Ethical Hacking Class part 2

Introduction

The Transmission Control Protocol/Internet Protocol (TCP/IP) suite is so dominant and important to ethical hacking that it is given wide coverage in this lesson. Many tools, attacks, and techniques that will be covered throughout this class are based on the use and misuse of TCP/IP protocol suite. Understanding its basic functions will advance your security skills. This lesson also spends time reviewing the attacker's process and some of the better known methodologies used by ethical hackers.

The Attacker's Process

Objective:

State the process or methodology hackers use to attack networks

Attackers follow a fixed methodology. To beat a hacker, you have to think like one, so it's important to understand the methodology. The steps a hacker follows can be broadly divided into six phases, which include pre-attack and attack phases:

1. Performing Reconnaissance
2. Scanning and enumeration
3. Gaining access
4. Escalation of privilege
5. Maintaining access
6. Covering tracks and placing backdoors

NOTE

A denial of service (DoS) might be included in the preceding steps if the attacker has no success in gaining access to the targeted system or network. Let's look at each of these phases in more detail so that you better understand the steps.

Performing Reconnaissance

Reconnaissance is considered the first pre-attack phase and is a systematic attempt to locate, gather, identify, and record information about the target. The hacker seeks to find out as much information as possible about the victim. This first step is considered a passive information gathering. As an example, many of you have probably seen a detective movie in which the policeman waits outside a suspect's house all night and then follows him from a distance when he leaves in the car. That's reconnaissance; it is passive in nature, and, if done correctly, the victim never even knows it is occurring.

Hackers can gather information in many different ways, and the information they obtain allows them to formulate a plan of attack. Some hackers might dumpster dive to find out more about the victim. Dumpster diving is the act of going through the victim's trash. If the organization does not have good media control policies, many types of sensitive information will probably go directly in the trash. Organizations should inform employees to shred sensitive information or dispose of it in an approved way.

Don't think that you are secure if you take adequate precautions with paper documents. Another favorite of the hacker is social engineering. A social engineer is a person who can smooth talk other individuals into revealing sensitive information. This might be accomplished by calling the help desk and asking someone to reset a password or by sending an email to an insider telling him he needs to reset an account.

If the hacker is still struggling for information, he can turn to what many consider the hacker's most valuable reconnaissance tool, the Internet. That's right; the Internet offers the hacker a multitude of possibilities for gathering information. Let's start with the company website. The company website might have key employees listed, technologies used, job listings probably detailing software and hardware types used, and some sites even have databases with employee names and email addresses.

Scanning and Enumeration

Scanning and enumeration is considered the second pre-attack phase. Scanning is the active step of attempting to connect to systems to elicit a response. Enumeration is used to gather more in-depth information about the target, such as open shares and user account information. At this step in the methodology, the hacker is moving from passive information gathering to active information gathering. Hackers begin injecting packets into the network and might start using scanning tools such as Nmap. The goal is to map open ports and applications. The hacker might use techniques to lessen the chance that he will be detected by scanning at a very slow rate. As an example, instead of checking for all potential applications in just a few minutes, the scan might take days to verify what applications are running. Many organizations use intrusion detection systems(IDS) to detect just this type of activity. Don't think that the hacker will be content with just mapping open ports. He will soon turn his attention to grabbing banners. He will want to get a good idea of what type of version of software applications you are running. And, he will keep a sharp eye out for down-level software and applications that have known vulnerabilities. An example of down-level software would be Windows 95.

One key defense against the hacker is the practice of deny all. The practice of the deny all rule can help reduce the effectiveness of the hacker's activities at this step. Deny all means that all ports and applications are turned off, and only the minimum number of applications and services are turned on that are needed to accomplish the organization's goals.

Unlike the elite black hat hacker who attempts to remain stealth, script kiddies might even use vulnerability scanners such as Nessus to scan a victim's network. Although the activities of the black hat hacker can be seen as a single shot in the night, the script kiddies scan will appear as a series of shotgun blasts, as their activity will be loud and detectable. Programs such as Nessus are designed to find vulnerabilities but are not designed to be a hacking tool; as such, they generate a large amount of detectable network traffic.

TIP

The greatest disadvantage of vulnerability scanners is that they are very noisy.

Gaining Access

As far as potential damage, this could be considered one of the most important steps of an attack. This phase of the attack occurs when the hacker moves from simply probing the network to actually attacking it. After the hacker has gained access, he can begin to move from system to system, spreading his damage as he progresses.

Access can be achieved in many different ways. A hacker might find an open wireless access point that allows him a direct connection or the help desk might have given him the phone number for a modem used for out-of-band management. Access could be gained by finding a vulnerability in the web server's software. If the hacker is really bold, he might even walk in and tell the receptionist that he is late for a meeting and will wait in the conference room with network access. Pity the poor receptionist who unknowingly provided network access to a malicious hacker. These things do happen to the company that has failed to establish good security practices and procedures.

The factors that determine the method a hacker uses to access the network ultimately comes down to his skill level, amount of access he achieves, network architecture, and configuration of the victim's network.

Escalation of Privilege

Although the hacker is probably happy that he has access, don't expect him to stop what he is doing with only a "Joe user" account. Just having the access of an average user probably won't give him much control or access to the network. Therefore, the attacker will attempt to escalate himself to administrator or root privilege. After all, these are the individuals who control the network, and that is the type of power the hacker seeks.

Privilege escalation can best be described as the act of leveraging a bug or vulnerability in an application or operating system to gain access to resources that normally would have been protected from an average user. The end result of privilege escalation is that the application performs actions that are running within a higher security context than intended by the designer, and the hacker is granted full access and control.

Maintaining Access

Would you believe that hackers are paranoid people? Well, many are, and they worry that their evil deeds might be uncovered. They are diligent at working on ways to maintain access to the systems they have attacked and compromised. They might attempt to pull down the etc/passwd file or steal other passwords so that they can access other user's accounts.

