JAMES MICHAEL STEWART

CompTIA Security+ REVIEW GUIDE

Fourth Edition

Covers 100% of exam objectives, including threats, attacks, and vulnerabilities; technologies and tools; architecture and design; identity and access management; risk management; cryptography and PKI; and much more...

Includes interactive online learning environment and study tools with:

+ 2 customer practice exams
+ Over 300 electronic flashcards
+ Searchable key term glossary

CompTIA Approved Quality Content

CompTIA Authorized Partner Gold

EXAM SY0-501

SYBEX
A Wiley Brand

CompTIA®

Security+®

Review Guide

CompTIA®

Security+® SY0-501

Review Guide

James Michael Stewart

SYBEX®
A Wiley Brand

Senior Acquisitions Editor: Kenyon Brown
Development Editor: Jim Compton
Technical Editor: Josh More
Senior Production Editor: Christine O'Connor
Copy Editor: Elizabeth Welch
Editorial Manager: Mary Beth Wakefield
Production Manager: Kathleen Wisor
Executive Editor: Jim Minatel
Book Designers: Judy Fung and Bill Gibson
Proofreader: Louise Watson, Word One New York
Indexer: John Sleeva
Project Coordinator, Cover: Brent Savage
Cover Designer: Wiley
Cover Image: Getty Images Inc./Jeremy Woodhouse

Published simultaneously in Canada

ISBN: 978-1-119-41694-4

ISBN: 978-1-119-41695-1 (ebk.)

ISBN: 978-1-119-41693-7 (ebk.)

Manufactured in the United States of America

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 Web site 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 Web site may provide or recommendations it may make. Further, readers should be aware that Internet Web sites 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 (877) 762-2974, outside the U.S. at (317) 572-3993 or fax (317) 572-4002.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at http://booksupport.wiley.com. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2017960021

10 9 8 7 6 5 4 3 2 1

To Catharine Renee Stewart:
You are my all and my everything, I love you.

Acknowledgments

Thanks to all those at Sybex/Wiley who continue to allow me to do what I enjoy most—impart knowledge to others. Thanks to Kenyon Brown, acquisitions editor, and the whole Sybex crew for professional juggling services adequately rendered. Thanks to my development editor, Jim Compton, and my technical editor, Josh More. To my wonder woman of a wife, Cathy, and my amazing kids, Slayde and Remi—you make life exciting and sweet! To my mom, Johnnie: thanks for your love and consistent support. To Mark: go away or I shall taunt you a second time! And finally, as always, to Elvis: I hear that you amazed the Beatles with your television remote control; it is good to be the King!

About the Author

James Michael Stewart has been working with computers and technology since 1983 (although officially as a career since 1994). His work focuses on Internet technologies, professional certifications, and IT security. Recently, Michael has been teaching job skill and certification courses, such as CISSP, CEH, CHFI, and Security+. Michael has contributed to many Security+ focused materials, including exam preparation guides, practice exams, DVD video instruction, and courseware. In addition, Michael has co-authored numerous books on other security and IT certification and administration topics. He has developed certification courseware and training materials and presented these materials in the classroom. He holds numerous certifications, including Sec+, CISSP, and CEH. Michael graduated in 1992 from the University of Texas at Austin with a bachelor's degree in philosophy. Despite his degree, his computer knowledge is self-acquired, based on seat-of-the-pants, hands-on, "street smarts" experience. You can reach Michael by email at `michael@impactonline.com`.

Contents at a Glance

Contents

CompTIA

CompTIA Security+

Becoming a CompTIA Certified IT Professional is Easy

It's also the best way to reach greater professional opportunities and rewards.

Why Get CompTIA Certified?

Growing Demand

Labor estimates predict some technology fields will experience growth of over 20% by the year 2020.* CompTIA certification qualifies the skills required to join this workforce.

Higher Salaries

IT professionals with certifications on their resumes command better jobs, earn higher salaries, and have more doors open to new multi-industry opportunities.

Verified Strengths

91% of hiring managers indicate CompTIA certifications are valuable in validating IT expertise, making certification the best way to demonstrate your competency and knowledge to employers.**

Universal Skills

CompTIA certifications are vendor neutral—which means that certified professionals can proficiently work with an extensive variety of hardware and software found in most organizations.

Learn

Learn more about what the exam covers by reviewing the following:

- Exam objectives for key study points.
- Sample questions for a general overview of what to expect on the exam and examples of question format.
- Visit online forums, like LinkedIn, to see what other IT professionals say about CompTIA exams.

Certify

Purchase a voucher at a Pearson VUE testing center or at CompTIAstore.com.

- Register for your exam at a Pearson VUE testing center.
- Visit pearsonvue.com/CompTIA to find the closest testing center to you.
- Schedule the exam online. You will be required to enter your voucher number or provide payment information at registration.
- Take your certification exam.

Work

Congratulations on your CompTIA certification!

- Make sure to add your certification to your resume.
- Check out the CompTIA Certification Roadmap to plan your next career move.

Learn more: **Certification.CompTIA.org/securityplus**

* Source: CompTIA 9th Annual Information Security Trends study: 500 U.S. IT and Business Executives Responsible for Security
** Source: CompTIA Employer Perceptions of IT Training and Certification

Introduction

The Security+ certification program was developed by the Computer Technology Industry Association (CompTIA) to provide an industry-wide means of certifying the competency of computer service technicians in the basics of computer security. The Security+ certification is granted to those who have attained the level of knowledge and security skills that show a basic competency in the security needs of both personal and corporate computing environments. CompTIA's exam objectives are periodically updated to keep their exams applicable to the most recent developments. The most recent update, labeled SY0–501, occurred in late 2017. This book focuses on these newly revised certification objectives.

What Is Security+ Certification?

The Security+ certification was created to offer an introductory step into the complex world of IT security. You need to pass only a single exam to become Security+ certified. However, obtaining this certification doesn't mean you can provide realistic security services to a company. In fact, this is just the first step toward true security knowledge and experience. By obtaining Security+ certification, you should be able to acquire more security experience in order to pursue more complex and in-depth security knowledge and certification.

For the latest pricing on the exam and updates to the registration procedures, please visit `www.vue.com`. If you have further questions about the scope of the exams or related CompTIA programs, refer to the CompTIA website at `www.comptia.org`.

Is This Book for You?

CompTIA Security+ Review Guide: SY0-501 is designed to be a succinct, portable exam review guide. It can be used in conjunction with a more typical full-sized study guide, such as Wiley's *CompTIA Security+ Study Guide: SY0-501* (ISBN: 978-1260026054), with computer-based training (CBT) courseware and a classroom/lab environment, or as an exam review for those who don't feel the need for more extensive (and/or expensive) test preparation. It isn't our goal to give away the answers, but rather to identify those topics on which you can expect to be tested and to provide sufficient focused coverage of these topics.

Perhaps you've been working with information technologies for years. The thought of paying lots of money for a specialized IT exam-preparation course probably doesn't sound appealing. What can they teach you that you don't already know, right? Be careful, though—many experienced network administrators have walked confidently into the test center only to walk sheepishly out of it after failing an IT exam. After you've finished reading this book, you should have a clear idea of how your understanding of the technologies involved matches up with the expectations of the Security+ test makers.

Or perhaps you're relatively new to the world of IT, drawn to it by the promise of challenging work and higher salaries. You've just waded through an 800-page study guide or taken a weeklong class at a local training center. Lots of information to keep track of, isn't there? Well, by organizing this book according to CompTIA's exam objectives, and by breaking up the information into concise, manageable pieces, we've created what we think is the handiest exam review guide available. Throw it in your backpack and carry it to work with you. As you read the book, you'll be able to quickly identify those areas you know best and those that require a more in-depth review.

How Is This Book Organized?

This book is organized according to the official objectives list prepared by CompTIA for the Security+ exam. The chapters correspond to the six major domains of objective and topic groupings. The exam is weighted across these six topical areas or domains as follows:

1.0 Threats, Attacks and Vulnerabilities (21%)

2.0 Technologies and Tools (22%)

3.0 Architecture and Design (15%)

4.0 Identity and Access Management (16%)

5.0 Risk Management (14%)

6.0 Cryptography and PKI (12%)

Within each chapter, the top-level exam objectives from each domain are addressed in turn and in order according to the official exam objectives directly from CompTIA. In addition to a thorough review of each objective, every chapter includes two specific features: Exam Essentials and Review Questions.

Exam Essentials At the end of each top-level objective section, you're given a short list of topics that you should explore fully before taking the test. Included in the Exam Essentials areas are notations of the key information you should have taken from that section, or from the corresponding content in the *CompTIA Security+ Study Guide*.

Review Questions This feature ends every chapter and provides 20 questions to help you gauge your mastery of the chapter.

Interactive Online Learning Environment and Test Bank

We've included several additional test-preparation features on the interactive online learning environment and test bank. These tools will help you retain vital exam content as well as prepare you to sit for the actual exams:

NOTE Go to www.wiley.com/go/sybextestprep to register and gain access to this interactive online learning environment and test bank with study tools.

Sample Tests In this section of the online test bank, you'll find the chapter tests, which present all the review questions from the end of each chapter, as well as two more practice tests of 90 questions each. Use these questions to test your knowledge of the study guide material. The online test bank runs on multiple devices.

Electronic Flashcards Questions are provided in digital flashcard format (a question followed by a single correct answer). You can use the flashcards to reinforce your learning and provide last-minute test prep before the exam.

Glossary of Terms in PDF We have included a very useful glossary of terms in PDF format so you can easily read it on any computer. If you have to travel and brush up on any key terms, you can do so with this useful resource.

Tips for Taking the Security+ Exam

Here are some general tips for taking your exam successfully:

- Bring two forms of ID with you. One must be a photo ID, such as a driver's license. The other can be a major credit card or a passport. Both forms must include a signature.
- Arrive early at the exam center so you can relax and review your study materials.
- Read the questions carefully. Don't be tempted to jump to an early conclusion. Make sure you know exactly what the question is asking.
- Read each question twice, read the answer options, and then read the question again before selecting an answer.
- You can move forward and backward through the exam, but only one question at a time. You can only move forward once you have given the current question an answer. Only after seeing the Review Page after the last question can you jump around questions at random.
- Don't leave any unanswered questions. Unanswered questions give you no opportunity for guessing correctly and scoring more points.
- Watch your clock. If you have not seen your last question when you have 5 minutes left, guess at the remaining questions.
- There will be questions with multiple correct responses. When there is more than one correct answer, a message on the screen will prompt you to either "Choose two" or "Choose all that apply." Be sure to read the messages displayed so you know how many correct answers you must choose.

- Questions needing only a single correct answer will use radio buttons to select an answer, whereas those needing two or more answers will use check boxes.
- When answering multiple-choice questions you're not sure about, use a process of elimination to get rid of the obviously incorrect answers first. Doing so will improve your odds if you need to make an educated guess.
- Try to expand your perspective from your own direct experience. Often the writers of the exam questions are from large enterprises; if you only consider answers in light of a small company or as an individual, you might not determine the correct answer.
- You can mark or flag a question to indicate you want to review it again before ending the exam. Flagged questions will be highlighted on the Review page.
- For the latest pricing on the exams and updates to the registration procedures, visit CompTIA's website at `www.comptia.org`.

Performance-Based Questions

CompTIA has begun to include performance-based (scenario-based) questions on its exams. These differ from the traditional multiple-choice questions in that the candidate is expected to perform a task or series of tasks. Tasks could include filling in a blank, answering questions based on a video or an image, reorganizing a set into an order, placing labels on a diagram, filling in fields based on a given situation or set of conditions, or setting the configuration on a network security management device. Don't be surprised if you are presented with a scenario and asked to complete a task. The performance-based questions are designed to be more challenging than standard multiple choice questions and thus are also worth more points. Take the time to answer these carefully. For an official description of performance-based questions from CompTIA, visit `http://certification.comptia.org/news/2012/10/09/What_Is_A_Performance-Based_Question.aspx` and `https://certification.comptia.org/testing/about-testing/performance-based-questions-explained` (this second link is from the CompTIA Security+ information page, so you can follow it from there instead of typing it in).

Exam Specifics

The Security+ SY0-501 exam consists of up to 90 questions with a time allotment of 90 minutes for the exam itself. Additional time is provided for the pre-exam elements, such as the NDA, and the post-exam survey. If you are assigned only multiple choice questions, then you will have the maximum of 90 questions. If you are assigned performance-based questions (which is most likely), then you will have fewer than 90 total questions. It is fairly common to have 5 or 6 performance-based questions and about 70 multiple choice questions, for a total of 75 or so questions. However, you could be assigned 8 or more performance-based questions with about 50 multiple choice questions, for a total of 55 questions. To pass, you must score at least 750 points on a scale of 100–900 (effectively 81.25%). At the completion of your test, you will receive a printout of your test results. This report will show your score and the objective topics about which you missed a question.

NOTE Although there is no clear statement from CompTIA, there seem to be some questions on the exam that are included for evaluation purposes but do not count toward your score. These questions are likely on topics not currently listed in the SY0-501 objectives list, and they will appear at random within your exam and will not be marked in any way.

NOTE These details are subject to change. For current information, please consult the CompTIA website: `www.comptia.org`.

How to Contact the Publisher

Sybex welcomes feedback on all of its titles. Visit the Sybex website at `www.sybex.com` for book updates and additional certification information. You'll also find forms you can use to submit comments or suggestions regarding this or any other Sybex title.

The Security+ Exam Objectives

For easy reference and clarification, the following is a complete listing of Security+ objectives. Also, we organized this book to correspond with the official objectives list. We use the objective list's order and organization throughout the book. Each domain is covered in one chapter. Each subobjective is a heading within a chapter.

NOTE Exam objectives are subject to change at any time without prior notice and at CompTIA's sole discretion. Please visit the Security+ Certification page of CompTIA's website (`www.comptia.org`) for a link to the most current exam objectives.

Domain 1.0 Threats, Attacks and Vulnerabilities

1.1 Given a scenario, analyze indicators of compromise and determine the type of malware.

- Viruses
  - Crypto-malware
  - Ransomware
  - Worm
  - Trojan

- Rootkit
- Keylogger
- Adware
- Spyware
- Bots
- RAT
- Logic bomb
- Backdoor

1.2 Compare and contrast types of attacks.

- Social engineering
  - Phishing
  - Spear phishing
  - Whaling
  - Vishing
  - Tailgating
  - Impersonation
  - Dumpster diving
  - Shoulder surfing
  - Hoax
  - Watering hole attack
  - Principles (reasons for effectiveness)
    - Authority
    - Intimidation
    - Consensus
    - Scarcity
    - Familiarity
    - Trust
    - Urgency
- Application/service attacks
  - DoS
  - DDoS
  - Man-in-the-middle
  - Buffer overflow

    - Injection
    - Cross-site scripting
    - Cross-site request forgery
    - Privilege escalation
    - ARP poisoning
    - Amplification
    - DNS poisoning
    - Domain hijacking
    - Man-in-the-browser
    - Zero day
    - Replay
    - Pass the hash
    - Hijacking and related attacks
        - Clickjacking
        - Session hijacking
        - URL hijacking
        - Typo squatting
- Driver manipulation
    - Shimming
    - Refactoring
        - MAC spoofing
        - IP spoofing
- Wireless attacks
    - Replay
    - IV
    - Evil twin
    - Rogue AP
    - Jamming
    - WPS
    - Bluejacking
    - Bluesnarfing
    - RFID
    - NFC
    - Disassociation

- Cryptographic attacks
  - Birthday
  - Known plain text/cipher text
  - Rainbow tables
  - Dictionary
  - Brute force
    - Online vs. offline
  - Collision
  - Downgrade
  - Replay
  - Weak implementations

1.3 Explain threat actor types and attributes.

- Types of actors
  - Script kiddies
  - Hacktivist
  - Organized crime
  - Nation states/APT
  - Insiders
  - Competitors
- Attributes of actors
  - Internal/external
  - Level of sophistication
  - Resources/funding
  - Intent/motivation
- Use of open-source intelligence

1.4 Explain penetration testing concepts.

- Active reconnaissance
- Passive reconnaissance
- Pivot
- Initial exploitation
- Persistence
- Escalation of privilege
- Black box

- White box
- Gray box
- Pen testing vs. vulnerability scanning

1.5 Explain vulnerability scanning concepts.

- Passively test security controls
- Identify vulnerability
- Identify lack of security controls
- Identify common misconfigurations
- Intrusive vs. non-intrusive
- Credentialed vs. non-credentialed
- False positive

1.6 Explain the impact associated with types of vulnerabilities.

- Race conditions
- Vulnerabilities due to:
    - End-of-life systems
    - Embedded systems
    - Lack of vendor support
- Improper input handling
- Improper error handling
- Misconfiguration/weak configuration
- Default configuration
- Resource exhaustion
- Untrained users
- Improperly configured accounts
- Vulnerable business processes
- Weak cipher suites and implementations
- Memory/buffer vulnerability
    - Memory leak
    - Integer overflow
    - Buffer overflow
    - Pointer dereference
    - DLL injection

- System sprawl/undocumented assets
- Architecture/design weaknesses
- New threats/zero day
- Improper certificate and key management

Domain 2.0 Technologies and Tools

2.1 Install and configure network components, both hardware- and software-based, to support organizational security.

- Firewall
    - ACL
    - Application-based vs. network-based
    - Stateful vs. stateless
    - Implicit deny
- VPN concentrator
    - Remote access vs. site-to-site
    - IPSec
        - Tunnel mode
        - Transport mode
        - AH
        - ESP
    - Split tunnel vs. full tunnel
    - TLS
    - Always-on VPN
- NIPS/NIDS
    - Signature-based
    - Heuristic/behavioral
    - Anomaly
    - Inline vs. passive
    - In-band vs. out-of-band
    - Rules
    - Analytics
        - False positive
        - False negative

- Router
    - ACLs
    - Antispoofing
- Switch
  - Port security
  - Layer 2 vs. Layer 3
  - Loop prevention
  - Flood guard
- Proxy
  - Forward and reverse proxy
  - Transparent
  - Application/multipurpose
- Load balancer
  - Scheduling
    - Affinity
    - Round-robin
  - Active-passive
  - Active-active
  - Virtual IPs
- Access point
  - SSID
  - MAC filtering
  - Signal strength
  - Band selection/width
  - Antenna types and placement
  - Fat vs. thin
  - Controller-based vs. standalone
- SIEM
  - Aggregation
  - Correlation
  - Automated alerting and triggers
  - Time synchronization
  - Event deduplication
  - Logs/WORM

- DLP
    - USB blocking
    - Cloud-based
    - Email
- NAC
    - Dissolvable vs. permanent
    - Host health checks
    - Agent vs. agentless
- Mail gateway
    - Spam filter
    - DLP
    - Encryption
- Bridge
- SSL/TLS accelerators
- SSL decryptors
- Media gateway
- Hardware security module

2.2 Given a scenario, use appropriate software tools to assess the security posture of an organization.

- Protocol analyzer
- Network scanners
    - Rogue system detection
    - Network mapping
- Wireless scanners/cracker
- Password cracker
- Vulnerability scanner
- Configuration compliance scanner
- Exploitation frameworks
- Data sanitization tools
- Steganography tools
- Honeypot
- Backup utilities
- Banner grabbing
- Passive vs. active

- Command line tools
  - ping
  - netstat
  - tracert
  - nslookup/dig
  - arp
  - ipconfig/ip/ifconfig
  - tcpdump
  - nmap
  - netcat

2.3 Given a scenario, troubleshoot common security issues.

- Unencrypted credentials/clear text
- Logs and events anomalies
- Permission issues
- Access violations
- Certificate issues
- Data exfiltration
- Misconfigured devices
  - Firewall
  - Content filter
  - Access points
- Weak security configurations
- Personnel issues
  - Policy violation
  - Insider threat
  - Social engineering
  - Social media
  - Personal email
- Unauthorized software
- Baseline deviation
- License compliance violation (availability/integrity)
- Asset management
- Authentication issues

2.4 Given a scenario, analyze and interpret output from security technologies.

- HIDS/HIPS
- Antivirus
- File integrity check
- Host-based firewall
- Application whitelisting
- Removable media control
- Advanced malware tools
- Patch management tools
- UTM
- DLP
- Data execution prevention
- Web application firewall

2.5 Given a scenario, deploy mobile devices securely.

- Connection methods
  - Cellular
  - WiFi
  - SATCOM
  - Bluetooth
  - NFC
  - ANT
  - Infrared
  - USB
- Mobile device management concepts
  - Application management
  - Content management
  - Remote wipe
  - Geofencing
  - Geolocation
  - Screen locks
  - Push notification services
  - Passwords and pins
  - Biometrics

  - Context-aware authentication
  - Containerization
  - Storage segmentation
  - Full device encryption
- Enforcement and monitoring for:
  - Third-party app stores
  - Rooting/jailbreaking
  - Sideloading
  - Custom firmware
  - Carrier unlocking
  - Firmware OTA updates
  - Camera use
  - SMS/MMS
  - External media
  - USB OTG
  - Recording microphone
  - GPS tagging
  - WiFi direct/ad hoc
  - Tethering
  - Payment methods
- Deployment models
  - BYOD
  - COPE
  - CYOD
  - Corporate-owned
  - VDI

2.6 Given a scenario, implement secure protocols.

- Protocols
  - DNSSEC
  - SSH
  - S/MIME
  - SRTP
  - LDAPS

  - FTPS
  - SFTP
  - SNMPv3
  - SSL/TLS
  - HTTPS
  - Secure POP/IMAP
- Use cases
  - Voice and video
  - Time synchronization
  - Email and web
  - File transfer
  - Directory services
  - Remote access
  - Domain name resolution
  - Routing and switching
  - Network address allocation
  - Subscription services

Domain 3.0 Architecture and Design

3.1 Explain use cases and purpose for frameworks, best practices and secure configuration guides.

- Industry-standard frameworks and reference architectures
  - Regulatory
  - Non-regulatory
  - National vs. international
  - Industry-specific frameworks
- Benchmarks/secure configuration guides
  - Platform/vendor-specific guides
    - Web server
    - Operating system
    - Application server
    - Network infrastructure devices
  - General purpose guides

- Defense-in-depth/layered security
    - Vendor diversity
    - Control diversity
        - Administrative
        - Technical
    - User training

3.2 Given a scenario, implement secure network architecture concepts.

- Zones/topologies
    - DMZ
    - Extranet
    - Intranet
    - Wireless
    - Guest
    - Honeynets
    - NAT
    - Ad hoc
- Segregation/segmentation/isolation
    - Physical
    - Logical (VLAN)
    - Virtualization
    - Air gaps
- Tunneling/VPN
    - Site-to-site
    - Remote access
- Security device/technology placement
    - Sensors
    - Collectors
    - Correlation engines
    - Filters
    - Proxies
    - Firewalls
    - VPN concentrators
    - SSL accelerators

    - Load balancers
    - DDoS mitigator
    - Aggregation switches
    - Taps and port mirror
- SDN

3.3 Given a scenario, implement secure systems design.

- Hardware/firmware security
    - FDE/SED
    - TPM
    - HSM
    - UEFI/BIOS
    - Secure boot and attestation
    - Supply chain
    - Hardware root of trust
    - EMI/EMP
- Operating systems
    - Types
        - Network
        - Server
        - Workstation
        - Appliance
        - Kiosk
        - Mobile OS
    - Patch management
    - Disabling unnecessary ports and services
    - Least functionality
    - Secure configurations
    - Trusted operating system
    - Application whitelisting/blacklisting
    - Disable default accounts/passwords
- Peripherals
    - Wireless keyboards
    - Wireless mice

  - Displays
  - WiFi–enabled MicroSD cards
  - Printers/MFDs
  - External storage devices
  - Digital cameras

3.4 Explain the importance of secure staging deployment concepts.

- Sandboxing
- Environment
  - Development
  - Test
  - Staging
  - Production
- Secure baseline
- Integrity measurement

3.5 Explain the security implications of embedded systems.

- SCADA/ICS
- Smart devices/IoT
  - Wearable technology
  - Home automation
- HVAC
- SoC
- RTOS
- Printers/MFDs
- Camera systems
- Special purpose
  - Medical devices
  - Vehicles
  - Aircraft/UAV

3.6 Summarize secure application development and deployment concepts.

- Development life-cycle models
  - Waterfall vs. Agile

- Secure DevOps
    - Security automation
    - Continuous integration
    - Baselining
    - Immutable systems
    - Infrastructure as code
- Version control and change management
- Provisioning and deprovisioning
- Secure coding techniques
    - Proper error handling
    - Proper input validation
    - Normalization
    - Stored procedures
    - Code signing
    - Encryption
    - Obfuscation/camouflage
    - Code reuse/dead code
    - Server-side vs. client-side execution and validation
    - Memory management
    - Use of third-party libraries and SDKs
    - Data exposure
- Code quality and testing
    - Static code analyzers
    - Dynamic analysis (e.g., fuzzing)
    - Stress testing
    - Sandboxing
    - Model verification
- Compiled vs. runtime code

3.7 Summarize cloud and virtualization concepts.

- Hypervisor
    - Type I
    - Type II
    - Application cells/containers

- VM sprawl avoidance
- VM escape protection
- Cloud storage
- Cloud deployment models
    - SaaS
    - PaaS
    - IaaS
    - Private
    - Public
    - Hybrid
    - Community
- On-premise vs. hosted vs. cloud
- VDI/VDE
- Cloud access security broker
- Security as a Service

3.8 Explain how resiliency and automation strategies reduce risk.

- Automation/scripting
    - Automated courses of action
    - Continuous monitoring
    - Configuration validation
- Templates
- Master image
- Non-persistence
    - Snapshots
    - Revert to known state
    - Rollback to known configuration
    - Live boot media
- Elasticity
- Scalability
- Distributive allocation
- Redundancy
- Fault tolerance
- High availability
- RAID

3.9 Explain the importance of physical security controls.

- Lighting
- Signs
- Fencing/gate/cage
- Security guards
- Alarms
- Safe
- Secure cabinets/enclosures
- Protected distribution/Protected cabling
- Airgap
- Mantrap
- Faraday cage
- Lock types
- Biometrics
- Barricades/bollards
- Tokens/cards
- Environmental controls
    - HVAC
    - Hot and cold aisles
    - Fire suppression
- Cable locks
- Screen filters
- Cameras
- Motion detection
- Logs
- Infrared detection
- Key management

Domain 4.0 Identity and Access Management

4.1 Compare and contrast identity and access management concepts.

- Identification, authentication, authorization and accounting (AAA)
- Multifactor authentication
    - Something you are
    - Something you have

    - Something you know
    - Somewhere you are
    - Something you do
- Federation
- Single sign-on
- Transitive trust

4.2 Given a scenario, install and configure identity and access services.

- LDAP
- Kerberos
- TACACS+
- CHAP
- PAP
- MSCHAP
- RADIUS
- SAML
- OpenID Connect
- OAUTH
- Shibboleth
- Secure token
- NTLM

4.3 Given a scenario, implement identity and access management controls.

- Access control models
    - MAC
    - DAC
    - ABAC
    - Role-based access control
    - Rule-based access control
- Physical access control
    - Proximity cards
    - Smart cards
- Biometric factors
    - Fingerprint scanner
    - Retinal scanner

  - Iris scanner
  - Voice recognition
  - Facial recognition
  - False acceptance rate
  - False rejection rate
  - Crossover error rate
- Tokens
  - Hardware
  - Software
  - HOTP/TOTP
- Certificate-based authentication
  - PIV/CAC/smart card
  - IEEE 802.1x
- File system security
- Database security

4.4 Given a scenario, differentiate common account management practices.

- Account types
  - User account
  - Shared and generic accounts/credentials
  - Guest accounts
  - Service accounts
  - Privileged accounts
- General Concepts
  - Least privilege
  - Onboarding/offboarding
  - Permission auditing and review
  - Usage auditing and review
  - Time-of-day restrictions
  - Recertification
  - Standard naming convention
  - Account maintenance
  - Group-based access control
  - Location-based policies

- Account policy enforcement
  - Credential management
  - Group policy
  - Password complexity
  - Expiration
  - Recovery
  - Disablement
  - Lockout
  - Password history
  - Password reuse
  - Password length

Domain 5.0 Risk Management

5.1 Explain the importance of policies, plans and procedures related to organizational security.

- Standard operating procedure
- Agreement types
  - BPA
  - SLA
  - ISA
  - MOU/MOA
- Personnel management
  - Mandatory vacations
  - Job rotation
  - Separation of duties
  - Clean desk
  - Background checks
  - Exit interviews
  - Role-based awareness training
    - Data owner
    - System administrator
    - System owner
    - User

    - Privileged user
    - Executive user
  - NDA
  - Onboarding
  - Continuing education
  - Acceptable use policy/rules of behavior
  - Adverse actions
- General security policies
  - Social media networks/applications
  - Personal email

5.2 Summarize business impact analysis concepts.

- RTO/RPO
- MTBF
- MTTR
- Mission-essential functions
- Identification of critical systems
- Single point of failure
- Impact
  - Life
  - Property
  - Safety
  - Finance
  - Reputation
- Privacy impact assessment
- Privacy threshold assessment

5.3 Explain risk management processes and concepts.

- Threat assessment
  - Environmental
  - Manmade
  - Internal vs. external
- Risk assessment
  - SLE
  - ALE
  - ARO

    - Asset value
    - Risk register
    - Likelihood of occurrence
    - Supply chain assessment
    - Impact
    - Quantitative
    - Qualitative
    - Testing
        - Penetration testing authorization
        - Vulnerability testing authorization
    - Risk response techniques
        - Accept
        - Transfer
        - Avoid
        - Mitigate
- Change management

5.4 Given a scenario, follow incident response procedures.

- Incident response plan
    - Documented incident types/category definitions
    - Roles and responsibilities
    - Reporting requirements/escalation
    - Cyber-incident response teams
    - Exercise
- Incident response process
    - Preparation
    - Identification
    - Containment
    - Eradication
    - Recovery
    - Lessons learned

5.5 Summarize basic concepts of forensics.

- Order of volatility
- Chain of custody
- Legal hold

- Data acquisition
  - Capture system image
  - Network traffic and logs
  - Capture video
  - Record time offset
  - Take hashes
  - Screenshots
  - Witness interviews
- Preservation
- Recovery
- Strategic intelligence/counterintelligence gathering
  - Active logging
- Track man-hours

5.6 Explain disaster recovery and continuity of operation concepts.

- Recovery sites
  - Hot site
  - Warm site
  - Cold site
- Order of restoration
- Backup concepts
  - Differential
  - Incremental
  - Snapshots
  - Full
- Geographic considerations
  - Off-site backups
  - Distance
  - Location selection
  - Legal implications
  - Data sovereignty
- Continuity of operation planning
  - Exercises/tabletop
  - After-action reports
  - Failover

    - Alternate processing sites
    - Alternate business practices

5.7 Compare and contrast various types of controls.

- Deterrent
- Preventive
- Detective
- Corrective
- Compensating
- Technical
- Administrative
- Physical

5.8 Given a scenario, carry out data security and privacy practices.

- Data destruction and media sanitization
    - Burning
    - Shredding
    - Pulping
    - Pulverizing
    - Degaussing
    - Purging
    - Wiping
- Data sensitivity labeling and handling
    - Confidential
    - Private
    - Public
    - Proprietary
    - PII
    - PHI
- Data roles
    - Owner
    - Steward/custodian
    - Privacy officer
- Data retention
- Legal and compliance

Domain 6.0 Cryptography and PKI

6.1 Compare and contrast basic concepts of cryptography.

- Symmetric algorithms
- Modes of operation
- Asymmetric algorithms
- Hashing
- Salt, IV, nonce
- Elliptic curve
- Weak/deprecated algorithms
- Key exchange
- Digital signatures
- Diffusion
- Confusion
- Collision
- Steganography
- Obfuscation
- Stream vs. block
- Key strength
- Session keys
- Ephemeral key
- Secret algorithm
- Data-in-transit
- Data-at-rest
- Data-in-use
- Random/pseudo-random number generation
- Key stretching
- Implementation vs. algorithm selection
  - Crypto service provider
  - Crypto modules
- Perfect forward secrecy
- Security through obscurity

- Common use cases
  - Low power devices
  - Low latency
  - High resiliency
  - Supporting confidentiality
  - Supporting integrity
  - Supporting obfuscation
  - Supporting authentication
  - Supporting non-repudiation
  - Resource vs. security constraints

6.2 Explain cryptography algorithms and their basic characteristics.

- Symmetric algorithms
  - AES
  - DES
  - 3DES
  - RC4
  - Blowfish/Twofish
- Cipher modes
  - CBC
  - GCM
  - ECB
  - CTM
  - Stream vs. block
- Asymmetric algorithms
  - RSA
  - DSA
  - Diffie-Hellman
    - Groups
    - DHE
    - ECDHE
  - Elliptic curve
  - PGP/GPG

- Hashing algorithms
  - MD5
  - SHA
  - HMAC
  - RIPEMD
- Key stretching algorithms
  - BCRYPT
  - PBKDF2
- Obfuscation
  - XOR
  - ROT13
  - Substitution ciphers

6.3 Given a scenario, install and configure wireless security settings.

- Cryptographic protocols
  - WPA
  - WPA2
  - CCMP
  - TKIP
- Authentication protocols
  - EAP
  - PEAP
  - EAP-FAST
  - EAP-TLS
  - EAP-TTLS
  - IEEE 802.1x
  - RADIUS Federation
- Methods
  - PSK vs. Enterprise vs. Open
  - WPS
  - Captive portals

6.4 Given a scenario, implement public key infrastructure.

- Components
  - CA
  - Intermediate CA

  - CRL
  - OCSP
  - CSR
  - Certificate
  - Public key
  - Private key
  - Object identifiers (OID)
- Concepts
  - Online vs. offline CA
  - Stapling
  - Pinning
  - Trust model
  - Key escrow
  - Certificate chaining
- Types of certificates
  - Wildcard
  - SAN
  - Code signing
  - Self-signed
  - Machine/computer
  - Email
  - User
  - Root
  - Domain validation
  - Extended validation
- Certificate formats
  - DER
  - PEM
  - PFX
  - CER
  - P12
  - P7B

Security+ Acronyms

Here are the acronyms of security terms that CompTIA deems important enough that they're included in the objectives list for the exam. We've repeated them here exactly as listed by CompTIA.

3DES	Triple Digital Encryption Standard
AAA	Authentication, Authorization, and Accounting
ABAC	Attribute-based Access Control
ACL	Access Control List
AES	Advanced Encryption Standard
AES256	Advanced Encryption Standards 256bit
AH	Authentication Header
ALE	Annualized Loss Expectancy
AP	Access Point
API	Application Programming Interface
APT	Advanced Persistent Threat
ARO	Annualized Rate of Occurrence
ARP	Address Resolution Protocol
ASLR	Address Space Layout Randomization
ASP	Application Service Provider
AUP	Acceptable Use Policy
AV	Antivirus
BAC	Business Availability Center
BCP	Business Continuity Planning
BIA	Business Impact Analysis
BIOS	Basic Input/Output System
BPA	Business Partners Agreement
BPDU	Bridge Protocol Data Unit
BYOD	Bring Your Own Device

CA	Certificate Authority
CAC	Common Access Card
CAN	Controller Area Network
CAPTCHA	Completely Automated Public Turing test to tell Computers and Humans Apart
CAR	Corrective Action Report
CBC	Cipher Block Chaining
CCMP	Counter-Mode/CBC-Mac Protocol
CCTV	Closed-circuit Television
CER	Certificate
CERT	Computer Emergency Response Team
CFB	Cipher Feedback
CHAP	Challenge Handshake Authentication Protocol
CIO	Chief Information Officer
CIRT	Computer Incident Response Team
CMS	Content Management System
COOP	Continuity of Operations Plan
COPE	Corporate Owned, Personally Enabled
CP	Contingency Planning
CRC	Cyclical Redundancy Check
CRL	Certificate Revocation List
CSO	Chief Security Officer
CSP	Cloud Service Provider
CSR	Certificate Signing Request
CSRF	Cross-site Request Forgery
CSU	Channel Service Unit
CTM	Counter-Mode
CTO	Chief Technology Officer

CYOD	Choose Your Own Device
DAC	Discretionary Access Control
DBA	Database Administrator
DDoS	Distributed Denial of Service
DEP	Data Execution Prevention
DER	Distinguished Encoding Rules
DES	Digital Encryption Standard
DHCP	Dynamic Host Configuration Protocol
DHE	Data-Handling Electronics
DHE	Diffie-Hellman Ephemeral
DLL	Dynamic Link Library
DLP	Data Loss Prevention
DMZ	Demilitarized Zone
DNAT	Destination Network Address Transaction
DNS	Domain Name Service (Server)
DoS	Denial of Service
DRP	Disaster Recovery Plan
DSA	Digital Signature Algorithm
DSL	Digital Subscriber Line
DSU	Data Service Unit
EAP	Extensible Authentication Protocol
ECB	Electronic Code Book
ECC	Elliptic Curve Cryptography
ECDHE	Elliptic Curve Diffie-Hellman Ephemeral
ECDSA	Elliptic Curve Digital Signature Algorithm
EFS	Encrypted File System
EMI	Electromagnetic Interference
EMP	Electro Magnetic Pulse

ERP	Enterprise Resource Planning
ESN	Electronic Serial Number
ESP	Encapsulated Security Payload
FACL	File System Access Control List
FDE	Full Disk Encryption
FRR	False Rejection Rate
FTP	File Transfer Protocol
FTPS	Secured File Transfer Protocol
GCM	Galois Counter Mode
GPG	Gnu Privacy Guard
GPO	Group Policy Object
GPS	Global Positioning System
GPU	Graphic Processing Unit
GRE	Generic Routing Encapsulation
HA	High Availability
HDD	Hard Disk Drive
HIDS	Host-based Intrusion Detection System
HIPS	Host-based Intrusion Prevention System
HMAC	Hashed Message Authentication Code
HOTP	HMAC-based One-Time Password
HSM	Hardware Security Module
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol over SSL/TLS
HVAC	Heating, Ventilation and Air Conditioning
IaaS	Infrastructure as a Service
ICMP	Internet Control Message Protocol
ICS	Industrial Control Systems

ID	Identification
IDEA	International Data Encryption Algorithm
IDF	Intermediate Distribution Frame
IdP	Identity Provider
IDS	Intrusion Detection System
IEEE	Institute of Electrical and Electronic Engineers
IKE	Internet Key Exchange
IM	Instant Messaging
IMAP4	Internet Message Access Protocol v4
IoT	Internet of Things
IP	Internet Protocol
IPSec	Internet Protocol Security
IR	Incident Response
IR	Infrared
IRC	Internet Relay Chat
IRP	Incident Response Plan
ISA	Interconnection Security Agreement
ISP	Internet Service Provider
ISSO	Information Systems Security Officer
ITCP	IT Contingency Plan
IV	Initialization Vector
KDC	Key Distribution Center
KEK	Key Encryption Key
L2TP	Layer 2 Tunneling Protocol
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
LEAP	Lightweight Extensible Authentication Protocol
MaaS	Monitoring as a Service

MAC	Mandatory Access Control
MAC	Media Access Control
MAC	Message Authentication Code
MAN	Metropolitan Area Network
MBR	Master Boot Record
MD5	Message Digest 5
MDF	Main Distribution Frame
MFD	Multi-function Device
MITM	Man-in-the-Middle
MMS	Multimedia Message Service
MOA	Memorandum of Agreement
MOU	Memorandum of Understanding
MPLS	Multi-protocol Label Switching
MSCHAP	Microsoft Challenge Handshake Authentication Protocol
MSP	Managed Service Provider
MTBF	Mean Time Between Failures
MTTF	Mean Time to Failure
MTTR	Mean Time to Recover or Mean Time to Repair
MTU	Maximum Transmission Unit
NAC	Network Access Control
NAT	Network Address Translation
NDA	Non-disclosure Agreement
NFC	Near Field Communication
NIDS	Network-based Intrusion Detection System
NIPS	Network-based Intrusion Prevention System
NIST	National Institute of Standards & Technology
NTFS	New Technology File System
NTLM	New Technology LAN Manager

NTP	Network Time Protocol
OAUTH	Open Authorization
OCSP	Online Certificate Status Protocol
OID	Object Identifier
OS	Operating System
OTA	Over The Air
OVAL	Open Vulnerability Assessment Language
P12	PKCS #12
P2P	Peer to Peer
PaaS	Platform as a Service
PAC	Proxy Auto Configuration
PAM	Pluggable Authentication Modules
PAP	Password Authentication Protocol
PAT	Port Address Translation
PBKDF2	Password-based Key Derivation Function 2
PBX	Private Branch Exchange
PCAP	Packet Capture
PEAP	Protected Extensible Authentication Protocol
PED	Personal Electronic Device
PEM	Privacy-enhanced Electronic Mail
PFS	Perfect Forward Secrecy
PFX	Personal Exchange Format
PGP	Pretty Good Privacy
PHI	Personal Health Information
PII	Personally Identifiable Information
PIV	Personal Identity Verification
PKI	Public Key Infrastructure
POP	Post Office Protocol

POTS	Plain Old Telephone Service
PPP	Point-to-Point Protocol
PPTP	Point-to-Point Tunneling Protocol
PSK	Pre-shared Key
PTZ	Pan-Tilt-Zoom
RA	Recovery Agent
RA	Registration Authority
RAD	Rapid Application Development
RADIUS	Remote Authentication Dial-in User Server
RAID	Redundant Array of Inexpensive Disks
RAS	Remote Access Server
RAT	Remote Access Trojan
RBAC	Role-based Access Control
RBAC	Rule-based Access Control
RC4	Rivest Cipher version 4
RFID	Radio Frequency Identifier
RIPEMD	RACE Integrity Primitives Evaluation Message Digest
ROI	Return on Investment
RPO	Recovery Point Objective
RSA	Rivest, Shamir, & Adleman
RTBH	Remotely Triggered Black Hole
RTO	Recovery Time Objective
RTOS	Real-time Operating System
RTP	Real-time Transport Protocol
S/MIME	Secure/Multipurpose Internet Mail Extensions
SaaS	Software as a Service
SAML	Security Assertions Markup Language
SAN	Storage Area Network

SAN	Subject Alternative Name
SCADA	System Control and Data Acquisition
SCAP	Security Content Automation Protocol
SCEP	Simple Certificate Enrollment Protocol
SCSI	Small Computer System Interface
SDK	Software Development Kit
SDLC	Software Development Life Cycle
SDLM	Software Development Life Cycle Methodology
SDN	Software Defined Network
SED	Self-encrypting Drive
SEH	Structured Exception Handler
SFTP	Secured File Transfer Protocol
SHA	Secure Hashing Algorithm
SHTTP	Secure Hypertext Transfer Protocol
SIEM	Security Information and Event Management
SIM	Subscriber Identity Module
SLA	Service Level Agreement
SLE	Single Loss Expectancy
SMS	Short Message Service
SMTP	Simple Mail Transfer Protocol
SMTPS	Simple Mail Transfer Protocol Secure
SNMP	Simple Network Management Protocol
SOAP	Simple Object Access Protocol
SoC	System on Chip
SPIM	Spam over Internet Messaging
SQL	Structured Query Language
SRTP	Secure Real-Time Protocol
SSD	Solid State Drive

SSH	Secure Shell
SSL	Secure Sockets Layer
SSO	Single Sign-on
STP	Shielded Twisted Pair
TACACS+	Terminal Access Controller Access Control System Plus
TCP/IP	Transmission Control Protocol/Internet Protocol
TGT	Ticket Granting Ticket
TKIP	Temporal Key Integrity Protocol
TLS	Transport Layer Security
TOTP	Time-based One-time Password
TPM	Trusted Platform Module
TSIG	Transaction Signature
UAT	User Acceptance Testing
UAV	Unmanned Aerial Vehicle
UDP	User Datagram Protocol
UEFI	Unified Extensible Firmware Interface
UPS	Uninterruptable Power Supply
URI	Uniform Resource Identifier
URL	Universal Resource Locator
USB	Universal Serial Bus
USB OTG	USB On The Go
UTM	Unified Threat Management
UTP	Unshielded Twisted Pair
VDE	Virtual Desktop Environment
VDI	Virtual Desktop Infrastructure
VLAN	Virtual Local Area Network
VLSM	Variable Length Subnet Masking
VM	Virtual Machine

VoIP	Voice over IP
VPN	Virtual Private Network
VTC	Video Teleconferencing
WAF	Web Application Firewall
WAP	Wireless Access Point
WEP	Wired Equivalent Privacy
WIDS	Wireless Intrusion Detection System
WIPS	Wireless Intrusion Prevention System
WORM	Write Once Read Many
WPA	WiFi Protected Access
WPA2	WiFi Protected Access 2
WPS	WiFi Protected Setup
WTLS	Wireless TLS
XML	Extensible Markup Language
XOR	Exclusive Or
XSRF	Cross-site Request Forgery
XSS	Cross-site Scripting

Chapter

1

Threats, Attacks, and Vulnerabilities

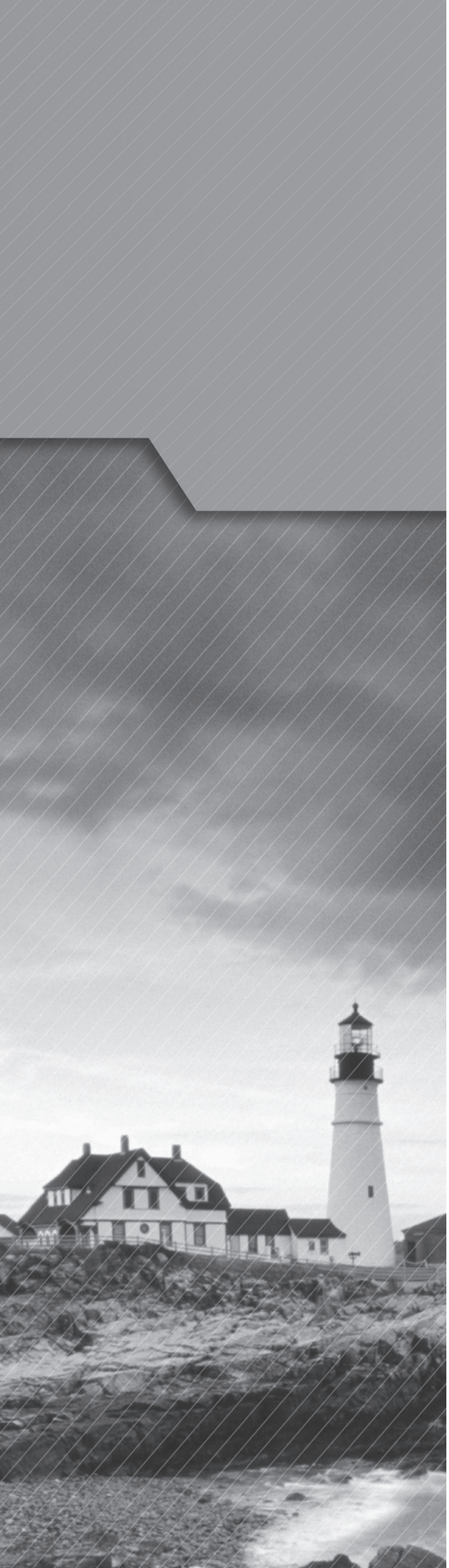
COMPTIA SECURITY+ EXAM OBJECTIVES COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:

✓ **1.1 Given a scenario, analyze indicators of compromise and determine the type of malware.**

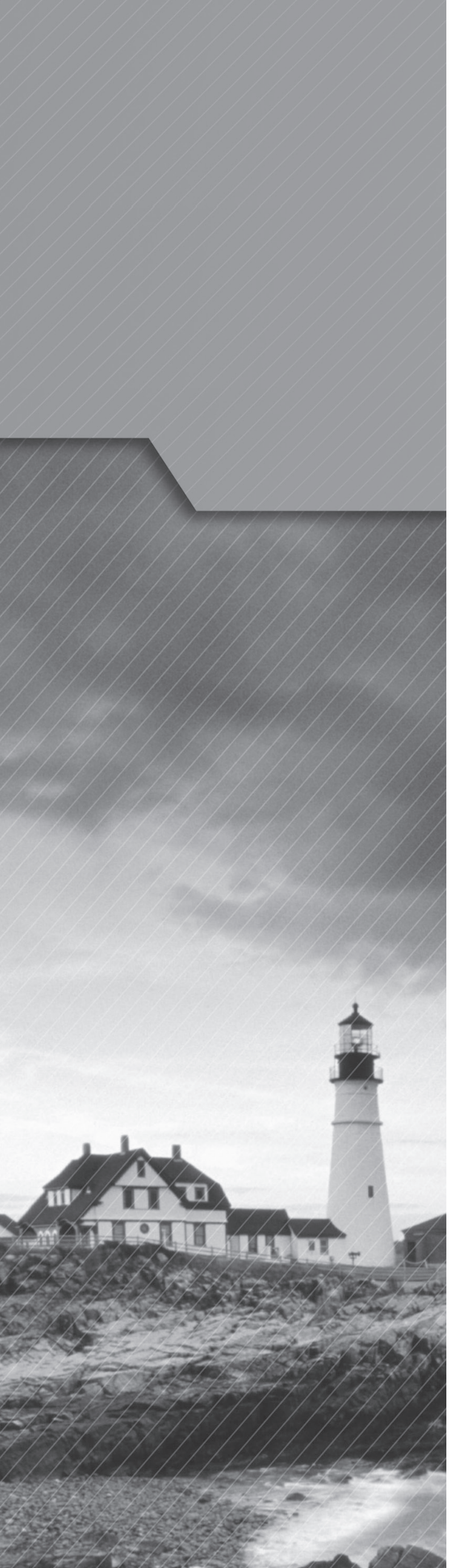
- Viruses
- Crypto-malware
- Ransomware
- Worm
- Trojan
- Rootkit
- Keylogger
- Adware
- Spyware
- Bots
- RAT
- Logic bomb
- Backdoor

✓ **1.2 Compare and contrast types of attacks.**

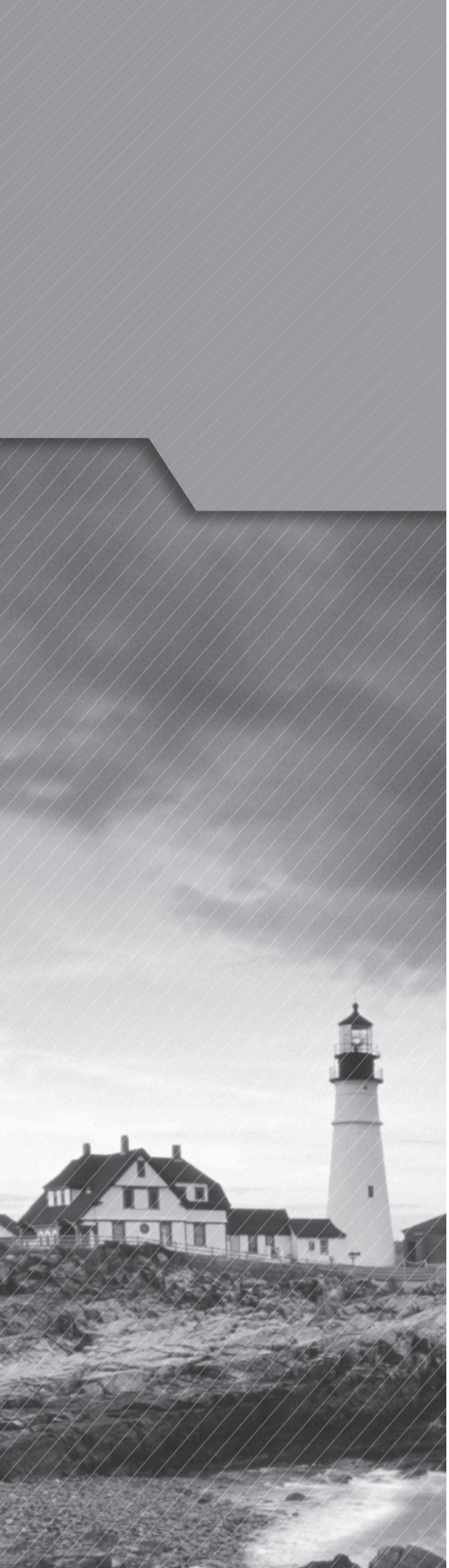
- Social engineering
  - Phishing
  - Spear phishing
  - Whaling
  - Vishing
  - Tailgating



    - Impersonation
    - Dumpster diving
    - Shoulder surfing
    - Hoax
    - Watering hole attack
    - Principles (reasons for effectiveness)
        - Authority
        - Intimidation
        - Consensus
        - Scarcity
        - Familiarity
        - Trust
        - Urgency
- Application/service attacks
    - DoS
    - DDoS
    - Man-in-the-middle
    - Buffer overflow
    - Injection
    - Cross-site scripting
    - Cross-site request forgery
    - Privilege escalation
    - ARP poisoning
    - Amplification
    - DNS poisoning
    - Domain hijacking
    - Man-in-the-browser
    - Zero day
    - Replay
    - Pass the hash



  - Hijacking and related attacks
    - Clickjacking
    - Session hijacking
    - URL hijacking
    - Typo squatting
  - Driver manipulation
    - Shimming
    - Refactoring
  - MAC spoofing
  - IP spoofing
- Wireless attacks
  - Replay
  - IV
  - Evil twin
  - Rogue AP
  - Jamming
  - WPS
  - Bluejacking
  - Bluesnarfing
  - RFID
  - NFC
  - Disassociation
- Cryptographic attacks
  - Birthday
  - Known plain text/cipher text
  - Rainbow tables
  - Dictionary
  - Brute force
    - Online vs. offline
  - Collision
  - Downgrade



    - Replay
    - Weak implementations

✓ 1.3 Explain threat actor types and attributes.

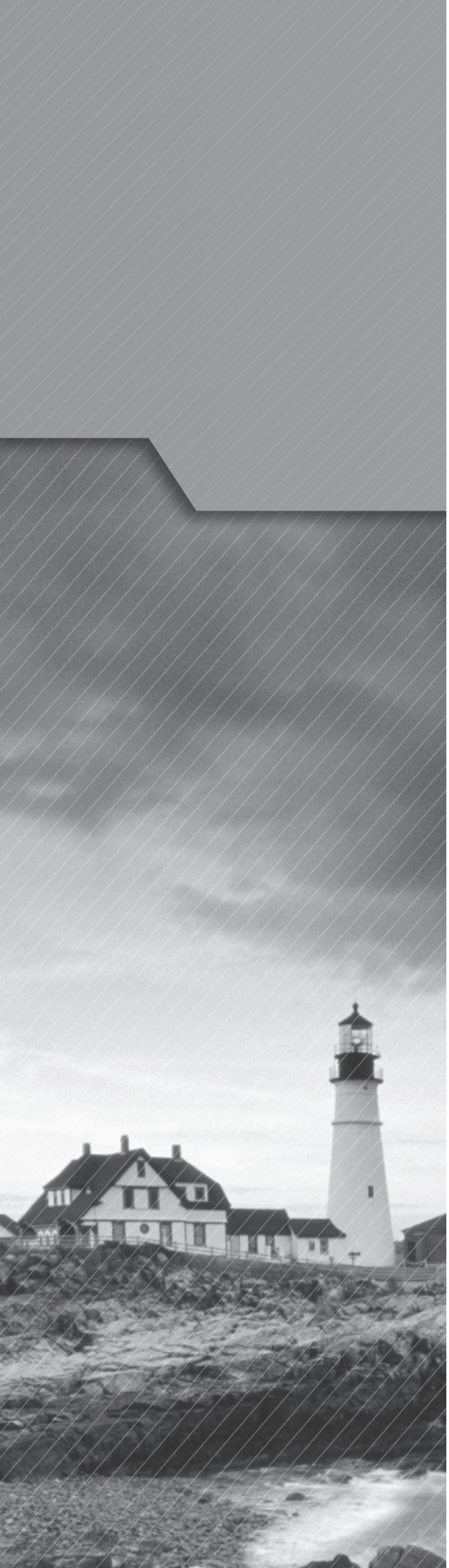
- Types of actors
    - Script kiddies
    - Hacktivist
    - Organized crime
    - Nation states/APT
    - Insiders
    - Competitors
- Attributes of actors
    - Internal/external
    - Level of sophistication
    - Resources/funding
    - Intent/motivation
- Use of open-source intelligence

✓ 1.4 Explain penetration testing concepts.

- Active reconnaissance
- Passive reconnaissance
- Pivot
- Initial exploitation
- Persistence
- Escalation of privilege
- Black box
- White box
- Gray box
- Pen testing vs. vulnerability scanning

✓ 1.5 Explain vulnerability scanning concepts.

- Passively test security controls
- Identify vulnerability



- Identify lack of security controls
- Identify common misconfigurations
- Intrusive vs. non-intrusive
- Credentialed vs. non-credentialed
- False positive

✓ 1.6 Explain the impact associated with types of vulnerabilities.

- Race conditions
- Vulnerabilities due to:
  - End-of-life systems
  - Embedded systems
  - Lack of vendor support
- Improper input handling
- Improper error handling
- Misconfiguration/weak configuration
- Default configuration
- Resource exhaustion
- Untrained users
- Improperly configured accounts
- Vulnerable business processes
- Weak cipher suites and implementations
- Memory/buffer vulnerability
  - Memory leak
  - Integer overflow
  - Buffer overflow
  - Pointer dereference
  - DLL injection
- System sprawl/undocumented assets
- Architecture/design weaknesses
- New threats/zero day
- Improper certificate and key management

The Security+ exam will test your knowledge of IT attacks and compromises. There are a wide range of hacks and compromises that both individuals and organizations must understand in order to defend against downtime and intrusion. To pass the test and be effective in reducing loss and harm, you need to understand the threats, attacks, vulnerabilities, concepts, and terminology detailed in this chapter.

1.1 Given a scenario, analyze indicators of compromise and determine the type of malware.

Malware or *malicious code* is any element of software that performs an unwanted function from the perspective of the legitimate user or owner of a computer system. This objective topic focuses on your ability to recognize a specific type of malware from a given scenario, list of symptoms, or general description of an infection or compromise. Malicious code includes a wide range of concepts, including viruses, ransomware, worms, Trojans, rootkits, keyloggers, adware, spyware, bots, RATs (Remote Access Trojan), logic bombs, and backdoors. Following is an overview of each.

Viruses

Viruses are just one example of malicious code, malicious software, or malware. *Viruses* get their name from their biological counterparts. They're programs designed to spread from one system to another through self-replication and to perform any of a wide range of malicious activities. The malicious activities performed by viruses include data deletion, corruption, alteration, and exfiltration. Some viruses replicate and spread so rapidly that they consume most of the available system and network resources, thus performing a type of denial-of-service (DoS) attack (discussed later in this chapter).

Most viruses need a host to latch onto. The host can be a file (as in the case of *common viruses*) or the boot sector of a storage device. Viruses that attach themselves to the boot sector of a storage device (including HDD, SSD, CD/DVD-ROM, Blu-ray, and USB), and thus are loaded in memory when the drive is activated, are known as *boot sector viruses*.

Within these categories, some specific virus types include the following:

Polymorphic viruses *Polymorphic viruses* have the ability to mask their own code using encryption in order to avoid detection by antivirus scanners.

Macro viruses *Macro viruses* live within documents or emails and exploit the scripting capabilities of productivity software.

Stealth viruses *Stealth viruses* attempt to avoid detection by masking or hiding their activities.

Armored viruses *Armored viruses* are any form of malware that has been crafted to avoid detection and make removal difficult. This can involve the use of complex compiling techniques, overly complex coding logic, and abnormal use of memory.

Retroviruses *Retroviruses* are specifically targeted at antivirus systems to render them useless.

Phage viruses *Phage viruses* modify or infect many aspects of a system so they can regenerate themselves from any remaining unremoved parts.

Companion viruses A *companion virus* borrows the root filename of a common executable and then gives itself the `.com` extension in an attempt to get itself launched rather than the intended application.

Multipart or multipartite viruses *Multipart* or *multipartite viruses* perform multiple tasks and may infect a system in numerous ways.

The best technology to serve as a countermeasure against viruses is an antivirus or antimalware scanner that is updated regularly and that monitors all local storage devices, memory, and communication pathways for viral activities. However, it is essential that modifying user behavior to avoid risky activities be a core part of the security strategy. Otherwise, without human risk reduction, no technological protections will be sufficient. Examples of activities to reduce or avoid risk include avoiding downloading software from nonvendor sources, not opening email attachments, and avoiding the use of removable media from other environments.

If a system is infected with a virus, some potential symptoms include corrupted or missing data files, applications that will no longer execute, slow system operation, lag between mouse click and system response, application or system crashes, ongoing hard drive activity, and the system's tendency to be unresponsive to mouse movements or keystrokes. Any of these symptoms could accompany a virus infection; however, they can be symptoms of other malware infections as well.

Crypto-malware

Crypto-malware is any form of malware that uses cryptography as a weapon or a defense. Crypto as a weapon is seen in malware such as ransomware, while crypto as a defense is seen in malware such as polymorphic and armored viruses.

Another potential form of crypto-malware is code that seeks out the encryption keys of encrypted storage devices and then discloses those keys to a remote attacker. The goal or purpose of such malware is to grant the attacker access to otherwise protected content.

Symptoms of crypto-malware infection include the inability to access data, missing data, a system that will not boot, a sluggish system (during the encryption processes), and pop-ups demanding payment to decrypt your data.

Ransomware

Ransomware is a form of malware that takes over a computer system, usually by encrypting user data, in order to hinder its use while demanding payment. Effectively, it's malware that holds a user's data hostage in exchange for a ransom payment. Often, the thieves behind ransomware request payment to be made in untraceable money cards, such as the MoneyPak Green Dot card, or in Bitcoins (a form of digital currency intended to be untraceable).

Countermeasures against ransomware include avoiding risky behaviors, running anti-malware software, and maintaining a reliable backup of your data. Unless absolutely no other option is available to you to regain access to your data, avoid paying the ransom. Paying a ransom to attackers only encourages them to continue their criminal activities.

Symptoms of ransomware infection include the inability to access data, missing data, a system that will not boot, a sluggish system (during the encryption processes), and pop-ups demanding payment to decrypt your data.

Worm

Another form of malware that is closely related to a virus is a worm. *Worms* are self-contained applications that don't require a host file or hard drive to infect. Worms typically are focused on replication and distribution, rather than on direct damage and destruction. Worms are designed to exploit a specific vulnerability in a system (operating system, protocol, service, or application) and then use that flaw to spread themselves to other systems with the same flaw. They may be used to deposit viruses, logic bombs, ransomware, backdoors, or zombies/agents/bots for botnets, or they may perform direct virus-like maelstrom activities on their own.

Countermeasures for worms are the same as for viruses, with the addition of keeping systems patched.

A worm infection may display symptoms that include a slow-to-respond system, applications that no longer will execute, a lack of free space on storage devices, CPU and memory utilization maxed out at 100 percent, system crashes, and abnormal network activity.

Trojan

A *Trojan horse* is a form of malicious software that is disguised as something useful or legitimate. The most common forms of Trojan horses are games and screensavers, but any software can be made into a Trojan. The goal of a Trojan horse is to trick a user into installing it on their computer. This allows the malicious code portion of the Trojan to gain access to the otherwise secured environment. A Trojan is crafted by combining a seemingly benign host file with a malicious payload. It is an integration of technology abuse with social engineering. The victim is tricked into accepting the Trojan on their system because they believe that the only thing they are obtaining is the obvious benign host. However, when the host is used, the malicious payload is released to infect the system. Some of the most common Trojans are tools that install distributed denial-of-service (DDoS), botnet agents, or remote-control backdoors onto systems.

Countermeasures for Trojan horses are the same as for viruses.

Scenarios involving a system becoming infected through Trojan horse delivery of malware can elicit any of the symptoms mentioned for other malware infections (see earlier and later malware concepts), since a Trojan horse can be used to deliver any sort of malicious code. In addition, a Trojan horse may cause system slowdown or unresponsiveness immediately after triggering or launching the Trojan horse while it is delivering the malicious payload.

Rootkit

A *rootkit* is a special type of hacker tool that embeds itself deep within an operating system (OS). The rootkit positions itself at the heart of an OS, where it can manipulate information seen by the OS. Often, a rootkit replaces the OS kernel, shims itself under the kernel, replaces device drivers, or infiltrates application libraries so that whatever information it feeds or hides from the OS, the OS thinks is normal and acceptable. This allows a rootkit to hide itself from detection, prevent its files from being viewed by file management tools, and prevent its active processes from being viewed by task management or process management tools. Thus, a rootkit is a type of invisibility shield. A rootkit can be used to hide other malicious tools and/or perform other functions. A rootkit or other tools hidden by a rootkit can capture keystrokes, steal credentials, watch URLs, take screen captures, record sounds via the microphone, track application use, or grant a remote hacker backdoor access or remote control over the compromised target system.

After a rootkit has infected a system, that system can no longer be trusted or considered secure. There are rootkits that are still undetectable and/or can't be effectively removed. Thus, any rootkit-compromised system can never be fully trusted again. To use a silly analogy: if you're fighting an invisible army, how can you be sure that you've defeated all of the soldiers?

There are several rootkit-detection tools, some of which are able to remove certain rootkits. However, once you suspect a rootkit is on a system, the only truly secure response is to reconstitute or replace the entire computer. *Reconstitution* involves performing a low-level formatting operation on all storage devices on that system, reinstalling the OS and all applications from trusted original sources, and then restoring files from trusted rootkit-free backups. Obviously, the best protection against rootkits is defense rather than response.

There are often no noticeable symptoms or indicators of compromise related to a rootkit infection. Rootkit authors often strive to minimize any noticeable activity that might indicate that a system has been compromised. In the moments after initial rootkit installation there might be some system sluggishness and unresponsiveness as the rootkit installs itself, but otherwise it will actively mask any symptoms.

Keylogger

A *keylogger* is a form of malware that records the keystrokes typed into a system's keyboard. Software keyloggers are often able to record input from both physical keyboards and on-screen keyboards. The captured keystrokes are then uploaded to the attacker for analysis and exploitation.

Many antimalware scanners include signatures for keyloggers; however, a potentially unwanted program (PUP) scanner, such as Malwarebytes, might also be necessary to detect this type of abusive software.

Hardware keyloggers are physical devices attached to the keyboard cable where it connects to the main system. Such devices are not detectable by software and thus require physical inspection to uncover. Some hardware keyloggers can upload captured content via Wi-Fi, Bluetooth, or cellular service, whereas others must be physically retrieved.

A keylogger infection might exhibit sluggish keyboard response, require typing keys twice to get them to be recognized by the system, and cause overall system performance degradation.

Adware

Adware is a variation on the idea of spyware (discussed later in this section). Adware displays pop-up advertisements to users based on their activities, URLs they have visited, applications they have accessed, and so on. Adware is used to customize advertisements to prospective customers. Unfortunately, most adware products arrive on client systems without the knowledge or consent of the user. Thus, legitimate commercial products are often seen as intrusive and abusive adware.

Some forms of adware display offerings for fake or false security products. They often display an animation that seems like the system is being scanned; they may even search for malicious code or intrusion events. The adware then displays a warning that problems were found and the solution is to download a "free" utility to remove or resolve the offense. This type of malware is also known as scareware.

Countermeasures for adware are the same as for spyware and viruses—antimalware software with added specific spyware/adware-scanning tools.

Indicators of adware compromise can include the pop-up display of advertisements even when a web browser is not already running, sluggish system response, and poor mouse responsiveness (especially when clicking on links).

Spyware

Spyware is any form of malicious code or even business or commercial code that collects information about users without their direct knowledge or permission. Spyware can be fully malicious when it seeks to gain information to perform identity theft or credential hijacking. However, many advertising companies use less malicious forms of spyware to gather demographics about potential customers. In either case, the user is often unaware that the spyware tool is present or that it's gathering information that is periodically transmitted to some outside entity. Spyware can collect keystrokes, names of launched applications, local files, sent or received emails and instant messages (IMs), and URLs visited; it can also record audio by turning on the microphone, or even record video by turning on a webcam. Spyware can be deposited by viruses, worms, or Trojan horses, or it can be installed as an extra element from commercial, freeware, or shareware applications.

Countermeasures for spyware are the same as for viruses, with the addition of specific spyware-scanning tools.

Spyware infections may cause noticeable symptoms such as slow system performance, poor keyboard and mouse responsiveness, the appearance of unknown files, and quickly dwindling available storage space.

Bots

The term *botnet* is a shortened form of the phrase *robot network*. It is used to describe a massive deployment of malicious code onto numerous compromised systems that are all remotely controlled by a hacker. A botnet is the culmination of traditional DoS attacks into a concept known as a *distributed denial-of-service (DDoS) attack*. A DDoS attack occurs when a hacker has deposited remote-controlled agents, zombies, or bots onto numerous secondary victims and then uses the deployed bots as a single entity to attack a primary target. (This is covered in more detail later in this chapter, when we review specific attack types.)

Botnets are either directly or indirectly controlled by a hacker. Sometimes the hacker is called a *bot herder*, a *master*, or even a *handler*. Direct control of a botnet occurs when the bot herder sends commands to each bot. Therefore, bots have a listening service on an open port waiting for the communication from the bot herder. Indirect control of a botnet can occur through any intermediary communication system, including Internet Relay Chat (IRC), IM, File Transfer Protocol (FTP), email, the Web, blogging, Facebook, Twitter, and so on. When indirect control is used, the bots access an intermediary communication service for messages from the bot herder. The intermediary communication service is often named a "command and control center," but instead of being a complex controlling interface, it is simply the locus of connection between the attacker and the bots where information is exchanged.

Botnets are possible because most computers around the world are accessible over the Internet, and many of those computers have weak security. A botnet creator writes their botnet code to exploit a common vulnerability in order to spread the botnet agent far and wide—often using the same techniques used by viruses, worms, and Trojan horses. Botnets typically include thousands (if not hundreds of thousands) of compromised secondary victims. The secondary victims are the hosts of the botnet agent itself and aren't affected or damaged beyond the initial intrusion and planting of the botnet agent. The hackers want the secondary victims fully functional so that when they launch their botnet attack against the primary victim, they can use all the resources of the secondary victims against the primary target.

A botnet can be used to perform any type of malicious activity. Although they're most often used to perform DoS flooding attacks, botnets can also be used to transmit spam, perform massively distributed parallel processing to crack passwords or encryption keys, perform phishing attacks, capture network packets, or perform any other conceivable activity.

The best defense against a botnet is to keep your systems patched and hardened and to not become the host of a botnet agent (in other words, don't become a secondary victim). Strict outbound firewall rules, spoofed source address filtering, and web content filtering on a unified threat management (UTM) device are also effective countermeasures. In addition, most antivirus software and antispyware/adware tools include well-known botnet agents in their detection databases.

If you're the primary victim of a botnet flooding attack, there is little you can do to stop the attack. Your responses are often limited to disconnecting from the Internet, contacting your ISP, and reporting the incident to law enforcement. There are several DDoS filtering services, which range from free services to quite expensive enterprise-class services.

The indicators of botnet compromise can include slow system performance, high levels of CPU and memory utilization, high levels of abnormal network traffic, strange files appearing on storage devices, unknown processes running, and odd program windows appearing on the desktop.

RAT

A remote-access Trojan (*RAT*) is a form of malicious code that grants an attacker some level of remote-control access to a compromised system. Often the remote-control backdoor component is hidden inside a host file that is linked to some current popular concept, such as a new movie, music album, or game. Once the victim uses or opens the host, the remote-control malware is installed on their system and a notification is sent to the attacker. Most RATs then initiate an outbound connection to the attacker's waiting system to grant them access to manipulate the victim's data and system operations.

RAT infections may result in noticeable symptoms such as odd network communications and traffic levels; a system that will not auto-engage the screensaver or timed sleep mode; higher levels of drive, CPU, and memory activity; and the appearance of unknown files on storage devices.

Logic bomb

A *logic bomb* is a form of malicious code that remains dormant until a triggering event occurs. The triggering event can be a specific time and date, the launching of a specific program, typing in a certain keystroke combination, or the accessing of a specific URL (such as your online banking logon page). Logic bombs can perform any malicious function the programmer wishes, from causing system crashes, to deleting data, to altering configurations, to stealing authentication credentials.

A logic bomb can also be a fork bomb, which triggers a duplication event where the original code is cloned and launched. Then, each of the new clones forks itself again. This forking/cloning process repeats until the system crashes due to complete resource consumption by the malware. A fork bomb also works by consuming storage space or using up the network bandwidth.

Symptoms of logic bomb compromise could include an abrupt change in system performance, crashing of applications or the system, and a loss of storage device free space.

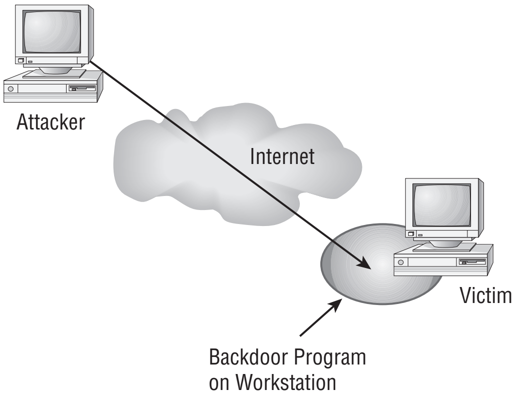
Backdoor

The term *backdoor* can refer to two types of problems or attacks on a system. The first and oldest type of backdoor was a developer-installed access method that bypassed all security restrictions. The backdoor was a special hard-coded user account, password, or command sequence that allowed anyone with knowledge of the access hook (sometimes called a *maintenance hook*) to enter the environment and make changes. This sounds great from a developer's perspective, especially during the coding and debugging process. Unfortunately, such programming shortcuts are often forgotten about when the product nears completion; thus, they end up in the final product. Fortunately, once a backdoor is discovered in a released product, the vendor usually releases a patch to remove the backdoor code from the installed product. The possible presence of backdoors is another good reason to stay current with vendor-released updates and patches.

The second meaning of backdoor is a hacker-installed remote-access client. These small, maliciously purposed tools can easily be deposited on a computer through a Trojan horse, a virus, a worm, a website mobile code download, or even as part of an intrusion activity. Once active on a system, the tool opens access ports and waits for an inbound connection. Thus, a backdoor serves as an access portal for hackers so that they can bypass any security restrictions and gain (or regain) access to a system. Some common backdoor tools include Back Orifice, NetBus, and Sub7 (all of which function on Windows). These and other common backdoor tools are detected and removed by virus scanners and spyware scanning tools.

Figure 1.1 shows a backdoor attack in progress.

FIGURE 1.1 A backdoor attack in progress



Preemptive measures against backdoors include restricting mobile code from being automatically downloaded to your systems, using software policies to prevent unauthorized software from being installed, monitoring inbound and outbound traffic, and requiring software and driver signing.

A backdoor compromise may elicit noticeable symptoms such as an unresponsive system, applications opening or closing seemingly on their own, abnormal network connections and activity, and missing or new files.

Exam Essentials

Understand viruses. Viruses are programs that are designed to spread from one system to another through self-replication and to perform any of a wide range of malicious activities.

Understand crypto-malware. Crypto-malware is any form of malware that uses cryptography as a weapon or a defense.

Understand ransomware. Ransomware is a form of malware that aims to take over a computer system in order to block its use while demanding payment.

Understand worms. Worms are designed to exploit a single flaw in a system (operating system, protocol, service, or application) and then use that flaw to replicate themselves to other systems with the same flaw.

Understand Trojan horses. A Trojan horse is a form of malicious software that is disguised as something useful or legitimate.

Understand rootkits. A rootkit is a type of malicious code that fools the OS into thinking that active processes and files don’t exist. Rootkits render a compromised system completely untrustworthy.

Understand keyloggers. A keylogger is a form of malware that records the keystrokes typed into a system’s keyboard.

Understand spyware and adware. Spyware gathers information about users and may employ that information to customize advertisements or steal identities. Adware gathers information about users and uses it to direct advertisements to the user. Both spyware and adware are usually unwanted software that gathers information without authorization.

Understand botnets. A botnet is a network of robots or malicious software agents controlled by a hacker in order to launch massive attacks against targets.

Understand a RAT. A remote-access Trojan (RAT) is a form of malicious code that grants an attacker some level of remote-control access to a compromised system.

Understand logic bombs. A logic bomb is a form of malicious code that remains dormant until a triggering event occurs. The triggering event can be a specific time and date, the launching of a specific program, or the accessing of a specific URL.

Understand backdoor attacks. There are two types of backdoor attacks: a developer-installed access method that bypasses any and all security restrictions, or a hacker-installed remote-access client.

Understand malicious code countermeasures. The best countermeasure to viruses and other malicious code is an antivirus scanner that is updated regularly and that monitors all local storage devices, memory, and communication pathways for malicious activity. Other countermeasures include avoiding downloading software from the Internet, not opening email attachments, and avoiding the use of removable media from other environments.

1.2 Compare and contrast types of attacks.

Any computer system connected to any type of network is subject to various types of attacks. The rate at which networked systems are attacked is increasing at an alarming rate. Even systems that aren't connected to the Internet, such as those isolated in a private network, may come under attack. There are myriad ways to attack a computer system. Your familiarity with a modest collection of these attacks and how to respond to them is an essential skill for the Security+ exam. The following sections discuss common attack methods.

Social engineering

Social engineering is a form of attack that exploits human nature and human behavior. Social engineering attacks take two primary forms: convincing someone to perform an unauthorized operation or convincing someone to reveal confidential information. For example, the victim may be fooled into believing that a received email is authoritative (such as an email hoax), that a person on the phone is someone to be respected and obeyed (such as someone claiming to be from tech support or a manager offsite), or that a person with them is who they claim to be (such as an air-conditioning [AC] repair technician). In just about every case, in social engineering the attacker tries to convince the victim to perform some activity or reveal a piece of information that they shouldn't. The result of a successful attack is information leakage or the attacker being granted logical or physical access to a secure environment.

Any form of advertisement could be considered a form of social engineering attack—ads appeal to you in an attempt to get you to purchase or use a product or service. Although an advertisement's motivation is profit, the motives for most social engineering attacks are more malevolent. In fact, hackers now have access to sophisticated technology to assist in their social engineering endeavors.

One such tool is the Social Engineering Toolkit (SET). As you can see on the `http://social-engineer.org` website, SET was specifically designed to perform advanced attacks against the human element. It integrates with the Metasploit framework to allow an attacker to take control of a remote computer by enticing the soon-to-be victim to click a pop-up of some sort. For instance, a gamer playing the latest version of the newest hot online video game could receive a pop-up stating that there is temporary Internet congestion. It might then say, "Please select Stay Online if performance is acceptable or select Disconnect to disconnect and reconnect." Either selection results in the attacker's code being run and possibly in the exploitation of the system. The user-interaction portion of the attack is why this is referred to as the Social Engineering Toolkit.

Here are some example scenarios of common social engineering attacks:

- A worker receives an email warning about a dangerous new virus spreading across the Internet. The message directs the worker to look for a specific file on the hard drive and delete it, because it indicates the presence of the virus. Often, however, the identified file is really an essential file needed by the system.
- A website claims to offer free temporary access to its products and services, but it requires web browser and/or firewall alterations in order to download the access software.
- A secretary receives a phone call from a person claiming to be a client who is running late to meet the CEO. The caller asks for the CEO's private cell phone number in order to call them.
- The helpdesk receives a call from an outside line. The caller claims to be a manager of a department who is currently involved in a sales meeting in another city. The caller claims to have forgotten their password and needs it to be reset so that they can log in remotely to download an essential presentation.
- Someone who looks like an AC repair technician enters the office and claims a service call was received for a malfunctioning unit in the building. The "technician" is sure the unit can be accessed from inside your office work area and asks to be given free rein to repair the AC system.
- An unexpected pop-up requires a selection of some sort.

These are just a few examples of possible social engineering attacks. They may also be legitimate and benign occurrences, but you can see how they could mask the motives and purposes of an attacker.

Methods to protect against social engineering include the following:

- Training personnel about social engineering attacks and how to recognize common signs
- Requiring authentication when performing activities for personnel over the phone
- Defining restricted information that is never communicated over the phone
- Always verifying the credentials of a repair person and verifying that a real service call was placed by authorized personnel
- Never following the instructions of an email without verifying the information with at least two independent and trusted sources
- Always erring on the side of caution when dealing with anyone you don't know or recognize, whether in person, over the phone, or over the Internet/network

The only real defense against social engineering attacks is user education and awareness training. A healthy dose of paranoia and suspicion will help users detect or notice social engineering attack attempts. Training should include role playing and numerous examples of the various forms of social engineering attacks.

Phishing

Phishing is a form of social engineering attack focused on stealing credentials or identity information from any potential target. It is based on the concept of fishing for information. Phishing is employed by attackers to obtain sensitive information such as usernames, passwords, credit card details, or other personally identifiable information by masquerading as a trustworthy entity (a bank, a service provider, or a merchant, for example) in electronic communication (usually email). Phishing can be waged in numerous ways using a variety of communication media, including email, the Web, live discussion forums, IM, message boards, and so on.

To defend against phishing attacks, end users should be trained to avoid clicking any link received via email, IM, or social network message. Instead, the user should visit the supposed site by using a preestablished bookmark or by searching for the site by name. If, after accessing their account on the site, a duplicate message does not appear in the online messaging or alert system, the original message is likely an attack or a fake. Any such false communications should be reported to the targeted organization, and then the message should be deleted.

All forms of phishing take advantage of people's willingness to extend trust to apparently legitimate third parties without applying rules of basic, commonsense information security (the most germane of these principles here are "never open unexpected email attachments" and "never share sensitive information via email").

Spear phishing

Spear phishing is a more targeted form of phishing where the message is crafted and directed specifically to a group of individuals, rather than being just a blind broadcast to anyone. Often, attackers will first compromise an online or digital business in order to steal their customer database. Then, false messages are crafted to seem like a communication from the compromised business, but with falsified source addresses and incorrect URLs. The hope of the attack is that someone who already has an online/digital relationship with an organization is more likely to fall for the false communication. If the victim responds, then the followup messages or the website they access is crafted to elicit their personal information in order to perform account takeover or full-fledged identity theft.

Whaling

Whaling is a form of phishing that targets specific high-value individuals (by title, by industry, from media coverage, and so forth), such as C-level executives or high-net-worth clients, and sends messages tailored to the needs and interests of those individuals. Whaling attacks require significantly more research, planning, and development on the part of the attackers in order to fool the victim. But a successful attack can be a significant payoff for the malicious hacker.

Vishing

Vishing is phishing done via *Voice-over-IP* (*VoIP*) services. VoIP is a technology that allows phone call–like conversations to take place over TCP/IP networks. Many companies and individuals use VoIP phones instead of traditional landline phones. The victims of vishing do not have to be using VoIP. Instead, the attack originates from a VoIP service. This allows the attacker to be located anywhere in the world and make a free phone call to the victim.

Vishing is simply another form of phishing attack. The main problem with vishing is that tracing the source or origin of the attacks is much more complicated, if not impossible. Thus, it's more important than ever to be suspicious of phone calls, even those with correct caller ID. Everyone should take the extra effort to verify the caller, or hang up on them and then call the claimed entity back using a known trusted phone number, such as the one on the back of your credit card or from the entity's official website. Users should be trained to be careful about volunteering information when prompted by a caller, such as being asked to provide account numbers, account passwords, secret PINs, billing address, and so on. These are fine to disclose to the valid entity when a user originates the call, but when someone else calls, there is no way to fully verify that they are the claimed entity.

Tailgating

Tailgating occurs when an unauthorized entity gains access to a facility under the authorization of a valid worker but without their knowledge. This attack can occur when a worker uses their valid credentials to unlock and open a door, then walks on into the building as the door closes, granting the attacker the opportunity to stop the door from closing and sneak in without the victim realizing. Tailgating is an attack that does not depend on the consent of the victim, just their obliviousness to what occurs behind them as they walk into a building.

Tailgate prevention by users is very simple. Each and every time a user unlocks or opens a door, they should ensure that it is closed and locked before walking away. This action alone eliminates tailgating. There is social pressure to hold open a door for someone who is walking up behind you, but this courtesy should not be extended to include secure entry points.

A problem similar to tailgating is *piggybacking*. Piggybacking occurs when an unauthorized entity gains access to a facility under the authorization of a valid worker but with their knowledge and consent. This could happen when the intruder feigns the need for assistance by holding a large box or lots of paperwork and asks someone to "hold the door." The goal is to distract the victim while the attacker gains access in order to prevent the victim from realizing that the attacker did not provide their own credentials.

Users should be trained to watch out for this type of attack. When someone asks for assistance in holding open a secured door, users should ask for proof of authorization or offer to swipe the person's access card on their behalf. This reduces the chance of an outsider bluffing their way into your secured areas.

In addition to user behavior changes, mantraps, turnstiles, and security guards all reduce tailgating and piggybacking significantly.

Impersonation

Impersonation is the act of taking on the identity of someone else. This can take place in person, over the phone, or through any other means of communication. The purpose of impersonation is to fool someone into believing you have the claimed identity so you can use the power or authority of that identity. Impersonation is a common element of social engineering. Impersonation can also be known as *masquerading*.

A form of impersonation known as *pretexting* can occur when an individual describes a false situation as a pretext for the social engineering attack.

Dumpster diving

Dumpster diving is the act of digging through trash, discarded equipment, or abandoned locations in order to obtain information about a target organization or individual. Although discovering confidential documentation or secret information would be a welcomed bonus to attackers, they are looking for more mundane documentation. Typical collected items include old calendars, calling lists, meeting notes, discarded forms, product boxes, user manuals, sticky notes, printed reports, or the test sheet from a printer. Dumpster diving can provide an attacker with information that could make social engineering attacks easier or more effective.

To prevent dumpster diving, or at least reduce its value, all documents should be shredded and/or incinerated before being discarded. Additionally, no storage media should ever be discarded in the trash; use a secure disposal technique or service.

Shoulder surfing

Shoulder surfing occurs when someone is able to watch a user's keyboard or view their display. This could allow them to learn a password or see information that is confidential, private, or simply not for their eyes. Often, shoulder surfing is stopped by dividing worker groups by sensitivity levels using locked doors. Additionally, users should not orient their displays to be visible through windows (from outside) or walkways/doorways (for internal issues). And they should not work on sensitive data while in a public space, such as a coffee shop or on a plane.

Hoax

A *hoax* is a form of social engineering designed to convince targets to perform an action that will cause problems or reduce their IT security. A hoax is often an email that proclaims some imminent threat is spreading across the Internet and that you must perform certain tasks in order to protect yourself. Victims may be instructed to delete files or change configuration settings, which results in a compromised OS, a nonbooting OS, or a reduction in their security defenses. Additionally, hoax emails often encourage the victim to forward the message to all their contacts in order to "spread the word."

Watering hole attack

A *watering hole attack* is a form of targeted attack against a region, a group, or an organization. The attack is performed in three main phases. The first phase is to observe the target's habits. The goal is to discover a common resource, site, or location that one or more members of the target frequent. This location is considered the watering hole. The second phase is to plant malware on watering hole systems. The third phase is to wait for members of the target to revisit the poisoned watering hole and then bring the infection back into the group. The name is derived from the concept of wiping out an animal population by poisoning its primary water source. This technique is fairly effective at infiltrating groups that are well secured, are difficult to breach, or operate anonymously. For an example of a watering hole attack performed by the FBI, see `www.wired.com/threatlevel/2013/09/freedom-hosting-fbi/`.

Principles (reasons for effectiveness)

Social engineering works so well because we're human. The principles of social engineering attacks are designed to focus on various aspects of human nature and take advantage of them. Although not every target succumbs to every attack, most of us are vulnerable to one or more of the following common social engineering principles.

Authority

Authority is an effective technique because most people are likely to respond to authority with obedience. The trick is to convince the target that the attacker is someone with valid authority. That authority can be from within an organization's internal hierarchy or from an external recognized authority, such as law enforcement, technical support, pest extermination, utility inspection, debt collection, and so on. Some attackers claim their authority verbally, and others assume authority by wearing a costume or uniform.

Intimidation

Intimidation can sometimes be seen as a derivative of the authority principle. Intimidation uses authority, confidence, or even the threat of harm to motivate someone to follow orders or instructions. Often, intimidation is focused on exploiting uncertainty in a situation where a clear directive of operation or response isn't defined. The attacker attempts to use perceived or real force to bend the will of the victim before the victim has time to consider and respond with a denial.

Consensus

Consensus or *social proof* is the act of taking advantage of a person's natural tendency to mimic what others are doing or are perceived as having done in the past. For example, bartenders often seed their tip jar with money to make it seem as if previous patrons were appreciative of the service. People visiting a tourist spot might carve their name in a railing because many previous visitors' names are present. People will stop walking down the street and join a crowd, just to see what is going on. As a social engineering principle, the attacker attempts to convince the victim that a particular action or response is preferred in order to be consistent with social norms or previous occurrences. For example, an attacker may claim that a worker who is currently out of the office promised a large discount on a purchase and that the transaction must occur now with you as the salesperson.

Scarcity

Scarcity is a technique used to convince someone that an object has a higher value based on the object's scarcity. For example, shoppers often feel motivated to make a purchase because of a limited-time offer, due to a dwindling stock level, or because an item is no longer manufactured.

Familiarity/liking

Familiarity or *liking* as a social-engineering principle attempts to exploit a person's native trust in that which is familiar. The attacker often tries to appear to have a common contact or relationship with the target, such as mutual friends or experiences, or uses a facade to take on the identity of another company or person. If the target believes a message is from a known entity, such as a friend or their bank, they're much more likely to trust in the content and even act or respond.

Trust

Trust as a social engineering principle involves an attacker working to develop a relationship with a victim. This may take seconds or months, but eventually the attacker attempts to use the value of the relationship (the victim's trust in the attacker) to convince the victim to reveal information or perform an action that violates company security.

Urgency

Urgency often dovetails with scarcity, because the need to act quickly increases as scarcity indicates a greater risk of missing out. Urgency is often used as a method to get a quick response from a target before they have time to carefully consider or refuse compliance.

Application/service attacks

Social engineering is not the only form of attack faced by modern environments. A wide variety of attacks and exploitations are used by attackers to exfiltrate data or gain logical or physical access to our organizations. In this section, I discuss several examples of attacks that take advantage of information technology. Keep in mind a phrase attributed to the NSA: "Attacks always get better; they never get worse," meaning that while you may not be vulnerable to a particular attack today, you might be tomorrow. It is also the case that new attacks are being developed by attackers, so you may not even be aware of a new method of exploitation until after your systems have fallen victim to it.

Arbitrary Code Execution/Remote Code Execution

Arbitrary code execution is the ability to run any software—particularly malicious shell code—on a target system. This ability is usually the focus of most hacker exploits and attacks, such as those mentioned in this section. When combined with privilege escalation, a hacker's capacity to arbitrarily run any software of their choosing at an administrator, root, or system level means they have the open-ended ability to perform any task on the system. Often, this capability is established using a remote attack as opposed to a local attack (an attack run on an authorized system from within an authorized user account). A remote exploitation of arbitrary code execution is also called *remote code execution*.

DoS

Denial of service (*DoS*) is a form of attack that has the primary goal of preventing the victimized system from performing legitimate activity or responding to legitimate traffic. There are two basic types of DoS attack. The first form exploits a weakness, an error, or a standard feature of software to cause a system to hang, freeze, consume all system resources, and so on. The end result is that the victimized computer is unable to process any legitimate tasks. The second form floods the victim's communication pipeline with garbage network traffic. Such garbage traffic can be false responses to nonexistent requests, partial establishment of a TCP session, or repeated requests for data from a service or application. The end result is that the victimized computer is unable to send or receive legitimate network communications. In any case, the victim is denied the ability to perform normal operations (services).

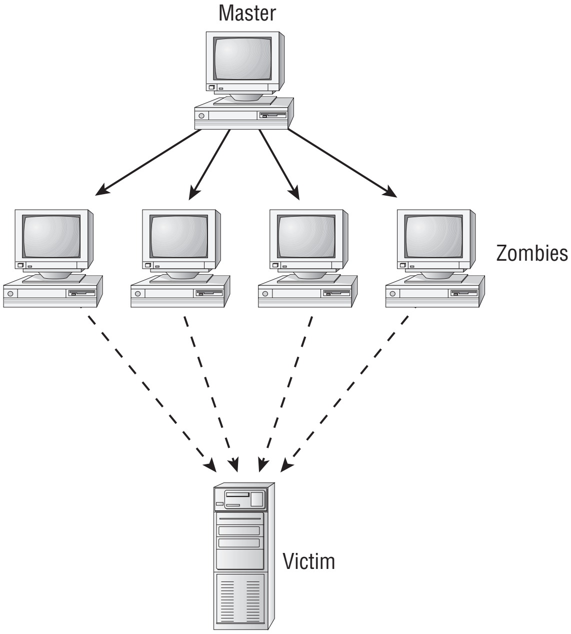
DoS isn't a single attack but rather an entire class of attacks. Some attacks exploit flaws in OS software, whereas others focus on installed applications, services, or protocols. Some attacks exploit specific protocols, including Internet Protocol (IP), Transmission Control Protocol (TCP), Internet Control Message Protocol (ICMP), and User Datagram Protocol (UDP).

DoS attacks typically occur between one attacker and one victim. However, they don't have to be waged in that simple a manner. Most DoS attacks employ some form of intermediary system (usually an unwilling and unknowing participant) in order to hide the attacker from the victim. For example, if an attacker sends attack packets directly to a victim, it's possible for the victim to discover who the attacker is. This is made more difficult, although not impossible, through the use of *spoofing* (discussed later in this chapter).

The next generation of DoS attacks is known as *distributed denial-of-service (DDoS)* attacks. These types of DoS attacks are waged by first compromising or infiltrating one or more intermediary systems that serve as launch points or attack platforms. These intermediary systems are commonly referred to as *secondary victims*. The attacker installs remote-control tools, often called *bots*, *zombies*, or *agents*, onto these systems. Then, at an appointed time or in response to a launch command from the attacker, the DoS attack is conducted against the victim, as shown in Figure 1.2. In this manner, the victim may be able to discover the zombied system(s) that are causing the DoS attack but probably won't be able to track down the actual attacker. Recently, such deployments of many bots or zombies across numerous unsuspecting secondary victims have become known as *botnets* (see the earlier section "Botnets").

In addition to DoS and DDoS, there is a third form known as *distributed reflective denial-of-service (DRDoS)*. This form of attack employs an *amplification* or *bounce* network that is an unknowing participant, unfortunately able to receive broadcast messages and create message responses, echoes, or bounces. In effect, the attacker sends spoofed message packets to the amplification network's broadcast address. This causes each single inbound received packet to be distributed to all the hosts in that network (which could be in the 10,000 or 100,000 range). Each host then responds to each packet, but because the source of the original packet was falsified, the response goes to the victim instead of the true sender (the attacker). So, what originated from the attacker as a single packet is transformed into numerous packets exiting the amplification network and ultimately flooding the victim's communication link.

FIGURE 1.2 DDoS attack



There are numerous specific DoS, DDoS, and DRDoS attack tools and methods. Here are a few that you should be able to recognize:

Smurf This form of DRDoS uses ICMP echo reply packets (ping packets). The attacker sends ICMP Type 8 echo request packets to several intermediary networks' broadcast addresses with the source IP address set to the primary victim. This causes multiple ICMP Type 0 replies to be sent to the victim. A smurf attack is also known as an amplification attack. See Figure 1.3 for an example.

Fraggle This form of DRDoS uses UDP packets commonly directed to port 7 (echo port) or 19 (chargen [character generator] port).

SYN flood This type of attack is an exploitation of a TCP three-way handshake. Every TCP session starts with the client sending a SYN (synchronize) packet to a server, the server responding with a SYN/ACK (synchronize/acknowledgment) packet, and the client sending a final ACK packet. The attack consists of the attacker posing as a client and sending numerous SYN packets but never any final ACK packets. This causes the server to consume all network resources by opening numerous incomplete communication sessions. Figure 1.4 shows an example of a TCP SYN flood attack.

FIGURE 1.3 A smurf attack underway against a network

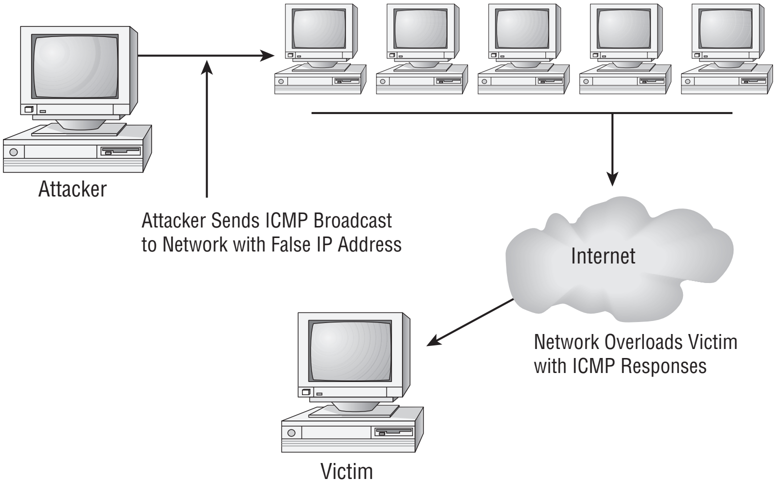
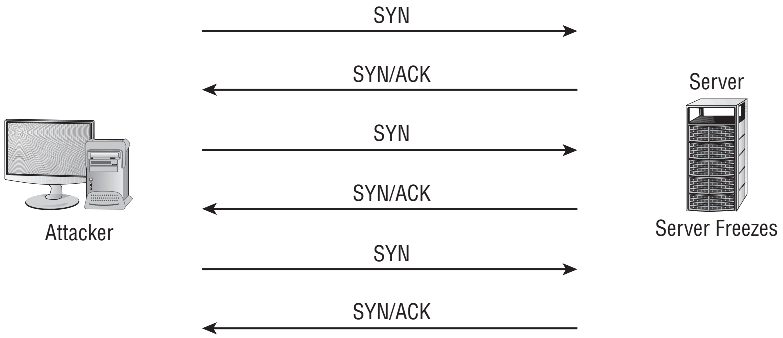

Attacker
Attacker Sends ICMP Broadcast to Network with False IP Address
Internet
Network Overloads Victim with ICMP Responses
Victim

FIGURE 1.4 TCP SYN flood attack



Ping of death The attacker sends oversized ping packets to the victim. The victim doesn't know how to handle invalid packets, and it freezes or crashes.

Xmas attack The *Xmas attack* is actually an Xmas scan. It's a form of port scanning that can be performed by a wide number of common port scanners, including Nmap, Xprobe, and hping2. The Xmas scan sends a TCP packet to a target port with the flags of URG, PSH, and FIN all turned on. This creates a flag byte of 00101001 in the TCP header, which is said to be representative of alternating flashing lights on a Christmas tree. According to the TCP specifications, ports should ignore any invalid construction of a packet if the port is open and send an RST back if the port is closed. This is true of all systems except for Windows OSs, which send RSTs for many invalid packets even if the port is open. An Xmas attack (or scan) occurs when someone sends Xmas-flagged packets to one or more ports on a computer. If the level of scanning packets is significant, this can affect the performance of the targeted system or consume some or all of the available bandwidth. Thus, an Xmas scan can escalate to a DoS and thus be considered an Xmas attack.

SYN floods, teardrops, land attacks, ping floods, pings of death, bonks, and boinks are typically labeled DoS attacks, but they can be waged as a DDoS if the attacker compromises several intermediary systems and uses those as launching points to attack the victim. Fortunately, most of the basic DoS attacks that exploit error-handling procedures (such as ping of death, land attack, teardrop, bonk, boink, and so on) are now automatically handled by improved versions of the protocols installed in the OS. However, many of the current DDoS and DRDoS attacks aren't as easy to safeguard against.

Some countermeasures and safeguards against these attacks are as follows:

- Work out a response plan with your ISP.
- Add firewalls, routers, and intrusion detection systems (IDSs) that detect DoS traffic and automatically block the port or filter out packets based on the source or destination address.
- Disable echo replies on external systems.
- Disable broadcast features on border systems.
- Block spoofed packets from entering or leaving your network.
- Keep all systems patched with the most current security updates from vendors.

Unfortunately, as security professionals develop better defenses, preventions, and detections of the various types of DoS attacks, so hackers are actively developing new means and methods of waging attacks that get around those defenses. In the fall of 2016, the most significant DDoS flooding attack ever took place in response to a blog posting by Brian Krebs on his site `https://www.krebsonsecurity.com/`, where he revealed the "secret" of the existence of DoS as a Service. This attack generated a peak load of 620 Mbps. For details about this attack and related concerns, check out `https://krebsonsecurity.com/2016/09/krebsonsecurity-hit-with-record-ddos/` and `https://krebsonsecurity.com/2017/02/how-google-took-on-mirai-krebsonsecurity/`.

DDoS

Distributed denial of service (DDoS) was discussed in the previous section.

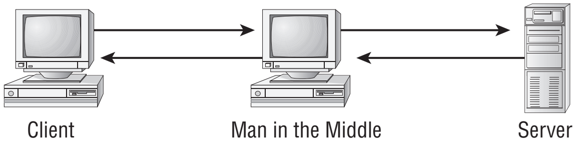
Man-in-the-middle

A *man-in-the-middle attack* is a communications eavesdropping attack. Attackers position themselves in the communication stream between a client and server (or any two communicating entities). The client and server believe that they're communicating directly with each other—they may even have secured or encrypted communication links. However, the attacker can access and potentially modify the communications.

Man-in-the-middle (MitM) attacks range from very simple to quite complex. Some MitM attacks exploit DHCP weaknesses to distribute false IP configurations, such as defining the attack system's IP address as the victim's default gateway. Other forms of MitM attacks focus on poisoning name-resolution systems—such as Domain Name System (DNS), Address Resolution Protocol (ARP), NetBIOS, and Windows Internet Name Service (WINS). Still other MitM attacks include the use of false proxy server settings or using MAC (media access control) address spoofing. Any form of MitM may fool the client into perceiving the attacker as the server and fool the server into perceiving the attacker as the client, or simply cause the attacker to be a transparent node along the communication pathway. When that charade is successful, the client submits its logon credentials to the fake server (the masked attacker), which in turn sends the credentials to the actual server while masquerading as the actual client. As a result, the client establishes a communication link (maybe even an encrypted link) with the attacker, and the attacker establishes a communication link with the server. As data is transmitted in either direction between the true client and server systems, the attacker can read and access all the data and can choose to modify the traffic to further the subterfuge.

Figure 1.5 shows a man-in-the-middle attack.

FIGURE 1.5 A man-in-the-middle attack occurring between a client and a web server



Such attacks are usually most successful when routing and name-resolution systems are first compromised in order to position the attacker before the client-to-server communication is initiated. However, in some cases man-in-the-middle attacks can be conducted against existing client-server communication links (usually assuming they aren't encrypted). One situation where this is possible is with open wireless connections. A deauthentication packet can be sent by an attacker to a wireless client victim; then, as they attempt to reconnect, the attacker fools the victim system into establishing a connection through it, instead of linking up with the valid base station. Even this style of MitM can be effective because it takes only a fraction of a second for the entire process to occur, and unless the short-lived disconnect interrupts an active data transfer, the user won't even notice the event.

Countermeasures to man-in-the-middle attacks include strong encryption protocols (such as IPsec, SSH, and TLS) and the use of strong authentication, such as Domain Name System Security Extensions (DNSSEC) and mutual certificate authentication.

Related to man-in-the-middle is the transitive access attack, or exploitation. Transitive access is a potential backdoor or way to work around traditional means of access control. The idea is that user A can use process B, and process B can use or invoke process C, and process C can access object D (see Figure 1.6). If process B exits (or is otherwise inaccessible) before process C completes, process C may return access to object D back to user A, even if user A doesn't directly or by intent have access to object D (see Figure 1.7). Some forms of access control don't specifically prevent this problem.

All subject to object accesses should be validated before access is granted, rather than relying on previous verifications.

FIGURE 1.6 Transitive access

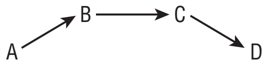
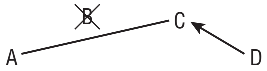


FIGURE 1.7 A transitive access exploit



Buffer overflow

Software exploitation attacks are directed toward known flaws, bugs, errors, and oversights, or toward normal functions of the OS, protocols, services, or installed applications. One of the most common forms of software exploitation is a *buffer overflow attack*.

A buffer overflow attack occurs when an attacker submits data to a process that is larger than the input variable is able to contain. Unless the program is properly coded to handle excess input, the extra data is dropped into the system's execution stack and may execute as a fully privileged operation. Buffer overflow attacks can result in system crashes, corrupted data, user privilege escalation, or just about anything a hacker can think of. The only countermeasures to buffer overflow attacks are to patch the software when issues are discovered and to properly code software to perform input-validation checks before accepting input for processing.

NOTE

If you are the user of open source software, then—assuming you know how to code—you have the opportunity to fix flawed code yourself, rather than having to rely on the original programmer or vendor.

Once a weakness is discovered in software, a hacker can craft an exploit or attack tool. These tools are easily accessible and widely distributed on the Internet. They allow anyone to grab the tool and point it at a victim they wish to attack, even when they have neither the knowledge of how the attack actually works nor the skill to craft an attack tool themselves. Those who only use preexisting exploitation tools are known as *script kiddies*.

A buffer overflow occurs when a program receives input that is larger than it was designed to accept or process. The extra data received by the program is shunted over onto the CPU without any security restrictions; it's then allowed to execute (assuming it's a valid command, script, system call, and so on) with system-level privileges. A hacker can achieve many possible results with a buffer overflow: crashing a program, freezing or crashing a system, opening a port, disabling a service, creating a user account, elevating the privileges of an existing user account, accessing a website, or executing a utility. Clever attackers can do just about anything they wish if they can execute a command or script with unrestricted access to a system.

Sometimes a buffer overflow attack can be considered a form of DoS attack, because a buffer overflow occurs when a system receives more data than it can handle (a bit like a flooding attack). This is especially true when the buffer overflow event prevents a system from processing legitimate data or requests.

Poor programming quality controls and a lack of input validation checks in software lead to buffer overflow attacks. Unfortunately, most commercial software is vulnerable to buffer overflow attacks; web server software is attacked most frequently. Fortunately, buffer overflow vulnerabilities are often easily patched with vendor updates or by skilled users when using open source software.

Injection

An *injection attack* is any exploitation that allows an attacker to submit code to a target system in order to modify its operations and/or poison and corrupt its data set. There are a wide range of potential injection attacks. Typically an injection attack is named after the type of backend system it takes advantage of or the type of payload delivered (injected) onto the target. Examples include SQL injection, LDAP injection, XML injection, command injection, HTML injection, code injection, and file injection. A few of these are presented in more detail in this section.

SQL injection attacks are even riskier than XSS attacks (see the following section) from an organization's perspective, because the targets of a SQL injection attack are organizational assets, whereas the targets of an XSS attack are customers or visitors to a website. SQL injection attacks use unexpected input to alter or compromise a web application. However, instead of using this input to attempt to fool a user, SQL injection attacks use it to gain unauthorized access to an underlying database and related assets.

In the early days of the web, all web pages were *static*, or unchanging. Webmasters created web pages containing information and placed them on a web server, where users could retrieve them using their web browsers. The web quickly outgrew this model because users wanted the ability to access customized information based on their individual needs. For example, visitors to a bank website aren't interested only in static pages containing information about the bank's locations, hours, and services. They also want to retrieve dynamic content containing information about their personal accounts. Obviously, the webmaster can't possibly create pages on the web server for each individual user with that user's personal account information. At a large bank, that would require maintaining millions of pages with up-to-the-minute information. That's where dynamic web applications come into play.

Web applications take advantage of a database to create content on demand when the user makes a request. In the banking example, the user logs in to the web application, providing an account number and password. The web application then retrieves current account information from the bank's database and uses it to instantly create a web page containing the user's current account information. If that user returns an hour later, the web server repeats the process, obtaining updated account information from the database.

What does this mean to you as a security professional? Web applications add complexity to the traditional security model. The web server, as a publicly accessible server, belongs in a separate network zone from other servers, commonly referred to as a *demilitarized zone (DMZ)*. The database server, on the other hand, isn't meant for public access, so it belongs on the internal network or at least a secured subnet separated from the DMZ. The web application needs access to the database, so the firewall administrator must create a rule allowing access from the web server to the database server. This rule creates a potential path for Internet users to gain access to the database server.

If the web application functions properly, it allows only authorized requests to the database. However, if there is a flaw in the web application, it may let individuals tamper with the database in an unexpected and unauthorized fashion through the use of SQL injection attacks. These attacks allow a malicious individual to perform SQL transactions directly against the underlying database. SQL injection attacks might enable an attacker to bypass authentication, reveal confidential data from database tables, change existing data, add new records into the database, destroy entire tables or databases, and even gain command line–like access through certain database capabilities (such as command shell stored procedures).

You can use two techniques to protect your web applications against SQL injection attacks:

Perform input validation. Input validation lets you limit the types of data a user provides in a form. There are numerous variations of input injection or manipulation attacks that require a broad-spectrum defense approach, including whitelisting and blacklisting filters. The primary forms of input sanitization that should be adopted include limiting the length of input, filtering on known malicious content patterns, and escaping *metacharacters*.

Metacharacters

Metacharacters are characters that have been assigned special programmatic meaning. Thus, they have special powers that standard, normal characters do not have. There are many common metacharacters, but typical examples include single and double quotation marks; open/close square brackets; the backslash; the semicolon; the ampersand; the caret; the dollar sign; the period, or dot; the vertical bar, or pipe symbol; the question mark; the asterisk; the plus sign; open/close curly braces; and open/close parentheses: ‘ “ [ ] \ ; & ^ $. | ? * + { } ( ).

Escaping a metacharacter is the process of marking the metacharacter as merely a normal or common character, such as a letter or number, thus removing its special programmatic powers. This is often done by adding a backslash in front of the character (\&), but there are many ways to escape metacharacters based on the programming language or execution environment.

Limit account privileges. The database account used by the web server should have the smallest set of privileges possible. If the web application needs only to retrieve data, it should have that ability only.

Ultimately, SQL injection is a vulnerability of the script used to handle the interaction between a front end (typically a web server) and the backend database. If the script was written defensively and included code to escape (invalidate or reject) metacharacters, SQL injection would not be possible.

LDAP injection is a variation of an input injection attack; however, the focus of the attack is on the backend of an LDAP directory service rather than a database server. If a web server front end uses a script to craft LDAP statements based on input from a user, then LDAP injection is potentially a threat. Just as with SQL injection, sanitization of input and defensive coding are essential to eliminate this threat.

XML injection is another variant of SQL injection, where the backend target is an XML application. Again, input sanitization is necessary to eliminate this threat.

Directory Traversal/Command Injection

A *directory traversal* is an attack that enables an attacker to jump out of the web root directory structure and into any other part of the filesystem hosted by the web server's host OS. A common, but historical, version of this attack was against IIS 4.0, hosted by Windows NT 4.0 Server. The attack used a modified URL to directory-traverse out of the web root, into the main OS folders, in order to access the command prompt executable. For example:

```
http://victim.com/scripts/..%c0%af../..%c0%af../..%c0%af../..%c0%af../
..%c 0%af../..%c0%af../winnt/system32/cmd.exe?/c+tftp+-i+get+exploit.exe
```

This URL includes a UNICODE equivalent of the "change to parent directory" command, which is `../` in ASCII, and also notice it uses the metacharacter of percent (%). This URL not only performed directory traversal, but also granted the attacker the ability to perform command injection. The example shows a command injection triggering a TFTP Get operation to download an exploit tool onto the victim web server. Any command that could be executed under the privileges of the IIS service and be crafted within the limitations of a URL could be used. The example performs a single directory listing of the C root. But with minor tweaking, TFTP commands could be used to download hacker tools to the target and subsequently launch those tools to grant greater remote control or true command shell access. This attack can be stopped with metacharacter escaping or filtering.

Cross-site scripting

Cross-site scripting (XSS) is a form of malicious code-injection attack in which an attacker is able to compromise a web server and inject their own malicious code into the content sent to other visitors. Hackers have discovered numerous and ingenious methods for injecting malicious code into websites via CGI scripts, web server software vulnerabilities, SQL injection attacks, frame exploitation, DNS redirects, cookie hijacks, and many other forms of attack. A successful XSS attack can result in identity theft, credential theft, data theft, financial losses, or the planting of remote-control software on visiting clients.

For the administrator of a website, defenses against XSS include maintaining a patched web server, using web application firewalls, operating a host-based intrusion detection system (HIDS), auditing for suspicious activity, and, most importantly, performing server-side input validation for length, malicious content, and metacharacter filtering. As a web user, you can defend against XSS by keeping your system patched, running antivirus software, and avoiding non-mainstream websites. There are add-ons for some web browsers, such as NoScript for Firefox and uBlock Origin for Chrome, that allow only scripts of your choosing to be executed.

Cross-site request forgery

Cross-site request forgery (XSRF) is an attack that is similar in nature to XSS. However, with XSRF, the attack is focused on the visiting user's web browser more than the website being visited. The main purpose of XSRF is to trick the user or the user's browser into performing actions they had not intended or would not have authorized. This could include logging out of a session, uploading a site cookie, changing account information, downloading account details, making a purchase, and so on. One form of XSRF infects a victim's system with malware that stays dormant until a specific website is visited. Then the malware forges requests as the user in order to fool the web server and perform malicious actions against the web server and/or the client.

One example of an exploit that used XSRF is Zeus, which would hide on a victim's system until the user visited their online bank site; then, after it checked their account balance and determined their bank account number, those details would be sent to the controlling attacker, who would initiate an ACH money transfer to another bank. Thus, this is an example of malware that assists in stealing money directly out of the victim's account.

Website administrators can implement prevention measures against XSRF by requiring confirmations or reauthentication whenever a sensitive or risky action is requested by a connected client. This could include requiring the user to reenter their password, sending a code to the user via text message or email that must be provided back to the website, triggering a phone call–based verification, or solving a CAPTCHA (a mechanism to differentiate between humans and software robots). Another potential protection mechanism is to add a randomization string (called a *nonce*) to each URL request and session establishment and check the client HTTP request header referrer for spoofing. End users can form more secure habits, such as running antimalware scanners; using a HIDS; running a firewall; avoiding non-mainstream websites; always logging off from sites instead of closing the browser, closing the tab, or moving on to another URL; keeping browsers patched; and clearing out temporary files and cached cookies regularly.

Privilege escalation

Privilege escalation occurs when a user is able to obtain greater permissions, access, or privileges than they're assigned by an organization. Privilege escalation can occur accidentally or due to administrative oversight, but usually this term refers to the specific and intentional abuse of a system to steal access.

Privilege escalation can take place via weaknesses in the OS. Often a hacker tool is used to exploit a programming flaw or buffer overflow that may allow the attacking user to obtain permanent or temporary access to the administrators group. This form of attack is known as vertical privilege escalation, since the current low-level user or access is itself elevated to a higher level of access. In other cases, privilege escalation occurs through identity theft or credential compromise, such as keystroke capturing or password cracking. This form of attack is known as horizontal privilege escalation, since the attacker switches over to another user account to gain a higher level of access.

Privilege escalation is a violation of security. Specifically, it's a breach of authorization restrictions and may be a breach of authentication. In order to prevent or stop privilege escalation, all OSs should be kept current with patches from the vendor. Additionally, auditing and monitoring should be configured to watch for privilege-escalation symptoms. These include repeated attempts to perform user account management by nonadministrators as well as repeated attempts to access resources beyond a user's assigned authorization level.

ARP poisoning

Address Resolution Protocol (ARP) poisoning is the act of falsifying the IP-to-MAC address resolution system employed by TCP/IP. ARP operates at Layer 2, the Data-Link layer of the OSI model. ARP is responsible for resolving IP addresses into MAC addresses. This allows Layer 2 to physically address transmissions before sending them to the Physical layer (Layer 1). Similar to DNS, ARP resolution is a multistep process:

1. Check the local ARP cache.
2. If that fails, transmit an ARP broadcast.

The ARP broadcast is a transmission to all possible recipients in the local subnet (more accurately, the ARP broadcast is received by all members of the same Ethernet broadcast domain, but that is almost always the same group of systems that is contained in the local subnet), asking all hosts if they own the IP address in question. If the owner of the IP address is present, it responds with a direct reply to the source system with its MAC address.

MAC addresses are essential for TCP/IP communications because transmissions occur from host to host and router to router, based not solely on IP address but primarily on

MAC addresses. When a host sends data to another host, if that host is in the same subnet, it transmits the signal from its MAC-addressed network interface card (NIC) to the target's MAC-addressed NIC. If the target is in a different subnet, it sends the message to the MAC-addressed NIC of the default gateway (which is the router interface in that subnet). Then, that router takes over and tries to find the target host, either with a subnet directly off one of its ports or by sending the message to another router that may have a greater chance of being connected to the target host's subnet. Without proper ARP activity, this process isn't possible.

ARP poisoning can take place in many ways. The most common ways are to poison the local ARP cache or to transmit poisoned ARP replies or announcements. In either case, if a host obtains a false MAC address for an IP address, its transmission is likely to go to the wrong location. This tactic is most effective within a single subnet, but it does have an effect across multiple subnets. ARP poisoning is commonly used in active sniffing attacks where false ARP announcements are used to redirect traffic to the hacker-controlled system, allowing the attacker to view the contents of all transactions. The attack must then forward each Ethernet frame to the correct MAC address destination in order to prevent a DoS and maintain the façade that nothing abnormal is occurring.

One popular tool used to monitor for ARP poisoning is arpwatch. However, the best defense against ARP-based attacks, including ARP poisoning, is port security on the switch. Switch port security can prohibit communications with unknown, unauthorized, rogue devices and may be able to determine which system is responding to all APR queries and block ARP replies from the offending system.

Amplification

In an *amplification attack* the amount of work or traffic generated by an attacker is multiplied in order to cause a significant volume of traffic to be delivered to the primary victim. An amplification attack can also be known as a reflective or bounce attack. Most amplification attacks involve innocent third-party systems or networks, which are used to cause the multiplicity of responses. An historical example of an amplification attack is the Smurf DRDoS discussed earlier. Any attack where a single packet from the attacker generates two or more packets sent to the primary target can be described as an amplification attack. These types of attacks grant the attacker more perceived power and capability than they would have without the multiplication benefit. While the Smurf DRDoS uses the broadcast address of intermediate networks for its amplification, other attacks can use a request, which generates a larger reply.

An example of this latter type of amplification attack was performed in February 2014. The attack took advantage of 4,529 NTP (Network Time Protocol) servers by sending an administrator request of MONLIST. This command generates a report of the last 600 IP addresses of systems requesting information from the NTP server. This caused a flood of traffic against the target victim that reached a peak of 400 Gbps.

Amplification attacks are popular among attackers because they assist in increasing the attack's effective power against larger targets. There have been a few non-amplification attacks that have caused larger levels of traffic against victims—such as the Mirai botnet, which in September 2016 generated 620 Gbps against `krebsonsecurity.com`—but only a few.

DNS poisoning

DNS poisoning is the act of falsifying the DNS information used by a client to reach a desired system. It can take place in many ways. Whenever a client needs to resolve a DNS name into an IP address, it may go through the following process:

1. Check the local cache (which includes content from the HOSTS file).
2. Send a DNS query to a known DNS server.
3. Send a broadcast query to any possible local subnet DNS server. (This step isn't widely supported.)

If the client doesn't obtain a DNS-to-IP resolution from any of these steps, the resolution fails and the communication can't be sent. DNS poisoning can take place at any of these steps, but the easiest way is to corrupt the HOSTS file or the DNS server query.

There are many ways to attack or exploit DNS. An attacker might use one of these techniques:

Deploy a rogue DNS server (also known as DNS spoofing or DNS pharming). A rogue DNS server can listen in on network traffic for any DNS query or specific DNS queries related to a target site. Then the rogue DNS server sends a DNS response to the client with false IP information. This attack requires that the rogue DNS server get its response back to the client before the real DNS server responds. Once the client receives the response from the rogue DNS server, the client closes the DNS query session, which causes the response from the real DNS server to be dropped and ignored as an out-of-session packet.

DNS queries are not authenticated, but they do contain a 16-bit value known as the Query ID or QID. The DNS response must include the same QID as the query to be accepted. Thus, a rogue DNS server must include the requesting QID in the false reply.

Perform DNS poisoning. *DNS poisoning* involves attacking the real DNS server and placing incorrect information into its zone file. This causes the real DNS server to send false data back to clients.

Alter the HOSTS file. Modifying the HOSTS file on the client by placing false DNS data into it redirects users to false locations.

Corrupt the IP configuration. Corrupting the IP configuration can result in a client having a false DNS server definition. This can be accomplished either directly on the client or on the network's DHCP server.

Use proxy falsification. This method works only against web communications. This attack plants false web proxy data into a client's browser, and then the attacker operates the rogue proxy server. A rogue proxy server can modify HTTP traffic packets to reroute requests to whatever site the hacker wishes.

Although there are many DNS poisoning methods, here are some basic security measures you can take that can greatly reduce their threat:

- Limit zone transfers from internal DNS servers to external DNS servers. This is accomplished by blocking inbound TCP port 53 (zone transfer requests) and UDP port 53 (queries).
- Limit the external DNS servers from which internal DNS servers pull zone transfers.

- Deploy a *network intrusion detection system (NIDS)* to watch for abnormal DNS traffic.
- Properly harden all DNS, server, and client systems in your private network.
- Use DNSSEC to secure your DNS infrastructure.
- Require internal clients to resolve all domain names through the internal DNS. This will require that you block outbound UDP port 53 (for queries) while keeping open outbound TCP port 53 (for zone transfers).

Another attack closely related to DNS poisoning and/or DNS spoofing is *DNS pharming*. *Pharming* is the malicious redirection of a valid website's URL or IP address to a fake website that hosts a false version of the original, valid site. This is often part of a phishing attack where the attacker is attempting to trick victims into giving up their logon credentials. If potential victims aren't careful or paying attention, they may be tricked into providing their logon information to the false, pharmed website. Pharming typically occurs either by modifying the local HOSTS file on a system or by poisoning or spoofing DNS resolution. Pharming is an increasingly problematic activity because hackers have discovered means to exploit DNS vulnerabilities to pharm various domain names for large groups of targeted users.

For a detailed review of DNS and its vulnerabilities, read "An Illustrated Guide to the Kaminsky DNS Vulnerability" at `www.unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html`.

Domain hijacking

Domain hijacking, or domain theft, is the malicious action of changing the registration of a domain name without the authorization of the valid owner. This may be accomplished by stealing the owner's logon credentials; using XSRF, session hijacking, or MitM; or exploiting a flaw in the domain registrar's systems.

Sometimes when another person registers a domain name immediately after the original owner's registration expires this is called domain hijacking, but it should not be. This is a potentially unethical practice, but it is not an actual hack or attack. It is taking advantage of the oversight of the original owner failing to manually extend their registration or configure auto-renewal. If an original owner loses their domain name by failing to maintain registration, there is often no recourse other than to contact the new owner and inquire regarding re-obtaining control. Many registrars have a "you snooze, you lose" policy for lapsed registrations.

When an organization loses their domain and someone else takes over control, this can be a devastating event both to the organization as well as its customers and visitors. The original website or online content will no longer be available (or at least not available on the same domain name). And the new owner might host completely different content or host a false duplicate of the previous site. This later activity might result in fooling visitors, similar to a phishing attack, where PII (personally identifiable information) might be extracted and collected.

Man-in-the-browser

The *man-in-the-browser* (MitB, MiTB, MiB, MIB) attack is effectively an MitM attack. The only real distinction is that the middleman malware is operating on the victim's system, where it is able to intercept and manipulate communications immediately after they leave the browser and before they exit the network interface. Often the MitB is a false proxy system where even encrypted connections can be infiltrated through the presentation of a false, cloned certificate.

The main defenses against MitB attacks are to avoid risky behaviors in order to minimize exposure to malware infection, run an antimalware scanner, use an HIDS, and have a stateful inspection firewall.

LSO (Local Shared Object)

LSOs (local shared objects) are small files or data sets that websites may store on a visitor's computer through the Adobe Flash Player. LSOs, also known as Flash cookies, are generally used to store user preferences and settings, but they do have some risk. LSOs can be used to track a user's web activities and are not cleared or removed when a browser's HTML cookies are cleared.

There are some options to limit the use of LSOs through Adobe Flash configuration settings. However, after each update of Flash, those settings are reset to the default of "Allow". And the most recent versions of Flash will not store LSOs while the browser is operating in privacy or incognito mode.

Malicious Add-ons

Most browsers and many other applications now allow for expansion through downloadable add-ons, *BHOs (browser helper objects)*, *plug-ins,* or *expansion packs*. These add-ons are additional targets for attackers. Hackers have crafted false versions of add-ons, converted add-ons into Trojan horses, and written add-ons to look legitimate but be nothing more than attack code. The purpose is to trick unsuspecting victims into installing the malicious add-ons so the attackers can either gain access to information or take control of the victim's system or identity. Browser add-on stores have started to require signing of add-ons to help address this issue, but it's more important than ever to be cautious about installing anything, to install only software from trusted sources, and to run current antivirus and antimalware scanners.

Header Manipulation

Header manipulation is a form of attack in which malicious content is submitted to a vulnerable application, typically a web browser or web server, under the guise of a valid HTML/HTTP header value. Header manipulation is usually a means to some other nefarious end, such as cross-user defacement, cache poisoning, cross-site scripting, page hijacking, cookie manipulation, open redirects, and so on. In most cases, preventing this attack involves using updated browsers/servers, filtering content from visitors, and rejecting/ignoring any header in violation of HTTP/HTML specifications.

Zero day

Zero-day attacks are newly discovered attacks for which there is no specific defense. A *zero-day exploit* aims to exploit flaws or vulnerabilities in targeted systems that are unknown or undisclosed to the world in general. Zero-day also implies that a direct or specific defense to the attack does not yet exist; thus, most systems with the targeted vulnerable asset are at risk. Another way of describing a zero-day attack is that it is one for which the vendor of the target product has not yet released a patch or update to address the vulnerability; however, there may be IDS or firewall filters that can reduce the risk of attack or exploit while the world waits for the vendor to resolve the concern.

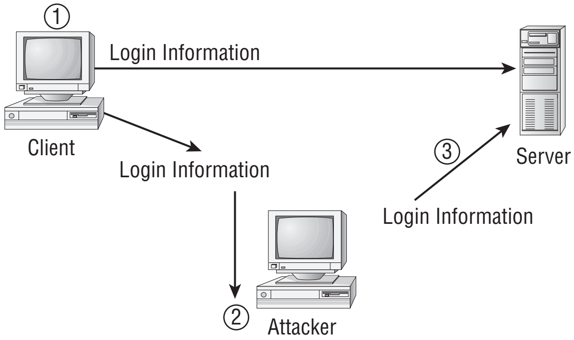
Many attacks take advantage of zero-day vulnerabilities—security flaws discovered by hackers that have not been thoroughly addressed by the security community. There are two main reasons systems are affected by these vulnerabilities. First, it may be the result of the necessary delay between the discovery of a new type of malicious code and the issuance of patches and antivirus updates. Second, it may be due to slowness in applying updates on the part of system administrators. The existence of zero-day vulnerabilities makes it vital that you have a strong patch-management program in your organization that ensures the prompt application of critical security updates. Additionally, you may wish to use a vulnerability scanner to scan your systems for known security issues on a regular basis.

Replay

A *replay attack* is just what it sounds like: an attacker captures network traffic and then replays (retransmits) the captured traffic in an attempt to gain unauthorized access to a system. Most commonly, the attacker focuses on network traffic that is the exchange between a client and server performing authentication. If an attacker can capture the authentication traffic—especially the packets containing the logon credentials, even if they're more than just username and password (such as certificates, token responses, or biometric values)—then a replay attack may grant the attacker the ability to log on to a system by retransmitting the captured packets.

Figure 1.8 shows a replay attack. As the client transmits its logon credentials to the server (1), the attacker intercepts and eavesdrops on that transmission (2) and then later can replay those captured authentication packets against the server to falsify a logon as the original client (3).

FIGURE 1.8 A replay attack occurring



If a replay attack succeeds, the attacker gains the same level of access as the user that originally submitted the authentication information. Fortunately, most modern OSs, networks, protocols, services, and applications use various replay-protection mechanisms to directly prevent such attacks. Common countermeasures are packet sequencing, time stamps, challenge-response, and ephemeral session encryption. Packet sequencing ensures that any packet received that isn't in the proper order (or within a reasonable margin) is dropped and ignored. Packet time stamps ensure that any packet received outside of a specific time window is dropped and ignored. A great example of this is Kerberos, which isn't vulnerable to replay attacks, thanks to its use of time stamps.

Challenge-response is a type of authentication where the server generates and issues a random number challenge to the connecting client. The client uses the challenge number and the hash of the user's password (or other authentication factor) to generate a response. The response is sent back to the server, where it is compared with the expected response generated by the server using the challenge number and the credentials pulled from the user account database. Since each challenge is valid only once and each challenge is randomly selected, replay attacks are not possible.

Ephemeral session key is the term for the use of DHE or ECDHE (see the Chapter 6 section, "Diffie-Hellman") to generate random, nonrepeating, nonreusable, nonpredictable, session-specific symmetric encryption keys. Meeting these criteria means that each authentication session is encrypted, and that encryption is valid only once. Again, it's a reliable method to thwart replay attacks.

Pass the hash

Pass the hash is an authentication attack that potentially can be used to gain access as an authorized user without actually knowing or possessing the plain text of the victim's credentials. This attack is mostly aimed at Windows systems, which maintain a set of cached credentials (this is the item being referenced with the term "hash" in the attack name, which is also known as the authentication token) on client systems for the Windows domains they have authenticated into. The cached credentials are used to grant a user access to the local system and the network in the event the authenticating domain controllers are not available the next time the user attempts to log in. In such a situation, the cached credentials are used, and whenever the domain controllers come back online, the user is automatically accepted by the domain controllers as having been properly authenticated because the user was granted access through the cached credentials from their previous successful domain logon. Although repeated attempts to secure this process have been implemented by Microsoft, hackers continue to exploit this fault-tolerant feature of Windows operating systems.

An attacker extracts the cached credentials from the Registry of a victim's system and then uses those credentials on their own rogue domain client. This may fool the domain controller into accepting the attacker as the authorized user, even though the attack did not actually participate in any authentication process.

Mitigations to this attack include disabling cached credentials, requiring network level authentication, and forcing NTLMv2 (disabling NTLM and LM). Restricted Admin mode is also a good defensive measure. Implementing two-factor authentication can also stop this in some cases.

Hijacking and related attacks

Hijack attacks are those where an attacker takes over control of a session from a valid user. Some forms of hijacking disconnect the client, whereas others grant the attacker a parallel connection into the system or service. This section includes definitions of several hijack-related exploitations.

Clickjacking

Clickjacking is a web page–based attack that causes a user's click to link someplace other than the user intended. This is often accomplished by using hidden or invisible layovers, frame sets, or image maps. When a user sees such an item or link, and then clicks their mouse pointer, the click is intercepted by the invisible or hidden layer, and thus the request is for something other than what the user actually intended.

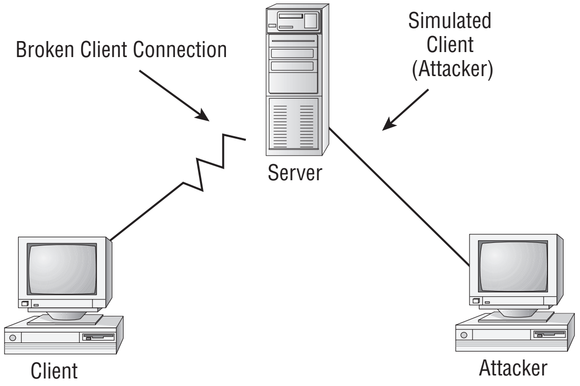
Clickjacking can be used to perform phishing attacks, hijacking, MitM, and MitB. Examples of clickjacking include hiding an Amazon Buy button behind an image of a Play button; tricking users into enabling their microphone or web camera; causing users to set their social media profiles to public; downloading malicious software, including backdoors, rootkits, and ransomware; and falsely generating advertisement clicks.

Session hijacking

TCP/IP hijacking, *or session hijacking*, is a form of attack in which the attacker takes over an existing communication session. The attacker can assume the role of the client or the server, depending on the purpose of the attack. In most forms of session hijacking, the other partner (most often the client) in the communication is disconnected—they're aware that they're no longer communicating and that their session was interrupted. However, they may not immediately realize that they were the collateral damage in a session hijacking attack. Some of the tools that can be used to perform session hijacking are Ettercap, Cain, Juggernaut, and Hunt.

Figure 1.9 shows a TCP/IP hijacking attack.

FIGURE 1.9 TCP/IP hijacking attack



Countermeasures to TCP/IP hijacking attacks include using encrypted protocols and performing periodic, mid-stream reauthentication during a session. Additionally, modern or secured protocols are often designed with preventive features that make session hijacking very difficult or impossible. These features include complex nonlinear sequencing rules as well as time stamps with short timeout values.

URL hijacking

URL hijacking, or *typo squatting*, is a practice employed to capture traffic when a user mistypes the domain name or IP address of an intended resource. A squatter predicts URL typos and then registers those domain names to direct traffic to their own site. This can be done for competition or for malicious intent. The variations used for typo squatting include common misspellings (such as `googel.com`), typing errors (such as `gooogle.com`), variations on a name or word (for example, plurality, as in `googles.com`), and different top-level domains (such as `google.org`).

URL hijacking is also the term applied to the practice of displaying a link or advertisement that looks like that of a well-known product, service, or site, but when clicked redirects the user to an alternate location, service, or product. This may be accomplished by posting sites and pages and exploiting SEO (search engine optimization) to cause your content to occur higher in search results, or through the use of adware that replaces legitimate ads and links with those leading to alternate or malicious locations.

Cookies

A *cookie* is a tracking mechanism developed for web servers to monitor and respond to a user's serial viewing of multiple web pages. A cookie is often used to maintain an e-commerce shopping cart, focus product placement, or track your visiting habits. However, the benign purposes of cookies have been subverted by malevolent entities. Now cookies are a common means of violating your privacy by gathering information about your identity, logon credentials, surfing habits, work habits, and much more.

A cookie can easily be exploited against a web browser to gather sufficient information about a user to allow the attacker to impersonate the victim online. It's generally recommended that you block third-party cookies from everyone and first-party cookies from all but the most trusted sites. Trusted sites are usually those entities that protect your identity by not including such details in a cookie. Instead, these sites only place a session ID in the cookie and keep all of your personal information in a backside database. If you don't allow trusted first-party cookies (aka *session cookies*), functions such as e-commerce shopping carts, online banking, and posting to discussion forums will be disabled.

Cookies can be used in a hijack of a web service connection, where the attacker gains a parallel connection while the original user maintains their connection. This is accomplished by the attacker stealing a copy of the cookie while it's in transit between the valid client and server or directly off the client's storage device. If the cookie serves as an access token, then anyone with possession of it will be recognized by the server as the original authenticated client. The attacker places the cookie on their system and uses their own browser to visit the target server, which mistakenly assumes the attacker is simply another valid connection from the previously authenticated client. Websites should be designed to detect and prevent multiple simultaneous (concurrent) connections.

The Risk of Email Attachments

Because email is so widely used, it has become the most prevalent delivery vehicle for malicious code such as viruses, logic bombs, and Trojan horses. Many of these email-delivered malware items can be used to perform MitM, MitB, or hijacking attacks. To combat this threat, you should deploy an antivirus scanner to scan email content and attachments. You should even consider stripping or blocking email attachments (especially those with known extensions of scripts or executables) as they enter your network (on an email gateway, firewall, and so on). It's always the more secure option to scan, check, and if necessary, strip email on SMTP servers before it reaches an end user's client system.

Typo squatting

See the previous section, "URL Hijacking."

Driver manipulation

Some forms of malicious code or attacker intrusions will take advantage of a form of software manipulation known as *driver manipulation*. Driver manipulation occurs when a malicious programmer crafts a system or device driver so that it behaves differently based on certain conditions. For example, a system benchmark tool may be used to test the performance of a computer, but if the drivers are tuned to provide favorable performance only when the specific benchmarking tool is used, this is an abuse of the evaluation known as driver manipulation. This type of operation occurred recently with Volkswagen, which designed its "fuel-efficient" diesel engines to provide high-performance measurements when being tested but to operate at a lower level during standard driving conditions.

Driver manipulation may be implemented by the original hardware vendor, the original software designer, or a third party, whether a legitimate systems designer or an attacker. Driver manipulation can be based on customized code within the driver itself or on non-driver software that takes advantage of driver features, capabilities, or vulnerabilities in order to achieve the desired goal or effect.

Driver manipulation may be used to achieve a specific goal or hide the fact that a specific goal is not being met. Driver manipulation can be used to optimize performance or diminish performance, improve security or circumvent security, create remote control and backdoor vulnerabilities, or block such abuses from being implemented.

Shimming

Shimming is a means of injecting alternate or compensation code into a system in order to alter its operations without changing the original or existing code. A rough analogy would be that when a table on a new floor is wobbly, a shim can be used to prop up the leg; this is preferable to rebuilding or modifying the table itself. A shim can be used as a quick fix for existing software or firmware code in order to alter operations in situ or to test new options before modifying the core code base.

A shim can be inserted anywhere between two programming objects or subroutines as long as it accepts the output from the preceding element and can produce acceptable input for the receiving element. The shim will intercept the API calls, output, or messages from the first element, perform processing on the captured information set, and then generate output that is compliant with the input of the next element.

Shims are widely used to support legacy applications when the hardware platform no longer provides essential functions. The shim acts as a compatibility interface between the old API and the new one.

Shims can also be employed by attackers to inject alternate commands into an operating environment, add hooks for eavesdropping and manipulation, or simply gain remote access to and control of a target.

Refactoring

Refactoring is a restricting or reorganizing of software code without changing its externally perceived behavior or produced results. Refactoring focuses on improving software's nonfunctional elements, such as quality attributes, nonbehavioral requirements, service requirements, or constraints. Refactoring can improve readability, reduce complexity, ease troubleshooting, and simplify future expansion and extension efforts. Refactoring may be able to simplify internal programmatic logic and eliminate hidden or unresolved bugs or weaknesses.

The goals of refactoring include maintaining the same external behavior and not introducing new bugs or flaws.

Refactoring is about simplifying code, removing redundancies, and avoiding long, monolithic code structures. By dividing computer code into distinct encapsulated elements, modules, objects, or subroutines, programmers ensure that the resulting code is easier to test, verify, and modify. Refactoring is touted by many as a key behavior of experienced programmers.

Refactoring can also be used as a means to focus on programming shortcuts or resolve inelegant solutions. Sometimes, to get code to work, programmers will effectively cheat by using shortcuts rather than crafting the longer valid and complete method. This may be fine initially, but the more elements of the code depend on the cheat, the more unstable and unreliable the whole software becomes. Some call this a technical debt, and like monetary debt, it can accumulate interest and make the resulting software unstable or insecure. Refactoring gives the programmer the opportunity to re-code shortcuts with proper instructions in order to model or craft behaviors more reliably and completely.

The lack of refactoring may leave weaknesses in code or flaws in logic that an attacker might discover and leverage to their advantage. These flaws may be discoverable using fuzzing tools; see the Chapter 3 section "Dynamic analysis (e.g., fuzzing)." Such discoveries are the foundation of unknown and zero-day exploits that anyone using such flawed and inelegant software is likely to be attacked by.

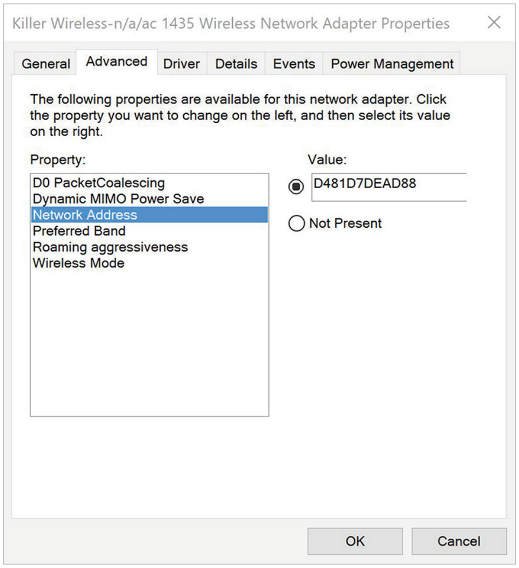
MAC spoofing

MAC (media access control) addresses are also known as physical addresses, hardware addresses, or Ethernet addresses. The MAC address is typically a 48-bit binary number assigned to a NIC by the manufacturer. The MAC address is composed of two equal-sized parts: an OUI and a NIC specific number. The OUI, or organizationally unique identifier, is issued by IEEE (Institute of Electrical and Electronics Engineers) to NIC vendors. Then NIC vendors generate their own NIC-specific number, which may include references to the model and build run as well as a unique value per device. The MAC address is then burned to the ROM chip on the NIC. Because the Ethernet protocol operates in computer memory, however, it reads this ROM-hosted MAC from the NIC but then stores its copy in a software configuration location (such as a CFG file in Linux or the Registry in Windows).

It is possible to eavesdrop on a network and take note of the MAC addresses in use. One of these addresses can then be spoofed into a system by altering the software copy of the NIC's MAC. This causes the Ethernet driver to create frames with the modified or spoofed MAC address instead of the original manufacturer's assigned MAC. Thus, it is quite simple to falsify a MAC address.

MAC spoofing is used to impersonate another system, often a valid or authorized network device, in order to bypass port security or MAC filtering limitations. MAC filtering is a security mechanism intended to limit or restrict network access to those devices with known specific MAC addresses. Its intention is to prevent rogue machines from participating in network communications. However, a simple-to-use Linux application called `macchanger` does just that with a few keystrokes. On the Windows platform, Technitium MAC Address Changer makes MAC spoofing easy. Windows 10 may offer the ability to change your MAC address if supported by the device driver for your NIC; to check, view the Advanced tab of the adapter's device properties dialog box (Figure 1.10). Thus, MAC filtering isn't a complete security solution, as MAC spoofing can bypass this defensive measure.

FIGURE 1.10 Changing a MAC address on a wireless adapter in Windows 10



Killer Wireless-n/a/ac 1435 Wireless Network Adapter Properties

The following properties are available for this network adapter. Click the property you want to change on the left, and then select its value on the right.

Property:

D0 PacketCoalescing
Dynamic MIMO Power Save
Network Address
Preferred Band
Roaming aggressiveness
Wireless Mode

Value:

● D481D7DEAD88

○ Not Present

Countermeasures to MAC spoofing include the following:

- Using intelligent switches that monitor for odd MAC address uses and abuses
- Using a NIDS that monitors for odd MAC address uses and abuses
- Maintaining an inventory of devices and their MAC addresses to confirm whether a device is authorized or unknown and rogue

IP spoofing

Spoofing is the act of falsifying data. Usually the falsification involves changing the source address of network packets. As a result of the changed source address, victims are unable to locate the true attackers or initiators of a communication. Also, by spoofing the source address, the attacker redirects packet responses, replies, and echoes to some other system (as in the case of Smurf, Fraggle, and land DoS attacks).

There are three main types of IP spoofing. One method is to craft IP packets for an attack by setting the source IP address to that of an innocent, uninvolved third party. This type of IP spoofing will result in a simplex or one-way communication for the attacker—any response from the primary victim will be sent to the innocent third party. All logs of the attack event will point to the innocent third-party device as the culprit. A second method is to DoS disconnect the owner/user of an IP address, and then temporarily take on that IP address on the attack system. This provides duplex communication for the attacker in order to retrieve information from the primary target. Once the attack is over, the IP address goes back to normal use by the original system. However, log files blame the attack on the innocent third party who was assigned the IP address, not the rogue system. A third method involves using an IP address from the subnet that is not currently assigned to a valid authorized system. This method also grants the attack duplex communication, but the logs indicate that an unassigned address was the source of the attack, which clearly indicates that a rogue machine must have been using the address to perform the attack.

Countermeasures against IP spoofing attacks include the following:

- Drop all inbound packets received by border systems that have a source destination from inside your private network (this indicates spoofing).
- Drop all outbound packets received by border systems that have a source destination from outside your private network (this also indicates spoofing).
- Drop all packets that have a LAN address in their header if that LAN address isn't officially issued to a valid system.
- Operate a NIDS that monitors for changes in where an IP address is used.

Email Spoofing

Spoofing is also a common activity for unsolicited email, commonly known as spam. Spoofed email means you're unable to reply to the email or determine where it originally came from.

There are innumerable forms of spoofing attacks. Spoofing can be used to redirect packets, bypass traffic filters, steal data, perform social engineering attacks, and even falsify websites.

Countermeasures against email spoofing attacks include using email spam filters as well as the spoofing preventions of IP spoofing.

Wireless attacks

Wireless communication is a quickly expanding field of technologies for networking, connectivity, communication, and data exchange. Literally thousands of protocols, standards, and techniques can be labeled as wireless. These include cell phones, Bluetooth, cordless phones, and wireless networking. As wireless technologies continue to proliferate, your organization's security must go beyond locking down its local network. Security should be an end-to-end solution that addresses all forms, methods, and techniques of communication.

Wireless networking has become common on both corporate and home networks. Properly managing wireless networking for reliable access as well as security isn't always a straightforward proposition. This section examines various wireless security issues.

War Driving

War driving is the act of using a detection tool to look for wireless networking signals. Often, war driving refers to someone looking for wireless networks they aren't authorized to access. In a way, war driving is performing a site survey for possibly malicious or at least unauthorized purposes. The name comes from the legacy attack concept of *war dialing*, which was used to discover active computer modems by dialing all the numbers in a prefix or an area code.

War driving can be performed with a dedicated handheld detector, with a PED (personal electronic device) with WiFi capabilities, or with a notebook that has a wireless network card. It can be performed using native features of the OS, or using specialized scanning and detecting tools.

Once a wireless network is detected, the next step is to determine whether the network is open or closed. An open network has no technical limitations to what devices can connect to it, whereas a closed network has technical limitations to prevent unauthorized connections. If the network is closed, an attacker may try to guess or crack the technologies preventing the connection. Often, the setting making a wireless network closed (or at least hidden) is the disabling of service set identifier (SSID) broadcasting. This restriction is easily overcome with a wireless SSID scanner. After this, the hacker determines whether encryption is being used, what type it is, and whether it can be compromised.

War Chalking

War chalking is a type of geek graffiti that some wireless hackers used during the early years of wireless (1997–2002). It's a way to physically mark an area with information about the presence of a wireless network. A closed circle indicated a closed or secured wireless network, and two back-to-back half circles indicated an open network. War chalking was often used to disclose to others the presence of a wireless network in order to share a discovered Internet link. However, now that Internet connectivity is nearly ubiquitous, with most of us carrying an Internet-connected device on our person (usually a smartphone), the popularity of portable WiFi hotspots, and many retail establishments offering free WiFi as an incentive for customers, the need for and occurrence of war chalking has faded. When an attacker uses war dialing to locate a wireless target to compromise, they don't mark up the area with special symbols to inform others of their intentions.

Replay

A *replay attack* is the retransmission of captured communications in hope of gaining access to the targeted system. This concept was discussed earlier in this chapter in the sections "Application/Service Attacks" and "Replay."

Replay attacks in relation to wireless environments specifically may continue to focus on initial authentication abuse. However, many other wireless replay attack variants exist. They include capturing the new connection request of a typical client, and then replaying that request in order to fool the base station into responding as if another new client connection request had been initiated. Wireless replay attacks can also focus on DoS by retransmitting connection requests or resource requests to the base station in order to keep it busy focusing on managing new connections rather than maintaining and providing service for existing connections.

Wireless replay attacks can be mitigated by keeping the firmware of the base station updated as well as operating a wireless focused NIDS. A W-IDS or W-NIDS will be able to detect such abuses and inform the administrators promptly about the situation.

IV

IV stands for *initialization vector*, a mathematical and cryptographic term for a random number. Most modern crypto functions use IVs in order to increase their security by reducing predictability and repeatability. An IV becomes a point of weakness when it's too short, exchanged in plain text, or selected improperly. Thus, an IV attack is an exploitation of how the IV is handled (or mishandled). One example of an IV attack is that of cracking Wireless Equivalent Privacy (WEP) encryption.

WEP is the original encryption option of 802.11 wireless networking. It's based on RC4. However, because of mistakes in its design and implementation, WEP's primary flaw is related to its IV. The WEP IV is only 24 bits long and is transmitted in plain text. This, coupled with the fact that WEP doesn't check for packet freshness, allows a live WEP crack to be successful in less than 60 seconds (see the Wesside-ng tool from the Aircrack-ng suite at `www.aircrack-ng.org`).

Evil twin

Evil twin is an attack in which a hacker operates a false access point that will automatically clone, or twin, the identity of an access point based on a client device's request to connect. Each time a device successfully connects to a wireless network, it retains a wireless profile in its history. These wireless profiles are used to automatically reconnect to a network whenever the device is in range of the related base station. Each time the wireless adapter is enabled on a device, it wants to connect to a network, so it sends out reconnection requests to each of the networks in its wireless profile history. These reconnect requests include the original base station's MAC address and the network's SSID. The evil twin attack system eavesdrops on the wireless signal for these reconnect requests. Once the evil twin sees a reconnect request, it spoofs its identity with those parameters and offers a plain-text connection to the client. The client accepts the request and establishes a connection with the false evil twin base station. This enables the hacker to eavesdrop on communications through a man-in-the-middle attack, which could lead to session hijacking, data manipulation credential theft, and identity theft.

This attack works because authentication and encryption are managed by the base station, not enforced by the client. Thus, even though the client's wireless profile will include authentication credentials and encryption information, the client will accept whatever type of connection is offered by the base station, including plain text.

To defend against evil twin attacks, pay attention to the wireless network your devices connect to. If you connect to a network that you know is not located nearby, it is a likely sign that you are under attack. Disconnect and go elsewhere for Internet access. You should also prune unnecessary and old wireless profiles from your history list to give attackers fewer options to target.

Rogue AP

A security concern commonly discovered during a site survey is the presence of *rogue wireless access points*. A rogue WAP may be planted by an employee for convenience or it may be operated externally by an attacker.

A wireless access point planted by an employee can be connected to any open network port. Such unauthorized access points usually aren't configured for security or, if they are, aren't configured properly or in line with the organization's approved access points. Rogue wireless access points should be discovered and removed in order to eliminate an unregulated access path into your otherwise secured network.

It's common for an attacker to find a way to visit a company (via a friend who is an employee or by going on a company tour, posing as a repair technician or breakfast taco seller, or even breaking in at night) in order to plant a rogue access point. After a rogue access point is positioned, an attacker can gain entry to the network easily from a modest distance away from your front door.

A rogue WAP can also be deployed by an attacker externally to target your existing wireless clients or future visiting wireless clients. An attack against existing wireless clients requires that the rogue WAP be configured to duplicate the SSID, MAC address, and wireless channel of the valid WAP, although operating at a higher power rating. This may cause clients with saved wireless profiles to inadvertently select or prefer to connect to the rogue WAP instead of the valid original WAP.

The second method focuses on attracting new visiting wireless clients. This type of rogue WAP is configured with a social engineering trick by setting the SSID to an alternate name that appears legitimate or even preferred over the original valid wireless network's SSID. For example, if the original SSID is "ABCcafe," then the rogue WAP SSID could be "ABCcafe-2," "ABCcafe-LTE," or "ABCcafe-VIP." The rogue WAP's MAC address and channel do not need to be clones of the original WAP. These alternate names may seem like better network options to new visitors and thus trick them into electing to connect to the false network instead of the legitimate one.

The defense against rogue WAPs is to be aware of the correct and valid SSID. It would also be beneficial for an organization to operate a wireless IDS to monitor the wireless signals for abuses, such as newly appearing WAPs, especially those operating with mimicked or similar SSID and MAC values.

Jamming

Wireless communications employ radio waves to transmit signals over a distance. There is a finite amount of radio wave spectrum; thus, its use must be managed properly to allow multiple simultaneous connections with little to no interference. The radio spectrum is measured or differentiated using *frequency*. Frequency is a measurement of the number of wave oscillations within a specific time, identified using the unit Hertz (Hz), or oscillations per second. Radio waves have a frequency between 3 Hz and 300 GHz. Different ranges of frequencies have been designated for specific uses, such as AM and FM radio, VHF and UHF television, and so on. Currently, the 900 MHz, 2.4 GHz, and 5 GHz frequencies are the most commonly used in commercial wireless products because of their unlicensed categorization. However, to manage the simultaneous use of the limited radio frequencies, several spectrum-use techniques were developed. These include *spread spectrum, frequency hopping spread spectrum (FHSS)*, *direct sequence spread spectrum (DSSS)*, and *orthogonal frequency-division multiplexing (OFDM).*

NOTE

Most devices operate within a small subsection of frequencies rather than all available frequencies. This is because of frequency-use regulations (determined by the Federal Communications Commission (FCC) in the United States), power consumption, and the expectation of interference.

Spread spectrum means that communication occurs over multiple frequencies at the same time. Thus, a message is broken into pieces, and each piece is sent at the same time but using a different frequency. Effectively, this is a parallel communication rather than a serial communication.

Frequency hopping spread spectrum (FHSS) was an early implementation of the spread spectrum concept. However, instead of sending data in a parallel fashion, it transmits data in a series while constantly changing the frequency in use. The entire range of available frequencies is employed, but only one frequency at a time is used. As the sender changes from one frequency to the next, the receiver has to follow the same hopping pattern to pick up the signal. FHSS was designed to help minimize interference by constantly shifting frequencies rather than using only a single frequency that could be affected.

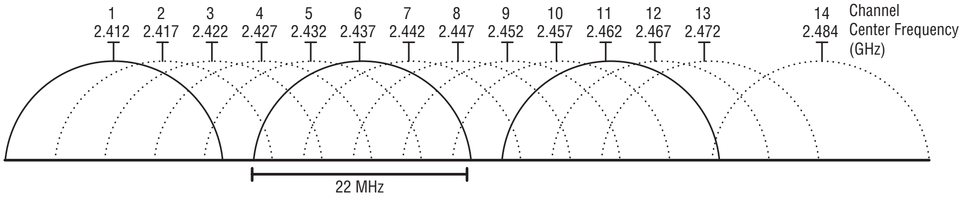
Direct sequence spread spectrum (DSSS) employs several available frequencies simultaneously in parallel. This provides a higher rate of data throughput than FHSS. DSSS also uses a special encoding mechanism known as *chipping code* to allow a receiver to reconstruct data even if parts of the signal were distorted due to interference. This occurs in much the same way that the parity of RAID 5 allows the data on a missing drive to be re-created.

Orthogonal frequency-division multiplexing (OFDM) is yet another variation on frequency use. OFDM employs a digital multicarrier modulation scheme that allows for a more tightly compacted transmission. The modulated signals are perpendicular (orthogonal) and thus don't interfere with each other. Ultimately, OFDM requires a smaller frequency set (aka *channel bands*) but can offer greater data throughput.

Wireless Channels

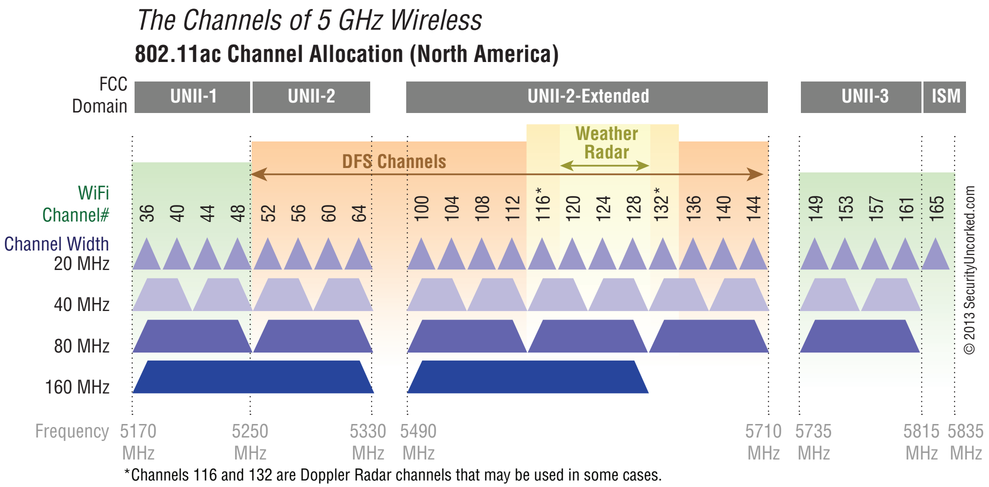
There are many more topics within the scope of wireless networking that we aren't addressing due to space limitations and because they're not covered on the exam. For instance, you may want to learn more about wireless channels. Within the assigned frequency of the wireless signal are subdivisions of that frequency known as *channels*. Think of channels as lanes on the same highway. In relation to the 2.4 GHz frequency, in the United States there are 11 channels, in Europe there are 13, and in Japan there are 14. The differences stem from local laws regarding frequency management (think international versions of the United States' FCC).

The Channels of 2.4 GHz Wireless



Adapted from `https://en.wikipedia.org/wiki/List_of_WLAN_channels`

Wireless communications take place between a client and an access point over a single channel. However, when two or more access points are relatively close to each other physically, signals on one channel can interfere with signals on another channel. One way to avoid this is to set the channels of physically close access points as far apart as possible to minimize channel overlap interference. This is most important for 2.4 GHz networks, where channels are only 5 MHz apart but are 22 MHz wide. This causes channels that are within three numbers of another to experience some level of interference. For example, if a building has four access points arranged in a line along the length of the building, the channel settings could be 1, 11, 1, and 11. But if the building is square and an access point is in each corner, the channel settings may need to be 1, 4, 8, and 11. Think of the signal within a single channel as being like a wide-load truck in a lane on the highway. The wide-load truck is using part of each lane on either side of it, thus making passing in those lanes dangerous. Likewise, wireless signals in adjacent channels will interfere with each other. Channel interference is not an issue with the 5 GHz frequency range (shown next)—the channels are 20 MHz apart and 20 MHz wide, and therefore even adjacent channels do not overlap or interfere.



Adapted from `https://www.networkcomputing.com/wireless/dynamic-frequency-selection-part-3-channel-dilemma/438580919`

Interference may occur by accident or intentionally. Intentional interference is a form of jamming. *Jamming* is the transmission of radio signals to prevent reliable communications by decreasing the effective signal-to-noise ratio. To avoid or minimize interference and jamming, start by adjusting the physical location of devices. Next, check for devices using the same frequency and/or channel. If there are conflicts, change the frequency or channel in use on devices you control. If an interference attack is occurring, try to triangulate the source of the attack and take appropriate steps to address the concern—that is, contact law enforcement if the source of the problem is outside of your physical location.

WPS

WiFi Protected Setup (WPS) is a security standard for wireless networks. It is intended to simplify the effort involved in adding new clients to a well-secured wireless network. It operates by autoconnecting the first new wireless client to seek the network once the administrator has triggered the feature by pressing the WPS button on the base station. However, the standard also calls for a code or PIN that can be sent to the base station remotely in order to trigger WPS negotiation without the need to physically press the button. This can lead to a brute-force guessing attack that could enable a hacker to guess the WPS code in hours (usually less than 6 hours), which in turn enables the hacker to connect their own unauthorized system to the wireless network.

NOTE The PIN code is composed of two four-digit segments, which can be guessed one segment at a time with confirmation from the base station.

WPS is a feature that is enabled by default on most wireless access points because it is a requirement for device WiFi Alliance certification. It's important to disable it as part of a security-focused predeployment process. If a device doesn't offer the ability to turn off WPS (or the Off switch doesn't work), upgrade or replace the base station's firmware or replace the whole device.

Generally, leave WPS turned off. Each time you upgrade your firmware, perform your security-focused predeployment process again to ensure all settings, including WPS, are set properly. If you need to add numerous clients to a network, you can temporarily re-enable WPS—just be sure to disable it immediately afterward.

Bluejacking

Bluejacking involves sending messages to Bluetooth-capable devices without the permission of the owner/user. These messages often appear on a device's screen automatically. Just about any Bluetooth-enabled device, such as a PDA, a cell phone, and even a notebook computer, can receive a bluejacked message. Most bluejacking involves sending a vCard (a virtual business card) to a target device over the Object Exchange (OBEX) protocol (which is also used by infrared communications). Many small portable devices have only a 1 mW power antenna, and Bluetooth may be accessible from 10 meters or less, whereas on a notebook, Bluetooth may be accessible from up to 100 meters (thanks to the 100 mW power antenna). However, even these distances can be exceeded by using a strong transmission antenna, which allows distances of a mile or more.

A bluejack message is often positioned in the name field of the vCard, with little or nothing else. This limits the messages to short strings of text. But this stunt can still be used to pull off various pranks, teasing, and advertisements. Some multimedia message–capable phones are also able to receive images and sound. Bluejacking is mostly harmless, because it doesn't contain malicious code—at least, not so far.

Many devices are configured with a level of defense against bluejacking by not automatically accepting Bluetooth-transmitted messages from unknown (that is, unpaired) sources. Instead, you may see a warning stating that a message from an unknown device has been received and asking whether you want to accept or reject the message. You can also minimize your exposure by keeping Bluetooth off when not in active use.

All Bluetooth devices are vulnerable to bluejacking, since it is just a transmission of a message or announcement. Other Bluetooth-based attacks that are of widespread concern are *bluesniffing* and *bluesmacking*. Bluesniffing is eavesdropping or packet-capturing Bluetooth communications. Since Bluetooth is mostly plain text, this attack can allow an attacker to monitor your Bluetooth activities, such as keystrokes, phone calls, and so on. Bluesmacking is a DoS attack against a Bluetooth device. The defenses for these risks are to minimize use of Bluetooth, especially in public locations.

Bluesnarfing

Bluesnarfing is the unauthorized access of data via a Bluetooth connection. Often the term bluejacking is mistakenly used to describe or label the activity of bluesnarfing. Successful bluesnarfing attacks against PDAs, cell phones, and notebooks have been able to extract calendars, contact lists, text messages, emails, pictures, videos, and more. Because bluesnarfing involves stealing data, it's illegal in most countries.

