NIST
National Institute of Standards and Technology
U.S. Department of Commerce

Special Publication 800-115

Technical Guide to Information Security Testing and Assessment

Recommendations of the National Institute of Standards and Technology

Karen Scarfone
Murugiah Souppaya
Amanda Cody
Angela Orebaugh

NIST Special Publication 800-115

Technical Guide to Information Security Testing and Assessment

Recommendations of the National Institute of Standards and Technology

Karen Scarfone
Murugiah Souppaya
Amanda Cody
Angela Orebaugh

C O M P U T E R S E C U R I T Y

Computer Security Division
Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8930

September 2008

U.S. Department of Commerce
Carlos M. Gutierrez, Secretary

National Institute of Standards and Technology
Dr. Patrick D. Gallagher, Deputy Director

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analysis to advance the development and productive use of information technology (IT). ITL's responsibilities include the development of technical, physical, administrative, and management standards and guidelines for the cost-effective security and privacy of sensitive unclassified information in Federal computer systems. This Special Publication 800-series reports on ITL's research, guidance, and outreach efforts in computer security and its collaborative activities with industry, government, and academic organizations.

National Institute of Standards and Technology Special Publication 800-115
Natl. Inst. Stand. Technol. Spec. Publ. 800-115, 80 pages (Sep. 2008)

Acknowledgements

The authors, Karen Scarfone and Murugiah Souppaya of the National Institute of Standards and Technology (NIST) and Amanda Cody and Angela Orebaugh of Booz Allen Hamilton, wish to thank their colleagues who reviewed drafts of this document and contributed to its technical content. The authors would like to acknowledge John Connor, Tim Grance, Blair Heiserman, Arnold Johnson, Richard Kissel, Ron Ross, Matt Scholl, and Pat Toth of NIST and Steve Allison, Derrick Dicoi, Daniel Owens, Victoria Thompson, Selena Tonti, Theodore Winograd, and Gregg Zepp of Booz Allen Hamilton for their keen and insightful assistance throughout the development of the document. The authors appreciate all the feedback provided during the public comment period, especially by Marshall Abrams, Karen Quigg, and others from MITRE Corporation; William Mills of SphereCom Enterprises; and representatives from the Financial Management Service (Department of the Treasury) and the Department of Health and Human Services (HHS).

Trademark Information

Table of Contents

List of Appendices

List of Tables

List of Figures

Executive Summary

An information security *assessment* is the process of determining how effectively an entity being assessed (e.g., host, system, network, procedure, person—known as the assessment *object*) meets specific security objectives. Three types of assessment methods can be used to accomplish this—testing, examination, and interviewing. *Testing* is the process of exercising one or more assessment objects under specified conditions to compare actual and expected behaviors. *Examination* is the process of checking, inspecting, reviewing, observing, studying, or analyzing one or more assessment objects to facilitate understanding, achieve clarification, or obtain evidence. *Interviewing* is the process of conducting discussions with individuals or groups within an organization to facilitate understanding, achieve clarification, or identify the location of evidence. Assessment results are used to support the determination of security control effectiveness over time.

This document is a guide to the basic technical aspects of conducting information security assessments. It presents technical testing and examination methods and techniques that an organization might use as part of an assessment, and offers insights to assessors on their execution and the potential impact they may have on systems and networks. For an assessment to be successful and have a positive impact on the security posture of a system (and ultimately the entire organization), elements beyond the execution of testing and examination must support the technical process. Suggestions for these activities—including a robust planning process, root cause analysis, and tailored reporting—are also presented in this guide.

The processes and technical guidance presented in this document enable organizations to:

- Develop information security assessment policy, methodology, and individual roles and responsibilities related to the technical aspects of assessment
- Accurately plan for a technical information security assessment by providing guidance on determining which systems to assess and the approach for assessment, addressing logistical considerations, developing an assessment plan, and ensuring legal and policy considerations are addressed
- Safely and effectively execute a technical information security assessment using the presented methods and techniques, and respond to any incidents that may occur during the assessment
- Appropriately handle technical data (collection, storage, transmission, and destruction) throughout the assessment process
- Conduct analysis and reporting to translate technical findings into risk mitigation actions that will improve the organization's security posture.

The information presented in this publication is intended to be used for a variety of assessment purposes. For example, some assessments focus on verifying that a particular security control (or controls) meets requirements, while others are intended to identify, validate, and assess a system's exploitable security weaknesses. Assessments are also performed to increase an organization's ability to maintain a proactive computer network defense. Assessments are not meant to take the place of implementing security controls and maintaining system security.

To accomplish technical security assessments and ensure that technical security testing and examinations provide maximum value, NIST recommends that organizations:

- **Establish an information security assessment policy**. This identifies the organization's requirements for executing assessments, and provides accountability for the appropriate

individuals to ensure assessments are conducted in accordance with these requirements. Topics that an assessment policy should address include the organizational requirements with which assessments must comply, roles and responsibilities, adherence to an established assessment methodology, assessment frequency, and documentation requirements.

- **Implement a repeatable and documented assessment methodology**. This provides consistency and structure to assessments, expedites the transition of new assessment staff, and addresses resource constraints associated with assessments. Using such a methodology enables organizations to maximize the value of assessments while minimizing possible risks introduced by certain technical assessment techniques. These risks can range from not gathering sufficient information on the organization's security posture for fear of impacting system functionality to affecting the system or network availability by executing techniques without the proper safeguards in place. Processes that minimize risk caused by certain assessment techniques include using skilled assessors, developing comprehensive assessment plans, logging assessor activities, performing testing off-hours, and conducting tests on duplicates of production systems (e.g., development systems). Organizations need to determine the level of risk they are willing to accept for each assessment, and tailor their approaches accordingly.

- **Determine the objectives of each security assessment, and tailor the approach accordingly**. Security assessments have specific objectives, acceptable levels of risk, and available resources. Because no individual technique provides a comprehensive picture of an organization's security when executed alone, organizations should use a combination of techniques. This also helps organizations to limit risk and resource usage.

- **Analyze findings, and develop risk mitigation techniques to address weaknesses**. To ensure that security assessments provide their ultimate value, organizations should conduct root cause analysis upon completion of an assessment to enable the translation of findings into actionable mitigation techniques. These results may indicate that organizations should address not only technical weaknesses, but weaknesses in organizational processes and procedures as well.

1. Introduction

1.1 Authority

The National Institute of Standards and Technology (NIST) developed this document in furtherance of its statutory responsibilities under the Federal Information Security Management Act (FISMA) of 2002, Public Law 107-347.

NIST is responsible for developing standards and guidelines, including minimum requirements, for providing adequate information security for all agency operations and assets; but such standards and guidelines shall not apply to national security systems. This guideline is consistent with the requirements of the Office of Management and Budget (OMB) Circular A-130, Section 8b (3), "Securing Agency Information Systems," as analyzed in A-130, Appendix IV: Analysis of Key Sections. Supplemental information is provided in A-130, Appendix III.

This guideline has been prepared for use by federal agencies. It may be used by nongovernmental organizations on a voluntary basis and is not subject to copyright, though attribution is desired.

Nothing in this document should be taken to contradict standards and guidelines made mandatory and binding on federal agencies by the Secretary of Commerce under statutory authority; nor should these guidelines be interpreted as altering or superseding the existing authorities of the Secretary of Commerce, Director of the OMB, or any other federal official.

1.2 Purpose and Scope

The purpose of this document is to provide guidelines for organizations on planning and conducting technical information security testing and assessments, analyzing findings, and developing mitigation strategies. It provides practical recommendations for designing, implementing, and maintaining technical information relating to security testing and assessment processes and procedures, which can be used for several purposes—such as finding vulnerabilities in a system or network and verifying compliance with a policy or other requirements. This guide is not intended to present a comprehensive information security testing or assessment program, but rather an overview of the key elements of technical security testing and assessment with emphasis on specific techniques, their benefits and limitations, and recommendations for their use.

This document replaces NIST Special Publication 800-42, *Guideline on Network Security Testing*.

1.3 Audience

This guide is intended for use by computer security staff and program managers, system and network administrators, and other technical staff who are responsible for the technical aspects of preparing, operating, and securing systems and network infrastructures. Managers can also use the information presented to facilitate the technical decision-making processes associated with security testing and assessments. Material in this document is technically oriented, and assumes that readers have at least a basic understanding of system and network security.

