Chapter 28

The Enemy (The Intruder's Genesis)

Dr. Pramod Pandya
CSU Fullerton

1. INTRODUCTION

In last few years, we have read and experienced the nature of such threats, resulting in loss of credit-card numbers, Social Security numbers, passwords, and other sensitive information. Security for the network has grown from a simpleminded approach to a multilevel approach, depending on the complexity of the network. As the number of Intranets connected to the Internet has grown, threat to network security has progressed from simple need for network security engineers to need for information security engineers. The primary function of network security engineers was to design network architecture to secure and protect network resources from unauthorized users, namely, hackers. The goal of information security engineers was to define and design the information architectural infrastructure to secure and protect information resources from being stolen by unauthorized users, namely hackers. Why then the need for a cybersecurity engineer? We have now seen that the Internet has more than a couple of billion users connecting to it from all over the world. Since all these users are not from one nation, it does pose a much greater security as the network traffic enters and leaves at the nation boundaries. Hence the Internet has morphed into a cyberspace and thus a need for a cybersecurity engineer. The reader can appreciate that the "Enemy" is not a just a hacker bounded by the nation-state boundary, but can be just from anywhere in cyberspace!

The latest incident-response report from the Industrial Control Systems Cyber Emergency Response Team (ICS-CERT)—part of the Department of Homeland Security (DHS)—warns of ongoing cyberattacks against the computer networks of U.S. natural gas pipeline companies. Cybersecurity experts believe that attacks on the critical commercial and public infrastructure will increase more rapidly in the future economic wars as the nation-states compete for natural resources and intelligence.

At the other extreme in the cyberspace is social media networking, which has dominated the conversation. The young and the mature alike all have accounts on Facebook to connect with each other. Facebook has a vital interest in making sure that its clientele feel confident about sharing their experiences and personal moments with their friends. Sharing becomes a lot less appealing when there is a risk of contagion. A growing number of companies have created their presence on Facebook and would want to make sure that the users on the Facebook would feel safe and confident to click on links that would direct them to a company's e-commerce site for marketing, customer support, and product reach. Presently, more information about individuals and companies has been made publicly available on Facebook, LinkedIn, Twitter, and other social media Web sites (cyberspace) than ever before. Social media sites can be used by companies to gather information about their competitors, and by hackers to exploit the information, the passwords, and much more.

A host of software-based hacking tools or toolkits are available freely from the various Web sites on the Internet. Hacker would initially take a survey of a network, scan for network devices, then look for open ports on those network devices, take an audit of available sensitive information on the vulnerable network devices, and finally design an attack plan to secure the sensitive information. Of course, this approach to compromise the network is designed so that the trace back to the hacker is lost in the maze of the Internet. It is not possible to completely secure the network (see Figure 28.1) against hackers, as most networks have at least one open port that permits communications with the rest of the Internet. This is the dilemma facing all network security professionals: how to safeguard the network from unauthorized access. The following steps are undertaken by those intending to hack a network:

- Gather information about nodes on the target network.
- Look for vulnerabilities in the target network—"holes."

Computer and Information Security Handbook. DOI: http://dx.doi.org/10.1016/B978-0-12-394397-2.00028-3

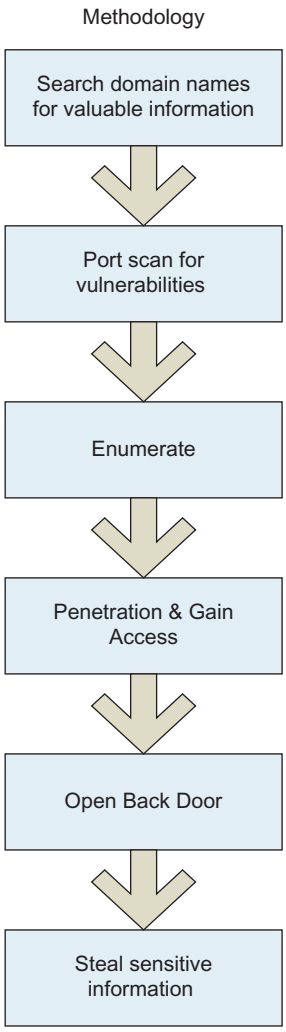

FIGURE 28.1 The six-step process necessary to gain control of a network.

