Pentest+

Disclosure attacks seek to gain unauthorized access to information or systems.
■■*Alteration* attacks seek to make unauthorized changes to information or systems.
■■*Denial* attacks seek to prevent legitimate use of information and systems.

**DAD TRIAD

***attackers' DAD triad directly corresponding to a leg of the CIA triad that is designed to counter those attacks. Confidentiality controls seek to prevent disclosure attacks. Integrity controls seek to prevent alteration attacks. Availability controls seek to keep systems running, preventing denial attacks.

controls

■■Security cameras in high risk areas
■■Auditing of cash register receipts
■■Theft detectors at the main entrance/exit of the store
■■Exit alarms on emergency exits

threat hunting seek to adopt the attacker's mind-set and imagine how hackers might seek to defeat an organization's security controls. The two disciplines diverge in what they accomplish with this information...

***Threat hunting builds upon a cybersecurity philosophy known as the "presumption of compromise."
This approach assumes that attackers have already successfully breached
an organization and searches out the evidence of successful attacks.

When threat hunters
discover a potential compromise, they then kick into incident-handling mode, seeking to contain, eradicate, and recover from the compromise.

***They also conduct a post-mortem
analysis of the factors that contributed to the compromise in an effort to remediate deficiencies. This post-event remediation is another similarity between penetration testing and threat hunting: organizations leverage the output of both processes in similar ways.

**[Payment Card
Industry Data Security Standard (PCI DSS).

This regulation is a private contractual obligation that governs all organizations involved in the storage, processing, or transmission

of credit and debit card transactions. Nestled among the more than 100 pages of detailed security requirements for cardholder data environments (CDEs) is section 11.3, which reads as follows:

Implement a methodology for penetration testing that includes the following:

;**Is based on industry accepted penetration testing approaches (for example, NIST SP800-115)
**Includes coverage for the entire CDE perimeter and critical systems

;**Includes testing from both inside and outside the network
;**Includes testing to validate any segmentation and scope-reduction controls
;**Defines application-layer penetration tests to include, at a minimum, ;the vulnerabilities listed in Requirement 6.5
;**Defines network-layer penetration tests to include components that ;support network functions as well as operating systems
;**Includes review and consideration of threats and vulnerabilities experienced in the last 12 months
;**Specifies retention of penetration testing results and remediation activities results

The standard goes on to include four additional requirements that describe the frequency and scope of penetration tests:

11.3.1. Perform external penetration testing at least annually and after any significant infrastructure or application upgrade or modification (such as an operating system upgrade, a sub-network added to the environment, or a web server added to the environment).
11.3.2 Perform internal penetration testing at least annually and after any significant infrastructure or application upgrade or modification (such as an operating system upgrade, a sub-network added to the environment, or a web server added to the environment).
11.3.3. Exploitable vulnerabilities found during penetration testing are corrected and the testing is repeated to verify the corrections.
11.3.4 If segmentation is used to isolate the CDE from other networks, perform penetration tests at least annually and after any changes to segmentation controls/methods to verify that the segmentation methods are operational and effective, and isolate all out-of-scope systems from systems in the CDE.

[That section of PCI DSS provides a useful set of requirements for anyone conducting a penetration test. It’s also a nice blueprint for penetration testing, even for organizations that don’t have PCI DSS compliance obligations.

The standard goes on to include four additional requirements that describe the frequency and scope of penetration tests:

**11.3.1. Perform external penetration testing at least annually and afterany significant infrastructure or application upgrade or modification(such as an operating system upgrade, a sub-network added to theenvironment, or a web server added to the environment).
**11.3.2 Perform internal penetration testing at least annually and after anysignificant infrastructure or application upgrade or modification (such asan operating system upgrade, a sub-network added to the environment, ora web server added to the environment).
11.3.3. Exploitable vulnerabilities found during penetration testing are corrected and the testing is repeated to verify the corrections.
11.3.4 If segmentation is used to isolate the CDE from other networks,perform penetration tests at least annually and after any changes tosegmentation controls/methods to verify that the segmentation methodsare operational and effective, and isolate all out-of-scope systems fromsystems in the CDE.

[Penetration tests may be conducted by either internal teams, comprising cybersecurity employees from the organization being tested, or external teams, comprising contractors.

Internal penetration testing teams consist of cybersecurity professionals from within the organization who conduct penetration tests on the organization's systems and applications.

There are **two** major benefits of using internal teams to conduct penetration testing. **First**, they have contextual knowledge of the organization that can improve the effectiveness of testing by providing enhanced subject matter expertise. **Second,** it's generally less expensive to conduct testing using internal employees than it is to hire a penetration testing fi rm, provided that you have enough work to keep your internal team busy!
The primary disadvantages to using internal teams to conduct penetration testing stem from the fact that you are using internal employees. These individuals may have helped to design and implement the security controls that they are testing, which may introduce conscious or unconscious bias toward demonstrating that those controls are secure. Similarly, the fact that they were involved in designing the controls may make it more difficult for them to spot potential flaws that could provide a foothold for an attacker.