1.4 Document Structure

The remainder of this document is organized into seven major sections:

- Section 2 presents an overview of information security assessments, including policies, roles and responsibilities, methodologies, and techniques.
- Section 3 provides a detailed description of several technical examination techniques, including documentation review, log review, network sniffing, and file integrity checking.
- Section 4 describes several techniques for identifying targets and analyzing them for potential vulnerabilities. Examples of these techniques include network discovery and vulnerability scanning.
- Section 5 explains techniques commonly used to validate the existence of vulnerabilities, such as password cracking and penetration testing.
- Section 6 presents an approach and process for planning a security assessment.
- Section 7 discusses factors that are key to the execution of security assessments, including coordination, the assessment itself, analysis, and data handling.
- Section 8 presents an approach for reporting assessment findings, and provides an overview of remediation activities.

This guide also contains the following appendices:

- Appendix A describes two live operating system (OS) CD distributions that allow the user to boot a computer to a CD containing a fully operational OS and testing tools.
- Appendix B provides a template for creating Rules of Engagement (ROE).
- Appendix C briefly discusses application security assessment.
- Appendix D contains recommendations for performing remote access testing.
- Appendix E offers a list of resources that may facilitate the security assessment process.
- Appendix F features a glossary of terms used throughout this document.
- Appendix G provides a list of acronyms and abbreviations.

2. Security Testing and Examination Overview

An information security *assessment* is the process of determining how effectively an entity being assessed (e.g., host, system, network, procedure, person—known as the assessment *object*) meets specific security objectives. Three types of assessment methods can be used to accomplish this—testing, examination, and interviewing. *Testing* is the process of exercising one or more assessment objects under specified conditions to compare actual and expected behaviors. *Examination* is the process of checking, inspecting, reviewing, observing, studying, or analyzing one or more assessment objects to facilitate understanding, achieve clarification, or obtain evidence. *Interviewing* is the process of conducting discussions with individuals or groups within an organization to facilitate understanding, achieve clarification, or identify the location of evidence. Assessment results are used to support the determination of security control effectiveness over time.

This publication addresses technical testing and examination techniques that can be used to identify, validate, and assess technical vulnerabilities and assist organizations in understanding and improving the security posture of their systems and networks. Security testing and examination is required by FISMA[1] and other regulations. It is not meant to take the place of implementing security controls and maintaining system security, but to help organizations confirm that their systems are properly secured and identify any organization security requirements that are not met as well as other security weaknesses that should be addressed.

This section provides an overview of information security assessment methodologies and technical testing and examination techniques.

2.1 Information Security Assessment Methodology

A repeatable and documented security assessment methodology is beneficial in that it can:

- Provide consistency and structure to security testing, which can minimize testing risks
- Expedite the transition of new assessment staff
- Address resource constraints associated with security assessments.

Because information security assessment requires resources such as time, staff, hardware, and software, resource availability is often a limiting factor in the type and frequency of security assessments. Evaluating the types of security tests and examinations the organization will execute, developing an appropriate methodology, identifying the resources required, and structuring the assessment process to support expected requirements can mitigate the resource challenge. This gives the organization the ability to reuse pre-established resources such as trained staff and standardized testing platforms; decreases time required to conduct the assessment and the need to purchase testing equipment and software; and reduces overall assessment costs.

A phased information security assessment methodology offers a number of advantages. The structure is easy to follow, and provides natural breaking points for staff transition. Its methodology should contain at minimum the following phases:

[1] Section 3544 requires the "periodic testing and evaluation of the effectiveness of information security policies, procedures, and practices, to be performed with a frequency depending on risk, but no less than annually." FISMA is available at http://csrc.nist.gov/drivers/documents/FISMA-final.pdf.

- **Planning**. Critical to a successful security assessment, the planning phase is used to gather information needed for assessment execution—such as the assets to be assessed, the threats of interest against the assets, and the security controls to be used to mitigate those threats—and to develop the assessment approach. A security assessment should be treated as any other project, with a project management plan to address goals and objectives, scope, requirements, team roles and responsibilities, limitations, success factors, assumptions, resources, timeline, and deliverables. Section 6 of this guide covers planning.
- **Execution**. Primary goals for the execution phase are to identify vulnerabilities and validate them when appropriate. This phase should address activities associated with the intended assessment method and technique. Although specific activities for this phase differ by assessment type, upon completion of this phase assessors will have identified system, network, and organizational process vulnerabilities. This phase is discussed in more depth in Section 7.
- **Post-Execution**. The post-execution phase focuses on analyzing identified vulnerabilities to determine root causes, establish mitigation recommendations, and develop a final report. Section 8 of this guide addresses reporting and mitigation.

Several accepted methodologies exist for conducting different types of information security assessments. References to several of these methodologies are found in Appendix E.[2] For example, NIST has created a methodology—documented in Special Publication (SP) 800-53A, *Guide for Assessing the Security Controls in Federal Information Systems*—which offers suggestions for assessing the effectiveness of the security controls outlined in NIST SP 800-53.[3] Another widely used assessment methodology is the Open Source Security Testing Methodology Manual (OSSTMM).[4] Because there are numerous reasons to conduct assessments, an organization may want to use multiple methodologies. This publication offers recommendations for technical testing and examination techniques that can be used for many assessment methodologies and leveraged for many assessment purposes.

2.2 Technical Assessment Techniques

Dozens of technical security testing and examination techniques exist that can be used to assess the security posture of systems and networks. The most commonly used techniques from the standpoint of this document will be discussed in more depth later in this guide, and are grouped into the following three categories:

- **Review Techniques**. These are examination techniques used to evaluate systems, applications, networks, policies, and procedures to discover vulnerabilities, and are generally conducted manually. They include documentation, log, ruleset, and system configuration review; network sniffing; and file integrity checking. Section 3 provides additional information on review techniques.
- **Target Identification and Analysis Techniques**. These testing techniques can identify systems, ports, services, and potential vulnerabilities, and may be performed manually but are generally performed using automated tools. They include network discovery, network port and service

[2] NIST does not endorse one methodology over another; the intent is to provide organizations with options that will allow them to make informed decisions to adopt an existing methodology or combine several to develop a unique methodology that suits the organization.

[3] NIST SP 800-53A discusses the framework for development of assessment procedures, describes the process of assessing security controls, and offers assessment procedures for each control. NIST SP 800-53A was developed to be used in conjunction with NIST SP 800-37, *Guide for the Security Certification and Accreditation of Federal Information Systems*. NIST SPs 800-53, 800-53A, and 800-37 are available at http://csrc.nist.gov/publications/PubsSPs.html.

[4] More information on OSSTMM is available at http://www.isecom.org/osstmm/.

identification, vulnerability scanning, wireless scanning, and application security examination. Further discussion of these techniques is presented in Section 4.

- **Target Vulnerability Validation Techniques**. These testing techniques corroborate the existence of vulnerabilities, and may be performed manually or by using automatic tools, depending on the specific technique used and the skill of the test team. Target vulnerability validation techniques include password cracking, penetration testing, social engineering, and application security testing. More information on these techniques is found in Section 5.

Since no one technique can provide a complete picture of the security of a system or network, organizations should combine appropriate techniques to ensure robust security assessments. For example, penetration testing usually relies on performing both network port/service identification and vulnerability scanning to identify hosts and services that may be targets for future penetration. Also, multiple technical ways exist to meet an assessment requirement, such as determining whether patches have been applied properly. This publication focuses on explaining *how* these different technical techniques can be performed, and does not specify *which* techniques should be used for which circumstances—thus providing organizations with the flexibility to choose the techniques that best meet their requirements.

In addition to the technical techniques described in this publication, there are many non-technical techniques that may be used in addition to or instead of the technical techniques. One example is physical security testing, which confirms the existence of physical security vulnerabilities by attempting to circumvent locks, badge readers, and other physical security controls, typically to gain unauthorized access to specific hosts. Another example of a non-technical technique is manual asset identification. An organization may choose to identify assets to be assessed through asset inventories, physical walkthroughs of facilities, and other non-technical means, instead of relying on technical techniques for asset identification. Details on non-technical techniques are outside the scope of this publication, but it is important to recognize the value of non-technical techniques and to consider when they may be more appropriate to use than their technical counterparts.

2.3 Comparing Tests and Examinations

Examinations primarily involve the review of documents such as policies, procedures, security plans, security requirements, standard operating procedures, architecture diagrams, engineering documentation, asset inventories, system configurations, rulesets, and system logs. They are conducted to determine whether a system is properly documented, and to gain insight on aspects of security that are only available through documentation. This documentation identifies the intended design, installation, configuration, operation, and maintenance of the systems and network, and its review and cross-referencing ensures conformance and consistency. For example, an environment's security requirements should drive documentation such as system security plans and standard operating procedures—so assessors should ensure that all plans, procedures, architectures, and configurations are compliant with stated security requirements and applicable policies. Another example is reviewing a firewall's ruleset to ensure its compliance with the organization's security policies regarding Internet usage, such as the use of instant messaging, peer-to-peer (P2P) file sharing, and other prohibited activities.