- Exploit the "holes" to access the nodes in the target network.
- Secure access to the target network, without leaving the traces behind, for launching future attacks on the target network, or third-party networks.

Figure 28.1 shows the six-step process necessary to gain control of a network.

2. ACTIVE RECONNAISSANCE

The first step involves searching for Internet domain names, to help to identify those entities that would hold valuable information worth gaining access to. The next step is to map the domain names to network addresses, and finally map out the detail infrastructure of that network. Now we can begin to discover IP addresses of the network nodes and attempt to identify DNS servers, database servers, email servers, and Web servers. These various servers would hold the sensitive information of value. The next phase would be to place the DNS servers, email, and Web servers and database servers on the network and reproduce a complete network, including its functional specification. Once the target network is mapped out, we will use network-based tools to get all the information about that server, and then make a preparation to design a scheme to attack the network.

The domain name registered by the target corporate network can be found by entering the organization's domain name in a search at www.internic.net/whois.html. Thus, we can learn addresses for the target networks' DNS servers, Web servers, and email servers. The GFI Languard NSS software has a utility "whois" that easily allows discovering all the information regarding a domain name registered to a corporate network. DNS Zone transfers refer to learning about the servers and their IP addresses from zone files. Information collected is used to determine what TCP and or UDP services such as HTTP, SMTP, or FTP are in either "listening," "wait," or "closed" state, including the types of operating system and applications currently in use.

The examples of port scanning, and enumeration illustrated in this chapter were obtained using the network in Figure 28.2. The network consists of the following computers:

- kailash a Windows 2000 server—Domain PANDYA
- kalidas a Windows XP workstation
- nanjun a Linux server

Network Mapping

Network mapping (see Figure 28.3) is the process of discovering information about the topology of the target network, thus finding the IP addresses of gateways, routers, email, Web, FTP servers, and database servers. The next step is to sweep the target network to find live nodes by sending ping packets and waiting for response from the target nodes. ICMP messages can be blocked, so an alternative is to send a TCP or UDP packet to a port such as 80 (http) that is frequently open, and live machines will send a SYN-ACK packet in response. Once the live nodes are mapped, standard utility such as traceroute can provide additional information about the network topology by discovering the paths taken by packets to each host, which provides information about the routers and gateways in the network and the general network layout.

The screenshot in Figure 28.3 is obtained using a network security scanner from GFI Languard (http://www.gfi.com). This software is a commercial product, but a trial version 6.0 can be downloaded for 14 days.

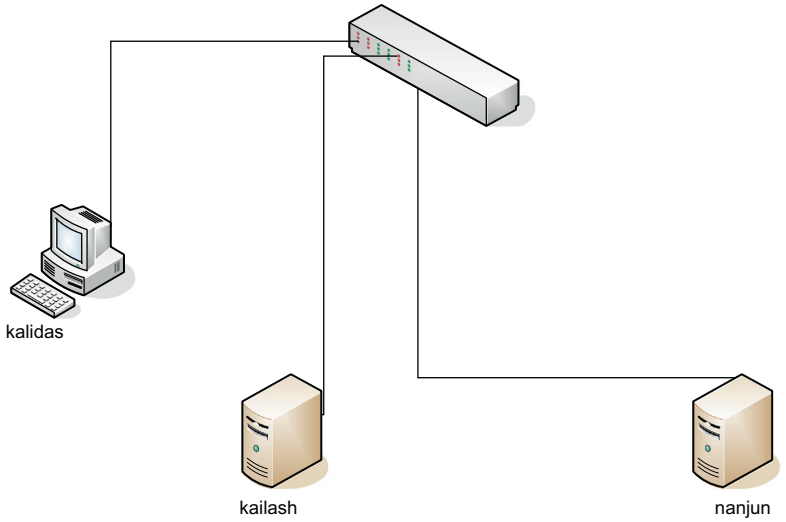

FIGURE 28.2 Switched Ethernet network.