Rootkits are one option for hackers. A rootkit is a set of tools used to help the attacker maintain his access to the system and use it for malicious purposes. Rootkits have the capability to mask the hacker, hide his presence, and keep his activity secret. They will be discussed in detail later on in the class.

Sometimes hackers might even fix the original problem that they used to gain access, where they can keep the system to themselves. After all, who wants other hackers around to spoil the fun? Sniffers are yet another option for the hacker and can be used to monitor the activity of legitimate users. At this point, hackers are free to upload, download, or manipulate data as they see fit.

Covering Tracks and Placing Backdoors

Nothing happens in a void, and that includes computer crime. Hackers are much like other criminals in that they would like to be sure to remove all evidence of their activities. This might include using rootkits or other tools to cover their tracks. Other hackers might hunt down log files and attempt to alter or erase them.

Hackers must also be worried about the files or programs they leave on the compromised system. File hiding techniques, such as hidden directories, hidden attributes, and Alternate Data Streams (ADS), can be used. As an ethical hacker, you will need to be aware of these tools and techniques to discover their activities and to deploy adequate countermeasures.

Backdoors are methods that the hacker can use to reenter the computer at will. The tools and techniques used to perform such activities are discussed later on in the

class. At this point, what is important is to identify the steps.

The Ethical Hacker's Process

As an ethical hacker, you will follow a similar process to one that an attacker uses. The stages you progress through will map closely to those the hacker uses, but you will work with the permission of the company and will strive to "do no harm." By ethical hacking and assessing the organizations strengths and weaknesses, you will perform an important service in helping secure the organization. The ethical hacker plays a key role in the security process. The methodology used to secure an organization can be broken down into five key steps. Ethical hacking is addressed in the first:

1. Assessment
   Ethical hacking, penetration testing, and hands-on security tests.

2. Policy Development
   Development of policy based on the organization's goals and mission. The focus should be on the organization's critical assets.

3. Implementation
   The building of technical, operational, and managerial controls to secure key assets and data.

4. Training
   Employees need to be trained as to how to follow policy and how to configure key security controls, such as Intrusion Detection Systems (IDS) and firewalls.

5. Audit
   Auditing involves periodic reviews of the controls that have been put in place to provide good security. Regulations such as Health Insurance Portability and Accountability Act (HIPAA) specify that this should be done yearly.

All hacking basically follows the same six-step methodology discussed in the previous section: reconnaissance, scanning and enumeration, gaining access, escalation of privilege, maintaining access, and covering tracks and placing backdoors.

Is this all you need to know about methodologies? No, different organizations have developed diverse ways to address security testing. There are some basic variations

you should be aware of. These include National Institute of Standards and Technology 800-42, Threat and Risk Assessment Working Guide, Operational Critical Threat, Asset, fand Vulnerability Evaluation, and Open Source Security Testing Methodology Manual. Each is discussed next.

National Institute of Standards and Technology (NIST)

The NIST 800-42 method of security assessment is broken down into four basic stages that Include:

1. Planning
2. Discovery
3. Attack
4. Reporting

NIST has developed many standards and practices for good security. This methodology is contained in NIST 800-42. This is just one of several documents available to help guide you through an assessment. Find out more at http://csrc.nist.gov/publications/nistpubs.

Threat and Risk Assessment Working Guide (TRAWG)

The Threat and Risk Assessment Working Guide provides guidance to individuals or teams carrying out a Threat and Risk Assessment (TRA) for an existing or proposed IT system. This document helps provide IT security guidance and helps the user determine which critical assets are most at risk within that system and develop recommendations for safeguards. Find out more at http://www.cse-cst.gc.ca/publication.../itsg04-e.html.

Operational Critical Threat, Asset, and Vulnerability Evaluation (OCTAVE)

OCTAVE focuses on organizational risk and strategic, practice-related issues. OCTAVE is driven by operational risk and security practices. OCTAVE is self-

directed by a small team of people from the organization's operational, business units, and the IT department. The goal of OCTAVE is to get departments to work together to address the security needs of the organization. The team uses the experience of existing employees to define security, identify risks, and build a robust security strategy. Find out more at www.cert.org/octave.

Open Source Security Testing Methodology Manual (OSSTMM)

One well-known open sourced methodology is the OSSTMM. The OSSTMM divides security assessment into six key points known as sections. They are as follows:

* Physical Security
* Internet Security
* Information Security
* Wireless Security
* Communications Security
* Social Engineering

The OSSTMM gives metrics and guidelines as to how many man-hours a particular assessment will require. Anyone serious about learning more about security assessment should review this documentation. The OSSTMM outlines what to do before, during, and after a security test. Find out more at www.isecom.org/osstmm.

Security and the Stack

To really understand many of the techniques and tools that hackers use, you need to understand how systems and devices communicate. Hackers understand this, and many think outside the box when planning an attack or developing a hacking tool. As an example, TCP uses flags to communicate, but what if a hacker sends TCP packets with no flags set? Sure, it breaks the rules of the protocol, but it might allow the attacker to illicit a response to help identify the server. As you can see, having the ability to know how a protocol, service, or application works and how it can be manipulated can be beneficial.

The OSI model and TCP/IP are discussed in the next sections. Pay careful attention to the function of each layer of the stack, and think about what role each layer plays in the communication process.

The OSI Model

Objective:

Understand the Open Systems Interconnect (OSI) Model

Once upon a time, the world of network protocols was much like the Wild West. Everyone kind of did their own thing, and if there were trouble, there would be a shoot-out on Main Street. Trouble was, you never knew whether you were going to get hit by a stray bullet. Luckily, the IT equivalent of the sheriff came to town. This was the International Standards Organization (ISO). The ISO was convinced that there needed to be order and developed the Open Systems Interconnect (OSI) model in 1984. The model is designed to provide order by specifying a specific hierarchy in which each layer builds on the output of each adjacent layer. Although its role as sheriff was not widely accepted by all, the model is still used today as a guide to describe the operation of a networking environment.