If at all possible, the penetration testing team should be organizationally separate from the cybersecurity team that designs and operates controls. However, this is usually not possible in any but the largest organizations due to staffi ng constraints.

External penetration testing teams are hired for the express purpose of performing a penetration test.

External penetration testing teams also generally bring a much higher degree of independence than internal teams. However, organizations using an external team should still

be aware of any potential confl icts of interest the testers may have. It might not be the best idea to hire the cybersecurity consultants that helped you design and implement your security controls to perform an independent test of those controls. They may be inclined to feel that any negative report they provide is a reflection on the quality of their own work.

Penetration testing is not a one-time process. While organizations may wish to require penetration testing for new systems upon deployment, it is important to repeat those tests on a periodic basis for three reasons.

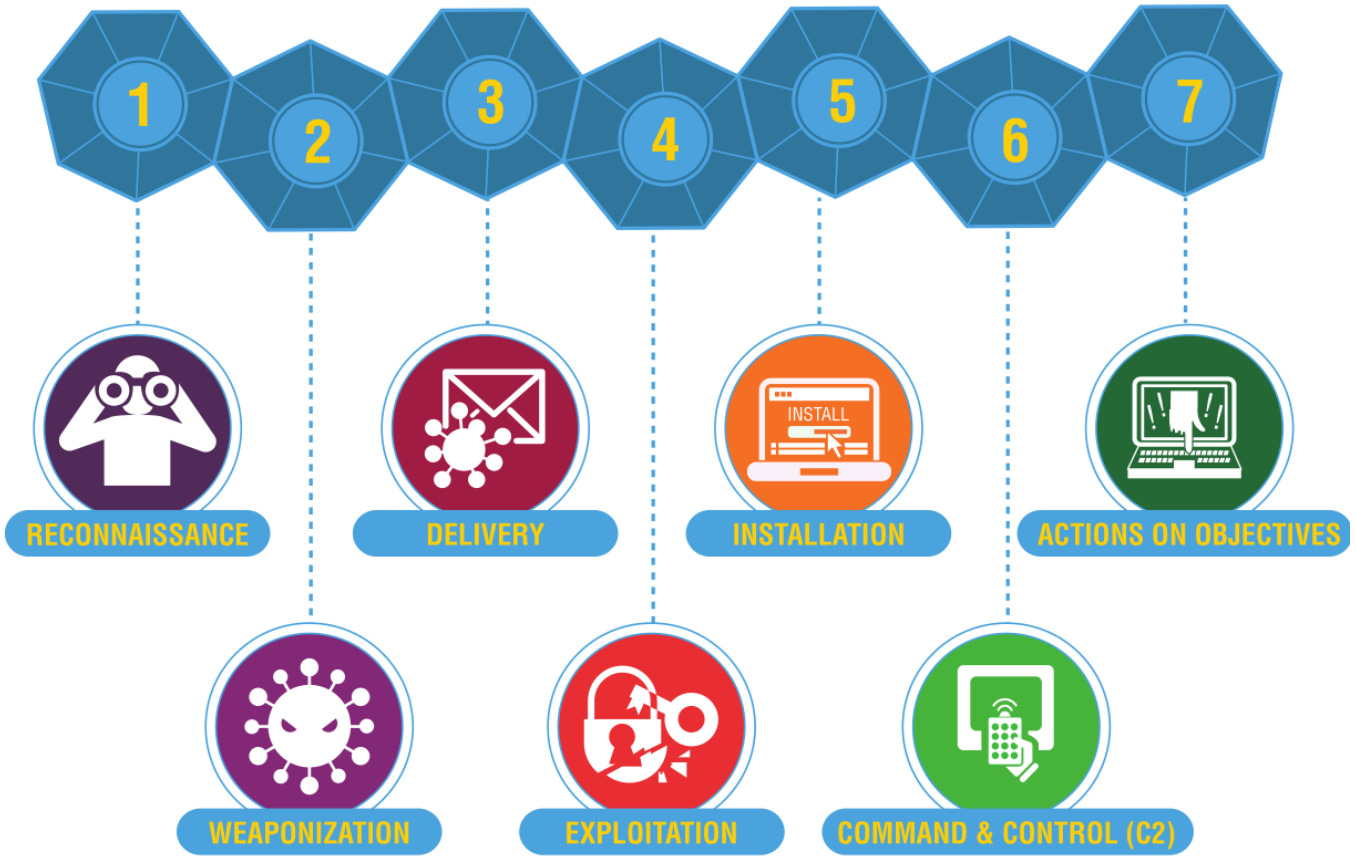
First, the technology environment changes. Systems are reconfigured, patches are applied, updates and tweaks are made on a regular basis. Considered in isolation, each of these changes may have only a minor impact on the environment and may not reach the threshold for triggering a “significant change” penetration test, but collectively they may change the security posture of the environment. Periodic penetration tests have a good chance of detecting security issues introduced by those environmental changes

Testers and their clients must have a
clear understanding of what will occur during the penetration test, outline clear rules of engagement, and decide what systems, data, processes, and activities are within the authorized scope of the test. There’s a fine line between penetration testing and hacking, and a written statement of work that includes clear authorization for penetration testing activities is crucial to ensuring that testers stay on the right side of the law and meet client expectations.