Examinations typically have no impact on the actual systems or networks in the target environment aside from accessing necessary documentation, logs, or rulesets.[5] However, if system configuration files or logs are to be retrieved from a given system such as a router or firewall, only system administrators and

[5] One passive testing technique that can potentially impact networks is network sniffing, which involves connecting a sniffer to a hub, tap, or span port on the network. In some cases, the connection process requires reconfiguring a network device, which could disrupt operations.

similarly trained individuals should undertake this work to ensure that settings are not inadvertently modified or deleted.

Testing involves hands-on work with systems and networks to identify security vulnerabilities, and can be executed across an entire enterprise or on selected systems. The use of scanning and penetration techniques can provide valuable information on potential vulnerabilities and predict the likelihood that an adversary or intruder will be able to exploit them. Testing also allows organizations to measure levels of compliance in areas such as patch management, password policy, and configuration management.

Although testing can provide a more accurate picture of an organization's security posture than what is gained through examinations, it is more intrusive and can impact systems or networks in the target environment. The level of potential impact depends on the specific types of testing techniques used, which can interact with the target systems and networks in various ways—such as sending normal network packets to determine open and closed ports, or sending specially crafted packets to test for vulnerabilities. Any time that a test or tester directly interacts with a system or network, the potential exists for unexpected system halts and other denial of service conditions. Organizations should determine their acceptable levels of intrusiveness when deciding which techniques to use. Excluding tests known to create denial of service conditions and other disruptions can help reduce these negative impacts.

Testing does not provide a comprehensive evaluation of the security posture of an organization, and often has a narrow scope because of resource limitations—particularly in the area of time. Malicious attackers, on the other hand, can take whatever time they need to exploit and penetrate a system or network. Also, while organizations tend to avoid using testing techniques that impact systems or networks, attackers are not bound by this constraint and use whatever techniques they feel necessary. As a result, testing is less likely than examinations to identify weaknesses related to security policy and configuration. In many cases, combining testing and examination techniques can provide a more accurate view of security.

2.4 Testing Viewpoints

Tests can be performed from a number of viewpoints—for example, how easily could an external attacker or malicious insider successfully attack a system? Section 2.4.1 of this guide compares testing performed from external and internal viewpoints. Section 2.4.2 discusses another aspect of viewpoints—namely, the previous knowledge that assessors have of the target or target environment.

2.4.1 External and Internal

External security testing is conducted from outside the organization's security perimeter. This offers the ability to view the environment's security posture as it appears outside the security perimeter—usually as seen from the Internet—with the goal of revealing vulnerabilities that could be exploited by an external attacker.

External testing often begins with reconnaissance techniques that search public registration data, Domain Name System (DNS) server information, newsgroup postings, and other publicly available information to collect information (e.g., system names, Internet Protocol [IP] addresses, operating systems, technical points of contact) that may help the assessor to identify vulnerabilities. Next, enumeration begins by using network discovery and scanning techniques to determine external hosts and listening services. Since perimeter defenses such as firewalls, routers, and access control lists often limit the types of traffic allowed into the internal network, assessors often use techniques that evade these defenses—just as external attackers would. Depending on the protocols allowed through, initial attacks are generally focused on commonly used and allowed application protocols such as File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), Simple Mail Transfer Protocol (SMTP), and Post Office Protocol

(POP). Servers that are externally accessible are tested for vulnerabilities that might allow access to internal servers and private information. External security testing also concentrates on discovering access method vulnerabilities, such as wireless access points, modems, and portals to internal servers.

For internal security testing, assessors work from the internal network and assume the identity of a trusted insider or an attacker who has penetrated the perimeter defenses. This kind of testing can reveal vulnerabilities that could be exploited, and demonstrates the potential damage this type of attacker could cause. Internal security testing also focuses on system-level security and configuration—including application and service configuration, authentication, access control, and system hardening.

Assessors who perform internal testing are often granted some level of access to the network, normally as general users, and are provided with information that users with similar privileges would have. This level of temporary access depends on the goals of the test, and can be up to and including the privileges of a system or network administrator. Working from whatever level of access they have been granted, assessors attempt to gain additional access to the network and systems through privilege escalation—i.e., increasing user-level privileges to administrator-level privileges, or increasing system administrator privileges to domain administrator privileges.

Internal testing is not as limited as external testing because it takes place behind perimeter defenses, even though there may be internal firewalls, routers, and switches in place that pose limitations. Examination techniques such as network sniffing may be used in addition to testing techniques.

If both internal and external testing is to be performed, the external testing usually takes place first. This is particularly beneficial if the same assessors will be performing both types of testing, as it keeps them from acquiring insider information on network architecture or system configuration that would not be available to an adversary—an advantage that would reduce the validity of the test.

2.4.2 Overt and Covert

Overt security testing, also known as white hat testing, involves performing external and/or internal testing with the knowledge and consent of the organization's IT staff, enabling comprehensive evaluation of the network or system security posture. Because the IT staff is fully aware of and involved in the testing, it may be able to provide guidance to limit the testing's impact. Testing may also provide a training opportunity, with staff observing the activities and methods used by assessors to evaluate and potentially circumvent implemented security measures. This gives context to the security requirements implemented or maintained by the IT staff, and also may help teach IT staff how to conduct testing.

Covert security testing, also known as black hat testing, takes an adversarial approach by performing testing without the knowledge of the organization's IT staff but with the full knowledge and permission of upper management. Some organizations designate a trusted third party to ensure that the target organization does not initiate response measures associated with the attack without first verifying that an attack is indeed underway (e.g., that the activity being detected does not originate from a test). In such situations, the trusted third party provides an agent for the assessors, the management, the IT staff, and the security staff that mediates activities and facilitates communications. This type of test is useful for testing technical security controls, IT staff response to perceived security incidents, and staff knowledge and implementation of the organization's security policy. Covert testing may be conducted with or without warning.

The purpose of covert testing is to examine the damage or impact an adversary can cause—it does not focus on identifying vulnerabilities. This type of testing does not test every security control, identify each vulnerability, or assess all systems within an organization. Covert testing examines the organization from

an adversarial perspective, and normally identifies and exploits the most rudimentary vulnerabilities to gain network access. If an organization's goal is to mirror a specific adversary, this type of testing requires special considerations—such as acquiring and modeling threat data. The resulting scenarios provide an overall strategic view of the potential methods of exploit, risk, and impact of an intrusion. Covert testing usually has defined boundaries, such as stopping testing when a certain level of access is achieved or a certain type of damage is achievable as a next step in testing. Having such boundaries prevents damage while still showing that the damage could occur.

Besides failing to identify many vulnerabilities, covert testing is often time-consuming and costly due to its stealth requirements. To operate in a stealth environment, a test team will have to slow its scans and other actions to stay "under the radar" of the target organization's security staff. When testing is performed in-house, training must also be considered in terms of time and budget. In addition, an organization may have staff trained to perform regular activities such as scanning and vulnerability assessments, but not specialized techniques such as penetration or application security testing. Overt testing is less expensive, carries less risk than covert testing, and is more frequently used—but covert testing provides a better indication of the everyday security of the target organization because system administrators will not have heightened awareness.

3. Review Techniques

Review techniques passively examine systems, applications, networks, policies, and procedures to discover security vulnerabilities.[6] They also gather information to facilitate and optimize other assessment techniques. Because review techniques are passive, they pose minimal risk to systems and networks. This section covers several common review techniques—documentation, log, ruleset, and system configuration review; network sniffing; and file integrity checking.

3.1 Documentation Review

Documentation review determines if the technical aspects of policies and procedures are current and comprehensive. These documents provide the foundation for an organization's security posture, but are often overlooked during technical assessments. Security groups within the organization should provide assessors with appropriate documentation to ensure a comprehensive review. Documents to review for technical accuracy and completeness include security policies, architectures, and requirements; standard operating procedures; system security plans and authorization agreements; memoranda of understanding and agreement for system interconnections; and incident response plans.

Documentation review can discover gaps and weaknesses that could lead to missing or improperly implemented security controls. Assessors typically verify that the organization's documentation is compliant with standards and regulations such as FISMA, and look for policies that are deficient or outdated. Common documentation weaknesses include OS security procedures or protocols that are no longer used, and failure to include a new OS and its protocols. Documentation review does not ensure that security controls are implemented properly—only that the direction and guidance exist to support security infrastructure.

Results of documentation review can be used to fine-tune other testing and examination techniques. For example, if a password management policy has specific requirements for minimum password length and complexity, this information can be used to configure password-cracking tools for more efficient performance.