There are seven layers of the OSI model: the Application, Presentation, Session, Transport, Network, Data Link, and Physical layers. The seven layers of the OSI model are shown in Figure 2.1, which overviews data moving between two systems up and down the stack, and described in the following list:

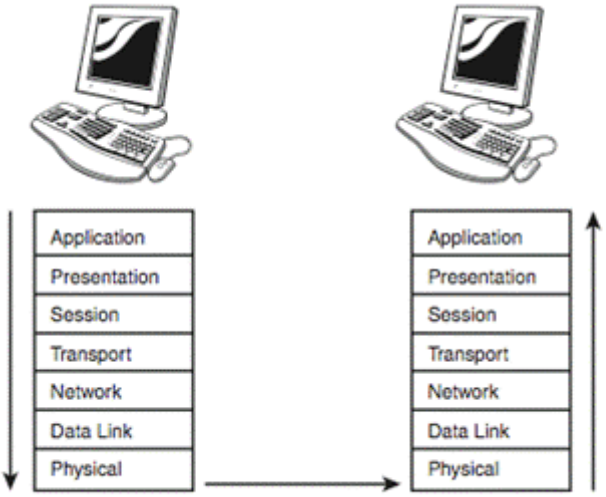

FIGURE 2.1 The OSI model.

Application layer

Layer 7 is known as the Application layer. Recognized as the top layer of the OSI model, this layer serves as the window for application services. The Application layer is one that most users are familiar with as it is the home of email programs, FTP, Telnet, web browsers, and office productivity suites, as well as many other applications. It is also the home of many malicious programs such as viruses, worms, Trojan horse programs, and other virulent applications.

Presentation layer

Layer 6 is known as the Presentation layer. The Presentation layer is responsible for taking data that has been passed up from lower levels and putting it into a format that Application layer programs can understand. These common formats include American Standard Code for Information Interchange (ASCII), Extended Binary-Coded Decimal Interchange Code (EBCDIC), and American National Standards Institute (ANSI). From a security standpoint, the most critical process handled at this layer is encryption and decryption. If properly implemented, this can help security data in transit.

Session layer

Layer 5 is known as the Session layer. Its functionality is put to use when creating, controlling, or shutting down a TCP session. Items such as the TCP connection establishment and TCP connection occur here. Session-layer protocols include items such as Remote Procedure Call and SQLNet from Oracle. From a security

standpoint, the Session layer is vulnerable to attacks such as session hijacking. A session hijack can occur when a legitimate user has his session stolen by a hacker. This will be discussed in detail in lesson 7, "Sniffers, Session Hijacking, and Denial of Service ".

Transport layer

Layer 4 is known as the Transport layer. The Transport layer ensures completeness by handling end-to-end error recovery and flow control. Transport-layer protocols include TCP, a connection-oriented protocol. TCP provides reliable communication through the use of handshaking, acknowledgments, error detection, and session teardown, as well as User Datagram Protocol (UDP), a connectionless protocol. UDP offers speed and low overhead as its primary advantage. Security concerns at the transport level include Synchronize(SYN) attacks, Denial of Service(DoS), and buffer overflows.

Network layer

Layer 3 is known as the Network layer. This layer is concerned with logical addressing and routing. The Network layer is the home of the Internet Protocol (IP), which makes a best effort at delivery of datagrams from their source to their destination. Security concerns at the network level include route poisoning, DoS, spoofing, and fragmentation attacks. Fragmentation attacks occur when hackers manipulate datagram fragments to overlap in such a way to crash the victim’s computer. IPSec is a key security service that is available at this layer.

Data Link layer

Layer 2 is known as the Data Link layer. The Data Link layer is responsible for formatting and organizing the data before sending it to the Physical layer. The Data Link layer organizes the data into frames. A frameis a logical structure in which data can be placed; it’s a packet on the wire. When a frame reaches the target device, the Data Link layer is responsible for stripping off the data frame and passing the data packet up to the Network layer. The Data Link layer is made up of two sub layers, including the logical link control layer (LLC) and the media access control layer (MAC). You might be familiar with the MAC layer, as it shares its name with the MAC addressing scheme. These 6-byte (48-bit) addresses are used to uniquely identify each device on the local network. A major security concern of the Data Link layer is the Address Resolution Protocol (ARP) process. ARP is used to resolve known Network layer addresses to unknown MAC addresses. ARP is a trusting

protocol and, as such, can be used by hackers for APR poisoning, which can allow them access to traffic on switches they should not have.

Physical layer

Layer 1 is known as the Physical layer. At Layer 1, bit-level communication takes place. The bits have no defined meaning on the wire, but the Physical layer defines how long each bit lasts and how it is transmitted and received. From a security standpoint, you must be concerned anytime a hacker can get physical access. By accessing a physical component of a computer network—such as a computer, switch, or cable—the attacker might be able to use a hardware or software packet snifferto monitor traffic on that network. Sniffers enable attacks to capture and decode packets. If no encryption is being used, a great deal of sensitive information might be directly available to the hacker.

TIP

For the exam, make sure that you know which attacks and defenses are located on each layer.

Anatomy of TCP/IP Protocols

Objectives:
Have a basic knowledge of the Transmission Control Protocol/Internet Protocol (TCP/IP) and their functionality Describe the basic TCP/IP frame structure

Four main protocols form the core of TCP/IP: the Internet Protocol (IP), the Transmission Control Protocol (TCP), the User Datagram Protocol (UDP), and the Internet Control Message Protocol (ICMP). These protocols are essential components that must be supported by every device that communicates on a TCP/IP network. Each serves a distinct purpose and is worthy of further discussion. The four layers of the TCP/IP stack are shown in Figure 2.2. The figure lists the Application, Host-to-host, Internet, and Network Access layers and describes the function of each.