We cover this topic in great detail in Chapter 2. Specifically, you’ll learn how to meet the four objectives of this domain:

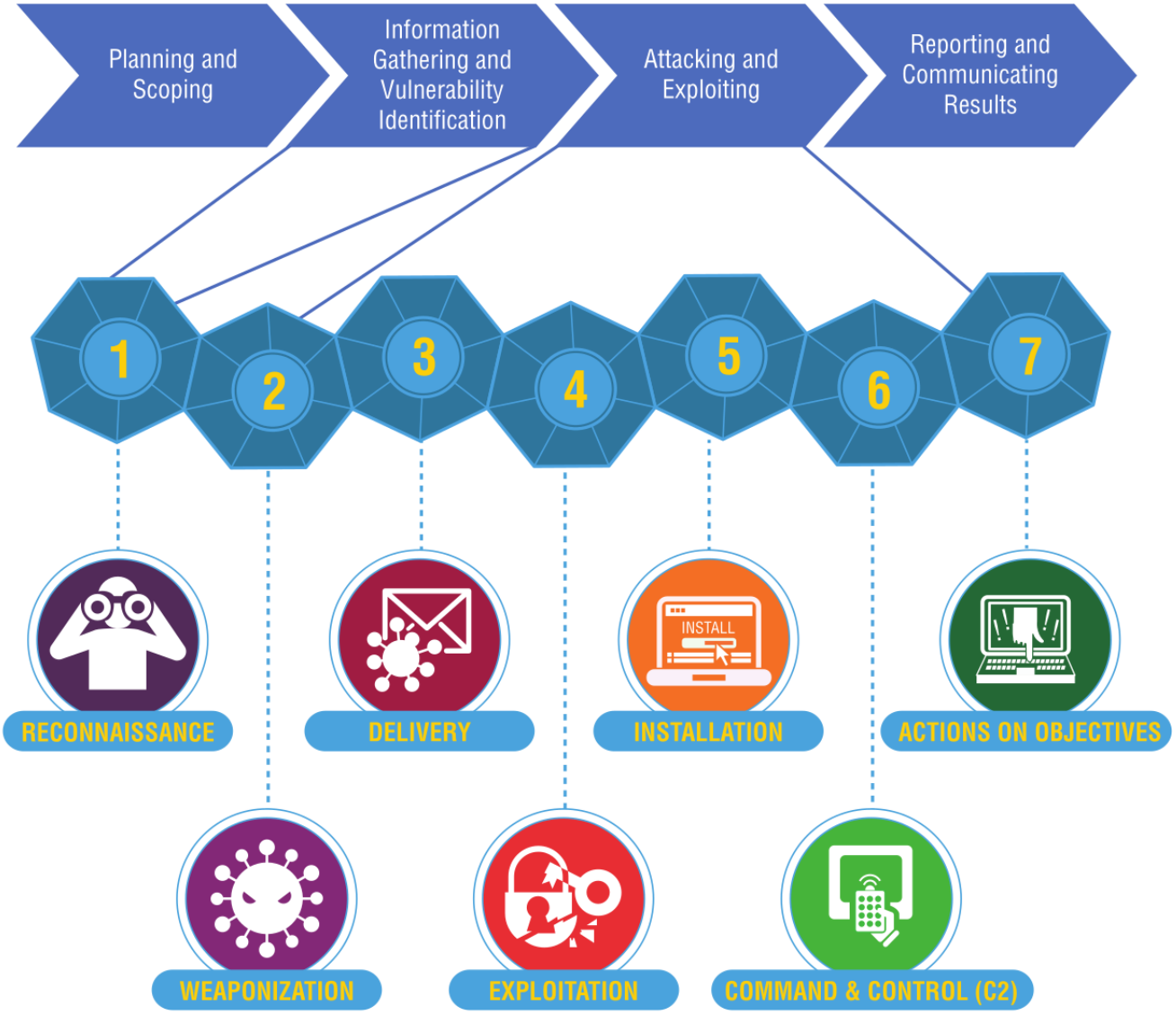
■■Explain the importance of planning for an engagement.
■■Explain key legal concepts.
■■Explain the importance of scoping an engagement properly.
■■Explain the key aspects of compliance-based assessments.

FIGURE 1.4 The Cyber Kill Chain model



Source: Lockheed Martin

FIGURE 1.5 Cyber Kill Chain in the context of the CompTIA model



Reconnaissance

The reconnaissance phase of the Cyber Kill Chain maps directly to the Information Gathering and Vulnerability Identification phase of the penetration testing process. During this phase, attackers gather open-source intelligence and conduct initial scans of the target environment to detect potential avenues of exploitation.

Weaponization

After completing the Reconnaissance phase of an attack, attackers move into the remaining six steps, which expand upon the Attacking and Exploiting phase of the penetration testing process.16

The first of these phases is **Weaponization.** During this stage, the attackers develop a specific attack tool designed to exploit the vulnerabilities identified during reconnaissance. They often use automated toolkits to develop a malware strain specifically tailored to infiltrate their target.