Bluesnarfing typically occurs over a paired link between the hacker's system and the target device. If the device isn't enabled to be seen by the public (that is, discoverable) or to allow pairing, bluesnarfing usually isn't possible. There was a Bluetooth flaw that could be exploited to perform bluesnarfing against phones that were set up as private, but this has long since been patched. It's true that bluesnarfing is also possible against nondiscoverable devices if you know their Bluetooth MAC addresses, but this usually isn't a practical attack because the 48-bit address must be guessed.

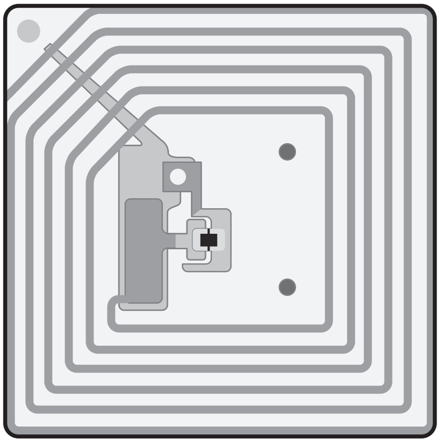
Another interesting Bluetooth attack is *bluebugging*. This attack grants an attacker remote control over the hardware and software of your devices over a Bluetooth connection. The name is derived from enabling the microphone on a compromised system in order to use it as a remote wireless bug.

Bluesnarfing and bluebugging are attacks based not on an inherent flaw of Bluetooth but on vulnerabilities in specific device implementations. Thus, these exploits are not widespread, but they are serious if your device is vulnerable. Be sure to keep your firmware current in order to minimize the risk.

RFID

RFID (Radio Frequency Identification) is a tracking technology based on the ability to power a radio transmitter using current generated in an antenna (Figure 1.11) when placed in a magnetic field. RFID can be triggered/powered and read from a considerable distance away (often hundreds of meters). RFID can be attached to or integrated into the structure of devices such as notebook computers, tablets, routers, switches, USB flash drives, portable hard drives, and so on. This can allow for quick inventory tracking without having to be in direct physical proximity of the device. Simply walking into a room with an RFID reader can collect the information transmitted by the activated chips in the area.

FIGURE 1.11 An RFID antenna



Adapted from `https://electrosome.com/rfid-radio-frequency-identification/`

There is some concern that RFID can be a privacy-violating technology. If you are in possession of a device with an RFID chip, then anyone with an RFID reader can take note of the signal from your chip. Mostly an RFID chip transmits a unique code or serial number—which is meaningless without the corresponding database that links the number to the specific object (or person). However, if you are the only one around and someone detects your RFID chip code, then they can associate you and/or your device with that code for all future detections of the same code.

NFC

Near field communication (NFC) is a standard that establishes radio communications between devices in close proximity. It lets you perform a type of automatic synchronization and association between devices by touching them together or bringing them within inches of each other. NFC is a derivative technology from RFID and is itself a form of field-powered or -triggered device.

NFC is commonly found on smartphones and many mobile device accessories. It's often used to perform device-to-device data exchanges, set up direct communications, or access more complex services such as WPA-2 encrypted wireless networks by linking with the wireless access point via NFC. Because NFC is a radio-based technology, it isn't without its vulnerabilities. NFC attacks can include man-in-the-middle, eavesdropping, data manipulation, and replay attacks.

Disassociation

Disassociation is one of the many types of wireless management frames. A disassociation can be used in several forms of wireless attacks, including the following:

- For networks with hidden SSIDs, a disassociation packet with a MAC address spoofed as that of the WAP is sent to a connected client that causes the client to lose its connection and then send a Reassociation Request packet, which includes the SSID in the clear.
- An attack can send repeated disassociation frames to a client in order to prevent reassociation, thus causing a DoS.
- A session hijack event can be initiated by using disassociation frames to keep the client disconnected while the attacker impersonates the client and takes over their wireless session with the WAP.
- A man-in-the-middle attack can be implemented by using a disassociation frame to disconnect a client. Then the attacker provides a stronger signal from their rogue/fake WAP using the same SSID and MAC as the original WAP; once the client connects to the false WAP, the attacker connects to the valid WAP.

The main defense against these attacks is to operate a wireless IDS, which monitors for wireless abuses.

Cryptographic attacks

Passwords are the most common form of authentication; at the same time, they're the weakest form. Reliance solely on passwords isn't true security. The strength of a password is generally measured in the amount of time and effort involved in breaking the password through various forms of cryptographic attacks. These attacks are collectively known as *password cracking* or *password guessing*. A weak password invariably uses only alphanumeric characters; often employs dictionary or other common words; and may include user profile–related information such as birthdates, Social Security numbers, and pet names. A strong password is longer, more complex, and unique, and is changed on a regular basis.

At least four attack methods are used to steal or crack passwords. All of them involve *reverse hash matching*. This is the process of stealing the hash of a password directly from an authentication server's account database or plucking it out of network traffic, and then reverse-engineering the original password. This is done by taking potential passwords, hashing them, and then comparing the stolen hash with the potential password hash. If a match is found, then the potential password is probably the actual password. (By the way, even if the potential password isn't the actual password, if it happens to produce the same hash, it will be accepted by the authentication system as the valid password.) The four password-cracking or -guessing attacks are *brute force* (aka *birthday attack*), *dictionary, hybrid*, and *rainbow tables*.

This section delves into these four main password-cracking techniques, as well as related issues and a few other attacks that focus on abusing cryptographically protected data.

Birthday

A *brute force or birthday attack* is used against hashing and other forms of cryptography involving finite sets (of either hashes or keys). The birthday attack gets its name from a bar bet that exploits the mathematical probability of shared birthdays. (The bar bet is that you'll drink for free if two people in the bar share a birthday; otherwise, you'll buy the house a round of drinks.) However, the bar bet is derived from the birthday statistical paradox, which is found in the area of mathematics known as *probability theory*.

The issue is that because there are only 366 possible birthdays (don't forget leap year!), the chance of two people sharing the same birth month and day increases exponentially as group size increases. It takes only 23 people for there to be a 50 percent chance that two share the same birthday, and only 75 people are needed for a 99.9 percent chance. When this logic is applied to cracking passwords (or encryption keys), it shows that because the target is part of a finite set (large, yes, but still finite), the likelihood of guessing correctly increases with each subsequent guess. In other words, each wrong guess removes one option from the remaining pool, so the next guess has a slightly greater chance of being correct. This is why brute force attacks are successful—given enough time to perform guesses, the probability of success continues to increase.

Birthday attacks can be waged against any use of hashing. However, they're most commonly employed during password-guessing attacks (discussed in the following section). In a password-guessing attack, a program compares possible passwords with passwords stored in an accounts database. But passwords stored in an accounts database are secured because only their hash values are stored there. Thus, the password-cracking program first performs the same hashing function used by the secured system on each possible password before scanning the accounts database for a match. If a match is found, then the password-guessing tool has discovered a password based on the `f(M)=f(M')` property. This is more specifically known as reverse hash matching. Generally, any form of password cracking is based on the birthday attack.

Known plain text/cipher text

The cryptographic attacks of *known plain text* and *known cipher text* are focused on encryption systems that use the same key repeatedly or that select keys in a sequential or otherwise predictable manner. The goal is to discover the key or a key of the series, and then use that key to determine other keys and thus be able to decrypt most or all of the data protected by the flawed encryption system.

The operation of symmetric encryption involves four main components: the original plain text, the algorithm, the key, and the resultant cipher text. When an attacker knows three of these four parts, they can solve for the final part.

The known plain-text attack starts off by knowing the original data which is to be encrypted. Then the target or victim encrypts the data with the known algorithm with an unknown key. The attacker obtains the cipher text result. From these three parts—plain text, algorithm, and cipher text—the attacker can solve for the key. A slight variant of this is *chosen plain text*, where the attacker provides the victim with the original data, which is then encrypted. Once the key is known, future or past keys can be derived.

The known cipher-text version of this attack is the same process, only in reverse. The attacker knows the cipher text and the algorithm. The victim decrypts the cipher text using the unknown key to produce the plain text, which the attacker obtains. Then the attacker solves for the key. Again, there is a variant of this, known as the *chosen cipher text*, where the attacker provides a data set to the victim to serve as the cipher text, and the attacker retrieves the resulting plain text in order to finally solve for the key.

NOTE In most cases, the cipher text is random and the resulting plain text is unintelligible, but the result is mathematically accurate.

Rainbow tables

Traditionally, password crackers hashed each potential password and then performed an *Exclusive Or (XOR)* comparison to check it against the stolen hash. The hashing process is much slower than the XOR process, so 99.99 percent of the time spent cracking passwords was actually spent generating hashes. A new form of password cracking was developed to remove the hashing time from the cracking time. This technique is known as *rainbow tables*.

Rainbow tables take advantage of a concept known as a *hash chain*. A hash chain is constructed using an initial starting password (often selected at random), and then hashing the starting password into its hash value. Then the hash value is converted into a new password using a process called the *reduction function*. (Note that hashes are not reversible; they are a one-way operation, so the reduction function does not re-create the original password, but a new one.) This process (the password to hash to new password) is repeated numerous times. A hash chain can be composed of just a few links (password to hash to new password sections) or a few thousand. Once crafted, only the starting and finishing passwords for each chain are retained. Many unique hash chains are produced in order to include or cover most or all of the potential passwords and hashes in the range of valid values for the hashing algorithm (password system) being attacked.

Once the rainbow table hash chain database is constructed, it can be used to compromise hashes obtained from a victim. The attacker will first run the stolen hash through the reduction function and then check to see if this value matches any of the hash chain end or stop elements. If so, then the attacker knows the plain text of the password is in that specific chain. If not, the attacker performs another set of hash and reduction functions and checks again. Eventually a matching end of hash chain will be discovered.

Once the correct hash chain is determined, the attacker starts the chain calculation, again starting with the original starting value for the chain, and performs the hash and reduction operations until they encounter the stolen hash. Once that is achieved, the attacker knows the immediately previous password used to produce the hash that matches the stolen hash, and thus the password hash has been cracked.

Although this process may seem complex, or even convoluted, it ends up being a fairly efficient means of compromising passwords. However, rainbow tables do have their limitations. It is difficult to know whether a particular set of hash chains is sufficient to cover or address all or even most of the potential passwords for a given hash. The size of the rainbow table depends on the range of possible passwords, the character options, and the lengths of the passwords. For poor password hashing algorithms—such as LM (LAN Manager), which is the oldest and now depreciated function in Microsoft Windows—a complete rainbow table is only 64 GB in size. A rainbow table covering NTLM passwords—limiting the focus to just U.S. keyboard characters (95 unique options) and passwords with a length of 1–8—would be at least 16 EB (or 16,000,000,000,000,000,000 bytes).

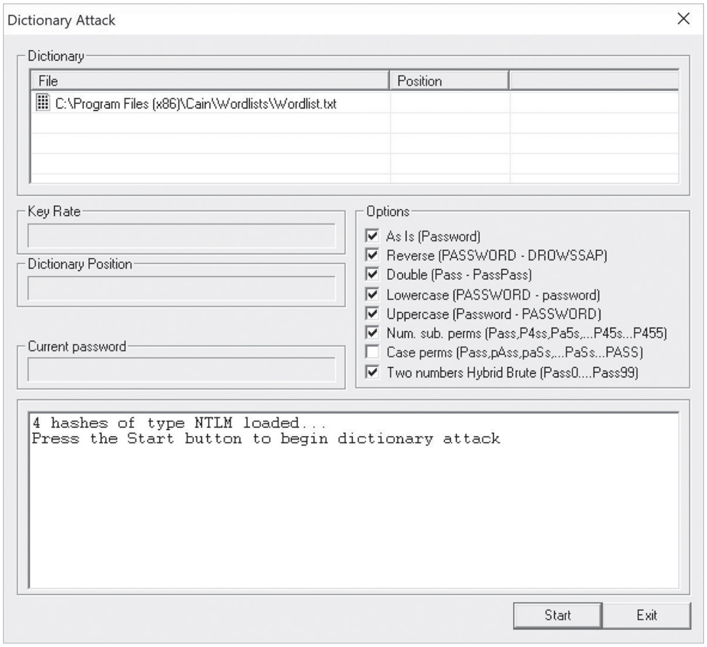
Sometimes rainbow tables are confused with a precomputed hash database. A database containing all possible input passwords and their corresponding output hash would be considerably larger than that of a rainbow table.

To protect yourself from this threat, change all of your passwords to a minimum of 16 characters with a mixture of character types—uppercase, lowercase, numbers, and symbols (when supported). Also be sure to use unique passwords for each logon whether an online site or service or a internal local network system. Whenever available, use multifactor authentication.

Dictionary

A *dictionary attack* (Figure 1.12) performs password guessing by using a preexisting list of possible passwords. Password lists can include millions of possible passwords. Often, password lists or dictionaries are constructed around topics. Thus, if an attacker knows basic information about you as a person, they can attempt to exploit human nature's propensity to select passwords using words common or familiar to you. For example, if an attacker knows that you work in the medical industry, you have cats, and you enjoy sailing, they can select password dictionaries that include words, acronyms, and phrases common to those subjects.

FIGURE 1.12 A dictionary attack configuration page from Cain & Abel



C:\Program Files (x86)\Cain\Wordlists\Wordlist.txt

- ☑ As Is (Password)
- ☑ Reverse (PASSWORD - DROWSSAP)
- ☑ Double (Pass - PassPass)
- ☑ Lowercase (PASSWORD - password)
- ☑ Uppercase (Password - PASSWORD)
- ☑ Num. sub. perms (Pass,P4ss,Pa5s,...P45s...P455)
- □ Case perms (Pass,pAss,paSs,...PaSs...PASS)
- ☑ Two numbers Hybrid Brute (Pass0....Pass99)

```
4 hashes of type NTLM loaded...
Press the Start button to begin dictionary attack
```

Dictionary attacks are surprisingly effective against users who haven't been trained in the methods and skills of creating complex passwords. These attacks are fairly simple in that they try only the passwords from the list in the exact form they have in the list. For example, "password" and "Password" and "PASSWORD" are all different, since uppercase and lowercase letters are different ASCII values. Thus, unless all case variants are included in a dictionary list, they would not be tested for by a dictionary attack. Only the specific constructions of passwords included in the dictionary list are used to attack the target password hashes.

Some dictionary lists attempt to include all passwords stolen to date and all valid words (from an actual dictionary, public domain books, or online encyclopedias) in order to cover a broad range of potential passwords and victims' ideas for selecting passwords. One such list is available from `crackstation.net`, where a list of nearly 1.5 billion passwords is available for download. The use of these aggregated password lists is critical for penetration testing and to create a hardened environment. Ethical penetration testers and system administrators should use password-only dictionary lists for security testing, while avoiding lists that include usernames and other personally identifiable information (PII) elements. Dictionary attacks are relatively fast operations, but they have a low rate of success against targets with any knowledge of password security or whose systems enforce reasonable levels of password length and complexity.

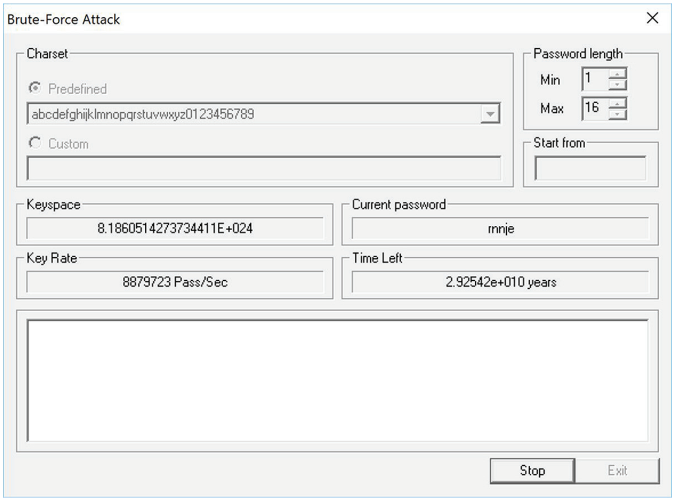
Brute force

A brute force attack (Figure 1.13) is designed to try every possible valid combination of characters to construct possible passwords, starting with single characters and adding characters as it churns through the process, in an attempt to discover the specific passwords used by user accounts. Such attacks are always successful, given enough time. Whereas simple and short passwords can be discovered amazingly quickly with a brute force approach, longer and complex passwords can take an outrageously long period of time (possibly into millions of years of computational time for complex passwords containing 16 or more characters).

Longer and more complex passwords make brute-force attacks less successful. However, given enough time, a brute-force attack will always succeed. But with a sufficiently long target password (16 or more characters), brute force attacks are rendered impractical. An important variant of the brute force attack is that of the hybrid password cracking attack. A *hybrid attack* uses a dictionary list as its password source but uses brute force techniques to make modifications on a progressively increasing level. For example, the first round takes each source password and makes all possible one-character modifications, and then the second round makes all possible two-character modifications, and so on. This includes replacing characters as well as adding characters. The hybrid method has the benefit of focusing on words the target users may have used based on their interests and backgrounds instead of having to try all possible combinations.

Hybrid attacks are often successful even against security professionals who think they're being clever by, for example, changing a to @ and o to 0 and adding the number 12 to the end of the name of their favorite movie character.

FIGURE 1.13 A brute force attack configuration page from Cain & Abel



Predefined: abcdefghijklmnopqrstuvwxyz0123456789

Custom:

Min: 1

Max: 16

Keyspace: 8.1860514273734411E+024

Current password: mnje

Key Rate: 8879723 Pass/Sec

Time Left: 2.92542e+010 years

Hybrid password attacks are most successful against users who are forced into complying with a company password policy, such as a nine-character minimum length with at least two examples of each of the four character types. Most users are not aware of why password complexity is important, nor does the company training program provide sufficient information on selecting passwords. So, the typical person will seek to adhere to the requirements by selecting a word they can easily remember, and then make modifications until it meets the requirements. Often, workers only work hard enough to be minimally compliant. For example, if a user selects the word "password" as their base word, then with just six alterations to change or add characters, they could produce "P@5sw0Rd!", which would be in compliance with a nine-character minimum with two examples of each character type password policy. However, this password is extremely poor, because it is based on one of the most likely words to be contained in a dictionary list and is only a six-change/character variant. A typical hybrid attack would find this user's password in less than 1 second once the base word was reached in the dictionary list.

We must make better password selections in order to defend against hybrid attacks. One method is to select three to five words that are strung together (technically now it would be called a passphrase), and then make sufficient character changes to meet the complexity requirements. This method is far superior to using just a single base word, mostly because the result is usually 10–25 characters long.

Password example	**Estimated time to crack**
montyp99	minutes
monty python grail	years
monty python movie holy grail	centuries
Monty Python 1975 and the Holy Grail!	millennia

Online vs. offline

An online password attack occurs against a live logon on prompt. In this type of attack, the attacker submits credentials, which are then processed by the authentication service of the target system. If the credentials are correct, then the attacker has successfully impersonated the user. If incorrect, a logon denied error occurs. Most logon prompts offer the user several attempts to provide the correct credentials. However, system managers do not want to grant infinite attempts because that enables hackers to continue attempting to guess the victim's credentials. Thus in most cases, *account lockout* is configured.

Account lockout is the security mechanism that provides a fixed number of logon attempts before the account is locked out (disabled for use). It is common to allow for three logon events before triggering account lockout. Once the lockout is triggered, it can last for a specific number of minutes, such as 15, or indefinitely. If account lockout is set to an indefinite time limit, an administrator will need to manually disable the lockout status in order to return the account to a usable state. Keep in mind that these are just the basics of lockout. There are some forms of lockout that lock the account completely, whereas others lock only the current source attempt location (thus another location or device can be used to attempt logon). Some lockouts will disable the primary means of logon (such as a fingerprint) and revert to a fallback method (such as a password). Some lockout systems also offer a user lockout clearing process (similar to that of password recovery) that may involve SMS or emailed recovery codes or answering identity verification security questions. Some lockout systems use an increasing delay between logon attempts so that the third failure causes a 5-minute delay, the fourth a 15-minute delay, the fifth a 30-minute delay, and so on.

An offline attack is one in which the attacker is not working against a live target system but instead is working on their own independent computers. An attacker will have had to obtain the target's password hashes and then transferred them to their own computers. Collecting hashes is a challenging task, since most systems are designed to specifically prevent theft of hashes. They are stored in the encrypted authentication database and only sent over network communications via encrypted channels. Thus, an attacker must use clever techniques to access the hashes. These can include direct physical contact with the target system, using a remote control tool to extract the hashes from memory or from network traffic, or obtaining access to a backup of the system files.

Once the attacker has the password hashes, password-cracking operations can take place on the attacker's own computers, either on their CPU or GPU, or using a cloud computing service (such as Amazon's EC2 service). An offline attack is not affected or limited by account lockout, since that security feature is part of the authentication service and not the hash itself. So the attacker has no limit to the number of password-cracking events to attempt other than their own system's computation speed and their patience to allow the attack to operate.

Collision

A *collision* occurs when the output of two cryptographic operations produce the same result. Collisions occur in relation to encryption operations as well as hashing operations (see Chapter 6, "Cryptography and PKI," for the full coverage of cryptography concepts).

When two encryption operations produce the same cipher text, a collision has occurred. When collisions occur in relation to symmetric encryption, it is a symptom of a serious problem. It can mean that the encryption system being used is not properly implementing the algorithm or that its use of randomization is flawed. A secure encryption system will guarantee that, even if you attempt to encrypt the same message a second time using the same algorithm and the same key, the randomization function (known as the initialization vector [IV]) will ensure unique cipher text each time. An encryption collision can also indicate that the encryption key is not being used in an ephemeral manner (that is, once, randomly, and in a nonrepeating way).

Hashing collisions are more likely the focus of discussion or concern. A hash collision occurs when two different data sets that are hashed by the same hashing algorithm produce the same hash value. This is always a possibility with hashing, since it is a process that takes any input to produce a fixed-length output. Thus, there is a guarantee that multiple inputs will produce the same output. The key is to understand how to use hashing properly in order to avoid allowing occurrences of collisions to be an actual violation of integrity.

Hashes are designed to provide protection from corruption, alteration, or counterfeiting that a person would not notice or would overlook. To this end, hashing algorithms employ features known as *avalanche effects*, which ensure that small changes in the input produce large changes in the output. Thus, if before and after hashes do not match, the current data set is not the same as the original data set. However, if the before and after hashes do match, then there is still one additional step before the current data can be accepted as being the unchanged original. That step is to look at the data, and if it does not look like the original, then the occurrence of a matching hash is a collision. This is therefore a situation where the data has been switched out with a different data set that happens to produce the same hash. In this case, the data should still be discarded because it is obviously not the original. Only when the before and after hashes match and the current data looks like the original data can you accept the current data as valid and unchanged (that is, it has retained its integrity).

Hash collision attacks are intended to fool a victim into accepting an alternate data set just because it happens to produce the same hash value. This can occur only if the victim does not look at their data in addition to checking the hash values. Hash collision is easier with hashes that are shorter, such as 128 bits in length, than longer hashes, such as those 512 bits in length. So when choosing a hashing algorithm, use the available option that produces the longest hash.

Downgrade

A *downgrade attack* attempts to prevent a client from successfully negotiating robust high-grade encryption with a server. This attack may be performed using a real-time traffic manipulation technique or through a man-in-the-middle attack (a false proxy) in order to forcibly downgrade the attempted negotiation to a lower quality level of algorithms and key exchange/generation. By keeping the client from setting up a high-grade encrypted session, the attacker is able to continue to eavesdrop and manipulate the conversation even after the "encrypted" session is established. This type of attack is possible if both the client and server retain older encryption options designed for backward compatibility. If the attacker can force the negotiation to select these older options, then the attacker may be able to exploit known weaknesses in the older solutions. One example of the downgrade attack is against SSL/TLS, where the attacker uses a technique known as the *POODLE attack*. POODLE stands for Padding Oracle On Downgraded Legacy Encryption. POODLE causes the client to fall back to using SSL 3.0, which has less robust encryption cipher suite options than TLS.

The best defense against downgrade attacks is to disable support for older encryption options and backward compatibility with less secure systems.

Replay

A *replay attack* occurs when an attacker captures network packets and then retransmits or replays them back onto the network. Often a replay attack focuses on authentication packets with the goal of being falsely granted access to a system or service. In a replay (or playback) attack, the attacker does not gain knowledge of the victim's credentials. Instead, the attacker holds the packets that contain the credentials and then sends them back out onto the network in hopes of fooling the authentication service into granting the attacker the same access as the original user (the victim).

Replay attacks are mostly relegated to legacy systems and services because most modern implementations of authentication have specific defenses against replay attacks integrated into them. Such defenses include using short time stamps in packets so they are valid only for a few moments, using random one-time-use challenge-response dialogs (that is, a random number is sent to the client system, which must calculate a response), and using one-time-use ephemeral session encryption keys.

Weak Implementations

Most failures of modern cryptography systems are due to poor or *weak implementations* rather than a true failure of the algorithm itself. The algorithms employed by most software products have been widely scrutinized and have survived focused analyses and attacks. Thus, any failures of the software products to provide reliable security are due to programming mistakes or errors in implementation.

It is challenging to properly implement randomization functions in software that produce truly random and non-predictable keys. Some programmers ignore this complexity (and it's important) and rely on the simple and predictable "randomization" function provided by their execution environment. Other failures include reusing the key across multiple data sets or sessions, using keys in sequential order or other patterns of use, storing keys in an insecure fashion, and distributing or exchanging keys in an insecure manner. All of these issues have resolutions, mostly adopting secure coding practices and consulting with other expert programs for tips and suggestions on avoiding common programming pitfalls. It would also be helpful to submit new code to thorough review and analysis by other skilled programmers and cryptography subject matter experts (SMEs).

Exam Essentials

Understand social engineering. Social engineering is a form of attack that exploits human nature and human behavior. Social engineering attacks take two primary forms: convincing someone to perform an unauthorized operation or convincing them to reveal confidential information.

Understand phishing. Phishing is the process of attempting to obtain sensitive information such as usernames, passwords, credit card details, or other personally identifiable information (PII) by masquerading as a trustworthy entity (a bank, a service provider, or a merchant, for example) in electronic communication (usually email).

Understand spear phishing. Spear phishing is a more targeted form of phishing where the message is crafted and directed specifically to an individual or group of individuals. The hope of the attack is that someone who already has an online/digital relationship with an organization is more likely to fall for the false communication.

Understand whaling. Whaling is a form of phishing that targets specific high-value individuals.

Understand vishing. Vishing is phishing done over VoIP services.

Understand tailgating and piggybacking. Tailgating occurs when an unauthorized entity gains access to a facility under the authorization of a valid worker but without their knowledge. Piggybacking occurs when an unauthorized entity gains access to a facility under the authorization of a valid worker but with their knowledge and consent.

Understand impersonation. Impersonation is the act of taking on the identity of someone else. The purpose of impersonation is to trick someone into believing you're the claimed identity so you can use the power or authority of that identity. Impersonation is also known as masquerading or spoofing.

Understand dumpster diving. Dumpster diving is the act of digging through trash in order to obtain information about a target organization or individual. It can provide an attacker with information that could make social engineering attacks easier or more effective.

Understand shoulder surfing. Shoulder surfing occurs when someone is able to watch your keyboard or view your display. This may allow them to learn your password or see information that is confidential, private, or simply not for their eyes.

Understand hoaxes. A hoax is a form of social engineering designed to convince targets to perform an action that will cause problems or reduce their IT security. A hoax is often an email that proclaims some imminent threat is spreading across the Internet and that you must perform certain tasks in order to protect yourself.

Understand watering hole attacks. A watering hole attack is a form of targeted attack against a region, a group, or an organization. It's waged by poisoning a commonly accessed resource.

Understand principles of social engineering. Many techniques are involved in social engineering attacks. These often involve one or more common principles such as authority, intimidation, consensus/social proof, scarcity, familiarity/liking, trust, and urgency.

Understand arbitrary code execution. Arbitrary code execution is the ability to run any software on a target system.

Understand DoS. Denial of service (DoS) is a form of attack that has the primary goal of preventing the victimized system from performing legitimate activity or responding to legitimate traffic. One form exploits a weakness, an error, or a standard feature of software to cause a system to hang, freeze, consume all system resources, and so on. The end result is that the victimized computer is unable to process any legitimate tasks. Another form floods the victim's communication pipeline with garbage network traffic. The end result is that the victimized computer is unable to send or receive legitimate network communications.

Understand a Smurf attack. This form of DRDoS uses ICMP echo reply packets (ping packets).

Understand Xmas attacks. The Xmas attack is actually an Xmas scan. It's a form of port scanning that can be performed by a wide number of common port scanners, including Nmap, Xprobe, and hping2. The Xmas scan sends a TCP packet to a target port with the flags URG, PSH, and FIN all turned on.

Understand DDoS. Distributed denial-of-service (DDoS) employs an amplification or bounce network that is an unwilling or unknowing participant that is unfortunately able to receive broadcast messages and create message responses, echoes, or bounces. In effect, the attacker sends spoofed message packets to the amplification network's broadcast address.

Understand man-in-the-middle attacks. A man-in-the-middle attack is a form of communications eavesdropping attack. Attackers position themselves in the communication stream between a client and server (or any two communicating entities). The client and server believe they're communicating directly with each other.

Understand buffer overflows. Buffer overflows occur due to a lack of secure defensive programming. The exploitation of a buffer overflow can result in a system crash or arbitrary code execution. A buffer overflow occurs when a program receives input that is larger than it was designed to accept or process. The extra data received by the program is shunted over to the CPU without any security restrictions; it's then allowed to execute. Results of buffer overflows can include crashing a program, freezing or crashing the system, opening a port, disabling a service, creating a user account, elevating the privileges of an existing user account, accessing a website, or executing a utility.

Understand injection attacks. An injection attack is any exploitation that allows an attacker to submit code to a target system in order to modify its operations and/or poison and corrupt its data set. Examples include SQL injection, LDAP injection, XML injection, command injection, HTML injection, code injection, and file injection.

Understand SQL injection. SQL injection attacks allow a malicious individual to perform SQL transactions directly against the underlying database through a website front end.

Understand directory traversal. A directory traversal is an attack that enables an attacker to jump out of the web root directory structure and into any other part of the filesystem hosted by the web server's host OS.

Understand cross-site scripting. Cross-site scripting (XSS) is a form of malicious code injection attack in which an attacker is able to compromise a web server and inject their own malicious code into the content sent to other visitors.

Understand cross-site scripting (XSS) prevention. The most effective ways to prevent XSS on a resource host are implemented by the programmer by validating input, coding defensively, escaping metacharacters, and rejecting all script-like input.

Understand cross-site request forgery (XSRF). Cross-site request forgery (XSRF) is an attack focused on the visiting user's web browser more than on the website being visited. The main purpose of XSRF is to trick the user or the user's browser into performing actions they had not intended or would not have authorized.

Understand cross-site request forgery (XSRF) prevention. XSRF prevention measures include adding a randomization string (called a nonce) to each URL request and session establishment and checking the client HTTP request header referrer for spoofing.

Understand privilege escalation. Privilege escalation occurs when a user account is able to obtain unauthorized access to higher levels of privileges, such as a normal user account that can perform administrative functions. Privilege escalation can occur through the use of a hacker tool or when an environment is incorrectly configured.

Understand ARP poisoning. ARP poisoning is the act of falsifying the IP-to-MAC address resolution system employed by TCP/IP.

Understand amplification. An amplification attack is one where the amount of work or traffic generated by an attacker is multiplied in order to cause a significant volume of traffic to be delivered to the primary victim. An amplification attack can also be known as a reflective or bound attack.

Understand DNS poisoning. DNS poisoning is the act of falsifying the DNS information used by a client to reach a desired system. This can be accomplished by deploying a rogue DNS server (also known as DNS spoofing and DNS pharming), using DNS poisoning, altering the HOSTS file, corrupting IP configuration, and using proxy falsification.

Understand pharming. Pharming is the malicious redirection of a valid website's URL or IP address to a fake website that hosts a false version of the original valid site.

Understand domain hijacking. Domain hijacking or domain theft is the malicious action of changing the registration of a domain name without the authorization of the valid owner. This may be accomplished by stealing the owner's logon credentials, using XSRF, hijacking sessions, using MitM, or exploiting a flaw in the domain registrar's systems.

Understand man-in-the-browser. The man-in-the-browser (MitB, MiTB, MiB, MIB) attack is effectively a MitM attack. The only real distinction is that the middle-man malware is operating on the victim's system, where it is able to intercept and manipulate communications immediately after they leave the browser and before they exit the network interface.

Understand zero day. Zero-day attacks are newly discovered attacks for which there is no specific defense. A zero-day exploit aims at exploiting flaws or vulnerabilities in targeted systems that are unknown or undisclosed to the world in general. Zero day also implies that a direct or specific defense to the attack does not yet exist; thus most systems with the targeted vulnerable asset are at risk.

Understand a replay attack. In a replay attack, an attacker captures network traffic and then replays the captured traffic in an attempt to gain unauthorized access to a system.

Understand pass the hash. Pass the hash is an authentication attack that potentially can be used to gain access as an authorized user without actually knowing or possessing the plain text of the victim's credentials. This attack is mostly aimed at Windows systems.

Understand hijacking attacks. Hijacking attacks are those where an attacker takes over control of a session from a valid user. Some forms of hijacking disconnect the client, whereas others grant the attacker a parallel connection into the system or service.

Understand clickjacking. Clickjacking is a web page–based attack that causes a user to click on something other than what the user intended to click. This is often accomplished by using hidden or invisible layovers, frame sets, or image maps.

Understand session hijacking. TCP/IP hijacking, or session hijacking, is a form of attack in which the attacker takes over an existing communication session. The attacker can assume the role of the client or the server, depending on the purpose of the attack.

Understand typo squatting/URL hijacking. Typo squatting, or URL hijacking, is a practice employed to capture traffic when a user mistypes the domain name or IP address of an intended resource.

Understand cookies. A cookie is a tracking mechanism developed for web servers to monitor and respond to a user's serial viewing of multiple web pages. It may allow identity theft.

Understand driver manipulation. Driver manipulation occurs when a malicious programmer crafts a system or device driver so that it behaves differently based on certain conditions.

Understand shimming. Shimming is a means of injecting alternate or compensation code into a system in order to alter its operations without changing the original or existing code.

Understand refactoring. Refactoring is a restricting or reorganizing of software code without changing its externally perceived behavior or produced results. Refactoring focuses on improving software's nonfunctional elements, such as quality attributes, non-behavioral requirements, service requirements, and constraints.

Understand spoofing. Spoofing is the act of falsifying data. Usually the falsification involves changing the source addresses of network packets. Because the source address is changed, victims are unable to locate the true attackers or initiators of a communication. Also, by spoofing the source address, attackers redirect responses, replies, and echoes of packets to some other system.

Understand MAC spoofing. MAC spoofing is used to impersonate another system, often a valid or authorized network device in order to bypass port security or MAC filtering limitations.

Understand IP spoofing. There are three main types of IP spoofing: crafting IP packets for an attack but setting the source IP address to that of an innocent, uninvolved third party; via DoS, disconnecting the owner/user of an IP address, then temporary taking on that IP address on the attack system; or using an IP address from the subnet that is not currently assigned to a valid authorized system.

Understand war driving. War driving is the act of using a detection tool to look for wireless networking signals. Often, war driving is the process of someone looking for a wireless network they aren't authorized to access.

Understand wireless replay attacks. Wireless replay attacks may focus on initial authentication abuse. They may be used to simulate numerous new clients or cause a DoS.

Understand initialization vector (IV). IV is a mathematical and cryptographic term for a random number. Most modern crypto functions use IVs in order to increase their security by reducing predictability and repeatability.

Understand evil twin attacks. During an evil twin attack, a hacker configures their system as a twin of a valid wireless access point. Victims are tricked into connecting to the fake twin instead of the valid original wireless network.

Understand rogue access points. A rogue WAP may be planted by an employee for convenience or it may be operated externally by an attacker. Rogue wireless access points should be discovered and removed in order to eliminate an unregulated access path into your otherwise secured network.

Understand jamming. Jamming is the transmission of radio signals to prevent reliable communications by decreasing the effective signal-to-noise ratio.

Understand WPS attacks. WPS is a security standard for wireless networks that was found to be flawed. The standard called for a code that could be sent to the base station remotely in order to trigger WPS negotiation. This led to a brute force guessing attack that could enable a hacker to guess the WPS code in just hours.

Understand bluejacking. Bluejacking is the sending of messages to Bluetooth-capable devices without the permission of the owner/user. Just about any Bluetooth-enabled device, such as a smartphone or notebook computer, can receive a bluejacked message.

Understand bluesnarfing. Bluesnarfing is the unauthorized accessing of data via a Bluetooth connection. Successful bluesnarfing attacks against smartphones and notebooks have been able to extract calendars, contact lists, text messages, emails, pictures, videos, and more.

Understand RFID. RFID (radio frequency identification) is a tracking technology based on the ability to power a radio transmitter using current generated in an antenna when placed in a magnetic field. RFID can be triggered/powered and read from up to hundreds of meters away.

Understand NFC. Near field communication (NFC) is a standard to establish radio communications between devices in close proximity. It lets you perform a type of automatic synchronization and association between devices by touching them together or bringing them within inches of each other.

Understand disassociation. Disassociation is one of the many types of wireless management frames. A disassociation can be used in several forms of wireless attacks, including discovering hidden SSIDs, causing a DoS, hijacking sessions, and using MitM.

Understand password attacks. The strength of a password is generally measured in the amount of time and effort involved in breaking the password through various forms of cryptographic attacks. These attacks are collectively known as password cracking or password guessing. Forms of password attacks include brute force (also known as a birthday attack), dictionary, hybrid, and rainbow tables.

Understand password guessing. Password guessing is an attack aimed at discovering the passwords employed by user accounts. It's often called password cracking. There are two primary categories of password-guessing tools based on the method used to select possible passwords for a direct logon prompt or birthday attack procedure: brute force and dictionary.

Understand password crackers. A password cracker is a tool used to reverse-engineer the secured storage of passwords in order to gain (or regain) access to an unknown or forgotten password. There are four well-known types of password-cracking techniques: dictionary, brute force, hybrid, and precomputed hash.

Understand birthday attacks. The birthday attack exploits a mathematical property that if the same mathematical function is performed on two values and the result is the same, then the original values are the same. This concept is often represented with the syntax `f(M)=f(M') therefore M=M'`.

Understand known plain text and known cipher text attacks. The cryptographic attacks of known plain text and known cipher text are focused on encryption systems that use the same key repeatedly or that select keys in a sequential or otherwise predictable manner. The goal is to discover the key or a key of the series, and then use that key to determine other keys and thus be able to decrypt most or all of the data protected by the flawed encryption system.

Understand rainbow tables. Rainbow tables take advantage of a concept known as a hash chain. It offers relatively fast password cracking, but at the expense of spending the time and effort beforehand to craft the rainbow table hash chain database.

Understand dictionary attacks. A dictionary attack performs password guessing by using a preexisting list of possible passwords.

Understand brute-force attacks. A brute force attack is designed to try every valid combination of characters to construct possible passwords, starting with single characters and adding characters as it churns through the process, in an attempt to discover the specific passwords used by user accounts.

Understand online vs. offline password cracking. An online password attack occurs against a live logon prompt. An offline attack is one where the attacker is not working against a live target system, but instead is working on their own independent computers to compromise a password hash.

Understand collision. A collision is when the output of two cryptography operations produces the same result. Collisions occur in relation to encryption operations as well as hashing operations.

Understand a downgrade attack. A downgrade attack attempts to prevent a client from successfully negotiating robust high-grade encryption with a server. This attack may be performed using a real-time traffic manipulation technique or through a man-in-the-middle attack (a false proxy) in order to forcibly downgrade the attempted negotiation to a lower quality level of algorithms and key exchange/generation.

Understand replay attacks. A replay attack is one in which an attacker captures network packets and then retransmits or replays them back onto the network.

Understand weak implementations. Most failures of modern cryptography systems are due to poor or weak implementations rather than a true failure of the algorithm itself.

1.3 Explain threat actor types and attributes.

A *threat actor* is the person or entity who is responsible for causing or controlling any security-violating incidents experienced by an organization or individual. Such incidents may or may not successfully breach the security infrastructure of the victim, but any attempt is still an event to be noticed, recorded, and evaluated. It is important to understand the threats faced by your organization, and knowing the types of threat actors who may be responsible will further help your preparedness efforts.

Types of actors

The actual perpetrators of attacks or exploits range from individuals to organizations. Some of the terms related to threat actors are included in the following sections.

Black Hat, White Hat, Gray Hat

There are many names that have been used to refer to those who attack computer systems and networks. These include hacker, cracker, phreaker, black hat, white hat, and gray hat. A hacker is someone skilled and knowledgeable in a system. Hackers may be able to take a system apart, alter its functions, repair broken elements, and reassemble it back into a working system. The term *hacker* simply denotes skill, not intention or authorization. A cracker is an attacker of computer systems and networks. It is the malicious form of hacker. However, due to media use, the term *hacker* has picked up a negative connotation. So, *ethical hacker* is often used to denote the benign nature of the skilled individual, versus *criminal or malicious hacker* for the bad guy. A *phreaker* is someone who attacks the telephone network and related systems. A *black hat* is a criminal or malicious attacker, whereas a *white hat* is an ethical hacker or skilled IT professional. A *gray hat* may be a reformed criminal or a skilled IT professional operating undercover to perform ethical hacking (also known as penetration testing).

Script kiddies

Script kiddies are threat actors who are less knowledgeable than a professional skilled attacker. A script kiddie is usually unable to program their own attack tools and may not understand exactly how the attack operates. However, a script kiddie is able to follow instructions and use attack tools crafted by other skilled and knowledgeable malicious programmers. Script kiddies are much more numerous than professional attackers. Script kiddies also pose a serious threat due to their number as well as the chance that they may have access to an attack tool that can exploit a vulnerability in your IT system.

Hacktivist

A *hacktivist* is someone who uses their hacking skills for a cause or purpose. A hacktivist commits criminal activities to further their cause. A hacktivist attacks targets even when they know they will be identified, apprehended, and prosecuted. They do this because they believe their purpose or cause is more important than themselves. Keep in mind that committing crimes is still illegal, no matter what the intention or purpose of the perpetrator is. Hacktivism may often be used as a form of protest, but it is not a legal one.

Organized crime

Organized crime is involved in cybercrime activities because it is yet another area of exploitation that may allow criminals to gain access, power, or money. Although not all hacks and attacks are funded or backed by organized crime groups, their involvement in cyberattacks is quite significant. Some organized crime syndicates actively recruit skilled hackers to join their ranks in the criminal enterprise.

Nation states/APT

Many governments and militaries—nation-states—are now using cyberattacks as yet another weapon in their arsenal against real or perceived enemies, whether internal or outside their borders. For examples of nation-state–sponsored cyber events, read up on the malicious code concepts of Stuxnet (uncovered by Symantec) (`https://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf`) and Flame (uncovered by Kaspersky) (`https://www.kaspersky.com/flame`).

APT (advanced persistent threat) is any form of cyberattack that is able to continually exploit a target over a considerable period of time. An APT often takes advantage of unknown flaws (that is, not publicly known) and tries to maintain stealth throughout the attack. The name is derived from the concept that the attacks are unique and exploit flaws that are not public knowledge (that is, this state of using an exploit against an unknown flaw is labeled as advanced), that the exploit grants the attackers ongoing remote access to and control over the target (that is, it's persistent), and that the attackers are likely nation-states (that is, it's a threat). Sometimes APT is used to refer to the threat actors who continue to focus on targets using all available exploitations in order to retain control and dominance over the victim.

Insiders

One of the biggest risks at any organization is its own internal personnel. Hackers work hard to gain what insiders already have: physical presence within the facility or a working user account on the IT infrastructure. When an insider performs malicious activities, the threat is significant, because they're already past most physical barriers and may have easy access that lets them compromise IT security.

Malicious insiders can bring in malicious code from outside on various storage devices, including smartphones, memory cards, optical discs, and USB drives. These same storage devices can be used to leak or steal internal confidential and private data in order to disclose it to the outside world. (Where do you think most of the content on WikiLeaks comes from?) Malicious insiders can execute malicious code, visit dangerous websites, or intentionally perform harmful activities.

The means to reduce the threat of malicious insiders include thorough background checks, strong policies with severe penalties, detailed user activity auditing and monitoring, prohibition of external and private storage devices, and use of whitelists to minimize unauthorized code execution.

Competitors

Another type of threat actor is that of competitors. Many organizations still elect to perform corporate espionage and sabotage against their competition while it is widely known that such actions are illegal. This might be due to a perceived advantage another company has or the lack of desire to put forth the time and effort to gain valid market share in a competitive marketplace. Organizations should always take care to closely monitor their competition for signs that they are benefiting from and launching cyberattacks. This concept is known as competitive intelligence gathering. Competitive intelligence gathering is a valid and legal means to keep track of another company by analyzing publicly available information.

Companies should also pay special attention to business partners, contractors, and employees who may have left an organization only to gain employment with a competitor (whether you hire someone from the other firm or they hire away your employees).

Attributes of actors

Threat actors can have a wide range of skills and attributes. When analyzing the threats to your organization, it is important to keep these variables in mind.

Internal/external

Threats can originate from inside your organization as well as outside. All too often, companies focus most of their analysis and security deployment efforts on external threats without providing sufficient attention to the threats originating from inside. All threats should be considered on their merits—their specific risk level to your organization and its assets—and not just based on someone's subjective perspective on the issues.

Level of sophistication

Threat actors can vary greatly in their skill level and level of sophistication. Some attackers are highly trained professionals who are applying their education to malicious activities, whereas others are simply bad guys who learned how to perform cyberattacks just to expand their existing repertoire.

Some attacks are structured or targeted, others are unstructured and opportunistic. A structured or targeted attack is one where a specific organization was always the focus and considerable effort was expended to find a means to compromise that organization's security. This type of an attack usually involves a higher level of sophistication because there is a need to be methodical and persistent in seeking to accomplish the goal. An unstructured or opportunistic attack is one that seeks out a target that happens to be vulnerable to a chosen attack or exploit. This type of attack is often performed by an attacker once they have crafted a new exploit tool; they seek out a target to show off or demonstrate that their tool actually works. This type of attack displays a much lower level of sophistication. Yes, building a new attack tool can be a complex process, but the action of hitting multiple targets until you finally locate one with a particular weakness is not complicated. This is the equivalent of giving a toddler a hammer; if they treat everything like a nail and hit it, maybe eventually they will actually hit a real nail.

Resources/funding

Some threat actors are well funded with broad resources, whereas others are not. Some threat actors self-fund; others find outside investors or paying customers. Self-funded threat actors might highjack or use advertisement platforms to obtain funds; others may use ransomware to extort money from their victims. Some hackers offer their services like mercenaries to clients who pay the attackers to harm a specific target or craft a new exploit for a particular vulnerability. Some actors are paid by individuals, some by corporations, and others by nation-states.

Intent/motivation

The intent or motivation of an attacker can be unique to the individual or overlap with your own. Some attackers are motivated by the obvious benefit of money and notoriety. Others attack from boredom or just to prove to themselves that they can. Others find a thrill in the attack or are encouraged by the challenge. Some attackers are just drones in a crime group who have a daily boring drudgery of attacking a list of targets provided to them (think cubicle farms of script kiddies). Some attackers do it for fun, and others out of necessity (to earn money to feed their families). Some attack based on philosophy, political ideology, religious views, perspective on the environment, or disagreement with a business plan. Sometimes attackers have motivations that we will never know about—or might not comprehend even if we learn about them.

Use of open-source intelligence

Open-source intelligence is the gathering of information from any publicly available resource. This includes websites, social networks, discussion forums, file services, public databases, and other online sources. It also includes non-Internet sources, such as libraries and periodicals. Any information that may have been distributed by a target or by any other entity about the target is the focus of open-source intelligence gathering. The process, techniques, and methodologies used to collect open-source intelligence can be called reconnaissance, information gathering, footprinting, fingerprinting, or target research in hacking methodologies.

Exam Essentials

Define a threat actor. A threat actor is the person or entity who is responsible for causing or controlling any security-violating incidents experienced by an organization or individual.

Define script kiddies. Script kiddies are threat actors who are less knowledgeable than a professional skilled attacker. A script kiddie is usually unable to program their own attack tools and may not understand exactly how the attack operates.

Define a hacktivist. A hacktivist is someone who uses their hacking skills for a cause or purpose. A hacktivist commits criminal activities to further their cause.

Understand how organized crime is involved in cybercrime. Organized crime is involved in cybercrime activities because it is yet another area of exploitation that may allow them to gain access, power, or money.

Understand how nation-states are using cyberattacks. Most nation-states are now using cyberattacks as yet another weapon in their arsenal against their real or perceived enemies, whether internal or outside their borders.

Define APT. APT (advanced persistent threat) is any form of cyberattack that is able to continually exploit a target over a considerable period of time. An APT often takes advantage of flaws not publicly known and tries to maintain stealth throughout the attack.

Understand the risks presented by insiders. One of the biggest risks at any organization is its own internal personnel. Hackers work hard to gain what insiders already have: physical presence within the facility or a working user account on the IT infrastructure.

Understand the risks presented by competitors. While it is widely known that such actions are illegal, many organizations still elect to perform corporate espionage and sabotage against their competition.

Understand the risks presented by internal and external threat actors. Threats can originate from inside your organization as well as outside. All too often, companies focus most of their analysis and security deployment efforts on external threats without providing sufficient attention to the threats originating from inside.

Understand threat actors' level of sophistication. Threat actors can vary greatly as to their skill level and level of sophistication. Some attackers are highly trained professionals who are applying their education to malicious activities, whereas others are simply bad guys who learned how to perform cyberattacks just to expand their existing repertoire.

Know how threat actors access resources and funding. Some threat actors are well funded with broad resources; others are not. Some threat actors self-fund, whereas others find outside investors or paying customers. Self-funded threat actors might highjack or use advertisement platforms to obtain funds; others may use ransomware to extort money from their victims.

Understand threat actors' intent and motivation. The intent or motivation of an attacker can be unique to the individual or may be similar to your own. Some attackers are motivated by the obvious benefits of money and notoriety. Others attack from boredom or just to prove to themselves that they can.

Understand open-source intelligence. Open-source intelligence is the gathering of information from any publicly available resource. This includes websites, social networks, discussion forums, file services, public databases, and other online sources. It also includes non-Internet sources, such as libraries and periodicals.

1.4 Explain penetration testing concepts.

Penetration testing is a form of security evaluation that involves the same tools, techniques, and methodologies used by criminal hackers but is performed by security professionals. Penetration testing is also known as *ethical hacking* or *pen testing*.

Active reconnaissance

Active reconnaissance is collecting information about a target through interactive means. By directly interacting with a target, a person can quickly collect accurate and detailed information, but at the expense of potentially being identified as an attacker rather than just an innocent, benign, random visitor. Examples of activities that are considered active reconnaissance include visiting the target's website, performing port scanning (discussed next), speaking with the target's tech support or help desk service, visiting their physical location, and performing vulnerability scans against the target's systems.

One common function or task performed during active reconnaissance is *port scanning*. A port scanner is a vulnerability assessment tool that sends probe or test packets to a target system's ports in order to learn about the status of those ports. A port can be in one of two states: open or closed. If a valid request for connection is sent to an open TCP port (a SYN flagged packet), a normal response can be expected (a SYN/ACK flagged packet). If the TCP port is closed, the response is an RST packet. However, if a firewall is present, the firewall can filter out connection attempts on closed ports, resulting in no packet being received by the probing system. This is known as *filtering*. Thus, a TCP port scanner will have direct proof that a port is open or closed but can assume a filtered port if no response is received.

Although this form of probing works effectively, it produces traffic that is likely recorded or logged by the target system or the firewall protecting it. Thus, many other forms of port scanning have been developed. Some scanning techniques use standard packets but in an unexpected context, such as FIN or ACK flagged packets. These packets have no valid meaning outside of a valid TCP setup or teardown handshake; thus when used out of context, they may illicit a response that is meaningful to the probing entity. Even a normal data packet, which doesn't have any header flags enabled, can be used in a *NULL scan*. There are even some methods of scanning that use invalid packet constructions, such as the Xmas scan, which has numerous header flags enabled (see the earlier section "Xmas attack").

The details of how these scans operate are a bit beyond the Security+ content. However, it's important to understand that port scans allow security testers and hackers to discover what ports are open on a system. Once the open ports are known on a target, this information can lead to other important details, such as the identity of the host OS and what types of services are hosted on the target. Many port-scanning tools, such as Nmap, can not only detect open, closed, and stealth ports, but also determine the OS and identify active services on a port. Sometimes these actions are performed using a database of characteristics, and sometimes they're performed using banner-grabbing queries. A *banner grab* (see Chapter 2, "Technologies and Tools") occurs when a request for data or identity is sent to a service on an open port and that service responds with information that may directly or indirectly reveal its identity.

Passive reconnaissance

Passive reconnaissance is the activity of gathering information about a target without interacting with the target. Instead, information is collected from sources not owned and controlled by the target (other websites and services) as well as by eavesdropping on communications from the target. A significant amount of information can be gathered through passive reconnaissance, but it may not be as accurate as data gathering through active means. Additionally, eavesdropping-based reconnaissance may require a significant length of time in order to gather useful information, because you will be waiting for the transmission of the data you wish to obtain based on the normal activities of the target.

Examples of activities performed during passive reconnaissance include visiting social network sites, reading third-party reports, searching discussion forums (not operated by the target), researching domain name and IP address registrations, and visiting any other online or offline source not owned, controlled, or monitored by the target.

Pivot

In penetration testing (or hacking in general), a *pivot* is the action or ability to compromise a system, and then use the privileges or access gained through the attack to focus attention on another target that may not have been visible or exploitable initially. It is the ability to adjust the focus or the target of an intrusion after an initial foothold is gained. It is potentially possible to pivot from the compromise of computer A to launch attacks against computer B, when computer B was not accessible earlier, or once computer A is compromised, information hosted on computer A can be accessed. This could include files, database contents, security settings, and account credentials.

Pivoting also relates to *daisy chaining*, the concept of performing several exploitations in a series in order to achieve a goal on the target. Often with modern defense-in-depth or diversity-of-defense security infrastructures, a single attack is insufficient to achieve a compromise goal. Instead, several successive attacks must be waged, each dependent on and building on the success of the previous exploits. For example, a daisy chaining attack could include an initial port scan to find an open port, followed by an exploitation of the application behind the open port, which executes code to change the firewall configuration to open another port, which leads to compromising another service (which was previously inaccessible), in order to launch an injection attack, which dumps out user account credentials. This series of attack events is also an example of pivoting, because each successful attack leads to another target and exploitation.

Initial exploitation

The *initial exploitation* in a penetration test or a real-world malicious attack is the event that grants the attacker/tester access to the system. It is the first successful breach of the organization's security infrastructure that grants the attacker/tester some level of command control or remote access to the target. All steps prior to the initial exploitation—reconnaissance, port scanning, enumeration, and vulnerability detection—lead up to and make possible the initial exploitation. Once the initial exploitation is successful, the later stages of attack can occur: establishing persistent connect and control over the target and hiding all traces of the intrusion.

Persistence

Persistence is the characteristic of an attack that maintains remote access to and control over a compromised target. Some attacks are quick one-off events where the initial compromise triggers some result, such as stealing data, planting malware, destroying files, or crashing the system. But such events are short-lived "one-time, then done" occurrences, not persistent attacks. A persistent attack grants the attacker ongoing prolonged access to and control over a victim system and/or network.

Escalation of privilege

Escalation of privilege is any attack or exploit that grants the attacker greater privileges, permissions, or access than may have been achieved by the initial exploitation or that a legitimate user was assigned. Privilege escalation can be either horizontal or vertical. A horizontal privilege escalation occurs when an attack is able to jump from controlling one lower-level user account into controlling a high-level user account. This is often accomplished through credential theft using keystroke loggers planted through the initial account's capabilities, which might be installing the malware directly or sending a Trojan horse installer via email attachment. A vertical privilege escalation occurs when an attack exploits a flaw in the system or software that makes the current user account a member of an admin group, converts the account into an admin account, or simply enables the execution of commands as the system or root.

Black box

It's important to understand various terms for penetration (and other forms of) testing. A *black box* is literally a device whose internal circuits, makeup, and processing functions are unknown but whose outputs in response to various kinds of inputs can be observed and analyzed. Black-box penetration testing proceeds without using any initial knowledge of how an organization is structured, what kinds of hardware and software it uses, or its security policies, processes, and procedures.

Black-box testing requires that the penetration testers spend significant time and effort during the earlier phases of hacking to discover as much as possible about the operations of the "black box" of the target network and systems. This causes a black-box test to take the most time and cost the most (among the black, white, and gray testing options), but it provides a realistic external criminal hacker perspective on the security stance of an organization.

White box

By contrast, a *white box* is a device whose internal structure and processing are known and understood. This distinction is important in penetration testing, where white-box testing makes use of knowledge about how an organization is structured, what kinds of hardware and software it uses, and its security policies, processes, and procedures. It could be called invisible box or transparent box testing. White-box testing seeks to exploit everything known about the operations and functions of the network to focus and guide testing efforts. White-box penetration testing uses all available knowledge to drive its efforts.

White-box testers need not devote significant time and effort to reconnaissance, but perform only enough initial research activities to confirm the information provided. The overall time for a white-box test is much shorter and thus it costs significantly less as well. However, the result is that it gives a rogue administrator a lot of information about the organization's security. This is the type or form of penetration testing that is most over overlooked or discounted, because it is hard for organizational leaders to conceive of their most trusted IT administrators ever turning on them.

Gray box

Gray-box testing combines the two other approaches to perform an evaluation based on partial knowledge of the target environment. This requires some time spent on reconnaissance, and costs are usually between those of white- and black-box testing. The results are a security evaluation from the perspective of a disgruntled employee. An employee has some knowledge of the organization and its security and has some level of physical and logical access.

Pen testing vs. vulnerability scanning

Vulnerability scanning and penetration testing are important aspects of detecting and responding to new vulnerabilities and weaknesses. In addition to these important tools, ongoing monitoring of performance, throughput, and protocol use can reveal trends toward downtime, change in job focus, and the need for infrastructure upgrades.

A penetration test is a form of vulnerability scan that is performed by a special team of trained, white-hat security specialists rather than by an internal security administrator using an automated tool. Penetration testing (also known as ethical hacking) uses the same tools, techniques, and skills of real-world criminal hackers as a methodology to test the deployed security infrastructure of an organization. Penetration testing gives you the perspective of real hackers, whereas typical vulnerability scanning offers only the security perspective of the scanner's vendor.

To best simulate a real-life situation, penetration testing is usually performed without the IT or security staff being aware of it. Senior management often schedules ethical hacking events. This allows the penetration test to assess the performance of the infrastructure and the response personnel. This is known as an *unannounced test*. An *announced test* means everyone in the organization knows the penetration assessment is taking place and when.

Penetration tests can take many forms, including hacking in from the outside, simulating a disgruntled employee, social engineering attacks, and physical attacks, as well as remote connectivity, wireless, and VPN attacks. The goal of penetration testing is to discover weaknesses before real criminals do. Most penetration testing requires high levels of knowledge and skill on the part of the testers. Automated tools are employed, but most of the benefit derived from a penetration test is from the skill of the testers modifying existing exploits or crafting custom code for attacks. This is because real hackers often write their own surgically precise attack tools and scripts based on their target. Security administrators do use automated tools for vulnerability scanning to check for policy compliance and known issues. Penetration testing is used to discover new weaknesses that these automated tools can't find.

In security terms, a *penetration* occurs when an attack is successful and an intruder is able to breach the perimeter around your environment. A breach can be as small as reading a few bits of data from your network or as big as logging in as a user with unrestricted privileges. A primary goal of security is to prevent penetrations.

One common method you can employ to test the strength of your security measures is to perform penetration testing, a vigorous attempt to break into your protected network using any means available. It's common for organizations to hire external consultants to perform penetration testing so testers aren't privy to confidential elements of the environment's security configuration, network design, and other internal secrets.

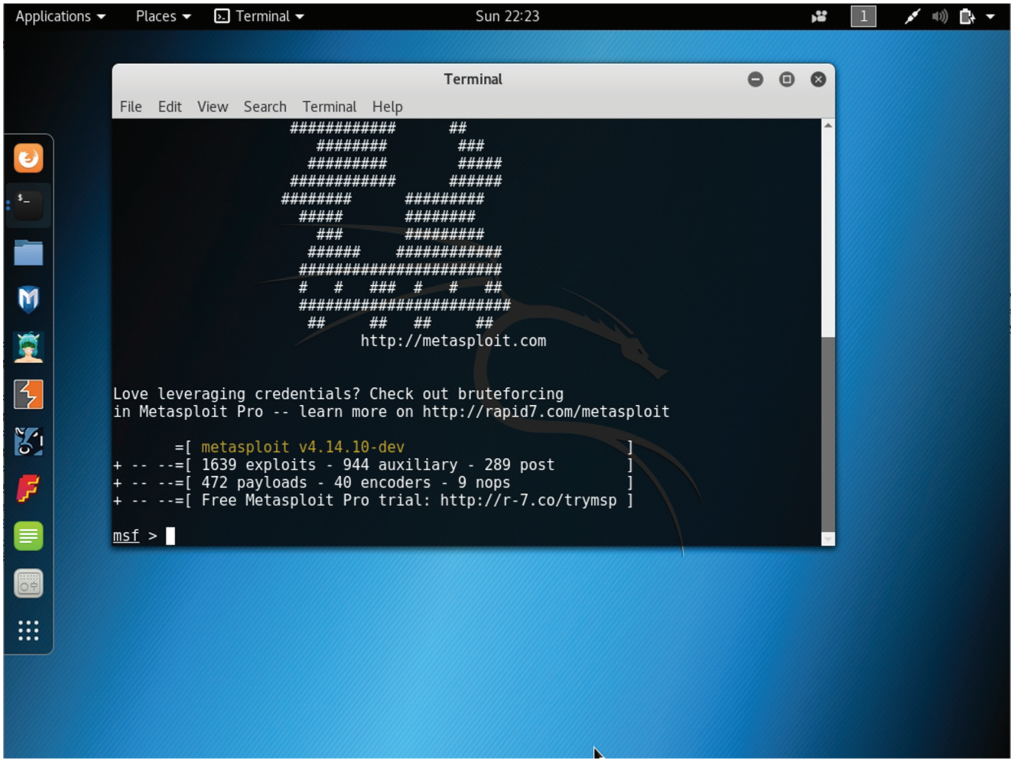
Penetration testing seeks to find any and all detectable weaknesses in your existing security perimeter. The operative term is *detectable*; there are undetected and presently unknowable threats lurking in the large-scale infrastructure of network software and hardware design that no amount of penetration testing can directly discover. Once a weakness is discovered, countermeasures can be selected and deployed to improve security in the environment. One significant difference between penetration testing and an actual attack is that once a vulnerability is discovered during a penetration test, the intrusion attempt ceases before a vulnerability exploit can cause any damage. There are open-source and commercial tools (such as Metasploit [Figure 1.14], Immunity's CANVAS, and CORE Impact) that can be considered active security scanners or exploitation frameworks; they allow you to take penetration testing one step further and attempt to exploit known vulnerabilities in systems and networks. These tools may be used by good guys and bad guys alike.

Penetration testing may use automated attack tools or suites or may be performed manually using common network utilities and scripts. Automated attack tools range from professional vulnerability scanners to wild, underground tools discovered on the Internet. Tools are also often used for penetration testing that's performed manually, but the real emphasis is on knowing how to perpetrate an attack.

Penetration testing should be performed only with the consent and knowledge of management (and security staff). Performing unapproved security testing could cause productivity losses, trigger emergency response teams, or even cost you your job and potentially earn you jail time.

Regularly staged penetration tests are a good way to accurately judge the security mechanisms deployed by an organization. Penetration testing can also reveal areas where patches or security settings are insufficient and where new vulnerabilities have developed. To evaluate your system, benchmarking and testing tools are available for download at `www.cisecurity.org`, and a somewhat comprehensive list of security assessment and hacker/penetration testing tools is available from `www.sectools.org`.

FIGURE 1.14 The CLI (command-line interface) of Metasploit on Kali Linux



```
                    ############      ##
                       ########        ###
                      #########        #####
                    ############      ######
                   ########      #########
                     #####       ########
                       ###       #########
                      ######    ############
                     #######################
                     #   #   ###  #   #   ##
                     ########################
                      ##     ##   ##     ##
                            http://metasploit.com


Love leveraging credentials? Check out bruteforcing
in Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.14.10-dev                         ]
+ -- --=[ 1639 exploits - 944 auxiliary - 289 post        ]
+ -- --=[ 472 payloads - 40 encoders - 9 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > 
```

Identifying and repelling attacks requires an explicit, well-defined body of knowledge about their nature and occurrence. Some attack patterns leave behind signatures that make them readily apparent to casual observation with IDS instrumentation; other forms of attack are esoteric or not conducive to pattern-matching engines and therefore must be measured against a baseline of acceptable activity.

What elements or properties signify an attack sequence rather than a benign traffic formation? Answering this question depends on careful, attentive security professionals keeping up with the latest attacks, vulnerabilities, exploits, and security bulletins (like those from the U.S. Computer Emergency Readiness Team at `www.us-cert.gov/cas/bulletins` or those from the Common Vulnerabilities and Exposures database at `http://cve.mitre.org`).

Before implementing a fix or a security control, it's important to verify that a problem actually exists. There is no point in protecting against a threat if your environment doesn't have the vulnerability. Likewise, if the threat doesn't exist or is extremely unlikely to ever become realized in your organization, implementing countermeasures may also be unwarranted.

Part of penetration testing is to confirm whether a vulnerability exists and whether a real threat exists. Based on the criticality of known threats, vulnerabilities, and risks, you can determine whether to respond by implementing a countermeasure, assigning the risk elsewhere, or accepting the risk.

Hackers often attempt to find a way to bypass security controls. An ethical hacker or penetration tester attempts many of these same techniques so that you can be aware of them before they're abused by someone malicious. Means of bypassing security controls vary greatly, but some common general categories include using alternate physical or logical pathways, overloading controls, and exploiting new flaws. If hackers know that a specific pathway of approach is secured, they may seek an alternate route. For example, if all Internet-sourced traffic is filtered by a firewall, a hacker may try to locate a modem or an unauthorized wireless access point on the network to bypass the firewall's security.

Sometimes DoS/DDoS attacks can be used to overload firewalls, IDS, IPS, auditing, and so on, so that these security tools are "distracted" while the real attack takes place. Also, new exploits are being crafted daily that may be able to compromise security through exploitation of faulty programming code. For examples, see the Exploit Database at `www.exploit-db.com` for a current list of exposed new and zero-day exploits.

Just because an electronic lock or other form of access control is in use, that doesn't ensure that bypassing the system is impossible. Ways to bypass electronic controls include turning off the power, creating a short circuit, introducing an alternative power supply, bypassing triggering circuits, and overloading detectors with false positives.

A penetration test should be used to find new flaws or unknown vulnerabilities as well as to test the abilities of the deployed security infrastructure. If current security controls aren't sufficient or can be easily bypassed, a thorough penetration test should reveal this. If your security posture isn't resilient enough to catch proficient ethical hackers, then it's unlikely that it's good enough to catch professional criminal hackers.

A penetration test should discover vulnerabilities and then exploit them to a predetermined extent. The testing should not be performed to the point of causing unrepairable damage or prolonged downtime. The whole point of penetration testing is for the testers to act ethically and within restrictions or boundaries imposed by the service-level agreement (SLA) or testing contract. Any test that might cause harm should gain specific preapproval before it's executed. Additionally, the target being tested should be prepared with recent backups and a recovery team just in case the tester's precautions aren't sufficient or the attack accidentally is more extensive than expected.

Exam Essentials

Understand active reconnaissance. Active reconnaissance is the idea of collecting information about a target through interactive means. By interacting with a target, accurate and detailed information can be collected quickly but at the expense of potentially being identified as an attacker rather than just an innocent, benign, random visitor.

Know how to use port scanners. A port scanner is a vulnerability assessment tool that sends probe or test packets to a target system's ports in order to learn about the status of those ports.

Understand passive reconnaissance. Passive reconnaissance is the activity of gathering information about a target without interacting with the target. Instead, information is collected from sources not owned and controlled by the target (other websites and services) as well as by eavesdropping on communications from the target.

Define pivoting. In penetration testing (or hacking in general), a pivot is the action or ability to compromise a system, and then using the privileges or access gained through the attack to focus attention on another target that may not have been visible or exploitable initially.

Understand initial exploitation. The initial exploitation in a penetration test or a real-world malicious attack is the event that grants the attacker/tester access to the system. It is the first successful breach of the organization's security infrastructure that grants the attacker/tester some level of command control or remote access to the target.

Define persistence. Persistence is the concept of an attack that maintains remote access to and control over a compromised target. A persistent attack grants the attacker ongoing prolonged access to and control over a victim system and/or network.

Understand escalation of privilege. Escalation of privilege is any attack or exploit that grants the attacker greater privileges, permissions, or access than what may have been achieved by the initial exploitation. Privilege escalation can be either horizontal or vertical.

Understand black-box testing. Black-box penetration testing proceeds without using any initial knowledge of how an organization is structured; what kinds of hardware and software it uses; or its security policies, processes, and procedures. It provides a realistic external criminal hacker perspective on the security stance of an organization.

Understand white-box testing. White-box testing makes use of knowledge about how an organization is structured, what kinds of hardware and software it uses, and its security policies, processes, and procedures. The result is that it gives a rogue administrator a lot of information about the organization's security.

Understand gray-box testing. Gray-box testing combines the two other approaches to perform an evaluation based on partial knowledge of the target environment. The results are a security evaluation from the perspective of a disgruntled employee.

Understand penetration testing. A penetration test is a form of vulnerability scan that is performed by a special team of trained white-hat security specialists rather than by an internal security administrator using an automated tool. Penetration testing (also known as ethical hacking) uses the same tools, techniques, and skills of real-world criminal hackers as a methodology to test the deployed security infrastructure of an organization.

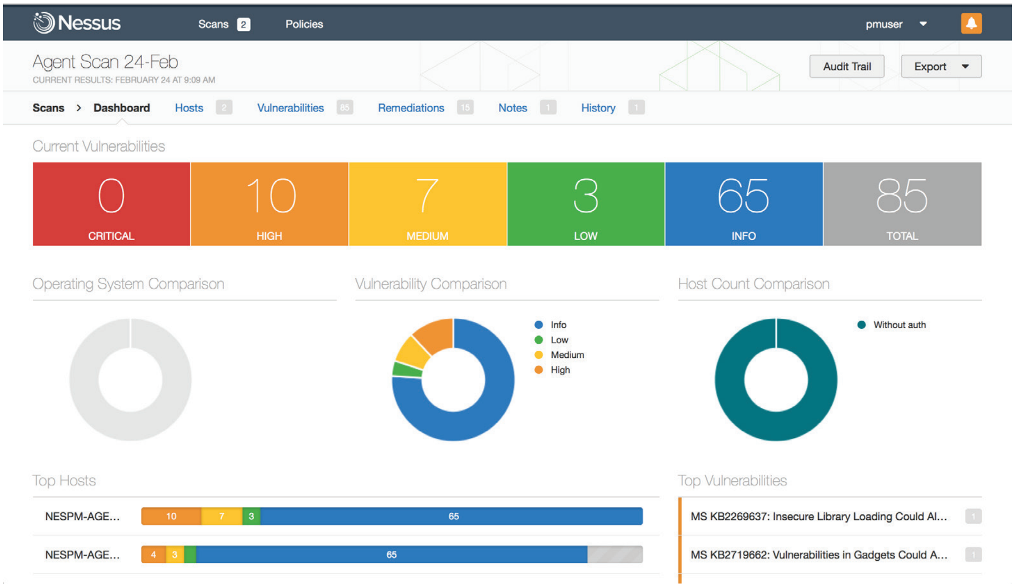
1.5 Explain vulnerability scanning concepts.

Vulnerability scanning is used to discover weaknesses in deployed security systems in order to improve or repair them before a breach occurs. By using a wide variety of assessment tools (such as vulnerability scanners, protocol analyzers, network scanners, and wireless scanners), security administrators can learn about deficiencies quickly. Only through vigilance and constant monitoring and assessment can a security endeavor prove successful.

Typically, vulnerability scanning should be performed by security administrators on a regular periodic basis (such as weekly). Additionally, only after thoroughly performing vulnerability scanning and responding to/addressing each alert item is an organization ready for a true penetration test. A penetration test requires dedicated full-time testing professionals, who are often external consultants. A vulnerability scan can be run by any reasonably skilled and knowledgeable IT or security administrator with a little training and lab testing.

Vulnerability scanners and security-assessment tools are used to test a system for known security vulnerabilities and weaknesses. They're used to generate reports that indicate the aspects of the system that need to be managed to improve security. The reports may recommend applying patches or making specific configuration or security setting changes to improve or impose security (Figure 1.15).

FIGURE 1.15 The scan summary report from the vulnerability scanner Nessus



A vulnerability scanner is only as useful as its database of security issues. Thus, the database must be updated from the vendor often to provide a useful audit of your system. The use of vulnerability scanners in conjunction with an IDS may help reduce false positives by the IDS and keep the total number of overall intrusions or security violations to a minimum. When discovered vulnerabilities are patched quickly and often, the system provides a more secure environment.

An extension to the concept of the IDS is the IPS. An IPS seeks to actively block unauthorized connection attempts or illicit traffic patterns as they occur. IPS designs fall under the same type (host- and network-based) and classification (behavior- and signature-based) as the IDS counterparts, and they're often deployed together for complete network coverage. Additionally, many IPS platforms are capable of dissecting higher-level application protocols in search of malicious payloads. The line between IDSs and IPSs can be blurred in that many self-professed IDSs have IPS capabilities. These days, detection and prevention systems occur together more often than they do separately.

The results of a vulnerability scan need to be interpreted by a knowledgeable security expert. Automated scanning tools can produce numerous false positives; thus it may be necessary to confirm the presence of a security flaw before implementing a fix, especially if the fix is costly or interferes with production. Another issue is that the criticality level reported by a scanning tool may not be accurate or relevant to your organization. Finally, the results of a vulnerability scan must be interpreted in light of the existing environment, known real threats, and budget.

Passively test security controls

A passive test of security controls is being performed when an automated vulnerability scanner is being used that seeks to identify weaknesses without fully exploiting discovered vulnerabilities. In most cases, automated vulnerability scanners detect the security control as it attempts a test. Additionally, because the security controls are operating while the automated vulnerability scan is being performed, the security controls get a workout at the same time the actual targets are the focus of the scan. Thus, passively testing security controls takes place any time tests are performed against targets but not specifically directed toward the security measures themselves.

Actively testing security controls involves attempting to fully exploit and breach a target system. This might be performed using active scanners, also known as exploitation frameworks, or using manual attacks.

Identify vulnerability

A scanner that is able to identify a vulnerability does so through a testing probing process defined in its database of evaluations. The goal of a vulnerability scanner is to inform you of any potential weaknesses or attack points on your network, within a system, or against an individual application. In most cases, a vulnerability scanner evaluates a target using surface probing activities, which does not fully exploit the potential flaw. Thus the report is listing all issues based on the symptoms of a vulnerability, not the confirmed result of an actual exploitation. This causes some of the items on the report to be false positives. Thus, it is important for system managers to investigate each item on a vulnerability scanner's report to confirm whether or not they are actual exploitable flaws before undertaking any mitigation activities.

Identify lack of security controls

An important task for a vulnerability scanner is to identify any necessary or best-practice security controls that are not present in the evaluated target. Such a report may indicate that updates and patches are not applied or that a specific security mechanism is not present, such as encryption, antivirus scanning, a firewall, and so on. If a vulnerability scanner can easily determine that your environment is missing key elements of a security infrastructure, so, too, can a hacker discover this and take advantage of your lack of sufficient protection.

Identify common misconfigurations

Many vulnerability scanners can determine whether or not you have improper, poor, or misconfigured systems and protections. If a vulnerability scanner is able to detect this issue, so can an attacker. Be sure to correct any discovered misconfigurations immediately.

Intrusive vs. non-intrusive

An *intrusive* vulnerability scan (also known as *active evaluation*) attempts to exploit any flaws or vulnerabilities detected. A *nonintrusive* vulnerability scan (also known as *passive evaluation*) only discovers the symptoms of flaws and vulnerabilities and doesn't attempt to exploit them. Traditionally, a vulnerability scanner is assumed to be nonintrusive, whereas a penetration test is assumed to be intrusive. However, a range of assessment tools can now provide either form of evaluation.

Credentialed vs. non-credentialed

A *credentialed* scan is one where the logon credentials of a user, typically a domain administrator, must be provided to the scanner in order for it to perform its work. The account credentials provided are most likely a domain account rather than a local account, such as root or administrator. A *noncredentialed* scan is one where no user accounts are provided to the scanning tool, so only those vulnerabilities that don't require credentials are discovered. Both forms of scanning should be used to provide a thorough evaluation of your security infrastructure.

False positive

A *false positive* is the occurrence of an alarm or alert due to a benign activity being initially classified as potentially malicious. The problem with false positives is they cause security administrators to waste time investigating nonmalicious events. Over time, and after repeated false positives, security admins may stop responding to alarms and assume all alerts are false.

An even more important issue to address is the *false negative*. Whereas a false positive is an alarm without a malicious event, a false negative is a malicious event without an alarm. When false negatives occur, it is assumed that only benign events are occurring; however, malicious activities are actually taking place. This is the equivalent of a building burning without fire alarms.

A *false negative* occurs when an alarm or alert is not triggered by malicious or abnormal events. False negatives occur when poor detection technologies are used, when detection databases are not kept current, or when an organization is facing a new, unknown zero-day threat. When malicious activities are occurring and are not detected, the victim is unaware of the situation. They are actively being harmed while not being aware that the harm is occurring. Thus, they do not know that they need to make any response or adjustment. This is the realm of the unknown unknown.

	Malicious events	**Benign events**
Alarm/alert	True positive	False positive
No alarm/alert	False negative	True negative

To reduce the risk of false negatives, organizations should adopt a deny-by-default or implicit-deny security stance. This stance centers on the idea that nothing is allowed to occur, such as execution, unless it is specifically allowed (placed on a whitelist or an exception list). It is also good practice to keep detection technologies, such as firewalls, intrusion detection systems (IDSs), and intrusion prevention systems (IPSs), current in terms of their core engines as well as their rule lists and detection databases.

Exam Essentials

Understand vulnerability scanning. Vulnerability scanning is used to discover weaknesses in deployed security systems in order to improve or repair them before a breach occurs. By using a wide variety of assessment tools, security administrators can learn about deficiencies quickly.

Understand passive testing of security controls. A passive test of security controls is being performed when an automated vulnerability scanner is being used that seeks to identify weaknesses without fully exploiting discovered vulnerabilities.

Understand vulnerability identification. A scanner that is able to identify a vulnerability does so through a testing probing process defined in its database of evaluations. The goal of a vulnerability scanner is to inform you of any potential weaknesses or attack points on your network, within a system, or against an individual application.

Understand the identification of a lack of security controls. An important task for a vulnerability scanner is to identify any necessary or best-practice security controls that are not present in the evaluated target. Such a report may indicate that updates and patches are not applied or that a specific security mechanism is not present.

Be able to identify common misconfigurations. Many vulnerability scanners can determine whether or not you have improper, poor, or misconfigured systems and protections. If a vulnerability scanner is able to detect this issue, so can an attacker.

Understand intrusive vs. nonintrusive. An intrusive vulnerability scan attempts to exploit any flaws or vulnerabilities detected (also known as active evaluation). A nonintrusive vulnerability scan only discovers the symptoms of flaws and vulnerabilities and doesn't attempt to exploit them (also known as passive evaluation).

Understand credentialed vs. noncredentialed. A credentialed scan is one where the logon credentials of a user, typically a system administrator or the root, must be provided to the scanner in order for it to perform its work. A noncredentialed scan is one where no user accounts are provided to the scanning tool, so only those vulnerabilities that don't require credentials are discovered.

Know what a false positive is. A false positive occurs when an alarm or alert is triggered by benign or normal events.

Know what a false negative is. A false negative occurs when an alarm or alert is not triggered by malicious or abnormal events.

1.6 Explain the impact associated with types of vulnerabilities.

There are many different forms and types of hacks, attacks, exploits, and intrusions. Many of these compromises can cause significant harm or damage to a system as well as impede the ability of an organization to continue normal operations. This section focuses on the impact that some forms of vulnerabilities and their exploitation may have on an organization.

Race conditions

Computer systems perform tasks with rigid precision. Computers excel at repeatable tasks. Attackers can develop attacks based on the predictability of task execution. The common sequence of events for an algorithm is to check that a resource is available and then access it if you are permitted. The time of check (TOC) is the time at which the subject checks on the status of the object. There may be several decisions to make before returning to the object to access it. When the decision is made to access the object, the procedure accesses it at the time of use (TOU). The difference between the TOC and the TOU is sometimes large enough for an attacker to replace the original object with another object that suits their own needs. *Time-of-check-to-time-of-use (TOCTTOU)* attacks are often called *race conditions* because the attacker is racing with the legitimate process to replace the object before it is used.

A classic example of a TOCTTOU attack is replacing a data file after its identity has been verified but before data is read. By replacing one authentic data file with another file of the attacker's choosing and design, an attacker can potentially direct the actions of a program in many ways. Of course, the attacker would have to have in-depth knowledge of the program and system under attack.

Likewise, attackers can attempt to take action between two known states when the state of a resource or the entire system changes. Communication disconnects also provide small windows that an attacker might seek to exploit. Anytime a status check of a resource precedes action on the resource, a window of opportunity exists for a potential attack in the brief interval between check and action. These attacks must be addressed in your security policy and in your security model. TOCTTOU attacks, race condition exploits, and communication disconnects are known as state attacks because they attack timing, dataflow control, and transition between one system state and another.

Another form of race condition attack occurs when two processes are running concurrently but one is designed to finish first and then provide its results to the second process in order for it to complete its tasks. If the first process is delayed in completing its task, this may cause the second process to be vulnerable to injection of malicious content (since it is not receiving the needed input from the first process), or it may cause the second process to fail.

Race condition attacks can result in system takeover, data leakage, and data destruction.

Vulnerabilities due to:

Every nontypical and specialized system places unique and often complex security strains on your organization. It is important to keep these in mind when designing your security policy and performing network segmentation.

End-of-life systems

End-of-life systems are those that are no longer receiving updates and support from the vendor. If an organization continues to use an end-of-life system, then the risk of compromise is high because any future exploitation will never be patched or fixed. It is of utmost important to move off end-of-life systems in order to maintain a secure environment. It might not seem initially cost-effective or practical to move away from a solution that still works, just because the vendor has terminated support. However, the security management efforts you will expend will likely far exceed the cost of developing and deploying a modern system–based replacement.

Embedded systems

An *embedded system* is any form of computing component added to an existing mechanical or electrical system for the purpose of providing automation and/or monitoring. Embedded systems can be a security risk because they are generally static systems, meaning that even the administrators who deploy them have no real means to alter the device's operations in order to address security vulnerabilities. Some embedded systems can be updated with patches from the vendor, but often patches are released months after a known exploit is found in the wild. It is essential that embedded systems be isolated from the Internet and from a private production network in order to minimize exposure to remote exploitation, remote control, or malware compromise.

Lack of vendor support

Any system, whether hardware or software, will become more insecure over time once it lacks vendor support. Lack of support can be a "feature" of the product all along, where the vendor does not provide any improvement, support, or patching/upgrading of the product after the initial sale. As a security manager, you should avoid products that lack vendor support and phase out products as they reach their end-of-life date.

Improper input handling

Many forms of exploitation are caused by the lack of *input sanitization* or validation. Only with proper input handling can software exploitation be reduced or eliminated. There are three main forms of input filtering that should be adopted by every programmer and included in every code they author:

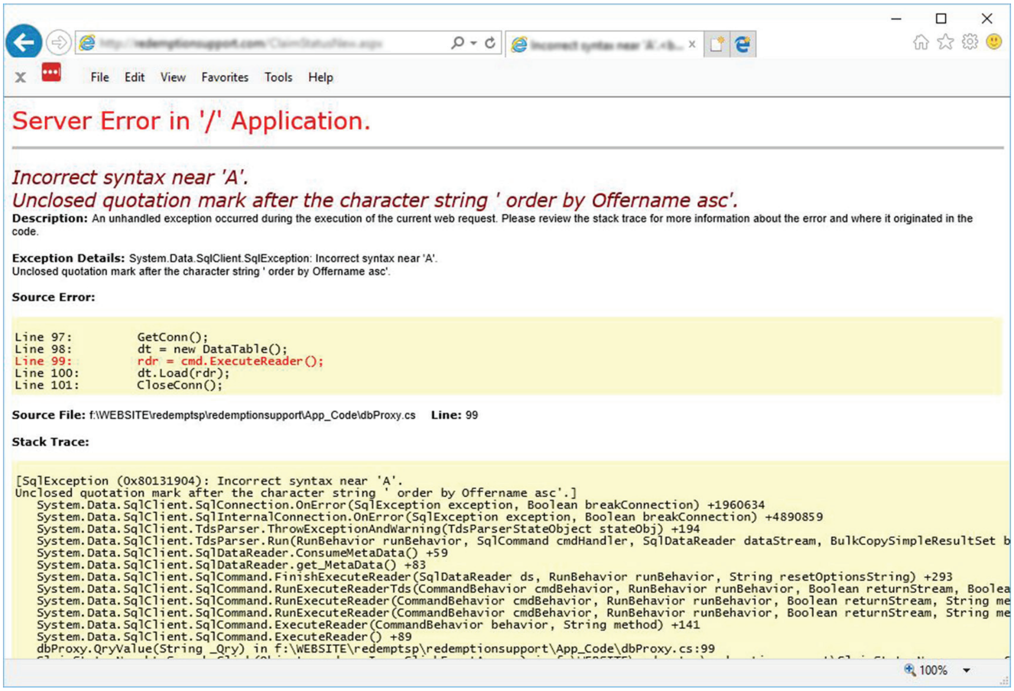
- Check for length.
- Filter for known malware patterns.
- Escape metacharacters.

Improper error handling

Improper error handling may allow for the leaking of essential information to attackers or enable attackers to force a system into an insecure state. If error messages are not handled properly, they may disclose details about a flaw or weakness that will enable an attacker to fine-tune their exploit. For example, if an attacker submits just a single quote to a target system, if the error response indicates that there is an unclosed quotation mark (Figure 1.16), then it informs the attacker that no metacharacter filtering is taking place. Otherwise, the error would have stated that invalid or out-of-bounds input was attempted but was rejected and that the user should try again.

If errors themselves are not handled properly, it could cause an application to disclose confidential data to a visitor, allow an attacker to bypass authentication, or even crash the system. Programmers should include an error management system in their products in order to handle invalid values, out-of-range data sets, or other forms of improper input. When an error is detected, the error management system should display a generic error message to the user, such as "error try again" or "error, contact technical support." The error management system should log all details about the error into a file for the administrator, but should not disclose those details to the user. Additionally, if an error could result in security violations, a general error fault response known as fail-secure should be initiated. A fail-secure system will revert to a secured, closed, protective state in the event of a failure rather than into an open, insecure, nonprotected state where information can be disclosed or modified.

FIGURE 1.16 An error page for a website that shows the lack of metacharacter filtering



Server Error in '/' Application.

Incorrect syntax near 'A'.
Unclosed quotation mark after the character string ' order by Offername asc'.

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it originated in the code.

Exception Details: System.Data.SqlClient.SqlException: Incorrect syntax near 'A'.
Unclosed quotation mark after the character string ' order by Offername asc'.

Source Error:

```
Line 97:         GetConn();
Line 98:         dt = new DataTable();
Line 99:         rdr = cmd.ExecuteReader();
Line 100:        dt.Load(rdr);
Line 101:        CloseConn();
```

Source File: f:\WEBSITE\redemptsp\redemptionsupport\App_Code\dbProxy.cs **Line:** 99

Stack Trace:

```
[SqlException (0x80131904): Incorrect syntax near 'A'.
Unclosed quotation mark after the character string ' order by Offername asc'.]
   System.Data.SqlClient.SqlConnection.OnError(SqlException exception, Boolean breakConnection) +1960634
   System.Data.SqlClient.SqlInternalConnection.OnError(SqlException exception, Boolean breakConnection) +4890859
   System.Data.SqlClient.TdsParser.ThrowExceptionAndWarning(TdsParserStateObject stateObj) +194
   System.Data.SqlClient.TdsParser.Run(RunBehavior runBehavior, SqlCommand cmdHandler, SqlDataReader dataStream, BulkCopySimpleResultSet b
   System.Data.SqlClient.SqlDataReader.ConsumeMetaData() +59
   System.Data.SqlClient.SqlDataReader.get_MetaData() +83
   System.Data.SqlClient.SqlCommand.FinishExecuteReader(SqlDataReader ds, RunBehavior runBehavior, String resetOptionsString) +293
   System.Data.SqlClient.SqlCommand.RunExecuteReaderTds(CommandBehavior cmdBehavior, RunBehavior runBehavior, Boolean returnStream, Boolea
   System.Data.SqlClient.SqlCommand.RunExecuteReader(CommandBehavior cmdBehavior, RunBehavior runBehavior, Boolean returnStream, String me
   System.Data.SqlClient.SqlCommand.RunExecuteReader(CommandBehavior cmdBehavior, RunBehavior runBehavior, Boolean returnStream, String me
   System.Data.SqlClient.SqlCommand.ExecuteReader(CommandBehavior behavior, String method) +141
   System.Data.SqlClient.SqlCommand.ExecuteReader() +89
   dbProxy.QryValue(String _Qry) in f:\WEBSITE\redemptsp\redemptionsupport\App_Code\dbProxy.cs:99
```

Misconfiguration/weak configuration

It is the responsibility of system implementers and those performing ongoing system management to verify that the correct and secure configuration items remain defined and enforced. When *misconfigurations* or *weak configurations* are allowed to remain while a system is in active productive use, the risk of data loss, data leakage, and overall system compromise is higher.

Default configuration

Default configurations should never be allowed to remain on a device or within an application. Defaults are intended for ease of installation and initial configuration in order to minimize support calls from new customers. As a system administrator, you should alter system settings from their defaults to a state that brings the system into compliance with your security policy. The tyranny of the default is the fact that defaults are usually insecure and thus leave a system open to simple compromise.

Resource exhaustion

Resource exhaustion occurs when applications are allowed to operate in an unrestricted and unmonitored manner so that all available system resources are consumed in the attempt to serve the requests of valid users or in response to a DoS attack. It is essential for system managers to monitor the baseline of productive valid resource consumption and watch for trends that may indicate a need to expand capacity or to respond to exploitative attacks.

Untrained users

Untrained users are more likely to make mistakes or abuse a system's resources and capabilities. Only trained workers should be allowed to use sensitive system resources. Organizations need to train new employees properly on test systems not directly tied to production. Only once a new employee shows proficiency in accomplishing tasks should he or she be moved into a live production system.

Improperly configured accounts

User accounts need to be properly configured in order to grant the correct level of resource access and system rights based on the job responsibilities of the employees. No matter what the means of authorization, users should be granted only enough powers to accomplish their work tasks. Any more than that minimum is simply increasing risk for the organization without any benefit. Improperly configured accounts violate the principle of least privilege. Workers should have only the object permissions or privileges and system user rights or capabilities that they need for their specific jobs.

Vulnerable business processes

All business tasks, processes, procedures, or functions should be assessed as to their importance to the organization and their relative vulnerabilities. This includes considering the confidentiality, integrity, and availability protections or deficiencies for each business process. Attention should also be given to the value and importance of the data sets that each business process processes.

Weak cipher suites and implementations

Not all ciphers or other algorithm elements in a cipher suite are secure. Many older algorithms or implementations of algorithms have known flaws, weaknesses, or means of compromise. These weaker ciphers should be avoided and disabled and replaced with stronger cipher suites with few or no issues. A cipher's age isn't necessarily an indication of strength or weakness. For a discussion about weak ciphers, cipher suite attacks, and Google's recommendations for the future, read "A roster of TLS cipher suites weaknesses" at `http://googleonlinesecurity.blogspot.com/2013/11/a-roster-of-tls-cipher-suites-weaknesses.html`.

Memory/buffer vulnerability

Memory is a key element of a computer system. It is the area holding data that was received as input, whether from the keyboard, network, or storage device. The area of memory set aside or assigned to hold input is known as a buffer. Memory or buffer attacks and exploits are serious security concerns.

Memory leak

A *memory leak* is the opposite of what the name might imply. A memory leak occurs when a program fails to release memory or continues to consume more memory. It's called a leak because the overall computer system ends up with less available free memory when an application is causing a memory leak. It might be more appropriate to call this issue a memory consumption flaw. Depending on the speed of the memory leak, the issue may not be noticeable in typical circumstances (such as when an application is closed after a few minutes of use) or may quickly degenerate, causing system failures. Programmers should focus on properly managing memory and releasing memory allocations once they are no longer needed. Otherwise, end users and system administrators should monitor system performance for software memory leaks and then elect to discontinue the use of offending products.

Integer overflow

An *integer overflow* is the state that occurs when a mathematical operation attempts to create a numeric value that is too large to be contained or represented by the allocated storage space or memory structure. For example, an 8-bit value can only hold the numbers 0 to 255. If an additional number is added to the maximum value, an integer overflow occurs. Often, the number value resets or rolls over to 0, similar to the way a vehicle odometer rolls over. However, in other cases, the result *saturates*, meaning the maximum value is retained. Thus, the result is another form of error (missing or lost information). In yet other cases, the rollover results in a negative number. If the programming logic assumes that a number will always be positive, then when a negative number is processed, it could have security-breaching results. Programmers need to understand the numeric limitations of their code and the platform for which they're developing. There are coding techniques programmers should adopt in order to test for integer-overflow results before an overflow can occur.

Buffer overflow

A *buffer overflow* is a memory exploitation that takes advantage of a software's lack of input length validation. By injecting larger than expected input into a system, this attack may result in the extra data "overflowing" the assigned buffer and thus overwrite memory in the following adjacent locations. Such a buffer overflow might simply cause a system freeze or execution malfunction. However, in some cases a buffer overflow can allow for the injection of shellcode (precompiled malicious code) into memory, where it may be executed with system-level privileges. This is known as a buffer overflow attack leading to arbitrary code execution. The primary defense against buffer overflow is input sanitization, specifically limiting the length of input.

Pointer dereference

*A pointer dereferenc*e is the programmatic activity of retrieving the value stored in a memory location by triggering the pulling of the memory based on its address or location as stored in a pointer (a type of variable that holds an address—that is, a memory space location). Invalid dereferencing can occur due to attempting to dereference a pointer that was not initialized (assigned a memory address), dereferencing a pointer that retrieves data to be assigned to a variable that is not configured as the same data type (binary vs. ASCII or numbers vs. text), and dereferencing a pointer that was deallocated due to a dynamic memory allocation change. If a programmer leaves in code that causes an invalid dereference, it could cause a crash of the application, cause the system to freeze, or even open vulnerabilities that can be exploited by other means (such as buffer overflow attacks).

DLL injection

DLL injection is an advanced software exploitation technique that manipulates a process's memory in order to trick it into loading additional code and thus performing operations the original author did not intend. A *DLL (dynamic link library)* is a collection of code that is designed to be loaded and used as needed by a process. Many DLLs are designed to perform common functions and thus are shared among many applications.

A DLL injection attack starts off by manipulating the memory of a live process in order to inject commands that trick the process into loading and executing the malicious DLL. This is similar to when you have a shopping list posted on the fridge that you grab as you head to the grocery store, only to discover when you return home that your neighbor broke into your apartment and added beer, chips, and dip to the list, and now that you have returned home with those items, he has declared that you are now hosting a poker party for him and his questionable friends.

System sprawl/undocumented assets

System sprawl or *server sprawl* is the situation where numerous underutilized servers are operating in your organization's server room. These servers are taking up space, consuming electricity, and placing demands on other resources, but their provided workload or productivity does not justify their presence. This can occur if an organization purchases cheap lower-end hardware in bulk instead of selecting optimal equipment for specific use cases. Consolidation of software onto optimized hardware designed to manage resource consumption with little resource contention is a response to system sprawl. It is also likely that using virtualization to run several guest OSs on a single hardware server can reduce the inefficiencies as well.

Undocumented assets are another form of wasted resources and lost opportunity. Without clear knowledge of what equipment is present in an organization, it is impossible to plan for future growth, adopt proper security measures, or track down offending elements. Every asset used in a business task should be identified and tracked. This will help maximize the production potential of existing hardware while minimizing the purchase of unneeded, superfluous, or ill-suited equipment.

Architecture/design weaknesses

Architecture or design flaws are distinct from coding bugs. Bugs are mistakes in the authoring of the software code, often typographical errors or the use of the wrong function. *Design flaws* are mistakes in the overall concept, theory, implementation, or structure of an application. Design flaws may exist because of a misunderstanding of the problem that was intended to be solved, not understanding the requirements of the solution, violating common or good practice design principles, or failing to account for security measures during initial conception.

Architecture or design flaws often fall into three main categories: omission flaws, commission flaws, or realization flaws. Omission flaws occur when a security requirement is overlooked or a key element of the development process is ignored. Commission flaws occur when poor decisions were made about how to perform certain actions, resolve problems, or strike a balance between performance and security. Realization flaws occur when the correct design concept was selected but it was improperly implemented in code, thus causing a problem that is indistinguishable from an omission or commission flaw.

For a more thorough explanation of design flaws, see Common Architecture Weakness Enumeration (CAWE) at `http://blog.ieeesoftware.org/2016/04/common-architecture-weakness.html` and visit the CWE (common weakness enumeration) catalog hosted by MITRE at `http://cwe.mitre.org/`.

New threats/zero day

New threats are being developed by hackers on a nearly daily basis. It is an essential part of security management to be aware of new threats. Performing daily research can assist you in remaining up to date. To see or track some of the concerns, security professionals can review various websites for threat information. Some useful sites of this ilk are `https://www.exploit-db.com`, `https://cve.mitre.org`, `https://nvd.nist.gov/`, and `https://www.us-cert.gov`.

By keeping an eye on the security trends and alerts related to new zero-day compromises, you will be better prepared to respond to incidents as well as defend against them.

Thousands of new virus and malware variations are crafted and released daily. Fortunately, only a small portion of these are significant threats. However, that is not cause to overlook the severity of the damage that even a single malicious code infection could cause.

Everyone needs a current antivirus scanner. This scanner should be configured to download updates daily on an automatic schedule. The system should be scanned fully at least once per week. The system's activity should be monitored in real time. Although antivirus software has advanced significantly in the last few years, it is still not a substitute for avoiding risky activity and controlling user behavior.

Please see the earlier coverage of zero-day issues in the sections "Application/Service Attacks" and "Zero Day."

Improper certificate and key management

Key management is always a concern when cryptography is involved. Most of the failures of a cryptosystem are based on improper key management rather than on the algorithms. Good key selection is based on the quality and availability of random numbers. Most mobile devices must rely locally on poor random number–producing mechanisms or access more robust random number generators (RNGs) over a wireless link. Once keys are created, they need to be stored in such a way as to minimize exposure to loss or compromise. The best option for key storage is usually removable hardware or the use of a trusted platform module (TPM), but these are rarely available on mobile phones and tablets.

For more discussion on key management in general, see the section "Key Escrow" in Chapter 6, "Cryptography and PKI."

Exam Essentials

Understand race conditions. Time-of-check-to-time-of-use (TOCTTOU) attacks are often called race conditions because the attacker is racing with the legitimate process to replace the object before it is used. Another form of race condition attack occurs when two processes are running concurrently and one process is designed to finish first, but the attack alters the processing to change the order of completion.

Comprehend end-of-life systems. End-of-life systems are those that are no longer receiving updates and support from their vendors. If an organization continues to use an end-of-life system, then the risk of compromise is high because no future exploitation will ever be patched or fixed.

Understand embedded systems. An embedded system is any form of computing component added to an existing mechanical or electrical system for the purpose of providing automation and/or monitoring.

Realize that there may be a lack of vendor support. Any system, whether hardware or software, will become more insecure over time once it lacks vendor support. The lack of vendor support can be due to end-of-life dropping of support, but it can also be a "feature" of the product all along, where the vendor does not provide any improvement, support, or patching/upgrading of the product after the initial sale.

Understand improper input handling. Many forms of exploitation are caused by the lack of input sanitization or validation. Only with proper input handling can software exploitation be reduced or eliminated.

Know proper input handling. There are three main forms of input filtering that should be adopted by every programmer and included in every code they author: check for length, filter for known malware patterns, and escape metacharacters.

Understand improper error handling. Improper error handling may allow for the leaking of essential information to attackers or enable attackers to force a system into an insecure state. If error messages are not handled properly, they may disclose details about a flaw or weakness that will enable an attacker to fine-tune their exploit.

Understand misconfiguration/weak configuration. When misconfigurations or weak configurations are allowed to remain while a system is in active productive use, the risk of data loss, data leakage, and overall system compromise is higher.

Know the risks of default configuration. Default configurations should never be allowed to remain on a device or within an application. The tyranny of the default is the fact that defaults are usually insecure and thus leave a system open to simple compromise.

Understand resource exhaustion. Resource exhaustion occurs when applications are allowed to operate in an unrestricted and unmonitored manner so that all available system resources are consumed in the attempt to serve the requests of valid users or in response to a DoS attack.

Understand untrained users. Untrained users are more likely to make mistakes or abuse a system's resources and capabilities.

Understand improperly configured accounts. The concept of improperly configured accounts is a violation of the principle of least privilege.

Understand vulnerable business processes. All business tasks, processes, procedures, and functions should be assessed as to their importance to the organization and their relative vulnerabilities.

Understand weak cipher suites and implementations. Many older algorithms or implementations of algorithms have known flaws, weaknesses, or means of compromise. These weaker ciphers should be avoided and disabled and replaced with stronger cipher suites with few or no issues.

Understand memory leaks. A memory leak occurs when a program fails to release memory or continues to consume more memory.

Understand integer overflow. An integer overflow is the state that occurs when a mathematical operation attempts to create a numeric value that is too large to be contained or represented by the allocated storage space or memory structure.

Understand buffer overflow. A buffer overflow is a memory exploitation that takes advantage of a software's lack of input length validation. In some cases a buffer overflow can allow for the injection of shellcode (precompiled malicious code) into memory, where it may become executed with system-level privileges.

Understand pointer dereference. Pointer dereferencing is the programmatic activity of retrieving the value stored in a memory location by triggering the pulling of the memory based on its address or location as stored in a pointer.

Understand DLL injection. DLL injection is an advanced software exploitation technique that manipulates a process's memory in order to trick it into loading additional code and thus perform operations the original author did not intend.

Comprehend system sprawl/undocumented assets. System sprawl or server sprawl is the situation where numerous underutilized servers are operating in your organization's server room. The existence of undocumented assets is a form of wasted resources and lost opportunity.

Understand architecture/design weaknesses. Architecture or design flaws are mistakes in the overall concept, theory, implementation, or structure of an application. Design flaws may exist because of a misunderstanding of the problem that was intended to be solved, not understanding the requirements of the solution, violating common or good practice design principles, or failing to account for security measures during initial conception.

Understand new threats. New threats are being developed by hackers on a nearly daily basis. It is an essential part of security management to be aware of new threats.

Understand improper certificate and key management. Most of the failures of a cryptosystem are based on improper key management rather than on the algorithms.

Review Questions

You can find the answers in the Appendix.

1. An attacker has decided to attempt to compromise your organization's network. They have already determined the ISP you are using and know your public IP addresses. They have also performed port scanning to discover your open ports. What communications technique can the hacker now use to identify the applications that are running on each open port facing the Internet?

   A. Credentialed penetration test

   B. Intrusive vulnerability scan

   C. Banner grabbing

   D. Port scanning

2. You are the security manager for a large organization. Your NIDS has reported abnormal levels of network activity and several systems have become unresponsive. While investigating the causes of these issues, you discover a rootkit on your mission-critical database server. What is the best step to take to return this system to production?

   A. Reconstitute the system.

   B. Run an antivirus tool.

   C. Install a HIDS.

   D. Apply vendor patches.

3. If user awareness is overlooked, what attack is more likely to succeed?

   A. Man-in-the-middle

   B. Reverse hash matching

   C. Physical intrusion

   D. Social engineering

4. A pirated movie-sharing service is discovered operating on company equipment. Administrators do not know who planted the service or who the users are. What technique could be used to attempt to trace the identity of the users?

   A. Typo squatting

   B. Integer overflow

   C. Watering hole attack

   D. Ransomware

5. You are the IT security manager for a retail merchant organization that is just going online with an e-commerce website. You hired several programmers to craft the code that is the backbone of your new web sales system. However, you are concerned that while the new code functions well, it might not be secure. You begin to review the code, systems design, and services architecture to track down issues and concerns. Which of the following do you hope to find in order to prevent or protect against XSS?

   A. Input validation

   B. Defensive coding

C. Allowing script input

D. Escaping metacharacters

6. What type of virus attempts to disable security features that are focused on preventing malware infection?

A. Retrovirus

B. Polymorphic

C. Companion

D. Armored

7. What does the acronym RAT stand for?

A. Random Access Token

B. Remote Authentication Testing

C. Random Authorization Trajectory

D. Remote Access Trojan

8. What form of social engineering attack focuses on stealing credentials or identity information from any potential target?

A. Phishing

B. Tailgating

C. Dumpster diving

D. Logic bomb

9. What type of service attack positions the attacker in the communication path between a client and a server?

A. Session hijacking

B. Man-in-the-middle

C. Amplification

D. Replay

10. What form of attack abuses a program's lack of length limitation on the data it receives before storing the input in memory and can lead to arbitrary code execution?

A. ARP poisoning

B. XSS

C. Domain hijacking

D. Buffer overflow

11. What is a programmatic activity that restricts or reorganizes software code without changing its externally perceived behavior or produced results?

A. Buffer overflow

B. Pass the hash

C. Refactoring

D. Shimming

12. What wireless attack is able to trick mobile device users into connecting into its man-in-the-middle style of attack by automatically appearing as if it is a trusted network that they have connected to in the past?

A. Replay

B. Evil twin

C. Bluesnarfing

D. Disassociation

13. What type of hacker hacks for a cause or purpose, knowing that they may be identified, apprehended, and prosecuted?

A. Hacktivist

B. Script kiddie

C. Nation-state hacker

D. Internal attacker

14. When an attacker selects a target, they must perform reconnaissance to learn as much as possible about the systems and their configuration before launching attacks. What is the term for the gathering of information from any publicly available resource, such as websites, social networks, discussion forums, file services, and public databases?

A. Banner grabbing

B. Port scanning

C. Open-source intelligence

D. Enumeration

15. What penetration testing or hacking term refers to the concept of continuing an intrusion after an initial compromise in order to further breach an organization by focusing on new targets that may not have been accessible initially?

A. Man-in-the-browser

B. Pivot

C. Daisy chaining

D. Shimming

16. What is the term for an attack or exploit that grants the attacker greater privileges, permissions, or access than what may have been achieved by the initial exploitation?

A. Hoax

B. Impersonation

C. Piggybacking

D. Privilege escalation

17. What type of information-gathering tactics rely on direct interaction with the target while attempting to avoid being detected as malicious?

    A. Passive reconnaissance

    B. Banner grabbing

    C. Active reconnaissance

    D. Social engineering

18. What type of test of security controls is performed with an automated vulnerability scanner that seeks to identify weaknesses while listening in on network communications?

    A. Active

    B. Passive

    C. External

    D. Noncredentialed

19. What is the term used to describe systems that are no longer receiving updates and support from their vendors?

    A. Passive

    B. Embedded

    C. End-of-life

    D. Static

20. What is present on a system for ease of installation and initial configuration in order to minimize support calls from new customers?

    A. Default configuration

    B. Resource exhaustion trigger

    C. Buffer overflow flaw

    D. Collision tool

Chapter

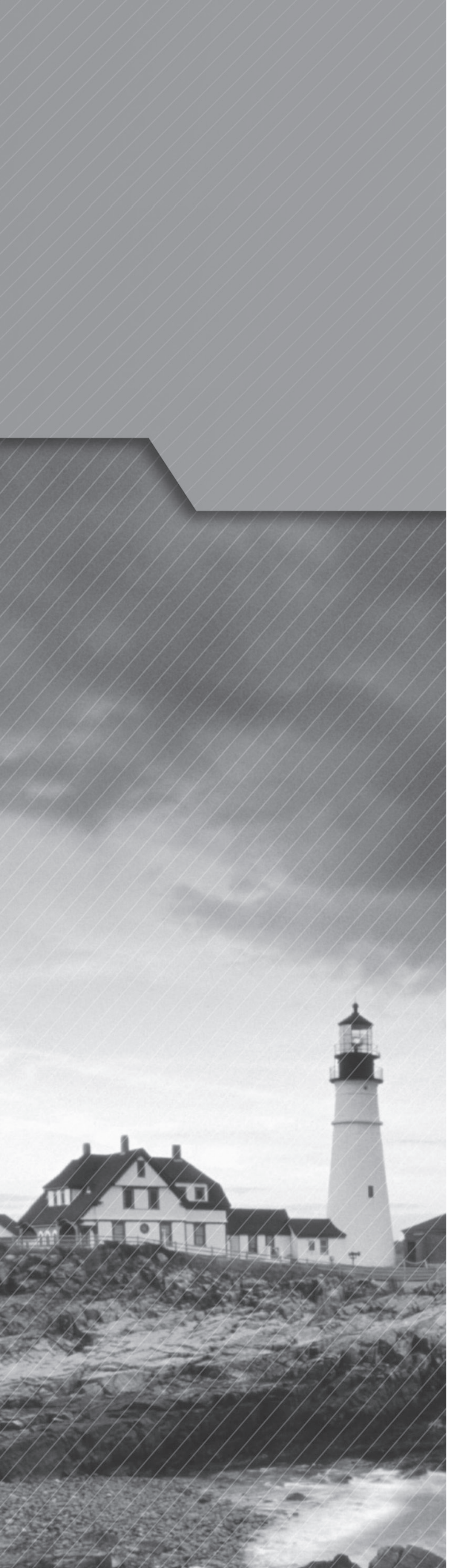
2

Technologies and Tools

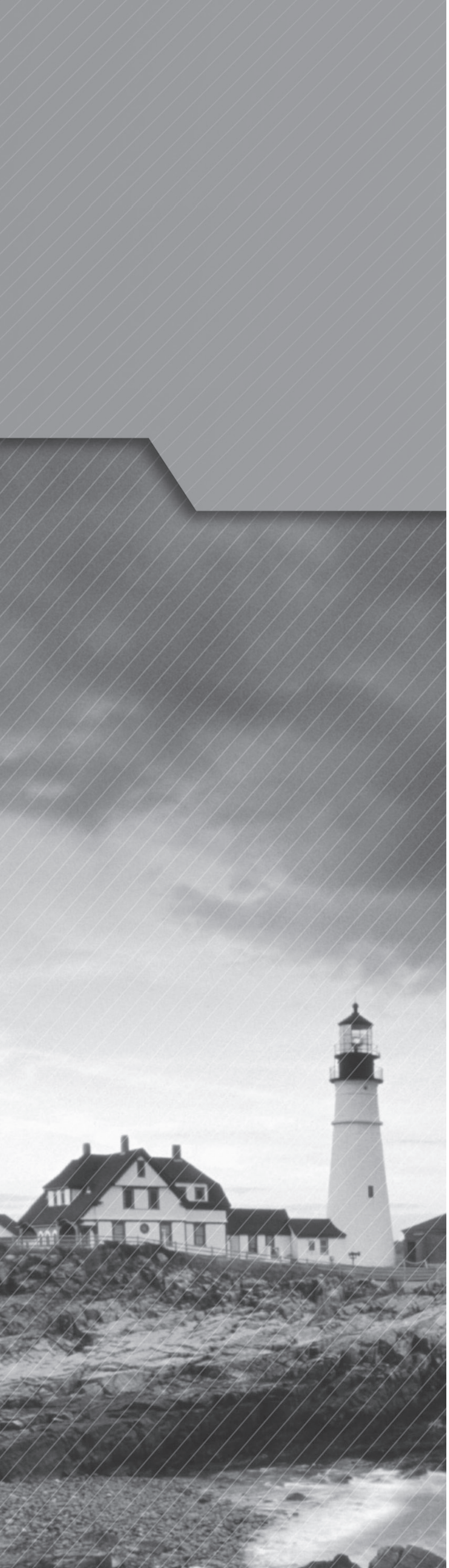
COMPTIA SECURITY+ EXAM OBJECTIVES COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:

✓ **2.1 Install and configure network components, both hardware- and software-based, to support organizational security.**

- Firewall
  - ACL
  - Application-based vs. network-based
  - Stateful vs. stateless
  - Implicit deny
- VPN concentrator
  - Remote access vs. site-to-site
  - IPSec
    - Tunnel mode
    - Transport mode
    - AH
    - ESP
  - Split tunnel vs. full tunnel
  - TLS
  - Always-on VPN
- NIPS/NIDS
  - Signature-based
  - Heuristic/behavioral
  - Anomaly
  - Inline vs. passive
  - In-band vs. out-of-band



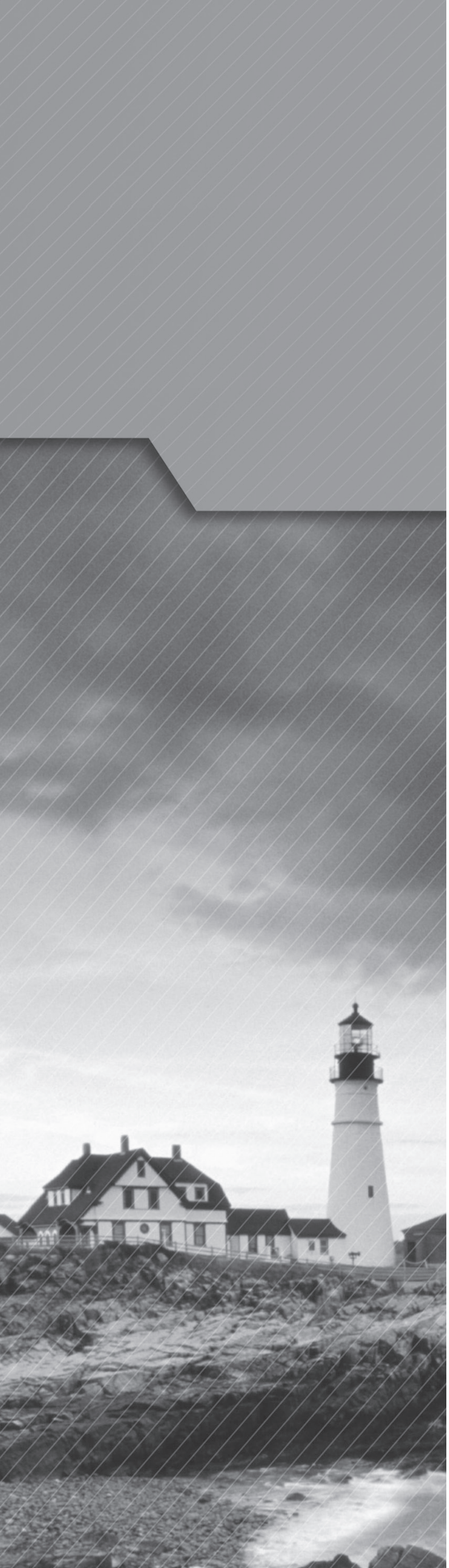
  - Rules
  - Analytics
    - False positive
    - False negative
- Router
  - ACLs
  - Antispoofing
- Switch
  - Port security
  - Layer 2 vs. Layer 3
  - Loop prevention
  - Flood guard
- Proxy
  - Forward and reverse proxy
  - Transparent
  - Application/multipurpose
- Load balancer
  - Scheduling
    - Affinity
    - Round-robin
  - Active-passive
  - Active-active
  - Virtual IPs
- Access point
  - SSID
  - MAC filtering
  - Signal strength
  - Band selection/width
  - Antenna types and placement
  - Fat vs. thin
  - Controller-based vs. standalone



- SIEM
  - Aggregation
  - Correlation
  - Automated alerting and triggers
  - Time synchronization
  - Event deduplication
  - Logs/WORM
- DLP
  - USB blocking
  - Cloud-based
  - Email
- NAC
  - Dissolvable vs. permanent
  - Host health checks
  - Agent vs. agentless
- Mail gateway
  - Spam filter
  - DLP
  - Encryption
- Bridge
- SSL/TLS accelerators
- SSL decryptors
- Media gateway
- Hardware security module

✓ 2.2 Given a scenario, use appropriate software tools to assess the security posture of an organization.

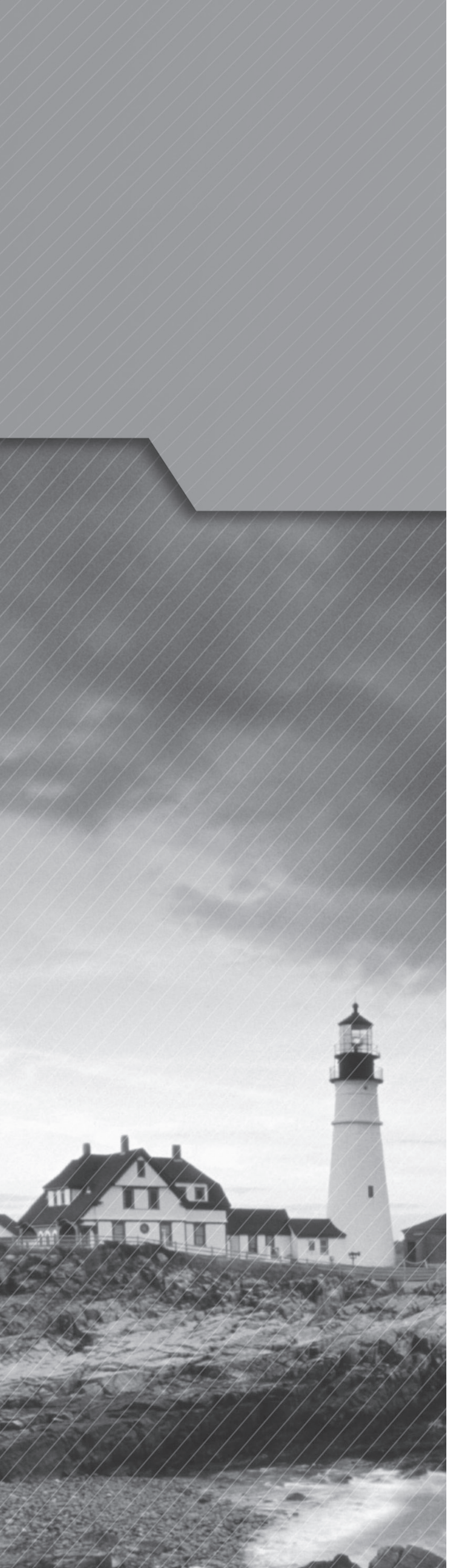
- Protocol analyzer
- Network scanners
  - Rogue system detection
  - Network mapping
- Wireless scanners/cracker
- Password cracker



- Vulnerability scanner
- Configuration compliance scanner
- Exploitation frameworks
- Data sanitization tools
- Steganography tools
- Honeypot
- Backup utilities
- Banner grabbing
- Passive vs. active
- Command line tools
  - ping
  - netstat
  - tracert
  - nslookup/dig
  - arp
  - ipconfig/ip/ifconfig
  - tcpdump
  - nmap
  - netcat

✓ 2.3 Given a scenario, troubleshoot common security issues.

- Unencrypted credentials/clear text
- Logs and events anomalies
- Permission issues
- Access violations
- Certificate issues
- Data exfiltration
- Misconfigured devices
  - Firewall
  - Content filter
  - Access points



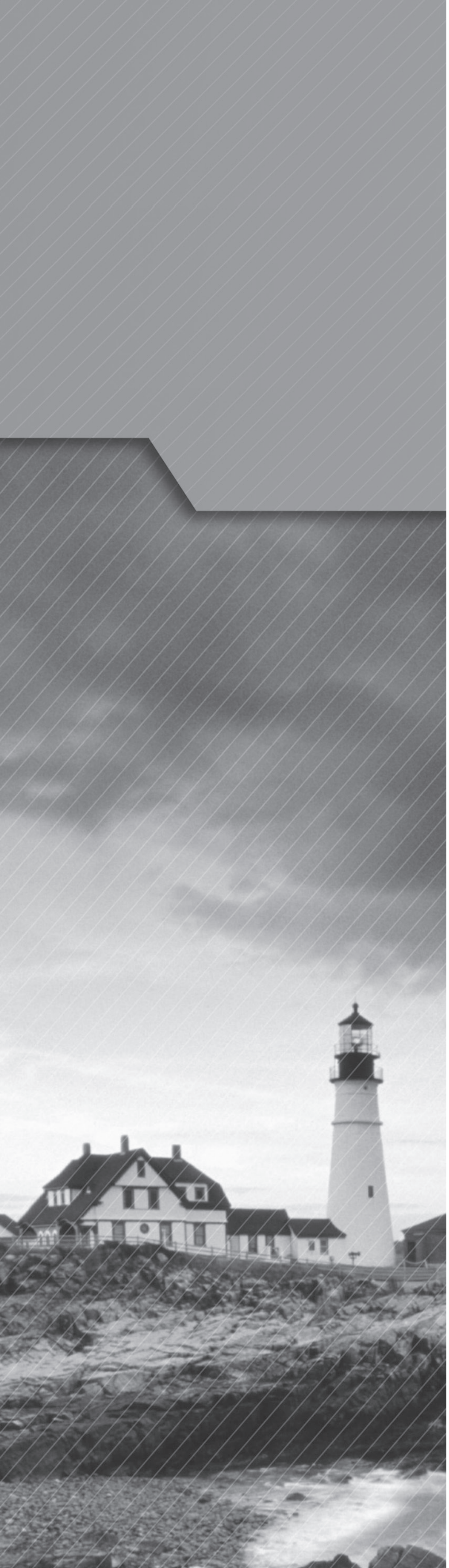
- Weak security configurations
- Personnel issues
  - Policy violation
  - Insider threat
  - Social engineering
  - Social media
  - Personal email
- Unauthorized software
- Baseline deviation
- License compliance violation (availability/integrity)
- Asset management
- Authentication issues

✓ **2.4 Given a scenario, analyze and interpret output from security technologies.**

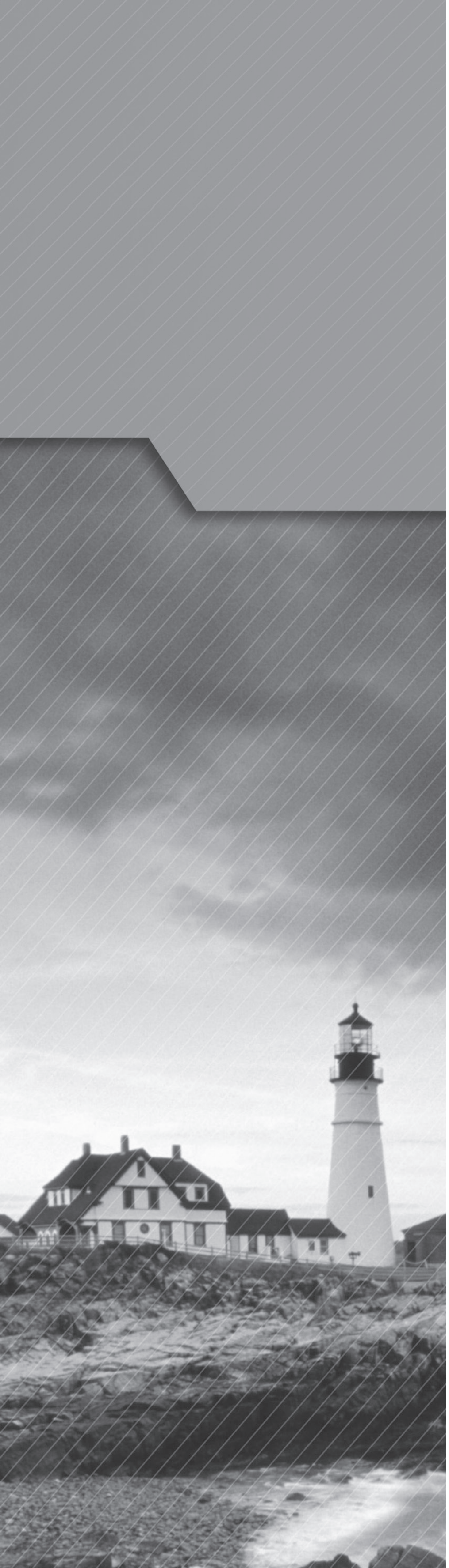
- HIDS/HIPS
- Antivirus
- File integrity check
- Host-based firewall
- Application whitelisting
- Removable media control
- Advanced malware tools
- Patch management tools
- UTM
- DLP
- Data execution prevention
- Web application firewall

✓ **2.5 Given a scenario, deploy mobile devices securely.**

- Connection methods
  - Cellular
  - WiFi
  - SATCOM
  - Bluetooth



  - NFC
  - ANT
  - Infrared
  - USB
- Mobile device management concepts
  - Application management
  - Content management
  - Remote wipe
  - Geofencing
  - Geolocation
  - Screen locks
  - Push notification services
  - Passwords and pins
  - Biometrics
  - Context-aware authentication
  - Containerization
  - Storage segmentation
  - Full device encryption
- Enforcement and monitoring for:
  - Third-party app stores
  - Rooting/jailbreaking
  - Sideloading
  - Custom firmware
  - Carrier unlocking
  - Firmware OTA updates
  - Camera use
  - SMS/MMS
  - External media
  - USB OTG
  - Recording microphone
  - GPS tagging
  - WiFi direct/ad hoc



  - Tethering
  - Payment methods
- Deployment models
  - BYOD
  - COPE
  - CYOD
  - Corporate-owned
  - VDI

✓ 2.6 Given a scenario, implement secure protocols.

- Protocols
  - DNSSEC
  - SSH
  - S/MIME
  - SRTP
  - LDAPS
  - FTPS
  - SFTP
  - SNMPv3
  - SSL/TLS
  - HTTPS
  - Secure POP/IMAP
- Use cases
  - Voice and video
  - Time synchronization
  - Email and web
  - File transfer
  - Directory services
  - Remote access
  - Domain name resolution
  - Routing and switching
  - Network address allocation
  - Subscription services

The Security+ exam will test your knowledge of security technology and tools both for the home office and in corporate environments. To pass the test and be effective in implementing security, you need to understand the concepts and terminology related to network and system security as detailed in this chapter. You will also need to be familiar with when and why to use various tools and technologies, given a scenario.

2.1 Install and configure network components, both hardware- and software-based, to support organizational security.

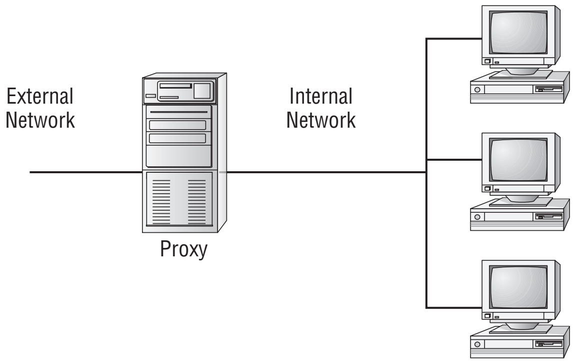
Security involves the implementation of hardware and software solutions designed to provide protection for the confidentiality, integrity, and availability of the IT infrastructure. There are a wide range of products you should be familiar with for the Security+ exam; this section reviews them.

Firewall

A *firewall* is a hardware or software component designed to protect one network from another (see Figure 2.1). Firewalls are deployed between areas of high and low trust, like a private network and a public network (such as the Internet), or between two networks that belong to the same organization but are used by different departments. Firewalls provide protection by controlling traffic entering and/or leaving a network.

Firewalls manage traffic using *filters*. A filter is just a rule or set of rules. Firewall filters can also be known as access control lists (ACLs) or tuples (collections of related data items). Firewalls usually have lots of filters, which are defined in a priority order. If a packet meets the identification criteria of a rule, the action of that rule is applied. If a packet doesn't meet the criteria of a rule, no action from that rule is applied, and the next rule is checked.

FIGURE 2.1 A proxy firewall blocking network access from external networks



The action of a filter rule is commonly *allow, deny,* or *log*. Some firewalls use a first-match mechanism when applying rules. Allow rules enable the packet to continue toward its destination. Deny rules block the packet from going any further (effectively discarding it). When first-match is used, the first rule that applies to the packet is followed, but no other rules are considered. Thus, rules need to be placed in a priority order. Filter lists are created with the most specific and detailed rules first, followed by successively more general rules, until a final default universal rule is reached, which often specifies a denial. The log action records information about the packet into a log file. However, some firewalls (such as iptables) allow for multiple rule matches. Or they perform a consolidated or accumulated result to apply that is an amalgamation of all the rules that apply to the packet.

Firewalls following a first-match approach should have a final written rule of deny all. So any packet that does not otherwise meet a previous allow or deny rule will be discarded. Those following an amalgamation approach will not have a written deny rule; instead they have an implicit deny stance that any packet not specifically allowed will be discarded.

Therefore, if a packet fails to meet the criteria of an allow rule, the discard option will be applied. This way, only packets meeting the custom-defined allow filters or rules are allowed to cross the security barrier. In other words, firewalls are deny-by-default or implicit deny security tools.

There are four basic types of firewalls:

Packet Filter *A packet filter firewall* filters traffic based on basic identification items found in a network packet's header. This includes source and destination IP address, port numbers, and protocols used. Packet-filtering firewalls operate at the Network layer (Layer 3) and the Transport layer (Layer 4) of the Open Systems Interconnection (OSI) model.

Circuit-Level Gateway A *circuit-level gateway firewall* filters traffic by filtering on the connection between an internal trusted host and an external untrusted host. This monitoring occurs at either the Network layer (Layer 3) or the Session layer (Layer 5) of the OSI model. This type of firewall ensures that the packets involved in establishing and maintaining the circuit (a virtual circuit or session) are valid and used in the proper manner. Once a circuit-level gateway allows a connection, no further filtering on that communication is performed.

Application-Level Gateway An *application-level gateway firewall* filters traffic based on user access, group membership, the application or service used, or even the type of resources being transmitted. This type of firewall operates at the Application layer (Layer 7) of the OSI model. Such a firewall can be called a *proxy*. Application-level gateways are focused on the aspects of a specific appliance and protocol combination as well as the content of the conversation. An application-aware firewall provides filtering services for specific applications.

Stateful Inspection Firewall A *stateful inspection firewall* is aware that any valid outbound communication (especially related to TCP) will trigger a corresponding response or reply from the external entity. Thus, this type of firewall automatically creates a response rule for the response on the fly. But that rule exists only as long as the conversation is taking place. This is unlike the static packet filter firewall, which requires that both an outbound rule and an inbound rule be defined at all times.

Additionally, stateful inspection firewalls can retain knowledge of previous packets in a conversation in order to detect unwanted or malicious traffic that isn't noticeable or detectable when evaluating only individual packets. This is known as *context analysis* or *contextual analysis*.

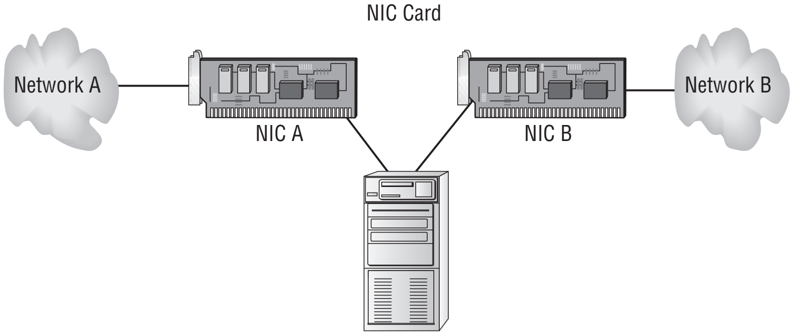
A stateful inspection firewall may also perform deep packet inspection, which is the analysis of the payload or content of a packet. This could even include virtual reassembly of the original (or final) payload through the recombination of the payloads across multiple packets.

Thus, a stateful inspection firewall can make more intelligent and complex filtering decisions based on higher-order information. One of the key functions of this type of firewall is to ensure that each packet is part of an established Transmission Control Protocol (TCP) communication session. All rogue, or unassociated, packets are blocked.

The first step in effectively designing, deploying, and implementing a firewall is to design or develop a *firewall policy*: a security policy that focuses on the purposes, uses, functions, and security of the firewalls in an organization. This policy clearly defines how the firewall should filter traffic and the types of traffic that should be blocked or allowed.

Most firewalls are deployed with at least two network interfaces. Such firewalls are called *dual-homed* (see Figure 2.2) or *multihomed* (for two or more NICs). Dual- or multihomed firewalls provide a clear security distinction between one network and another; thus, packets must successfully pass the filters of a firewall in order to move from one network to another. In this manner, firewalls provide strong and reliable security.

FIGURE 2.2 A dual-homed firewall segregating two networks from each other



Make sure routing or IP forwarding is disabled in operating system.

Some firewalls with three or more network interfaces can manage access to multiple networks simultaneously. A common deployment uses one of these additional network interfaces to connect to a demilitarized zone (DMZ). The DMZ hosts publicly accessible servers, such as the Web or File Transfer Protocol (FTP). The firewall provides secured but public access to the DMZ, but it prevents unauthorized access to the private network. If such a multihomed firewall is compromised, only the systems in the DMZ are directly threatened or exposed.

When a port is opened in a firewall to allow a virtual private network (VPN) connection to take place, keep in mind that all encrypted data will pass through the firewall without being inspected or filtered. Unless the firewall can see the unencrypted data, perhaps as a VPN termination point, it can't inspect the communication and, therefore, can't provide filtering security.

An *ingress filter* is a traffic filter on packets coming into a secured area from outside (that is, inbound communications). An *egress filter* is a traffic filter on packets leaving a secured area toward the outside (outbound communications). Common ingress and egress filters perform the following functions:

- Blocking inbound packets claiming to have an internal source address
- Blocking outbound packets claiming to have an external source address
- Blocking packets with source or destination addresses listed on a block list (a list of known malicious IPs)
- Blocking packets that have source or destination addresses from the local area network (LAN) but haven't been officially assigned to a host

Additional firewall rules are added to these common spoofing-prevention and common-sense protections based on the needs of the organization and the design of the infrastructure.

ACL

Access control list (ACL) is a term that is normally used in the context of object permissions and privileges, but it is also used in relation to firewalls. The rules or filters on a firewall can be referred to as ACLs. Most cloud solutions or hosted systems use an ACL-based approach rather than traditional firewalling.

Application-based vs. network-based

An *application firewall* is a device, server add-on, virtual service, or system filter that defines a strict set of communication rules for a service and all users. It's intended to be an application-specific server-side firewall to prevent application-specific protocol and payload attacks. A *web application firewall* is an example of an application firewall. It's intended to be an application-specific firewall to prevent cross-site scripting, SQL injection, and other web application attacks.

A *network firewall* is a hardware device, typically called an *appliance*, designed for general network filtering. A network firewall is designed to provide broad protection for an entire network.

Both of these types of firewalls are important and may be relevant in many situations. Every network needs a network firewall. Many application servers need an application firewall. However, the use of an application firewall generally doesn't negate the need for a network firewall. You should use both types in a series to complement each other, rather than seeing them as competitive solutions.

Stateful vs. stateless

A *stateless firewall* analyzes packets on an individual basis against the filtering ACLs. The context of the communication (that is, any previous packets) is not used to make an allow or deny decision on the current packet. A *stateful firewall* monitors the state or session of the communication; it evaluates previous packets and potentially other communications and conditions when making an allow or deny decision for the current packet. A stateful firewall considers the context of the communication, whereas a stateless firewall does not.

Implicit deny

Implicit deny is the default security stance and ensures that any communication not specifically granted access or privileges is denied access by default. A default-deny statement is implicit in the permission-management system and doesn't need to be specifically defined. This may differ on firewall and router access rule sets when operating on a first-match apply basis. In this situation, a default deny-all rule is included as the last rule. Implicit deny is the default response when an explicit allow or deny isn't present. In a firewall context where all rules are considered as a collective against traffic, no explicit deny-all rule is defined, since the traffic will be blocked by the implicit deny if it does not meet an allow rule.

VPN concentrator

A *virtual private network (VPN)* is a communication *tunnel* between two entities across an intermediary network. In most cases, the intermediary network is an untrusted network, such as the Internet, and therefore the communication tunnel is also encrypted. VPNs are discussed further in Chapter 3, "Architecture and Design," in the section "Tunneling/VPN."

A *VPN concentrator* is a dedicated hardware device designed to support a large number of simultaneous VPN connections, often hundreds or thousands. It provides high availability, high scalability, and high performance for secure VPN connections. With the ever-increasing need for secured communications, VPNs have become an essential tool for securing communications traversing private networks and the Internet.

A VPN concentrator can also be called a VPN server, a VPN gateway, a VPN firewall, a VPN remote access server (RAS), a VPN device, a VPN proxy, or a VPN appliance.

Remote access vs. site-to-site

A *remote access VPN* is a variant of the *site-to-site VPN*. The difference is that with a remote access VPN one endpoint is the single entity of a remote user that connects into an organizational network. A remote access VPN is also known as a host-to-site VPN. A site-to-site VPN is a VPN between two organizational networks. Both remote access VPNs and site-to-site VPNs are known as tunnel mode VPNs, and they offer link encryption. This means they provide encryption only when the traffic is inside the tunnel itself. In both types of tunnel mode VPN, on the side of the VPN that is a site or an organizational network, traffic exiting the tunnel will go back to plain text to traverse the private network.

The other main type of VPN is the transport mode VPN. It provides end-to-end encryption and can be described as a host-to-host VPN. In this type of VPN, all traffic is fully encrypted between the endpoints, but those endpoints are only individual systems, not organizational networks.

IPSec

Internet Protocol Security (IPSec) is a VPN protocol for IPv4 derived from the security features of IPv6. You can use IPSec in dial-up or network-to-network connections. When it's employed over dial-up, it usually functions as the encryption protocol in an L2TP link. IPSec by itself is more suitable for network-to-network connections across normal LAN connections, high-speed WAN links, and the Internet.

IPSec isn't a single protocol but rather a collection of protocols. Two of the primary protocols of IPSec are *Authentication Header (AH)* and *Encapsulating Security Payload (ESP)*.

IPSec provides for encryption security using symmetric cryptography. This means communication partners use shared secret keys to encrypt and decrypt traffic over the IPSec VPN link. One of the mechanisms used by IPSec to manage cryptography is *Internet Key Exchange (IKE)*; it ensures the secure exchange of secret keys between communication partners in order to establish the encrypted VPN tunnel. IKE is composed of three elements: Oakley, SKEME, and ISAKMP.

Oakley is a key generation and exchange protocol similar to Diffie-Hellman (see Chapter 6, "Cryptography and PKI"). *Secure Key Exchange MEchanism (SKEME)* is a means to exchange keys securely.

Internet Security Association and Key Management Protocol (ISAKMP) is used to organize and manage the encryption keys that have been generated and exchanged by Oakley and SKEME. A security association is the agreed-on method of authentication and encryption used by two entities. Without a common method of authentication, a VPN link can't be established. So, ISAKMP is used to negotiate and provide authenticated keying material (a common method of authentication) for security associations in a secured manner. The four major functional components of ISAKMP are as follows:

- Authentication of communications peers
- Threat mitigation
- Security association creation and management
- Cryptographic key establishment and management

IPSec is a standard architecture set forth by the Internet Engineering Task Force (IETF) for setting up a secure channel to exchange information between two entities. The two entities could be two systems, two routers, two gateways, or any combination of entities. Although generally used to connect two networks, IPSec can be used to connect individual computers, such as a server and a workstation or a pair of workstations (sender and receiver, perhaps). IPSec doesn't dictate all implementation details but is an open, modular framework that allows many manufacturers and software developers to develop IPSec solutions that work well with products from other vendors.

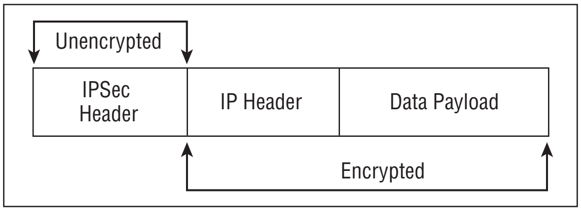
IPSec uses public-key cryptography to provide encryption, access control, nonrepudiation, and message authentication, all using Internet protocols. The primary use of IPSec is for VPNs, so IPSec operates in either transport or tunnel mode. IPSec is commonly paired with L2TP as L2TP/IPSec.

The IPSec protocol provides a complete infrastructure for secured network communications. It has gained widespread acceptance and is now offered in a number of commercial operating systems out of the box.

Tunnel mode

IPSec can operate in two modes: *tunnel mode* and *transport mode*. In tunnel mode, IPSec provides encryption protection for both the payload and message header by encapsulating the entire original LAN protocol packet and adding its own temporary IPSec header (see Figure 2.3).

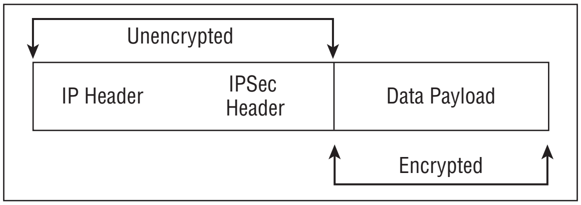
FIGURE 2.3 IPSec's encryption of a packet in tunnel mode



Transport mode

In transport mode, IPSec provides encryption protection for just the payload and leaves the original message header intact (see Figure 2.4). You should use tunnel mode when you're connecting over an untrusted network.

FIGURE 2.4 IPSec's encryption of a packet in transport mode



AH

The *Authentication Header (AH)* provides assurances of message integrity and nonrepudiation. AH also provides authentication and access control and prevents replay attacks.

ESP

The *Encapsulating Security Payload (ESP)* provides confidentiality and integrity of packet contents. It provides encryption and limited authentication, and prevents replay attacks.

NOTE ESP also provides some limited authentication, but not to the degree of the AH. Although ESP is sometimes used without AH, it's rare to see AH used without ESP.

Split tunnel vs. full tunnel

A *split tunnel* is a VPN configuration that allows a VPN-connected system to access both the organizational network over the VPN and the Internet directly at the same time. The split tunnel thus simultaneously grants an open connection to the Internet and to the organizational network.

A *full tunnel* is a VPN configuration in which all of the client's traffic is sent to the organizational network over the VPN link, and then any Internet-destined traffic is routed out of the organizational network's proxy or firewall interface to the Internet. A full tunnel ensures that all traffic is filtered and managed by the organizational network's security infrastructure.

TLS

Secure Sockets Layer (SSL) was developed by Netscape to provide client-server encryption for web traffic. HTTPS uses port 443 to negotiate encrypted communications sessions between web servers and browser clients. Although SSL originated as a standard for Netscape browsers, Microsoft also adopted it as a security standard for its popular Internet Explorer browser. The incorporation of SSL into both of these products made it the de facto Internet standard.

SSL relies on the exchange of server digital certificates to negotiate RSA encryption/decryption parameters between the browser and the web server. SSL's goal is to create secure communications channels that remain open for an entire web browsing session.

SSL relies on a combination of symmetric and asymmetric cryptography. When a user accesses a website, the browser retrieves the web server's certificate and extracts the server's public key from it. The browser then creates a random symmetric key, uses the server's public key to encrypt it, and sends the encrypted symmetric key to the server. The server then decrypts the symmetric key using its own private key, and the two systems exchange all future messages using the symmetric encryption key. This approach allows SSL to use the advanced functionality of asymmetric cryptography while encrypting and decrypting the vast majority of the data exchanged using the faster symmetric algorithm.

SSL forms the basis for a newer security standard, the *Transport Layer Security (TLS)* protocol, specified in RFC 2246. TLS is quickly surpassing SSL in popularity. SSL and TLS both support server authentication (mandatory) and client authentication (optional).

TLS has replaced SSL due to exploitable flaws discovered in SSL. Since November 2016, most browsers disable SSL by default and leave only TLS active. For further discussion on TLS (and SSL), see the section "SSL/TLS" later in this chapter.

Always-on VPN

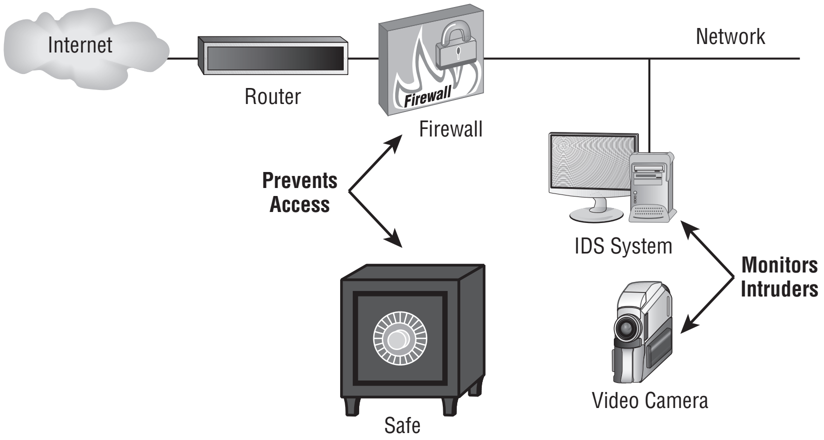
An always-on VPN is one that attempts to auto-connect to the VPN service every time a network link becomes active. Some always-on VPNs can be configured to engage only when an Internet link is established rather than a local network link or only when a WiFi link is established rather than a wired link. Due to the risks of using an open public Internet link, whether wireless or wired, having an always-on VPN will ensure that a secure connection is established every time when attempting to use online resources.

NIPS/NIDS

Intrusion detection is an important security capability. *Intrusion detection systems (IDSs)* are designed to detect the presence of an unauthorized intruder or unwanted activity. Generally, IDSs are used in a passive manner; they detect problems rather than eliminate them. Intrusion prevention systems (IPSs) are designed to detect attempts to gain unauthorized access and stop the attempts from becoming successful. IPSs are generally used more actively; they interact and interfere with communications of unwanted entities.

IDS and IPS security solutions are considered complementary to firewalls (see Figure 2.5). IDS and IPS systems can be two independent solutions, or one combined product.

FIGURE 2.5 An IDS and a firewall working together to secure a network

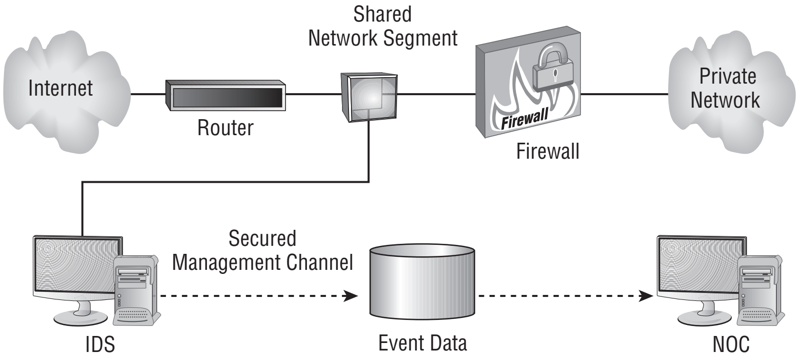


There are two primary types of IDS/IPS: network (NIDS/NIPS) and host (HIDS/HIPS). A NIDS can detect malicious activity that occurs within the network (it doesn't cross the firewall) and activity that is able to pass through the firewall. A HIDS can detect malicious activity that occurs on a single host.

The most common problem with an IDS/IPS, excluding misconfiguration, is the occurrence of false positives. A *false positive* occurs when legitimate traffic or user activity is mistaken for intruder activity.

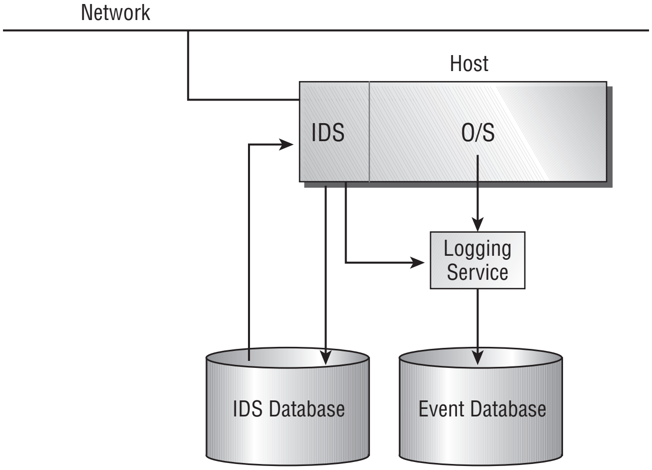
A *network-based IDS/IPS* watches network traffic in real time (see Figure 2.6). It monitors network traffic patterns, scans packet header information, and may examine the contents of packets to detect security violations or attacks. A network-based IDS/IPS is reliable for detecting network-focused attacks, such as bandwidth-based denial-of-service (DoS) attacks. A NIDS/NIPS monitors network traffic, looking for any abnormal or malicious content. Based on what it detects and how it's configured, it can respond in real time to notify administrators (a passive NIDS response) or interfere with any attack or intrusion attempts before they're successful against the network or any internal targets (an active NIPS reaction). Most commonly, the response to malicious packets is to drop them, thus rendering their payloads ineffective. However, NIDS/NIPS can also be configured to disconnect sessions and reconfigure firewalls, as well as initiate alerts, expand monitoring, and quarantine intruders in honeypots or padded cells. A *honeypot* is a fictitious environment designed to fool attackers and intruders and lure them away from the private secured network (see the section "Honeypot" later in this chapter). A *padded cell* is a containment area that is activated only when an intrusion is detected.

FIGURE 2.6 A network-based IDS/IPS placement in a network determines what data will be analyzed.



A *host-based IDS/IPS* watches the audit trails and log files of a host system (see Figure 2.7). This type of IDS/IPS is limited to the auditing and logging capabilities of the host system (which includes the OS and installed applications and services). A host-based IDS/IPS can detect problems only if sufficient information is captured by the host's auditing capabilities. It's reliable for detecting attacks directed against a host, whether they originate from an external source or are perpetrated by a user locally logged into the host.

FIGURE 2.7 A host-based IDS/IPS interacting with the OS



Common examples of HIDSs are antivirus software, antispyware scanners, and security anomaly detectors.

An IDS/IPS with active detection and response is designed to take the quickest action to reduce the potential damage caused by an intruder (see Figure 2.8). This response may include shutting down the server or just the affected service or disconnecting suspicious connections (see Figure 2.9 and Figure 2.10).

FIGURE 2.8 The components of an IDS/IPS working together to provide network monitoring

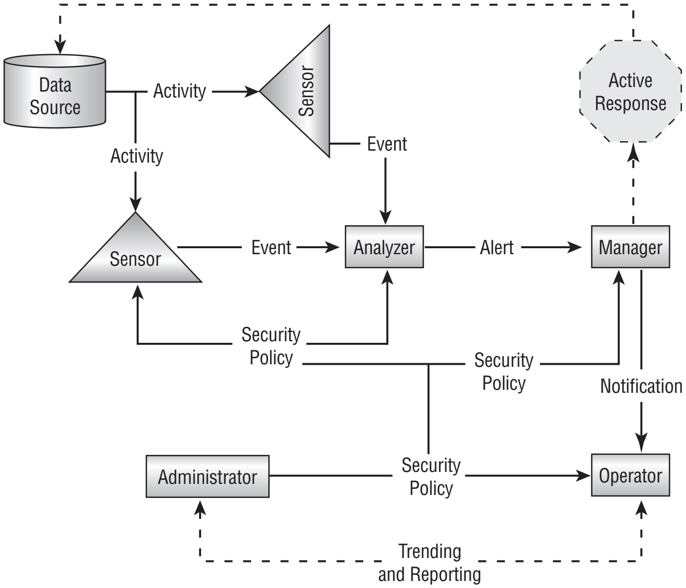


FIGURE 2.9 IDS/IPS instructing TCP to reset all connections

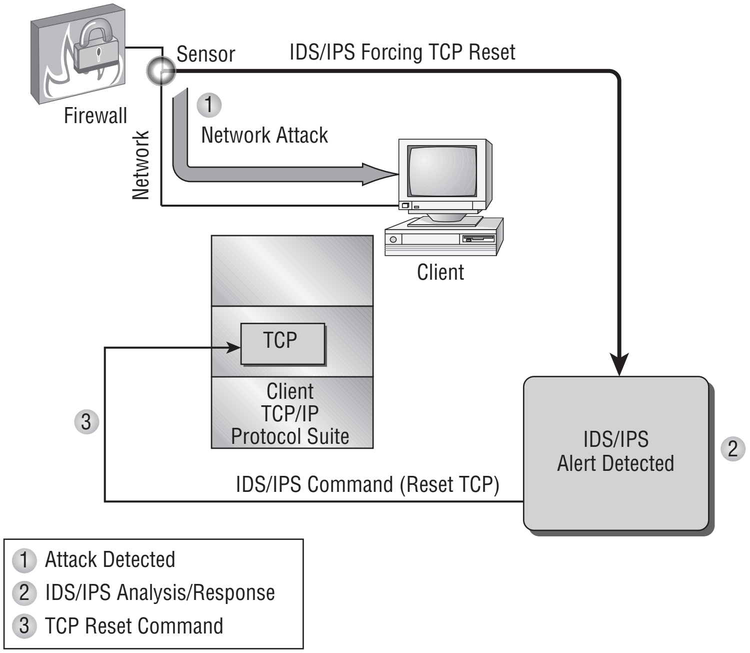
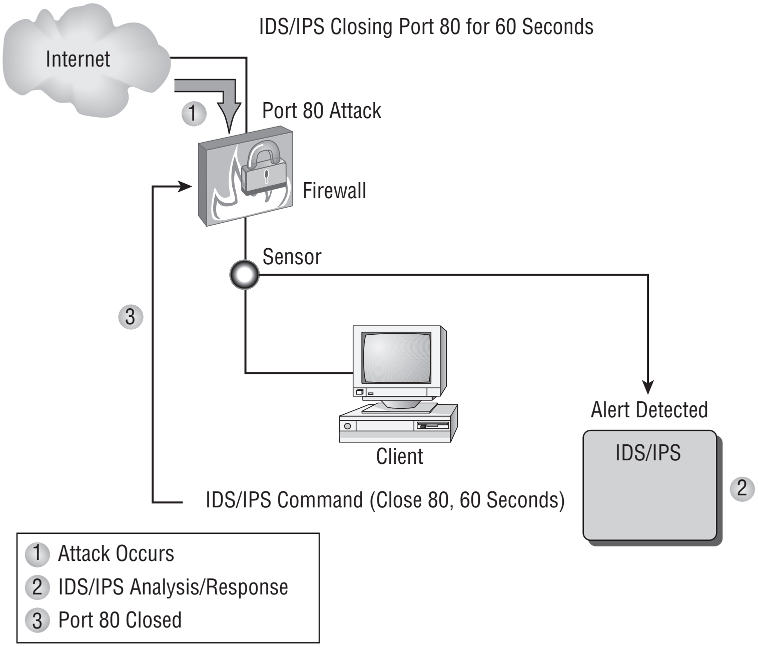

Firewall
Sensor
IDS/IPS Forcing TCP Reset
1
Network Attack
Network
Client
TCP
Client TCP/IP Protocol Suite
3
IDS/IPS Alert Detected
2
IDS/IPS Command (Reset TCP)
1 Attack Detected
2 IDS/IPS Analysis/Response
3 TCP Reset Command

FIGURE 2.10 IDS/IPS instructing the firewall to close port 80 for 60 seconds to thwart an Internet Information Services (IIS) attack



IDS/IPS Closing Port 80 for 60 Seconds
Internet
1
Port 80 Attack
Firewall
Sensor
3
Client
Alert Detected
IDS/IPS
2
IDS/IPS Command (Close 80, 60 Seconds)
1 Attack Occurs
2 IDS/IPS Analysis/Response
3 Port 80 Closed

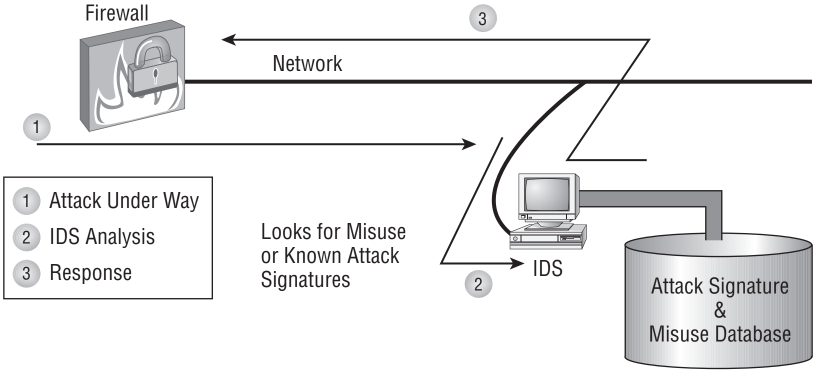
An IDS/IPS with passive detection and response takes no direct action against the intruder; instead, it may increase the amount of data being audited and recorded and notify administrators about the intrusion. An IDS/IPS is good at detecting DoS attacks; exploiting bugs, flaws, or hidden features; and port scanning. It isn't reliable for detecting spoofed email. Passive IDS/IPS responses are usually unseen by intruders and don't directly affect the violating activity, whereas active IDS/IPS responses are seen by intruders because they directly interrupt and interfere with violating activities.

Many tools are used for monitoring and overseeing the activities within the complex infrastructures of networks and systems, such as performance monitors, system monitors, IDSs, protocol analyzers, and so on. Many of these tools also support one or more methodologies of monitoring. These methodologies determine how a tool knows when a measurement or event is normal, abnormal, benign, malicious, and so on.

Signature-based

Signature-based detection (see Figure 2.11) compares event patterns against known attack patterns (signatures) stored in the IDS/IPS database. The strength of a signature-based system is that it can quickly and accurately detect any event from its database of signatures. However, the primary weakness of a signature-based system is that it's unable to detect new and unknown activities or events. Thus, new zero-day attacks are unseen by a signature-based system. As new attacks are discovered and the pattern database is improved, the deployed signature-based tools need to have their local databases updated.

FIGURE 2.11 A signature-detection IDS/IPS in action



Heuristic/behavioral

A behavior-based monitoring or detection method relies on the establishment of a baseline or a definition of normal and benign. Behavior-based monitoring is a form of anomaly detection, but instead of using a database of rules to determine anomalies, a recording of real production activity is used. Once this baseline is established, the monitoring tool is able to detect activities that vary from that standard of normal. The strength of a behavior-based system is that it can detect any type of change or difference, including previously unseen and unknown issues such as zero-day intrusion attacks. However, a weakness of behavior-based attacks monitoring is that defining what is normal is a very difficult challenge. Determining what is benign or malicious when nonstandard activity occurs is also not easy or often possible with an automated behavior-based tool.

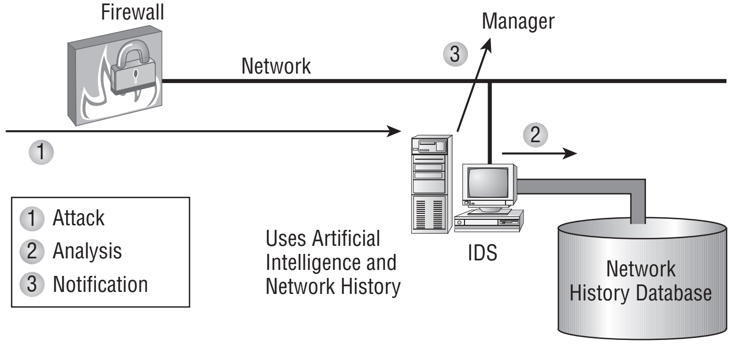
Heuristic analysis functions by comparing suspicious or new programs against known examples of malware. This can be accomplished in many ways. One method is to run the suspicious program in a sandbox or virtual machine and watch its activities. If it exhibits activities similar enough to those of known malicious code, then it's classified as malicious.

Another method is to decompile the new program and look for known malicious subroutines or duplicates of code sections from known malware. This method is known as *static analysis*.

Anomaly

Anomaly-based detection (see Figure 2.12) watches the ongoing activity in the environment and looks for abnormal occurrences. An anomaly-based monitoring or detection method relies on definitions of all valid forms of activity. This database of known valid activity allows the tool to detect any and all anomalies. Anomaly-based detection is commonly used for protocols. Because all the valid and legal forms of a protocol are known and can be defined, any variations from those known valid constructions are seen as anomalies. Anomaly detection is very effective at stopping abnormal events. However, traffic or events falling within normal values doesn't necessarily mean the contents of that event or traffic aren't malicious in nature.

FIGURE 2.12 An anomaly-detection IDS/IPS using expert system technology to evaluate risks



Inline vs. passive

An *inline IPS* has two interfaces and all traffic must traverse through the IPS. Traffic enters either interface, is evaluated by the IPS analysis engine, and then exits the other interface on its way to the destination. This technique enables the IPS to stop or block abusive traffic.

A *passive IDS or IPS* uses a promiscuous mode NIC to eavesdrop on network communication. A passive IDS is often deployed off the SPAN (Switched Port Analyzer) port on a switch, where it receives a copy of every communication occurring across the switch. Sometimes this port is called the auditing port, IDS port, or mirror port. This type of monitoring allows only for reactive responses to discovered problems, rather than proactive responses.

In-band vs. out-of-band

An *in-band IDS* is configured to monitor and filter both the pre-connect activities and the post-connect activities of each session. Pre-connect activities can include authentication as well as verifying compliance with minimal security requirements before a session is allowed to be established. Post-connect activities include traffic monitoring, content filtering, identity-based access controls, and ongoing verification that the connection that was granted is still valid and should be allowed to continue.

An *out-of-band IDS* is configured to perform pre-connect activity monitoring, but then not be involved with any post-connect activity monitoring.

Rules

IDS rules are used to define what is considered benign allowed traffic versus malicious/suspicious/abnormal disallowed traffic. Often anomaly detection is implemented through the defining of rules. Any event that meets a rule defining valid benign activity is allowed to occur, whereas any event that meets a rule defining suspicious activity is blocked, logged, or flagged for more detailed analysis.