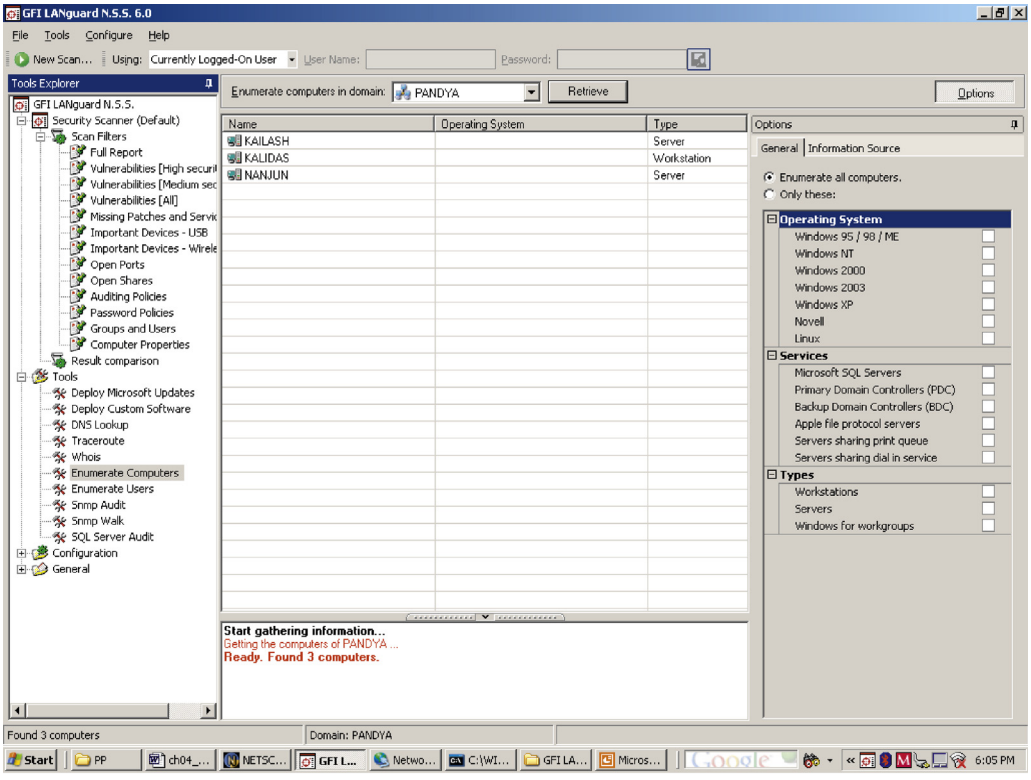

FIGURE 28.3 Network mapping of computers in Figure 28.2.

Nmap

The Nmap main page is described as a security/network exploration tool and port-scanner. The basic command line syntax to invoke Nmap is as follows:

- nmap [scan type(s)] [options] {target specification}
- Nmap has a huge list of command-line options, generally categorized into target specification, host listing, port specifications, service identification, scan technique, scripted scans, and output options. Some of the Nmap switches only work when run as the root (superuser).
- nmap -sL 192.168.1.0/24—Lists all the hosts scanned (all responding IPs in the subnet from 192.168.1.1 to 192.168.1.254).
- nmap -p80,443 192.168.1.10-20—Scans the IP address range looking for open ports 80 and 443.

- nmap -p T:80,8080,6588,800 172.16.0.1/22 — Scans all hosts between 172.16.0.1 and 172.16.3.254, looking for open TCP ports 80, 8080, 6588, and 800 (the default listening ports for various proxy servers).
- nmap -sP 192.168.1.10,20 — Ping scans two hosts in a fast scan.
- nmap -PN 192.168.1.0/29 — Scans all the hosts in the 192.168.1.1 to 192.168.1.6 range. Sometimes host-based firewalls deny ping requests, and it is difficult to scan such hosts. The `-PN` scan is useful in such cases; it scans the hosts assuming them to be online.
- nmap -A -F 192.168.1.1 — Detects target OS and services running on it, in fast-scan mode.

Idlescan

This scan will probe 192.168.1.95 while pretending that the scan packets come from another host; the target's logs will show that the scan originated from 192.168.1.10. This is called a zombie host.

Zombie hosts are those controlled by other hosts on the network. Not all hosts can be used as zombies, as certain conditions are required to be met before this is possible. (Using packages such as hping may enable you to find a zombie host on the network.) The `-v` switch increases the verbosity of the output.

Decoy Host

This command is especially useful while testing IDS/IPS. The `-sS` option will perform a SYN scan on the target host. While doing so, it will spoof the packet contents to make the target host see them as coming from the specified (`-D`) decoy hosts. The `-sI` and `-D` switches can't be combined, for obvious reasons.