Delivery

After developing and testing their malware weapon, attackers next must deliver that malware to the target. This may occur through a variety of means, including exploiting a network or application vulnerability, conducting a social engineering attack, distributing malware on an infected USB drive or other media, sending it as an email attachment, or through other means.

Exploitation

Once the malware is delivered to the target organization, the attacker or the victim takes some action that triggers the malware's payload, beginning the Exploitation phase of the Cyber Kill Chain. During this phase, the malware gains access to the targeted system. This may occur when the victim opens a malicious file or when the attacker exploits a vulnerability over the network or otherwise gains a foothold on the target network.

Installation

The initial malware installation is designed only to enable temporary access to the target system. During the next phase of the Cyber Kill Chain, Installation, the attacker uses the initial access provided by the malware to establish permanent, or persistent, access to the target system. For this reason, many people describe the objective of this phase as establishing persistence in the target environment. Attackers may establish persistence by creating a back door that allows them to return to the system at a later date, by creating Registry entries that reopen access once an administrator closes it, or by installing a web shell that allows them to access the system over a standard HTTPS connection.

Command and Control

After establishing persistent access to a target system and network, the attacker may then use a remote shell or other means to remotely control the compromised system. The attacker may manually control the system using the shell or may connect it to an automated command-and-control (C2C) network that provides it instructions. This automated approach is common in distributed denial of service (DDoS) attacks where the attacker simultaneously directs the actions of thousands of compromised systems, known as a botnet.

Ta b l e 1 .1

Penetration testing tools covered by the PenTest+ exam

Scanners

Credential Testing Tools

Nikto Hashcat

OpenVA SMedusa

sqlmap Hydra

Nessus CeWL

Nmap John the Ripper

Cain and Abel

OSINT

Mimikatz

WHOIS Patator
Nslookup DirBuster
FOCA W3AF
theHarvester
Shodan
Maltego
Recon-ng
Censys
Remote Access Tools
Secure Shell (SSH)
Ncat
Netcat
Proxychains
Wireless
Aircrack-ng
Kismet
WiFite
Networking Tools
Wireshark
Hping
Debuggers
OllyDbg

Immunity Debugger AFL
GDB
SonarQube
WinDbg
YASCA
19
IDA
Social Engineering Tools
Web Proxies
OWASP ZAP
SET
BeEF
Burp Suite
Miscellaneous Tools
Mobile Tools
Drozer
APKX
APK Studio
Software Assurance
FindBugs/find-sec-bugs

SearchSploit
PowerSploit
Responder
Impacket
Empire
Metasploit framework
Peach
If attackers aren't able to gain access to credentials through social engineering techniques, they may be able to use tools to reverse engineer hashed passwords.
The PenTest+ exam includes coverage of a large set of tools designed to assist with these activities:
■■
■■

■■
Hashcat, **John the Ripper**, **Hydra, Medusa, Patator, and Cain and Abel are password** cracking tools used to reverse engineer hashed passwords stored in files.
CeWL is a custom wordlist generator that searches websites for keywords that may be used in password guessing attacks.
Mimikatz retrieves sensitive credential information from memory on Windows systems.
DirBusterr is a brute-forcing tool used to enumerate files and directories on a web server.
We'll cover all of these tools in more detail in Chapter 10, "Exploiting Host Vulnerabilities."

Debugging tools provide insight into software and assist with reverse engineering activities. Penetration testers preparing for the exam should be familiar with five debugging tools:
■■
■■
Immunity Debugger is designed specifically to support penetration testing and the reverse engineering of malware.
GDB is a widely used open-source debugger for Linux that works with a variety of programming languages.22

OllyDbg is a Windows debugger that works on binary code at the assembly language level.
WinDbg is another Windows-specific debugging tool that was created by Microsoft.
IDA is a commercial debugging tool that works on Windows, Mac, and Linux platforms.
In addition to decompiling traditional applications, penetration testers also may find themselves attempting to exploit vulnerabilities on mobile devices.
You should be familiar
with three mobile device security tools for the exam.

■■**Drozer** is a security audit and attack framework for Android devices and apps.
■■**APKX and APK** Studio decompile Android application packages (APKs).
We'll provide detailed coverage of these tools in Chapter 9, "**Exploiting Application Vulnerabilities."**

Software Assurance

In addition to debuggers, penetration testers also make use of other software assurance and testing tools. Some that you'll need to be familiar with for the exam include:

FindBugs and find-sec-bugss are Java software testing tools that perform static analysis of code.

Peach and AFL are fuzzing tools that generate artificial input designed to test applications.

SonarQube is an open-source continuous inspection tool for software testing.

YASCA (Yet Another Source Code Analyzer) is another open-source software testing tool that includes scanners for a wide variety of languages. YASCA leverages FindBugs, among other tools.

You'll learn more about each of these tools in Chapter 9, "Exploiting Application Vulnerabilities."

Network Testing

In addition to exploiting software vulnerabilities, penetration testers also often exploit flaws in networks as they seek access to systems.