Snort is an example of a rules-based NIDS. Snort uses a simple rule description language to craft rules that can have significant benefit in protecting a network from identified unwanted traffic. Snort rules are comprised of a rule action (such as alert, log, or pass), a protocol (such as TCP or UDP), a source IP address, a source port number, a direction, a destination IP address, a destination port number, and finally what to do if a packet matches the rule's requirements. Please see online Snort rules documentation (`http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node27.html`) for more specific and non–exam-related information.

Analytics

IDS analytics is the review, investigation, and understanding of the results from an IDS. An IDS will consider an event or traffic either benign or malicious and in turn will either trigger an alarm/response or not. This allows for four possible result states from an IDS; the first two are true positive and the latter two are true negative. The most desired is the true negative, in which only benign events are occurring and no alarms are sounding. The second most desired state is the true positive, when malicious events are occurring and the alarm is sounding.

False positive

The third state occurs when a benign event triggers an alarm; this is a false positive. It is undesired because it wastes response resources, time, and attention and, if it occurs repeatedly, could cause a disbelief in the quality of the IDS system, thus resulting in future alarms being ignored. Care must be taken to ensure that response teams always respect the IDS alarms even after repeated false positives. Please see the "False Positive" section in Chapter 1 for more discussion on this topic.

False negative

The fourth state occurs when a malicious event does not trigger an alarm—a false negative. This is the worst possible state, because harm is actively occurring and it remains unknown. Thus, there is no response to the harm, so it continues to occur. This type of state cannot be directly addressed, since it is the unknown unknown. The only response is to continue to improve and tune the detection of the IDS in hopes that it will minimize the occurrences of this state.

Router

A *router* (see Figure 2.13) is used to connect several network segments. Routers enable traffic from one network segment to traverse into another network segment (see Figure 2.14). However, the traffic must pass through the router's filters in order to make the transition. A router with access control lists (ACLs) can be considered a simple firewall.

FIGURE 2.13 A router connecting two networks, such as a LAN to a WAN

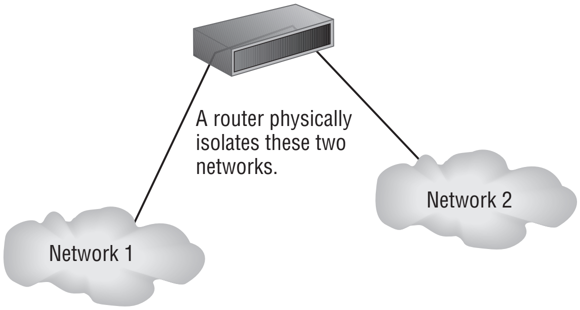
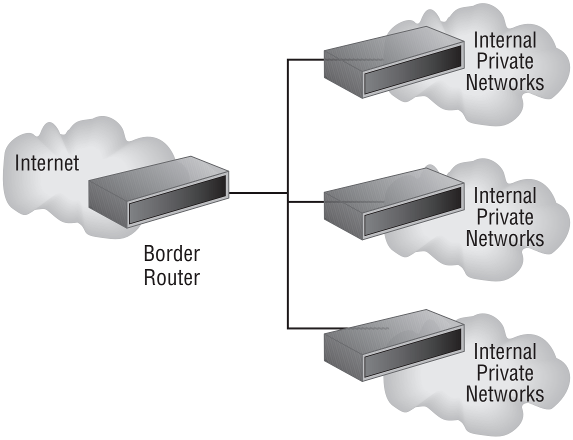


FIGURE 2.14 A corporate network implementing routers for segmentation and security


Internal Private Networks
Internet
Border Router
Internal Private Networks
Internal Private Networks

Routers direct traffic based on a routing table and grant or deny access using ACLs, such as rules or filters. The routing table informs the router which direction to transmit a received packet based on the best-known pathway (route).

Routers can manage traffic for both inbound and outbound communications. The router's collection of information about the network is stored in a routing table. The routing table can be managed statically or by dynamic routing protocols.

A secure router configuration is one in which malicious or unauthorized route changes are prevented. This can be done using a few simple settings:

- Set the router's administrator password to something unique and secret.
- Set the router to ignore all Internet Control Message Protocol (ICMP) type 5 redirect messages.

- Use a secure routing protocol that requires authentication and data encryption to exchange route data.
- Preconfigure the IP addresses of other trusted routers with which routing data can be exchanged.
- Configure management interfaces to operate only on internal interfaces, use secure protocols, and potentially be accessible only on dedicated networks.

With these simple precautions, you can improve the security of a router's configuration. For more advanced router configuration tips, see the CIS benchmarks and configuration guides at `https://www.cisecurity.org/cis-benchmarks`.

ACLs

Access control lists (ACLs) are used to define who is allowed or denied permission to perform a specified activity or action. ACLs are commonly associated with object access but also apply to communications. In many cases, firewalls, routers, and even switches can use ACLs as a method of security management. In fact, the rules of these devices can be called ACLs or filters. It's all roughly the same concept. As with many other security control mechanisms, ACLs deny by default and allow by exception. If a user/IP/device is present in an ACL (specifically an access control entry [ACE], which is a single line in an ACL), then the specified action or activity is either allowed or denied.

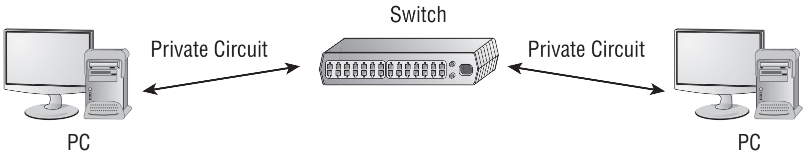
Antispoofing

Antispoofing rules or ACLs can be added to a router in order to drop communications that are obviously false. These rules simply indicate that if the source address of a packet does not exist within the subnet from which the traffic is originating, then it is spoofed and should be dropped. Antispoofing rules are most effective on routers managing traffic crossing the internal-to-external boundary. Any outbound traffic that does not originate from an internal IP address and any inbound traffic that does not originate from an external IP address can be dropped because it is spoofed.

Switch

A *switch* (see Figure 2.15) is a networking device used to connect many other devices together and potentially implement traffic management on their communications.

FIGURE 2.15 Switching between two systems



Switches generally link individual hosts, but they can also be used to link networks together. Switches receive signals in one port and transmit them out the port where the intended recipient is connected. Switches accomplish this traffic-control task by maintaining a table of the media access control (MAC) addresses of devices located off each switch port. The switch examines the source MAC address of each packet it receives and records the MAC address and the related port in its CAM (Content Addressable Memory) table. Thus the CAM table is dynamic and is constantly being updated. The switch analyzes the header of each packet it receives to determine the destination MAC address and then transmits each packet only out the port where that MAC address is known to reside. If a MAC address is encountered that isn't known (it's not in the CAM table), the unknown destination packet is transmitted out all ports except the ingress port.

Switches are good defenses against *sniffing attacks* from random clients within a network. Sniffing is the act of capturing network traffic for analysis; sniffing attacks occur when sniffing is done without authorization. Switches transmit messages only on those specific network links between the source and destination systems.

A sniffer can only intercept traffic that happens to be transmitted on the segment it's connected to. Thus, using switches instead of hubs is a great defense against sniffing.

However, there are logical and physical attacks to overcome this protection. If a hacker can gain physical access, he can connect to the audit/monitor/mirror ports or reconfigure the switch to obtain full access to all the data it sees. If a hacker has only logical (network) access to the switch, then a MAC flooding attack can overload a switch's CAM table in order to drop valid MAC addresses and populate the table with invalid MAC addresses.

When this attack is successful, the switch may revert to a hub-like fault-tolerance mode, transmitting data out all ports instead of only a single port. This type of attack is often called *active sniffing*, because the hacker has to attack the switch (or sometimes hosts on the network with Address Resolution Protocol [ARP] flooding attacks) to obtain access to all network traffic. Advanced switches have native IDS-like detection and defense features to prevent MAC flooding attacks from being successful.

Port security

Port security in IT can mean several things. It can mean the physical control of all connection points, such as RJ-45 wall jacks or device ports (such as those on a switch, router, or patch panel), so that no unauthorized users or unauthorized devices can attempt to connect into an open port. This can be accomplished by locking down the wiring closet and server vaults and then disconnecting the workstation run from the patch panel (or punch-down block) that leads to a room's wall jack. Any unneeded or unused wall jacks can (and should) be physically disabled in this manner. Another option is to use a smart patch panel that can monitor the MAC address of any device connected to each wall port across a building and detect not just when a new device is connected to an empty port, but also when a valid device is disconnected or replaced by an invalid device.

Another meaning for *port security* is the management of TCP and User Datagram Protocol (UDP) ports. If a service is active and assigned to a port, then that port is open. All the other 65,535 ports (of TCP or UDP) are closed if a service isn't actively using them.

Hackers can detect the presence of active services by performing a port scan. Firewalls, IDSs, IPSs, and other security tools can detect this activity and either block it or send back false/misleading information. This measure is a type of port security that makes port scanning less effective.

Port security can also refer to *port knocking,* a security system in which all ports on a system appear closed. However, if the client sends packets to a specific set of ports in a certain order, a bit like a secret knock, then the desired service port becomes open and allows the client software to connect to the service. Port knocking doesn't prevent a hacker from eavesdropping on the port-knocking sequence and repeating it, but it does defeat the use of port scanners that randomly target Internet-facing systems.

Port security can also refer to the need to authenticate to a port before being allowed to communicate through or across the port. This may be implemented on a switch, router, smart patch panel, or even a wireless network. This concept is often referred to as IEEE 802.1x. See the section "IEEE 802.1x" in Chapter 4, "Identity and Access Management."

Layer 2 vs. Layer 3

A switch is normally a Layer 2 device since it manages traffic based on the MAC address. A switch can create VLANs to segment off communications to only members of the same VLAN. But when cross-VLAN communications are needed, a Layer 3 switch can be used; it provides routing between its own VLANs. Thus, a Layer 3 switch includes some router capabilities that it can offer to its VLANs.

Loop prevention

A *loop* in networking terms is a transmission pathway that repeats itself. It's the network equivalent of going around in a circle. The problem with looping in a network environment is that it wastes resources, specifically network throughput capacity. Loops can occur at Layer 2 and at Layer 3, typically related to Ethernet and IP, respectively.

Ethernet looping is resolved using the Spanning Tree Protocol (STP) on the bridges and switches of a network. STP learns all available paths and then makes traffic-management decisions that prevent looping pathways. Effectively, STP erects transmission blockades to prevent loops from being created.

IP resolves looping using a different technique. Instead of preventing the use of pathways that cause looping, IP controls the distance a packet travels before it's discarded. So, instead of preventing loops, IP minimizes the amount of looping before packets are terminated. This is controlled using a countdown timer in the IP header, specifically the time-to-live (TTL) value. The TTL is set at an initial OS-specific default (for example, the Windows TTL is now 128 but was 32 in some older versions, whereas the TTL on Linux systems ranges from 64 to 255), and then each router decrements the TTL as it retransmits the IP packet. When a router receives a TTL that has a value of 1, that router stops forwarding the packet toward its destination and sends it back to the source address with an error message ("ICMP Type 11—Timeout Exceeded").

Flood guard

A *flood guard* is a defense against flooding or massive-traffic DoS attacks. The purpose of a flood guard is to detect flooding activity and then automatically begin blocking it. This prevents this type of malicious traffic from entering a private network.

Floods can be used in a variety of attack variations. One form of flood attack can be used to overload a switch in order to break VLAN segmentation. This is accomplished by flooding a switch with Ethernet frames with randomized source MAC addresses. The switch will attempt to add each newly discovered source MAC address to its CAM table. Once the CAM table is full, older entries will be dropped in order to make room for new entries. Once the CAM is full of only false addresses, the switch is unable to properly forward traffic, so it reverts to flooding mode, where it acts like a hub or a multiport repeater and sends each received Ethernet frame out of every port. This effectively violates VLAN segmentation, which relies on the switch only sending frames out the ports that are members of the correct VLAN. Flood guards are often effective at minimizing this risk, whether the attack origin is internal or external.

The formal command `floodguard` in the Cisco IOS can be used to enable or disable Flood Defender, the Cisco solution that addresses flooding attacks.

Proxy

A *proxy server* is a variation of an application firewall or circuit-level firewall. A proxy server is used as a proxy or middleman between clients and servers. Often a proxy serves as a barrier against external threats to internal clients. This is usually performed by utilizing network address translation (NAT). NAT hides the Internet Protocol (IP) configuration of internal clients and substitutes the IP configuration of the proxy server's own public external network interface card (NIC) in outbound requests. This effectively prevents external hosts from learning the internal configuration of the network. A proxy server typically has the default setting to ignore all external queries and only manage communications that are responses from previous queries. In addition to features such as NAT, proxy servers can provide caching and site or content filtering.

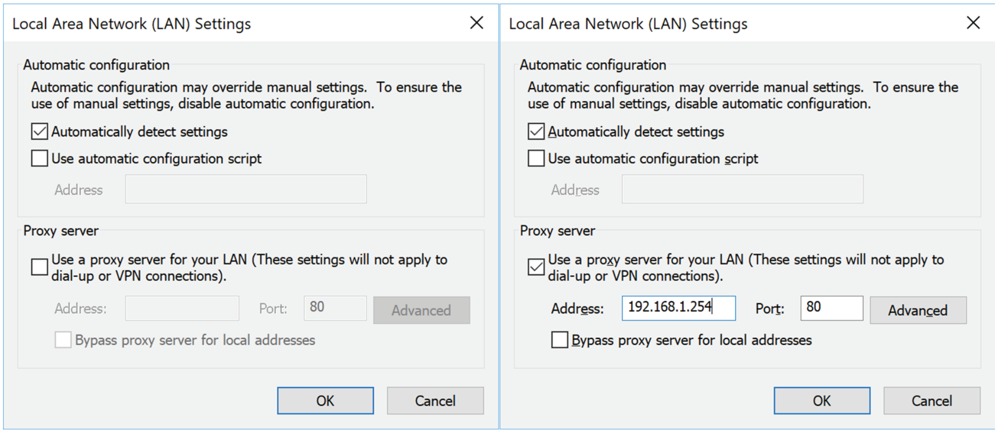
Forward and reverse proxy

A forward proxy is a standard proxy that acts as an intermediary or middleman for queries of external resources. A forward proxy handles queries from internal clients when accessing outside services. A *reverse proxy* provides the opposite function; it handles inbound requests from external systems to internally located services. A reverse proxy is similar to the functions of port forwarding and static NAT.

Transparent

If a client is not configured (Figure 2.16, left) to send queries directly to a proxy but the network routes outbound traffic to a proxy anyway, then a *transparent proxy* is in use. A *nontransparent proxy* is in use when a client is configured (Figure 2.16, right) to send outbound queries directly to a proxy.

FIGURE 2.16 The configuration dialog boxes for a transparent (left) vs. a nontransparent (right) proxy



Local Area Network (LAN) Settings

Automatic configuration

Automatic configuration may override manual settings. To ensure the use of manual settings, disable automatic configuration.

☑ Automatically detect settings

□ Use automatic configuration script

Address:

Proxy server

□ Use a proxy server for your LAN (These settings will not apply to dial-up or VPN connections).

Address: Port: 80

□ Bypass proxy server for local addresses

Local Area Network (LAN) Settings

Automatic configuration

Automatic configuration may override manual settings. To ensure the use of manual settings, disable automatic configuration.

☑ Automatically detect settings

□ Use automatic configuration script

Address:

Proxy server

☑ Use a proxy server for your LAN (These settings will not apply to dial-up or VPN connections).

Address: 192.168.1.254 Port: 80

□ Bypass proxy server for local addresses

Application/multipurpose

An application proxy or an application-specific proxy is a proxy server configured to handle the communications for a single application and its related protocols. For example, a web application proxy is designed to manage only HTTP- and HTTPS-based communications. An application proxy operates at Layer 7, where it is able to handle the payloads of a specific application and related application-layer protocols.

A multipurpose proxy is not limited to a single application or set of protocols but can provide proxy functions for any application and protocol. A multipurpose proxy operates at Layers 3 and 4, where it manages communications based on IP address and/or port number.

Load balancer

A *load balancer* is used to spread or distribute network traffic load across several network links or network devices. The purpose of load balancing is to obtain more optimal infrastructure utilization, minimize response time, maximize throughput, reduce overloading, and eliminate bottlenecks. Although load balancing can be used in a variety of situations, a common implementation is spreading a load across multiple members of a server farm or cluster. A load balancer might use a variety of techniques to perform load distribution, as described in Table 2.1.

TABLE 2.1 Common load-balancing techniques

Technique	Description
Random choice	Each packet or connection is assigned a destination randomly.
Round robin	Each packet or connection is assigned the next destination in order, such as 1, 2, 3, 4, 5, 1, 2, 3, 4, 5, and so on.
Load monitoring	Each packet or connection is assigned a destination based on the current load or capacity of the targets. The device/path with the lowest current load receives the next packet or connection.
Preferencing	Each packet or connection is assigned a destination based on a subjective preference or known capacity difference. For example, suppose system 1 can handle twice the capacity of systems 2 and 3; in this case, preferencing would look like 1, 2, 1, 3, 1, 2, 1, 3, 1, and so on.

Load balancing can be either a software service or a hardware appliance. Load balancing can also incorporate many other features, depending on the protocol or application, including caching, Secure Sockets Layer (SSL) offloading, compression, buffering, error checking, filtering, and even firewall and IDS capabilities.

Scheduling

Scheduling or load balancing methods are the means by which a load balancer distributes the work, requests, or loads among the devices behind it. Scheduling can be very basic, such as round-robin, or highly advanced and sophisticated, such as monitoring devices' reported loads, response times, active sessions, and other aspects of performance in order to maintain optimal workload distribution.

Affinity

Affinity is a configured preference for a client request to be sent to a specific server within the cluster or device cloud managed by the load balancer. Affinity is implemented using information gathered from layers below the Application layer (Layer 7) to preference a client request to a specific server. Persistence is when Application layer information is used to associate a client with a specific server. Although persistence is more accurate in ensuring that the desired server handles a specific client, implementation of persistence is not always possible, so affinity is used instead. Persistence ensures that a request is handled by a specific server, but affinity is an attempt to cause a server to handle a specific request.

Round-robin

Round-robin is one of the basic forms of load balancing, in which each next request or load is handed to the next server in line. For example, if there are four servers, then the requests are distributed in order to server 1, then server 2, then server 3, then server 4, then again to server 1. This pattern is the same way you pass out cards to play games like poker or Go Fish.

Active-passive

An active-passive system is a form of load balancing that keeps some pathways or system in an unused dormant state during normal operations. If one of the active elements fails, then a passive element is brought online and takes over the workload for the failed element. This technique is used when the level of throughput or workload needs to be consistent between normal states and failure states.

Active-active

An active-active system is a form of load balancing that uses all available pathways or systems during normal operations. In the event of a failure of one or more of the pathways, the remaining active pathways must support the full load that was previously handled by all. This technique is used when the traffic levels or workload during normal operations need to be maximized, but reduced capacity will be tolerated during times of failure.

Virtual IPs

Virtual IP addresses are sometimes used in load balancing; an IP address is perceived by clients and even assigned to a domain name, but the IP address is not actually assigned to a physical machine. Instead, as communications are received at the IP address, they are distributed in a load-balancing schedule to the actual systems operating on some other set of IP addresses.

Access point

Wireless networking has become common on both corporate and home networks. Properly managing wireless networking for reliable access as well as security isn't always easy. This section examines various wireless security issues.

Wireless cells are the areas in a physical environment where a wireless device can connect to a wireless access point. Wireless cells can leak outside the secured environment and allow intruders easy access to the wireless network. You should adjust the strength of the wireless access point (WAP) to maximize authorized user access and minimize intruder access. Doing so may require unique placement of WAPs, shielding, and noise transmission.

802.11 is the IEEE standard for wireless network communications. Various versions (technically called *amendments*) of the standard have been implemented in wireless networking hardware, including 802.11a, 802.11b, 802.11g, and 802.11n. *802.11x* is sometimes used to collectively refer to all of these specific implementations as a group; however, 802.11 is preferred because 802.11x is easily confused with 802.1x, which is an authentication technology independent of wireless. Each version or amendment of the 802.11 standard has offered slightly better throughput: 2 Mbps, 11 Mbps, 54 Mbps, and 200+ Mbps, respectively, as described in Table 2.2. The 802.11 standard also defines Wired Equivalent Privacy (WEP), which provides eavesdropping protection for wireless communications.

TABLE 2.2 802.11 wireless networking amendments

Amendment	Speed	Frequency
802.11	2 Mbps	2.4 GHz
802.11a	54 Mbps	5 GHz
802.11b	11 Mbps	2.4 GHz
802.11g	54 Mbps	2.4 GHz
802.11n	200+ Mbps	2.4 GHz or 5 GHz
802.11ac	1.3 Gbps	5 GHz
802.11ad	7 Gbps	60 GHz

Wireless networking has made networking more versatile than ever before. Workstations and portable systems are no longer tied to a cable but can roam freely around an office or environment—anywhere within the signal range of the deployed WAPs. However, this freedom comes at the cost of additional vulnerabilities. Wireless networks are subject to the same threats and risks as any cabled network, plus there are the additional issues of distance eavesdropping and packet sniffing as well as new forms of DoS and intrusion.

When you're deploying wireless networks, you should deploy WAPs configured to use *infrastructure mode* rather than *ad hoc mode*. Ad hoc mode means that any two wireless networking devices, including two wireless network interface cards (NICs), can communicate without a centralized control authority. Infrastructure mode means that a WAP is required, wireless NICs on systems can't interact directly, and the restrictions of the WAP for wireless network access are enforced.

Infrastructure mode includes several variations, including stand-alone, wired extension, enterprise extended, and bridge. A *stand-alone* mode infrastructure occurs when there is a WAP connecting wireless clients to each other but not to any wired resources. The WAP serves as a wireless hub exclusively. A *wired extension* mode infrastructure occurs when the WAP acts as a connection point to link the wireless clients to the wired network. An *enterprise extended* mode infrastructure occurs when multiple WAPs are used to connect a large physical area to the same wired network. Each WAP uses the same extended service set identifier (ESSID) so that clients can roam the area while maintaining network connectivity, even if their wireless NICs change associations from one WAP to another. A *bridge* mode infrastructure occurs when a wireless connection is used to link two wired networks. This often uses dedicated wireless bridges and is used when wired bridges are inconvenient, such as when linking networks between floors or buildings.

NOTE

The term SSID (which stands for service set identifier) is typically misused to indicate the name of a wireless network. Technically there are two types of SSIDs: extended service set identifier (ESSID) and basic service set identifier (BSSID). An ESSID is a wireless network in which a wireless base station or WAP is used (that is, infrastructure mode). A BSSID is a wireless network that is in ad hoc or peer-to-peer mode (that is, when a base station or WAP isn’t used). However, when operating in infrastructure mode, the BSSID is the MAC address of the base station hosting the ESSID in order to differentiate multiple base stations supporting a single extended wireless network.

Wireless Channels

There are many topics within wireless networking that I’m not addressing due to space limitations and because they’re not covered on the exam. For instance, you may want to learn more about wireless channels. Within the assigned frequency of the wireless signal are subdivisions of that frequency known as channels. Think of channels as lanes on the same highway. In the United States, there are 11 channels; in Europe, there are 13; and in Japan, there are 17. The differences stem from local laws regulating frequency management (think international versions of the United States’ Federal Communications Commission).

Wireless communications take place between a client and WAP over a single channel. However, when two or more WAPs are relatively close to each other physically, signals on one channel can interfere with signals on another channel. One way to avoid this is to set the channels of physically close WAPs as differently as possible to minimize channel-overlap interference. For example, if a building has four WAPs arranged in a line along the length of the building, the channel settings could be 1, 11, 1, and 11. But if the building is square, and a WAP is in each corner, the channel settings may need to be 1, 4, 8, and 11.

Think of the signal within a single channel as being like a wide-load truck in a lane on the highway. The wide-load truck is using part of each lane on either side of it, thus making passing the truck in those lanes dangerous. Likewise, wireless signals in adjacent channels will interfere with each other.

Data Emanations

Data emanation is the transmission of data across electromagnetic signals. Almost all activities within a computer or across a network are performed using some form of data emanation. However, this term is often used to focus on emanations that are unwanted or on data that is at risk due to the emanations.

Emanations occur whenever electrons move. Movement of electrons creates a magnetic field. If you can read that magnetic field, it could be re-created elsewhere in order to reproduce the electron stream. If the original electron stream was used to communicate data, then the re-created electron stream is also a re-creation of the original data. This form of electronic eavesdropping sounds like science fiction, but it's science fact. The U.S. government has been researching emanation security since the 1950s under the TEMPEST project.

Protecting against eavesdropping and data theft requires a multipronged effort. First, you must maintain physical access control over all electronic equipment. Second, where unauthorized personnel can still achieve physical access or proximity, use shielded devices and media. Third, always transmit any sensitive data using secure encryption protocols.

The IEEE 802.11 standard defines two methods that wireless clients can use to authenticate to WAPs before normal network communications can occur across the wireless link. These two methods are open system authentication (OSA) and shared key authentication (SKA). OSA means no real authentication is required. As long as a radio signal can be transmitted between the client and WAP, communications are allowed. It's also the case that wireless networks using OSA typically transmit everything in clear text, thus providing no secrecy or security. SKA means that some form of authentication must take place before network communications can occur. The 802.11 standard defines one optional technique for SKA known as Wired Equivalent Privacy (WEP). More information about wireless encryption is located the Chapter 6 section "Given a scenario, install and configure wireless security settings."

Site Surveys

A site survey is a formal assessment of wireless signal strength, quality, and interference using an RF signal detector. You perform a site survey by placing a wireless base station in a desired location and then collecting signal measurements from throughout the area. These measurements are overlaid onto a blueprint of the building to determine whether sufficient signal is present where needed while minimizing signals outside of the desired location. If the base station is adjusted, then the site survey should be repeated. The goal of a site survey is to maximize performance in the desired areas (such as within a home or office) while minimizing ease of access in external areas.

SSID

Wireless networks are assigned an SSID (either BSSID or ESSID) to differentiate one wireless network from another. If multiple base stations or WAPs are involved in the same wireless network, an ESSID is defined. The SSID is similar to the name of a workgroup. If a wireless client knows the SSID, it can configure its wireless NIC to communicate with the associated WAP. Knowledge of the SSID doesn't always grant entry, though, because the WAP can use numerous security features to block unwanted access. SSIDs are defined by default by vendors, and because these default SSIDs are well known, standard security practice dictates that the SSID should be changed to something unique before deployment.

The SSID is broadcast by the WAP via a special transmission called a *beacon frame*. This allows any wireless NIC within range to see the wireless network and make connecting as simple as possible. However, this default broadcasting of the SSID should be disabled to keep the wireless network secret. Even so, attackers can still discover the SSID with a wireless sniffer, because the SSID must be used in transmissions between wireless clients and the WAP. Thus, disabling SSID broadcasting isn't a true security mechanism. Instead, use WPA2 as a reliable authentication and encryption solution rather than trying to hide the existence of the wireless network.

MAC filtering

A MAC filter is a list of authorized wireless client interface MAC addresses that is used by a WAP to block access to all unauthorized devices. Although it's a useful feature to implement, it can only be used in environments with a small (fewer than 20), static set of wireless clients. Additionally, a hacker with basic wireless hacking tools can discover the MAC address of a valid client and then spoof that address onto their attack wireless client.

Signal strength

Some WAPs provide a physical or logical adjustment of the antenna power levels. Power-level controls are typically set by the manufacturer to a setting that is suitable for most situations. However, if, after performing site surveys and adjusting antenna placement, wireless signals are still not satisfactory, you may need to adjust the power levels. Keep in mind that changing channels, avoiding reflective and signal-scattering surfaces, and reducing interference can often be more significant in improving connectivity reliability.

When adjusting power levels, make minor adjustments instead of attempting to maximize or minimize the setting. Also, take note of the initial/default setting so you can return to that setting if desired. After each power-level adjustment, reset/reboot the WAP before re-performing the site survey and quality tests. Sometimes, lowering the power level can improve performance.

Band selection/width

WiFi band selection should be based on the purpose or use of the wireless network as well as the level of existing interference. For external networks, 2.4 GHz is often preferred because it can provide good coverage over a distance but at slower speeds; 5 GHz is often preferred for internal networks because it provides higher throughput rates (but less coverage area). Higher-frequency radio waves do not penetrate solid objects, like walls and furniture as well, so the 5 GHz band is best suited for open internal environments.

As discussed and illustrated in the Chapter 1 section “Wireless Channels,” within each frequency are divisions of the wireless band known as channels, with 2.4 GHz it is important to select a channel that has little or no interference, because the channels are spaced 5 MHz apart but the channels are used at 22 MHz width. Thus, any 2.4 GHz channel interferes with the three channels above and below it. The 5 GHz band is divided into channels 20 MHz wide, which are spaced 20 MHz apart. Thus, there is no interference between adjacent channels. This also allows adjacent channels to be bonded to create wider channels, which in turn supports faster data throughput.

Antenna types and placement

A wide variety of antenna types can be used for wireless clients and base stations. Many devices’ standard antennas can be replaced with stronger (signal-boosting) antennas.

The standard straight or pole antenna is an omnidirectional antenna that can send and receive signals in all directions perpendicular to the line of the antenna itself. This is the type of antenna found on most base stations and some client devices. It’s sometimes also called a *base antenna* or a *rubber duck antenna* (because most such antennas are covered in a flexible rubber coating).

Most other types of antenna are *directional*: they focus their sending and receiving capabilities in one primary direction. Some examples of directional antennas include Yagi, cantenna, panel, and parabolic. A *Yagi antenna* is similar in structure to a traditional roof TV antenna; it’s crafted from a straight bar with cross sections to catch specific radio frequencies in the direction of the main bar. *Cantennas* are constructed from tubes with one sealed end. They focus along the direction of the open end of the tube. Some of the first cantennas were crafted from Pringles cans. *Panel antennas* are flat devices that focus from only one side of the panel. *Parabolic antennas* are used to focus signals from very long distances or weak sources.

Antenna placement should be a concern when you’re deploying a wireless network. Don’t fixate on a specific location before a proper site survey has been performed. Place the WAP and/or its antenna in a likely position, and then test various locations for signal strength and connection quality. Only after you confirm that a potential antenna placement provides satisfactory connectivity should it be made permanent.

Consider the following guidelines when seeking optimal antenna placement:

- Use a central location.
- Avoid solid physical obstructions.

- Avoid reflective or other flat metal surfaces.
- Avoid electrical equipment.

If a base station has an external omnidirectional antenna, typically it should be positioned pointing straight up vertically. If a directional antenna is used, point the focus toward the area of desired use. Keep in mind that wireless signals are affected by interference, distance, and obstructions.

Fat vs. thin

A fat access point is a base station that is a fully managed wireless system, which operates as a stand-alone wireless solution. A thin access point is little more than a wireless transmitter/receiver, which must be managed from a separate external centralized management console. Most of the management functions have been shifted to an offloading management device so the wireless access point only has to handle the radio signals. The benefit of using thin access points is that management, security, routing, filtering, and more can be concentrated in one location, while there may be dozens or more deployed thin access points throughout a facility. Most fat access points require device-by-device configuration and thus are not as flexible for enterprise use.

Controller-based vs. standalone

Controller-based wireless access points are thin access points that are managed by a central controller. A stand-alone access point is a fat access point that handles all management functions locally on the device.

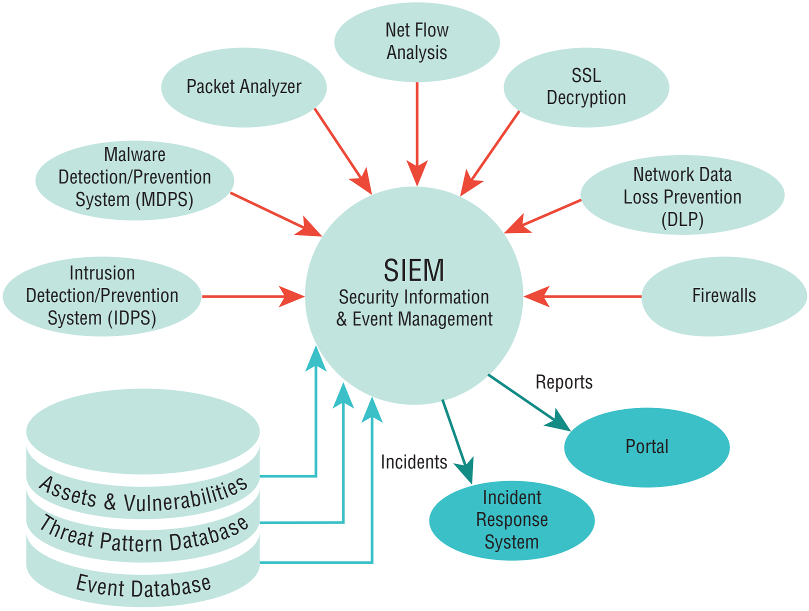
SIEM

A centralized application to automate the monitoring of network systems is essential to many organizations. There are many terms used to describe such a solution, including *Security Information and Event Management (SIEM)*, Security Information Management (SIM), and Security Event Management (SEM). An entire organization’s IT infrastructure can be monitored through real-time event analysis using these types of enterprise-level monitoring systems. SIEM can use triggers or thresholds that oversee specific features, elements, or events that will send alerts or initiate alarms when specific values or levels are breached. This can be seen as a more advanced system than that provided by SNMP (Simple Network Management Console). SNMP focuses on pulling information from network devices; however, it can use trap messages to inform the management console when an event or threshold violation occurs on a monitored system.

SIEMs can be used to monitor the activity of a cluster of email servers. As the email servers log events, the SIEM agent monitors the log for events of interest. If one is noticed, then the SIEM agent forwards the details of the event to the central management SIEM system. The master SIEM analysis engine determines the severity of the event and whether it relates to other recent events, and it may trigger either a notification to be sent to an administrator or an alarm requesting immediate response. An example of SIEM operating to prevent a problem from becoming significant enough to interrupt mission-critical communications would be if an email server begins to have a backlog of unprocessed messages due to a strong surge or increase in inbound messages. The SIEM can notice this change in normal operations and inform the administrators long before the situation becomes a DoS and the system fails to keep up with communications or begins to drop, delete, overwrite, or otherwise lose messages.

SIEMs typically have a wide range of configuration options that allow IT personnel to select which events and occurrences are of importance to the organization. Thus, SIEM allows for customization of monitoring and alerting based on the organization's specific business processes, priorities, and risks. A SIEM solution will include agents for any type of server and may include hooks into network appliances, such as switches, routers, firewalls, IDSs, IPSs, VPNs, and WAPs (Figure 2.17). Thus, a SIEM is an important monitoring and analysis tool for large organizations with a wide variety of systems and devices. The reports from a SIEM solution will keep the IT and security staff informed of the overall state of the environment, and alarms and alerts will enable them to respond promptly to concerns or compromises.

FIGURE 2.17 The concept of SIEM



SIEM can often perform more tasks than just event monitoring; it can also serve as a NAC (network access control) solution by monitoring the configuration and patch status of systems throughout the network. SIEM can provide asset tracking, MAC monitoring, IP management, and system inventory oversight, and can even monitor for unauthorized software installations—whether implemented by a user or via malware infection.

Aggregation

SIEM performs aggregation of logs, event details, and system measurements pulled from the range of devices throughout the network into the centralized management server. This enables SIEM to monitor the entire IT infrastructure and provide administrators with insight into the health and stability of their network as a whole. Aggregation is essential to the benefits of SIEM.

Correlation

Correlation is the comparison and analysis of logged events in order to find similarities or repeating occurrences. The correlation analysis performed by SIEM enables it to notice repeated breaches, trends toward failure, and other forms of escalating or recurring incidents.

Automated alerting and triggers

SIEM uses automated alerting and triggers to keep the network and security administrators informed of any event that may violate security or be deemed suspicious on the network. This is a key feature of SIEM, whose ability to proactively inform the IT staff of concerns before serious harm takes place is what makes it stand out as an essential enterprise tool.

Time synchronization

Time synchronization is always an important element of security solution implementation, including SIEM, which can assist with maintaining the synchronization of the clocks on networked devices. Time synchronization is an important part of re-creating, or at least analyzing, the events of a violation. If the time stamps of log entries are not synchronized, it may be difficult, if not impossible, to actually determine the order in which the events occurred.

Event deduplication

SIEM endeavors to keep the level or load of collected materials from becoming an overwhelming burden on storage capacity. To that end, SIEM performs event deduplication by merging exact duplicates of the same event. Duplicate events may occur because of repetitions of an offending event over time or may simply be multiple recordings of the same event by the same or different monitors due to latency, processing load, or time desynchronization.

Logs/WORM

Logs need to be protected against accidental and intentional malicious change. Centralized logging services such as SIEM and Syslog can assist with protecting the integrity of logs through several techniques. One technique is to create additional duplicate copies of a log in various locations throughout the network. This is the primary log protection technique employed by centralized logging solutions. The primary log still resides on the original system, but a duplicate copy of the log is maintained on one or more log management servers. An additional technique can be to store the log copies on a WORM (write-once, read-many) storage device. These are storage media that prohibit the change of any data item once it has been written. Common examples of WORMs include optical discs and ROM chips, but WORM hard drives and tapes are also available.

DLP

Data loss prevention (DLP) refers to systems specifically implemented to detect and prevent unauthorized access to, use of, or transmission of sensitive information. DLP can include hardware and software elements designed to support this primary goal. It may involve deep packet inspection, storage and transmission encryption, contextual assessment, monitoring authorizations, and centralized management.

A wide range of security measures can be implemented that provide DLP benefits; these include blocking use of email attachments, setting strict job-specific authorization, blocking cut-and-paste, preventing use of portable drives, and setting all storage to be encrypted by default. However, keep in mind that while DLP will reduce the occurrence of accidental data loss and data leakage, anyone intentionally violating company security rules may still find a means to subvert such protections.

Many regulations, such as the Health Insurance Portability and Accountability Act (HIPAA), the Health Information Technology for Economic and Clinical Health (HITECH) Act, the Gramm-Leach-Bliley Act (GLBA), Basel II, and PCI DSS, either directly require DLP solutions or strongly imply the need for DLP.

USB blocking

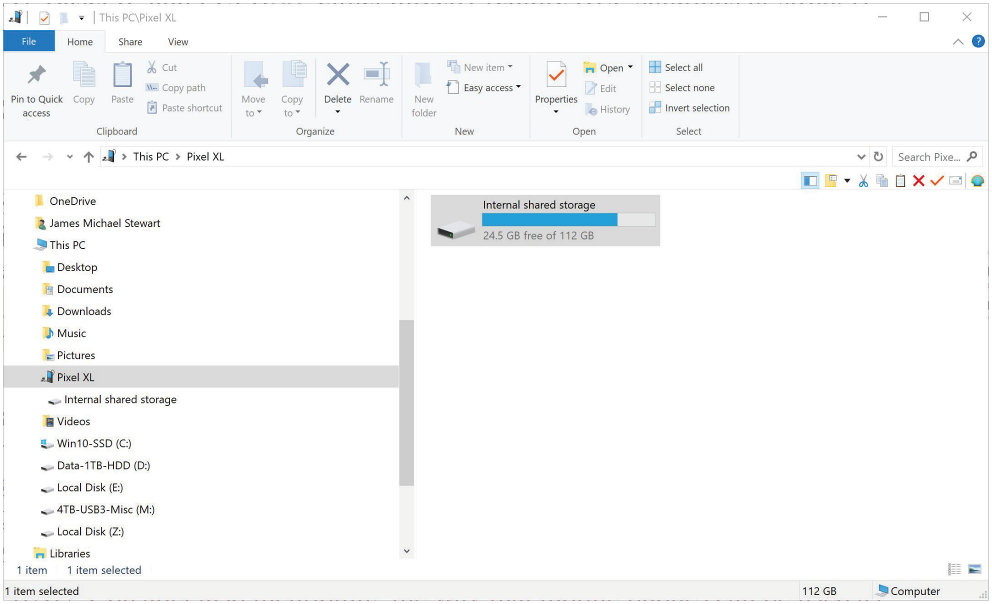
Many data loss and data leakage events take place via USB storage devices, which have significant capacity compared to their size. They are convenient to use as well as easy to hide, lose, or steal. Disabling the use of USB storage devices when setting network use policy can significantly reduce data loss and data leakage that occurs based on portable USB storage.

It might also be worth considering blocking the use of memory cards, such as SD cards and microSD cards. These types of memory cards are almost as ubiquitous as USB drives. And, don't overlook the fact that most mobile phones can function as USB storage when attached to a computer via a USB cable (Figure 2.18). Mobile phones can thus provide writable access to both their internal memory storage and any expanded storage, such as SD or microSD cards.

Cloud-based

Cloud-based DLP must focus on strict authorization in order to prevent unauthorized entities from viewing, accessing, or downloading sensitive data. Cloud-based services may be the data source for portable equipment, point-of-sale devices, desktop applications, and smart devices. Cloud DLP needs to include both storage encryption and transportation encryption in order to restrict access of resources to authorized users, software, and devices.

FIGURE 2.18 A mobile phone's storage is accessible from a Windows system after connecting via a USB cable.



Email

Email DLP often involves blocking or filtering of attachments as well as limiting the use of HTML/web features that may enable data and/or code transmission. Email DLP can also be tied with a blocking of cut-and-paste to prevent users from transmitting body content of sensitive documents to unauthorized recipients.

NAC

Network Access Control (NAC) involves controlling access to an environment through strict adherence to and implementation of security policy. The goals of NAC are to prevent or reduce zero-day attacks, enforce security policy throughout the network, and use identities to perform access control. These goals can be achieved through the use of strong, detailed security policies that define all aspects of security control, as well as filtering, prevention, detection, and response for every device from client to server and for every internal or external communication. NAC is meant to be an automated detection and response system that can react in real time to ensure that all monitored systems are current on patches and updates and are in compliance with the latest security configurations.

NAC can be implemented with either a pre-admission philosophy or a post-admission philosophy. Using the pre-admission philosophy, a system must meet all current security requirements (such as patch application and antivirus updates) before it's allowed to communicate with the network. The post-admission philosophy says that allow/deny decisions are made based on user activity, which is based on a predefined authorization matrix. NAC can also be deployed with aspects of both of these philosophies.

Other issues related to NAC include using a client/system agent versus overall network monitoring (agentless); using out-of-band versus in-band monitoring; and resolving any remediation, quarantine, or captive portal strategies.

A typical operation of an agent-based NAC system would be to install a NAC monitoring agent on each managed system. The NAC agent downloads a configuration file on a regular basis, possibly daily, to check the current configuration baseline requirements against the local system. If the system is not compliant, it can be quarantined into a remediation subnet where it can communicate only with the NAC server. The NAC agent can download and apply updates and configuration files to bring the system into compliance. Once compliance is achieved, the NAC agent returns the system to the normal production network.

Many organizations have released products with the NAC concept in mind (often in the title of their offering), including Cisco, McAfee, Symantec, and so on. There are many open-source solutions as well.

Dissolvable vs. permanent

NAC agents can be either dissolvable or permanent. A dissolvable NAC agent is usually written in a web/mobile language, such as Java or ActiveX, and is downloaded and executed to each local machine when the specific management web page is accessed. The dissolvable NAC agent can be set to run once and then terminate, or it may remain resident in memory until the system reboots.

A permanent NAC agent is installed onto the monitored system as a persistent software background service.

Host health checks

A NAC system can check on the security and performance health of monitored systems. This host health check can be used as part of the procedure for determining what updates and configurations to apply to the system. It can also be used to establish an ongoing record of systems in order to monitor for trends or establish baselines.

Agent vs. agentless

An agent-based system installs an assessment and monitoring software tool on each monitored system. The agents can be dissolvable or persistent/permanent. An agent-based system may be able to apply updates and configuration changes automatically, whereas an agentless system typically requires an administrator to manually resolve any discovered issues. Although agent-based systems provide more data, agentless systems are more trustworthy in the data they do provide, since they cannot be as easily compromised by malware.

Mail gateway

A *mail gateway* or email gateway is an add-on security filter used to reduce the risk of malicious and wasteful emails. A mail gateway filters out malware, phishing scams, and spam messages from inbound mail before they are deposited into a recipient's inbox folder. A mail gateway can also be used to filter outbound messages. Such egress filtering can be a component of DLP or provide some additional protection against distribution of PII (personally identifiable information).

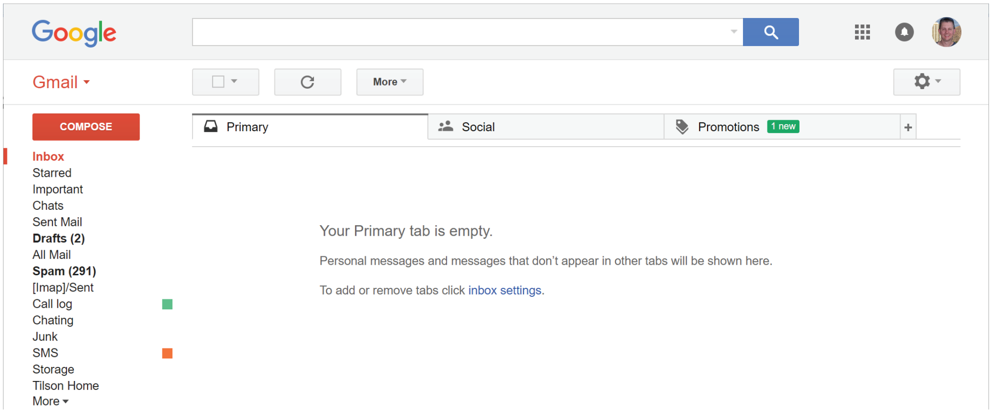
Spam filter

Spam is any type of email that is undesired and/or unsolicited. Think of spam as the digital equivalent of junk mail and door-to-door solicitations.

Spam is a problem for numerous reasons:

- Some spam carries malicious code such as viruses, logic bombs, or Trojan horses.
- Some spam carries social engineering attacks (also known as hoax email).
- Unwanted email wastes your time while you sort through it looking for legitimate messages (Figure 2.19).
- Spam wastes Internet resources: storage capacity, computing cycles, and throughput.

FIGURE 2.19 Notice the spam counter on my Gmail account; this is just the message count for the one week since the last time I cleared it out!



Your Primary tab is empty.

Personal messages and messages that don't appear in other tabs will be shown here.

To add or remove tabs click inbox settings.

The primary countermeasure against spam is an email filter. An *email filter* is a list of email addresses, domain names, or IP addresses where spam is known to originate. If a message is received from one of the listed spam sources, the email filter blocks or discards it. Some email filters are becoming as sophisticated as antivirus scanners. These email filters can examine the header, subject, and contents of a message to look for keywords or phrases that identify it as a known type of spam, and then take the appropriate actions to discard, quarantine, or block the message.

In addition to client application or client-side spam filters, there are enterprise spam tools. Some enterprise tools are stand-alone devices, often called *anti-spam appliances*, whereas others are software additions to internal enterprise email servers. The benefit of enterprise spam filtering is that it reduces spam distribution internally by blocking and discarding unwanted messages before they waste storage space on email servers or make their way to clients.

However, email spam filters are problematic. Just because a message includes keywords that are typically found in spam doesn't mean every message with those words is spam. Some legitimate, if not outright essential, messages include spam words. One method of addressing this issue is for the spam-filtering tool to place all suspected spam messages into a quarantine folder. Users can peruse this folder for misidentified messages and retrieve them.

Another important issue to address when managing spam is spoofed email. A *spoofed* email is a message that has a fake or falsified source address. When an email server receives an email message, it should perform a reverse lookup on the source address of the message. If the source address is fake or nonexistent, the message should be discarded. Other methods of detecting or blocking spoofed messages include checking source addresses against blacklists and filtering on invalid entries in a message header.

Spim

Spim is a term sometimes used to refer to spam over IM. It's also called just spam, instant spam, or IM marketing. No matter what the name, it consists of unwanted messages transmitted through some form of instant messaging service, which can include Short Message Service (SMS).

A *spam filter* is a software or hardware tool whose primary purpose is to identify and block, filter, or remove unwanted messages (that is, spam). Spam is most commonly associated with email, but spam also exists in instant messaging (IM), Short Message Service (SMS), Usenet, and web discussions, forums, comments, and blogs. Because spam consumes a substantial amount of resources worldwide, it's essential to filter and block it at every opportunity. Failing to block spam allows it to waste resources, consume bandwidth, and distract workers from productive activities. Spam can also be a common source of malware infections via links and attachments.

DLP

Mail gateways can be configured to filter the contents of outbound messages against a list of sensitive materials or keywords. This can help to prevent data loss or data leakage events that take place over email.

Encryption

Encryption can be used as a tool for security or a means to bypass filters. A mail gateway will be unable to filter on the contents of encrypted messages, but it can still filter on the source email address, destination, email address, and any other value left in plain text in the message header. A mail gateway can be configured to block all unencrypted messages to specific clients in order to enforce confidentiality on those communications.

Bridge

A traditional network bridge was a device used to link local LANs together. The local LANs were originally hub-based networks. With the implementation of switches, the network bridge is no longer a common device used in a typical network deployment.

Another new concern that is related to bridging is when a single system has two active network interfaces, such as one to the company network and the other to an Internet connection. In these situations, it is usually important to ensure that bridging is not enabled. On some systems, this feature is known as IP forwarding. Bridging or IP forwarding allows traffic to flow directly between the two connected interfaces without filters. This is not a secure network configuration.

As IPv6 is deployed, many organizations will adopt an IPv4-to-IPv6 tunnel, proxy, or bridge. This may allow an organization to retain IPv4 use in some sections of their network or interface with an Internet service provider that only supports IPv4.

SSL/TLS accelerators

SSL accelerators or *TLS accelerators* are used to offload the operation of encryption to a dedicated hardware device. This frees up resources on a server or system itself while still maintaining the security of the connection. By allowing a dedicated SSL/TLS hardware accelerator to manage secure connections, more efficient encryption is available to more concurrent sessions.

SSL decryptors

An *SSL decryptor* or *TLS decryptor* is a dedicated device used to decode secure communications for the purpose of filtering and monitoring. An SSL/TLS decryptor can be deployed in line or be used for out-of-band management. When deployed in line, the decryptor serves as an SSL/TLS offloader; it is where the encryption and decryption of a communication is handled rather than being managed by the Web or email server itself. Such an inline implementation also enables the use of load balancing to distribute communication loads across a number of hosting servers.

An out-of-band monitoring tool only needs to decrypt communications for filtering rather than also needing to encrypt outbound messages. Such a configuration provides the filtering or IDS/IPS systems with a plain-text version of the communications.

Media gateway

A *media gateway* is any device or service that converts data from one communication format to another. A media gateway is often located at the intersection of two different types of networks. Media gateways are commonly used with VoIP systems, where a conversion from IP-based communications to analog or digital is needed.

Hardware security module

The *hardware security module (HSM)* is a cryptoprocessor used to manage and store digital encryption keys, accelerate crypto operations, support faster digital signatures, and improve authentication. An HSM is often an add-on adapter or peripheral, or it can be a TCP/IP network device. HSMs include tamper protection to prevent their misuse even if an attacker gains physical access.

HSMs provide an accelerated solution for large (2,048+ bit) asymmetric encryption calculations and a secure vault for key storage. Many certificate authority systems use HSMs to store certificates; ATM and POS bank terminals often employ proprietary HSMs; hardware SSL accelerators can include HSM support; and DNSSEC-compliant DNS servers use HSM for key and zone file storage.

One common example of an HSM is the *trusted platform module (TPM)*. This special chip found on many portable system motherboards can be used to store the master encryption key used for whole drive encryption.

Exam Essentials

Understand firewalls. Firewalls provide protection by controlling traffic entering and leaving a network. They manage traffic using filters or rules.

Know the types of firewalls. The three basic types of firewalls are packet filtering, circuit-level gateway, and application-level gateway. A fourth type combines features from these three and is called a stateful inspection firewall.

Understand implicit deny. Implicit deny is the default security stance that says if you aren't specifically granted access to or privileges for a resource, you're denied access by default.

Comprehend application-based vs. network-based firewalls. An application firewall is a device, server add-on, virtual service, or system filter that defines a strict set of communication rules for a service and all users. A network firewall is a hardware device, typically called an appliance, designed for general network filtering. A network firewall is designed to provide broad protection for an entire network.

Understand stateful vs. stateless firewalls. A stateless firewall analyzes packets on an individual basis against the filtering ACLs. The context of the communication or previous packets are not used to make an allow or deny decision on the current packet. A stateful firewall monitors the state or session of the communication; it evaluates previous packets and potentially other communications and conditions when making an allow or deny decision for the current packet.

Understand VPN concentrators. A VPN concentrator is a dedicated hardware device designed to support a large number of simultaneous VPN connections, often hundreds or thousands. It provides high availability, high scalability, and high performance for secure VPN connections.

Know IPsec. IPSec is a security architecture framework that supports secure communication over IP. IPSec establishes a secure channel in either transport mode or tunnel mode. It can be used to establish direct communication between computers or to set up a VPN between networks.

Understand AH and ESP. IPSec isn't a single protocol but rather a collection of protocols. Two of the primary protocols of IPSec are Authentication Header (AH) and Encapsulating Security Payload (ESP). AH provides authentication of the sender's data; ESP provides encryption of the transferred data as well as limited authentication.

Understand tunnel mode and transport mode. In tunnel mode, IPSec provides encryption protection for both the payload and the message header by encapsulating the entire original LAN protocol packet and adding its own temporary IPSec header. In transport mode, IPSec provides encryption protection for just the payload and leaves the original message header intact.

Understand IKE. Internet Key Exchange (IKE) ensures the secure exchange of secret keys between communication partners in order to establish an encrypted VPN tunnel.

Understand ISAKMP. Internet Security Association and Key Management Protocol (ISAKMP) is used to negotiate and provide authenticated keying material (a common method of authentication) for security associations in a secured manner. The four major functional components of ISAKMP are authentication of communications peers, threat mitigation, security association creation and management, and cryptographic key establishment and management.

Know split tunnel. A split tunnel is a VPN configuration that allows a VPN-connected system to access both the organizational network over the VPN and the Internet directly at the same time. The split tunnel thus grants a simultaneously open connection to the Internet and the organizational network.

Understand IDS. An intrusion detection system (IDS) is an automated system that either watches activity in real time or reviews the contents of audit logs in order to detect intrusions or security policy violations. The two types of IDS are network-based and host-based.

Understand NIDS. A network-based IDS (NIDS) watches network traffic in real time. It's reliable for detecting network-focused attacks, such as bandwidth-based DoS attacks.

Understand HIDS. A host-based IDS (HIDS) watches the audit trails and log files of a host system. It's reliable for detecting attacks directed against a host, whether they originate from an external source or are being perpetrated by a user locally logged into the host.

Know detection mechanisms. Signature detection compares event patterns against known attack patterns (signatures) stored in the IDS database. Anomaly detection watches the ongoing activity in the environment and looks for abnormal occurrences.

Understand response methods. An IDS with active detection and response is designed to take the quickest action to reduce potential damage caused by an intruder. This response may include shutting down the server or the affected service or disconnecting suspicious connections. An IDS with passive detection and response takes no direct action against the intruder; instead, it may increase the amount of data being audited and recorded and notify administrators about the intrusion.

Understand behavior-based detection. A behavior-based monitoring or detection method relies on the establishment of a baseline or a definition of normal and benign. Once this baseline is established, the monitoring tool is able to detect activities that vary from that standard of normal.

Understand signature-based detection. A signature-based monitoring or detection method relies on a database of signatures or patterns of known malicious or unwanted activity. The strength of a signature-based system is that it can quickly and accurately detect any event from its database of signatures.

Understand anomaly-based detection. An anomaly-based monitoring or detection method relies on definitions of all valid forms of activity. This database of known valid activity allows the tool to detect all anomalies.

Know routers. Routers enable traffic from one network segment to traverse into another network segment. However, the traffic must pass through the router's filters in order to make the transition.

Understand router access control lists. Access control lists (ACLs) are used to define who is allowed or denied permission to perform a specified activity or action. ACLs are commonly associated with object access but also apply to communications. In many cases, firewalls, routers, and switches use ACLs as a method of security management.

Understand switches. A switch is a networking device used to connect other devices together and potentially implement traffic management on their communications. It receives signals in one port and transmits them out the port where the intended recipient is connected. Switches are often used to create virtual local area networks (VLANs).

Comprehend loop protection. A loop in networking terms is a transmission pathway that repeats itself. Loop protection includes STP for Ethernet and the IP header TTL value.

Understand proxy. A proxy server is a variation of an application-level firewall or circuit-level firewall. A proxy server is used as a proxy or middleman between clients and servers.

Understand load balancers. A load balancer is used to spread or distribute network traffic load across several network links or network devices. The purpose of load balancing is to obtain optimal infrastructure utilization, minimize response time, maximize throughput, reduce overloading, and eliminate bottlenecks.

Understand wireless access points. A wireless access point is the network management device that supports and manages an infrastructure mode wireless network.

Be familiar with 802.11 and 802.11a, b, g, n, and 802.11n (150+ Mbps). 802.11 is the IEEE standard for wireless network communications. Versions include 802.11a (2 Mbps), 802.11b (11 Mbps), and 802.11g (54 Mbps). The 802.11 standard also defines Wired Equivalent Privacy (WEP).

Understand MAC filters. A MAC filter is a list of authorized wireless client interface MAC addresses that is used by a WAP to block access to all unauthorized devices.

Understand SSID broadcast. Wireless networks traditionally announce their SSIDs on a regular basis in a special packet known as the beacon frame. When the SSID is broadcast, any device with an automatic detect and connect feature can see the network and initiate a connection with it.

Know the antenna types. A wide variety of antenna types can be used for wireless clients and base stations. These include omnidirectional pole antennas as well as many directional antennas such as Yagi, cantenna, panel, and parabolic.

Understand site surveys. A site survey is the process of investigating the presence, strength, and reach of WAPs deployed in an environment. This task usually involves walking around with a portable wireless device, taking note of the wireless signal strength, and mapping it on a plot or schematic of the building.

Understand SIEM. Security Information and Event Management (SIEM) is a centralized application to automate the monitoring and real-time event analysis of network systems.

Comprehend DLP. Data loss prevention (DLP) is the idea of systems specifically implemented to detect and prevent unauthorized access to, use of, or transmission of sensitive information. DLP can include hardware and software elements designed to support this primary goal.

Understand NAC. Network Access Control (NAC) means controlling access to an environment through strict adherence to and implementation of security policies. The goals of NAC are to prevent or reduce zero-day attacks, enforce security policy throughout the network, and use identities to perform access control.

Understand a mail gateway. A mail gateway or email gateway is an add-on security filter used to reduce the risk of malicious and wasteful emails. A mail gateway filters out malware, phishing scams, and spam messages from inbound mail before they are deposited into a recipient's inbox folder.

Be aware of spam. Spam is undesired or unsolicited email. It's a problem for numerous reasons:

- Spam can be the carrier for malicious code such as viruses, logic bombs, and Trojan horses.
- Spam can be the carrier of a social engineering attack (hoax email).
- Unwanted email wastes your time while you're sorting through it looking for legitimate messages.
- Spam wastes Internet resources such as storage capacity, computing cycles, and throughput.

Understand SPIM. SPIM is a term used to refer to spam over IM (instant messaging).

Understand bridges. A traditional network bridge was a device used to link local LANs together. Another new concern that is related to bridging is when a single system has two active network interfaces.

Know SSL/TLS accelerators. SSL accelerators or TLS accelerators are used to offload the operation of encryption to a dedicated hardware device. This frees up resources on a server or system itself while still maintaining the security of the connection.

Understand SSL decryptors. An SSL decryptor or TLS decryptor is a dedicated device used to decode secure communications for the purpose of filtering and monitoring.

Understand media gateways. A media gateway is any device or service that converts data from one communication format to another. A media gateway is often located at the intersection of two different types of networks.

Understand HSMs. The hardware security module (HSM) is a cryptoprocessor used to manage and store digital-encryption keys, accelerate crypto operations, support faster digital signatures, and improve authentication.

2.2 Given a scenario, use appropriate software tools to assess the security posture of an organization.

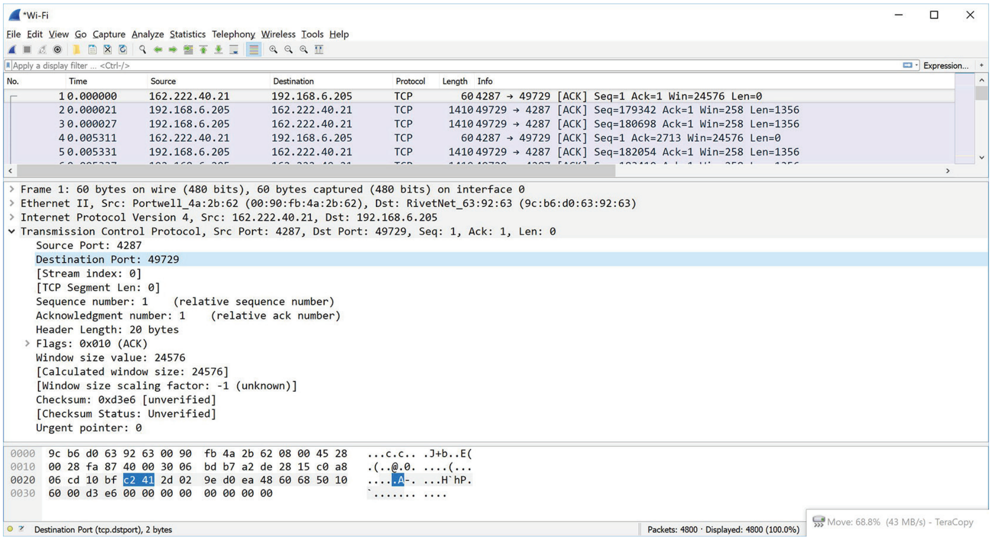
Security management starts immediately after initial security deployment. Security management never ends. There is always something new to learn about the security of an environment. This is due to the changing nature of attacks and exploitation as well as the evolution of production networks. This section looks at many of the security tools used to assess the security stance of an IT infrastructure.

Protocol analyzer

A *protocol analyzer* is a tool used to examine the contents of network traffic. Commonly known as a sniffer, a protocol analyzer can be a dedicated hardware device or software installed on a typical host system. In either case, a protocol analyzer is first a packet-capturing tool that can collect network traffic and store it in memory or on a storage device. Once a packet is captured, it can be analyzed either with complex automated tools and scripts or manually. A protocol analyzer usually places the NIC into promiscuous mode in order to see and capture all packets on the local network segment rather than just those with the destination MAC address of the computer's local NIC. In promiscuous mode, the NIC ignores the destination MAC addresses of packets and collects each one it sees.

Once a network packet is collected, it's either saved to the hard drive in a log file or retained in memory in a buffer. The protocol analyzer can examine individual packets down to the binary level. Most analyzers or sniffers automatically parse out the contents of the header into an expandable outline form (Figure 2.20). Any configuration or setting can be easily seen in the header details. The payload of packets is often displayed in both hexadecimal and ASCII.

FIGURE 2.20 Wireshark, a network sniffer, showing an expanded view of the TCP header of a captured communication


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Expression...

```
No.  Time      Source          Destination     Protocol Length Info
  1 0.000000   162.222.40.21   192.168.6.205   TCP        60 4287 → 49729 [ACK] Seq=1 Ack=1 Win=24576 Len=0
  2 0.000021   192.168.6.205   162.222.40.21   TCP      1410 49729 → 4287 [ACK] Seq=179342 Ack=1 Win=258 Len=1356
  3 0.000027   192.168.6.205   162.222.40.21   TCP      1410 49729 → 4287 [ACK] Seq=180698 Ack=1 Win=258 Len=1356
  4 0.005311   162.222.40.21   192.168.6.205   TCP        60 4287 → 49729 [ACK] Seq=1 Ack=2713 Win=24576 Len=0
  5 0.005331   192.168.6.205   162.222.40.21   TCP      1410 49729 → 4287 [ACK] Seq=182054 Ack=1 Win=258 Len=1356
```

```
> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: Portwell_4a:2b:62 (00:90:fb:4a:2b:62), Dst: RivetNet_63:92:63 (9c:b6:d0:63:92:63)
> Internet Protocol Version 4, Src: 162.222.40.21, Dst: 192.168.6.205
v Transmission Control Protocol, Src Port: 4287, Dst Port: 49729, Seq: 1, Ack: 1, Len: 0
    Source Port: 4287
    Destination Port: 49729
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 1    (relative sequence number)
    Acknowledgment number: 1    (relative ack number)
    Header Length: 20 bytes
  > Flags: 0x010 (ACK)
    Window size value: 24576
    [Calculated window size: 24576]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0xd3e6 [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
```

```
0000  9c b6 d0 63 92 63 00 90  fb 4a 2b 62 08 00 45 28   ...c.c.. .J+b..E(
0010  00 28 fa 87 40 00 30 06  bd b7 a2 de 28 15 c0 a8   .(..@.0. ....(...
0020  06 cd 10 bf c2 41 2d 02  9e d0 ea 48 60 68 50 10   .....A-. ...H`hP.
0030  60 00 d3 e6 00 00 00 00  00 00 00 00               `....... ....
```

```
Destination Port (tcp.dstport), 2 bytes          Packets: 4800 · Displayed: 4800 (100.0%)
```

Sniffers typically offer both capture filters and display filters. A capture filter is a set of rules to govern which packets are saved into the capture file or buffer and which are discarded. Capture filters are used to collect only packets of interest and keep the number of retained packets to a minimum. A display filter is used to show only those packets from the packet file or buffer that match your requirements. Display filters act like search queries to locate packets of interest.

Protocol analyzers vary from simple raw packet-capturing tools to fully automated analysis engines. There are both open-source (such as Wireshark) and commercial (such as Omnipeek and NetScout) options.

Protocol analyzers can be used to discover communication problems caused by hardware and software issues. They can detect protocol anomalies that may be due to misconfiguration, malfunction, or malicious intent. Often, when security administrators attempt to track down a network communication problem or discover the source of an attack, they use a protocol analyzer.

Sniffer may either be a synonym for *protocol analyzer* or may mean a distinct type of product. A sniffer is generally a packet- (or frame-) capturing tool, whereas a protocol analyzer is able to decode and interpret packet/frame contents.

A protocol analyzer is used by network administrators throughout their internal network, within a DMZ, and even on the open Internet to evaluate network communications. When there are odd or unexplained network events occurring, a protocol analyzer might be useful in capturing traffic related to the event to diagnose and troubleshoot the issue.

Protocol analyzers can capture live traffic to assist administrators in determining the cause of communication failures or service and application issues based on header values and payload data.

Network scanners

A network scanner is usually a form of port scanner that adds enumeration techniques in order to inventory the devices found on a network. A network scanner can use a variety of detection techniques to discover the presence of a system on a network, whether a valid device or a rogue system. These techniques include ping sweeping, port scanning, and promiscuous mode detections.

Ping sweeping is the activity of using ICMP Type 8 Echo Request packets to trigger ICMP Type 0 Echo Reply packets from any system within a specific subnet. However, only systems that are not filtering or ignoring ICMP will respond.

Port scanning can be used to detect the presence of an open port. If an open port is detected, it means that there is a system present at the IP address probed. Open TCP ports will always respond with a SYN/ACK reply if they are sent a SYN-flagged initial packet. However, if port probes are sent too quickly, intelligent firewalls can block open port responses.

Promiscuous mode detection is accomplished by sending out queries or requests but addressing them to a MAC address that is not in use on the network. Any NIC in normal mode will ignore the request, but NICs in promiscuous mode will accept the query and respond. This trick or technique will detect any system that is in promiscuous mode, but may otherwise not respond to other techniques.

A network scanner is often used to locate and identify devices on a network. Administrators may use network scanners to inventory the network and look for rogue or out-of-place systems.

Rogue system detection

A rogue system is any device not authorized to be present on a private network. Rogue systems may be wired, wireless, or virtual machines. A network scanner can assist with detecting rogue systems by detecting machines that are not present on a preapproved system index.

Network mapping

Network mapping is often an important part of security management. It is used in addition to rogue machine detection to ensure that every system present on the network is authorized and that all expected systems are accounted for. Tools used for network mapping may either produce a text listing of the discovered systems or a visual diagram including typical details such as IP address, possibly MAC address, subnet group, OS type, and system name or identity.

Wireless scanners/cracker

A wireless scanner is used to detect the presence of a wireless network. Any wireless network that is not enclosed in a Faraday cage can be detected, since the base station will be transmitting radio waves. A Faraday cage is an enclosure that filters or blocks all target frequencies of radio waves in order to prevent cross-boundary eavesdropping. Wireless networks that have their SSID broadcast disabled are detectable, since they are still transmitting radio signals.

A wireless scanner is able to quickly determine whether there are wireless networks in the area, what frequency and channel they are using, their network name, and what level of encryption is in use. A wireless scanner is also able to discern the MAC addresses of the base station and all connected clients because the Ethernet header in the wireless communications will be in plain text even with WPA-2 encryption.

Once a wireless network is discovered, WEP network encryption can be compromised in moments due to its poor implementation of RC4. WPA networks, which are also based on RC4, are better, but their encryption can be cracked in less than 12 hours. Only WPA-2 encryption based on AES is currently impossible to crack.

Most organizations that are not using a Faraday cage to contain their wireless signals are providing a potential attack avenue to hackers. Even with a WPA-2 encrypted network, an attacker can discover the MAC addresses of all wireless devices, take note of the volume and timing of traffic, and implement effective DoS attacks.

Wireless scanners are used by network administrators to monitor and evaluate the health of their wireless networks. A wireless scanner can be used to confirm WAP configuration, inventory wireless clients, and assist in tracking down rogue or unauthorized wireless devices.

Password cracker

The strength of a password is generally measured in the amount of time and effort required to break the password through various forms of cryptographic attacks. These attacks are collectively known as *password cracking* or *password guessing*. A weak password invariably uses only alphanumeric characters; often employs dictionary or other common words; and may include user profile–related information such as birthdates, Social Security numbers, and pet names. A strong password is longer, more complex, unique, and changed on a regular basis.

A password is typically stored as its hash. A password hash does not contain the password characters, but it is a representation of the password produced by the hashing algorithm. Future authentication events hash the user's newly presented characters to the stored hash. If the two hashes match, the user is authenticated; if not, they are rejected.

Password hashes can be attacked using reverse engineering, reverse hash matching (aka rainbow table attack), or a birthday attack. These attack methods are commonly used by password-cracking tools. Hashes can't be reversed or "decrypted," so this is generally a secure system. But because the hash algorithm used by commercial software is known (or can be easily discovered), password crackers can be written to exploit the stored password hashes.

Passwords are usually stored in a hashed format for the security provided by the one-way process. However, even though it isn't possible to reverse the hash process directly, it's possible to reverse-engineer a hash. Reverse-engineering a hash (aka reverse hash matching) is the idea of taking a potential data set, hashing it, and then comparing it to the hash you wish to crack. By repeating that process until it succeeds or the options are exhausted with different potential data sets (possible passwords), the hacker can reveal (crack) passwords.

This form of hashing attack exploits the mathematical characteristic that if two messages are hashed and their hashes are the same, the messages must be the same. This can be written as `H(M)=H(M') therefore M=M'`.

Weak passwords are short or are otherwise easy to guess. Weak passwords often allow hackers or unscrupulous employees to obtain access to another person's logon credentials. Compromising weak passwords is possible through a wide variety of attacks, including password guessing or cracking.

Password guessing is an attack aimed at discovering the passwords employed by user accounts. There are several forms of password-guessing attack tools: some attempt to guess passwords by attacking a logon prompt, others try to extract passwords directly from an accounts database, and still others attempt to capture authentication traffic and extract passwords out of the network packet. In most cases, the latter two options employ birthday attack (reverse hash matching) methods to discover the password used by a user account.

There are innumerable password-guessing and cracking tools on the Internet. No matter what tool is used to discover passwords, the most important countermeasure against password crackers is to use long, complex passwords and change them on a regular basis.

Password-cracking tools compare hashes from potential passwords with the hashes stored in the accounts database (obtained or stolen through any number of means). Potential passwords are either generated on the fly using all possible combinations of characters or pulled from a precompiled list of passwords (known as *dictionary lists*). Each potential password is hashed, and that hash value is compared with the accounts database. If a match is found, the password-cracker tool has discovered a password for a user account. Birthday attacks, rainbow table attacks, dictionary attacks, and brute-force attacks, initially described in Chapter 1, are prime examples of password-cracking attacks.

A password cracker is used by system administrators to stress-test the strength of their users' passwords. Any password discovered by an administrator-controlled password cracker must be changed to something stronger and more resistant to password-cracking techniques. Password crackers should be used on an isolated offline system in order to prevent attackers from using legitimate password-auditing activities as a means to steal credentials.

Vulnerability scanner

A *vulnerability scanner* is a tool used to scan a target system for known holes, weaknesses, or vulnerabilities. These automated tools have a database of attacks, probes, scripts, and so on that are run against one or more systems in a controlled manner. Vulnerability scanners are designed to probe targets and produce a report of the findings. They can be used from within a private network to test internal systems directly or from outside the network to test border devices against breaching attacks.

Note that *vulnerability scanner* is often used as a general term for a tool that performs any sort of security assessment or that could be used in a security evaluation. This is evident in that this term appears multiple times on the objectives list. Here, the term is used to refer to a specific tool that checks for symptoms of weakness. Be sure to consider this on the exam and look for context clues to decipher the intention of a question.

Vulnerability scanners are designed not to cause damage while they probe for weaknesses, but they can still inadvertently cause errors, slower network performance, and downtime. Thus, it's important to plan their use and prepare for potential recovery actions.

Vulnerability scanners can be commercial products, such as Retina, or open-source, such as Nessus. Most organizations take advantage of several vulnerability scanners in order to gain the most complete perspective on their security status. Each time a vulnerability scanner is to be used, it should be updated from the vendor.

A vulnerability scanner should be used on a regular basis to identify vulnerabilities, weaknesses, and misconfigurations in all parts of a company network.

Configuration compliance scanner

A *configuration compliance scanner* is a form of manually operated NAC. It is a tool that quickly scans a system to check whether approved updates and patches are installed and whether the system is in compliance with security and general system configuration settings.

A configuration compliance scanner should be used by system administrators on a regular basis to check for and monitor the settings status of the devices on the network.

Exploitation frameworks

An exploitation framework is a vulnerability scanner that is able to fully exploit the weaknesses it discovers. It can be an automated or manual exploit assessment tool. For example, Metasploit is an open-source exploitation framework, and Immunity Canvas and Core Impact are commercial exploitation frameworks. Often an exploitation framework allows for customization of the test elements as well as the crafting of new tests to deploy against your environment's targets.

An exploitation framework does have additional risk compared to that of a vulnerability scanner, as it attempts to fully exploit any discovered weaknesses. Thus, it is important to make sure a reliable backup has been created and that an incident response policy (IRP) is in effect that can recover damaged data or systems promptly. The purpose or goal of an exploitation framework is not to intentionally cause harm, but the thoroughness of the philosophy of testing can inadvertently cause data loss or downtime.

An exploitation framework is an advanced vulnerability scanner that should be used by system administrators and security administrators to stress-test the security stance of the IT infrastructure. Regular scans and evaluations using an exploitation framework will assist IT managers with finding and resolving security concerns before they are discovered by an attacker.

Data sanitization tools

Data sanitization is the concept of removing data from a storage device so that it is no longer recoverable. Standard operating system functions of deletion and formatting leave data remnants behind that can be recovered by undelete data recovery utilities. This is because deletion and formatting only mark storage device sectors as available without actually removing any existing data. Data sanitization overwrites existing data with new data in order to prevent data recovery. The overwriting process can write random data, all 1's, all 0's (known as zeroization), or some repeated pattern of 1's and 0's.

Data sanitization tools should be used by anyone discarding, recycling, or reselling a computer system or storage device. Due to the risk of data remnant recovery and data loss/leakage, all storage devices should be sanitized before they leave your secured environment.

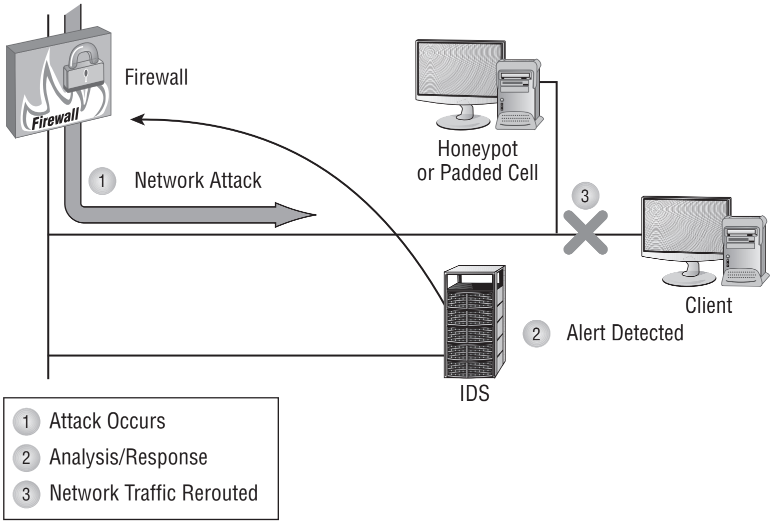
Steganography tools

See the Chapter 6 section "Steganography" for a discussion of this topic.

Honeypot

A *honeypot* is a fictitious environment designed to fool attackers and intruders and lure them away from the private secured network (see Figure 2.21). A honeypot is often deployed as a buffer network between an untrusted network, such as the Internet, business partners, or a DMZ, and the private network. In this position, the honeypot serves as a decoy and distraction for attackers.

FIGURE 2.21 A network honeypot deceives an attacker and gathers intelligence.



The honeypot looks and acts like a real system or network, but it doesn't contain any valuable or legitimate data or resources. Intruders may be fooled into wasting their time attacking and infiltrating a honeypot instead of your actual network. All the activity in the honeypot is monitored and recorded.

The purpose of deploying a honeypot is to provide an extra layer of security, specifically a detection mechanism, for your private network and to gather information about attacks and, potentially, sufficient evidence for prosecution against malicious intruders and attackers. A honeypot can often gather sufficient information to determine the identity of the intruder; the type of data, resource, or system being attacked or focused on; and the methods and tools of attack.

Honeypots are effective if they're easier for a hacker or intruder to find than the real private LAN being protected. They should be modestly secured so they seem like a real network, but not overly secured. The goal is to distract attackers and lure them away from your intranet so you can learn about new attacks and potentially be able to track down criminals for prosecution. If a honeypot seems too easy to access or doesn't react and behave like a real production network, experienced hackers and intruders won't be fooled and may be provoked to find and attack your actual production network.

Another form of honeypot is known as a *padded cell*. Whereas a honeypot is usually a distracting network that is always on, a padded cell is a containment area that is activated only when an intrusion is detected or when an unauthorized command or software launch or execution is attempted.

A honeypot can be used whenever there is a risk of an attacker finding a way to breach a shared resource that is not for public consumption. Honeypots are of little value in front of a public web server or a known email system, but can be effective in front of systems that are not publicly accessible.

Backup utilities

Backup utilities create backups of data onto alternate storage devices. Backups are insurance against data loss. Only when a backup is available can damaged, deleted, or corrupted data be restored. Backup utilities can be configured to perform backups on an automated basis at specific periods or time intervals.

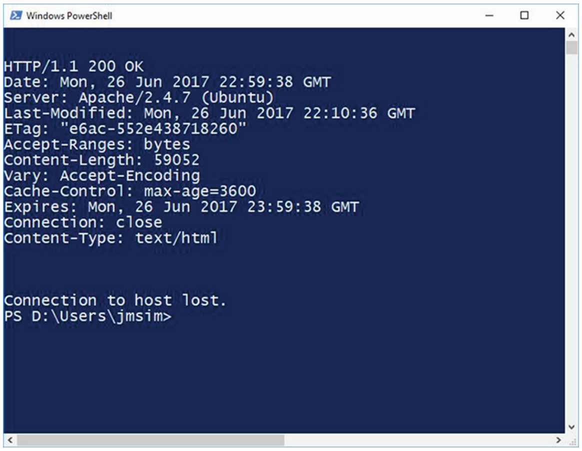
Please see the Chapter 5 section, "Backup concepts," for details about various types of backups.

Backup utilities should be used to provide a recovery option for data sets, including system and device configurations. Backups should either be continuous or implemented on a periodic basis. Backups are the only available insurance against data loss.

Banner grabbing

Banner grabbing is the process of capturing the initial response or welcome message from a network service. A *banner grab* occurs when a request for data or identity is sent to a service on an open port and that service responds with information that may directly or indirectly reveal its identity. Often the banner discloses the application's identity, version information, and potentially much more. A common method of banner grabbing against web servers is to use the telnet client to send a plain-text query. This can be accomplished by opening a command prompt, typing **`telnet www.apache.org 80`**, pressing Enter, typing **`HEAD / HTTP/1.0`**, and pressing Enter a few more times. This should result in the display of an HTTP 200 OK message (Figure 2.22), which often includes a Server line that identifies the specific web server product in use. Banner grabbing is a common technique used by both hackers and researchers to learn more about an unknown system across a network connection.

FIGURE 2.22 The result of banner grabbing `www.apache.org`



```
HTTP/1.1 200 OK
Date: Mon, 26 Jun 2017 22:59:38 GMT
Server: Apache/2.4.7 (Ubuntu)
Last-Modified: Mon, 26 Jun 2017 22:10:36 GMT
ETag: "e6ac-552e438718260"
Accept-Ranges: bytes
Content-Length: 59052
Vary: Accept-Encoding
Cache-Control: max-age=3600
Expires: Mon, 26 Jun 2017 23:59:38 GMT
Connection: close
Content-Type: text/html


Connection to host lost.
PS D:\Users\jmsim>
```

Banner grabbing can be used when an attacker or an administrator wishes to learn more about a targeted system. If a banner can be retrieved from a target system, it often includes details as to the product and version of the application running on a port as well as information regarding the underlying operating system.

Passive vs. active

A *passive* tool, technique, or technology is one that monitors a situation but doesn't do anything about it. This can include recording details, launching analysis engines, and notifying administrators. Passive actions or tools don't affect an event and are unseen (or unnoticed) by the event (or subject of the event).

An *active* tool, technique, or technology is one that intercedes in a situation in order to alter events or chance outcomes. This can include altering settings, opening or closing ports, rebooting devices, restarting services, launching applications, disconnecting clients, restoring data, and so on. Active actions or tools affect the event and are thus detectable by the event or the subjects of the event.

Passive tools are to be used when monitoring is preferred over reaction, such as when watching allowed activities. Active tools are to be used when response and containment, such as stopping breach attempts, are more important than ongoing information gathering.

Command line tools

Some security tools are command-line tools. Here is a list of some of the command-line tools you should be familiar with for the Security+ exam.

A scenario that might involve the use of several of these tools is during an investigation to seek out a potential rogue system in a private network. The `nmap` tool can be used to detect the presence of systems by performing an array of port scans. The `ping` tool can be used to verify that the target's IP address is active and in use. The `tracert` command can be used to determine the router closest to the target system. The `arp` command can be used to determine the MAC address of the rogue system from its IP address. The `nslookup` or `dig` tool might be used to determine whether the rogue machine is registered with the directory service's DNS system. The `tcpdump` tool can be run to collect packets sent to or received from the target system. And `netcat` might be used to attempt to connect to any open ports on the target system in order to perform banner grabbing or other information discovery probing activities.

Ping

Internet Control Message Protocol (ICMP) is a network health and link-testing protocol. ICMP operates in Layer 3 as the payload of an IP packet. It's the protocol commonly used by tools such as `ping`, `traceroute`, and `pathping`. Most uses of ICMP revolve around its echo-request to echo-reply system. ICMP is also used for error announcement or transmission. However, ICMP provides information only when a packet is actually received. If ICMP request queries go unanswered, or ICMP replies are lost or blocked, then ICMP provides no information.

ICMP is also a protocol commonly used for network scanning and malicious attacks. When it's used as a network-scanning protocol, ping sweeps are used to identify the IP addresses in use. However, because ICMP can be ignored or blocked, this makes it an unreliable host-discovery tool. As for malicious attacks, ICMP abuses include Ping of Death, Smurf, and Loki.

The Ping of Death creates multiple packet fragments that are reassembled on the target to create an ICMP/IP packet that is larger than the maximum valid size of 65,535 bytes. On unprotected systems, this can cause freezing or rebooting.

Smurf abuses ICMP by using it in a flooding attack. An attacker sends ICMP echo requests to the directed broadcast addresses of numerous networks with insecure Internet-accessible router or firewall interfaces. These requests are spoofed so they appear to come from the victim's IP address. Each recipient of the echo request sends back an echo reply to the victim, causing a flood of traffic to DoS the victim.

Loki is a tool that uses ICMP as an encapsulation or tunnel protocol. Effectively, Loki uses ICMP like an unencrypted VPN. It operates across network boundaries that allow outbound ICMP echo requests and their corresponding inbound echo replies.

ICMP functions or operates around a signaling system known as Type and Code. There are roughly 40 defined types for ICMP; the five most common (and relevant for the Security+ exam) are listed in Table 2.3.

TABLE 2.3 Common ICMP types

ICMP type	Description
8	Echo request
0	Echo reply
11	Time exceeded
3	Destination unreachable
5	Redirect

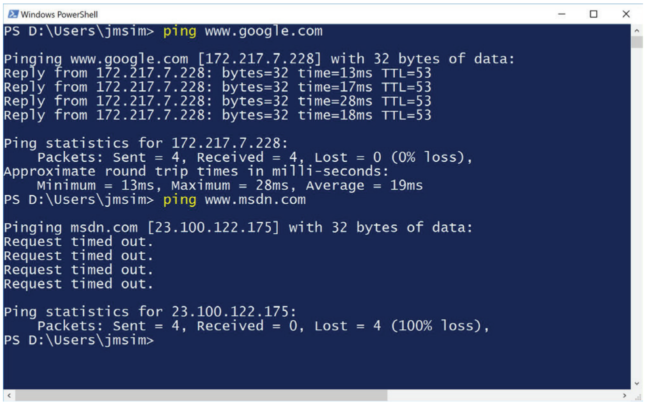
Some types have further detailed designations using codes. For example, Type 3 destination unreachable has 14 codes used to provide more specific detail about the reason or cause of the type. A common example, Type 3, Code 3—which means destination unreachable, destination port unreachable—is the standard response from a closed UDP port when packets are sent to it.

The `ping` command employs the ICMP Type 8 and Type 0 messages. On the Windows platform, `ping` sends out four echo requests (Figure 2.23), whereas on most other platforms, such as Linux, it indefinitely repeats the transmission of an echo request until the tool is terminated. If the target system is operating and able to respond, an echo request is sent back to the requesting system. If the echo request is received, `ping` confirms this information by displaying messages about the replies, which include the size, round-trip time, and resulting TTL. If no reply is received, the tool displays the error "Request timed out." A positive result confirms the ability to access the remote system. The "Request timed out" error, however, does not necessarily mean the remote system is offline—it can also mean the system is too busy to respond, the routing to the target is flawed, the system is blocking ICMP, or the system is not responding to ICMP. Thus, using ICMP-based ping is an unreliable means to determine whether a system is present and online.

Experiment with the `ping` command from your own system's command prompt. Use the `ping -h` command to view the syntax details.

ICMP Type 11 time exceeded in addition to Types 8 and 0 are used by `tracert` (see later section, "tracert"). Type 3 is often received when access to a destination is denied or fails. There is usually a sub-code returned with the Type 3 ICMP message that gives more specific information regarding the reason for the failure (`https://en.wikipedia.org/wiki/Internet_Control_Message_Protocol`). The Type 5 redirect was used to temporarily implement a static route detour, but due to hacker abuses this type is generally ignored by public routers.

FIGURE 2.23 The ping command

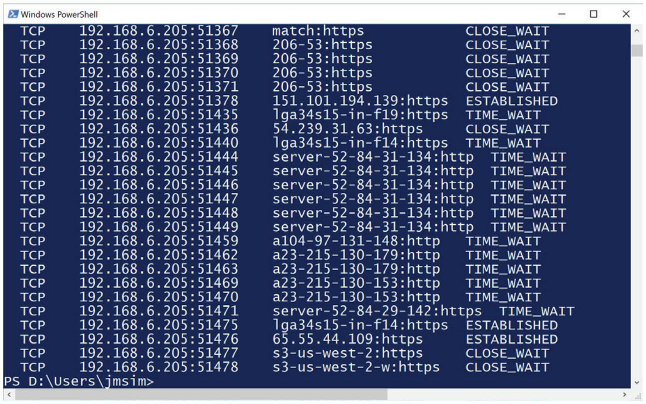


```
PS D:\Users\jmsim> ping www.google.com

Pinging www.google.com [172.217.7.228] with 32 bytes of data:
Reply from 172.217.7.228: bytes=32 time=13ms TTL=53
Reply from 172.217.7.228: bytes=32 time=17ms TTL=53
Reply from 172.217.7.228: bytes=32 time=28ms TTL=53
Reply from 172.217.7.228: bytes=32 time=18ms TTL=53

Ping statistics for 172.217.7.228:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 13ms, Maximum = 28ms, Average = 19ms
PS D:\Users\jmsim> ping www.msdn.com

Pinging msdn.com [23.100.122.175] with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 23.100.122.175:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PS D:\Users\jmsim>
```

netstat

The command netstat displays information about TCP sessions of a system. The output options include displaying the source and destination IP address and port number of active connections (Figure 2.24), listing the program associated with a connection, showing traffic bytes, displaying Ethernet statistics, showing the FQDN for external addresses, and displaying the routing table.

FIGURE 2.24 The output of a netstat command



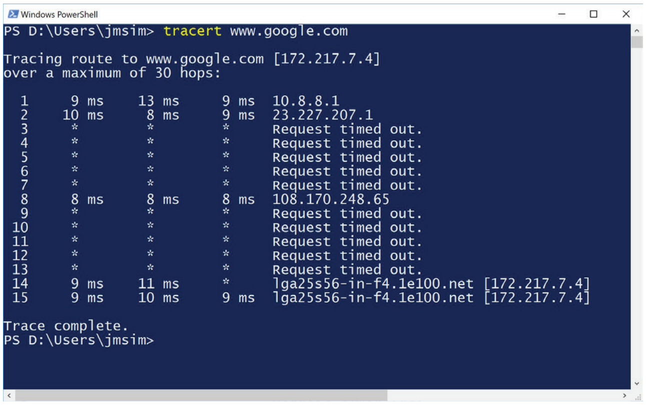
```
  TCP    192.168.6.205:51367    match:https            CLOSE_WAIT
  TCP    192.168.6.205:51368    206-53:https           CLOSE_WAIT
  TCP    192.168.6.205:51369    206-53:https           CLOSE_WAIT
  TCP    192.168.6.205:51370    206-53:https           CLOSE_WAIT
  TCP    192.168.6.205:51371    206-53:https           CLOSE_WAIT
  TCP    192.168.6.205:51378    151.101.194.139:https  ESTABLISHED
  TCP    192.168.6.205:51435    lga34s15-in-f19:https  TIME_WAIT
  TCP    192.168.6.205:51436    54.239.31.63:https     CLOSE_WAIT
  TCP    192.168.6.205:51440    lga34s15-in-f14:https  TIME_WAIT
  TCP    192.168.6.205:51444    server-52-84-31-134:http  TIME_WAIT
  TCP    192.168.6.205:51445    server-52-84-31-134:http  TIME_WAIT
  TCP    192.168.6.205:51446    server-52-84-31-134:http  TIME_WAIT
  TCP    192.168.6.205:51447    server-52-84-31-134:http  TIME_WAIT
  TCP    192.168.6.205:51448    server-52-84-31-134:http  TIME_WAIT
  TCP    192.168.6.205:51449    server-52-84-31-134:http  TIME_WAIT
  TCP    192.168.6.205:51459    a104-97-131-148:http   TIME_WAIT
  TCP    192.168.6.205:51462    a23-215-130-179:http   TIME_WAIT
  TCP    192.168.6.205:51463    a23-215-130-179:http   TIME_WAIT
  TCP    192.168.6.205:51469    a23-215-130-153:http   TIME_WAIT
  TCP    192.168.6.205:51470    a23-215-130-153:http   TIME_WAIT
  TCP    192.168.6.205:51471    server-52-84-29-142:https  TIME_WAIT
  TCP    192.168.6.205:51475    lga34s15-in-f14:https  ESTABLISHED
  TCP    192.168.6.205:51476    65.55.44.109:https     ESTABLISHED
  TCP    192.168.6.205:51477    s3-us-west-2:https     CLOSE_WAIT
  TCP    192.168.6.205:51478    s3-us-west-2-w:https   CLOSE_WAIT
PS D:\Users\jmsim>
```

Experiment with the netstat command from your own system's command prompt. Use the netstat -h command to view the syntax details.

tracert

The command `tracert` (Windows) or `traceroute` (Linux) is used to discover the route between a local system and a remote system. `tracert` uses the ICMP protocol. It sends toward the destination the same ICMP Type 8 echo request that is used by `ping`, but it manipulates the IP header's TTL. The first wave of three requests has a TTL of only 1. The first router decrements the TTL by 1 to discover that the TTL is zero. Once the TTL has reached zero, the router discards the request packet and crafts a new ICMP Type 11 Time Exceeded message, which is sent back to the origin. The origin system uses the source IP address of the Type 11 packet, which is the IP address of a router, in a reverse DNS lookup. If there is a PTR (pointer) record for the IP address, the domain name of the router is displayed along with the IP address. If there is no domain name associated with the IP address, then only the IP address is shown. Each subsequent query has an incremented TTL, which continues until a default maximum 30 hops is reached or a Type 0 echo reply is received from the target (Figure 2.25).

FIGURE 2.25 The `tracert` command



```
PS D:\Users\jmsim> tracert www.google.com

Tracing route to www.google.com [172.217.7.4]
over a maximum of 30 hops:

  1     9 ms    13 ms     9 ms  10.8.8.1
  2    10 ms     8 ms     9 ms  23.227.207.1
  3     *        *        *     Request timed out.
  4     *        *        *     Request timed out.
  5     *        *        *     Request timed out.
  6     *        *        *     Request timed out.
  7     *        *        *     Request timed out.
  8     8 ms     8 ms     8 ms  108.170.248.65
  9     *        *        *     Request timed out.
 10     *        *        *     Request timed out.
 11     *        *        *     Request timed out.
 12     *        *        *     Request timed out.
 13     *        *        *     Request timed out.
 14     9 ms    11 ms     *     lga25s56-in-f4.1e100.net [172.217.7.4]
 15     9 ms    10 ms     9 ms  lga25s56-in-f4.1e100.net [172.217.7.4]

Trace complete.
PS D:\Users\jmsim>
```

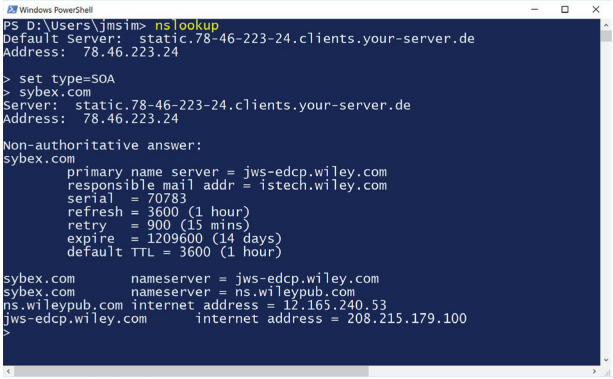
Experiment with the `tracert` command from your own system's command prompt. Use the `tracert -h` command to view the syntax details.

nslookup/dig

The command-line tools `nslookup` and `dig` are used to perform manual DNS queries. The `nslookup` tool is found on Windows, and the `dig` tool is on Linux. These tools initially perform queries against the system's configured DNS server. However, it is possible to refocus the tools to an alternate DNS server to perform queries.

Experiment with the `nslookup` and `dig` commands from your own system's command prompt. On Windows, the `nslookup` tool can be used in interactive or noninteractive mode. The interactive mode allows for numerous sequential commands to be issued while inside the nslookup interface (Figure 2.26). To launch nslookup in interactive mode, issue the command **nslookup** and then, to see a list of syntax, enter **?**. Noninteractive mode singular commands can be issued using command syntax. To view the syntax, enter **nslookup -?**. On Linux, issue the **dig -?** command to view the syntax of this tool.

FIGURE 2.26 The nslookup tool

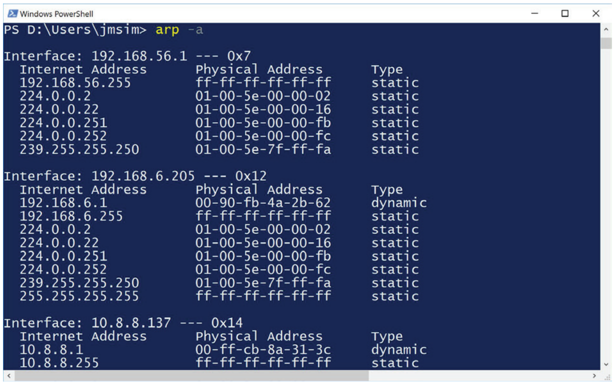


```
PS D:\Users\jmsim> nslookup
Default Server:  static.78-46-223-24.clients.your-server.de
Address:  78.46.223.24

> set type=SOA
> sybex.com
Server:  static.78-46-223-24.clients.your-server.de
Address:  78.46.223.24

Non-authoritative answer:
sybex.com
        primary name server = jws-edcp.wiley.com
        responsible mail addr = istech.wiley.com
        serial  = 70783
        refresh = 3600 (1 hour)
        retry   = 900 (15 mins)
        expire  = 1209600 (14 days)
        default TTL = 3600 (1 hour)

sybex.com       nameserver = jws-edcp.wiley.com
sybex.com       nameserver = ns.wileypub.com
ns.wileypub.com internet address = 12.165.240.53
jws-edcp.wiley.com      internet address = 208.215.179.100
>
```

arp

The arp command is used to display or manipulate the contents of the ARP cache. The ARP cache shows the current table of associations between a MAC address and an IP address. With the arp tool you can view the current ARP cache (Figure 2.27), delete entries, or add new entries.

FIGURE 2.27 The arp command



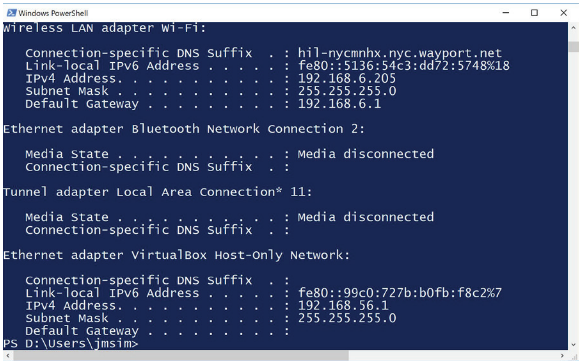
```
PS D:\Users\jmsim> arp -a

Interface: 192.168.56.1 --- 0x7
  Internet Address      Physical Address      Type
  192.168.56.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.2             01-00-5e-00-00-02     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static

Interface: 192.168.6.205 --- 0x12
  Internet Address      Physical Address      Type
  192.168.6.1           00-90-fb-4a-2b-62     dynamic
  192.168.6.255         ff-ff-ff-ff-ff-ff     static
  224.0.0.2             01-00-5e-00-00-02     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

Interface: 10.8.8.137 --- 0x14
  Internet Address      Physical Address      Type
  10.8.8.1              00-ff-cb-8a-31-3c     dynamic
  10.8.8.255            ff-ff-ff-ff-ff-ff     static
```

Experiment with the `arp` command from your own system's command prompt. Use the `arp -?` command to view the syntax details.

ipconfig/ip/ifconfig

The Windows command-line tool `ipconfig` is used to display IP configuration and make some modifications to the interface. The `ipconfig` command can display summary or full interface configurations (Figure 2.28), release a DHCP-assigned IP address, trigger a DHCP renewal of an IP address, purge the DNS cache, and show the contents of the DNS cache.

FIGURE 2.28 The Windows `ipconfig` command



```
Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . : hil-nycmnhx.nyc.wayport.net
   Link-local IPv6 Address . . . . . : fe80::5136:54c3:dd72:5748%18
   IPv4 Address. . . . . . . . . . . : 192.168.6.205
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.6.1

Ethernet adapter Bluetooth Network Connection 2:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Tunnel adapter Local Area Connection* 11:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

Ethernet adapter VirtualBox Host-Only Network:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::99c0:727b:b0fb:f8c2%7
   IPv4 Address. . . . . . . . . . . : 192.168.56.1
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :
PS D:\Users\jmsim>
```

Experiment with the `ipconfig` command from your own Windows system's command prompt. Use the command `ipconfig /?` to view the syntax details.

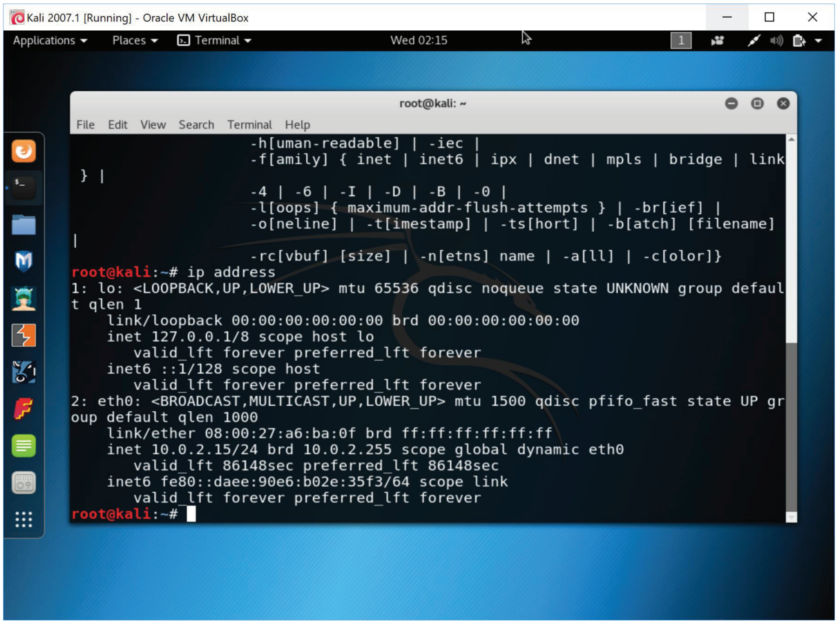
The Linux command tools `ifconfig` and `ip` are used to manipulate the configuration settings of network interface cards. The `ifconfig` command is older and is slated to be replaced by the `ip` command. These tools can be used to show current NIC configuration (Figure 2.29), enable and disable an interface, set an IP address, and remove an IP address. The `ip` command can be used to perform many other network-related functions, including adding ARP cache entries, showing the routing table, and changing the routing table.

Experiment with the `ifconfig` and `ip` commands from your own Linux system's command prompt. Use the command `ifconfig -h` or `ip -h` to view the syntax details.

tcpdump

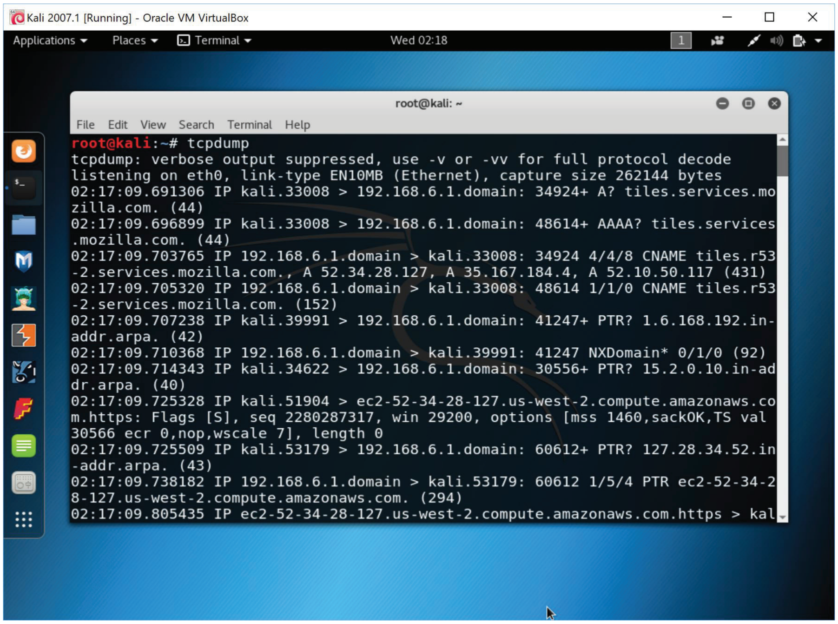
The command tool `tcpdump` is a raw packet-capturing utility (Figure 2.30) found on Linux. It can be used to capture packets into a capture file. It supports command-line capture filters in order to collect specific packets. The output capture file can be examined by a number of other tools, including GUI packet analysis utilities such as Wireshark.

FIGURE 2.29 The Linux ip command



```
                   -h[uman-readable] | -iec |
                   -f[amily] { inet | inet6 | ipx | dnet | mpls | bridge | link
 } |
                   -4 | -6 | -I | -D | -B | -0 |
                   -l[oops] { maximum-addr-flush-attempts } | -br[ief] |
                   -o[neline] | -t[imestamp] | -ts[hort] | -b[atch] [filename]
|
                   -rc[vbuf] [size] | -n[etns] name | -a[ll] | -c[olor]}
root@kali:~# ip address
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaul
t qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP gr
oup default qlen 1000
    link/ether 08:00:27:a6:ba:0f brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
       valid_lft 86148sec preferred_lft 86148sec
    inet6 fe80::daee:90e6:b02e:35f3/64 scope link
       valid_lft forever preferred_lft forever
root@kali:~#
```

FIGURE 2.30 The tcpdump command



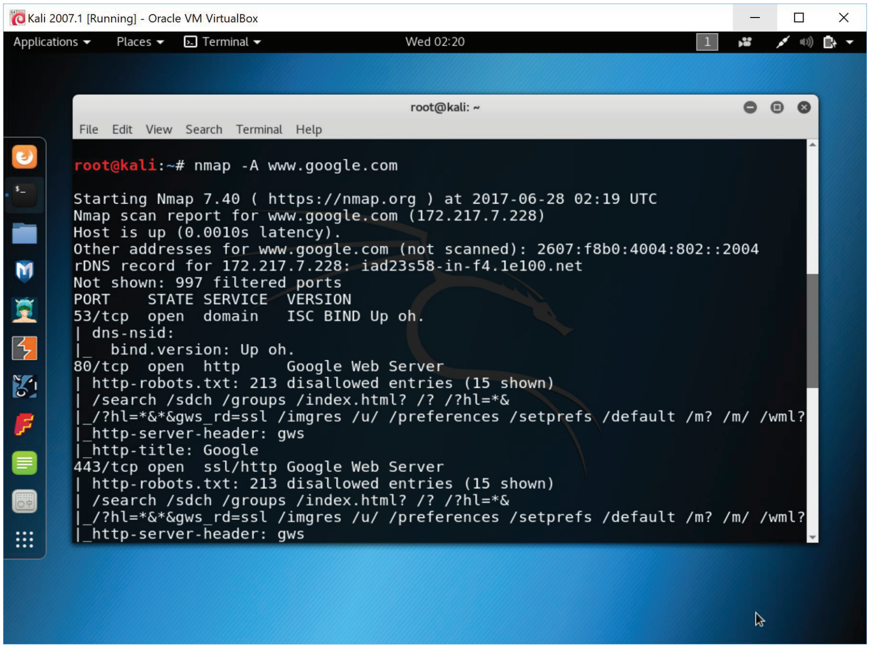
```
root@kali:~# tcpdump
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes
02:17:09.691306 IP kali.33008 > 192.168.6.1.domain: 34924+ A? tiles.services.mo
zilla.com. (44)
02:17:09.696899 IP kali.33008 > 192.168.6.1.domain: 48614+ AAAA? tiles.services
.mozilla.com. (44)
02:17:09.703765 IP 192.168.6.1.domain > kali.33008: 34924 4/4/8 CNAME tiles.r53
-2.services.mozilla.com., A 52.34.28.127, A 35.167.184.4, A 52.10.50.117 (431)
02:17:09.705320 IP 192.168.6.1.domain > kali.33008: 48614 1/1/0 CNAME tiles.r53
-2.services.mozilla.com. (152)
02:17:09.707238 IP kali.39991 > 192.168.6.1.domain: 41247+ PTR? 1.6.168.192.in-
addr.arpa. (42)
02:17:09.710368 IP 192.168.6.1.domain > kali.39991: 41247 NXDomain* 0/1/0 (92)
02:17:09.714343 IP kali.34622 > 192.168.6.1.domain: 30556+ PTR? 15.2.0.10.in-ad
dr.arpa. (40)
02:17:09.725328 IP kali.51904 > ec2-52-34-28-127.us-west-2.compute.amazonaws.co
m.https: Flags [S], seq 2280287317, win 29200, options [mss 1460,sackOK,TS val
30566 ecr 0,nop,wscale 7], length 0
02:17:09.725509 IP kali.53179 > 192.168.6.1.domain: 60612+ PTR? 127.28.34.52.in
-addr.arpa. (43)
02:17:09.738182 IP 192.168.6.1.domain > kali.53179: 60612 1/5/4 PTR ec2-52-34-2
8-127.us-west-2.compute.amazonaws.com. (294)
02:17:09.805435 IP ec2-52-34-28-127.us-west-2.compute.amazonaws.com.https > kal
```

Experiment with the tcpdump command from your own system's command prompt. Use the command tcpdump /h to view the syntax details.

nmap

The command nmap is a network mapper or port scanner. The nmap tool can be used to perform a wide range of network discovery and enumeration functions, including ping sweeping, port scanning (Figure 2.31), application identification, operating system identification, firewall and IDS evasion, and a plethora of script functions to discover details about target applications and OSs. Zenmap is a GUI interface to nmap.

FIGURE 2.31 The nmap tool.



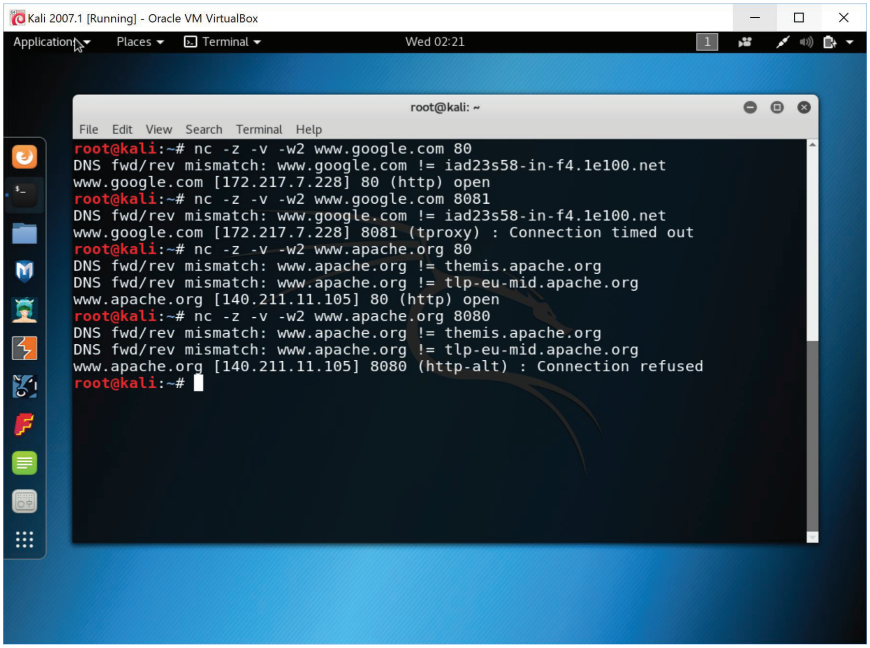
```
root@kali:~# nmap -A www.google.com

Starting Nmap 7.40 ( https://nmap.org ) at 2017-06-28 02:19 UTC
Nmap scan report for www.google.com (172.217.7.228)
Host is up (0.0010s latency).
Other addresses for www.google.com (not scanned): 2607:f8b0:4004:802::2004
rDNS record for 172.217.7.228: iad23s58-in-f4.1e100.net
Not shown: 997 filtered ports
PORT    STATE SERVICE  VERSION
53/tcp  open  domain   ISC BIND Up oh.
| dns-nsid:
|_  bind.version: Up oh.
80/tcp  open  http     Google Web Server
| http-robots.txt: 213 disallowed entries (15 shown)
| /search /sdch /groups /index.html? /? /?hl=*&
|_/?hl=*&*&gws_rd=ssl /imgres /u/ /preferences /setprefs /default /m? /m/ /wml?
|_http-server-header: gws
|_http-title: Google
443/tcp open  ssl/http Google Web Server
| http-robots.txt: 213 disallowed entries (15 shown)
| /search /sdch /groups /index.html? /? /?hl=*&
|_/?hl=*&*&gws_rd=ssl /imgres /u/ /preferences /setprefs /default /m? /m/ /wml?
|_http-server-header: gws
```

Please experiment with the nmap command from your own system's command prompt. Use the command nmap -h to view the syntax details.

netcat

The netcat command is a flexible network utility used to write to or read from TCP and UDP network connections. Its command tool is just nc. This tool can be used to redirect standard input and output over network pathways, even for tools and utilities which do not have network capabilities natively. In addition to redirecting input and output, it can also be used as a basic port scanner (Figure 2.32), perform file transfers, act as a port listener, and even serve as a remote control backdoor.

FIGURE 2.32 The nc (netcat) command


Kali 2007.1 [Running] - Oracle VM VirtualBox
Applications
Places
Terminal
Wed 02:21
root@kali: ~
File Edit View Search Terminal Help

```
root@kali:~# nc -z -v -w2 www.google.com 80
DNS fwd/rev mismatch: www.google.com != iad23s58-in-f4.1e100.net
www.google.com [172.217.7.228] 80 (http) open
root@kali:~# nc -z -v -w2 www.google.com 8081
DNS fwd/rev mismatch: www.google.com != iad23s58-in-f4.1e100.net
www.google.com [172.217.7.228] 8081 (tproxy) : Connection timed out
root@kali:~# nc -z -v -w2 www.apache.org 80
DNS fwd/rev mismatch: www.apache.org != themis.apache.org
DNS fwd/rev mismatch: www.apache.org != tlp-eu-mid.apache.org
www.apache.org [140.211.11.105] 80 (http) open
root@kali:~# nc -z -v -w2 www.apache.org 8080
DNS fwd/rev mismatch: www.apache.org != themis.apache.org
DNS fwd/rev mismatch: www.apache.org != tlp-eu-mid.apache.org
www.apache.org [140.211.11.105] 8080 (http-alt) : Connection refused
root@kali:~# 
```

Please experiment with the nc command from your own system's command prompt. Use the command nc -h to view the syntax details. Netcat is getting increasingly difficult to find, as it is no longer a stand-alone project. To get the command, readers will need to either install nmap and get it bundled with that project, or install one of the forks like cryptocat.

Exam Essentials

Understand protocol analyzers A protocol analyzer is a tool used to examine the contents of network traffic.

Understand network scanners A network scanner is usually a form of port scanner which adds enumeration techniques in order to inventory the devices found on a network.

Comprehend wireless scanners/crackers A wireless scanner is used to detect the presence of a wireless network. Once a wireless network is discovered, WEP network encryption can be compromised with a wireless cracker in moments, due to its poor implementation of RC4. WPA networks, which are also based on RC4, are better, but their encryption can be cracked in less than 12 hours.

Understand hashing attacks. Hashing can be attacked using reverse engineering, reverse hash matching, or a birthday attack. These attack methods are commonly used by password-cracking tools.

Know vulnerability scanners. A vulnerability scanner is a tool used to scan a target system for known holes, weaknesses, or vulnerabilities. These automated tools have a database of attacks, probes, scripts, and so on that are run against one or more systems in a controlled manner.

Understand configuration compliance scanners A configuration compliance scanner is a form of manually operated NAC. It is a tool that quickly scans a system to check whether or not approved updates and patches are installed and whether the system is in compliance with security and general system configuration settings.

Be aware of exploitation frameworks An exploitation framework is a vulnerability scanner that is able to fully exploit the weaknesses it discovers.

Understand data sanitization Data sanitization is the concept of removing data from a storage device so that it is no longer recoverable.

Understand honeypots. A honeypot is a fictitious environment designed to fool attackers and intruders and lure them away from the private secured network. The purpose of deploying a honeypot is to provide an extra layer of protection for your private network and to gather sufficient evidence for prosecution against malicious intruders and attackers.

Understand backup utilities Backup utilities create backups of data on alternate storage devices.

Know banner grabbing. Banner grabbing is the process of capturing the initial response or welcome message from a network service. Often the banner discloses the application's identity, version information, and potentially much more.

Comprehend a variety of command, line tools You should be familiar with several command, line tools, including `ping`, `netstat`, `tracrt`, `nslookup`/`dig`, `arp`, `ipconfig`/`ip`/`ifconfig`, `tcpdump`, `nmap`, and `netcat`.

Understand ICMP Internet Control Messaging Protocol (ICMP) is a network health and link-testing protocol. It operates in Layer 3 as the payload of an IP packet. ICMP is the protocol commonly used by tools such as `ping`, `traceroute`, and `pathping`.

2.3 Given a scenario, troubleshoot common security issues.

A key component of security management is being able to resolve security issues as they occur. Developing knowledge and skill related to common security issues is not only essential to real-world system management, but is important for the Security+ exam.

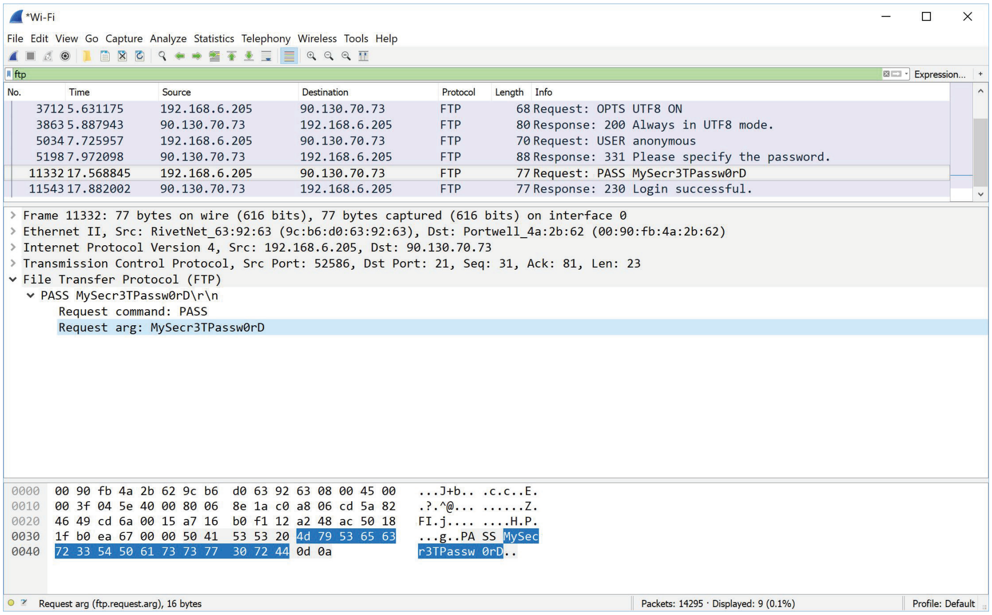
Unencrypted credentials/clear text

It is no longer an acceptable practice to allow authentication to take place over a plain-text or clear-text communication channel. All authentication, without exception, should be encrypted. When an authentication mechanism is discovered to be using clear-text transmission, it is important to cease the use of that system until its authentication solution is secured.

All credentials that might have been transmitted over plain text should be changed. A preferred means of securing authentication would be to salt passwords before they are hashed (see the Chapter 6 section "Salt, IV, nonce"), then use a robust transmission encryption system such as TLS, SSH, or IPSec.

When older insecure versions of services, such as FTP (Figure 2.33), Telnet, or even HTTP, are in use, troubleshooting the presence of unencrypted credentials that are transmitted in clear text is valuable. These older protocols and services do not encrypt authentication by default, and thus are a target for an eavesdropping event in which an attacker could discover account credentials.

FIGURE 2.33 A captured plain-text FTP password using Wireshark


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ftp
Expression...

```
No.    Time       Source          Destination     Protocol  Length Info
  3712 5.631175   192.168.6.205   90.130.70.73    FTP           68 Request: OPTS UTF8 ON
  3863 5.887943   90.130.70.73    192.168.6.205   FTP           80 Response: 200 Always in UTF8 mode.
  5034 7.725957   192.168.6.205   90.130.70.73    FTP           70 Request: USER anonymous
  5198 7.972098   90.130.70.73    192.168.6.205   FTP           88 Response: 331 Please specify the password.
 11332 17.568845  192.168.6.205   90.130.70.73    FTP           77 Request: PASS MySecr3TPassw0rD
 11543 17.882002  90.130.70.73    192.168.6.205   FTP           77 Response: 230 Login successful.
```

```
> Frame 11332: 77 bytes on wire (616 bits), 77 bytes captured (616 bits) on interface 0
> Ethernet II, Src: RivetNet_63:92:63 (9c:b6:d0:63:92:63), Dst: Portwell_4a:2b:62 (00:90:fb:4a:2b:62)
> Internet Protocol Version 4, Src: 192.168.6.205, Dst: 90.130.70.73
> Transmission Control Protocol, Src Port: 52586, Dst Port: 21, Seq: 31, Ack: 81, Len: 23
v File Transfer Protocol (FTP)
  v PASS MySecr3TPassw0rD\r\n
      Request command: PASS
      Request arg: MySecr3TPassw0rD
```

```
0000  00 90 fb 4a 2b 62 9c b6  d0 63 92 63 08 00 45 00   ...J+b.. .c.c..E.
0010  00 3f 04 5e 40 00 80 06  8e 1a c0 a8 06 cd 5a 82   .?.^@... ......Z.
0020  46 49 cd 6a 00 15 a7 16  b0 f1 12 a2 48 ac 50 18   FI.j.... ....H.P.
0030  1f b0 ea 67 00 00 50 41  53 53 20 4d 79 53 65 63   ...g..PA SS MySec
0040  72 33 54 50 61 73 73 77  30 72 44 0d 0a            r3TPassw 0rD..
```

Request arg (ftp.request.arg), 16 bytes | Packets: 14295 · Displayed: 9 (0.1%) | Profile: Default

Logs and events anomalies

A securely managed environment should be recording logs of all system and user events. When an anomaly in the logged events is discovered, the response should address the specific violation. However, when the anomaly is with the logging system itself, this also requires specific and immediate attention.

When the logging, auditing, and even tracking systems of the environment are malfunctioning, it may be prudent to block all external access to the system until the issue is resolved. If possible, restrict access to the more sensitive and valuable data systems as long as monitoring is not operational.

Promptly determine whether the issue can be resolved from within the current system or if a backup version of the system needs to be restored to re-enable the logging mechanisms.

Be sure to back up and preserve the logs as they currently exist. Verify that proper authorization is still assigned to the services performing the logging and auditing to ensure they can still write to the log files. Verify that there is sufficient storage capacity on the target drives. Recheck that user authorization is properly configured, which typically means that only specific administrators have any level of access to the log files.

If your organization suspects intrusions, other security violations, or simply odd system or application behavior, it would be a good idea to review log files and event records for anomalies. Look for anything that stands out as atypical for the device, system, or network.

Permission issues

Permissions or privileges are abilities granted to users over individual objects, such as files and printers. (By contrast, user rights are abilities granted to users over the operating system, such as the ability to reboot or install device drivers.) It is important to assign permissions so that users have sufficient privileges to accomplish their work tasks, but do not have any substantial additional capabilities. This is known as the principle of least privilege.

When users have too much privilege, the organization is at a higher risk than necessary. When users have too little privilege, they are unable to accomplish their work responsibilities.

To assess permission issues, an administrator first must understand what permissions and privileges a user needs to do their job, and then determine the current effective permissions for the user on the objects they need access to. This is done by accumulating the permissions granted, either through group memberships or to the user account directly, and then removing any denials of permissions. If the resulting effective permissions are not correct for the position, then adjustments need to be made by adding or removing the user from a group or adding or removing user-specific permissions. Permission issues can also be related to users having access to resources they should not. These are addressed by removing any specific user allows, removing a user from a group, removing a group from access, or adding a specific user or group denial for the object.

Troubleshooting permission issues is necessary whenever a user is unable to access a resource that they previously were able to access or that they should be able to access. In large environments, it is common for a user account to be a member of numerous groups. Thus, it is possible that the permissions and privileges granted by one group are removed by another. Review the effective permissions for the affected user on the respective resources and compare the settings to other similar resources that the user is able to access.

Access violations

An access violation can be described as either an unauthorized logon event or an unauthorized resource access event, which occurs when a person accesses a system for which they do not have authorization. However, if they performed a valid logon with their credentials, the fault is with the configuration of the authentication and authorization systems. The administrator needs to adjust the configurations to prevent the logon event from occurring in the future.

Similarly, if a valid user is able to access a resource they should not be able to access, this is a failure of authorization. An administrator should reassess and reconfigure the authorization configuration, specifically effective permissions for the user and from the object's perspective (see the previous section).

Whenever there is an anomaly in system activities, executables or other files that are not authorized appear, or expected files and applications are not present, it is good troubleshooting practice to consider whether access violations have occurred. If an unauthorized local or remote access event has taken place, it may have left behind changes to the system, which may be noticed by attentive administrators or users.

Certificate issues

Certificate issues can be related to a wide range of potential misconfigurations, policy violations, or missing information. If an end user is unable to verify a digital certificate, they might not trust the CA that issued it. If this is the case, the CA should be reviewed to determine if they are an entity worthy of being trusted, and if they are, the CA's public key should be added to the trusted roots list (TRL) on the client.

If a customer of a CA abuses their certificate by using it in a criminal activity, changing their confirmed identity, or otherwise violating the terms of the certificate policy, then the CA should revoke the certificate.

If an end user is still accepting a revoked certificate, it may be that their client utility is unable to download the certificate revocation list (CRL) or is unable to query the online certificate status protocol (OCSP) service. To resolve this issue, update the client utility and/or alter its configuration to use the correct address of the CA to access these revocation status–checking resources.

Troubleshooting certificate authentication should take place whenever an authentication event fails, especially if the event has been successful previously. Look for any changes to the system, software, configuration, or networking. Review the root CA's certificate and verify that the subject's certificate is still valid and has not been revoked.

For more information on certificates and certificate management, please see the Chapter 6 section "Given a scenario, implement public key infrastructure."

Data exfiltration

When data exfiltration occurs, an outsider or unauthorized entity has gained access to internal data. This is a data loss or data leakage event. To respond to this situation, first determine what information or data was involved and what risk or consequences are likely due to the leakage. Next, address the entity violating the security policy. If they are an employee, this may mean a stern reprimand, a termination, or filing criminal charges. If they are an external entity, filing criminal charges may be the only option. Discover the means by which the exfiltration occurred and implement new countermeasures to prevent the same violation from taking place again.

Troubleshooting data exfiltration should include reviewing logs of user activity, checking authorization settings, and investigating whether new vulnerabilities have been recently discovered related to your systems.

Misconfigured devices

Attackers will take every advantage possible when attempting to violate a target organization. This includes seeking out misconfigured devices to be used as a point of intrusion. A misconfigured device may interfere with normal communications or may allow for security breaches. Troubleshooting misconfigured devices should include evaluating the current settings against the documented settings baseline, checking access logs for recent use or modification, and reviewing vulnerability disclosures for new concerns.

Firewall

A misconfigured firewall may allow communications that were intended to be blocked to cross a network boundary. It is important to carefully review firewall rules to prevent any loopholes from emerging due to complex and conflicting filter entries. Third-party evaluation tools are available that can be used to find mistakes in firewall rule sets.

Other firewall configuration mistakes include not keeping current on updates and patches and failing to manage access to the management interface. Always review and update firewalls promptly whenever a new update is released from the vendor. This will minimize the number of known and exposed vulnerabilities. Always change the default password, but be sure not to use something simple or use the same password across multiple devices. Disable plain text access to the management interface and require encrypted connections. Be sure that only internal systems can initiate connections to the management interface, and block any WAN interface attempts to access the management interface.

Content filter

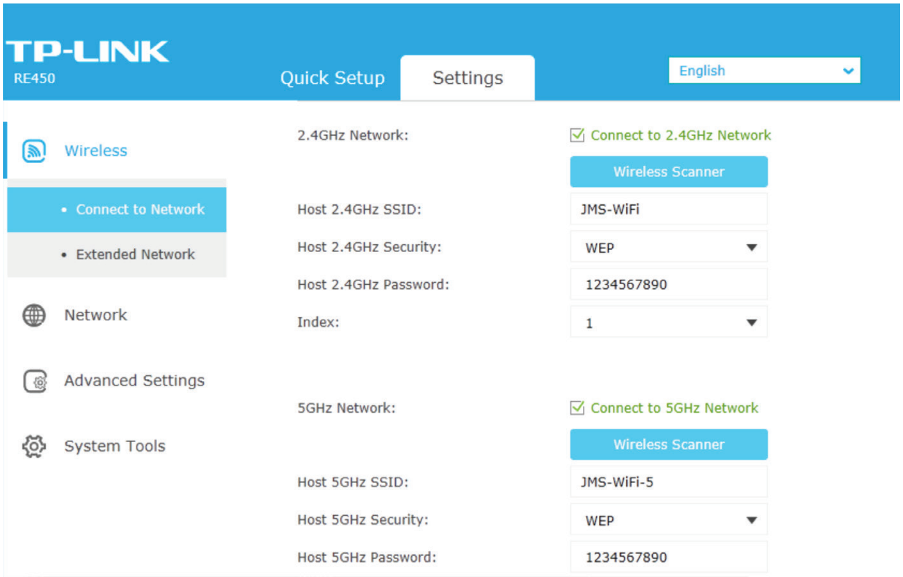
A content filter can fail when it is not properly or thoroughly checking communications. A content filter should be positioned in a network architecture where it is able to gain access to the plain text payload of the application protocol. Otherwise, if the content filter is unable to view the application protocol payload or the payload is encrypted, the filter will not be properly applied. It is also possible to bypass content filters using alternate encoding techniques, such as Hex or Unicode. Be sure that the content filter is checking not just for direct specific ASCII matches, but also for processed results.

A common oversight in content filtering is to fail to escape metacharacters (see Chapter 1). Be sure that in addition to blocking content that is too long or that matches a known unwanted data set, your content filtering escapes metacharacters so that their programmatic power is removed.

Access points

A misconfigured wireless access point is a popular target for attackers seeking to gain access into a private network. Common problems include running out-of-date firmware, leaving default configurations in place, not securing access to the management console, and failing to implement strong authentication and encryption (Figure 2.34). These issues are resolved by installing current firmware, customizing settings (especially security), locking down management interface access to require encryption and strong authentication from a wired connection, and using WPA-2 encryption either with a long and complex PER/PSK password or by leveraging robust multifactor authentication from a network-accessible AAA service using ENT/IEEE 802.1x.

FIGURE 2.34 A WiFi access point with poor security configuration



2.4GHz Network: ☑ Connect to 2.4GHz Network

Host 2.4GHz SSID: JMS-WiFi

Host 2.4GHz Security: WEP

Host 2.4GHz Password: 1234567890

Index: 1

5GHz Network: ☑ Connect to 5GHz Network

Host 5GHz SSID: JMS-WiFi-5

Host 5GHz Security: WEP

Host 5GHz Password: 1234567890

For more information on wireless security, please see the wireless sections of Chapter 3, “Given a scenario, implement secure network architecture concepts” and Chapter 6, “Given a scenario, install and configure wireless security settings.”

Weak security configurations

Poor or weak security configurations are to be avoided. Old or nonstandard compliance features should be disabled and replaced with current standards-compliant security settings. This may require upgrading equipment, updating firmware, and electing to set more robust configurations that might block older systems from being able to gain access.

Troubleshooting weak security configurations should occur on a regular basis. Every IT device should be evaluated as to its current and preferred configuration state. IT administrators should review baseline security recommendations as well as vulnerability disclosures. Keep in mind that default settings are never secure settings.

Personnel issues

People are always the weakest link in security because they can make mistakes, be fooled into causing harm, or intentionally violate company security. It is important to consider the risks that personnel represent to your organization and implement security strategies to minimize and handle those risks.

Troubleshooting personnel issues should include verifying that all personnel have attended awareness training on standard minimum security activities and requirements, evaluating the use and activity logs of personnel, and determining whether the violation was intentional, coerced, accidental, or due to ignorance.

Policy violation

A policy violation occurs when a user breaks a rule. Users need to be trained on the security policies of the organization and know their specific responsibilities with regard to abiding by security rules. If a violation occurs, an internal investigation should evaluate whether it was an accident or an intentional event. If accidental, the worker needs to be trained on how to avoid the accident in the future, and new countermeasures may need to be implemented. If intentional, the severity of the issue may dictate a range of responses, including retraining, reassignment, and termination.

An example of a policy violation could be the distribution of an internal company memo to external entities via a social network posting. Depending on the content of the memo, this could be a minor violation (such as posting a memo due to hilarious or pointless content according to the worker) or a major issue (such as posting a memo that discloses a company secret).

Insider threat

An insider threat is someone on the inside of your organization who is violating the company security policy. Once an insider threat is identified, they need to be removed from the organization. If necessary, contact law enforcement to file criminal charges. Any resources accessed by the threat agent should be evaluated and re-secured.

An example of an insider threat is when a worker purposely brings malicious code into the building on a USB drive in order to infect the network. Such malware might be destructive, or it may grant remote-control backdoor access to an external entity (whether or not that was the insider's intention).

Social engineering

Social engineering attacks can range from email communications to face-to-face encounters. Whenever a security breach occurs, an investigation should be performed to determine what was affected and whether the attack is ongoing. Personnel should be retrained to detect and avoid social engineering attacks in the future. If the attack resulted in data leakage or the attacker gaining remote access, those issues need to be addressed. For a data leakage event, the value and risk of the leaked data must be assessed and an appropriate response crafted. For a remote access event, the connection needs to be terminated, any malware removed, and additional defenses installed to prevent the same remote access event from taking place again.

An example of social engineering is when a worker opens an email attachment that was crafted by an attacker to seem like legitimate business communications but was simply a ruse to trick the victim into installing a remote-control service.

Social media

Social media can be a distraction as well as a potential vulnerability to an organization. Workers can easily waste time and system resources by interacting with social media when that task is not part of their job description. The company's acceptable user policy (AUP) should indicate that workers need to focus on work while at work rather than spending time on personal or non–work-related tasks.

Social media can be a means by which workers intentionally or accidentally distribute internal, confidential, proprietary, or PII data to outsiders. This may be accomplished by typing in messages or participating in chats in which they reveal information that they should have kept secret. This can also be accomplished by distributing or publishing documents from internal file stores. Often social media is a tool used by attackers to initiate or further a social engineering attack.

Responses to social media issues can be to block access to social media sites by adding IP blocks to firewalls and resolution filters to DNS. Violating workers need to be reprimanded or even terminated.

An example of social media abuse is when a worker wastes time on a social media site or app rather than accomplishing their work tasks.

Personal email

Personal email can also serve as a distraction, a means to disclose data to outsiders, or a method by which malware infection can occur. In addition to the steps discussed in the previous sections, it may be necessary to block access to personal email on company equipment.

Unauthorized software

Unauthorized software can be a cause of malware infection or a violation of use licenses. Workers should not be given authority to install software of their choosing; instead users should only be able to use software installed by system administrators. Stand-alone or portable programs can be limited by using whitelisting so that only preapproved executables are allowed to function on a system.

If unauthorized software is discovered on a system, determine who installed the application, and whether it is one of the following:

- A legitimate application useful for work tasks
- Potentially malicious
- Just not work-related

The person should be reprimanded and potentially fired if they have repeatedly violated company policy. If the user circumvented software installation prevention measures, then reinforce those security measures or supplement them with more restrictive prevention techniques.

An example of unauthorized software is when a worker monitors the network in order to collect credentials, PII, or other sensitive data by installing a network sniffer. Troubleshooting unauthorized software should include implementing a whitelisting policy that prohibits the installation or execution of unauthorized code, monitoring execution activity of workers, and tracking abnormal network communications back to their system of origin (which can indicate the use of unauthorized software).

Baseline deviation

All company systems should be operating within expected parameters and compliant with a defined baseline. If a system is determined to be out of baseline, then the system should be removed from the production network in order to investigate the cause. If the deviation was caused by a malicious event, then investigate and respond as discussed in earlier sections. If the deviation was due to normal work-related actions and activities, it may be necessary to update the baseline and/or implement more restrictive system modification policies, such as whitelisting or using static systems. A static system is an environment where users can make either no changes or only temporary and minor changes that are discarded once the user logs out.

License compliance violation (availability/integrity)

License compliance is important to an organization to avoid legal complications. All software in use on company equipment needs to be used in accordance with its license. If software is discovered that is not properly licensed, it should be removed immediately. An investigation should determine how the software made its way onto the system. If the software is needed for a business task, then a proper and valid license should be obtained before reinstalling it.

One common license compliance violation is to purchase a specific number of installation or use licenses for a software product but then accidentally or intentionally install more versions than were licensed. This might be seen as a means to support availability of a business task or resource, but it is at the cost of the integrity of the organization.

Asset management

Asset management is the process of keeping track of the hardware and software implemented by an organization. This management process is used to ensure that updates, revisions, replacements, and upgrades are properly implemented as well as to make sure that all company assets are accounted for. If asset management fails, new equipment may be obtained unnecessarily as sufficient equipment is on premises, but not inventoried properly.

This could result in loss, theft, or mistakenly discarding equipment misidentified as excess or old that is actually needed for business tasks.

On a regular basis, maybe quarterly or yearly, a manual inventory should be performed in order to compare and adjust any automated or digital inventory and asset management system. If the process shows reliable asset management, the frequency of manual verification can be relaxed.

Authentication issues

Authentication is a key element in system security. Authentication is the first element of AAA services, which also include authorization and accounting. Without reliable authentication, it is not possible to hold users accountable for their actions.

Authentication issues include when user credentials are violated, when a user is impersonated, or when a user is unable to log in. If user credentials are known to have been violated, such as when a user database has been remotely accessed by attackers or user credentials were transmitted in clear text, then all user credentials need to be invalidated and reset.

If a user has been impersonated, the account should be disabled during the investigation. If the issue can be resolved, the credentials on the account can be reset and use returned to the original user. If the violation was severe or criminally related, keep the violated account disabled and create a new account with robust credentials for the user.

If a user was unable to log in, this could be due to an authentication service failure or a communications issue. Rebooting and/or resetting systems should resolve minor problems. The user may have provided the wrong credentials, forgotten them, or attempted to use previous credentials. This may require that the credentials be reset and that the user attempt authorization again. If the user has never attempted authentication from a specific system, check for compatibility issues. If the authentication failure is a new issue, look for anything that may have changed since the last successful logon. Changes to the system, whether intentional and approved or accidental, may be the cause of the problem. Reversing changes or reconfiguring the system may be needed to restore authentication function for the user. It may also be helpful to determine whether the issue is unique to one user, or some or all other users are affected.

Exam Essentials

Know the issue of unencrypted credentials. It is no longer an acceptable practice to allow authentication to take place over a plain-text or clear-text communication channel. All authentication, without exception, should be encrypted.

Understand access violations. An access violation can be either an unauthorized logon event or an unauthorized resource access event.

Comprehend troubleshooting certificate issues. Certificate issues can be related to a wide range of potential misconfigurations, policy violations, or missing information.

Understand data exfiltration. When data exfiltration occurs, an outsider or unauthorized entity has gained access to internal data. This is a data loss or data leakage event.

Know about personnel issues. People are always the weakest link in security—they can make mistakes, be fooled into causing harm, or intentionally violate company security.

Understand the issue of unauthorized software. Unauthorized software can be a cause of malware infection or a violation of use licenses.

Understand the issue of baseline deviation. All company systems should be operating within expected parameters and compliant with a defined baseline. If a system is determined to be out of baseline, the system should be removed from the production network in order to investigate the cause.

Be aware of the issue of license compliance violation. License compliance is important to an organization in order to avoid legal complications. All software in use on company equipment needs to be used in accordance with its license.

2.4 Given a scenario, analyze and interpret output from security technologies.

Security management includes responding to violations or alerts. Knowing how to respond to the various issues that are detected by your security infrastructure is important to minimize downtime and data loss. This section focuses on the various security tools that may produce output related to security violations.

HIDS/HIPS

A host-based IDS (HIDS) monitors a local machine for symptoms of unwanted activity. See the discussion of HIDS/HIPS earlier in this chapter in the section “NIPS/NIDS.”

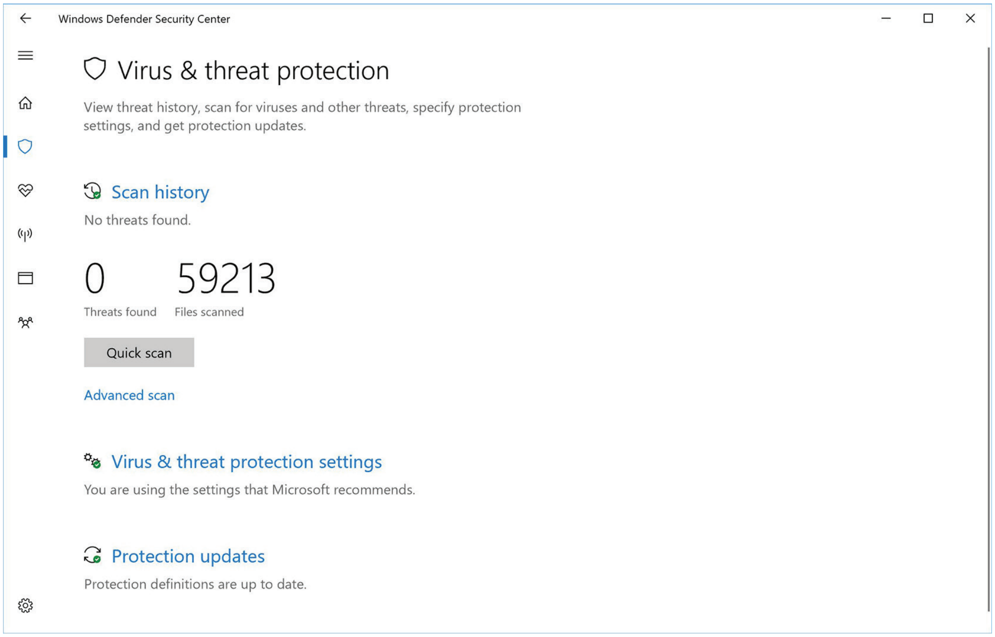
The output of a HIDS or HIPS can include alerts or reports. An alert is an indicator of an immediate event that has just occurred or is continuing to occur. After reviewing the contents of the alert, the administrator should formulate a response to address the issue promptly with a focus on minimizing further harm. A report is a record of alerts and other information related to the time frame of monitoring and the systems being monitored. These reports should be reviewed regularly. If serious issues are discovered in a report, actions should be taken to stop further harm and prevention technologies should be implemented to stop a recurrence.

If an intrusion event has taken place, there may be a record of it. By analyzing and interpreting the output of a HIDS/HIPS, we can determine whether an intrusion was attempted, whether the attack was successful, and what systems were targeted by the attacker.

Antivirus

Antivirus software is an essential security application (Figure 2.35). It's one example of a host IDS. It monitors the local system for evidence of malware in memory, in active processes, and in storage. Most antivirus products can remove detected malicious code and repair most damage it causes.

FIGURE 2.35 An antivirus application status page



Windows Defender Security Center

Virus & threat protection

View threat history, scan for viruses and other threats, specify protection settings, and get protection updates.

Scan history

No threats found.

0 Threats found

59213 Files scanned

Quick scan

Advanced scan

Virus & threat protection settings

You are using the settings that Microsoft recommends.

Protection updates

Protection definitions are up to date.

In order for antivirus software to be effective, it must be kept current with daily signature-database updates. It's also important to use the most recent engine, because new methods of detection and removal are found only in the most current versions of antivirus software.

The output from an antivirus or antimalware product is an alert or alarm when malware is discovered by the live system monitor, or a report if discovered by the systemwide file scan. A live malware event may cause the antivirus product to respond automatically or prompt the user. Responses may include malware removal or quarantine. Removal of infected files may result in lost data, and quarantine may provide an option for removing the malware elements while retaining the data. If a backup exists, then removal may be preferred, but if there is no backup of the infected files, quarantine is preferred.

Antispam software is a variation on the theme of antivirus software. It specifically monitors email communications for spam and other forms of unwanted email in order to stop hoaxes, identity theft, waste of resources, and possible distribution of malicious software. Some antivirus software products include an antispam component. An antispam product may produce a brief report of the messages it discarded or quarantined. However, in most cases the user will need to view the contents of the quarantined spam folder to see what was identified as suspicious.

Spyware monitors your actions and transmits important details to a remote system that spies on your activity. For example, spyware might wait for you to log in to a banking website and then transmit your username and password to the creator of the spyware. Alternatively, it might wait for you to enter your credit card number on an e-commerce site and then transmit the number to a fraudster to resell on the black market.

Adware, although quite similar to spyware in form, has a different purpose. It uses a variety of techniques to display advertisements on infected computers. The simplest forms of adware display pop-up ads on your screen while you surf the Web. More nefarious versions may monitor your shopping behavior and redirect you to competitor websites.

In both cases, you need an antispyware scanner to detect, remove, and repel spyware and adware. Some antivirus products include antispyware features. However, it may be a good idea to run an antispyware scanner that comes from a vendor different from the antivirus scanner vendor. The output of antispyware and anti-adware products is very similar to that of antivirus products.

NOTE

Pop-up blockers are used to prevent websites from opening additional web browser windows without your consent. Often these pop-up windows are used for advertisements or possibly to distribute malicious code or interact with questionable content. Pop-up blockers simply prevent active web browser processes or code from websites from launching or initiating new windows. There is usually an easy bypass for those times when you want to allow pop-ups; one common bypass is to hold down the Ctrl key while the pop-up opens. Pop-up blockers are common components of modern web browsers, but they may also be part of antivirus software or standalone third-party applications.

When a system is infected by a known malware, the antivirus should detect the unwanted code and initiate either a removal or a quarantining of the offending file(s). The output from an antivirus tool can be analyzed and interpreted in order to understand what specific form of malware was detected and the response performed by the protection software.

File integrity check

File integrity checking is the activity of comparing the current hash of a file to the stored/previous hash of a file. A file integrity checking utility will either display an alert or produce a report of the files that do not pass their hash-based integrity check. When a file's integrity is violated, the response should be to replace the file with a valid version from backup. Review the log files to determine the source of the change, and then take appropriate action to prevent the reoccurrence of the integrity violation.

The output of a file integrity check utility can be analyzed and interpreted in order to understand which files did not have integrity. With this knowledge, it may be possible to review file change logs to determine when the files were modified and what person or software performed the modifications.

Host-based firewall

A host-based or personal software firewall is a security application that is installed on client systems. A client firewall is used to provide protection for the client system from the activities of the user and from communications from the network or Internet. A personal firewall must be kept current with patches and updates. It can often limit communications to approved applications and protocols and can usually prevent externally initiated connections.

The output of a host-based firewall may be to prompt the user whether or not to grant outbound communication privileges to a software program or alert the user of an attempt to violate existing inbound and outbound firewall rules. If a valid program is requesting network access, it can be granted. But if network access is not authorized or the program is unknown, this request should be denied.

Analyzing and interpreting the output of a host-based firewall can determine if intrusion or DoS attacks were attempted against the host or if host software attempted to egress the system to attack other systems.

Application whitelisting

Application whitelisting is a security option that prohibits unauthorized software from executing. Whitelisting is also known as *deny by default* or *implicit deny*. In application security, whitelisting prevents any software, including malware, from executing unless it's on the preapproved exception list: the whitelist. This is a significant departure from the typical device-security stance, which is to allow by default and deny by exception (also known as *blacklisting*).

Due to the growth of malware, an application whitelisting approach is one of the few options remaining that shows real promise in protecting devices and data. However, no security solution is perfect, including whitelisting. All known whitelisting solutions can be circumvented with kernel-level vulnerabilities and application configuration issues.

A whitelisting solution may produce an output report detailing the attempts to launch unapproved software and a record of each approved software's execution. This report can be used to determine whether additional software needs to be added to the whitelist or its existing approved applications should be reconsidered.

Application whitelisting may produce logs regarding the attempts of users to execute software that is not included on the approved list. Analyzing and interpreting this output can help determine whether additional software needs to be approved or whether users are attempting to perform unauthorized tasks for personal benefit or attempt work tasks for which they are not trained, skilled, or authorized.

Removable media control

Removable media drives, and removable storage in general, are considered both a convenience and a security vulnerability. The ability to add storage media to and remove it from a computer system makes it more versatile. However, using removable media also makes the hosted content vulnerable to data theft and malicious code planting.

Removable media include the electronic, logical, and digital storage mechanisms listed in the following sections as well as printed materials. When media are no longer needed, they should be properly destroyed to prevent disclosure of sensitive and confidential information to unauthorized entities. For example, failing to destroy printouts or burned CDs may provide dumpster-diving attackers with treasures.

Tape is a removable medium commonly used for backup purposes. It's a form of sequential storage, so data elements are written and read in sequential order rather than semirandomly as with hard drives. Tape media often support larger storage capacities than most removable media, excluding hard drives. This makes them suited for backup operations.

Recordable compact disks (CD-Rs) include the wide range of optical media that can be written to. These include CDR, CD-RW, DVD-R, DVD-RW, Blu-Ray disc recordable (BD-R), and numerous other variants. Writable CDs and DVDs are often inappropriate for network backups due to their size (a maximum of 650 MB for CD-R/RW and 4 GB or more for DVD-R/RW), but they're useful for personal (home) or client-level backups. BD-Rs have a capacity of 25 GB to 50 GB, which can prove useful in some environments (such as SoHo), but they aren't a widely implemented solution. Regardless, the data on a CD isn't protected and thus is vulnerable to unauthorized access if you don't maintain physical control over the media.

Hard drives are usually thought of as a computer's permanent internal storage device. This is true, but hard drives are also available in removable formats. These include hard drives that are plugged into the case or attached by SCSI, eSATA, USB, or IEEE 1394 (FireWire) connections with their own external power-supply connections.

Diskettes, or *floppies*, are removable media that can store only a small amount of data (about 1.4 MB). However, even though they're small, they represent a significant security threat to a protected environment if they get into the wrong hands—not to mention the possibility that they can be used to introduce malware onto a system. Although this type of storage media is becoming less common, it is still a security concern when present.

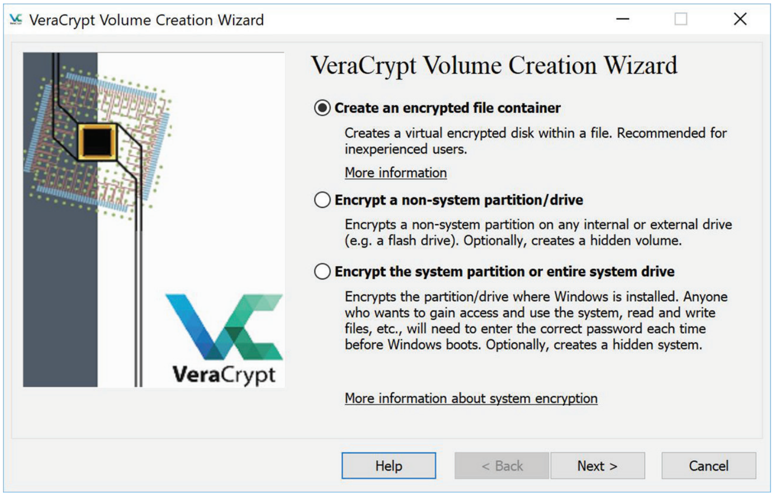
A *flashcard*, or memory card, is a form of storage that uses EEPROM or NVRAM memory chips in a small-form-factor case. Flashcards often use USB connectors or are themselves inserted into devices, such as MP3 players and digital cameras. Some flashcards are almost as small as a quarter and are therefore easy to conceal.

Smartcards can be used for a wide variety of purposes. They can be used as an authentication factor (specifically, as a Type 2 authentication factor, commonly known as *something you have*). When used as such, the smartcard hosts a memory chip that stores a password, PIN, certificate, private key, or digital signature. The authentication system uses this stored data item to verify a user's identity. Smartcards are used as an authentication mechanism by networks, portable computers, PDAs, satellite phones, Public Key Infrastructure (PKI) devices, and more. A smartcard can even function as a credit card (like the American Express Blue card).

A smartcard can also be used as a storage device. Most smartcards have a limited amount of storage, but sometimes, being able to move a few kilobytes of data is all someone needs to steal something of great value. Account numbers, credit card numbers, and a user's private key are all small items that can be very valuable.

Any removable media can typically be secured using file-by-file encryption or whole-drive encryption (Figure 2.36). This may let you move the media from place to place with reasonable assurance that the stored data can't be easily accessed if lost or stolen.

FIGURE 2.36 VeraCrypt, a drive encryption tool



VeraCrypt Volume Creation Wizard

VeraCrypt Volume Creation Wizard

● **Create an encrypted file container**

Creates a virtual encrypted disk within a file. Recommended for inexperienced users.

More information

○ **Encrypt a non-system partition/drive**

Encrypts a non-system partition on any internal or external drive (e.g. a flash drive). Optionally, creates a hidden volume.

○ **Encrypt the system partition or entire system drive**

Encrypts the partition/drive where Windows is installed. Anyone who wants to gain access and use the system, read and write files, etc., will need to enter the correct password each time before Windows boots. Optionally, creates a hidden system.

More information about system encryption

Removable media controls are used to prohibit and monitor the use of portable storage devices. These tools should create a log of the user of valid and approved devices as well as any attempts to use unauthorized devices. Such controls may create audit logs of allowed and attempted/denied use of various media. These logs can be analyzed and interpreted in order to track down users who are violating company policy or attempting to perform unauthorized activities, such as data leakage or inappropriate data storage or sharing.

Advanced malware tools

Advanced malware tools may relate to scanners that include ransomware, rootkits, and potentially unwanted programs (PUPs) in their detection database. A PUP can include any type of questionable software, such as sniffers, password crackers, network mappers, port scanners, and vulnerability scanners. Although these are all legitimate applications for authorized administrators, they are unlikely to be approved tools for standard users.

When an advanced malware tool detects an unapproved executable, it will alert the administrator. The tool should also initiate an automated removal or quarantine whenever possible. In some cases, automated removal is not possible, so an administrator will have to address the concern manually. The specific steps to be performed are based on the type of malicious code or unwanted software discovered and what effect it has had on the system so far. In the worst-case scenarios, the current system will be removed from the production network, drives may be replaced, and a new secure and safe system image will be restored.

The output of advanced malware tools can be analyzed and interpreted to discover business tasks that are exposing the organization to infection as well as to pinpoint which users are performing risky behaviors.

Patch management tools

Patch management is the formal process of ensuring that updates and patches are properly tested and applied to production systems. Security is always a moving target. A system that is secure today may be vulnerable tomorrow. New methods of attacks, new attack tools, new viruses, new weaknesses, accidents in your environment, and much more can cause new risks, threats, and vulnerabilities at any time. Staying vigilant in the face of new security issues is essential in today's business environment. One method for staying as secure as possible is to install updates from vendors.

Using vendor *updates* to OSs, applications, services, protocols, device drivers, and any other software is the absolute best way to protect your environment from known attacks and vulnerabilities. Not all vendor updates are security related, but any error, bug, or flaw that can be exploited to result in damaged data, disclosure of information, or obstructed access to resources should be addressed.

The best way to keep your systems updated is by using a good patch-management system that includes the following steps:

1. Watch vendor websites for information about updates.
2. Sign up for newsletters, discussion groups, or notifications.
3. Download all updates as they're made available. Be sure to verify all downloads against the vendor-provided hashes.
4. Test all updates on nonproduction systems.
5. Document changes to your test systems, and plan the implementation on production systems.
6. Back up production systems before implementing updates.
7. Implement updates on production systems.
8. Evaluate the effect of the updates on the production systems.
9. If negative effects are discovered, roll back the update.

Patch management can be implemented via a manual process, or you can use an intelligent software tool to automate this essential activity. An example of intelligent patch-management software for Windows environments is Microsoft's Windows Server Update Services (WSUS) software. WSUS provides administrators with a centralized means of patch management, distribution, and installation. There are similar product solutions for other OSs and mixed-OS environments. Although security involves more than just patch management, security management requires that patches and updates are properly installed.

A *hotfix* is often a single-issue update (however, there are some multi-issue hotfixes) that corrects a single problem. Hotfixes aren't as thoroughly tested as other updates—they're quickly designed and released to deal with immediate issues and problems. You should install them if you're experiencing the problem they're designed to correct or if you're threatened by the vulnerability they're designed to address.

Service packs are collections of hotfixes and other previously unreleased updates and features as a single entity. They're thoroughly tested and generally should be applied to all systems once they're made available. Service packs may be cumulative, so you need to apply only the most recent service pack to keep your systems current. When a service pack isn't cumulative, it requires a specific base level of previous patches before it can be applied.

A *patch* is an update that corrects programming flaws that cause security vulnerabilities. Patches are single-issue utilities that are more thoroughly tested than hotfixes.

The output from a patch management tool will be a report indicating that monitored systems are or are not in compliance with the approved updates. Some patch management systems may be able to automatically apply approved patches, whereas others require the administrator to install updates manually.

UTM

An all-in-one security appliance is a hardware device designed to operate inline between an Internet connection and a network. Its goal is to detect and filter all manner of malicious, wasteful, or otherwise unwanted traffic. These devices can be called security gateways or *unified threat management* (UTM) systems. They're implemented to perform firewall, IDS, IPS, and NATing functions and to provide DoS protection, spam filtering, virus scanning, privacy protection, web filtering, spyware blocking, and activity tracking. Some all-in-one security appliances also provide server-side services for hosting web applications and wireless security features.

For some organizations, a single product that provides so many features is a cost-saving measure. In other environments, especially larger enterprises, it may not be the optimum choice.

Since a UTM is a combination of products, the output of the UTM will either be unique reports from each of the subfeatures or a single report with an amalgamation of results from all tools. Responses to the UTM report should be based on the specific item discovered and will follow the same procedures as discussed earlier in this section.

DLP

Data loss prevention (DLP) is the system designed to reduce and/or prevent data loss or data leakage to external unauthorized entities. If a violation of DLP occurs, its report should indicate the data that was involved, the user(s) related to the breach, and the applications involved in the exfiltration.

The response to a DLP issue is to evaluate the value of the leaked asset to determine the severity or priority of the response. It may dictate that legal proceedings be started against the internal and external entities involved with the violation. The user involved with violating DLP restrictions may need to be retrained, have their job privileges reduced, and potentially be terminated. The applications and systems that enabled the loss event to occur need to be adjusted to block the reoccurrence of the violation. This might be an indicator that the existing DLP solution is insufficient and needs additional buttressing.

Data execution prevention

Data execution prevention (DEP) is a memory security feature of many operating systems aimed at blocking a range of memory abuse attacks, including buffer overflows. DEP blocks the execution of code stored in areas of memory designated as data-only areas. However, DEP is not foolproof and some forms of buffer overflow attacks are unhampered by it.

When DEP fails, there may not be any specific official output, log, or report of the situation. Some DEP solutions may create a memory abuse attempt log, but it may not record events that were designed to specifically violate the DEP protections. Often the only result is when an administrator happens to notice that malware or unauthorized code is executing on a system where DEP was present. The response should be to terminate the offending code and remove it from the system. If the means by which the code gained execution access to the system is determined, then additional patches or filters should be installed to prevent the exploitation of the same process.

Web application firewall

A *web application firewall* is a device, server add-on, virtual service, or system filter that defines a strict set of communication rules for a website and all visitors. It's intended to be an application-specific firewall to prevent cross-site scripting, SQL injection, and other web application attacks.

A related device is the *web security gateway*, which is a web-content filter (often URL and content keyword–based) that also supports malware scanning. In most cases, a web security gateway is implemented by an organization to provide better enforcement of employee web activity policies. Some web security gateways incorporate non-web features as well, including instant messaging (IM) filtering, email filtering, spam blocking, and spoofing detection.

URL filtering, also known as web filtering, is the act of blocking access to a site based on all or part of the URL used to request access. URL filtering can focus on all or part of a fully qualified domain name (FQDN), specific path names, specific filenames, specific file extensions, or entire specific URLs. Many URL-filtering tools can obtain updated master URL block lists from vendors as well as allow administrators to add or remove URLs from a custom list.

Content inspection is the security-filtering function in which the contents of the application protocol payload are inspected. Often such inspection is based on keyword matching. A master blacklist of unwanted terms, addresses, or URLs is used to control what is or isn't allowed to reach a user.

Malware inspection is the use of a malware scanner (also known as an antivirus scanner or spyware scanner) to detect unwanted software content in network traffic. If malware is detected, it can be blocked or logged and/or trigger an alert.

Many firewalls, especially application firewalls and proxies, include URL filtering, content inspection, and malware inspection as additional security features.

Application-aware devices are security devices, such as firewalls, IDSs, IPSs, and proxies, that operate at the higher layers of the protocol stack in order to provide focused security filtering and analysis of the content of specific communications. Such devices are designed around a specific application or service, such as the Web, email, IM, file transfers, database interactions, and so on. Often, application-aware devices are able to provide deep content inspection and filtering based on their focus on specific applications and protocols.

The output of a web application firewall or web security gateway is similar to that of a firewall, IDS, or UTM. The output may be a real-time alert or an after-the-fact report. The output should be evaluated and appropriate responses determined, as discussed in prior sections of this chapter.

Exam Essentials

Understand antivirus software. Antivirus software is an essential security application. Antivirus software is one example of a host IDS. It monitors the local system for evidence of malware in memory, in active processes, and in storage.

Comprehend file integrity checking. File integrity checking is the activity of comparing the current hash of a file to the stored/previous hash of a file.

Understand host-based firewalls. A host-based or personal software firewall is a security application that is installed on client systems. A client firewall is used to provide protection for the client system from the activities of the user and from communications from the network or Internet.

Know about application whitelisting. Application whitelisting is a security option that prohibits unauthorized software from executing. Whitelisting is also known as deny by default or implicit deny.