Now, a word of caution: Be careful not to cause an unintended denial-of-service (DoS) attack while using the `-D` option. To understand how this could happen, we need to know how a TCP handshake operates. TCP, being a connection-oriented protocol that guarantees delivery of packets, operates with a three-way handshake:

- The client initiates the communication by a SYN.
- The server acknowledges with a SYN-ACK.
- The client again sends an ACK, and now they can communicate.

If the `-D` switch is used, and there is a live host at the decoy IP address, then the SYN-ACK reaches the actual host at the decoy IP address, and not the host running the Nmap scan. Since the real host at the decoy address did not initiate the connection, it closes the connection by sending a TCP Reset (RST). There's no problem with this.

However, a problem occurs if the decoy IP address is not active on the network — there is no RST sent to the scan target, which keeps the connection open. As Nmap continues to generate more and more requests to the target with the decoy IP as the source, the scan target has a growing list of open connections for which it maintains the "connection-initiated" state. This ends up consuming more and more resources on the target and may cause a DoS to other, legitimate hosts and communications.

FIN Scan

The Nmap FIN scan comes in handy in such circumstances. The standard use of a FIN packet is to terminate the TCP connection — typically after the data transfer is complete. Instead of a SYN packet, Nmap initiates a FIN scan by using a FIN packet. Since there is no earlier communication between the scanning host and the target host, the target responds with an RST packet to reset the connection. However, by doing so, it reveals its presence. A FIN scan is initiated using a command like `nmap -sF 192.168.1.1`.

Port Scanning

The second step in reconnaissance is known as port scanning. All networks are secured by one firewall on the perimeter of the network, and this firewall is configured to permit HTTP and SMTP traffic to pass through. Other application traffic is forced to use a secured tunnel to pass through the network. Of course, the perimeter firewall is configured to monitor the traffic, and a log is kept for analysis. Internal network is built using Ethernet segments to reflect the infrastructure of the organization. IP network segments are then superimposed on the Ethernet segments. Each IP network segment is secured from each other by a firewall. Each of the IP segments is connected to the layer-3 switch, thus further protecting each IP segment from an external attack. The IP traffics from the layer-3 switch are directed to pass through a Demilitarized ZONE (DMZ) before it enters the perimeter router. The nodes in the DMZ are DNS, SMTP, and HTTP servers, which are permitted for both inbound and outbound traffic. The attacker would scan the ports on the perimeter firewall and look for open ports on the firewall. The firewall would have the ports such as 80 and 25 (well-known) open for Web and email services. The goal of the attacker is to find which ports in "listen," "wait," or "closed" state.

TCP Full Connect. Full TCP connection is a three-way handshake between a source host and a target host to establish a normal connection. This is used to determine the open TCP ports on the target network, even though the packets have to pass through the firewall. If an intrusion detection system (IDS) is installed on the target

network and configured to trigger an alarm to indicate an anomalous behavior on the network, then this activity will be recorded by IDS.

Ping

This mode sends a short UDP packet to the target's UDP ports and looks for an ICMP "Port Unreachable" message in response. The absence of that message indicates either that the port is in use or the target does not return the ICMP message, which can lead to false positives (A *false positive* occurs when an IDS reports as an intrusion an event that is in fact legitimate network activity). This mode, too, is easily recognized by IDS.

TCP SYN Half Open

In Chapter 27, we talked about the mode of the TCP session. This mode normally sends out a SYN packet to the target port and listens for the appropriate response. Open ports respond with SYN + ACK, and closed ports respond with ACK + RST or RST. This mode is less likely to be recorded by IDS, since the TCP connection is not fully complete, and consequently the attacker might get away with this mode of intrusion.

Fragmentation Scanning

In this method of scanning, you break up IP packets into a number of fragments. Consequently, you are splitting up the TCP header over several packets to make it harder for packet filters and so forth to detect what you are doing. IP fragmentation can also lead to a DoS.

Port Numbers

Public IP addresses are controlled by the Internet Assigned Numbers Authority (IANA) www.iana.org, and are unique globally. Port numbers are unique only within a computer system, and they are 16-bit unsigned numbers. The port numbers are divided into three ranges: the Well Known Ports (0..1023), the Registered Ports (1024..49151), and the Dynamic and/or Private Ports (49152..65535).