Wireshark is a protocol analyzer that allows penetration testers to eavesdrop on and dissect network traffic.

Hping is a command-line tool that allows testers to artificially generate network traffic.

Aircrack-ng, WiFite, and Kismet are wireless network security testing tools.

You'll learn more about each of these tools in Chapter 7, "Exploiting Network Vulnerabilities."

Remote Access

After gaining initial access to a network, penetration testers seek to establish persistence so that they may continue to access a system. These are some of the tools used to assist with this task:

Secure Shell (SSH) provides secure encrypted connections between systems.

Ncat and Netcat provide an easy way to read and write data over network connections.

Proxychains allows testers to force connections through a proxy server where they may be inspected and altered before being passed on to their final destination.

You'll learn more about each of these tools in Chapter 10, "Exploiting Host Vulnerabilities."

Exploitation

As attackers work their way through a network, they use a variety of exploits to compromise new systems and escalate the privileges they have on systems they've already compromised.

Exploitation toolkits make this process easy and automated. For the exam, you should be familiar with the following exploitation tools:

Metasploit is, by far, the most popular exploitation framework and supports thousands of plug-ins covering different exploits.
SearchSploit is a command-line tool that allows you to search through a database of known exploits.
PowerSploit and Empire are Windows-centric sets of PowerShell scripts that may be used to automate penetration testing tasks.
Responder is a toolkit used to answer NetBIOS queries from Windows systems on a network.
Impacket is a set of network tools that provide low-level access to network protocols.
You'll learn more about each of these tools in Chapter 6, "Exploit and Pivot."
Summary
Penetration testing is an important practice that allows cybersecurity professionals to assess the security of environments by adopting the **hacker mind-set**. By thinking like an attacker, testers are able to identify weaknesses in the organization's security infrastructure and potential gaps that may lead to future security breaches.
The CompTIA penetration testing process includes four phases: Planning and Coping, Information Gathering and Vulnerability Identification, Attacking and Exploiting, and24

Reporting and Communicating Results. Penetration testers follow each of these phases to ensure that they have a well-designed test that operates using agreed-upon rules of engagement.
Penetration testers use a wide variety of tools to assist in their work. These are many of the same tools used by cybersecurity professionals, hackers, network engineers, system administrators, and software developers. Tools assist with all stages of the penetration testing process, especially information gathering, vulnerability identification, and exploiting vulnerabilities during attacks.

The first step in most penetration testing engagements is **determining what should be tested, or the scope of the assessment.** [The scope of the assessment determines what pen-etration testers will do and how their time will be spent.

Determining the scope requires working with the person or organization for whom the pen-etration test will be performed. Testers need to understand all of the following: why the test is being performed; whether specific requirements such as compliance or business needs are driv-ing the test; what systems, networks, or services should be tested and when; what informationcan and cannot be accessed during testing; what the rules of engagement for the test are; what echniques are permitted or forbidden; and to whom the final report will be presented.

Assessment Types
There are quite a few ways to categorize and describe assessments, but it helps to have some

broad categories to sort them into. The PenTest+ exam objectives describe three major types of assessment:

Goals-based or objectives-based assessments are conducted for specific reasons. Examples include

--validation of a new security design,

--testing an application or service

--infrastructure before it enters production, and

--assessing the security of an organization

that has recently been acquired.

Compliance-based assessments are[designed around the compliance objectives of a law, standard, or other guidance and may require engaging a specific provider or assessor that is certified to perform the assessment.

Red-team assessments are typically more targeted than normal penetration tests. Red teams attempt to act like an attacker, targeting sensitive data or systems with the goal of acquiring data and access. Unlike other types of penetration tests, red-team assessments are not intended to provide details of all of the security flaws a target has. This means that red-team assessments are unlikely to provide as complete a view of flaws in the environment, but they can be very useful as a security exercise to train incident responders or to help validate security designs and practices.

Red teams test the effectiveness of a security program or system by acting like attackers. Red teams are sometimes called tiger teams. **Blue teams are defenders and may operate against red teams or actual attackers.**

Some security professionals also describe other colors of teams, such as **purple teams that work to integrate red- and blue-team efforts to improve organizational security, white teams that control the environment during an exercise, or green teams that tackle long-term vulnerability remediation or act as trainers.**

[White Box, Black Box, or Gray Box?

Once the type of assessment is known, one of the fi rst things to decide about a penetration test is how much knowledge testers will have about the environment. There are three typical classifications that are used to describe this:

■✓

White box tests, sometimes called “crystal box” or “full knowledge” tests, as in you see everything inside, are performed with full knowledge of the underlying technology, configurations, and settings that make up the target. Testers will typically have information including network diagrams, lists of systems and IP network ranges, and even credentials to the systems they are testing. White box tests allow effective testing of systems without requiring testers to spend time identifying targets and determining which of them may allow a way in. This means that a white box test is often more Scoping and Planning Engagements

complete, as testers can get to every system, service, or other target that is in scope and

will have credentials and other materials that will allow them to be tested. Of course, since testers can see everything inside an environment, they may not provide an accurate view of what an external attacker would see, and controls that would have been effective against most attackers may be bypassed.