Understand removable media control. Removable media drives, and removable storage in general, are considered both a convenience and a security vulnerability. The ability to add storage media to and remove it from a computer system makes it more versatile. However, using removable media also makes the hosted content vulnerable to data theft and malicious code planting.

Understand advanced malware tools. Advanced malware tools may relate to scanners that include ransomware, rootkits, and potentially unwanted programs (PUPs) in their detection database.

Be aware of patch management tools. Patch management is the formal process of ensuring that updates and patches are properly tested and applied to production systems.

Understand UTMs. An all-in-one security appliance or unified threat management (UTM) is a hardware device designed to operate inline between an Internet connection and a network. Its goal is to detect and filter all manner of malicious, wasteful, or otherwise unwanted traffic.

Understand DLP. Data loss prevention (DLP) is the system designed to reduce the occurrence of and/or prevent data loss or data leakage to external unauthorized entities. If a violation of DLP occurs, its report should indicate the data that was involved, the user(s) related to the breach, and the applications involved in the exfiltration.

Know about DEP. Data execution prevention (DEP) is a memory security feature of many operating systems aimed at blocking a range of memory abuse attacks, including buffer overflows. DEP blocks the execution of code stored in areas of memory designated as data-only areas.

Understand web application firewalls. A web application firewall is a device, server add-on, virtual service, or system filter that defines a strict set of communication rules for a website and all visitors. It's intended to be an application-specific firewall to prevent cross-site scripting, SQL injection, and other web application attacks.

2.5 Given a scenario, deploy mobile devices securely.

Mobile devices are a central part of modern life and business operations. Whether mobile devices are brought into the organization by employees or are provided by the company, mobile device security is just as important as the security of mission-critical servers and standard endpoint network access devices.

Connection methods

Mobile devices may support a number of various connection options. These may be network connections that link to an external provider, such as a telco, or the local private network. A basic understanding of each concept is important for the Security+ exam.

For any organization, it is important to consider the scenarios where workers are in need of reliable communications. These may be standard in-office employees, telecommuters, or even those on location at a client's facility. Only consider deploying those services that can provide reliable and secure (encrypted) communications.

Cellular

A cellular network or a wireless network is the primary communications technology that is used by many mobile devices, especially cell phones and smartphones. The network is organized around areas of land called cells, which are centered around a primary transceiver, known as a cell site, cell tower, or base station. Cellular communications can support audio, text, and data transmissions. The services provided over cellular networks are often referred to by a generational code, which is only loosely defined, such as 2G, 3G, and 4G (with 5G just starting to be implemented in 2017). These generational terms are used to refer to the communications technology deployed by each subsequent improvement of the networks. For example, 2G refers to Global System for Mobile Communications (GSM), which is still used to support a majority of audio communications; 3G refers to Universal Mobile Telecommunications System (UMTS); and 4G refers to Long-Term Evolution (LTE).

Generally, cellular service is encrypted, but only while the communication is being transmitted from the mobile device to a transmission tower. Communications are effectively plain text once they are being transmitted over wires. So, avoid performing any task over cellular that is sensitive or confidential in nature. Use an encrypted communications application to pre-encrypt communications before transmitting them over a cellular connection.

WiFi

WiFi or wireless networking was originally defined by the IEEE 802.11 standard. WiFi is a nearly ubiquitous communication scheme available in most homes, offices, and public retail locations, such as restaurants and stores.

More information about wireless networking or WiFi is located earlier in this chapter in the section "Access Points" and in the Chapter 6 section, "Given a scenario, install and configure wireless security settings."

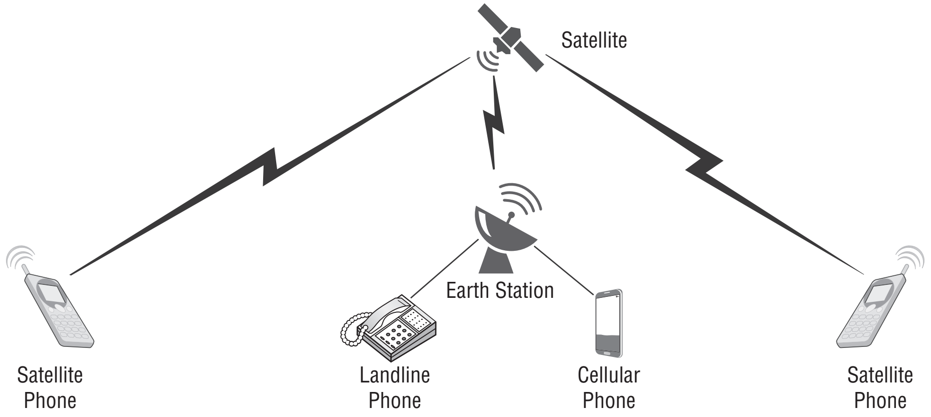
WiFi is not always encrypted, and even when it is, the encryption is only between the portable device and the base station. For end-to-end encryption of communications, use a VPN or an encrypted communications application to pre-encrypt communications before transmitting them over WiFi.

SATCOM

SATCOM, or satellite communication, is a means of audio and data transmission using satellites orbiting in near-earth orbit (Figure 2.37). SATCOM devices benefit from nearly complete service coverage, thanks to the broad footprint of a signal transmitted from 100+ miles above the surface of the planet. The data transmission speeds of SATCOM are rather poor compared to those of terrestrial solutions, but it may be the only available option in many remote locations.

SATCOM communications are encrypted in most cases to prevent eavesdropping from others elsewhere in the transmission footprint of the signal from the satellite. However, the encryption is only applied to the communication between the portable device and the satellite and between the satellite and the ground station. Once the communication reaches the ground station, it is likely then transmitted over a landline, cellular, or data connection in plain text form. Always pre-encrypt any sensitive communications before sending them over a SATCOM connection.

FIGURE 2.37 Satellite communications for voice and data



Bluetooth

Bluetooth is defined in IEEE 802.15 and uses the 2.4 GHz frequency (which is also used by some forms of WiFi). Bluetooth is plain text by default in most implementation and usage scenarios, but can be encrypted with specialty transmitters and peripherals. Bluetooth operates between devices that have been paired, which is a means of loosely associating devices with each other either using a default pair code, often 0000 or 1234, or a random 8-character code displayed on one device that must be typed into the other device. Bluetooth is generally a short distance communication method, but that distance is based on the relative strengths of the paired devices' antennas. Standard or official use of Bluetooth ranges up to 100 meters; 10 meters is most common.

Bluetooth is vulnerable to a wide range of attacks, including bluesniffing, bluesmacking, bluejacking, bluesnarfing, and bluebugging. Please see the Chapter 1 sections "Bluejacking" and "Bluesnarfing."

Since Bluetooth is typically a plain-text communication, do not use it to support sensitive or confidential transactions. Use an alternate means of communications that can provide encrypted transactions. Even if you are using a special implementation of Bluetooth that does encrypt the wireless signal, that encryption ends at the Bluetooth transmitter/receiver device on each end of the wireless signal.

NFC

Near field communication (NFC) is a standard to establish radio communications between devices in close proximity. It lets you perform a type of automatic synchronization and association between devices by touching them together or bringing them within inches of each other. See the Chapter 1 section "NFC" for more.

NFC is designed to be a secure communications system, and its signals are encrypted or encoded in most cases. NFC is not used to support ongoing or large data transmissions, such as WiFi, cellular, or even Bluetooth, so the risks are minimal simply based on its limited data transmission uses.

ANT

ANT is a proprietary protocol owned by Garmin that is an open access multicast sensor network technology. It uses the 2.4 GHz frequency band to support interactions between sensor devices and management devices (such as a smartphone). It is similar in nature to Bluetooth LE (Low Energy), but with a primary focus on gathering data from low-power and low-bit-rate sensors. ANT is found in many fitness trackers, heart rate monitors, watches, cycling meters, and pedometers.

ANT offers the ability to encrypt communications, but it is not always enabled. Some implementations of ANT, such as ANT+, do not offer any encryption options because they focus on cross-vendor interoperability rather than security. Similar to NFC, ANT has limited risk due to its current use limitations. However, always be cautious when using any plain text communications system.

Infrared

Infrared is not as common a communication technology as wireless is for modern devices. However, there are still plenty of infrared implementations; they often revolve around cameras transmitting imagery to printers or storage devices or remote controls of cameras, video systems, A/V systems, and environmental sensors. Infrared is a line-of-sight–based system and can be easily interrupted. Infrared communications are typically in plain text. It is unlikely you will use infrared communications; if you do, however, be cautious of transmitting valuable or sensitive data. Some modern mobile phones continue to include an infrared port for use as a transmitter for controlling televisions and other A/V entertainment equipment.

USB

USB (Universal Serial Bus) is a standard for connecting peripheral devices and primary computers over a wired link. USB is almost always a connection option for devices manufactured since 2000. There are a range of specifications and adapter/connection variations. Although USB is an easy-to-use mechanism for exchanging data between devices, it does not provide any security over the data transfer. Once devices are connected via USB, they typically appear in standard file management tools as USB storage devices, where reading and writing of data can take place. The only real protection provided by USB is that it is a wired connection as opposed to wireless and that an encrypted and screen-locked device is likely to disable the USB port. Only when the screen lock is cleared does the USB port become enabled for data exchange.

Mobile device management concepts

Smartphones and other mobile devices present an ever-increasing security risk as they become more and more capable of interacting with the Internet as well as corporate networks. Mobile devices often support memory cards and can be used to smuggle malicious code into or confidential data out of organizations. Mobile devices often contain sensitive data such as contacts, text messages, email, and possibly notes and documents. The loss or theft of a mobile device could mean the compromise of personal and/or corporate secrets.

Mobile devices are becoming the target of hackers and malicious code. It's important to keep nonessential information off portable devices, run a firewall and antivirus product (if available), and keep the system locked and/or encrypted (if possible).

Many mobile devices also support USB connections to perform synchronization of communications and contacts with desktop and/or notebook computers as well as the transfer of files, documents, music, video, and so on.

Additionally, mobile devices aren't immune to eavesdropping. With the right type of sophisticated equipment, most mobile phone conversations can be tapped into—not to mention the fact that anyone within 15 feet can hear you talking. Be careful what you discuss over a mobile phone, especially when you're in a public place.

A wide range of security features are available on mobile devices. However, support for a feature isn't the same thing as having a feature properly configured and enabled. A security benefit is gained only when the security function is in force. Be sure to check that all desired security features are operating as expected on your device.

When personally owned devices are allowed to enter and leave a secured facility without limitation, oversight, or control, the potential for harm is significant. Most portable electronics, especially mobile phones, audio players, and digital cameras, can be used as storage devices. This can allow malicious code to be brought in or sensitive data secreted out. Additionally, any device with a camera feature can take photographs of sensitive information or locations. A device owned by an individual can be referenced using any of these terms: portable device, mobile device, personal mobile device (PMD), personal electronic device or portable electronic device (PED), and personally owned device (POD).

Mobile device management (MDM) is a software solution to the challenging task of managing the myriad mobile devices that employees use to access company resources. The goals of MDM are to improve security, provide monitoring, enable remote management, and support troubleshooting. Many MDM solutions support a wide range of devices and can operate across many service providers. You can use MDM to push or remove apps, manage data, and enforce configuration settings both over the air (across a carrier network) and over WiFi connections. MDM can be used to manage company-owned devices as well as personally owned devices (such as in a bring-your-own-device [BYOD] environment).

Device security is the range of potential security options or features that may be available for a mobile device. Not all portable electronic devices (PEDs) have good security features. But even if devices have security features, they're of no value unless they're enabled and properly configured. Be sure to consider the security options of a new device before you make a purchase decision.

Application management

Application control or *application management* is a device management solution that limits which applications can be installed onto a device. It can also be used to force specific applications to be installed or to enforce the settings of certain applications, in order to support a security baseline or maintain other forms of compliance. Using application control can often reduce exposure to malicious applications by limiting the user's ability to install apps that come from unknown sources or that offer non–work-related features.

Although security features must be enabled to have any beneficial effect, it's just as important to remove apps and disable features that aren't essential to business tasks or common personal use. The wider the range of enabled features and installed apps, the greater the chance that an exploitation or software flaw will cause harm to the device and/or the data it contains. Following common security practices, such as hardening, reduces the attack surface of mobile devices.

In addition to managing the security of mobile devices, you need to focus on the applications and functions used on those devices. Most of the software security concerns on desktop or notebook systems apply to mobile devices just as much as common-sense security practices do.

Content management

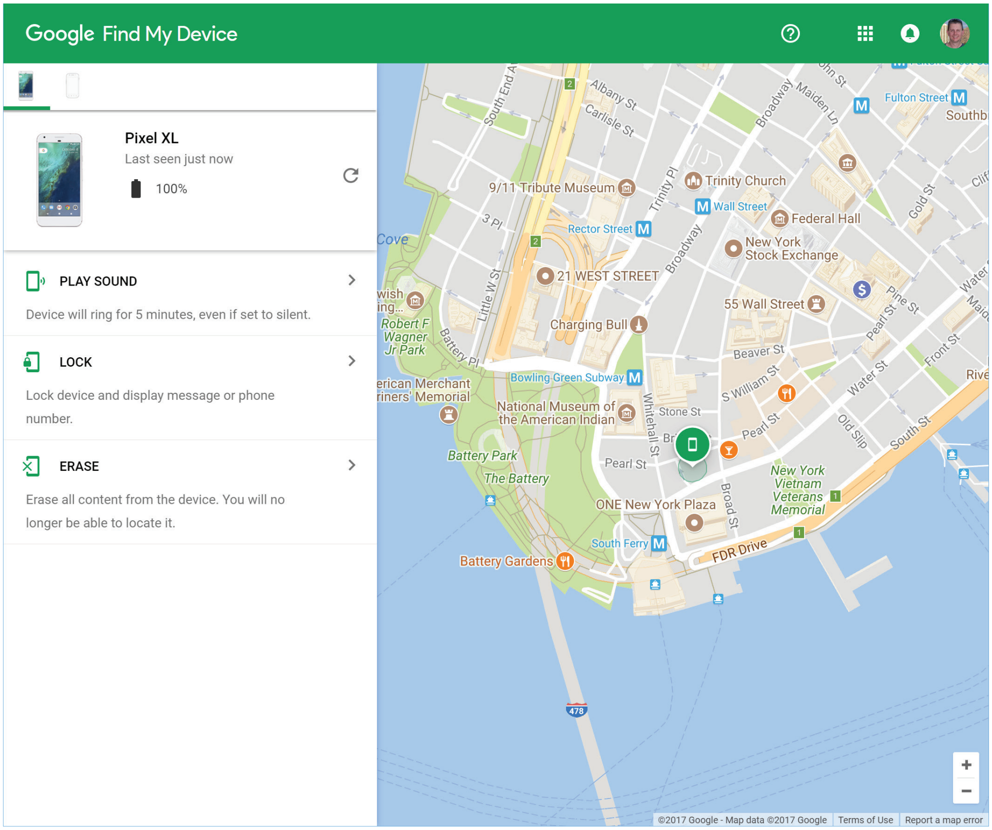
Content management is the control over mobile devices and their access to content hosted on company systems as well as controlling access to company data stored on mobile devices. Typically an MCM (mobile content management) system is used to control company resources and the means by which they are accessed or used on mobile devices. An MCM can take into account a device's capabilities, storage availability, screen size, bandwidth limitations, memory (RAM), and processor capabilities when rendering or sending data to mobile devices.

The goal of a content management system for mobile devices is to maximize performance and work benefit while reducing complexity, confusion, and inconvenience. An MCM may also be tied to an MDM to ensure secure use of company data.

Remote wipe

Remote wipe or remote sanitation is to be performed if a device is lost or stolen. A *remote wipe* lets you delete all data and possibly even configuration settings from a device remotely. The wipe process can be triggered over mobile phone service or sometimes over any Internet connection (Figure 2.38). However, a remote wipe isn't a guarantee of data security. Thieves may be smart enough to prevent connections that would trigger the wipe function while they dump out the data. Additionally, the remote wipe is usually just a deletion of user data and resetting the device back to factory conditions. A skilled thief may be able to undelete data files after the wiping process. A way to improve the benefit of remote wipe is to keep the mobile device's storage encrypted. Thus, an undelete operation would only recover encrypted files and most likely not allow the attacker to decode the data.

FIGURE 2.38 The remote wipe (erase) function available through Google Find My Device



Google Find My Device

Pixel XL
Last seen just now
100%

PLAY SOUND

Device will ring for 5 minutes, even if set to silent.

LOCK

Lock device and display message or phone number.

ERASE

Erase all content from the device. You will no longer be able to locate it.

Geofencing

Geofencing is the designation of a specific geographical area that is then used to implement features on mobile devices. A geofence can be defined by GPS coordinates, wireless indoor positioning system (IPS), or presence or lack of a specific wireless signal. A device can be configured to enable or disable features based on a geofenced area. For example, a geofence may trigger a mobile device to disable WiFi and the camera while in a company building, or it might enable mobile payments once the user enters a retail store.

Geolocation

Geolocation or *geotagging* is the ability of a mobile device to include details about its location in any media created by the device. Geolocation data is commonly used in navigation tools and by many location-based services, such as offering discounts or coupons to nearby retail stores.

Many mobile devices include a GPS chip to support and benefit from localized services, such as navigation, so it's possible to track those devices. The GPS chip itself is usually just a receiver of signals from orbiting GPS satellites. However, applications on the mobile device can record the GPS location of the device and then report it to an online service. You can use GPS tracking to monitor your own movements, track the movements of others (such as minors or delivery personnel), or track down a stolen device. But for GPS tracking to work, the mobile device must have Internet or wireless phone service over which to communicate its location information.

Mobile devices with GPS support enable the embedding of geographical location in the form of latitude and longitude as well as date/time information on photos taken with these devices. This allows a would-be attacker (or angry ex) to view photos from social networking or similar sites and determine exactly when and where a photo was taken. This geotagging can be used for nefarious purposes, such as determining when a person normally performs routine activities.

Once a geotagging photo has been uploaded to the Internet, a potential cyberstalker may have access to more information than the uploader intended. This is prime material for security-awareness briefs for end users.

Asset Tracking and Inventory Control

Asset tracking is the management process used to maintain oversight over an inventory, such as deployed mobile devices. An asset-tracking system can be passive or active. Passive systems rely on the asset itself to check in with the management service on a regular basis, or the device is detected as being present in the office each time the employee arrives at work. An active system uses a polling or pushing technology to send out queries to devices in order to elicit a response.

You can use asset tracking to verify that a device is still in the possession of the assigned authorized user. Some asset-tracking solutions can locate missing or stolen devices.

Some asset-tracking solutions expand beyond hardware inventory management and can oversee the installed apps, app usage, stored data, and data access on a device. You can use this type of monitoring to verify compliance with security guidelines or check for exposure of confidential information to unauthorized entities.

The term "inventory control" may describe hardware asset tracking (as discussed in the previous topic). However, it can also refer to the concept of using a mobile device as a means to track inventory in a warehouse or storage cabinet. Most mobile devices have a camera. Using a mobile device camera, apps that can take photos or scan bar codes can be used to track physical goods. Those mobile devices with RFID or NFC capabilities may be able to interact with objects or their containers that have been electronically tagged.

Screen locks

A *screen lock* is designed to prevent someone from casually picking up and being able to use your phone or mobile device. However, most screen locks can be unlocked by swiping a pattern or typing a number on a keypad display. Neither of these is truly a secure operation. Screen locks may have workarounds, such as accessing the phone application through the emergency calling feature. And a screen lock doesn't necessarily protect the device if a hacker connects to it over Bluetooth, wireless, or a USB cable.

Screen locks are often triggered after a timeout period of non-use. Most PCs auto-trigger a password-protected screen saver if the system is left idle for a few minutes. Similarly, many tablets and mobile phones trigger a screen lock and dim or turn off the display after 30–60 seconds. The lockout feature ensures that if you leave your device unattended or it's lost or stolen, it will be difficult for anyone else to be able to access your data or applications. To unlock the device, you must enter a password, code, or PIN; draw a pattern; offer your eyeball or face for recognition; scan your fingerprint; or use a proximity device such as a near-field communication (NFC) or radio-frequency identification (RFID) ring or tile.

Push notification services

Push notification services are able to send information to your device rather than having the device (or its apps) pull information from an online resource. Push notifications are useful in being notified about a concern immediately, but they can also be a nuisance if they are advertising or spam. Many apps and services can be configured to use push and/or pull notifications. Consider the benefits and trade-offs of each application and whether allowing push notifications is worth the distraction.

Passwords and pins

A strong password would be a great idea on a phone or other mobile device if locking the phone provided true security. But most mobile devices aren't secure, so even with a strong password, the device is still accessible over Bluetooth, wireless, or a USB cable. If a specific mobile device blocked access to the device when the system lock was enabled, this would be a worthwhile feature to set to trigger automatically after a period of inactivity or manual initialization. This benefit is usually obtained when you enable both a device password and storage encryption.

You should consider any means that reduces unauthorized access to a mobile device. Many MDM solutions can force screen-lock configuration and prevent a user from disabling the feature.

Authentication on or to a mobile device is often fairly simple, especially for mobile phones and tablets. However, a swipe or pattern access shouldn't be considered true authentication. Whenever possible, use a password, provide a PIN, offer your eyeball or face for recognition, scan your fingerprint, or use a proximity device such as an NFC or RFID ring or tile. These means of device authentication are much more difficult for a thief to bypass. As mentioned previously, it's also prudent to combine device authentication with device encryption to block access to stored information via a connection cable.

Lockout on a mobile device is similar to account lockout on a company workstation. When a user fails to provide their credentials after repeated attempts, the account or device is disabled (locked out) for a period of time or until an administrator clears the lockout flag.

Mobile devices may offer a lockout feature, but it's in use only if a screen lock has been configured. Otherwise, a simple screen swipe to access the device doesn't provide sufficient security, because an authentication process doesn't occur. Some devices trigger ever longer delays between access attempts as a greater number of authentication failures occur. Some devices allow for a set number of attempts (such as three) before triggering a lockout that lasts minutes. Other devices trigger a persistent lockout and require the use of a different account or master password/code to regain access to the device.

Biometrics

Biometrics are a convenient means of authenticating to mobile devices. However, they are not as accurate as we may wish them to be. A password must match exactly; otherwise, an authentication attempt is rejected, but a biometric only has to satisfy an approximation of the reference profile of the stored biometric value. This is why your finger does not have to be oriented in the same way each time, nor does the same exact part of your finger have to be located on the sensor. Even when you train the device for your biometric factor, the device takes numerous samples from your selected body part to create the reference profile. Most of the biometric sensors on mobile devices are rather simple and can be fooled by false versions of the biometric factor. If someone lifts your fingerprint off a smooth surface, like a drinking glass or tabletop, they can create a gummy equivalent. Many facial recognition systems can be fooled by photos of the valid subjects.

Thus, biometrics should not be employed as the only means or mechanism to authenticate to a device. If the device holds highly valuable and sensitive content, then don't use single-factor biometrics. Instead use a biometric only as one element of a multifactor authentication.

If single-factor biometric authentication is desired, configure biometric lockout to engage after two or three failed attempts, and then have the fallback authentication be a long, complex password.

Context-aware authentication

Context-aware authentication is an improvement on traditional authentication means. Contextual authentication evaluates the origin and context of a user's attempt to access a system. If the user originates from a known trusted system, such as a system inside the company facility, then a low-risk context is present and a modest level of authentication is mandated for gaining access. If the context and origin of the user is from an unknown device and/or external/unknown location, the context is high risk. The authentication system will then demand that the user traverse a more complex multifactor authentication gauntlet in order to gain access. Context-aware authentication is thus an adaptive authentication that may be able to reduce the burden of authentication during low-risk scenarios but thwart impersonation attempts during high-risk scenarios.

Containerization

Containerization is the next stage in the evolution of the virtualization trend for both internally hosted systems and cloud providers and services. A virtual machine–based system uses a hypervisor installed onto the bare metal of the host server and then operates a full guest operating system within each virtual machine, and each virtual machine often supports only a single primary application. This is very resource-wasteful design and reveals its origins as separate physical machines.

Containerization is based on the concept of eliminating the duplication of OS elements and removing the hypervisor altogether. Instead, each application is placed into a container that includes only the actual resources needed to support the enclosed application. The containers run on a standard shared operating system. There is effectively a hypervisor replacement, generically known as the container engine, but it consumes far fewer resources than the hypervisor, because it simply facilitates OS resource and service access for the containerized applications. Containerization is able to provide 10 to 100 times more application density per physical server than that provided by hypervisor virtualization solutions.

Containerization can be used in relation to mobile devices by hosting the primary OS on a containerization host in the company cloud so that the actual mobile device is only used as a remote control interface to the OS container rather than having the business apps and company data on the device itself.

Storage segmentation

Storage segmentation is used to artificially compartmentalize various types or values of data on a storage medium. On a mobile device, the device manufacturer and/or the service provider may use storage segmentation to isolate the device's OS and preinstalled apps from user-installed apps and user data. Some mobile device–management systems further impose storage segmentation in order to separate company data and apps from user data and apps. This allows for ownership and rights over user data to be retained by the user, while granting ownership and rights over business data to the organization, even on devices owned by the employee.

Full device encryption

Encryption is often a useful protection mechanism against unauthorized access to data, whether in storage or in transit. Most mobile devices provide some form of storage encryption. When this is available, it should be enabled. However, encryption isn't a guarantee of protection for data, especially if the device is stolen while unlocked or if the system itself has a known backdoor attack vulnerability.

Communication Encryption

Some mobile devices offer native support for communications encryption, but most can run add-on software (apps) that can add encryption to data sessions, voice calls, and/or video conferences.

Voice encryption may be possible on mobile devices when Voice over IP (VOIP) services are used. VOIP service between computer-like devices is more likely to offer an encryption option than VOIP connections to a traditional landline phone or typical mobile phone. When a voice conversation is encrypted, eavesdropping becomes worthless because the contents of the conversation are undecipherable.

Enforcement and monitoring for:

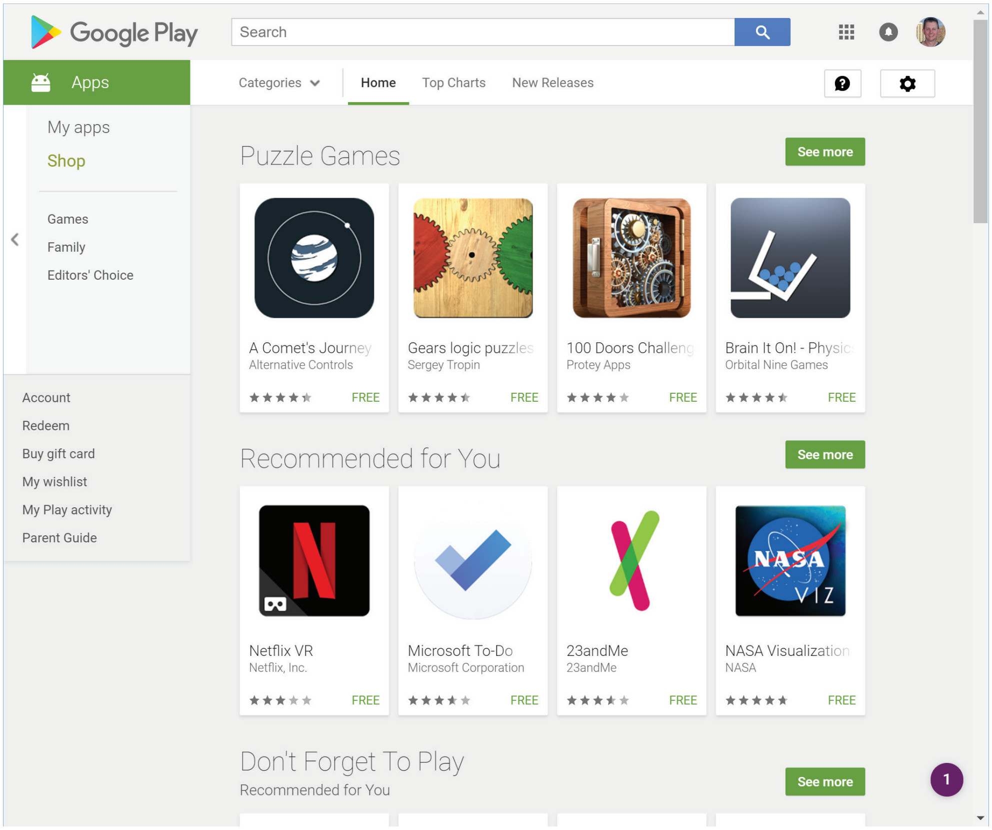
Allowing mobile devices to connect to or interact with company networks and resources puts the organization at greater risk. A company should define mobile device security policies that attempt to address and minimize the security issues related to the following list of concerns.

Third-party app stores

The first-party app stores of Apple iTunes and Google Play (Figure 2.39) are reasonable sources for apps for use on the typical or standard iOS and Android smartphone or device. For Android devices, the second-party Amazon Underground app store is also a worthwhile source of apps. However, most other sources of apps for either smart device platform are labeled as third-party app stores. These app stores often have less rigorous rules regarding hosting an app. On Android devices, simply enabling a single feature to install apps from unknown sources allows the use of the Amazon Underground app store as well as any other non-Google source. For Apple iOS devices, you are limited to the official iTunes App Store unless you jailbreak or root the device (which is not usually a security recommendation).

When a mobile device is being managed by an organization, especially when using an MDM, most third-party sources of apps will be blocked. Such third-party app sources represent a significant increase in risk of data leakage or malware intrusion to an organizational network.

FIGURE 2.39 The Google Play app store



Rooting/jailbreaking

Rooting or jailbreaking (the special term for rooting Apple devices) is the action of breaking the digital rights management (DRM) security on the bootloader of a mobile device in order to be able to operate the device with root or full system privileges. Most devices are locked in such a way as to limit end-user activity to that of a limited user. But a root user can manipulate the OS, enable or disable hardware features, and install software applications that are not available to the limited user. Rooting may enable a user to change the core operating system or operate apps that are unavailable in the standard app stores. However, this is not without its risks. Operating in rooted status also reduces security, since any executable also launches with full root privileges. There are many forms of malicious code that cannot gain footing on normal mode devices but that can easily take root (pun intended) when the user has rooted or jailbroken their device.

An organization should prohibit the use of rooted devices on the company network or even access to company resources whenever possible. For some, a rooted device may provide benefits that exceed the risks, but such devices should be operated as stand-alone equipment and not as endpoint devices of a company network. Also, even users should consider keeping their personal information and credentials on a normal limited device and employ a second rooted device for those limited occasions when rooting is beneficial.

Rooting a fully owned device is legal as of 2017, but the exemption to the Digital Millennium Copyright Act allowing for this will expire at the end of 2017. If other exemptions or new legislation do not reestablish the legality of rooting devices, then it will become illegal again starting in 2018. Be sure to review the legality of rooting devices before you attempt to do so. The legality of the issue may not change your mind about your decision, but you do need to be aware of the legality of the activity and the potential consequences. If you are caught crafting tools to help others root their devices, this is seen as a much more severe activity than rooting your own device.

Even if rooting is legal, keep in mind that there are still some restrictions. First, it is only legal if you fully own the device; if you are in a one- or two-year contract with a hardware fee; or if you are in a lease-to-own contract and you do not fully own the device until that contract is fulfilled. Thus, it is illegal to root until you fully own the device. Second, legal root does not require a manufacturer, vendor, or telco to honor any warranty. In most cases, any form of system tampering, including rooting, voids your warranty. Rooting may also void your support contract or replacement contract. Third, rooting is actively suppressed by the telcos and some product vendors, Apple being the main example. A rooted device might not be allowed to operate over a telco network, or it might be prohibited from accessing resources, downloading apps, or receiving future updates.

Sideloading

Sideloading is the activity of installing an app onto a device by bringing the installer file to the device through some form of file transfer or USB storage method rather than installing from an app store. Sideloading is not possible on Apple iOS devices unless they are jailbroken. Sideloading is possible on Android devices if Install Apps From Unknown Sources is enabled.

Most organizations should prohibit user sideloading, because it may be a means to bypass security restrictions imposed by an app store or the MDM.

Custom firmware

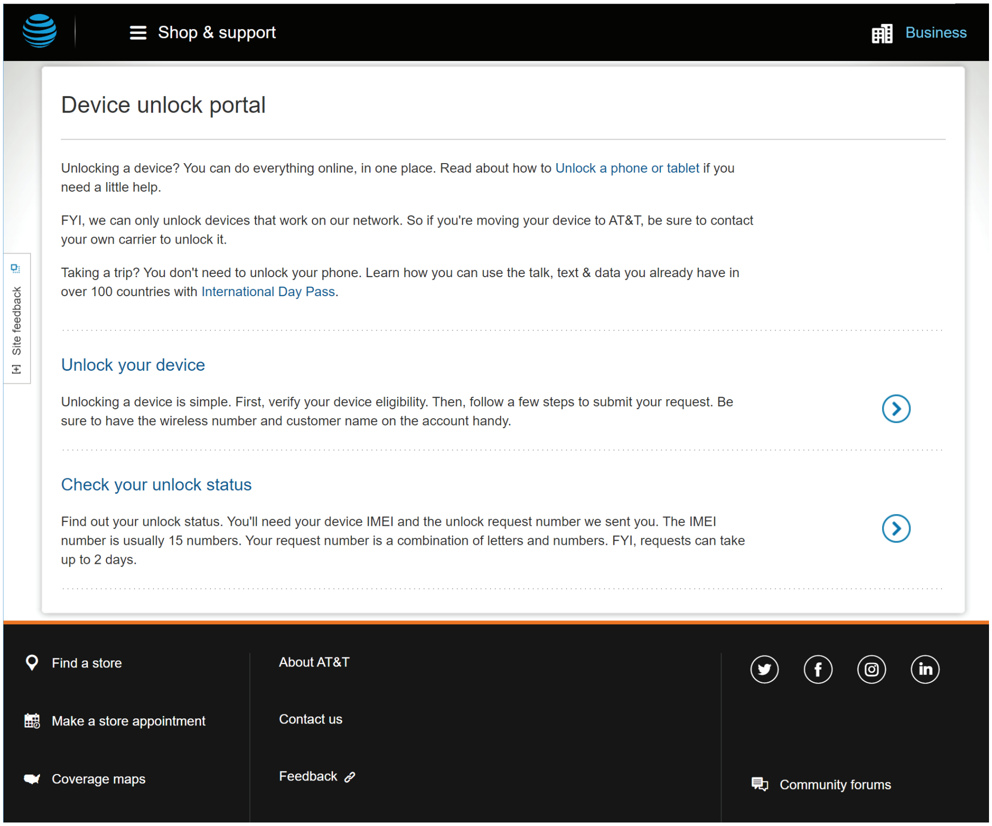
Mobile devices come preinstalled with a vendor- or telco-provided firmware or core operating system. The firmware can be updated by an upgrade provided by the vendor or telco. If a device is rooted or jailbroken, it can allow the user to install alternate custom firmware in place of the default firmware. Custom firmware may remove bloatware included by the vendor or telco, may add or remove features, and can streamline the OS to optimize performance. There are online discussion forums and communities that specialize in custom firmware for Apple and Android devices, such as `xda-developers.com` (be careful and include the dash in the name!) and `howardforums.com`.

An organization should not allow users to operate mobile devices that have custom firmware unless that firmware is preapproved by the organization. Some custom firmware can be returned to a non-rooted state once the firmware install is completed.

Carrier unlocking

Most mobile devices purchased directly from a telco are carrier locked. This means you are unable to use the device on any other telco network until the carrier lock is removed or *carrier unlocked*. Once you fully own a device, the telco should freely carrier unlock the phone, but you will have to ask for it specifically, as shown in Figure 2.40; they do not do so automatically. If you have an account in good standing and are traveling to another country with compatible telco service, you may be able to get a telco to carrier unlock your phone for your trip so you can temporarily use another SIM card for local telco services.

FIGURE 2.40 The AT&T mobile device unlock request page



Device unlock portal

Unlocking a device? You can do everything online, in one place. Read about how to Unlock a phone or tablet if you need a little help.

FYI, we can only unlock devices that work on our network. So if you're moving your device to AT&T, be sure to contact your own carrier to unlock it.

Taking a trip? You don't need to unlock your phone. Learn how you can use the talk, text & data you already have in over 100 countries with International Day Pass.

Unlock your device

Unlocking a device is simple. First, verify your device eligibility. Then, follow a few steps to submit your request. Be sure to have the wireless number and customer name on the account handy.

Check your unlock status

Find out your unlock status. You'll need your device IMEI and the unlock request number we sent you. The IMEI number is usually 15 numbers. Your request number is a combination of letters and numbers. FYI, requests can take up to 2 days.

Having a device carrier unlocked is not the same as rooting. Carrier unlocked status only allows the switching out of SIMs in order to use the service from another telco (which is technically possible only if your device uses the same radio frequencies as the telco).

A carrier unlocked device should not represent any additional risk to an organization; thus there is likely no need for a prohibition of carrier unlocked devices on company networks.

Firmware OTA updates

Firmware OTA updates are upgrades, patches, and improvements to the existing firmware of a mobile device that are downloaded from the telco or vendor over the air (OTA). Some telcos do not count firmware downloads against data caps, but you may find downloading OTA updates over WiFi to be faster and more reliable anyway. Generally, as a mobile device owner, you should install new firmware OTA updates onto a device once they become available. However, some updates may alter the device configuration or interfere with MDM restrictions. Organizations should attempt to test new updates before allowing managed devices to receive them. There simply may need to be a waiting period established so the MDM vendor can update their management product to properly oversee the deployment and configuration of the new firmware update.

Camera use

The company security policy needs to address mobile devices with onboard cameras. Some environments disallow cameras of any type. This would require that authorized equipment be without a camera. If cameras are allowed, a description of when they may and may not be used should be clearly documented and explained to workers. A mobile device can act as a storage device, provide an alternate wireless connection pathway to an outside provider or service, and also be used to collect images and video that disclose confidential information or equipment.

If geofencing is available, it may be possible to use MDM to implement a location-specific hardware-disable profile in order to turn off the camera (or other components) while the device is on company premises but return the feature to operational status once the device leaves the geofenced area.

SMS/MMS

SMS (Short Messaging Service), also known as texting, and MMS (Multimedia Messaging Service) are communication functions provided by telcos and commonly used on mobile devices. MMS allows for images, video, and potentially other files to be sent to a recipient along with text messages.

SMS and MMS represent generally the same level of risk and benefit as that of email. It is a good idea to block attachments and file exchange, spam filtering is needed (although it may be called SPIM for Spam over Instant Messaging), social engineering defenses need to be established, and users must be trained on avoiding risk and minimizing distractions.

External media

Many mobile devices support removable storage. Some smartphones support microSD cards whereas most larger mobile devices, such as tablets and notebook computers, support

SD cards and other media card formats, which can be used to expand available storage on a mobile device. However, most mobile phones require the removal of a back plate and sometimes removal of the battery in order to add or remove a storage card. Larger mobile phones, tablets, and notebook computers may support an easily accessible card slot on the side of the device.

In addition, there are mobile storage devices that can provide Bluetooth- or WiFi-based access to stored data through an onboard wireless interface.

Organizations need to consider whether the use of removable storage on portable and mobile devices is a convenient benefit or a significant risk vector. If the former, proper access limitations and use training are necessary. If the latter, then a prohibition of removable storage can be implemented via MDM.

USB OTG

USB is a specification that allows a mobile device with a USB port to act as a host and use other standard peripheral USB equipment, such as storage devices, mice, keyboards, and digital cameras. USB OTG is a feature that can be disabled via MDM if it is perceived as a risk vector for mobile devices used within an organization.

Recording microphone

Most mobile devices with a speaker also have a microphone. The microphone can be used to record audio, noise, and voices nearby. Many devices also support external microphones connected by a USB adapter or a 1/8″ stereo jack. If microphone recording is deemed a security risk, this feature should be disabled using an MDM or deny presence of mobile devices in sensitive areas or meetings.

GPS tagging

GPS tagging is the same as geolocation and geotagging. Please see the earlier section “Geolocation.” This is also a feature that can be disabled using an MDM.

WiFi direct/ad hoc

WiFi Direct is the new name for the wireless topology of ad hoc or peer-to-peer connections. It is a means for wireless devices to connect directly to each other without the need for a middleman base station. WiFi Direct supports WPA-2, but not all devices are capable of supporting this optional encryption scheme. WiFi Direct is used for a wide range of capabilities, including transmitting media for display on a monitor or television, sending print jobs to printers, controlling home automation products, controlling security cameras, and controlling photo frames.

In a business environment, WiFi Direct should only be used where WPA-2 can be used. Otherwise, the plain-text communication presents too much risk.

Tethering

Tethering is the activity of sharing the cellular network data connection of a mobile device with other devices. The sharing of data connection can take place over WiFi, Bluetooth, or

USB cable. Some service providers include tethering in their service plans, whereas others charge an additional fee and a few block tethering completely.

Tethering may represent a risk to the organization. It is a means for a user to grant Internet access to devices that are otherwise network isolated, and it can be used as a means to bypass the company's filtering, blocking, and monitoring of Internet use. Thus, tethering should be blocked while a mobile device is within a company facility.

Payment methods

There are a number of mobile device–based payment systems. Some are based on NFC, others on RFID, some on SMS, and still others on optical camera–based solutions, such as scanning Quick Response (QR) codes. Mobile payments are convenient for the shopper but might not always be a secure mechanism. Users should only employ mobile payment solutions that require a per-transaction confirmation or that require the device to be unlocked and an app launched in order to perform a transaction. Without these precautions, it may be possible to clone your device's contactless payment signals and perform transaction abuse.

An organization is unlikely to see any additional risk based on mobile payment solutions. However, caution should still be taken when implementing them on company-owned equipment or when they are linked to the company's financial accounts.

Deployment models

A number of deployment models are available for allowing and/or providing mobile devices for employees to use while at work and to perform work tasks when away from the office. However, before discussing these, we'll look at several additional concerns that a mobile device policy must address regarding the use of a personal or portable electronic device (PED) in relation to the organization's IT infrastructure and business tasks.

Data Ownership When a personal device is used for business tasks, comingling of personal data and business data is likely to occur. Some devices can support storage segmentation, but not all devices can provide data-type isolation. Establishing data ownership can be complicated. For example, if a device is lost or stolen, the company may wish to trigger a remote wipe, clearing the device of all valuable information. However, the employee will often be resistant to this, especially if there is any hope that the device will be found or returned. A wipe removes all business and personal data, which may be a significant loss to the individual—especially if the device is recovered, because then the wipe would seem to have been an overreaction. Clear policies about data ownership should be established. Some MDM solutions can provide data isolation/segmentation and support business data sanitization without affecting personal data.

The mobile device policy regarding data ownership should address backups for mobile devices. Business data and personal data should be protected by a backup solution—either a single solution for all data on the device or separate solutions for each type or class of data. This reduces the risk of data loss in the event of a remote-wipe event as well as device failure or damage.

Support Ownership When an employee's mobile device experiences a failure, a fault, or damage, who is responsible for the device's repair, replacement, or technical support? The mobile device policy should define what support will be provided by the company and what support is left to the individual and, if relevant, their service provider.

Patch Management The mobile device policy should define the means and mechanisms of patch management for a personally owned mobile device. Is the user responsible for installing updates? Should the user install all available updates? Should the organization test updates prior to on-device installation? Are updates to be handled over the air (via service provider) or over WiFi?

Antivirus Management The mobile device policy should dictate whether antivirus, antimalware, and antispyware scanners are to be installed on mobile devices. The policy should indicate which products and apps are recommended for use, as well as the settings for those solutions.

Forensics The mobile device policy should address forensics and investigations as related to mobile devices. Users need to be aware that in the event of a security violation or a criminal activity, their devices might be involved. This would mandate gathering evidence from those devices. Some processes of evidence-gathering can be destructive, and some legal investigations require the confiscation of devices.

Privacy The mobile device policy should address privacy and monitoring. When a personal device is used for business tasks, the user often loses some or all of the privacy they enjoyed prior to using their mobile device at work. Workers may need to agree to be tracked and monitored on their mobile devices, even when not on company property and outside of work hours. A personal device in use under the mobile device policy should be considered by the individual to be quasi-company property.

On-boarding/off-boarding The mobile device policy should address personal mobile device on-boarding and off-boarding procedures. Mobile device on-boarding includes installing security, management, and productivity apps along with implementing secure and productive configuration settings. Mobile device off-boarding includes a formal wipe of the business data along with the removal of any business-specific applications. In some cases, a full device wipe and factory reset may be prescribed.

Adherence to Corporate Policies A mobile device policy should clearly indicate that using a personal mobile device for business activities doesn't exclude a worker from adhering to corporate policies. A worker should treat mobile equipment as company property and thus stay in compliance with all restrictions, even when off premises and during off hours.

User Acceptance A mobile device policy needs to be clear and specific about all the elements of using a personal device at work. For many users, the restrictions, security settings, and MDM tracking implemented under a mobile device policy will be much more onerous than they expect. Thus, organizations should make the effort to fully explain the details of a mobile device policy prior to allowing a personal device into the production environment. Only after an employee has expressed consent and acceptance, typically through a signature, should their device be on-boarded.

Architecture/Infrastructure Considerations When implementing a mobile device policy, organizations should evaluate their network and security design, architecture, and infrastructure. If every worker brings in a personal device, the number of devices on the network may double. This requires planning to handle IP assignments, communications isolation, data-priority management, increased intrusion detection system (IDS)/intrusion prevention system (IPS) monitoring load, as well as increased bandwidth consumption, both internally and across any Internet link. Most mobile devices are wireless enabled, so this will likely require a more robust wireless network and dealing with WiFi congestion and interference. A mobile device policy needs to be considered in light of the additional infrastructure costs it will trigger.

Legal Concerns Company attorneys should evaluate the legal concerns of mobile devices. Using personal devices in the execution of business tasks probably means an increased burden of liability and risk of data leakage. Mobile devices may make employees happy, but they might not be worthwhile or cost-effective for the organization.

Acceptable Use Policy The mobile device policy should either reference the company acceptable use policy or include a mobile device–specific version focusing on unique issues. With the use of personal mobile devices at work, there is an increased risk of information disclosure, distraction, and accessing inappropriate content. Workers should remain mindful that the primary goal when at work is to accomplish productivity tasks.

BYOD

BYOD is a policy that allows employees to bring their own personal mobile devices to work and use those devices to connect to business resources and/or the Internet through the company network. Although BYOD may improve employee morale and job satisfaction, it increases security risk to the organization. If the BYOD policy is open-ended, any device is allowed to connect to the company network. Not all mobile devices have security features, and thus such a policy may allow noncompliant devices onto the production network.

Users need to understand the benefits, restrictions, and consequences of using their own devices at work. Reading and signing off on the BYOD policy, along with attending an overview or training program, may be sufficient to accomplish reasonable awareness.

COPE

The concept of corporate owned, personally enabled (COPE) means the organization purchases devices and provides them to employees. Each user is then able to customize the device and use it for both work activities and personal activities. COPE allows the organization to select exactly which devices are to be allowed on the organizational network—specifically only those devices that can be configured into compliance with the security policy.

CYOD

The concept of choose your own device (CYOD) provides users with a list of approved devices from which to select the device to implement. A CYOD can be implemented so that employees purchase their own devices from the approved list (a BYOD variant) or the company can purchase the devices for the employees (a COPE variant).

Corporate-owned

A corporate-owned mobile strategy is when the company purchases mobile devices that can support compliance with the security policy. These devices are to be used exclusively for company purposes, and users should not perform any personal tasks on them. This often requires workers to carry a second device for personal use.

VDI

Virtual desktop infrastructure (VDI) is a means to reduce the security risk and performance requirements of end devices by hosting virtual machines on central servers that are remotely accessed by users. VDI has been adopted for mobile devices and has already been widely used on tablets and notebook computers. It is a means to retain storage control on central servers, gain access to higher levels of system processing and other resources, and allow lower-end devices access to software and services beyond their hardware's capacity.

This has led to virtual mobile infrastructure (VMI), where the operating system of a mobile device is virtualized on a central server. Thus most of the actions and activities of the traditional mobile device are no longer occurring on the mobile device itself. This remote virtualization allows an organization greater control and security than when using a standard mobile device platform. It can also enable personally owned devices to interact with the VDI without increasing the risk profile. This concept requires a dedicated isolated wireless network to keep BYOD devices from interacting directly with company resources other than through the VDI solution.

Exam Essentials

Know the basics of various connection methods. You should have a basic understanding of the various mobile device connection methods, including cellular, WiFi, SATCOM, Bluetooth, NFC, ANT, infrared, and USB.

Understand mobile device security. Device security involves the range of potential security options or features that may be available for a mobile device. Not all portable electronic devices (PEDs) have good security features. PED security features include full device encryption, remote wiping, lockout, screen locks, GPS, application control, storage segmentation, asset tracking, inventory control, mobile device management, device access control, removable storage, and disabling of unused features.

Be familiar with mobile device security management concepts. Smartphones and other mobile devices present an ever-increasing security risk as they become more and more capable of interacting with the Internet as well as corporate networks. Mobile devices are becoming the target of hackers and malicious code. A wide range of security features are available on mobile devices. Mobile device management (MDM) is a software solution to the challenging task of managing the myriad mobile devices that employees use to access company resources.

Understand mobile device application management. Application control or application management is a device management solution that limits which applications can be installed on a device. It can also be used to force specific applications to be installed or to enforce the settings of certain applications in order to support a security baseline or maintain other forms of compliance.

Understand mobile device content management. Content management involves controlling mobile devices and their access to content hosted on company systems, as well as controlling access to company data stored on mobile devices.

Know how to perform a mobile device remote wipe. Remote wipe or remote sanitation should be performed if a device is lost or stolen. A remote wipe lets you delete all data and possibly even configuration settings from a device remotely.

Understand mobile device geofencing. Geofencing is the designation of a specific geographical area, which is then used to implement features on mobile devices. A geofence can be defined by GPS coordinates, a wireless indoor positioning system (IPS), or the presence or lack of a specific wireless signal.

Understand mobile device geolocation. Geolocation or geotagging is the ability of a mobile device to include details about its location in any media created by the device.

Know how to use mobile device screen lock. A screen lock is designed to prevent someone from being able to casually pick up and use your phone or mobile device.

Be familiar with mobile device push notification services. Push notification services are able to send information to your device rather than having the device (or its apps) pull information from an online resource.

Understand mobile device context-aware authentication. Context-aware authentication is an improvement on traditional authentication means. Contextual authentication will evaluate the origin and context of a user's attempt to access a system.

Be able to describe mobile device containerization. Containerization is the next stage in the evolution of the virtualization trend for internally hosted systems and cloud providers/services. Containerization can be used in relation to mobile devices by hosting the primary OS on a containerization host in the company cloud; then the actual mobile device is used only as a remote-control interface to the OS container, rather than having the business apps and company data on the device itself.

Understand mobile device storage segmentation. Storage segmentation is used to artificially compartmentalize various types or values of data on a storage medium. On a mobile device, the device manufacturer and/or the service provider may use storage segmentation to isolate the device's OS and preinstalled apps from user-installed apps and user data.

Know the security issues with third-party app stores. Third-party app sources represent a significant increase in risk of data leakage or malware intrusion to an organizational network.

Define rooting and jailbreaking. Rooting or jailbreaking is the action of breaking the digital rights management (DRM) security on the bootloader of a mobile device in order to be able to operate the device with root or full-system privileges.

Understand sideloading. Sideloading is the activity of installing an app on a device by bringing the installer file to the device through some form of file transfer or USB storage method rather than installing from an app store.

Be familiar with custom firmware. Mobile devices come preinstalled with a vendor- or telco-provided firmware or core operating system. If a device is rooted or jailbroken, it can allow the user to install alternate custom firmware in place of the default firmware. Custom firmware may remove bloatware included by the vendor or telco, add or remove features, and streamline the OS to optimize performance.

Understand carrier unlock. Most mobile devices purchased directly from a telco are carrier locked. This means you are unable to use the device on any other telco network until the carrier lock is removed or carrier unlocked.

Know how to use firmware OTA updates. Firmware OTA updates are upgrades, patches, and improvements to the existing firmware of a mobile device that are downloaded from the telco or vendor over the air (OTA).

Define USB OTG. USB On-The-Go (OTG) is a specification that allows mobile devices with a USB port to act as a host and use other standard peripheral USB equipment, such as storage devices, mice, keyboards, and digital cameras.

Define WiFi Direct. WiFi Direct is a means for wireless devices to connect directly to each other without the need for a middleman base station.

Define tethering. Tethering is the activity of sharing the cellular network data connection of a mobile device with other devices. The sharing of data connection can take place over WiFi, Bluetooth, or USB cable.

Understand mobile device application security. The applications and functions used on a mobile device need to be secured. Related concepts include key management, credential management, authentication, geotagging, encryption, application whitelisting, and transitive trust/authentication.

Define BYOD. Bring your own device (BYOD) is a policy that allows employees to bring their own personal mobile devices to work and then use those devices to connect to (or through) the company network to access business resources and/or the Internet. Although BYOD may improve employee morale and job satisfaction, it increases security risks to the organization. Related issues include data ownership, support ownership, patch management, antivirus management, forensics, privacy, on-boarding/off-boarding, adherence to corporate policies, user acceptance, architecture/infrastructure considerations, legal concerns, acceptable use policies, and onboard cameras and video.

Define COPE. COPE stands for corporate owned, personally enabled. It allows the organization to purchase devices and provide them to employees. Each user is then able to customize the device and use it for both work activities and personal activities.

Define CYOD. CYOD stands for choose your own device. This concept provides users with a list of approved devices from which to select the device to implement.

Be familiar with corporate-owned mobile strategies. A corporate-owned mobile strategy is when the company purchases mobile devices that can support compliance with the security policy. These devices are to be used exclusively for company purposes, and users should not perform any personal tasks on the devices.

Understand VDI and VMI. Virtual desktop infrastructure (VDI) is a means to reduce the security risk and performance requirements of end devices by hosting virtual machines on central servers that are remotely accessed by users. This has led to virtual mobile infrastructure (VMI), in which the operating system of a mobile device is virtualized on a central server.

2.6 Given a scenario, implement secure protocols.

A significant improvement in the security stance of an organization can be achieved by implementing secure communications protocols. This section discusses many protocols that can be used to add encryption to communications.

Protocols

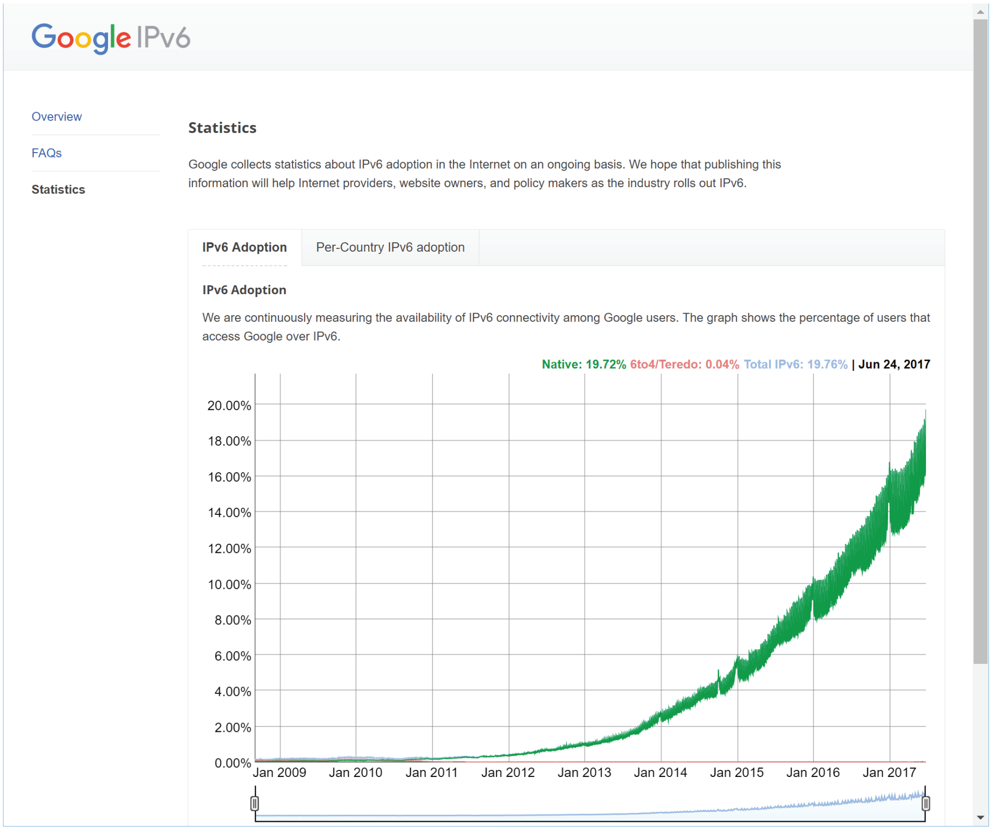
TCP/IP is the primary protocol suite in use on the Internet and most private networks across the planet. TCP/IP is a protocol suite that wasn't originally designed around a global network concept, nor was security a primary feature. However, TCP/IP is the primary protocol used on the Internet, and many security protocols and add-on features are supported by IP and TCP.

General knowledge of the TCP/IP suite is necessary for the Security+ exam, but it's assumed to be a prerequisite knowledge base primarily derived from the CompTIA Network+ certification. If you aren't generally versed in TCP/IP, please consult Network+ study materials or research TCP/IP online.

IPv4 is in widespread use with a 32-bit addressing scheme. Most of the public network is still IPv4 based; however, available public IPv4 addresses are scarce. IPv4 (as well as IPv6) operates at the Network layer, or Layer 3, of the OSI protocol stack.

IPv6 was finalized in RFC 2460 in 1998. It uses a 128-bit addressing scheme, eliminates broadcasts and fragmentation, and includes native communication-encryption features. It was enabled officially on the Internet on June 6, 2012. The move to IPv6 is still occurring slowly, but the pace is beginning to increase. To see the progress of worldwide public deployment of IPv6, see Google's IPv6 Statistics page (Figure 2.41) at `https://www.google.com/intl/en/ipv6/statistics.html`.

FIGURE 2.41 Google's IPv6 statistics



Statistics

Google collects statistics about IPv6 adoption in the Internet on an ongoing basis. We hope that publishing this information will help Internet providers, website owners, and policy makers as the industry rolls out IPv6.

IPv6 Adoption

We are continuously measuring the availability of IPv6 connectivity among Google users. The graph shows the percentage of users that access Google over IPv6.

There is not a scenario where using a secure protocol would be a bad idea. In every data communication, use a secure protocol if one exists. And if a secure form of a specific protocol does not exist, then configure a VPN between endpoints to run the insecure protocol across in order to gain protection for the transaction.

DNSSEC

The Domain Name System (DNS) is the hierarchical naming scheme used in both public and private networks. DNS links IP addresses and human-friendly fully qualified domain names (FQDNs) together. A FQDN consists of three main parts:

- Top-level domain (TLD)—The `com` in `www.google.com`
- Registered domain name—The `google` in `www.google.com`
- Subdomain(s) or hostname—The `www` in `www.google.com`

The TLD can be any number of official options, including six of the original seven TLDs—com, org, edu, mil, gov, and net—as well as many newer ones, such as info, museum, telephone, mobi, biz, and so on. There are also country variations known as *country codes*. (See www.iana.org/domains/root/db/ for details on current TLDs and country codes.) (Note: The seventh original TLD was int, for international, which was replaced by the country codes.)

The registered domain name must be officially registered with one of any number of approved domain registrars, such as Network Solutions or GoDaddy.

The far-left section of an FQDN can be either a single hostname, such as www, ftp, and so on, or a multisectioned subdomain designation, such as server1.group3.bldg5.mycompany.com.

The total length of an FQDN can't exceed 253 characters (including the dots). Any single section can't exceed 63 characters. FQDNs can only contain letters, numbers, and hyphens.

Every registered domain name has an assigned authoritative name server. The authoritative name server hosts the original zone file for the domain. A *zone file* is the collection of resource records or details about the specific domain. There are dozens of possible resource records (see http://en.wikipedia.org/wiki/List_of_DNS_record_types); the most common are listed in Table 2.4.

TABLE 2.4 Common resource records

Record	Type	Description
A	Address record	Links an FQDN to an IPv4 address
AAAA	Address record	Links an FQDN to an IPv6 address
PTR	Pointer record	Links an IP address to a FQDN (for reverse lookups)
CNAME	Canonical name	Links an FQDN alias to another FQDN
MX	Mail exchange	Links a mail- and messaging-related FQDN to an IP address
NS	Name server record	Designates the FQDN and IP address of an authorized name server
SOA	Start of authority record	Specifies authoritative information about the zone file, such as primary name server, serial number, timeouts, and refresh intervals

Originally, DNS was handled by a static local file known as the HOSTS file. This file still exists, but a dynamic DNS query system has mostly replaced it, especially for large private networks as well as the Internet. When client software points to a FQDN, the protocol stack initiates a DNS query in order to resolve the name into an IP address that can be used in the construction of the IP header. The resolution process first checks the local DNS cache to see if the answer is already known. The DNS cache consists of preloaded content from the local HOSTS file plus any DNS queries performed during the current boot session (that haven't timed out). If the needed answer isn't in the cache, a DNS query is sent to the DNS server indicated in the local IP configuration. The process of resolving the query is interesting and complex, but most of it isn't relevant to the Security+ exam. To explore DNS in more detail, see `http://en.wikipedia.org/wiki/Domain_Name_System` and `http://unixwiz.net/techtips/iguide-kaminsky-dns-vuln.html`.

DNS operates over TCP port 53 and UDP port 53. TCP port 53 is used for zone transfers. These are zone file exchanges between DNS servers for special manual queries, or when a response exceeds 512 bytes. UDP port 53 is used for most typical DNS queries.

Domain Name System Security Extensions (DNSSEC) is a security improvement to the existing DNS infrastructure. The primary function of DNSSEC is to provide reliable authentication between devices during DNS operations. DNSSEC has been implemented across a significant portion of the DNS system. Each DNS server is issued a digital certificate, which is then used to perform mutual certificate authentication. The goal of DNSSEC is to prevent a range of DNS abuses where false data can be injected into the resolution process. Once fully implemented, DNSSEC will significantly reduce server-focused DNS abuses.

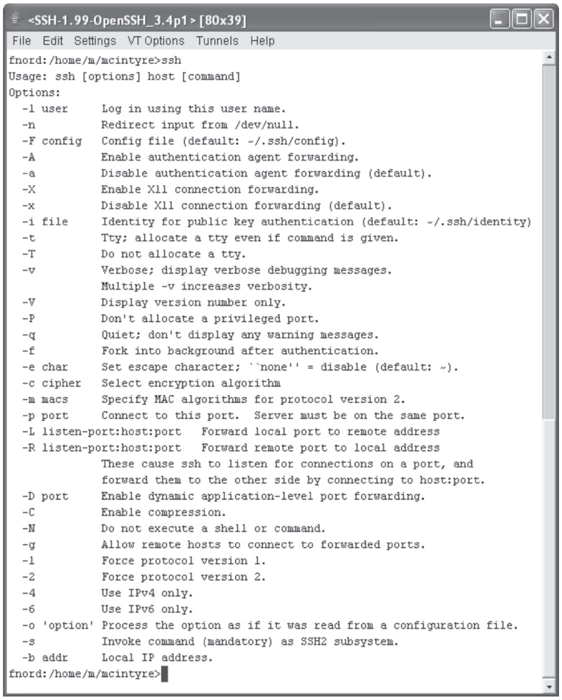
SSH

Secure Shell (SSH) is a secure replacement for Telnet and many of the Unix "r" tools, such as `rlogon`, `rsh`, `rexec`, and `rcp`. While Telnet provides remote access to a system at the expense of plain-text communication, SSH transmissions are cipher text and thus are protected from eavesdropping. SSH operates over TCP port 22. SSH is the protocol most frequently used with a terminal editor program such as HyperTerminal in Windows, Minicom in Linux, or PuTTY in both. An example of SSH use would involve remotely connecting to a switch or router in order to make configuration changes.

SSH offers a means by which a secure command-line, text-only interface connection with a server, router, switch, or similar device can be established over any distance. You can perform many command-line or scriptable activities through the SSH connection, as shown in Figure 2.42.

SSH transmits both authentication traffic and data in a secured encrypted form. Thus, no information is exchanged in clear text. SSH is a very flexible tool. It can be used as a secure Telnet replacement; it can be used to encrypt protocols similar to TLS, such as SFTP; and it can be used as a VPN protocol.

FIGURE 2.42 A Unix version of SSH, showing a list of available command-line options


<SSH-1.99-OpenSSH_3.4p1> [80x39]
File Edit Settings VT Options Tunnels Help

```
fnord:/home/m/mcintyre>ssh
Usage: ssh [options] host [command]
Options:
  -l user     Log in using this user name.
  -n          Redirect input from /dev/null.
  -F config   Config file (default: ~/.ssh/config).
  -A          Enable authentication agent forwarding.
  -a          Disable authentication agent forwarding (default).
  -X          Enable X11 connection forwarding.
  -x          Disable X11 connection forwarding (default).
  -i file     Identity for public key authentication (default: ~/.ssh/identity)
  -t          Tty; allocate a tty even if command is given.
  -T          Do not allocate a tty.
  -v          Verbose; display verbose debugging messages.
              Multiple -v increases verbosity.
  -V          Display version number only.
  -P          Don't allocate a privileged port.
  -q          Quiet; don't display any warning messages.
  -f          Fork into background after authentication.
  -e char     Set escape character; ``none'' = disable (default: ~).
  -c cipher   Select encryption algorithm
  -m macs     Specify MAC algorithms for protocol version 2.
  -p port     Connect to this port.  Server must be on the same port.
  -L listen-port:host:port   Forward local port to remote address
  -R listen-port:host:port   Forward remote port to local address
              These cause ssh to listen for connections on a port, and
              forward them to the other side by connecting to host:port.
  -D port     Enable dynamic application-level port forwarding.
  -C          Enable compression.
  -N          Do not execute a shell or command.
  -g          Allow remote hosts to connect to forwarded ports.
  -1          Force protocol version 1.
  -2          Force protocol version 2.
  -4          Use IPv4 only.
  -6          Use IPv6 only.
  -o 'option' Process the option as if it was read from a configuration file.
  -s          Invoke command (mandatory) as SSH2 subsystem.
  -b addr     Local IP address.
fnord:/home/m/mcintyre>
```

Telnet

Telnet is a terminal-emulation network application that supports remote connectivity for executing commands and running applications but doesn't support transfer of files. Telnet uses TCP port 23. Because it's a clear-text protocol and service, it should be avoided and replaced with SSH.

S/MIME

Because email is natively insecure, several encryption options have been developed to add security to email used over the Internet. Two of the most common solutions are *Secure/Multipurpose Internet Mail Extensions (S/MIME)* and *Pretty Good Privacy (PGP)*.

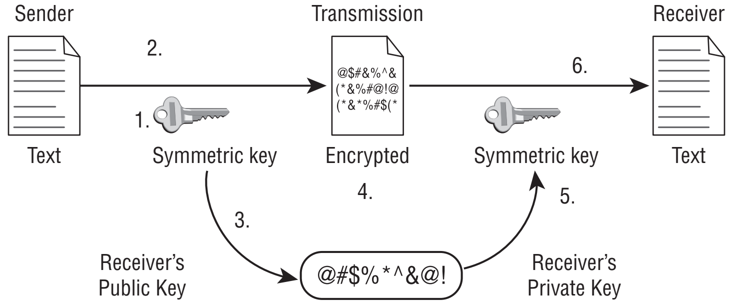
S/MIME is an Internet standard for encrypting and digitally signing email. S/MIME takes the standard MIME element of email, which enables email to carry attachments and higher-order textual information (fonts, color, size, layout, and so on), and expands this to include message encryption. S/MIME uses a hybrid encryption system that combines RSA (an asymmetric encryption scheme) and AES (a symmetric encryption algorithm) to encrypt and protect email.

S/MIME works by taking the original message from the server, encrypting it using a symmetric encryption key, and then attaching it to a new blank email. The symmetric encryption key is itself encrypted or enveloped using the recipient's public key, and then attached to the blank email as well. The new blank email includes the sender's and receiver's email addresses to control routing of the message to its destination. The receiver must then strip off the attachments, open the envelope using their private key to extract the symmetric key, and then decrypt the original message. When email encryption is used in this manner, confidentiality is protected.

As shown in Figure 2.43, the basic process is as follows:

1. The sender's system generates a random symmetric key.
2. The sender encrypts the message with the random symmetric key.
3. The symmetric key is enveloped (encrypted) using the recipient's public key.
4. The message and the envelope are sent to the recipient.
5. The recipient opens (decrypts) the envelope using the recipient's private key to extract the symmetric key.
6. The recipient decrypts the message using the symmetric key.

FIGURE 2.43 The hybrid cryptography-based email encryption process



The process of encrypting email isn't complex; however, it's cumbersome in implementation. Fortunately, the native S/MIME support in most email clients automates the process. The only restriction to the S/MIME email solutions is that all communication partners must have compatible S/MIME products installed and use a common or compatible source for their asymmetric encryption key pairs.

S/MIME is a standards-based email security solution. An example of a proprietary, open-source email security solution is Pretty Good Privacy (PGP). Please see this concept discussion in the Chapter 6 section "PGP/GPG."

SRTP

SRTP (Secure Real-Time Transport Protocol, or Secure RTP) is a security improvement over RTP (Real-Time Transport Protocol) that is used in many VoIP (Voice over IP) communications. SRTP aims to minimize the risk of VoIP DoS through robust encryption and reliable authentication.

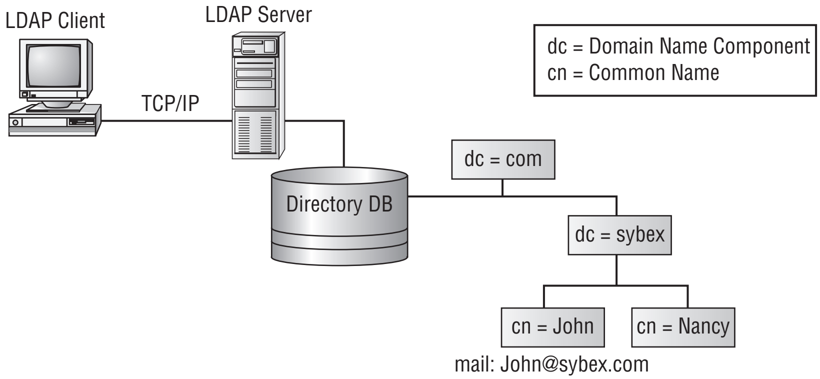
LDAPS

A *directory service* is a managed list of network resources. It is effectively a network index or network telephone book of the systems and their shared resources. Through the use of a directory service, large networks are easier to navigate, manage, and secure. Active Directory from Microsoft, OpenLDAP, and legacy eDirectory (or NDS) from Novell are examples of directory services. All three are based on *Lightweight Directory Access Protocol (LDAP)*.

LDAP is a standardized protocol that enables clients to access resources within a directory service. A directory service is a network service that provides access to a central database of information, which contains detailed information about the resources available on a network. LDAP follows the *x.500 standard*, which defines what a directory service is and how it is to be constructed and organized (at least from a foundational infrastructure perspective). Clients can interact with directory service resources through LDAP by using authentication that consists of at least a username and password.

LDAP directory structures are hierarchical data models that use branches like a tree and that have a clearly identified and defined root (see Figure 2.44). LDAP operates over TCP ports 389 (plain text) and 636 (secure). There are two connection mechanisms used for plain-text authentication. They are known by the terms anonymous bind (no authentication) and simple bind (plain-text password authentication). It's important to secure LDAP rather than allow it to operate in a plain-text insecure form. This is accomplished by enabling the Simple Authentication and Security Layer (SASL) on LDAP, which implements Transport Layer Security (TLS) on the authentication of clients as well as all data exchanges. This results in *LDAP Secured (LDAPS)*. This isn't the only means to secure LDAP, but it's the method addressed on Security+.

FIGURE 2.44 An example of an LDAP-based directory services structure



FTPS

The antiquated protocol of file transfer or exchange is *File Transfer Protocol (FTP)*. This protocol is often used to move files between one system and another, either over the Internet or within private networks. Understanding the basics of FTP and the secured alternative file-transfer solutions is important for the Security+ exam.

FTP is an in-the-clear file-exchange protocol. It's supported by any computer system that uses TCP/IP. An FTP server system is configured to allow authenticated or anonymous FTP clients to log on in order to upload or download files. FTP employs TCP ports 20 and 21 by default. Port 21 is used for session management, and port 20 is used for data transmission. There are two connection options for FTP: active and passive. Active FTP is the original method of FTP connection. The process of Active FTP is as follows:

The client initiates the session management connection to the FTP server on TCP port 21 using a random source port number (such as 1060).

The FTP server initiates the data transmission connection to the FTP client from TCP source port 20 to the client's port that is one increment above the client's original source port (such as 1061).

This technique works only if the client is not protected by a firewall, proxy, or NAT function because it requires inbound initiation from the FTP server to the FTP client. In most cases today, since a firewall, proxy, or NAT is likely present, FTP is used in passive mode. The process of passive FTP is as follows:

The client initiates the session management connection to the FTP server on TCP port 21 using a random source port number (such as 1060).

The server selects a random port number to open in order to receive a second client-initiated connection (such as port 4081), and sends that number to the client over the existing communication session.

The client initiates another connection to the FTP server for data transmission, using an incremented client source port number (such as 1061) to the server's suggested destination port number (such as 4081).

The exchange of files is a common practice on the Internet, intranets, and extranets. FTP is an independent platform and thus makes file exchanges between different OSs simple. It's one of the common services deployed in a DMZ—an extension of a private network where Internet users can access services such as the Web and email—in order to provide controlled public access to company resources while still allowing internal clients to access the services.

Because all FTP traffic is transmitted in the clear, it's vulnerable to packet sniffing and other forms of eavesdropping. It's important not to use the same user account and password on FTP that you use in a secure environment. Otherwise, if an attacker captures your FTP logon traffic, they also obtain the logon credentials needed to log into your secured network. Always use a separate and distinct user account for FTP logons. Sniffers and protocol analyzers are discussed in the "Understand protocol analyzers" section earlier in this chapter.

Anonymous FTP is a form of nameless logon to an FTP server. Usually, visitors to an FTP site who wish to log on anonymously use the word *anonymous* as the logon name. They're then prompted to provide their email address as the password, but any text string suffices.

Site administrators should carefully configure FTP servers that allow anonymous access. Anonymous users shouldn't be able to download (or, in many cases, view) any files uploaded by anonymous users. Anonymous upload and download should be enabled only if absolutely necessary. When possible, don't allow both authenticated and anonymous FTP logons on the same FTP site. Most FTP servers have anonymous FTP enabled by default, so usually it must be specifically disabled in order to limit access to authenticated users.

If FTP upload is allowed—especially when anonymous FTP uploading is allowed—ensure that it isn't possible to access upload folders from a web URL. If you don't take this precaution, web visitors may be able to download files from the FTP site through HTTP, or they may be able to execute uploaded files. Both of these tactics are commonly used by hackers in a wide variety of intrusion attacks.

Blind FTP is a configuration of anonymous FTP or authenticated FTP in which uploaded files are unseen and unreadable by visitors. Thus, users can upload files but not see the resulting uploads. Additionally, even if a user knows the exact pathname and filename of a file deposited onto your blind FTP site, the deposited files are write-only, and thus reading or downloading isn't possible. This ensures that your FTP site isn't overrun by file swappers using your system as a file-exchange point. File swappers often exchange illegal (unlicensed) copies of software, music, and movies through unsecured FTP servers. Uploaded files on a blind FTP server become accessible only after the administrator has either changed the files' permissions or moved them into a folder configured to allow downloads.

FTPS is FTP Secure or FTP SSL, which indicates that it's a variation of FTP secured by SSL (or now TLS). This is an FTP service variation distinct from SSH-secured FTP (SFTP). Although in general use they're similar, in that both provide for cryptographically protected file transfers, they aren't interoperable.

FTPS is supported by FTP servers in either an implicit or an explicit mode (FTPIS or FTPES, respectively). Implicit implies that the client must specifically challenge the FTPS server with a TLS/SSL ClientHello message. This assumes that only FTPS clients will connect. In order to allow traditional FTP clients to continue to operate over ports 20 (data channel) and 21 (control channel), FTPS is delegated to ports 990 (control channel) and 989 (data channel). It's important, however, to note that implicit mode is now considered deprecated.

Explicit (FTPES) mode implies that the FTPS client must specifically request an FTPS connection on ports 20 and 21; otherwise, an insecure FTP connection will be attempted. More information regarding explicit mode is available in RFC 2228 and RFC 4218.

TFTP

Trivial File Transfer Protocol (TFTP) is a simple file-exchange protocol that doesn't require authentication. It has fewer commands and capabilities than FTP (mostly PUT for uploading and GET for downloading). TFTP operates on UDP port 69. It can be used to host device-configuration files. This allows those devices to download their configuration if it's lost, such as due to a power failure. Thus essential network devices can self-restore quickly. However, this function is often replaced by a locally installed SD card or other flash memory product. TFTP is also used in multicasting to serve as a caching system for links that are otherwise unable to keep up with the default transmission speed of a multicast signal.

SFTP

Secure FTP (SFTP) is a secured alternative to standard FTP. Standard FTP sends all data, including authentication traffic, in the clear. Thus, there is no confidentiality protection. SFTP encrypts both authentication and data traffic between the client and server by employing SSH to provide secure FTP communications. Thus, SFTP provides protection for both the authentication traffic and the data transfer occurring between a client and server.

No matter what secure FTP solution is employed, both the server and the client must have the same solution. The client and the server must have compatible or interoperable FTP tools in order to establish a connection and support the exchange of files. Otherwise, FTP session establishment and subsequent file-transfer communications won't be possible.

SNMPv3

Simple Network Management Protocol (SNMP) is a standard network-management protocol supported by most network devices and TCP/IP-compliant hosts. These include routers, switches, bridges, WAPs, firewalls, VPN appliances, modems, printers, and so on. Through the use of a management console, you can use SNMP to interact with various network devices to obtain status information, performance data, statistics, and configuration details. Some devices support the modification of configuration settings through SNMP.

Early versions of SNMP relied on plain-text transmission of community strings as authentication. Communities were named collections of network devices that SNMP management consoles could interact with. The original default community names were *public* and *private*. The latest version of SNMP allows for encrypted communications between devices and the management console, as well as robust authentication protection customized authentication factors.

SNMP operates over UDP ports 161 and 162. UDP port 161 is used by the SNMP agent (that is, network device) to receive requests, and UDP port 162 is used by the management console to receive responses and notifications (also known as trap messages).

SSL/TLS

Transport Layer Security (TLS) is the updated replacement for the Netscape Corporation's SSL. TLS is generally the same as SSL, but it uses more secure cryptographic protocols and algorithms. It's currently the preferred protocol for securing a wide variety of Layer 5+ protocol–based communications.

Secure Sockets Layer (SSL) and *Transport Layer Security (TLS)* are used to encrypt traffic between a web browser and a web server. Through the use of SSL or TLS, web surfers can make online purchases, interact with banks, and access private information without disclosing the contents of their communications. SSL and TLS can make web transactions private and secure. Although they aren't true VPN protocols, SSL and TLS operate in much the same manner as VPNs.

SSL was originally developed by Netscape, but it quickly became an Internet standard and has been replaced by TLS. TLS is based on SSL, but the two aren't interoperable. SSL operates over TCP port 443, whereas TLS can operate over either of the default TCP ports, 443 and 80 (as does HTTP).

In addition to web communications, SSL can be used to secure FTP, Network News Transfer Protocol (NNTP), email, Telnet, and other Application layer TCP/IP protocols. However, when SSL is used for protecting other application protocols, the destination port is different than that of HTTPS, which uses 443; other examples include SMTP over SSL at 465, IMAP over SSL at 993, and POP3 over SSL at 995.

SSL/TLS can also be used to provide encrypted sessions for other Application layer protocols, such as Telnet, FTP, and email. SSL/TLS functions at the top of Layer 4 (the Transport layer) of the OSI model. Thus, any protocol in Layers 5–7 can be secured using SSL/TLS.

When you use SSL/TLS to secure communications between a web browser and a web server, a multistep handshake process must be completed to establish the secured session:

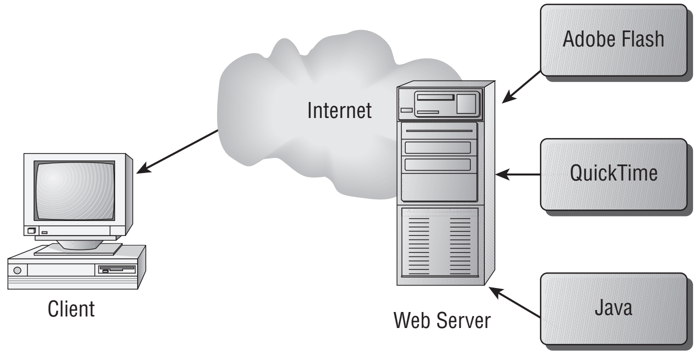
1. The client requests a secure connection.
2. The server responds with its certificate, the name of its certificate authority (issuing CA), and its public key.
3. The client verifies the server's certificate, produces a session (symmetric) encryption key, encrypts the key with the server's public key, and sends the encrypted key to the server.
4. The server unpacks the session key and sends a summary of the session details to the client, encrypted with the session key.
5. The client reviews the summary and sends its own summary back to the server, likewise encrypted with the session key.
6. After both entities receive a matching session summary, secured SSL communications are initiated.

SSL/TLS uses symmetric keys as the session keys. The session keys available for SSL include 40-bit and 128-bit strengths. TLS session keys can currently span between 128 bit and 256 bit.

HTTPS

The World Wide Web is a vast, global, ad hoc collection of online information and storefronts. The primary protocol that supports the Web is *Hypertext Transfer Protocol (HTTP)*. HTTP enables the transmission of *Hypertext Markup Language (HTML)* documents (the base page elements of a website) and embedded multimedia components such as graphics and mobile code (see Figure 2.45). Without HTTP, there would be no Web. However, HTTP is an insecure protocol: it doesn't offer anything in the way of secure authentication or data encryption for web communications. Fortunately, numerous add-on protocols and mechanisms provide these and other security services to the information superhighway.

FIGURE 2.45 A web server providing streaming video, animations, and HTML data to a client



HTTP operates over TCP port 80. It's a plain-text or clear-text communication protocol; thus, it offers no security or privacy to transactions. When SSL or TLS is used to secure transactions, this is known as *Hypertext Transfer Protocol over SSL (HTTPS)* or *Hypertext Transfer Protocol Secured (HTTPS)*. You can recognize when secure web communications are occurring using SSL or TLS because the URL begins with `HTTPS` and a locked padlock icon appears in the status bar at the bottom of the browser window.

It's important not to confuse HTTPS with a similarly named protocol, Secure HTTP (S-HTTP). S-HTTP isn't in widespread use. The primary differences are that S-HTTP doesn't use SSL; it encrypts individual web page elements rather than the entire web communication session using SSH, and it can only be used to support HTTP. Overall, S-HTTP is less secure than HTTPS.

Secure POP/IMAP

Post Office Protocol (POP) and Internet Message Access Protocol (IMAP) are secured by implementing TLS (or SSL in the past) encryption. This converts these protocols into POPS (or POP3S) and IMAPS (or IMAP4S) and also alters their ports from 110 to 995 and 143 to 993, respectively. POP and IMAP are email retrieval protocols, unlike SMTP which is an email sending or forwarding protocol. When using SMTP, POP, or IMAP in their TLS encrypted form, the security is only between the email client and the local email server. Any subsequent email communications between email servers may or may not be using encrypted SMTP. But those communication pathways are not under the control of the email client system, but of the administrators of each SMTP server.

Use cases

There is an ever-expanding collection of use cases where data transfers, audio communications, or other information exchanges are in need of secure protocols. This section highlights several of these that may be mentioned on the Security+ exam.

Voice and video

Voice communications have shifted from traditional landlines to mobile phones employing cellular to VoIP. The trend toward using VoIP as a business phone, home phone, and mobile phone will only continue to increase. Although a few VoIP solutions can provide end-to-end encryption, this does not seem to be the industry standard, and often encryption is not possible between dissimilar systems. There is a need for a more universal or assured VoIP encryption solution. Whenever you have the option to employ an end-to-end encryption VoIP system, it will be a better security choice than any other.

Time synchronization

Time synchronization is an important element of security management as well as overall network and system management. Many essential functions and services are dependent on reliable time information. The protocol Network Time Protocol (NTP), which operates over UPD port 123, is used to synchronize system clocks with each other and with an external reliable time source.

NTP is a plain-text protocol, so there is risk associated with using NTP on its own. There are NTP attacks that may simply eavesdrop on the time synchronization events. This can give an attacker more information about the systems on your network, since the NTP connection will disclose the source and destination addresses of the systems involved.

While some cryptography is available in NTPv4, it is mostly for integrity checking and not for confidentiality. It may be necessary to implement IPSec or other VPN sessions between systems and then tunnel the NTP through the encrypted channel.

Email and web

Email and web communications can both be easily protected using TLS-encrypted forms of their respective protocols. See the discussions in the earlier sections “S/MIME,” “SSL/TLS,” “HTTPS,” and “Secure POP/IMAP.”

File transfer

Secure file transfer is easily accomplished using either SSH or TLS-encrypted forms of FTP; see the sections “SSH,” “FTPS,” “SFTP,” and “SSL/TLS” earlier in this chapter. There are numerous other alternate file transfer protocols as well; some offer encryption, but not all. So be sure to investigate the security features before implementing or using any file transfer solution.

One option is to use a P2P (peer-to-peer) solution, such as BitTorrent. BitTorrent is one example of a P2P solution that provides communication encryption as well as integrity checking of delivered data.

It is also important to evaluate the internal file transfer protocols employed by your operating systems or third-party applications. It is fairly common to use the native Windows service of Server Message Block (SMB) or the Network File System (NFS) on Linux and Unix. These file transfer tools are convenient but they are plain text. You should establish an IPSec or other VPN connection, and then encapsulate the file transfer session within the VPN in order to gain security for these tools.

Directory services

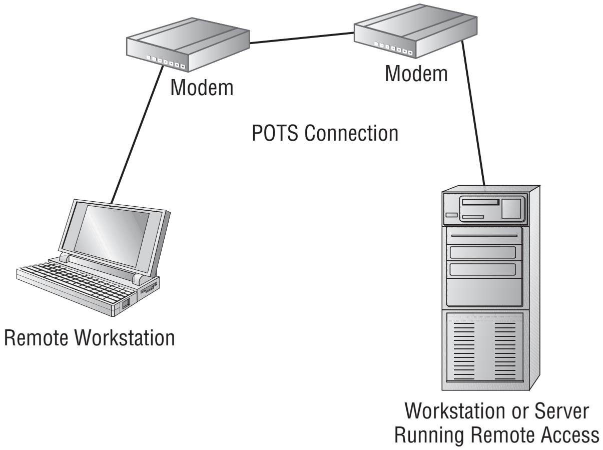
A directory service is a key feature of modern networks. Most clients should support the secured form of directory service interaction, such as LDAPS (or SASL), and thus this secure means should be required. Please see the earlier section "LDAPS."

Remote access

A *remote access server (RAS)* is a network server that supports connections from distant users or systems. RAS systems often support modem banks, VPN links, and even terminal services connections.

A *modem* (see Figure 2.46) is a device that creates a network communication link between two computers (or networks) over a telephone line. Modems are one of the slowest remote-connection methods still widely supported by OSs. Most connections are limited to a maximum throughput of 56 Kbps. However, because portable systems can use them to connect to corporate offices using any available telephone line, modems will probably be around for years to come.

FIGURE 2.46 A RAS connection between a remote workstation and a Windows server using modems



A common security protection added to dial-up modems is *callback*, a feature that disconnects the remote user immediately after authentication and then calls back the remote user at a predefined number. Callback ensures that the authenticated user is located at the correct phone number before access to the network is granted.

War dialing is a common attack against dial-up modems on a company network. Such an attack dials all the numbers in a prefix range in order to locate modems connected to computer systems. Once attackers locate a modem that answers a computer call, they can focus their efforts on breaking through the logon security barrier.

As networks grow, it becomes more common for them to support remote connections, whether dial-up, wireless wide area networks (WWANs), or virtual private networks (VPNs). The access-control and -protection issues involved in managing and administering remote access connections are generally called *communications security*.

Networks exist to share resources. In order to share resources, all entities on a network must share a common protocol. But in order for the protocol to function, a communication medium must be in place to provide support for the transfer of that protocol and its hosted communication data between one system and another. Often that medium is a network cable, such as a *Cat5e* (also known as *twisted-pair cabling*).

However, the communication medium could be wireless, a VPN link, a dial-up link, a terminal services link, or even a remote-control link. In any case, understanding the technology and the security implications of each of these communication media is an essential part of administering an environment.

One mechanism often used to help control the complexities of remote connectivity is a remote access policy. Remote access server policies (RAS policies) are additional gauntlets of requirements that remote users must be in compliance with to gain access to the internal resources of the LAN. RAS policies can require specific OSs and patch levels, restrict time and date access, mandate authentication mechanisms, and confirm the caller ID and/or MAC address of the remote client. After a connection is established, RAS policies can be used to enforce idle timeout disconnects, define the maximum connect time, mandate minimal encryption levels, enforce IP packet filters, define IP address parameters, and force specific routing paths.

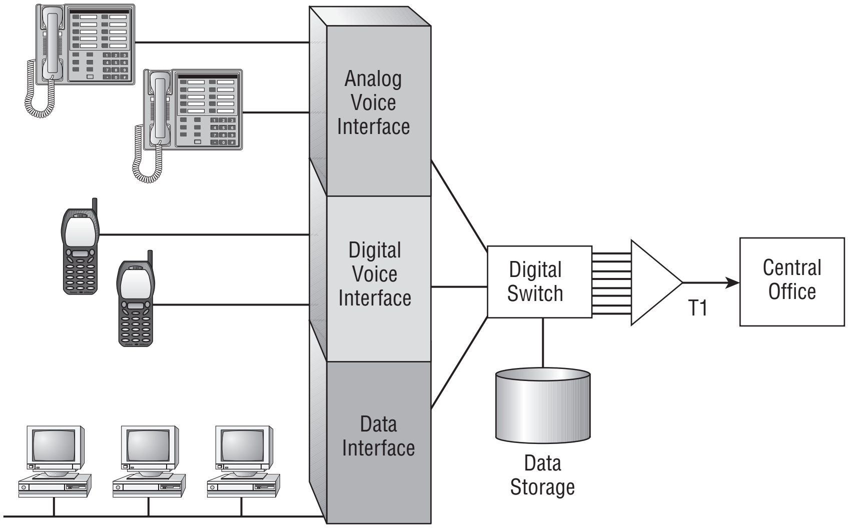
Remote authentication is a catchphrase that refers to any mechanism used to verify the identity of remote users. Several well-known examples of remote authentication include RADIUS, TACACS, 802.1x, and Challenge Handshake Authentication Protocol (CHAP). Originally, remote authentication referred to solutions that supported authentication mechanisms for dial-up telecommuters. Today, it includes any authentication technology that can be used for remote users, whether connecting over dial-up, VPN, or wireless.

Telephony is the collection of methods by which telephone services are provided to an organization or the mechanisms by which an organization uses telephone services for either voice and/or data communications. Traditionally, telephony included plain old telephone service (POTS) or public switched telephone network (PSTN) service combined with modems. However, this has expanded to include PBX, VoIP, and VPN.

A *private branch exchange (PBX)* (also known as *telecom*), shown in Figure 2.47, is a computer- or network-controlled telephone system. PBXs are deployed in large organizations; they offer a wide range of telephone services, features, and capabilities, including conference calls, call forwarding, paging, call logging, voicemail, call routing, and remote calling.

Remote calling is the ability to dial in to a PBX system from outside and then access a dial tone in order to place a call. The second call can be long distance, and all toll charges are accumulated on the PBX system, not on the user's telephone. This is a commonly attacked feature of PBX systems.

FIGURE 2.47 A modern digital PBX system integrating voice and data onto a single network connection



Methods to secure PBX systems include the following:

- Disabling maintenance features
- Changing all default passwords, accounts, and access codes
- Enabling logging
- Restricting long-distance calling
- User awareness and training

Voice over IP (VoIP) is a tunneling mechanism used to transport voice and/or data over a TCP/IP network. VoIP has the potential to replace or supplant PSTN because it's often less expensive and offers a wider variety of options and features. VoIP can be used as a direct telephone replacement on computer networks as well as mobile devices. However, VoIP is able to support video and data transmission to allow videoconferencing and remote collaboration on projects. VoIP is available in both commercial and open-source options. Some VoIP solutions require specialized hardware to either replace traditional telephone handsets/base stations or allow these to connect to and function over the VoIP system. Some VoIP solutions are software only, such as Skype, and allow the user's existing speakers, microphone, or headset to replace the traditional telephone handset. Others are more hardware based, such as magicJack, which allows the use of existing PSTN phone devices plugged into a USB adapter to take advantage of VoIP over the Internet. Often, VoIP-to-VoIP calls are free (assuming the same or compatible VoIP technology), whereas VoIP-to-landline calls are usually charged a per-minute fee. While most VoIP protocols support encryption, it is often disabled by default or not available due to connecting to a noncompatible recipient (such as calling from a computer VoIP tool to a landline.

Domain name resolution

DNS security should be taken seriously. If your DNS resolutions are being falsified or monitored, your system and/or organization can be harmed. DNSSEC is a means to improve the security of DNS resolutions; please see the earlier section "DNSSEC" for that discussion.

Routing and switching

The communications between routers and switches are yet another potential target for attackers. If they are able to interfere with the convergence of routing tables or STP, then attackers can either redirect traffic down pathways to which they have physical or logical access or they can simply implement a DoS. Employing VPNs or other encryption services between network management devices can reduce these risks.

Network address allocation

Subnetting is a divisioning process used on networks to divide larger groups of hosts into smaller collections. The act of subnetting may be mandated by the maximum size of a subnet based on desired IP class restrictions, physical limitations, differentiation of business functions, or other concerns. Subnetting is mainly a logical activity, but it can be used to direct or guide physical divisioning. In fact, many large organizations mimic their logical subnetting infrastructure in their physical deployment for easier troubleshooting and maintenance.

NOTE

Subnet size is no longer strictly limited to the IP class range restrictions, such as only 254 hosts per Class C network, if Classless Inter-Domain Routing (CIDR) subnetting is used. This topic isn't directly relevant to Security+, because it's a Network+ topic, so please search for the term CIDR on the Internet for more information.

Ultimately, in the TCP/IPv4 protocol, subnetting is defined by the assigned host IP address and its related subnet mask. The subnet mask is a 32-bit binary number that indicates which portions of a host IP address (also a 32-bit binary number, at least for TCP/IPv4) define the network ID (or subnet ID) and which portions define the host ID. Network and subnet IDs are unique within each organization's private network or across the public Internet. Host IDs are unique only within the local subnet. In much the same way that an area code defines the general area where a phone number resides, a network ID defines where a subnet resides. Within one area code and another, there are duplicate seven-digit phone numbers, and within multiple subnets there are duplicate host IDs. However, unlike phone numbers, IP addresses are always presented with their entire complement of numbers and, when necessary or important, their related subnet mask.

As an example, IP address 193.25.172.56 with a subnet mask of 255.255.0.0 can be converted from this dotted decimal notation to binary as follows:

IP Address	11000001000110011010110000111000
Subnet Mask	11111111111111110000000000000000
Network ID	11000001000110010000000000000000
Host ID	00000000000000001010110000111000

By reading only the portions of the IP address marked or masked by the 1s from the subnet mask, the network ID is revealed: 11000001000110010000000000000000, or 193.25.0.0.

By reading only the portions of the IP address marked or masked by the 0s from the subnet mask, the host ID is revealed: 00000000000000001010110000111000, or 0.0.172.56.

A host within a subnet is able to communicate directly with any other host in that same subnet. However, to communicate with hosts in other subnets, traffic must be directed out of the subnet toward the destination host's subnet. This is done by sending the data stream to the default gateway of the local subnet. The default gateway is just the interface of a router in your local subnet. The router then reads the destination IP address and directs the traffic toward its destination subnet.

You can use subnetting to control communications, block access, divide security zones, and much more. This is only a general and generic overview of the topic. If you aren't already familiar with how to subnet TCP/IP, please consult Network+ study materials or search for this content online.

The allocation of network addresses should be considered carefully. Under IPv4, it is still important to use the private IP addresses from RFC 1918 internally. This will prevent external entities from initiating communications with your internal systems. However, this will require that you implement NAT/PAT in order to support communications from internal systems to external systems. The use of RFC 1918 and NAT/PAT will reduce the ability of an external entity to easily determine your network size and internal address allocation.

Another issue with IPv4 is that there are very few remaining available public IP addresses. This is a problem, since a server typically needs its own public IP address. Some services, like the Web, can host multiple sites on the same web server host IP address, but not every service can be configured this way. Clients can be expanded potentially indefinitely using NAT. But once there are no more remaining available public IPv4 addresses, no new servers can come online in that address space. The migration to IPv6 is essential to the future expansion of the Internet and other internetworking technologies.

Subscription services

Subscription services are becoming a common tool employed by businesses and individuals alike. Subscriptions for all types of services are gaining widespread support; these include email, document editing, cloud storage, cloud backup, gaming, video entertainment, VoIP, and remote hosting.

No matter what subscription service is in use, care should be taken to ensure that the connection between the online service server and the client/subscriber/customer system is encrypted. It is also important to determine whether the online service provides proper security for the customer database and any files or data stored online.

Exam Essentials

Understand TCP/IP. TCP/IP is the primary protocol suite in use on the Internet and most private networks across the planet.

Know IPv4. IPv4 is in widespread use with a 32-bit addressing scheme and operates at the Network layer or Layer 3 of the OSI protocol stack.

Understand IPv6. IPv6 uses a 128-bit addressing scheme, eliminates broadcasts and fragmentation, and includes native communication-encryption features.

Be familiar with DNS. DNS is the hierarchical naming scheme used in both public and private networks. It links IP addresses and human-friendly fully qualified domain names (FQDNs) together.

Understand DNSSEC. DNSSEC (Domain Name System Security Extensions) is a security improvement to the existing DNS infrastructure. The primary function of DNSSEC is to provide reliable authentication between devices when performing DNS operations.

Comprehend SSH. Secure Shell (SSH) is a secure replacement for Telnet, rlogon, rsh, and RCP. It can be called a remote-access or remote-terminal solution. SSH encrypts authentication and data traffic, and it operates over TCP port 22.

Understand Telnet. Telnet is a terminal-emulation network application that supports remote connectivity for executing commands and running applications but doesn't support transfer of files. Telnet uses TCP port 23.

Know the common applications of cryptography to secure electronic mail. The emerging standard for encrypted messages is the Secure/Multipurpose Internet Mail Extensions (S/MIME) protocol. The other popular email security protocol is Phil Zimmerman's Pretty Good Privacy (PGP).

Understand SRTP SRTP (Secure Real-Time Transport Protocol or Secure RTP) is a security improvement over Real-Time Transport Protocol (RTP) that is used in many Voice over IP (VoIP) communications. SRTP aims to minimize the risk of VoIP DoS through robust encryption and reliable authentication.

Know LDAP. Lightweight Directory Access Protocol (LDAP) is used to allow clients to interact with directory service resources. LDAP is based on x.500 and uses TCP ports 389 and 636. It uses a tree structure with a district root.

Understand LDAPS. LDAPS (LDAP Secured) is accomplished by enabling the Simple Authentication and Security Layer (SASL) on LDAP, which implements Transport Layer Security (TLS) on the authentication of clients as well as all data exchanges.

Be aware of FTP. File Transport Protocol (FTP) is an in-the-clear file-exchange solution. An FTP server system is configured to allow authenticated or anonymous FTP clients to log on in order to upload or download files. FTP employs TCP ports 20 and 21.

Understand FTPS. FTPS is FTP Secure or FTP SSL, which indicates that it's a variation of FTP secured by SSL (or now TLS). This FTP service variation is distinct from SFTP, which is SSH-secured FTP.

Understand SFTP Secure FTP (SFTP) is a secured alternative to standard or basic FTP that encrypts both authentication and data traffic between the client and server. SFTP employs SSH to provide secure FTP communications.

Know TFTP. Trivial File Transfer Protocol (TFTP) is a simple file-exchange protocol that doesn't require authentication. It operates on UDP port 69.

Understand SNMP. Simple Network Management Protocol (SNMP) is a standard network-management protocol supported by most network devices and TCP/IP-compliant hosts. These include routers, switches, bridges, WAPs, firewalls, VPN appliances, modems, printers, and so on.

Be familiar with SSL and TLS. Secure Sockets Layer (SSL) and Transport Layer Security (TLS) are used to encrypt traffic between a web browser and a web server. TLS is the updated replacement for Netscape's SSL. Through the use of SSL or TLS, web surfers can make online purchases, interact with banks, and access private information without disclosing the contents of their communications. SSL and TLS can make web transactions private and secure.

Understand HTTPS. When SSL or TLS is used to secure transactions, it's known as Hypertext Transfer Protocol over SSL or Hypertext Transfer Protocol Secured (HTTPS).

Be aware of secure POP/IMAP. Securing Post Office Protocol (POP) and Internet Message Access Protocol (IMAP) is accomplished by implementing TLS (or SSL in the past) encryption. This converts these protocols into POPS (or POP3S) and IMAPS (or IMAP4S) and also alters their ports from 110 to 995 and 143 to 993, respectively.

Review Questions

You can find the answers in the Appendix.

1. You are implanting a new network for a small office environment. The network includes a domain controller, four resource servers, a network printer, a wireless access point, and three dozen client systems. In addition to standard network management devices, such as switches and routers, why would you want to deploy a firewall?

A. To watch for intrusions

B. To control traffic entering and leaving a network

C. To require strong passwords

D. To prevent misuse of company resources

2. As the security administrator for a moderate-sized network, you need to deploy security solutions to reduce the risk of a security breach. You elect to install a network-based IDS. However, after deployment you discover that the NIDS is not suitable for detecting which of the following?

A. Email spoofing

B. Denial-of-service attacks

C. Attacks against the network

D. Attacks against an environment that produces significant traffic

3. Illegal or unauthorized zone transfers are a significant and direct threat to what type of network server?

A. Web

B. DHCP

C. DNS

D. Database

4. What mechanism of loop protection is based on an element in a protocol header?

A. Spanning Tree Protocol

B. Ports

C. Time to live

D. Distance vector protocols

5. What type of wireless antenna can be used to send or receive signals in any direction?

A. Cantenna

B. Yagi

C. Rubber duck

D. Panel

6. What mechanism of wireless security is based on AES?
   - A. TKIP
   - B. CCMP
   - C. LEAP
   - D. WEP

7. What technology provides an organization with the best control over BYOD equipment?
   - A. Encrypted removable storage
   - B. Mobile device management
   - C. Geotagging
   - D. Application whitelisting

8. What is the most effective means to reduce the risk of losing the data on a mobile device, such as a notebook computer?
   - A. Encrypt the hard drive.
   - B. Minimize sensitive data stored on the mobile device.
   - C. Use a cable lock.
   - D. Define a strong logon password.

9. Which security stance will be most successful at preventing malicious software execution?
   - A. Deny by exception
   - B. Whitelisting
   - C. Allow by default
   - D. Blacklisting

10. LDAP operates over what TCP ports?
    - A. 636 and 389
    - B. 110 and 25
    - C. 443 and 80
    - D. 20 and 21

11. What type of NAC agent is written in a web or mobile language and is temporarily executed on a system only when the specific management page is accessed?
    - A. Permanent
    - B. Dissolvable
    - C. Passive
    - D. Stateless

12. What is the purpose or use of a media gateway?

A. It is a fictitious environment designed to fool attackers and intruders and lure them away from the private secured network.

B. It is used to connect several network segments and enable traffic from one network segment to traverse into another network segment.

C. It is used to spread or distribute network traffic load across several network links or network devices.

D. It is any device or service that converts data from one communication format to another.

13. Which of the following is true regarding an exploitation framework? (Select all that apply.)

A. Is a passive scanner

B. Fully exploits vulnerabilities

C. Only operates in an automated fashion

D. Allows for customization of test elements

E. Represents additional risk to the environment

F. Can only assess systems over IPv4

14. What is the purpose of a banner grabbing activity?

A. Detecting the presence of a wireless network

B. Capturing the initial response or welcome message from a network service that may directly or indirectly reveal its identity

C. Preventing access to a network until the client has accepted use terms or fully authenticated

D. Altering the source IP address of an outbound request

15. How are effective permissions determined or calculated?

A. Accumulate allows, remove any denials

B. Count the number of users listed in the ACL

C. View the last access time stamp of the asset

D. Review the user’s group memberships

16. What is a content filter mechanism that can reduce the possibility of malicious executable code being accepted as input?

A. Checking length

B. Blocking hex characters

C. Escaping metacharacters

D. Filtering on known patterns of malicious content

17. What is an example of a PUP?

A. A backdoor

B. Unwanted marketing pop-ups

C. A Trojan horse

D. A password cracker

18. What is the purpose of DEP being present in an operating system?

A. To block buffer overflows

B. To prevent social-engineering attacks

C. To stop ransomware infections

D. To interrupt backdoor installations

19. What is the term used to describe the designation of a specific geographical area that is then used to implement features on mobile devices, which can be defined by GPS coordinates, a wireless indoor positioning system (IPS), or the presence or lack of a specific wireless signal?

A. Bluesmacking

B. Geofencing

C. Banner grabbing

D. CYOD

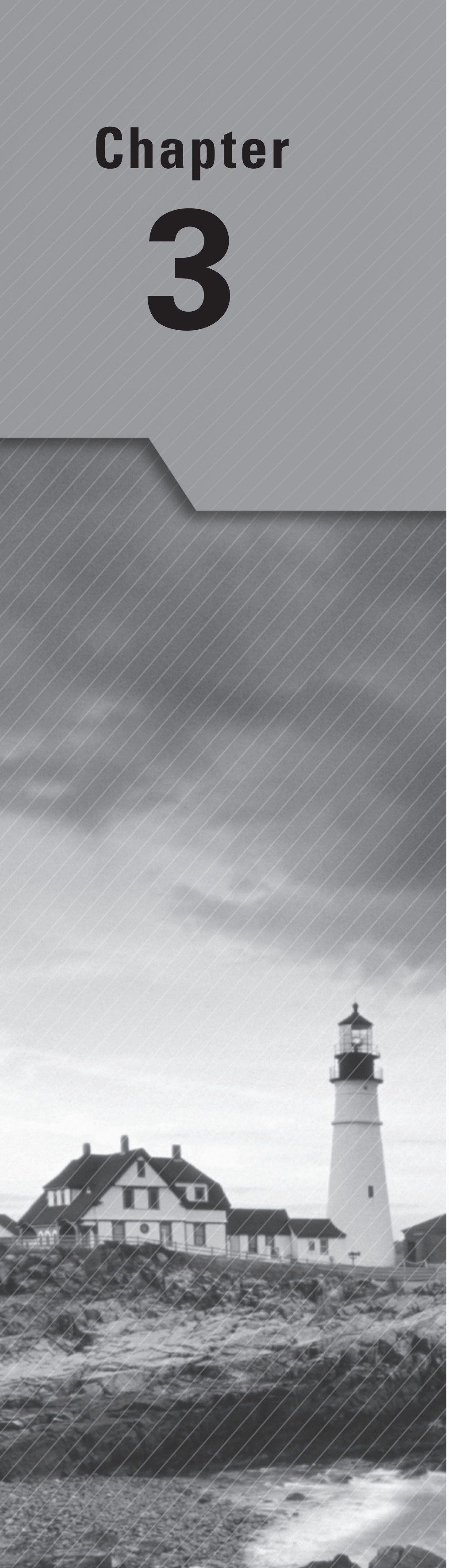
20. What is the definition of DNSSEC?

A. It is an Internet standard for encrypting and digitally signing email.

B. It can be used as a secure Telnet replacement, it can be used to encrypt protocols similar to TLS, and it can be used as a VPN protocol.

C. It is a standard network-management protocol supported by most network devices and TCP/IP-compliant hosts used to obtain status information, performance data, statistics, and configuration details.

D. It is a security improvement to the existing name resolution infrastructure. The primary function of this tool is to provide reliable authentication between devices when performing resolution operations.

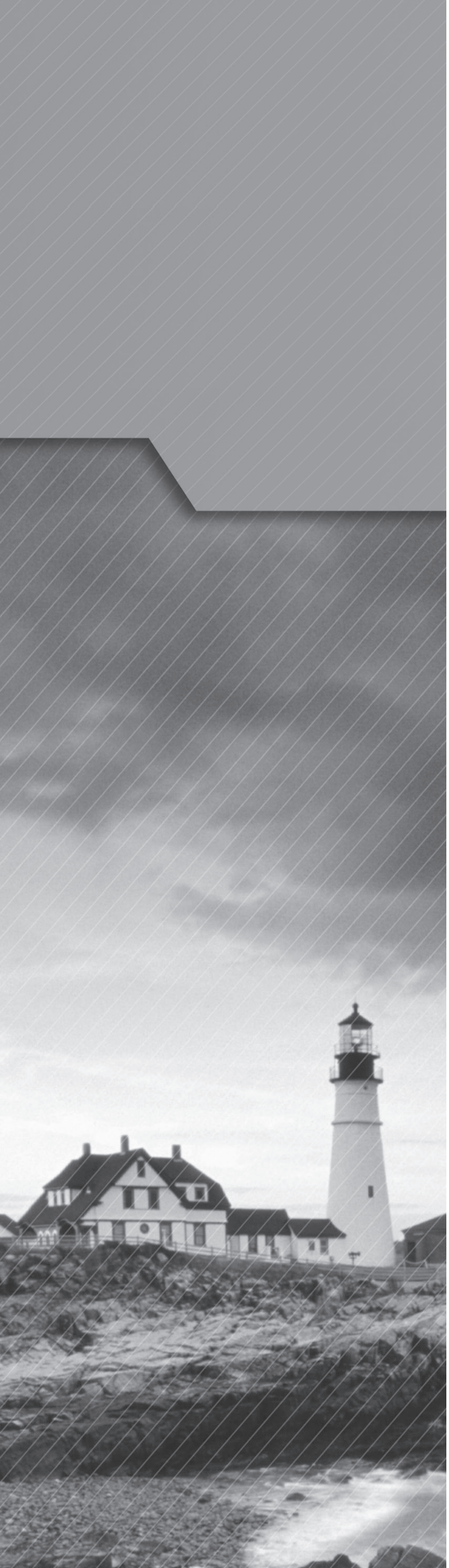


Architecture and Design

COMPTIA SECURITY+ EXAM OBJECTIVES COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:

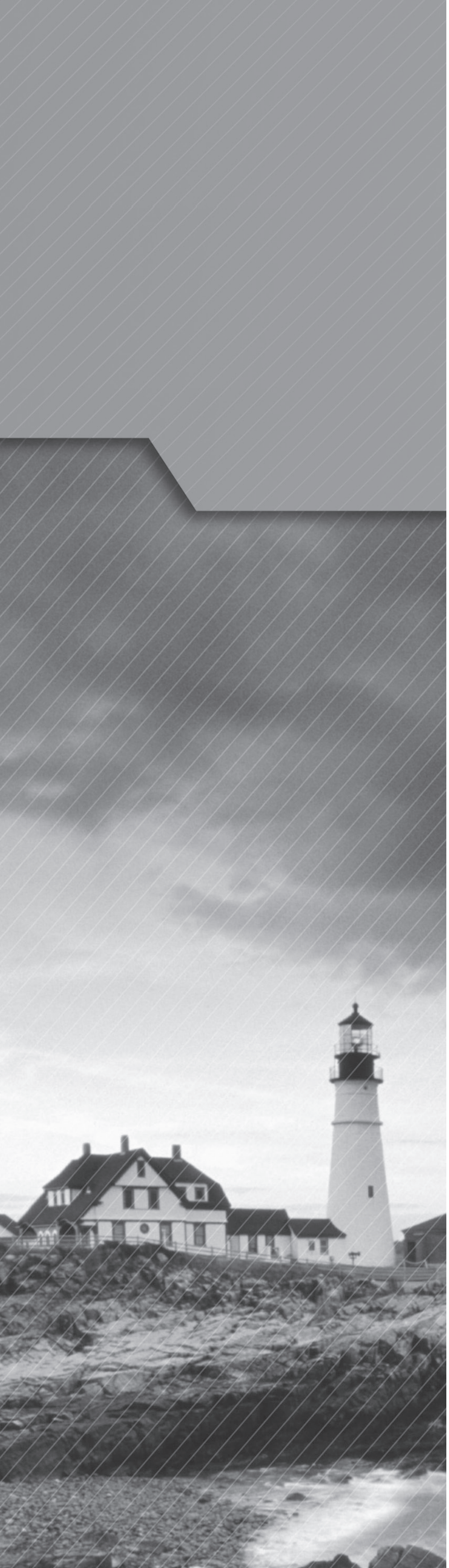
✓ **3.1 Explain use cases and purpose for frameworks, best practices and secure configuration guides.**

- Industry-standard frameworks and reference architectures
  - Regulatory
  - Non-regulatory
  - National vs. international
  - Industry-specific frameworks
- Benchmarks/secure configuration guides
  - Platform/vendor-specific guides
    - Web server
    - Operating system
    - Application server
    - Network infrastructure devices
  - General purpose guides
- Defense-in-depth/layered security
  - Vendor diversity
  - Control diversity
    - Administrative
    - Technical
  - User training



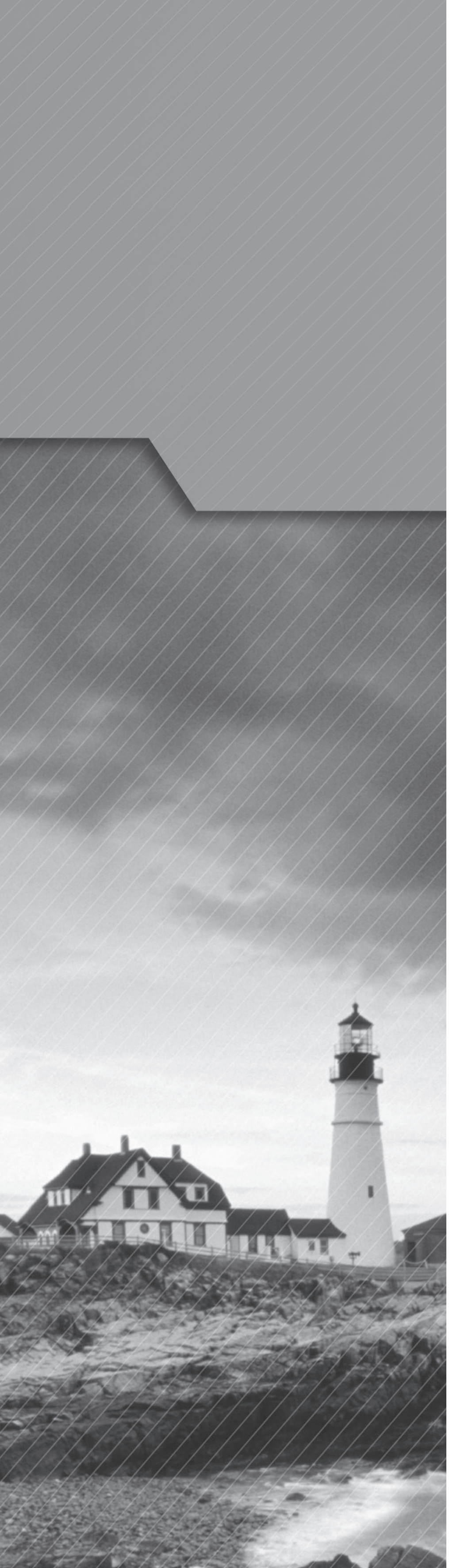
✓ **3.2 Given a scenario, implement secure network architecture concepts.**

- Zones/topologies
  - DMZ
  - Extranet
  - Intranet
  - Wireless
  - Guest
  - Honeynets
  - NAT
  - Ad hoc
- Segregation/segmentation/isolation
  - Physical
  - Logical (VLAN)
  - Virtualization
  - Air gaps
- Tunneling/VPN
  - Site-to-site
  - Remote access
- Security device/technology placement
  - Sensors
  - Collectors
  - Correlation engines
  - Filters
  - Proxies
  - Firewalls
  - VPN concentrators
  - SSL accelerators
  - Load balancers
  - DDoS mitigator
  - Aggregation switches
  - Taps and port mirror
- SDN



✓ **3.3 Given a scenario, implement secure systems design.**

- Hardware/firmware security
  - FDE/SED
  - TPM
  - HSM
  - UEFI/BIOS
  - Secure boot and attestation
  - Supply chain
  - Hardware root of trust
  - EMI/EMP
- Operating systems
  - Types
    - Network
    - Server
    - Workstation
    - Appliance
    - Kiosk
    - Mobile OS
  - Patch management
  - Disabling unnecessary ports and services
  - Least functionality
  - Secure configurations
  - Trusted operating system
  - Application whitelisting/blacklisting
  - Disable default accounts/passwords
- Peripherals
  - Wireless keyboards
  - Wireless mice
  - Displays
  - WiFi-enabled MicroSD cards



    - Printers/MFDs
    - External storage devices
    - Digital cameras

✓ **3.4 Explain the importance of secure staging deployment concepts.**

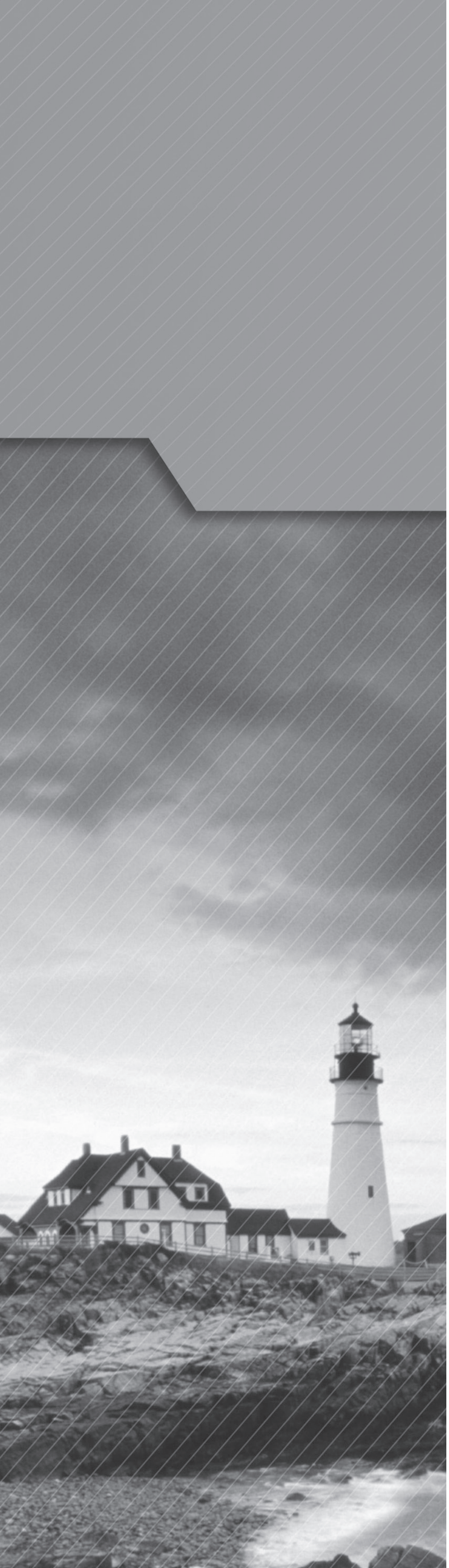
- Sandboxing
- Environment
    - Development
    - Test
    - Staging
    - Production
- Secure baseline
- Integrity measurement

✓ **3.5 Explain the security implications of embedded systems.**

- SCADA/ICS
- Smart devices/IoT
    - Wearable technology
    - Home automation
- HVAC
- SoC
- RTOS
- Printers/MFDs
- Camera systems
- Special purpose
    - Medical devices
    - Vehicles
    - Aircraft/UAV

✓ **3.6 Summarize secure application development and deployment concepts.**

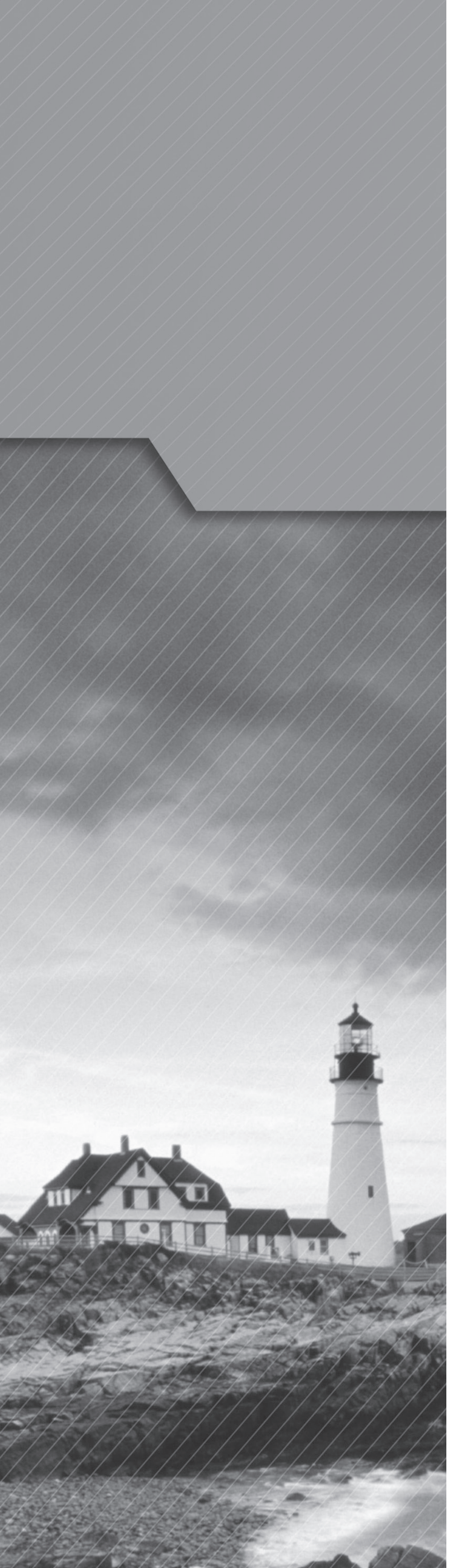
- Development life-cycle models
    - Waterfall vs. Agile



- Secure DevOps
  - Security automation
  - Continuous integration
  - Baselining
  - Immutable systems
  - Infrastructure as code
- Version control and change management
- Provisioning and deprovisioning
- Secure coding techniques
  - Proper error handling
  - Proper input validation
  - Normalization
  - Stored procedures
  - Code signing
  - Encryption
  - Obfuscation/camouflage
  - Code reuse/dead code
  - Server-side vs. client-side execution and validation
  - Memory management
  - Use of third-party libraries and SDKs
  - Data exposure
- Code quality and testing
  - Static code analyzers
  - Dynamic analysis (e.g., fuzzing)
  - Stress testing
  - Sandboxing
  - Model verification
- Compiled vs. runtime code

✓ **3.7 Summarize cloud and virtualization concepts.**

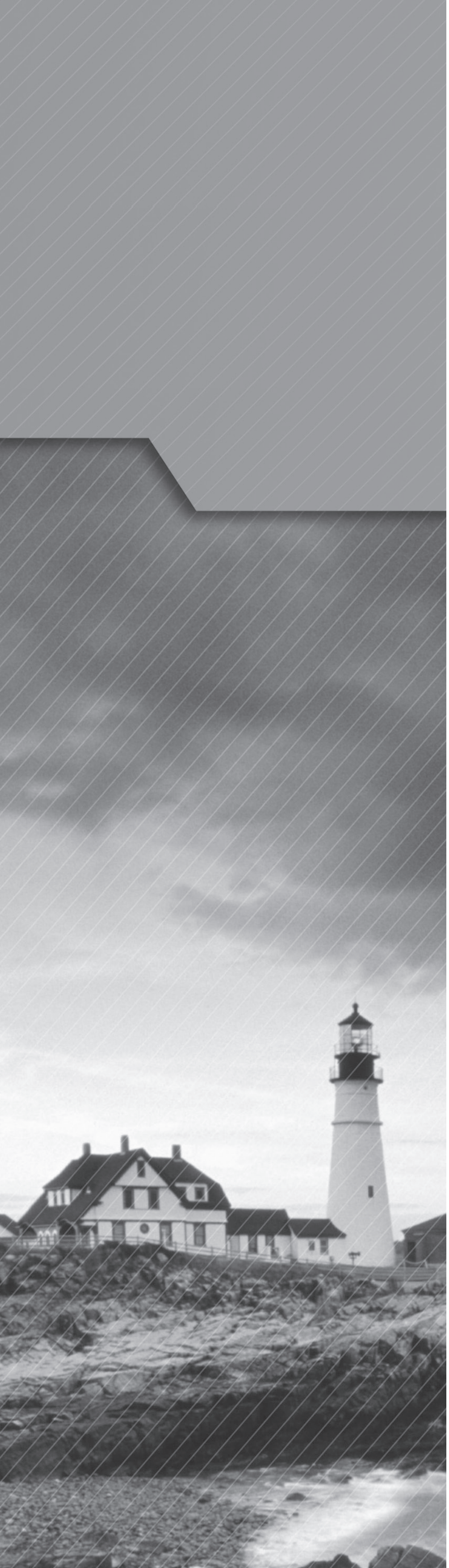
- Hypervisor
  - Type I
  - Type II
  - Application cells/containers



- VM sprawl avoidance
- VM escape protection
- Cloud storage
- Cloud deployment models
    - SaaS
    - PaaS
    - IaaS
    - Private
    - Public
    - Hybrid
    - Community
- On-premise vs. hosted vs. cloud
- VDI/VDE
- Cloud access security broker
- Security as a Service

✓ **3.8 Explain how resiliency and automation strategies reduce risk.**

- Automation/scripting
    - Automated courses of action
    - Continuous monitoring
    - Configuration validation
- Templates
- Master image
- Non-persistence
    - Snapshots
    - Revert to known state
    - Rollback to known configuration
    - Live boot media
- Elasticity
- Scalability
- Distributive allocation
- Redundancy



- Fault tolerance
- High availability
- RAID

✓ **3.9 Explain the importance of physical security controls.**

- Lighting
- Signs
- Fencing/gate/cage
- Security guards
- Alarms
- Safe
- Secure cabinets/enclosures
- Protected distribution/Protected cabling
- Airgap
- Mantrap
- Faraday cage
- Lock types
- Biometrics
- Barricades/bollards
- Tokens/cards
- Environmental controls
  - HVAC
  - Hot and cold aisles
  - Fire suppression
- Cable locks
- Screen filters
- Cameras
- Motion detection
- Logs
- Infrared detection
- Key management

The Security+ exam will test your understanding of the architecture and design of an IT environment and its related security. To pass the test and be effective in implementing security, you need to understand the basic concepts and terminology related to network security design and architecture as detailed in this chapter.

3.1 Explain use cases and purpose for frameworks, best practices and secure configuration guides.

Security is complicated. The task of designing and implementing security can be so daunting that many organizations may put it off until it's too late and they have experienced a serious intrusion or violation. A means to simplify the process, or at least to get started, is to adopt predefined guidance and recommendations from trusted entities. There are many government, open-source, and commercial security frameworks, best practices, and secure configuration guides that can be used as both a starting point and a goalpost for security programs for large and small organizations.

Industry-standard frameworks and reference architectures

A security framework is a guide or plan for keeping your organizational assets safe. It provides a structure to the implementation of security for both new organizations and those with a long history. A security framework should provide perspective that security is not just an IT concern, but an important business operational function. A well-designed security framework should address personnel issues, network security, portable and mobile equipment, operating systems, applications, servers and endpoint devices, network services, business processes, user tasks, communications, and data storage.

Industry-standard frameworks are those that are adopted and respected by a majority of organizations within a specific line of business. A reference architecture may accompany a security framework. Often a reference architecture is a detailed description of a fictitious organization and how security could be implemented. This concept serves as a guide for real-world organizations to use as a template to follow for adapting and implementing a framework.

Some security frameworks are designed to help new organizations implement their initial and foundational security elements, whereas others are designed to improve the existing in-place security infrastructure.

Regulatory

A regulatory security framework is a security guidance established by a government regulation or law. Regulatory frameworks are thus crafted or sponsored by government agencies. However, this does not necessarily limit their use to government entities. Many regulatory frameworks are publicly available and thus can be adopted and applied to private organizations as well.

A security framework does not have to be designed specifically for an organization, nor does an entire framework need to be implemented. Each organization is unique and thus should use several frameworks to assemble a solution that addresses their specific security needs.

Non-regulatory

A nonregulatory security framework is any security guidance crafted by a nongovernment entity. This would include open-source communities as well as commercial entities. Nonregulatory frameworks may require a licensing fee or a subscription fee in order to view and access the details of the framework. Some commercial entities will even provide customized implementation guidance or compliance auditing.

National vs. international

A national security framework is any security guidance designed specifically for use within a particular country. The author of a national framework may attempt to restrict access to the details of their framework in order to control or limit implementation to just their local industries. National frameworks also may include country-specific limitations, requirements, utilities, or other concerns that are not applicable to any or most other countries. Such national nuances may also serve as a limiting factor for the use of such frameworks in other lands.

International security frameworks are designed on purpose to be nation independent. These are crafted with the goal of avoiding any country-specific limitations or idiosyncrasies in order to support worldwide adoption of the framework. Compliance with international security frameworks simplifies the interactions between organizations located across national borders by ensuring they have compatible and equivalent security protections.

Industry-specific frameworks

Industry-specific frameworks are those crafted to be applicable to one specific industry, such as banking, health care, insurance, energy management, transportation, or retail. These types of frameworks are tuned to address the most common issues within an industry and may not be as easily applicable to organizations outside of that target.

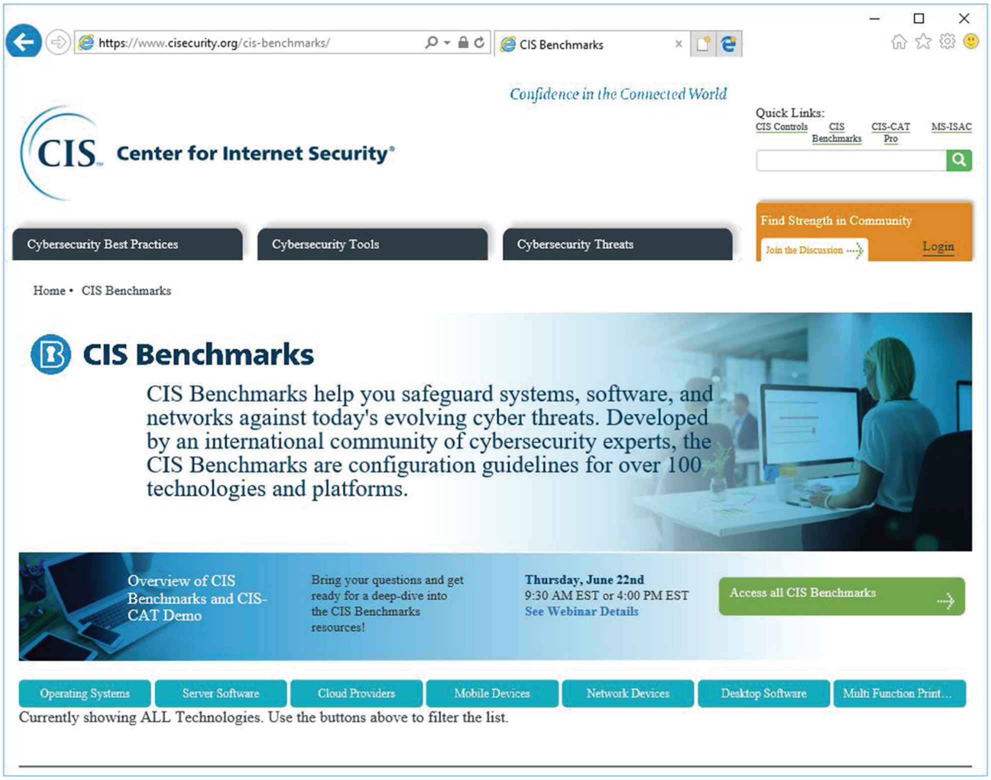
Security Frameworks Whitepaper

I wrote a white paper on security frameworks for Global Knowledge titled "Cybersecurity Frameworks to Consider for Organization-wide Integration." You might find it interesting—it includes details that are not exam relevant and thus not included here. You can access the white paper here: `https://www.globalknowledge.com/us-en/resources/resource-library/white-papers/cybersecurity-frameworks-to-consider-for-organization-wide-integration/`.

Benchmarks/secure configuration guides

A benchmark is a documented list of requirements that is used to determine whether or not a system, device, or software solution is allowed to operate within a securely management environment (Figure 3.1). A secure configuration guide is another term for a benchmark. It can also be known as a standard or a baseline.

FIGURE 3.1 The CIS Benchmarks website



CIS Benchmarks help you safeguard systems, software, and networks against today's evolving cyber threats. Developed by an international community of cybersecurity experts, the CIS Benchmarks are configuration guidelines for over 100 technologies and platforms.

Overview of CIS Benchmarks and CIS-CAT Demo

Bring your questions and get ready for a deep-dive into the CIS Benchmarks resources!

Thursday, June 22nd
9:30 AM EST or 4:00 PM EST
See Webinar Details

Currently showing ALL Technologies. Use the buttons above to filter the list.

A benchmark can include specific instructions on installation and configuration of a product. It may also suggest alterations, modifications, and supplemental tools, utilities, drivers, and controls to improve the security of the system. A benchmark may also recommend operational steps, SOPs (standard operation procedures), and end-user guides to maintain security while business tasks are taking place.

A benchmark can be adopted from external entities, such as government regulations, commercial guidance, or community recommendations. But ultimately, a benchmark should be customized for the organization's assets, threats, and risks.

CIS Security Benchmarks

There are several excellent online resources to see specific examples of OS, application, and hardware security configuration guides. Here are two you should investigate:

Center for Internet Security (CIS) at `https://www.cisecurity.org/cis-benchmarks/`

National Institute of Standards and Technology (NIST) National Vulnerability Database (NVD) National Checklist Program Repository at `https://nvd.nist.gov/ncp/repository`

Platform/vendor-specific guides

Security configuration guides are often quite specific to an operating system/platform, application, or product vendor. These types of guides can be quite helpful in securing a product since they may provide step-by-step, click-by-click, command-by-command instructions on securing a specific application, OS, or hardware product.

Web server

Benchmarks and security configuration guides can focus on specific web server products, such as Microsoft's Internet Information Service or Apache Web Server.

Operating system

Benchmarks and security configuration guides can focus on specific operating systems, such as Microsoft Windows, Apple Macintosh, Linux, or Unix.

Application server

Benchmarks and security configuration guides can focus on specific application servers, such as Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP), databases, Network Area Storage (NAS), Storage Area Network (SAN), directory services, virtual private network (VPN), or Voice over Internet Protocol (VoIP).

Network infrastructure devices

Benchmarks and security configuration guides can focus on specific network infrastructure devices, such as firewalls, switches, routers, wireless access points, VPN concentrators, web security gateways, virtual machines/hypervisors, or proxies.

General purpose guides

General-purpose security configuration guides are more generic in their recommendations rather than being focused on a single software or hardware product. This makes them useful in a wide range of situations, but they provide less detail and instruction on exactly how to accomplish the recommendations. A product-focused guide might provide hundreds of steps of configuring a native firewall, whereas a general-purpose guide may provide only a few dozen general recommendations. This type of guide leaves the specific actions to accomplish the recommendations up to the system manager to determine how to accomplish the goals or implement the suggestions.

Defense-in-depth/layered security

Defense in depth is the use of multiple types of access controls in literal or theoretical concentric circles or layers. This form of layered security helps an organization avoid a monolithic security stance. A monolithic mentality is the belief that a single security mechanism is all that is required to provide sufficient security.

Only through the intelligent combination of countermeasures can you construct a defense that will resist significant and persistent attempts at compromise. Intruders or attackers would need to overcome multiple layers of defense to reach the protected assets.

As with any security solution, relying on a single security mechanism is unwise. *Defense in depth*, *multilayered security*, or *diversity of defense* uses multiple types of access controls in literal or theoretical concentric circles or layers. By having security control redundancy and diversity, an environment can avoid the pitfalls of a single security feature failing; the environment has several opportunities to deflect, deny, detect, and deter any threat. Of course, no security mechanism is perfect. Each individual security mechanism has a flaw or a workaround just waiting to be discovered and abused by a hacker.

Vendor diversity

Vendor diversity is important for establishing defense in depth in order to avoid security vulnerabilities due to one vendor's design, architecture, and philosophy of security. No one vendor can provide a complete end-to-end security solution that protects against all known and unknown exploitations and intrusions. Thus, to improve the security stance of an organization, it is important to integrate security mechanisms from a variety of vendors, manufacturers, and programmers.

Control diversity

Control diversity is essential in order to avoid a monolithic security structure. Do not depend on a single form or type of security; instead, integrate a variety of security mechanisms into the layers of defense. Using three firewalls is not as secure as using a firewall, an IDS, and strong authentication.

Administrative

Administrative controls typically include security policies as well as mechanisms for managing people and overseeing business processes. It is important to ensure a diversity of administrative controls rather than relying on a single layer or single type of security mechanism.

Technical

Technical controls include any logical or technical mechanism used to provide security to an IT infrastructure. Technical security controls need to be broad and varied in order to provide a robust wall of protection against intrusions and exploit attempts. Single defenses, whether a single layer or repetitions of the same defense, can fall to a singular attack. Diverse and multilayered defenses require a more complex attack approach requiring numerous exploitations to be used in a series, successfully, without detection in order to compromise the target. The concept of attacking with a series of exploits is known as *daisy-chaining*.

User training

User training is always a key part of any security endeavor. Users need to be trained in how to perform their work tasks in accordance with the limitations and restrictions of the security infrastructure. Users need to understand, believe in, and support the security efforts of the organization; otherwise, users will by default cause problems with compliance, cause a reduction in productivity, and may cause accidental or intentional security control sabotage.

Exam Essentials

Be aware of industry-standard frameworks. A security framework is a guide or plan for keeping your organizational assets safe. It provides guidance and a structure to the implementation of security for organizations. Security frameworks may be regulatory, nonregulatory, national, international, and/or industry-specific.

Understand benchmarks. A benchmark is a documented list of requirements that is used to determine whether a system, device, or software solution is allowed to operate within a secure management environment. Benchmarks may be platform- or vendor-specific or general-purpose.

Define defense in depth. Defense in depth or layered security is the use of multiple types of access controls in literal or theoretical concentric circles or layers. Defense in depth should include vendor diversity and control diversity.

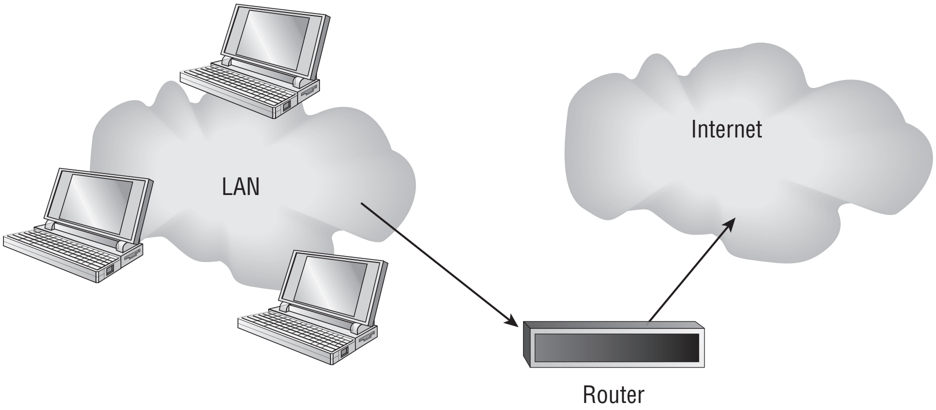
3.2 Given a scenario, implement secure network architecture concepts.

Reliable network security depends on a solid foundation. That foundation is the network architecture. Network architecture is the physical structure of your network, the divisions or segments, the means of isolation and traffic control, whether or not remote access is allowed, the means of secure remote connection, and the placement of sensors and filters. This section discusses many of the concepts of network architecture.

Zones/topologies

A *network zone* is an area of a network designed for a specific purpose, such as internal use or external use. Network zones are logical and/or physical divisions or segments of a LAN that allow for supplementary layers of security and control (see Figure 3.2). Each security zone is an area of a network that has a single defined level of security. That security may focus on encoding authorized access, preventing access, protecting confidentiality and integrity, or limiting traffic flow. Different security zones usually host different types of resources with different levels of sensitivity. Zones are often designated and isolated through the use of unique IP subnets and firewalls. Another term for network zone is network topology. There are many types of network zones; several are covered in the next sections.

FIGURE 3.2 A typical LAN connection to the Internet



DMZ

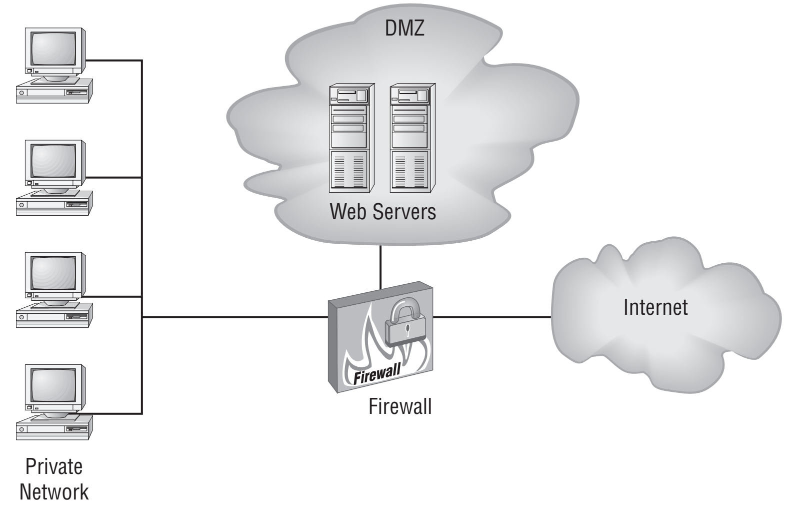
A *demilitarized zone (DMZ)* is a special-purpose subnet. A network consists of networking components (such as cables and switches) and hosts (such as clients and servers). Often, large networks are logically and physically subdivided into smaller interconnected networks. These smaller networks are known as *subnets*. Subnets are usually fairly generic, but some have special uses and/or configurations.

A DMZ is an area of a network that is designed specifically for low-trust users to access specific systems, such as the public accessing a web server. If the DMZ (as a whole or as individual systems within the DMZ) is compromised, the private LAN isn't necessarily affected or compromised. Access to a DMZ is usually controlled or restricted by a firewall and router system.

The DMZ can act as a buffer network between the public untrusted Internet and the private trusted LAN. This implementation is known as a screened subnet. It is deployed by placing the DMZ subnet between two firewalls, where one firewall leads to the Internet and the other to the private LAN.

A DMZ can also be deployed through the use of a multihomed firewall (see Figure 3.3). Such a firewall has three interfaces: one to the Internet, one to the private LAN, and one to the DMZ.

FIGURE 3.3 A multihomed firewall DMZ



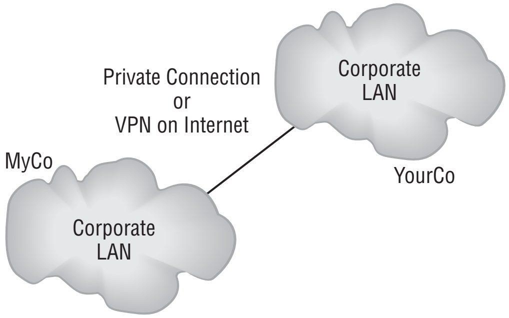
A DMZ gives an organization the ability to offer information services, such as web browsing, FTP, and email, to both the public and internal clients without compromising the security of the private LAN.

A typical scenario where a DMZ would be deployed is when an organization wants to offer resources, such as web server, email server, or file server, to the general public.

Extranet

An *extranet* (see Figure 3.4) is a privately controlled network segment or subnet that functions as a DMZ for business-to-business transactions. It allows an organization to offer specialized services to business partners, suppliers, distributors, or customers. Extranets are based on TCP/IP and often use the common Internet information services, such as web browsing, FTP, and email. Extranets aren't accessible to the general public. They often require outside entities to connect using a VPN. This restricts unauthorized access and ensures that all communications with the extranet are secured. Another important security concern with extranets is that companies that are partners today may be competitors tomorrow. Thus, you should never place data into an extranet that you're unwilling to let a future competitor have access to.

FIGURE 3.4 A typical extranet between two organizations



A common scenario for use of an extranet is when an organization needs to grant resource access to a business partner or external supplier. This allows the external entity to access the offered resources without exposing those resources to the open Internet and does not allow the external entities access into the private LAN.

Intranet

An *intranet* is a private network or private LAN. This term was coined in the 1990s when there was a distinction between traditional LANs and those adopting Internet technologies, such as the TCP/IP protocol, web services, and email. Now that most networks use these technologies, the term intranet is no longer distinct from LAN.

All organizations that have a network have an intranet. Thus, any scenario involving a private LAN is also an intranet.

Wireless

A wireless network is a network that uses radio waves as the communication media instead of copper or fiber-optic cables. A wireless network zone can be isolated using encryption (such as WPA-2) and unique authentication (so that only users and devices authorized for a specific network zone are able to log into that wireless zone).

Scenarios where wireless is a viable option include workspaces where portable devices are needed or when running network cables is cost prohibitive.

Guest

A guest zone or a guest network is an area of a private network designated for use by temporary authorized visitors. It allows nonemployee entities to partially interact with your private network, or at least with a subset of strictly controlled resources, without exposing your internal network to unauthorized user threats. A guest network can be a wireless or wired network. A guest network can also be implemented using VLAN enforcement.

Any organization that has a regularly recurring need to grant visitors and guests some level of network access—even if just to grant them Internet connectivity–should consider implementing a guest network.

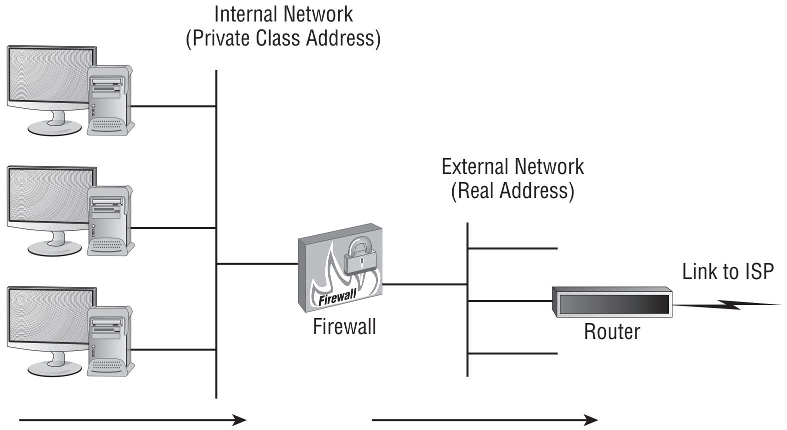
Honeynets

A *honeynet* consists of two or more networked honeypots used in tandem to monitor or re-create larger, more diverse network arrangements. Often, these honeynets facilitate IDS deployment for the purposes of detecting and catching both internal and external attackers. See the section "Honeynet" in Chapter 2, "Technologies and Tools," for more information.

NAT

In order for systems to communicate across the Internet, they must have an Internet-capable TCP/IP address. Unfortunately, leasing a sufficient number of public IP addresses to assign one to every system on a network is expensive. Plus, assigning public IP addresses to every system on the network means those systems can be accessed (or at least addressed) directly by external benign and malicious entities. One way around this issue is to use *network address translation (NAT)* (see Figure 3.5).

FIGURE 3.5 A typical Internet connection to a local network



NAT converts the private IP addresses (see the discussion of RFC 1918) of internal systems found in the header of network packets into public IP addresses. It performs this operation on a one-to-one basis; thus, a single leased public IP address can allow a single internal system to access the Internet. Because Internet communications aren't usually permanent or dedicated connections, a single public IP address could effectively support three or four internal systems if they never needed Internet access simultaneously. So, when NAT is used, a larger network needs to lease only a relatively small number of public IP addresses.

NAT provides the following benefits:

- It hides the IP addressing scheme and structure from external entities.
- It serves as a basic firewall by only allowing incoming traffic that is in response to an internal system's request.

- It reduces expense by requiring fewer leased public IP addresses.
- It allows the use of *private IP addresses* (RFC 1918).

RFC 1918

RFC 1918 defines the ranges of private IP addresses that aren't routable across the Internet. These ranges of addresses were specifically reserved for use by private networks. Anyone can use them at no expense; however, a NAT gateway must be deployed in order for systems using RFC 1918 addresses to communicate with the Internet. The ranges of IP addresses reserved for this purpose by RFC 1918 are as follows:

- 10.0.0.0–10.255.255.255 (10.0.0.0 /8 subnet): 1 Class A range
- 172.16.0.0–172.31.255.255 (172.16.0.0 /12 subnet): 16 Class B ranges
- 192.168.0.0–192.168.255.255 (192.168.0.0 /16 subnet): 256 Class C ranges

Closely related to NAT is *port address translation (PAT)*, which allows a single public IP address to host up to 65,536 simultaneous communications from internal clients (a theoretical maximum; in practice, you should limit the number to 100 or fewer in most cases). Instead of mapping IP addresses on a one-to-one basis, PAT uses the Transport layer port numbers to host multiple simultaneous communications across each public IP address.

The use of the term *NAT* in the IT industry has come to include the concept of PAT. Thus, when you hear or read about NAT, you can assume that the material is referring to PAT. This is true for most OSs and services; it's also true of the Security+ exam.

Another issue to be familiar with is that of NAT traversal (NAT-T). Traditional NAT doesn't support IPSec VPNs, because of the requirements of the IPSec protocol and the changes NAT makes to packet headers. However, NAT-T was designed specifically to support IPSec and other tunneling VPN protocols, such as Layer 2 Tunneling Protocol (L2TP), so organizations can benefit from both NAT and VPNs across the same border device/interface.

As the conversion from IPv4 to IPv6 takes place, there will be a need for NATing between these two IP structures. V4-to-v6 gateways or NAT servers will become more prevalent as the migration gains momentum, in order to maintain connectivity between legacy IPv4 networks and updated IPv6 networks. Once a majority of systems are using IPv6, the number of v4-to-v6 NATing systems will decline.

Scenarios where NAT implementation is essential include when using private IP addresses from RFC 1918 or when wanting to prevent external initiation of communications to internal devices.

Ad hoc

Ad hoc is a form of wireless network also known as the peer-to-peer network. It is a form of wireless network in which individual hosts connect directly to each other rather than going through a middleman. For more on this topic, see the Chapter 2 section "WiFi direct/ad hoc."

Segregation/segmentation/isolation

Network segmentation involves controlling traffic among networked devices. Complete or physical network segmentation occurs when a network is isolated from all outside communications, so transactions can only occur between devices within the segmented network. Logical network segmentation can be imposed with switches using VLANs, or through other traffic-control means, including MAC addresses, IP addresses, physical ports, TCP or UDP ports, protocols, or application filtering, routing, and access control management. Network segmentation can be used to isolate static environments in order to prevent changes and/or exploits from reaching them.

Security layers exist where devices with different levels of classification or sensitivity are grouped together and isolated from other groups with different security levels. This isolation can be absolute or one-directional. For example, a lower level may not be able to initiate communication with a higher level, but a higher level may initiate with a lower level. Isolation can also be logical or physical. *Logical isolation* requires the use of classification labels on data and packets, which must be respected and enforced by network management, OSs, and applications. *Physical isolation* requires implementing network segmentation or air gaps between networks of different security levels.

Bridging between networks can be a desired feature of network design. Network bridging is self-configuring, is inexpensive, maintains collision-domain isolation, is transparent to Layer 3+ protocols, and avoids the 5-4-3 rule's Layer 1 limitations (see `https://en.wikipedia.org/wiki/5-4-3_rule`). However, network bridging isn't always desirable. It doesn't limit or divide broadcast domains, doesn't scale well, can cause latency, and can result in loops. In order to eliminate these problems, you can implement network separation or segmentation. There are two means to accomplish this. First, if communication is necessary between network segments, you can implement IP subnets and use routers. Second, you can create physically separate networks that don't need to communicate. This can also be accomplished using firewalls instead of routers to implement secured filtering and traffic management.

All networks are involved in scenarios where segregation, segmentation, and isolation are needed. Without establishing a distinction between internal private networks and external public networks, maintaining privacy, security, and control is very challenging for the protection of sensitive data and systems. Network segmentation should be used to divide communication areas based on sensitivity of activities, value of data, risk of data loss or disclosure, level of classification, physical location, or any other distinction deemed important to an organization.

Physical

Physical segmentation occurs when no links are established between networks. This is also known as an air gap. If there are no cables and no wireless connections between two networks, then a physical network segregation/segmentation/isolation has been achieved. This is the most reliable means of prohibiting unwanted transfer of data. However, this configuration is also the most inconvenient for the rare events where communications are desired or necessary.

Logical (VLAN)

A *virtual local area network (VLAN)* is a hardware-imposed network segmentation created by switches. By default, all ports on a switch are part of VLAN 1. But as the switch administrator changes the VLAN assignment on a port-by-port basis, various ports can be grouped together and kept distinct from other VLAN port designations.

VLANs are used for traffic management. Communications between ports within the same VLAN occur without hindrance, but communications between VLANs require a routing function, which can be provided either by an external router or by the switch's internal software (one reason for the term *multilayer switch*). VLANs are treated like subnets but aren't subnets. VLANs are created by switches. Subnets are created by IP address and subnet mask assignments.

VLAN management is the use of VLANs to control traffic for security or performance reasons. VLANs can be used to isolate traffic between network segments. This can be accomplished by not defining a route between different VLANs or by specifying a deny filter between certain VLANs (or certain members of a VLAN). Any network segment that doesn't need to communicate with another in order to accomplish a work task/function shouldn't be able to do so. Use VLANs to allow what is necessary and to block/deny anything that isn't necessary. Remember, "deny by default; allow by exception" isn't a guideline just for firewall rules, but for security in general.

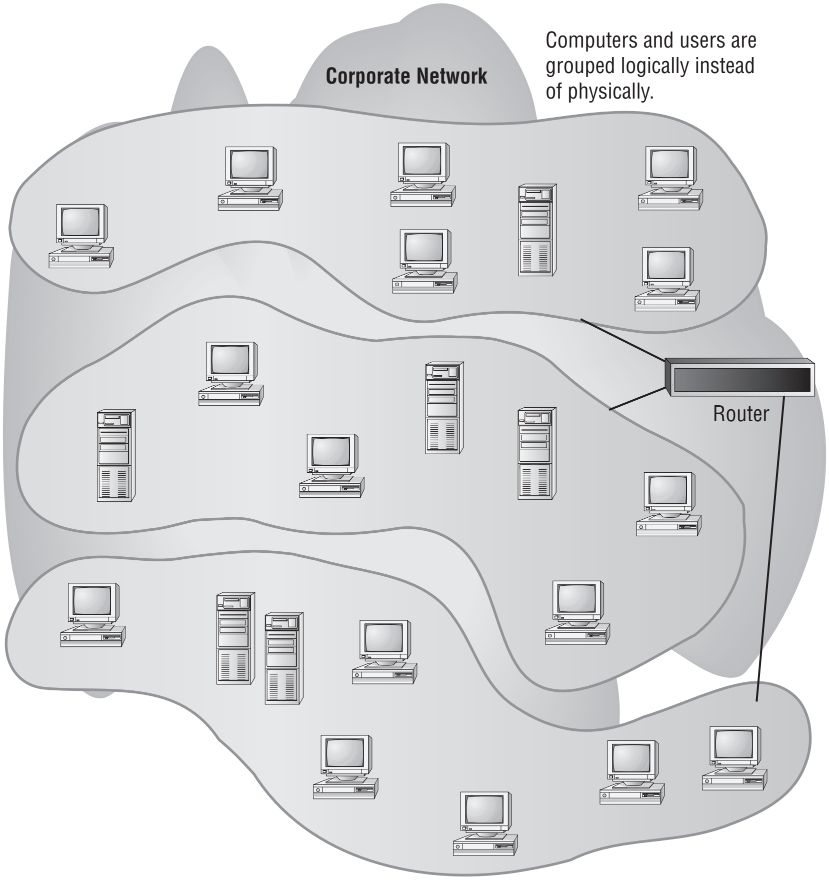
A VLAN consists of network divisions that are logically created out of a physical network. They're often created using switches (see Figure 3.6). Basically, the ports on a switch are numbered; each port is assigned the designation VLAN1 by default. Through the switch's management interfaces, the device administrator can assign ports other designations, such as VLAN2 or VLAN3, in order to create additional virtual networks.

VLANs function in much the same way as traditional subnets. In order for communications to travel from one VLAN to another, the switch performs routing functions to control and filter traffic between its VLANs.

VLANs are used to segment a network logically without altering its physical topology. They're easy to implement, have little administrative overhead, and are a hardware-based solution (specifically a Layer 3 switch). As networks are being crafted in virtual environments or in the cloud, software switches are often used. In these situations, VLANs are not hardware-based on or implemented by the software of a switch whether a physical device or a virtual system.

VLANs let you control and restrict broadcast traffic and reduce a network's vulnerability to sniffers, because a switch treats each VLAN as a separate network division. In order to communicate between segments, the switch must provide a routing function. It's the routing function that blocks broadcasts between subnets and VLANs, because a router (or any device performing Layer 3 routing functions, such as a Layer 3 switch) doesn't forward Layer 2 Ethernet broadcasts. This feature of a switch blocks Ethernet broadcasts between VLANs and so helps protect against broadcast storms. A *broadcast storm* is a flood of unwanted Ethernet broadcast network traffic.

FIGURE 3.6 A typical segmented VLAN


Corporate Network
Computers and users are grouped logically instead of physically.
Router

Virtualization

Virtualization technology is used to host one or more OSs in the memory of a single host computer. This mechanism allows virtually any OS to operate on any hardware. It also lets multiple OSs work simultaneously on the same hardware. Common examples include VMware, Microsoft's Virtual PC or Hyper-V, VirtualBox, and Apple's Parallels.

Virtualization offers several benefits, such as the ability to launch individual instances of servers or services as needed, real-time scalability, and the ability to run the exact OS version required for an application. Virtualized servers and services are indistinguishable from traditional servers and services from a user's perspective. Additionally, recovery from damaged, crashed, or corrupted virtual systems is often quick: you simply replace the virtual system's main hard drive file with a clean backup version, and then relaunch the affected virtual system.

With regard to security, virtualization offers several benefits. It's often easier and faster to make backups of entire virtual systems rather than the equivalent native hardware installed system. Plus, when there is an error or problem, the virtual system can be replaced by a backup in minutes. Malicious code compromises of virtual systems rarely affect the host OS. This allows for safer testing and experimentation.

Custom virtual network segmentation can be used in relation to virtual machines in order to make guest OSs members of the same network division as that of the host, or guest OSs can be placed into alternate network divisions. A virtual machine can be made a member of a different network segment from that of the host or placed into a network that only exists virtually and does not relate to the physical network media. See the later section "SDN" for more about this technique, known as software-defined networking.

Air gaps

An air gap is another term for physical network segregation, as discussed in the earlier section "Physical."

Tunneling/VPN

A *virtual private network (VPN)* is a communication *tunnel* between two entities across an intermediary network. In most cases, the intermediary network is an untrusted network, such as the Internet, and therefore the communication tunnel is usually encrypted. Numerous scenarios lend themselves to the deployment of VPNs; for example, VPNs can be used to connect two networks across the Internet (see Figure 3.7) or to allow distant clients to connect into an office local area network (LAN) across the Internet (see Figure 3.8). Once a VPN link is established, the network connectivity for the VPN client is exactly the same as a LAN connected by a cable connection. The only difference between a direct LAN cable connection and a VPN link is speed.

FIGURE 3.7 Two LANs being connected using a VPN across the Internet

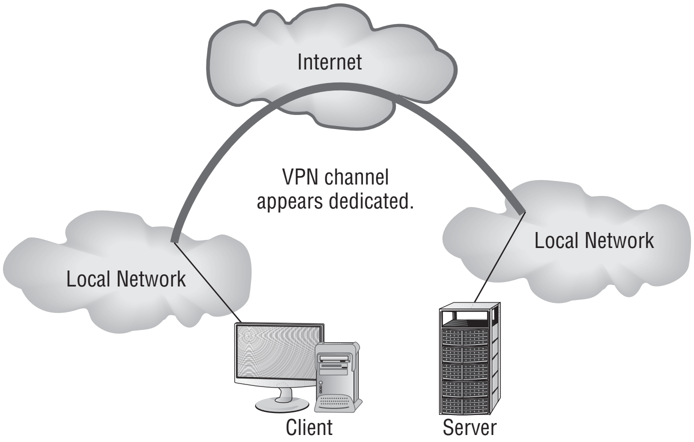
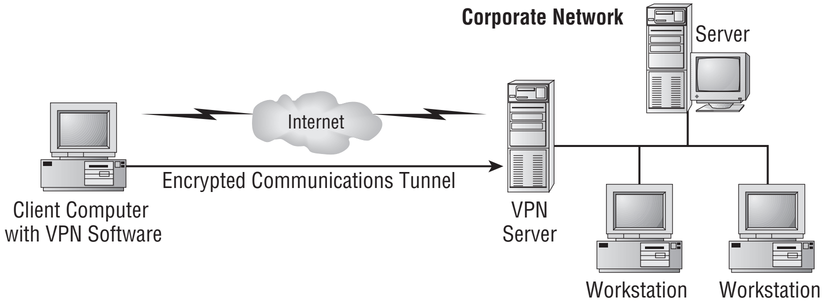


FIGURE 3.8 A client connecting to a network via a VPN across the Internet


Corporate Network
Server
Internet
Encrypted Communications Tunnel
Client Computer with VPN Software
VPN Server
Workstation
Workstation

VPNs offer an excellent solution for remote users to access resources on a corporate LAN. They have the following advantages:

- They eliminate the need for expensive dial-up modem banks, including landline and ISDN.
- They do away with long-distance toll charges.
- They allow a user anywhere in the world with an Internet connection to establish a VPN link with the office network.
- They provide security for both authentication and data transmission.

