gateways**—As you move up in the network protocol stack, you find gateways. Gateways normally work around the transport and session layers of the OSI model. Typically, on and above the transport layer, there are numerous protocols and standards proposed by different vendors. The Internet uses the Transmission Control Protocol (TCP) at the Transport layer, but other protocols (mostly proprietary) do exist in this layer. Some of the other Transport layer protocols include:

- X.25
- Systems Network Architecture (SNA)
- Asynchronous Transfer Mode (ATM)

Gateways are used when dealing with multiprotocol Transport layers and above. All of the following are important specifications at the gateway level of network architecture:

- Form factor
- Network type
- Performance
- Port
- Processor specifications
- Memory
- Features

Common features for network gateways include stackable, rack mount, LED indicators, integrated firewall, and Virtual Private Networks. Application layer gateways are ideal choices for integrating multiple security services. Firewalls and intrusion detection systems are ideally suited to be at this layer of the network stack.

## Subnetting, Switching, and VLANs

Addressing is one of the main issues that network architecture is concerned with. Two major addresses are involved with all the major public access networks such as the Internet. They are the hardware (MAC) address and IP addresses. MAC addresses are used to uniquely identify individual machines as hosts and are not as important from a routing standpoint. They are hard coded into the network card and most people are not even aware what they are. The most important addresses seen all over the Internet are IP addresses. An IP address points to a logical entity on the Internet and is normally unique in identifying itself. Addressing in IP version 4 (IPv4) uses 32 bits. Rather than providing for random addresses for incoming hosts, the Internet follows a particular hierarchy that could be logically used for various vital services such as routing, name resolution, and so on.

IPv4 divides the whole address range into five major classes: Classes A, B, C, D, and E. The 32-bit address is split into three distinct regions, as follows:

- ♦ Class-ID
- ♦ Net-ID
- ♦ Host-ID

The class-ID is usually represented in the first to the fourth bits of an IP address. This is followed by the net-ID and then the host-ID. With such a scheme, it becomes easy to discern to what network and class a particular IP address's host belongs with the inspection of the address. This is highly useful when routing and resource discovery are in the picture. Figure 13-4 shows an IPv4 addressing representation. Classes A, B, and C are typically allotted to individual host depending on how big the particular network segment is. Classes D and E are for multicasting and future use, respectively. Class A encompasses a relatively small number of networks compared to class B and class C (which holds the most number of network addresses). Huge networks are generally addressed using class A, rather than class B and class C (which can hold the minimum number of hosts among the different classes).

#### Class

|   |      |           |         |
|---|------|-----------|---------|
| A | 0    | net-id    | host-id |
| B | 10   | net-id    | host-id |
| C | 110  | net-id    | host-id |
| D | 1110 | Multicast |         |
| E | 1111 | Reserved  |         |

**Figure 13-4:** IPv4 class addressing

Subnetting is a technique followed in network architecture to reduce the burden of the routers in maintaining routing tables. Class B networks have approximately 64,000 different host addresses, though most organizations registered for class B addresses do not require that many hosts as they do not employ that many systems. In this case, many addresses get wasted and cumulatively there is a dearth of IP addresses with IPv4, because each network segment typically demands its own net-ID in the addressing scheme. Subnetting schemes are built in such a way that the traditional net-id/host-id barriers are broken so that any combination of addressing would fit per the size of the network. A subnet mask (a series of 1s followed by a series of 0s) is applied to the IP address to determine which subnet a particular destination is on in a particular network. This is a highly recommended feature for conserving addressing space and controlling router table explosions.

Switching is then done based on the resolved subnet address and host address. A new technique called *classless interdomain routing (CIDR)* offers new features in this regard.

Switching techniques can be used to incorporate interesting architectural twists in networking. One such network architecture concept is the virtual local area network (VLAN). This architecture is useful in situations where organizations have geographically distributed divisions or departments and would still like to place all the entities under a single network segment. VLANs make this possible. VLANs use switching to achieve same broadcast domain relationships. Figure 13-5 shows how virtual networking for four different LAN segments (A, B, C, and D) is possible using switching technologies.