TCP/IP is the foundation of all modern networks. In many ways, you can say that TCP/IP has grown up along with the development of the Internet. Its history can be traced back to standards adopted by the U.S. government’s Department of Defense

(DoD) in 1982. Originally, the TCP/IP model was developed as a flexible, fault tolerant set of protocols that were robust enough to avoid failure should one or more nodes go down. After all, the network was designed to these specifications to withstand a nuclear strike, which might destroy key routing nodes. The designers of this original network never envisioned the Internet we use today. Because TCP/IP was designed to work in a trusted environment, many TCP/IP protocols are now considered insecure. As an example, Telnet is designed to mask the password on the user's screen, as the designers didn't want shoulder surfers stealing a password; however, the password is sent in clear text on the wire. Little concern was ever given to the fact that an untrustworthy party might have access to the wire and be able to sniff the clear text password. Most networks today run TCP/IPv4. Many security mechanisms in TCP/IPv4 are add-ons to the original protocol suite. As the layers are stacked one atop another, encapsulation takes place. Encapsulation is the technique of layering protocols in which one layer adds a header to the information from the layer above. An example of this can be seen in Figure 2.3. This screenshot from a sniffer program has UDP highlighted.

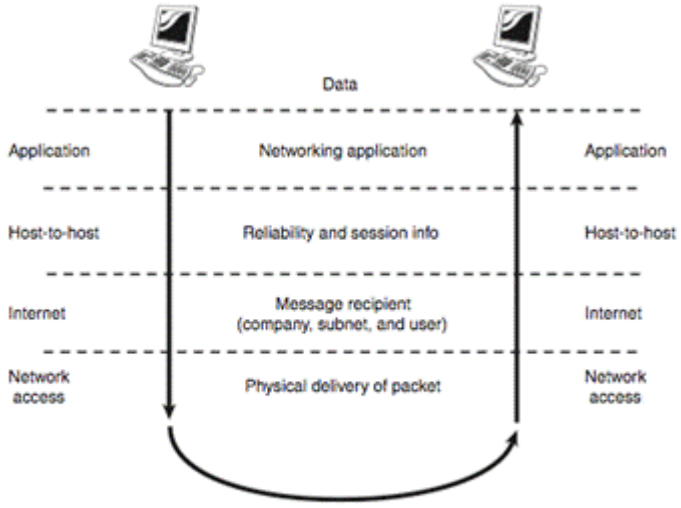

FIGURE 2.2 The TCP/IP stack.

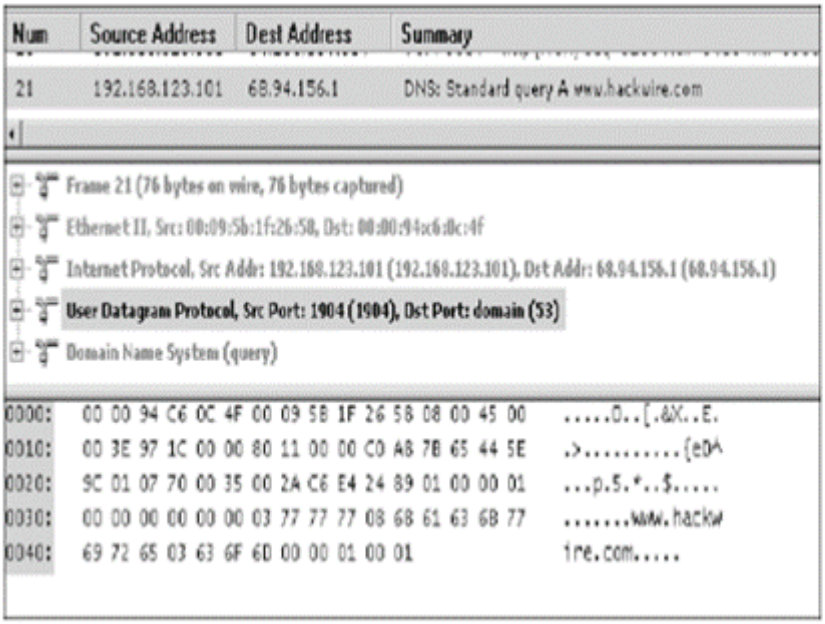

FIGURE 2.3 Encapsulation.

TIP

A lot of free packet sniffing utilities are available on the Internet. Consider evaluating Packetyzer for Windows or Ethereal for Linux. There are also many commercial sniffing tools, such as Sniffer by Network General. These tools can help you learn more about encapsulation and packet structure.

Let’s take a look at each of the four layers of TCP/IP and discuss some of the security concerns lassociated with each layer and specific protocols. The four layers

of TCP/IP include

1. The Application layer
2. The Host-to-host layer
3. The Internet layer
4. The Network access layer

The Application Layer

Objective:

Describe application ports and how they are numbered The Application layer sets at the top of the protocol stack. This layer is responsible for application support. Applications are typically mapped not by name, but by their corresponding port. Ports are placed into TCP and UDP packets so that the correct application can be passed to the required protocols below.

Although a particular service might have an assigned port, nothing specifies that services cannot listen on another port. A common example of this is Simple Mail Transfer Protocol (SMTP). The assigned port of this is 25. Your cable company might block port 25 in an attempt to keep you from running a mail server on your local computer; however, nothing prevents you from running your mail server on another local port. The primary reason services have assigned ports is so that a client can easily find that service on a remote host. As an example, FTP servers listen at port 21, and Hypertext Transfer Protocol (HTTP) servers listen at port 80. Client applications, such as a File Transfer Protocol (FTP) program or browser, use randomly assigned ports typically greater than 1023.

There are approximately 65,000 ports; they are divided into well-known ports (0–1023), registered ports (1024–49151), and dynamic ports (49152–65535). Although there are hundreds of ports and corresponding applications in practice, less than a hundred are in common use. The most common of these are shown in Table 2.1. These are some of the ports that a hacker would look for first on a victim’s computer systems.