Well-Known Ports

Port numbers 0 to 1023 are well-known ports. These well-known ports (also called standard ports) are assigned to services by the IANA. On Unix, the text file named `/etc/services` (on Windows 2000 the file named `%windir%\system32\drivers\etc\services`) lists these service names and the ports they use. Here are a few lines extracted from this file:

- echo 7/tcp Echo
- ftp-data 20/udp File Transfer [Default Data]
- ftp 21/tcp File Transfer [Control]
- ssh 22/tcp SSH Remote Login Protocol
- telnet 23/tcp Telnet
- domain 53/udp Domain Name Server
- www-http 80/tcp World Wide Web HTTP

Nonstandard Ports

By a nonstandard port, we simply mean a port whose number is higher than 1023. In this range also, several services are "standard." For example:

- wins 1512/tcp # Microsoft Windows Internet Name Service
- yahoo 5010 # Yahoo! Messenger
- x11 6000-6063/tcp # X Window System

Once the IP address of a target system is known, an attacker can then begin the process of port scanning, looking for holes in the system through which the attacker can gain access to the network nodes. We have already discussed the significance of TCP and UDP port numbers, and the well-known and not so well-known services that run at these ports. Each of these ports is a potential entryway or "hole" into the network. If a port is open, there is a service listening on it; well-known services have assigned port numbers, such as http on TCP port 80 or telnet on TCP port 23. Port scanning is the process of sending packets (TCP or UDP) to each port on a system to find out which ones are open.

A port scanner such as Nmap is capable of a wider variety of TCP scans that are harder to detect. Nmap allows an option for a TCP SYN stealth scan in which the third message is not an ACK but a FIN that forces the TCP connection to be closed before fully opening. This half-open connection is not logged at the target, but may be noticed by routers or firewalls that record the original SYN packet.

Nmap also allows options that give the attacker more control over the packets sent. The attacker can set the rate at which packets are sent, since changing the timing to space out the packets can help avoid raising the target's suspicions that it is being scanned. If the rate is set too fast, packets can be lost, and incorrect results will be returned. The attacker can also fragment the packets to avoid intrusion detection systems, many of which only look for the whole suspicious packet to be sent at once. Nmap even allows the attacker to set the source port, for example, to 80 to appear as Web traffic to a packet filter, as well as to set a decoy source address to obscure the real address by sending an extra packet per decoy address.

Bounce Scans

In the bounce scan, the attacker would attempt to fool or mislead the victim into believing that the attack originated from a different source IP address, often known as the distributed denial-of-service attacks (DDoS). Such an attack would make it difficult to trace the attacker's IP address. Most commercial Internet sites such as Yahoo, Google, Microsoft, and others support proxy services so that all Web traffic can be directed to a single server for filtering as well as caching to improve performance. We have seen cases of DDOS in spite of the proxy servers' setup to protect the networks.

Vulnerability Scanning

One essential type of tool for any attacker or defender is the vulnerability scanner. These tools allow the attacker to connect to a target system and check for such vulnerabilities as configuration errors, default configuration settings that allow attackers access, and the most recently reported system vulnerabilities. Most commercial Network Security Services (NSS) are expensive and do not come with the source code, while the open-source NSSs are free and the source code is readily available. The open-source tool Nessus is an extremely powerful network scanner and can be configured to run a variety of attacks. Nessus includes a variety of plug-ins that can be enabled, depending on the type of security checks the user wishes to perform. Nessus includes its own scripting language, called Nessus Attack Scripting Language (NASL), which can be used to create individualized attacks and incorporate them with the other plug-ins. Although attacks could be written in C, Perl, Python, or a variety of other languages, NASL was designed to be an attack language. The screenshot in Figure 28.4 was obtained using a network security scanner from GFI Languard.

3. ENUMERATION

Now we should be ready to generate a laundry list of resources that we identified to be vulnerable using the scanning devices; this is known as Enumeration. Our list would have at a minimum the following resources that we discovered to be vulnerable; DNS, Web, and email servers in the DMZ. If we had managed to penetrate through the firewall, then of course, most of the resources on the Intranet would now be vulnerable to our attacks. We have now completed the Active Reconnaissance, phase one of Figure 28.1.