3.2 Log Review

Log review determines if security controls are logging the proper information, and if the organization is adhering to its log management policies.[7] As a source of historical information, audit logs can be used to help validate that the system is operating in accordance with established policies. For example, if the logging policy states that all authentication attempts to critical servers must be logged, the log review will determine if this information is being collected and shows the appropriate level of detail. Log review may also reveal problems such as misconfigured services and security controls, unauthorized accesses, and attempted intrusions. For example, if an intrusion detection system (IDS) sensor is placed behind a firewall, its logs can be used to examine communications that the firewall allows into the network. If the sensor registers activities that should be blocked, it indicates that the firewall is not configured securely.

---

[6] This publication discusses reviews strictly from the aspect of assessment. Reviews should also be conducted periodically as part of regular system monitoring and maintenance, such as to identify operational problems, security misconfigurations, malicious activity, and other types of security events. Organizations can choose to use findings from operational reviews for their assessments.

[7] NIST SP 800-92, *Guide to Security Log Management*, provides more information on security log management methods and techniques, including log review. It is available at http://csrc.nist.gov/publications/PubsSPs.html.

Examples of log information that may be useful when conducting technical security assessments include:

- Authentication server or system logs may include successful and failed authentication attempts.
- System logs may include system and service startup and shutdown information, installation of unauthorized software, file accesses, security policy changes, account changes (e.g., account creation and deletion, account privilege assignment), and privilege use.
- Intrusion detection and prevention system logs may include malicious activity and inappropriate use.
- Firewall and router logs may include outbound connections that indicate compromised internal devices (e.g., rootkits, bots, Trojan horses, spyware).
- Firewall logs may include unauthorized connection attempts and inappropriate use.
- Application logs may include unauthorized connection attempts, account changes, use of privileges, and application or database usage information.
- Antivirus logs may include update failures and other indications of outdated signatures and software.
- Security logs, in particular patch management and some IDS and intrusion prevention system (IPS) products, may record information on known vulnerable services and applications.

Manually reviewing logs can be extremely time-consuming and cumbersome. Automated audit tools are available that can significantly reduce review time and generate predefined and customized reports that summarize log contents and track them to a set of specific activities. Assessors can also use these automated tools to facilitate log analysis by converting logs in different formats to a single, standard format for analysis. In addition, if assessors are reviewing a specific action—such as the number of failed logon attempts in an organization—they can use these tools to filter logs based on the activity being checked.

3.3 Ruleset Review

A ruleset is a collection of rules or signatures that network traffic or system activity is compared against to determine what action to take—for example, forwarding or rejecting a packet, creating an alert, or allowing a system event. Review of these rulesets is done to ensure comprehensiveness and identify gaps and weaknesses on security devices and throughout layered defenses such as network vulnerabilities, policy violations, and unintended or vulnerable communication paths. A review can also uncover inefficiencies that negatively impact a ruleset's performance.

Rulesets to review include network- and host-based firewall and IDS/IPS rulesets, and router access control lists. The following list provides examples of the types of checks most commonly performed in ruleset reviews:

- For router access control lists
  - Each rule is still required (for example, rules that were added for temporary purposes are removed as soon as they are no longer needed)
  - Only traffic that is authorized per policy is permitted, and all other traffic is denied by default
- For firewall rulesets

  - Each rule is still required
  - Rules enforce least privilege access, such as specifying only required IP addresses and ports
  - More specific rules are triggered before general rules
  - There are no unnecessary open ports that could be closed to tighten the perimeter security
  - The ruleset does not allow traffic to bypass other security defenses
  - For host-based firewall rulesets, the rules do not indicate the presence of backdoors, spyware activity, or prohibited applications such as peer-to-peer file sharing programs

- For IDS/IPS rulesets
  - Unnecessary signatures have been disabled or removed to eliminate false positives and improve performance
  - Necessary signatures are enabled and have been fine-tuned and properly maintained.

3.4 System Configuration Review

System configuration review is the process of identifying weaknesses in security configuration controls, such as systems not being hardened or configured according to security policies. For example, this type of review will reveal unnecessary services and applications, improper user account and password settings, and improper logging and backup settings. Examples of security configuration files that may be reviewed are Windows security policy settings and Unix security configuration files such as those in */etc*.

Assessors using manual review techniques rely on security configuration guides or checklists to verify that system settings are configured to minimize security risks.[8] To perform a manual system configuration review, assessors access various security settings on the device being evaluated and compare them with recommended settings from the checklist. Settings that do not meet minimum security standards are flagged and reported.

The Security Content Automation Protocol (SCAP) is a method for using specific standards to enable automated vulnerability management, measurement, and policy compliance evaluation.[9] NIST SCAP files are written for FISMA compliance and NIST SP 800-53A security control testing. Other tools can be used to retrieve and report security settings and provide remediation guidance. Automated tools are often executed directly on the device being assessed, but can also be executed on a system with network access to the device being assessed. While automated system configuration reviews are faster than manual methods, there may still be settings that must be checked manually. Both manual and automated methods require root or administrator privileges to view selected security settings.

Generally it is preferable to use automated checks instead of manual checks whenever feasible. Automated checks can be done very quickly and provide consistent, repeatable results. Having a person manually checking hundreds or thousands of settings is tedious and error-prone.

[8] NIST maintains a repository of security configuration checklists for IT products at http://checklists.nist.gov/.
[9] More information on SCAP is located at http://scap.nist.gov/.

3.5 Network Sniffing

Network sniffing is a passive technique[10] that monitors network communication, decodes protocols, and examines headers and payloads to flag information of interest. Besides being used as a review technique, network sniffing can also be used as a target identification and analysis technique (see Section 4.1). Reasons for using network sniffing include the following:

- Capturing and replaying network traffic
- Performing passive network discovery (e.g., identifying active devices on the network)
- Identifying operating systems, applications, services, and protocols, including unsecured (e.g., telnet) and unauthorized (e.g., peer-to-peer file sharing) protocols
- Identifying unauthorized and inappropriate activities, such as the unencrypted transmission of sensitive information
- Collecting information, such as unencrypted usernames and passwords.

Network sniffing has little impact on systems and networks, with the most noticeable impact being on bandwidth or computing power utilization. The sniffer—the tool used to conduct network sniffing—requires a means to connect to the network, such as a hub, tap, or switch with port spanning. Port spanning is the process of copying the traffic transmitted on all other ports to the port where the sniffer is installed. Organizations can deploy network sniffers in a number of locations within an environment. These commonly include the following:

- At the perimeter, to assess traffic entering and exiting the network
- Behind firewalls, to assess that rulesets are accurately filtering traffic
- Behind IDSs/IPSs, to determine if signatures are triggering and being responded to appropriately
- In front of a critical system or application to assess activity
- On a specific network segment, to validate encrypted protocols.

One limitation to network sniffing is the use of encryption. Many attackers take advantage of encryption to hide their activities—while assessors can see that communication is taking place, they are unable to view the contents. Another limitation is that a network sniffer is only able to sniff the traffic of the local segment where it is installed. This requires the assessor to move it from segment to segment, install multiple sniffers throughout the network, and/or use port spanning. Assessors may also find it challenging to locate an open physical network port for scanning on each segment. In addition, network sniffing is a fairly labor-intensive activity that requires a high degree of human involvement to interpret network traffic.

3.6 File Integrity Checking

File integrity checkers provide a way to identify that system files have been changed computing and storing a checksum for every guarded file, and establishing a file checksum database. Stored checksums are later recomputed to compare their current value with the stored value, which identifies file

[10] Sniffers may perform domain name lookups for the traffic they collect, during which they generate network traffic. Domain name lookups can be disabled for stealthy network sniffing.

modifications. A file integrity checker capability is usually included with any commercial host-based IDS, and is also available as a standalone utility.

Although an integrity checker does not require a high degree of human interaction, it must be used carefully to ensure its effectiveness. File integrity checking is most effective when system files are compared with a reference database created using a system known to be secure—this helps ensure that the reference database was not built with compromised files. The reference database should be stored offline to prevent attackers from compromising the system and covering their tracks by modifying the database. In addition, because patches and other updates change files, the checksum database should be kept up-to-date.

For file integrity checking, strong cryptographic checksums such as Secure Hash Algorithm 1 (SHA-1) should be used to ensure the integrity of data stored in the checksum database. Federal agencies are required by Federal Information Processing Standard (FIPS) PUB 140-2, *Security Requirements for Cryptographic Modules*[11], to use SHA (e.g., SHA-1, SHA-256).

3.7 Summary

Table 3-1 summarizes the major capabilities of review techniques discussed in Section 3.