TABLE 2.1 Common Ports and Protocols

Code:

```
Port   Service      Protocol
21       FTP           TCP
22       SSH           TCP
23       Telnet        TCP
25       SMTP          TCP
53       DNS           TCP/UDP
67/68    DHCP          UDP
69       TFTP          UDP
79       Finger        TCP
80       HTTP          TCP
88       Kerberos      UDP
110      POP3          TCP
111      SUNRPC        TCP/UDP
135      MS RPC        TCP/UDP
139      NB Session    TCP/UDP
161      SNMP          UDP
162       SNMP Trap     UDP
389     LDAP          TCP
443      SSL           TCP
445      SMB over IP   TCP/UDP
1433     MS-SQL        TCP
```

Blocking these ports if they are not needed is a good idea, but it's better to practice the principle of least privilege. The principle of least privilege means that you give an entity the least amount of access only to perform its job and nothing more. If a port is not being used, it should be closed. Remember that security is a never ending process; just because the port is closed today, doesn't mean that it will be closed tomorrow. You will want to periodically test for open ports. Not all applications are created equally. Although some, such as SSH, are relatively secure, others, such as Telnet, are not.

The following list discusses the operation and security issues of some of the

common applications:

File Transfer Protocol (FTP)

FTP is a TCP service and operates on ports 20 and 21. This application is used to move files from one computer to another. Port 20 is used for the data stream and transfers the data between the client and the server. Port 21 is the control stream and is used to pass commands between the client and the FTP server. Attacks on FTP target misconfigured directory permissions and compromised or sniffed clear-text passwords. FTP is one of the most commonly hacked services.

Telnet

Telnet is a TCP service that operates on port 23. Telnet enables a client at one site to establish a session with a host at another site. The program passes the information typed at the client's keyboard to the host computer system. Although Telnet can be configured to allow anonymous connections, it should be configured to require usernames and passwords. Unfortunately, even then, Telnet sends them in clear text. When a user is logged in, he or she can perform any allowed task. Applications, such as Secure Shell (SSH), should be considered as a replacement. SSH is a secure replacement for Telnet and does not pass cleartext username and passwords.

Simple Mail Transfer Protocol (SMTP)

This application is a TCP service that operates on port 25. It is designed for the exchange of electronic mail between networked systems. Messages sent through SMTP have two parts: an address header and the message text. All types of computers can exchange messages with SMTP. Spoofing and spamming are two of the vulnerabilities associated with SMTP.

Domain Name Service (DNS)

This application operates on port 53 and performs address translation. Although we sometimes realize the role DNS plays, it serves a critical function in that it converts fully qualified domain names (FQDNs) into a numeric IP address or IP addresses into FQDNs. If someone were to bring down DNS, the Internet would continue to function, but it would require that Internet users know the IP address of every site they want to visit. For all practical purposes, the Internet would not be useable without DNS.

The DNS database consists of one or more zone files. Each zone is a collection of structured resource records. Common record types include the Start of Authority(SOA) record, A record, CNAME record, NS record, PTR record, and the MX record. There is only one SOA record in each zone database file. It describes the zone name space. The A record is the most common, as it contains IP addresses and names of specific hosts. The CNAME record is an alias. For example, the outlaw William H. Bonney went by the alias of Billy the Kid. The NS record lists the IP address of other name servers. An MX recordis a mail exchange record. This record has the IP address of the server where email should be delivered. Hackers can target DNS servers with many types of attacks. One such attack is DNS cache poisoning. This type of attack sends fake entries to a DNS server to corrupt the information stored there. DNS can also be susceptible to DoS attacks and to unauthorized zone transfers. DNS uses UDP for DNS queries and TCP for zone transfers.

Trivial File Transfer Protocol (TFTP)

TFTP operates on port 69. It is considered a down-and-dirty version of FTP as it uses UDP to cut down on overhead. It not only does so without the session management offered by TCP, but it also requires no authentication, which could pose a big security risk. It is used to transfer router configuration files and by cable companies to configure cable modems. TFTP is a favorite of hackers and has been used by programs, such as the Nimda worm, to move data without having to use input usernames or passwords.

Hypertext Transfer Protocol (HTTP)

HTTP is a TCP service that operates on port 80. This is one of the most well-known applications. HTTP has helped make the Web the popular protocol it is today. The HTTP connection model is known as a stateless connection. HTTP uses a request response protocol in which a client sends a request and a server sends a response. Attacks that exploit HTTP can target the server, browser, or scripts that run on the browser. Code Red is an example of code that targeted a web server.

Simple Network Management Protocol(SNMP)

SNMP is a UDP service and operates on ports 161 and 162. It was envisioned to be an efficient and inexpensive way to monitor networks. The SNMP protocol allows agents to gather information, including network statistics, and report back to their management stations. Most large corporations have implemented some type of SNMP management. Some of the security problems that plague SNMP are caused by the fact that community strings can be passed as clear text and that the default

community strings (public/private) are well known. SNMP version 3 is the most current, and it offers encryption for more robust security.

TIP

A basic understanding of these applications' strengths and weaknesses will be needed for the exam.

The Host-to-Host Layer

Objectives:
Describe the TCP packet structure
Know the TCP flags and their meaning
Understand how UDP differs from TCP

The host-to-host layer provides end-to-end delivery. Two primary protocols are located at the host-to-host layer, which includes Transmission Control Protocol (TCP) and User Datagram Protocol (UDP).

Transmission Control Protocol (TCP)

TCP enables two hosts to establish a connection and exchange data reliably. To do this, TCP performs a three-step handshake before data is sent. During the data-transmission process, TCP guarantees delivery of data by using sequence and acknowledgment numbers. At the completion of the data-transmission process, TCP performs a four-step shutdown that gracefully concludes the session. The startup and shutdown sequences are shown in Figure 2.4.