Sometimes VPN protocols are called *tunneling protocols*. This naming convention is designed to focus attention on the tunneling capabilities of VPNs.

VPNs work through a process called *encapsulation*. As data is transmitted from one system to another across a VPN link, the normal LAN TCP/IP traffic is encapsulated (encased, or enclosed) in the VPN protocol. The VPN protocol acts like a security envelope that provides special delivery capabilities (for example, across the Internet) as well as security mechanisms (such as data encryption).

When firewalls, intrusion detection systems, antivirus scanners, or other packet-filtering and -monitoring security mechanisms are used, you must realize that the data payload of VPN traffic won't be viewable, accessible, scannable, or filterable, because it's encrypted. Thus, in order for these security mechanisms to function against VPN-transported data, they must be placed outside of the VPN tunnel to act on the data after it has been decrypted and returned back to normal LAN traffic.

VPNs provide the following critical functions:

- *Access control* restricts users from accessing resources on a network.
- *Authentication* proves the identity of communication partners.
- *Confidentiality* prevents unauthorized disclosure of secured data.
- *Data integrity* prevents unwanted changes of data while in transit.

VPN links are established using VPN protocols. There are several VPN protocols, but these are the four you should recognize:

- *Point-to-Point Tunneling Protocol (PPTP)*
- *Layer 2 Tunneling Protocol (L2TP)*

- *OpenVPN (SSL VPN, TLS VPN)*
- *Internet Protocol Security* (IPsec) (see the Chapter 2 section "IPSec")

PPTP was originally developed by Microsoft. L2TP was developed by combining features of Microsoft's proprietary implementation of PPTP and Cisco's Layer 2 Forwarding (L2F) VPN protocols. Since its development, L2TP has become an Internet standard (RFC 2661) and is quickly becoming widely supported.

Both L2TP and PPTP are based on Point-to-Point Protocol (PPP) and thus work well over various types of remote-access connections, including dial-up. L2TP can support just about any networking protocol. PPTP is limited to IP traffic. L2TP uses UDP port 1701, and PPTP uses TCP port 1723.

PPTP can use any of the authentication methods supported by PPP, including the following:

- Challenge Handshake Authentication Protocol (CHAP)
- Extensible Authentication Protocol (EAP)
- Microsoft Challenge Handshake Authentication Protocol version 1 (MS-CHAP v.1)
- Microsoft Challenge Handshake Authentication Protocol version 2 (MS-CHAP v.2)
- Shiva Password Authentication Protocol (SPAP)
- Password Authentication Protocol (PAP)

Not all implementations of PPTP can provide data encryption. For example, when working with a PPTP VPN between Windows systems, the authentication protocol MS-CHAP v.2 enables data encryption.

L2TP can rely on PPP and thus on PPP's supported authentication protocols. This is typically referenced as IEEE 802.1x (see Chapter 4, "Identity and Access Management," and Chapter 6, "Cryptography and PKI," for their sections on IEEE 802.1x), which is a derivative of EAP from PPP. IEEE 802.1x enables L2TP to leverage or borrow authentication services from any available AAA server on the network, such as RADIUS or TACACS+. L2TP does not offer native encryption, but it supports the use of encryption protocols, such as Internet Protocol Security (IPSec). Although it isn't required, L2TP is most often deployed using IPsec.

L2TP can be used to tunnel any routable protocol but contains no native security features. When L2TP is used to encapsulate IPsec, it obtains authentication and data-encryption features because IPsec provides them. The main reason to use L2TP-encapsulated IPSec instead of naked IPSec is when needing to traverse a Layer 2 network that is either untrustworthy or its security is unknown. This can include a telco's business connection offerings, such as Frame Relay and Asynchronous Transfer Mode (ATM) or the public switched telephone network (PSTN). Otherwise, IPSec can be used without the extra overhead of L2TP.

OpenVPN is based on TLS (formally SSL) and provides an easy-to-configure but robustly secured VPN option. OpenVPN is an open-source implementation that can use either preshared secrets (such as passwords) or certificates for authentication. Many wireless access points support OpenVPN, which has a native VPN option for using a home or business WAP as a VPN gateway.

Site-to-site

A site-to-site VPN is a connection between two organizational networks. See the Chapter 2 section “Remote access vs. site-to-site” for more information.

Remote access

A *remote-access VPN* is a variant of the *site-to-site VPN*. The difference is that with a remote-access VPN one endpoint is the single entity of a remote user that connects into an organizational network. See the Chapter 2 section “Remote access vs. site-to-site” for more information.

Site-to-site and remote access VPNs are variants of tunnel mode VPN. Another type of VPN is the transport mode VPN, which provides end-to-end encryption and can be described as a host-to-host VPN. In this type of VPN, all traffic is fully encrypted between the endpoints, but those endpoints are only individual systems, not organizational networks.

Security device/technology placement

When designing the layout and structure of a network, it is important to consider the placement of security devices and related technology. The goal of planning the architecture and organization of the network infrastructure is to maximize security while minimizing downtime, compromises, or other interruptions to productivity.

Sensors

A sensor is a hardware or software tool used to monitor an activity or event in order to record information or at least take notice of an occurrence. A sensor may monitor heat, humidity, wind movement, doors and windows opening, the movement of data, the types of protocols in use on a network, when a user logs in, any activity against sensitive servers, and much more.

For sensors to be effective, they need to be located in proper proximity to be able to take notice of the event of concern. This might require the sensor to monitor all network traffic, monitor a specific doorway, or monitor a single computer system.

Collectors

A security collector is any system that gathers data into a log or record file. A collector’s function is similar to the functions of auditing, logging, and monitoring. A collector watches for a specific activity, event, or traffic, and then records the information into a record file. Targets could be, for example, logon events, door opening events, all launches of a specific executable, any access to sensitive files, or all activity on mission-critical servers.

A collector, like any auditing system, needs sufficient space on a storage device to record the data it collects. Such data should be treated as more sensitive than the original data, programs, or systems it was collected from. A collector should be placed where it has the ability to review and retrieve information on the system, systems, or network that it is intended to monitor. This might require a direct link or path to the monitored target, or it may be able to operate on a cloned or mirrored copy of communications, such as the SPAN, audit, mirror, or IDS port of a switch.

Correlation engines

A correlation engine is a type of analysis system that reviews the contents of log files or live events. It is programmed to recognize related events, sequential occurrences, and interdependent activity patterns in order to detect suspicious or violating events. Through a correlation engine's ability to aggregate and analyze system logs using fuzzy logic and predictive analytics, it may be able to detect a problem or potential problem long before a human administrator would have taken notice.

A correlation engine does not need to be in line on the network or installed directly onto monitored systems. It must have access to the recorded logs or the live activity stream in order to perform its analysis. This could allow it to operate on or near a data warehouse or centralized logging server (which is a system that maintains a real-time cloned copy of all live logs from servers and other critical systems) or off a switch SPAN port.

Filters

A filter is used to recognize or match an event, address, activity, content, or keyword and trigger a response. In most cases a filter is used to block or prevent unwanted activities or data exchanges. The most common example of a filtering tool is a firewall.

A filter should be located in line along any communication path where control of data communications is necessary. Keep in mind that filters cannot inspect encrypted traffic, so filtering of such traffic must be done just before encryption or just after decryption.

Proxies

For an introduction to proxies, see the Chapter 2 section "Proxy."

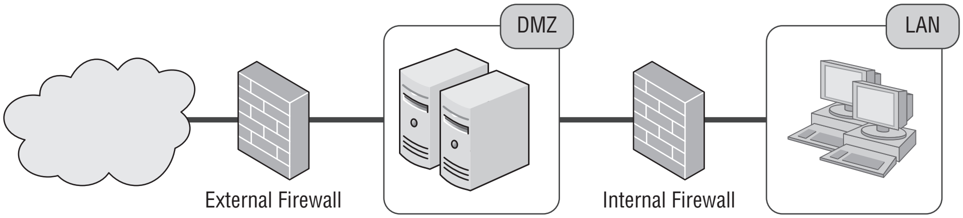
The placement or location of a proxy should be between source or origin devices and their destination systems. The location of a transparent proxy must be along the routed path between source and destination, whereas a nontransparent proxy can be located along an alternate or indirect routed path, since source systems will direct traffic to the proxy themselves.

Firewalls

For an introduction to firewalls, see the Chapter 2 section "Firewall."

The placement or location of a firewall should be at any transition between network segments where there is any difference in risk, sensitivity, security, value, function, or purpose. It is standard security practice to deploy a hardware security firewall between an internal network and the Internet as well as between the Internet, a DMZ or extranet, and an intranet (Figure 3.9).

FIGURE 3.9 A potential firewall deployment related to a DMZ



Software firewalls are also commonly deployed on every host system, both servers and clients, throughout the organizational network.

It may also be worth considering implementing firewalls between departments, satellite offices, VPNs, and even different buildings or floors.

VPN concentrators

For an introduction to VPN concentrators, see the Chapter 2 section "VPN concentrator."

A VPN concentrator should be located on the boundary or border of the organizational network at or near the primary Internet connection. The VPN concentrator may be located inside or outside of the primary network appliance firewall. If the security policy is that all traffic is filtered entering the private network, then the VPN concentrator must be located outside the firewall. If traffic from remote locations over VPNs is trusted, then the VPN concentrator can be located inside the firewall.

SSL accelerators

For an introduction to SSL/TLS accelerators, see the Chapter 2 section "SSL/TLS accelerators."

An SSL/TLS accelerator should be located at the boundary or border of the organizational network at or near the primary Internet connection and before the resource server being accessed by those protected connections. Usually the SSL/TLS accelerator is located in line with the communication pathway so that no abusive network access can reach the network segment between the accelerator and the resource host. The purpose of this device or service is to offload the computational burden of encryption in order for a resource host to devote its system resources to serving visitors.

Load balancers

For an introduction to Load balancers, see the Chapter 2 section "Load balancer."

A load balancer should be located in front of a group of servers, often known as a cluster, which all support the same resource. The purpose of the load balancer is to distribute the workload of connection requests among the members of the cluster group, so this determines its location. A load balancer is placed between requesting clients and the cluster or group of servers hosting a resource.

DDoS mitigator

A DDoS mitigator is a software solution, hardware device, or cloud service that attempts to filter and/or block traffic related to DoS attacks. See Chapter 1, "Threats, Attacks, and Vulnerabilities," sections "DoS" and "DDoS" for information about these attacks.

A DDoS mitigator will attempt to differentiate legitimate packets from malicious packets. Benign traffic will be sent toward its destination, whereas abusive traffic will be discarded. Low-end DDoS mitigators may be called flood guards. Flood guarding is often a feature of firewalls. However, such solutions only change the focus of the DDoS attack rather than eliminate it. A low-level DDoS mitigator or flood guard solution will prevent the malicious traffic from reaching the target server, but the filtering system may itself be overloaded. This can result in the DoS event still being able to cut off communications for the network, even when the targeted server is not itself harmed in the process.

Commercial-grade DDoS solution, especially those based on a cloud service, operate differently. Instead of simply filtering traffic on the spot, they reroute traffic to the cloud provider's core filtering network. The cloud-based DDoS mitigator will often use a load balancer in front of 10,000+ virtual machines in order to dilute and distribute the traffic for analysis and filtering. All garbage packets are discarded, and legitimate traffic is routed back to the target network.

A DDoS mitigator should be positioned in line along the pathway into the intranet, DMZ, and extranet from the Internet. This provides the DDoS mitigator with the ability to filter all traffic from external attack sources before it reaches servers or the network as a whole.

Aggregation switches

An aggregation switch is the main or master switch used as the interconnection point for numerous other switches. In the past, this device may have been known as the master distribution frame (MDF), central distribution frame, core distribution frame, or primary distribution frame. In large network deployments, an initial master primary switch is deployed near the demarcation point (which is the point where internal company wiring meets the external telco wiring), and then additional switches for various floors, departments, or network segments are connected off the master primary switch.

Taps and port mirror

A tap is a means to eavesdrop on network communications. In the past taps were physical connections to the copper wires themselves, often using a mechanical means to strip or pierce the insulation to make contact with the conductors. These types of taps were often called vampire taps. Today, taps can be installed in line without damaging the existing cable. To install an inline tap, first the original cable must be unplugged from the switch (or other network management device) and then plugged into the tap. Then the tap is plugged into the vacated original port. A tap should be installed wherever traffic monitoring on a specific cable is required and when a port mirroring function is either not available or undesired.

A port mirror is a common feature found on managed switches; it will duplicate traffic from one or more other ports out a specific port. A switch may have a hardwired Switched Port Analyzer (SPAN) port, which duplicates the traffic for all other ports, or any port can be set as the mirror, audit, IDS, or monitoring port for one or more other ports. Port mirroring takes place on the switch itself.

SDN

The concept of OS virtualization has given rise to other virtualization topics, such as virtualized networks. A virtualized network or network virtualization is the combination of hardware and software networking components into a single integrated entity. The resulting system allows for software control over all network functions: management, traffic shaping, address assignment, and so on. A single management console or interface can be used to oversee every aspect of the network, a task that required physical presence at each hardware component in the past. Virtualized networks have become a popular means of infrastructure deployment and management by corporations worldwide. They allow organizations to implement or adapt other interesting network solutions, including software-defined networks, virtual SANs, guest operating systems, and port isolation.

Software-defined networking (SDN) is a unique approach to network operation, design, and management. The concept is based on the theory that the complexities of a traditional network with on-device configuration (routers and switches) often force an organization to stick with a single device vendor, such as Cisco, and limit the flexibility of the network to adapt to changing physical and business conditions. SDN aims at separating the infrastructure layer (hardware and hardware-based settings) from the control layer (network services of data transmission management). Furthermore, this also negates the need for the traditional networking concepts of IP addressing, subnets, routing, and the like to be programmed into or deciphered by hosted applications.

SDN offers a new network design that is directly programmable from a central location, is flexible, is vendor neutral, and is based on open standards. Using SDN frees an organization from having to purchase devices from a single vendor. It instead allows organizations to mix and match hardware as needed, such as to select the most cost-effective or highest throughput–rated devices, regardless of vendor. The configuration and management of hardware are then controlled through a centralized management interface. In addition, the settings applied to the hardware can be changed and adjusted dynamically as needed.

Another way of thinking about SDN is that it is effectively network virtualization. It allows data transmission paths, communication decision trees, and flow control to be virtualized in the SDN control layer rather than being handled on the hardware on a per-device basis.

Another interesting development arising out of the concept of virtualized networks is the virtual storage area network (SAN). A SAN is a network technology that combines multiple individual storage devices into a single consolidated network-accessible storage container. A virtual SAN or a software-defined shared storage system is a virtual re-creation of a SAN on top of a virtualized network or an SDN.

A *storage area network (SAN)* is a secondary network (distinct from the primary communications network) used to consolidate and manage various storage devices. SANs are often used to enhance networked storage devices such as hard drives, drive arrays, optical jukeboxes, and tape libraries so they can be made to appear to servers as if they were local storage.

SANs can offer greater storage isolation through the use of a dedicated network. This makes directly accessing stored data difficult and forces all access attempts to operate against a server's restricted applications and interfaces.

Fibre Channel

Fibre Channel is a form of network data-storage solution (storage area network [SAN] or network-attached storage [NAS]) that allows for high-speed file transfers at upward of 16 Gbps. It was designed to be operated over fiber-optic cables; support for copper cables was added later to offer less expensive options. Fibre Channel typically requires its own dedicated infrastructure (separate cables). However, Fibre Channel over Ethernet (FCoE) can be used to support it over the existing network infrastructure.

FCoE

FCoE is used to encapsulate Fibre Channel communications over Ethernet networks. It requires 10 Gbps Ethernet in order to support the Fibre Channel protocol. With this technology, Fibre Channel operates as a Network layer or OSI Layer 3 protocol, replacing IP as the payload of a standard Ethernet network.

FCIP

Fiber Channel over IP (FCIP) further expands the use of Fibre Channel signaling to no longer require any specific speed of network. It is the SAN equivalent of VoIP.

iSCSI

Internet Small Computer System Interface (iSCSI) is a networking storage standard based on IP. This technology can be used to enable location-independent file storage, transmission, and retrieval over LAN, WAN, or public Internet connections. iSCSI is often viewed as a low-cost alternative to Fibre Channel.

Exam Essentials

Comprehend network zones. A network zone is an area of a network designed for a specific purpose, such as internal use or external use. Network zones are logical and/or physical divisions or segments of a LAN that allow for supplementary layers of security and control.

Understand DMZs. A demilitarized zone (DMZ) is an area of a network that is designed specifically for public users to access. The DMZ is a buffer network between the public untrusted Internet and the private trusted LAN. Often a DMZ is deployed through the use of a multihomed firewall.

Understand extranets. An extranet is an intranet that functions as a DMZ for business-to-business transactions. Extranets let organizations offer specialized services to business partners, suppliers, distributors, or customers.

Understand intranets. An intranet is a private network or private LAN.

Know about guest networks. A guest zone or a guest network is an area of a private network designated for use by temporary authorized visitors.

Understand honeynets. A honeynet consists of two or more networked honeypots used in tandem to monitor or re-create larger, more diverse network arrangements.

Be aware of NAT. NAT converts the IP addresses of internal systems found in the headers of network packets into public IP addresses. It hides the IP addressing scheme and structure from external entities. NAT serves as a basic firewall by only allowing incoming traffic that is in response to an internal system's request. It reduces expense by requiring fewer leased public IP addresses, and it allows the use of private IP addresses (RFC 1918).

Understand PAT. Closely related to NAT is port address translation (PAT), which allows a single public IP address to host multiple simultaneous communications from internal clients. Instead of mapping IP addresses on a one-to-one basis, PAT uses the Transport layer port numbers to host multiple simultaneous communications across each public IP address.

Know RFC 1918. RFC 1918 defines the ranges of private IP addresses that aren't routable across the Internet: 10.0.0.0–10.255.255.255 (10.0.0.0 /8 subnet), 1 Class A range; 172.16.0.0–172.31.255.255 (172.16.0.0 /12 subnet), 16 Class B ranges; and 192.168.0.0–192.168.255.255 (192.168.0.0 /16 subnet), 256 Class C ranges.

Understand network segmentation. Network segmentation involves controlling traffic among networked devices. Logical network segmentation can be imposed with switches using VLANs, or through other traffic-control means, including MAC addresses, IP addresses, physical ports, TCP or UDP ports, protocols, or application filtering, routing, and access control management.

Comprehend VLANs. Switches are often used to create virtual LANs (VLANs)—logical creations of subnets out of a single physical network. VLANs are used to logically segment a network without altering its physical topology. They are easy to implement, have little administrative overhead, and are a hardware-based solution.

Understand virtualization. Virtualization technology is used to host one or more OSs within the memory of a single host computer. Related issues include snapshots, patch compatibility, host availability/elasticity, security control testing, and sandboxing.

Understand VPNs. A virtual private network (VPN) is a communication tunnel between two entities across an intermediary network. In most cases, the intermediary network is an untrusted network, such as the Internet, and therefore the communication tunnel is also encrypted.

Know VPN protocols. PPTP, L2TP, OpenVPN, and IPSec are VPN protocols.

Understand PPTP. Point-to-Point Tunneling Protocol (PPTP) is based on PPP, is limited to IP traffic, and uses TCP port 1723. PPTP supports PAP, SPAP, CHAP, EAP, and MS-CHAP v.1 and v.2.

Know L2TP. Layer 2 Tunneling Protocol (L2TP) is based on PPTP and L2F, supports any LAN protocol, uses UDP port 1701, and often uses IPSec for encryption.

Understand OpenVPN. OpenVPN is based on TLS (formerly SSL) and provides an easy-to-configure but robustly secured VPN option.

Realize the importance of security device placement. When designing the layout and structure of a network, it is important to consider the placement of security devices and related technology. The goal of planning the architecture and organization of the network infrastructure is to maximize security while minimizing downtime, compromises, or other interruptions to productivity.

Understand software-defined networking. Software-defined networking (SDN) is a unique approach to network operation, design, and management. SDN aims at separating the infrastructure layer (hardware and hardware-based settings) from the control layer (network services of data transmission management).

3.3 Given a scenario, implement secure systems design.

Any effective security infrastructure is built following the guidelines of a security policy and consists of secure systems. A secure system must be planned and developed with security not just as a feature but as a central core concept. This section discusses some of the important design concepts that contribute to secure systems.

Hardware/firmware security

Security is an integration of both hardware/firmware components and software elements. This section looks at several hardware and firmware security technologies.

FDE/SED

Full-disk encryption (FDE) or *whole-disk encryption* is often used to provide protection for an OS, its installed applications, and all locally stored data. FDE encrypts all of the data on a storage device with a single master symmetric encryption key. Anything written to the encrypted storage device, including standard files, temporary files, cached data, memory swapped data, and even the remnants of deletion and the contents of slack space, is encrypted when FDE is implemented.

However, whole-disk encryption provides only reasonable protection when the system is fully powered off. If a system is accessed by a hacker while it's active, there are several ways around hard drive encryption. These include a FireWire direct memory access (DMA) attack, malware stealing the encryption key out of memory, slowing down memory-decay rates with liquid nitrogen, or even just user impersonation. The details of these attacks aren't important for this exam. However, you should know that whole-disk encryption is only a partial security control.

To maximize the defensive strength of whole-disk encryption, you should use a long, complex passphrase to unlock the system on bootup. This passphrase shouldn't be written down or used on any other system or for any other purpose. Whenever the system isn't actively in use, it should be powered down (hibernation is fine, but not sleep mode) and physically locked against unauthorized access or theft. Hard drive encryption should be viewed as a delaying tactic, rather than as a true prevention of access to data stored on the hard drive.

Hard drive encryption can be provided by a software solution, as discussed previously, or through a hardware solution. One option is self-encrypting drives (SED). Some hard drive manufacturers offer hard drive products that include onboard hardware-based encryption services. However, most of these solutions are proprietary and don't disclose their methods or algorithms, and some have been cracked with relatively easy hacks.

Using a trusted software encryption solution can be a cost-effective and secure choice. But realize that no form of hard drive encryption, hardware- or software-based, is guaranteed protection against all possible forms of attack.

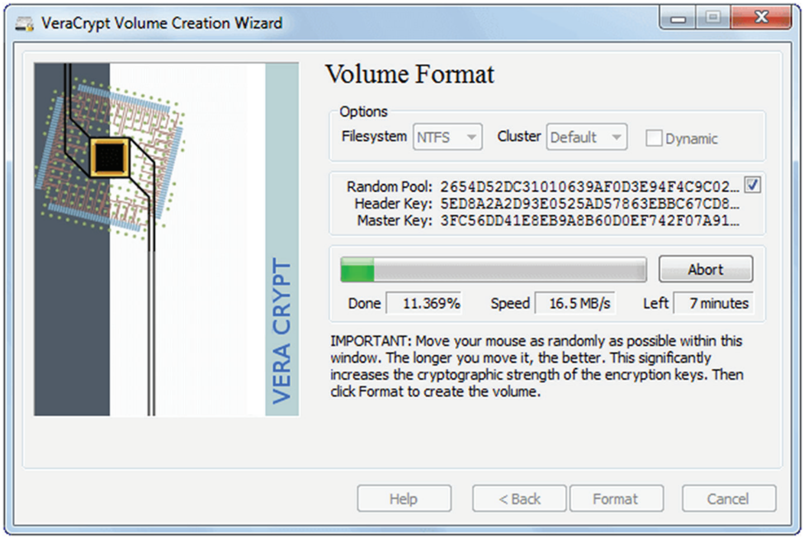
USB encryption is usually related to USB storage devices, which can include both USB-connected hard drives as well as USB thumb drives. Some USB device manufacturers include encryption features in their products. These often have an autorun tool that is used to gain access to encrypted content once the user has been authenticated. An example of an encrypted USB device is an IronKey.

If encryption features aren't provided by the manufacturer of a USB device, you can usually add them through a variety of commercial or open-source solutions. One of the best-known, respected, and trusted open-source solutions is VeraCrypt (Figure 3.10) (the revised and secure replacement for its abandoned predecessor, TrueCrypt). This tool can be used to encrypt files, folders, partitions, drive sections, or whole drives, whether internal, external, or USB.

TPM

The *trusted platform module (TPM)* is both a specification for a cryptoprocessor and the chip in a mainboard supporting this function. A TPM chip is used to store and process cryptographic keys for a hardware-supported/implemented hard drive encryption system. Generally, a hardware implementation rather than a software-only implementation of hard drive encryption is considered more secure.

FIGURE 3.10 VeraCrypt encryption dialog box



VeraCrypt Volume Creation Wizard

Volume Format

Options

Filesystem: NTFS Cluster: Default □ Dynamic

Random Pool: 2654D52DC31010639AF0D3E94F4C9C02... ☑
Header Key: 5ED8A2A2D93E0525AD57863EBBC67CD8...
Master Key: 3FC56DD41E8EB9A8B60D0EF742F07A91...

Done: 11.369% Speed: 16.5 MB/s Left: 7 minutes

IMPORTANT: Move your mouse as randomly as possible within this window. The longer you move it, the better. This significantly increases the cryptographic strength of the encryption keys. Then click Format to create the volume.

When TPM-based whole-disk encryption is in use, the user/operator must supply a password or physical USB token device to the computer to authenticate and allow the TPM chip to release the hard drive encryption keys into memory. Although this seems similar to a software implementation, the primary difference is that if the hard drive is removed from its original system, it can't be decrypted. Only with the original TPM chip can an encrypted hard drive be decrypted and accessed. With software-only hard drive encryption, the hard drive can be moved to a different computer without any access or use limitations.

HSM

A *hardware security module (HSM)* is a special-purpose cryptoprocessor used for a wide range of potential functions. The functions of an HSM can include accelerated cryptography operations, managing and storing encryption keys, offloading digital signature verification, and improving authentication. An HSM can be a chip on a motherboard, an external peripheral, a network-attached device, or an add-on or extension adapter or card (which is inserted into a device, such as a router, firewall, or rack-mounted server blade). Often an HSM includes tamper protection technology in order to prevent or discourage abuse and misuse even if physical access is obtained by the attacker. One example of an HSM is the TPM (see the previous section).

UEFI/BIOS

Basic input/output system (BIOS) is the basic low-end firmware or software embedded in the hardware's *electrically erasable programmable read-only memory (EEPROM)*. The BIOS identifies and initiates the basic system hardware components, such as the hard drive, optical drive, video card, and so on, so that the bootstrapping process of loading an OS can begin. This essential system function is a target of hackers and other intruders because it may provide an avenue of attack that isn't secured or monitored.

BIOS attacks, as well as complementary metal-oxide-semiconductor (CMOS) and device firmware attacks, are becoming common targets of physical hackers as well as of malicious code. If hackers or malware can alter the BIOS, CMOS, or firmware of a system, they may be able to bypass security features or initiate otherwise prohibited activities.

Protection against BIOS attacks requires physical access control for all sensitive or valuable hardware. Additionally, strong malware protection, such as current antivirus software, is important.

A replacement or improvement to BIOS is Unified Extensible Firmware Interface (UEFI). UEFI provides support for all of the same functions as BIOS with many improvements, such as support for larger hard drives (especially for booting), faster boot times, enhanced security features, and even the ability to use a mouse when making system changes (BIOS was limited to keyboard control only). UEFI also includes a CPU-independent architecture, a flexible pre-OS environment with networking support, secure boot (see the next section), and backward and forward compatibility. It also runs CPU-independent drivers (for system components, drive controllers, and hard drives).

Secure boot and attestation

Secure boot is a feature of UEFI that aims to protect the operating environment of the local system by preventing the loading or installing of device drivers or an operating system that is not signed by a preapproved digital certificate. Secure boot thus protects systems against a range of low-level or boot-level malware, such as certain rootkits and backdoors. Secure boot ensures that only drivers and operating systems that pass attestation (the verification and approval process accomplished through the validation of a digital signature) are allowed to be installed and loaded on the local system.

Although the security benefits of secure boot attestation are important and beneficial to all systems, there is one important drawback to consider: if a system has a locked UEFI secure boot mechanism, it may prevent the system's owner from replacing the operating system (such as switching from Windows to Linux) or block them from using third-party vendor hardware that has not been approved by the motherboard vendor (which means the third-party vendor did not pay a fee to have their product evaluated and their drivers signed by the motherboard vendor). If there is any possibility of using alternate OSs or changing hardware components of a system, be sure to use a motherboard from a vendor that will provide unlock codes/keys to the UEFI secure boot.

Supply chain

Supply chain security is the concept that most computers are not built by a single entity. In fact, most of the companies we know of as computer manufacturers, such as Dell, HP, Asus, Acer, and Apple, mostly perform the final assembly rather than manufacture all of the individual components. Often the CPU, memory, drive controllers, hard drives, SSDs, and video cards are created by other third-party vendors. Even these vendors are unlikely to have mined their own metals or processed the oil for plastics or etched the silicon of their chips. Thus, any finished system has a long and complex history, known as its supply chain, that enabled it or caused it to come into existence.

A secure supply chain is one in which all of the vendors or links in the chain are reliable, trustworthy, reputable organizations that disclose their practices and security requirements to their business partners (although not necessarily to the public). Each link in the chain is responsible and accountable to the next link in the chain. Each hand-off, from raw materials to refined products to electronics parts to computer components to finished product, is properly organized, documented, managed, and audited. The goal of a secure supply chain is to ensure that the finished product is of sufficient quality, meets performance and operational goals, and provides stated security mechanisms, and that at no point in the process was any element subjected to unauthorized or malicious manipulation or sabotage.

Hardware root of trust

A hardware root of trust is based or founded on a secure supply chain. The security of a system is ultimately dependent upon the reliability and security of the components that make up the computer as well as the process it went through to be crafted from original raw materials. If the hardware that is supporting an application has security flaws or a backdoor, or fails to provide proper HSM-based cryptography functions, then the software is unable to accommodate those failings. Only if the root of the system—the hardware itself—is reliable and trustworthy can the system as a whole be considered trustworthy. System security is a chain of many interconnected links; if any link is weak, then the whole chain is untrustworthy.

EMI/EMP

Electromagnetic interference (EMI) is the noise caused by electricity when used by a machine or when flowing along a conductor. Copper network cables and power cables can pick up environmental noise or EMI, which can corrupt the network communications or disrupt the electricity feeding equipment. An electromagnetic pulse (EMP) is an instantaneous high-level EMI, which can damage most electrical devices in the vicinity.

EMI shielding is important for network-communication cables as well as for power-distribution cables. EMI shielding can include upgrading from UTP (unshielded twisted pair) to STP (shielded twisted pair), running cables in shielding conduits, or using fiber-optic networking cables. EMI-focused shielding can also provide modest protection against EMPs, although that depends on the strength and distance of the EMP compared to the device or cable. Generally, these two types of cables (networking and electrical) should be run in separate conduits and be isolated and shielded from each other. The strong magnetic fields produced by power-distribution cables can interfere with network-communication cables.

Operating systems

Any secure system design requires the use of a secure operating system. Although no operating system is perfectly secure, the selection of the right operating system for a particular task or function can reduce the ongoing burden of security management.

Types

There many ways to categorize or group operating systems. This section includes several specific examples of OS types, labels, and groupings. In all cases, the selection of an OS should focus on features and capabilities without overlooking the native security benefits. Although security can often be added through software installation, native security features are often superior.

Network

A network operating system (NOS) is any OS that has native networking capabilities and was designed with networking as a means of communication and data transfer. Most OSs today are NOSs, but not all OSs are network capable. There are still many situations where a stand-alone or isolated OS is preferred for function, stability, and security.

Server

A server is a form of NOS. It is a resource host that offers data, information, or communication functions to other requesting systems. *Servers* are the computer systems on a network that support and maintain the network. They require greater physical and logical security protections than workstations because they represent a concentration of assets, value, and capabilities. End users should be restricted from physically accessing servers, and they should have no reason to log on directly to a server—they should interact with servers over a network through their workstations.

Workstation

A workstation is another form of NOS. A workstation is a resource consumer. A workstation is typically where an end user will log in and then from the workstation reach out across the network to servers to access resources and retrieve data. Workstations are also called *clients*, *terminals*, or *end-user computers*. Access to workstations should be restricted to authorized personnel. One method to accomplish this is to use strong authentication, such as two-factor authentication with a smartcard and a password or PIN.

Appliance

An appliance OS is yet another variation of NOS. An appliance NOS is a stripped-down or single-purpose OS that is typically found on network devices, such as firewalls, routers, switches, wireless access points, and VPN gateways. An appliance NOS is designed around a primary set of functions or tasks and usually does not support any other capabilities.

Kiosk

A kiosk OS is either a stand-alone OS or a variation of NOS. A kiosk OS is designed for end-user use and access. The end user might be an employee of an organization or anyone from the general public. A kiosk OS is locked down so that only preauthorized software products and functions are enabled. A kiosk OS will revert to the locked-down mode each time it is rebooted, and some will even revert if they experience a flaw, crash, error, or any attempt to perform an unauthorized command or launch an unapproved executable. The goal and purpose of a kiosk OS is to provide a robust information service to a user while preventing accidental or intentional misuse of the system. A kiosk OS is often deployed in a public location and thus must be configured to implement security effectively for that situation.

Mobile OS

A mobile OS is yet another form of NOS. A mobile OS is designed to operate on a portable device. Although a portable device can be defined as any device with a battery, a mobile OS is designed for portable devices for which traditional NOSs are too large or too resource demanding. Many portable devices have less CPU processing capacity, RAM memory availability, and storage capabilities than full computer notebooks or workstation systems. A mobile OS is designed to optimize performance of limited resources. It may be designed around a few specific mobile device features, such as phone calls, text messages, and taking photographs, or may be designed to support a wide range of user-installed software or applications ("apps"). Some mobile OSs are extremely limited in their functions, whereas others can provide capabilities nearly equivalent to those of a traditional workstation NOS.

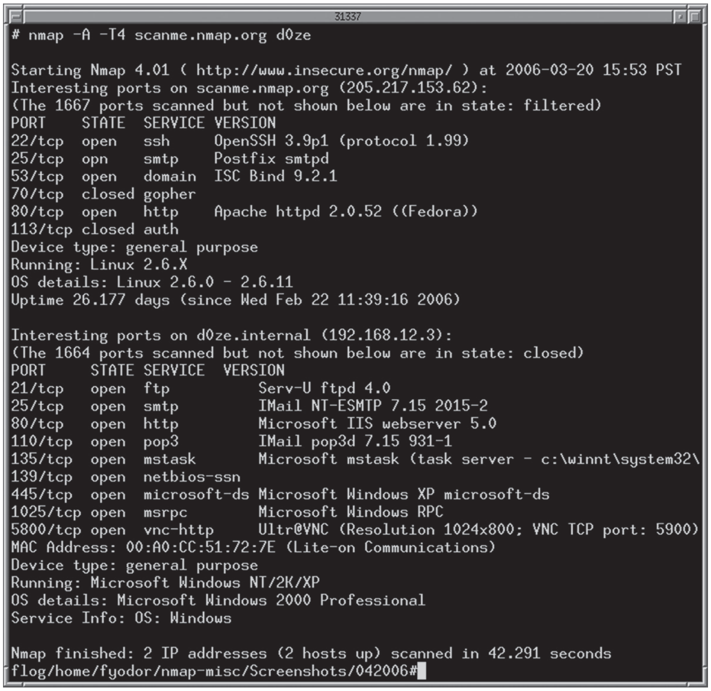
Patch management

See the Chapter 2 section "Patch management tools" for the security implications of patch management.

Disabling unnecessary ports and services

A key element in securing a system is to reduce its attack surface. The *attack surface* is the area that is exposed to untrusted networks or entities and that is vulnerable to attack. If a system is hosting numerous services and protocols, its attack surface is larger than that of a system running only essential services and protocols. The image in Figure 3.11 is from nmap.org, the source site for this tool, `https://nmap.org/images/nmap-401-demoscan-798x774.gif`.

FIGURE 3.11 Output from nmap showing open ports on scanned target systems



```
# nmap -A -T4 scanme.nmap.org d0ze

Starting Nmap 4.01 ( http://www.insecure.org/nmap/ ) at 2006-03-20 15:53 PST
Interesting ports on scanme.nmap.org (205.217.153.62):
(The 1667 ports scanned but not shown below are in state: filtered)
PORT    STATE  SERVICE VERSION
22/tcp  open   ssh     OpenSSH 3.9p1 (protocol 1.99)
25/tcp  opn    smtp    Postfix smtpd
53/tcp  open   domain  ISC Bind 9.2.1
70/tcp  closed gopher
80/tcp  open   http    Apache httpd 2.0.52 ((Fedora))
113/tcp closed auth
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.0 - 2.6.11
Uptime 26.177 days (since Wed Feb 22 11:39:16 2006)

Interesting ports on d0ze.internal (192.168.12.3):
(The 1664 ports scanned but not shown below are in state: closed)
PORT     STATE SERVICE  VERSION
21/tcp   open  ftp          Serv-U ftpd 4.0
25/tcp   open  smtp         IMail NT-ESMTP 7.15 2015-2
80/tcp   open  http         Microsoft IIS webserver 5.0
110/tcp  open  pop3         IMail pop3d 7.15 931-1
135/tcp  open  mstask       Microsoft mstask (task server - c:\winnt\system32\
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds Microsoft Windows XP microsoft-ds
1025/tcp open  msrpc        Microsoft Windows RPC
5800/tcp open  vnc-http     Ultr@VNC (Resolution 1024x800; VNC TCP port: 5900)
MAC Address: 00:A0:CC:51:72:7E (Lite-on Communications)
Device type: general purpose
Running: Microsoft Windows NT/2K/XP
OS details: Microsoft Windows 2000 Professional
Service Info: OS: Windows

Nmap finished: 2 IP addresses (2 hosts up) scanned in 42.291 seconds
flog/home/fyodor/nmap-misc/Screenshots/042006#
```

It's tempting to install every service, component, application, and protocol available to you on every computer system you deploy. However, this temptation is in direct violation of the security best practice that each system should host only those services and protocols that are absolutely essential to its mission-critical operations.

Any unused application service ports should be specifically blocked or disabled. *Port or interface disabling* is a physical option that renders a connection port electrically useless. *Port blocking* is a service provided by a software or hardware firewall that blocks or drops packets directed toward disallowed ports.

Layer 4, the Transport layer, uses ports to indicate the protocol that is to receive the payload/content of the TCP or UDP packet. Ports also assist in supporting multiple simultaneous connections or sessions over a single IP address. There are 65,535 potential ports. See `www.iana.org/assignments/port-numbers` for a current complete list of ports and protocol associations.

There are a number of common protocol default ports you may want to know for exam purposes. Table 3.1 is a brief list of ports to consider memorizing. All listed ports are default ports, and custom configurations can use alternate port selections.

TABLE 3.1 Common protocols and default ports

Protocol/service	Port(s)	Notes
FTP	TCP ports 20 (data) and 21 (control)	
SSH	TCP port 22	All protocols encrypted by SSH also use TCP port 22, such as SFTP, SHTTP, SCP, SExec, and slogin.
SMTP	TCP port 25	
DNS	TCP and UDP port 53	TCP port 53 is used for zone transfers, whereas UDP port 53 is used for queries.
HTTP	TCP port 80 or TCP port 8080	
Post Office Protocol v3 (POP3)	TCP port 110	
NetBIOS Session service	TCP port 139	

TABLE 3.1 Common protocols and default ports *(continued)*

Protocol/service	Port(s)	Notes
Internet Message Access Protocol v4 (IMAP4)	TCP port 143	
HTTPS	TCP port 443	(TCP port 80 in some configurations of TLS)
Remote Desktop Protocol (RDP)	TCP port 3389	

The real issue is that software isn't trusted. Software (services, applications, components, and protocols) is written by people, and therefore, in all likelihood, it isn't perfect. But even if software lacked bugs, errors, oversights, mistakes, and so on, it would still represent a security risk. Software that is working as expected can often be exploited by a malicious entity. Therefore, every instance of software deployed onto a computer system represents a collection of additional vulnerability points that may be exposed to external, untrusted, and possibly malicious entities.

From this perspective, you should understand that all nonessential software elements should be removed from a system before it's deployed on a network, especially if that network has Internet connectivity. But how do you know what is essential and what isn't? Here is a basic methodology:

1. Plan the purpose of the system.
2. Identify the services, applications, and protocols needed to support that purpose. Make sure these are installed on the system.
3. Identify the services, applications, and protocols that are already present on the system. Remove all that aren't needed.

Often, you won't know if a specific service that appears on a system by default is needed. Thus, a trial-and-error test is required. If software elements aren't clearly essential, disable them one by one and test the capabilities of the system. If the system performs as you expect, the software probably isn't needed. If the system doesn't perform as expected, then the software needs to be re-enabled. This process is known as *application and system hardening*.

You may discover that some services and protocols offer features and capabilities that aren't necessary to the essential functions of your system. If so, find a way to disable or restrict those characteristics. This may include restricting ports or reconfiguring services through a management console.

The essential services on a system are usually easy to identify—they generally have recognizable names that correspond to the function of the server. However, you must determine which services are essential on your specific system. Services that are essential on a web server may not be essential on a file server or an email server. Some examples of possible essential services are as follows:

- File sharing
- Email
- Web
- File Transfer Protocol (FTP)
- Telnet
- SSH
- Remote access
- Network News Transfer Protocol (NNTP)
- Domain Name Service (DNS)
- Dynamic Host Configuration Protocol (DHCP)

Nonessential services are more difficult to identify. Just because a service doesn't have the same name as an essential function of your server doesn't mean it isn't used by the underlying OS or as a support service. It's extremely important to test and verify whether any service is being depended on by an essential service. However, several services are common candidates for nonessential services that you may want to locate and disable first (assuming you follow the testing method described earlier). These may include the following:

- NetBIOS
- Unix RPC
- Network File System (NFS)
- X services
- R services
- Trivial File Transfer Protocol (TFTP)
- NetMeeting
- Instant messaging
- Remote-control software
- Simple Network Management Protocol (SNMP)

OSI Relevance

The OSI model (ISO/IEC 7498-1) was developed over three decades ago as a conceptual reference model for describing protocols, as well as potentially to guide their design. But over the years, protocols were not designed to adhere to the OSI model, so its relevance has waned. Many IT professionals still use OSI as a standard reference. Most discussions of protocols and hardware continue to use its seven-layer model as a point of reference in order to maintain clear communications and to relate more easily to existing documentation and prior technology. However, using the OSI model as a protocol reference is a bit like using Imperial units (United States customary/standard units) of measurement while the rest of the world uses metric.

The most widely used protocol in the world today is TCP/IP (a protocol suite rather than an individual protocol). TCP/IP operates on four layers rather than the seven layers of the OSI model. This four-layer model is also referred to as the DARPA model, or the DoD model. Here is a quick cross-reference between the two models:

TCP/IP model	OSI model
Process layer or Application layer(4)	Application layer (7) Presentation layer (6) Session layer (5)
Host-to-host layer or Transport layer(3)	Transport layer (4)
Internetworking layer or Internet layer(2)	Network layer (3)
Link layer or Network Access layer (1)	Data Link layer (2) Physical layer (1)

On the Security+ exam, when a layer name or number is mentioned, assume that it means the OSI model. Only if the exam uses one of the TCP/IP-specific layer names or calls out the four-layer model directly is it referencing the TCP/IP model.

Least functionality

One rule of thumb to adopt when designing and implementing security is that of least functionality. If you always select and install the solution with the least functionality or without any unnecessary additional capabilities and features, you will likely have a more secure result than opting for any solution with more options than necessary. This is another perspective on minimizing your attack surface. Rather than removing and blocking components that are unneeded or unwanted, select hardware and software systems that have minimal additional capabilities beyond what is strictly needed for the business function or task.

Secure configurations

All company systems should be operating within expected parameters and compliant with a defined baseline of secure configuration. Any system that is determined to be out of baseline should be removed from the production network in order to investigate the cause. If the deviation was due to a malicious event, then investigate and respond. If the deviation was due to normal work-related actions and activities, it may be necessary to update the baseline and/or implement more restrictive system modification policies, such as whitelisting or using static systems. A static system is an environment in which users cannot make changes or the few changes users can implement are only temporary and are discarded once the user logs out.

For more discussion of secure configurations and establishing a baseline, see the section "Secure baseline" later in this chapter.

Trusted operating system

Trusted OS is an access-control feature that requires a specific OS to be present in order to gain access to a resource. By limiting access to only those systems that are known to implement specific security features, resource owners can be assured that violations of a resource's security will be less likely.

Another formal definition of *trusted OS* is any OS that has security features in compliance with government and/or military security standards that enable the enforcement of multilevel security policies (that is, enforcing mandatory access control using classification labels on subjects and objects). Examples of trusted OSs include Trusted Solaris, Apple macOS X, HP-UX, and AIX. Many other OSs can be altered to become trusted, such as Windows, Windows Server, and SELinux.

Application whitelisting/blacklisting

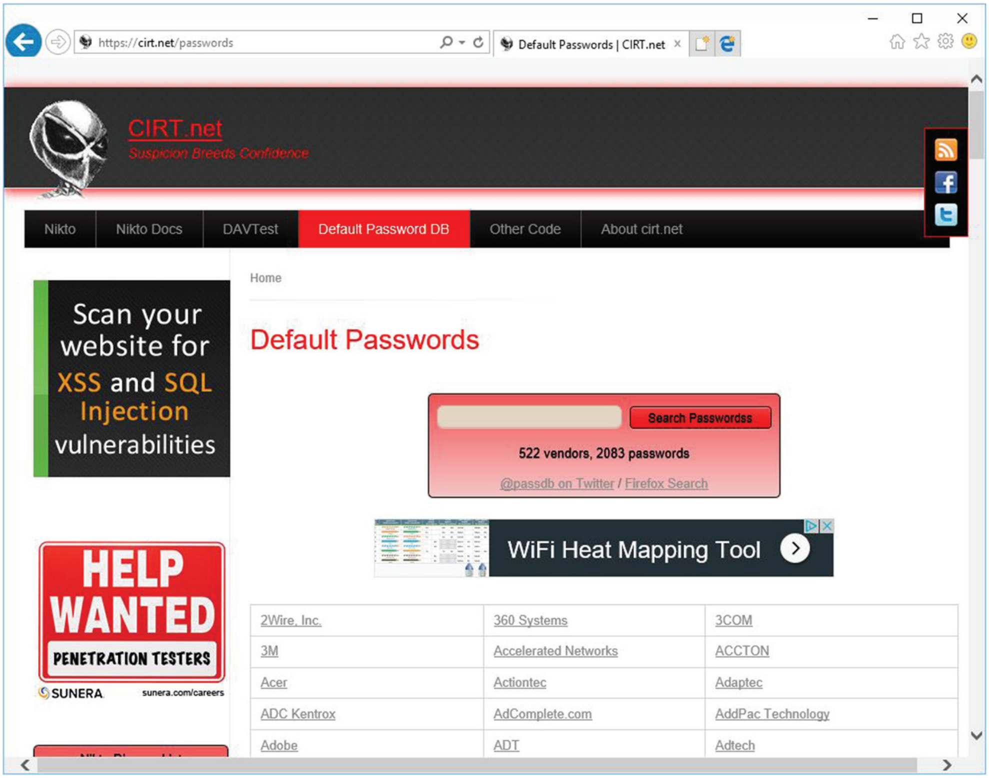
Applications can be specifically allowed or disallowed; see the Chapter 2 section "Application whitelisting" for details.

Disable default accounts/passwords

If you don't need it, don't keep it. This may be an optional mantra for you in real life, but in terms of security, it's the first of two—the second is, lock down what's left. Getting rid of unnecessary services and accounts is just the beginning of proper security and environment hardening. Leaving behind default or unused accounts gives hackers and attackers more potential points of compromise.

Always change default passwords to something unique and complex. All default passwords are available online (Figure 3.12). If available, always turn on password protection and set a complex password. Don't assume physical access control is good enough or that logical remote access isn't possible. If you discover you have a device with a hard-coded default password (meaning the original from-the-vendor password cannot be disabled or replaced), then remove that device from the network and replace it with a device that offers real security configuration options.

FIGURE 3.12 CIRT.net's default password database


https://cirt.net/passwords
Default Passwords | CIRT.net
CIRT.net
Suspicion Breeds Confidence
Nikto
Nikto Docs
DAVTest
Default Password DB
Other Code
About cirt.net
Home
Scan your website for XSS and SQL Injection vulnerabilities
HELP WANTED
PENETRATION TESTERS
SUNERA
sunera.com/careers
Search Passwordss
@passdb on Twitter / Firefox Search
WiFi Heat Mapping Tool

Default Passwords

522 vendors, 2083 passwords

2Wire, Inc.	360 Systems	3COM
3M	Accelerated Networks	ACCTON
Acer	Actiontec	Adaptec
ADC Kentrox	AdComplete.com	AddPac Technology
Adobe	ADT	Adtech

Peripherals

System hardware and peripherals require physical access controls and protections in order to maintain the logical security imposed by software. Without access control over the facility and physical environment, otherwise secured systems can be quickly compromised. Physical protections are used to protect against physical attacks, whereas logical protections protect only against logical attacks. Without adequate layers of protection, security is nonexistent. This section discusses several issues related to peripherals that often lead to security compromise because they're overlooked or deemed non-serious threats.

Wireless keyboards

A wireless keyboard connects to a computer over Bluetooth, WiFi, or some other radio wave–based communication. In most cases, wireless keyboards do not use encryption, so any listening device within range may be able to eavesdrop on the characters typed into a wireless keyboard. Do not use wireless keyboards for sensitive systems or when you are typing sensitive, confidential, or valuable information.

Wireless mice

Like wireless keyboards, wireless mice connect to a computer over Bluetooth, WiFi, or some other radio wave–based communication. In most cases, wireless mice do not use encryption, so any listening device within range may be able to eavesdrop on the movements and activities of a wireless mouse. Do not use a wireless mouse on sensitive systems.

Displays

A display can show sensitive, confidential, or personal information. It is important to orient your system display so it is hard to see unless you are sitting or standing in your work position. You do not want others in the general area or who just walk by to be able to see the contents of your screen. It may be worthwhile to install screen filters, also called privacy filters. As discussed under "Screen filters" later in the chapter, these devices reduce the range of visibility of a screen down to a maximum of 30 degrees from perpendicular.

Some displays are wireless, and signals from the core computer to the display itself are unlikely to be encrypted. Anyone in the area may be able to eavesdrop on your display communications in order to see what is being shown through your monitor.

WiFi-enabled MicroSD cards

Many portable devices that do not have native wireless support can be enhanced using a microSD or SD card with WiFi. These memory storage cards include their own WiFi adapter, which can in turn provide wireless connectivity to the mobile device. While not turning the mobile device into a full-fledged NOS, it often allows for uploading or backing up of files (that are saved to the storage expansion card) to a cloud service or a network share. These WiFi–enabled SD and microSD cards may support wireless encryption, but not necessarily.

When these cards are used in a device that already has networking capabilities, the device can serve as an additional attack path for hackers—especially if it automatically connects to plain-text WiFi networks. In general, avoid the use of any WiFi–enabled storage expansion card. Although it's less convenient, manually moving the SD card to a full computer system in order to upload files is faster and likely more secure.

Printers/MFDs

Many printers are network attached printers, meaning they can be directly connected to the network without being directly attached to a computer. A network-attached printer serves as its own print server. It may connect to the network via cable or through wireless. Some devices are more than just printers and may include fax, scanning, and other functions. These are known as multifunction devices (MFDs). Any device connected to a network can be a potential breach point. This may be due to flaws in the firmware of the device as well as whether or not the device uses communication encryption.

External storage devices

Universal Serial Bus (USB) devices are ubiquitous these days. Nearly every worker who uses a computer possesses a USB storage device, and most portable devices (such as phones, music players, and still or video cameras) connect via USB. However, this convenience comes at a cost to security. There are at least four main issues:

- Just about any USB device can be used to either bring malicious code into or leak sensitive, confidential, and/or proprietary data out of an otherwise secure environment. Even a device not specifically designed as a storage device, such as a mobile phone, might still serve that function.
- Most computers have the ability to boot off USB. This could allow a user to boot a computer to an alternate OS (such as Kali, a live Linux distribution used for hacking and/or penetration testing), which fully bypasses any security the native OS imposes.
- Some more recent malware uses the Autoruns feature of Windows to spread from infected USB storage devices to the host computer. Such malware will succeed if security measures such as updating, patching, and hardening systems with up-to-date antivirus protection aren't in place.
- USB auto-typers have the ability to brute-force logins with thousands of attempts per second.

To protect against USB threats, the only real option is to fully disallow the use of USB devices and lock down all USB ports. Some organizations not only disable USB functionality but also physically fill USB ports with silicon, epoxy, or a similar material, thus ensuring that USB devices can't be used. As businesses move to USB keyboards and mice, the epoxy trick is less effective: users can simply remove their input devices and attach a USB drive either directly or through a hub. Instead, more businesses are disabling USB boot in the (then-locked) BIOS and disabling USB Autoruns in the OS. Otherwise, allowing the use of USB typically leaves your organization's system vulnerable to threats.

Digital cameras

Digital cameras can be a security risk when they are used to take photographs of sensitive documents or information on a computer screen. A digital camera can serve as a storage device when connected via a cable to a computer system, thus allowing the user to transmit confidential files to outside the organization or to bring in malware from outside. A digital camera might support wireless communications natively or have that feature added through the use of a WiFi–enabled storage card. Most digital cameras include GPS chips in order to geotag photos and videos created on the device. This can reveal sensitive or secret locations, as well as the time and date of a photo being taken or a video being recorded.

Exam Essentials

Understand hardware security. System hardware and peripherals require physical access controls and protections in order to maintain the logical security imposed by software. Without access control over the facility and physical environment, otherwise secured systems can be quickly compromised.

Know about FDE and SED. Full-disk encryption (FDE) or whole-disk encryption is often used to provide protection for an OS, its installed applications, and all locally stored data. FDE encrypts all of the data on a storage device with a single master symmetric encryption key. Another option is self-encrypting drives (SEDs).

Understand TPM. The trusted platform module (TPM) is both a specification for a cryptoprocessor and the chip in a mainboard supporting this function. A TPM chip is used to store and process cryptographic keys for a hardware-supported and -implemented hard drive encryption system.

Define HSM. A hardware security module (HSM) is a special-purpose cryptoprocessor used for a wide range of potential functions. The functions of an HSM can include accelerated cryptography operations, managing and storing encryption keys, offloading digital signature verification, and improving authentication.

Understand UEFI and BIOS. Basic input/output system (BIOS) is the basic low-end firmware or software embedded in the hardware's electrically erasable programmable read-only memory (EEPROM). BIOS identifies and initiates the basic system hardware components. A replacement or improvement to BIOS is Unified Extensible Firmware Interface (UEFI). UEFI provides support for all of the same functions as that of BIOS with many improvements, such as support for larger hard drives (especially for booting), faster boot times, enhanced security features, and even the ability to use a mouse when making system changes.

Comprehend OS security. There is no fully secure OS. All of them have security flaws. Every OS needs some level of security management imposed on it.

Understand EMI shielding. Shielding is used to restrict or control interference from electromagnetic or radio frequency disturbances. This can include using shielded cabling or cabling that is resistant to interference, or running cables through shielded conduits.

Realize the importance of disabling unnecessary ports and services. If a system is hosting numerous services and protocols, its attack surface is larger than that of a system running only essential services and protocols.

Understand least functionality. One rule of thumb to adopt when designing and implementing security is that of least functionality. If you always select and install the solution with the least functionality or without any unnecessary additional capabilities and features, then you will likely have a more secure result than opting for any solution with more options than necessary.

Know the concept of trusted OS. Trusted OS is an access-control feature that requires a specific OS to be present in order to gain access to a resource. Another formal definition of trusted OS is any OS that has security features in compliance with government and/or military security standards that enable the enforcement of multilevel security policies.

Understand peripheral security. System hardware and peripherals require physical access controls and protections in order to maintain the logical security imposed by software.

3.4 Explain the importance of secure staging deployment concepts.

Secure staging is the controlled process of configuration and deployment for new systems, whether hardware or software. The goal of a secure staging process is to ensure compliance with the organization's security policies and configuration baselines while minimizing risks associated with exposing an insecure system to a private network or even the Internet. This section discusses several elements that may be part of a secure staging system.

Sandboxing

Sandboxing is a means of quarantine or isolation. It's implemented to keep new or otherwise suspicious software from being able to cause harm to production systems. It can be used against applications or entire OSs.

Sandboxing is simple to implement in a virtualization context because you can isolate a virtual machine with a few mouse clicks or commands. Once the suspect code is deemed safe, you can release it to integrate with the environment. If it's found to be malicious, unstable, or otherwise unwanted, it can quickly be removed from the environment with little difficulty.

Environment

The organization's IT environment must be configured and segmented to properly implement staging. This often requires at least four main network divisions: development, test, staging, and production.

Development

The development network is where new software code is being crafted by on-staff programmers and developers. For some organizations, this might also be where custom-built hardware is being created. This network is to be fully isolated from all other network divisions in order to prevent ingress of malware or egress of unfinished products.

Test

The test network is where in-development products or potentially final versions of products are subjected to a battery of evaluations, stress tests, vulnerability scans, and even attack attempts in order to determine whether the product is stable, secure, and ready for deploying into the production network.

Staging

The staging network is where new equipment, whether developed in-house or obtained from external vendors, is configured to be in compliance with the company's security policy and configuration baseline. Once a system has been staged, it can be moved to the test network for evaluation. After the system has passed evaluation, it can be deployed into the production network.

Production

The production network is where the everyday business tasks and work processes are accomplished. It should only be operating on equipment and systems that have been properly staged and tested. The production network should be managed so that it is not exposed to the risk and unreliability of new systems and untested solutions. The goal of the production network is to support the confidentiality, integrity, and availability (among other goals) of the organization's data and business tasks.

Secure baseline

The *security posture* is the level to which an organization is capable of withstanding an attack. An organization may have good or poor posture. A plan and implementation are parts of the security posture often known as the *secure baseline* or *security baseline*. These include detailed policies and procedures, implementation in the IT infrastructure and the facility, and proper training of all personnel.

One mechanism often used to help maintain a hardened system is to use a security baseline, a standardized minimal level of security that all systems in an organization must comply with. This lowest common denominator establishes a firm and reliable security structure on which to build trust and assurance. The security baseline is defined by the organization's security policy. Creating or defining a baseline requires that you examine three key areas of an environment: the OS, the network, and the applications. It may include requirements of specific hardware components, OS versions, service packs, patches and upgrades, configuration settings, add-on applications, service settings, and more.

The basic procedure for establishing a security baseline or hardening a system is as follows:

1. Remove unneeded components, such as protocols, applications, services, and hardware (including device drivers).
2. Update and patch the OS and all installed applications, services, and protocols.
3. Configure all installed software as securely as possible.
4. Impose restrictions on information distribution for the system, its active services, and its hosted resources.

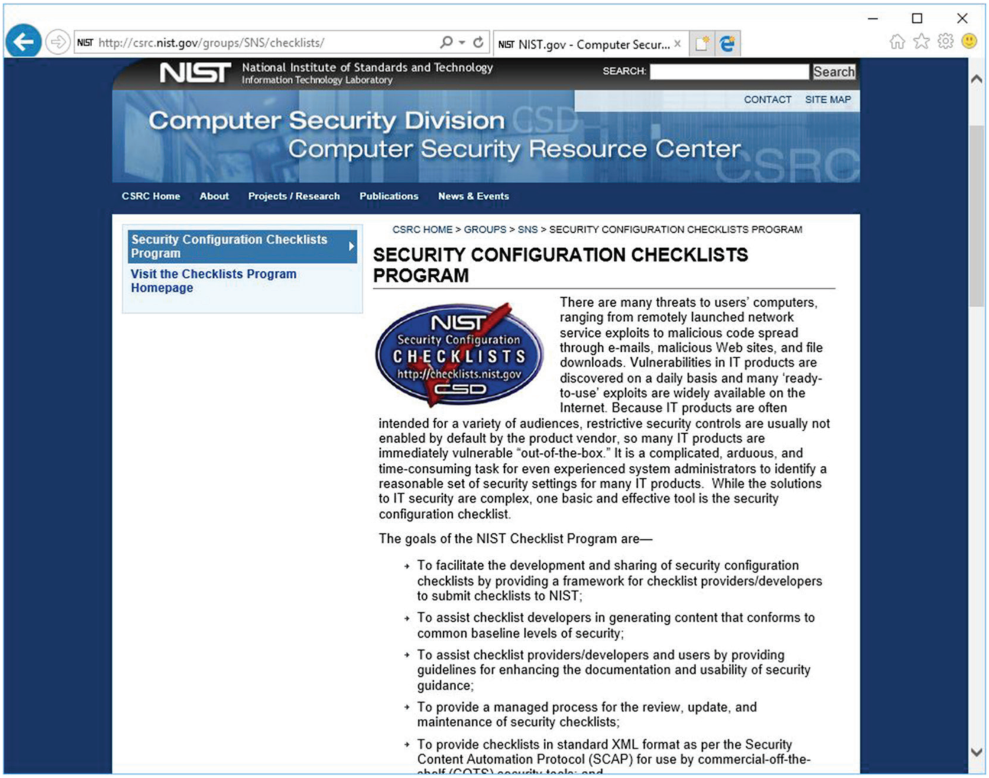
Documentation is an important aspect of establishing a security baseline and implementing security in an environment. Every aspect of a system, from design to implementation, tuning, and securing, should be documented. A lack of sufficient documentation is often the primary cause of difficulty in locking down or securing a server. Without proper documentation, all the details about the OS, hardware configuration, applications, services, updates, patches, configuration, and so on must be discovered before security improvements can be implemented. With proper documentation, a security professional can quickly add to the existing security without having to reexamine the entire environment.

A *security template* is a set of security settings that can be mechanically applied to a computer to establish a specific configuration. Security templates can be used to establish baselines or bring a system up to compliance with a security policy. They can be custom-designed for workstations and server functions or purposes. Security templates are a generic concept; however, specific security templates can be applied via Windows' Group Policy system.

Security templates can be built by hand or by extracting settings from a preconfigured master. Once a security template exists, you can use it to configure a new or existing machine (by applying the template to the target either manually or through a Group Policy object [GPO]), or to compare the current configuration to the desired configuration. This latter process is known as *security template analysis* and often results in a report detailing the gaps in compliance.

Operating system hardening is the process of reducing vulnerabilities, managing risk, and improving the security provided by or for an OS. This is usually accomplished by taking advantage of an OS's native security features and supplementing them with add-on applications such as firewalls, antivirus software, and malicious-code scanners. There are several online sources of security configuration hardening and configuration checklists, such as the NIST Security Configuration Checklists Program site at `http://csrc.nist.gov/groups/SNS/checklists/` (Figure 3.13), which can be used as a starting point for crafting an organization-specific set of SOPs.

FIGURE 3.13 The NIST Security Configuration Checklists Program site


http://csrc.nist.gov/groups/SNS/checklists/
NIST NIST.gov - Computer Secur...
NIST National Institute of Standards and Technology
Information Technology Laboratory
SEARCH:
Search
CONTACT
SITE MAP
Computer Security Division CSD
Computer Security Resource Center CSRC
CSRC Home
About
Projects / Research
Publications
News & Events
Security Configuration Checklists Program
Visit the Checklists Program Homepage
CSRC HOME > GROUPS > SNS > SECURITY CONFIGURATION CHECKLISTS PROGRAM
NIST Security Configuration CHECKLISTS http://checklists.nist.gov CSD

SECURITY CONFIGURATION CHECKLISTS PROGRAM

There are many threats to users' computers, ranging from remotely launched network service exploits to malicious code spread through e-mails, malicious Web sites, and file downloads. Vulnerabilities in IT products are discovered on a daily basis and many 'ready-to-use' exploits are widely available on the Internet. Because IT products are often intended for a variety of audiences, restrictive security controls are usually not enabled by default by the product vendor, so many IT products are immediately vulnerable "out-of-the-box." It is a complicated, arduous, and time-consuming task for even experienced system administrators to identify a reasonable set of security settings for many IT products. While the solutions to IT security are complex, one basic and effective tool is the security configuration checklist.

The goals of the NIST Checklist Program are—

- To facilitate the development and sharing of security configuration checklists by providing a framework for checklist providers/developers to submit checklists to NIST;
- To assist checklist developers in generating content that conforms to common baseline levels of security;
- To assist checklist providers/developers and users by providing guidelines for enhancing the documentation and usability of security guidance;
- To provide a managed process for the review, update, and maintenance of security checklists;
- To provide checklists in standard XML format as per the Security Content Automation Protocol (SCAP) for use by commercial-off-the-shelf (COTS) security tools; and

Hardening an OS includes protecting the system from both intentional directed attacks and unintentional or accidental damage. This can include implementing security countermeasures as well as fault-tolerant solutions for both hardware and software. Some of the actions that are often included in a system-hardening procedure include the following:

- Deploy the latest version of the OS.
- Apply any service packs or updates to the OS.
- Update the versions of all device drivers.
- Verify that all remote-management or remote-connectivity solutions that are active are secure. Avoid FTP, Telnet, and other clear-text or weak authentication protocols.
- Disable all unnecessary services, protocols, and applications.
- Remove or securely configure Simple Network Management Protocol (SNMP).
- Synchronize time zones and clocks across the network with an Internet time server.
- Configure event-viewer log settings to maximize capture and storage of audit events.
- Rename default accounts, such as administrator, guest, and admin.
- Enforce strong passwords on all accounts.
- Force password changes on a periodic basis.
- Restrict access to administrative groups and accounts.
- Hide the last-logged-on user's account name.
- Enforce account lockout.
- Configure a legal warning message that's displayed at logon.
- If file sharing is used, force the use of secure sharing protocols or use virtual private networks (VPNs).
- Use a security and vulnerability scanner against the system.
- Scan for open ports.
- Disable Internet Control Message Protocol (ICMP) functionality on publicly accessible systems.
- Consider disabling NetBIOS.
- Configure auditing.
- Configure backups.

The filesystem in use on a system greatly affects the security offered by that system. A filesystem that incorporates security, such as access control and auditing, is a more secure choice than a filesystem without incorporated security. One great example of a secured filesystem is the Microsoft New Technology File System (NTFS). It offers file- and folder-level access permissions and auditing capabilities. Examples of filesystems that don't include security are file allocation table (FAT) and FAT32.

Integrity measurement

The primary means of integrity measurement or assessment is the use of a hash. See the Chapter 6 section, "Hashing," for details.

Exam Essentials

Understand secure staging. Secure staging is the controlled process of configuration and deployment for new systems, whether hardware or software. The goal of a secure staging process is to ensure compliance with the organization's security policies and configuration baselines while minimizing risks associated with exposing an insecure system to a private network or even the Internet.

Comprehend sandboxing. Sandboxing is a means of quarantine or isolation. It's implemented to restrict new or otherwise suspicious software from being able to cause harm to production systems.

Understand a secure IT environment. The organization's IT environment must be configured and segmented to properly implement staging. This often requires at least four main network divisions: development, test, staging, and production.

Realize the importance of a security baseline. A plan and implementation are parts of the security posture often known as the secure baseline or security baseline. These include detailed policies and procedures, implementation in the IT infrastructure and the facility, and proper training of all personnel.

3.5 Explain the security implications of embedded systems.

An *embedded system* is a computer implemented as part of a larger system. The embedded system is typically designed around a limited set of specific functions in relation to the larger product of which it's a component. It may consist of the same components found in a typical computer system, or it may be a microcontroller (an integrated chip with on-board memory and peripheral ports). Examples of embedded systems include network-attached printers, smart TVs, HVAC controls, smart appliances, smart thermostats, embedded smart systems in vehicles, and medical devices.

Security concerns regarding embedded systems include the fact that most are designed with a focus on minimizing cost and extraneous features. This often leads to a lack of security and difficulty with upgrades or patches. Because an embedded system is in control of a mechanism in the physical world, a security breach could cause harm to people and property.

A *static environment* is a set of conditions, events, and surroundings that don't change. In theory, once understood, a static environment doesn't offer new or surprising elements. In technology, static environments are applications, OSs, hardware sets, or networks that are configured for a specific need, capability, or function, and then set to remain unaltered. A *static IT environment* is any system that is intended to remain unchanged by users and administrators. The goal is to prevent or at least reduce the possibility of a user implementing change that could result in reduced security or functional operation.

However, although the term *static* is used, there are no truly static systems. There is always the chance that a hardware failure, a hardware configuration change, a software bug, a software-setting change, or an exploit may alter the environment, resulting in undesired operating parameters or actual security intrusions. Many embedded systems are implemented as static solutions. It is important to understand the various ways to protect the stability and security of embedded and/or static systems. Static environments, embedded systems, and other limited or single-purpose computing environments need security management. Although they may not have as broad an attack surface and aren't exposed to as many risks as a general-purpose computer, they still require proper security governance.

Manual Updates *Manual updates* should be used in static environments to ensure that only tested and authorized changes are implemented. Using an automated update system would allow untested updates to introduce unknown security reductions.

Firmware Version Control Similar to manual software updates, strict control over firmware versions in a static environment is important. Firmware updates should be implemented on a manual basis, only after testing and review. Oversight of firmware version control should focus on maintaining a stable operating platform while minimizing exposure to downtime or compromise.

Wrappers A *wrapper* is something used to enclose or contain something else. Wrappers are well known in the security community in relation to Trojan horse malware. A wrapper of this sort is used to combine a benign host with a malicious payload.

Wrappers are also used as encapsulation solutions. Some static environments may be configured to reject updates, changes, or software installations unless they're introduced through a controlled channel. That controlled channel can be a specific wrapper. The wrapper may include integrity and authentication features to ensure that only intended and authorized updates are applied to the system.

SCADA/ICS

Supervisory control and data acquisition (SCADA) is a type of *industrial control system (ICS)*. An ICS is a form of computer-management device that controls industrial processes and machines. SCADA is used across many industries, including manufacturing, fabrication, electricity generation and distribution, water distribution, sewage processing, and oil refining. A SCADA system can operate as a stand-alone device, be networked together with other SCADA systems, or be networked with traditional IT systems.

Most SCADA systems are designed with minimal human interfaces. Often, they use mechanical buttons and knobs or simple LCD screen interfaces (similar to what you might have on a business printer or a GPS navigation device). However, networked SCADA devices may have more complex remote-control software interfaces.

In theory, the static design of SCADA and the minimal human interface should make the system fairly resistant to compromise or modification. Thus, little security was built into SCADA devices, especially in the past. But there have been several well-known compromises of SCADA; for example, Stuxnet delivered the first-ever rootkit to a SCADA system located in a nuclear facility. Many SCADA vendors have started implementing security improvements into their solutions in order to prevent or at least reduce future compromises.

Smart devices/IoT

Smart devices are a range of mobile devices that offer the user a plethora of customization options, typically through installing apps, and may take advantage of on-device or in-the-cloud artificial intelligence (AI) processing. The products that can be labeled "smart devices" are constantly expanding and already include smartphones, tablets, music players, home assistants, extreme sport cameras, and fitness trackers.

Android is a mobile device OS based on Linux, which was acquired by Google in 2005. In 2008, the first devices hosting Android were made available to the public. The Android source code is made open source through the Apache license, but most devices also include proprietary software. Although it's mostly intended for use on phones and tablets, Android is being used on a wide range of devices, including televisions, game consoles, digital cameras, microwaves, watches, e-readers, cordless phones, and ski goggles.

The use of Android in phones and tablets isn't a good example of a static environment. These devices allow for a wide range of user customization: you can install both Google Play Store apps and apps from unknown external sources (such as Amazon's App Store), and many devices support the replacement of the default version of Android with a customized or alternate version. However, when Android is used on other devices, it can be implemented as something closer to a static system.

Whether static or not, Android has numerous security vulnerabilities. These include being exposed to malicious apps, running scripts from malicious websites, and allowing insecure data transmissions. Android devices can often be rooted (breaking their security and access limitations) in order to grant the user full root-level access to the device's low-level configuration settings. Rooting increases a device's security risk, because all running code inherits root privileges.

Improvements are made to Android security as new updates are released. Users can adjust numerous configuration settings to reduce vulnerabilities and risks. Also, users may be able to install apps that add additional security features to the platform.

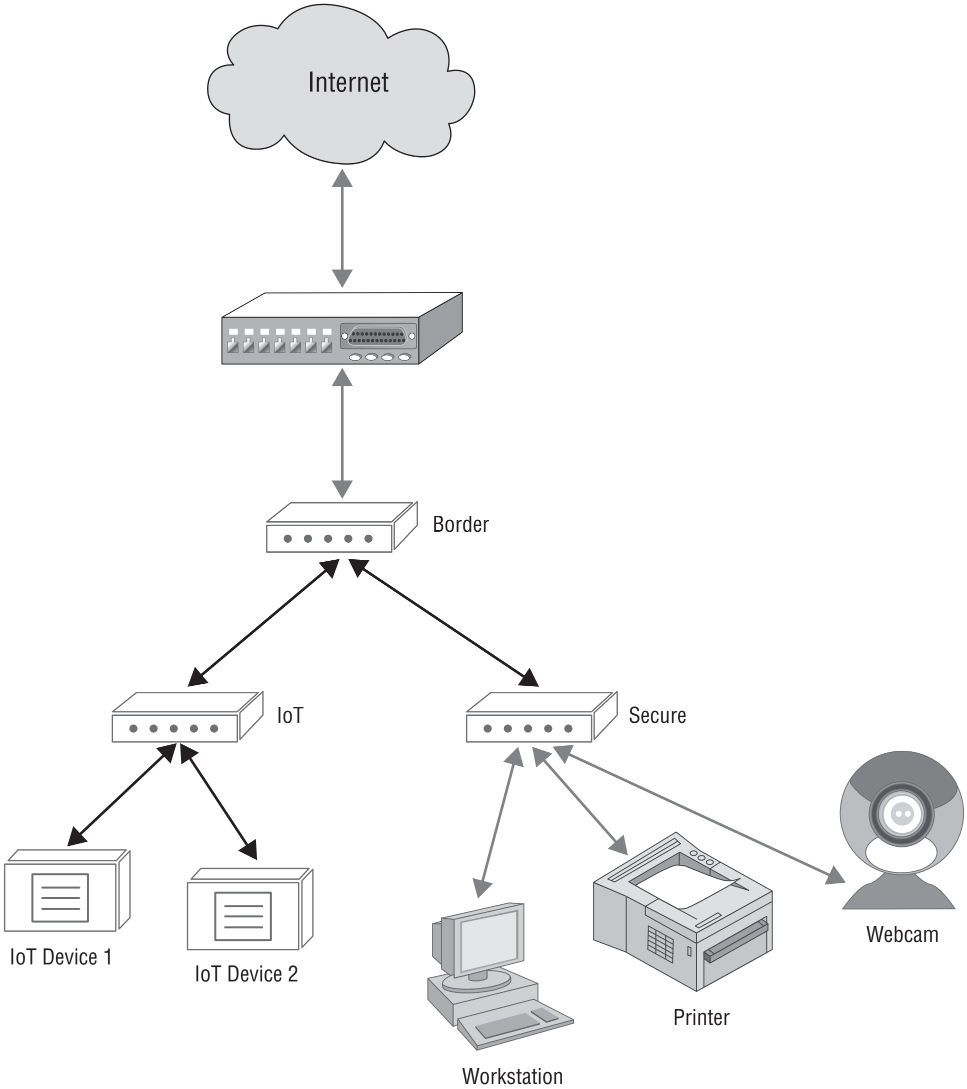
iOS is the mobile device OS from Apple that is available on the iPhone, iPad, iPod, and Apple TV. iOS isn't licensed for use on any non-Apple hardware. Thus, Apple is in full control of the features and capabilities of iOS. However, iOS is also a poor example of a static environment, because users can install any of over one million apps from the Apple App Store. Also, it's often possible to jailbreak iOS (breaking Apple's security and access restrictions), allowing users to install apps from third parties and gain greater control over low-level settings. Jailbreaking an iOS device reduces its security and exposes the device to potential compromise. Users can adjust device settings to increase an iOS device's security and install many apps that can add security features.

The Internet of Things (IoT) is a new subcategory or even a new class of devices that are Internet-connected in order to provide automation, remote control, or AI processing to traditional or new appliances or devices in a home or office setting. IoT devices are sometimes revolutionary adaptations of functions or operations we may have been performing locally and manually for decades, which we would not want to ever be without again. Other IoT devices are nothing more than expensive gimmicky gadgets that, after the first few moments of use, are forgotten about and/or discarded. The security issues related to IoT are about access and encryption. All too often an IoT device was not designed with security as a core concept or even an afterthought. This has already resulted in numerous home and office network security breaches. Additionally, once an attacker has remote access to or through an IoT device, they may be able to access other devices on the compromised network. When electing to install IoT equipment, evaluate the security of the device as well as the security reputation of the vendor. If the new device does not have the ability to meet or accept your existing security baseline, then don't compromise your security just for a flashy gadget.

One possible secure compromise is to deploy a distinct network for the IoT equipment, which is kept separate and isolated from the primary network. This configuration is often known as the three dumb routers (Figure 3.14) (see `https://www.grc.com/sn/sn-545.pdf` or `https://www.pcper.com/reviews/General-Tech/Steve-Gibsons-Three-Router-Solution-IOT-Insecurity`).

While we often associate smart devices and IoT with home or personal use, they are also a concern to every organization. This is partly due to the use of mobile devices by employees within the company's facilities and even on the organizational network. These concerns are often addressed in a BYOD, COPE, or CYOD policy (see the Chapter 2 section "Deployment models" for more information). Another aspect of network professional concern is that many IoT or networked automation devices are being added to the business environment. This includes environmental controls, such as HVAC management, air quality control, debris and smoke detection, lighting controls, door automation, personnel and asset tracking, and consumable inventory management and auto-reordering (such as coffee, snacks, printer toner, paper, and other office supplies). Thus, both smart devices and IoT devices are potential elements of a modern business network that need appropriate security management and oversight. For some additional reading on the importance of proper security management of smart devices and IoT equipment, please see "NIST Initiatives in IoT" at `https://www.nist.gov/itl/applied-cybersecurity/nist-initiatives-iot`.

FIGURE 3.14 Three-dumb-router network layout



Wearable technology

Wearable technology are offshoots of smart devices and IoT devices that are specifically designed to be worn by an individual. The most common examples of wearable technology are smart watches and fitness trackers. There are an astounding number of options in these categories available, with a wide range of features and security capabilities. When selecting a wearable device, consider the security implications. Is the data being collected in a cloud service that is secured for private use or is it made publicly available? What alternative uses is the collected data going to be used for? Is the communication between the device and the collection service encrypted? And can you delete your data and profile from the service completely if you stop using the device?

Home automation

A very popular element of smart devices and IoT is home automation devices. These include smart thermostats, ovens, refrigerators, garage doors, doorbells, door locks, and security cameras. These IoT devices may offer automation or scheduling of various mundane, tedious, or inconvenient activities, such as managing the household heating and cooling systems, adding groceries to an online shopping list, automatically opening or unlocking doors as you approach, recording visitors to your home, and cooking dinner so it is ready just as you arrive home from work.

The precautions related to home automation devices are the same as for smart devices, IoT, and wearables. Always consider the security implications, evaluate the included or lacking security features, consider implementing the devices in an isolated network away from your other computer equipment, and only use solutions that provide robust authentication and encryption.

HVAC

HVAC (heating, ventilation, and air conditioning) can be controlled by an embedded solution (which might be also known as a smart device or an IoT device). See the previous discussion on smart devices for security issues, and see the later section “HVAC” under “Environmental controls.” Physical security controls protect against physical attacks, while logical and technical controls only protect against logical and technical attacks.

SoC

A System on a Chip (SoC) is an integrated circuit (IC) or chip that has all of the elements of a computer integrated into a single chip. This often includes the main CPU, memory, a GPU, WiFi, wired networking, peripheral interfaces (such as USB), and power management. In most cases the only item missing from a SoC compared to a full computer is bulk storage. Often a bulk storage device must be attached or connected to the SoC to store its programs and other files, since the SoC usually contains only enough memory to retain its own firmware or OS.

The security risks of an SoC include the fact that the firmware or OS of an SoC is often minimal, which leaves little room for most security features. An SoC may be able to filter input (such as by length or to escape metacharacters), reject unsigned code, provide basic firewall filtering, use communication encryption, and offer secure authentication. But these features are not universally available on all SoC products. A few devices that use a SoC include the mini-computer Raspberry Pi, fitness trackers, smart watches, and some smartphones.

RTOS

A real-time operating system (RTOS) is designed to process or handle data as it arrives on the system with minimal latency or delay. An RTOS is usually stored on read-only memory (ROM) and is designed to operate in a hard real-time or soft real-time condition. A hard real-time solution is for mission-critical operations where delay must be eliminated or minimized for safety, such as autonomous cars. A soft real-time solution is used when some level of modest delay is acceptable under typical or normal conditions, as it is for most consumer electronics, such as the delay between a digitizing pen and a graphics program on a computer.

RTOSs can be event-driven or time-sharing. An event-driven RTOS will switch between operations or tasks based on preassigned priorities. A time-sharing RTOS will switch between operations or tasks based on clock interrupts or specific time intervals. An RTOS is often implemented when scheduling or timing is the most critical part of the task to be performed.

A security concern using RTOSs is that these systems are often very focused and single-purpose, leaving little room for security. They often use custom or proprietary code, which may include unknown bugs or flaws that could be discovered by attackers. An RTOS might be overloaded or distracted with bogus data sets or process requests by malware. When deploying or using RTOSs, use isolation and communication monitoring to minimize abuses.

Printers/MFDs

See the earlier section “Printers/MFDs” under “Peripherals” for an introduction to their security implications. A printer or multifunction device (MFD) can be considered an embedded device if it has integrated network capabilities that allow it to operate as an independent network node rather than a direct-attached dependent device. Thus, network-attached printers and other similar devices pose an increased security risk because they often house full-fledged computers within their chassis. Network security managers need to include all such devices in their security management strategy in order to prevent these devices from being the targets of attack, used to house malware or attack tools, or grant outsiders remote-control access.

Camera systems

See the earlier section “Digital cameras” under “Peripherals” for an introduction to their security implications. Some camera systems include an SoC or embedded components that grant them network capabilities. Cameras that operate as network nodes can be remotely controlled; can provide automation functions; and may be able to perform various specialty functions, such as time-lapse recording, tracking, facial recognition, or infrared or color-filtered recording. Such devices may be targeted by attackers, be infected by malware, or be remotely controlled by hackers. Network security managers need to include all such devices in their security management strategy to prevent these compromises.

Special purpose

The concept of embedded systems is rapidly expanding as computer control, remote access, remote management, automation, monitoring, and AI processing are being applied to professional and personal events, activities, and tasks. In addition to the concepts mentioned previously in this section, there are a handful of additional special purpose embedded systems you should be familiar with. These include mainframes, game consoles, medical devices, vehicles, and aircraft/UAVs.

Mainframes are high-end computer systems used to perform highly complex calculations and provide bulk data processing. Older mainframes may be considered static environments because they were often designed around a single task or supported a single mission-critical application. These configurations didn't offer significant flexibility, but they did provide for high stability and long-term operation. Many mainframes were able to operate for decades.

Modern mainframes are much more flexible and are often used to provide high-speed computation power in support of numerous virtual machines. Each virtual machine can be used to host a unique OS and in turn support a wide range of applications. If a modern mainframe is implemented to provide fixed or static support of one OS or application, it may be considered a static environment.

Game consoles, whether home systems or portable systems, are potentially examples of static systems and embedded systems. The OS of a game console is generally fixed and is changed only when the vendor releases a system upgrade. Such upgrades are often a mixture of OS, application, and firmware improvements. Although game console capabilities are generally focused on playing games and media, modern consoles may offer support for a range of cultivated and third-party applications. The more flexible and open-ended the app support, the less of a static system it becomes.

Medical devices

A growing number of medical devices have been integrated with IoT technology to make them remotely accessible for monitoring and management. This may be a great innovation for medical treatment, but it also has security risks. All computer systems are subject to attack and abuse. All computer systems have faults and failings that can be discovered and abused by an attacker. Although most medical device vendors strive to provide robust and secure products, it is not possible to consider and test for every possibility of attack, access, or abuse. There have already been several instances of medical devices being remotely controlled, disabled, accessed, or attacked with a DoS. When using any medical device, consider whether remote access, wired or wireless, is essential to the medical care it is providing. If not, it may still make sense to disable the network feature of the medical device. Although the breach of a personal computer or smartphone may be inconvenient and/or embarrassing, the breach of a medical device can be life-threatening.

Vehicles

In-vehicle computing systems can include the components used to monitor engine performance and optimize braking, steering, and suspension, but can also include in-dash elements related to driving, environment controls, and entertainment. Early in-vehicle systems were static environments with little or no ability to be adjusted or changed, especially by the owner/driver. Modern in-vehicle systems may offer a wider range of capabilities, including linking a mobile device or running custom apps. In-vehicle computing systems may or may not have sufficient security mechanisms. Even if the system is only providing information, such as engine performance, entertainment, and navigation, it is important to consider what, if any, security features are included in the solution. Does it connect to cloud services, are communications encrypted, how strong is the authentication, is it easily accessible to unauthorized third parties? If the in-vehicle computing system is controlling the vehicle, which might be called automated driving or self-driving, it is even more important that security be a major design element of the system. Otherwise, a vehicle can be converted from a convenient means of transference to a box of death.

Aircraft/UAV

Automated pilot systems have been part of aircraft for decades. In most of the airplanes that you have flown on, a human pilot was likely only in full control of the craft during takeoff and landing, and not always even then. For most of the flight, the autopilot system was likely in control of the aircraft. The military, law enforcement, and hobbyists have been using unmanned aerial vehicles (UAVs) for years, but usually under remote control. Now, with flight automation systems, UAVs can take off, fly to a destination, and land fully autonomously. There are even many retail businesses experimenting with, and in some countries implementing, UAV delivery of food and/or other packages.

The security of automated aircraft and UAVs is a concern for all of us. Are these systems secure against malware infection, signal disruption, remote control takeover, AI failure, and remote code execution? Does the UAV have authenticated connections to the authorized control system? Are the UAV's communications encrypted? What will the aircraft do in the event that all contact with the control system is blocked through DoS or signal jamming? A compromised UAV could result in the loss of your pizza, a damaged product, a few broken shingles, or severe bodily injury.

Exam Essentials

Understand embedded systems. An embedded system is a computer implemented as part of a larger system. The embedded system is typically designed around a limited set of specific functions in relation to the larger product of which it's a component.

Comprehend static environments. Static environments are applications, OSs, hardware sets, or networks that are configured for a specific need, capability, or function, and then set to remain unaltered. Examples include SCADA, embedded systems, Android, iOS, mainframes, game consoles, and in-vehicle computing systems.

Understand static environment security methods. Static environments, embedded systems, and other limited or single-purpose computing environments need security management. These techniques may include network segmentation, security layers, application firewalls, manual updates, firmware version control, wrappers, and control redundancy and diversity.

Know SCADA and ICS. Supervisory control and data acquisition (SCADA) is a type of industrial control system (ICS). An ICS is a form of computer-management device that controls industrial processes and machines. SCADA is used across many industries.

Understand smart devices. A smart device is a mobile device that offers the user a plethora of customization options, typically through installing apps, and may take advantage of on-device or in-the-cloud artificial intelligence (AI) processing.

Comprehend IoT. The Internet of Things (IoT) is a new subcategory or maybe even a new class of devices connected to the Internet in order to provide automation, remote control, or AI processing to traditional or new appliances or devices in a home or office setting.

Understand SoC. A System on a Chip (SoC) is an integrated circuit (IC) or chip that has all of the elements of a computer integrated into a single chip.

Know RTOS. A real-time operating system (RTOS) is designed to process or handle data as it arrives onto the system with minimal latency or delay. An RTOS is usually stored on read-only memory (ROM) and is designed to operate in a hard real-time or soft real-time condition.

3.6 Summarize secure application development and deployment concepts.

Secure software starts with a secure development and deployment system. Only if a software product was designed, crafted, and distributed in a secure fashion is it possible for the final product to provide reliable and trustable security. This section discusses several aspects of secure software deployment and development.

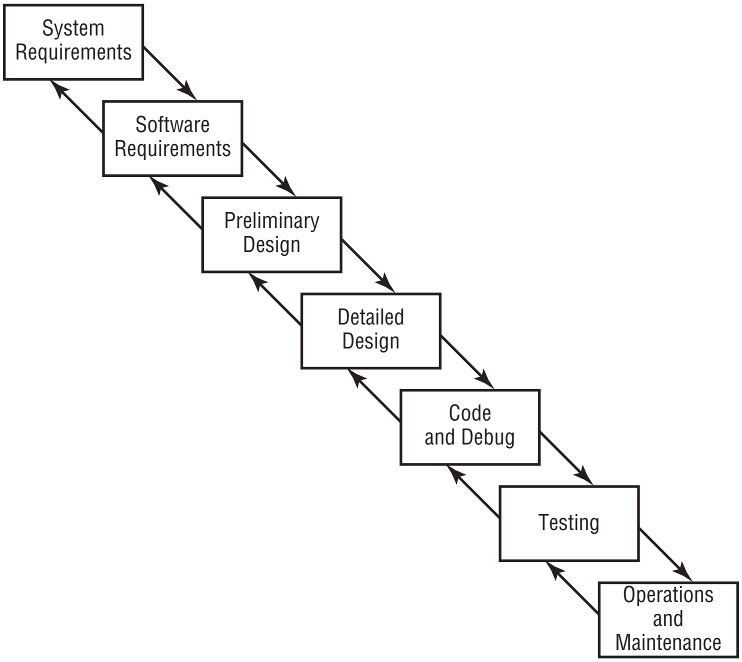
Development life-cycle models

A development life-cycle model is a methodical ordering of the tasks of creating a new product or revising an existing one. A formal *software development life-cycle (SDLC)* model helps to ensure a more reliable and stable product by establishing a standardized process by which new ideas become actual software. Software development has only existed as long as computers—less than 100 years. Modern software is only 30 or 40 years old. The earliest forms of software development management were forged in the 1970s and 1980s, but it wasn't until 1991, when the Software Engineering Institute established the Capability Maturity Model (CMM), that software management concepts were formally established and widely adopted.

Waterfall vs. Agile

Two of the dominant SDLC concepts are the waterfall model and the Agile model. The waterfall model (Figure 3.15) consists of seven stages, or steps. The original idea was that project development would proceed through these steps in order from first to last, with the restriction that returning to an earlier phase was not allowed. The name waterfall is derived from the concept of steps of rocks in a waterfall, where water falls onto each step to then move on down to the next, and the water is unable to flow back up. A more recent revision of the waterfall model allows for some movement back into earlier phases (hence the up arrows in the image) in order to address oversights or mistakes discovered in later phases.

FIGURE 3.15 The waterfall model

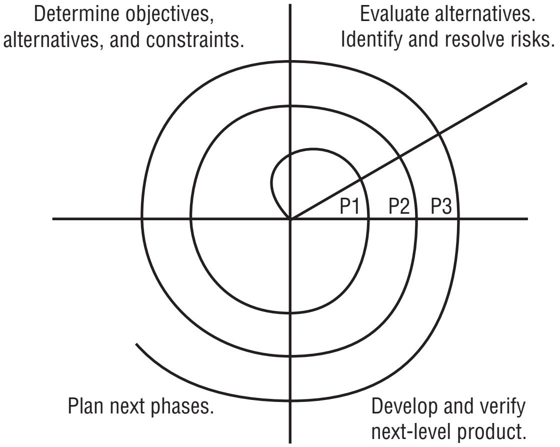


The primary criticism of the waterfall model is the limitation to only return to the immediately previous phase. This prevents returning to the earliest phases to correct concept and design issues that are not discovered until later in the development process. Thus, it forces the completion of a product that is known to be flawed or not to fulfill goals.

To address this concern, the modified waterfall model was crafted. This version adds a verification and validation process to each phase so that as a phase is completed, a review process ensures that each phase's purposes, functions, and goals were successfully and correctly fulfilled.

However, this model modification was not widely adopted before another variation was crafted, known as the spiral model. The spiral model (Figure 3.16) is designed around repeating the earlier phases multiple times, known as iterations, in order to ensure that each element and aspect of each phase is fulfilled in the final product.

FIGURE 3.16 The spiral model


Determine objectives, alternatives, and constraints.
Evaluate alternatives. Identify and resolve risks.
P1
P2
P3
Plan next phases.
Develop and verify next-level product.

In the diagram, each spiral traverses the first four initial phases. At the completion of an iteration, a prototype of the solution (P1, P2, …) is developed and tested. Based on the prototype, the spiral path is repeated. Multiple iterations are completed until the prototype fulfills all or most of the requirements of the initial phase or design goals and functions, at which point the final prototype becomes the final product.

One of the most modern SDLC models is Agile, based around adaptive development, where focusing on a working product and fulfilling customer needs is prioritized over rigid adherence to a process, use of specific tools, and detailed documentation. Agile focuses on an adaptive approach to development; it supports early delivery, continuous improvement, and flexible and prompt response to changes.

In 2001, 17 Agile development pioneers crafted the *Manifesto for Agile Software Development* (`http://agilemanifesto.org`), which states the core philosophy as follows:

> We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:
>
> **Individuals and interactions** over processes and tools
>
> **Working software** over comprehensive documentation
>
> **Customer collaboration** over contract negotiation
>
> **Responding to change** over following a plan
>
> That is, while there is value in the items on the right, we value the items on the left more.

Furthermore, the *Agile Manifesto* prescribed 12 principles that guide the development philosophy:

> "Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
>
> Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
>
> Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
>
> Business people and developers must work together daily throughout the project.
>
> Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
>
> The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
>
> Working software is the primary measure of progress.
>
> Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
>
> Continuous attention to technical excellence and good design enhances agility.
>
> Simplicity—the art of maximizing the amount of work not done—is essential.
>
> The best architectures, requirements, and designs emerge from self-organizing teams.
>
> At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly."

Agile is quickly becoming the dominant SDLC model, adopted by programming groups both large and small.

Secure DevOps

DevOps, or development and operations, is a new IT movement in which many elements and functions of IT management are being integrated into a single automated solution. DevOps typically consists of IT development, operations, security, and quality assurance. Secure DevOps is a variant of DevOps that prioritizes security in the collection of tasks performed under this new umbrella concept. DevOps is adopted by organizations crafting software solutions for internal use as well as products destined for public distribution. DevOps has many goals, including reducing time to market, improving quality, maintaining reliability, and implementing security into the design and development process.

Transforming DevOps into secure DevOps, or at least prioritizing security within DevOps, often includes several components, as discussed in the next sections.

Security automation

Security automation is important to DevOps in order to ensure that issues and vulnerabilities are discovered earlier so they can be properly addressed before product release. This will include automating vulnerability scans and code attacks against preproduction code, using fuzz testing techniques to discover logic flaws or the lack of input sanitization, and using code scanners that evaluate software for flaws and input management mistakes.

Continuous integration

In order for security to be successful in any development endeavor, it must be integrated and maintained at the beginning and throughout the development process. Secure DevOps must adopt a continuous integration approach to ensure that automated tools, automated testing, and manual injection of security elements are included throughout the process of product development. Programmers need to adopt secure coding practices, security experts need to train programmers, and security auditors need to monitor code throughout development for proper security elements.

Baselining

Every development project needs a baseline. A baseline is a minimum level of function, response, and security that must be met in order for the project to proceed toward release. Any software product that does not meet or exceed the baseline is rejected and must return to development to be improved. A baseline is a form of quality control. Without quality control, neither the needs of the customer nor the reputation of the vendor are respected.

Immutable systems

An immutable system is a server or software product that, once configured and deployed, is never altered in place. Instead, when a new version is needed or a change is necessary, a revised version is crafted and the new system is then deployed to replace the old one. The concept of immutable systems is to prevent minor tweaks and changes to one system or another causing a complexity of configuration differences. In many organizations today, a single server is no longer sufficient to support a resource and its users, so numerous computers, often in a clustered arrangement, are deployed. Immutable systems ensure that each member of the server group is exactly the same, so when something needs to change, it is first developed and tested in a staging area, and then when finalized the new version fully replaces the previous version.

Infrastructure as code

Infrastructure as code is a change in how hardware management is perceived and handled. Instead of seeing hardware configuration as a manual, direct hands-on, one-on-one administration hassle, it is viewed as just another collection of elements to be managed in the same way that software and code are managed under DevOps. This alteration in hardware management approach has allowed many organizations to streamline infrastructure changes so that they occur more easily, more rapidly, more securely and safely, and more reliably than before. Infrastructure as code often requires the implementation of hardware management software, such as Puppet. Such solutions provide version control, code review, continuous integration, and code review to the portion of an IT infrastructure that was not able to be managed in this manner in the past.

Version control and change management

Change in a secure environment can introduce loopholes, overlaps, missing objects, and oversights that can lead to new vulnerabilities. The only way to maintain security in the face of change is to manage change systematically. Change management usually involves extensive planning, testing, logging, auditing, and monitoring of activities related to security controls and mechanisms. The records of changes to an environment are then used to identify agents of change, whether those agents are objects, subjects, programs, communication pathways, or the network itself.

The goal of change management is to ensure that no change leads to reduced or compromised security. Change management is also responsible for making it possible to roll back any change to a previous secured state. Change management can be implemented on any system, no matter what its level of security. Ultimately, change management improves the security of an environment by protecting implemented security from unintentional, tangential, or affected diminishments. Although an important goal of change management is to prevent unwanted reductions in security, its primary purpose is to make all changes subject to detailed documentation and auditing and thus able to be reviewed and scrutinized by management.

Change management should be used to oversee alterations to every aspect of a system, including hardware configuration and OS and application software. Change management should be included in design, development, testing, evaluation, implementation, distribution, evolution, growth, ongoing operation, and modification. It requires a detailed inventory of every component and configuration. It also requires the collection and maintenance of complete documentation for every system component, from hardware to software and from configuration settings to security features.

The change-control process of configuration, version control, or change management has several goals or requirements:

- Implement changes in a monitored and orderly manner. Changes are always controlled.
- A formalized testing process is included to verify that a change produces expected results.
- All changes can be reversed.

- Users are informed of changes before they occur to prevent loss of productivity.
- The effects of changes are systematically analyzed.
- The negative impact of changes on capabilities, functionality, and performance is minimized.

One example of a change-management process is a *parallel run*, which is a type of new system deployment testing where the new system and the old system are run in parallel. Each major or significant user process is performed on each system simultaneously to ensure that the new system supports all required business functionality that the old system supported or provided.

Change is the antithesis of security. In fact, change often results in reduced security. Therefore, security environments often implement a system of change management to minimize the negative impact of change on security. *Change documentation* is one aspect of a change-management system: it's the process of writing out the details of changes to be made to a system, a computer, software, a network, and so on before they're implemented. Then, the change documentation is transformed into a procedural document that is followed to the letter to implement the desired changes. After the changes are implemented, the system is tested to see whether security was negatively affected. If security has decreased, the change documentation can be used to guide the reversal of the changes to restore the system to a previous state in which stronger security was enforced.

Provisioning and deprovisioning

Provisioning is preallocation. When needing to deploy several new instances of a server to increase resource availability, the IT manager must provision hardware server resources to allocate to the new server instances. Provisioning is used to ensure that sufficient resources are available to support and maintain a system, software, or solution. Provisioning helps prevent the deployment of a new element without sufficient resources to support it.

Deprovisioning can be focused on two elements. It can focus on streamlining and fine-tuning resource allocation to existing systems for a more efficient distribution of resources. This can result in freeing sufficient resources to launch additional instances of a server. Deprovisioning can also focus on the release of resources from a server being decommissioned so that those resources return to the availability pool for use by other future servers.

Secure coding techniques

Secure coding concepts are those efforts designed to implement security into software as it's being developed. Security should be designed into the concept of a new solution, but programmers still need to code the security elements properly and avoid common pitfalls and mistakes while coding.

Proper error handling

When errors occur, the program should fall back to a secure state. This is generally known as *fail-secure design*. However, the programmer must code this into the application in order for a true fail-secure response to take place. This should include error and exception handling. When a process, a procedure, or an input results in or causes an error, the system should revert to a more secure state. This could include resetting back to a previous state of operation, rebooting back into a secured state, or recycling the connection state to revert to secured communications. Errors should also provide minimal information to visitors and users, especially outside/external visitors and users. All detailed error messages should be stored in an access-restricted log file for the programmers and administrators. Whenever an exception is encountered, it should be rejected and the fail-secure response should be triggered.

Proper input validation

Input validation is an aspect of defensive programming intended to ward off a wide range of input-focused attacks, such as buffer overflows and fuzzing. Input validation checks each and every input received before it's allowed to be processed. The check could be a length, a character type, a language type, a domain, or even a timing check to prevent unknown, unwanted, or unexpected content from making it to the core program.

Normalization

Normalization is a database programming and management technique used to reduce redundancy. The goal of normalization is to prevent redundant data, which is a waste of space and can also increase processing load. A normalized database is more efficient and can allow for faster data retrieval operations. Removing duplicate and redundant data ensures that sensitive data will exist in only one table or database (the original source), rather than being repeated within many others. This can reduce the difficulty of security sensitive data by allowing database security managers to implement access control over the original data source instead of having to attempt to lock down every duplicate copy.

Stored procedures

A stored procedure is a subroutine or software module that can be called on or accessed by applications interacting with a relational database management system (RDBMS). Stored procedures may be used for data validation during input, managing access control, assessing the logic of data, and more. Stored procedures can make some database applications more efficient, consistent, and secure.

Code signing

Code signing is the activity of crafting a digital signature of a software program in order to confirm that it was not changed and who it is from. See the Chapter 6 section "Digital signatures."

Encryption

Encryption should be used to protect data in storage and data in transit. Programmers should adopt trusted and reliable encryption systems into their applications. See Chapter 6 for the broad discussion of encryption and cryptography.

Obfuscation/camouflage

Obfuscation or camouflage is the coding practice of crafting code specifically to be difficult to decipher by other programmers. These techniques might be adopted in order to prevent unauthorized third parties from understanding proprietary solutions. These techniques can also be adopted by malicious programmers to hide the true intentions and purposes of software.

Code reuse/dead code

Code reuse is the inclusion of preexisting code in a new program. Code reuse can be a way to quicken the development process by adopting and reusing existing code. However, care should be taken not to violate copyright or intellectual property restrictions when reusing code. It is also important to fully understand the reused code to ensure that backdoors or other exploitable flaws are not introduced to the new product through the recycled code.

Dead code is any section of software that is executed but whose output or result is not used by any other process. Effectively the execution of dead code is a waste of time and resources. Programmers should strive to minimize and eliminate dead code from their products in order to improve efficiency and minimize the potential for exploitable errors or flaws. Dead code is sometimes used as part of obfuscation.

Server-side vs. client-side execution and validation

Server-side validation is suited for protecting a system against input submitted by a malicious user. Most client-side executions of scripts or mobile applets can be easily bypassed by a skilled web hacker. Thus, any client-side filtering is of little defense if the submission to the server bypasses those protections. A web hacker can edit JavaScript or HTML, modify forms, alter URLs, and much more. Thus, never assume any client-side filtering was effective—all input should be reassessed on the server side before processing. Server-side validation should include a check for input length, a filter for known scriptable or malicious content (such as SQL commands or script calls), and a filter for metacharacters (see Chapter 1, “Threats, Attacks, and Vulnerabilities.”).

Client-side validation is also important, but its focus is on providing better responses or feedback to the typical user. Client-side validation can be used to indicate whether input meets certain requirements, such as length, value, content, and so on. For example, if an email address is requested, a client-side validation check can confirm that it uses supported characters and is of the typical construction `username@FQDN`.

Although all the validation can take place on the server side, it is often a more complex process and introduces delays to the interaction. A combination of server-side and client-side validation allows more efficient interaction while maintaining reasonable security defenses.

Memory management

Programmers should include code in their software that focuses on proper memory management. Software should preallocate memory buffers but also limit the input sent to those buffers. Including input limit checks is part of secure coding practices, but it may be seen as busy work during the initial steps of software creation. Some programmers focus on getting new code to function with the intention of returning to the code in the future to improve security and efficiency. Unfortunately, if the functional coding efforts take longer than expected, it can result in the security revisions being minimized or skipped. Always be sure to use secure coding practices, such as proper memory management, to prevent a range of common software exploitations, such as buffer overflow attacks.

Use of third-party libraries and SDKs

Third-party software libraries and software development kits (SDKs) are often essential tools for a programmer. Using preexisting code can allow programmers to focus on their custom code and logic. SDKs provide guidance on software crafting as well as solutions, such as special APIs, subroutines, or stored procedures, which can simplify the creation of software for complex execution environments.

However, when you are using third-party software libraries, the precrafted code may include flaws, backdoors, or other exploitable issues that are unknown and yet undiscovered. Attempt to vet any third-party code before relying on it. Similarly, an SDK might not have security and efficiency as a top priority, so evaluate the features and capabilities provided via the SDK for compliance with your own programming and security standards.

Data exposure

When software does not adequately protect the data it processes, it may result in unauthorized data exposure. Programmers need to include authorization, authentication, and encryption schemes in their products in order to protect against data leakage, loss, and exposure.

Code quality and testing

No amount of network hardening, auditing, or user training can compensate for bad programming. Solid application security is essential to the long-term survival of any organization. Application security begins with secure coding and design, which is then maintained over the life of the software through testing and patching. Code quality needs to be assessed prior to execution. Software testing needs to be performed prior to distribution.

Before deploying a new application into the production environment, you should install it into a lab or pilot environment. Once testing is complete, the deployment procedure should include the crafting of an installation how-to, which must include not only the steps for deployment but also the baseline of initial configuration. This can be a written baseline or a template file that can be applied. The purpose of an application configuration baseline is to ensure compliance with policy and reduce human error. Baselines can be reapplied periodically or validated against changing work conditions as needed.

Static code analyzers

Static code analyzers review the raw source code of a product without executing the program. This debugging effort is designed to locate flaws in the software code before the program is run on a target or customer system. Static code analysis is often a first step in software quality and security testing.

Dynamic analysis (e.g., fuzzing)

Dynamic analysis is the testing and evaluation of software code while the program is executing. The executing code is then subjected to a range of inputs to evaluate its behavior and responses. One method of performing dynamic analysis is known as fuzzing.

Fuzzing is a software-testing technique that generates inputs for targeted programs. The goal of fuzz testing is to discover input sets that cause errors, failures, and crashes, or to discover other unknown defects in the targeted program. Basically, a fuzz-tester brute-force attack generates inputs within given parameters far in excess of what a normal, regular user or environment would ever be able to do. The information discovered by a fuzzing tool can be used to improve software as well as develop exploits for it.

Once a fuzz-testing tool discovers a constructed input that causes an abnormal behavior in the target application, the input and response are recorded into a log. The log of interesting inputs is reviewed by a security professional or a hacker. With the right skills and tools, the results of fuzzing can be transformed into a patch that fixes discovered defects or an exploit that takes advantage of them.

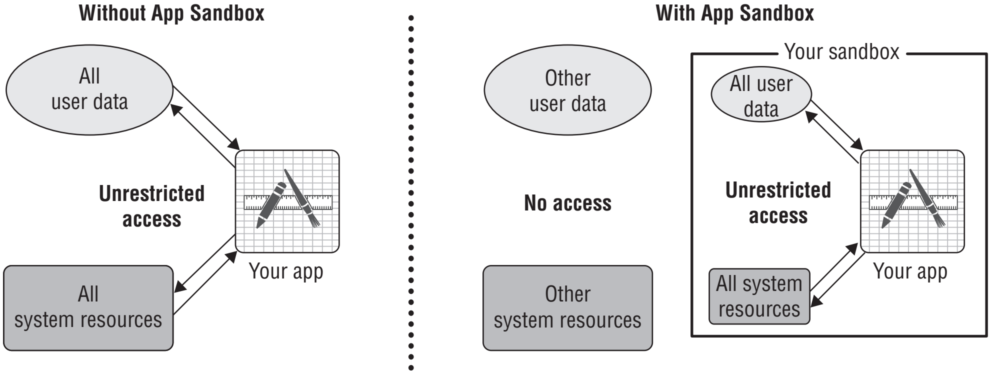
Stress testing

Stress testing is another variation of dynamic analysis in which a hardware or software product is subjected to various levels of workload in order to evaluate its ability to operate and function under stress. Stress testing can start with a modest level of traffic and then increase to abnormally high levels. The purpose of stress testing is to gain an understanding of how a product will perform, react, or fail in the various circumstances between normal conditions and DoS level traffic or load.

Sandboxing

A *sandbox* is a software implementation of a constrained execution space used to contain an application. Sandboxing (Figure 3.17) is often used to protect the overall computer from a new, unknown, untested application. The sandbox provides the contained application with direct or indirect access to sufficient system resources to execute, but not the ability to make changes to the surrounding environment or storage devices (beyond its own files). Sandboxing is commonly used for software testing and evaluating potential malware, and it is the basis for the concept of virtualization.

FIGURE 3.17 Application sandboxing



Model verification

Model verification is a part of the software development process that is often used to ensure that the crafted code remains in compliance with a development process, an architectural model, or design limitations. Model verification can also extend to ensuring that a software solution is able to achieve the desired real-world results by performing operational testing. Model verification can ensure that a product maintains compliance with security baseline requirements during development.

Compiled vs. runtime code

Most applications are written in a high-level language that is more similar to human language, such as English, than to the 1s and 0s that make up machine language. High-level languages are easier for people to learn and use in crafting new software solutions. However, high-level languages must ultimately be converted to machine language in order to execute the intended operations.

If the code is converted to machine language using a compiler crafting an output executable, then the language is described as compiled. The resulting executable file can be run at any time.

If the code remains in its original human-readable form and is converted into machine language only at the moment of execution, the language is a runtime compiled language. Some runtime languages will compile/convert the entire code at once into machine language for execution, whereas others will compile/convert only one line at a time (sometimes known as just-in-time execution or compilation).

Compiled code is harder for an attacker to inject malware into, but it is harder to detect such malware. Runtime code is easier for an attacker to inject malware into, but it is easier to detect such malware.

Exam Essentials

Understand SDLC. A development life-cycle model is a methodical ordering of the tasks of creating a new product or revising an existing one. A formal software development life-cycle (SDLC) model helps ensure a more reliable and stable product by establishing a standard process by which new ideas become actual software.

Know the waterfall model. The waterfall model consists of seven stages or steps. The original idea was that project development would proceed through these steps in order from first to last with the restriction that returning to an earlier phase was not allowed.

Understand Agile. The Agile model is based on adaptive development, where focusing on a working product and fulfilling customer needs is prioritized over rigid adherence to a process, use of specific tools, or detailed documentation. Agile focuses on an adaptive approach to development and supports early delivery and continuous improvement, along with flexible and prompt responses to changes.

Comprehend secure DevOps. DevOps, or development and operations, is a new IT movement in which many elements and functions of IT management are being integrated into a single automated solution. DevOps typically consists of IT development, operations, security, and quality assurance. Secure DevOps is a variant of DevOps that prioritizes security in the collection of tasks performed under this new umbrella concept.

Understand change management. The goal of change management is to ensure that change does not lead to reduced or compromised security. Change in a secure environment can introduce loopholes, overlaps, missing objects, and oversights that can lead to new vulnerabilities. The only way to maintain security in the face of change is to systematically manage change. This usually involves extensive planning, testing, logging, auditing, and monitoring of activities related to security controls and mechanisms.

Know provisioning and deprovisioning. Provisioning is preallocation. Provisioning is used to ensure that sufficient resources are available to support and maintain a system, software, or solution. Deprovisioning can focus on streamlining and fine-tuning resource allocation to existing systems for a more efficient distribution of resources. It can also focus on the release of resources from a server that is being decommissioned so that those resources return to the availability pool for use by other future servers.

Understand secure coding concepts. Secure coding concepts are those efforts designed to implement security into software as it's being developed. Security should be designed into the concept of a new solution, but programmers still need to code the security elements properly and avoid common pitfalls and mistakes while coding.

Comprehend error handling. When a process, a procedure, or an input causes an error, the system should revert to a more secure state. This could include resetting to a previous state of operation, rebooting back into a secured state, or recycling the connection state to revert to secured communications.

Understand input validation. Input validation checks each and every input received before it's allowed to be processed. The check could be a length, a character type, a language type, a domain, or even a timing check to prevent unknown, unwanted, or unexpected content from making it to the core program.

Know about normalization. Normalization is a database programming and management technique used to reduce redundancy. The goal of normalization is to prevent redundant data, which is a waste of space and can also increase processing load.

Understand stored procedures. A stored procedure is a subroutine or software module that can be called upon or accessed by applications interacting with an RDBMS.

Know code signing. Code signing is the activity of crafting a digital signature of a software program in order to confirm that it was not changed and who it is from.

Understand obfuscation and camouflage. Obfuscation or camouflage is the coding practice of crafting code specifically to be difficult for other programmers to decipher.

Comprehend code reuse. Code reuse is the inclusion of preexisting code in a new program. Code reuse can be a way to quicken the development process.

Understand dead code. Dead code is any section of software that is executed but the output or result of the execution is not used by any other process. Effectively the execution of dead code is a waste of time and resources.

Know server-side validation. Server-side validation is suited for protecting a system against input submitted by a malicious user. It should include a check for input length, a filter for known scriptable or malicious content (such as SQL commands or script calls), and a metacharacter filter.

Understand client-side validation. Client-side validation focuses on providing better responses or feedback to the typical user. It can be used to indicate whether input meets certain requirements, such as length, value, content, and so on.

Know memory management. Software should include proper memory management, such as preallocating memory buffers but also limiting the input sent to those buffers. Including input limit checks is part of secure coding practices.

Understand third-party libraries and SDKs. Third-party software libraries and software development kits (SDKs) are often essential tools for a programmer. Using preexisting code can allow programmers to focus on their custom code and logic.

Comprehend code quality and testing. Application security begins with secure coding and design, which is then maintained over the life of the software through testing and patching.

Understand static code analyzers. Static code analyzers review the raw source code of a product without the program being executed. This debugging effort focuses on locating flaws in the software code before the program is run on a target or customer system.

Know dynamic analysis. Dynamic analysis is the testing and evaluation of software code while the program is executing. The executing code is then subjected to a range of inputs to evaluate its behavior and responses.

Understand fuzzing. Fuzzing is a software-testing technique that generates inputs for targeted programs. The goal of fuzz-testing is to discover input sets that cause errors, failures, and crashes, or to discover other defects in the targeted program.

Know about stress testing. Stress testing is another variation of dynamic analysis in which a hardware or software product is subjected to various levels of workload in order to evaluate its ability to operate and function under stress.

Understand sandboxing. A sandbox is a software implementation of a constrained execution space used to contain an application. Sandboxing is often used to protect the overall computer from a new, unknown, untested application.

Comprehend model verification. Model verification is often part of software development processes; it is used to ensure that the crafted code remains in compliance with a development process, architectural model, or design limitations.

Understand compiled code. If the code is converted to machine language using a compiler crafting an output executable, then the language is a compiled language.

Know about runtime code. If the code remains in its original human-readable form and then gets converted into machine language only at the moment of execution, the language is a runtime compiled language.

3.7 Summarize cloud and virtualization concepts.

Virtualization technology is used to host one or more OSs in the memory of a single host computer. This mechanism allows practically any OS to operate on any hardware. It also lets multiple OSs work simultaneously on the same hardware. Cloud computing is often remote virtualization. Please review the earlier section "Virtualization" in this chapter.

Cloud computing and virtualization, especially when you are virtualizing in the cloud, have serious risks associated with them. Once sensitive, confidential, or proprietary data leaves the confines of the organization, it also leaves the protections imposed by the organizational security policy and resultant infrastructure. Cloud services and their personnel might not adhere to the same security standards as your organization. It is important to investigate the security of a cloud service before adopting it.

With the increased burden of industry regulations, such as the Sarbanes-Oxley Act of 2002 (SOX), Health Insurance Portability and Accountability Act (HIPAA), and Payment Card Industry Data Security Standards (PCI DSS), it is essential to ensure that a cloud service provides sufficient protections to maintain compliance. Additionally, cloud service providers may not maintain your data in close proximity to your primary physical location. In fact, they may distribute your data across numerous locations, some of which may reside outside your country of origin. It may be necessary to add to a cloud service contract a limitation to house your data only within specific logical and geographic boundaries.

It is important to investigate the encryption solutions employed by a cloud service. Do you send your data to them pre-encrypted, or is it encrypted only after reaching the cloud? Where are the encryption keys stored? Is there segregation between your data and that belonging to other cloud users? An encryption mistake can reveal your secrets to the world or render your information unrecoverable.

What is the method and speed of recovery or restoration from the cloud? If you have system failures locally, how do you get your environment back to normal? Also consider whether the cloud service has its own disaster-recovery solution. If it experiences a disaster, what is its plan to recover and restore services and access to your cloud resources?

Other issues include the difficulty with which investigations can be conducted, concerns over data destruction, and what happens if the current cloud-computing service goes out of business or is acquired by another organization.

Snapshots are backups of virtual machines. They offer a quick means to recover from errors or poor updates. It's often easier and faster to make backups of entire virtual systems rather than the equivalent native hardware installed system.

Virtualization doesn't lessen the security management requirements of an OS. Thus, patch management is still essential. Patching or updating virtualized OSs is the same process as for a traditionally hardware installed OS. Also, don't forget that you need to keep the virtualization host updated as well.

When you're using virtualized systems, it's important to protect the stability of the host. This usually means avoiding using the host for any purpose other than hosting the virtualized elements. If host availability is compromised, the availability and stability of the virtual systems are also compromised.

Elasticity refers to the flexibility of virtualization and cloud solutions to expand or contract based on need. In relation to virtualization, *host elasticity* means additional hardware hosts can be booted when needed and then used to distribute the workload of the virtualized services over the newly available capacity. As the workload becomes smaller, you can pull virtualized services off unneeded hardware so it can be shut down to conserve electricity and reduce heat.

Virtualized systems should be security tested. The virtualized OSs can be tested in the same manner as hardware installed OSs, such as with vulnerability assessment and penetration testing. However, the virtualization product may introduce additional and unique security concerns, so the testing process needs to be adapted to include those idiosyncrasies.

Hypervisor

The *hypervisor*, also known as the virtual machine monitor (VMM), is the component of virtualization that creates, manages, and operates the virtual machines. The computer running the hypervisor is known as the host OS, and the OSs running within a hypervisor-supported virtual machine are known as guest OSs.

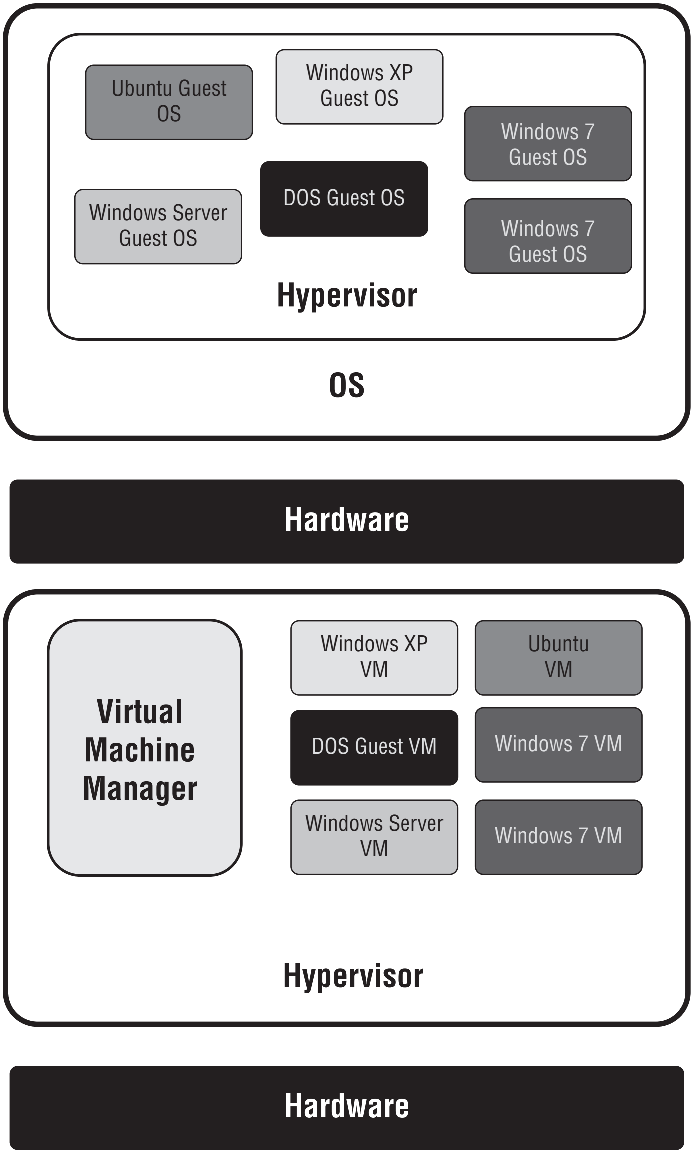
Type I

A type I hypervisor (Figure 3.18, top) is a native or bare-metal hypervisor. In this configuration, there is no host OS; instead, the hypervisor installs directly onto the hardware where the host OS would normally reside. Type 1 hypervisors are often used to support server virtualization. This allows for maximization of the hardware resources while eliminating any risks or resource reduction caused by a host OS.

Type II

A type II hypervisor (Figure 3.18, bottom) is a hosted hypervisor. In this configuration, a standard regular OS is present on the hardware, and then the hypervisor is installed as another software application. Type II hypervisors are often used in relation to desktop deployments, where the guest OSs offer safe sandbox areas to test new code, allow the execution of legacy applications, support apps from alternate OSs, and provide the user with access to the capabilities of a host OS.

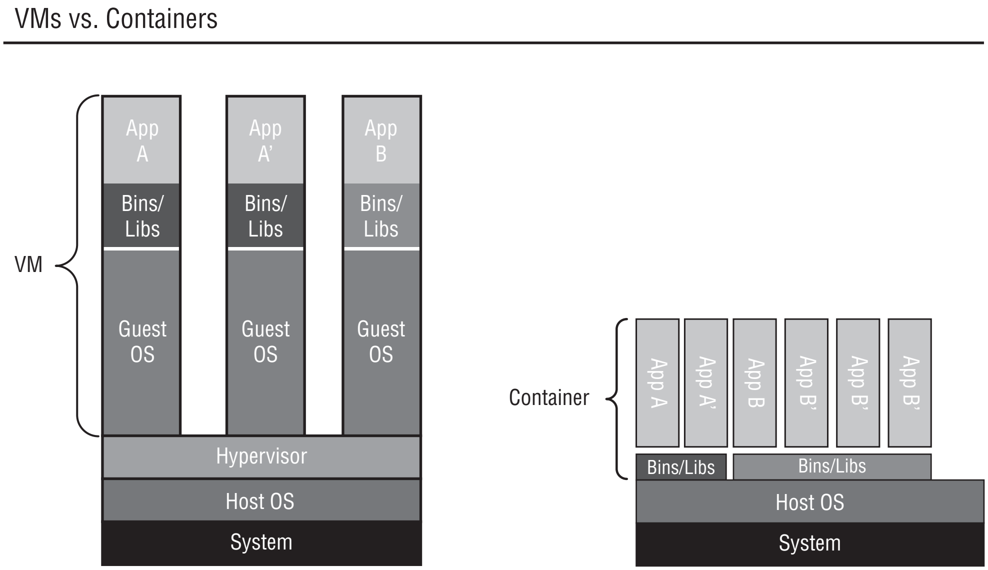
FIGURE 3.18 Hosted vs. bare-metal hypervisor



Application cells/containers

Another variation of virtualization is that focusing on applications instead of entire operating systems. Application cells or application containers (Figure 3.19) are used to virtualize software so they can be ported to almost any OS.

FIGURE 3.19 Application containers vs. a hypervisor



VM sprawl avoidance

VM sprawl occurs when an organization deploys numerous virtual machines without an overarching IT management or security plan in place. Although VMs are easy to create and clone, they have the same licensing and security management requirements as a metal installed OS. Uncontrolled VM creation can quickly lead to a situation where manual oversight cannot keep up with system demand. To prevent or avoid VM sprawl, a policy for developing and deploying VMs must be established and enforced. This should include establishing a library of initial or foundation VM images that are to be used to develop and deploy new services.

VM escape protection

VM escaping occurs when software within a guest OS is able to breach the isolation protection provided by the hypervisor in order to violate the container of other guest OSs or to infiltrate a host OS. VM escaping can be a serious problem, but steps can be implemented to minimize the risk. First, keep highly sensitive systems and data on separate physical machines. An organization should already be concerned about overconsolidation resulting in a single point of failure, so running numerous hardware servers so each supports a handful of guest OSs helps with this risk. Keeping enough physical servers on hand to maintain physical isolation between highly sensitive guest OSs will further protect against VM escaping. Second, keep all hypervisor software current with vendor-released patches. Third, monitor attack, exposure, and abuse indexes for new threats to your environment.

Cloud storage

Cloud storage is the idea of using storage capacity provided by a cloud vendor as a means to host data files for an organization. Cloud storage can be used as a form of backup or support for online data services. Cloud storage may be cost effective, but it is not always high speed or low latency. Most do not yet consider cloud storage as a replacement for physical backup media solutions, but rather as a supplement for organizational data protection.

Cloud deployment models

Cloud computing is a popular term that refers to performing processing and storage elsewhere, over a network connection, rather than locally. Cloud computing is often thought of as Internet-based computing. Ultimately, processing and storage occur on computers somewhere, but the distinction is that the local operator no longer needs to have that capacity or capability locally. Thus more users can use cloud resources on an on-demand basis. From the end users' perspective, all the work of computing is performed "in the cloud," so the complexity is isolated from them.

Cloud computing is a natural extension and evolution of virtualization, the Internet, distributed architecture, and the need for ubiquitous access to data and resources. However, it does have some security and IT policy issues: privacy concerns, regulation compliance difficulties, use of open-/closed-source solutions, adoption of open standards, and whether cloud-based data is actually secured (or even securable). The primary security concerns related to cloud computing are determining and clarifying what security responsibilities belong to the cloud provider and which are the customer's obligation—this should be detailed in the SLA/contract.

SaaS

Software as a Service (SaaS) is a derivative of Platform as a Service. It provides on-demand online access to specific software applications or suites without the need for local installation (and with no local hardware and OS requirements, in many cases). Software as a Service can be implemented as a subscription service, a pay-as-you-go service, or a free service.

PaaS

Platform as a Service (PaaS) is the concept of providing a computing platform and software solution stack to a virtual or cloud-based service. Essentially, it involves paying for a service that provides all the aspects of a platform (that is, an OS and a complete solution package). A PaaS solution grants the customer the ability to run custom code of their choosing without needing to manage the environment. The primary attraction of Platform as a Service is that you don't need to purchase and maintain high-end hardware and software locally.

IaaS

Infrastructure as a Service (IaaS) takes the platform as a service model another step forward and provides not just on-demand operating solutions but complete outsourcing options. These can include utility or metered computing services, administrative task automation, dynamic scaling, virtualization services, policy implementation and management services, and managed/filtered Internet connectivity. Ultimately, Infrastructure as a Service allows an enterprise to quickly scale up new software- or data-based services/solutions through cloud systems without having to install massive hardware locally.

Private

A private cloud is a cloud service that is within a corporate network and isolated from the Internet. The private cloud is for internal use only.

A virtual private cloud is a service offered by a public cloud provider that provides an isolated subsection of a public or external cloud for exclusive use by an organization internally. In other words, an organization outsources its private cloud to an external provider.

Public

A public cloud is a cloud service that is accessible to the general public, typically over an Internet connection. Public cloud services may require some form of subscription or pay per use or may be offered for free. Although an organization's or individual's data is usually kept separated and isolated from other customers' data in a public cloud, the overall purpose or use of the cloud is the same for all customers.

Hybrid

A hybrid cloud is a mixture of private and public cloud components. For example, an organization could host a private cloud for exclusive internal use but distribute some resources onto a public cloud for the public, business partners, customers, the external sales force, and so on.

Community

A community cloud is a cloud environment maintained, used, and paid for by a group of users or organizations for their shared benefit, such as collaboration and data exchange. This may allow for some cost savings compared to accessing private or public clouds independently.

On-premise vs. hosted vs. cloud

An on-premise solution is the traditional deployment concept in which an organization owns the hardware, licenses the software, and operates and maintains the systems on its own, usually within their own building.

A cloud solution is a deployment concept where an organization contracts with a third-party cloud provider. The cloud provider owns, operates, and maintains the hardware and software. The organization pays a monthly fee (often based on a per-user multiplier) to use the cloud solution.

A hosted solution is a deployment concept where the organization must license software and then operates and maintains the software. The hosting provider owns, operates, and maintains the hardware that supports the organization's software.

On-premise solutions do not have ongoing monthly costs, but may be more costly because of initial up-front costs of obtaining hardware and licensing. On-premise solutions offer full customization, provide local control over security, do not require Internet connectivity, and provide local control over updates and changes. However, they also require significant administrative involvement for updates and changes, require local backup and management, and are more challenging to scale.

Cloud solutions often have lower up-front costs, lower maintenance costs, vendor-maintained security, and scalable resources, and they usually have high levels of uptime and availability from anywhere (over the Internet). However, cloud solutions do not offer customer control over OS and software, such as updates and configuration changes; offer minimal customization; and are often inaccessible without Internet connectivity. In addition, the security policies of the cloud provider might not match those of the organization.

VDI/VDE

See the Chapter 2 section "VDI" for a description of the virtual desktop infrastructure (VDI) model. Virtual desktop environment (VDE) is an alternate term for VDI.

Cloud access security broker

A *cloud access security broker (CASB)* is a security policy enforcement solution that may be installed on-premise or may be cloud-based. The goal of a CASB is to enforce proper security measures and ensure that they are implemented between a cloud solution and a customer organization.

Security as a Service

Security as a Service (SECaaS) is a cloud provider concept in which security is provided to an organization through or by an online entity. The purpose of an SECaaS solution is to reduce the cost and overhead of implementing and managing security locally. SECaaS often implements software-only security components that do not need dedicated on-premise hardware. SECaaS security components can include a wide range of security products, including authentication, authorization, auditing/accounting, antimalware, intrusion detection, penetration testing, and security event management.

Exam Essentials

Understand the risks associated with cloud computing and virtualization. Cloud computing and virtualization, especially when combined, have serious risks associated with them. Once sensitive, confidential, or proprietary data leaves the confines of the organization, it also leaves the protections imposed by the organizational security policy and resultant infrastructure. Cloud services and their personnel might not adhere to the same security standards as your organization.

Comprehend cloud computing. Cloud computing involves performing processing and storage elsewhere, over a network connection, rather than locally. Cloud computing is often thought of as Internet-based computing.

Understand hypervisors. The hypervisor, also known as the virtual machine monitor (VMM), is the component of virtualization that creates, manages, and operates the virtual machines.

Know about the type I hypervisor. A type I hypervisor is a native or bare-metal hypervisor. In this configuration, there is no host OS; instead, the hypervisor installs directly onto the hardware where the host OS would normally reside.

Know about the type II hypervisor. A type II hypervisor is a hosted hypervisor. In this configuration, a standard regular OS is present on the hardware, and the hypervisor is then installed as another software application.

Understand application cells/containers. Application cells or application containers are used to virtualize software so they can be ported to almost any OS.

Comprehend VM sprawl avoidance. VM sprawl occurs when an organization deploys numerous virtual machines without an overarching IT management or security plan in place. To prevent or avoid VM sprawl, a policy must be established and enforced regarding the procedure for developing and deploying VMs.

Understand VM escaping. VM escaping occurs when software within a guest OS is able to breach the isolation protection provided by the hypervisor in order to violate the container of other guest OSs or to infiltrate a host OS.

Know about cloud storage. Cloud storage is the idea of using storage capacity provided by a cloud vendor as a means to host data files for an organization. Cloud storage can be used as form of backup or support for online data services.

Understand cloud deployment models. Cloud deployment models include SaaS, PaaS, IaaS, private, public, hybrid, and community.

Define CASB. A cloud access security broker (CASB) is a security policy enforcement solution that may be installed on-premise, or may be cloud based.

Understand SECaaS. Security as a Service (SECaaS) is a cloud provider concept in which security is provided to an organization through or by an online entity.

3.8 Explain how resiliency and automation strategies reduce risk.

Risk reduction, mitigation, and even elimination should be a core strategy for every organization. Security management consists of the efforts to establish, administer, and maintain security throughout the organization. Many elements of security management focus on establishing resiliency as well as automation in order to reduce risk, improve uptime, and minimize expense.

Automation/scripting

Automation is the control of systems on a regular scheduled, periodic, or triggered basis that does not require manual hands-on interaction. Automation is often critical to a resilient security infrastructure. Automation includes concepts such as scheduled backups, archiving of log files, blocking of failed access attempts, and blocking communications based on invalid content in initial packets or due to traffic seeming like a port scan. Automation can also be implemented using scripting. Scripting is the crafting of a file of individual lines of commands that are executed one after another. Scripts can be set to launch on a schedule or based on a triggering event.

Automated courses of action

Automated courses of action ensure that a specific series of steps or activities are performed in the correct order each and every time. This helps ensure consistency of results, which in turn establishes consistent security.

Continuous monitoring

In order for security monitoring to be effective, it must be continuous in several ways. First, it must always be running and active. There should be no intentional time frame when security monitoring isn't functioning. If security monitoring goes offline, all user activity should cease and administrators should be notified.

Second, security monitoring should be continuous across all user accounts, not just end users. Every single person has responsibilities to the organization to maintain its security. Likewise, everyone needs to abide by their assigned job-specific responsibilities and privileges. Any attempts to exceed or violate those limitations should be detected and dealt with.

Third, security monitoring should be continuous across the entire IT infrastructure. On every device possible, recording of system events and user activities should be taking place.

Fourth, security monitoring should be continuous for each user from the moment of attempted logon until the completion of a successful logoff or disconnect. At no time should the user expect to be able to perform tasks without security monitoring taking place.

Configuration validation

Automation is only effective if accurate. Repeating execution of a flawed program may leave the environment with reduced security rather than improved or maintained security. All systems need to have a defined baseline of configuration that is clearly documented. The configuration documentation should be used to validate all in-production systems on a regular basis. Only when systems are in proper compliance with a configuration baseline is security likely to be resilient; baseline compliance also supports the results of automated processes.

Templates

A template is a preestablished starting point. A template can be crafted for a plethora of concerns in an environment, including a security policy, a procedure, a contract, a submission form, a system image, a software configuration, and a firewall rule set. Starting security documentation, configuration, or management with a template is likely to produce more consistent and reliable results.

Master image

A master image or gold master is a crafted setup and configuration of a software product or an entire computer system. A master image is created just after the target system has been manually installed, patched, and configured. A master image is employed to quickly roll out new versions of a system. For example, when deploying 100 new workstations, you can install the master image of the preferred workstation software deployment configuration to quickly bring the new devices into compliance with production needs and security requirements.

Non-persistence

A nonpersistent system is a computer system that does not allow, support, or retain changes. Thus, between uses and/or reboots, the operating environment and installed software are exactly the same. A persistent system is one where changes are possible. Changes may be performed by authorized users, administrators, automated processes, or malware. To reduce the risk of change, various protection and recovery measures may need to be established.

Snapshots

A snapshot is a copy of the live current operating environment. Snapshots are mostly known relative to virtual machines and guest OSs. However, the term can be loosely employed to refer to any systemwide backup that can be restored to a previous state or condition of configuration and operation. A VM snapshot might only take a few minutes to create and restore, while a hard drive–based snapshot or cloning may take hours to create and restore.

Revert to known state

Revert to known state is a type of backup or recovery process. Many databases support a known state reversion in order to return to a state of data before edits or changes were implemented. Some systems will automatically create a copy of a known state in order to provide a rollback option, whereas others may require a manual creation of the rollback point. An example of a revert-to-known-state system is the restore point system of Windows. Whenever a patch or software product is installed, Windows can create a restore point that can be used to return the system to a previous configuration state.

Rollback to known configuration

Rollback to known configuration is a concept similar to that of reverting to a known state, but the difference is that a state retention may address a larger portion of the environment than just configuration. A known configuration is just a collection of settings, not likely to include any software elements, such as code present before a patch was applied. A rollback to known configuration is useful after a setting change that had undesired consequences, but not after installing a new version of a software product (for that use revert to known state, snapshot, or backup). One example of a rollback to known configuration is the Last Known Good Configuration (LKGC) found in Windows. Each time a user successfully logs into a Windows system, a copy of the registry is made at that moment and stored in the LKGC container. If the system is altered in such a way that the operating environment is unusable, then upon the next reboot an advanced boot option is to restore the LKGC. However, this option is available only once; if the user logs in while the system is still malfunctioning, the current configuration is stored in the LKGC container.

Live boot media

Live boot media is a portable storage device that can be used to boot a computer. Live boot media contains a read-to-run or portable version of an operating system. Live boot media may include CDs, DVDs, flash memory cards, and USB drives. Live boot media can be used as a portable OS when the local existing OS is not to be trusted (such as the computer sitting in a library or hotel lobby). Live boot media can also be used as a recovery and repair strategy to gain access to tools and utilities to operate on a target system without the system's OS running.

Elasticity

Elasticity is the ability of a system to adapt to workload changes by allocating or provisioning resources in an automatic responsive manner. Elasticity is a common feature of cloud computing, where additional system resources or even additional hardware resources can be provisioned to a server when demand for its services increases.

Scalability

Scalability is the ability for a system to handle an ever-increasing level or load of work. It can also be the potential for a system to be expanded to accommodate future growth. Some amount of additional capacity can be implemented into a system so it can take advantage of the dormant resources automatically as need demands. A cloud system can further automate scalability by enabling servers to auto-clone across to other virtualization hosts as demand requires.

Distributive allocation

Distributive or distributed allocation is the concept of provisioning resources across multiple servers or services as needed, rather than preallocation or concentrating resources based exclusively on physical system location. This is a form of load balancing but with a focus on the supporting resources rather than the traffic or request load.

Redundancy

This concept applies to various aspects of operational security, including business continuity, backups, and avoiding single points of failure as a means to protect availability.

Redundancy is the implementation of secondary or alternate solutions. Commonly, redundancy refers to having alternate means to perform work tasks or accomplish IT functions. Redundancy helps reduce single points of failure and improves fault tolerance. When there are multiple pathways, copies, devices, and so on, there is reduced likelihood of downtime when something fails.

When backup systems or redundant servers exist, there needs to be a means by which you can switch over to the backup in the event the primary system is compromised or fails. *Rollover*, or *failover*, means redirecting workload or traffic to a backup system when the primary system fails. Rollover can be automatic or manual. Manual rollover, also known as *cold rollover*, requires an administrator to perform some change in software or hardware configuration to switch the traffic load over from the down primary to a secondary server. With automatic rollover, also known as *hot rollover*, the switch from primary to secondary system is performed automatically as soon as a problem is encountered. *Fail-secure*, *fail-safe*, and *fail-soft* are terms related to these issues. A system that is fail-secure is able to resort to a secure state when an error or security violation is encountered (also known as *fail-closed*). Fail-safe is a similar feature, but human safety is protected in the event of system failure. However, these two terms are often used interchangeably in logical or technical context to mean a system that is secure after a failure. Fail-soft describes a refinement of the fail-secure capability: only the portion of a system that encountered or experienced the failure or security breach is disabled or secured, whereas the rest of the system continues to function normally.

The insecure inverse of these is the fail-open response. With a fail-open result, all defenses or preventions are disabled or retracted. Thus, a door defaults to being unlocked or even wide open, and electric security defaults to open, unlimited access.

Fault tolerance

Fault tolerance is the ability of a system to handle or respond to failure smoothly. This can include software, hardware, or power failure.

Any element in your IT infrastructure, component in your physical environment, or person on your staff can be a single point of failure. A single point of failure is any element—such as a device, service, protocol, or communication link—that would cause total or significant downtime if compromised, violated, or destroyed, affecting the ability of members of your organization to perform essential work tasks. To avoid single points of failure, you should design your networks and your physical environment with redundancy and backups by doing such things as deploying dual-network backbones. By using systems, devices, and solutions with fault-tolerant capabilities, you improve resistance to single-point-of-failure vulnerabilities. Taking steps to establish a way to provide alternate processing, failover capabilities, and quick recovery also helps avoid single points of failure.

Another type of redundancy related to servers is clustering. *Clustering* means deploying two or more duplicate servers in such a way as to share the workload of a mission-critical application. Users see the clustered systems as a single entity. A cluster controller manages traffic to and among the clustered systems to balance the workload across all clustered servers. As changes occur on one of the clustered systems, they are immediately duplicated to all other cluster partners.

The use of *redundant servers* is another example of avoiding single points of failure. A redundant server is a mirror or duplicate of a primary server that receives all data changes immediately after they are made on the primary server. In the event of a failure of the primary server, the secondary or redundant server can immediately take over and replace the primary server in providing services to the network.

This switchover system can be either hot or cold. A *hot switchover* or *hot failover* is an automatic system that can often perform the task nearly instantaneously. A *cold switchover* or *cold failover* is a manual system that requires an administrator to perform the manual task of switching from the primary to the secondary system, and thus it often involves noticeable downtime.

Redundant servers can be located in the same server vault as the primary or can be located offsite. Offsite positioning of the redundant server offers a greater amount of security so that whatever disaster damaged the primary server is unlikely to be able to damage the secondary, offsite server. However, offsite redundant servers are more expensive due to the cost of housing them, as well as real-time communication links needed to support the mirroring operations.

High availability

Availability is the assurance of sufficient bandwidth and timely access to resources. *High availability* means the availability of a system has been secured to offer very reliable assurance that the system will be online, active, and able to respond to requests in a timely manner, and that there will be sufficient bandwidth to accomplish requested tasks in the time required. Both of these concerns are central to maintaining continuity of operations. Availability is often measured in terms of the nines (Table 3.2)—a percentage of availability within a given time frame, such as a year, month, week, or day. Many organizations strive to achieve five or size nines of availability.

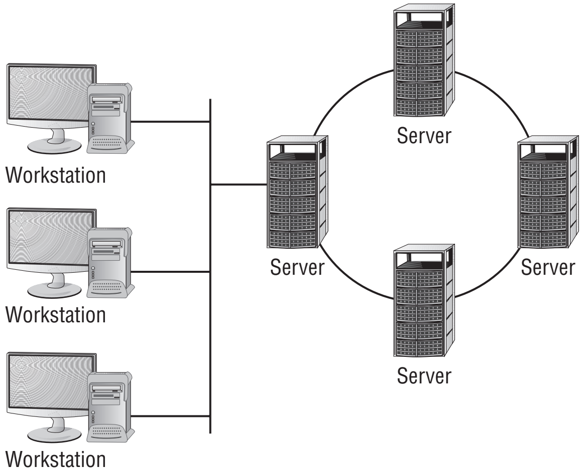
TABLE 3.2 Availability percentages and downtimes

Availability %	Downtime per year	Downtime per month	Downtime per week	Downtime per day
90% ("one nine")	36.5 days	72 hours	16.8 hours	2.4 hours
99% ("two nines")	3.65 days	7.20 hours	1.68 hours	14.4 minutes
99.9% ("three nines")	8.76 hours	43.8 minutes	10.1 minutes	1.44 minutes
99.99% ("four nines")	52.56 minutes	4.38 minutes	1.01 minutes	8.64 seconds
99.999% ("five nines")	5.26 minutes	25.9 seconds	6.05 seconds	864.3 milliseconds
99.9999% ("six nines")	31.5 seconds	2.59 seconds	604.8 milliseconds	86.4 milliseconds

High availability is a form of fault tolerance—or, rather, a benefit of providing reliable fault tolerance. *Fault tolerance* is the ability of a network, system, or computer to withstand a certain level of failures, faults, or problems and continue to provide reliable service. Fault tolerance is also a means of avoiding single points of failure. As mentioned earlier, a single point of failure is any system, software, or device that is mission-critical to the entire environment. If that one element fails, then the entire environment fails. Your environments should be designed with redundancy so that there are no single points of failure. Such a redundant design is fault tolerant.

Another example of a high-availability solution is server clustering (see Figure 3.20). *Server clustering* is a technology that connects several duplicate systems together so they act cooperatively. If one system in a cluster fails, the other systems take over its workload. From a user's perspective, the cluster is a single entity with a single resource access name.

FIGURE 3.20 Server clustering



Maintaining an onsite stash of spare parts can reduce downtime. Having an in-house supply of critical parts, devices, media, and so on enables fast repair and function restoration. A replacement part can then be ordered from the vendor and returned to the onsite spare-parts storage. Unexpected downtime due to hardware failure is a common cause of loss of availability. Planning for faster repairs improves uptime and eliminates lengthy downtimes caused by delayed shipping from vendors.

To avoid single points of failure completely, every communication pathway should be redundant. Thus, every link from the LAN to a carrier network or ISP should be duplicated. This can be accomplished by leasing two lines from the same ISP (which is the most basic form of redundant connection) or from different ISPs. The use of redundant ISPs reduces the likelihood that a failure at a single ISP will cause your organization significant connectivity downtime. However, the best redundant ISP configuration requires the two (or more) selected ISPs to use distinct Internet or network backbones.

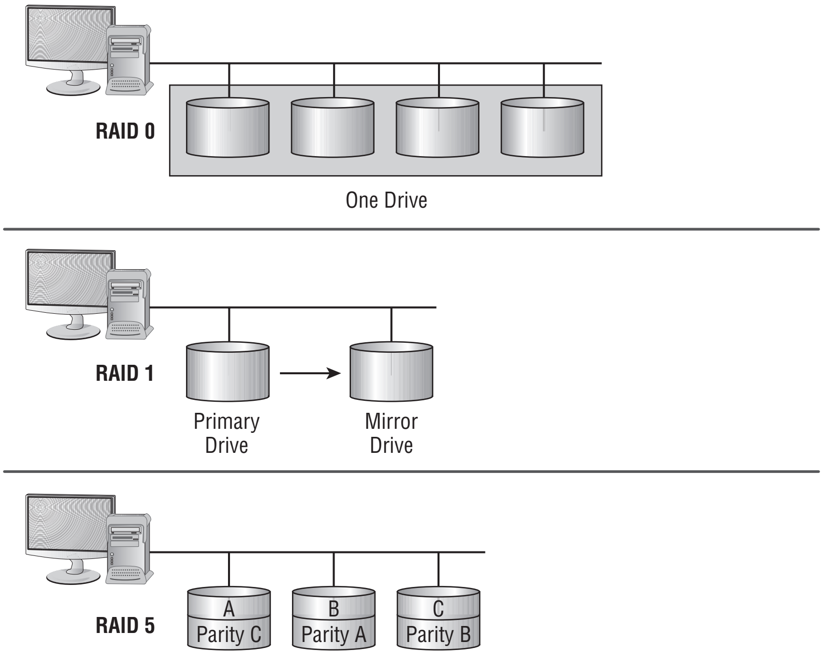
Power is an essential utility for any organization, but especially those dependent on their IT infrastructure. In addition to basic elements such as power conditioners and UPS devices, many organizations opt for an onsite backup generator to provide power during complete blackouts. A variety of backup generators are available, in terms of both size and fuel.

An *uninterruptible power supply (UPS)* is an essential element of any computing environment. A UPS provides several important services and features. First, a UPS is a power conditioner to ensure that only clean, pure, nonfluctuating power is fed to computer equipment. Second, in the event of a loss of power, the internal battery can provide power for a short period of time. The larger the battery, the longer the UPS can provide power. Third, when the battery reaches the end of its charge, it can signal the computer system to initiate a graceful shutdown in order to prevent data loss.

RAID

One example of a high-availability solution is a *redundant array of independent disks (RAID)*. A RAID solution employs multiple hard drives in a single storage volume, as illustrated in Figure 3.21. RAID 0 provides performance improvement but not fault tolerance known as *striping*, it uses multiple drives as a single volume. RAID 1 provides *mirroring*, meaning the data written to one drive is exactly duplicated to a second drive in real time. RAID 5 provides *striping with parity*: three or more drives are used in unison, and one drive's worth of space is consumed with parity information. The parity information is stored across all drives. If any single drive of a RAID 5 volume fails, the parity information is used to rebuild the contents of the lost drive on the fly. A new drive can replace the failed drive, and the RAID 5 system rebuilds the contents of the lost drive onto the replacement drive. RAID 5 can only support the failure of one disk drive.

FIGURE 3.21 Examples of RAID implementations



Exam Essentials

Understand automation and scripting. Automation is the control of systems on a regular scheduled, periodic, or triggered basis that does not require manual hands-on interaction. Automation is often critical to a resilient security infrastructure. Scripting is the crafting of a file of individual lines of commands that are executed one after another. Scripts can be set to launch on a schedule or based on a triggering event.

Know about master images. A master image is a crafted setup and configuration of a software product or an entire computer system. A master image is created just after the target system has been manually installed, patched, and configured.

Understand nonpersistence. A nonpersistent system is one where changes are possible. Changes may be performed by authorized users, administrators, automated processes, or malware. Due to the risk of change, various protection and recovery measures may need to be established.

Comprehend snapshots. A snapshot is a copy of the live current operating environment.

Understand revert to known state. Revert to known state is a type of backup or recovery process. Many databases support a known state reversion in order to return back to a state of data before edits or changes were implemented.

Know about roll back to known configuration. Roll back to known configuration is a concept similar to that of revert to known state, but a known configuration is just a collection of settings, not likely to include any software elements, such as code present before a patch was applied.

Understand live boot media. Live boot media is a portable storage device that can be used to boot a computer. Live boot media contains a read-to-run or portable version of an operating system.

Comprehend elasticity. Elasticity is the ability of a system to adapt to workload changes by allocating or provisioning resources in an automatic responsive manner.

Understand scalability. Scalability is the ability of a system to handle an ever-increasing level or load of work. It can also be the potential for a system to be expanded to handle or accommodate future growth.

Know about distributive allocation. Distributive allocation or distributed allocation is the concept of provisioning resources across multiple servers or services as needed, rather than using preallocation or concentrating resources based exclusively on physical system location.

Understand redundancy. Redundancy is the implementation of secondary or alternate solutions. Commonly, redundancy refers to having alternate means to perform work tasks or accomplish IT functions. Redundancy helps reduce single points of failure and improves fault tolerance.

Comprehend fault tolerance. Fault tolerance is the ability of a network, system, or computer to withstand a certain level of failures, faults, or problems and continue to provide reliable service. Fault tolerance is also a form of avoiding single points of failure. A single point of failure is any system, software, or device that is mission-critical to the entire environment.

Understand high availability. High availability means the availability of a system has been secured to offer very reliable assurance that the system will be online, active, and able to respond to requests in a timely manner, and that there will be sufficient bandwidth to accomplish requested tasks in the time required. RAID is a high-availability solution.

Know about the continuity of operations/high availability. Availability is the assurance of sufficient bandwidth and timely access to resources. High availability means the availability of a system has been secured to offer very reliable assurance that the system will be online, active, and able to respond to requests in a timely manner, and that there will be sufficient bandwidth to accomplish requested tasks in the time required. Both of these concerns are central to maintaining continuity of operations.

Understand RAID. One example of a high-availability solution is a redundant array of independent disks (RAID). A RAID solution employs multiple hard drives in a single storage volume with some level of drive loss protection (with the exception of RAID 0).

3.9 Explain the importance of physical security controls.

Without physical security, there is no security. No amount or extent of logical and technical security controls can compensate for lax physical security protection. Thus, physical security controls need to be assessed and implemented in the same manner as security controls for the IT infrastructure.

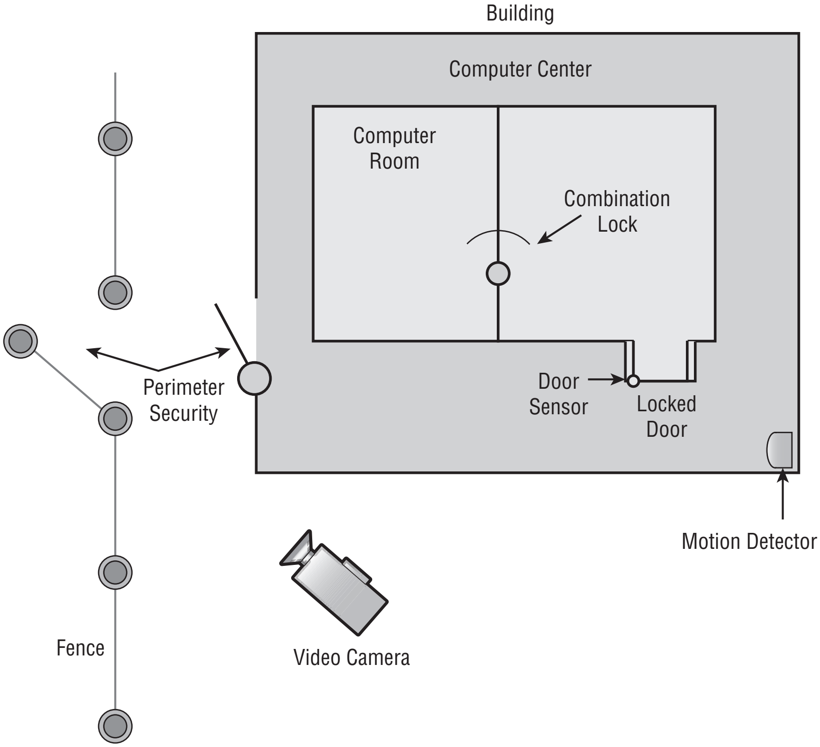
Physical security is an area that is often overlooked when security for an environment is being designed. As you prepare for the Security+ exam, don't overlook the aspects and elements of physical security. As a security professional, you need to reduce overall opportunities for intrusions or physical security violations. This can be accomplished using various mechanisms, including prevention, deterrence, and detection.

To ensure proper physical security, you should design the layout of your physical environment with security in mind. This means you should place all equipment in locations that can be secured, and control and monitor access or entrance into those locations. Good physical security access control also recognizes that some computers and network devices are more important or mission-critical than others and therefore require greater physical security protection.

Mission-critical servers and devices should be placed in dedicated equipment rooms that are secured from all possible entrance and intrusion (see Figure 3.22). These rooms shouldn't have windows or floor-to-roof walls (rather than short walls that end at a drop ceiling). Equipment rooms should be locked at all times, and only authorized personnel should be granted entrance. The rooms should be monitored, and all access should be logged and audited.

Physical barriers are erected to control access to a location. Some of the most basic forms of physical barriers are walls and fences. Fences are used to designate the borders of a geographic area where entrance is restricted; a high fence, the presence of barbed wire, or electrified fencing all provide greater boundary protection. Walls provide protection as well, preventing entry except at designated points such as doors and windows. The stronger the wall, the more security it provides. And the greater the number of walls between the untrusted outside and the valuable assets located inside, the greater the level of physical security.

FIGURE 3.22 An example of a multilayered physical security environment



Lighting

Lighting is a commonly used form of perimeter security control. The primary purpose of lighting is to discourage casual intruders, trespassers, prowlers, or would-be thieves who would rather perform their misdeeds in the dark, such as vandalism, theft, and loitering. However, lighting is not a strong deterrent. It should not be used as the primary or sole protection mechanism except in areas with a low threat level.

Lighting should be combined with guards, dogs, CCTV, or some other form of intrusion detection or surveillance mechanism. Lighting must not cause a nuisance or problem for nearby residents, roads, railways, airports, and so on. It should also never cause glare or a reflective distraction to guards, dogs, and monitoring equipment, which could otherwise aid attackers during break-in attempts.

Signs

Signs can be used to declare areas off limits to those who are not authorized, indicate that security cameras are in use, and disclose safety warnings. Signs are useful in deterring minor criminal activity, establishing a basis for recording events, and guiding people into compliance or adherence with rules or safety precautions.

Fencing/gate/cage

A *fence* is a perimeter-defining device. Fencing protects against casual trespassing and clearly identifies the geographic boundaries of a property. Fences are used to clearly differentiate between areas that are under a specific level of security protection and those that aren't. Fencing can include a wide range of components, materials, and construction methods. It can consist of stripes painted on the ground, chain-link fences, barbed wire, concrete walls, or invisible perimeters that use laser, motion, or heat detectors. Various types of fences are effective against different types of intruders:

- Fences 3 to 4 feet high deter casual trespassers.
- Fences 6 to 7 feet high are too hard to climb easily and deter most intruders except determined ones.
- Fences 8 or more feet high with three strands of barbed wire deter even determined intruders.

A *gate* is a controlled exit and entry point in a fence. The deterrent level of a gate must be equivalent to the deterrent level of the fence to sustain the effectiveness of the fence as a whole. Hinges and locking/closing mechanisms should be hardened against tampering, destruction, or removal. When a gate is closed, it should not offer any additional access vulnerabilities. Keep the number of gates to a minimum. They can be manned by guards, or not. When they're not protected by guards, the use of dogs or electronic monitoring is recommended.

A *cage* is an enclosed fence area that can be used to protect assets from being accessed by unauthorized individuals. Cages can be used inside or outside. For a cage to be most effective, it needs to have a secured floor and ceiling.

Security guards

All physical security controls, whether static deterrents or active detection and surveillance mechanisms, ultimately rely on personnel to intervene and stop actual intrusions and attacks. *Security guards* exist to fulfill this need. Guards can be posted around a perimeter or inside to monitor access points or watch detection and surveillance monitors. The real benefit of guards is that they are able to adapt and react to various conditions or situations. Guards can learn and recognize attack and intrusion activities and patterns, adjust to a changing environment, and make decisions and judgment calls. Security guards are often an appropriate security control when immediate situation handling and decision-making onsite is necessary.

Unfortunately, using security guards is not a perfect solution. There are numerous disadvantages to deploying, maintaining, and relying on security guards. Not all environments and facilities support security guards. This may be because of actual human incompatibility or the layout, design, location, and construction of the facility. Not all security guards are themselves reliable. Prescreening, bonding, and training do not guarantee that you won't end up with an ineffective or unreliable security guard.

Even if a guard is initially reliable, guards are subject to physical injury and illness, take vacations, can become distracted, and are vulnerable to social engineering. In addition, security guards usually offer protection only up to the point at which their lives are endangered. Security guards are usually unaware of the scope of the operations in a facility and therefore are not thoroughly equipped to know how to respond to every situation. Finally, security guards are expensive.

The presence of security guards at an entrance or around the perimeter of a security boundary serves as a deterrent to intruders and provides a form of physical barrier. Guard dogs can also protect against intrusion by detecting the presence of unauthorized visitors.

A security guard can check each person's credentials before granting entry. You can also use a biometrically controlled door. In either entrance-control system, a log or list of entries and exits, along with visitors and escorts, can be maintained. Such a log will assist in tracking down suspects or verifying that all personnel are accounted for in the event of an emergency.

In the realm of physical security, *access controls* are mechanisms designed to manage and control entrance into a location such as a building, a parking lot, a room, or even a specific box or server rack. Being able to control who can gain physical proximity to your environment (especially your computers and networking equipment) lets you provide true security for your data, assets, and other resources.

One method to control access is to issue each valid worker an ID badge that can be either a simple photo ID or an electronic smartcard. A photo ID requires a security guard to view, discriminate, and then grant or deny access. In this process, the security guard can also add the name and action to an access roster. A smartcard can be used with an automated system that can electronically unlock and even open doors when a valid smartcard is swiped. Smartcard use is also easy to log and monitor. Additionally, the same smartcard used for facility access can also serve as a photo ID as well as an authentication factor for accessing the company network.

Alarms

Alarms or physical IDSs are systems—automated or manual—designed to detect an attempted intrusion, breach, or attack; the use of an unauthorized entry point; or the occurrence of some specific event at an unauthorized or abnormal time. IDSs used to monitor physical activity may include security guards, automated access controls, and motion detectors as well as other specialty monitoring techniques.

Physical IDSs, also called *burglar alarms*, detect unauthorized activities and notify the authorities (internal security or external law enforcement). The most common type of system uses a simple circuit (aka dry contact switch) consisting of foil tape in entrance points to detect when a door or window has been opened.

An intrusion detection mechanism is useful only if it is connected to an intrusion alarm. An intrusion alarm notifies authorities about a breach of physical security.

Two aspects of any intrusion detection and alarm system can cause it to fail: how it gets its power and how it communicates. If the system loses power, the alarm will not function. Thus, a reliable detection and alarm system has a battery backup with enough stored power for 24 hours of operation.

If communication lines are cut, an alarm may not function, and security personnel and emergency services will not be notified. Thus, a reliable detection and alarm system incorporates a heartbeat sensor for line supervision. A *heartbeat sensor* is a mechanism by which the communication pathway is either constantly or periodically checked with a test signal. If the receiving station detects a failed heartbeat signal, the alarm triggers automatically. Both measures are designed to prevent intruders from circumventing the detection and alarm system.

Whenever a motion detector registers a significant or meaningful change in the environment, it triggers an alarm. An alarm is a separate mechanism that triggers a deterrent, a repellent, and/or a notification:

Deterrent Alarms Alarms that trigger deterrents may engage additional locks, shut doors, and so on. The goal of such an alarm is to make further intrusion or attack more difficult.

Repellent Alarms Alarms that trigger repellents usually sound an audio siren or bell and turn on lights. These kinds of alarms are used to discourage intruders or attackers from continuing their malicious or trespassing activities and force them off the premises.

Notification Alarms Alarms that trigger notification are often silent from the intruder/attacker perspective but record data about the incident and notify administrators, security guards, and law enforcement. A recording of an incident can take the form of log files and/or CCTV tapes. The purpose of a silent alarm is to bring authorized security personnel to the location of the intrusion or attack in hopes of catching the person(s) committing the unwanted or unauthorized acts.

Alarms are also categorized by where they are located:

Local Alarm System Local alarm systems must broadcast an audible (up to 120 decibel [db]) alarm signal that can be easily heard up to 400 feet away. Additionally, they must be protected from tampering and disablement, usually by security guards. For a local alarm system to be effective, a security team or guards who can respond when the alarm is triggered must be positioned nearby.