■■

■■

Black box tests, sometimes called **“zero knowledge” tests**, are intended to **replicate what an attacker would encounter.** Testers are not provided with access to or information about an environment, and instead, they must gather information, discover vulnerabilities, and make their way through an infrastructure or systems as an attacker would.

This can be time-consuming for the penetration tester, but it can better reveal what vulnerabilities might be exploited by someone starting with nothing. It can also help provide a reasonably accurate assessment of how secure the target is against an attacker of similar or lesser skill. It is important to note that the quality and skill set of your penetration tester or team is very important when conducting a black box penetration test—if the threat actor you expect to target your organization is more capable, a black box tester can’t provide you with a realistic view of what they could do.

Gray box tests are a **blend of black box and white box testing.** A gray box test may provide some information about the environment to the penetration testers without giving full access, credentials, or configuration details. [A gray box test can help focus penetration testers’ time and effort while also providing a more accurate view of what an attacker would actually encounter.

Understanding Your Adversaries

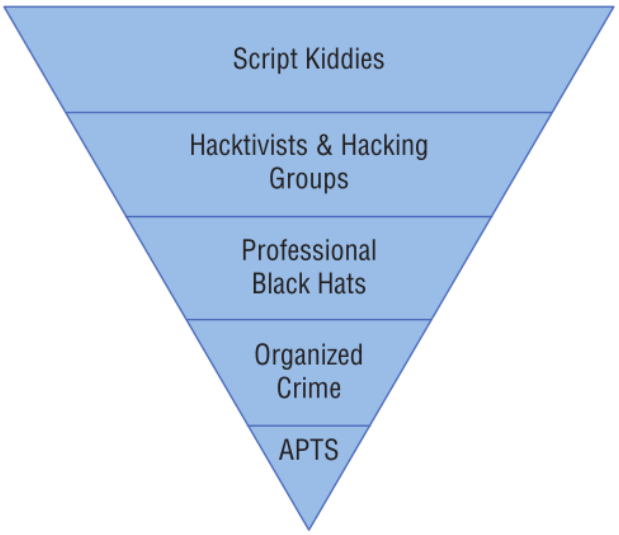
When an organization conducts a **black box** penetration test, one of the first questions it will ask is, **Who would attack us and why?** Answering that question can help management make decisions about how a penetration test is conducted, what techniques are considered in the engagement, the scope of the test, and who they will hire to conduct it.

Threat actors are often rated by their capabilities. For example, script kiddies and casual hackers use prebuilt tools to conduct their attacks, and most organizations will consider their attacks nuisance-level threats. But as you continue down the threat actors adversary tiers shown in Figure 2.1, capabilities and resources, and thus the threat an adversary poses, increase. As professional hackers, organized crime, and nation-state–level attackers like advanced persistent threats (APTs) enter your threat radar, the likelihood of a successful attack and compromise increases. This means that you should assume that a breach will occur and plan accordingly!

Each of these potential adversaries is likely to have a different intent: hacktivists may want to make a political or social point, while black hats and organized crime are likely to

have a profit motive. APT actors are usually focused on a nation-state's goals, with other attacks driven by that purpose.

FIGURE 2.1 Adversary tiers



The Rules of Engagement

Once you have determined the type of assessment and the level of knowledge testers will have about the target, the rest of the rules of engagement (RoE) can be written. Key elements include these:

[The timeline for the engagement and when testing can be conducted. Some assessments will intentionally be scheduled for noncritical time frames to minimize the impact of potential service outages, while others may be scheduled during normal business hours to help test the organization's reaction to attacks.

What locations, systems, applications, or other potential targets are included or excluded**this also often includes discussions about third-party service providers that may be impacted by the test, such as Internet service providers, Software as a Service**or other cloud service providers, or outsourced security monitoring services. Any special technical constraints should also be discussed in the RoE.

Data handling requirements for information gathered during the penetration test. This is particularly important when engagements cover sensitive organizational data or systems. Penetration tests cannot, for example, legally expose protected health information (PHI), even under an NDA.

Requirements for handling often include **confidentiality requirements** for the findings, such as **encrypting data during and after the test,** [and contractual require-ments for disposing of the penetration test data and results after the engagement is over.

***What behaviors to expect from the target. Defensive behaviors like shunning, blacklisting, or other active defenses may limit the value of a penetration test. If the test is meant to evaluate defenses, this may be useful. If the test is meant to test a complete

infrastructure, shunning or blocking the penetration testing team's efforts can waste time and resources.

[***What resources are committed to the test. In white and gray box testing scenarios, time commitments from the administrators, developers, and other experts on the targets ofthe test are not only useful, they can be necessary for an effective test.
Legal concerns should also be addressed, including a synopsis of any regulatory concerns affecting the target organization, pentest team, any remote locations, and any service providers who will be in-scope.
When and how communications will occur. Should the engagement include daily or weekly updates regardless of progress, or will the penetration testers simply report out when they are done with their work?
Whom to contact in case of particular events, such as evidence of ongoing compromise, accidental breach of RoE, a critical vulnerability discovered, and other events that warrant immediate attention.
Who is permitted to engage the pentest team; for example, can the CFO request an update? Including this in RoE helps avoid potentially awkward denials.