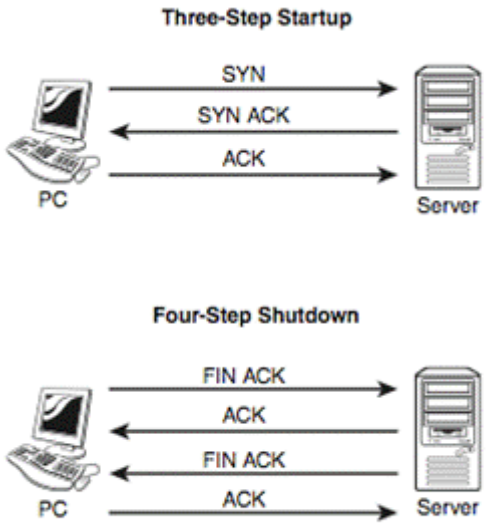

FIGURE 2.4 TCP operation.

TCP has a fixed packet structure that is used to provide flow control, maintain reliable communication, and ensure that any missing data is resent. At the heart of TCP is a 1-byte flag field. Flags help control the TCP process. Common flags include synchronize (SYN), acknowledgement (ACK), push (PSH), and finish (FIN). Figure 2.5 details the TCP packet structure. TCP security issues include TCP sequence number attacks, session hijacking, and SYN flood attacks. Programs, such as Nmap, manipulate TCP flags to attempt to identify active hosts.

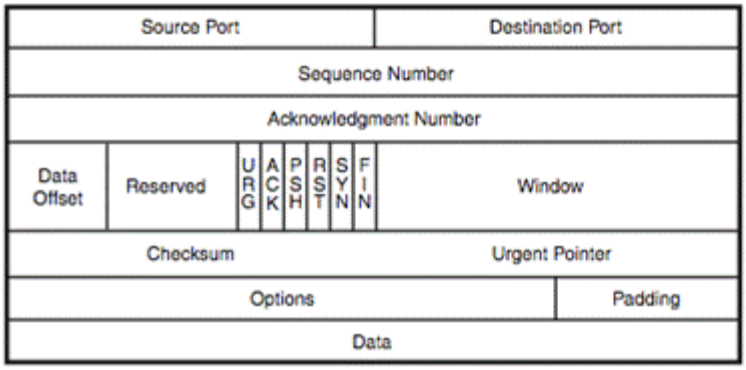

FIGURE 2.5 TCP packet structure.

The ports shown previously in Table 2.1 identify the source and target application, whereas the sequence and acknowledgement numbers are used to assemble packets into their proper order. The flags are used to manage TCP sessions—for example, the synchronize (SYN) and acknowledge (ACK) flags are used in the three-way handshaking, whereas the reset (RST) and finish (FIN) flags are used to

tear down a connection. FIN is used during a normal four-step shutdown, whereas RST is used to signal the end of an abnormal session. The checksum is used to ensure that the data is correct, although an attacker can alter a TCP packet and the checksum to make it appear valid. Other flags include urgent (URG). If no flags are set at all, the flags can be referred to as Null, as none are set.

TIP
Not all hacking tools play by the rules; most port scanners can tweak TCP flags and send them in packets that should not normally exist in an attempt to illicit a response for the victim's server. One such variation is the XMAS tree scan, which sets the SYN, URG, and PSH flags. Another is the NULL scan, which sets no flags in the TCP header.

User Datagram Protocol (UDP)

UDP performs none of the handshaking processes that we see performed with TCP. Although that makes it considerably less reliable than TCP, it does offer the benefit of speed. It is ideally suited for data that requires fast delivery and is not sensitive to packet loss. UDP is used by services such as DHCP and DNS. UDP is easier to spoof by attackers than TCP as it does not use sequence and acknowledgement numbers. Figure 2.6 shows the packet structure of UDP.

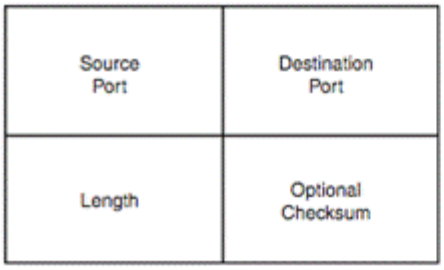

FIGURE 2.6 UDP packet structure.

The Internet Layer

Objective:
Describe how Internet Control Message Protocol (ICMP) functions and its purpose

The Internet layer contains two important protocols: Internet Protocol (IP) and

Internet Control Messaging Protocol (ICMP). IP is a routable protocol whose function is to make a best effort at delivery. The IP header is shown in Figure 2.7. Spend a few minutes reviewing it to better understand each field's purpose and structure. While reviewing the structure of UDP, TCP, and IP, packets might not be the most exciting part of security work. A basic understanding is desirable because many attacks are based on manipulation of the packets. For example, the total length field and fragmentation is tweaked in a ping of death attack.

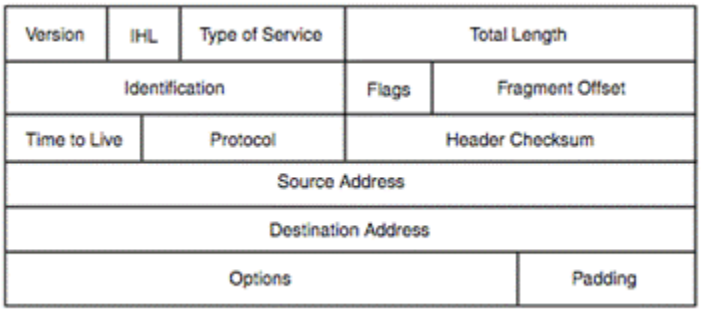

FIGURE 2.7 IP header structure.

IP addresses are laid out in a dotted decimal notation format. IPv4 lays out addresses into a four decimal number format that is separated by decimal points. Each of these decimal numbers is one byte in length to allow numbers to range from 0–255. Table 2.2 shows IPv4 addresses and the number of available networks and hosts.