Central Station System A central station system alarm is usually silent locally, but offsite monitoring agents are notified so they can respond to the security breach. Most residential security systems are of this type. Most central station systems are well-known or national security companies, such as Brinks and ADT. A proprietary system is similar to a central station system, but the host organization has its own onsite security staff waiting to respond to security breaches.

Auxiliary Station System Alarm systems can be added to either local or centralized alarm systems. When the security perimeter is breached, emergency services are notified to respond to the incident at the location. This could include fire, police, and medical services.

Two or more of these types of intrusion and alarm systems can be incorporated into a single solution.

Safe

Any device or removable media containing highly sensitive information should be kept locked securely in a *safe* when not in active use. You can install a department-wide safe that is managed by a single person, or you can install per-desk safes. A per-desk safe is often smaller, but it lets workers store devices and documentation securely while also allowing quick access.

Long-term storage of media and devices may require safes as well. Safes may be present onsite, or you can contract with an offsite storage facility to provide a safe for secured storage.

Secure cabinets/enclosures

Cabinets, device enclosures, rack-mounting systems, patch panels, wiring closets, and other equipment and cable containers can provide additional physical security through the use of locking mechanisms. Locking cabinets and other forms of containers can block or reduce access to power switches, adapter ports, media bays, and cable runs. Locking cabinets can be used in server rooms or in workspace areas. These can also include desks that give workers access to the monitor, mouse, and keyboard but sequester the main system chassis inside a locked desk compartment.

Protected distribution/Protected cabling

Protected distribution or *protective distribution systems* (PDSs) (also known as *protected cabling* systems) are the means by which cables are protected against unauthorized access or harm. The goals of PDSs are to deter violations, detect access attempts, and otherwise prevent compromise of cables. Elements of PDS implementation can include protective conduits, sealed connections, and regular human inspections. Some PDS implementations require intrusion or compromise detection within the conduits.

Airgap

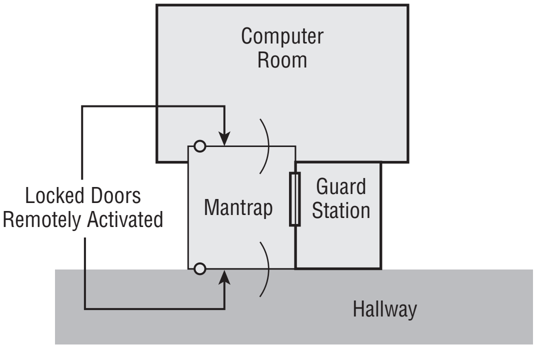
See the earlier section “Physical” for a discussion of the security benefits of this type of network segmentation.

Mantrap

Some high-value or high-security environments may also employ mantraps as a means to control access to the most secured, dangerous, or valuable areas of a facility. A *mantrap* is a form of high-security barrier entrance device (see Figure 3.23). It’s a small room with two doors: one in the trusted environment and one opening to the outside. The mantrap works like this:

1. A person enters the mantrap.
2. Both doors are locked.
3. The person must properly authenticate to unlock the inner door to gain entry. If the authentication fails, security personnel are notified, and the intruder is detained in the mantrap.

FIGURE 3.23 A mantrap

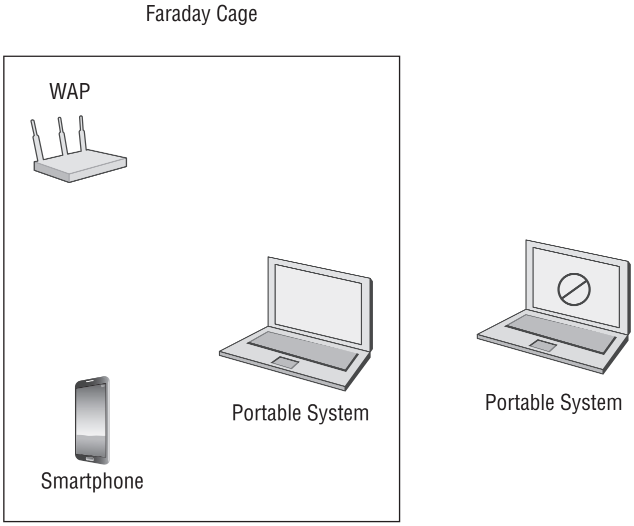


Mantraps often contain scales and cameras in order to prevent *piggybacking.* Piggybacking occurs when one person authenticates, opens a door, and lets another person enter without that second person authenticating to the system.

Faraday cage

A *Faraday cage* (Figure 3.24) is an enclosure that blocks or absorbs electromagnetic fields or signals. Faraday cage containers, computer cases, rack-mount systems, rooms, or even building materials are used to create a blockage against the transmission of data, information, metadata, or other emanations from computers and other electronics. Devices inside a Faraday cage can use electromagnetic (EM) fields for communications, such as wireless or Bluetooth, but devices outside the cage will not be able to eavesdrop on the signals of the systems within the cage.

FIGURE 3.24 A Faraday cage prevents WiFi (EM) access outside of the container.



Lock types

Although you need walls and fences to protect boundaries, there must be a means for authorized personnel to cross these barriers into the secured environment. Doors and gates can be locked and controlled in such a way that only authorized people can unlock and/or enter through them. Such control can take the form of a lock with a key that only authorized people possess. Locks are used to keep doors and containers secured in order to protect assets.

Hardware conventional locks and even electronic or smart locks are used to keep specific doors or other access portals closed and prevent entry or access to all but authorized individuals. With the risks of lock picking and bumping, locks resistant to such attacks must be used whenever valuable assets are to be protected from tampering or theft.

Doors used to control entrance into secured areas can be protected by locks that are keyed to biometrics. A biometric lock requires that the person present a biometric factor, such as a finger, a hand, or a retina to the scanner, which in turn transmits the fingerprint, hand, or retina scan to the validation mechanism. Only after the biometric is verified is the door unlocked and the person allowed entry. When biometrics are used to control entrance into secured areas, they serve as a mechanism of identity proofing as well as authentication.

However, door access systems need not be exclusively biometric. Smartcards and even traditional metal keys can function as authentication factors for physical entry points.

Many door access systems, whether supporting biometrics, smartcards, or even PINs, are designed around the electronic access control (EAC) concept. An EAC system is a door-locking and -access mechanism that uses an electromagnet to keep a door closed, a reader to accept access credentials, and a door-close spring and sensor to ensure that the door recloses within a reasonable timeframe.

Biometrics

Biometrics is the term used to describe the collection of physical attributes of the human body that can be used as identification or authentication factors. Biometrics fall into the authentication factor category of something you are: you, as a human, have the element of identification as part of your physical body. Biometrics include fingerprints, palm scans (use of the entire palm as if it were a fingerprint), hand geometry (geometric dimensions of the silhouette of a hand), retinal scans (pattern of blood vessels at the back of the eye), iris scans (colored area of the eye around the pupil), facial recognition, voice recognition, signature dynamics, and keyboard dynamics.

Although biometrics are a stronger form of authentication than passwords alone, biometrics in and of themselves aren’t the best solution. Even with biometrics, implementing multifactor authentication is the most secure solution.

The key element in deploying biometrics as an element of authentication is a biometric device or a biometric reader. This is the hardware designed to read, scan, or view the body part that is to be presented as proof of identification.

See the Chapter 4 section “Biometric factors” for more about the benefits and limitations of biometrics.

Barricades/bollards

Barricades, in addition to fencing (discussed earlier), are used to control both foot traffic and vehicles. K-rails (often seen during road construction), large planters, zigzag queues, bollards, and tire shredders are all examples of barricades. When used properly, they can control crowds and prevent vehicles from being used to cause damage to your building.

Tokens/cards

A token device or an access card can be used as an element in authentication when gaining physical entry into a facility. See the Chapter 4 sections “Tokens,” “Physical access control,” and “Certificate-based authentication.”

Environmental controls

Environmental monitoring is the process of measuring and evaluating the quality of the environment within a given structure. This can focus on general or basic concerns, such as temperature, humidity, dust, smoke, and other debris. However, more advanced systems can include chemical, biological, radiological, and microbiological detectors.

When you’re designing a secure facility, it’s important to keep various environmental factors in mind. These include the following:

- Controlling the temperature and humidity
- Minimizing smoke and airborne dust and debris
- Minimizing vibrations
- Preventing food and drink from being consumed near sensitive equipment
- Avoiding strong magnetic fields
- Managing electromagnetic and radio frequency interference
- Conditioning the power supply
- Managing static electricity
- Providing proper fire detection and suppression

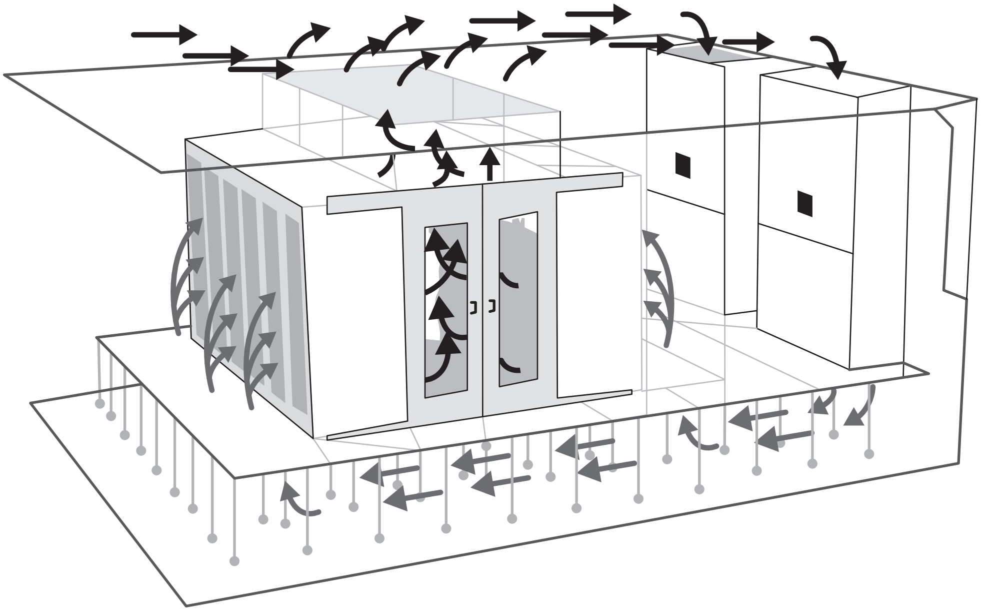
HVAC

Heating, ventilating, and air-conditioning (HVAC) management is important for two reasons: temperature and humidity. In the mission-critical server vault or room, the temperature should be maintained around a chosen set point to support optimal system operation. For many, the “optimal” temperature or preferred set point is in the mid-60s Fahrenheit. However, some organizations are operating as low as 55 degrees and others are creeping upward into the 90s. With good airflow management and environmental monitoring, many companies are saving 4 to 5 percent on their cooling bills for every one degree they increase their server room temperature. Throughout the organization, humidity levels should be managed to keep the relative humidity between 40 and 60 percent. Low humidity allows static electricity buildup, with discharges capable of damaging most electronic equipment. High humidity can allow condensation, which leads to corrosion.

Hot and cold aisles

Hot and cold aisles are a means of maintaining optimum operating temperature in large server rooms. The overall technique is to arrange server racks in lines separated by aisles (Figure 3.25). Then the airflow system is designed so hot, rising air is captured by air-intake vents on the ceiling, whereas cold air is returned in opposing aisles from either the ceiling or the floor. Thus, every other aisle is hot, then cold. This creates a circulating air pattern that is intended to optimize the cooling process.

FIGURE 3.25 A hot aisle/cold aisle air management system



Fire suppression

Fire is a common problem that must be addressed in the design of any facility. Electrical fires are common causes of building fires; they may result from overheated computer or networking equipment or improperly managed electrical power cables and distribution nodes (power strips).

Early fire detection and suppression is important because the earlier the discovery, the less damage is caused to the facility and equipment. Personnel safety is always of utmost importance. However, in a dedicated, secured, mission-critical server room (often called a *server cage*, *server vault*, or *datacenter*), the fire-suppression system can be gas discharge–based rather than water-based. A gas discharge–based system removes oxygen from the air and may even suppress the chemical reaction of combustion, often without damaging computer equipment, but such systems are harmful to people. If a water-based system must be used, employ a pre-action system that allows the release of the water to be turned off in the event of a false alarm.

The safety of the facility and personnel should be a priority of a security effort. Human life and safety are without question the top concerns, but sufficient focus needs to be placed on providing physical security for buildings and other real-world assets. The following sections discuss many aspects of security and safety.

Every building needs an escape plan, and a backup escape plan, and even a backup backup escape plan. An *escape route* is the path someone should take out of a building to reach safety. The preferred and alternate escape routes should be identified, marked, and clearly communicated to all personnel. Accommodations for those with disabilities need to be made.

Employees need to be trained in safety and escape procedures. Once they are trained, their training should be tested using drills and simulations. Having workers go through the routine of escape helps to reinforce their understanding of the escape plans and available routes, and it also helps reduce anxiety and panic in case of a threatening event.

All elements of physical security, especially those related to human life and safety, should be tested on a regular basis. It is mandated by law that fire extinguishers, fire detectors/alarms, and elevators be inspected regularly. A self-imposed schedule of control testing should be implemented for door locks, fences, gates, mantraps, turnstiles, video cameras, and all other physical security controls.

Cable locks

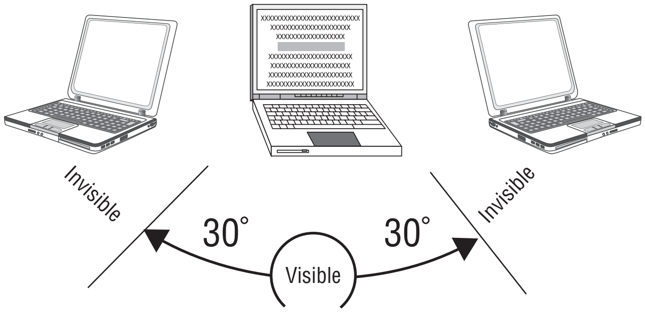
A *cable lock* is used to keep smaller pieces of equipment from being easy to steal. Many devices, most commonly portable computers, have a Kensington Security Slot (K-Slot) that is designed as a connection point for a cable lock. The K-Slot was originally developed by Kensington, which continues to develop new cable lock security devices.

A cable lock usually isn't an impenetrable security device, since most portable systems are constructed with thin metal and plastic. However, a thief will be reluctant to swipe a cable-locked device, because the damage caused by forcing the cable lock out of the K-Slot will be obvious when they attempt to pawn or sell the device.

Screen filters

It may be worthwhile to install *screen filters*, also called privacy filters, which reduce the range of visibility of a screen down to a maximum of 30 degrees from perpendicular (Figure 3.26). These types of screens are designed to prevent someone sitting directly next to you, such as on an airplane, from being able to see the contents of your display.

FIGURE 3.26 The viewing angle for a screen filter



Cameras

Video surveillance, video monitoring, closed-circuit television (CCTV), and security cameras are all means to deter unwanted activity and create a digital record of the occurrence of events. Cameras should be positioned to watch exit and entry points allowing any change in authorization level—for example, doors allowing entry into a facility from outside, doors allowing entry into work areas from common areas, and doors allowing entry into high-security areas from work areas. Cameras should also be used to monitor activities around valuable assets and resources, such as server rooms, safes, vaults, and component closets, as well as to provide additional protection in public areas such as parking structures and walkways.

Cameras should be configured to record to storage media. This has traditionally been some sort of tape, such as VCR tape. However, modern systems may record to DVD, NVRAM, or hard drives and may do so over a wired or even an encrypted wireless connection.

Cameras vary in type. Typical security cameras operate by recording visible-light images and often require additional lighting in low-light areas. Alternative camera types include those that record only when motion is detected, those that are able to record in infrared, and those that can automatically track movement.

Video records may be used to detect policy violations, track personnel movements, or capture an intruder on film. Video recordings should be monitored in real time or reviewed on a periodic basis in order to provide a detective benefit. Just the visible presence of video cameras can provide a deterrent effect to would-be perpetrators.

A camera is primarily used to detect and record unwanted or unauthorized activity. If someone is aware that a camera is present and will record their actions, that person is less likely to perform actions that are violations. This is generally known as a *deterrent*.

A security guard is able to move around a facility to potentially view places a camera is unable to see. Security guards are often as much a deterrent as they are a detective control. They can respond to varying issues and can adjust their actions based on changing conditions.

Both cameras and guards have useful security features, but both require proper use to be beneficial, both have their own unique requirements for use, and both are costly in their own ways.

Motion detection

A *motion detector*, or *motion sensor*, is a device that senses movement or sound in a specific area. Many types of *motion detection* exist, including infrared, heat, wave pattern, capacitance, photoelectric, and passive audio:

- An infrared motion detector monitors for significant or meaningful changes in the infrared lighting pattern of a monitored area.
- A heat-based motion detector monitors for significant or meaningful changes in the heat levels and patterns in a monitored area.
- A wave-pattern motion detector transmits a consistent low-frequency ultrasonic or high-frequency microwave signal into a monitored area and monitors for significant or meaningful changes or disturbances in the reflected pattern.
- A capacitance motion detector senses changes in the electrical or magnetic field surrounding a monitored object.
- A photoelectric motion detector senses changes in visible light levels for the monitored area. Photoelectric motion detectors are usually deployed in internal rooms that have no windows and are kept dark.
- A passive audio motion detector listens for abnormal sounds in the monitored area.

The proper technology of motion detection should be selected for the environment where it will be deployed, in order to minimize false positives and false negatives.

Logs

Logs of physical access should be maintained. These can be created automatically through the use of smartcards for gaining access into the facility or manually by a security guide who will indicate entrance after inspecting each person's ID. The purpose of physical access logs is to establish context for logical logs produced by servers, workstations, and networking equipment. The logs are also helpful in the event of an emergency in order to determine whether everyone has escaped a building safely or if rescue teams should be sent in.

Infrared detection

Infrared detection is often used by security cameras to see in perceived darkness or to detect movement in an area. These concepts were discussed in the previous sections "Cameras" and "Motion detection."

Key management

Key management in relation to physical security focuses on the issuance of physical metal keys to those who need access into secured rooms, areas, or containers. A detailed log should be maintained of who was issued which key for which room/container and for what purpose. Regular auditing should be performed to ensure the responsible party still possesses the key and discloses whether or not the key has been exposed to theft or duplication. Keys should be numbered and, when possible, labeled or identified as not to be duplicated (so a locksmith will refuse to make a copy). Keys should be returned to the key manager when access is no longer required by that individual. When a worker who was in possession of a key is terminated, that lock and key should be replaced.

Digital or electronic key management relates to cryptography and is covered throughout Chapter 6.

Exam Essentials

Understand physical access control. Physical access control refers to mechanisms designed to manage and control entrance into a location. Being able to control who can gain physical proximity to your environment (especially your computers and networking equipment) allows you to provide true security for your data, assets, and other resources. Without physical access control, you have no security.

Know about lighting. Lighting is a commonly used form of perimeter security control. The primary purpose of lighting is to discourage casual intruders, trespassers, prowlers, or would-be thieves who would rather perform their misdeeds in the dark, such as vandalism, theft, and loitering.

Understand signs. Signs can be used to declare areas as off limits to those who are not authorized, to indicate that security cameras are in use, and to disclose safety warnings.

Know about fencing, gates, and cages. A fence is a perimeter-defining device. Fencing protects against casual trespassing and clearly identifies the geographic boundaries of a property. Fences are used to clearly differentiate between areas that are under a specific level of security protection and those that aren't. A gate is a controlled exit and entry point in a fence. A cage is an enclosed fence area that can be used to protect assets from being accessed by unauthorized individuals.

Understand security guards. All physical security controls, whether static deterrents or active detection and surveillance mechanisms, ultimately rely on personnel to intervene and stop actual intrusions and attacks. Security guards exist to fulfill this need.

Comprehend alarms. Physical IDSs, also called burglar alarms, detect unauthorized activities and notify the authorities (internal security or external law enforcement).

Understand safes. Any device or removable media containing highly sensitive information should be kept locked securely in a safe when not in active use.

Know about PDS. Protected distribution or protective distribution systems (PDSs) (also known as protected cabling systems) are the means by which cables are protected against unauthorized access or harm.

Understand mantraps. A mantrap is a form of high-security barrier entrance device. It's a small room with two doors: one to the trusted environment and one to the outside. A person must properly authenticate to unlock the inner door and gain entry.

Realize the importance of a Faraday cage. A Faraday cage is an enclosure that blocks or absorbs electromagnetic fields or signals.

Understand biometrics. Biometrics is the collection of physical attributes of the human body that can be used as authentication factors (something you are). Biometrics include fingerprints, palm scans (use of the entire palm as if it were a fingerprint), hand geometry (geometric dimensions of the silhouette of a hand), retinal scans (pattern of blood vessels at the back of the eye), iris scans (colored area of the eye around the pupil), facial recognition, voice recognition, signature dynamics, and keyboard dynamics.

Comprehend environmental monitoring. Environmental monitoring is the process of measuring and evaluating the quality of the environment within a given structure.

Understand humidity management. Throughout the organization, humidity levels should be managed to keep the relative humidity between 40 and 60 percent. Low humidity allows static electricity buildup, with discharges capable of damaging most electronic equipment. High humidity can allow condensation, which leads to corrosion.

Know about hot and cold aisles. Hot and cold aisles are a means of maintaining optimum operating temperature in large server rooms.

Understand fire suppression. Early fire detection and suppression is important because the earlier the discovery, the less damage will be caused to the facility and equipment. Personnel safety is always of utmost importance.

Know about cable locks. A cable lock is used to keep smaller pieces of equipment from being easy to steal.

Understand screen filters. It may be worthwhile to install screen filters that reduce the range of visibility of a screen down to a maximum of 4 degrees from perpendicular.

Know about cameras. Video surveillance, video monitoring, closed-circuit television (CCTV), and security cameras are all means to deter unwanted activity and create a digital record of the occurrence of events.

Understand motion detection. A motion detector, or motion sensor, is a device that senses movement or sound in a specific area. Many types of motion detection exist, including infrared, heat, wave pattern, capacitance, photoelectric, and passive audio.

Review Questions

You can find the answers in the Appendix.

1. Which of the following allows the deployment of a publicly accessible web server without compromising the security of the private network?

   A. Intranet

   B. DMZ

   C. Extranet

   D. Switch

2. An organization has a high-speed fiber Internet connection that it uses for most of its daily operations, as well as its offsite backup operations. This represents what security problem?

   A. Single point of failure

   B. Redundant connections

   C. Backup generator

   D. Offsite backup storage

3. A security template can be used to perform all but which of the following tasks?

   A. Capture the security configuration of a master system

   B. Apply security settings to a target system

   C. Return a target system to its precompromised state

   D. Evaluate compliance with security of a target system

4. What technique or method can be employed by hackers and researchers to discover unknown flaws or errors in software?

   A. Dictionary attacks

   B. Fuzzing

   C. War dialing

   D. Cross-site request forgery

5. What is a security risk of an embedded system that is not commonly found in a standard PC?

   A. Power loss

   B. Access to the Internet

   C. Control of a mechanism in the physical world

   D. Software flaws

6. To ensure that whole-drive encryption provides the best security possible, which of the following should not be performed?

   A. Screen lock the system overnight.

   B. Require a boot password to unlock the drive.

   C. Lock the system in a safe when it is not in use.

   D. Power down the system after use.

7. In order to avoid creating a monolithic security structure, organizations should adopt a wide range of security mechanisms. This concept is known as ___________.

   A. Defense in depth

   B. Control diversity

   C. Intranet buffering

   D. Sandboxing

8. When offering a resource to public users, what means of deployment provides the most protection for a private network?

   A. Intranet

   B. Wireless

   C. Honeynet

   D. DMZ

9. When you are implementing a security monitoring system, what element is deployed in order to detect and record activities and events?

   A. Correlation engine

   B. Tap

   C. Sensor

   D. Aggregation switch

10. When an enterprise is using numerous guest OSs to operate their primary business operations, what tool or technique can be used to enable communications between guest OSs hosted on different server hardware but keep those communications distinct from standard subnet communications?

    A. VPN

    B. SDN

    C. EMP

    D. FDE

11. What type of OS is designed for public end-user access and is locked down so that only preauthorized software products and functions are enabled?

    A. Kiosk

    B. Appliance

    C. Mobile

    D. Workstation

12. When you need to test new software whose origin and supply chain are unknown or untrusted, what tool can you use to minimize the risk to your network or workstation?

    A. Hardware security module

    B. UEFI

    C. Sandboxing

    D. SDN

13. What is the concept of a computer implemented as part of a larger system that is typically designed around a limited set of specific functions (such as management, monitoring, and control) in relation to the larger product of which it's a component?

    A. IoT

    B. Application appliance

    C. SoC

    D. Embedded system

14. What is an industrial control system (ICS) that provides computer management and control over industrial processes and machines?

    A. SCADA

    B. HSM

    C. OCSP

    D. MFD

15. Which SDLC model is based around adaptive development where focusing on a working product and fulfilling customer needs is prioritized over rigid adherence to a process, use of specific tools, and detailed documentation?

    A. Waterfall

    B. Agile

    C. Spiral

    D. Ad hoc

16. When an organization wishes to automate many elements and functions of IT management, such as development, operations, security, and quality assurance, they are likely to be implementing which of the following?

    A. SCADA

    B. UTM

    C. IaaS

    D. DevOps

17. What is not a cloud security benefit or protection?

    A. CASB

    B. SECaaS

    C. VM sprawl

    D. VM isolation

18. What form of cloud service provides the customer with the ability to run their own custom code but does not require that they manage the execution environment or operating system?

    A. SaaS

    B. PaaS

    C. IaaS

    D. SECaaS

19. What recovery mechanism is used to return a system back to a previously operating condition when a new software install corrupts the operating system?

A. Revert to known state

B. Roll back to known configuration

C. Live boot media

D. Template

20. What type of security mechanism can be used to prevent a vehicle from damaging a facility?

A. Fencing

B. Lighting

C. Bollard

D. Access cards

Chapter 4

Identity and Access Management

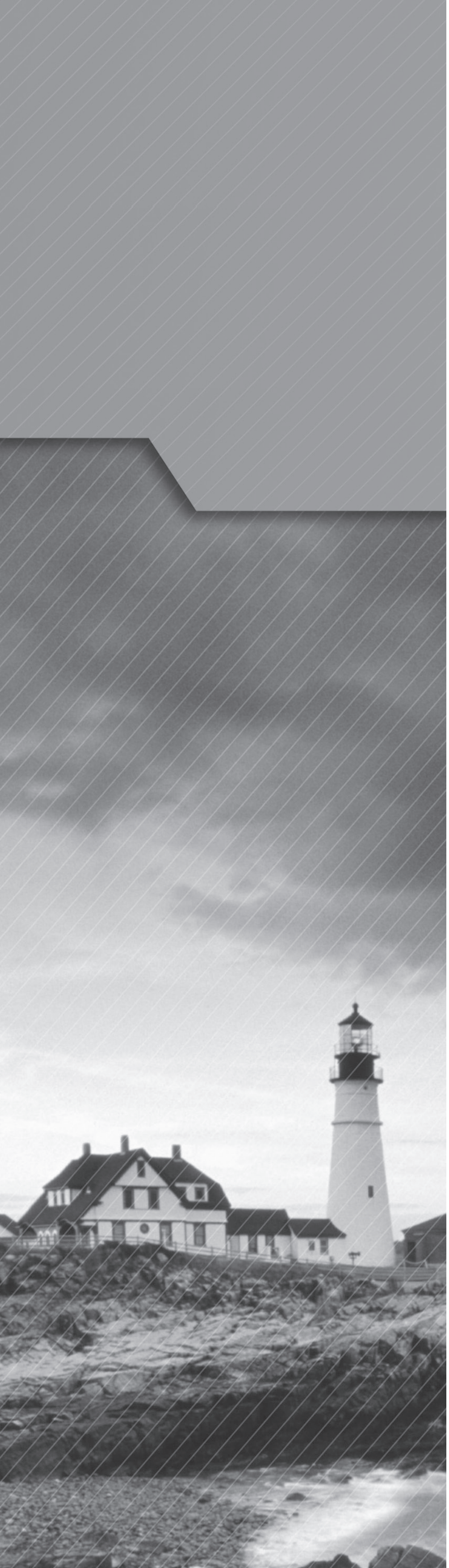
COMPTIA SECURITY+ EXAM OBJECTIVES COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:

✓ **4.1 Compare and contrast identity and access management concepts.**

- Identification, authentication, authorization and accounting (AAA)
- Multifactor authentication
  - Something you are
  - Something you have
  - Something you know
  - Somewhere you are
  - Something you do
- Federation
- Single sign-on
- Transitive trust

✓ **4.2 Given a scenario, install and configure identity and access services.**

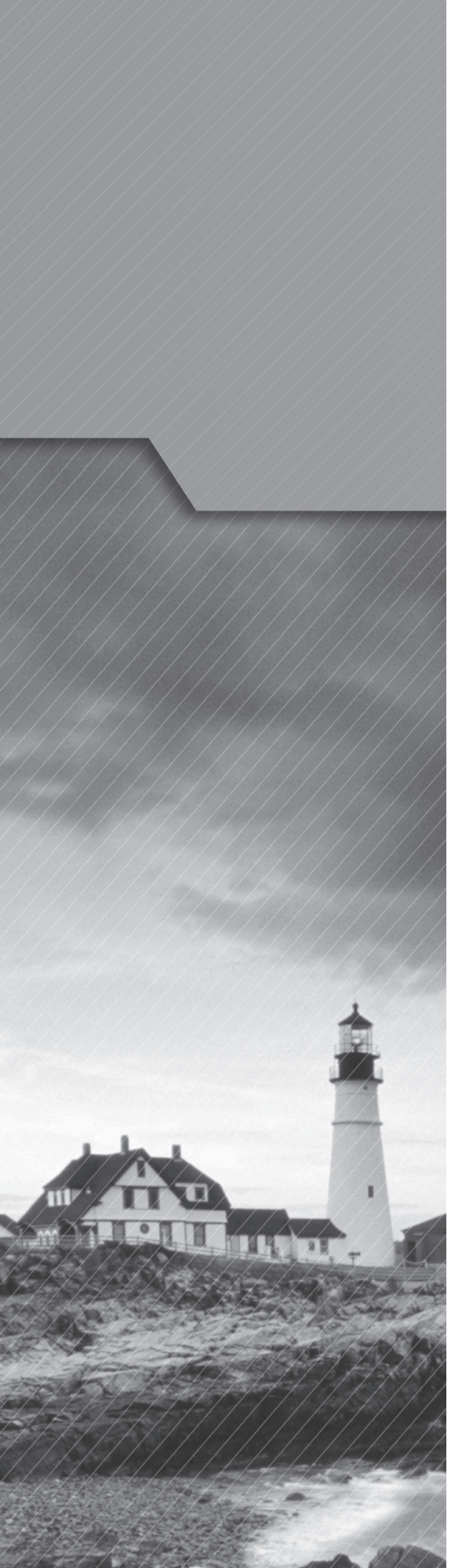
- LDAP
- Kerberos
- TACACS+
- CHAP
- PAP
- MSCHAP
- RADIUS
- SAML
- OpenID Connect



- OAuth
- Shibboleth
- Secure token
- NTLM

✓ **4.3 Given a scenario, implement identity and access management controls.**

- Access control models
  - MAC
  - DAC
  - ABAC
  - Role-based access control
  - Rule-based access control
- Physical access control
  - Proximity cards
  - Smart cards
- Biometric factors
  - Fingerprint scanner
  - Retinal scanner
  - Iris scanner
  - Voice recognition
  - Facial recognition
  - False acceptance rate
  - False rejection rate
  - Crossover error rate
- Tokens
  - Hardware
  - Software
  - HOTP/TOTP
- Certificate-based authentication
  - PIV/CAC/smart card
  - IEEE 802.1x



- File system security
- Database security

✓ **4.4 Given a scenario, differentiate common account management practices.**

- Account types
  - User account
  - Shared and generic accounts/credentials
  - Guest accounts
  - Service accounts
  - Privileged accounts
- General Concepts
  - Least privilege
  - Onboarding/offboarding
  - Permission auditing and review
  - Usage auditing and review
  - Time-of-day restrictions
  - Recertification
  - Standard naming convention
  - Account maintenance
  - Group-based access control
  - Location-based policies
- Account policy enforcement
  - Credential management
  - Group policy
  - Password complexity
  - Expiration
  - Recovery
  - Disablement
  - Lockout
  - Password history
  - Password reuse
  - Password length

The Security+ exam will test your knowledge of identity and access management concepts, since they relate to secure networked systems both for the home office and in corporate environments. To pass the test and be effective in implementing security, you need to understand the basic concepts and terminology related to network security as detailed in this chapter. You will also need to be familiar with when and why to use various tools and technologies, given a scenario.

4.1 Compare and contrast identity and access management concepts.

Identity is the concept of uniquely naming and referencing each individual user, program, and system component in order to authenticate, authorize, and audit for the purposes of holding users accountable for their actions. This is also known as "identification followed by authentication." Access management is the concept of defining and enforcing what can and cannot be done by each identified subject. This is also known as authorization.

Identification, authentication, authorization and accounting (AAA)

It's important to understand the differences between identification, authentication, and authorization. Although these concepts are similar and are essential to all security mechanisms, they're distinct and must not be confused.

Identification and authentication are commonly used as a two-step process, but they're distinct activities. *Identification* is the assertion of an identity. This needs to occur only once per authentication or access process. Any one of the common authentication factors can be employed for identification. Once identification has been performed, the authentication process must take place. *Authentication* is the act of verifying or proving the claimed identity. The issue is both checking that such an identity exists in the known accounts of the secured environment and ensuring that the human claiming the identity is the correct, valid, and authorized human to use that specific identity.

A *username* is the most common form of identification. It's any name used by a subject in order to be recognized as a valid user of a system. Some usernames are derived from a person's actual name, some are assigned, and some are chosen by the subject. Using a consistent username across multiple systems can help establish a consistent reputation across those platforms. However, it's extremely important to keep all authentication factors unique between locations, even when duplicating a username.

Authentication can take many forms, most commonly of one-, two-, or multifactor configurations. The more unique factors used in an authentication process, the more resilient and reliable the authentication itself becomes. If all the proffered authentication factors are valid and correct for the claimed identity, it's then assumed that the accessing person is who they claim to be. Then the permission- and action-restriction mechanisms of authorization take over to control the activities of the user from that point forward.

Identity proofing—that is, authentication—typically takes the form of one or more of the following authentication factors:

- *Something you know* (such as a password, code, PIN, combination, or secret phrase)
- *Something you have* (such as a smartcard, token device, or key)
- *Something you are* (such as a fingerprint, a retina scan, or voice recognition; often referred to as *biometrics*, discussed later in this chapter)
- *Somewhere you are* (such as a physical or logical location); this can be seen as a subset of something you know.
- *Something you do* (such as your typing rhythm, a secret handshake, or a private knock). This can be seen as a subset of something you know.

The authentication factor of something you know is also known as a Type 1 factor, something you have is also known as a Type 2 factor, and something you are is also known as a Type 3 factor. The factors of somewhere you are and something you do are not given Type labels.

When only one authentication factor is used, this is known as *single-factor authentication* (or, rarely, *one-factor authentication*).

Authorization is the mechanism that controls what a subject can and can't do, access, use, or view. Authorization is commonly called *access control* or *access restriction*. Most systems operate from a default authorization stance of deny by default or implicit deny. Then all needed access is granted by exception to individual subjects or to groups of subjects.

Once a subject is authenticated, its access must be *authorized*. The process of authorization ensures that the requested activity or object access is possible, given the rights and privileges assigned to the authenticated identity (which we refer to as the *subject* from this point forward). Authorization indicates who is trusted to perform specific operations. In most cases, the system evaluates an access-control matrix that compares the subject, the object, and the intended activity. If the specific action is allowed, the subject is authorized; if it's disallowed, the subject isn't authorized.

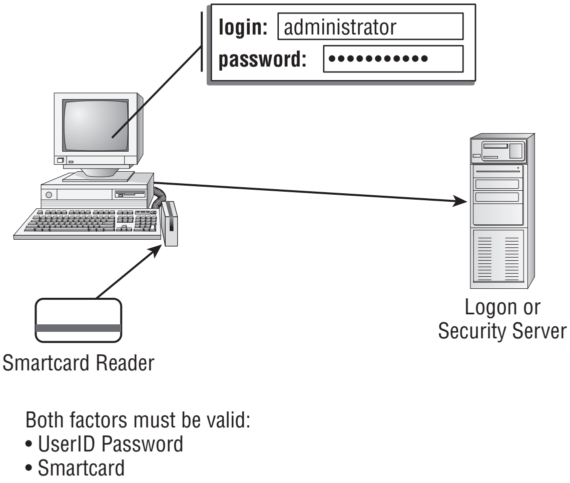
Keep in mind that just because a subject has been identified and authenticated, that doesn't automatically mean it has been authorized. It's possible for a subject to log on to a network (in other words, be identified and authenticated) and yet be blocked from accessing a file or printing to a printer (by not being authorized to perform such activities). Most network users are authorized to perform only a limited number of activities on a specific collection of resources. Identification and authentication are "all-or-nothing" aspects of access control. Authorization occupies a wide range of variations between all and nothing for each individual subject or object in the environment. Examples would include a user who can read a file but not delete it, or print a document but not alter the print queue, or log on to a system but not be allowed to access any resources.

Multifactor authentication

Multifactor authentication is the requirement that a user must provide two or more authentication factors in order to prove their identity. There are three generally recognized categories of authentication factors.

When two different authentication factors are used, the strategy is known as *two-factor authentication* (see Figure 4.1). If two or more authentication factors are used but some of them are of the same type, it is known as *strong authentication*. Using different factors (whether two or three) is always a more secure solution than any number of factors of the same authentication type, because with two or more different factors, two or more different types of attacks must take place to capture the authentication factor. With strong authentication, even if 10 passwords are required, only a single type of password-stealing attack needs to be waged to break through the authentication security.

FIGURE 4.1 Two-factor authentication



Authentication factors are the concepts used to verify the identity of a subject.

Something you are

Something you are is often known as biometrics. Examples include fingerprints, a retina scan, or voice recognition. See "Biometric factors" later in the chapter for more information.

Something you have

Something you have requires the use of a physical object. Examples include a smartcard, token device, or key.

Something you know

Something you know involves information you can recall from memory. Examples include a password, code, PIN, combination, or secret phrase.

Somewhere you are

Somewhere you are is a location-based verification. Examples include a physical location or a logical address, such as a domain name, an IP address, or a MAC address.

Something you do

Something you do involves some skill or action you can perform. Examples include solving a puzzle, a secret handshake, or a private knock. This concept can also include activities that are biometrically measured and semi-voluntary, such as your typing rhythm, patterns of system use, or mouse behaviors.

Federation

Federation or *federated identity* is a means of linking a subject's accounts from several sites, services, or entities in a single account. It's a means to accomplish single sign-on. Federated solutions often implement trans-site authentication using SAML (see the later section "SAML").

Federation creates authentication trusts between systems in order to facilitate single sign-on benefits. Federation trusts can be one-way or two-way and can be transitive or nontransitive. In a one-way trust, as when system A is trusted by system B, users from A can access resources in both A and B systems, but users from B can only access resources in B. In a two-way trust, such as between system A and system B, users from either side can access resources on both sides. If three systems are trust-linked using two-way nontransitive trusts, such as A links to B which links to C, then A resources are accessible by users from A and B, B resources are accessible by users from A, B, and C, and C resources are accessible by users from B and C. If three systems are trust-linked using two-way transitive trusts, then all users from all three systems can access resources from all three systems.

Single sign-on

Single sign-on (SSO) means that once a user (or other subject) is authenticated into the realm, they don't need to reauthenticate to access resources on any realm entity. (*Realm* is another term for domain or network.) This allows users to access all the resources, data, applications, and servers they need to perform their work tasks with a single authentication procedure. SSO eliminates the need for users to manage multiple usernames and passwords, because only a single set of logon credentials is required. Some examples of single sign-on include Kerberos, SESAME, NetSP, KryptoKnight, directory services, thin clients, and scripted access. Kerberos is one of the SSO solution options you should know about for the Security+ exam; it is discussed in the later section "Kerberos."

Transitive trust

Transitive trust or *transitive authentication* is a security concern when a block can be bypassed using a third party. A transitive trust is a linked relationship between entities (such as systems, networks, or organizations) where trust from one endpoint crosses over or through middle entities to reach the farthest linked endpoint. For example, if four systems are transitive trust linked, such as A-B-C-D, then entities in A can access resources in B, C, and D thanks to the nature of a transitive trust. It can be thought of as a shared trust.

A real-world example of transitive trust occurs when you order a pizza. The cook makes the pizza and passes it on to the assistant, who packages the pizza in a box. The assistant then hands the pizza to the delivery person, the delivery person brings it to your location to hand it to your roommate, and then your roommate brings the pizza into the kitchen, where you grab a slice to eat. Since you trust each link in the chain, you are experiencing transitive trust.

Keep in mind that transitive trust can be both a beneficial feature of linked systems as well as a source of risk or compromise. If the cook placed pineapple, mushrooms, or anchovies on the pizza that you did not want or order, then the trust is broken.

Attackers often seek out transitive trust situations in order to bypass defenses and blockades against a direct approach. For example, the company firewall prevents the attacker from launching a direct attack against the internal database server. The attacker instead targets a worker who uses social networks. After friending the target, the attacker sends the worker a link that leads to a malware infector. If the worker clicks on the link, their system may become infected by remote-control malware. Then, when the worker takes the compromised system back into the office, it provides the attacker with an access pathway to attack the internal database. Thus, the transitive trust of attacker through social network to worker to company network allowed a security breach to take place.

Exam Essentials

Understand identification. Identification is the act of claiming an identity using just one authentication factor.

Define authentication. Authentication is the act of proving a claimed identity using one or more authentication factors.

Understand multifactor authentication. Multifactor authentication is the requirement that users must provide two or more authentication factors in order to prove their identity.

Know about multifactor authentication. Multifactor authentication or strong authentication occurs when two or more authentication factors are used but some of them are of the same type.

Understand two-factor authentication. Two-factor authentication occurs when two different authentication factors are used.

Comprehend federation. Federation or federated identity is a means of linking a subject's accounts from several sites, services, or entities in a single account.

Understand single sign-on. Single sign-on means that once a user (or other subject) is authenticated into a realm, they need not reauthenticate to access resources on any realm entity.

Know about transitive trust. Transitive trust or transitive authentication is a security concern when a block can be bypassed using a third party. A transitive trust is a linked relationship between entities where trust from one endpoint crosses over or through middle entities to reach the farthest linked endpoint.

4.2 Given a scenario, install and configure identity and access services.

Authentication is the mechanism by which users prove their identity to a system. It's the process of proving that a subject is the valid user of an account. Often, the authentication process involves a simple username and password. But other more complex authentication factors or credential-protection mechanisms are involved in order to provide strong protection for the logon and account-verification processes. The authentication process requires that the subject provide an identity and then proof of that identity.

Many systems and technologies are involved with identification, authentication, and access control. Several of these are discussed in this section.

LDAP

Please see the "LDAPS" section in Chapter 2, "Technologies and Tools," for an introduction to this technology.

LDAP is usually present by default in every private network because it is the primary foundation of network directory services, such as Active Directory. LDAP is used to grant access to information about available resources in the network. The ability to view or search network resources can be limited through the use of authorization restrictions.

Kerberos

Early authentication transmission mechanisms sent logon credentials from the client to the authentication server in clear text. Unfortunately, this solution is vulnerable to eavesdropping and interception, thus making the security of the system suspect. What was needed was a solution that didn't transmit the logon credentials in a form that could be easily captured, extracted, and reused.

One such method for providing protection for logon credentials is *Kerberos*, a trusted third-party authentication protocol that was originally developed at MIT under Project Athena. The current version of Kerberos in widespread use is version 5. Kerberos is used to authenticate network principles (subjects) to other entities on the network (objects, resources, and servers). Kerberos is platform independent; however, some OSs require special configuration adjustments to support true interoperability (for example, Windows Server with Unix).

Kerberos is a centralized authentication solution. The core element of a Kerberos solution is the *key distribution center (KDC)*, which is responsible for verifying the identity of principles and granting and controlling access within a network environment through the use of secure cryptographic keys and tickets.

Kerberos is a trusted third-party authentication solution because the KDC acts as a third party in the communications between a client and a server. Thus, if the client trusts the KDC and the server trusts the KDC, then the client and server can trust each other.

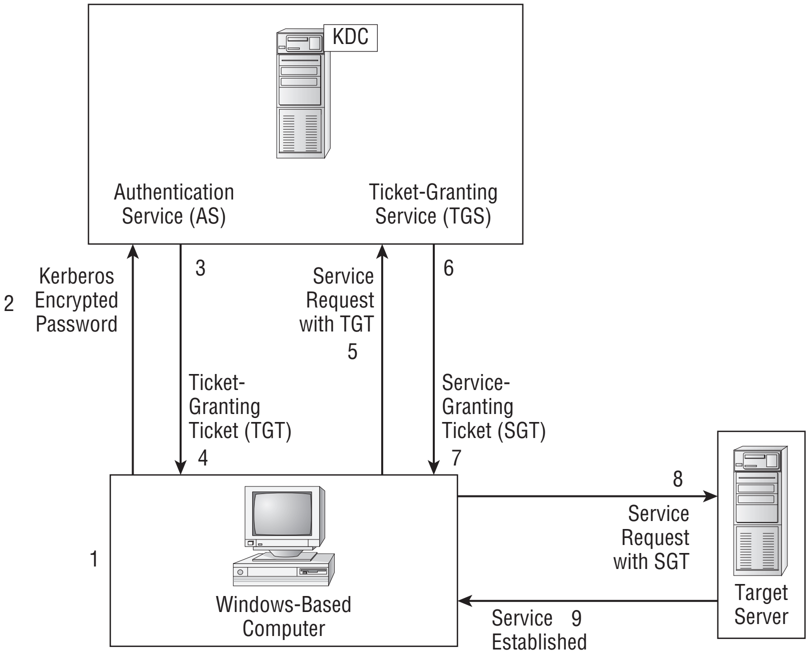
Kerberos is also a single sign-on solution. *Single sign-on* means that once a user (or other subject) is authenticated into the realm, they need not reauthenticate to access resources on any realm entity. (A *realm* is the network protected under a single Kerberos implementation.)

The basic process of Kerberos authentication is as follows:

1. The subject provides logon credentials.
2. The Kerberos client system encrypts the password and transmits the protected credentials to the KDC.
3. The KDC verifies the credentials and then creates a *ticket-granting ticket (TGT)*—a hashed form of the subject's password with the addition of a time stamp that indicates a valid lifetime. The TGT is encrypted and sent to the client.
4. The client receives the TGT. At this point, the subject is an authenticated principle in the Kerberos realm.
5. The subject requests access to resources on a network server. This causes the client to request a *service ticket (ST)* from the KDC.
6. The KDC verifies that the client has a valid TGT and then issues an ST to the client. The ST includes a time stamp that indicates its valid lifetime.
7. The client receives the ST.
8. The client sends the ST to the network server that hosts the desired resource.
9. The network server verifies the ST. If it's verified, it initiates a communication session with the client. From this point forward, Kerberos is no longer involved.

Figure 4.2 shows the Kerberos authentication process.

FIGURE 4.2 The Kerberos authentication process


KDC
Authentication Service (AS)
Ticket-Granting Service (TGS)
2 Kerberos Encrypted Password
3
Ticket-Granting Ticket (TGT) 4
Service Request with TGT 5
6
Service-Granting Ticket (SGT) 7
1
Windows-Based Computer
8 Service Request with SGT
Target Server
Service Established 9

The Kerberos authentication method helps ensure that logon credentials aren't compromised while in transit from the client to the server. The inclusion of a time stamp in the tickets ensures that expired tickets can't be reused. This prevents replay and spoofing attacks against Kerberos.

Kerberos supports *mutual authentication* (client and server identities are proven to each other). It's scalable and thus able to manage authentication for large networks. Being centralized, Kerberos helps reduce the overall time involved in accessing resources within a network.

NOTE

Kerberos is used to provide security and protection for authentication credentials alone. It isn't used in any way to provide encryption or security for other types of data transfer.

TACACS+

Terminal Access Controller Access Control System (TACACS) is another example of an AAA server. TACACS is an Internet standard (RFC 1492). Similar to RADIUS, it uses ports TCP 49 and UDP 49. XTACACS was the first proprietary Cisco revision of the standard RFC form. TACACS+ was the second major revision by Cisco of this service into yet another proprietary version. None of these three versions of TACACS are compatible with each other. TACACS and XTACACS are utilized on many older systems but have been all but replaced by TACACS+ on current systems.

TACACS+ differs from RADIUS in many ways. One major difference is that RADIUS combines authentication and authorization (the first two As in AAA), whereas TACACS+ separates the two, allowing for more flexibility in protocol selection. For instance, with TACACS+, an administrator may use Kerberos as an authentication mechanism while choosing something entirely different for authorization. With RADIUS these options are more limited.

Scenarios where TACACS+ would be used include any remote access situation where Cisco equipment is present. Cisco hardware is required in order to operate a TACACS+ AAA service for authenticating local or remote systems and users.

CHAP

Challenge Handshake Authentication Protocol (CHAP) is an authentication protocol used over a wide range of Point-to-Point Protocol (PPP) connections (including dial-up, ISDN, DSL, and cable) as a means to provide a secure transport mechanism for logon credentials. It was developed as a secure alternative and replacement for PAP, which transmitted authentication credentials in clear text.

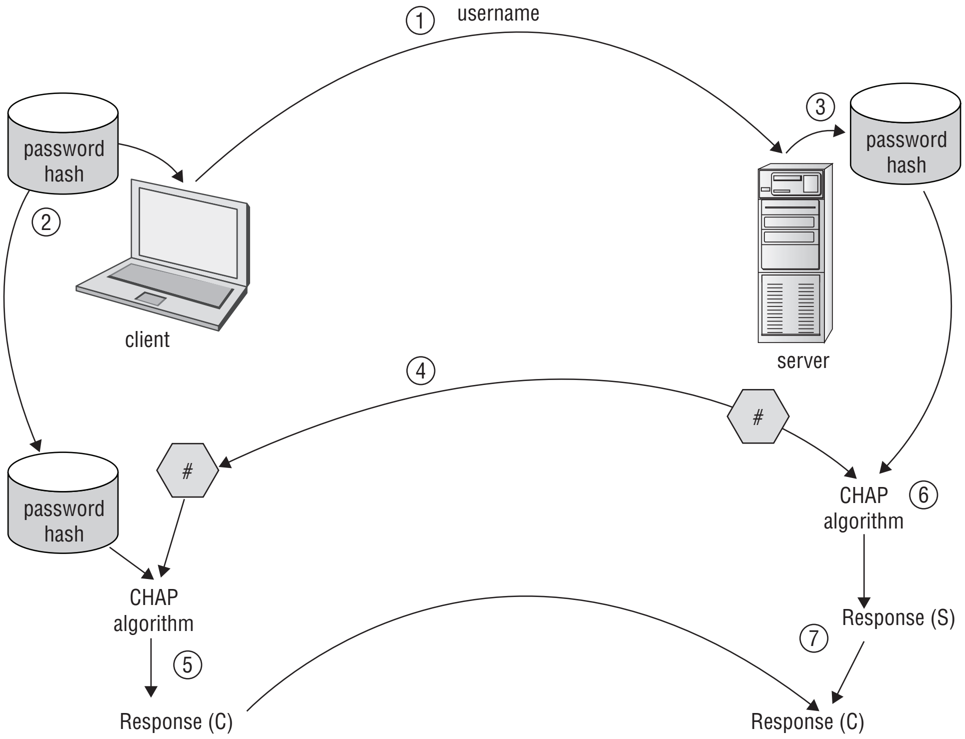
CHAP uses an initial authentication-protection process to support logon and a periodic midstream reverification process to ensure that the subject/client is still who they claim to be. The process is as follows:

1. The user is prompted for their name and password. Only the username is transmitted to the server.
2. The authentication process performs a one-way hash function on the subject's password.
3. The authentication server compares the username to its accounts database to verify that it is a valid existing account.
4. If there is a match, the server transmits a random challenge number to the client.
5. The client uses the password hash and the challenge number as inputs to the CHAP algorithm to produce a response, which is then transmitted back to the server.
6. The server retrieves the password hash from the user account stored in the account database and then, using it along with the challenge number, computes the expected response.
7. The server compares the response it calculated to that received from the client.

If everything matches, the subject is authenticated and allowed to communicate over the connection link. Figure 4.3 shows the CHAP authentication process.

Once the client is authenticated, CHAP periodically sends a challenge to the client at random intervals. The client must compute the correct response to the issued challenge; otherwise, the connection is automatically severed. This post-authentication verification process ensures that the authenticated session hasn't been hijacked.

FIGURE 4.3 CHAP authentication



Whenever a CHAP or CHAP-like authentication system is supported, use it. The only other authentication option that is more secure than CHAP is mutual certificate–based authentication.

PAP

Password Authentication Protocol (PAP) is an insecure plain-text password-logon mechanism. PAP was an early plain old telephone service (POTS) authentication mechanism. PAP is mostly unused today, because it was superseded by CHAP and numerous EAP add-ons. Don't use PAP—it transmits all credentials in plain text.

MSCHAP

MSCHAP is Microsoft's customized or proprietary version of CHAP. The original MSCHAPv1 was integrated into the earliest versions of Windows but was dropped with the release of Windows Vista. MSCHAPv2 was originally added to Windows NT 4.0 through Service Pack 4, as well as Windows 95 and Windows 98 with network update packages. MSCHAPv1 and MSCHAPv2 were both available on Windows NT 4.0 through Windows

XP and Windows Server 2003. MSCHAP is often associated with the Point-to-Point Tunneling Protocol (PPTP), a VPN protocol, and Protected Extensible Authentication Protocol (PEAP). One of the key differences between MSCHAP and CHAP is support for mutual authentication rather than client-only authentication. MSCHAPv2 uses DES encryption to encrypt the transmitted NTML password hash, which is weak and easily cracked. Thus, MSCHAP should generally be avoided and not used in any scenario where other, stronger authentication options are available.

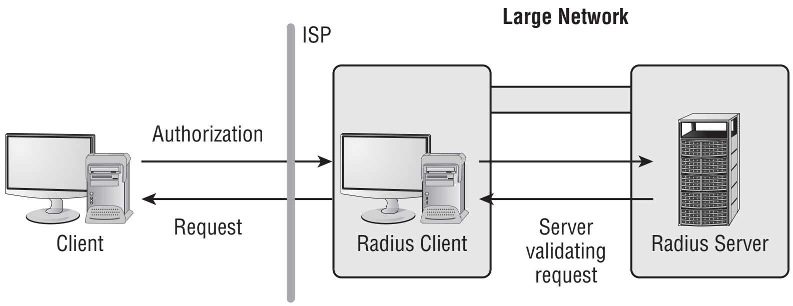
RADIUS

Remote Authentication Dial-In User Service (RADIUS) is a centralized authentication system. It's often deployed to provide an additional layer of security for a network. By offloading authentication of remote access clients from domain controllers or even the remote access server itself to a dedicated authentication server such as RADIUS, you can provide greater protection against intrusion for the network as a whole. RADIUS can be used with any type of remote access, including dial-up, virtual private network (VPN), and terminal services.

RADIUS is known as an *AAA server.* AAA stands for authentication, authorization (or access control), and accounting (sometimes referred to as auditing). RADIUS provides for distinct AAA functions for remote-access clients separate from those of normal local domain clients. RADIUS isn't the only AAA server, but it's the most widely deployed.

When RADIUS is deployed, it's important to understand the terms RADIUS client and RADIUS server, both of which are depicted in Figure 4.4. The *RADIUS server* is obviously the system hosting the RADIUS service. However, the *RADIUS client* is the *remote-access server (RAS),* not the remote system connecting to RAS. As far as the remote-access client is concerned, it sees only the RAS, not the RADIUS server. Thus, the RAS is the RADIUS client. RADIUS is a tried-and-true AAA solution, but alternatives include the Cisco proprietary TACACS+ as well as the direct RADIUS competitor Diameter.

FIGURE 4.4 The RADIUS client manages the local connection and authenticates against a central server.



RADIUS can be used in any remote-access authentication scenario. RADIUS is platform independent and thus does not require any specific vendor's hardware. Most RADIUS products from any vendor are interoperable with all others. RADIUS is a widely supported

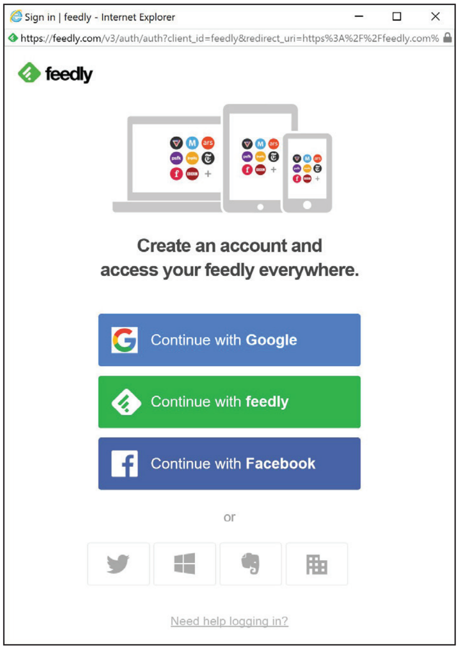
AAA service and can be used as the authentication system for most implementations of IEEE 802.1x (see the section "IEEE 802.1x," later in this chapter), including the ENT authentication option on wireless access points.

SAML

Security Assertion Markup Language (SAML) is an open-standard data format based on XML for the purpose of supporting the exchange of authentication and authorization details between systems, services, and devices. SAML was designed to address the difficulties related to the implementation of single sign-on (SSO) over the web. SAML's solution is based on a trusted third-party mechanism in which the subject or user (the *principle*) is verified through a trusted authentication service (the *identity provider*) in order for the target server or resource host (the *service provider*) to accept the identity of the visitor. SAML doesn't dictate the authentication credentials that must be used, so it's flexible and potentially compatible with future authentication technologies.

SAML is used to create and support federation of authentication. The success of SAML can be seen online wherever you are offered the ability to use an alternate site's authentication to access an account. For example, if you visit `feedly.com` and click the "Get started for free" button, a dialog box appears (Figure 4.5) where you can select to link a new Feedly account to an existing account at Google, Facebook, Twitter, Windows, Evernote, or your own company's enterprise authentication if you don't want to use a unique email and password for Feedly.

FIGURE 4.5 An example of a SAML/OAuth single sign-on interface



Sign in | feedly - Internet Explorer
https://feedly.com/v3/auth/auth?client_id=feedly&redirect_uri=https%3A%2F%2Ffeedly.com%
feedly

Create an account and access your feedly everywhere.

Continue with Google
Continue with feedly
Continue with Facebook
or
Need help logging in?

SAML should be used in any scenario where linking of systems, services, or sites is desired but the authentication solutions are not already compatible. SAML allows for the creation of interfaces between authentication solutions in order to allow federation.

OpenID Connect

OpenID Connect is an Internet-based single sign-on solution. It operates over the OAuth protocol (see next section) and can be used in relation to Web services as well as smart device apps. The purpose or goal of OpenID Connect is to simplify the process by which applications are able to identify and verify users. For more detailed information and programming guidance, please see `https://openid.net/connect/`.

OpenID Connect should be considered for use as an authentication solution for any online application or web service. It is a simple solution that may be robust enough for your soon-to-be virally popular digital app.

OAuth

OAuth is an open standard for authentication and access delegation (federation). OAuth is widely used by websites, web services, and mobile device applications. OAuth is an easy means of supporting federation of authentication between primary and secondary systems. A primary system could be Google, Facebook, or Twitter, and secondary systems are anyone else. OAuth is often implemented using SAML. OAuth can be recognized as being in use when you are offered the ability to use an existing authentication from a primary service as the authentication at a secondary one (see Figure 4.5 earlier).

OAuth can be used in any scenario where a new, smaller secondary entity wants to employ the access tokens from primary entities as a means of authentication. In other words, OAuth is used to implement authentication federation.

Shibboleth

Shibboleth is another example of an authentication federation and single sign-on solution. Shibboleth is a standards-based open-source solution that can be used for website authentication across the Internet or within private networks. Shibboleth was developed for use by Internet2 and is now available for use in any networking environment, public or private. Shibboleth is based on SAML.

For more information on Shibboleth, please visit `https://shibboleth.net/`.

Secure token

A secure token is a protected, possibly encrypted authentication data set that proves a particular user or system has been verified through a formal logon procedure. Access tokens include web cookies, Kerberos tickets, and digital certificates. A secure token is an access token that does not leak any information about the subject's credentials or allow for easy impersonation.

A secure token can also refer to a physical authentication device known as a TOTP or HOTP device (see the "HOTP/TOTP" section later in this chapter).

Secure tokens should be considered for use in any private or public authentication scenario. Minimizing the risk of information leakage or impersonation should be a goal of anyone designing, establishing, or managing authentication solutions.

NTLM

New Technology LAN Manager (NTLM) is a password hash storage system used on Microsoft Windows. NTLM exists in two versions. NTLMv1 is a challenge-response protocol system that, using a server-issued random challenge along with the user's password (in both LM hash and MD4 hash), produces two responses that are sent back to the server (this is assuming a password with 14 or fewer characters; otherwise only an MD4 hash-based response is generated). NTLMv2 is also a challenge-response protocol system, but it uses a much more complex process that is based on MD5. Both versions of NTLM use a challenge response–based hashing mechanism whose result is nonreversible and thus much more secure than LM hashing. However, reverse-engineering password-cracking mechanisms can ultimately reveal NTLMv1 or v2 stored passwords if the passwords are relatively short (under 15 characters) and the hacker is given enough processing power and time.

LANMAN, or what is typically referred to as LM or LAN Manager, is a legacy storage mechanism developed by Microsoft to store passwords. LM was replaced by NTLM on Windows NT 4.0 and should be disabled (usually left disabled) and avoided on all current versions of Windows.

One of the most significant issues with LM is that it limited passwords to a maximum of 14 characters. Shorter passwords were padded out to 14 characters using null characters. The 14 characters of the password were converted to uppercase and then divided into two seven-character sections. Each seven-character section was then used as a DES encryption key to encrypt the static ASCII string "KGS!@#$%". The two results were recombined to form the LM hash. Obviously, this system is fraught with problems. Specifically, the process is reversible and not truly a one-way hash, and all passwords are ultimately no stronger than seven characters.

As a user, you can completely avoid LM by using passwords of at least 15 characters. LM has been disabled by default on all versions of Windows since Windows 2000. However, this disabling only addresses the initial request for and the default transmission of LM for the authentication process. The Security Accounts Manager (SAM) still contains an LM equivalent of all passwords with 14 or fewer characters through Windows Vista, at least by default. Windows 7 and later versions of Windows do not even create the LM version of user passwords to store in the user account database by default. Settings are available in the Registry and Group Policy Objects to turn on this backward-compatibility feature.

You should leave LM disabled and disable it when it isn't. If you need LM to support a legacy system, you should find a way to upgrade the legacy system rather than continue to use LM. The use of LM is practically equivalent to using only plain text.

NTLM is used in nearly every scenario of Windows-to-Windows authentication. Although it is not the most robust or secure form of authentication, it is secure enough in most circumstances. When NTLM is deemed insufficient or incompatible (such as when connecting to non-Windows systems), then digital certificate-based authentication should be used.

Exam Essentials

Understand Kerberos. Kerberos is a trusted third-party authentication protocol. It uses encryption keys as tickets with time stamps to prove identity and grant access to resources. Kerberos is a single sign-on solution employing a key distribution center (KDC) to manage its centralized authentication mechanism.

Know about TACACS+. TACACS is a centralized remote access authentication solution. It's an Internet standard (RFC 1492); however, Cisco's proprietary implementations of XTACACS and now TACACS+ have quickly gained popularity as RADIUS alternatives.

Understand CHAP. The Challenge Handshake Authentication Protocol (CHAP) is an authentication protocol used primarily over dial-up connections (usually PPP) as a means to provide a secure transport mechanism for logon credentials. CHAP uses a one-way hash to protect passwords and periodically reauthenticate clients. A good example of CHAP usage is a point-to-point link between two corporate routers.

Define PAP. Password Authentication Protocol (PAP) is an insecure plain-text password-logon mechanism. PAP was an early plain old telephone service (POTS) authentication mechanism.

Understand MSCHAP. MSCHAP is Microsoft's customized or proprietary version of CHAP. One of the key differences between MSCHAP and CHAP is that MSCHAP supports mutual authentication, rather than client-only authentication. MSCHAPv2 uses DES encryption to encrypt the transmitted NTML password hash, which is weak and easily cracked.

Comprehend RADIUS. RADIUS is a centralized authentication system. It's often deployed to provide an additional layer of security for a network.

Understand SAML. Security Assertion Markup Language is an open-standard data format based on XML for the purpose of supporting the exchange of authentication and authorization details between systems, services, and devices.

Know about OpenID Connect. OpenID Connect is an Internet-based single sign-on solution. It operates over the OAuth protocol and can be used in relation to web services as well as smart device apps.

Understand OAuth. OAuth is an open standard for authentication and access delegation (federation). OAuth is widely used by websites/services and mobile device applications.

Define Shibboleth. Shibboleth is another example of an authentication federation and single sign-on solution. Shibboleth is a standards-based, open source-solution that can be used for website authentication across the Internet or within private networks.

Understand secure tokens. A secure token is a protected, possibly encrypted authentication data set that proves a particular user or system has been verified through a formal logon procedure. Access tokens include web cookies, Kerberos tickets, and digital certificates. A secure token is an access token that does not leak any information about the subject's credentials or allow for easy impersonation.

Know about NTLM. New Technology LAN Manager (NTLM) is a password hash storage system used on Microsoft Windows. It's a challenge-response protocol system that is nonreversible and thus much more secure than LM hashing. One place where NTLM is frequently used is in Microsoft Active Directory for user logon authentication in lieu of a RADIUS or TACACS solution.

4.3 Given a scenario, implement identity and access management controls.

Authorization is the second element of AAA services. Thus, authorization or access control is an essential part of security through an organization. Understanding the variations and options for identity verification and access control management is important for security management.

Access control models

The mechanism by which users are granted or denied the ability to interact with and use resources is known as *access control*. Access control is often referred to using the term *authorization*. Authorization defines the type of access to resources that users are granted—in other words, what users are authorized to do. Authorization is often considered the next logical step immediately after authentication. *Authentication* is proving your identity to a system or the act of logging on. With proper authorization or access control, a system can properly control access to resources in order to prevent unauthorized access.

There are three common access control methods:

- Mandatory access control (MAC)
- Discretionary access control (DAC)
- Role-based access control (RBAC)

These three models are widely used in today's IT environments. Familiarity with these models is essential for the Security+ exam.

In most environments, DAC is a sufficient authorization mechanism to use to control and manage a subject's access to and use of resources. Most operating systems are DAC by default. In government or military environments, where classifications are deemed an essential control mechanism, MAC should be used to directly enforce and restrict access based on a subject's clearance. RBAC is a potential alternative in many environments, but it is most appropriate in those situations where there is a high rate of employee turnover.

MAC

Mandatory access control (MAC) is a form of access control commonly employed by government and military environments. MAC specifies that access is granted based on a set of rules rather than at the discretion of a user. The rules that govern MAC are hierarchical in nature and are often called *sensitivity labels*, *security domains*, or *classifications*. MAC environments define a few specific security domains or sensitivity levels and then use the associated labels from those domains to impose access control on objects and subjects.

A government or military implementation of MAC typically includes the following five levels (in order from least sensitive to most sensitive):

- Unclassified
- Sensitive but unclassified
- Confidential
- Secret
- Top secret

Objects or resources are assigned sensitivity labels corresponding to one of these security domains. Each specific security domain or level defines the security mechanisms and restrictions that must be imposed in order to provide protection for objects in that domain.

MAC can also be deployed in private sector or corporate business environments. Such cases typically involve the following four security domain levels (in order from least to most sensitive):

- Public
- Sensitive
- Private
- Confidential/Proprietary

The primary purpose of a MAC environment is to prevent *disclosure*: the violation of the security principle of *confidentiality*. When an unauthorized user gains access to a secured resource, it is a security violation. Objects are assigned a specific sensitivity label based on the damage that would be caused if disclosure occurred. For example, if a top-secret resource was disclosed, it could cause grave damage to national security.

A MAC environment works by assigning subjects a *clearance level* and assigning objects a *sensitivity label*—in other words, everything is assigned a classification marker. The name of the clearance level is the same as the name of the sensitivity label assigned to *objects* or resources. A person (or other subject, such as a program or a computer system) must have the same or greater assigned clearance level as the resources they wish to access. In this manner, access is granted or restricted based on the rules of classification (that is, sensitivity labels and clearance levels).

MAC is so named because the access control it imposes on an environment is mandatory. Its assigned classifications and the resulting granting and restriction of access can’t be altered by users. Instead, the rules that define the environment and judge the assignment of sensitivity labels and clearance levels control authorization.

MAC isn’t a security environment with very granular control. An improvement to MAC includes the use of *need to know*: a security restriction in which some objects (resources or data) are restricted unless the subject has a need to know them. The objects that require a specific need to know are assigned a sensitivity label, but they're compartmentalized from the rest of the objects with the same sensitivity label (in the same security domain). The need to know is a rule in itself, which states that access is granted only to users who have been assigned work tasks that require access to the cordoned-off object. Even if users have the proper level of clearance, without need to know, they're denied access. "Need to know" is the MAC equivalent of the principle of least privilege from DAC (described in the following section).

DAC

Discretionary access control (DAC) is the form of access control or authorization that is used in most commercial and home environments. DAC is user-directed or, more specifically, controlled by the owner and creators of the objects (resources) in the environment. DAC is identity-based: access is granted or restricted by an object's owner based on user identity and on the discretion of the object owner. Thus, the owner or creator of an object can decide which users are granted or denied access to their object. To do this, DAC uses ACLs.

An *access control list (ACL)* is a security logical mechanism attached to every object and resource in the environment. It defines which users are granted or denied the various types of access available based on the object type. Individual user accounts or user groups can be added to an object's ACL and granted or denied access.

If your user account isn't granted access through an object's ACL, then often your access is denied by default (note: not all OSs use a deny-by-default approach). If your user account is specifically granted access through an object's ACL, then you're granted the specific level or type of access defined. If your user account is specifically denied access through an object's ACL, then you're denied the specific level or type of access defined. In some cases (such as with Microsoft Windows), a Denied setting in an ACL overrides all other settings. Table 4.1 shows an access matrix for a user who is a member of three groups, and the resulting access to specific files within a folder on a network server. As you can see, the presence of the Denied setting overrides any other access granted from another group. Thus, if your membership in one user group grants you write access over an object, but another group specifically denies you write access to the same object, then you're denied write access to the object.

TABLE 4.1 Cumulative access based on group memberships

Sales Group	User group	Research group	Resulting access	Filename
Change	Read	None specified	Change	`SalesReport.xls`
Read	Read	Change	Change	`ProductDevelopment.doc`
None specified	Read	Denied	Denied	`EmailPolicy.pdf`
Full control	Denied	None specified	Denied	`CustomerContacts.doc`

User-assigned privileges are permissions granted or denied on a specific individual user basis. This is a standard feature of DAC-based OSs, including Linux and Windows. All objects in Linux have an owner assigned. The owner (an individual) is granted specific privileges. In Windows, an access control entry (ACE) in an ACL can focus on an individual user to grant or deny permissions on the object.

In a DAC environment, it is common to use groups to assign access to resources in aggregate rather than only on an individual basis. This often results in users being members of numerous groups. In these situations, it is often important to determine the effective permissions for a user. This is accomplished by accumulating all allows or grants of access to a resource, and then subtracting or removing any denials for that resource.

ABAC

Attribute-based access control (ABAC) is a mechanism for assigning access and privileges to resources through a scheme of attributes or characteristics. The attributes can be related to the user, the object, the system, the application, the network, the service, time of day, or even other subjective environmental concerns. ABAC access is then determined through a set of Boolean logic rules, similar to if-then programming statements, that relate who is making a request, what the object is, what type of access is being sought, and results the action would cause. ABAC is a dynamic, context-aware authorization scheme that can modify access based on risk profiles and changing environmental conditions (such as system load, latency, whether or not encryption is in use, and whether the requesting system has the latest security patches). ABAC is also known by the terms policy-based access control (PBAC) and claims-based access control (CBAC).

Role-based access control

Role-based access control (RBAC) is another strict form of access control. It may be grouped with the nondiscretionary access control methods along with MAC. The rules used for RBAC are basically job descriptions: users are assigned a specific role in an environment, and access to objects is granted based on the necessary work tasks of that role. For example, the role of backup operator may be granted the ability to back up every file on a system to a tape drive. The user given the backup operator role can then perform that function.

RBAC is most suitable for environments with a high rate of employee turnover. It allows a job description or role to remain static even when the user performing that role changes often. It's also useful in industries prone to privilege creep, such as banking.

Rule-based access control

Rule-based access control (RBAC or rule-BAC) is typically used in relation to network devices that filter traffic based on filtering rules, as found on firewalls and routers. Rule-based access control (RBAC) systems enforce rules independent of the user or the resource, as the rules are the rules. If a firewall rule sets a port as closed, then it is closed regardless of who is attempting to access the system. These filtering rules are often called *rules*, *rule sets*, *filter lists*, *tuples*, or ACLs. Be sure you understand the context of the Security+ exam question before assuming *role* or *rule* when you see RBAC.

NOTE

On the exam, CompTIA will either spell out confusing acronyms or will provide context to indicate the intended meaning of an ambiguous acronym.

Physical access control

Often overlooked when considering IT security is the need to manage physical access. Physical access controls are needed to restrict physical access violations, whereas logical access controls are needed to restrict logical access violations.

Physical access controls should be implemented in any scenario in which there is a difference in value, risk, or use between one area of a facility and another. Any place where it would make sense to have a locked door, technology-managed physical access controls should be implemented.

Proximity cards

In addition to smart and dumb cards, proximity devices can be used to control physical access. A *proximity device* or *proximity card* can be a passive device, a field-powered device, or a transponder. The proximity device is worn or held by the authorized bearer. When it passes a proximity reader, the reader is able to determine who the bearer is and whether they have authorized access. A passive device reflects or otherwise alters the electromagnetic field generated by the reader. This alteration is detected by the reader.

The passive device has no active electronics; it is just a small magnet with specific properties (like antitheft devices commonly found on DVDs). A field-powered device has electronics that activate when the device enters the electromagnetic (EM) field that the reader generates. Such devices generate electricity from an EM field to power themselves (such as card readers that only require the access card be waved within inches of the reader to unlock doors). A transponder device is self-powered and transmits a signal received by the reader. This can occur continuously or only at the press of a button (like a garage door opener or car alarm key fob).

In addition to smart/dumb cards and proximity readers, physical access can be managed with radio frequency identification (RFID) or biometric access-control devices.

Smart cards

See the later section "PIV/CAC/smart card" in the discussion of implementing certificate-based authentication.

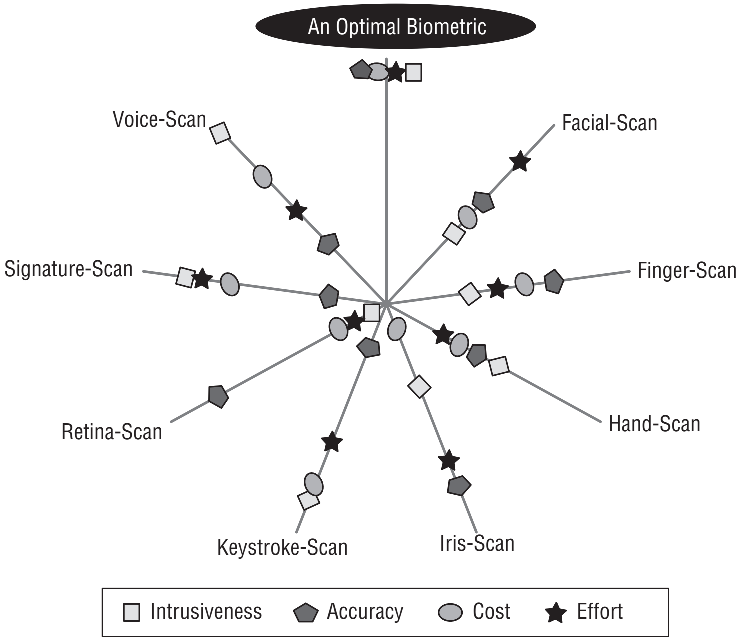
Biometric factors

Biometrics is the term used to describe the collection of physical attributes of the human body that can be used as an identification or authentication factor. Biometrics fall into the authentication factor category of something you are: you, as a human, have the element of identification as part of your physical body.

Numerous biometric factors can be considered for identification and authentication purposes. Several of these options are discussed in the following sections.

However, when an organization decides to implement a biometric factor, it is important to evaluate the available options in order to select a biometric solution that is most in line with the organization's security priorities. One method to accomplish this is to consult a Zephyr analysis chart (Figure 4.6). This type of chart presents the relative strengths and weaknesses of various characteristics of biometric factor options. The specific example shown in Figure 4.6 evaluates eight biometric types on four characteristics (intrusiveness, accuracy, cost, and effort). The security administrator should select a form of biometric based on their organization's priorities for the evaluated characteristics.

FIGURE 4.6 A Zephyr analysis chart



Once the type of biometric is selected, then a specific make and model needs to be purchased. Finding the most accurate device to implement is accomplished using a crossover error rate analysis (see the section "Crossover error rate" later in this chapter).

Biometric factor devices or biometric scanners should be used as an element in multifactor authentication. Any scenario in which there is sensitive data carries a corresponding need for greater security. One element of stronger security is more robust authentication. Any form of multifactor authentication is stronger than a single-factor authentication solution.

Fingerprint scanner

A fingerprint scanner is used to analyze the visible patterns of skin ridges on the fingers and thumbs of people. Fingerprints are thought to be unique to an individual and have been used for decades in physical security for identification, and are now often used as an electronic authentication factor as well. Fingerprint readers are now commonly used on laptop computers, smartphones, and USB flash drives as a method of identification and authentication. Although fingerprint scanners are common and seemingly easy to use, they can sometimes be fooled by photos of fingerprints, black-powder and tape-lifted fingerprints, or gummy re-creations of fingerprints.

Retinal scanner

Retinal scanners focus on the pattern of blood vessels at the back of the eye. Retinal scans are the most accurate form of biometric authentication and are able to differentiate between identical twins. However, they are the least acceptable biometric scanning method for employees because they can reveal medical conditions, such as high blood pressure and pregnancy. Older retinal scans blew a puff of air into the user's eye (which is uncomfortable), but newer ones typically use an infrared light instead. Retinal patterns can also change as people age and retinas deteriorate.

Iris scanner

Iris scanners focus on the colored area around the pupil. They are the second most accurate form of biometric authentication. Iris scans are often recognized as having a longer useful authentication life span than other biometric factors because the iris remains relatively unchanged throughout a person's life (barring eye damage or illness). Iris scans are considered more acceptable by general users than retina scans because they don't reveal personal medical information. However, some scanners can be fooled with a high-quality image in place of an actual person's eye; sometimes a contact lens can be placed on the photo to improve the subterfuge. Additionally, accuracy can be affected by changes in lighting.

Voice recognition

Voice recognition is a type of biometric authentication that relies on the characteristics of a person's speaking voice, known as a voiceprint. The user speaks a specific phrase, which is recorded by the authentication system. To authenticate, the user repeats the same phrase and it is compared to the original. Voice pattern recognition is sometimes used as an additional authentication mechanism but is rarely used by itself.

Facial recognition

Facial recognition is based on the geometric patterns of faces for detecting authorized individuals. Face scans are used to identify and authenticate people before accessing secure spaces, such as a secure vault. Many photo sites now include facial recognition, which can automatically recognize and tag individuals once they have been identified in other photos.

False acceptance rate

As with all forms of hardware, there are potential errors associated with biometric readers. Two specific error types are a concern: *false rejection rate (FRR)* or *Type I* errors and *false* *acceptance rate (FAR)* or *Type II* errors. The FRR is the number of failed authentications for valid subjects based on device sensitivity, whereas the FAR is the number of accepted invalid subjects based on device sensitivity.

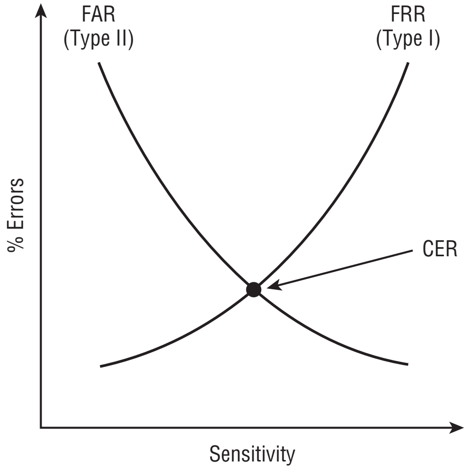
False rejection rate

Discussed in the previous section.

Crossover error rate

The two error measurements of biometric devices (FRR and FAR) can be mapped on a graph comparing sensitivity level to rate of errors. The point on this graph where these two rates intersect is known as the *crossover error rate (CER)*; see Figure 4.7. Notice how the number of FRR errors increases with sensitivity, whereas FAR errors decrease with an increase in sensitivity. The CER point (as measured against the error scale) is used to determine which biometric device for a specific body part from various vendors or of various models is the most accurate. The comparatively lowest CER point is the more accurate biometric device for the relevant body part.

FIGURE 4.7 A graphing of FRR and FAR, which reveals the CER



Tokens

A *token* is a form of authentication factor that is something you have. It's usually a hardware device, but it can be implemented in software as a logical token. A token is used to generate temporary single-use passwords for the purpose of creating stronger authentication. In this way, a user account isn't tied to a single static password. Instead, the user must be in physical possession of the password-generating device. Users enter the currently valid password from the token as their password during the logon process.

There are several forms of tokens. Some tokens generate passwords based on time (see Figure 4.8 later in the chapter), whereas others generate passwords based on challenges from the authentication server. In either case, users can use (or attempt to use) the generated password just once before they must either wait for the next time window or request another challenge. Passwords that can be used only once are known as *one-time passwords (OTP)*. This is the most secure form of password, because regardless of whether its use results in a successful logon, that one-use password is never valid again. One-time passwords can be employed only when a token is used, due to the complexity and ever-changing nature of the passwords. However, a token need not be a device; there are paper-based options as well as smartphone app–based solutions.

A token may be a device (Figure 4.8), like a small calculator with or without a keypad. It may also be a high-end smartcard. When properly deployed, a token-based authentication system is more secure than a password-only system.

A token should be used in any scenario in which multifactor authentication is needed or warranted. Almost every authentication event would be improved by implementing a multifactor solution as opposed to remaining with single-factor authentication.

Hardware

An authentication token can be a hardware device that must be present each time the user attempts to log on. Often hardware tokens are designed to be small and attach to a keychain or lanyard. They are often referred to as keychain tokens or key fobs.

Software

An authentication token can be a software solution, such as an app on a smart device. Since many of us carry a smartphone with us almost everywhere we go, having an app that provides OTP when necessary can eliminate the need for carrying around another hardware device or physical token. Software token apps are widely available and implemented on many Internet services; thus, they are easy to adopt for use as an authentication factor for a private network.

HOTP/TOTP

HMAC-based one-time password (HOTP) tokens, or asynchronous dynamic password tokens, are devices or applications that generate passwords based not on fixed time intervals but on a nonrepeating one-way function, such as a hash or hash message authentication code (HMAC—a type of hash that uses a symmetric key in the hashing process) operation. These tokens often generate a password after the user enters a PIN into the token device. The authentication process commonly includes a challenge and a response in which a server sends the user a PIN and the user enters the PIN to create the password. These tokens have a unique seed (or random number) embedded along with a unique identifier for the device. See the earlier section "CHAP" for a description of this operation.

There is a potential downside to using HOTPs, known as the *off-by-one problem*. If the non–time-based seed or key synchronization gets desynchronized, the client may be calculating a value that the server has already tossed or has not yet generated. This requires the device to be resynced with the authentication server.

Time-based one-time password (TOTP) tokens, or synchronous dynamic password tokens, are devices or applications that generate passwords at fixed time intervals, such as every 60 seconds. Time-interval tokens must have their clocks synchronized to an authentication server. To authenticate, the user enters the password shown along with a PIN or passphrase as a second factor of authentication. The generated one-time password provides identification, and the PIN/passphrase provides authentication.

Certificate-based authentication

Certificates or digital certificates are a trusted third-party authentication technology derived from asymmetric public key cryptography. Please see Chapter 6 for detailed coverage of digital certificates.

Certificate-based authentication is often a reliable mechanism for verifying the identity of devices, systems, services, applications, networks, and organizations. However, certificates alone are insufficient to identify or authenticate individuals, since the certificate is a digital file and can be lost, stolen, or otherwise abused for impersonation attacks. However, when implemented as a multifactor authentication process, certificates can be a significant improvement in logon security for a wide range of scenarios.

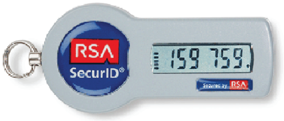
Certificates Might Not Be Usable as a Second Factor!

Recent PCI guidance has clarified the use of certificates as multifactor authentication and points out that a certificate-based second factor (something you have) doesn't properly function as that factor if it is only protected by a username and password. See `https://blog.pcisecuritystandards.org/understanding-new-pci-guidance-on-mfa` for more.

PIV/CAC/smart card

Smartcards (Figure 4.8) are credit card–sized IDs, badges, or security passes with embedded integrated circuit chips. They can contain information about the authorized bearer that can be used for identification and/or authentication purposes. Some smartcards can even process information or store reasonable amounts of data in a memory chip. Many smartcards are used as the means of hardware-based removable media storage for digital certificates. This enables users to carry a credit card–sized device on their person, which is then used as an element in multifactor authentication, specifically supporting certificate authentication as one of those factors.

FIGURE 4.8 The RSA SecurID token device



A smartcard may be known by several terms:

- An identity token containing integrated circuits (ICs)
- A processor IC card
- An IC card with an ISO 7816 interface

Smartcards are often viewed as a complete security solution, but they should not be considered complete by themselves. Like any single security mechanism, smartcards are subject to weaknesses and vulnerabilities. They can fall prey to physical attacks, logical attacks, Trojan horse attacks, or social engineering attacks.

Memory cards are machine-readable ID cards with a magnetic strip or a read-only chip, like a credit card, a debit card, or an ATM card. Memory cards can retain a small amount of data but are unable to process data like a smartcard. Memory cards often function as a type of two-factor control: the card is something you have, and its PIN is something you know. However, memory cards are easy to copy or duplicate and are insufficient for authentication purposes in a secure environment.

The *Common Access Card (CAC)* is the name given to the smartcard used by the U.S. government and military for authentication purposes. Although the CAC name was assigned by the Department of Defense (DoD), the same technology is widely used in commercial environments. This smartcard is used to host credentials, specifically digital certificates, that can be used to grant access to a facility or to a computer terminal.

Personal identification verification (PIV) cards, such as badges, identification cards, and security IDs, are forms of physical identification and/or electronic access control devices. A badge can be as simple as a name tag indicating whether you're a valid employee or a visitor. Or it can be as complex as a smartcard or token device that employs multifactor authentication to verify and prove your identity and provide authentication and authorization to access a facility, specific rooms, or secured workstations. Badges often include pictures, magnetic strips with encoded data, and personal details to help a security guard verify identity.

Badges can be used in environments in which physical access is primarily controlled by security guards. In such conditions, the badge serves as a visual identification tool for the guards. They can verify your identity by comparing your picture to your person and consult a printed or electronic roster of authorized personnel to determine whether you have valid access.

Badges can also serve in environments guarded by scanning devices rather than security guards. In such conditions, a badge can be used either for identification or for authentication. When a badge is used for identification, it's swiped in a device, and then the badge owner must provide one or more authentication factors, such as a password, passphrase, or biological trait (if a biometric device is used). When a badge is used for authentication, the badge owner provides an ID, username, and so on, and then swipes the badge to authenticate.

IEEE 802.1x

IEEE 802.1x is a port-based authentication mechanism. It's based on Extensible Authentication Protocol (EAP) and is commonly used in closed-environment wireless networks. However, 802.1x isn't exclusively used on wireless access points (WAPs); it can also be used on firewalls, proxies, VPN gateways, and other locations where an authentication handoff service is desired. Think of 802.1x as an authentication proxy. When you wish to use an existing authentication system rather than configure another, 802.1x lets you do that.

When 802.1x is in use, it makes a port-based decision about whether to allow or deny a connection based on the authentication of a user or service. 802.1x was initially used to compensate for the weaknesses of Wired Equivalent Privacy (WEP), but today it's often used as a component in more complex authentication and connection-management systems, including Remote Authentication Dial-In User Service (RADIUS), Diameter, Cisco System's Terminal Access Controller Access-Control System Plus (TACACS+), and Network Access Control (NAC).

Like many technologies, 802.1x is vulnerable to man-in-the-middle and hijacking attacks because the authentication mechanism occurs only when the connection is established.

802.1x is a standard port-based network-access control that ensures that clients can't communicate with a resource until proper authentication has taken place. Effectively, 802.1x is a handoff system that allows any device to use the existing network infrastructure's authentication services. Through the use of 802.1x, other techniques and solutions such as RADIUS, TACACS, certificates, smartcards, token devices, and biometrics can be integrated into any communications system. 802.1x is most often associated with wireless access points, but its use isn't limited to wireless.

File system security

Filesystem security is usually focused on authorization instead of authentication. To protect a filesystem, either access to the computer through which the storage device is accessed needs to be locked down in order to deny access to anyone not specifically authorized (such as using multifactor authentication), *or* the storage device should be encrypted to block access to all but the intentionally authorized. See the section "Access control models" earlier in this chapter for details on authorization control via DAC, MAC, RBAC, and others.

Filesystem security should be used in all scenarios to define what access users have. Such access should be granted based on their work responsibilities in order to enable users to complete work tasks without placing the organization at any significant level of additional and unwarranted risk. This concept is known as the principle of least privilege, and it should be adopted and enforced across all means of resource access management.

Database security

Database security is an important part of any organization that uses large sets of data as an essential asset. Without database security efforts, business tasks can be interrupted and confidential information disclosed. The wide array of topics that are part of database security includes aggregation, inference, aggregation, data mining, data warehousing, and data analytics.

Structured Query Language (SQL), the language used to interact with most databases, provides a number of functions that combine records from one or more tables to produce potentially useful information. This process, known as aggregation, is not without its security vulnerabilities. Aggregation attacks are used to collect numerous low-level security items or low-value items and combine them to create something of a higher security level or value.

For example, suppose a low-level military records clerk is responsible for updating records of personnel and equipment as they are transferred from base to base. As part of his duties, this clerk may be granted the database permissions necessary to query and update personnel tables.

The military might not consider an individual transfer request (in other words, Sergeant Jones is being moved from Base X to Base Y) to be classified information. The records clerk has access to that information because he needs it to process Sergeant Jones's transfer. However, with access to aggregate functions, the records clerk might be able to count the number of troops assigned to each military base around the world. These force levels are often closely guarded military secrets, but the low-ranking records clerk could deduce them by using aggregate functions across a large number of unclassified records.

For this reason, it's especially important for database security administrators to strictly control access to aggregate functions and adequately assess the potential information they may reveal to unauthorized individuals.

The database security issues posed by inference attacks are very similar to those posed by the threat of data aggregation. Inference attacks involve combining several pieces of nonsensitive information to gain access to information that should be classified at a higher level. However, inference makes use of the human mind's deductive capacity rather than the raw mathematical ability of modern database platforms.

A commonly cited example of an inference attack is that of the accounting clerk at a large corporation who is allowed to retrieve the total amount the company spends on salaries for use in a top-level report but is not allowed to access the salaries of individual employees. The accounting clerk often has to prepare those reports with effective dates in the past and so is allowed to access the total salary amounts for any day in the past year. Say, for example, that this clerk must also know the hiring and termination dates of various employees and has access to this information. This opens the door for an inference attack. If an employee was the only person hired on a specific date, the accounting clerk can now retrieve the total salary amount on that date and the day before and deduce the salary of that particular employee—sensitive information that the user would not be permitted to access directly.

As with aggregation, the best defense against inference attacks is to maintain constant vigilance over the permissions granted to individual users. Furthermore, intentional blurring of data may be used to prevent the inference of sensitive information. For example, if the accounting clerk were able to retrieve only salary information rounded to the nearest million, he would probably not be able to gain any useful information about individual employees. Finally, you can use database partitioning, dividing up a single database into multiple distinct databases according to content value, risk, and importance, to help subvert these attacks.

Many organizations use large databases, known as data warehouses (a predecessor to the idea of big data), to store large amounts of information from a variety of databases for use with specialized analysis techniques. These data warehouses often contain detailed historical information not normally stored in production databases because of storage limitations or data security concerns.

A data dictionary is commonly used for storing critical information about data, including usage, type, sources, relationships, and formats. Database management software (DBMS) reads the data dictionary to determine access rights for users attempting to access data.

Data mining techniques allow analysts to comb through data warehouses and look for potential correlated information. For example, an analyst might discover that the demand for lightbulbs always increases in the winter months and then use this information when planning pricing and promotion strategies. Data mining techniques result in the development of data models that can be used to predict future activity.

The activity of data mining produces metadata—information about data. Metadata is not exclusively the result of data mining operations; other functions or services can produce metadata as well. Think of metadata from a data mining operation as a concentration of data. It can also be a superset, a subset, or a representation of a larger data set. Metadata can be the important, significant, relevant, abnormal, or aberrant elements from a data set.

One common security example of metadata is that of a security incident report. An incident report is the metadata extracted from a data warehouse of audit logs through the use of a security auditing data mining tool. In most cases, metadata is of a greater value or sensitivity (due to disclosure) than the bulk of data in the warehouse. Thus, metadata is stored in a more secure container known as the data mart.

Data warehouses and data mining are significant to security professionals for two reasons. First, as previously mentioned, data warehouses contain large amounts of potentially sensitive information vulnerable to aggregation and inference attacks, and security practitioners must ensure that adequate access controls and other security measures are in place to safeguard this data. Second, data mining can actually be used as a security tool when it's used to develop baselines for statistical anomaly–based intrusion detection systems.

Data analytics is the science of raw data examination with the focus of extracting useful information out of the bulk information set. The results of data analytics could focus on important outliers or exceptions to normal or standard items, a summary of all data items, or some focused extraction and organization of interesting information. Data analytics is a growing field as more organizations are gathering an astounding volume of data from their customers and products. The sheer volume of information to be processed has demanded a whole new category of database structures and analysis tools. It has even picked up the nickname of "big data."

Big data refers to collections of data that have become so large that traditional means of analysis or processing are ineffective, inefficient, and insufficient. Big data involves numerous difficult challenges, including collection, storage, analysis, mining, transfer, distribution, and results presentation. Such large volumes of data have the potential to reveal nuances and idiosyncrasies that more mundane sets of data fail to address. The potential to learn from big data is tremendous, but the burdens of dealing with big data are equally great. As the volume of data increases, the complexity of data analysis increases as well. Big data analysis requires high-performance analytics running on massively parallel or distributed processing systems. With regard to security, organizations are endeavoring to collect an ever more detailed and exhaustive range of event data and access data. This data is collected with the goal of assessing compliance, improving efficiencies, improving productivity, and detecting violations.

A relational database is a means to organize and structure data in a flat two-dimensional table. The row and column–based organizational scheme is widely used, but it isn't always the best solution. Relational databases can become difficult to manage and use when they grow extremely large, especially if they're poorly designed and managed. Their performance can be slowed when significant numbers of simultaneous users perform queries. And they might not support data mapping needed by modern complex programming techniques and data structures. In the past, most applications of RDBMSs did not experience any of these potential downsides. However, in today's era of big data and services the size of Google, Amazon, Twitter, and Facebook, RDBMSs aren't sufficient solutions to some data-management needs.

NoSQL is a database approach that employs nonrelational data structures, such as hierarchies or multilevel nesting and referencing. A *hierarchical data structure* is one in which every data object can have a single data-parent relation and none, one, or many data-child relations. A *data parent* is an item upward or closer to the root of the hierarchy, whereas a *data child* is an item downward or further away from the root. DNS and XML data are excellent examples of hierarchical data structures.

A *multilevel nesting and cross-referencing data structure* is a system in which a data object can have multiple data-parent and data-child links and may even have links across multiple levels or among "peer" data items. Effectively, any data item can be linked to any other data item, with no structural limitations. The organization of Facebook, Twitter, and Google+ relationships is of this nature. This DBMS structure is also known as a *distributed database model*.

NoSQL databases or SQL databases? That is a common argument waged between DBMS managers and database programmers alike. However, using the term SQL here isn't entirely accurate, because SQL is a means to interact with a database rather than a form or type of database. More specifically, the comparison is between relational database management systems (RDBMSs) and nonrelational databases. Databases that are labeled as NoSQL may actually support SQL commands, and thus instead should be labeled as NotRDBMS or NotRelational. The nickname NoSQL is more of a slight against Microsoft SQL Server than an indication that a DBMS used for big data does not support SQL queries.

In recent years, services, applications, and websites that have employed SQL databases (again, for clarity, a DBMS that supports SQL expressions) have been found vulnerable to a range of attacks, most notably SQL injection. However, this attack has less to do with the DBMS and SQL expressions than it does with the tendency for sites to be configured with minimal security and to use nondefensive scripts. Scripts that receive input from users but aren't written to specifically defend against SQL injection are by default vulnerable. This vulnerability, tied in with loose security controls on the DBMS, has enabled the proliferation of SQL injection attacks across the Internet.

Although this is a serious issue, it isn't the reason to switch to a NoSQL solution. There are many RDBMS options that can allow SQL to be disabled or don't support SQL as an expression language. NoSQL DBMS options can often support SQL as an expression language. Thus, switching to NoSQL doesn't resolve the SQL injection attack vulnerability on its own. The reason to switch to NoSQL solutions is to obtain a data structure and have access to data-management features that are better suited for a particular data set or programming need.

NoSQL is also known for not supporting ACID, which is a standard benefit or feature of most RDBMSs. ACID stands for the following:

- Atomicity—Each transaction occurs in an all-or-nothing state.
- Consistency—Each transaction maintains valid data and a valid state of the database.
- Isolation—Each transaction occurs individually without interference.
- Durability—Each applied transaction is resilient.

A discussion of NoSQL often brings up the topic of JSON. *JavaScript Object Notation (JSON)* is a common organizational and referencing format used by some NoSQL database options. The use of JSON as the basis for a NoSQL solution is a popular option for Internet services. However, it's only one of the many NoSQL options available.

Exam Essentials

Understand authorization. Authorization is the mechanism that controls what a subject can and can't do, access, use, or view. Authorization is commonly called access control or access restriction.

Know about access control. Access control or privilege management can be addressed using one of three primary schemes: user, group, or role. These schemes correspond directly to the access-control methodologies DAC, MAC, and RBAC.

Understand MAC. Mandatory access control (MAC) is based on classification rules. Objects are assigned sensitivity labels. Subjects are assigned clearance labels. Users obtain access by having the proper clearance for the specific resource. Classifications are hierarchical.

Know common MAC hierarchies. Government or military MAC uses the following levels: unclassified, sensitive but unclassified, confidential, secret, and top secret. Private sector or corporate business environment MAC uses these: public, sensitive, private, and confidential.

Understand DAC. Discretionary access control (DAC) is based on user identity. Users are granted access through ACLs on objects, at the discretion of the object's owner or creator.

Comprehend ACLs. An ACL is a security logical device attached to every object and resource in the environment. It defines which users are granted or denied the various types of access available based on the object type.

Understand ABAC. Attribute-based access control (ABAC) is a mechanism of assigning access and privileges to resources through a scheme of attributes or characteristics.

Know about role-based access control (RBAC). Role-based access control (RBAC) is based on job description. Users are granted access based on their assigned work tasks. RBAC is most suitable for environments with a high rate of employee turnover.

Know about rule-based access control (RBAC). Rule-based access control (RBAC) is typically used in relation to network devices that filter traffic based on filtering rules, such as those found on firewalls and routers. Rule-based systems enforce rules independent of the user or the resource, since the rules are the rules.

Understand proximity systems. A proximity device or proximity card can be a passive device, a field-powered device, or a transponder.

Comprehend biometric device selection. It is important to evaluate the available options in order to select a biometric solution that is most in line with the organization's security priorities; this can be accomplished by consulting a Zephyr analysis chart.

Understand FRR and FAR. False rejection rate (FRR, or Type I) errors are the number of failed authentications for valid subjects based on device sensitivity. False acceptance rate (FAR, or Type II) errors are the number of accepted invalid subjects based on device sensitivity.

Define CER. The crossover error rate (CER) is the point where the FRR and FAR lines cross on a graph. The comparatively lowest CER point is the more accurate biometric device for the relevant body part.

Understand tokens. A token is a form of authentication factor that is something you have. It's usually a hardware device, but it can be implemented in software as a logical token.

Comprehend TOTP. Time-based one-time password (TOTP) tokens or synchronous dynamic password tokens are devices or applications that generate passwords at fixed time intervals.

Comprehend HOTP. HMAC-based one-time password (HOTP) tokens or asynchronous dynamic password tokens are devices or applications that generate passwords based not on fixed time intervals but on a nonrepeating one-way function, such as a hash or HMAC operation.

Understand personal identification verification cards. Personal identification verification cards, such as badges, identification cards, and security IDs, are forms of physical identification and/or electronic access-control devices.

Know about smartcards. Smartcards are credit card–sized IDs, badges, or security passes with embedded integrated circuit chips. They can contain information about the authorized bearer that can be used for identification and/or authentication purposes.

Understand 802.1x. 802.1x is a port-based authentication mechanism. It's based on EAP and is commonly used in closed-environment wireless networks. However, 802.1x isn't exclusively used on WAPs; it can also be used on firewalls, proxies, VPN gateways, and other locations where an authentication handoff service is desired. Think of 802.1x as an authentication proxy.

4.4 Given a scenario, differentiate common account management practices.

Account management is an element of authentication and authorization management. Secure account management includes an understanding of the various account types allowed in the IT environment, comprehension of a wide range of concepts, and understanding of account policy restrictions enforcement. These issues are discussed in this section.

Account types

User account types are the starting point for the type, level, and restriction settings related to a subject's access to resources. Organizations should consider which types of accounts to use in their network and which types should be prohibited for use.

User account

A user account is also known as a standard account, limited account, regular account, or even a normal account. A user account is the most common type of account in a typical network, since everyone is assigned a user account if they have computer and network privileges. A user account is limited because this type of account is to be used for regular, normal daily operation tasks. A user account is prohibited, in most environments, from installing software or making significant system/OS changes (such as installing device drivers or updates).

Even system administrators should be assigned a standard user account to use for most of their work activities. The powerful administrator account should be reserved for use only when absolutely necessary.

Shared and generic accounts/credentials

Under no circumstances should a standard work environment implement shared accounts. It isn't possible to distinguish between the actions of one person and another if several people use a shared account. Shared accounts should be used only for public systems (such as kiosks) or anonymous connections (which should be avoided as well).

Generic account prohibition is the rule that no generic or shared or anonymous accounts should be allowed in private networks or on any system where security is important. Only when each subject has a unique account is it possible to track the activities of individuals and hold them accountable for their actions and any violations of company policy or the law.

Generic credentials can refer either to the shared knowledge of credentials for a shared account or to the default credentials of a built-in account. Neither form of generic credentials is secure, and both should be avoided. All native and/or default accounts should be assigned a complex password.

Guest accounts

Guest accounts can be of two forms. One option is to use a shared group guest account that all visitors use. A second option is to create a unique account for each guest, with limited privileges. The former concept of a guest account is to be avoided because it does not support holding individuals accountable for their actions. The latter concept for a guest account is more desirable since it does allow for holding individuals accountable. A per-user unique guest account can also be used to customize and target access and permission for the needs and job requirements of the temporary visitor or guest.

Guest accounts are not for every person who visits a facility; instead, they should be issued only to those who have a valid and legitimate work need to be on the company network. This might include consultants, temporary workers, visiting workers from other locations, interns, investigators, and auditors.

Service accounts

A service account is a user account that is used to control the access and capabilities of an application. Through the use of a service account, an application can be granted specific authorization related to its function and data access needs. This is a more secure solution than configuring applications to operate as an administrator, root, or the system. Most applications and services do not need full and complete systemwide power; a service account allows for fine-tuned customization of permissions, privileges, and user rights for the exact needs of the software.

Privileged accounts

Administrative personnel need two user accounts: a standard account and an administrative or *privileged account*. Their standard account should have the normal privileges that every other typical worker has. This account should be used for the mundane tasks that most workdays consist of. Their administrative account should be configured to have only the special privileges needed to accomplish the assigned administrative functions. This account should not be able to perform the mundane tasks of everyday work. This restriction forces the user to employ the correct account for the task at hand. It also limits the amount of time the administrative account is in use and prevents it from being used when administrative access is a risk rather than a benefit, such as when an administrator account is used to access the Internet, open email, or perform general file transfers or executions.

For users with multiple roles within the organization, especially multiple administrative roles, each role should have its own administrative user account. This could mean a worker has a single standard user account and two or more administrative accounts. This places an extra burden on the worker to keep authentication distinct, but it prevents a single account from being too powerful. The use of multifactor authentication should be required on all privileged accounts in order to improve security and prevent a single basic password from being defined for the account.

General Concepts

This section includes descriptions of numerous account management concepts that are essential to the secure management of an IT environment.

Least privilege

The principle of *least privilege* is the security stance that users are granted only the minimum access, permissions, and privileges that are required for them to accomplish their work tasks. This ensures that users are unable to perform any task beyond the scope of their assigned responsibilities.

The assignment of privileges, permissions, rights, access, and so on should be periodically reviewed to check for privilege creep or misalignment with job responsibilities. Privilege creep occurs when workers accumulate privileges over time as their job responsibilities change. The end result is that a worker has more privileges than the principle of least privilege would dictate based on that individual's current job responsibilities.

Least privilege is a staple of the information security realm. Simply put, where users are concerned, the principle of least privilege states that a user should be granted only the minimal privileges necessary to perform their work or to accomplish a specific task. This principle should be applied to all facets of a LAN, MAN, WAN, or any secure environment. For instance, a typical end user should not normally be granted administrative privileges. A trouble-call technician might require local administrative privileges but doesn't normally require domain administrative privileges. Basically, as a security administrator, you should limit the damage that can be done by user error, a disgruntled employee, or a hijacked account. Least privilege is one of the easiest ways to protect against these and myriad other potential security risks.

Onboarding/offboarding

Onboarding is the process of adding new employees to the identity and access management (IAM) system of an organization. The onboarding process is also used when an employee's role or position changes or when that person is awarded additional levels of privilege or access.

Offboarding is the reverse of this process. It is the removal of an employee's identity from the IAM system once that person has left the organization.

The procedures for onboarding and offboarding should be clearly documented in order to ensure consistency of application as well as compliance with regulations or contractual obligations.

Onboarding can also refer to organizational socialization. This is the process by which new employees are trained in order to be properly prepared for performing their job responsibilities. It can include training, job skill acquisition, and behavioral adaptation in an effort to integrate employees efficiently into existing organizational processes and procedures. Well-designed onboarding can result in higher levels of job satisfaction, higher levels of productivity, faster integration with existing workers, a rise in organizational loyalty, stress reduction, and a decreased occurrence of resignation.

Permission auditing and review

Permissions are the access activities granted or denied users, often through the use of per-object access control lists (ACLs). An ACL is a collection of individual access control entries (ACEs). Each object in a discretionary access control (DAC) environment has an ACL. Each ACE focuses on either one user account or a group and then grants or denies an object-specific permission, such as read, write, or execute.

Permissions should be assigned on a job responsibility basis. Users should only have sufficient permissions to accomplish their work tasks. This is one aspect of the *principle of least privilege*.

User access, user rights, and *permission auditing and review* are often based on a comparative assessment of assigned resource privileges. A *privilege* or *permission* is an ability or activity that a user account is granted permission to perform. User accounts are often assigned privileges to access resources based on their work tasks and their normal activities. The *principle of least privilege* is a security rule of thumb that states that users should be granted only the level of access needed for them to accomplish their assigned work tasks, and no more. Furthermore, those privileges should be assigned for the shortest time period possible.

A user right is an ability to alter the operating environment as a whole. User rights include changing the system time, being able to shut down and reboot a system, and installing device drivers. Standard user accounts are granted few user rights, whereas administrators often require user rights in order to accomplish their privilege system management tasks.

Exploitation of privileges is known as *privilege abuse* or *privilege escalation*. Privilege escalation occurs when a user account is able to obtain unauthorized access to higher levels of privileges, such as a normal user account that can perform administrative functions. Privilege escalation can occur through the use of a hacker tool or when an environment is incorrectly configured. It can also occur when lazy administrators fail to remove older privileges as a user is granted new privileges based on new job descriptions. An accumulation of privileges can be considered a form of privilege escalation.

Auditing and review of access and privilege should be used to monitor and track not just the assignment of privilege and the unauthorized escalation of privilege, but also privilege usage. Knowing what users are doing and how often they do it may assist administrators in assigning and managing privileges.

Usage auditing and review

Part of security is holding users accountable for their actions. This can be accomplished only if every user has a unique user account. Thus, shared or group accounts aren't sufficient to provide accountability. Each user should be required to provide strong authentication credentials to prevent account takeover. Each account needs to have clearly defined access-control and authorization restrictions. Finally, all activities of users should be recorded in an auditing or logging mechanism. By having these elements in place, you can carry out *user auditing and review* in order to determine whether users have been performing their work tasks appropriately or whether there have been failed and/or successful attempts at violating company policies or the law.

Time-of-day restrictions

Time-of-day restrictions is an access control concept that limits a user account to be able to log into a system or network only during specific hours and days of the week. For example, a daily worker may only be able to log into the work network from 7 a.m. to 6 p.m. Monday through Friday. Although this might have some effect on preventing employees from working late or accumulating overtime, the main purpose is to prevent abuse of the account during evenings and weekends when the account should generally not be in use. This is a tool and technique for limiting access to sensitive environments to normal business hours, when oversight and monitoring can be performed to prevent fraud, abuse, or intrusion.

Recertification

Recertification can be used to refer to a variety of important security management concepts.

Recertification can mean performing a periodic assessment of workers' job responsibilities in relation to their user account's permissions and rights. Recertification is a means to ensure that the principle of least privilege is being adhered to.

Recertification is used in relation to formal certification procedures, such as establishing proof of knowledge and/or skill of a subject, and may relate to the assignment, repeal, or extension of a license or an approval to operate.

The term recertification can also refer to the act of assessing an organization's compliance with regulations, standards, and their own written security policy.

Finally, recertification can reference the concept of evaluating the IT infrastructure's mechanisms of account management and privilege assignment to ensure that they continue to provide sufficient authentication and authorization security.

Standard naming convention

Some organizations have adopted a *standard naming convention* to control the names of systems, shares, user account names, and email addresses. Such systems can make creating new names easier and more straightforward, which in turn makes recovery of a forgotten name simple as well. However, this can also make it easy for outsiders to predict names if they discover the naming convention in use. Still, since names of objects and users are not as sensitive as passwords, PII, and company intellectual property, adopting a standard naming convention can be seen as a streamlining effort to prevent or curtail questionable or offensive names that some users might select on their own.

Account maintenance

Account maintenance is the regular or periodic activity of reviewing and assessing the user accounts of an IT environment. Any accounts that are no longer needed should be disabled, such as those used by previous employees or related to services that have been uninstalled. Once an account has been disabled for a reasonable length of time for any security auditing concerns (for some companies this might be 2 weeks, whereas others may need 6 months), the account should be deleted. Keep in mind that once an account is deleted, all audit records related to that account now have no user object to point to and thus might be grouped in a catch-all category in any system or security audit.

Account maintenance can also include ongoing password auditing or cracking to discover poor passwords before attackers do, in order to have users change them to something more robust.

Account management should also review group memberships, user rights, time restrictions, and resource access in relation to each worker's individual job description and work task responsibilities.

Group-based access control

Group-based privileges assign a privilege or access to a resource to all members of a group as a collective. Group-based access control grants every member of the group the same level of access to a specific object. Group-based privileges are common in many OSs, including Linux and Windows. Linux (as well as Unix) uses group-based privileges on each object. In fact, each object has three types of permissions: those for the owner, those for the group of the owner, and those for other users (known as World or Everyone). The second permission set, which defines permissions for all members of the group, is associated with the object because the owner is a member of that group.

Windows uses group management differently. Each object has an ACL. The ACL can contain one or more access control entries (ACEs). Each ACE focuses on either a single user or a group. If an ACE focuses on a group, then all members of the group are granted (or denied) the related permissions on the object.

When using group-assigned privileges, it's important to consider whether doing so violates the principle of least privilege as well as whether you actually want to grant all members of a specific group the same access to a specific object. If not, you need to alter the permissions assignment.

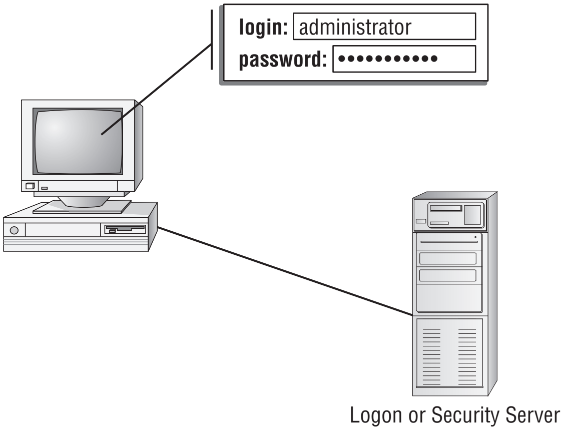
Location-based policies

Location-based policies for controlling authorization grant or deny resource access based on where the subject is located. This might be based on whether the network connection is local wired, local wireless, or remote. Location-based policies can also grant or deny access based on MAC address, IP address, OS version, patch level, and/or subnet in addition to logical or geographical location. Location-based policies should be used only in addition to standard authentication processes, not as a replacement for them.

Account policy enforcement

The combination of a username and a password is the most common form of authentication (see Figure 4.9). If the provided password matches the password stored in a system's accounts database for the specified user, then that user is authenticated to the system. However, just because using a username and password is the most common form of authentication, that doesn't mean it's the most secure. On the contrary, it's generally considered to be the least secure form of authentication.

FIGURE 4.9 A basic logon process employing a username and password



Numerous means to improve the basic username/password combination have been developed. First is the storage of passwords in an accounts database in an encrypted form. Typically that form is the hash value from a one-way hash function. Second is the use of an authentication protocol (or mechanism) that prevents the transmission of passwords in an easily readable form over a network or especially the Internet. Third, strong (complex) passwords are often enforced at a programmatic level. This is done to ensure that only passwords that are difficult for a password-cracking tool to discover are allowed by the system.

Whatever means of authentication is adopted by an organization, it is important to consider best secure business practices and to establish a standard operating procedure to follow. Once an account management policy is established, it should be enforced. Only with consistent application of security can consistent and reliable results be expected.

Strong passwords have the following characteristics:

- Consist of numerous characters (in 2017, 12 or more with at least 16 preferred)
- Include at least three types of characters (uppercase and lowercase letters, numerals, and keyboard symbols)
- Are changed on a regular basis (every 90 days)
- Don't include any dictionary or common words or acronyms
- Don't include any part of the subject's real name, username, or email address.

These features can be implemented as a requirement through *account policy enforcement*. This is the collection of password requirement features in the OS, often called a *password policy*.

Passwords should be strong enough to resist discovery through attack but easy enough for the person to remember. This can sometimes be a difficult line to walk. Training users on picking strong passphrases and memorizing them is an important element of modifying risky behavior.

Continuous monitoring stems from the need to have user accountability through the use of user access reviews. It's becoming a standard element in government regulations and security contracts that the monitoring of an environment be continuous in order to provide a more comprehensive overview of the security stance and user compliance with security policies. Effectively, continuous monitoring requires that all users be monitored equally, that users be monitored from the moment they enter the physical or logical premises of an organization until they depart or disconnect, and that all activities of all types on any and all services and resources be tracked. This comprehensive approach to auditing, logging, and monitoring increases the likelihood of capturing evidence related to abuse or violations.

Credential management

Credential management is a service or software product designed to store, manage, and even track user credentials. Many credential management options are available for enterprise networks, where hundreds or thousands of users must be managed. However, most credential management solutions are designed for end-user deployment. Credential management products allow users to store all their online (and even local) credentials in a local or cloud-based secured digital container. Examples of products of this type include LastPass, 1Password, KeePass, and Dashlane. By using a credential manager, users can define longer and more random credentials for their various accounts without the burden of having to remember them or the problem of writing them down.

The storage of credentials in a central location is referred to as *credential management*. Given the wide range of Internet sites and services, each with its own particular logon requirements, it can be a burden to use unique names and passwords. Credential management solutions offer a means to securely store a plethora of credential sets. Often these tools employ a master credential set (multifactor being preferred) to unlock the data set when needed. Some credential management options can even provide auto-login options for apps and websites.

Group policy

Group Policy is the mechanism by which Windows systems can be managed in a Windows network domain environment. A *Group Policy Object (GPO)* is a collection of Registry settings that can be applied to a system at the time of bootup or at the moment of user login. Group Policy enables a Windows administrator to maintain consistent configurations and security settings across all members of a large network. In the vast array of setting options available in a GPO, there are numerous settings related to credentials, such as password complexity requirements, password history, password length, and account lockout settings.

Password complexity

A *password policy* is both a set of rules written out as part of the organizational security policy that dictates the requirements of user and device passwords, and a technical enforcement tool (typically a native part of an OS) that enforces the password rules. The password policy typically spells out the requirements for minimum password length, maximum password age, minimum password age, password history retention, and some sort of password complexity requirement. This latter setting, *password complexity*, often enforces a minimum of three out of four standard character types (uppercase and lowercase letters, numbers, and symbols) to be represented in the password and does not allow the username, real name, and email address to appear in the password.

Generally, passwords over 12 characters are considered fairly secure, and those over 15 characters are considered very secure. Usually, the more characters in a password, along with some character type complexity, the more resistant it is to password-cracking techniques, specifically brute-force attacks. Requiring regular password changes, such as every 90 days, and forbidding the reuse of previous passwords (password history) improves the security of a system that uses passwords as the primary means of authentication.

Passwords are notoriously weak forms of authentication. Any environment that still relies on passwords alone is at greater risk for account compromise than organizations that have adopted stronger forms of authentication. Multifactor authentication should be seriously considered by every organization as a means to improve authentication security.

Good passwords can be crafted. However, most users revert to default or easier behaviors if left to their own devices. It is not uncommon for users—even when they are trained to pick passwords that are strong, long, and easier to remember—to write them down, be fooled by a social engineer, or reuse the password in other environments.

Bad password behaviors also include the following:

- Reusing old or previous passwords
- Sharing passwords with co-workers, friends, or family
- Using a nonencrypted password storage tool
- Allowing passwords to be used over nonencrypted protocols
- Failing to check for hardware keystroke loggers, video cameras, or shoulder-surfing onlookers.

Most of these poor password behaviors can be addressed with security policy, technology limitations, and user training.

Good password behaviors include selecting a passphrase of at least 15 characters, ensuring that at least three character types are represented (uppercase, lowercase, numbers, symbols, higher-order ASCII characters, and foreign language characters), memorizing passwords, using an encrypted password-storage tool only with authorized permission, following password-change rules, and not reusing passwords on the same or even on different systems.

Expiration

It has been common practice for years for passwords to automatically expire after a specific length of time in order to force users to change them. The length of time for a password to remain static can vary based on risk and threat levels. However, a common traditional rule of thumb for password *expiration* is for passwords to be changed every 90 days. This may still be considered the "right" answer for Security+, but the idea that a password needs to be changed due to its age is now considered to be invalid (see the accompanying sidebar). A password needs to be changed only if it

- Isn't in compliance with company password policy
- Is obviously insecure
- Has been reused
- Is likely compromised due to a system intrusion

Otherwise, a strong (long and complex) password can remain static.

Long Live Passwords!

According to NIST Special Publication 800-63B: Digital Identity Guidelines, published in June 2017 (`http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63b.pdf`), the new government guidelines for passwords are focusing on improving security for users based on historical compromise issues, research, and science. A key element of this revision is to no longer recommend that passwords be forcibly changed at specific intervals. The document states, "Verifiers SHOULD NOT require memorized secrets to be changed arbitrarily (e.g., periodically). However, verifiers SHALL force a change if there is evidence of compromise of the authenticator." Instead, NIST now recommends using length and complexity requirements and to allow passwords to remain static as long as no significant evidence of account compromise is detected.

However, for the exam, keep in mind that CompTIA may still recommend setting a maximum password age.

Recovery

Password recovery is usually a poor security solution. When a password is forgotten, it should be changed. The ability to recover and/or reveal a password requires that the password storage mechanism be reversible or that passwords be stored in multiple ways. A more secure option is to require passwords to be changed rather than recovered.

Most systems have moved to password replacement and away from actual password recovery. However, password replacement is not necessarily a secure process. Organizations often have a secure option that requires workers to visit an account manager's office in person to show a photo ID in order to have an account password changed or reset. However, websites often use one of two poor password reset options. If you have forgotten your password to a website, it often offers an "I forgot my password" link. Then you are prompted to provide the email address related to your account or answer several security or identity proofing questions.

If a password is sent to your email address and you then happen to recognize it as the password you had forgotten, you know that the web service is storing passwords in plaintext form. If they were storing passwords in a secure manner, even just a simple hash, they would not have the ability to instantly send you your actual existing password. What you hope to see in the recovery email is a new, hopefully random, password. But since email is itself a plain-text communication system, you need to use the new password quickly to log into your account and immediately change the password to something new.

If you are asked several security questions, then you need to know the facts that are being requested about you from some database source, your purchase history or credit history, or the answers to the questions you were asked or you selected at the time you set up the account originally. Unfortunately, it is all too common for these questions to be rather mundane and standardized across various websites. For example, many ask about your favorite vacation, favorite food, favorite music, favorite movie, third-grade teacher's name, first pet name, first vehicle make and model, first job, high school mascot, or best man/maid of honor at your wedding. These common questions are often information about you that many others know, especially friends, family, and some enemies. When setting the answers to security questions, consider recommending to users that they implement an obfuscation technique so the typed-in answers are not as obvious. Some options include spelling the correct answer backward; answering the opposite of the question; or adding a padding statement, such as ABC123, to each answer. Although no obfuscation technique is foolproof and any will be disclosed if the answer database is breached, it will at least provide some additional protection against those who would attempt to impersonate you via password recovery vulnerabilities.

Disablement

Disablement, or *account expiration*, is a little-used feature of some OS user accounts that automatically disables a user account or causes the account to expire at a specific time and on a specific day. Account expiration is a secure feature to employ on user accounts for temporary workers, interns, or consultants. Workers who need valid user accounts but whose employment or access will expire at a specific known date and time can be set up with accounts that are preconfigured to become disabled. In most cases, such accounts can be re-enabled after they expire, and new or updated expiration dates can be established at any time.

Lockout

Account lockout automatically disables an account when someone attempts to log on but fails repeatedly because they type in an incorrect password. Account lockout is often configured to lock out an account after three to five failed logon attempts within a short time (such as 15 minutes). Accounts that are locked out may remain permanently disabled until an administrator intervenes or may return to functional status after a specified period of time.

Password history

Password history is an authentication protection feature that tracks previous passwords (by archiving hashes) in order to prevent password reuse. For password history to be effective, it must typically be combined with a minimum password age requirement. For example, if five password histories are being retained, a worker could change their password six times to return to their preferred password. However, if there was also the requirement to keep a password a minimum of three days, it would take the person eighteen days to be able to get back to their preferred password. This lengthy delay is often a sufficient deterrent against password reuse.

Password reuse

Password reuse occurs when a user attempts to use a password they had used previously on the same system. The management of password history prevents password reuse.

Password length

Password length, in combination with complexity, is an important factor in determining a password's strength. Generally, longer passwords are better. Passwords of seven characters or less are likely to be cracked within hours. Passwords of eight or nine characters are likely to be cracked within days to weeks. Passwords of ten or more characters are unlikely to be cracked.

These relative strengths are based on the range of character types, the use of a strong hashing mechanism for storage, and never transmitting the password in plain text. The mathematical predictions of strength aren't a guarantee. Additionally, lazy actions on the part of the user or poor security management in the environment can provide other means to learn or bypass strong passwords.

Exam Essentials

Know about shared accounts. Under no circumstances should a standard work environment implement shared accounts. It isn't possible to distinguish between the actions of one person and another if they both use a shared account.

Understand the principle of least privilege. The principle of least privilege is a security rule of thumb that states that users should be granted only the level of access needed for them to accomplish their assigned work tasks, and no more.

Define onboarding/offboarding. Onboarding is the process of adding new employees to the organization's identity and access management (IAM) system. It can also mean organizational socialization, which is the process by which new employees are trained in order to be properly prepared for performing their job responsibilities. Offboarding is the removal of an employee's identity from the IAM system once they have left the organization.

Understand privileges. Group-based privileges assign a privilege or access to a resource to all members of a group as a collective. User-assigned privileges are permissions that are granted or denied on a specific individual user basis.

Know about time-of-day restrictions. Time-of-day restrictions is an access control concept that limits a user account to be able to log into a system or network only during specific hours.

Understand account maintenance. Account maintenance is the regular or periodic activity of reviewing and assessing the user accounts of an IT environment.

Comprehend group-based access control. Group-based access control grants every member of the group the same level of access to a specific object.

Understand location-based policies. Location-based policies for controlling authorization grant or deny resource access based on where the subject is located.

Know about credential management. Credential management is a service or software product designed to store, manage, and even track user credentials.

Understand Group Policy. Group Policy is the mechanism by which Windows systems can be managed in a Windows network domain environment. A Group Policy Object (GPO) is a collection of Registry settings that can be applied to a system at the time of bootup or at the moment of user login.

Comprehend password management. Password management is the system used to manage passwords across a large network environment. It typically includes a requirement for users to create complex passwords. It also addresses the issues of complexity, expiration, recovery, account disablement, lockout, history, reuse, and length.

Review Questions

You can find the answers in the Appendix.

1. What method of access control is best suited for environments with a high rate of employee turnover?

   A. MAC

   B. DAC

   C. RBAC

   D. ACL

2. What mechanism is used to support the exchange of authentication and authorization details between systems, services, and devices?

   A. Biometric

   B. Two-factor authentication

   C. SAML

   D. LDAP

3. Which is the strongest form of password?

   A. More than eight characters

   B. One-time use

   C. Static

   D. Different types of keyboard characters

4. Which of the following technologies can be used to add an additional layer of protection between a directory services–based network and remote clients?

   A. SMTP

   B. RADIUS

   C. PGP

   D. VLAN

5. Which of the following is not a benefit of single sign-on?

   A. The ability to browse multiple systems

   B. Fewer usernames and passwords to memorize

   C. More granular access control

   D. Stronger passwords

6. Federation is a means to accomplish __________.

   A. Accountability logging

   B. ACL verification

   C. Single sign-on

   D. Trusted OS hardening

7. You have been tasked with installing new kiosk systems for use in the retail area of your company's store. The company elected to use standard equipment and an open-source Linux operating system. You are concerned that everyone will know the default password for the root account. What aspect of the kiosk should be adjusted to prevent unauthorized entities from being able to make system changes?

   A. Authorization

   B. Accounting

   C. Authentication

   D. Auditing

8. Your company has several shifts of workers. Overtime and changing shifts is prohibited due to the nature of the data and the requirements of the contract. To ensure that workers are able to log into the IT system only during their assigned shifts, you should implement what type of control?

   A. Multifactor authentication

   B. Time-of-day restrictions

   C. Location restrictions

   D. Single sign-on

9. Place the following steps (represented by the letters A through I) in the correct order:

   A. The KDC verifies that the client has a valid TGT and then issues an ST to the client. The ST includes a time stamp that indicates its valid lifetime.

   B. The client receives the TGT. At this point, the subject is an authenticated principle in the Kerberos realm.

   C. The client sends the ST to the network server that hosts the desired resource.

   D. The Kerberos client system encrypts the password and transmits the protected credentials to the KDC.

   E. The subject requests access to resources on a network server. This causes the client to request a service ticket (ST) from the KDC.

   F. The subject provides logon credentials.

   G. The network server verifies the ST. If it's verified, it initiates a communication session with the client. From this point forward, Kerberos is no longer involved.

   H. The KDC verifies the credentials and then creates a ticket-granting ticket (TGT)—a hashed form of the subject's password with the addition of a time stamp that indicates a valid lifetime. The TGT is encrypted and sent to the client.

   I. The client receives the ST.

      A. F, D, H, B, E, A, I, C, G

      B. H, I, C, D, G, F, E, A, B

      C. A, B, C, D, E, F, G, H, I

      D. I, A, E, B, C, G, F, H, D

10. Your company has recently purchased Cisco networking equipment. When you are setting up to allow remote access, what means of AAA service is now available to your organization?

    A. RADIUS

    B. X.500

    C. TACACS+

    D. X.509 v3

11. Your organization has recently decided to allow some employees to work from home two days a week. While configuring the network to allow for remote access, you realize the risk this poses to the organization's infrastructure. What mechanism can be implemented to provide an additional barrier against remote access abuse?

    A. Kerberos

    B. Single sign-on

    C. Stronger authorization

    D. RADIUS

12. You are developing a smart app that will control a new IoT device that automates blinking light fixtures in time with the beat of music. You want to make using the device as simple as possible, so you want to adopt an authentication technique that is seamless for the user. Which technology should you integrate into your app and device?

    A. OpenID Connect

    B. Shibboleth

    C. A secure token

    D. Role-based access control

13. How are effective permissions calculated?

    A. Count the number of allows, subtract the number of denials

    B. Accumulate allows, remove denials

    C. Look at the user's clearance level

    D. Count the number of groups the user is a member of

14. What form of authorization is based on a scheme of characteristics related to the user, the object, the system, the application, the network, the service, time of day, or even other subjective environmental concerns?

    A. RBAC

    B. MAC

    C. DAC

    D. ABAC

15. Your organization wants to integrate a biometric factor into the existing multifactor authentication system. To ensure alignment with company priorities, what tool should be used in selecting which type or form of biometric to use?

    A. CER comparison

    B. OAuth verifier

    C. Zephyr analysis chart

    D. Federation assessment

16. What type of biometric error increases as the sensitivity of the device increases?

    A. FAR

    B. FRR

    C. CER

    D. False positive

17. You are installing a new network service application. The application requires a variety of permissions on several resources and even a few advanced user rights in order to operate properly. Which type of account should be created for this application to operate under?

    A. Service

    B. User

    C. Privileged

    D. Generic

18. Failing to perform regular permissions auditing can result in a violation of what security concept?

    A. Implicit deny

    B. Security by obscurity

    C. Least privilege

    D. Diversity of defense

19. What type of access management can involve restrictions based on MAC address, IP address, OS version, patch level, and/or subnet in addition to logical or geographical position?

    A. Geography-based access control

    B. Physical access control

    C. Logical access control

    D. Location-based access control

20. Which of the following is a recommended basis for reliable password complexity?

    A. Minimum of eight characters; include representations of at least three of the four character types

    B. Allow for a maximum of three failed logon attempts before locking the account for 15 minutes

    C. Require that a password remain static for at least three days and prevent the reuse of the five most recently used passwords

    D. Require that each administrator have a normal user account in addition to a privileged account

Chapter

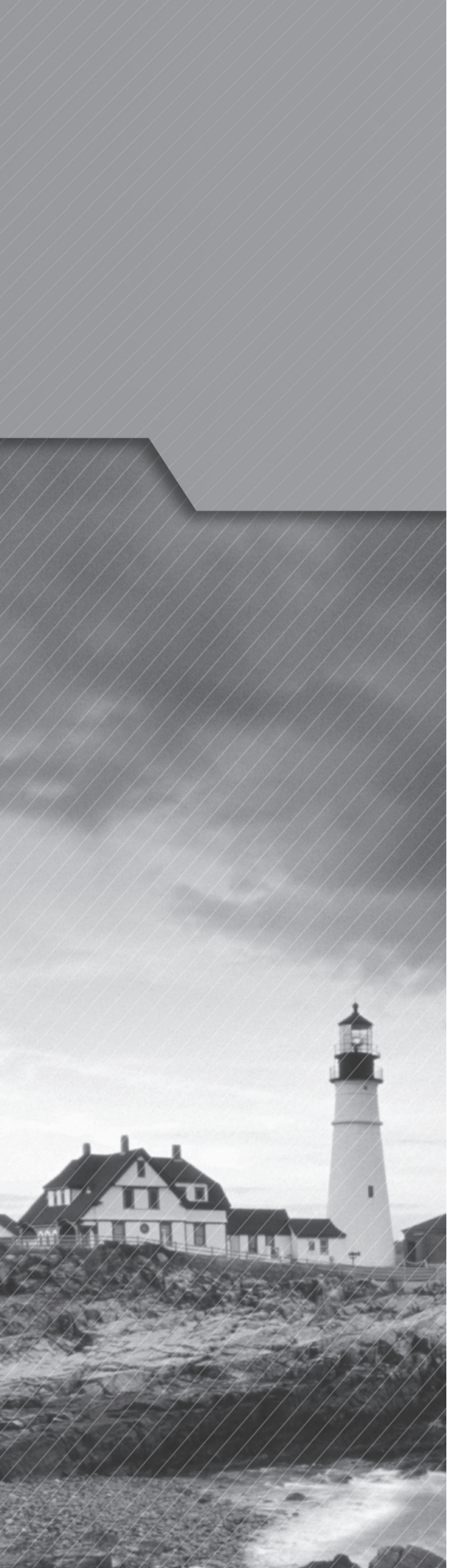
5

Risk Management

COMPTIA SECURITY+ EXAM OBJECTIVES COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:

✓ **5.1 Explain the importance of policies, plans and procedures related to organizational security.**

- Standard operating procedure
- Agreement types
  - BPA
  - SLA
  - ISA
  - MOU/MOA
- Personnel management
  - Mandatory vacations
  - Job rotation
  - Separation of duties
  - Clean desk
  - Background checks
  - Exit interviews
  - Role-based awareness training
    - Data owner
    - System administrator
    - System owner
    - User
    - Privileged user
    - Executive user
  - NDA
  - Onboarding
  - Continuing education



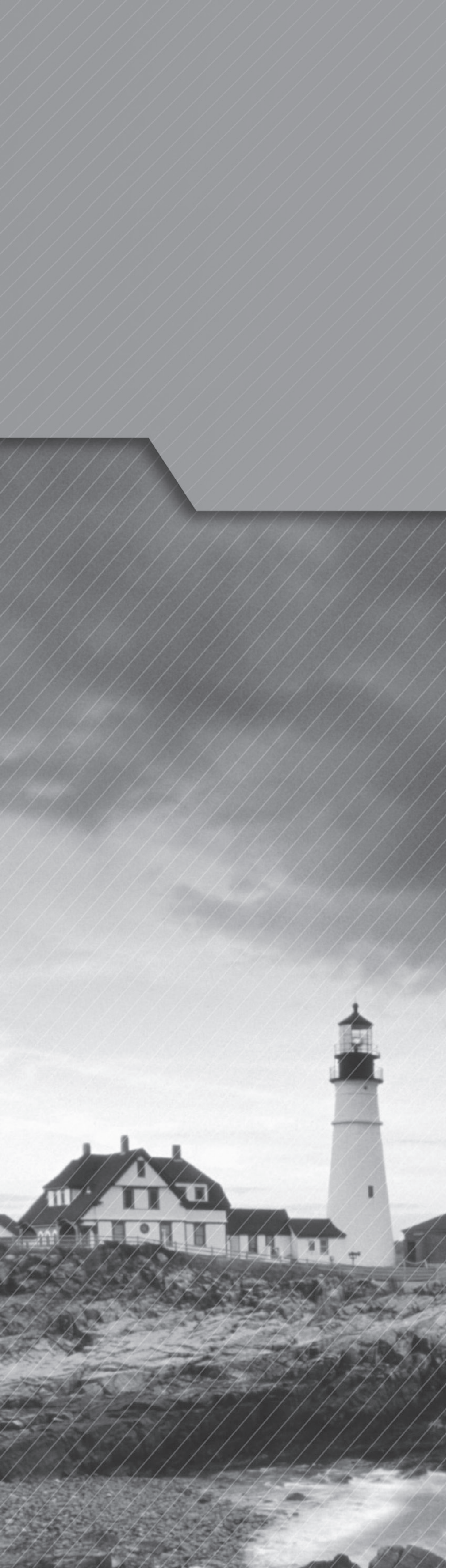
    - Acceptable use policy/rules of behavior
    - Adverse actions
  - General security policies
    - Social media networks/applications
    - Personal email

✓ **5.2 Summarize business impact analysis concepts.**

  - RTO/RPO
  - MTBF
  - MTTR
  - Mission-essential functions
  - Identification of critical systems
  - Single point of failure
  - Impact
    - Life
    - Property
    - Safety
    - Finance
    - Reputation
  - Privacy impact assessment
  - Privacy threshold assessment

✓ **5.3 Explain risk management processes and concepts.**

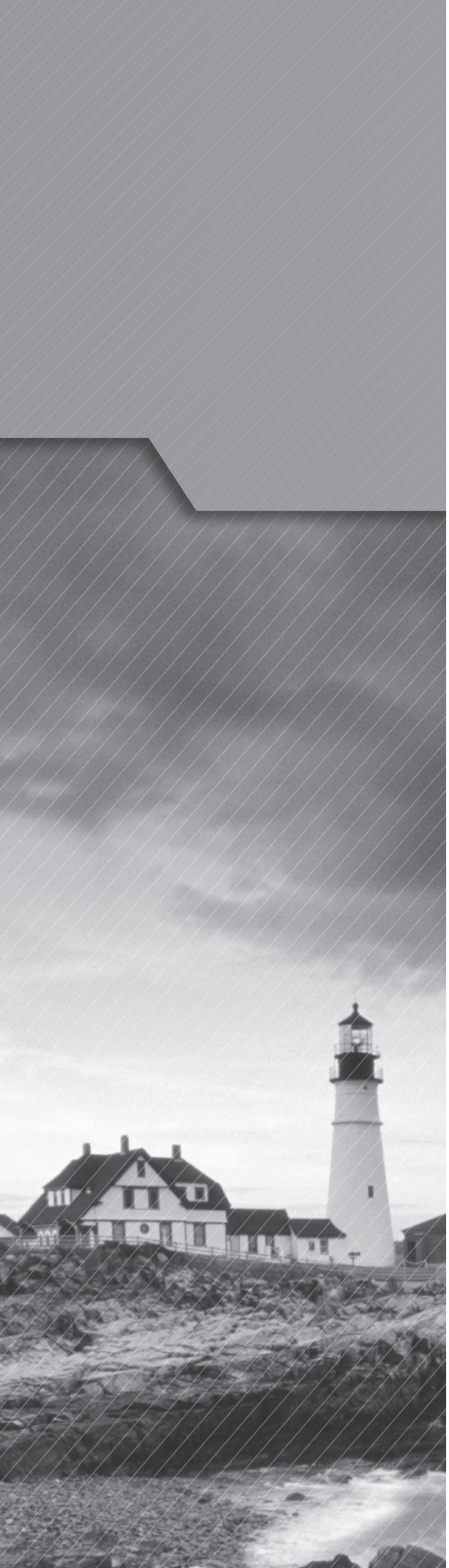
  - Threat assessment
    - Environmental
    - Manmade
    - Internal vs. external
  - Risk assessment
    - SLE
    - ALE
    - ARO
    - Asset value



- Risk register
- Likelihood of occurrence
- Supply chain assessment
- Impact
- Quantitative
- Qualitative
- Testing
  - Penetration testing authorization
  - Vulnerability testing authorization
- Risk response techniques
  - Accept
  - Transfer
  - Avoid
  - Mitigate
- Change management

✓ **5.4 Given a scenario, follow incident response procedures.**

- Incident response plan
  - Documented incident types/category definitions
  - Roles and responsibilities
  - Reporting requirements/escalation
  - Cyber-incident response teams
  - Exercise
- Incident response process
  - Preparation
  - Identification
  - Containment
  - Eradication
  - Recovery
  - Lessons learned

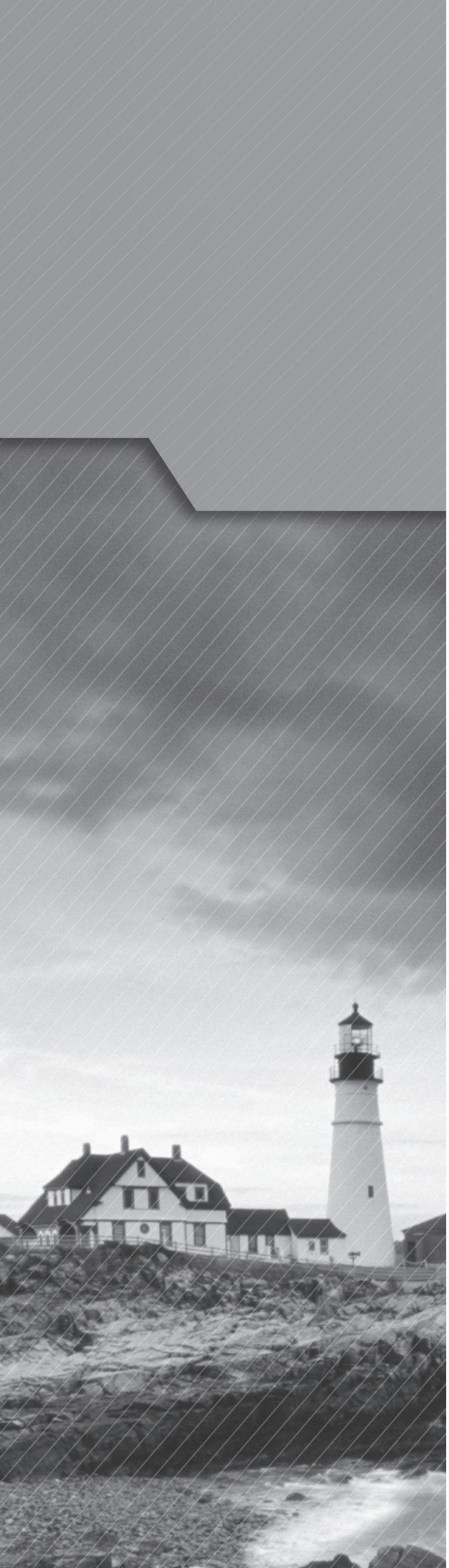


✓ **5.5 Summarize basic concepts of forensics.**

- Order of volatility
- Chain of custody
- Legal hold
- Data acquisition
  - Capture system image
  - Network traffic and logs
  - Capture video
  - Record time offset
  - Take hashes
  - Screenshots
  - Witness interviews
- Preservation
- Recovery
- Strategic intelligence/counterintelligence gathering
  - Active logging
- Track man-hours

✓ **5.6 Explain disaster recovery and continuity of operation concepts.**

- Recovery sites
  - Hot site
  - Warm site
  - Cold site
- Order of restoration
- Backup concepts
  - Differential
  - Incremental
  - Snapshots
  - Full
- Geographic considerations
  - Off-site backups
  - Distance



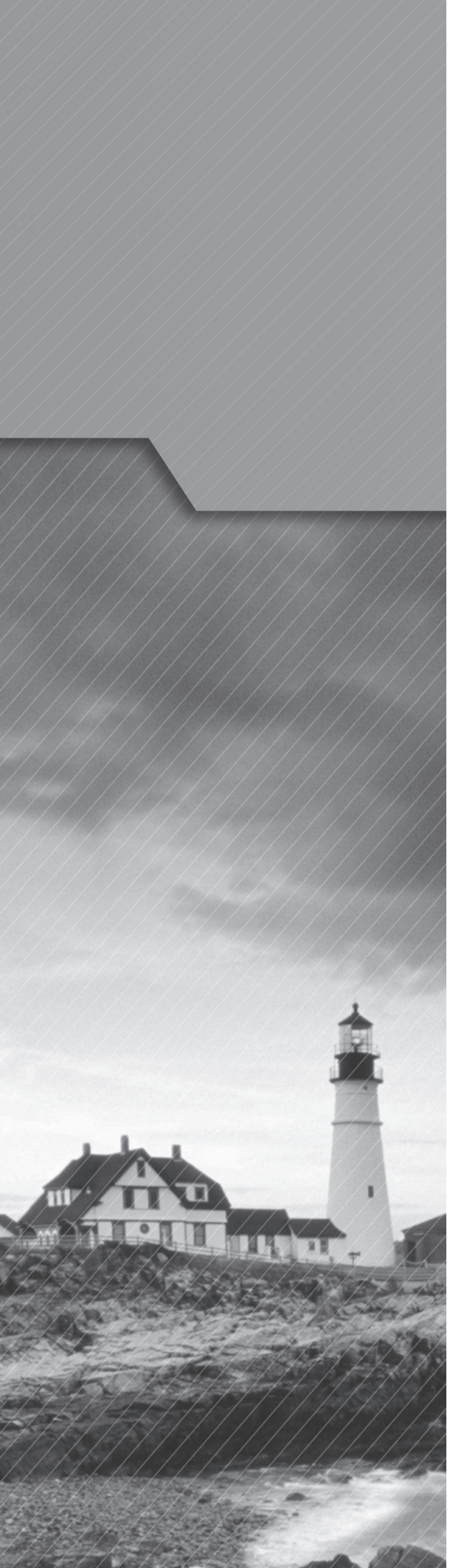
  - Location selection
  - Legal implications
  - Data sovereignty
- Continuity of operation planning
  - Exercises/tabletop
  - After-action reports
  - Failover
  - Alternate processing sites
  - Alternate business practices

✓ 5.7 Compare and contrast various types of controls.

- Deterrent
- Preventive
- Detective
- Corrective
- Compensating
- Technical
- Administrative
- Physical

✓ 5.8 Given a scenario, carry out data security and privacy practices.

- Data destruction and media sanitization
  - Burning
  - Shredding
  - Pulping
  - Pulverizing
  - Degaussing
  - Purging
  - Wiping
- Data sensitivity labeling and handling
  - Confidential
  - Private



  - Public
  - Proprietary
  - PII
  - PHI
- Data roles
  - Owner
  - Steward/custodian
  - Privacy officer
- Data retention
- Legal and compliance

The Security+ exam will test your knowledge about preparing for and handling risk, managing incident response, dealing with forensic investigations, and addressing business continuity and disaster recovery level issues. To pass the test and be effective in implementing security, you need to comprehend risk management, incident response, forensics, business continuity planning, and disaster recovery planning as detailed in this chapter.

5.1 Explain the importance of policies, plans and procedures related to organizational security.

Organizational security requires a written security policy in order to be successful. Only with a written policy is it possible to properly implement the prescribed security, and it also makes it possible to properly assess the security. A security policy will include specific plans and procedures defining how to install and configure security components as well as how workers should accomplish tasks in compliance with the security policy. The following sections focus on the areas of concern related to these topics as covered by Security+.

Standard operating procedure

A *standard operating procedure (SOP)* is an organizational policy that provides detailed or granular step-by-step instructions to accomplish a specific task. The goal of an SOP is to improve consistency in worker activities, especially as related to performance and security compliance. If all workers abide by the same SOPs, the results of their work will be more consistent, efficient, productive, and compliant with security restrictions.

Agreement types

Whenever a third party is involved in your IT infrastructure, there is an increased risk of data loss, leakage, or compromise. The security implications of integrating systems and data with third parties need to be considered carefully before implementation.

When you are integrating systems and data with third parties, be sure to compare and contrast the security stance of each organization. Each side of an agreement or connection should follow security policies and procedures as defined by their organization. However, both sides need to ensure that their expectations of security are satisfied by the partner's security infrastructure. If a significant gap exists between the levels of security or even the maturity of security between organizations, then the less secure entity will put the other entity at greater risk of compromise—in the same way that having one unpatched system puts all other systems in a network at risk.

To verify compliance and performance standards, you should review agreement requirements. Each side should assess or audit its partner for compliance with the mutual agreements as well as with any regulations or contractual obligations. In the event a partner is found in violation of regulations, both partners may be held responsible for the oversight. Even without regulations, it is beneficial to both parties to assess the other's security acumen and ability to support reasonable levels of productivity and performance for the purpose of ensuring the mutual benefit of the various agreements and contracts.

Data backups are essential because they are the only means of recovering data in the event of loss or corruption. However, when third parties are involved in a data system or information exchange, the issue of what is to be backed up and by whom needs to be addressed. Which side of a communication stream should be backing up the data? Should both sides back up the data? Does data ownership need to be considered during backups? If partnerships are dissolved, how is the comingled data to be handled in archived backups and during restoration activities?

Unauthorized data sharing can lead to the disclosure of private, confidential, or proprietary data to outsiders or unapproved entities. When you work with a third party with regard to data and systems integration, the risk of unauthorized data sharing increases. Data encryption, strong authentication, granular authorization controls, and detailed monitoring of activities are required to reduce and/or eliminate such disclosures.

An *interoperability agreement* is a formal contract (or at least a written document) that defines some form of arrangement where two entities agree to work with each other in some capacity. It defines the specifics of an exchange or sharing so there is little room for misunderstanding or for changing the terms of the agreement after the fact. The agreement could be between a supplier and customer or between equals. Such an agreement may discuss the sharing of a single resource or an exchange of resources of equivalent values.

There is a wide range of forms and types of interoperability agreements (Table 5.1). Some of these are discussed in the following sections.

TABLE 5.1 Types of agreements

Agreement type	Description
BPA	Business partners agreement—between organizational entities
SLA	Service-level agreement—between customer and supplier
ISA	Interconnection security agreement—sets IT networking security requirements
MOU/MOA	Memorandum of understanding/Memorandum of agreement—an expression of agreement or aligned intent

BPA

A *business partners agreement (BPA)* is a contract between two entities dictating the terms of their business relationship. It clearly defines the expectations and obligations of each partner in the endeavor. A BPA should include details about the decision-making process; management style; how business capital is to be allocated; the level of salary, benefits, and other distributions; whether new partners can be added; dispute resolution; outside competing activities/conflicts of interest; and how death or dissolution should be handled.

SLA

A *service-level agreement (SLA)* is a contract between a supplier and a customer. The SLA defines what is provided for a specific cost, barter, or other compensation. It specifies the range, values, quality, time frame, performance, and other attributes of the service or product. If the provider does not fulfill their obligations, the SLA lists the customer's options of compensation or recompense. It also defines the customer's penalties in the event of late payment or nonpayment.

ISA

An *interconnection security agreement (ISA)* is a formal declaration of the security stance, risks, and technical requirements of a link between two organizations' IT infrastructures. The goal of an ISA is to define the expectations and responsibilities of maintaining security over a communications path between two networks. Connecting networks can be mutually beneficial, but it also raises additional risks that need to be identified and addressed. An ISA is a means to accomplish that.

MOU/MOA

A *memorandum of understanding (MOU)* or *memorandum of agreement (MOA)* is an expression of agreement or aligned intent, will, or purpose between two entities. It is not typically a legal agreement or commitment, but rather a more formal form of a reciprocal agreement or handshake (neither of which is typically written down). An MOU can also be called a *letter of intent*. It is a means to document the specifics of an agreement or arrangement between two parties without necessarily legally binding them to the parameters of the document.

Personnel management

Implementing proper security involves the use of technology but also mandates the modification of user behaviors. If personnel do not believe in and support security, they are often opposed to the best security efforts of the organization. The weakest link of any security structure is the people who work in it. Understanding that your employees either support security or are dismantling it is critical to proper policy design, security implementation, and user training.

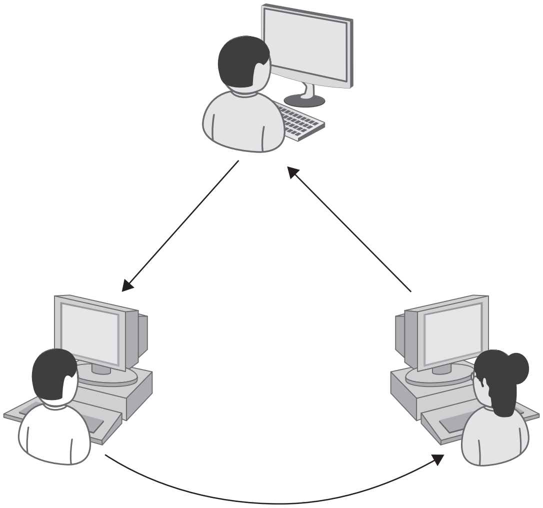
Mandatory vacations

Mandatory vacations are a form of user peer auditing. The process works by requiring each employee to be on vacation (or just away from the office and without remote access) for a minimal amount of time each year (typically one to two weeks). While the employee is away, another worker performs their work tasks using the original employee's privileged account. This process is used to detect fraud, abuse, or incompetence. The technique is often employed in financial environments or where high-value assets are managed.

Job rotation

Job rotation or *cross-training* or *rotation of duties* is a counterbalance to the application of separation of duties (see the next section). If all high-level tasks are performed by individual administrators, what happens if one person leaves the organization? If no one else has the knowledge or skill to perform the tasks, the organization suffers. Job rotation is the periodic shifting of assigned work tasks or job descriptions among a small collection of workers, sometimes known as a rotation group (Figure 5.1).

FIGURE 5.1 Job rotation, cross-training, and rotation of duties



When job rotation is implemented, multiple people have the knowledge to perform each task. Those people do not always have the permissions to perform those tasks, but they can be called on if needed to perform them once the privileges are granted. This reduces the risk of a person leaving the organization who happens to be the only individual with the proprietary knowledge or know-how of a mission-critical function. The implementation of job rotation reduces the administrative impact to the organization by employing several administrators who are cross-trained in their respective job roles. This helps ensure continued administrative support for a specific role or job function in the event of a loss in administrative personnel.

Separation of duties

Separation of duties (SoD) is the division of administrative or privileged tasks into distinct groupings; in turn, each grouping is individually assigned to unique administrators. The application of separation of duties results in no one user having complete access or power over an entire network, server, or system. Each administrator has their own uniquely defined area of responsibility and privileges only within that specifically assigned area. If an administrator goes rogue or their account is compromised, the entire network is not automatically compromised.

Separation of duties applies the principle of least privilege to administrative users. However, it also requires that several administrators work together to perform high-risk, sweeping tasks in an organization. This helps prevent fraud, reduce errors, and prevent conflicts of interest. For example, those who configure security should not be the same people who test security; those who are in accounts receivable should not be performing accounts payable; and those who are programmers should not be the same people who test code and approve applications for deployment.

Think of separation of duties as a control mechanism designed to limit the damage that could be done by a single individual due to error or fraud. For example, it's generally a bad idea to have the same personnel responsible for both LAN administration and LAN security. A better model would be to have an IT department as well as a security department. More stringent applications separate reporting as well to prevent the "make it work" philosophy of many IT departments from outweighing the less popular "make it work in a secure manner" alternative. SoD is one reason that you now find both a chief information officer (CIO) and a chief operations officer (COO) in many corporations. Also see the related discussion in the section "Secure DevOps" in Chapter 3, "Architecture and Design."

Clean desk

A *clean-desk policy* is used to instruct workers how and why to clean off their desks at the end of each work period. In relation to security, such a policy has a primary goal of reducing disclosure of sensitive information. This can include passwords, financial records, medical information, sensitive plans or schedules, and other confidential materials. If at the end of each day/shift a worker places all work materials into a lockable desk drawer or file cabinet, this prevents exposure, loss, and/or theft of these materials.

Users are also well known for failing to handle data properly. Users should be instructed where to keep data files. Typically, data files should be stored on servers that are included in the company's backup process, rather than being stored on clients. Users should not employ removable media unless authorized and approved. Any removable media containing sensitive or valuable data should be treated with additional care to prevent loss or theft. Users should not install software, because it may be infected with malware that could steal data or otherwise compromise the network. Users should not transmit sensitive data through any unencrypted means, including Internet email, file transfer, peer-to-peer file sharing, IM, or VoIP collaboration tools.

Users who are allowed to take home, or otherwise use out of the office, a portable computer or removable media with sensitive data should take only the minimum amount of data required to perform immediate work tasks. Any resource that leaves company premises is at significantly greater risk of being lost or stolen. It is important to protect entire data sets, databases, or record collections from being exposed to this unnecessary risk.

Background checks

Background checks are used to verify that a worker is qualified for a position and not disqualified. For example, a new applicant for a job might have the right education and certifications but lack the minimum required work experience, thus being both qualified and disqualified. Background checks are used to verify work history, education history, criminal background if any, certifications, and clearance verification, as well as personal and professional references. The goal or purpose of background checks is to verify that a current or potential worker is a good fit for a job position and for the organization and will not be a threat or detriment to the organization.

Exit interviews

An *exit interview* is a controlled and respectful process of termination or employee firing. The goal of an exit interview is to control the often emotionally charged event of a termination in order to minimize property damage, information leakage, or other unfortunate or embarrassing occurrences. Often an exit interview is held in the office of the worker's manager or in the HR manager's office. Once the worker enters the meeting, they will be accompanied by a third individual, such as another manager, supervisor, executive, or security guard, who serves as a witness to the interactions. The worker is then informed that they are being relieved of their job and released to pursue other work opportunities. They are reminded of the legal requirement to adhere to signed nondisclosure agreements and any other related contracts. Any items on their person that are company property are turned over at this time; these can include keys, badges, smartcards, pagers, and smartphones. The now ex-employee is then escorted directly off the premises. If the ex-employee has any personal property in their work area, it is collected for them by a security guard and returned to them off the premises. In some organizations, there may be a requirement to allow a security guard to drive the ex-employee's vehicle out of the corporate parking structure. If the worker has any additional company property at their home, a security guard will follow them in a separate vehicle and wait on public ground for the ex-employee to enter their home, collect the relevant items, and present them to the security guard for inventory and evaluation.

There are many variations of and additions to an exit interview that may be added to this basic structure. An exit interview should be handled in a consistent and respectful manner. A properly handled termination process will leave the ex-employee with their dignity and provide them with knowledge on how to address post-employment issues (such as health insurance, unemployment benefits, retirement account management, and final paycheck delivery), while preventing damage to company property, altercations with managers and other employees, or theft or corruption of company data.

Role-based awareness training

The successful implementation of a security solution requires changes in user behavior. These changes primarily consist of alterations in normal work activities to comply with the standards, guidelines, and procedures mandated by the security policy. Behavior modification involves some level of learning on the part of the user. There are three commonly recognized learning levels: awareness, training, and education.

A prerequisite to actual security training is *awareness*. The goal of creating awareness is to bring security to the forefront and make it a recognized entity for users. Awareness establishes a common baseline or foundation of security understanding across the entire organization. Awareness is not created exclusively through a classroom type of exercise but through the work environment. Many tools can be used to create awareness, such as posters, notices, newsletter articles, screen savers, T-shirts, rally speeches by managers, announcements, presentations, mouse pads, office supplies, and memos, as well as traditional instructor-led training courses. Awareness focuses on key or basic topics and issues related to security that all employees, no matter which position or classification they have, must comprehend.

Awareness is a tool for establishing a minimum standard common denominator or foundation of security understanding. All personnel should be fully aware of their security responsibilities and liabilities. They should be trained to know what to do and what not to do.

The issues that users need to be aware of include avoiding waste, fraud, and unauthorized activities. All members of an organization, from senior management to temporary interns, need the same level of awareness. The awareness program in an organization should be tied in with its security policy, incident-handling plan, and disaster-recovery procedures. For an awareness-building program to be effective, it must be fresh, creative, and updated often. The awareness program should also be tied to an understanding of how the corporate culture affects and impacts security for individuals as well as the organization as a whole. If employees do not see enforcement of security policies and standards, especially at the awareness level, then they may not feel obligated to abide by them.

Role-based training involves teaching employees to perform their work tasks and to comply with the security policy (Table 5.2). All new employees require some level of training so that they can comply with all standards, guidelines, and procedures mandated by the security policy. New users need to know how to use the IT infrastructure, where data is stored, and how and why resources are classified. Many organizations choose to train new employees before they are granted access to the network, whereas others grant new users limited access until their training in their specific job position is complete. Training is an ongoing activity that must be sustained throughout the lifetime of the organization for every employee. It is considered an administrative security control.

TABLE 5.2 Security roles

Role	Definition
Data owner	Has the responsibility of prescribing security needs for resources
System administrator	Implements security
System owner	Sets requirements for the system or network
User	The operator of a system
Privileged user	Has additional capabilities and responsibilities beyond that of a user
Executive user	Corporate management personnel

Awareness and role-based training are often provided in-house. That means these teaching tools are created and deployed by and within the organization. However, the next level of knowledge distribution is usually obtained from an external third-party source.

Education is a more detailed endeavor in which students and users learn much more than they actually need to know to perform their work tasks. Education is most often associated with users pursuing certification or seeking job promotion. It is typically a requirement for personnel seeking security professional positions. A security professional requires extensive knowledge of security and the local environment for the entire organization and not just their specific work tasks.

Data owner

When a third party is involved in an IT system or data exchange, it is important to clearly establish rules and restrictions regarding data ownership. Does the original possessor of the data retain ownership? Does anyone receiving the data now have ownership? Or does the intermediary supporting network or communications path have potential ownership of transferred data?

The data owner has full control over the objects that they own. This grants them the ability to make content changes as well as set authorization for others.

User, Owner, Custodian

When discussing access to objects, three subject labels are used: user, owner, and custodian. A *user* is any subject who accesses objects on a system to perform some action or accomplish a work task. An *owner*, or information owner, is the person who has final corporate responsibility for classifying and labeling objects and protecting and storing data. The owner may be liable for negligence if they fail to perform due diligence in establishing and enforcing security policies to protect and sustain sensitive data. A *custodian* or *steward* is a subject who has been assigned or delegated the day-to-day responsibility of properly storing and protecting objects.

System administrator

A system administrator is the individual tasked with the responsibilities of implementing the security and functionality policies and requirements established by the organization and the system owner. A system administrator's job functions may include installing updates, setting configurations, installing software, backing up data and settings, performing system evaluations, and responding to any troubleshooting- or breach-related issues.