Table 3-1. Review Techniques

Technique	Capabilities
Documentation Review	• Evaluates policies and procedures for technical accuracy and completeness
Log Review	• Provides historical information on system use, configuration, and modification • Could reveal potential problems and policy deviations
Ruleset Review	• Reveals holes in ruleset-based security controls
System Configuration Review	• Evaluates the strength of system configuration • Validates that systems are configured in accordance with hardening policy
Network Sniffing	• Monitors network traffic on the local segment to capture information such as active systems, operating systems, communication protocols, services, and applications • Verifies encryption of communications
File Integrity Checking	• Identifies changes to important files; can also identify certain forms of unwanted files, such as well-known attacker tools

Risks are associated with each technique and their combinations. To ensure that all are executed safely and accurately, each assessor should have a certain baseline skill set. Table 3-2 provides guidelines for the minimum skill set needed for each technique presented in Section 3.

Table 3-2. Baseline Skill Set for Review Techniques

Technique	Baseline Skill Set
Documentation Review	General knowledge of security from a policy perspective
Log Review	Knowledge of log formats and ability to interpret and analyze log data; ability to use automated log analysis and log correlation tools
Ruleset Review	Knowledge of ruleset formats and structures; ability to correlate and analyze rulesets from a variety of devices
System Configuration Review	Knowledge of secure system configuration, including OS hardening and security policy configuration for a variety of operating systems; ability to use automated security configuration testing tools

[11] FIPS PUB 140-2 is available at http://csrc.nist.gov/publications/PubsFIPS.html.

Technique	Baseline Skill Set
Network Sniffing	General Transmission Control Protocol/Internet Protocol (TCP/IP) and networking knowledge; ability to interpret and analyze network traffic; ability to deploy and use network sniffing tools
File Integrity Checking	General file system knowledge; ability to use automated file integrity checking tools and interpret the results

4. Target Identification and Analysis Techniques

This section addresses technical target identification and analysis techniques, which focus on identifying active devices and their associated ports and services, and analyzing them for potential vulnerabilities. The assessor uses this information to continue to explore devices that will validate existence of the vulnerabilities. Organizations often use non-technical techniques in addition or instead of technical techniques to identify the assets to be analyzed. For example, organizations may have existing asset inventories or other lists of assets to be targeted; another example is assessors performing a walkthrough of a facility to identify assets that were not found by technical techniques, such as hosts that were shut off or disconnected from the network when the technical techniques were used.

Target identification and analysis techniques for application security examination are briefly discussed in Appendix C.

4.1 Network Discovery

Network discovery uses a number of methods to discover active and responding hosts on a network, identify weaknesses, and learn how the network operates. Both passive (examination) and active (testing) techniques exist for discovering devices on a network. Passive techniques use a network sniffer to monitor network traffic and record the IP addresses of the active hosts, and can report which ports are in use and which operating systems have been discovered on the network. Passive discovery can also identify the relationships between hosts—including which hosts communicate with each other, how frequently their communication occurs, and the type of traffic that is taking place—and is usually performed from a host on the internal network where it can monitor host communications. This is done without sending out a single probing packet. Passive discovery takes more time to gather information than does active discovery, and hosts that do not send or receive traffic during the monitoring period might not be reported.

Active techniques send various types of network packets, such as Internet Control Message Protocol (ICMP) pings, to solicit responses from network hosts, generally through the use of an automated tool. One activity, known as OS fingerprinting, enables the assessor to determine the system's OS by sending it a mix of normal, abnormal, and illegal network traffic. Another activity involves sending packets to common port numbers to generate responses that indicate the ports are active. The tool analyzes the responses from these activities, and compares them with known traits of packets from specific operating systems and network services—enabling it to identify hosts, the operating systems they run, their ports, and the state of those ports. This information can be used for purposes that include gathering information on targets for penetration testing, generating topology maps, determining firewall and IDS configurations, and discovering vulnerabilities in systems and network configurations.

Network discovery tools have many ways to acquire information through scanning. Enterprise firewalls and intrusion detection systems can identify many instances of scans, particularly those that use the most suspicious packets (e.g., SYN/FIN scan, NULL scan). Assessors who plan on performing discovery through firewalls and intrusion detection systems should consider which types of scans are most likely to provide results without drawing the attention of security administrators, and how scans can be conducted in a more stealthy manner (such as more slowly or from a variety of source IP addresses) to improve their chances of success. Assessors should also be cautious when selecting types of scans to use against older systems, particularly those known to have weak security, because some scans can cause system failures. Typically, the closer the scan is to normal activity, the less likely it is to cause operational problems.

Network discovery may also detect unauthorized or rogue devices operating on a network. For example, an organization that uses only a few operating systems could quickly identify rogue devices that utilize

different ones. Once a wired rogue device is identified,[12] it can be located by using existing network maps and information already collected on the device's network activity to identify the switch to which it is connected. It may be necessary to generate additional network activity with the rogue device—such as pings—to find the correct switch. The next step is to identify the switch port on the switch associated with the rogue device, and to physically trace the cable connecting that switch port to the rogue device.

A number of tools exist for use in network discovery, and it should be noted that many active discovery tools can be used for passive network sniffing and port scanning as well. Most offer a graphical user interface (GUI), and some also offer a command-line interface. Command-line interfaces may take longer to learn than GUIs because of the number of commands and switches that specify what tests the tool should perform and which an assessor must learn to use the tool effectively. Also, developers have written a number of modules for open source tools that allow assessors to easily parse tool output. For example, combining a tool's Extensible Markup Language (XML) output capabilities, a little scripting, and a database creates a more powerful tool that can monitor the network for unauthorized services and machines. Learning what the many commands do and how to combine them is best achieved with the help of an experienced security engineer. Most experienced IT professionals, including system administrators and other network engineers, should be able to interpret results, but working with the discovery tools themselves is more efficiently handled by an engineer.

Some of the advantages of active discovery, as compared to passive discovery, are that an assessment can be conducted from a different network and usually requires little time to gather information. In passive discovery, ensuring that all hosts are captured requires traffic to hit all points, which can be time-consuming—especially in larger enterprise networks.

A disadvantage to active discovery is that it tends to generate network noise, which sometimes results in network latency. Since active discovery sends out queries to receive responses, this additional network activity could slow down traffic or cause packets to be dropped in poorly configured networks if performed at high volume. Active discovery can also trigger IDS alerts, since unlike passive discovery it reveals its origination point. The ability to successfully discover all network systems can be affected by environments with protected network segments and perimeter security devices and techniques. For example, an environment using network address translation (NAT)—which allows organizations to have internal, non-publicly routed IP addresses that are translated to a different set of public IP addresses for external traffic—may not be accurately discovered from points external to the network or from protected segments. Personal and host-based firewalls on target devices may also block discovery traffic. Misinformation may be received as a result of trying to instigate activity from devices. Active discovery presents information from which conclusions must be drawn about settings on the target network.

For both passive and active discovery, the information received is seldom completely accurate. To illustrate, only hosts that are on and connected during active discovery will be identified—if systems or a segment of the network are offline during the assessment, there is potential for a large gap in discovering devices. Although passive discovery will only find devices that transmit or receive communications during the discovery period, products such as network management software can provide continuous discovery capabilities and automatically generate alerts when a new device is present on the network. Continuous discovery can scan IP address ranges for new addresses or monitor new IP address requests. Also, many discovery tools can be scheduled to run regularly, such as once every set amount of days at a particular time. This provides more accurate results than running these tools sporadically.

[12] See Section 4.4 for information on locating wireless rogue devices.

4.2 Network Port and Service Identification

Network port and service identification involves using a port scanner to identify network ports and services operating on active hosts—such as FTP and HTTP—and the application that is running each identified service, such as Microsoft Internet Information Server (IIS) or Apache for the HTTP service. Organizations should conduct network port and service identification to identify hosts if this has not already been done by other means (e.g., network discovery), and flag potentially vulnerable services. This information can be used to determine targets for penetration testing.

All basic scanners can identify active hosts and open ports, but some scanners are also able to provide additional information on the scanned hosts. Information gathered during an open port scan can assist in identifying the target operating system through a process called *OS fingerprinting*. For example, if a host has TCP ports 135, 139, and 445 open, it is probably a Windows host, or possibly a Unix host running Samba. Other items—such as the TCP packet sequence number generation and responses to packets—also provide a clue to identifying the OS. But OS fingerprinting is not foolproof. For example, firewalls block certain ports and types of traffic, and system administrators can configure their systems to respond in nonstandard ways to camouflage the true OS.