TABLE 2.2 Ipv4 Addressing

Code:

```
Address Class | Address Range | Number of Networks | Number of Hosts
       A            1-126               126           16,777,214
       B          128-191            16,384               65,534
       C          192-223         2,097152                  254
       D          224-239               NA                   NA
       E          240-255               NA                   NA
```

A number of addresses have also been reserved for private use. These addresses are non-routable and normally should not been seen on the Internet. Table 2.3 defines the private address ranges.

TABLE 2.3 Private Address Ranges

Code:

```
Address Class            Address Range             Default Subnet Mask
       A          10.0.0.0 - 10.255.255.255.255       255.0.0.0
       B          172.16.0.0 - 172.31.255.255         255.255.0.0
       C          192.168.0.0 - 192.168.255.255       255.255.255.0
```

IP does more than just addressing. It can dictate a specific path by using strict or loose source routing, and IP is also responsible for datagram fragmentation. Fragmentation normally occurs when files must be split becauseof maximum transmission unit (MTU) size limitations.

Source Routing: The Hackers Friend

Source routing was designed to allow individuals the ability to specify the route that a packet should take through a network. It allows the user to bypass network problems or congestion. IP's source routing informs routers not to use their normal routes for delivery of the packet but to send it via the router identified in the packet's header. This lets a hacker use another system's IP address and get packets returned to him regardless of what routes are in between him and the destination. This type of attack can be used if the victim's web server is protected by an access list based on source addresses. If the hacker were to simply spoof one of the permitted source addresses, traffic would never be returned to him. By spoofing an address and setting the loose source routing option to force the response to return to the hacker's network, the attack might succeed. The best defense against this type of attack is to block loose source routing and not respond to packets set with this option.

If IP must send a datagram larger than allowed by the network access layer that it uses, the datagram must be divided into smaller packets. Not all network topologies can handle the same datagram size; therefore, fragmentation is an important function. As IP packets pass through routers, IP reads the acceptable size for the network access layer. If the existing datagram is too large, IP performs fragmentation and divides the datagram into two or more packets. Each packet is labeled with a length, an offset, and a more bit. The length specifies the total length

of the fragment, the offset specifies the distance from the first byte of the original datagram, and the more bit is used to indicate if the fragment has more to follow or if it is the last in the series of fragments. An example is shown in Figure 2.8.

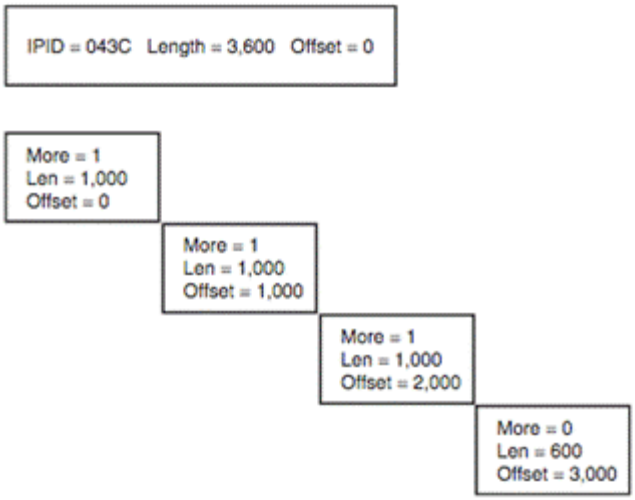

FIGURE 2.8 Fragmentation (3,600).

The first fragment has an offset of 0 and occupies bytes 0–999. The second fragment has an offset of 1,000 and occupies bytes 1,000–1,999. The third fragment has an offset of 2,000 and occupies bytes 2,000–2,999, and the final fragment has an offset 3,000 and occupies bytes 3,000–3,599. Whereas the first three fragments have the more bit set to 1, the final fragment has the more bit set to 0 because no more fragments follow. These concepts are important to understand how various attacks function. If you are not completely comfortable with these concepts, you might want to review a general TCP/IP network book. TCP/IP Illustrated by Richard Stevens is recommended.

TIP

On modern networks, there should be very little fragmentation. Usually such traffic will indicate malicious activities.

To get a better idea of how fragmentation can be exploited by hackers, consider the following: Normally, these fragments follow the logical structured sequence as shown in Figure 2.8. Hackers can manipulate packets to cause them to overlap abnormally, as shown in Figure 2.9.

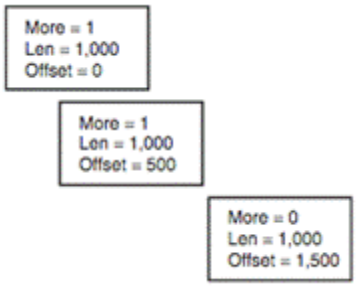

FIGURE 2.9 Overlapping fragment attack.

Hackers can also craft packets so that instead of overlapping, there will be gaps between various packets. These nonadjacent fragmented packets are similar to overlapping packets because they can crash or hang older operating systems that have not been patched.

TIP

A good example of the overlapping fragmentation attack is the teardrop attack. The teardrop attack exploits overlapping IP fragment and can crash Windows 95, Windows NT, and Windows 3.1 machines.

One of the other protocols residing at the Internet layer is ICMP. Its purpose is to provide feedback used for diagnostics or to report logical errors. ICMP messages follow a basic format. The first byte of an ICMP header indicates the type of ICMP message. The following byte contains the code for each particular type of ICMP. The ICMP type generally defines the problem, whereas the code is provided to allow a specific reason of what the problem is. As an example, a Type 3, Code 3 ICMP means that there was a destination error and that the specific destination error is that the targeted port is unreachable. Eight of the most common ICMP types are shown in Table 2.4.