System owner

A system owner is the entity responsible for setting the requirements for a system. The system owner may be the organization as a whole or an individual network or IT manager. The system owner crafts security policies, sets the baseline, and defines the configuration requirements of the system. In many organizations, the role of system owner and system administrator are combined.

User

The user or operator is the individual who uses a deployed computer system to perform assigned day-to-day work tasks. The user has the responsibilities to accomplish their assigned work activities, perform those actions within the confines of the security policy, and report any suspicious or abnormal events to the security staff.

Privileged user

A privileged user is someone granted additional capabilities, permissions, privileges, and user rights beyond those of a typical standard user. A privileged user is often referred to as an administrator. A privileged user or administrator may have special powers over an area of the environment, such as user accounts or the database or the firewall, but should not have special powers over all areas throughout the IT infrastructure (since this would be a violation of the concept of separation of duties).

Executive user

An executive user is someone in a corporate management position who is granted additional privileges and capabilities beyond those of a typical standard user but likely less than an administrator or privileged user. Care should be given to how much extra power is provided an executive user, especially when the individual is not well versed in security and IT protections.

NDA

An *NDA (nondisclosure agreement)* is a contract that prohibits specific confidential, secret, proprietary, and/or personal information from being shared or distributed outside of a specific prescribed set of individuals or organizations. Many employees must, upon being hired, sign an NDA that prohibits them from disclosing internal details to any outside entity, either while they are active employees or after their employment with the organization has ended.

Onboarding

See the section "Onboarding/offboarding" in Chapter 4, "Identity and Access Management," for a description of this personnel management operation.

Continuing education

Security is useless if users aren't properly trained to perform their work tasks within the confines of the secured environment. Security training for employees is essential to the success of any security endeavor. It should be part of your security policy and business operations. This should include communication, awareness training, education, and support through online resources. *Continuing education* of employees should primarily focus on improving efficiency, productivity, and security compliance. However, it can also enable employees to improve their knowledge base and job skills in order to advance within the organization or leave to pursue other external opportunities.

As a security professional in any organization, you must keep the lines of communication open. This means you should be up front about security requirements for all personnel. Clearly train users on how to perform their work tasks while maintaining security.

As a manager, be open to discussing security issues with users from every level and classification. Be ready to discuss good and bad aspects of security. And be willing to assist users in learning to be efficient and productive while sustaining security. By keeping communication open and abundant, you can help prevent users from giving up on security and intentionally bypassing security procedures.

User awareness is an effort to make security a regular thought for all employees. It begins with security training and orientation when a new worker is hired. However, user awareness must continue throughout the life of the organization. It includes regular reminders, refresher seminars, emails with security updates, newsletters, intranet websites, posters with security facts or rules, and so on—whatever is necessary to keep users aware of the importance of security.

If an organization fails to maintain user awareness of security, it will experience a slow erosion of security that may ultimately allow a serious intrusion to occur. Unfortunately, user awareness of security is generally the most overlooked element of security management. In fact, the lack of security awareness is the primary reason that social engineering attacks succeed. With proper information, users can be equipped to recognize social engineering attacks and avoid being taken in by them.

Education is broad security training, usually focused on teaching users to perform their work tasks securely. Security education has the ultimate goal of certification. *Certification* is the act of passing one or more exams in order to earn certification credentials (impressive acronyms to add to your résumé), which verify that you possess certain knowledge, skills, and expertise.

Security documentation, especially work task–specific instructions, should be posted to an intranet website for easy access. In this way, users can keep themselves current on changing security policies and procedures. In contrast, security documentation about known vulnerabilities and active investigations should almost never be made available to everyone (even all internal employees).

Acceptable use policy/rules of behavior

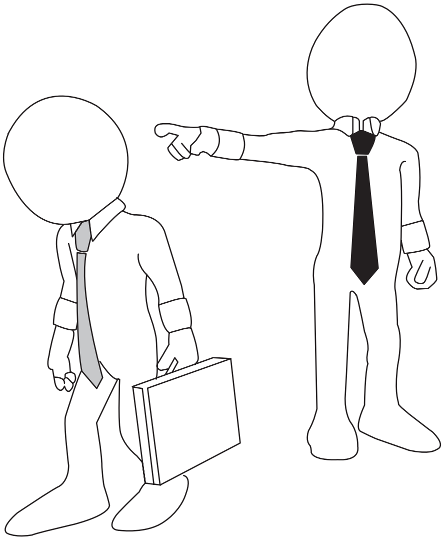
An *acceptable use policy (AUP)* defines what is and what is not an acceptable activity, practice, or use for company equipment and resources. The acceptable use policy is specifically designed to assign security roles within the organization as well as prescribe the responsibilities tied to those roles. This policy defines a level of acceptable performance and expectation of behavior and activity. Failure to comply with the policy may result in job action warnings, penalties, or termination.

Not having an acceptable use policy leads many users to the false assumption that any activity is permitted and that they enjoy privacy even on company equipment. However, there is often little to no privacy on company equipment. Although this varies by country, companies often have the right to audit, monitor, and record all activities that occur using their equipment and access services. An acceptable use policy (in addition to the privacy policy) outlines these monitoring tactics, dictates what users can and can't do, and clearly states that users don't have privacy. Often, employees must read and sign an acceptable use policy as part of the hiring and training process.

Adverse actions

Adverse actions are the consequences of failing to abide by company policies or actively committing criminal violations within the organization. Adverse actions can include lectures from management, retraining, requirements to make up work during evening or weekend hours, having pay reduced or held back, having a reduction or increase in work hours, loss of benefits, being skipped over for promotion or advancement, having to pay fees or penalties, facing termination (Figure 5.2), or even facing criminal or civil prosecution. Most company security policies, especially those such as the AUP that dictate employee activities and behaviors, will often include prescriptions of consequences for violations of the rules and tenets of the organization.

FIGURE 5.2 Termination can be a consequence of security policy violation.



General security policies

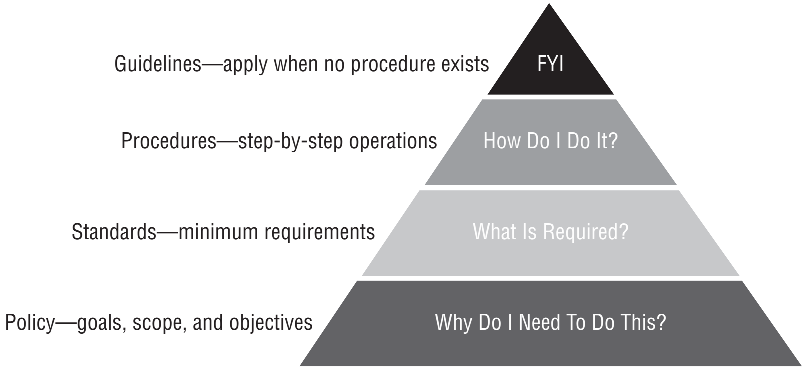
The top tier of formalizing an organization's essential protection-plan documentation is known as a *security policy*. A security policy:

- Is a document that defines the scope of security needed by the organization and discusses the assets that need protection and the extent to which security solutions should go in order to provide the necessary protection
- Is the foundational element of any successful security endeavor
- Is an overview or generalization of an organization's security needs
- Defines the main security objectives and outlines the security framework of an organization
- Identifies the major functional areas of data processing and clarifies and defines all relevant terminology
- Should clearly define why security is important and what assets are valuable
- Is a strategic plan for implementing security
- Should broadly outline the security goals and practices that should be employed to protect the organization's vital interests
- Discusses the importance of security to every aspect of daily business operations and the importance of the support of senior staff for the implementation of security
- Is used to assign responsibilities, define roles, specify audit requirements, outline enforcement processes, indicate compliance requirements, and define acceptable risk levels
- Is often used as proof that senior management has exercised due care in protecting itself against intrusion, attack, and disaster
- Is compulsory

Many organizations employ several types of security policies to define or outline their overall security strategy (Figure 5.3). An organizational security policy focuses on issues relevant to every aspect of an organization. An issue-specific security policy focuses on a specific network service, department, function, or other aspect that is distinct from the organization as a whole. A system-specific security policy focuses on individual systems or types of systems and prescribes approved hardware and software, outlines methods for locking down a system, and even mandates firewalls or other specific security controls.

In addition to these focused types of security policies, there are three overall categories of security policies: regulatory, advisory, and informative. A *regulatory* policy is required whenever industry or legal standards are applicable to your organization. This policy discusses the regulations that must be followed and outlines the procedures that should be used to elicit compliance. An *advisory* policy discusses behaviors and activities that are acceptable and defines consequences of violations. It explains senior management's desire for security and compliance within an organization. Most policies are advisory. An *informative* policy is designed to provide information or knowledge about a specific subject, such as company goals, mission statements, or how the organization interacts with partners and customers. An informative policy provides support, research, or background information relevant to the specific elements of the overall policy.

FIGURE 5.3 Components of a security policy


Guidelines—apply when no procedure exists
FYI
Procedures—step-by-step operations
How Do I Do It?
Standards—minimum requirements
What Is Required?
Policy—goals, scope, and objectives
Why Do I Need To Do This?

From the security policies flow many other documents or subelements necessary for a complete security solution. Policies are broad overviews, whereas standards, baselines, guidelines, and procedures include more specific, detailed information about the actual security solution.

You should document everything as a standard rule of thumb for security. Documentation is often seen as a keystone of security. Hence, you need to fully write out all security elements into security policies, standards, baselines, guidelines, and procedures. If you have exhaustive documentation, there will always be a detailed record of configurations, actions, procedures, and so on, which will assist you in the event of an incident, a disaster, or an implementation change.

With complete, detailed, exhaustive documentation, every aspect of your environment and every event in your secured environment is known. Documentation can be reviewed and referenced as new incidents or conditions arise. With proper documentation, the security of an organization is easier to maintain.

Social media networks/applications

Social media networks such as Facebook, Twitter, and LinkedIn, as well as social media applications such as Instagram, WeChat, and Pinterest, can be very useful tools for both individuals and organizations. These social media services and software can be used to distribute messages, attract new customers, provide support, increase market exposure, and much more. However, unlike with traditional advertising media, such as print, audio, and video ads, organizations do not have full control over the message received by the public. There is the risk that the public will view a message they don't agree with or simply use your platform to direct attention to their own areas of interest. When attempting to use social media as an interface to customers, clients, and the public, be cautious—and be prepared when your message gets lost in the noise.

If interacting with current or future customers through Internet-based services is important to your organization, you can choose to brave the risks of public social networks or host your own services. Self-hosted services can include discussion forums, text chats, and videoconferencing. When the organization is in full control of the medium as well as the message, it can tamp down any unwanted counter-messages.

Personal email

Many organizations have a security policy regarding use of or access to personal email while on company equipment. Some organizations strictly prohibit the use of personal email, whereas others allow access as long as it does not impose a risk to the organization or serve as a significant distraction from accomplishing work tasks. When workers access personal email, there may be a higher risk of system exposure to malicious attachments, malicious scripting in message bodies, access to malicious or inappropriate content via hyperlinks, or disclosure of confidential company data. Thus, it may be a more secure idea for the organization to prohibit personal email use on company equipment.

Workers may prefer accessing personal email on their own personal devices rather than through company equipment in order to limit the exposure to company filters, blocking, and auditing. Any data sent through or stored on company systems can be viewed, accessed, and retained by the organization. Thus, in order to maximize personal privacy protections, workers should restrict personal communication activities to their own devices.

Exam Essentials

Define SOP. A standard operating procedure (SOP) is an organizational policy that provides detailed or granular step-by-step instructions to accomplish a specific task. The goal of an SOP is to improve consistency in worker activities, especially as related to performance and security compliance.

Understand the security implications of integrating systems and data with third parties. Whenever a third party is involved in your IT infrastructure, there is an increased risk of data loss, leakage, or compromise. The security implications of integrating systems and data with third parties need to be considered carefully before implementation.

Comprehend interoperability agreements. Interoperability agreements are formal contracts (or at least written documents) that define some form of arrangement where two entities agree to work with each other in some capacity.

Define BPAs. A business partners agreement (BPA) is a contract between two entities, dictating their business relationship.

Define SLAs. A service-level agreement (SLA) is a contract between a supplier and a customer.

Define ISAs. An interconnection security agreement (ISA) is a formal declaration of the security stance, risks, and technical requirements of a link between two organizations' IT infrastructures.

Define MOUs. A memorandum of understanding (MOU) is an expression of agreement or aligned intent, will, or purpose between two entities.

Understand user habits. Implementing proper security involves using technology but also mandates the modification of user behaviors. If personnel do not believe in and support security, they are often opposed to the best security efforts of the organization. This includes addressing the issues of password behaviors, data handling, clean-desk policies, preventing tailgating, and personally owned devices.

Comprehend job rotation. Job rotation serves two functions: it provides a type of knowledge redundancy, and moving personnel around reduces the risk of fraud, data modification, theft, sabotage, and misuse of information.

Know about mandatory vacations. Mandatory vacations of one to two weeks are used to audit and verify the work tasks and privileges of employees. This often results in detection of abuse, fraud, or negligence.

Understand the importance of separation of duties. Separation of duties is the division of administrator or privileged tasks into distinct groupings, with each group in turn assigned to unique administrators. The application of separation of duties results in no single user having complete access to or power over an entire network, server, or system.

Define clean-desk policy. A clean-desk policy is used to instruct workers how and why to clean off their desks at the end of each work period.

Understand background checks. Background checks are used to verify that a worker is qualified for a position but not disqualified.

Know about exit interviews. An exit interview is a controlled and respectful process of termination or employee firing. The goal of an exit interview is to control the often emotionally charged event of a termination in order to minimize property damage, information leakage, or other unfortunate or embarrassing occurrences.

Understand role-based training. Role-based training involves teaching employees to perform their work tasks and to comply with the security policy.

Comprehend security education. Education means security training, usually focused on teaching a user to perform their work tasks securely. Security education is broader and has the ultimate goal of certification.

Understand user awareness. User awareness is an effort to make security a common and regular thought for all employees. Unfortunately, user security awareness is generally the most overlooked element of security management. The lack of security awareness is the primary reason social engineering attacks succeed.

Define NDA. An NDA (nondisclosure agreement) is a contract that prohibits specific confidential, secret, proprietary, and/or personal information from being shared or distributed outside of a specific prescribed set of individuals or organizations.

Understand what an acceptable use policy is. An acceptable use policy defines what is and what is not an acceptable activity, practice, or use for company equipment and resources.

Comprehend what a security policy is. A security policy is the overall purpose and direction of security in an environment, as well as the detailed procedural documents that indicate how various activities are to be performed in compliance with security.

5.2 Summarize business impact analysis concepts.

Disaster-recovery planning and procedures enable an organization to maintain or recover its mission-critical processes in spite of events that threaten its infrastructure. Maintaining business continuity means maintaining the organization's networking and IT infrastructure so that mission-critical functions continue to operate. This must be done in spite of reduced resources and damaged equipment. As long as business operations aren't stopped, business continuity is used to sustain the organization. If business operations are stopped, disaster recovery takes over.

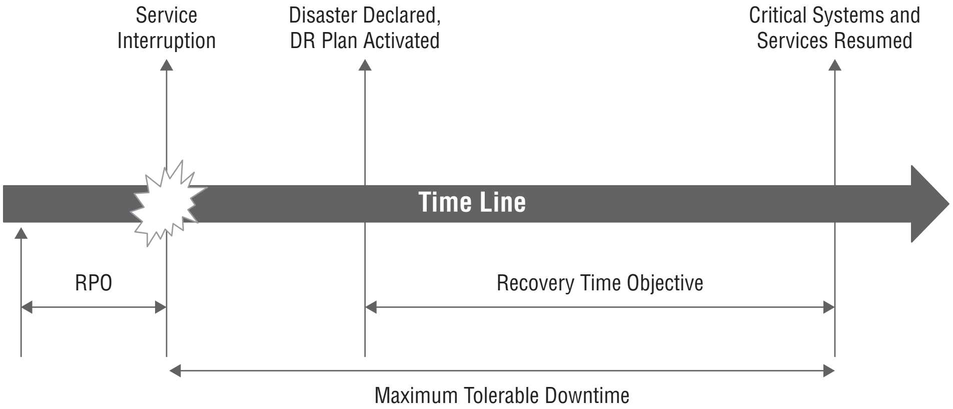
Business impact analysis (BIA) is the process of performing risk assessment on business tasks and processes rather than on assets. The purpose of BIA is to determine the risks to business processes and design protective and recovery solutions. The goal is to maintain business continuity, prevent and/or minimize downtime, and prepare for fast recovery and restoration in the event of a disaster.

The BIA identifies resources that are critical to an organization's ongoing viability and the threats posed to those resources. It also assesses the likelihood that each threat will actually occur and the impact those occurrences would have on the business. The results of this analysis provide you with quantitative measures that can help you prioritize the commitment of business continuity resources to the various risks your organization faces.

RTO/RPO

The *maximum tolerable downtime (MTD)* is the maximum length of time a business function can be inoperable without causing irreparable harm to the business. The MTD (Figure 5.4) provides valuable information when you're performing both business continuity planning (BCP) and disaster-recovery planning (DRP). Once you have defined your recovery objectives, you can design and plan the procedures necessary to accomplish the recovery tasks.

FIGURE 5.4 MTD, RTO, and RPO timeline


Service Interruption
Disaster Declared, DR Plan Activated
Critical Systems and Services Resumed
Time Line
RPO
Recovery Time Objective
Maximum Tolerable Downtime

This leads to another metric, the *recovery time objective (RTO)*, for each business function. This is the amount of time in which you think you can feasibly recover the function in the event of a disruption. The goal of the BCP process is to ensure that your RTOs are less than your MTDs, resulting in a situation in which a function should never be unavailable beyond the maximum tolerable downtime.

A metric related to RTO is the *recovery point objective (RPO)*. The RPO is a measurement of how much loss can be accepted by the organization when a disaster occurs. This acceptable loss is measured in time. The RPO measurement is independent from RTO. For example, if an organization can survive only two hours of lost data, then the RPO is two hours. The RPO is a measurement of how much data can be lost prior to the point in time of a disaster, whereas the RTO is how much time after the disaster the company has to recover operations. Generally, backup systems are designed to prevent data loss over the RPO limit, and recovery solutions are designed to return things to normal before the RTO is exceeded.

MTBF

Aging hardware should be scheduled for replacement and/or repair. The schedule for such operations should be based on the *mean time to failure (MTTF)*, *mean time between failures (MTBF)*, and *mean time to repair/restore (MTTR)* estimates established for each device or on prevailing best organizational practices for managing the hardware life cycle. MTTF is the expected typical functional lifetime of the device, given a specific operating environment. MTBF is the expected typical time lapse between failures, such as between the first failure and the second failure. If the MTTF and MTBF are the same values (or nearly so), some manufacturers list only the MTBF rating and use it to address both concepts. MTTR is the average length of time required to perform a repair on the device. A device can often undergo numerous repairs before a catastrophic failure is expected. Be sure to schedule all devices to be replaced before their MTTF expires.

When a device is sent out for repairs, you need to have an alternate solution or a backup device to fill in for the duration of the repair time. Often, waiting until a minor failure occurs before a repair is performed is satisfactory, but waiting until a complete failure occurs before replacement is a risky security practice.

MTTR

See the previous section, “MTBF.”

Mission-essential functions

Mission-essential functions or *mission-critical functions* are any core business tasks that are central to the operation of the organization. These are the functions, processes, or tasks that if stopped, interrupted, or terminated may cause the overall failure of the entire organization. Some organizations can survive for a brief period of time without functional mission-critical processes, but the MTD of these situations is often quite short.

Identification of critical systems

In the process of evaluating risk and determining the best response to it, the critical elements of an organization need to be identified. Mission-critical systems, functions, processes, or tasks are the core components of an organization. Without the mission-critical operations, the organization would cease to exist. The most critical systems and components are identified via the BIA process. BIA is effectively the same process as risk assessment. The only significant difference is that risk assessment focuses on assets, whereas BIA focuses on business tasks. For each business task, an ALE (annualized loss expectancy) is calculated (see the later section "ALE"). The processes, systems, or components that have the largest ALE are the elements most critical to the organization.

Single point of failure

A *single point of failure* is any individual or sole device, connection, or pathway that is moderately to mission-critically important to the organization. If that one item fails, the whole organization suffers loss. Infrastructures should be designed with redundancies of all moderately or highly important elements in order to avoid single points of failure. Removing single points of failure involves adding redundancy, recovery options, or alternative means to perform business tasks and processes. Avoiding or resolving single points of failure will improve stability, uptime, and availability.

Impact

Impact is a measurement of the amount of damage or loss that could be or would be caused if a potential threat is ever realized. The impact of a threat is indicated by the *EF (exposure factor*): the percentage of asset value loss that would occur if a risk was realized (for example, if an attack took place).

Life

Any time human life is at risk related to a business task or function, it is imperative to implement protections to reduce or eliminate the possibility of loss of life. Risks that may have a serious impact such as loss of life include natural disasters, fire, and man-made attacks (such as explosions, poisons, and gunfire). Most organizations endeavor to reduce life-threatening risks through prevention, detection, and response means, such as fire prevention, improved detection, escape plans, availability of handheld extinguishers, and building suppression systems.

Property

Property damage is a potential impact of a wide range of physical security breaches, but might also be a result of social engineering and logical/technical attacks. Property damage includes facility and equipment damage as well as loss of utilities, communications, networking, and storage systems.

Safety

Safety relates to the well-being of personnel. Safety violation defenses should include protections against physical and physiological harm to individuals. Organizations should provide a safe work environment, such as proper lighting; stable flooring; handrails on stairways; markings where vehicles travel; and safety stations for first aid, eye wash, or chemical exposure cleanup.

Finance

A security breach can have an impact on finances. Recovering from an intrusion can be expensive, especially if hardware must be replaced, if lawsuits are filed against the organization, and if extensive investigation is needed to weed out damaging or malicious code or to collect evidence. Some organizations may be able to obtain hacker, intrusion, or security breach insurance, but care should be taken to evaluate its cost-effectiveness and whether it covers any issues that might actually be experienced by the organization.

Reputation

A security breach can have a detrimental effect on an organization's reputation. If a security breach becomes public knowledge, then news outlets and Internet bloggers might emphasize the negative aspects of the intrusion rather than focusing on successful detection, response, and recovery. Such negative press may be able to sway public opinion against the organization, which can have a direct negative effect on sales and future business.

Privacy impact assessment

A *privacy impact assessment (PIA)* is a tool used to determine privacy risks, how to mitigate those risks, and whether to notify the affected parties as related to a new or future project or endeavor. A PIA should be drafted when an organization will be collecting PII from its employees and/or its customers through a new software application, outreach program, or any other type of real-world or digital interaction. The PIA should be used to determine what specific PII elements will be collected and for what purposes. The PIA should then evaluate the means of collection, storage, protection, use, distribution, access, and sharing of the PII.

The goals of a PIA are to ensure compliance with privacy laws, regulations, contracts, and policies; to predict risks and consequences of breaches; and to assist in the evaluation of the effectiveness of protections and determine additional safeguards to implement.

For examples of PIAs from the Department of Homeland Security (DHS), please see `https://www.dhs.gov/privacy-impact-assessments`.

Privacy threshold assessment

A *privacy threshold assessment (PTA)* is used to evaluate the data that an organization has already collected to determine whether such data is PII, business confidential, or nonsensitive data. If PII is discovered, then its source must be determined and its intended uses uncovered. If necessary, additional safeguards to protect against PII distribution are to be implemented. The PTA can then be used to determine whether the PII is a legitimate asset and how it should be managed, or whether the PII is illegitimate and should be purged and the related parties notified.

Exam Essentials

Define BIA. Business impact analysis (BIA) is the process of performing risk assessment on business tasks and processes rather than on assets. The purpose of BIA is to determine the risks to business processes and design protective and recovery solutions.

Understand MTD. The maximum tolerable downtime (MTD) is the maximum length of time a business function can be inoperable without causing irreparable harm to the business.

Know about RTO and RPO. Recovery time objective (RTO) is the amount of time in which you think you can feasibly recover the function in the event of a disruption. Recovery point objective (RPO) is a measurement of how much loss can be accepted by the organization when a disaster occurs.

Define MTTF, MTBF, and MTTR. Aging hardware should be scheduled for replacement and/or repair. The schedule for such operations should be based on the mean time to failure (MTTF), mean time between failures (MTBF), and mean time to repair/restore (MTTR) estimates established for each device- or on prevailing best organizational practices for managing the hardware life cycle.

Understand mission-essential functions. Mission-essential functions or mission-critical functions are any business tasks that are core and central to the operation of the organization.

Comprehend single point of failure. A single point of failure is any individual or sole device, connection, or pathway that is moderately to mission-critically important to the organization.

Know about impact. Impact is a measurement of the amount of damage or loss that could be or would be caused if a potential threat is ever realized. The impact of a threat is indicated by the exposure factor (EF): the percentage of asset value loss that would occur if a risk was realized.

Define PIA. A privacy impact assessment (PIA) is a tool used to determine privacy risks, how to mitigate those risks, and whether to notify the affected parties as related to a new or future project or endeavor.

Define PTA. A privacy threshold assessment (PTA) is used to evaluate the data that an organization has already collected to determine whether such data is PII, business confidential, or nonsensitive data. If PII is discovered, then its source must be determined and its intended uses uncovered. If necessary, additional safeguards to protect against PII distribution are to be implemented.

5.3 Explain risk management processes and concepts.

Security is aimed at preventing loss or disclosure of data while sustaining authorized access. The possibility that something could happen to damage, destroy, or disclose data or other resources is known as *risk*.

Managing risk is therefore an element of sustaining a secure environment. Risk management is a detailed process of identifying factors that could damage or disclose data, evaluating those factors in light of data value and countermeasure cost, and implementing cost-effective solutions for mitigating or reducing risk. The overall process of risk management is used to develop and implement information security strategies. The goals of these strategies are to reduce risk and to support the mission of the organization.

Thus, the primary goal of risk management is to reduce risk to an acceptable level. What that level actually is depends on the organization, the value of its assets, the size of its budget, and many other factors. What is deemed acceptable risk to one organization may be an unreasonably high level of risk to another. It is impossible to design and deploy a totally risk-free environment; however, significant risk reduction is possible, often with little effort.

Risks to an IT infrastructure are not all computer based. In fact, many risks come from nontechnical sources. It is important to consider all possible risks when performing risk evaluation for an organization. When it fails to properly evaluate and respond to all forms of risk, a company remains vulnerable. Keep in mind that IT security, commonly referred to as *logical* or *technical* security, can provide protection only against logical or technical attacks. To protect IT against physical attacks, physical protections must be erected.

The process by which the goals of risk management are achieved is known as *risk analysis*. It includes analyzing an environment for risks, evaluating each risk as to its likelihood of occurring and the cost of the damage it would cause if it did occur, assessing the cost of various countermeasures for each risk, and creating a cost/benefit report for safeguards to present to upper management. In addition to these risk-focused activities, risk management also requires evaluation, assessment, and the assignment of value for all assets within the organization. Without proper asset valuations, it is not possible to prioritize and compare risks with possible losses.

Threat assessment

Despite our best wishes, disasters of one form or another eventually strike every organization. Whether it's a natural disaster, such as a hurricane or earthquake, or a man-made calamity, such as a building fire or burst water pipes, every organization encounters events that threaten their very existence. Strong organizations have plans and procedures in place to help mitigate the effects a disaster has on their continuing operations and to speed the return to normal operations.

Environmental

Environmental threats are natural disasters that are triggered by Mother Nature. Environmental threats can result in minor interruptions to business tasks, significant damage to systems and infrastructure, or total disaster for an organization.

Manmade

Man-made threats are any event or occurrence caused by humans. This can range from acts of war and terrorism to data theft, sabotage, and embezzlement.

Internal vs. external

Organizations need to treat any and all threats seriously. They need to understand that there are significant security risks that originate both internally and externally. Outside attackers are often given more media time, but it is the insider threats that have the more likely chance to cause significant and lasting damage to an organization.

Risk assessment

Risk identification and risk calculation are essential parts of an organization's security endeavor. Without performing a risk assessment and analysis, you won't know what problems your security policy needs to address. Computer systems and networks can never be completely secure. However, that fact shouldn't prevent you from securing your environment as much as possible. By using asset identification, risk assessment, threat identification, and vulnerability management, you can focus your security endeavors on those areas that pose the greatest threat to your assets.

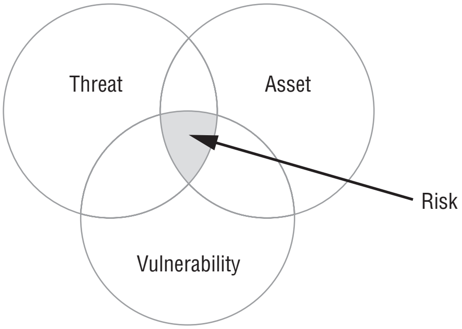
You don't know what to protect if you don't know what you have. A thorough asset inventory must be performed to identify mission-critical systems as well as everyday items (such as paper clips and sticky notes) that your organization needs to perform its services and produce its products. Once you have a master inventory, you can prioritize your assets. You can then perform risk assessment for the most important items first; after you provide additional protection for them, you can move on to less important items.

The goal of risk assessment, risk management, and risk analysis is to minimize the impact of risks on an organization. This is done through *mitigation* (applying safeguards or countermeasures), *transfer* or *assignment* (outsourcing or obtaining insurance), or *acceptance* (accepting the potential losses). This process identifies potential threats, evaluates the potential impact of those threats, and weighs the cost in terms of protection mechanisms needed and the potential loss or interruption of business continuity.

A *threat* is any person or tool that can take advantage of a vulnerability. Threat identification is a formal process of outlining the potential threats to a system. A *threat vector* is the path or means by which an attack can gain access to a target in order to cause harm. This is also known as the *attack vector*. *Threat probability* or *threat likelihood* is a calculation of the potential for a threat to cause damage to an asset.

A *vulnerability* is a weakness, an error, or a hole in the security protection of a system, a network, a computer, software, and so on. When a vulnerability exists, threats may exist to exploit it. A vulnerability allows for harm to occur when a threat is realized. You can use countermeasures and safeguards to patch vulnerabilities. If a vulnerability is patched, then that threat no longer poses a danger to your systems. However, although many vulnerabilities can be addressed with a patch, some are in legacy systems and are unpatchable. Others may be at the protocol or design levels and a new design must be created that is not vulnerable to that attack. Figure 5.5 shows the relationship between asset, threat, vulnerability, and risk.

FIGURE 5.5 Relationship of asset, threat, vulnerability, and risk



Risk awareness involves evaluating assets, vulnerabilities, and threats in order to clearly define an organization's risk levels. There are several important values and formulas involved in risk calculations; these are discussed in the following sections.

SLE

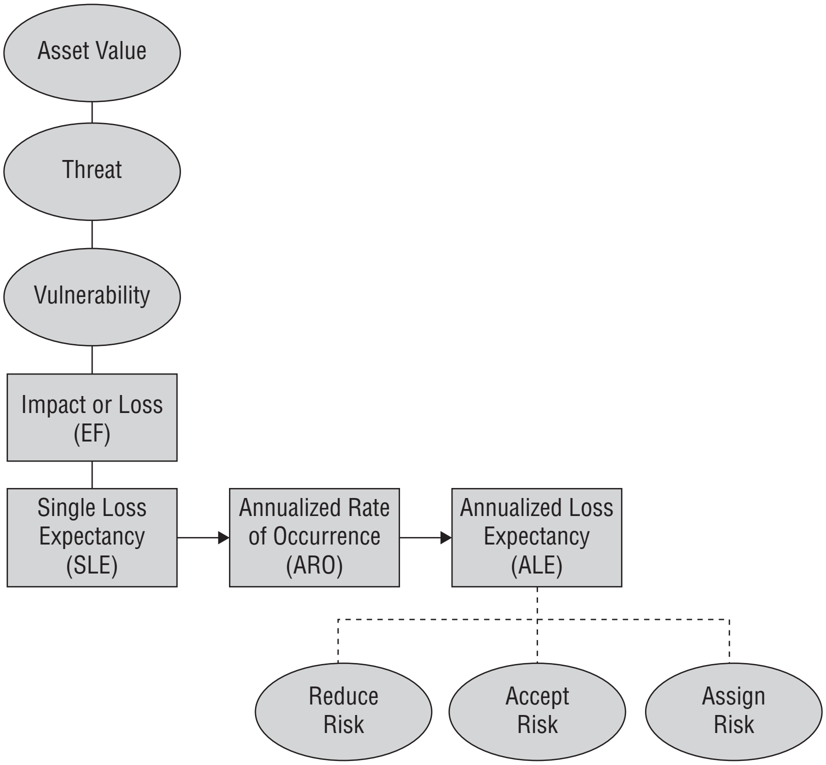
Single loss expectancy (SLE) is the potential dollar value loss from a single risk-realization incident. It's calculated by multiplying the EF by the asset value. The *exposure factor (EF)* is the percentage of asset value loss that would occur if a risk was realized (if an attack took place). EF is calculated by using historical data from previous occurrences in order to predict the amount or percentage of loss that might occur when and if the threat causes harm in the future.

ALE

Annualized loss expectancy (ALE) is the potential dollar value loss per year per risk. It's calculated by multiplying the SLE by the ARO (see the next section).

Once an ALE is calculated for each asset and the related threat to that asset (Figure 5.6), the ALEs are ordered from biggest to smallest. This establishes a relative measurement of the biggest risk to the organization versus the smallest. From this ordered priority list, security solutions are designed, starting from the top.

FIGURE 5.6 Assets to ALE calculation


Asset Value
Threat
Vulnerability
Impact or Loss (EF)
Single Loss Expectancy (SLE)
Annualized Rate of Occurrence (ARO)
Annualized Loss Expectancy (ALE)
Reduce Risk
Accept Risk
Assign Risk

Most organizations do not have an unlimited budget, especially in the area of security. Thus, prioritizing security dollars is important. Security controls should be implemented based on risk. Once an ALE has been calculated for each asset and threat, a priority order of need is established. The combination of asset and threat that produces the largest ALE is the most important security concern for the organization. A security solution should be selected based on the control with the most favorable cost-benefit result.

Selecting a countermeasure within the realm of risk management relies heavily on the cost-benefit analysis results. However, you should consider several other factors:

- The cost of the countermeasure should be less than the value of the asset.
- The cost of the countermeasure should be less than the benefit of the countermeasure.
- The result of the applied countermeasure should make the cost of an attack greater for the perpetrator than the derived benefit from an attack.
- The countermeasure should provide a solution to a real and identified problem. (Don't install countermeasures just because they are available, are advertised, or sound cool.)

- The benefit of the countermeasure should not be dependent on its secrecy. This means “security through obscurity” is not a viable countermeasure and that any viable countermeasure can withstand public disclosure and scrutiny.
- The benefit of the countermeasure should be testable and verifiable.
- The countermeasure should provide consistent and uniform protection across all users, systems, protocols, and so on.
- The countermeasure should have few or no dependencies to reduce cascade failures.
- The countermeasure should require minimal human intervention after initial deployment and configuration.
- The countermeasure should be tamper-proof.
- The countermeasure should have overrides accessible to privileged operators only.
- The countermeasure should provide fail-safe and/or fail-secure options.

Fortunately, you do not need to select an individual countermeasure for every ALE. As priority ALEs are addressed, those countermeasures will also address numerous lesser ALE concerns. Each time the top ALE asset or threat is resolved, the overall list of remaining issues will shrink.

ARO

Annualized rate of occurrence (ARO) is the statistical probability that a specific risk may be realized a certain number of times in a year (often written as #/year). It’s obtained from a risk assessment company, from an insurance company’s actuarial tables, through analyzing internal historical records, or sometimes by guessing. The ARO is an assessment of how many times a threat has the opportunity to cause harm and how often that harm might occur in the future.

Asset value

Asset value (often written as AV) is the value or worth of an asset to an organization. It is a calculation based on a mixture of tangible and intangible value, expense, and costs. AV predicts the amount of loss the organization would suffer if the asset was no longer available or viable.

Risk register

A *risk register* or risk log is a document that inventories all of the identified risks to an organization or system or within an individual project. A risk register is used to record and track the activities of risk management, including:

1. Identify risks.
2. Evaluate the severity and prioritize those risks.
3. Prescribe responses to reduce or eliminate the risks.
4. Track the progress of risk mitigation.

Likelihood of occurrence

Likelihood is the measurement of probability that a threat will become realized within a specific period of time. Within the scope of risk assessment, likelihood is measured on a yearly basis. This measurement is called the ARO. See the earlier section "ARO."

Supply chain assessment

When evaluating organizational risk, it is important to consider external factors that can affect the organization, especially related to company stability and resource availability. An organization's supply chain should be assessed in order to determine what risks it places on the organization. Is the organization operating on a just-in-time basis where materials are delivered just before or just as they are needed by manufacturing? If there is any delay in delivery, is there any surplus or buffer of materials that can be used to maintain production while the supply chain operations are reconstituted?

Impact

The goal of risk management is to minimize impact. This may be accomplished through prevention of risk realization or through prompt and sufficient responses to harm. See the earlier section "Impact" under objective 5.2, "Summarize business impact analysis concepts."

Quantitative

Most risk management processes start with taking an inventory of assets. Next, each asset is assigned an AV. Then, the threats for each asset are considered. Each asset-threat pairing should be individually evaluated for EF, ARO, and ultimate impact on the organization. There are two risk assessment methodologies: quantitative and qualitative. *Quantitative* risk analysis assigns real dollar figures to the loss of an asset. *Qualitative* risk analysis assigns subjective and intangible values to the loss of an asset. Both methods are necessary for a complete risk analysis.

The quantitative method results in concrete probability percentages. That means it creates a report that has dollar figures for levels of risk, potential loss, cost of countermeasures, and value of safeguards. This report is usually fairly easy to understand, especially for anyone with knowledge of spreadsheets and budget reports. Think of quantitative analysis as the act of assigning a quantity to risk: in other words, placing a dollar figure on each asset and threat. However, a purely quantitative analysis is not possible; not all elements and aspects of the analysis can be quantified, because some are qualitative, subjective, or intangible. The process of quantitative risk analysis starts with asset valuation and threat identification. Next, you estimate the potential and frequency of each risk. This information is then used to calculate various cost functions that are used to evaluate safeguards.

The six major steps or phases in quantitative risk analysis are as follows:

1. Inventory assets and assign a value (AV).
2. Research each asset and produce a list of all possible threats to each individual asset. For each listed threat, calculate the EF and SLE.
3. Perform a threat analysis to calculate the likelihood of each threat being realized within a single year—that is, the ARO.

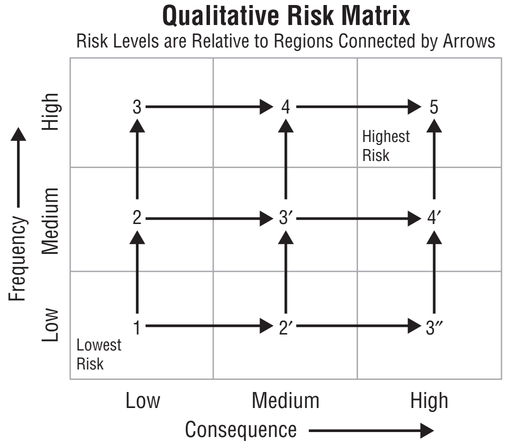
4. Derive the overall loss potential per threat by calculating the ALE.
5. Research countermeasures for each threat, and then calculate the changes to ARO and ALE based on an applied countermeasure.
6. Perform a cost-benefit analysis of each countermeasure for each threat for each asset. Select the most appropriate response to each threat.

Qualitative

Qualitative risk analysis is more scenario-based than it is calculator-based. Rather than assign exact dollar figures to possible losses, you rank threats on a scale to evaluate their risks, costs, and effects. A qualitative risk matrix (Figure 5.7) can be used to show the relationships between frequency and consequence. The process of performing qualitative risk analysis involves judgment, intuition, and experience. You can use many techniques to perform qualitative risk analysis:

- Brainstorming—Collecting spontaneous ideas from a group or individual
- Delphi technique—A means by which a group reaches anonymous consensus through the use of blind votes
- Storyboarding—Drawing pictures to represent concepts and timelines
- Focus groups—Using study, research, or discussion groups centered around a single topic
- Surveys—A broad-range data-gathering technique that seeks to pull relevant information from any source
- Questionnaires—Asking a series of questions
- Checklists—An inventory list that must be assessed against a process, task, or storage
- One-on-one meeting—A meeting between peers to discuss a topic
- Interview—A face-to-face interaction with subject matter experts or those with direct experience of an event or situation

FIGURE 5.7 A qualitative risk matrix



Determining which mechanism to employ is based on the culture of the organization and the types of risks and assets involved. It is common for several methods to be used simultaneously and for their results to be compared and contrasted in the final risk-analysis report to upper management.

Testing

Risk management may require the use of testing to determine some forms or types of risks. Risk discovery or risk testing can involve penetration testing and vulnerability testing in order to include system- and software-level issues in the mitigation process.

Penetration testing authorization

Before performing criminal violation simulations, it is important to obtain penetration testing authorization from senior management. Unauthorized penetration testing is a violation of company security policy and potentially criminal activity.

Vulnerability testing authorization

Vulnerability testing authorization should be obtained before performing the security assessment. Although vulnerability testing is not usually associated with lost data or system crashing, it is often a prohibited task for the typical user. That's because it could reveal to a user a flaw that could be used to cause serious harm to the organization. Thus, most organizations do not allow anyone to do vulnerability testing unless specifically authorized.

Risk response techniques

The documented results of risk analysis are many:

- A complete and detailed valuation of all assets
- An exhaustive list of all threats and risks, rates of occurrence, and extent of losses if realized
- A list of threat-specific safeguards and countermeasures that identifies their effectiveness and ALEs
- A cost-benefit analysis of each safeguard

This information is essential for management to make educated, intelligent decisions about safeguard implementation and security policy alterations.

Once the risk analysis is complete, management must address each specific risk. There are several possible responses to risk:

- Accept or tolerate
- Assign or transfer
- Avoid
- Reduce or mitigate
- Reject or ignore

Accept, transfer, avoid, and mitigate risk responses are covered in the following subsections.

A final but unacceptable possible response to risk is to *reject risk* or *ignore risk*. Denying that a risk exists or hoping that it will never be realized is not a valid, prudent, due-care response to risk.

Once countermeasures are implemented, the risk that remains is known as *residual risk*. Residual risk consists of any threats to specific assets against which upper management chooses not to implement a safeguard. In other words, residual risk is the risk that management has chosen to accept rather than mitigate. In most cases, the presence of residual risk indicates that the cost-benefit analysis showed that the available safeguards were not cost-effective deterrents.

Total risk is the amount of risk an organization would face if no safeguards were implemented. A formula for total risk is *threats * vulnerabilities * asset value = total risk*. (Note that the * here does not imply multiplication, but a combination function; this is not a true mathematical formula.) The difference between total risk and residual risk is known as the *controls gap*: the amount of risk that is reduced by implementing safeguards. A formula for *residual risk* is *total risk – controls gap = residual risk*.

As with risk management in general, handling risk is not a one-time process. Instead, security must be continually maintained and reaffirmed. In fact, repeating the risk assessment and analysis process is a mechanism to assess the completeness and effectiveness of the security program over time. Additionally, it helps locate deficiencies and areas where change has occurred. Because security changes over time, reassessing on a periodic basis is essential to maintaining reasonable security.

Obviously, there is more to properly managing risk than slapping on a patch. Risk management is a detailed, rigorous process that should be performed periodically to assess the state of an organization's security.

Accept

Accepting risk or *tolerating risk* is the valuation by management of the cost-benefit analysis of possible safeguards and the determination that the cost of the countermeasure greatly outweighs the possible cost of loss due to a risk. It also means management has agreed to accept the consequences and the loss if the risk is realized. In most cases, accepting risk requires a clearly written statement that indicates why a safeguard was not implemented, who is responsible for the decision, and who will be responsible for the loss if the risk is realized, usually in the form of a "sign-off" letter. An organization's decision to accept risk is based on its risk tolerance. *Risk tolerance* is the ability of an organization to absorb the losses associated with realized risks.

Transfer

Assigning risk, or transferring risk, is placing the cost of loss that a risk represents onto another entity or organization. Purchasing insurance and outsourcing are common forms of assigning or transferring risk.

Avoid

A variation of assigning risk is risk avoidance. This is the process of selecting alternate options or activities that have less associated risk than the default, common, expedient, or cheap option. For example, choosing to fly to a destination instead of drive is a form of risk avoidance. Another example is to locate a business in Arizona instead of Florida to avoid hurricanes.

Yet another variation on risk assignment or avoidance is risk deterrence. This is the process of implementing deterrents to would-be violators of security and policy. Some examples include implementation of auditing, security cameras, security guards, motion detectors, and strong authentication and making it known that the organization is willing to cooperate with authorities and prosecute those who participate in cybercrime.

Mitigate

Reducing risk, or *risk mitigation*, is the implementation of safeguards and countermeasures to eliminate vulnerabilities or block threats. Picking the most cost-effective or beneficial countermeasure is part of risk management, but it is not an element of risk assessment. In fact, countermeasure selection is a post–risk assessment or post–risk analysis activity. Another potential variation of risk mitigation is risk avoidance. The risk is avoided by eliminating the risk cause. A simple example is removing FTP from a server to avoid FTP attacks, and a larger example is to move to an inland location to avoid the risks from hurricanes.

Change management

See the Chapter 3 section "Version control and change management." The overall goal of risk management is to understand and reduce risks. The overall goal of change management is to prevent change from causing unnecessary downtime or reductions in security. Thus, change management is an important element of any risk management strategy.

Exam Essentials

Understand risk management. Risk management is a detailed process of identifying factors that could damage or disclose data, evaluating those factors in light of data value and countermeasure cost, and implementing cost-effective solutions for mitigating or reducing risk.

Know about risk assessment. The goal of risk assessment is to minimize the impact of risks on an organization. This is done through mitigation, assignment, or acceptance. This process identifies potential threats, evaluates the potential impact of those threats, and weighs the cost in terms of protection mechanisms needed and the potential loss or interruption of business continuity.

Understand asset identification. A thorough asset inventory must be performed to identify mission-critical systems as well as everyday items (such as paper clips and sticky notes) that your organization needs to perform its services and produce its products. Once you have a master inventory, you can prioritize your assets.

Know the risk assessment formulas and variables. The different risk assessment formulas and variables are exposure factor (EF), single loss expectancy (SLE), annualized rate of occurrence (ARO), and annualized loss expectancy (ALE).

Understand threats. A threat is any person or tool that can take advantage of a vulnerability. Threat identification is a formal process of outlining the potential threats to a system.

Know about vulnerabilities. A vulnerability is a weakness, an error, or a hole in the security protection of a system, a network, a computer, software, and so on. When a vulnerability exists, threats may exist to exploit it. You can use countermeasures and safeguards to patch vulnerabilities.

Define SLE. Single loss expectancy (SLE) is the potential dollar value loss from a single risk-realization incident. It's calculated by multiplying the EF by the asset value.

Define ALE. Annualized loss expectancy (ALE) is the potential dollar value loss per year per risk. It's calculated by multiplying the SLE by the ARO.

Define ARO. Annualized rate of occurrence (ARO) is the statistical probability that a specific risk may be realized a certain number of times in a year (often written as #/year).

Understand risk register. A risk register or risk log is a document that inventories all of the identified risks to an organization or system or within an individual project.

Comprehend quantitative risk analysis. Quantitative risk analysis focuses on hard values and percentages. A complete quantitative analysis is not possible because of the intangible aspects of risk. The process involves assigning value to assets and identifying threats and then determining a threat's potential frequency and the resulting damage; the result is a cost-benefit analysis of safeguards.

Comprehend qualitative risk analysis. Qualitative risk analysis is based more on scenarios than calculations. Exact dollar figures are not assigned to possible losses; instead, threats are ranked on a scale to evaluate their risks, costs, and effects. Such an analysis assists those responsible for creating proper risk management policies.

Know the options for handling risk or responding to risk. Reducing risk, or risk mitigation, is the implementation of safeguards and countermeasures. Assigning risk or transferring a risk places the cost of loss that a risk represents onto another entity or organization. Purchasing insurance is one form of assigning or transferring risk. Accepting risk means management has evaluated the cost-benefit analysis of possible safeguards and has determined that the cost of the countermeasure greatly outweighs the possible cost of loss due to a risk. It also means management has agreed to accept the consequences and the loss if the risk is realized.

5.4 Given a scenario, follow incident response procedures.

When an incident occurs, you must handle it in a manner that is outlined in your security policy and consistent with local laws and regulations. The first step in incident management or handling an incident properly is recognizing when one occurs. You should understand the following two terms related to incident handling:

- Event—Any occurrence that takes place during a certain period of time
- Incident—An event that has a negative outcome affecting the confidentiality, integrity, or availability of an organization's data

Incident response plan

The most common reason incidents are not reported is that they are never identified. You could have many security policy violations occurring each day, but if you don't have a way of identifying them, you will never know. Therefore, your security policy should identify and list all possible violations and ways to detect them. It's also important to update your security policy as new types of violations and attacks emerge.

What you do when you find that an incident has occurred depends on the type of incident and the scope of the damage. Laws dictate that some incidents must be reported, such as those that impact government or federal interest computers (a federal interest computer is one that is used by financial institutions or by infrastructure systems such as water and power systems) or certain financial transactions, regardless of the amount of damage. Most U.S. states now have laws that require organizations that experience an incident involving certain types of personally identifying information (for example, credit card numbers, Social Security numbers, and driver's license numbers) to notify affected individuals of the breach.

In addition to laws, many companies have contractual obligations to report different types of security incidents to business partners. For example, the Payment Card Industry Data Security Standard (PCI DSS) requires any merchant that handles credit card information to report incidents involving that information to Visa, Discovery, MasterCard, or American Express promptly.

Every organization needs an *incident response plan (IRP)*. The IRP is the SOP that defines how to prevent incidents, how to detect incidents, how to respond to incidents, and how to return to normal when the incident is concluded. An incident is any violation of the company security policy. Minor incidents are handled by the deployed security infrastructure, and criminal incidents are handled by law enforcement. All other incidents must be handled by the organization in accordance with their IRP.

Documented incident types/category definitions

An incident occurs when an attack, or other violation of your security policy, is carried out against your system. There are many ways to classify incidents; here is a general list of categories:

- Scanning
- Data breach

- Malicious code
- Denial of service

These four areas are the basic entry points for attackers to impact a system. You must focus on each of these areas to create an effective monitoring strategy that detects system incidents. Each incident area has representative signatures that can tip off an alert security administrator that an incident has occurred. Make sure you know your OS environment and where to look for the telltale signs of each type of incident.

Roles and responsibilities

The IRP should detail the roles and responsibilities of each member of the incident response team. The overall responsibilities include prevention, detection, and response. A more stable and secure environment exists when incidents are prevented rather than reacted to. When an incident is not prevented, it is essential to detect it in order to be able to properly respond. Incident response usually includes containment, eradication, and restoration to normal operations (see later section "Incident response process"). There should be a primary incident response leader, there may be several individuals who can trigger the team's response, and then most of the team's roles are based on the technology of the organization and types of violations that may be experienced.

Reporting requirements/escalation

After an incident has been contained, the incident response team is responsible for fully documenting the incident and making recommendations about how to improve the environment to prevent a recurrence. The documentation or reporting of an incident is used to provide a record of the incident (for use internally or to share with outsiders), provide support for due care and due diligence defense, serve as support for security decisions, and assist with training incident response team members.

Once you have a basic understanding of what the incident consists of, you can follow a staged procedure of escalation and notification. Information about security breaches is not to be shared publicly or with the entire employee base. Instead, only those in specific positions of authority or responsibility should receive notification of breaches. This may include legal, PR, IT staff, security staff, human resources, and so on. If the incident is related to a criminal event, then contacting law enforcement is in order. As the details of an incident are uncovered, the depth, complexity, or level of damage caused may increase, thus requiring an escalation of personnel and response.

Cyber-incident response teams

Many organizations now have a dedicated team responsible for investigating any computer security incidents that take place. These teams are commonly known as *cyber incident response teams (CIRTs)* or computer security incident response teams (CSIRTs). When an incident occurs, the response team has four primary responsibilities:

- Determine the amount and scope of damage caused by the incident.
- Determine whether any confidential information was compromised during the incident.

- Implement any necessary recovery procedures to restore security and recover from incident-related damages.
- Supervise the implementation of any additional security measures necessary to improve security and prevent recurrence of the incident.

When putting together your incident response team, be sure to design a cross-functional group of individuals who represent the management, technical, and functional areas of responsibility most directly impacted by a security incident.

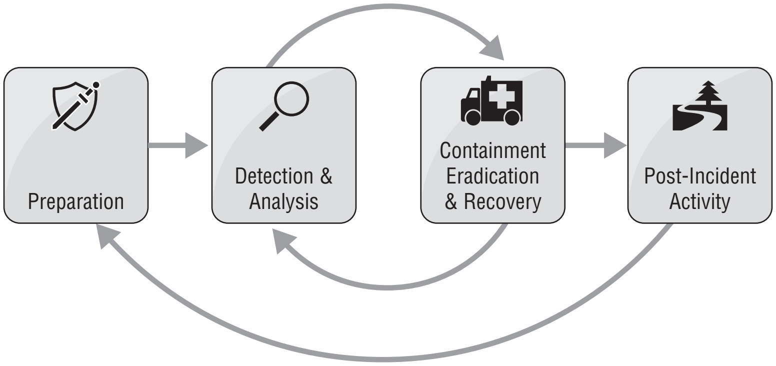
Exercise

Some key elements of CIRT preparedness are simulation, drill, and exercise. The members of the CIRT should be involved in continuing education and research to keep their knowledge current on new attacks and exploits as well as responses and repair options. But this knowledge needs to be tested in a real-world simulation before it is used in an actual incident event. Some CIRTs will have computer-based simulations, others will perform drills within the live organization, and others will have duplicate partial facilities to perform live walk-throughs of response options (sometimes called exercises).

Incident response process

An *incident response procedure* is to be followed when a security breach or security violation has occurred. One of the most important goals of incident response is *containment*: the protection and preservation of evidence. This may require taking systems offline, duplicating hard drives using imaging software, making photographs of monitor displays, documenting strange conditions or activities, disconnecting a server from the network, and so on. Figure 5.8 illustrates one strategy.

FIGURE 5.8 An incident response strategy



For end users, the incident response policy is simple and direct: they should step away from their computer system and contact the incident response team. For the computer incident response team, the incident response policy is more involved. The following sections discuss the responsibilities or concerns of a CIRT.

Preparation

Preparation is necessary to ensure a successful outcome of unplanned downtime, security breaches, or disasters. Being prepared includes defining a procedure to follow in response to incidents, buttressing an environment against incidents, and improving detection methods. Having a plan and preallocating resources to address incidents improves recovery time while minimizing loss and costs. An incident response policy should be developed that addresses the various aspects of handling incidents.

Identification

The first step in responding to an incident is to detect and become aware that an incident is occurring. Without detection, incidents would instead be false negatives (the lack of an alarm in the presence of malicious activity)—in other words, unknown unknowns. If an organization is not aware that it is actively being harmed, then it doesn't know there is a need to respond or make changes. Thus, improved means of detecting security violations is essential. This includes detailed security logging, use of IDSs and IPSs, and monitoring of performance for trends of abnormal activity levels.

Once an incident is recognized, data about the incident should be collected and documented. The incident response team should take an account of the status of the environment and attempt to deduce the cause of the incident. This will assist them in determining the scope of the concern. Many other questions may need to be asked as well, such as these:

- What systems were affected?
- Is the source internal or external?
- Is the compromise engorging or concluded?
- Was it a network traffic–based attack?
- Which subnets were affected?
- Which systems may have been accessed by the intruder?
- What resources were accessed?
- What level of privilege was used?
- What information or data was put at risk?
- Was the attack from a single source/vector or multiple?
- Is this a repeat of a previous attack?
- Was malicious code infection involved?
- Is the compromise contagious?
- Was privacy violated?
- Which other systems have similar vulnerabilities?

As these questions are answered, the information should be included in the incident documentation.

Containment

In the process of responding to an incident, important goals are to contain the problem (that is, the potential for further damage) and control or prevent loss. *Containment* means to limit the scope of damage and prevent other systems or resources from being negatively affected. Containment is especially important when the incident includes virus infection, remote control access, a Trojan horse, a logic bomb, or the use of hacker tools. Malicious use of these components may leave residual elements that are activated at a later time. Thus, after containment, eradication of malware should be part of the recovery process in order to prevent further damage or loss.

When a security breach or perimeter violation is detected, *incident response* must be initiated by the first responders. The goal of a planned and documented incident response is to limit the amount of damage caused by the incident, to recover the environment as quickly as possible, and to gather information about the incident and the perpetrator in order to prevent a recurrence and pursue legal prosecution.

The best way to respond to security violations is to have an incident response plan to follow. This plan defines and describes the procedures to perform in the event of an incident. One of the first steps that should occur when an incident is detected or suspected is to contact the incident response team, which then follows the procedures in the incident response plan.

It's important not to log off the system or shut down the computer, because these actions may damage or alter evidence. The CIRT team guides the first responder as to what action to take. When relevant, one common action is to remove the network cable but otherwise leave the affected or compromised system untouched. Often, the first responder is an end user; thus it is important that end users have proper training and awareness of how to handle and report incidents.

During the initial incident identification process, incident response team members may become aware that a system is a target or is the cause of a compromise. In these cases, alternate response actions may be considered, such as quarantine or device removal.

Quarantine sets something apart from the rest of an environment in order to provide protection and prevent interaction between the element quarantined and the environment. This technique can be implemented to protect a mission-critical server from a network compromise or to protect a network from a compromised server by quarantining the server.

Once a piece of hardware has been identified as the culprit or source of a system breach, it is often essential to remove the device from the production network. This might be performed on a temporary basis, such as placing the device in quarantine while it is being cleaned or repaired. But in most cases, device removal means removing a device that is not to be used again in production. This may be because the device is damaged or compromised beyond a reasonable ability to repair or restore the system. In such cases, the offending device may be destroyed and a new device obtained to be put into production.

Eradication

When the potential for additional damage has been eliminated (or at least significantly reduced), the process of eradication can take place. *Eradication* includes the processes used to remove or eliminate the causes of the incident, such as removing software, deleting malware, changing configurations, firing personnel, disabling compromised accounts, and blocking IP addresses and ports. In some instances, eradication is not necessary or is handled in the recovery phase. In some cases, the act of restoring systems from backups performs the eradication of the offending application or malicious software.

Recovery

Recovery is the process of removing any damaged elements from the environment and replacing them. This can apply to corrupted data being restored from backup and to malfunctioning hardware or software being replaced with updated or new versions. In some cases, entire computer systems need to be reconstituted (rebuilt from new parts) in order to eradicate all elements of compromise and return into production a functioning and trustworthy system.

The recovery and reconstitution procedures can also include alterations of configuration settings and adding new security features or components. This is especially important if a vulnerability remains that could be exploited to cause the incident to reoccur. The environment is returned to normal operations by the end of the recovery phase.

Lessons learned

A final step in incident response is to evaluate the response plan and procedures and improve them as necessary. This review can also serve as a means to extract or clarify lessons learned during an incident response. Often things go wrong during a response, and learning from errors or mistakes will improve future responses.

Exam Essentials

Know about incident management. When an incident occurs, you must handle it in a manner that is outlined in your security policy and consistent with local laws and regulations. The first step in incident management or handling an incident properly is recognizing when one occurs.

Understand the idea of an incident response policy. An incident response policy is the procedure to follow when a security breach or security violation has occurred. One of the most important goals of an incident response policy is containment: the protection and preservation of evidence.

Comprehend incident response. The goal of a planned and documented incident response is to limit the amount of damage caused by an incident, to recover the environment as quickly as possible, and to gather information about the incident and the perpetrator in order to prevent a recurrence and pursue legal prosecution.

Understand preparation. Preparation is necessary to ensure a successful outcome of unplanned downtime, security breaches, or disasters. Being prepared includes defining a procedure to follow in response to incidents, buttressing an environment against incidents, and improving detection methods.

Know about incident identification. The first step in responding to an incident is to become aware that an incident is occurring. This should then lead to documenting all details about the incident.

Understand escalation and notification. Once you have a basic understanding of what the incident consists of, a staged procedure of escalation and notification can be followed. Only those in specific positions of authority or responsibility should receive notification of breaches.

Comprehend eradication. Eradication consists of the processes used to remove or eliminate the causes of the incident.

Understand recovery/reconstitution procedures. Recovery is the process of removing any damaged elements from the environment and replacing them, altering configuration settings, and adding new security features or components.

Know about first responders. When a security breach or perimeter violation is detected, incident response must be initiated by the first responders. The CIRT team guides the first responders as to what action to take.

Understand damage and loss control. In the process of responding to an incident, important goals are to contain the problem (the potential for further damage) and control or prevent loss. Containment means to limit the scope of damage and prevent other systems or resources from being negatively affected.

5.5 Summarize basic concepts of forensics.

Forensics is the collection, protection, and analysis of evidence from a crime in order to present the facts of the incident in court. One of the most critical aspects of forensics is the initial gathering and protection of evidence. In order for evidence to be admissible in court, you must be able to show that the chain of custody wasn't broken, that the evidence was properly preserved, and that the evidence was collected properly. One aspect of this is to perform analysis on copies of evidence and not on the original evidence when the evidence is a storage medium.

Evidence should be protected from alteration, damage, and corruption from the moment of its discovery through the rest of its lifetime, which may be concluded after it's presented in court. Evidence preservation includes properly managing the chain-of-custody document, collecting the evidence into transportable containers, clearly labeling those containers, and then providing a secure environment for the evidence. A secure environment prevents damage and theft, but it also maintains the proper temperature and humidity while avoiding dust, smoke, debris, magnetic fields, and vibrations.

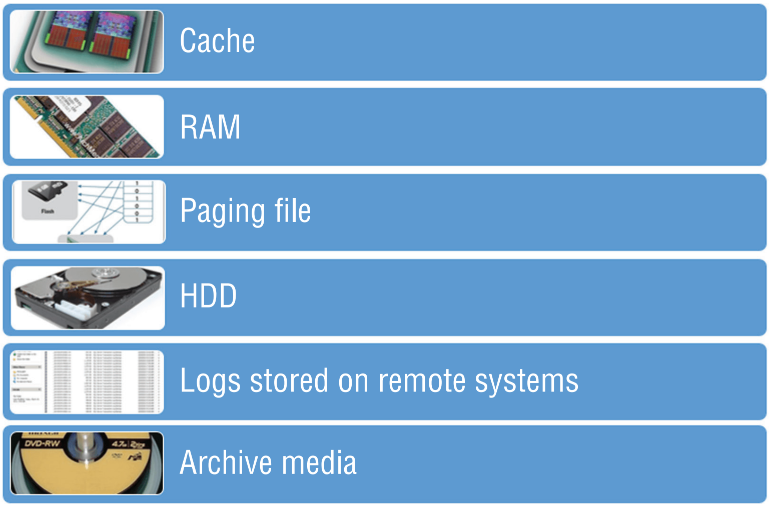
Collection of evidence is the procedure of securing evidence by collecting it. This process is often called *bag and tag*. Basically, evidence is gathered, placed in a container, and labeled, and then its chain-of-custody document is filled out. It's the responsibility of the crime scene technician to collect evidence.

Order of volatility

When collecting evidence, it is important to consider the volatility of data and resources. Volatility is the likelihood that data will be changed or lost due to the normal operations of a computer system and the passing of time. Collection of potential evidence should be prioritized based on the type of event, incident, or crime as well as the order of volatility. Generally, the order of volatility shown in Figure 5.9 is reliable to follow.

- Registers, cache
- Network connections
- Routing table, ARP cache, process table, kernel statistics, memory
- Temporary filesystems
- Disk
- Remote logging and monitoring data that is relevant to the system in question
- Physical configuration, network topology
- Archival media

FIGURE 5.9 Order of volatility



This volatility order was taken from RFC 3227: Guidelines for Evidence Collection and Archiving (`www.faqs.org/rfcs/rfc3227.html`). This is an excellent RFC to read for general knowledge about evidence collection.

Chain of custody

The *chain of custody* is a document that indicates various details about evidence across its life cycle. It begins with the time and place of discovery and identifies who discovered the evidence, who secured it, who collected it, who transported it, who protected it while in storage, and who analyzed it. Ultimately, the chain-of-custody document details all persons who had controlling authority over and access to the evidence. Any gaps in this record cast doubt on the integrity of the evidence, because there is a possibility that the evidence was out of authoritative control. The chain of custody must be created and maintained from the moment evidence is discovered through the presentation of the evidence in court.

Legal hold

A *legal hold* is an early step in the evidence collection or e-discovery process. It is a legal notice to a data custodian that that specific data or information must be preserved and that good-faith efforts must be engaged to preserve the indicated evidence. The custodian must maintain and preserve the data until they are notified that the obligation is no longer necessary. A data custodian may be an employer whose employee is under investigation, an online provider of a service or resource whose user is under investigation, or an ISP whose customer is under investigation.

Data acquisition

Data acquisition is the processes and procedures by which data relevant to a criminal action is discovered and collected. In most situations, evidence collection should be performed by licensed and trained forensics specialists, usually those associated with law enforcement. However, some familiarity with the concepts related to forensic evidence or data acquisition is necessary for the Security+ exam.

Big data refers to collections of data that have become so large that traditional means of analysis or processing are ineffective, inefficient, and insufficient. Big data involves numerous difficult challenges, including collection, storage, analysis, mining, transfer, distribution, and results presentation. Such large volumes of data have the potential to reveal nuances and idiosyncrasies that smaller sets of data fail to address. The potential to learn from big data is tremendous, but the burdens of dealing with it are equally great. As the volume of data increases, the complexity of data analysis increases as well. Big data analysis requires high-performance analytics running on massively parallel or distributed processing systems. In some situations, the sheer volume of data gathered related to a criminal activity means that forensic data acquisition and evidence analysis must take advantage of big data tools.

Capture system image

Because most computer crime evidence takes the form of bits on magnetic storage devices, it is fairly easy to manipulate and alter. Computers can be used to fabricate and counterfeit almost any form of record or data. In order to preserve data as well as establish and verify the integrity of that data, images are taken of suspect storage devices.

In most cases, a forensic imaging program is used that creates a bitstream image copy of a storage device. The image copy of the original media is stored on forensically clean storage devices, which have all existing data removed using a zeroization process.

The process of creating the image is performed with checks and balances. The forensic duplication system calculates a hash of the original media before and after the bitstream image copy is performed. If these hashes match, then the process of duplication did not alter the original during the duplication process. Additionally, the image copies are hashed. If the imaging process worked properly, each image copy's hash matches that of the original.

Network traffic and logs

When a computer crime or policy violation takes place, it is important to collect all possible sources of evidence. These can include network traffic captures as well as network device logs. In some network environments, it may be possible to maintain an ongoing recording of network traffic. However, because this would result in a massive need for storage capacity, such a recording only maintains a sliding window of recent network activity—often measured in minutes or, at most, hours. If a violation is detected promptly, the window of network traffic can be preserved for more detailed offline analysis.

Many network devices, including routers, switches, smart patch panels, firewalls, proxies, and VPN appliances, can be configured to record log files of the events, activities, or packets that occur on, over, or through them. These logs need to be collected and preserved in order to use them in an investigation.

Capture video

There are two issues related to video. First, if security cameras are present and video was captured of a security violation, those captured video images need to be preserved as evidence. Video (and audio) recordings may also track sensitive data that has been input, such as credit card numbers. In such circumstances, that data must be protected at the same level as or higher than the original data.

Second, while performing an investigation, especially while seeking out physical and/or logical evidence, it can be important to have someone videotape the process. The videotaped observation can assist in crime scene reenactments, orientation, and the proper explanation of evidence during a presentation in court.

In addition to videotaping the act of evidence gathering, it is a good idea to take copious photographs from multiple angles when moving or disassembling physical objects in association with an investigation.

Record time offset

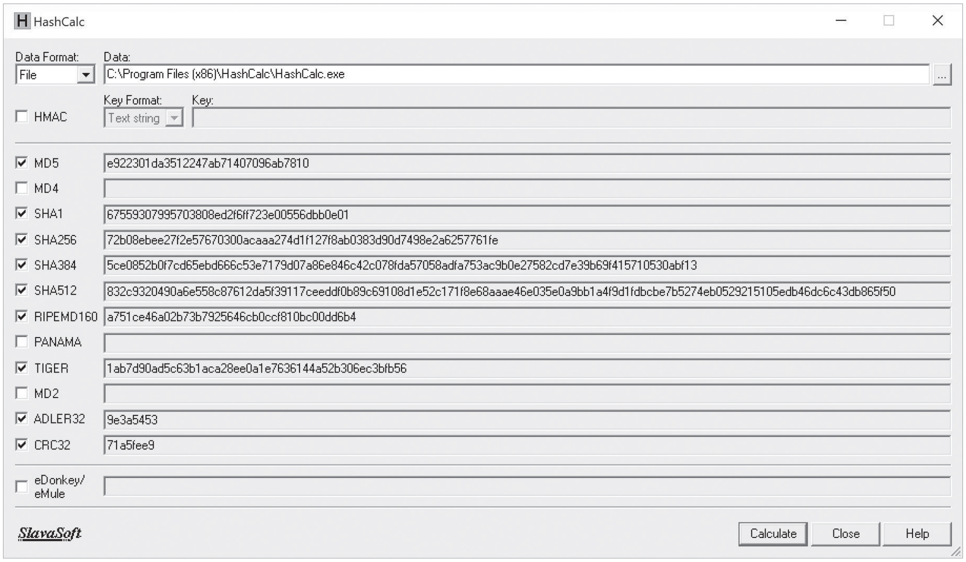
As an event is recorded into a log file, it is encoded with a time stamp. The time stamp is pulled from the clock on the local device where the log file is written or sent with the event from the originating device if remote logging is performed. However, it is all too common for the clocks of the devices and computers in a network to be out of time sync to some degree. Thus, it is important to establish a known time standard, such as one of the atomic clocks accessible through NIST (`http://tf.nist.gov/tf-cgi/servers.cgi`) in the United States; other nations also have nationalized time sources.

Then, each time a log file is pulled, the clock of the host device is checked and compared to the time standard. Recording the *time offset* is taking note of the difference between the device clock and the standard; it is used to adjust the time of log entries in order to sync events and activities across multiple network devices. Management of log times is essential for the chronological reconstruction of attack or compromise events.

Take hashes

As mentioned in the "Capture system image" section earlier, it is important to take a hash of a storage device, as shown in Figure 5.10, before and after image duplication. Additionally, it is important to periodically verify that the hash of the image copy being used for forensic investigation has not changed. Doing so ensures that the findings from the copy will legally apply to the source original.

FIGURE 5.10 A hashing tool, HashCalc



Data Format: File
Data: C:\Program Files (x86)\HashCalc\HashCalc.exe
□ HMAC Key Format: Text string Key:

☑ MD5 e922301da3512247ab71407096ab7810
□ MD4
☑ SHA1 67559307995703808ed2f6ff723e00556dbb0e01
☑ SHA256 72b08ebee27f2e57670300acaaa274d1f127f8ab0383d90d7498e2a6257761fe
☑ SHA384 5ce0852b0f7cd65ebd666c53e7179d07a86e846c42c078fda57058adfa753ac9b0e27582cd7e39b69f415710530abf13
☑ SHA512 832c9320490a6e558c87612da5f39117ceeddf0b89c69108d1e52c171f8e68aaae46e035e0a9bb1a4f9d1fdbcbe7b5274eb0529215105edb46dc6c43db865f50
☑ RIPEMD160 a751ce46a02b73b7925646cb0ccf810bc00dd6b4
□ PANAMA
☑ TIGER 1ab7d90ad5c63b1aca28ee0a1e7636144a52b306ec3bfb56
□ MD2
☑ ADLER32 9e3a5453
☑ CRC32 71a5fee9
□ eDonkey/eMule

Screenshots

When performing a forensic investigation, never trust the software on the suspect's computer. Thus, using native screen-capture tools or features is not recommended. Instead, use a camera to take photographs of anything being displayed. This includes monitors, smaller LCD screens (such as on printers), as well as any LEDs that might indicate status or function.

Witness interviews

A *witness* is someone who experienced an event or incident through one or more of their five senses. A witness can provide information about what occurred, where the occurrence took place, and the chronological order of related events. A witness is often called on during an investigation or during a court case to provide testimony of his or her experiences.

Preservation

Forensic preservation aims at preventing any change from occurring as related to collected evidence. These efforts include removing relevant storage devices from their systems, using write-blocking adapters to block any writing signals from being received by storage devices, using hash calculations before and after every operation, and analyzing only cloned copies of storage devices and never the original device. If the original data is corrupted or changed, then it usually becomes inadmissible in court. Thus, forensic experts take extreme caution when working with the original source drives. Once proper hash-validated clones are created, the original is sealed and secured, and then all analysis occurs on the clones.

Recovery

During forensic investigations there is often a need to recover or restore data in order to make it usable and to determine whether it is related to the criminal activity. One means of recovery is to restore files from backup, whether those backups are hard drives, tapes, optical discs, or the cloud. Another recovery mechanism is to gain access to the shadow copies of a file being maintained by the filesystem. For example, NTFS-formatted storage devices operating under Windows Server systems may be using the Volume Shadow Copy Service, which retains previous versions of files when new versions are written to the storage device.

Recovery might also include analyzing the data in slack space. Slack space is the unused storage space in the last cluster used by a file which is not fully used up by the file. Any unused portion of the last cluster will retain any previous data that was stored there. Slack space analysis tools can extract slack space stored data remnants that might have some evidentiary value.

Recovery might include using undelete tools that can restore data files back into a normal file after they have been deleted—or at least until the clusters containing the data are overwritten by new files being stored onto the device. Once some of a deleted file has been overwritten, undeleting cannot fully restore or recover the data.

Strategic intelligence/counterintelligence gathering

Strategic intelligence gathering consists of the investigative and interviewing skills that some law enforcement officers, military, and deputized civilians in certain cases use to discover information that may be relevant to a criminal activity. Evidence of a crime may not always be obvious and located where expected. It takes the skill, expertise, and experience of a seasoned investigator to approach each investigation with fresh eyes and a flexible methodology. Although many crimes may produce similar evidence, not all crimes fall in line with previous investigations. It is important for an investigator to consider the environment, technology, victim, opportunity, skill level, and other related items when considering how to look for and detect additional evidence during an investigation.

Strategic intelligence gathering can include more thorough interviews with direct and indirect witnesses of the criminal event. It can include evaluating off-SOP tasks performed by the suspect or victim. It can include considering physical logistics, chronologies, and human nature to discover new evidence that might not have been noticed during a rigid, abrupt, or cursory survey of affected systems and environments.

Counterintelligence, or anti-forensics, includes the actions that might be taken by a perpetrator in order to minimize relevant evidence or to misdirect an investigation. This can include scrubbing log files by surgically editing out the events related to the criminal activities or planting false records of events that did not occur or of entities that were innocent of committing crimes. System destruction can be used to remove evidence as well as standard personal or business data. Anti-forensics can also plant booby-trap code that lies dormant in files, slack space, or log files and is directed toward the standard investigative and analysis tools used by law enforcement. Such planted code might trigger storage device wipes, corrupt reading activities, overload/DoS scanning tools, generate random data, or attempt to destroy analysis systems. Investigators must take care to determine the skill level of the perpetrator and pay attention to symptoms that anti-forensics may have been used.

Active logging

Many organizations are realizing just how important forensic evidence collection is for understanding as well as responding to, detecting, and prosecuting a security breach. To that end, active logging is used to gather and maintain a wide range of security- and system-related events. Active logging may be a more thorough collection of events as well as a more robust means of preserving and retaining the audit trails for the purpose of forensic investigation rather than just overall system management and uptime optimization.

Track man-hours

Throughout the implementation of an incident response procedure or forensic investigation, you should document every action taken by end users and the incident-response/investigative teams. This documentation will serve as an audit trail to retrace the actions taken and the events that occurred during the incident. Learning from the incident's documentation includes taking precautions to prevent the recurrence of the incident, updating the security policy and related procedural documents, and assessing the overall impact of asset loss, damage, and risk imposed on the environment by the incident.

After an incident has been resolved, disclosure of the details and the results of the incident should be restricted to authorized parties. Often, the authorized parties are limited to senior management, the legal team, and some members of the security staff.

It is also important to review the man-hours involved in the response and mediation of an event. This helps determine whether the expense of the event was justified. Such information can be used to adjust budgets or response policies.

Exam Essentials

Know basic forensic procedures. Forensics is the collection, protection, and analysis of evidence from a crime in order to present the facts of the incident in court. One of the most critical aspects of forensics is the initial gathering and protection of evidence. In order for evidence to be admissible in court, you must be able to show that the chain of custody wasn't broken, that the evidence was properly preserved, and that the evidence was collected properly. This also includes issues such as the order of volatility, capturing a system image, collecting network traffic and logs, capturing video, recording time offsets, taking screenshots, interviewing witnesses, and tracking man-hours and expenses.

Understand order of volatility. When collecting evidence, it is important to consider the volatility of data and resources. Volatility is the likelihood that data will be changed or lost due to the normal operations of a computer system and the passing of time. Collection of potential evidence should be prioritized based on the type of event, incident, or crime as well as the order of volatility.

Comprehend the chain of custody. The chain of custody is a document that indicates various details about evidence across its life cycle. It begins with the time and place of discovery and identifies who discovered the evidence, who secured it, who collected it, who transported it, who protected it while in storage, and who analyzed it.

Understand legal hold. A legal hold is an early step in the evidence collection or e-discovery process. It is a legal notice to a data custodian that specific data or information must be preserved and that good-faith efforts must be engaged to preserve the indicated evidence.

Know about evidence preservation. Evidence should be protected from alteration, damage, and corruption from the moment of its discovery through the rest of its lifetime, which may be concluded after it's presented in court.

Understand the collection of evidence. Collection of evidence is the procedure of securing evidence by collecting it. This process is often called "bag and tag." Basically, evidence is gathered, placed in a container, and labeled, and its chain-of-custody document is filled out. It's the responsibility of the crime scene technician to collect evidence.

Comprehend big data analysis. Big data analysis requires high-performance analytics running on massively parallel or distributed processing systems.

5.6 Explain disaster recovery and continuity of operation concepts.

A *risk* is the possibility or likelihood that a threat will exploit a vulnerability, resulting in a loss such as harm to an asset. A *threat* is a potential occurrence that can be caused by anything or anyone and can result in an undesirable outcome. Natural occurrences such as floods and earthquakes, accidental acts by employees, and intentional attacks can all be threats to an organization. A *vulnerability* is any type of weakness. The weakness can be due to, for example, a flaw, a limitation, or the absence of a security control.

Risk management attempts to reduce or eliminate vulnerabilities or reduce the impact of potential threats by implementing controls or countermeasures. It is not possible, or desirable, to eliminate risk. Instead, an organization focuses on reducing the risks that can cause the most harm. Understanding risk management concepts is essential to the establishment of a sufficient security stance, proper security governance, and legal proof of due care and due diligence.

Managing risk is therefore an element of sustaining a secure environment. Risk management is a detailed process of identifying factors that could damage or disclose data, evaluating those factors in light of data value and countermeasure cost, and implementing cost-effective solutions for mitigating or reducing risk. The overall process of risk management is used to develop and implement information security strategies. The goals of these strategies are to reduce risk and to support the mission of the organization.

The primary goal of risk management is to reduce risk to an acceptable level. What that level is depends on the organization, the value of its assets, the size of its budget, and many other factors. What is deemed acceptable risk to one organization may be an unreasonably high level of risk to another. It is impossible to design and deploy a totally risk-free environment; however, significant risk reduction is possible, often with little effort.

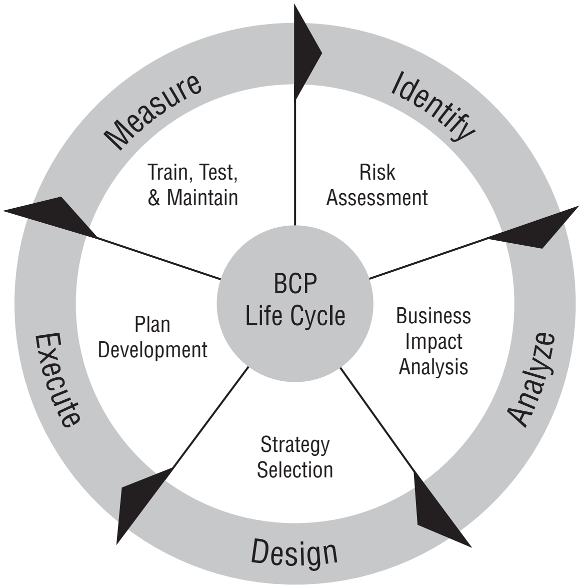
Risks to an IT infrastructure are not all computer-based. In fact, many risks come from noncomputer sources. Consider all possible risks when performing risk evaluation for an organization. Failing to properly evaluate and respond to all forms of risk leaves a company vulnerable. Keep in mind that IT security, commonly referred to as *logical* or *technical* security, can provide protection only against logical or technical attacks. To protect IT against physical attacks, physical protections must be erected.

The process by which the goals of risk management are achieved is known as *risk analysis*. It includes examining an environment for risks, evaluating each threat event as to its likelihood of occurring and the cost of the damage it would cause if it did occur, assessing the cost of various countermeasures for each risk, and creating a cost-benefit report for safeguards to present to upper management. In addition to these risk-focused activities, risk management requires evaluation, assessment, and the assignment of value for all assets within the organization. Without proper asset valuations, it is not possible to prioritize and compare risks with possible losses.

Business continuity planning (BCP) involves assessing a variety of risks to organizational processes and creating policies, plans, and procedures to minimize the impact those risks might have on the organization if they were to occur. BCP is used to maintain the continuous operation of a business in the event of an emergency situation. The goal of BCP planners is to implement a combination of policies, procedures, and processes such that a potentially disruptive event has as little impact on the business as possible. Figure 5.11 illustrates the BCP cycle.

BCP focuses on maintaining business operations with reduced or restricted infrastructure capabilities or resources. As long as the continuity of the organization's ability to perform its mission-critical work tasks is maintained, BCP can be used to manage and restore the environment. If the continuity is broken, then business processes have stopped, and the organization is in disaster mode; thus, disaster recovery planning (DRP) takes over.

FIGURE 5.11 BCP creation and maintenance life cycle



NOTE The top priority of BCP and DRP is always *people*. The primary concern is to get people out of harm's way; then you can address IT recovery and restoration issues.

You should understand the distinction between business continuity planning and disaster recovery planning. One easy way to remember the difference is that BCP comes first, and if the BCP efforts fail, DRP steps in to fill the gap.

Many industries are bound by federal, state, and local laws or regulations that require them to implement various degrees of BCP. We've already discussed one example in this chapter—the officers and directors of publicly traded firms have a fiduciary responsibility to exercise due diligence in the execution of their business continuity duties. In other circumstances, the requirements (and consequences of failure) may be more severe. Emergency services, such as police, fire, and emergency medical operations, have a responsibility to the community to continue operations in the event of a disaster. Indeed, their services become even more critical in an emergency when public safety is threatened. Failure on their part to implement a solid BCP could result in the loss of life and/or property and the decreased confidence of the population in their government.

A *disaster-recovery plan (DRP)* is an essential element of an overall security management plan. Disaster recovery is an expansion of BCP. Basically, when business continuity is interrupted, a disaster has occurred. Ultimately, both BCP and DRP rely on proper backup procedures.

A DRP is the collection of detailed procedures used in the event that business functions are interrupted by a significant damaging event. When the primary site is unable to support business functions, the disaster recovery plan is initiated. This plan outlines the procedures for getting the mission-critical functions of the business up and running at an alternate site while the primary site is restored to normal operations.

A disaster recovery plan is developed through critical process inventory and prioritization, a risk analysis and assessment process, and a detailed examination of dependencies of resources.

The overall disaster recovery plan should include plan maintenance and distribution of revisions. Over time, as the environment changes, the disaster recovery plan should be adjusted to comply with those changes. After the plan has been altered to a specified change level (meaning some specified amount of change), it must be redistributed throughout the organization. Only the most current version of the plan should be in existence; all older copies of the plan should be destroyed.

You should consider the implications of your facility's location. For example, what is the local crime rate? What is your proximity to highways? How close are emergency services? Is the area in a flood zone, subject to earthquakes, or liable to experience excessive rain or snow? Knowledge of these characteristics assists in the planning and design of the facility, as well as the selection of the location.

After a plan has been developed and implemented in an organization, it is important to regularly exercise or drill the plan. Just like a fire drill, drilling and exercising a disaster recovery plan helps train personnel on what to do in an emergency and reveals any oversights or omissions. Disaster recovery exercises are important maintenance elements that are essential to the long-term success of an organization.

BCP and DRP consist of the following elements:

Risk Analysis and Assessment This element includes itemizing the risks to each mission-critical aspect of the organization, and then performing qualitative and quantitative analyses of the risks to determine which risk is the most critical.

Business Impact Analysis You must determine how much any individually realized risk will negatively affect the business's continuity and also compute the maximum tolerable downtime.

Strategic Planning for Mitigation of Risks You need to determine what countermeasures, safeguards, or responses can be used to minimize the effect of risks.

Integration and Validation of the Plan This step includes putting the plan into practice in the daily work habits of users, integrating it into the security policy, and validating it through senior management approval and testing.

Training and Awareness The organization needs to properly train users on their responses and responsibilities in an emergency and maintain awareness between training periods.

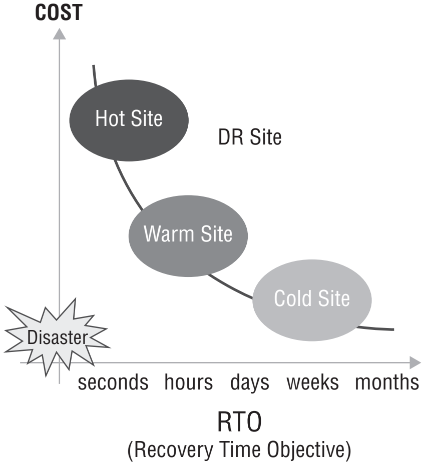
Maintenance and Auditing of the Plan You must regularly update the plan as the environment changes and constantly monitor the environment, the plan itself, testing, and training regarding the plan for areas where it can be improved.

Disaster recovery should encompass every aspect of an organization. A disaster recovery plan focuses on restoration of business processes. Additionally, DRP encourages the deployment of redundancy in order to prevent downtime. The following sections discuss several redundancy and DRP options.

Recovery sites

An organization-wide secure-recovery procedure involves the use of an alternate site—a secondary location where the business can move and continue performing mission-critical business operations. The recovery sites are known as alternate sites, alternate processing sites, backup locations, or secondary locations. There are three levels of alternate sites: hot, warm, and cold (Figure 5.12).

FIGURE 5.12 Alternate site options



Hot site

A *hot site* is a real-time, moment-to-moment mirror image of the original site. It contains a complete network environment that is fully installed and configured with live current business data. The moment the original site becomes inoperable due to a disaster, the hot site can be used to continue business operations without a moment of downtime.

Hot sites are the most expensive type, but they offer the least amount of downtime. Thus, while being a reliable means of recovery, they are not cost effective. Hot sites have significantly high security risk because live current business data is stored at both the primary site and the hot site, and there are real-time communications between them. Additionally, a hot site requires dedicated support staff to maintain it and keep it consistent with the primary site.

Warm site

A *warm site* is a partially configured alternate site with most of the server and networking infrastructure installed. In the event of a disaster, some final software installation and configuration are needed, and data must be restored from a backup set. A warm site may require hours or a day to get it ready for real-time operation to support the business's mission-critical functions. A warm site is moderately costly, but it is a realistic option for recovery if the organization can survive a few days of downtime.

When you return from the alternate site, whether hot, warm, or cold, the disaster could be repeated. The primary site is a new environment, because the original network and computer systems were damaged beyond their ability to support the business; significant changes, repairs, and replacements have occurred to restore the environment. The restored primary site should be stress-tested before the mission-critical operations of the business are transferred back to it. So, the least critical functions should be moved back to the primary site first. Then, after the site shows resiliency, you can move more critical functions as the network proves its ability to support the organization once again.

Cold site

A *cold site* is often little more than an empty room. It can be a location with no equipment or communications at all, or it can be a site with equipment in boxes and essential communications and utilities connected. In either case, it may require weeks of work to set up and configure in order to support the company's processing needs. A cold site is the least expensive option, but it does not offer a realistic hope of recovery.

Order of restoration

Order of restoration is the order in which a recovery effort should proceed. In most situations, when a disaster strikes, the most mission-critical business processes should be restored first. Then, other processes in descending criticality should be repaired.

However, use this order only when initially dealing with a disaster at the primary site, especially when moving operations over to a second, backup, or alternative processing site. When the disaster at the primary site has been addressed and the organization imitates the process of returning to the now repaired primary site, extreme caution should be shown-since the primary site is not the same as it was prior to the disaster. In fact, if the damage was significant, the primary site is actually now a tertiary site that just happens to be located in the same facility that was once the primary site. Thus, the return to the primary site should be slow and methodical and the order of restoration should be reversed from that followed during the immediate disaster response. The least critical processes should be restored to the repaired primary, followed by processes of ever-increasing levels of importance. This approach lets you evaluate the stability and reliability of the repaired primary prior to subjecting the mission-critical processes to an untested system.

Backup concepts

Backups are an essential part of business continuity because they provide insurance against damage or loss of data files. The mantra of all security professionals should be: backup, backup, backup. Backups are the only means of insurance available to your data resources in the event of a loss, disruption, corruption, intrusion, destruction, infection, or disaster.

Backups should be tested in order to prove reliable and usable. Testing a backup means restoring data from the backup media to verify that restoration can be done. If you don't test your restoration process, there is no guarantee that your backup was successful.

There are three primary forms of backup:

Full A *full backup* copies all files to the backup media regardless of the archive bit setting. It clears or resets the archive bit.

Incremental An *incremental backup* copies only those files with a set or flagged archive bit. It clears or resets the archive bit, thus selecting only those files that are new or that have changed.

Differential A *differential backup* copies only those files with a set or flagged archive bit. It doesn't alter the archive bit, thus selecting only those files that are new or that have changed.

Incremental and differential backups are performed in concert with full backups. For example, a full backup could be performed at the beginning of each week, and then daily incremental or differential backups could be performed the other six days. Daily incremental backups consume approximately the same amount of time and storage space each day, whereas differentials grow larger and take longer each day. When restoring, the full backup is used to restore the initial file set; then either all incremental backups are restored in chronological order or just the last differential is restored in order to regain access to the most current version of the files.

NOTE The *archive bit* is a file header flag indicating that a file either is new or has changed since the last backup (when set to 1), or is unchanged (when set to 0). The archive bit is a common feature on Windows filesystems. Other operating systems and filesystems may rely on time stamps instead of archive bits for backup file selection.

Backup media should be stored securely at an offsite location to prevent them from being damaged or destroyed by the same catastrophe that affects the business continuity of the primary site. They should be stored in a fire-protected safe, vault, or safety deposit box. Backup tapes should be moved offsite soon after a backup is complete, and the transportation of the backup tapes should be secured. The tapes should be protected at all times from physical damage, theft, alteration, and destruction.

Secure recovery and restoration ensure that mission-critical, sensitive, or secured servers can be restored after a disaster with minimal loss or security violations. Secure recovery ensures that affected systems reboot into a secured state, and that all resources open and active at the time of the fault, failure, or security violation are restored and have their security restrictions reimposed properly. Any damaged files are restored from backup, and their proper security labels are reapplied.

Another option for backups is to use the cloud as the storage medium. Many online storage services are available for individuals as well as enterprises to perform backups into the cloud. One example of a cloud backup solution is CrashPlan (`www.crashplan.com`). This is the cloud backup solution I use; it is designed for protecting data from a single machine, all the machines in a family, or the numerous systems of a small to medium-sized business. It performs automatic near-real-time backups of changed files very quickly after the changes are saved. There is also some retention of previous versions of files for recovery of an earlier edit of a document or a nonmodified version of an image file. However, using a cloud backup does require a high-speed data connection. Most local backup solutions can transfer data much faster than an online backup can. Amazon offers cloud backup solutions, which may include having a tractor-trailer full of physical storage (named the AWS Snowmobile) drive to a new customer location to make the initial copy of data that can handle up to 100 PB (petabytes, which is 1,000 terabytes). A local fiber link at 100 Gbps can copy 100 PB of data in just over 11 days, while it would take over 20 years to transfer over a 1 Gbps connection.

Differential

See the previous “Backup concepts” section for a description of the differential backup type.

Incremental

See the previous “Backup concepts” section for a description of the incremental backup type.

Snapshots

Snapshots are typically related to virtual machines (VMs) where the hypervisor is able to make a live copy of the active guest OS. Snapshots are complete copies of a VM that might take only a few minutes to create, compared to hours for cloning hard drives (which typically must be performed offline). After a snapshot is created, changes can be applied to the guest OS; if the changes are not satisfactory, the snapshot can be restored quickly to return the guest OS back to its previous saved state.

Full

See the previous “Backup concepts” section for a description of the full backup type.

Geographic considerations

When crafting business continuity and disaster recovery plans, you should assess the geographic considerations and their impact on preparation and responses.

Off-site backups

A backup strategy should include storage of backup media offsite to provide a reliable means of restoration and recovery in case of a significant event that may harm primary business operations. Onsite backups are useful for resolutions to minor issues, such as drive failures and accidental deletions. Only offsite backups are a reliable means of recovery due to major damage to the primary production environment.

Distance

Alternate processing facilities and offsite backup storage should be a reasonable distance away from the primary site. What is reasonable is subjective, but it depends on the value of the assets and the risk to an organization. Generally, alternate facilities should be far enough away that they will not be affected by the same disaster that harms the primary location—but not so far away that it is overly inconvenient for workers to travel to the alternate facility while the primary location is repaired. If it requires more than 3 hours to travel to the alternate location, lodging may have to be provided to workers so they don't have to commute 6 hours per day on top of working a typical 8-hour shift.

Backup media should also be stored far enough away from the primary location so it will not be harmed by the same disaster that damaged the primary location.

Location selection

Location selection should include consideration of accessibility, such as the number and size of roads leading to the facility, utility access and reliability, local crime rate, local hazards, and the likelihood of natural disasters (such as floods, fires, earthquakes, hurricanes, and tornadoes) affecting the area. Each of these concerns must be addressed in the design and construction of the facilities.

Legal implications

Some industries and individual organizations may be bound or limited by laws and regulations as to where their primary processing facilities can be located. This may be a limitation to stay within a country's borders or to stay away from populated areas. Some regulations include restrictions on the number of people allowed to operate within a specific building, construction requirements or supplements to local building codes, and proximity to airports, train stations, or seaports.

Data sovereignty

Data sovereignty is the concept that, once information has been converted into a binary form and stored as digital files, it is subject to the laws of the country within which the storage device resides. In light of the growing use of cloud computing, data sovereignty is an important consideration if there are regulations in your industry that require data to remain in your country of origin or if the country of storage has vastly different laws as compared to your country of origin. Data sovereignty can have an impact on privacy, confidentiality, and accessibility of your data.

Continuity of operation planning

IT contingency planning is a plan focused on the protection and/or recovery of an IT infrastructure. It is usually part of BCP or DRP, although separate plans for IT can be crafted. IT contingency planning focuses on providing alternate means to provide IT services in the event of a disaster. These plans can include backups as well as alternate, secondary, and backup processing locations.

Continuity of operation planning includes the creation of a security policy, BCP, DRP, and many other aspects of preparing for the worst and planning to avoid the consequences of downtime and data loss whenever possible.

Succession planning (Figure 5.13) is the process of identifying and preparing specific people, usually existing personnel, who will be called on to replace those in key leadership or critical role positions. The replacement may be planned due to a known retirement date, a scheduled company departure, or an unexpected event (such as prolonged sickness). For the long-term success of an organization using succession planning, focused training and development of the future replacements is essential. In some cases, succession planning is focused on replacing personnel with processes or services rather than filling a position with another human.

FIGURE 5.13 Succession planning

Exercises/tabletop

A *tabletop exercise* is a discussion meeting focused on a potential emergency event. It is usually performed verbally or with minimal visual aids (blueprints, charts, or board game miniatures representing resources). It is a means to walk through and evaluate an emergency plan in a stress-free environment. A tabletop exercise is also known as a *structured walkthrough*. A group can discuss the steps of an emergency response or recovery plan in order to clarify roles, assess responsibilities, detect deficiencies, address oversights, and conceive of alternative options.

After-action reports

The BRP, DRP, or CSIRT team should facilitate a *postmortem review*, or *after-action report (AAR)*, of the incident within a week of the occurrence to ensure that key players in the incident share their knowledge and develop best practices to assist in future incident response efforts. The AAR should detail the issues involved, the responses attempted, the successful resolution, and any oversights, mistakes, or lessons learned. These reports are useful in preparing for future incidents, defending the organization in court, and training future members of response teams.

Failover

The use of redundant servers is another example of avoiding single points of failure. A redundant server is a mirror or duplicate of a primary server that receives all data changes immediately after they are made on the primary server. In the event of a failure of the primary server, the secondary or redundant server can immediately take over and replace the primary server in providing services to the network. This takeover process is known as *failover*.

A failover system can be either hot or cold. A hot failover is an automatic system that can often perform the task nearly instantaneously. A cold failover requires an administrator to manually perform the task of switching from the primary to the secondary system, and thus it often involves noticeable downtime.

Redundant servers can be located in the same server vault as the primary or can be located offsite. Offsite positioning of the redundant server offers a greater amount of security so that the disaster that damaged the primary server is unlikely to be able to damage the secondary, offsite server. However, offsite redundant servers are more expensive due to the cost of housing, as well as real-time communication links needed to support the mirroring operations.

Alternate processing sites

See the earlier section "Recovery sites."

Alternate business practices

Alternate business practices are any secondary, backup, fail-back, or fallback plans that can be used in the event that the preferred recovery strategies and planning procedures fail. A *backup contingency plan* is an alternate solution or response in case the primary plan fails or is not as successful as planned.

A *backout contingency plan* is the plan to return to the primary site after moving to the alternate processing location. Since the primary site would have been severely damaged to necessitate the move to the secondary location, the primary would need to be significantly repaired in order to support the business operations. The repaired primary site is technically a new system and thus needs to be carefully stress-tested for resiliency before moving mission-critical processes.

Exam Essentials

Understand risk. A risk is the possibility or likelihood that a threat will exploit a vulnerability, resulting in a loss such as harm to an asset.

Know about business continuity planning and testing. Business continuity planning (BCP) involves assessing a variety of risks to organizational processes and creating policies, plans, and procedures to minimize the impact those risks might have on the organization if they were to occur. BCP is used to maintain the continuous operation of a business in the event of an emergency situation.

Understand disaster recovery. A disaster recovery plan is a collection of detailed procedures used in the event that business functions are interrupted by a significant damaging event. When the primary site is unable to support business functions, the disaster recovery plan is initiated. This plan outlines the procedures for getting the mission-critical functions of the business up and running at an alternate site while the primary site is restored to normal operations.

Comprehend alternate recovery sites. An alternate site is a secondary location where the business can move and continue performing mission-critical business operations. There are three levels of alternate sites: hot, warm, and cold.

Understand succession planning. Succession planning is the process of identifying and preparing specific people, usually existing personnel, who will be called on to replace those in key leadership positions.

Know about backups. Backups are the only means of insurance available to your data resources in the event of a loss, disruption, corruption, intrusion, destruction, infection, or disaster. Backups should be tested to ensure that they are reliable and usable.

Know the common types of backups. The three common types of backups are full, incremental, and differential.

Understand the importance of offsite storage. Backup media should be stored securely at an offsite location to prevent them from being damaged or destroyed by the same catastrophe that affects the business continuity of the primary site. This location should be a fire-protected safe, vault, or safety deposit box.

Define tabletop exercises. A tabletop exercise is a discussion meeting focused on a potential emergency event. It is a means to walk through and evaluate an emergency plan in a stress-free environment.

Understand backup/backout contingency plans or policies. A backup contingency plan provides an alternate solution or response if the primary plan fails or is not as successful as planned. A backout contingency plan prepares an organization to pull back from preparations, contracts, or agreements. Backout plans should include considerations that there may be legal or financial consequences to backing out of certain contracts or signed agreements.

5.7 Compare and contrast various types of controls.

A *control* is anything used to implement security. It can be an additional new product, a modification of an existing product, a redesign of the infrastructure, or the removal of something from the environment. Controls are necessary to protect the *confidentiality*, *integrity*, and *availability* of objects (and by extension, their information and data). Confidentiality addresses access control in the sense that it ensures that only authorized subjects can access objects. Integrity addresses the preservation of information in that unauthorized or unwanted changes to objects are denied (and checked). Availability addresses the ability to obtain access within a reasonable amount of time on request, in the sense that authorized requests for objects must be granted as quickly as system and network parameters allow.

The term *access control* refers to a broad range of controls that perform such tasks as ensuring that only authorized users can log on (thus an authentication focus) and preventing unauthorized users from gaining access to resources (thus an authorization focus). Controls mitigate a wide variety of information security risks.

Whenever possible, you want to prevent any type of security problem or incident. Of course, this isn't always possible, and unwanted events occur. When they do, you want to detect the events as soon as possible. And once you detect an event, you want to correct it.

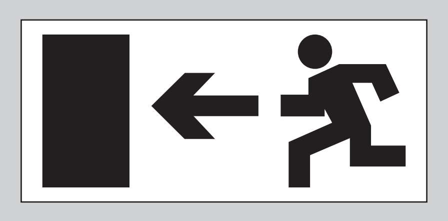
As you read the control descriptions, notice that some are listed as examples of more than one access control type. For example, a fence (or perimeter-defining device) placed around a building can be a preventive control (physically barring someone from gaining access to a building compound) and/or a deterrent control (discouraging someone from trying to gain access).

Deterrent

A *deterrent* access control is deployed to discourage violation of security policies. Deterrent and preventive controls are similar, but deterrent controls often depend on individuals deciding not to take an unwanted action. In contrast, a preventive control actually blocks the action. Some examples of deterrent access controls are policies, security-awareness training, locks, fences, security badges, guards, mantraps, and security cameras.

A *directive* access control is deployed to direct, confine, or control the actions of subjects to force or encourage compliance with security policies. Examples of directive access controls include security policy requirements or criteria, posted notifications, escape route exit signs (Figure 5.14), monitoring, supervision, and procedures.

FIGURE 5.14 Directive sign



Preventive

A *preventive* access control is deployed to thwart or stop unwanted or unauthorized activity from occurring. Examples of preventive access controls include fences, locks, biometrics, mantraps, lighting, alarm systems, separation of duties, job rotation, data classification, penetration testing, access control methods, encryption, auditing, presence of security cameras or CCTV, smartcards, callback procedures, security policies, security-awareness training, antivirus software, firewalls, and IPSs.

Detective

A *detective* access control is deployed to discover or detect unwanted or unauthorized activity. Detective controls operate after the fact and can discover the activity only after it has occurred. Examples of detective access controls include security guards, motion detectors, recording and reviewing of events captured by security cameras or CCTV, job rotation, mandatory vacations, audit trails, honeypots or honeynets, IDSs, violation reports, supervision and reviews of users, and incident investigations.

Corrective

A *corrective* access control modifies the environment to return systems to normal after an unwanted or unauthorized activity has occurred. It attempts to correct any problems that occurred as a result of a security incident. Corrective controls can be simple, such as terminating malicious activity or rebooting a system. They also include antivirus solutions that can remove or quarantine a virus, backup and restore plans to ensure that lost data can be restored, and active IDSs/IPSs that can modify the environment to stop an attack in progress. The access control is deployed to repair or restore resources, functions, and capabilities after a violation of security policies.

Recovery controls are an extension of corrective controls but have more advanced or complex abilities. Examples of recovery access controls include backups and restores, fault-tolerant drive systems, system imaging, server clustering, antivirus software, and database or virtual machine shadowing.

Compensating

A *compensation* access control is deployed to provide various options to other existing controls to aid in enforcement and support of security policies. It can be any control used in addition to, or in place of, another control. For example, an organizational policy may dictate that all PII must be encrypted. A review discovers that a preventive control is encrypting all PII data in databases, but PII transferred over the network is sent in clear text. A compensation control can be added to protect the data in transit. Examples of compensation controls include backups and alternate processing facilities.

Technical

Controls can be implemented administratively, logically/technically, or physically. Any of the access control types mentioned previously can include any of these types of implementation.

Technical or *logical* access involves the hardware or software mechanisms used to manage access and to provide protection for resources and systems. As the name implies, it uses technology. Examples of logical or technical access controls include authentication methods (such as usernames, passwords, smartcards, and biometrics), encryption, constrained interfaces, access control lists, protocols, firewalls, routers, IDSs, and clipping levels.

Administrative

Administrative access controls are the policies and procedures defined by an organization's security policy and other regulations or requirements. They are sometimes referred to as management controls. These controls focus on personnel and business practices. Examples of administrative access controls include policies, procedures, hiring practices, background checks, data classifications and labeling, security awareness and training efforts, vacation history, reports and reviews, work supervision, personnel controls, and testing.

Physical

Physical access controls are items you can physically touch. They include physical mechanisms deployed to prevent, monitor, or detect direct contact with systems or areas within a facility. Examples of physical access controls include guards, fences, motion detectors, locked doors, sealed windows, lights, cable protection, laptop locks, badges, swipe cards, guard dogs, video cameras, mantraps, and alarms.

Exam Essentials

Understand control types. The term access control refers to a broad range of controls that perform such tasks as ensuring that only authorized users can log on and preventing unauthorized users from gaining access to resources. Control types include preventive, detective, corrective, deterrent, recovery, directive, and compensation. Controls can also be categorized by how they are implemented: administrative, logical, or physical.

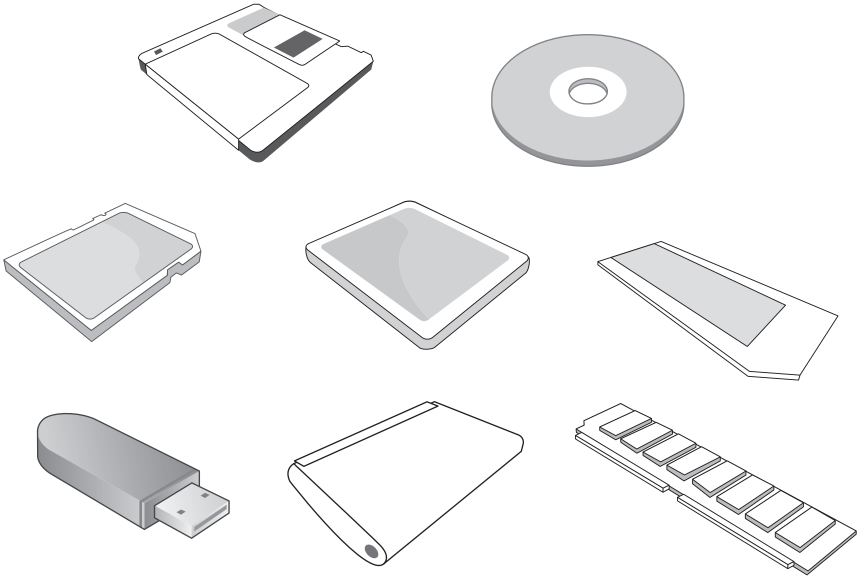
5.8 Given a scenario, carry out data security and privacy practices.

Every organization needs to be prepared to handle a variety of data security management activities that support or enforce general confidentiality protection and privacy. These security and privacy practices focus on properly disposing of data, labeling storage media, and retaining or archiving data. Scenarios where these management practices are relevant include implementing backups, retiring older systems, replacing damaged equipment, using portable drives, and providing a means of recovery in the event of a disaster.

Data destruction and media sanitization

Security policy should dictate how printed material and used storage media are to be handled after their useful lifetime. Secure disposal and destruction of printed material often involves shredding and incineration. Secure disposal and destruction of computers typically focuses on the secure disposal and destruction of storage media (Figure 5.15). After these items have been properly destroyed, they no longer pose a threat to the organization. If would-be intruders or attackers obtained these items after disposal and destruction, they would be unable to glean any useful information from them. In effect, proper disposal and destruction are countermeasures to dumpster diving and scavenging.

FIGURE 5.15 Types of storage media



If a medium is still useful to the organization but the data stored on the medium is to be removed, then media or data sanitization processes can be used. The reason for sanitization processes is that deletion and formatting are insufficient for removing data from storage devices. Deletion and formatting only remove the entries for files from the directory structure and mark the related storage clusters as available for use; these processes do not actually remove the data. Only when other data overwrites the older data is the older data no longer accessible or retrievable. Prior to overwriting, an undelete operation can restore the files. Any leftover data that can be recovered can be called data remnants. A proper sanitization process will remove any and all data remnants in order to prevent any and all forms of recovery.

Traditional magnetic media, such as spinning platter hard disk drives (HDDs), will overwrite areas in the outside or lower numbered tracks in preference over the inside or higher numbered tracks. Thus, deleted data on an inside track might remain available for undelete recovery much longer than a file on an outside track. Solid-state drives (SSDs) or flash media use a wear leveling process, which endeavors to use each storage cell evenly across the entire available storage capacity. This results in deleted data remaining available for undelete recovery until all of a drive's available free space has been used (at least once) before writing operations "wrap around" and eventually overwrite the deleted cells.

Once a storage device is of no further use to an organization, the only secure means of disposal is some form of physical destruction. These means can include incineration, an acid bath, and crushing. The remaining debris should then be handled by certified recycling services that will attempt to recover usable metals and properly dispose of any harmful or toxic materials.

There are generally recognized to be three levels of data remnant management: clearing, purging, and destruction. Clearing is any data destruction technique that only prevents data remnant recovery using any standard or common means, such as looking through the filesystem with a native file viewer or recovering data from a recycle bin or shadow copy service. Purging is any data destruction technique that attempts to prevent advanced or laboratory level data remnant recovery techniques from being able to restore access to data. Destruction is any data destruction technique in which the storage device itself is physically damaged beyond use.

Burning

Burning or incineration can be an effective means to destroy paperwork as well as media storage devices. Care should be taken to address any toxic fumes that may be produced when burning hardware, and the leftover materials may be recycled. When devices are incinerated it prohibits any data remnant recovery. Burning is considered a data destruction technique.

Shredding

Shredding can be an effective destruction technique for both paperwork and media storage devices; however, different equipment will be needed for these two techniques. Paperwork shredding should generally be accomplished using a cross-cut shredder that produces pieces no larger than 1/8 inch (4 mm) across. However, if you are operating in a highly secured facility, there may be specific government or industry regulations that require more severe levels of document destruction. There is some risk that shredded paperwork can be reassembled, so it may not be considered sufficient protection in some high-security situations. Paper shredding is considered a data clearing or purging technique.

Media device shredding is accomplished using an industrial metal shredder or chipper, which reduces devices down to metal fragments (shrapnel is metal fragments from a bomb, mine, or shell). When devices are shredded, it prohibits any data remnant recovery. Device shredding is considered a data destruction technique.

Pulping

Pulping is a paperwork destruction process that involves shredding paper and mixing it with a liquid to create a fibrous mush. The pulp can be formed into logs or bricks for starting fires in a fireplace or at a campsite or used to form new paper products, such as cardboard, hardboard, packaging, shipping containers, padding materials, or even recycled paper. Documents that have been pulped cannot be recovered. Pulping is considered a data destruction technique.

Pulverizing

Pulverizing is a means of device destruction that goes beyond the shredding level to a point where the devices are reduced to a grain or powder. When devices are pulverized, it prohibits any data remnant recovery. Pulverizing is considered a data destruction technique.

Degaussing

Degaussing is a means of media storage device data destruction using strong magnetic fields. It is effective only on magnetic media, such as hard drives and tapes; it is not effective on other forms of media, such as optical discs, SSDs, and flash memory cards. However, since the use of magnetic fields for data destruction leaves no visible trace of its effect, it might not always be sufficient to prevent data remnant recovery. The magnetic force might not have been applied correctly, it may not have been strong enough, or the activity may not have been applied long enough. Degaussing should not be considered a reliable means of data destruction. However, it may allow a device to be reused in the same security environment. Degaussing is considered a data clearing (when ineffective) or purging (when effective) technique.

Purging

Purging seems to be used as a reference to the level or quality of a data destruction process rather than an actual procedure that should be followed. See the earlier discussion at the end of the section "Data destruction and media sanitization."

Wiping

Data wiping is the process of removing data from a storage device. Often the intention is to prevent data remnants from being recovered that would lead to data leakage. Wiping may also be called *purging* or *sanitization*. Wiping procedures can include degaussing, random data overwriting, and zeroization (which is writing a zero to every location on the storage device). However, these techniques only provide sufficient data removal to use the storage device in the same security environment. There are no guaranteed wiping processes that allow for completely safe use of the device in less secure environments. Wiping may be considered a data purging or destruction technique.

Some HDD and SSH drives provide *on-device encryption*. These are sometimes known as "self-encrypting" drives. If the key for these drives is lost, replaced, removed, or "dropped," it prevents all access to the data stored on the drive. This can be an effective method to prevent access to data. Although the data is still present, it is in an encrypted form, and without the encryption key, regaining access to that data, whether authorized or not, will be a nearly impossible task.

Data sensitivity labeling and handling

Classification is the process of labeling objects (assets, data, information, and so on) with sensitivity labels and subjects (users) with clearance labels. After a resource is classified, the IT infrastructure and all users should read and respect the assigned label. Thus, each object receives the security it needs.

Data classification is the primary means by which data is protected based on its need for secrecy, sensitivity, or confidentiality. It is inefficient to treat all data the same when designing and implementing a security system, because some data items need more security than others. Securing everything at a low security level means sensitive data is easily accessible. Securing everything at a high security level is too expensive and restricts access to unclassified, noncritical data. Data classification is used to determine how much effort, money, and resources are allocated to protect the data and control access to it.

The primary objective of data classification schemes is to formalize and stratify the process of securing data based on assigned labels of importance and sensitivity. Data classification is used to provide security mechanisms for storing, processing, and transferring data. It also addresses how data is removed from a system and destroyed.

The following are benefits of using a data classification scheme:

- It demonstrates an organization's commitment to protecting valuable resources and assets.
- It assists in identifying those assets that are most critical or valuable to the organization.
- It lends credence to the selection of protection mechanisms.
- It is often required for regulatory compliance or legal restrictions.
- It helps to define access levels, types of authorized uses, and parameters for declassification and/or destruction of resources that are no longer valuable.

The criteria by which data is classified vary based on the organization performing the classification. However, you can glean numerous generalities from common or standardized classification systems:

- Usefulness of the data
- Timeliness of the data
- Value or cost of the data
- Maturity or age of the data
- Lifetime of the data (or when it expires)
- Association with personnel
- Data disclosure damage assessment (that is, how disclosure of the data would affect the organization)
- Data modification damage assessment (that is, how modification of the data would affect the organization)
- National security implications of the data
- Authorized access to the data (that is, who has access to the data)
- Restriction from the data (that is, who is restricted from the data)
- Maintenance and monitoring of the data (that is, who should maintain and monitor the data)
- Storage of the data

Using whatever criteria are appropriate for the organization, data is evaluated, and an appropriate data classification label is assigned to it. In some cases, the label is added to the data object. In other cases, labeling is simply assigned by the placement of the data into a storage mechanism or behind a security protection mechanism.

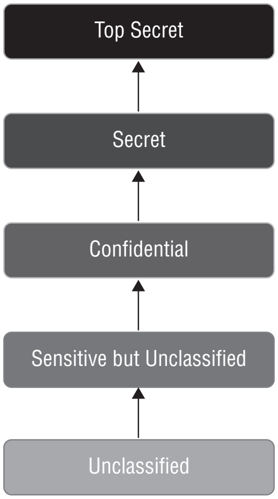
To implement a classification scheme, you must perform seven major steps, or phases:

1. Identify the custodian, and define his or her responsibilities.
2. Specify the evaluation criteria—how the information will be classified and labeled.
3. Classify and label each resource. (The owner conducts this step, but a supervisor should review it.)
4. Document any exceptions to the classification policy that are discovered, and integrate them into the evaluation criteria.
5. Select the security controls that will be applied to each classification level to provide the necessary level of protection.
6. Specify the procedures for declassifying resources and the procedures for transferring custody of a resource to an external entity.
7. Create an enterprise-wide awareness program to instruct all personnel about the classification system.

Declassification is often overlooked when designing a classification system and documenting usage procedures. Declassification is required once an asset no longer warrants or needs the protection of its currently assigned classification or sensitivity level. When assets fail to be declassified as needed, security resources are wasted, and the value and protection of the higher sensitivity levels is degraded.

The two common classification schemes are government/military classification and commercial business/private sector classification. As shown in Figure 5.16, there are five levels of government/military classification (listed here from highest to lowest).

FIGURE 5.16 Levels of classification



Top Secret The highest level of classification. The unauthorized disclosure of top-secret data will have drastic effects and cause grave damage to national security.

Secret Used for data of a restricted nature. The unauthorized disclosure of data classified as secret will have significant effects and cause critical damage to national security.

Confidential Used for data of a confidential nature. The unauthorized disclosure of data classified as confidential will have noticeable effects and cause serious damage to national security. This classification is used for all data labels or groupings between secret and sensitive but unclassified.

Sensitive but Unclassified Used for data that is of a sensitive or private nature, but the disclosure of which would not cause significant damage.

Unclassified The lowest level of classification. This is used for data that is neither sensitive nor classified. The disclosure of unclassified data would not compromise confidentiality or cause any noticeable damage.

NOTE

An easy way to remember the names of the five levels of the government or military classification scheme order from least secure to most secure is with a mnemonic device: U.S. Can Stop Terrorism. Notice that the five uppercase letters represent the five named classification levels, from least secure on the left to most secure on the right (or from bottom to top in the preceding list of items).

The classifications of confidential, secret, and top secret are collectively known or labeled as *classified*. Often, revealing the classification of data to unauthorized individuals is a violation of that data. Thus, the term “classified” is generally used to refer to any data that is ranked above the sensitive but unclassified level. All classified data is exempt from the Freedom of Information Act as well as many other laws and regulations.

The U.S. military classification scheme is most concerned with the sensitivity of data and focuses on the protection of confidentiality (that is, the prevention of disclosure). You can roughly define each level or label of classification by the level of damage that would be caused in the event of a confidentiality violation. Data from the top-secret level would cause grave damage to national security, whereas data from the unclassified level would not cause any serious damage to national or localized security.

Commercial business/private sector classification systems can vary widely, because they typically do not have to adhere to a standard or regulation. As an example, here are four possible business classification levels (listed from highest to lowest security):

Confidential The highest level of classification. This is used for data that is extremely sensitive and for internal use only. A significant negative impact could occur for a company if confidential data were disclosed. Sometimes the label *proprietary* is substituted for *confidential*.

NOTE

Another classification often used in the commercial business/private sector is *proprietary*. Proprietary data is a form of confidential information. If proprietary data is disclosed, it can have drastic effects on an organization’s competitive edge.

Private Used for data that is of a private or personal nature and intended for internal use only. A significant negative impact could occur for the company or individuals if private data is disclosed.

NOTE

Confidential and private data in a commercial business/private sector classification scheme both require roughly the same level of security protection. The real difference between the two labels is that confidential data is used for company data, whereas private data is used only for data related to individuals, such as medical data.

Sensitive Used for data that is more classified than public data. A negative impact could occur for the company if sensitive data is disclosed.

Public The lowest level of classification. This is used for all data that does not fit in one of the higher classifications. Its disclosure does not have a serious negative impact on the organization.

A *need-to-know* security policy grants and restricts access by compartmentalizing resources, objects, or data in a security domain. Compartmentalized resources can be located within a larger classification grouping. To gain access to compartmentalized items, the subjects (users) must obtain or prove the need to know—the necessity to have access to a resource based on assigned work tasks. Without need-to-know policies, such data is restricted from view even for users with sufficient security clearance. This form of access control is used in mandatory access control (MAC) environments, similar to the principle of least privilege that's used in discretionary access control (DAC) environments.

Although government/military terms and business classification levels are useful, generic references also can be used. The following are some common alternative classification terms:

High High, medium, and low can be used as generic references to classifications rather than using the government/military terms. High is comparable to top secret.

Medium This level is close to classified in government/military terms.

Low This level is close to unclassified in government/military terms.

Confidential Confidential relates to the most valuable and sensitive data level in a business classification scheme.

Private Private relates to individually related data (PII) in a business classification scheme. See the earlier section "Information classification."

Public Public relates to the least sensitive data level in a business classification scheme. See the earlier section "Information classification."

Confidential

See the previous discussion in the section "Data sensitivity labeling and handling."

Private

Private is a data classification label that can be used in a variety of contexts and for many purposes. Although most believe it to focus on the protection of information related to them as individuals, it can also refer to the collective data of employees and customers. Some organizations may even stretch the definition to include any internal-use-only information, which is more business confidential than actually private.

A privacy policy specifies the protections of privacy, or the lack thereof, within an organization. However, *privacy* can be a difficult entity to define. The term is used frequently in numerous contexts without much quantification or qualification. Here are some possible partial definitions of privacy:

- Active prevention of unauthorized access to information that is personally identifiable (that is, data points that can be linked directly to a person or an organization)
- Freedom from unauthorized access to information deemed personal or confidential
- Freedom from being observed, monitored, or examined without consent or knowledge

When addressing privacy in the realm of IT, it usually becomes a balancing act between individual rights and the rights or activities of an organization. Some claim that individuals have the right to control whether information can be collected about them and what can be done with it. This often brings up the issue of personally identifiable information (PII). PII is any data item that can be easily and/or obviously traced back to the person of origin or concern.

Others claim that any activity performed in public view, such as most activities performed over the Internet or activities performed on company equipment, can be monitored without knowledge of or permission from the individuals being watched and that the information gathered from such monitoring can be used for whatever purposes an organization deems appropriate or desirable.

On one hand, protecting individuals from unwanted observation, direct marketing, and disclosure of private, personal, or confidential details is considered a worthy effort. On the other, organizations profess that demographic studies, information gleaning, and focused marketing improve business models, reduce advertising waste, and save money for all parties.

Whatever your personal or organizational stance is on the issue of online privacy, it should be addressed in an organizational policy. Privacy is an issue not just for external visitors to your online offerings but also for your customers, employees, suppliers, and contractors. If you gather any type of information about any person or company, you must address privacy.

In most cases, especially when privacy is being violated or restricted, the individuals and companies must be informed; otherwise, you may face legal ramifications. Privacy issues must also be addressed when allowing or restricting personal use of email, retaining email, recording phone conversations, gathering information about surfing or spending habits, and so on.

Public

See the previous discussion in the section “Data sensitivity labeling and handling.”

Proprietary

See the previous discussion in the section “Data sensitivity labeling and handling.”

PII

Personally identifiable information (PII) is any data item that is linked back to the human from whom it was gleaned. PII that is medically related is protected under HIPAA laws. However, in the United States, most PII is not generally protected. Companies should clearly disclose what PII is collected and how it will be used in the acceptable use policy (AUP).

Privacy is the level of confidentiality and isolation a user is given in a system. Most users falsely assume that they have privacy on company computers. Privacy assumes that the activities and communications performed are hidden from others or at least protected from being viewed by all but the intended recipients. However, no activity on company property is hidden from the auditing and monitoring components of the network. As mentioned previously, whatever the stance of the company on privacy, this must be detailed and disclosed in a privacy policy.

PHI

Protected Health Information (PHI), according to the laws of the United States, is any data that relates to the health status, use of health care, payment for health care, and other information collected about an individual in relation to their health. The U.S. Health Insurance Portability and Accountability Act (HIPAA) defines PHI in relation to 18 types of information that must be handled securely to protect against disclosure and misuse. These 18 elements are:

- Names
- All geographic identifiers smaller than a state (so address, city, and zip are protected)
- Dates directly related to an individual, other than year
- Phone numbers
- Fax numbers
- Email addresses
- Social Security numbers
- Medical record numbers
- Health insurance beneficiary numbers
- Account numbers
- Certificate/license numbers
- Vehicle identifiers and serial numbers
- Device identifiers and serial numbers
- Web URLs
- IP address numbers
- Biometric identifiers
- Photographic images
- Any other unique identifying number, characteristic, or code except the unique code assigned by the collecting entity to code the data

Such data is often collected for statistical analysis and community health evaluation, but in these situations the data must be anonymized or de-identified. Anonymization is performed by removing or changing PHI items to prohibit linking the data back to the original source or individual. De-identification is performed by specifically removing the 18 HIPAA elements from the PHI. HIPAA and other legislation define strict parameters within which PHI can be collected, stored, and used.

Data roles

Data roles are discussed in the earlier section “Role-based awareness training.”

Owner

See the earlier discussion in the section "Role-based awareness training" and the sidebar "User, Owner, Custodian."

Steward/custodian

See the earlier discussion in the section "Role-based awareness training" and the sidebar "User, Owner, Custodian."

Privacy officer

A privacy officer is a company executive tasked with the responsibilities of crafting the company privacy policy, implementing that policy, and overseeing its operation and management. The goal of the privacy officer is to ensure that personal data related to employees and customers is properly handled and protected.

Data retention

A *retention policy* defines what data is to be maintained and for what period of time. A retention policy defines the parameters and operations of *data retention*. Retention policies may also need to define the purpose of the held data, the security means implemented to protect the held data, and the officers of the organization who are authorized to access or handle the held data. Various industry regulations as well as contractual obligations may mandate minimum retention time frames for certain types of data.

A *storage policy* defines the means, mechanisms, and locations for long-term housing of storage devices. No current storage device technology lasts forever, so you must make plans to provide a storage facility that can maintain the best environment (in terms of heat, light, humidity, vibration, and so on) and reliable security. A procedure for transferring data from aging storage devices to new devices is also essential if data is to be retained for longer than the predicted lifetime of the storage device.

Legal and compliance

Every organization needs to verify that its operations and policies are legal and in compliance with their stated security policies, industry obligations, and regulations. Auditing is necessary for *compliance testing*, also called *compliance checking*. Verification that a system complies with laws, regulations, baselines, guidelines, standards, best practices, and policies is an important part of maintaining security in any environment. Compliance testing ensures that all necessary and required elements of a security solution are properly deployed and functioning as expected. Compliance checks can take many forms, such as vulnerability scans and penetration testing. They can also use log analysis tools to determine whether any vulnerabilities for which countermeasures have been deployed have been attempted or exploited on the system.

Exam Essentials

Know about data labeling, handling, and disposal. Labeling is part of a classification system used to guide security, specifically in the areas of access, handling, and disposal.

Understand the concepts of data destruction and media sanitization. Three levels of data remnant management are generally recognized: clearing, purging, and destruction. You should consider numerous options for media sanitization, including burning, shredding, pulping, pulverizing, degaussing, and wiping.

Define information classification. Classification is the process of labeling objects (assets, data, information, and so on) with sensitivity labels and subjects (users) with clearance labels. After a resource is classified, the IT infrastructure and all users should read and respect the assigned label. Thus, each object receives the security it needs.

Understand the goal of a privacy policy. A privacy policy has a goal of protecting the confidentiality of personally identifiable information (PII).

Comprehend the need to protect PII. Personally identifiable information (PII) is any data item that is linked back to the human from whom it was gleaned. PII that is medically related is protected under HIPAA laws. However, in the United States, most PII is not generally protected. Companies should clearly disclose what PII is collected and how it will be used in the acceptable use policy (AUP).

Understand compliance with laws, best practices, and standards. Auditing is commonly used for compliance testing, also called compliance checking. Verifying that a system complies with laws, regulations, baselines, guidelines, standards, best practices, and policies is an important part of maintaining security in any environment. Compliance testing ensures that all necessary and required elements of a security solution are properly deployed and functioning as expected.

Review Questions

You can find the answers in the Appendix.

1. Which of the following risk assessment formulas represents the total potential loss a company may experience within a single year due to a specific risk to an asset?
   - **A.** EF
   - **B.** SLE
   - **C.** ARO
   - **D.** ALE

2. Which of the following is more formal than a handshake agreement but not a legal binding contract?
   - **A.** SLA
   - **B.** BIA
   - **C.** DLP
   - **D.** MOU

3. When a user signs a(n) __________, it's a form of consent to the monitoring and auditing processes used by the organization.
   - **A.** Acceptable use policy
   - **B.** Privacy policy
   - **C.** Separation of duties policy
   - **D.** Code of ethics policy

4. When is business continuity needed?
   - **A.** When new software is distributed
   - **B.** When business processes are interrupted
   - **C.** When a user steals company data
   - **D.** When business processes are threatened

5. You run a full backup every Monday. You also run a differential backup every other day of the week. You experience a drive failure on Friday. Which of the following restoration procedures should you use to restore data to the replacement drive?
   - **A.** Restore the full backup and then each differential backup.
   - **B.** Restore the full backup and then the last differential backup.
   - **C.** Restore the differential backup.
   - **D.** Restore the full backup.

6. Which of the following is a security control type that is not usually associated with or assigned to a security guard?

   A. Preventive

   B. Detective

   C. Corrective

   D. Administrative

7. You are the security manager for a brokerage firm. New company policy requires that all administrators be evaluated for compliance or violations in regard to adherence to the security policy and ethics agreement. Which of the following is a technique that can be used to accomplish this task?

   A. Separation of duties

   B. Clean desk

   C. Background checks

   D. Mandatory vacations

8. Separation of duties has recently been implemented at your organization. Due to the size of the company, a single person has been assigned to each compartmented management area. There is some concern that over time the company will be at risk of being unable to perform critical tasks if one or more administrators are unavailable due to illness, vacation, retirement, or termination. What tool can be used to reduce this risk?

   A. Job rotation

   B. Principle of least privilege

   C. Exit interviews

   D. Awareness training

9. Downtime is a violation of availability. Avoiding downtime is an essential tenet of your organization's mission and security policy. What element of system management and maintenance needs to be monitored and tracked in order to avoid device failure resulting in unplanned downtime?

   A. RTO

   B. MTTF

   C. ALE

   D. NDA

10. You are the network manager for a large organization. Over the weekend a storm caused a power surge, which damaged the main router between the company network and the Internet service. On Monday morning you realize that the entire intranet is unable to connect to any outside resource and mission-critical tasks are not functioning. What is the problem that the organization is experiencing?

    A. Sustained redundancy

    B. Maintaining of availability

    C. Load-balanced distribution of job tasks

    D. A single point of failure

11. What form of risk analysis can involve the Delphi technique, interviews, and focus groups?

    A. Quantitative

    B. Residual

    C. Qualitative

    D. Cost-benefit

12. You are the security manager for a large organization. During the yearly risk management reassessment, a specific risk is being left as is. You thoroughly document the information regarding the risk, the related assets, and the potential consequences. What is this method of addressing risk known as?

    A. Mitigation

    B. Tolerance

    C. Assignment

    D. Ignoring

13. What type of security policy or plan has the following main phases: Preparation, Identification, Containment, Eradication, Recovery, and Lessons Learned?

    A. IRP

    B. BCP

    C. DRP

    D. BPA

14. In what phase of an incident response plan does the organization return to normal operations after handling a violating event?

    A. Containment

    B. Lessons Learned

    C. Recovery

    D. Eradication

15. When an organization is sent a lawyer's letter demanding that they retain specific records, logs, and other files pertaining to suspected illegal activity, what is this known as?

    A. Audit

    B. Forensics

    C. Investigation

    D. Legal hold

16. Which of the following are important elements in gathering data from storage devices related to a suspect's system during a forensic investigation? (Select all that apply.)

    A. Calculating a hash of the original storage device

    B. Creating bitstream copy clones of the original

    C. Using read-block adapters

    D. Removing the storage device from the suspect's system

17. What is the main goal of BCP?

A. Recover from disasters

B. Minimize the impact of a disruptive event

C. Keep costs to a minimum

D. Prevent intrusions

18. What form of alternate processing facility is a reliable means of recovery but is not usually considered to be cost effective?

A. Warm

B. Cold

C. Onsite

D. Hot

19. A corrective control is used for what purpose?

A. To thwart or stop unwanted or unauthorized activity from occurring

B. To discover or detect unwanted or unauthorized activity

C. To modify the environment to return systems to normal after an unwanted or unauthorized activity has occurred

D. To provide various options to other existing controls to aid in enforcement and support of security policies

20. Which of the following may be considered protected health information? (Select all that apply.)

A. Phone numbers

B. Medical record numbers

C. Email address

D. Vehicle identifiers

E. Web URLs

F. IP address numbers

G. Biometric identifiers

H. Photographic images

Chapter

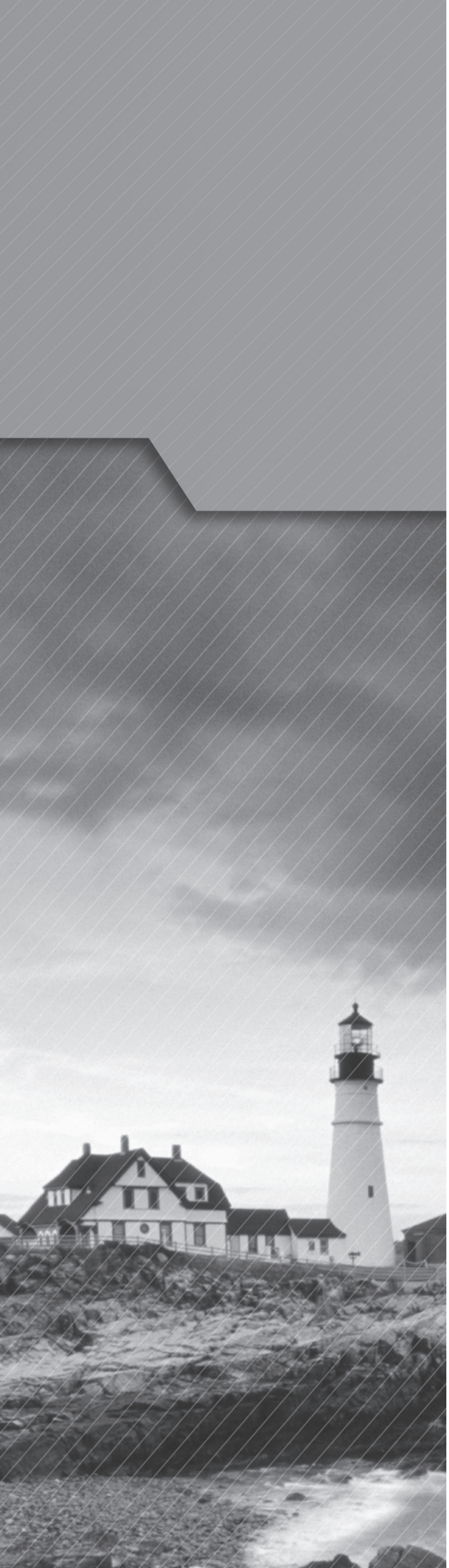
6

Cryptography and PKI

COMPTIA SECURITY+ EXAM OBJECTIVES COVERED IN THIS CHAPTER INCLUDE THE FOLLOWING:

✓ **6.1 Compare and contrast basic concepts of cryptography.**

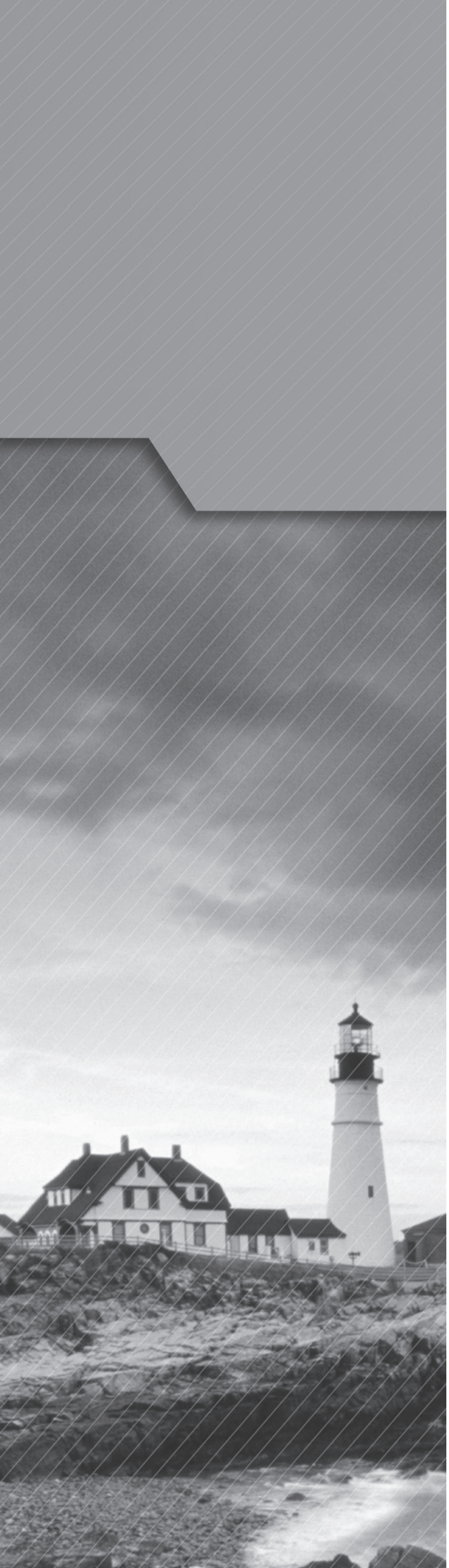
- Symmetric algorithms
- Modes of operation
- Asymmetric algorithms
- Hashing
- Salt, IV, nonce
- Elliptic curve
- Weak/deprecated algorithms
- Key exchange
- Digital signatures
- Diffusion
- Confusion
- Collision
- Steganography
- Obfuscation
- Stream vs. block
- Key strength
- Session keys
- Ephemeral key
- Secret algorithm
- Data-in-transit
- Data-at-rest
- Data-in-use



- Random/pseudo-random number generation
- Key stretching
- Implementation vs. algorithm selection
  - Crypto service provider
  - Crypto modules
- Perfect forward secrecy
- Security through obscurity
- Common use cases
  - Low power devices
  - Low latency
  - High resiliency
  - Supporting confidentiality
  - Supporting integrity
  - Supporting obfuscation
  - Supporting authentication
  - Supporting non-repudiation
  - Resource vs. security constraints

✓ **6.2 Explain cryptography algorithms and their basic characteristics.**

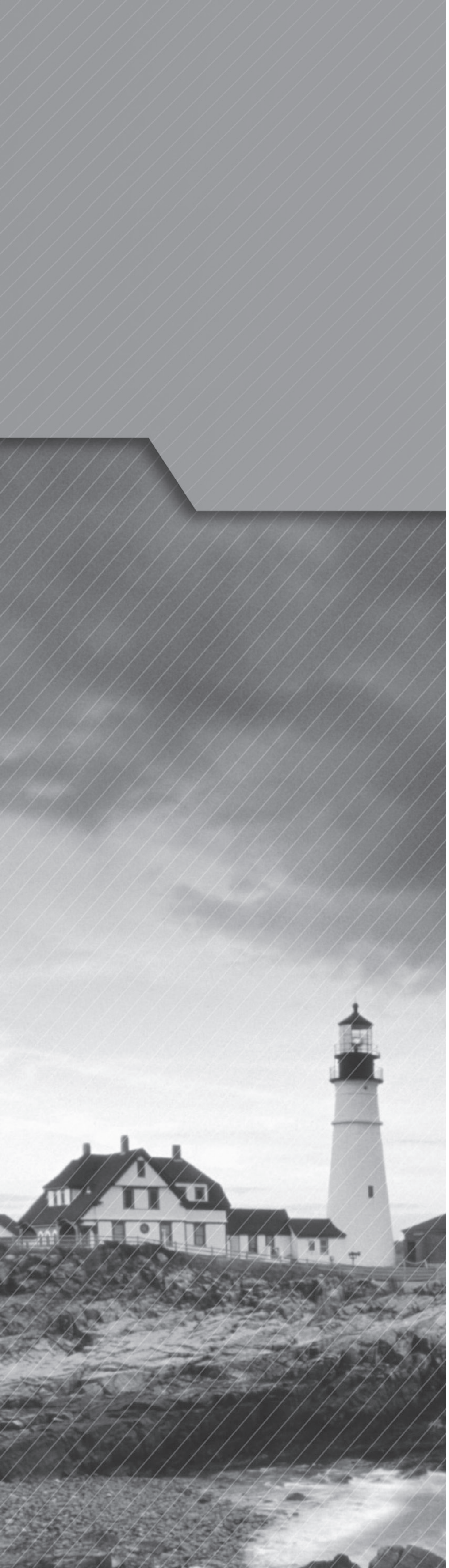
- Symmetric algorithms
  - AES
  - DES
  - 3DES
  - RC4
  - Blowfish/Twofish
- Cipher modes
  - CBC
  - GCM
  - ECB
  - CTM
  - Stream vs. block



- Asymmetric algorithms
  - RSA
  - DSA
  - Diffie-Hellman
    - Groups
    - DHE
    - ECDHE
  - Elliptic curve
  - PGP/GPG
- Hashing algorithms
  - MD5
  - SHA
  - HMAC
  - RIPEMD
- Key stretching algorithms
  - BCRYPT
  - PBKDF2
- Obfuscation
  - XOR
  - ROT13
- Substitution ciphers

✓ 6.3 Given a scenario, install and configure wireless security settings.

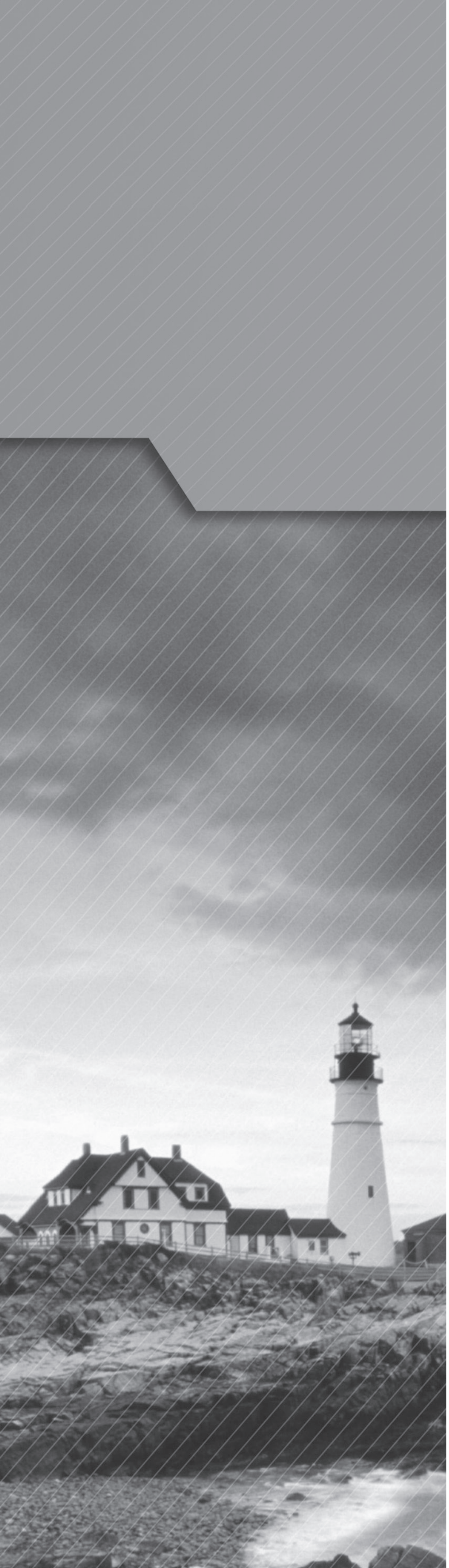
- Cryptographic protocols
  - WPA
  - WPA2
  - CCMP
  - TKIP
- Authentication protocols
  - EAP
  - PEAP



    - EAP-FAST
    - EAP-TLS
    - EAP-TTLS
    - IEEE 802.1x
    - RADIUS Federation
  - Methods
    - PSK vs. Enterprise vs. Open
    - WPS
    - Captive portals

✓ **6.4 Given a scenario, implement public key infrastructure.**

- Components
  - CA
  - Intermediate CA
  - CRL
  - OCSP
  - CSR
  - Certificate
  - Public key
  - Private key
  - Object identifiers (OID)
- Concepts
  - Online vs. offline CA
  - Stapling
  - Pinning
  - Trust model
  - Key escrow
  - Certificate chaining
- Types of certificates
  - Wildcard
  - SAN



  - Code signing
  - Self-signed
  - Machine/computer
  - Email
  - User
  - Root
  - Domain validation
  - Extended validation
- Certificate formats
  - DER
  - PEM
  - PFX
  - CER
  - P12
  - P7B

The Security+ exam will test your knowledge of cryptography and how it relates to the security of stand-alone and networked systems in a corporate environment. To pass the test and be effective in implementing security, you need to be familiar with both symmetric and asymmetric cryptography, as well as hashing, certificates, digital signatures, and other cryptographic issues you'll find detailed in this chapter.

6.1 Compare and contrast basic concepts of cryptography.

There is a wide range of topics related to cryptography. Some of these are foundational issues, some are security services, and others are solutions or implementations. This section discusses many important general cryptography concepts that are addressed on the Security+ exam.

Security practitioners utilize cryptographic systems to meet several fundamental goals, including confidentiality, integrity, and authentication. Achieving each of these goals requires the satisfaction of a number of design requirements, and not all cryptosystems are intended to achieve all possible goals.

Confidentiality ensures that data remains private while at rest, such as when stored on a disk, or in motion, such as during transmission between two or more parties. This is perhaps the most widely cited goal of cryptosystems—the facilitation of secret communications between individuals and groups. Two main types of cryptosystems enforce confidentiality. Symmetric key cryptosystems use a shared secret key available to all users of the cryptosystem. Asymmetric cryptosystems use individual combinations of public and private keys for each user of the system.

When developing a cryptographic system for the purpose of providing confidentiality, you must think about two different types of data: data at rest and data in motion. Data at rest, or stored data, resides in a permanent location awaiting access. Examples of data at rest include data stored on hard drives, backup tapes, USB devices, and other storage media. Data in motion, or data "on the wire," is being transmitted across a network between two systems. Data in motion might be traveling on a corporate network, a wireless network, or the public Internet. Both data in motion and data at rest pose different types of confidentiality risks that cryptography can protect against. For example, data in motion may be susceptible to eavesdropping attacks, whereas data at rest is more susceptible to the theft of physical devices.

Integrity ensures that a message isn't altered while in transit. If integrity mechanisms are in place, the recipient of a message can be certain that the message received is identical to the message that was sent. This protects against all forms of alteration: intentional alteration by a third party attempting to insert false information and unintentional alteration by faults in the transmission process. Message integrity is enforced through the use of digitally signed message digests created upon transmission of a message. The recipient of the message simply verifies that the message's digest and signature are valid, ensuring that the message wasn't altered in transit. Integrity can be enforced by both public and secret key cryptosystems.

Authentication verifies the claimed identity of system users and is a major function of cryptosystems. For example, suppose that Jim wants to establish a communications session with Bob, and they're both participants in a shared-secret communications system. Jim might use a challenge-response authentication technique to ensure that Bob is who he claims to be.

Another important benefit or goal of cryptography is nonrepudiation. This is the idea that a sender can't deny having sent a signed message. This is discussed in its own section later in this chapter.

As with any science, you must be familiar with certain terminology before you study cryptography. Let's look at a few of the key terms used to describe codes and ciphers. Before a message is put into a coded form, it's known as a *plain text* message and is represented by the letter *P* when encryption functions are described. The sender of a message uses a cryptographic algorithm to *encrypt* the plain text message and produce a *cipher text* message, represented by the letter *C*. This message is transmitted by some physical or electronic means to the recipient. The recipient then uses a predetermined algorithm to decrypt the cipher text message and retrieve the plain text version.

All cryptographic algorithms rely on keys to maintain their security. For the most part, a key is nothing more than a number. It's usually a very large binary number, but a number nonetheless. Every algorithm has a specific *keyspace*. The keyspace is the range of values that are valid for use as a key for a specific algorithm. A keyspace is defined by its bit size. Bit size is nothing more than the number of binary bits (0s and 1s) in the key. The keyspace is the range between the key that has all 0s and the key that has all 1s. Or to state it another way, the keyspace is the range of numbers from 0 to 2n, where *n* is the bit size of the key. So, a 128-bit key can have a value from 0 to 2128 (which is roughly 3.40282367 * 1038—that is, a very big number!). Even though a key is just a number, it's a very important number. In fact, if the algorithm is known, then all the security you gain from cryptography rests on your ability to keep the keys used private.

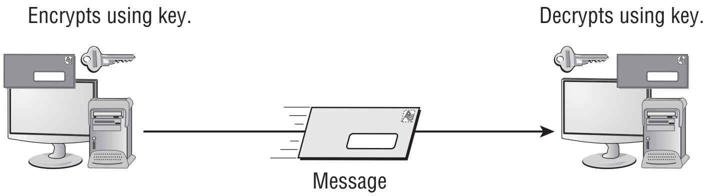
Different types of algorithms require different types of keys. In private key (or secret key) cryptosystems, all participants use a single shared key. In public key cryptosystems, each participant has his or her own pair of keys. Cryptographic keys are sometimes referred to as *cryptovariables*.

Symmetric algorithms

Symmetric cryptography is also called *private key cryptography* or *secret key cryptography*. It uses a single shared encryption key to encrypt and decrypt data (see Figure 6.1).

Keep in mind the word “same” when thinking about symmetric encryption, because the same key is used for both encryption and decryption. When symmetric cryptography is used to encrypt files on a hard drive, the user is the only person in possession of the single secret key. When symmetric cryptography is used to encrypt communications traffic, the two communication partners each have a copy of the one shared secret key. For example, the secure communication session protocol Secure Sockets Layer (SSL) uses symmetric cryptography. In either use, symmetric cryptography protects confidentiality.

FIGURE 6.1 A symmetric encryption system



Symmetric cryptography is very fast in comparison to asymmetric cryptography (discussed next), thanks to the way its algorithms are designed and the fact that a single key is used to encrypt and decrypt data.

Symmetric cryptography provides strong encryption protection when larger keys are used. However, the protection is secure only as long as the keys are kept private. If a symmetric key is compromised or stolen, it no longer offers true protection (just as your door lock no longer provides security if someone gets a copy of your house key).

Key exchange or distribution under symmetric cryptography is a common problem. To use symmetric cryptography to encrypt communications traffic between you and someone else over the Internet (or some other untrusted network), you must have a means to exchange the secret keys securely. If you already have a secure means of exchange, why aren’t you using that mechanism to communicate? Thus, some out-of-band communication solution must be implemented to securely exchange keys. Mechanisms of the past included shipping a floppy with a key, reading it over the phone, or using a different network to transmit the key. However, the preferred method now is to deploy a complete *Public Key Infrastructure (PKI)* solution that employs asymmetric cryptography to exchange symmetric cryptographic keys. The exchanged secret keys are used to encrypt the traffic for a single communication session, and then they’re discarded. PKI is simply a concept of how to deploy different aspects of various cryptography mechanisms into a single, complete, real-world solution.

Because each member of a network in a symmetric cryptography solution needs to have a secret key shared with every other member in order to support secure communications, $n\ (n - 1) / 2$ keys are needed. Thus, symmetric cryptography isn’t scalable when used alone.

The symmetric cryptography algorithms related to the Security+ exam are listed in Table 6.1.

TABLE 6.1 Common symmetric cryptography solutions

Name	Block size	Key size (in bits)
Advanced Encryption Standard (AES; uses the Rijndael block cipher algorithm)	128	128, 192, and 256
Triple Data Encryption Standard (3DES)	64	168
Data Encryption Standard (DES)	64	56
International Data Encryption Algorithm (IDEA)	64	128
Blowfish	64	32 to 448
Twofish	128	128, 192, or 256
Rivest Cipher 5 (RC5)	32, 64, 128	0–2040
Rivest Cipher 6 (RC6)	128	128, 192, or 256
Carlisle Adams/Stafford Tavares (CAST-128)	64	40 to 128 in increments of 8

Modes of operation

Most symmetric algorithms are block ciphers or offer a block cipher function. As detailed in the later section "Stream vs. block," a block cipher divides a message or data set into fixed-length sections or blocks before performing the encryption. There are many methods by which the plain text blocks are encrypted into cipher text blocks. These methods are known as *modes of operation*. There are several modes of operation, including ECB, CBC, CTM, and GCM.

ECB (Electronic Codebook) mode is the simplest and least secure of the symmetric modes. In this mode, each block of the message is encrypted in a straightforward process using the selected secret key. If two blocks in a message happen to contain the same data, then this mode would result in both blocks having matching cipher text as well. This mode is to be avoided for general message or data protection, but it is still used for very small data sets or when exchanging initialization values (IVs, effectively random numbers) or other values within other modes.

CBC (Cipher Block Chaining) mode is used to prevent the creation of duplicate cipher text blocks. This is accomplished by adding an IV into the operation of encryption. The IV is integrated with the first block using XOR. The result is encrypted using the selected secret key. The cipher text of the first block is then used as the IV for the second block. This linking or chaining of the blocks for use as an IV ensures that every block results in cipher text that is unique. The IV must be sent to the recipient, so it is encrypted using ECB and added as the first block of the cipher text. The downside of CBC is that errors propagate. Thus, if there is a computation error or a block goes missing, the remainder of the data after the problematic block cannot be decrypted. However, CBC is considered strong and is widely used by symmetric encryption–based solutions.

CTM (Counter Mode) is similar to CBC in that an additional value is added or incorporated into each block prior to encryption; the difference is that CTM does not use a random number and does not chain the blocks. Instead, CTM uses an independent counter, which both the sender and receiver have access to. Each block uses a counter value as the IV, and then the counter is incremented for the next block. This method avoids error propagation but does require a synchronized counter, which can be difficult to implement and keep secret. CTC is considered strong only for ciphers using block sizes of 128 bits or larger.

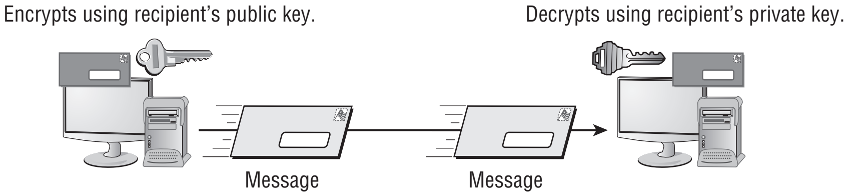
GCM (Galois Counter Mode) is an advancement over CTR that adds a hashing function to confirm the integrity of deciphered data. However, while GCM uses hashing to check integrity, it is described as an authentication code. GCM, like CTR, requires blocks of at least 128 bits. GCM is considered a secure mode and is widely used, especially in TLS cipher suites with AES (often listed as AES_128_GCM or AES_256_GCM).

Asymmetric algorithms

Asymmetric cryptography is often called public key cryptography. However, these terms aren't exactly synonymous. All public key cryptography systems are asymmetric, but there are asymmetric systems that aren't public key cryptography. These non–key-based asymmetric systems include Diffie-Hellman and ElGamal, both discussed later in this section.

Public key cryptography uses key pairs consisting of a public key and a private key (see Figure 6.2). Each communication partner in an asymmetric cryptography solution needs its own unique key pair set (a private key and a public key); this makes asymmetric cryptography much more scalable than symmetric. The private key of the key pair must be kept private and secure. The public key of the key pair is distributed freely and openly.

FIGURE 6.2 An asymmetric encryption system


Encrypts using recipient's public key.
Decrypts using recipient's private key.
Message
Message

The public and private keys are related mathematically, but possession of the public key doesn't allow someone to generate the private key. Thus, the integrity of the private key is protected. The mechanism that provides this security is called a *one-way function*. A one-way function is a mathematical operation that easily produces output values for each possible combination of inputs but makes it impossible to retrieve the input values. Public key cryptosystems are all based on some sort of one-way function. In practice, however, it's never been proven that any specific known function is truly one-way. Cryptographers rely on functions that they suspect may be one-way, but it's theoretically possible that those functions might be broken by future cryptanalysts. But for a one-way function to be useful in encrypted communications, it must be possible to decrypt the received communication.

A variation of the one-way function is the trapdoor one-way function. This is a mathematical system where the encryption provided by one key cannot be reversed or undone by the same key, but another, different key can be used to perform the decryption function. This second, different key is known as the trapdoor. In public key cryptography the public and private keys are used in a trapdoor one-way function system where each key can perform one-way encryption that it cannot decrypt, but the other key of the pair set is the trapdoor to reverse the process. In other words, the private key's encryption is one-way for the private key, but the public key is the trapdoor that can decrypt the private key's encryption, and the public key's encryption is one-way for the public key, but the private key is the trapdoor that can decrypt the public key's encryption. As will be discussed later, when the private key is used to perform encryption, it produces a feature known as a digital signature, and when the public key is used to perform encryption, the result is a feature known as a digital envelope.

Thus, the keys always work in unison. If the public key is used to encrypt data, only the private key can decrypt it. Likewise, if the private key is used to encrypt data, only the public key can decrypt it. Keep in mind the term "different" when thinking about asymmetric cryptography. Either different keys are used for different purposes, such as when the public and private keys are used for opposite operations (such as the private key is used to create a digital signature then the public key is used to verify the signature), or the asymmetric algorithm uses something other than keys to perform an operation, such as Diffie-Hellman.

Asymmetric public key cryptography can be implemented as follows:

1. The sender writes a message.
2. The sender encrypts the message with the sender's private key to produce the interim message package.
3. The sender encrypts the interim message package with the recipient's public key to produce the message package.
4. The sender transmits the message package to the recipient.
5. The recipient decrypts the message package using the recipient's private key to produce the interim message package.
6. The recipient decrypts the interim message package using the sender's public key to extract the original message.

Asymmetric cryptography is much slower than symmetric cryptography, so it isn't generally suited for encrypting a large amount of data. It's often used as the secure exchange mechanism for symmetric cryptographic keys. It provides several security services: authentication, integrity protection, and nonrepudiation.

The most widely used asymmetric cryptography solutions are as follows:

- Rivest, Shamir, and Adleman (RSA)
- Diffie-Hellman
- ElGamal
- Elliptic curve cryptography (ECC)

RSA, the most famous public key cryptosystem, is named after its creators. In 1977, Ronald Rivest, Adi Shamir, and Leonard Adleman proposed the RSA public key algorithm that remains a worldwide standard. They patented their algorithm and formed a commercial venture known as RSA Security to develop mainstream implementations of their security technology. Today, the RSA algorithm forms the security backbone of a large number of well-known security infrastructures produced by companies like Microsoft, Google, Apple, Amazon, and Cisco.

The RSA algorithm depends on the computational difficulty inherent in factoring the product of two large prime numbers. Each user of the cryptosystem generates a pair of public and private keys using a wonderfully complex trapdoor one-way algorithm.

In some cases, neither public key encryption nor offline distribution is sufficient. Two parties might need to communicate with each other, but they have no physical means to exchange key material, and no public key infrastructure is in place to facilitate the exchange of secret keys. In situations like this, key-exchange algorithms like the Diffie-Hellman algorithm prove to be extremely useful mechanisms.

Diffie-Hellman uses a series of one-way functions and nonshared secrets to generate a shared number (which is used as a symmetric key) between two parties across an insecure conversation medium. The Diffie-Hellman algorithm represented a major advance in the state of cryptographic science when it was released in 1976. It's still in use today.

In 1985, Dr. Taher Elgamal published an article describing how the mathematical principles behind the Diffie-Hellman key-exchange algorithm could be extended to support an entire public key cryptosystem used for encrypting and decrypting messages.

At the time of its release, one of the major advantages of *ElGamal* over the Diffie-Hellman and RSA algorithms was that it was released into the public domain. Dr. Elgamal didn't obtain a patent on his extension of Diffie-Hellman, and it's freely available for use, unlike the then-patented RSA and Diffie-Hellman technologies. RSA released its algorithm into the public domain in 2000 and the Diffie-Hellman patent expired in 1994.

However, ElGamal has a major disadvantage: the algorithm doubles the length of any message it encrypts. This presents a major hardship when encrypting long messages or data that will be transmitted over a narrow-bandwidth communications circuit.

In 1985, two mathematicians, Neal Koblitz from the University of Washington and Victor Miller from IBM, independently proposed the application of the *elliptic curve cryptography* (ECC) theory to develop secure cryptographic systems. The mathematical concepts behind ECC are quite complex and well beyond the scope of this book. However, you should be generally familiar with the elliptic curve algorithm and its potential applications when preparing for the Security+ exam.

Computer scientists and mathematicians believe that it's extremely hard to find the solution to the elliptic curve discrete logarithm problem, which forms the basis of elliptic curve cryptography. It's widely believed that this problem is harder to solve than both the prime-factorization problem that the RSA cryptosystem is based on and the standard discrete logarithm problem utilized by Diffie-Hellman and ElGamal. The end result of this mathematical magic is a cryptosolution that can be used on lower-powered devices (those with less CPU capabilities and less memory capacity than a typical computer or notebook, such as mobile phones, netbooks, tablet PCs, e-book readers, and handheld computers), because it is a simpler algorithm using shorter length keys but still provides equivalent security protection. For example, a 1,024-bit RSA key is cryptographically equivalent to a 160-bit ECC key.

Fundamental Differences and Encryption Methods

Some of the differences between symmetric and asymmetric encryption were mentioned in the previous sections. These differences include key length, use of one key or multiple keys (or no keys at all in some cases), and speed.

The length of the cryptographic key is perhaps the most important security parameter that can be set at the discretion of the security administrator. It's important to understand the capabilities of your encryption algorithm and choose a key length that provides an appropriate level of protection. This judgment can be made by weighing the difficulty of defeating a given key length (measured in the amount of processing time required to defeat the cryptosystem) against the importance of the data.