Some scanners can help identify the application running on a particular port through a process called service identification. Many scanners use a services file that lists common port numbers and typical associated services—for example, a scanner that identifies that TCP port 80 is open on a host may report that a web server is listening at that port—but additional steps are needed before this can be confirmed. Some scanners can initiate communications with an observed port and analyze its communications to determine what service is there, often by comparing the observed activity to a repository of information on common services and service implementations. These techniques may also be used to identify the service application and application version, such as which Web server software is in use—this process is known as *version scanning*. A well-known form of version scanning, called *banner grabbing*, involves capturing banner information transmitted by the remote port when a connection is initiated. This information can include the application type, application version, and even OS type and version. Version scanning is not foolproof, because a security-conscious administrator can alter the transmitted banners or other characteristics in hopes of concealing the service's true nature. However, version scanning is far more accurate than simply relying on a scanner's services file.

Scanner models support the various scanning methods with strengths and weaknesses that are normally explained in their documentation. For example, some scanners work best scanning through firewalls, while others are better suited for scans inside the firewall. Results will differ depending on the port scanner used. Some scanners respond with a simple open or closed response for each port, while others offer additional detail (e.g., filtered or unfiltered) that can assist the assessor in determining what other types of scans would be helpful to gain additional information.

Network port and service identification often uses the IP address results of network discovery as the devices to scan. Port scans can also be run independently on entire blocks of IP addresses—here, port scanning performs network discovery by default through identifying the active hosts on the network. The result of network discovery and network port and service identification is a list of all active devices operating in the address space that responded to the port scanning tool, along with responding ports. Additional active devices could exist that did not respond to scanning, such as those that are shielded by firewalls or turned off. Assessors can try to find these devices by scanning the devices themselves,

placing the scanner on a segment that can access the devices, or attempting to evade the firewall through the use of alternate scan types (e.g., SYN/FIN or Xmas scan).[13]

It is recommended that if both external and internal scanning are to be used and the assessors are intentionally performing the testing "blind," that external scanning be performed first. Done in this order, logs can be reviewed and compared before and during internal testing. When performing external scanning, assessors may use any existing stealth techniques to get packets through firewalls while evading detection by IDS and IPS.[14] Tools that use fragmentation, duplication, overlap, out-of-order, and timing techniques to alter packets so that they blend into and appear more like normal traffic are recommended. Internal testing tends to use less aggressive scanning methods because these scans are blocked less often than external scans. Using more aggressive scans internally significantly increases the changes of disrupting operations without necessarily improving scan results. Being able to scan a network with customized packets also works well for internal testing, because checking for specific vulnerabilities requires highly customized packets. Tools with packet-builder ability are helpful with this process. Once built, packets can be sent through a second scanning program that will collect the results. Because customized packets can trigger a denial of service (DoS) attack, this type of test should be conducted during periods of low network traffic—such as overnight or on the weekend.

Although port scanners identify active hosts, operating systems, ports, services, and applications, they do not identify vulnerabilities. Additional investigation is needed to confirm the presence of insecure protocols (e.g., Trivial File Transfer Protocol [TFTP], telnet), malware, unauthorized applications, and vulnerable services. To identify vulnerable services, the assessor compares identified version numbers of services with a list of known vulnerable versions, or perform automated vulnerability scanning as discussed in Section 4.3. With port scanners, the scanning process is highly automated but interpretation of the scanned data is not.

Although port scanning can disrupt network operations by consuming bandwidth and slowing network response times, it enables an organization to ensure that its hosts are configured to run only approved network services. Scanning software should be carefully selected to minimize disruptions to operations. Port scanning can also be conducted after hours to cause minimal impact to operations.

4.3 Vulnerability Scanning

Like network port and service identification, vulnerability scanning identifies hosts and host attributes (e.g., operating systems, applications, open ports), but it also attempts to identify vulnerabilities rather than relying on human interpretation of the scanning results. Many vulnerability scanners are equipped to accept results from network discovery and network port and service identification, which reduces the amount of work needed for vulnerability scanning. Also, some scanners can perform their own network discovery and network port and service identification. Vulnerability scanning can help identify outdated software versions, missing patches, and misconfigurations, and validate compliance with or deviations from an organization's security policy. This is done by identifying the operating systems and major software applications running on the hosts and matching them with information on known vulnerabilities stored in the scanners' vulnerability databases.

Vulnerability scanners can:

- Check compliance with host application usage and security policies

[13] Many firewalls can recognize and block various alternate scan types, so testers may not be able to use them to evade firewalls in many environments.

[14] This can be particularly helpful in improving the tuning and configuration of IDSs and IPSs.

- Provide information on targets for penetration testing
- Provide information on how to mitigate discovered vulnerabilities.

Vulnerability scanners can be run against a host either locally or from the network. Some network-based scanners have administrator-level credentials on individual hosts and can extract vulnerability information from hosts using those credentials. Other network-based scanners do not have such credentials and must rely on conducting scanning of networks to locate hosts and then scan those hosts for vulnerabilities. In such cases, network-based scanning is primarily used to perform network discovery and identify open ports and related vulnerabilities—in most cases, it is not limited by the OS of the targeted systems. Network-based scanning without host credentials can be performed both internally and externally—and although internal scanning usually uncovers more vulnerabilities than external scanning, testing from both viewpoints is important. External scanning must contend with perimeter security devices that block traffic, limiting assessors to scanning only the ports authorized to pass traffic.

Assessors performing external scanning may find challenges similar to those faced with network discovery, such as the use of NAT or personal and host-based firewalls. To overcome the challenges of NAT and conduct successful network-based scanning, assessors can ask the firewall administrator to enable port forwarding on specific IP addresses or groups of addresses if this is supported by the firewall, or request network access behind the device performing NAT. Assessors can also request that personal or host-based firewalls be configured to permit traffic from test system IP addresses during the assessment period. These steps will give assessors increased insight into the network, but do not accurately reflect the capabilities of an external attacker—although they may offer a better indication of the capabilities available to a malicious insider or an external attacker with access to another host on the internal network. Assessors can also perform scanning on individual hosts.

For local vulnerability scanning, a scanner is installed on each host to be scanned. This is done primarily to identify host OS and application misconfigurations and vulnerabilities—both network-exploitable and locally exploitable. Local scanning is able to detect vulnerabilities with a higher level of detail than network-based scanning because local scanning usually requires both host (local) access and a root or administrative account. Some scanners also offer the capability of repairing local misconfigurations.

A vulnerability scanner is a relatively fast and easy way to quantify an organization's exposure to surface vulnerabilities. A surface vulnerability is a weakness that exists in isolation, independent from other vulnerabilities. The system’s behaviors and outputs in response to attack patterns submitted by the scanner are compared against those that characterize the signatures of known vulnerabilities, and the tool reports any matches that are found. Besides signature-based scanning, some vulnerability scanners attempt to simulate the reconnaissance attack patterns used to probe for exposed, exploitable vulnerabilities, and report the vulnerabilities found when these techniques are successful.

One difficulty in identifying the risk level of vulnerabilities is that they rarely exist in isolation. For example, there could be several low-risk vulnerabilities that present a higher risk when combined. Scanners are unable to detect vulnerabilities that are revealed only as the result of potentially unending combinations of attack patterns. The tool may assign a low risk to each vulnerability, leaving the assessor falsely confident in the security measures in place. A more reliable way of identifying the risk of vulnerabilities in aggregate is through penetration testing, which is discussed in Section 5.2.

Another problem with identifying the risk level of vulnerabilities is that vulnerability scanners often use their own proprietary methods for defining the levels. For example, one scanner might use the levels low, medium, and high, while another scanner might use the levels informational, low, medium, high, and critical. This makes it difficult to compare findings among multiple scanners. Also, the risk levels assigned by a scanner may not reflect the actual risk to the organization—for example, a scanner might

label an FTP server as a moderate risk because it transmits passwords in cleartext, but if the organization only uses the FTP server as an anonymous public server that does not use passwords, then the actual risk might be considerably lower. Assessors should determine the appropriate risk level for each vulnerability and not simply accept the risk levels assigned by vulnerability scanners.

Network-based vulnerability scanning has some significant weaknesses. As with network sniffing and discovery, this type of scanning uncovers vulnerabilities only for active systems. This generally covers surface vulnerabilities, and is unable to address the overall risk level of a scanned network. Although the process itself is highly automated, vulnerability scanners can have a high false positive error rate (i.e., reporting vulnerabilities when none exist). An individual with expertise in networking and OS security should interpret the results. And because network-based vulnerability scanning requires more information than port scanning to reliably identify the vulnerabilities on a host, it tends to generate significantly more network traffic than port scanning. This may have a negative impact on the hosts or network being scanned, or on network segments through which scanning traffic is traversing. Many vulnerability scanners also include network-based tests for DoS attacks that, in the hands of an inexperienced assessor, can have a marked negative impact on scanned hosts. Scanners often allow all DoS attack tests to be suppressed so as to reduce the risk of impacting hosts through testing.