***The tools and techniques we will cover in this book are the bread and butter of a penetration tester's job, but they are very likely illegal to use on another owner's equipment without permission. Before you plan (and especially before you execute) a penetration test, you must have appropriate permission. In most cases, you should be sure to have appro-priate documentation for that permission in the form of a signed agreement, a memo rom senior management, or a similar "get out of jail free" card from a person or people in the target organization with the rights to give you permission.
Why is it called a "get out of jail free" card? It's the document that you would produce if something went wrong. Permission from the appropriate party can help you stay out of trouble if something goes wrong!

Scoping agreements and the **rules of engagement** must define more than just what willbe tested. In fact, **documenting** the limitations of the test can be just as important as docu-menting what will be included.
The testing agreement or scope documentation should contain disclaimers explaining that the test is valid only at the point in time when it is conducted and that the scope and methodology chosen can impact the comprehensiveness of the test. After all, **a white box penetration test is** far more likely to find issues buried layers deep in a design than a **black box test of well-secured systems!**
***Problem handling and resolution is another key element of the rules of engagement.

While penetration testers and clients always hope that the tests will run smoothly and won't cause any disruption, testing systems and services, particularly in production environments using actual attack and exploit tools, can cause outages and other problems. [In those cases, having a clearly defined communication, notification, and escalation path on

both sides of the engagement can help minimize downtime and other issues for the target organization.

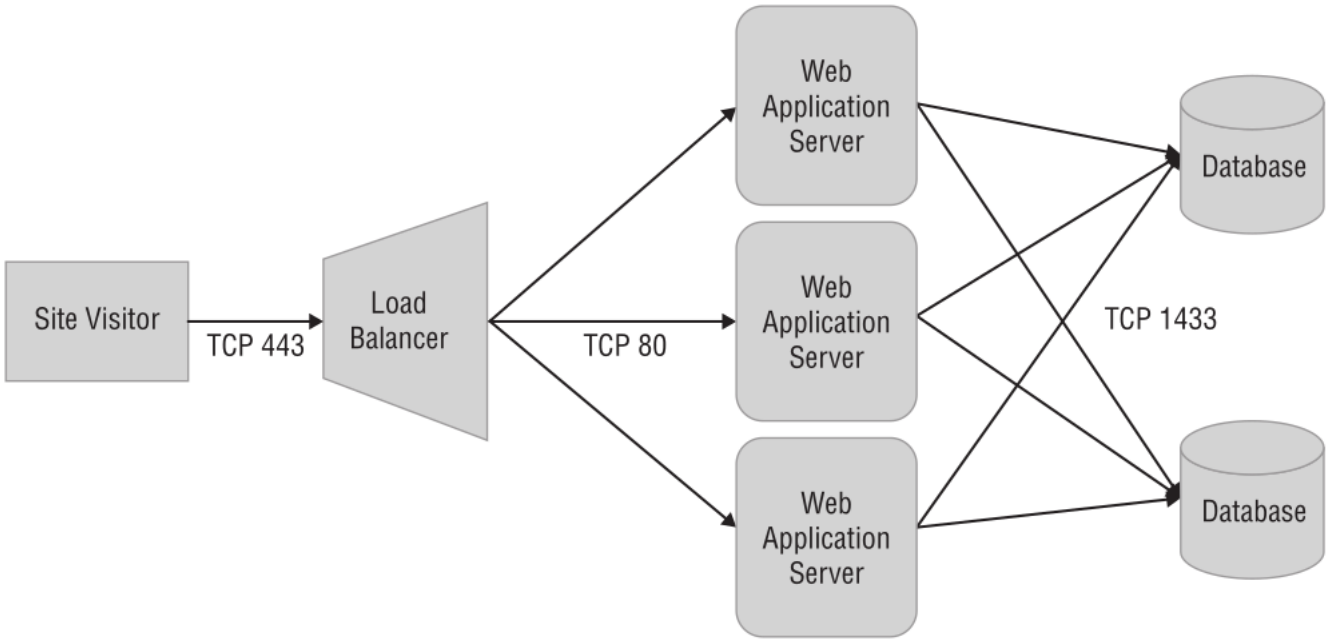
Penetration testers should carefully document their responsibilities and limitations of liability and ensure that clients know what could go wrong and that both sides agree on how it should be handled. This ensures that both the known and unknown impacts of the test can be addressed appropriately.

Scoping Considerations: A Deeper Dive

As you've likely already realized, determining the detailed scope of a test can involve a significant amount of work! Even a small organization may have a complex set of systems,applications, and infrastructure, and determining the scope of a penetration test can be challenging unless the organization has detailed and accurate architecture, dataflow, and system documentation. Of course, if the engagement is a black box test, the detail available to penetration testers may be limited, so they will need to know how to avoid going outside of the intended scope of the test.