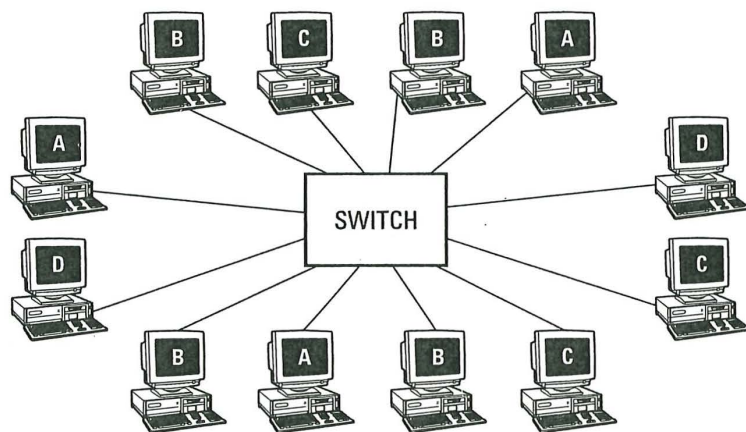


Figure 13-5: VLAN of four LAN segments (A, B, C, D) using a switch

## Address Resolution Protocol and Media Access Control Addresses

The Address Resolution Protocol (ARP) and the Media Access Control are basically Layer 2 and 3 control issues in the Transmission Control Protocol/ Internet Protocol (TCP/IP) stack. It is impossible to associate individual IP addresses to every physical machine that has been manufactured. Instead, every machine can be uniquely identified by a second address called the Media Access Control (MAC) address, commonly known as a hardware address. The ARP is used to determine the 48-bit MAC address corresponding to a 32-bit IP address.

Normally, one would intend to use static translation maps to achieve this process, but the enormity of IP and MAC addresses available that must be maintained make it almost impossible for static maintenance. TCP/IP designers have a novel way of dynamically solving this problem using ARP. Normally, any host that requires determining the hardware addresses of a particular IP address broadcasts address request packets all over the domain it is in. The host whose actual IP address is found on the requests replies to the intended source of the presence of the particular host with its IP and MAC address. If no host can be determined, default routing actions are taken. This information can be used for further communication between the two hosts. To speed up the resolution process, a small segment of memory is maintained to store short-term IP and MAC address mapping of other hosts on the network. This memory is called the ARP cache.

The ARP cache is the first thing that a host looks at for address resolution before it can start issuing the broadcast process. Because two parties that were involved in communication at some point in time are likely to communicate at a subsequent time, cache memory becomes very vital in speeding up processing in the protocol. The ARP caches are required to be refreshed at regular intervals of time (approximately every 25 or 30 minutes). This is very vital in determining changes that have occurred in recent history. In hosts where storage mechanisms are not available, a similar protocol called the Reverse Address Resolution Protocol (RARP) is used for IP address determination at bootup. Figure 13-6 shows the pseudo data structure followed in ARP messages.

| Hardware Type             |        | Protocol Type |
|---------------------------|--------|---------------|
| HD LEN                    | IP LEN | Operation     |
| Sender Hardware Address   |        |               |
| Sender IP Address         |        |               |
| Receiver Hardware Address |        |               |
| Receiver Hardware Address |        |               |

**Figure 13-6:** ARP message data structure

## Dynamic Host Configuration Protocol and Addressing Control

The Dynamic Host Configuration Protocol (DHCP) is a commonly employed technique to distribute IP addresses on networks where static address allocation may not be appropriate. Auto-configuration IP address distribution techniques such as DHCP may be very easily done where centralized servers are commonly available. However, there is also a need for autoconfiguration in architectures where individual clients communicate directly with each other and there is no centralized service. An example of this type of communication in wireless networks is the ad hoc network. Ad hoc networks are infrastructureless, multihop wireless networks that can be deployed without any pre-existing setup. Ad-hoc networks are mobile in nature and any node can join and leave the network at any time. Due to their mobility, ad-hoc networks must be able to configure themselves without human intervention. Configuration (such as address assignment) of a node in such a network is a critical issue. The nodes in an ad-hoc network are basically a plug-and-play type, wherein any node can enter and exit a network configuration without much intervention from other nodes in the network. Zero-Configuration networks have a similar setup, but the main problem that arises when applying the techniques followed in Zero-Configuration networks to ad hoc networks is that a set of reserved IP address (169.254) exist for use in such networks, which may not be feasible for ad hoc network set ups. This section reviews some of the existing techniques for dynamic host configuration in ad hoc networks and their applicability.

The best method to assign IP addresses to network nodes in any network would be to assign them statically for each node in the network. This process could become highly tedious and vulnerable to errors for large-scale networks. This is one of the main reasons that DHCP was designed to automate the address assignment in IP-based networks. There are basically two modes of address assignment to a network configuration: stateful and stateless. In the stateful mode of configuration, a predefined set of IP addresses is dynamically issued either permanently or on lease to the individual nodes. In the stateless configuration (usually applied to IPv6 networks) a function of the hardware address is used to assign an IP address. For a variety of reasons, one generally cannot assign an IP address based on the MAC address even in IPv6 networks where one-to-one mapping of IP and MAC addresses is possible. For example, security reasons constrain the IP address to originate from a distinct set, and unavailability of unique hardware addresses is another reason. The fact that ad-hoc networks are dynamically configured, combined, and divided makes them quite unsuitable for stateless modes of auto-configuration. Moreover, the IETF has recommended that stateful autoconfiguration be implemented for ad-hoc networks. The mobile nature of ad-hoc networks makes devising a suitable mechanism for dynamic host configuration with a predefined set of IP addresses very difficult. It should be noted that mobility is quite different from connectivity. A network can be independent of physical hardware connectivity (a wireless medium, for example) and yet be nonmobile. Ad-hoc wireless networks combine these two aspects, independent connectivity and mobility, making it quite complex for most present-day configuration protocols.