Another significant limitation of vulnerability scanners is that, like virus scanners and IDSs, they rely on a repository of signatures. This requires the assessors to update these signatures frequently to enable the scanner to recognize the latest vulnerabilities. Before running any scanner, an assessor should install the latest updates to its vulnerability database. Some vulnerability scanner databases are updated more regularly than others—this update frequency should be a major consideration when selecting a vulnerability scanner.

Most vulnerability scanners allow the assessor to perform different levels of scanning that vary in terms of thoroughness. While more comprehensive scanning may detect a greater number of vulnerabilities, it can slow the overall scanning process. Less comprehensive scanning can take less time, but identifies only well-known vulnerabilities. It is generally recommended that assessors conduct a thorough vulnerability scan if resources permit.

Vulnerability scanning is a somewhat labor-intensive activity that requires a high degree of human involvement to interpret results. It may also disrupt network operations by taking up bandwidth and slowing response times. Nevertheless, vulnerability scanning is extremely important in ensuring that vulnerabilities are mitigated before they are discovered and exploited by adversaries.

As with all pattern-matching and signature-based tools, application vulnerability scanners typically have high false positive rates. Assessors should configure and calibrate their scanners to minimize both false positives and false negatives to the greatest possible extent, and meaningfully interpret results to identify the real vulnerabilities. Scanners also suffer from the high false negative rates that characterize other signature-based tools—but vulnerabilities that go undetected by automated scanners can potentially be caught using multiple vulnerability scanners or additional forms of testing. A common practice is to use multiple scanners—this provides assessors with a way to compare results.

4.4 Wireless Scanning

Wireless technologies, in their simplest sense, enable one or more devices to communicate without the need for physical connections such as network or peripheral cables. They range from simple technologies like wireless keyboards and mice to complex cell phone networks and enterprise wireless local area networks (WLAN). As the number and availability of wireless-enabled devices continues to increase, it

is important for organizations to actively test and secure their enterprise wireless environments.[15] Wireless scans can help organizations determine corrective actions to mitigate risks posed by wireless-enabled technologies.

The following factors in the organization's environment should be taken into consideration when planning technical wireless security assessments:

- The location of the facility being scanned, because the physical proximity of a building to a public area (e.g., streets and public common areas) or its location in a busy metropolitan area may increase the risk of wireless threats
- The security level of the data to be transmitted using wireless technologies
- How often wireless devices connect to and disconnect from the environment, and the typical traffic levels for wireless devices (e.g., occasional activity or fairly constant activity)—this is because only active wireless devices are discoverable during a wireless scan
- Existing deployments of wireless intrusion detection and prevention systems (WIDPS[16]), which may already collect most of the information that would be gathered by testing.

Wireless scanning should be conducted using a mobile device with wireless analyzer software installed and configured—such as a laptop, handheld device, or specialty device. The scanning software or tool should allow the operator to configure the device for specific scans, and to scan in both passive and active modes. The scanning software should also be configurable by the operator to identify deviations from the organization's wireless security configuration requirements.

The wireless scanning tool should be capable of scanning all Institute of Electrical and Electronics Engineers (IEEE) 802.11a/b/g/n channels, whether domestic or international. In some cases, the device should also be fitted with an external antenna to provide an additional level of radio frequency (RF) capturing capability. Support for other wireless technologies, such as Bluetooth, will help evaluate the presence of additional wireless threats and vulnerabilities. Note that devices using nonstandard technology or frequencies outside of the scanning tool's RF range will not be detected or properly recognized by the scanning tool. A tool such as an RF spectrum analyzer will assist organizations in identifying transmissions that occur within the frequency range of the spectrum analyzer. Spectrum analyzers generally analyze a large frequency range (e.g., 3 to 18 GHz) —and although these devices do not analyze traffic, they enable an assessor to determine wireless activity within a specific frequency range and tailor additional testing and examination accordingly.

Some devices also support mapping and physical location plotting through use of a mapping tool, and in some cases support Global Positioning System (GPS)-based mapping. Sophisticated wireless scanning tools allow the user to import a floor plan or map to assist in plotting the physical location of discovered devices. (It is important to note that GPS has limited capabilities indoors.)

Individuals with a strong understanding of wireless networking—especially IEEE 802.11a/b/g/n technologies—should operate wireless scanning tools. These operators should be trained on the functionality and capability of the scanning tools and software to better understand the captured information and be more apt to identify potential threats or malicious activity. Individuals with similar

[15] For proper measures to secure IEEE 802.11-based WLANs, please refer to NIST SP 800-97, *Establishing Wireless Robust Security Networks: A Guide to IEEE 802.11i*, and NIST SP 800-48 Revision 1, *Guide to Securing Legacy IEEE 802.11 Wireless Networks*, available at http://csrc.nist.gov/publications/PubsSPs.html.

[16] For more information, see NIST SP 800-94, *Guide to Intrusion Detection and Prevention Systems (IDPS)*, which is available at http://csrc.nist.gov/publications/PubsSPs.html.

skills should be employed to analyze the data and results acquired from wireless scans. Scanning tool operators should be aware of other RF signals authorized for use within the area being scanned.

4.4.1 Passive Wireless Scanning

Passive scanning should be conducted regularly to supplement wireless security measures already in place, such as WIDPSs.[17] Wireless scanning tools used to conduct completely passive scans transmit no data, nor do the tools in any way affect the operation of deployed wireless devices. By not transmitting data, a passive scanning tool remains undetected by malicious users and other devices. This reduces the likelihood of individuals avoiding detection by disconnecting or disabling unauthorized wireless devices.

Passive scanning tools capture wireless traffic being transmitted within the range of the tool's antenna. Most tools provide several key attributes regarding discovered wireless devices, including service set identifier (SSID), device type, channel, media access control (MAC) address, signal strength, and number of packets being transmitted. This information can be used to evaluate the security of the wireless environment, and to identify potential rogue devices and unauthorized ad hoc networks discovered within range of the scanning device. The wireless scanning tool should also be able to assess the captured packets to determine if any operational anomalies or threats exist.

Wireless scanning tools scan each IEEE 802.11a/b/g/n channel/frequency separately, often for only several hundred milliseconds at a time. The passive scanning tool may not receive all transmissions on a specific channel. For example, the tool may have been scanning channel 1 at the precise moment when a wireless device transmitted a packet on channel 5. This makes it important to set the dwell time of the tool to be long enough to capture packets, yet short enough to efficiently scan each channel. Dwell time configurations will depend on the device or tool used to conduct the wireless scans. In addition, security personnel conducting the scans should slowly move through the area being scanned to reduce the number of devices that go undetected.

Rogue devices can be identified in several ways through passive scanning:

- The MAC address of a discovered wireless device indicates the vendor of the device's wireless interface. If an organization only deploys wireless interfaces from vendors A and B, the presence of interfaces from any other vendor indicates potential rogue devices.
- If an organization has accurate records of its deployed wireless devices, assessors can compare the MAC addresses of discovered devices with the MAC addresses of authorized devices. Most scanning tools allow assessors to enter a list of authorized devices. Because MAC addresses can be spoofed, assessors should not assume that the MAC addresses of discovered devices are accurate—but checking MAC addresses can identify rogue devices that do not use spoofing.
- Rogue devices may use SSIDs that are not authorized by the organization.
- Some rogue devices may use SSIDs that are authorized by the organization but do not adhere to its wireless security configuration requirements.

The signal strength of potential rogue devices should be reviewed to determine whether the devices are located within the confines of the facility or in the area being scanned. Devices operating outside an

[17] In some environments, the WIDPS implementation might be performing most of the same functions as passive wireless scanning. Some WIDPS products offer mobile sensors similar to the wireless scanning device setup described in Section 4.4. Organizations with WIDPS implementations should use the wireless scanning techniques described in this publication to supplement, not duplicate, WIDPS functionality.

organization's confines might still pose significant risks because the organization's devices might inadvertently associate to them.

4.4.2 Active Wireless Scanning

Organizations can move beyond passive wireless scanning to conduct active scanning. This builds on the information collected during passive scans, and attempts to attach to discovered devices and conduct penetration or vulnerability-related testing. For example, organizations can conduct active wireless scanning on their authorized wireless devices to ensure that they meet wireless security configuration requirements—including authentication mechanisms, data encryption, and administration access if this information is not already available through other means.

Organizations should be cautious in conducting active scans to make sure they do not inadvertently scan devices owned or operated by neighboring organizations that are within range. It is important to evaluate the physical location of devices before actively scanning them. Organizations should also be cautious in performing active scans of rogue devices that appear to be operating within the organization's facility. Such devices could belong to a visitor to the organization who inadvertently has wireless access enabled, or to a neighboring organization with a device that is close to, but not within, the organization's facility. Generally, organizations should focus on identifying and locating potential rogue devices rather than performing active scans of such devices.