Detailed scoping starts by determining the acceptable targets. Are they internally or externally hosted, and are they on site or off site? Are they hosted by the organization itself, by a third party, or by an Infrastructure as a Service or other service provider? Are they virtual, physical, or a hybrid, and does this impact the assessment?

Equally important is an understanding of what applications, services, and supporting infrastructure are in scope. It may be desirable or necessary to target elements of infrastructure or systems that are not directly related to the target to access the target. [For example, one of the authors of this book targeted the network administration infrastructure for an organization to gain access to the real target of the test he was conducting—a database server that was otherwise too well protected by firewalls. With access to network administration functions, he was able to pivot and get access to unencrypted dataflows between the database and application server that were his real target as shown in figure 2.2



User accounts and privileged accounts are both commonly part of penetration tests, and they can be some of the most important targets for penetration testers. That means determining which accounts are in scope and which aren't. For a black box penetration tester, limitations on accounts can create challenges if you aren't allowed to use an account that you may be able to access. Of course, with a white box test (and possibly with a gray box test), you should have access to the accounts you need to perform the test.
**Wireless and wired network scoping often comes into play for penetration testers who will conduct on-site work, or when the network itself is in scope. Thus it's important to know which SSIDs belong to your target and which are valid targets. At the same time, knowing which subnets or IP ranges are in scope is also key to avoid targeting third parties or otherwise going outside of the penetration test's scope.
***It is important to keep careful logs of the actions you take while conducting a penetration test. That way, if a problem occurs, you can show what was going on at that time. The authors of this book have used their logs to demonstrate which systems were being vulnerability scanned when a service crashed in multiple cases. In some, the scanner wasn't the cause; in others it was, showing that the service wasn't up to being scanned!

As you work through all of the details for a scoping exercise, [you should also make sure you have an in-depth discussion about the target organization's risk acceptance and company policies.

***Are the organization and the sponsor ready and able to accept that a penetration test could cause an outage or service disruption?
If not, is there a way to conduct the
test in a way that will either minimize risk or prevent it? What is the organization's impact tolerance? Is a complete outage acceptable as part of the test? What if an account lockout happens? Is there a particular time of day or part of the business or recurring IT mainte-nance cycle when a test would be less disruptive?

***The PenTest+ objectives specifically call out pre-merger and supply chain tests as business areas that a penetration tester may be asked to review. In pre-merger scenarios, the penetration test is typically intended to help the acquiring company understand the securitycapabilities and status of the acquired company. Supply chain testing, on the other hand,is usually targeted at companies and organizations that the client organization wants to review to determine if suppliers have effective security controls in place. It is common prac-tice to ask suppliers to provide audit and assessment documentation, so you might also beasked to provide an assessment suitable for sharing with prospective customers or partners.
In addition to these specific business reasons, a complete scope review for a customer or organization is likely to include at least some discussion of business processes and prac-tices that the tester may encounter. These could include administrative processes, accountmanagement, or any other business process that the tester might target or disrupt as partof their testing process. As a penetration tester, make sure that you discuss the potential formpact, and inquire about any processes that should be treated with care or avoided.

Scope creep, or the addition of more items and targets to the scope of the assessment, is a constant danger for penetration tests. During the scoping phase, you are unlikely to know all of the details of what you may uncover, and during the assessment itself you may encounter unexpected new targets. It is important to ensure that you have planned for thiswith the sponsor of the penetration test and know how you will handle it. They may opt to retain the original scope, engage you to perform further work, or request an estimate onthe new scope.

***[Support Resources for Penetration Tests

Penetration testers can take advantage of internal documentation to help plan their testing (and black box testers may manage to acquire this documentation during their work!). While there are a multitude of possible documents that each organization may have, documentation, accounts and access, and budget are all specifically described in the PenTest+ objectives.

Documentation

The documentation that an organization creates and maintains to support its infrastructure and services can be incredibly useful to a penetration tester. While there are a multitude of possible documents that each organization may have, a few of the most common are described in the PenTest+ objectives, including these:

■■

XML documentation like Web Services Description Language (**WSDL**), Web Application Description Language **(WADL), SOAP,** or other **XML-**based schema definitions. There are a multitude of XML-based standards that penetration testers may encounter. Fortunately, XML code is usually reasonably human-readable, and you should be able to get a general idea of what the definition or documentation describes by reading through it. Figure 2.3 shows an example of Amazon's Product Advertising WSDL (found at http://webservices.amazon.com/AWSECommerceService/AWSECommerceService.wsdl), which shows value types, operation definitions, and request/response formats.

Figure 2.3

An example of an API WSDL

```
<xs:element name="ItemSearch">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="MarketplaceDomain" type="xs:string" minOccurs="0"/>
      <xs:element name="AWSAccessKeyId" type="xs:string" minOccurs="0"/>
      <xs:element name="AssociateTag" type="xs:string" minOccurs="0"/>
      <xs:element name="XMLEscaping" type="xs:string" minOccurs="0"/>
      <xs:element name="Validate" type="xs:string" minOccurs="0"/>
      <xs:element name="Shared" type="tns:ItemSearchRequest" minOccurs="0"/>
      <xs:element name="Request" type="tns:ItemSearchRequest" minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```