TABLE 2.4 ICMP Types and Codes

Code:

Type	Code	Function
0/8	0	Echo Response/Request (Ping)
3	0-15	Destination Unreachable
4	0	Source Quench
5	0-3	Redirect

11	0-1	Time Exceeded
12	0	Parameter Fault
13/14	0	Time Stamp Request/Response
17/18	0	Subnet Mask Request/Response

The most common ICMP type in Table 2.4 is the type 0 and 8, which is a ICMP ping request and reply. Although a ping is useful to determine if a host is up, it is also a useful tool for the attacker. The ping can be used to inform a hacker if a computer is online. Although the designers of ICMP envisioned a protocol that would be helpful and informative, hackers use ICMP to send the ping of death, craft Smurf DoS packets, query the timestamp of a system or its netmask, or even send ICMP type 5 packets to redirect traffic. A complete list of Type 3 codes are provided in Table 2.5.

TABLE 2.5 Type 3 Codes

Code:

Code	Function
0	Net Unreachable
1	Host Unreachable
2	Protocol Unreachable
3	Port Unreachable
4	Fragmentation Needed and Don't Fragment was Set
5	Source Route Failed
6	Destination Network Unknown
7	Destination Host Unknown
8	Source Host Isolated
9	Communication with Destination Network is Administratively Prohibited
10	Communication with Destination Host is Administratively Prohibited
11	Destination Network Unreachable for Type of Service
12	Destination Host Unreachable for Type of Service
13	Communication Administratively Prohibited

EXAM ALERT

Type 11 ICMP time exceeded messages are used by most traceroute programs to determine the IP addresses of intermediate routers.

Address Resolution Protocol (ARP) is the final protocol reviewed at the IP layer. ARP's role in the world of networking is to resolve known IP addresses to unknown MAC addresses. ARP's two-step resolution process is performed by first sending a broadcast message requesting the target's physical address. If a device recognizes the address as its own, it issues an ARP reply containing its MAC address to the original sender. The MAC address is then placed in the ARP cache and used to address subsequent frames. You discover that hackers are interested in the ARP process as it can be manipulated to bypass the functionality of a switch. Because ARP was developed in a trusting world, bogus ARP responses are accepted as valid, which can allow attackers to redirect traffic on a switched network. Proxy ARPs can be used to extend a network and enable one device to communicate with a device on an adjunct node. ARP attacks play a role in a variety of man-in-the middle attacks, spoofing, and in-session hijack attacks.

EXAM ALERT

ARP is unauthenticated and, as such, can be used for unsolicited ARP replies, for poisoning the ARP table, and for spoofing another host.

The Network Access Layer

The network access layer is the bottom of the stack. This portion of the TCP/IP network model is responsible for the physical delivery of IP packets via frames. Ethernet is the most commonly used LAN frame type. Ethernet frames are addressed with MAC addresses that identify the source and destination device. MAC addresses are 6 bytes long and are unique to the Network Interface card (NIC) card in which they are burned. To get a better idea of what MAC addresses look like, review Figure 2.10, as it shows a packet with both the destination and source MAC addresses. Hackers can use a variety of programs to spoof MAC addresses. Spoofing MAC addresses can be a potential target to attackers attempting to bypass 802.11 wireless controls or when switches are used to control traffic by locking ports to specific MAC addresses.

MAC addresses can be either unicast, multicast, or broadcast. Although a destination MAC address can be any one of these three types, a frame will always originate from a unicast MAC address.

The three types of MAC addresses can be easily identified, as follows:

Code:

```
Type        Identified by
Unicast     The first byte is always an even value.
Multicast   The low order bit in the first byte is always on, and a multicast
MAC addresses is an odd value. As an example, notice the first byte (01) of
the following MAC address, 0x-01-00-0C-CC-CC-CC.
Broadcast   They are all binary 1s or will appear in hex as FF FF FF FF FF
FF.
```

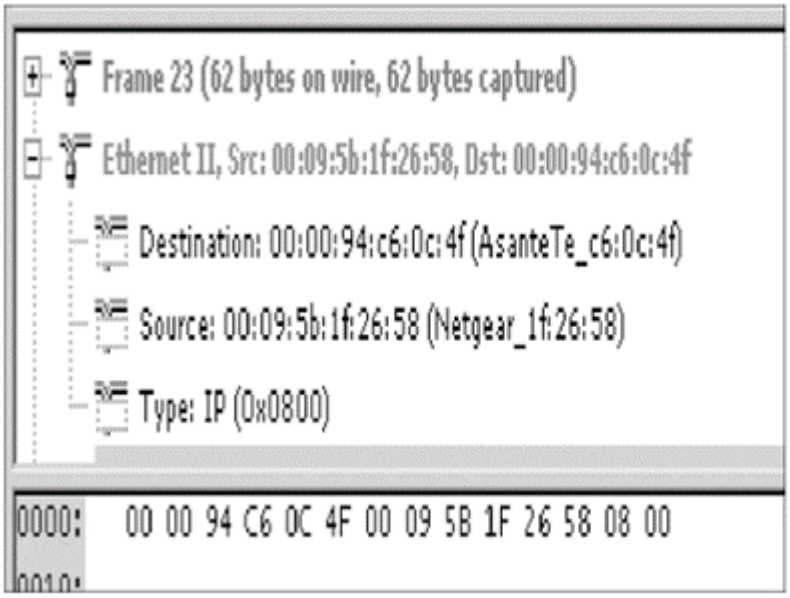

FIGURE 2.10 MAC addresses.

Summary

This lesson discusses the attacker's methodology, as well as some of the methodologies used by ethical hackers. Ethical hackers differ from malicious hackers in that ethical hackers seek to do no harm and work to improve an organization's security by thinking like a hacker. This lesson also discusses the OSI model and the TCP/IP protocol suite. It looks at some of the most commonly used

protocols in the suite and examines how they are used and misused by hackers. Common ports are discussed; as is the principle of deny all. Starting with all ports and protocols blocked leaves the organization in much more of a secure stance than simply blocking ports that are deemed dangerous or unneeded.

Source: http://www.go4expert.com/articles/ethical-hacking-class-part-2-t11988/