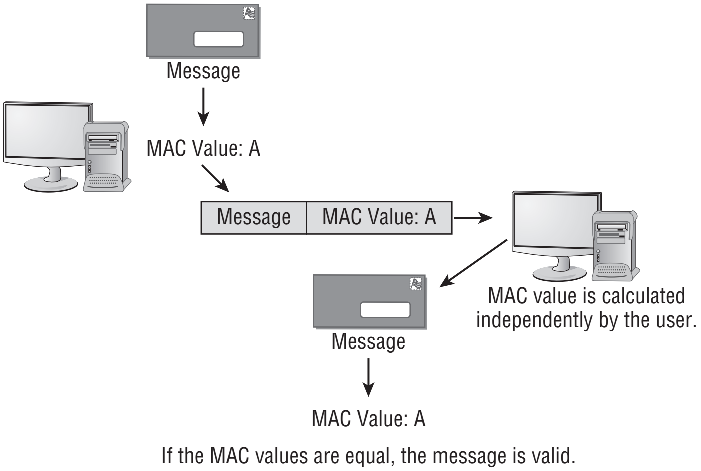
Generally speaking, the more critical your data, the stronger the key you use to protect it should be. Timeliness of the data is also an important consideration. You must take into account the rapid growth of computing power—the famous Moore's law states that computing power doubles approximately every 18 months. If it takes current computers one year of processing time to break your code, it will take only three months if the attempt is made with contemporary technology three years down the road. If you expect that your data will still be sensitive at that time, you should choose a much longer cryptographic key that will remain secure well into the future.

Hashing

Hashing is a type of cryptography that isn't an encryption algorithm. Instead, hashing is used to produce a unique identifier—known as a hash value, hash, checksum, message authentication code (MAC), fingerprint, thumbprint, or message digest—of data. Hashing is a one-way function that creates a fixed-length output from an input of any length. A hash serves as an ID code to detect when the original data source has been altered, since the altered file will produce a different hash value. The data could be a file, a hard drive, a network traffic packet, or an email message. The hash value is used to detect when changes have been made to a resource. In other words, hashing is used to detect violations of data integrity.

For example, a hash value computed now may be compared with a hash value created last week. If the two values are the same, the data hasn't been changed. If the two values are different, the data has been modified. Figure 6.3 shows the basic functionality of a hash or MAC value.

FIGURE 6.3 The MAC value is calculated by the sender and the receiver using the same algorithm.


Message
MAC Value: A
Message
MAC Value: A
MAC value is calculated independently by the user.
Message
MAC Value: A
If the MAC values are equal, the message is valid.

Unlike traditional cryptography, which transforms data into cipher text, hashing produces a hash value without modifying the original data. Because of this special feature, hashing can be used to protect or verify data integrity. It can also be used to verify whether a copy procedure produced an exact duplicate of a data set. For example, when a hard drive is being imaged to create an exact duplicate (as is done in forensic investigations), a hash is produced of the original drive before the duplication process. Then a hash is produced of the original drive and the duplicate drive after the duplication process. If the two hashes of the original drive are the same, no modifications have occurred to the original drive. If the duplicate drive's hash value is the same as the original drive's hash value, it proves the duplicate is an exact copy of the original.

Hashing takes a variable-length input and produces a fixed-length output. For example, Message Digest 5 (MD5) is a 128-bit hash algorithm. This means no matter what the size of the input data, the output hash is always 128 bits long.

The strength of hashing is the fact that it can be performed in only one direction. It isn't mathematically possible to convert a hash value back to its original data. Thus, if someone obtains your hash value, they can't re-create the original data that produced the hash.

Table 6.2 lists well-known hashing algorithms and their resultant hash value lengths in bits. Bookmark this table for memorization.

TABLE 6.2 Hash algorithm memorization chart

Name	Hash value length
Secure Hash Algorithm (SHA-1)	160
SHA-224 (an SHA-2 family member)	224
SHA-256 (an SHA-2 family member)	256
SHA-384 (an SHA-2 family member)	384
SHA-512 (an SHA-2 family member)	512
SHA3-224 (an SHA-3 family member)	224
SHA3-256 (an SHA-3 family member)	256
SHA3-384 (an SHA-3 family member)	384
SHA3-512 (an SHA-3 family member)	512
Message Digest 5 (MD5)	128
Message Digest 4 (MD4)	128
Message Digest 2 (MD2)	128
RIPEMD	160
Hash Message Authentication Code (HMAC)	Variable
Hash of Variable Length (HAVAL)—an MD5 variant	128, 160, 192, 224, and 256 bits

According to RSA Security, the five basic requirements for a cryptographic hash function are as follows:

- The input can be of any length.
- The output has a fixed length.
- The hash function is relatively easy to compute for any input.
- The hash function is one-way (meaning it's extremely hard to determine the input when provided with the output).
- The hash function is collision-free (meaning it's extremely hard to find two messages that produce the same hash value).

However, these requirements don't mean hashing is totally attack-proof. Hashing can be attacked using reverse engineering, reverse hash matching (also known as a rainbow table attack), or a birthday attack.

This form of hashing attack exploits the mathematical characteristic that if two messages are hashed and their hashes are the same, the messages must be the same. This can be written as `H(M)=H(M') then M=M'`. The occurrence of two different data sets that produce the same hash value is known as a collision.

Salt, IV, nonce

A *salt* is secret data added to input material prior to the hashing process. Salting hashes makes the process of attaching hashes much more complicated and computationally intensive. Salts are sometimes part of an authentication system, such as on Linux and many websites, but not on Windows. Authentication salts add additional characters to a password just before it is hashed. Salting passwords makes the act of password cracking more difficult for an attacker.

An *IV (initialization vector)* is a random number added to a cryptographic operation in order to add more chaos to the output cipher text. The recipient must somehow be provided the IV in addition to the key in order to decrypt any data encrypted using a cipher employing an IV.

A *nonce*, or "number used once," is a mathematical term, often a placeholder in a formula. The nonce placeholder is replaced by a random number when the formula is processed. An IV is an example of a nonce.

Elliptic curve

Elliptic curve cryptography (ECC) is cryptographic mathematical magic—or at least that's the way it seems to most mortals who don't have a Ph.D. in mathematics. Basically, it's a method of applying cryptography in order to obtain stronger encryption from shorter keys. For example, an ECC RSA 160-bit key provides the same protection as an RSA 1,024-bit key. ECC is further discussed in the section "Asymmetric algorithms" earlier in this chapter.

Quantum Cryptography

Quantum cryptography takes advantage of the dual nature of light at the quantum level, where it acts as both a wave and a particle. At the quantum level, cryptography could be designed so that communication would be completely protected from eavesdropping or tampering, because the act of listening in on such a secured transmission would affect it enough to damage the data stream. This would make it impossible for the attacker to collect the data and would allow the recipient to detect the attempted interception. For more information on the topic of quantum cryptography, please see the related article on Wikipedia.

Weak/deprecated algorithms

Weak or deprecated algorithms are those that are known to have flaws or those whose protections have been overcome by the computational capabilities of computers. Examples of weak and deprecated algorithms include DES, 3DES, and RC4.

Key exchange

In-band key exchange takes place in the existing and established communication channel or pathway. It's often considered less secure because there is greater risk of an eavesdropping or man-in-the-middle attack being able to capture and/or intercept the exchange.

Out-of-band key exchange takes place outside the current communication channel or pathway, such as through a secondary channel, via a special secured exchange technique in the channel, or with a completely separate pathway technology. Out-of-band key exchange is generally considered more secure, because any attack monitoring the initial channel is less likely to be monitoring or have access to the alternate or separate communications path.

Examples of out-of-band key exchange include using a separate communication session with alternate ports, using an asymmetric key-exchange solution (digital envelopes or Diffie-Hellman), and physical exchange methods (NFC sync, Bluetooth exchange, or QR code scanning).

Digital signatures

A *digital signature* is an electronic mechanism to prove that a message was sent from a specific user (that is, it provides nonrepudiation) and that the message wasn't changed while in transit (it also provides integrity). Digital signatures operate using a hashing algorithm and either a symmetric or an asymmetric encryption solution.

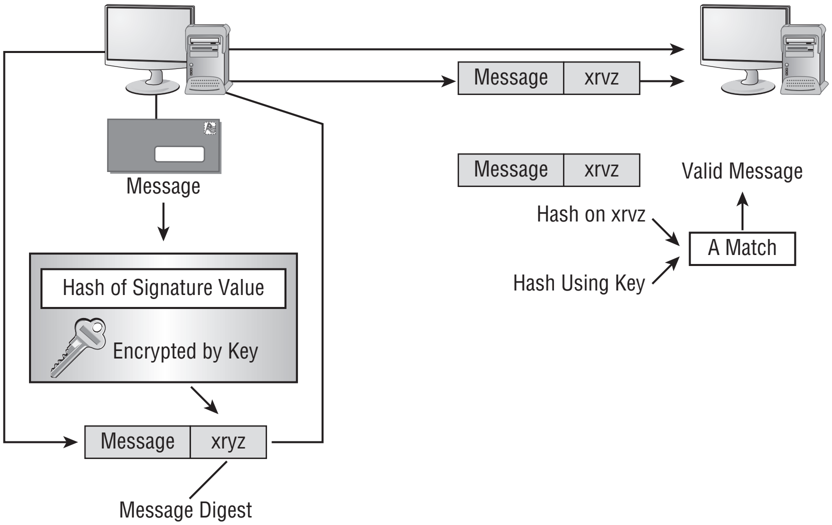
Digital signatures using asymmetric encryption solutions (specifically, public key cryptography, which uses a key pair consisting of a public key and a private key) operate as follows:

1. The sender writes a message.
2. The sender computes a hash of the message.
3. The sender uses the sender's private key to encrypt the hash.
4. The sender attaches the encrypted hash to the message.
5. The complete message package is sent to the receiver.
6. The receiver strips off the encrypted hash (the digital signature).
7. The receiver uses the sender's public key to decrypt the sender's private key and thus extract the hash from the digital signature.
8. The receiver computes a hash of the message.
9. The receiver compares the two hash values.

Digital signatures using symmetric encryption solutions (specifically, where a single shared symmetric key is used) operate as follows (see Figure 6.4):

1. The sender writes a message.
2. The sender computes a hash of the message.
3. The sender uses the shared secret key (symmetric key) to encrypt the hash.
4. The sender attaches the encrypted hash to the message.
5. The complete message package is sent to the receiver.
6. The receiver strips off the encrypted hash.
7. The receiver uses the shared secret key to decrypt the hash.
8. The receiver computes a hash of the message.
9. The receiver compares the two hash values.

FIGURE 6.4 A digital signature process using symmetric encryption



In either case, if the hash values match, the recipient gets verification that integrity was maintained and that the sender did send the message (nonrepudiation and authentication of source). If the hash values don't match, the recipient doesn't have verification of any security benefit.

A simple way to keep this in mind is to remember that a digital signature is built using the sender's private key to encrypt or sign the hash of the message, the signature is verified by the recipient using the sender's public key, and it provides the security benefits of integrity, authentication of source, and nonrepudiation of source.

Digital Envelopes

A *digital envelope* is the alternate process that can be performed using public key cryptography. When confidentiality is needed for a communication, two methods are available to exchange a symmetric key between the endpoints. One is to use a key exchange service such as Diffie-Hellman, and the other is to generate a key locally that is exchanged using a digital envelope. So, once the locally randomly generated symmetric key is generated and used to encrypt the communication or message, the symmetric key is encrypted or enveloped using the recipient's public key. This ensures that only the intended recipient can open or unlock the envelope using his or her corresponding recipient's private key.

A simple way to keep this in mind is to remember that a digital envelope is built using the recipient's public key to encrypt a symmetric key, it is opened or unlocked by the recipient using the recipient's private key, and it provides the security benefits of confidentiality and authentication of the recipient.

Diffusion

Diffusion is a feature or function of a cryptographic algorithm to ensure that small changes in input or plain text would result in large changes in output or cipher text. It is also known as the avalanche effect. This feature minimizes information disclosure based on repeated encryption of the same or similar data. Most modern algorithms implement diffusion.

Confusion

Confusion is a feature or function of a cryptographic algorithm to ensure that details about the key used during the encryption process are not disclosed in the cipher text. Thus, analyzing the cipher text will not reveal information about the key, such as its length or content. Most modern algorithms implement confusion.

Collision

A *collision* occurs when two different data sets produce the same hash value. Thus, hashing is effective at differentiating errors, corruption, or counterfeits. However, an attacker might be able to discover or craft other data sets with a matching hash, so recipients must look at and compare the data sets themselves. Fortunately, it is improbable that a data set with a colliding hash will be similar to the original because of the diffusion feature.

Steganography

Steganography is a process by which one communication is hidden inside another communication. This can be as simple as hiding a code within a sentence that can be extracted by reading only every fifth word or as complex as embedding a text document inside a movie or audio file. One of the most common forms of steganography is to hide text inside graphics.

Steganography often uses passwords as secrets to prevent third parties from extracting the stored communication and may also employ encryption to prevent or hinder brute-force attempts at extraction. Steganography can be used to detect theft, fraud, or modification when the hidden communication is a watermark.

Obfuscation

Obfuscation is the intentional hiding or masking of a communication or its meaning. Either the resulting communication is not recognized as a communication or the meaning of the communication is unknown or unclear. Obfuscation does not require the use of cryptography or steganography, since it is a means of hiding data or performing a communication that will be unrecognized by others. Examples of obfuscation include blinking a light on a device in Morse code so that only the intended recipient knows to look, receive, and decode the communication, or sending flowers to a location with a note using confusion or ambiguous language so that only the intended recipient knows how to extract meaning from the note.

Stream vs. block

Symmetric cryptography is divided into two subforms: block and stream. *Block ciphers* operate on "chunks," or blocks, of a message and apply the encryption algorithm to an entire message block at the same time. The transposition ciphers are examples of block ciphers. The simple mechanism used in the challenge-response algorithm takes an entire word and reverses its letters. The more complicated columnar transposition cipher works on an entire message (or a piece of a message) and encrypts it using the transposition algorithm and a secret keyword. Most modern encryption algorithms implement some type of block cipher.

Stream ciphers are ciphers that operate on each character or bit of a message (or data stream) one character/bit at a time. The Caesar cipher (or C3 cipher) is a three-letter shifted monoalphabetic substitution cipher and is an example of a stream cipher. The one-time pad is also a stream cipher because the algorithm operates on each letter of the plain text message independently. Stream ciphers can also function as a type of block cipher. In such operations, a buffer fills with real-time data that is then encrypted as a block and transmitted to the recipient.

Other than the basic difference in whether the original data is preexisting and static or produced on the fly, both ciphers function in much the same manner. Unless the symmetric cryptography solution is based on a one-time pad (meaning every key is used only once), the same encryption key is used on each block or buffer block for a given data set or communication session.

One-time Pads

A *one-time pad* is the basis of many forms of modern cryptography, from SSL to IPSec to dynamic one-time password tokens. The concept is that a real or virtual paper pad contains codes or keys on each page that are random and don't repeat. Each page of the pad (each key or code) can be used once for a single operation, and then it's discarded—never to be reused or valid again. This concept defines the most secure form of encryption possible. However, because computers can't create true random numbers, we're using pseudo–one-time pad systems, which are very good—just not perfect.

Key strength

Key strength is based on length and randomness. Generally, the longer the key in binary digits, the more strength it provides. Specifically, that strength is often measured in the predicted amount of time or computational effort that would be involved in attempting a full brute-force attack against an encrypted file in order to discover the corresponding secret key. When the key is longer, there are more keys in the keyspace to be evaluated during the attack. Thus, the additional time needed to try all the options of valid keys means the attacker will be less likely to be successful in determining the correct key.

Key strength is also based on the key being selected at random from the full breadth of the keyspace. The keyspace is the totality of all valid keys usable by an algorithm, which is typically the range of keys between all bits being zero (the bottom of the keyspace) and all bits being one (the top of the keyspace). Keys should not be reused between messages or sessions. Keys should not be selected in a pattern or on a sequential incrementation.

Session keys

Session keys are encryption keys used for a communications session. Typically, session keys are randomly selected (or generated) and then used for only one session. Session keys are often symmetric keys, but asymmetric session keys can be used as well.

Some of the most commonly occurring session keys are those used by SSL/TLS. Session keys can be further secured by using a secure key-exchange mechanism (see the next section) and by using them on a limited basis. The more often an encryption key is used, the less security it provides. This is the case because each new use of an encryption key on another message provides additional information to a potential attacker that may simplify the complexity or shorten the time required for a key-cracking or -guessing attack.

A way to combat this is to perform *rekeying*, which is a means of using keys on a limited basis. Rekeying is the process of discarding a key and creating a new one. Rekeying can be triggered by a wide number of events or circumstances, such as the length of time a conversation lasts, the amount of data transmitted, or a gap or idle period in the transaction.

Ephemeral key

An *ephemeral key* is a key generated at the time of need for use in a short or temporary time frame. An ephemeral key might be used only once or could be used for an entire single communication session before being discarded. Most session keys are (or at least should be) ephemeral. Ephemeral keys are a key element of perfect forward secrecy (see the later section on this topic). Ephemeral keys are in contrast to static or fixed keys, which never or rarely change and are reused over and over again. They are also different from shared or preshared keys, which are used by a number of entities, whereas ephemeral keys are used uniquely and exclusively by the endpoints of a single transaction or session.

Secret algorithm

A secret algorithm is a cryptographic cipher whose content and mechanisms are kept hidden from the public. Some secret algorithms are labeled as proprietary and are included in commercial products, but they are prohibited from being analyzed or evaluated by unauthorized third parties. The NSA has a collection of classified algorithms, labeled as Suite A, which are used internally by the NSA and by some other government and military divisions.

Most algorithms are known and public, allowing anyone to view the mathematical operations of the cipher. Public algorithms are subjected to intense scrutiny from both legitimate ethical cryptographers and malicious hackers. When flaws or attacks are discovered, the algorithm can be discarded or patched to address the issue. With secret algorithms, very few third parties are authorized to evaluate the cipher. Thus, if there are flaws or attacks, they may not be discovered quickly or they might be discovered by attackers.

Data-in-transit

Data isn't always stored statically on a storage device. Thus, you need a range of security mechanisms to provide reasonable protection over a range of events and circumstances. *Data in transit* is data being communicated over a network connection. Session encryption should be used to protect data in transit.

Data-at-rest

Data at rest is data stored statically on a storage device. Storage encryption, such as file encryption or whole-drive encryption, should be used to protect data at rest.

Database and Individual File Encryption

Database encryption uses a DBMS product that includes native encryption features. This is sometimes preferred over whole-drive encryption, which is implemented using a separate or independent solution. Native DBMS database encryption integrates the cryptography functions directly into the database software. This feature is now offered by most commercial or enterprise-grade databases, including Oracle and Microsoft SQL Server.

A benefit of database encryption over whole-drive encryption is that data remains secured until an authorized user makes a valid request to access a data element. With whole-drive encryption, the decryption key is in memory, and any file can potentially be opened and decrypted on the fly. Thus, database encryption provides a greater measure of security against outside attackers, unauthorized users, and invalid requests than whole-drive encryption.

Individual file encryption, or file-by-file encryption, is another option, but it is generally thought to provide less security than a whole-drive solution.

File-by-file encryption typically generates a symmetric encryption random key for each file and then stores that key in an encrypted form using the user's public key on the encrypted file. This allows the user to return with their private key, unlock the stored symmetric key, and then unlock the file itself. Each time the file is viewed, it's resaved using a newly selected random symmetric key.

Problems with individual file encryption include the potential for data loss and recovery abuse. If the user loses her private key or it's corrupted, she will be unable to unlock secured files. If a recovery agent is defined, then a recovery agent can restore the files for the user. A recovery agent must be defined before encryption is set; then, when the symmetric key is stored after being encrypted with the user's public key, it's also stored using the public key of the recovery agent. This system provides a backdoor for the user, but at the same time, another entity—the recovery agent—has access to previously secured data. If the recovery agent is untrustworthy, they may abuse their privileges.

Data-in-use

Data in use is data being actively processed by an application. Open and active data is secure only if the logical and physical environment is secure. A well-established security baseline and physical access control are needed to provide reasonable protection for data in use.

Random/pseudo-random number generation

A *random number generator* is a device or system that can produce numbers that cannot be reasonably predicted. A pseudo-random number generator is a device or system that produces numbers that may be predictable in specific situations or under specific conditions. Traditionally, computers have only been able to function as *pseudo-random* *number generators,* since their random number system was based on a time-stamp clocking measurement. Modern systems may be able to adopt truly random number-generating systems by using numerous sources of unpredictable entropy, such as user mouse movements, light levels, noise picked up by the microphone, information gathered through a camera, network bandwidth utilization, CPU temperature, and current dissipation rate of battery power. These and other additional sources of entropy may require additional hardware.

Key stretching

Key stretching is a collection of techniques that can potentially take a weak key or password and stretch it to become more secure, at least against brute-force attacks. Often, key stretching involves adding iterative computations that increase the effort involved in creating the improved key result, usually by several orders of magnitude. This increased workload may be indistinguishable by the typical end user, but it increases the difficulty of reverse-engineering the key by the same orders of magnitude.

A common example of key stretching is to convert a user's password into an encryption key. A typical user password is 8 to 12 characters long, representing only 64 to 96 bits. A symmetric encryption key should be at least 128 bits for reasonable security, or longer for very strong security. A user's password can be run through a series of variable-length hash operations, which may increment the length by 0, 1, or 2 bits per operation, eventually resulting in a 128-bit (or 192, or 256, and so on) result. There could be hundreds (or hundreds of thousands) of hash iterations performed between the initial input and the final encryption key output. Any attempt to crack the key would require either a brute-force attack on the key itself or a password crack/guess followed by the same hash gauntlet for every attempt. Either means of attack would be daunting and would thus often result in a hacker giving up long before being successful.

Implementation vs. algorithm selection

The strength and reliability of a cryptosystem is not based solely on the algorithms in use. There is also the concern of the software code used to implement the algorithm. A robust algorithm can be improperly coded, resulting in an insecure software solution. It is important to consider the quality of the software code and the reputation of the vendor, as well as the algorithms offered.

Crypto service provider

The *crypto service provider (CSP)* is a software library that implements the CAPI (Microsoft CryptoAPI). CAPI and CSP serve to provide standardized cryptographic functions to applications. This frees software programmers from the burden of having to code algorithms into their own software by enabling them to use straightforward API calls to the native OS support cryptography solutions.

Crypto modules

A *crypto module* is a hardware or software component that can be used to provide cryptographic services to a device, application, and operation system. A crypto module may provide random number generation, perform hashing, perform encryption and decryption functions, and serve as a secure storage container for encryption keys.

Perfect forward secrecy

Perfect forward secrecy is a means of ensuring that the compromising of an entity's digital certificates or public/private key pairs doesn't compromise the security of any session's keys. Perfect forward secrecy is implemented by using ephemeral keys for each and every session; these keys are generated at the time of need and used for only a specific period of time or volume of data transfer before being discarded and replaced.

Each subsequent rekeying operation in a session is performed independently of any previous keys, so each key is nondependent and nondeterminant of any other key employed by the current session (and absolutely no previous or future sessions). This technique ensures that the compromising of a session key would result only in the disclosure of the subsection of the overall conversation encrypted by that key. All other subsections of the overall conversation would remain confidential. Perfect forward secrecy also ensures that if the original asymmetric keys are obtained or disclosed, they can't be used to unlock any prior sessions captured by an eavesdropper or man-in-the-middle attack.

Security through obscurity

Security through obscurity is the concept of attempting to gain security by hiding or not being noticed among the crowd of other targets. This is effectively security hide-and-seek. It is not considered a valid security approach for any organization. Security through obscurity can also be used in relation to system design by assuming that custom code and proprietary design is itself a form of security. The thought here is that if no one knows how the system works or how the code was written, then an attacker will not be able to find a flaw to exploit. Unfortunately, this is rarely sufficient security, because most systems with flaws can be probed by various methods, such as fuzzing, to discover flaws, errors, and programming bugs on an automated random basis.

Common use cases

Cryptography is in widespread use across private networks, over the Internet, and on and through most endpoint devices. In this section, we'll look at several common use cases.

Low power devices

Low-power, or lower-powered, devices, such as smartphones, tablets, and some notebook computers, may have limited CPU capabilities or memory capacities. These devices require cryptography functions that will not place an undue burden of computation on the device and will also minimize latency and delay caused by heavy computational loads. Operating systems and applications for low-power devices may limit the range of available algorithms to those that are more favorable to the hardware's capabilities. Such crypto-solutions may also restrict key sizes to only one or a few generally secure options.

Low latency

Low-latency systems are those that need real-time or near-real-time response and communications, such as navigation, VoIP, and some high-end web services. Encrypting and decrypting communications takes time and effort, so in order to minimize latency many devices have a dedicated crypto-processor that will offload the cryptographic operations from the CPU in order to provide faster security services.

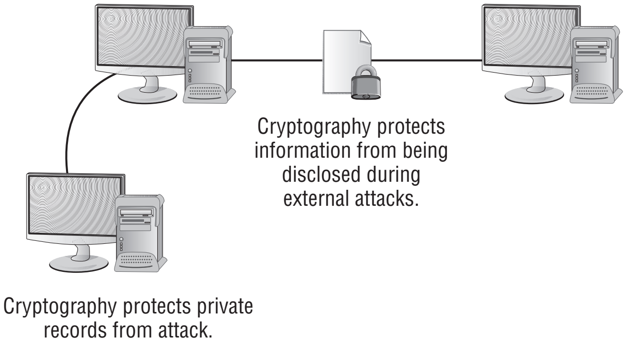
High resiliency

High-resiliency systems are those that want to ensure reliable communications and data storage, often at the expense of higher latency and by requiring more computational capabilities, such as banking and military weapons control systems. Highly resilient systems will often perform one or more reverification passes to ensure data integrity; implement more extensive authentication requirements; and provide robust backup, key storage, and key-recovery options.

Supporting confidentiality

Confidentiality protects the secrecy of data, information, or resources. It prevents or minimizes unauthorized access to data (see Figure 6.5). It ensures that no one other than the intended recipient of a message receives it or is able to read it. Confidentiality protection provides a means for authorized users to access and interact with resources, but it actively prevents unauthorized users from doing so. A wide range of security controls can provide protection for confidentiality, including encryption, access controls, and steganography.

FIGURE 6.5 Cryptographic systems protect data from internal and external disclosure.

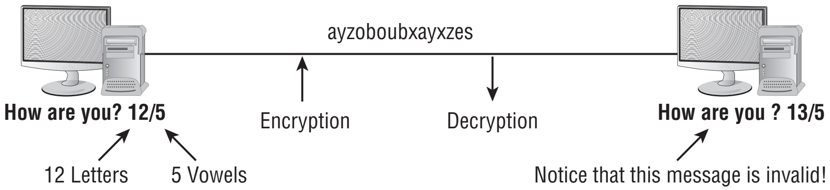

Cryptography protects information from being disclosed during external attacks.
Cryptography protects private records from attack.

Systems that focus on supporting confidentiality will often use keys of longer-than-average length and have a reliable key randomization system.

Supporting integrity

Integrity is the security service that protects the reliability and correctness of data. Integrity protection prevents unauthorized alterations of data (see Figure 6.6). It ensures that data remains correct, unaltered, and preserved. Integrity protection provides a means for authorized changes to be implemented, but it actively prevents unauthorized changes to protected data. Integrity protection resists changes by unauthorized activities (such as viruses or intrusions) and accidents by authorized users (such as mistakes or oversights). Often an integrity check uses a hashing function to verify that data remains unchanged in storage or after transit. Hashing is then used as part of other cryptographic technologies where integrity verification is essential, such as digital signatures, certifications, and nonrepudiation.

FIGURE 6.6 A simple integrity-checking process for an encrypted message



Systems that focus on supporting integrity may calculate several hash values using a range of hashing algorithms in order to reduce the chance of collision.

Availability

Availability is the security service that provides protection for the use of a resource in a timely and effective manner. Often, availability-protection controls support sufficient bandwidth and timeliness of processing as deemed necessary by the organization or situation. When availability is protected, users can perform their work tasks in an efficient and timely manner. When availability is violated, workers cannot accomplish their assigned tasks.

Availability can be violated through the destruction or modification of a resource, overloading of a resource host, interference with communications to a resource host, or prevention of a client from being able to communicate with a resource host. Some of the technologies or concepts that focus on protecting availability are redundancy, fault tolerance, and patching.

Supporting obfuscation

There may be some circumstances or situations where obfuscation is more desirable than actual encryption. These might include communications that are valid or relevant only for a second or two so that the time and effort to perform encryption is greater than the risk of disclosure during the small time frame of value.

Systems that focus on supporting obfuscation will support prompt real-time communications with minimal latency.

Supporting authentication

Authentication is often a key element of a cryptography solution. When sending secured communications, it is essential to control who the recipient is and to verify the recipient before disclosing the content. Cryptographic-based authentication can include knowing a preshared password or key, possessing the correct private key to open a digital envelope, or possessing the correct digital certificate to verify the subject's identity.

Supporting non-repudiation

Nonrepudiation prevents the senders of a message or the perpetrators of an activity from being able to deny that they sent the message or performed the activity. In asymmetric cryptography, nonrepudiation is supported when a sender's private key is used to successfully decrypt a message. This proves that the sender's private key was used to encrypt the data. Because the sender is the only user who has possession of the sender's private key, no one else could have encrypted and sent the message. Often, the security service of nonrepudiation is dependent on authentication and authorization (access control) mechanisms.

Authentication verifies the identity of the sender or recipient of a message. In cryptography terms, authentication occurs differently in symmetric cryptography than it does in asymmetric cryptography. In symmetric cryptography, a single shared secret key is held only by the two communication partners. Thus, when an encrypted message is received and is properly decrypted by the recipient's copy of the shared secret key, authentication occurs. The recipient is authenticated because possession of the correct key proves that this is the correct recipient of the encrypted message. Likewise, the sender is authenticated because the recipient's ability to extract an intelligible message from the received encrypted material using the secret key proves that the sender, the only other user with possession of the same secret key, encrypted and sent the message.

In asymmetric cryptography, a sender uses the recipient's public key to encrypt data. This forces authentication of the recipient because the recipient is the only user in possession of the corresponding private key. Likewise, when the sender's private key is used to encrypt data, any recipient can verify the sender's identity by decrypting that data with the sender's public key.

Access control restricts access to secured data to authorized users. Cryptographic access control is enforced through the concept of possession of encryption keys. In a symmetric cryptography solution, a maximum of two people have valid possession of the shared secret key. Thus possession of the shared secret key is proof of authorization: the holder of the shared secret key is authorized to access anything encrypted with that key. In asymmetric cryptography, only one person is in valid possession of the private key. Thus, possession of the private key is proof of authorization: the holder of the private key is authorized to access anything encrypted with the corresponding public key.

Resource vs. security constraints

Cryptography is able to provide the security benefits of confidentiality, integrity, authentication (of source or recipient or both), and nonrepudiation. It is not able to provide for availability. So, the lack of availability protection is a restriction or constraint of cryptography as a solution.

Cryptography itself may be constrained by resource limitations or security policy or requirements. Resources can include CPU processing capabilities, memory, network capacity, and storage space. If resources are limited or constrained, then cryptography may be restricted or unable to operate. A company security policy or government legislation may prohibit certain uses of cryptography, such as prohibiting the use of a VPN or requiring that all communications be content filtered by a IDS/IPS and firewall.

Exam Essentials

Understand the role confidentiality plays in cryptosystems. Confidentiality is one of the major goals of cryptography. It protects the secrecy of data while it’s at rest and in transit. Confidentiality can be assured by both symmetric and asymmetric cryptosystems.

Know the role integrity plays in cryptosystems. Integrity provides the recipient of a message with the assurance that data wasn’t altered (intentionally or unintentionally) between the time it was created and the time it was accessed. Integrity can be assured by both symmetric and asymmetric cryptosystems.

Be familiar with the basic terminology of cryptography. When a sender wants to transmit a private message to a recipient, the sender takes the plain text (unencrypted) message and encrypts it using an algorithm and a key. This produces a cipher text message that is transmitted to the recipient. The recipient then uses a similar algorithm and key to decrypt the cipher text and re-create the original plain text message for viewing.

Understand symmetric cryptography. Symmetric cryptography is also called private key cryptography or secret key cryptography. Symmetric cryptography uses a single shared encryption key to encrypt and decrypt data. It provides the security service with confidentiality protection.

Know the strengths and weaknesses of symmetric cryptography. Symmetric cryptography is very fast compared to asymmetric cryptography. It provides strong encryption protection when larger keys are used. However, the protection is secure only as long as the keys are kept private. Key exchange under symmetric cryptography is a common problem. Symmetric cryptography isn’t scalable when used alone.

Know common symmetric cryptography solutions. The common symmetric solutions are Advanced Encryption Standard (AES), Triple Data Encryption Standard (3DES), Data Encryption Standard (DES), International Data Encryption Algorithm (IDEA), Blowfish, Twofish, Rivest Cipher 5 (RC5), and Carlisle Adams/Stafford Tavares (CAST-128).

Understand modes of operation. Most symmetric algorithms support several modes of operations, including ECB, CBC, CTM, and GCM.

Understand asymmetric cryptography. Asymmetric cryptography is also called public key cryptography. It uses key pairs consisting of a public key and a private key. Each communication partner in an asymmetric cryptography solution needs only a key pair.

Know the strengths and weaknesses of asymmetric cryptography. Asymmetric cryptography is scalable. The private key of the key pair must be kept private and secure. The public key of the key pair is distributed freely and openly. Possession of the public key doesn't allow someone to generate the private key. Asymmetric cryptography is much slower than symmetric cryptography. It provides three security services: authentication, integrity protection, and nonrepudiation.

Be familiar with the three primary asymmetric algorithms. RSA is the most famous public key cryptosystem; it was developed by Rivest, Shamir, and Adleman in 1977. It depends on the difficulty of factoring the product of prime numbers. ElGamal is an extension of the Diffie-Hellman key-exchange algorithm that depends on modular arithmetic. The elliptic curve algorithm depends on the elliptic curve discrete logarithm problem and provides more security than other algorithms when both are used with keys of the same length.

Know the differences between symmetric and asymmetric cryptosystems. Symmetric key cryptosystems (or secret key cryptosystems) rely on the use of a shared secret key. They're much faster than asymmetric algorithms, but they lack support for scalability, easy key distribution, and nonrepudiation. Asymmetric cryptosystems use public-private key pairs for communication between parties but operate much more slowly than symmetric algorithms.

Understand hashing. Hashing is used to produce a unique data identifier. Hashing takes a variable-length input and produces a fixed-length output. It can be performed in only one direction. The hash value is used to detect violations of data integrity.

Know about hashing attacks. Hashing can be attacked using reverse engineering, reverse hash matching, or a birthday attack. These attack methods are commonly used by password-cracking tools.

Know common hash algorithms. The common hash algorithms are Secure Hash Algorithm (SHA-1), which is a 160-bit hash value; SHA-2, which is a family of four members: SHA-224, SHA-256, SHA-384, and SHA-512; SHA-3, which is a family of SHA3-224, SHA3-256, SHA3-384, and SHA3-512; and Message Digest 5 (MD5), which is a 128-bit hash value.

Understand salts. A salt is secret data added to input material prior to the hashing process. Salting hashes makes the process of attaching them much more complicated and computationally intensive.

Define IV. An IV (initialization vector) is a random number added into a cryptographic operation in order to add more chaos to the output cipher text.

Understand elliptic curve. Elliptic curve cryptography (ECC) is a method of applying cryptography in order to obtain stronger encryption from shorter keys.

Know in-band vs. out-of-band key exchange. In-band key exchange takes place in the existing and established communication channel or pathway. Out-of-band key exchange takes place outside of the current communication channel or pathway, such as through a secondary channel, via a special secured exchange technique in the channel, or with a completely separate pathway technology.

Understand digital signatures. A digital signature is an electronic mechanism used to prove that a message was sent from a specific user and that the message wasn't changed while in transit. Digital signatures operate using a hashing algorithm and either a symmetric or an asymmetric encryption solution. A digital signature is built using the sender's private key to encrypt or sign the hash of the message; the signature is verified by the recipient using the sender's public key; and it provides the security benefits of integrity, authentication of source, and nonrepudiation of source.

Comprehend digital envelopes. A digital envelope is built using the recipient's public key to encrypt a symmetric key, it is opened or unlocked by the recipient using the recipient's private key, and it provides the security benefits of confidentiality and authentication of the recipient.

Know about diffusion. Diffusion is a feature or function of a cryptographic algorithm to ensure that small changes in input or plain text would result in large changes in output or cipher text. This is also known as the avalanche effect.

Know about confusion. Confusion is a feature or function of a cryptographic algorithm to ensure that details about the key used during the encryption process are not disclosed in the cipher text.

Define collision. A collision occurs when two different data sets produce the same hash value.

Understand steganography. Steganography is a process by which one communication is hidden inside another communication.

Define obfuscation. Obfuscation is the intentional hiding or masking of a communication or its meaning. The resulting communication is either not recognized as a communication or the meaning of the communication is unknown or unclear.

Understand block ciphers. A block cipher is a solution that works against a complete static data set. That data set is broken into fixed-length segments called blocks, and each block is encrypted separately.

Understand stream ciphers. A stream cipher is a solution that works against data that is constantly being produced on the fly. Stream ciphers can operate on a bit, character, or buffer basis of encrypting data in real time. A buffer, much like a block, waits to be filled by data as it's produced. When the buffer block is full, that block is encrypted and then transmitted to the receiver.

Know the concept of a one-time pad. A one-time pad is the basis for many forms of modern cryptography, from SSL to IPSec to dynamic one-time password tokens. The concept is that a real or virtual paper pad contains codes or keys on each page that are random and don't repeat. Each page of the pad (each key or code) can be used once for a single operation, and then it's discarded—never to be reused or be valid again.

Understand session keys. Session keys are encryption keys used for a communication session. Typically, session keys are randomly selected (or generated) and then used for only one session.

Know about ephemeral keys. An ephemeral key is a key generated at the time of need for use in a short or temporary time frame. An ephemeral key might be used only once or could be used for a communication session before being discarded. Most session keys are (or at least should be) ephemeral.

Understand key stretching. Key stretching is a collection of techniques that can potentially take a weak key or password and stretch it to become more secure, at least against brute-force attacks. Often, key stretching involves adding iterative computations that increase the effort involved in creating the improved key result, usually by several orders of magnitude.

Know about the crypto service provider. The crypto service provider (CSP) is a software library that implements the CAPI (Microsoft CryptoAPI). CAPI and CSP serve to provide standardized cryptographic functions to applications.

Understand perfect forward secrecy. Perfect forward secrecy is a means of ensuring that compromising an entity's digital certificates or public/private key pairs doesn't compromise the security of any session's keys. Perfect forward secrecy is implemented by using ephemeral keys for each and every session; these keys are generated at the time of need and used for only a specific period of time or volume of data transfer before being discarded and replaced.

Comprehend security through obscurity. Security through obscurity is the concept of attempting to gain security by hiding or not being noticed among the crowd of other targets. This is effectively security hide-and-seek. It is not considered a valid security approach for any organization.

6.2 Explain cryptography algorithms and their basic characteristics.

Algorithms in cryptography are the procedures that dictate how plain text becomes cipher text, and vice versa. A general understanding of a range of algorithms is necessary for the Security+ exam.

Symmetric algorithms

Symmetric algorithms are those ciphers that use a single shared key between sender and receiver to provide secure communications. As discussed previously, symmetric algorithms provide the primary security benefit of confidentiality.

AES

In October 2000, NIST announced that the AES/Rijndael (pronounced “rhine-doll”) block cipher had been chosen as the replacement for DES. In December of that same year, the U.S. Secretary of Commerce approved FIPS 197, which mandated the use of AES/Rijndael for the encryption of all sensitive but unclassified data by the U.S. government.

The original specification for *Advanced Encryption Standard (AES)* called for the processing of 128-bit blocks, but Rijndael exceeded this specification, allowing cryptographers to use a block size equal to any of three key lengths. The number of encryption rounds depends on the key length chosen:

- 128-bit keys require 10 rounds of encryption.
- 192-bit keys require 12 rounds of encryption.
- 256-bit keys require 14 rounds of encryption.

NOTE The other AES finalists were Twofish, Multivariate Adaptive Regression Splines (MARS), and Serpent.

As of 2017, AES is uncracked and has no known weaknesses or flaws in its algorithm. It’s considered one of the best encryption solutions currently available and should be the go-to solution for most users and organizations. In most cases, selecting AES over other options is the best choice in terms of providing long-term, reliable confidentiality protection for your data, whether in transit or in storage.

DES

The U.S. government published the *Data Encryption Standard (DES)* in 1977 as a proposed standard cryptosystem for all government communications. Many government entities continue to use DES for cryptographic applications today, even though it was superseded by AES in December 2001. DES is a 64-bit block cipher that has five modes of operation:

- Electronic Codebook (ECB) mode
- Cipher Block Chaining (CBC) mode
- Cipher Feedback (CFB) mode
- Output Feedback (OFB) mode
- Counter (CTR) mode

All the DES modes operate on 64 bits of plain text at a time to generate 64-bit blocks of cipher text. The key used by DES is 56 bits long. The modes of DES aren't relevant for the Security+ exam, so visit the DES Wikipedia article if you want to know more about them.

DES utilizes a long series of exclusive OR (XOR) operations to generate the cipher text. This process is repeated 16 times for each encryption/decryption operation. Each repetition is commonly referred to as a *round* of encryption, explaining the statement that DES performs 16 rounds of encryption.

As mentioned, DES uses a 56-bit key to drive the encryption and decryption process. However, you may read in some literature that DES uses a 64-bit key. This isn't an inconsistency—there's a perfectly logical explanation. The DES specification calls for a 64-bit key. But of those 64 bits, only 56 actually contain keying information. The remaining 8 bits are supposed to contain parity information to ensure that the other 56 bits are accurate. In practice, though, those parity bits are rarely used. You should commit only the 56-bit figure to memory.

DES is now easily cracked, through either brute-force or precomputed hash techniques, in a matter of minutes. Whenever possible, any other encryption algorithm alternative is preferable to DES. If the software you're using only supports DES, you should seek out an alternative solution; DES isn't providing you with meaningful security.

3DES

As just mentioned, the DES 56-bit key is no longer considered adequate in the face of modern cryptanalytic techniques and supercomputing power. However, an adapted version of DES, *Triple DES (3DES)*, uses the same algorithm to produce a more secure encryption.

There are four versions of 3DES. The first simply encrypts the plain text three times, using three different keys: K_1, K_2, and K_3. It's known as DES-EEE3 mode (the *E*s indicate that there are three encryption operations, whereas the numeral 3 indicates that three different keys are used). DES-EEE3 can be expressed using the following notation, where $E(K,P)$ represents the encryption of plain text P with key K:

$$E(K_1,E(K_2,E(K_3,P)))$$

DES-EEE3 has an effective key length of 168 bits.

The second variant (DES-EDE3) also uses three keys but replaces the second encryption operation with a decryption operation:

$$E(K_1,D(K_2,E(K_3,P)))$$

The third version of 3DES (DES-EEE2) uses only two keys, K_1 and K_2, as follows:

$$E(K_1,E(K_2,E(K_1,P)))$$

The fourth variant of 3DES (DES-EDE2) also uses two keys but uses a decryption operation in the middle:

$$E(K_1,D(K_2,E(K_1,P)))$$

Both the third and fourth variants have an effective key length of 112 bits.

NOTE

Technically, there is a fifth variant of 3DES, DES-EDE1, which uses only one cryptographic key. However, it results in the same algorithm (and strength) as standard DES and is provided only for backward-compatibility purposes.

These four variants of 3DES were developed over the years because several cryptologists put forth theories that one variant was more secure than the others. But the current belief is that all modes are equally secure or relatively insecure. 3DES was a useful product in the late 1990s and early 2000s before AES became widely available. However, 3DES withstands cracking attempts only about three times as long as DES, and thus it no longer provides adequate security for most uses and applications. 3DES should be replaced with AES whenever possible.

RC4

Rivest Cipher 4 (RC4) is a 128-bit stream cipher. It's the foundation of the WEP and WPA encryption used for wireless networking. Please see the discussion of WPA and WEP earlier in this chapter. RC4 is no longer considered a reliable encryption scheme. The later versions of the RC cipher, namely RC5 and RC6, are block ciphers and thus not replacements for RC4.

Blowfish/Twofish

Bruce Schneier's *Blowfish* block cipher is another alternative to DES and IDEA. Like its predecessors, Blowfish operates on 64-bit blocks of text. However, it extends IDEA's key strength even further by allowing the use of variable-length keys ranging from a relatively insecure 32 bits to an extremely strong 448 bits. Obviously, the longer keys result in a corresponding increase in encryption/decryption time. But time trials have established Blowfish as a much faster algorithm than both IDEA and DES. Also, Schneier released Blowfish for public use with no license required.

Blowfish encryption is built into a number of open-source and commercial software products and OSs. In addition, a number of Blowfish libraries are available for software developers. Blowfish can be an acceptable option for encryption, but only when you're using key lengths of at least 128 bits.

The *Twofish* algorithm, also developed by Schneier and derived from Blowfish, was another of the AES finalists. Like Rijndael, Twofish is a block cipher. It operates on 128-bit blocks of data and is capable of using cryptographic keys up to 256 bits in length. If Twofish is an available option in a software product, it's almost the equivalent of AES and thus a secure solution.

Cipher modes

As detailed in the earlier section "Modes of operation," cipher modes are the methods by which the plain text blocks are encrypted into cipher text blocks. You'll find more about each of the following modes in that section.

CBC

CBC (Cipher Block Chaining) mode is used to prevent the creation of duplicate cipher text blocks. See the previous section "Modes of operation."

GCM

GCM (Galois Counter Mode) is an advancement of CTR that adds a hashing function to confirm the integrity of deciphered data. See the previous section "Modes of operation."

ECB

ECB (Electronic Codebook) mode is the simplest and least secure of the symmetric modes. See the previous section "Modes of operation."

CTM

CTM (Counter Mode) is similar to CBC in that an additional value is added or incorporated into each block prior to encryption; however, the difference is that CTM does not use a random number and does not chain the blocks. See the previous section "Modes of operation."

Stream vs. block

Block ciphers operate on "chunks," or blocks, of a message and apply the encryption algorithm to an entire message block at the same time. Stream ciphers are ciphers that operate on each character or bit of a message (or data stream) one character or bit at a time.

Asymmetric algorithms

Asymmetric algorithms are those ciphers that use either a key pair set or no keys at all to perform a range of cryptographic security functions. As discussed previously, asymmetric algorithms may provide the security benefits, including authentication of source and/or recipient and nonrepudiation.

RSA

RSA (Rivest, Shamir, and Adleman) encryption was addressed in the "Asymmetric algorithms" section earlier in this chapter. You'll recall that it's a form of public key cryptography. RSA is still a reliable and secure hashing algorithm, even over 35 years after its initial design in the late 1970s. The only difference between original RSA and modern RSA implementations is the length of the public and private keys. RSA continues to maintain its reliability, security, and speed today, and it's a go-to solution for use in any environment that requires public key cryptography for storage or transmission.

DSA

DSA (Digital Signature Standard) is a method for creating digital signatures and is a FIPS (Federal Information Processing Standard; specifically FIPS 186). Unlike RSA, DSA does not use a private key to directly encrypt the hash of the data or message to create signatures. Instead, DSA uses a unique mathematical function that creates a signature consisting of two 160-bit numbers. These two numbers are produced using a proprietary process involving the original data's hash value and the sender's private key. The sender's public key is then used by the recipient to verify and validate the signature. The verification process is both different from that of RSA and more complex. Most consider the RSA digital signature and that of DSA to be equal in strength and reliability—in terms of verifying the integrity of the delivered message as well as confirming the identity of the sender.

The term DSS (Digital Signature Standard) is another way of referencing the DSA.

Diffie-Hellman

Diffie-Hellman key exchange (D-H or DH) is a means of securely generating symmetric encryption keys across an insecure medium. See the earlier section "Asymmetric algorithms."

Groups

A *Diffie-Hellman group* is a set of starting parameters that determines the length of the initial prime and integer starting values. The bit length listed in the group name is the required length for both of the starting values for that group. Table 6.3 lists the current or in-use groups.

TABLE 6.3 Diffie-Hellman groups currently in use

Group number	Group name	RFC	Predefined
Group 1	768-bit modulus MODP Group	RFC 7296	Yes
Group 2	1024-bit modulus MODP Group	RFC 7296	Yes
Group 5	1536-bit modulus MODP Group	RFC 3526	Yes
Group 14	2048-bit modulus MODP Group	RFC 3526	Yes
Group 15	3072-bit modulus MODP Group	RFC 3526	No
Group 16	4096-bit modulus MODP Group	RFC 3526	No
Group 17	6144-bit modulus MODP Group	RFC 3526	No
Group 18	8192-bit modulus MODP Group	RFC 3526	No
Group 19	256-bit random Elliptic Curve Group	RFC 5903	Yes
Group 20	384-bit random Elliptic Curve Group	RFC 5903	Yes
Group 24	2048-bit MODP Group with 256-bit Prime Order Subgroup	RFC 5114	No

The elliptic curve groups (19 and 20) are preferred because of their high-performance operations. Predefined groups are those that have more rigidly predefined elements and in turn allow for auditing and security verification. Those groups that are not predefined allow for random elements, which may or may not be more secure, but auditing is not possible to make that determination.

DHE

Diffie-Hellman Ephemeral (DHE, aka *Ephemeral Diffie-Hellman [EDH])* is a variation of D-H that is used by TLS to implement perfect forward secrecy by ensuring that random, unpredicted, and nonrepeated starting values are used for each session. This ensures that no session is protected by a predetermined symmetric key and that compromising any of a session's ephemeral keys doesn't assist with the compromising of the other ephemeral keys used during other sessions.

ECDHE

Elliptic Curve Diffie-Hellman Ephemeral, or *Elliptic Curve Ephemeral Diffie-Hellman (ECDHE),* implements perfect forward secrecy through the use of elliptic curve cryptography (ECC). ECC has the potential to provide greater security with less computational burden than that of DHE.

Elliptic curve

Introduced earlier in the chapter, elliptic curve cryptography is an alternate approach to traditional asymmetric public key cryptography that relies on a special class of mathematics known as elliptic curves over finite fields. The result of applying elliptic curve to cryptographic solutions is to obtain stronger encryption, with less computational effort required, while using significantly shorter key lengths.

PGP/GPG

PGP (Pretty Good Privacy) and GPG (GNU Privacy Guard) are email security products. Phil Zimmerman's PGP can also be used to encrypt and digitally sign email messages. PGP is a public-private key system that uses a variety of encryption algorithms to encrypt files and email messages. The first version of PGP used RSA, and the second version used IDEA, but later versions offered a spectrum of algorithm options. PGP uses a proprietary certificate system that is not interoperable with X.509 v3. PGP certificates are not a formal standard but rather an independently developed product that has wide Internet grassroots support, thus becoming a de facto standard.

PGP appeared on the computer security scene in 1991. It combines the certificate authority (CA) hierarchy described later in this chapter under the section "CA" with the "web of trust" concept—that is, you must become trusted by one or more PGP users to begin using the system. You then accept their judgment regarding the validity of additional users and, by extension, trust a multilevel "web" of users descending from your initial trust judgments. PGP initially encountered a number of hurdles to widespread use. The most difficult obstruction was U.S. government export regulations, which treated encryption technology as munitions and prohibited the distribution of strong encryption technology outside the United States. Fortunately, this restriction has since been repealed, and PGP may be freely distributed to most countries.

PGP started off as a free product for all to use, but it has since split into two divergent but compatible products. One is available as a commercial product, and the other is a GNU project now known as GNU Privacy Guard (GnuPG or GPG). GnuPG currently supports ElGamal, DSA, RSA, AES, 3DES, Blowfish, Twofish, CAST5, MD5, SHA-1, RIPE-MD-160, and TIGER. If you haven't used PGP before, we recommend downloading the appropriate GnuPG version for your preferred email platform. This secure solution is sure to improve your email privacy and integrity. You can learn more about GnuPG at `http://gnupg.org`. You can learn more about PGP by visiting its pages on Wikipedia.

Hashing algorithms

Hashing is a one-way function that creates a fixed-length output from an input of any length. Hashing is used to check for integrity violations. This section presents several hashing algorithms.

MD5

The *Message Digest 2 (MD2)* hash algorithm was developed by Ronald Rivest (the same Rivest of Rivest, Shamir, and Adleman fame) in 1989 to provide a secure hash function for 8-bit processors. MD2 pads the message so that its length is a multiple of 16 bytes. It then computes a 16-byte checksum and appends it to the end of the message. A 128-bit message digest is then generated by using the entire original message along with the appended checksum.

Cryptanalytic attacks exist against the MD2 algorithm. Specifically, Nathalie Rougier and Pascal Chauvaud discovered that if the checksum isn't appended to the message before digest computation, collisions may occur. Frederic Mueller later proved that MD2 isn't a one-way function. Therefore, it should no longer be used.

In 1990, Rivest enhanced his message digest algorithm to support 32-bit processors and increase the level of security. This enhanced algorithm is known as *MD4*. It first pads the message to ensure that the message length is 64 bits smaller than a multiple of 512 bits. For example, a 16-bit message would be padded with 432 additional bits of data to make it 448 bits, which is 64 bits smaller than a 512-bit message.

The MD4 algorithm then processes 512-bit blocks of the message in three rounds of computation. The final output is a 128-bit message digest.

Several mathematicians have published papers documenting flaws in the full version of MD4 as well as improperly implemented versions of MD4. In particular, Hans Dobbertin published a paper in 1996 outlining how a modern PC could be used to find collisions for MD4 message digests in less than one minute. For this reason, MD4 is no longer considered to be a secure hashing algorithm, and its use should be avoided if at all possible.

In 1991, Rivest released the next version of his message digest algorithm, which he called *MD5 (Message Digest 5)*. It also processes 512-bit blocks of the message, but it uses four distinct rounds of computation to produce a digest of the same length as the MD2 and MD4 algorithms (128 bits). MD5 has the same padding requirements as MD4—the message length must be 64 bits less than a multiple of 512 bits.

MD5 implements additional security features that reduce the speed of message digest production significantly. Unfortunately, recent cryptanalytic attacks demonstrated that the MD5 protocol is subject to collisions, making it not a one-way function. Specifically, Arjen Lenstra and others demonstrated in 2005 that it's possible to create two digital certificates from different public keys that have the same MD5 hash.

MD5 is probably the most widely used hashing algorithm in the world today and will remain so for at least several more years, because it is coded into operating systems and popular software products. Only when OSs and common software tools shift to SHA-1 or another more advanced hashing system will MD5 use decline. MD5 is generally regarded as sufficient for most situations. Unlike weak encryption, older hashing systems aren't as much of a risk. They have an increased possibility of collision, whereas weak encryption schemes have an increased chance of having confidentiality violated. A greater chance of collisions only means it's slightly less likely that two data sets will produce the same hash value. This doesn't mean you would be fooled by the counterfeit data set; instead, it means the mathematical possibility of a collision being discovered or crafted is greater.

Although there is nothing wrong or flawed with MD5, it has a shorter length hash value and a higher propensity toward collision, and it does require higher levels of computational effort compared to newer algorithms. If you're given a choice of a better hashing algorithm, such as SHA-1, then take it.

SHA

The *Secure Hash Algorithm (SHA)* and its successors are government standard hash functions developed by the National Institute of Standards and Technology (NIST) and are specified in an official government publication—the Secure Hash Standard (SHS), also known as Federal Information Processing Standard (FIPS) 180.

SHA-1 takes an input of virtually any length (in reality, there is an upper bound of approximately 2,097,152 terabytes on the algorithm) and produces a 160-bit message digest. The SHA-1 algorithm processes a message in 512-bit blocks. Therefore, if the message length isn't a multiple of 512, the SHA algorithm pads the message with additional data until the length reaches the next highest multiple of 512. Recent cryptanalytic attacks demonstrated that there are weaknesses in the SHA-1 algorithm. This led to the creation of SHA-2, which has four variants:

- SHA-224 produces a 224-bit message digest using a 512-bit block size.
- SHA-256 produces a 256-bit message digest using a 512-bit block size.
- SHA-512 produces a 512-bit message digest using a 1,024-bit block size.
- SHA-384 uses a truncated version of the SHA-512 hash to produce a 384-bit digest using a 1,024-bit block size.

SHA-3 was released by NIST in 2015. The design of SHA-3 has a completely different basis from that of SHA-2, which makes it more flexible and often more efficient. SHA-3 can be directly substituted for SHA-2 since it produces the same hash lengths: SHA3-224, SHA3-256, SHA3-384, and SHA3-512.

When SHA is offered, it's a better choice than MD5 or other older hashing algorithms. Generally, SHA-3 is preferred over SHA-2, which is preferred over SHA-1. If you're a programmer developing code that uses or needs hashing, you should select SHA-3 (or SHA-2) hashing over other options.

HMAC

The *Hash-Based Message Authentication Code (HMAC)* algorithm implements a partial digital signature—it guarantees the integrity of a message during transmission, but it doesn't provide for nonrepudiation.

HMAC can be combined with any standard message digest–generation algorithm, such as SHA-2, by using a shared secret key. Therefore, only communicating parties who know the key can generate or verify the digital signature. If the recipient decrypts the message digest but can't successfully compare it to a message digest generated from the plain text message, the message was altered in transit.

Because HMAC relies on a shared secret key, it doesn't provide any nonrepudiation functionality (as previously mentioned). However, it may be suitable for applications in which symmetric key cryptography is appropriate. In short, it represents a halfway point between unencrypted use of a message digest algorithm and computationally expensive digital-signature algorithms based on public key cryptography.

HMAC isn't usually a hashing option that is presented to an administrator or even an end user. Instead, specific cryptographic solutions are designed and programmed to take advantage of HMAC. For example, IPSec uses HMAC to reduce the possibility of data collision to a near impossibility.

RIPEMD

RIPEMD-160 is a 160-bit hashing algorithm that is a derivative of *RACE (Research and Development in Advanced Communications Technologies in Europe) Integrity Primitives Evaluation Message Digest (RIPEMD)*, which was itself a variant of MD4. RIPEMD-160 was developed as an alternative to SHA-1, but it hasn't gained wide popularity and thus isn't widely implemented. Use of RIPEMD should generally be avoided when possible; SHA is a much better alternative.

Key stretching algorithms

Key stretching is discussed in the earlier section "Key stretching." Here are two commonly used key stretching algorithms.

BCRYPT

Bcrypt is an example of a key-stretching technology. It's based on the Blowfish cipher, it uses salting, and it includes an adaptive function to increase iterations over time. Bcrypt's adaptive function allows it to operate more slowly over time as the number of iterative operations increases; this reduces the effectiveness of a brute-force attack.

PBKDF2

Password-Based Key Derivation Function 2 (PBKDF2) is another example of a key-stretching technology. It uses a hashing operation, an encryption cipher function, or an HMAC operation (a symmetric key is used in the hashing process) on the input password, which is combined with a salt. This process is then repeated thousands of times. It prevents precomputed hash table attacks and significantly hinders brute-force attacks.

Obfuscation

Obfuscation was covered in the earlier section "Obfuscation." Here are a few examples of obfuscation techniques.

XOR

XOR (eXclusive OR) is an exclusive disjunction, which means that it produces an output of truth (or 1) whenever the two inputs differ (such as one is a 0 [false] and the other is a 1 [true]). It's referred to in mathematical literature as the XOR function and is commonly represented by the ⊕ symbol. The XOR function returns a true value when only one of the input values is true. If both values are false or both values are true, the output of the XOR function is false. Here is the truth table for the XOR operation:

X	Y	X ⊕ Y
0	0	0
0	1	1
1	0	1
1	1	0

The following operation shows sample X and Y values *when they are used as input to the XOR function:*

X:	0 1 1 0 1 1 0 0
Y:	1 0 1 0 0 1 1 1
X ⊕ Y:	1 1 0 0 1 0 1 1

The X value could be an original message and the Y value could be a secret password or an encryption key, and the result can be stored or sent to a recipient. The recipient can extract the original X message by performing another XOR operation using Y.

X: 1 1 0 0 1 0 1 1 (the original $X \oplus Y$)

Y: 1 0 1 0 0 1 1 1

$X \oplus Y$: 0 1 1 0 1 1 0 0 (the original X: value)

XOR is a very commonly used function. XOR is used to compare hash values, implement symmetric stream ciphers, integrate an IV with a block, and use obfuscation.

ROT13

ROT13, or rotation 13, is a substitution cipher based on the 26 letters of the English alphabet (or the basic Latin alphabet). The operation of ROT13 is to shift or substitute each original plain text letter with the letter located in 13 positions further down the alphabet. Rotation is the concept that when you reach the end of the alphabet, the Z, you rotate or restart back at the beginning, the A. A ROT13 key can be written as follows:

```
PT: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
CT: N O P Q R S T U V W X Y Z A B C D E F G H I J K L M
```

Similar to XOR, ROT13 is its own inverse, so reversing the operation simply requires another ROT13 function to return to the original data.

ROT13 and other rotation variations are often implemented in a ring or circle device (such as the prize in a cereal box), which allows for ease of use, portability, and customization of the rotation count. This was and is a simple means of covert communications between friends where the original message is obfuscated using ROT13.

Substitution ciphers

Substitution ciphers use the encryption algorithm to replace each character or bit of the plain text message with a different character. The Caesar cipher (C3) discussed earlier in this chapter is a good example of a substitution cipher. The C3 functions by shifting each letter three places to the right in the message to generate the cipher text. When the end of the alphabet is reached, it wraps around to the beginning of the alphabet so that the plain text character Z becomes the cipher text character C.

```
Original: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
C3:       D E F G H I J K L M N O P Q R S T U V W X Y Z A B C
```

There are many substitution ciphers that are more sophisticated than the examples provided in this chapter. Polyalphabetic substitution ciphers use multiple alphabets in the same message to hinder decryption efforts. One of the most notable examples of a polyalphabetic substitution cipher system is the Vigenère cipher, which uses an encryption/decryption chart, as shown here:

```
PT: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
01: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
02: B C D E F G H I J K L M N O P Q R S T U V W X Y Z A
03: C D E F G H I J K L M N O P Q R S T U V W X Y Z A B
04: D E F G H I J K L M N O P Q R S T U V W X Y Z A B C
05: E F G H I J K L M N O P Q R S T U V W X Y Z A B C D
06: F G H I J K L M N O P Q R S T U V W X Y Z A B C D E
07: G H I J K L M N O P Q R S T U V W X Y Z A B C D E F
08: H I J K L M N O P Q R S T U V W X Y Z A B C D E F G
09: I J K L M N O P Q R S T U V W X Y Z A B C D E F G H
10: J K L M N O P Q R S T U V W X Y Z A B C D E F G H I
11: K L M N O P Q R S T U V W X Y Z A B C D E F G H I J
12: L M N O P Q R S T U V W X Y Z A B C D E F G H I J K
13: M N O P Q R S T U V W X Y Z A B C D E F G H I J K L
14: N O P Q R S T U V W X Y Z A B C D E F G H I J K L M
15: O P Q R S T U V W X Y Z A B C D E F G H I J K L M N
16: P Q R S T U V W X Y Z A B C D E F G H I J K L M N O
17: Q R S T U V W X Y Z A B C D E F G H I J K L M N O P
18: R S T U V W X Y Z A B C D E F G H I J K L M N O P Q
19: S T U V W X Y Z A B C D E F G H I J K L M N O P Q R
20: T U V W X Y Z A B C D E F G H I J K L M N O P Q R S
21: U V W X Y Z A B C D E F G H I J K L M N O P Q R S T
22: V W X Y Z A B C D E F G H I J K L M N O P Q R S T U
23: W X Y Z A B C D E F G H I J K L M N O P Q R S T U V
24: X Y Z A B C D E F G H I J K L M N O P Q R S T U V W
25: Y Z A B C D E F G H I J K L M N O P Q R S T U V W X
26: Z A B C D E F G H I J K L M N O P Q R S T U V W X Y
```

Notice that the chart is simply the alphabet written repeatedly (26 times) under the master heading, shifting by one letter each time. You need a key to use the Vigenère system. For example, the key could be the word "secret." Then, you would perform the following encryption process:

1. Write out the plain text.
2. Underneath, write out the encryption key, repeating the key as many times as needed to establish a line of text that is the same length as the plain text.

3. Convert each letter position from plain text to cipher text.
    - a. Locate the column headed by the first plain text character (a).
    - b. Next, locate the row headed by the first character of the key (s).
    - c. Finally, locate where these two items intersect, and write down the letter that appears there (s). This is the cipher text for that letter position.
4. Repeat step 3 for each letter in the plain text message.

```
Plain text:  a t t a c k a t d a w n
Key:         s e c r e t s e c r e t
Cipher text: s x v r g d s x f r a g
```

Although *polyalphabetic substitution* protects against direct *frequency analysis* (the frequency occurrence of letters within a written language, which is retained in the substitution cipher text), it is vulnerable to a second-order form of frequency analysis called *period analysis*, which is an examination of frequency based on the repeated use of the key or secret that reveals a pattern of encryption.

Exam Essentials

Know the Advanced Encryption Standard (AES). The Advanced Encryption Standard utilizes the Rijndael algorithm and is the U.S. government standard for the secure exchange of sensitive but unclassified data. AES uses key lengths and block sizes of 128, 192, and 256 bits to achieve a much higher level of security than that provided by the older DES algorithm.

Understand the basics of the Data Encryption Standard (DES) and Triple DES (3DES). The Data Encryption Standard (DES) is a 64-bit block cipher that provides 56 bits of key strength. 3DES is a variation of DES that has an effective key strength of either 168 bits or 112 bits.

Know RC4 Rivest Cipher 4 (RC4) is a 128-bit stream cipher. It's the foundation of the WEP and WPA encryption used for wireless networking.

Define Blowfish and Twofish. Bruce Schneier's Blowfish block cipher supports keys of 32–488 bits in length. The Twofish algorithm developed by Schneier, derived from Blowfish, was an AES finalist.

Know DSA. DSA (Digital Signature Standard) is a method for creating digital signatures and is a FIPS (Federal Information Processing Standard) (specifically FIPS 186).

Understand DHE and ECDHE. Diffie-Hellman (D-H) Ephemeral (DHE, aka Ephemeral Diffie-Hellman [EDH]) is a variation of D-H that is used by TLS to implement perfect forward secrecy by performing multiple rekey operations in a session. Elliptic Curve Diffie-Hellman Ephemeral, or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE), implements perfect forward secrecy through the use of elliptic curve cryptography (ECC).

Know PGP and GPG. PGP (Pretty Good Privacy) and GPG (GNU Privacy Guard) are email security products that can also be used to encrypt and digitally sign email messages. PGP uses proprietary certificates.

Understand HMAC. The Hash-Based Message Authentication Code (HMAC) algorithm implements a partial digital signature—it guarantees the integrity of a message during transmission, but it doesn't provide for nonrepudiation.

Be able to define RIPEMD. RIPEMD-160 is a 160-bit hashing algorithm that is a derivative of RACE (Research and Development in Advanced Communications Technologies in Europe) Integrity Primitives Evaluation Message Digest (RIPEMD), which was itself a variant of MD4.

Be able to define BCRYPT. Bcrypt is an example of a key-stretching technology. It's based on the Blowfish cipher, it uses salting, and it includes an adaptive function to increase iterations over time.

Be able to define PBKDF2. Password-Based Key Derivation Function 2 (PBKDF2) is an example of a key-stretching technology. It uses a hashing operation, an encryption cipher function, or an HMAC operation (a symmetric key is used in the hashing process) on the input password, which is combined with a salt.

Be able to define XOR. XOR (eXclusive OR) is an exclusive disjunction, which means that it produces an output of truth (or 1) whenever the two inputs differ (such as one is a 0 [false] and the other is a 1 [true]).

Be able to define ROT13. ROT13, or rotation 13, is a substitution cipher based on the 26 letters of the English alphabet (or the basic Latin alphabet). The operation of ROT13 is to shift or substitute each original plain text letter with the letter located in 13 positions further down the alphabet.

Understand substitution ciphers. Substitution ciphers use the encryption algorithm to replace each character or bit of the plain text message with a different character. The Caesar cipher (C3) is a good example of a substitution cipher.

Be familiar with the polyalphabetic substitution cipher. A polyalphabetic substitution cipher is a substitution cipher using multiple alphabets, such as the 26 alphabets of the Vigenère cipher.

Be able to define frequency analysis. Frequency analysis is the frequency occurrence of letters within a written language that are retained in the substitution cipher text.

Be able to define period analysis. Period analysis is an examination of frequency based on the repeated use of the key or secret that reveals a pattern of encryption.

6.3 Given a scenario, install and configure wireless security settings.

Wireless networking, or WiFi, is covered in numerous sections in this book based on the official CompTIA Security+ SY0-501 objectives. Please review the Chapter 2 section “Access point” and the Chapter 1 section “Wireless attacks” before studying this section of Chapter 6.

Cryptographic protocols

You must consider numerous security or cryptographic protocols when implementing wireless. This section lists the standard options of wireless security.

WEP

Wired Equivalent Privacy (WEP) is defined by the IEEE 802.11 standard. WEP uses a predefined shared secret key; however, the shared key is static and shared among all WAPs and device interfaces.

WEP was cracked almost as soon as it was released. Today, it’s possible to crack WEP in less than a minute, thus rendering it a worthless security precaution. Fortunately, there are alternatives to WEP: WPA and WPA2.

WEP is based on RC4, but due to flaws in design and implementation, WEP is weak in several areas, two of which are the use of a static common key and poor implementation of initiation vectors (IVs). When the WEP key is discovered, the attacker can join the network and then listen in on all other wireless client communications.

There are no scenarios where WEP should be considered for use. All wireless devices manufactured since 2000 have support for better wireless security options.

WPA

An early alternative to WEP was *WiFi Protected Access (WPA)*. This technique was an improvement but was itself not fully secure. WPA is an improvement over WEP and still uses the RC4 algorithm. WPA employs the Temporal Key Integrity Protocol (TKIP) or the Cisco alternative Lightweight Extensible Authentication Protocol (LEAP) in order to ensure that each connected session is issued a random encryption key.

WPA supports two forms of authentication: PSK and ENT. PSK, or preshared key, is also known as personal. PSK is the use of a static fixed password for authentication. ENT or enterprise is also known as IEEE 802.1x/EAP. ENT enables the leveraging of an existing AAA service, such as RADIUS or TACACS+, to be used for authentication.

Although WPA was a welcome improvement over WEP, it is no longer considered secure because the RC4 algorithm can now be overcome, thanks to the advancements in computational abilities of high-end computers.

There are scenarios where WPA might still be considered, such as when legacy devices that do not support WPA2 are needed or when physical isolation would prevent an attacker from gaining access to the wireless signal. However, most devices manufactured in the last 10 to 12 years support WPA2 and thus there is rarely a true need to use WPA any longer.

WPA2

The latest standards-based form of wireless encryption is *WPA2 (WiFi Protected Access version 2,* sometimes written as WPA-2). WPA2 uses the same two authentication options as WPA: PSK and ENT. WPA2 uses the AES encryption algorithm and a key exchange technique known as Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP). WPA2 was defined in IEEE 802.11i.

Most scenarios involving wireless communications should be using WPA2. Although PSK may be convenient and easy to use, especially in home or small environments, ENT should be implemented whenever a dedicated AAA service is available.

CCMP

Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP) was created to replace WEP and WPA's TKIP. CCMP is based on AES. It's the preferred standard security protocol of 802.11 wireless networking indicated by 802.11i. To date, no attacks have been successful against AES/CCMP encryption. Most wireless scenarios would benefit from using CCMP/AES/WPA-2.

TKIP

Temporal Key Integrity Protocol (TKIP) was designed as the replacement for WEP without requiring replacement of legacy wireless hardware. TKIP was implemented into 802.11 wireless networking under the name WPA. TKIP improvements include a key-mixing function that combines the initialization vector (IV—a random number) with the secret root key before using that key with RC4 to perform encryption; a sequence counter is used to prevent packet-replay attacks; and a strong integrity check named Michael is used.

TKIP and WPA were officially replaced by WPA2 in 2004. Additionally, attacks specific to WPA and TKIP (coWPAtty and a GPU-based cracking tool) have rendered WPA's security unreliable.

Generally, WPA2 should be implemented instead of WPA with TKIP, so there are few scenarios where the use of TKIP would be preferred.

LEAP

Lightweight Extensible Authentication Protocol (LEAP) is a Cisco proprietary alternative to TKIP for WPA. It was developed to address deficiencies in TKIP before the 802.11i/WPA2 system was ratified as a standard. An attack tool known as AsLEAP was released in 2004 that could exploit the ultimately weak protection provided by LEAP. LEAP should be avoided when possible; use of EAP-TLS as an alternative is recommended, but if LEAP is used, a complex password is strongly recommended.

Authentication protocols

Wireless communications need to be encrypted and properly authenticated in order to be considered secure. This section examines various options of authentication available for wireless.

EAP

Extensible Authentication Protocol (EAP) isn't a specific mechanism of authentication; rather, it's an authentication framework. Effectively, EAP allows for new authentication technologies to be compatible with existing wireless or point-to-point connection technologies. More than 40 different EAP methods of authentication are widely supported. These include the wireless methods LEAP, EAP-TLS, EAP-SIM, EAP-FAST, EAP-TLS, and EAP-TTLS.

EAP-SIM

EAP-SIM (Subscriber Identity Module) is a means of authenticating mobile devices over the Global System for Mobile communications (GSM) network. Each device/subscriber is issued a subscriber identity module (SIM) card, which is inserted into the device to verify and validate it onto the network and associate its use with the subscriber's account and service level.

PEAP

Protected Extensible Authentication Protocol (PEAP) encapsulates EAP methods within a TLS tunnel that provides authentication and potentially encryption. Because EAP was originally designed for use over physically isolated channels and hence assumed secured pathways, EAP usually isn't encrypted. So, PEAP can provide encryption for EAP methods.

EAP-FAST

Flexible Authentication via Secure Tunneling (EAP-FAST) is a Cisco protocol proposed to replace LEAP, which is now obsolete-thanks to the development of WPA2.

EAP-TLS

EAP Transport Layer Security (EAP-TLS) is an open IETF standard that is an implementation of the TLS protocol for use in protecting authentication traffic. EAP-TLS is considered one of the strongest EAP standards available. EAP-TLS is most effective when both client and server (wireless endpoint device and wireless base station) have a digital certificate.

EAP-TTLS

EAP Tunneled Transport Layer Security (EAP-TTLS) is an extension of EAP-TLS that creates a VPN-like tunnel between endpoints prior to authentication. This ensures that the client's username is never transmitted in clear text.

IEEE 802.1x

This topic is covered in Chapter 4's section "IEEE 802.1x."

RADIUS Federation

RADIUS Federation is an option under 802.1x in which the various devices of the wireless network are able to authenticate using RADIUS rather than a fixed static password that is shared by all. The federation concept here is the ability to authenticate devices regardless of vendor hardware, operating system, or application software, since RADIUS is an interoperable open standard.

Methods

This section examines the various authentication or access control techniques that may be used on a wireless network.

PSK vs. Enterprise vs. Open

A *preshared key (PSK)* is exactly what it sounds like. Two separate parties share a key via an out-of-band communication method prior to communication. This was part of the problem with WEP, because the same value used for encryption was also used for authentication. Under WEP, this is known as shared-key authentication (SKA) and everyone connecting to the same wireless network used the same value. The PSK under WPA and WPA2 is still a fixed value for all users of a wireless network, but it can be defined as a much stronger password/passphrase than the PSK of WEP, and the PSK isn't involved in the key assignment for wireless encryption. PSK should be used in scenarios where ENT authentication is not available.

Enterprise, or *ENT*, is also known as *IEEE 802.1x/EAP*. ENT enables the leveraging of an existing AAA service, such as RADIUS or TACACS+, to be used for authentication. This enables a unique authentication per connected user/device rather than a fixed shared authentication as with PSK. ENT authentication should be used in any scenario where a leverageable AAA service is available.

An *open wireless network* has no authentication and thus usually no encryption. Open wireless networks should be avoided when possible. Most public wireless network scenarios use open networks to keep the hassle and troubleshooting of offering Internet connectivity to users, visitors, and customers to a minimum. Open wireless networks are insecure. If using an open wireless network is the only available option for you at a given location, then connect to a VPN service. The VPN will encrypt all traffic from your device, across the open WiFi network, and across a portion of the Internet (at least until the VPN service's server is reached). This use of a VPN will greatly reduce the risk of using a public open WiFi network.

WPS

This topic is covered in the Chapter 1 section "WPS."

Captive portals

A *captive portal* is an authentication technique that redirects a newly connected wireless web client to a portal access control page. The portal page may require the user to input payment information, provide logon credentials, or input an access code. A captive portal is also used to display an acceptable use policy, a privacy policy, and a tracking policy to the user, who must consent to the policies before being able to communicate across the network.

Captive portals are most often located on wireless networks implemented for public use, such as at hotels, restaurants, bars, airports, libraries, and so on. However, they can also be used on cabled Ethernet connections. Captive portals can be used in any scenario where the owner or administrator of a connection wants to limit access to authorized entities (which might include paying customers, overnight guests, known visitors, or those who agree to a security policy and/or terms of service).

Exam Essentials

Define WEP. Wired Equivalent Privacy (WEP) is defined by the IEEE 802.11 standard. It was designed to provide the same level of security and encryption on wireless networks as that found on wired or cabled networks. WEP provides protection from packet sniffing and eavesdropping against wireless transmissions. A secondary benefit of WEP is that it can be configured to prevent unauthorized access to the wireless network. WEP uses a predefined shared secret key.

Define WPA. An early alternative to WEP was WiFi Protected Access (WPA). This technique was an improvement but was itself not fully secure. It's based on the LEAP and TKIP cryptosystems and employs a secret passphrase.

Define WPA2. WPA2 is a new encryption scheme known as the Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP), which is based on the AES encryption scheme.

Be familiar with CCMP. Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP) was created to replace WEP and WPA. CCMP is based on AES.

Be familiar with TKIP. Temporal Key Integrity Protocol (TKIP) was designed as a replacement for WEP without requiring replacement of legacy wireless hardware. TKIP was implemented into 802.11 wireless networking under the name WiFi Protected Access (WPA).

Define EAP. Extensible Authentication Protocol (EAP) isn't a specific mechanism of authentication; rather, it's an authentication framework. Effectively, EAP allows for new authentication technologies to be compatible with existing wireless or point-to-point connection technologies. More than 40 different EAP methods of authentication are widely supported. These include the wireless methods LEAP, EAP-TLS, EAP-SIM, EAP-FAST, EAP-TLS, and EAP-TTLS.

Understand PEAP. Protected Extensible Authentication Protocol (PEAP) encapsulates EAP methods within a TLS tunnel that provides authentication and potentially encryption.

Define LEAP. Lightweight Extensible Authentication Protocol (LEAP) is a Cisco proprietary alternative to TKIP for WPA. It was developed to address deficiencies in TKIP before the 802.11i/WPA2 system was ratified as a standard.

Understand captive portals. A captive portal is an authentication technique that redirects a newly connected wireless web client to a portal access control page.

6.4 Given a scenario, implement public key infrastructure.

Public key cryptography is technically a subset of asymmetric cryptography. Furthermore, *Public Key Infrastructure (PKI)* is a framework for deploying asymmetric (or public key) cryptography, along with symmetric cryptography, hashing, and certificates, to obtain a real-world flexible and functional secure communications system. The following sections discuss various aspects of PKI and its subelements (rather than focusing only on public key cryptography).

PKI isn't a product; rather, it's a blueprint or concept for a solution. It dictates what should happen and which standards you should comply with, but it doesn't indicate which technologies or algorithms you should use. PKI focuses on proving the identity of communication partners, providing a means to exchange session-based symmetric encryption keys securely through asymmetric cryptographic solutions, and providing a means to protect message integrity through the use of hashing. Most PKI solutions are based on certificates and the use of a CA.

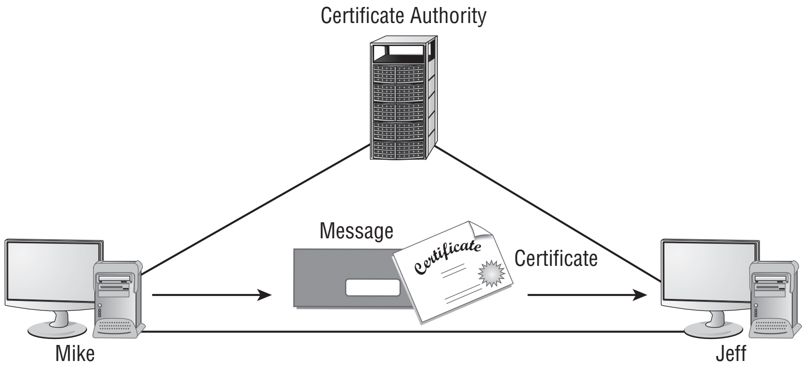
A PKI solution should be implemented in any scenario where a complete end-to-end cryptographic solution is needed. PKI can provide reliable storage and communication encryption, authentication, digital signatures, digital envelopes, and integrity verification.

Components

Digital certificates serve a single purpose: proving the identity of a user or the source of an object. They don't provide proof as to the reliability or quality of the object or service to which they're attached; they only provide proof of where that product or service originated.

Certificates work under a theory known as the *trusted third party*. This theory states that if user A trusts user C and user B trusts user C, then user A can trust B, and vice versa. With certificates, the trusted third party is a certificate authority (CA) (see Figure 6.7). If two users have certificates issued by the CA, then the two users can trust each other's identity. Certificates work this way on the Internet and within private organizations.

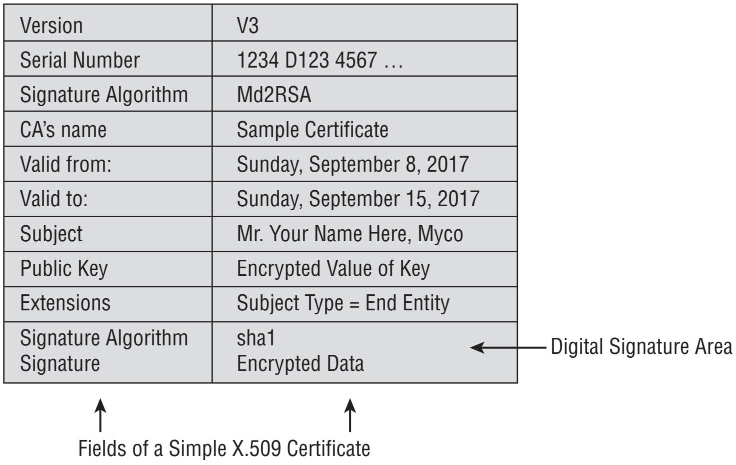
FIGURE 6.7 The CA process



Jeff can verify that the
message with the
certificate from Mike is
valid if he trusts the CA.

Most certificates used on the Internet and within private networks are based on the X.509 version 3 certificate standard. This standard dictates how certificates are to be constructed and their required components, such as the subject's public key, the CA's distinguishing name, a unique serial number, and the type of symmetric algorithm used for the certificate's encryption (see Figure 6.8).

FIGURE 6.8 A certificate illustrating some of the information it stores



Version	V3
Serial Number	1234 D123 4567 …
Signature Algorithm	Md2RSA
CA's name	Sample Certificate
Valid from:	Sunday, September 8, 2017
Valid to:	Sunday, September 15, 2017
Subject	Mr. Your Name Here, Myco
Public Key	Encrypted Value of Key
Extensions	Subject Type = End Entity
Signature Algorithm Signature	sha1 Encrypted Data

Digital Signature Area

Fields of a Simple X.509 Certificate

A user or a subject uses the following procedure to obtain a certificate:

1. The subject requests a certificate from a CA. The request process includes proof of the subject's identity and the subject's public key.
2. The CA verifies the identity of the subject.
3. The CA creates the certificate.
4. The CA validates the certificate by signing it with the CA's private key.
5. The CA issues the certificate to the subject.

CA

Certificates are a means of authentication. They typically involve a trusted third-party *certificate authority (CA)*: a private or public entity that issues certificates to entities serving as either clients (subjects) or servers (objects). It's the responsibility of the CA to verify the identity of each entity before issuing a certificate. After a certificate is issued, the entity can use the certificate as proof of its identity.

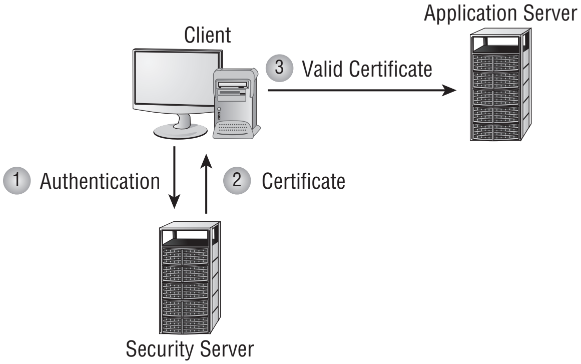
When you're using certificates in a private network, you're your own trusted third party (that is, you're your own CA). Such a private certificate solution is known as a *Public Key Infrastructure (PKI)*. A PKI is the definition (like a blueprint or schematic) of the mechanisms involved in implementing certificates. For the most part, you must deploy one or more servers with certificate services in order to create your own hierarchy of CAs.

Certificates can be used logically or electronically only (such as by the OS or web browser directly) or via physical access control devices (such as a smartcard). In a logical deployment, certificates are installed into a client OS or a specific application. Whenever identity proofing is required, the OS transmits its certificate to the requesting party. In a physical deployment, the certificate is stored on a smartcard or some form of removable media. When the system needs user authentication, it requests the physical access control device.

When users request a certificate, they must usually provide proof of their identity along with their public key. This means they must use the same PKI solution as the CA issuing the certificate. The CA then uses the public key from the subject as the basis to generate the certificate returned to the user (see Figure 6.9). In this fashion, the certificate is tied to the subject's public key–private key pair and provides a mechanism for identity proofing.

Certificates are commonly used over the Internet as a means of logical or electronic identity proofing. As long as both parties in a communication or transaction trust a specific third-party CA, such as VeriSign, then the two entities can trust that each is who they claim to be. It's very important to understand that a certificate only provides proof of identity, and that proof is based on trusting a third party (the CA that performed testing on the subject's identity). It in no way ensures that the subject or object so identified has benevolent motives or will function, perform, or operate in any specific manner.

FIGURE 6.9 A certificate being issued after identification has been verified



A *certificate policy* is a PKI document that serves as the basis for common interoperability standards and common assurance criteria. Certificate policies are acceptable-use policies for certificates: they dictate what is and isn't acceptable with regard to how certificates can be used in an organization. The policies are a set of rules that control how certificates are used, managed, and deployed.

Certificate policies must be all of the following:

- Clear and concise
- Endorsed by senior management
- Restricted to a maximum length of two pages
- Written in bullet-point statements
- Able to provide users (also referred to as *subjects*) with a clear understanding of the acceptable-use policies with regard to certificates

A *certificate practice statement (CPS)* describes how a CA will manage the certificates it issues. The CPS details how certificate management is performed, how security is maintained, and the procedures the CA must follow to perform any type of certificate management from creation to revocation.

Another entity, known as a registration authority (RA), may be deployed in a CA solution. The RA is used to offload the work of receiving new certificate requests and verifying the subject's identity. Once an RA has completed the identity verification process, it sends the CA a formal *certificate signing request (CSR)* for the CA to then actually build, sign, and issue the certificate to the requested subject.

Intermediate CA

An *intermediate CA* is any CA positioned below a root or another CA, but above any leaf CAs. An intermediate CA is a full-fledged CA but simply located in an intermediary or subordinate position in a CA's deployment hierarchy or trust structure.

A *leaf CA* is located at the bottom of a CA trust structure and is the set of CAs that interact directly with customers or end users.

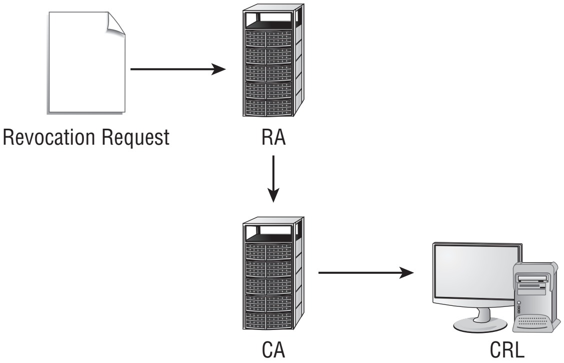
CRL

Certificates have a specific expiration date, which is sometimes called the *lifetime date*. When that date is reached, a certificate is automatically rejected as invalid. However, a CA may have cause to revoke or invalidate a certificate before its predefined expiration date. *Revocation* is the process by which a CA invalidates, cancels, or revokes a certificate.

Revocation may occur because the subject's identity information has changed, the subject used the certificate to commit a crime, or the subject used the certificate in such a way as to violate the CA's certificate policy.

When a certificate is revoked, it's added to the CA's *certificate revocation list (CRL),* its database of revoked certificates. When a certificate's expiration date is reached, it's removed from the CRL because the time stamp automatically invalidates the certificate. Thus, the CRL contains only those certificates that have been revoked but have yet to expire. Figure 6.10 shows the certification revocation process.

FIGURE 6.10 Certification revocation process



The CRL is freely distributed to all users and applications. You should always consult it before accepting a certificate. When issued to a requesting user or application, a CRL is assigned a lifetime date as well. When the CRL exceeds this lifetime, it can no longer be relied on, and you should obtain a new, updated version of the CRL.

When a user application, such as a web browser, receives a certificate from a server, such as a web server, the application first verifies that the date on the certificate is still valid. Next, it checks the local copy of the CA's CRL. If the CRL is no longer valid, an updated copy of the CRL is obtained. The application checks to see whether the certificate appears on the CRL. If it doesn't, the application presents the certificate to the user for a final acceptance choice. The user can elect to accept or reject the certificate as well as to indicate whether to make this same acceptance or rejection choice for all future instances of this certificate.

The CRL process is widely used, but it isn't the only mechanism for informing users and applications about the status of certificates. The *Online Certificate Status Protocol (OCSP)* is another solution that functions on a direct query basis. Each time an application receives a new certificate, it sends a query to an OCSP CA server. The CA responds directly to indicate whether the certificate is still valid or has been revoked. By using OCSP, large CRLs aren't transmitted repeatedly to every requesting system, and queries are direct, immediate, and current.

Most cryptographic keys and all certificates have a built-in expiration date. Upon reaching that date, the key or certificate becomes invalid, and no system will accept it. Keys and certificates are assigned a lifetime with control settings known as valid from and valid to dates. Keys and certificates past their valid to dates should be discarded or destroyed.

If the valid to date for a key or certificate is approaching, you should request a renewal. If you fail to renew before the lifetime expires, then you must perform the complete request process from scratch.

Suspension is an alternative to revocation. Suspension can be used when a key or certificate will be temporarily removed from active use but the subject (or the CA) doesn't wish to invalidate it. When a key or certificate is suspended, it can't be used to sign or encrypt any new items, but previously signed or encrypted items can be verified or decrypted. The key or certificate can be reactivated at a later date.

Suspension status checking is an extension of revocation status checking. However, the results indicate whether a certificate or key is currently valid or in suspension (such a status would be labeled *certification hold*).

Renewal is the process by which a key or certificate is reissued with an extended lifetime date before the key or certificate expires. The renewal process doesn't require a complete repeat of the request and identity proofing process; rather, the old key (which is about to expire) is used to sign the request for the new key. This allows the CA to quickly determine whether the end user's key or certificate can be immediately extended (or reissued with a new lifetime date) or should be rejected and revoked according to its existing lifetime dates. The decision of the CA often depends on the end user's compliance with the organization's certificate policy (the acceptable-use policy for the key or certificate).

After a key or certificate is no longer needed, or when it has expired or been revoked, it should be properly disposed of. This process is known as *key destruction*: the removal of the key or certificate from all software and hardware storage devices. For keys and certificates that are still valid, the CA should be informed about the destruction of the key or certificate. This action allows the CA to update its CRL and OCSP servers.

Reasons to use key destruction include going out of business, changing identity, or having to obtain replacement keys or certificates.

OCSP

See the discussion of Online Certificate Status Protocol (OCSP) in the previous section "CRL."

CSR

A *certificate signing request (CSR)* is the message sent to a certificate authority from an RA on behalf of a user or organization to request and apply for a digital certificate. A CSR often follows the PKCS#10 specification or the Signed Public Key and Challenge (SPKAC) format. A CSR typically includes the generated public key from the applicant's key pair set and the subject's details as defined by the applicable certificate standard (such as x.509 v3); these often include distinguished name, organization name, address/location details, email address, and other contact information.

Certificate

Registration is the process of obtaining a certificate. The specifics may vary based on whether the CA is public or private- as well as the actual software used for the CA services, but the basics of the process are the same throughout. The registration process is as follows:

1. A subject crafts a private key and then generates a public key.
2. The public key is sent to the CA along with proof-of-subject identity.
3. The CA verifies the subject's identity using whatever level of due diligence is warranted.
4. The CA crafts the certificate by digitally signing the subject's public key with the CA's private key, and then it adds a text file containing the details mandated by the X.509 v3 certificate standard.
5. The CA sends the certificate to the subject via a secured pathway.

Some private organizations may generate the initial subject key set on the CA and then issue them both to the subject.

The current certificate standard *X.509 v3* (as defined by RFC 5280) requires the components and structure of a certificate to be as follows:

- Version Number
- Serial Number
- Signature Algorithm ID
- Issuer Name
- Validity Period
- Subject Name
- Subject Public Key Info
- Public Key Algorithm
- Subject Public Key
- Issuer Unique Identifier (optional)
- Subject Unique Identifier (optional)
- Extensions (optional)
- Certificate Signature Algorithm
- Certificate Signature

Most of these items are disclosed in clear text and are visible to anyone viewing the digital certificate. Ultimately, a digital certificate is the public key of a subject signed by the private key of the CA with a clear text document attached disclosing the require component details.

Public key

A *public key* is the key from the public key cryptography key pair set that is designed to be sent out into the public world. Anyone can obtain a person's public key and use it to initiate secure communications with that person. The public key is derived from the private key, but it isn't feasible to reverse the process in order to discover the private key.

Private key

A *private key* is the key from the public key cryptography key pair set that is designed to be kept secured locally and accessible only to the one individual to whom it belongs. The private key is used to unlock communications sent using the corresponding public key. The private key can also be used in crafting digital signatures. The main points regarding the private key is that it should be safeguarded and that it's normally used for decryption.

Object identifiers (OID)

An *Object IDentifier (OID)* is used to name or reference most object types in an X.509 certificate, such as Distinguished Names and Certificate Practices Statements. The OID is a standardized identifier mechanism defined by the International Telecommunications Union (ITU) and International Organization for Standardization/International Electrotechnical Commission (ISO/IEC) that is used to name any object, concept, or thing with an unambiguous persistent name that is unique globally.

Every OID starts off with an initial node from one of three options: 0: ITU-T, 1: ISO, or 2: joint-iso-itu-t. Then, each next element or node is selected from the official OID tree (see a live version of the OID tree at `www.oid-info.com/cgi-bin/display?tree=`). An example of a CA's OID is 2.16.840.1.114412, which references the CA of DigiCert (`https://www.digicert.com`). The breakdown of this OID is as follows:

2 - joint-iso-itu-t

16 – country

840 – United States

1 – organization

114412 – DigiCert

The values in the lower levels of an OID hierarchy, such as the 114412 for DigiCert, must be registered with an OID repository. There are several repositories, most RAs can function as a repository, and all data in a repository is publicly viewable.

Concepts

In any PKI implementation, there are a wide range of concepts to be familiar with. This section points out many of these concepts that are potentially included on the Security+ exam.

Cipher suites

A *cipher suite* is a standardized collection of authentication, encryption, and hashing algorithms used to define the parameters for a security network communication. Most often the term cipher suite is used in relation to SSL/TLS connections. An official TLS Cipher Suite Registry is maintained by the International Assigned Numbers Authority (IANA) at `www.iana.org/assignments/tls-parameters/tls-parameters.xhtml`.

A cipher suite consists of and is named by four elements (for example, TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384):

- A key-exchange mechanism (ECDHE)
- An authentication mechanism used for digital signatures (RSA)
- A symmetric cipher (AES_256_GCM)
- A hashing or message-authenticating code (MAC) mechanism (SHA384)

A client requesting a TLS session sends a preference-ordered list of client-side supported cipher suites as part of the initiation handshake process. The server replies and negotiates with the client based on the highest-preference cipher suite they have in common.

Online vs. offline CA

An *offline CA* is a root CA of a hierarchy that is disconnected from the network and often powered off to be stored in a powered-off state in a physically secure container (such as a vault). The offline CA must be brought back online in order to re-sign or reissue certificates to intermediary, subordinate, or even leaf CAs when their respective certificates are about to expire or have expired. The typical purpose of keeping a CA in an offline state is to prevent compromise of the entire trust hierarchy. The concept is that if the root CA is compromised, then the entire trust environment is compromised. If the root CA is offline, then it cannot be compromised, so at least the root or foundation of the trust structure is protected and can be used to re-create or re-establish the trust structure if it was somehow compromised.

An important consideration with offline CAs is that a root-hosted CRL will not be possible, since that system will not be available. Thus, there will be either no hierarchy-wide CRL or a CRL that is only updated each time the root is brought online. Some CA systems do allow for the delegation of the CRL management to another CA in the hierarchy while the root is offline.

An *online CA* is kept online and network-connected at all times. Online CAs must be kept secure physically and logically in order to manage the risk that an online CA poses.

There is some disagreement about whether an offline CA is actually more secure than an online CA with proper security management. Because several significant CA compromises have occurred with offline CAs (DigiNotar is one example; see `https://en.wikipedia.org/wiki/DigiNotar`), this technique is not as reliable a security mechanism as it was once considered. In fact, many consider offline CAs a form of security by obscurity. A recommended alternative is to use a hardware security module (HSM) to protect the authentication services and encryption keys of all CA systems.

Stapling

OCSP stapling (or stapling, certificate stapling, or previously TLS Certificate Status Request) is a means of checking the revocation status of X.509 digital certificates. This mechanism enables the presenter of a certificate to append or staple a time-stamped OCSP response signed by the issuing CA. This stapling process allows the client or recipient to verify the revocation status of the offered digital certificate without having to interact with the issuing CA's OCSP server directly. Stapling is often viewed as an improvement to the previous OCSP solution; it reduces the workload for the client as well as the OCSP server, while minimizing the risk of DoS against the OCSP system, which could force the client into a default-accept mode if no OCSP response was received.

Pinning

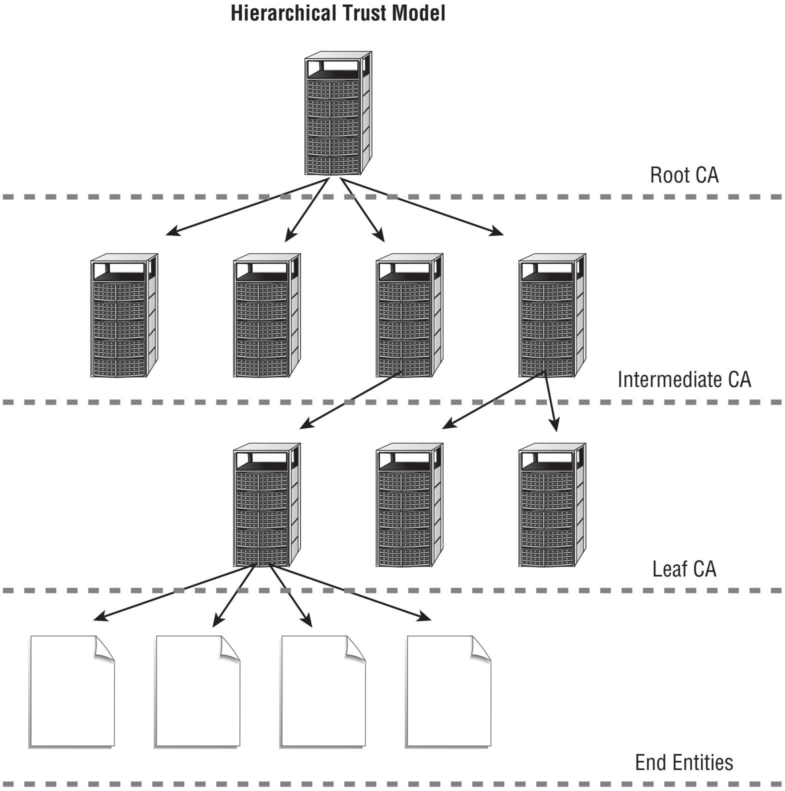
Pinning, or *HTTP Public Key Pinning (HPKP),* is a security mechanism operating over HTTP that enables an HTTPS (TLS secured web service) system to prevent impersonation by attackers through the use of fraudulently issued digital certificates. Pinning operates by providing the visitor with an HTTP response header field value, named Public key-Pins, which includes the hashes of the certificates used by the server along with a time stamp for how long to keep these certificates pinned. Thus, the initial visit to a new site is not secured by pinning. Assuming the initial visit is to the valid site and no man-in-the-middle attacks are taking place, the client receives the list of valid certificates to accept for/from this specific server. Future visits to the same server will compare the offered digital certificate's hash to that of the pinned certificates' hash on the client. If there is a match, then interaction with the website continues unimpeded. If there is no match, then the connection is rejected and a warning message is displayed, indicating that the digital certificate provided by the site did not match one expected by the pinning system.

HPKP is supported by Chrome, Firefox, and Opera, but is not supported by Internet Explorer or Edge.

Trust model

The term *trust model* refers to the structure of the trust hierarchy used by a certificate authority system. The basic trust-model scheme used by CAs is a hierarchical structure with a single top-level root CA. A *root CA* self-signs its own certificate in order to begin the tree of trust. Below the root CA are one, two, or more subordinate CAs. Below each subordinate CA may be one, two, or more subordinate CAs, and so on. Subordinate CAs can sometimes be called *intermediate* or *leaf CAs*. In this model, all CAs have a single parent CA, but they may have multiple child CAs (see Figure 6.11). The root CA is the start of trust; all CAs and participants in a hierarchical trust model ultimately rely on the trustworthiness of the root CA.

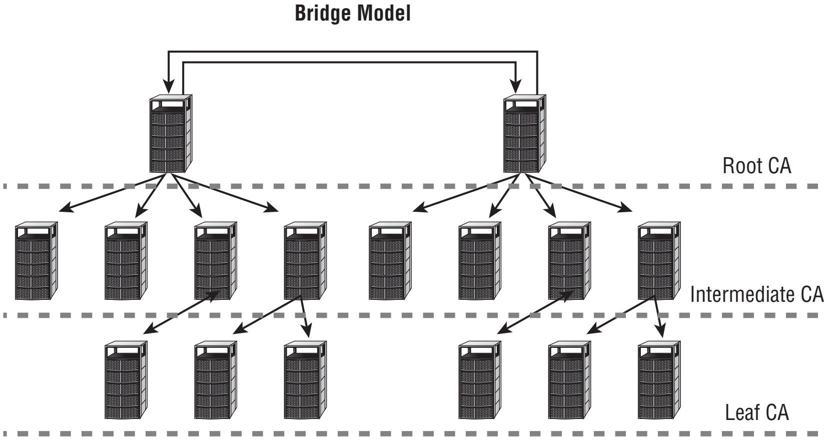
FIGURE 6.11 A hierarchical trust structure



Cross-certification occurs when a CA from one organization elects to trust a CA from another organization (see Figure 6.12). This is also called a *bridge trust structure*. In this way, certificates from either organization are accepted by the other organization. In most cases, the root CA is configured to trust the other root CAs; however, a separate bridge CA may be deployed as a new superior root that other hierarchical root CAs trust. If multiple root CAs trust each other directly to create the bridge trust, this can also be known as a *mesh trust*. A mesh trust requires that each member trust each other member directly; thus this technique is difficult to scale because the trust relationships increase exponentially as the number of trusted CAs increases.

A *trust list* is a form of trust model where a web browser or similar application is provided with a list of root certificates of trusted CAs. The web browser trusts numerous sources of certificates because of the presence of the trusted CA's root certificate on the list of trusted CAs.

FIGURE 6.12 A cross or bridge trust structure



Another trust model option is a *peer trust*, or *web of trust*. This is similar to a mesh trust, but the main difference is that a peer trust does not involve hierarchies or third-party trust structures. Peer trust links are between individual entities without a third-party CA. Instead, each member of the trust is their own CA and self-issues and self-signs certificates that others must accept at face value.

Key escrow

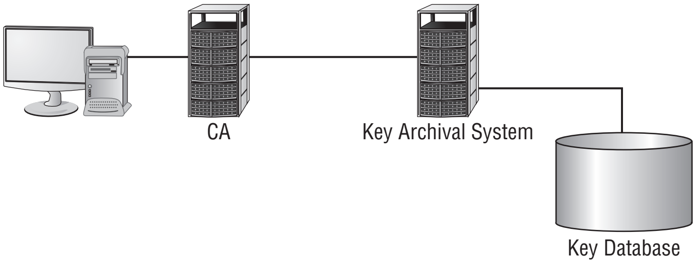
In a symmetric system, all entities in possession of the shared secret key must protect the privacy and secrecy of that key. If the key is compromised anywhere or by anyone, the entire solution (all entities using the same key) is compromised (everything protected by that key).

In an asymmetric system, each end user must protect his or her private key. If an end user's private key is ever compromised, then only that one end user's security is lost.

Key escrow is a storage process by which copies of private keys and/or secret keys are retained by a centralized management system (see Figure 6.13). This system securely stores the encryption keys as a means of insurance or recovery in the event of a disaster. In terms of cryptography, a disaster is when a key is lost or damaged. If such a key is stored in escrow, it can be recovered by a key-escrow agent and used to recover any data encrypted with the damaged or lost key.

However, escrow can be seen in another light if you're an end user who is intent on obtaining complete and total security. If you're assigned your private key or secret key, then the issuing CA (or cryptographic server) probably retains a copy of the key in escrow. This means that at any time, a key-recovery agent could pull your key out of escrow and use it to decrypt anything you've encrypted with your public key or your secret key without your permission. Obviously, key escrow is great for private corporate environments, but it doesn't apply well to the public Internet.

FIGURE 6.13 A key archiving or escrow system



Recovery is the process of pulling a key or certificate from escrow. Recovery can be used when users lose their key or their key has been corrupted. This process can also be used to extract a key for the purpose of decrypting data even when users still have valid possession of their key. The latter option may be necessary in a private corporate environment, but it's unacceptable in a public environment such as the Internet.

Key recovery can only be performed by a key-recovery agent. The *key-recovery agent* is an administrative-level user who has the encryption key to the escrow database. They can decrypt and extract the necessary key from the escrow database and either give a copy to the user or use the key to decrypt all data. If the latter occurs (as is common in most cryptographic solution implementations), the end user must be issued a new key, which must be used in turn to re-encrypt all the data that should be secured.

A key-recovery agent should be a trusted individual. If the environment doesn't warrant the trust of a single key-recovery agent, a mechanism known as *M of N control* can be implemented. M of N control indicates that there are multiple key-recovery agents (M) and that a specific minimum number of these key-recovery agents (N) must be present and working in tandem in order to extract keys from the escrow database. The use of M of N control ensures accountability among the key-recovery agents and prevents any one individual from having complete control over or access to a cryptographic solution.

Key management is the term used to describe the various mechanisms, techniques, and processes used to protect, use, distribute, store, and control cryptographic keys. A key-management solution should follow these basic rules:

- The key should be long enough to provide the necessary level of protection.
- Keys should be stored and transmitted securely.
- Keys should be truly random, should use the full spectrum of the *keyspace* (the range of valid values that can be used as a key for a specific algorithm), and should never repeat.
- The lifetime of a key should correspond to the sensitivity of the data it's protecting.
- The more a key is used, the shorter its lifetime should be.

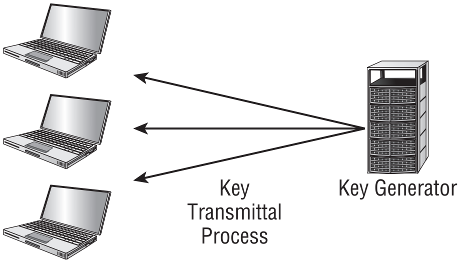
- The shorter the key length or bit length of the algorithm, the shorter the lifetime of the key.
- Keys should be backed up or escrowed in case of emergency.
- Keys should be properly destroyed at the end of their lifetime.

Centralized key management gives complete control of cryptographic keys to the organization and takes control away from the end users. A centralized key-management solution requires a significant investment in infrastructure, processing capabilities, administrative oversight, and communication bandwidth.

In a centralized management solution, copies of all or most cryptographic keys are often stored in escrow. This allows administrators to recover keys in the event that a user loses their key, but it also allows management to access encrypted data whenever it chooses.

Other important aspects of centralized key management are that keys must be created only on secured, dedicated servers; keys can be distributed only to authorized users and only in a secure fashion; keys can be modified only by administrators; and revocation of keys and corresponding digital certificates is at the discretion of the organization. Figure 6.14 shows an example of a centralized key-management solution.

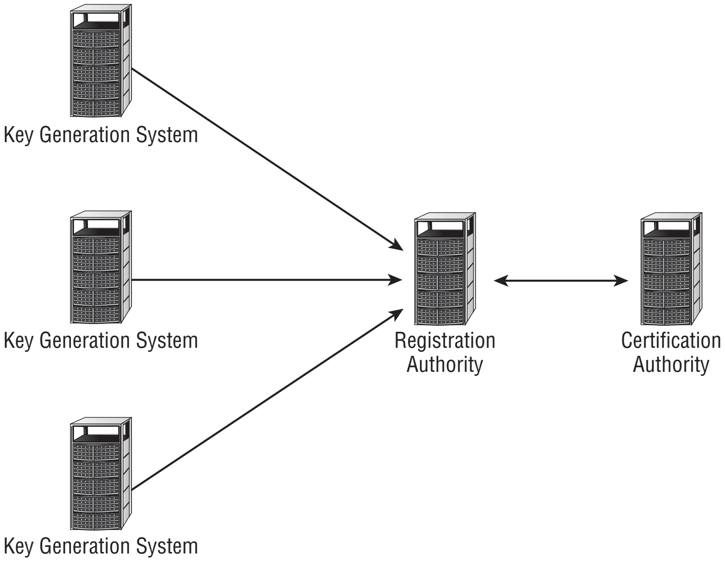
FIGURE 6.14 A centralized key-generating facility



Centralized key management is often unacceptable to a public or open user community because it doesn't provide any control over privacy, confidentiality, or integrity. In fact, every key generated by the centralized system is usually stored in escrow. Therefore, nothing encrypted by an end user is completely private, because an administrator could extract the key from escrow and use it to decrypt a message or file.

An alternate scheme is known as *decentralized key management*. In this type of environment, end users generate their keys (whether symmetric or asymmetric) and submit keys only as needed to centralized authorities (see Figure 6.15). For example, to request a digital certificate, an end user would transmit only their public key to the CA. The end user's private key is always kept private so the end user is the only entity in possession of it. Plus, because the public key is already public, its compromise doesn't result in a complete compromise of the end user's secured solution.

FIGURE 6.15 A decentralized key-generating system



In a decentralized key-management system, end users are ultimately responsible for managing their own keys and using escrow to provide fault tolerance. If an end user fails to take the necessary precautions, a lost or corrupted key could mean the loss of all data encrypted with that key.

Cryptographic keys and digital certificates should be stored securely. If a private key (asymmetric) or a secret key (symmetric) is ever compromised, then the security of all data encrypted with that key is lost. Reliable storage mechanisms must be used to protect cryptographic keys. There are two methods or mechanisms for storing keys: hardware based or software based.

Keys can be stored in either software solutions or hardware solutions. Both offer unique benefits and shortcomings. A software solution offers flexible storage mechanisms and, often, customizable options. However, such a solution is vulnerable to electronic attacks (viruses or intrusions), may not properly control access (privilege-elevation attacks), and may be deleted or destroyed. Most software solutions rely on the security of the host operating system, which may not be sufficient.

Hardware solutions aren't as flexible. However, they're more reliable and more secure than software solutions. Hardware solutions may be expensive and are subject to physical theft. If a user isn't in physical possession of the hardware storage solution, they can't gain access to the secured or encrypted resources. Some common examples of hardware key-storage solutions include smartcards and flash memory drives.

In some situations, you may use multiple key pairs. One key set might be used for authentication and encryption and the other for digital signatures. This allows the first key pair to be escrowed and included on data backups of a centralized key-management scheme. The second key set is then protected from compromise, and the privacy of the owner's digital signature is protected, preventing misuse and forgery.

Certificate chaining

A certificate chain is the relationship between the root CA and the end-user entities. The certificate chain is the linking of the root CA to a first level of intermediate CA, then that CA to potentially other intermediate CAs at other levels, then to the bottom-level leaf CA, then finally to the end-user entity.

Certificate chaining, or establishing a chain of trust, can also occur within a single system or within a private network environment. This concept allows subordinate components to know and trust the element above them without having to directly know and trust the root of the environment (although they may sometimes call the root anchor).

Types of certificates

Many CAs offer a wide range of certificate types. This section lists many types of certificates.

Wildcard

A *wildcard certificate* provides validation for all subdomains under a registered domain. A wildcard of *.mycompany.com on a certificate would also provide authentication for ftp.myexamplecompany.com, www.myexamplecompany.com, stuff.myexamplecompany.com, info.myexamplecompany.com, and anything else with a base of myexamplecompany.com.

SAN

Subject alternative name (SAN) certificates support a range of names for a single entity, such as hostname, site name, IP address, and common name. A SAN certificate is used to provide authentication to multiple names, but only those names specifically defined. Thus, it operates differently than a wildcard.

Code signing

A *code signing certificate* is used to verify the source of source code or compiled code. It is a means for a programmer, developer, or vendor to prove the authenticity and integrity of their software solutions.

Self-signed

A *self-signed certificate* is a certificate signed by the same entity for which it identifies. A root CA issues its first certificate for itself as a self-signed certificate and any peer trust members also issue self-signed certificates.

Machine/computer

A *machine or computer certificate* is issued to verify the identity of a device rather than a service or a user.

Email

An *email certificate* is used to verify a specific email address.

User

A *user certificate* is used to verify a specific individual person.

Root

A *root certificate* is the self-signed certificate issued by a root CA to itself as the means to establish its trust structure, which other devices can enter into by being issued trusted third-party certificates (initially issued only from the root but later to be issued by other intermediary or subordinate CAs within the trust hierarchy).

Domain validation

Domain validation certificates validate a domain name rather than a specific device, server, or system hardware.

Extended validation

An *extended validation (EV) certificate* is issued when the CA has expended considerable additional effort to validate and verify the identity of the subject prior to issuing the certificate. The requirements and criteria for issuing EV certificates are defined by the Guidelines for Extended Validation (`https://cabforum.org/extended-validation/`). Using EV certificates gives end users a sense that they can trust in the verified entities (and improve confidence) because the CA has spent extra effort to verify that they are who they claim to be.

Keep in mind that no CA verification process is foolproof. There are many people in an organization, and not all are always fully trustworthy. And trusted systems are compromised on an all too common basis. Thus, when presented with a standard certificate or an EV certificate, rather than thinking that the verified and validated certificate allows you to extend trust and confidence to the remote entity, consider that it increases your assurance that you will know who to blame when things go wrong. Administrators and end users alike always need to make their own trust and confidence decisions based on the subjective perspective of the target's reputation.

Certificate formats

There are several certificate formats. A certificate format is how the information defined in an X.509 v3 certificate is encoded into a file. Although several formats are interchangeable or interoperable, this is not universally true.

DER

DER (Distinguished Encoding Rules) is a certificate file encoding technique and file extension. DER is a binary formatting rather than ASCII (as is used by PEM). A DER-encoded certificate can be stored in a file with a `.der` or `.cer` extension. DER can be used to store server certificates, intermediate certificates, and private keys. DER can be used for most scenarios, but it is typically used in relation to Java.

PEM

PEM (Privacy-Enhanced Electronic Mail) is a certificate format that uses Base64 (ASCII) to encode the certificate details into a file with a `.pem`, `.crt`, `.cer`, or `.key` extension. PEM certificate files include `"-----BEGIN CERTIFICATE-----"` and `"-----END CERTIFICATE-----"` statements. PEM can be used to store server certificates, intermediate certificates, and private keys. PEM is the most commonly used format.

PFX

PFX (personal information exchange) or *PKCS#12* is a certificate format that stores certificate data in binary. PFX files have extensions of `.pfx` or `.p12`. PFX is most commonly used on Windows systems to import and export certificates and private keys. PFX can be used to store server certificates, intermediate certificates, and private keys.

CER

CER (CERtificate) (an alternate form of `.crt`) is a file extension that can be used to store a DER or PEM formatted certificate.

P12

P12 (PKCS#12) is a file extension option for PFX-formatted certificates.

P7B

P7B or PKCS#7 B is a certificate format that stores certificate data in Base64-encoded ASCII files. P7B-formatted data can be stored in a file with a `.p7b` or `.p7c` extension. P7B certificate files include `"-----BEGIN PKCS7-----"` and `"-----END PKCS7-----"` statements. P7B can be used to store only certificates and chain certificates, not private keys.

Exam Essentials

Understand trusted third parties. Certificates work under a theory known as the trusted third party: if user A trusts user C and user B trusts user C, then user A can trust B, and vice versa. With certificates, the trusted third party is a certificate authority (CA).

Be familiar with certificates. Certificates serve a single purpose: proving the identity of a user or the source of an object. Certificates don't provide any proof as to the reliability or quality of the object or service to which they're attached; they only provide proof of where that product or service originated.

Understand the X.509 version 3 certificate standard. Most certificates are based on the X.509 version 3 certificate standard. Some of the required components are the subject's public key, the CA's distinguishing name, a unique serial number, and the type of symmetric algorithm used for the certificate's encryption.

Define PKI. The Public Key Infrastructure (PKI) focuses on proving the identity of communication partners, providing a means to securely exchange session-based symmetric encryption keys through asymmetric cryptographic solutions, and providing a means to protect message integrity through the use of hashing.

Understand the procedure for requesting a certificate. To request a certificate, a subject submits a request to a CA with proof of their identity and their public key.

Be familiar with certificate policies. A certificate policy is a PKI document that serves as the basis for common interoperability standards and common assurance criteria. It's a statement that governs the use of digital certificates within an organization. Certificate policies are acceptable-use policies for certificates.

Understand certificate practice statements. A certificate practice statement (CPS) describes how a CA will manage the certificates it issues. It details how certificate management is performed, how security is maintained, and the procedures the CA must follow to perform any type of certificate management from creation to revocation.

Understand revocation. A CA may have cause to revoke or invalidate a certificate before its predefined expiration date. Revocation may occur because the subject's identity information has changed, the subject used the certificate to commit a crime, or the subject used the certificate in such a way as to violate the CA's certificate policy.

Understand certificate revocation lists. When a certificate is revoked, it's added to the CA's certificate revocation list (CRL). The CRL is freely distributed to all users and applications. It should always be consulted before recipients accept a certificate and whatever it's associated with.

Define OCSP. The Online Certificate Status Protocol (OCSP) is a revocation solution that functions on a direct query basis. Each time an application receives a new certificate, it sends a query to an OCSP CA server. The CA responds directly to indicate whether the certificate is still valid or has been revoked.

Understand key expiration. Most cryptographic keys and all certificates have a built-in expiration date. Upon reaching that date, the key or certificate becomes invalid, and no system will accept it. Keys and certificates are assigned a lifetime with control settings known as valid from and valid to dates. Keys and certificates past their valid to dates should be discarded or destroyed.

Know about key revocation and status checking. Keys and certificates can be revoked before they reach their lifetime expiration date. Status checking is the process of checking the lifetime dates against the current system date, checking the CRL, and/or querying an OCSP server.

Define key suspension. Suspension is an alternative to revocation. It can be used when a key or certificate will be temporarily removed from active use, but the subject (or the CA) doesn't wish to invalidate the key or certificate outright. Suspension allows a key or certificate to be reactivated at a later date.

Define key recovery. Recovery is the process of pulling a key or certificate from escrow. The recovery process can be used when a user loses their key or their key has been corrupted. Only a key-recovery agent can perform key recovery.

Define key renewal. Renewal is the process by which a key or certificate is reissued with an extended lifetime date before it expires. The renewal process doesn't require a complete repeat of the request and identity proofing process; rather, the old key (which is about to expire) is used to sign the request for the new key.

Define key destruction. After a key or certificate is no longer needed or it has expired or been revoked, it should be properly disposed of. For keys and certificates that are still valid, the CA should be informed about the destruction of the key or certificate. This action allows the CA to update its CRL and OCSP servers.

Understand how a web browser handles new certificates. When a web browser receives a certificate from a web server, it verifies that the date on the certificate is still valid. Next, it checks the local copy of the CA's CRL. If the CRL is no longer valid, an updated copy of the CRL is obtained. The application checks to see if the certificate appears on the CRL. If it doesn't, the application presents the certificate to the user for a final acceptance choice.

Be familiar with CSRs. A certificate signing request (CSR) is the message sent to a certificate authority from an RA on behalf of a user or organization to request and apply for a digital certificate. A CSR often follows the PKCS#10 specification or the Signed Public Key and Challenge (SPKAC) format.

Understand OIDs. An OID (Object IDentifier) is used to name or reference most object types in an X.509 certificate, such as Distinguished Names and Certificate Practices Statements. The OID is a standardized identifier mechanism defined by the ITU and ISO/IEC that is used to name any object, concept, or thing with an unambiguous persistent name that is unique globally.

Be familiar with cipher suites. A cipher suite is a standardized collection of authentication, encryption, and hashing algorithms used to define the parameters for a security network communication. A cipher suite consists of and is named by four elements: key-exchange mechanism, authentication mechanism, symmetric cipher, and hashing mechanism.

Understand stapling. OCSP Stapling (or stapling, certificate stapling, or previously TLS Certificate Status Request) is a means for checking the revocation status of X.509 digital certificates. It is a mechanism that enables the presenter of a certificate to append or staple a time-stamped OCSP response signed by the issuing CA.

Define pinning. Pinning, or HTTP Public Key Pinning (HPKP), is a security mechanism operating over HTTP that enables an HTTPS (TLS secured web service) system to prevent impersonation by attackers through the use of fraudulently issued digital certificates.

Understand trust models. The term trust model refers to the structure of the trust hierarchy used by a certificate authority system. The basic and most common trust model scheme used by CAs is a hierarchical structure.

Understand hierarchical trust models. A hierarchical structure has a single top-level root CA. Below the root CA are one, two, or more subordinate CAs. The root CA is the start of trust. All CAs and participants in a hierarchical trust model ultimately rely on the trustworthiness of the root CA.

Know about cross-certification. Cross-certification or a bridge trust occurs when a CA from one organization elects to trust a CA from another organization. In this way, certificates from either organization are accepted by the other organization. In most cases, the root CA is configured to trust the other root CA.

Understand authentication. In relation to cryptography, authentication is the security service that verifies the identity of the sender or receiver of a message.

Define nonrepudiation. Nonrepudiation prevents the sender of a message or the perpetrator of an activity from being able to deny that they sent the message or performed the activity.

Know how cryptosystems can be used to achieve authentication goals. Authentication provides assurance as to the identity of a user. One possible scheme that uses authentication is the challenge-response protocol, in which the remote user is asked to encrypt a message using a key known only to the communicating parties. Authentication can be achieved with both symmetric and asymmetric cryptosystems.

Know the common applications of cryptography to secure web activity. The de facto standard for secure web traffic is the use of HTTP over Secure Sockets Layer (SSL), otherwise known as HTTPS. Secure HTTP (S-HTTP) also plays an important role in protecting individual messages. Most web browsers support both standards.

Understand the importance of providing nonrepudiation capability in cryptosystems. Nonrepudiation provides undeniable proof that the sender of a message actually authored it. It prevents the sender from subsequently denying that they sent the original message. Nonrepudiation is possible only with asymmetric cryptosystems.

Understand key escrow. Key escrow is a storage process in which copies of private keys and/or secret keys are retained by a centralized management system. This system securely stores the encryption keys as a means of insurance or recovery in the event of a lost or corrupted key.

Know key-management basics. Keys should be long enough to provide the necessary level of protection, should be stored and transmitted securely, should be random, and should use the full spectrum of the keyspace. In addition, they should be escrowed, properly destroyed at the end of their lifetime, used in correspondence with the sensitivity of the protected data, and have a shortened use lifespan if they're used repeatedly.

Understand centralized key management. Centralized key management gives complete control of cryptographic keys to the organization and takes control away from the end users. In a centralized management solution, copies of all cryptographic keys are stored in escrow.

Understand decentralized key management. In decentralized key management, end users generate their keys (whether symmetric or asymmetric) and submit keys only as needed to centralized authorities. The end user's private key is always kept private, so the end user is the only entity in possession of it.

Know about key storage. Cryptographic keys and digital certificates should be stored securely. If a private key (asymmetric) or a secret key (symmetric) is ever compromised, then the security of all data encrypted with the key is lost.

Comprehend M of N control. If the environment doesn't warrant the trust of a single key-recovery agent, a mechanism known as M of N control can be implemented. M of N control indicates that there are multiple key-recovery agents (M) and that a specific minimum number of these key-recovery agents (N) must be present and working in tandem in order to extract keys from the escrow database.

Understand software key storage. A software solution offers flexible storage mechanisms and, often, customizable options. However, a software solution is vulnerable to electronic attacks (viruses or intrusions), may not properly control access (privilege-elevation attacks), and may be deleted or destroyed. Most software solutions rely on the security of the host OS, which may not be sufficient.

Understand hardware key storage. Hardware solutions aren't as flexible as software solutions; however, they're more reliable and more secure. Hardware solutions may be expensive and are subject to physical theft. If a user isn't in physical possession of the hardware storage solution, they can't gain access to the secured or encrypted resources. Some common examples of hardware key storage solutions include smartcards and flash memory drives.

Know about private-key protection. In a symmetric system, all entities in possession of the shared secret key must protect the privacy and secrecy of that key. If the key is compromised anywhere or by anyone, the entire solution (all entities using the same key) is compromised (everything protected by that key).

Comprehend the use of multiple key pairs. In some situations, you may use multiple key pairs. One key set may be used for authentication and encryption and the other for digital signatures. This allows the first key pair to be escrowed and included on data backups of a centralized key-management scheme. The second key set is then protected from compromise, and the privacy of the owner's digital signature is protected, preventing misuse and forgery.

Define certificate chaining. A certificate chain is the relationship between the root CA and the end-user entities.

Know the types of certificates. Many CAs offer a wide range of certificate types, including wildcard, subject alternative name (SAN), code signing, self-signed, machine/computer, email, user, root, domain validation, and extended validation.

Know the common certificate formats. A certificate format is how the information defined in an X.509 v3 certificate is encoded into a file. Format options include DER, PEM, PFX, P12, and P7B. File extensions for certificates include `.cer`, `.der`, `.pem`, `.crt`, `.key`, `.pfx`, `.p12`, `.p7b`, and `.p7c`.

Review Questions

You can find the answers in the Appendix.

1. Which of the following is most directly associated with providing or supporting perfect forward secrecy?
   A. PBKDF2
   B. ECDHE
   C. HMAC
   D. OCSP

2. Which of the following symmetric-encryption algorithms offers the strength of 168-bit keys?
   A. Data Encryption Standard
   B. Advanced Encryption Standard
   C. Triple DES
   D. IDEA

3. The security service that protects the secrecy of data, information, or resources is known as what?
   A. Integrity
   B. Authentication
   C. Nonrepudiation
   D. Confidentiality

4. Digital signatures can be created using all but which of the following?
   A. Asymmetric cryptography
   B. Hashing
   C. Key escrow
   D. Symmetric cryptography

5. When a subject or end user requests a certificate, they must provide which of the following items? (Choose all that apply.)
   A. Proof of identity
   B. A hardware storage device
   C. A public key
   D. A private key

6. From a private corporate perspective, which of the following is most secure?
   A. Decentralized key management
   B. Centralized key management
   C. Individual key management
   D. Distributed key management

7. When should a key or certificate be renewed?

   A. Every year

   B. Every quarter

   C. Just after it expires

   D. Just before it expires

8. Which mode of operation used by symmetric encryption algorithms ensures unique cipher text by integrating an IV into the operation and linking each cipher text block to the next plain text block?

   A. Cipher Block Chaining

   B. Electronic Codebook

   C. Galois Counter Mode

   D. Counter Mode

9. You are the communications officer for a large organization. Your data transfer system encrypts each file before sending it across the network to the recipient. There have been issues with the keys being intercepted as they are sent along the same path as the protect files. What alternative system should be used for key exchange?

   A. Ephemeral

   B. Out-of-band

   C. Sequential

   D. Synchronized

10. Place the steps for creating a digital signature in correct order.

    A. The receiver uses the sender's public key to decrypt the sender's private key and thus extract the hash from the digital signature.

    B. The sender computes a hash of the message.

    C. The complete message package is sent to the receiver.

    D. The sender attaches the encrypted hash to the message.

    E. The receiver computes a hash of the message.

    F. The sender writes a message.

    G. The receiver compares the two hash values.

    H. The receiver strips off the encrypted hash (the digital signature).

    I. The sender uses the sender's private key to encrypt the hash.

       A. H, G, E, B, D, F, A, C, I

       B. F, I, G, A, D, H, C, B, E

       C. C, A, E, I, B, D, G, H, F

       D. F, B, I, D, C, H, A, E, G

11. You are a programmer with a new app for use on smartphones. Your app provides users with a means to securely store personal data, such as their calendar, financial information, and personal contacts in an encrypted container. There is concern that users will be unable to remember a long random encryption key, but you want to use something stronger than just a remembered password. What technique can be used to minimize the information remembered by the user while maximizing the security of the encryption?

A. Session key

B. Ephemeral key

C. Key stretching

D. Secret algorithm

12. What is the least effective form of security?

A. Ephemeral keys

B. Security through obscurity

C. Implicit deny

D. Authentication using certificates

13. As a security-focused systems designer, you need to select the means by which symmetric keys are generated and exchanged between communication endpoints. Which of the following will provide your product with the most secure solution?

A. Digital envelopes

B. Static keys

C. ECDHE

D. Sequential keys

14. What is the result of the following calculation: 1 0 0 1 0 0 1 1 ⊕ 0 0 0 1 1 1 0 0?

A. 1 0 0 1 1 1 1 1

B. 0 0 0 1 0 0 0 0

C. 0 1 1 1 0 0 0 0

D. 1 0 0 0 1 1 1 1

15. Given the ROT13 matrix that follows, what is the plain text of the following cipher text: frphevgl ebpxf?

```
PT: A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
CT: N O P Q R S T U V W X Y Z A B C D E F G H I J K L M
```

A. Windows rules

B. Security rocks

C. Ephemeral crypto

D. Cracking keys

16. What form of wireless can use a RADIUS server to authenticate a wireless client?

A. WEP

B. WPA PSK

C. WPA-2 ENT

D. WPS

17. What form of EAP is considered one of the strongest options, negotiates security using digital certificates similar to HTTPS, and can function over wireless connections?

A. EAP-FAST

B. EAP-SIM

C. PEAP

D. EAP-TLS

18. You are the manager of a restaurant and want to offer your customers wireless connectivity to the Internet. You are concerned that non-patrons will abuse the system and you therefore want to limit access to paying customers. Which of the following solutions would be able to accomplish this?

A. Use an open WiFi network with a hidden SSID.

B. Use a captive portal requiring a code that is provided to customers just after they place their drink order.

C. Post a sign on the wall with the WiFi name and password.

D. Track the MAC addresses of each wireless user and block those that abuse the system.

19. You are implementing a new web server for your organization. There have been issues in the past with hackers impersonating your site in order to harm your clients and visitors. What certificate-based tool can be used to reduce the risk of site impersonation?

A. Pinning

B. Stapling

C. Key escrow

D. Offline CA

20. What type of certificate will enable an organization to verify six specific subdomains with a single certificate but not allow other subdomains to be included?

A. Wildcard

B. SAN

C. Root

D. Domain validation

Appendix

Answers to Review Questions

Chapter 1: Threats, Attacks, and Vulnerabilities

1. C. Banner grabbing is the communications technique a hacker can use to identify the product that is running on an open port facing the Internet.

2. A. The only real option to return a system to a secure state after a rootkit is reconstitution.

3. D. Social engineering is more likely to occur if users aren't properly trained to detect and prevent it. The lack of user awareness training won't have as much impact on man-in-the-middle, reverse hash-matching, or physical intrusion attacks.

4. C. A watering hole attack could be used to plant phone-home-to-identity malware on the systems of subsequent visitors.

5. A, B, and D. A programmer can implement the most effective way to prevent XSS by validating input, coding defensively, escaping metacharacters, and rejecting all script-like input.

6. A. Retroviruses specifically target antivirus systems to render them useless.

7. D. A RAT is a remote access Trojan. A RAT is a form of malicious code that grants an attacker some level of remote control access to a compromised system.

8. A. Phishing is a form of social engineering attack focused on stealing credentials or identity information from any potential target. It is based on the concept of fishing for information. Phishing is employed by attackers to obtain sensitive information such as usernames, passwords, credit card details, or other personally identifiable information by masquerading as a trustworthy entity (a bank, a service provider, or a merchant, for example) in electronic communication (usually email).

9. B. A man-in-the-middle attack is a communications eavesdropping attack. Attackers position themselves in the communication stream between a client and server (or any two communicating entities). The client and server believe that they're communicating directly with each other—they may even have secured or encrypted communication links.

10. D. A buffer overflow attack occurs when an attacker submits data to a process that is larger than the input variable is able to contain. Unless the program is properly coded to handle excess input, the extra data is dropped into the system's execution stack and may execute as a fully privileged operation.

11. C. Refactoring is restricting or reorganizing software code without changing its externally perceived behavior or produced results. Refactoring focuses on improving software's nonfunctional elements (quality attributes, nonbehavioral requirements, service requirements, or constraints). Refactoring can improve readability, reduce complexity, ease troubleshooting, and simplify future expansion and extension efforts.

12. B. Evil twin is an attack where a hacker operates a false access point that will automatically clone or twin the identity of an access point based on a client device's request to connect. Each time a device successfully connects to a wireless network, it retains a wireless profile in its history.

13. A. A hacktivist is someone who uses their hacking skills for a cause or purpose. A hacktivist commits criminal activities for the furtherance of their cause. A hacktivist attacks targets even when they know they will be identified, apprehended, and prosecuted. They do this because they believe their purpose or cause is more important than themselves.

14. C. Open source intelligence is the gathering of information from any publicly available resource. This includes websites, social networks, discussion forums, file services, public databases, and other online sources. This also includes non-Internet sources, such as libraries and periodicals.

15. B. In penetration testing (or hacking in general), a pivot is the action or ability to compromise a system, then using the privileges or accessed gained through the attack to focus attention on another target that may not have been visible or exploitable initially. It is the ability to adjust the focus or the target of an intrusion after an initial foothold is gained.

16. D. Privilege escalation is an attack or exploit that grants the attacker greater privileges, permissions, or access than may have been achieved by the initial exploitation.

17. C. Active reconnaissance is the idea of collecting information about a target through interactive means. By directly interacting with a target, the attacker can collect accurate and detailed information quickly but at the expense of potentially being identified as an attacker rather than just an innocent, benign, random visitor.

18. B. A passive test of security controls is being performed when an automated vulnerability scanner is being used that seeks to identify weaknesses while listening in on network communications.

19. C. End-of-life systems are those that are no longer receiving updates and support from their vendor. If an organization continues to use an end-of-life system, the risk of compromise is high because any future exploitation will never be patched or fixed. It is of utmost important to move off end-of-life systems in order to maintain a secure environment.

20. A. Default configurations should never be allowed to remain on a device or within an application. Defaults are such for ease of installation and initial configuration in order to minimize support calls from new customers.

Chapter 2: Technologies and Tools

1. B. Firewalls provide protection by controlling traffic entering and leaving a network; thus, this is an essential foundational security device that should be deployed in any network, large or small.

2. A. Network-based IDSs aren't suitable for detecting email spoofing. Detecting email spoofing is not a feature of an NIDS because email is the payload of network communications and an NIDS mostly focuses on the headers of protocols. Furthermore, even those NIDS that do analyze payloads will often be unable to detect spoofed email elements if those elements are technically valid (such as proper values) and represent real entities (although not the actual author and sender of the message).

3. C. Illegal or unauthorized zone transfers are a significant and direct threat to DNS servers. If a zone transfer is performed against an internal DNS server by an outsider, the result is the leakage of information about every system with an IP address. This is due to the fact that most internal networks use LDAP-based directory services, LDAP is DNS-based, and DHCP auto-registers devices with LDAP and DNS.

4. C. Time to live (TTL) is a value in the IP header used to prevent loops at Layer 3. The TTL value sets the maximum number of routers that an IP packet will traverse before it is discarded if it has not reached its intended destination. Each router will decrement the TTL by 1, then check to see if the result is a non-zero value (to then forward the packet) or a zero value (to discard the packet). If the packet is discarded due to TTL exhaustion, the router will create an ICMP Type 11 Timeout Exceeded message, which is sent to the originator of the discarded communication.

5. C. A rubber duck antenna is an omnidirectional antenna. Cantenna, Yagi, and panel antennas are all examples of directional antennae.

6. B. Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (CCMP) is based on the AES encryption scheme.

7. B. Mobile device management (MDM) is a software solution to the challenging task of managing the myriad mobile devices that employees use to access company resources. The goals of MDM are to improve security, provide monitoring, enable remote management, and support troubleshooting. Not all mobile devices support removable storage, and even fewer support encrypted removable storage. Geotagging is used to mark photos and social network posts, not for BYOD management. Application whitelisting may be an element of BYOD management, but it is only part of a full MDM solution.

8. B. The risk of a lost or stolen notebook is the data loss, not the loss of the system itself. Thus, keeping minimal sensitive data on the system is the only way to reduce the risk. Hard drive encryption, cable locks, and strong passwords, although good ideas, are preventive tools, not means of reducing risk. They don't keep intentional and malicious data compromise from occurring; instead, they encourage honest people to stay honest.

9. B. Whitelisting is a security option that prohibits unauthorized software from being able to execute. Whitelisting is also known as deny by default or implicit deny. Blacklisting, also known as deny by exception or allow by default, is the least successful means of preventing malware execution.

10. A. LDAP operates over TCP ports 636 and 389. POP3 and SMTP operate over TCP ports 110 and 25, respectively. TLS operates over TCP ports 443 and 80 (SSL operates only over TCP port 443; HTTP operates over TCP port 80). FTP operates over TCP ports 20 and 21.

11. B. NAC agents can be dissolvable or permanent. A dissolvable NAC agent is usually written in a web/mobile language, such as Java or ActiveX, and is downloaded and executed to each local machine when the specific management web page is accessed. The dissolvable NAC agent can be set to run once and then terminate or remain resident in memory until the system reboots. A permanent NAC agent is installed on the monitored system as a persistent software background service.

12. D. A media gateway is any device or service that converts data from one communication format to another. A media gateway is often located at the intersection of two different types of networks. Media gateways are commonly used with VoIP systems, where a conversion from IP-based communications to analog or digital is needed.

13. B, D, E. An exploitation framework is a vulnerability scanner that is able to fully exploit the weaknesses it discovers. It can be an automated or manual exploit assessment tool. Often an exploitation framework allows for customization of the test elements as well as the crafting of new tests to deploy against your environment's targets. An exploitation framework does have additional risk compared to that of a vulnerability scanner, since it attempts to fully exploit any discovered weaknesses.

14. B. Banner grabbing is the process of capturing the initial response or welcome message from a network service. A banner grab occurs when a request for data or identity is sent to a service on an open port and that service responds with information that may directly or indirectly reveal its identity.

15. A. Determining effective permissions is accomplished by accumulating the grants or allows of permissions, either through group memberships or to the user account directly, and then removing any denials of permissions.

16. C. A common oversight in content filtering is to fail to escape metacharacters. Be sure that in addition to blocking content that is too long or that matches a known unwanted dataset, your filter escapes metacharacters so that their programmatic power is removed.

17. D. PUPs (potentially unwanted programs) can include any type of questionable software, such as sniffers, password crackers, network mappers, port scanners, and vulnerability scanners. PUPs are distinct from malware, spyware, and adware.

18. A. DEP (data execution prevention) is a memory security feature of many operating systems aimed at blocking a range of memory abuse attacks, including buffer overflows. DEP blocks the execution of code stored in areas of memory designated as data-only areas.

19. B. Geofencing is the designation of a specific geographical area that is then used to implement features on mobile devices. A geofence can be defined by GPS coordinates, IPS (wireless indoor positioning system), or the presence or lack of a specific wireless signal. A device can be configured to enable or disable features based on a geofenced area.

20. D. DNSSEC (Domain Name System Security Extensions) is a security improvement to the existing DNS infrastructure. The primary function of DNSSEC is to provide reliable authentication between devices when performing DNS operations. DNSSEC has been implemented across a significant portion of the DNS system. Each DNS server is issued a digital certificate, which is then used to perform mutual certificate authentication. The goal of DNSSEC is to prevent a range of DNS abuses where false data can be injected into the resolution process.

Chapter 3: Architecture and Design

1. B. A DMZ provides a network segment where publicly accessible servers can be deployed without compromising the security of the private network.

2. A. Having only a single high-speed fiber Internet connection represents the security problem of a single point of failure.

3. C. A security template alone cannot return a system to its precompromised state.

4. B. Fuzzing is a software-testing technique that generates input for targeted programs. The goal of fuzzing is to discover input sets that cause errors, failures, and crashes, or to discover other unknown defects in the targeted program.

5. C. Because an embedded system is in control of a mechanism in the physical world, a security breach could cause harm to people and property. This typically is not true of a standard PC. Power loss, Internet access, and software flaws are security risks of both embedded systems and standard PCs.

6. A. An attack can steal the encryption key from memory, so systems with whole drive encryption that are only screen-locked are vulnerable. Requiring a boot password, locking the system, and powering down ensure the protection of whole drive encryption.

7. B. Control diversity is essential in order to avoid a monolithic security structure. Do not depend on a single form or type of security; instead, integrate a variety of security mechanisms into the layers of defense.

8. D. A demilitarized zone (DMZ) is a special-purpose subnet. A DMZ is an area of a network that is designed specifically for public users to access. If the DMZ (as a whole or as individual systems within the DMZ) is compromised, the private LAN isn't necessarily affected or compromised.

9. C. A sensor is a hardware or software tool used to monitor an activity or event in order to record information or at least take notice of an occurrence.

10. B. Software-defined networking (SDN) is a unique approach to network operation, design, and management. A virtualized network or network virtualization is the combination of hardware and software networking components into a single integrated entity. SDN offers a new network design that is directly programmable from a central location, is flexible, is vendor neutral, and is open standards–based. Another way of thinking about SDN is that it is effectively network virtualization. It allows data transmission paths, communication decision trees, and flow control to be virtualized in the SDN control layer rather than being handled on the hardware on a per-device basis.

11. A. A kiosk OS is either a stand-alone OS or a variation of an NOS. A kiosk OS is designed for end-user use and access. The end user might be an employee of an organization or might be anyone from the general public. A kiosk OS is locked down so that only preauthorized software products and functions are enabled.

12. C. Sandboxing is a means of quarantine or isolation. It's implemented to restrict new or otherwise suspicious software from being able to cause harm to production systems. It can be used against applications or entire OSs.

13. D. An embedded system is a computer implemented as part of a larger system. The embedded system is typically designed around a limited set of specific functions in relation to the larger product of which it's a component. It may consist of the same components found in a typical computer system, or it may be a microcontroller.

14. A. Supervisory control and data acquisition (SCADA) is a type of industrial control system (ICS). An ICS is a form of computer management device that controls industrial processes and machines. SCADA is used across many industries, including manufacturing, fabrication, electricity generation and distribution, water distribution, sewage processing, and oil refining. A SCADA system can operate as a stand-alone device, be networked together with other SCADA systems, or be networked with traditional IT systems.

15. B. Agile is based around adaptive development, where focusing on a working product and fulfilling customer needs is prioritized over rigid adherence to a process, use of specific tools, and detailed documentation. Agile focuses on an adaptive approach to development, supports early delivery, and provides continuous improvement, along with flexible and prompt response to changes.

16. D. DevOps, or development and operations, is a new IT movement where many elements and functions of IT management are being integrated into a single automated solution. DevOps typically consists of IT development, operations, security, and quality assurance.

17. C. VM sprawl occurs when an organization deploys numerous virtual machines without an overarching IT management or security plan in place. Although VMs are easy to create and clone, they have the same licensing and security management requirements as a metal installed OS. Uncontrolled VM creation can quickly lead to a situation where manual oversight is unable to keep up with system demand.

18. B. Platform as a service (PaaS) is the concept of providing a computing platform and software solution stack to a virtual or cloud-based service. Essentially, it involves paying for a service that provides all the aspects of a platform (that is, OS and complete solution package). A PaaS solution grants the customer the ability to run custom code of their choosing without needing to manage the environment.

19. A. Revert to known state is a type of backup or recovery process. Many databases support a known state reversion in order to return to a state of data before edits or changes were implemented. Some systems will automatically create a copy of a known state in order to provide a rollback option, whereas others may require a manual creation of the rollback point.

20. C. Barricades, in addition to fencing (discussed earlier), are used to control both foot traffic and vehicles. K-rails (often seen during road construction), large planters, zigzag queues, bollards, and tire shredders are all examples of barricades. When used properly, they can control crowds and prevent vehicles from being used to cause damage to your building.

Chapter 4: Identity and Access Management

1. C. Role-based access control (RBAC) is best suited for environments with a high rate of employee turnover, because access is defined against static job descriptions rather than transitive user accounts (DAC and ACL) or assigned clearances (MAC).

2. C. SAML is an open standard data format based on XML for the purpose of supporting the exchange of authentication and authorization details between systems, services, and devices. A biometric is an authentication factor, not a means of exchanging authentication information. Two-factor authentication is the use of two authentication factors. LDAP is a protocol used by directory services and is not directly related to authentication.

3. B. A one-time password is always the strongest form of password. A static password is always the weakest form of password. Passwords with more than eight characters and those that use different types of keyboard characters are usually strong, but these factors alone are unable to indicate their strength.

4. B. RADIUS is a centralized authentication solution that adds an additional layer of security between a network and remote clients. SMTP is the email-forwarding protocol used on the Internet and intranets. PGP is a security solution for email. VLANs are created by switches to logically divide a network into subnets.

5. C. Single sign-on doesn't address access control and therefore doesn't provide granular or nongranular access control. Single sign-on provides the benefits of the ability to browse multiple systems, fewer credentials to memorize, and the use of stronger passwords.

6. C. Federation or federated identity is a means of linking a subject's accounts from several sites, services, or entities in a single account. Thus it is a means to accomplish single sign-on. Accountability logging is used to relate digital activities to humans. ACL verification is a means to verify that correct permissions are assigned to subjects. Trusted OS hardening is the removal of unneeded components and securing the remaining elements.

7. C. Since the open-source Linux system likely has a default root password, changing the default password to something unique will have the effect of preventing unauthorized entities from making system changes. Passwords are part of the authentication system. Authorization is access control or the ability to interact with resource objects.

8. B. Time-of-day restrictions are used to limit or restrict what time of day, and often what day of the week, a specific user account can log on to the network or a specific system can be accessed by users.

9. A. The proper order of the steps of Kerberos's third-party trusted authentication process starts with the subject providing logon credentials and ends with the network server verifying the ST.

10. C. TACACS+ is a Cisco proprietary AAA service that is available only when using Cisco hardware.

11. D. RADIUS is an AAA server that can be used as an additional security barrier between external connections and the private network. A remote access–focused AAA service protects the internal domain controllers from abuse caused by remote connections.

12. A. OpenID Connect is an Internet-based single sign-on solution. It operates over the OAuth protocol (OAuth is an open standard for authentication and access delegation [federation]) and can be used in relation to web services as well as smart-device apps. The purpose or goal of OpenID Connect is to simplify the process by which applications are able to identify and verify users. Shibboleth is optimized for websites, not for devices and apps.

13. B. Effective permissions are calculated by accumulating all allows or grants of access to a resource, and then subtracting or removing any denials to that resource.

14. D. Attribute-based access control (ABAC) is a mechanism for assigning access and privileges to resources through a scheme of attributes or characteristics. The attributes can be related to the user, the object, the system, the application, the network, the service, the time of day, or even other subjective environmental concerns.

15. C. When an organization decides to implement a biometric factor, it is important to evaluate the available options in order to select a biometric solution that is most in line with the organization's security priorities. One method to accomplish this is to consult a Zephyr analysis chart. This type of chart presents the relative strengths and weaknesses of various characteristics of biometric factor options.

16. B. FRR (false rejection rate) errors increase with sensitivity, whereas FAR (false acceptance rate) errors decrease with an increase in sensitivity.

17. A. A service account is a user account that is used to control the access and capabilities of an application. Through the use of a service account, an application can be granted specific authorization related to its function and data access needs.

18. C. Failing to regularly audit permissions can result in users gaining more access over time that is not required by their current work responsibilities. This situation is a violation of the principle of least privilege.

19. D. Location-based access control is a means of authorization that grants or denies resource access based on where the subject is located. This might be based on whether the network connection is local wired, local wireless, or remote. Location-based policies can also grant or deny access based on MAC address, IP address, OS version, patch level, and/or subnet in addition to logical or geographical location.

20. A. Password complexity sets the rules regarding password content, which should be a minimum of eight characters (although 12–16 would be much better) and include representations of at least three of the four character types (uppercase, lowercase, numbers, and symbols).

Chapter 5: Risk Management

1. D. The annualized loss expectancy (ALE) represents the total potential loss a company may experience within a single year due to a specific risk to an asset. EF is the percentage of asset value loss that would occur if a risk was realized. SLE is the potential dollar value loss from a single risk-realization incident. ARO is the statistical probability that a specific risk may be realized a certain number of times in a year.

2. D. A memorandum of understanding (MOU) is an expression of agreement or aligned intent, will, or purpose between two entities. An MOU is not typically a legal agreement or commitment, but rather a more formal form of a reciprocal agreement or gentleman's handshake (neither of which is typically written down). An SLA is a formal control. BIA is business impact assessment. DLP is data loss prevention.

3. A. When a user signs an acceptable use policy, it's a form of consent to the monitoring and auditing processes used by the organization. A privacy policy usually explains that there is no privacy on company systems. A separation of duties policy indicates that administrative functions are divided among several people. The code of ethics policy describes decision-making processes to use when faced with ethical dilemmas.

4. D. Business continuity is used when business processes are threatened. Security policy is used when new software is distributed. Disaster recovery is used when business processes are interrupted. Incident response is used when a user steals company data.

5. B. The proper procedure is to restore the full backup and then the last differential backup. The other three options are incorrect or incomplete.

6. D. A security guard is not an administrative control. A security guard can be considered a preventive, detective, and/or corrective control.

7. D. Mandatory vacations are a form of user peer auditing. The process works by requiring each employee to be on vacation (or just away from the office and without remote access) for a minimal amount of time each year (typically one to two weeks). While the employee is away, another worker sits at their desk and performs their work tasks using the original employee's privileged account. This process is used to detect fraud, abuse, or incompetence. The technique is often employed in financial environments or where high-value assets are managed.

8. A. Job rotation, cross-training, or rotation of duties is a counterbalance to the application of separation of duties. If all high-level tasks are performed by individual administrators, what happens if one person leaves the organization? If no one else has the knowledge or skill to perform the tasks, the organization suffers. Job rotation is the periodic shifting of assigned work tasks or job descriptions among a small collection of workers, sometimes known as a rotation group.

9. B. Aging hardware should be scheduled for replacement and/or repair. The schedule for such operations should be based on the mean time to failure (MTTF), mean time between failures (MTBF), and mean time to repair/restore (MTTR) estimates established for each device or on prevailing best organizational practices for managing the hardware life cycle. MTTF is the expected typical functional lifetime of the device, given a specific operating environment. MTBF is the expected typical time frame between failures, such as between the first failure and the second failure. If the MTTF and MTBF are the same values (or nearly so), some manufacturers only list the MTBF rating and use it to address both concepts. MTTR is the average length of time required to perform a repair on the device. A device can often undergo numerous repairs before a catastrophic failure is expected. Be sure to schedule all devices to be replaced before their MTTFs expire.

10. D. A single point of failure is any individual or sole device, connection, or pathway that is of moderate to mission-critical importance to the organization. If that one item fails, the whole organization suffers loss. Infrastructures should be designed with redundancies of all moderately or highly important elements in order to avoid single points of failure. Removing single points of failure involves adding redundancy, recovery options, or alternative means to perform business tasks and processes. Avoiding or resolving single points of failure will improve stability, uptime, and availability.

11. C. Qualitative risk analysis is more scenario, based than calculator, based. Rather than assign exact dollar figures to possible losses, you rank threats on a scale to evaluate their risks, costs, and effects. The process of performing qualitative risk analysis involves judgment, intuition, and experience. You can use many techniques to perform qualitative risk analysis, including brainstorming, the Delphi technique, storyboarding, focus groups, surveys, checklists, questionnaires, one-on-one meetings, and interviews.

12. B. Accepting risk, or tolerating risk, is the valuation by management of the cost-benefit analysis of possible safeguards and the determination that the cost of the countermeasure greatly outweighs the possible cost of loss due to a risk. It also means management has agreed to accept the consequences and the loss if the risk is realized. In most cases, accepting risk requires a clearly written statement that indicates why a safeguard was not implemented, who is responsible for the decision, and who will be responsible for the loss if the risk is realized, usually in the form of a “sign-off” letter. An organization’s decision to accept risk is based on its risk tolerance.

13. A. An incident response plan (IRP) consists of Preparation, Identification, Containment, Eradication, Recovery, and Lessons Learned.

14. C. Recovery is the process of removing any damaged elements from the environment and replacing them. This can apply to corrupted data being restored from backup and to malfunctioning hardware or software being replaced with updated or new versions. In some cases, entire computer systems need to be reconstituted (rebuilt from new parts) in order to eradicate all elements of compromise and return into production a functioning and trustworthy system. The recovery and reconstitution procedures can also include alterations of configuration settings and adding new security features or components. This is especially important if a vulnerability remains that could be exploited to cause the incident to reoccur. The environment is returned to normal operations by the end of the recovery phase.

15. D. A legal hold is an early step in the evidence collection or e-discovery process. It is a legal notice to a data custodian that that specific data or information must be preserved and that good-faith efforts must be engaged to preserve the indicated evidence. The custodian must maintain and preserve the data until they are notified that the obligation is no longer necessary.

16. A, B, D. Forensic preservation aims at preventing any change from occurring as related to collected evidence. These efforts include removing relevant storage devices from their systems, using write-blocking adapters to block any writing signals from being received by storage devices, using hash calculations before and after every operation, and only analyzing cloned copies of storage devices and never the original device. If the original data is corrupted or changed, then it usually becomes inadmissible in court. Thus, forensic experts take extreme caution when working with the original source drives.

17. B. The goal of BCP (business continuity planning) planners is to implement a combination of policies, procedures, and processes such that a potentially disruptive event has as little impact on the business as possible.

18. D. A hot site is a real-time, moment-to-moment mirror image of the original site. It contains a complete network environment that is fully installed and configured with live current business data. The moment the original site becomes inoperable due to a disaster, the hot site can be used to continue business operations without a moment of downtime. Hot sites are the most expensive, but they offer the least amount of downtime. Thus, while being a reliable means of recovery, they are not cost effective.

19. C. A corrective access control modifies the environment to return systems to normal after an unwanted or unauthorized activity has occurred. A preventive access control is deployed to thwart or stop unwanted or unauthorized activity from occurring. A detective access control is deployed to discover or detect unwanted or unauthorized activity. A compensation access control is deployed to provide various options to other existing controls to aid in enforcement and support of security policies.

20. A, B, C, D, E, F, G, H. Protected Health Information (PHI), according to the laws of the United States, is any data that relates to the health status, use of health care, payment for health care, and other information collected about an individual in relation to their health. HIPAA defines PHI in relation to 18 types of information that must be handled securely to protect against disclosure and misuse. These 18 elements are names, all geographic identifiers smaller than a state (so address, city, and zip are protected), dates directly related to an individual, other than year, phone numbers, fax numbers, email addresses, social security numbers, medical record numbers, health insurance beneficiary numbers, account numbers, certificate/license numbers, vehicle identifiers and serial numbers, device identifiers and serial numbers, Web URLs, IP address numbers, biometric identifiers, photographic images, and any other unique identifying number, characteristic, or code except the unique code assigned by the collecting entity to code the data.

Chapter 6: Cryptography and PKI

1. B. Elliptic Curve Diffie-Hellman Ephemeral, or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE), implements perfect forward secrecy through the use of elliptic curve cryptography (ECC). PBKDF2 is an example of a key-stretching technology not directly supporting perfect forward secrecy. HMAC is a hashing function. OCSP is used to check for certificate revocation.

2. C. Triple DES (3DES) offers the strength of 168-bit keys. The Data Encryption Standard (DES) offers the strength of 56-bit keys. The Advanced Encryption Standard (AES) offers the strength of 128-, 192-, or 256-bit keys. The International Data Encryption Algorithm (IDEA) offers the strength of 128-bit keys.

3. D. The security service that protects the secrecy of data, information, or resources is known as confidentiality. Integrity protects the reliability and correctness of data. Authentication verifies the identity of the sender or receiver of a message. Nonrepudiation prevents the sender of a message or the perpetrator of an activity from being able to deny that they sent the message or performed the activity.

4. C. Key escrow isn't used in digital signatures, but it's a fault-tolerance feature of certificate and key management. Asymmetric and symmetric cryptography, along with hashing, are used in digital signatures.

5. A, C. Proof of identity and the subject's public key must be provided to the CA when the subject requests a certificate. The private key should never be revealed to anyone, not even the CA. A hardware storage device is used after a key or certificate has been issued, not as part of the requesting process.

6. B. Centralized key management is more secure, or at least more desirable, from a private corporate perspective. From a public or individual perspective, decentralized key management is more secure. Individual and distributed key management are nonstandard terms that could be used to refer to decentralized key management.

7. D. Keys and certificates should be renewed just before they expire. All the other choices are incorrect.

8. A. CBC (Cipher Block Chaining) mode is used to prevent the creation of duplicate ciphertext blocks. This is accomplished by adding an IV into the operation of encryption. The IV is integrated with the first block using XOR. The result is then encrypted using the selected secret key. The cipher text of the first block is then used as the IV for the second block. This linking or chaining of the blocks for use as an IV ensures that every block results in cipher text that is unique.

9. B. Out-of-band key exchange takes place outside of the current communication channel or pathway, such as through a secondary channel, via a special secured exchange technique in the channel, or with a complete separate pathway technology. Out-of-band key exchange is generally considered more secure, because any attack monitoring the initial channel is less likely to be monitoring or have access to the alternate or separate communications path.

10. D. Digital signatures using asymmetric encryption solutions (specifically, public key cryptography where a key pair of a public key and a private key is used) operate as follows:

    1. The sender writes a message.
    2. The sender computes a hash of the message.
    3. The sender uses the sender's private key to encrypt the hash.
    4. The sender attaches the encrypted hash to the message.
    5. The complete message package is sent to the receiver.
    6. The receiver strips off the encrypted hash (the digital signature).
    7. The receiver uses the sender's public key to decrypt the sender's private key and thus extract the hash from the digital signature.
    8. The receiver computes a hash of the message.
    9. The receiver compares the two hash values.

11. C. Key stretching is a collection of techniques that can take a weak key or password and stretch it to make it more secure, at least against brute-force attacks.

12. B. Security through obscurity is the concept of attempting to gain security by hiding or not being noticed among the crowd of other targets. This is effectively security hide-and-seek. It is not considered a valid security approach for any organization.

13. C. Elliptic Curve Diffie-Hellman Ephemeral, or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE), implements perfect forward secrecy through the use of elliptic curve cryptography (ECC). ECC has the potential to provide greater security with less computational burden than that of DHE. Digital envelopes allow one side of a communication to select the key and provide it to the other side; this can result in less secure keys than if both sides participate in the key generation process. Static keys and sequential keys are insecure.

14. D. XOR (eXclusive OR) is an exclusive disjunction, which means that it produces an output of truth (or 1) whenever the two inputs differ (such as one is a 0 [false] and the other is a 1 [true]). It's referred to in mathematical literature as the XOR function and is commonly represented by the ⊕ symbol. When the two values being XORed are both 0 or both 1, the result is 0. When the values are different, the result is 1.

15. B. ROT13 (rotation 13) is a substitution cipher based on the 26 letters of the English alphabet (or the basic Latin alphabet). The operation of ROT13 is to shift or substitute each original plain text letter with the letter located in 13 positions further down the alphabet. The ROT13 cipher text of "frphevgl ebpxf" decrypts into the plain text of "security rocks."

16. C. WPA and WPA-2 support two forms of authentication: PSK and ENT. PSK, or preshared key, is also known as personal. PSK is the use of a static fixed password for authentication. ENT, or enterprise, is also known as IEEE 802.1x/EAP. ENT enables the leveraging of an existing AAA service, such as RADIUS or TACACS+, to be used to authenticate. WEP only supports fixed key authentication. WPS (WiFi Protected Setup) adds a new client to a wireless network, but other than pressing the WPS button or sending the PIN to the WAP, no authentication is taking place.

17. D. EAP-TLS (EAP Transport Layer Security) is an open IETF standard which is an implementation of the TLS protocol for use in protecting authentication traffic. EAP-TLS is considered one of the strongest EAP standards available. EAP-TLS is most effective when both client and server (wireless endpoint device and wireless base station) have digital certificates.

18. B. A captive portal is an authentication technique that redirects a newly connected wireless web client to a portal access-control page. The portal page may require the user to input payment information, provide logon credentials, or input an access code. Providing patrons with the access code once a drink order is placed allows the business to limit access to actual customers.

19. A. Pinning, or HTTP Public Key Pinning (HPKP), is a security mechanism operating over HTTP that enables an HTTPS (TLS secured web service) system to prevent impersonation by attackers through the use of fraudulently issued digital certificates. Pinning operates by providing the visitor with an HTTP response header field value, named Public-Key-Pins, which includes the hashes of the certificates used by the server along with a time stamp for how long to keep these certificates pinned.

20. B. SAN (subject alternative name) certificates support a range of names for a single entity, such as hostname, site name, IP address, and common name. A SAN certificate is used to provide authentication to multiple names, but only those that are specifically defined.

Index

Note to the Reader: Throughout this index **boldfaced** page numbers indicate primary discussions of a topic. *Italicized* page numbers indicate illustrations.