Organizations may use active scanning when conducting penetration testing on their own wireless devices. Tools are available that employ scripted attacks and functions, attempt to circumvent implemented security measures, and evaluate the security level of devices. For example, tools used to conduct wireless penetration testing attempt to connect to access points (AP) through various methods to circumvent security configurations. If the tool can gain access to the AP, it can obtain information and identify the wired networks and wireless devices to which the AP is connected. Some active tools may also identify vulnerabilities discovered on the wireless client devices, or conduct wired network vulnerability tests as outlined in Section 4.

While active scanning is being performed, the organization's WIDPSs can be monitored to evaluate their capabilities and performance. Depending on assessment goals, assessors conducting these scans may need to inform the WIDPS administrators and wireless network administrators of pending scanning to prepare them for possible alarms and alerts. In addition, some WIDPSs can be configured to ignore alarms and alerts triggered by a specific device—such as one used to perform scanning.

Tools and processes to identify unauthorized devices and vulnerabilities on wired networks can also be used to identify rogue and misconfigured wireless devices. Wired-side scanning is another process that can be conducted to discover, and possibly locate, rogue wireless devices. Sections 3.5 and 4.1 discuss wired scanning.

4.4.3 Wireless Device Location Tracking

Security personnel who operate the wireless scanning tool should attempt to locate suspicious devices. RF signals propagate in a manner relative to the environment, which makes it important for the operator to understand how wireless technology supports this process. Mapping capabilities are useful here, but the main factors needed to support this capability are a knowledgeable operator and an appropriate wireless antenna.

If rogue devices are discovered and physically located during the wireless scan, security personnel should ensure that specific policies and processes are followed on how the rogue device is handled—such as

shutting it down, reconfiguring it to comply with the organization's policies, or removing the device completely. If the device is to be removed, security personnel should evaluate the activity of the rogue device before it is confiscated. This can be done through monitoring transmissions and attempting to access the device.

If discovered wireless devices cannot be located during the scan, security personnel should attempt to use a WIDPS to support the location of discovered devices. This requires the WIDPS to locate a specific MAC address that was discovered during the scan. Properly deployed WIDPSs should have the ability to assist security personnel in locating these devices, and usually involves the use of multiple WIDPS sensors to increase location identification granularity. Because the WIDPS will only be able to locate a device within several feet, a wireless scanning tool may still be needed to pinpoint the location of the device.

4.4.4 Bluetooth Scanning

For organizations that want to confirm compliance with their Bluetooth security requirements, passive scanning for Bluetooth-enabled wireless devices should be conducted to evaluate potential presence and activity. Because Bluetooth has a very short range (on average 9 meters [30 feet], with some devices having ranges of as little as 1 meter [3 feet]), scanning for devices can be difficult and time-consuming. Assessors should take range limitations into consideration when scoping this type of scanning. Organizations may want to perform scanning only in areas of their facilities that are accessible by the public—to see if attackers could gain access to devices via Bluetooth—or to perform scanning in a sampling of physical locations rather than throughout the entire facility. Because many Bluetooth-enabled devices (such as cell phones and personal digital assistants [PDA]) are mobile, conducting passive scanning several times over a period of time may be necessary. Organizations should also scan any Bluetooth infrastructure, such as access points, that they deploy. If rogue access points are discovered, the organization should handle them in accordance with established policies and processes.

A number of tools are available for actively testing the security and operation of Bluetooth devices. These tools attempt to connect to discovered devices and perform attacks to surreptitiously gain access and connectivity to Bluetooth-enabled devices. Assessors should be extremely cautious of performing active scanning because of the likelihood of inadvertently scanning personal Bluetooth devices, which are found in many environments. As a general rule, assessors should use active scanning only when they are certain that the devices being scanned belong to the organization. Active scanning can be used to evaluate the security mode in which a Bluetooth device is operating, and the strength of Bluetooth password identification numbers (PIN). Active scanning can also be used to verify that these devices are set to the lowest possible operational power setting to minimize their range. As with IEEE 802.11a/b/g rogue devices, rogue Bluetooth devices should be dealt with in accordance with policies and guidance.

4.5 Summary

Table 4-1 summarizes the major capabilities of the target identification and analysis techniques discussed in Section 4.

Table 4-1. Target Identification and Analysis Techniques

Technique	Capabilities
Network Discovery	• Discovers active devices • Identifies communication paths and facilitates determination of network architectures
Network Port and Service Identification	• Discovers active devices • Discovers open ports and associated services/ applications

Technique	Capabilities
Vulnerability Scanning	• Identifies hosts and open ports • Identifies known vulnerabilities (note: has high false positive rates) • Often provides advice on mitigating discovered vulnerabilities
Wireless Scanning	• Identifies unauthorized wireless devices within range of the scanners • Discovers wireless signals outside of an organization's perimeter • Detects potential backdoors and other security violations

There are risks associated with each technique and combination of techniques. To ensure that all are executed safely and accurately, each assessor should have a certain baseline skill set. Table 4-2 provides guidelines for the minimum skill set needed for each technique presented in Section 4.

Table 4-2. Baseline Skill Set for Target Identification and Analysis Techniques

Technique	Baseline Skill Set
Network Discovery	General TCP/IP and networking knowledge; ability to use both passive and active network discovery tools
Network Port and Service Identification	General TCP/IP and networking knowledge; knowledge of ports and protocols for a variety of operating systems; ability to use port scanning tools; ability to interpret results from tools
Vulnerability Scanning	General TCP/IP and networking knowledge; knowledge of ports, protocols, services, and vulnerabilities for a variety of operating systems; ability to use automated vulnerability scanning tools and interpret/analyze the results
Wireless Scanning	General knowledge of computing and radio transmissions in addition to specific knowledge of wireless protocols, services, and architectures; ability to use automated wireless scanning and sniffing tools

5. Target Vulnerability Validation Techniques

This section addresses target vulnerability validation techniques, which use information produced from target identification and analysis to further explore the existence of potential vulnerabilities. The objective is to prove that a vulnerability exists, and to demonstrate the security exposures that occur when it is exploited. Target vulnerability validation involves the greatest amount of risk in assessments, since these techniques have more potential to impact the target system or network than other techniques.

Target vulnerability validation techniques for application security testing are briefly discussed in Appendix C.

5.1 Password Cracking

When a user enters a password, a hash of the entered password is generated and compared with a stored hash of the user's actual password. If the hashes match, the user is authenticated. Password cracking is the process of recovering passwords from password hashes stored in a computer system or transmitted over networks. It is usually performed during assessments to identify accounts with weak passwords. Password cracking is performed on hashes that are either intercepted by a network sniffer while being transmitted across a network, or retrieved from the target system, which generally requires administrative-level access on, or physical access to, the target system. Once these hashes are obtained, an automated password cracker rapidly generates additional hashes until a match is found or the assessor halts the cracking attempt.

One method for generating hashes is a *dictionary attack*, which uses all words in a dictionary or text file. There are numerous dictionaries available on the Internet that encompass major and minor languages, names, popular television shows, etc. Another cracking method is known as a *hybrid attack*, which builds on the dictionary method by adding numeric and symbolic characters to dictionary words. Depending on the password cracker being used, this type of attack can try a number of variations, such as using common substitutions of characters and numbers for letters (e.g., p@ssword and h4ckme). Some will also try adding characters and numbers to the beginning and end of dictionary words (e.g., password99, password$%).

Yet another password-cracking method is called the *brute force* method. This generates all possible passwords up to a certain length and their associated hashes. Since there are so many possibilities, it can take months to crack a password. Although brute force can take a long time, it usually takes far less time than most password policies specify for password changing. Consequently, passwords found during brute force attacks are still too weak. Theoretically, all passwords can be cracked by a brute force attack, given enough time and processing power, although it could take many years and require serious computing power. Assessors and attackers often have multiple machines over which they can spread the task of cracking passwords, which greatly shortens the time involved.

Password cracking can also be performed with *rainbow tables*, which are lookup tables with pre-computed password hashes. For example, a rainbow table can be created that contains every possible password for a given character set up to a certain character length. Assessors may then search the table for the password hashes that they are trying to crack. Rainbow tables require large amounts of storage space and can take a long time to generate, but their primary shortcoming is that they may be ineffective against password hashing that uses *salting*. Salting is the inclusion of a random piece of information in the password hashing process that decreases the likelihood of identical passwords returning the same hash. Rainbow tables will not produce correct results without taking salting into account—but this dramatically increases the amount of storage space that the tables require. Many operating systems use