4. PENETRATION AND GAIN ACCESS

Vulnerable resources on the network have been itemized, so we are ready now to devise an attack scheme and to

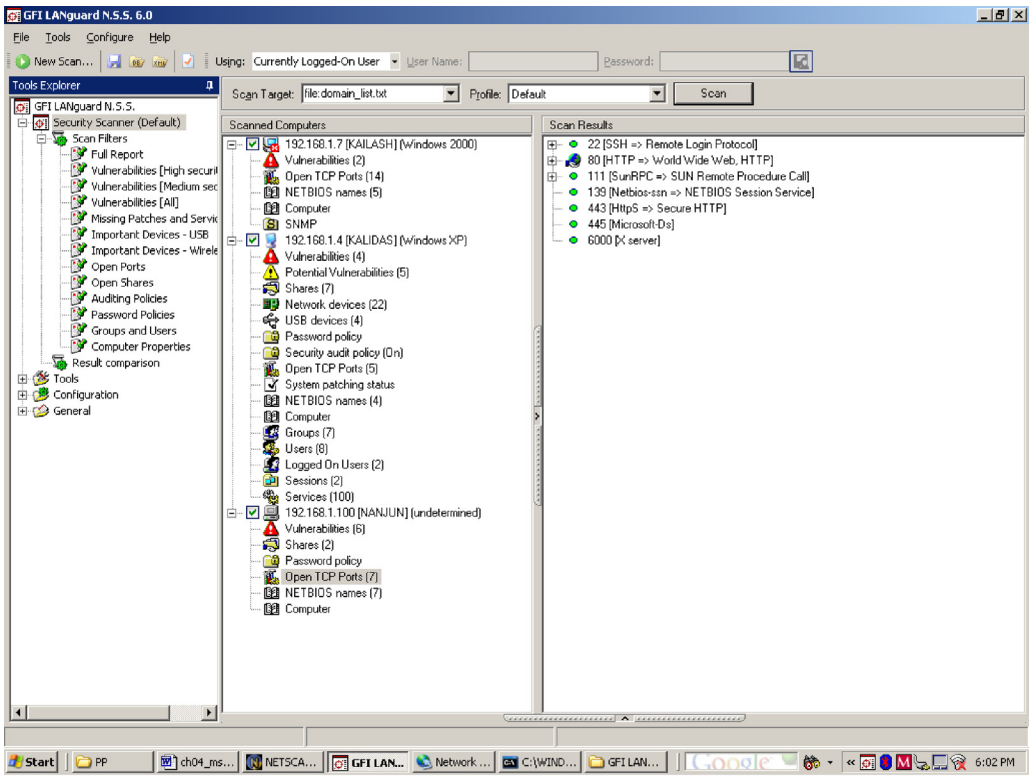

FIGURE 28.4 Vulnerable open ports on the computers in Figure 28.2.

proceed with penetration techniques. Database and HTTP servers are now ready to be compromised.

Stack-Based Buffer Overflow Attacks

Stack-based buffer overflow attacks take advantage of poorly written applications and operating systems architecture. A computer program is a process that is executed by the central processing unit (CPU). Each process manages its memory and input/output. Typically, a process is broken into functions and one main function. This main function is the entry point to the program. Once the program enters the execution mode, the functions in the program get executed as they are called by other functions in a given order. When a called function has completed execution, the control must be returned to the calling function. Herein lay\\ies the problem: When a function is in the execution mode, it needs to store data it is manipulating; this data is stored in a region of memory called a stack. If this region of memory gets overwritten, this will lead to crashing the program—buffer overflow results.

A special-purpose register keeps track of the currently executing instruction, and points to the address of this instruction —Instruction Pointer (IP). This IP is stored on the stack during the function call. Now as you can see, if buffer overflow takes place, then the return addresses get smashed, and the CPU will return to whatever address is available in the IP. The attacker will take advantage of this flow in the architecture by placing the address in the IP.

Once a stack overflow is successful, the returned address gets altered. This altered address is where the attacker will place the payload which could be a virus, a malware, or a Trojan horse. The playload is injected thus, and the attacker will have control of the target network node.

Password Attacks

Every node on a network is secured by assigning a password to access the resources available on the network or on that node. The current policy set by the network administrators is to force the users to change the password over a defined time interval. A new password cannot be a previous password, and it must be alphanumeric and of a certain minimum length to make it difficult for someone to break it. Users' passwords are normally encrypted and stored in a file. If a user forgets the password, then the password will need to be reset as there is no way of recovering the password. An attacker would attempt to get hold of the file that has encrypted passwords. Using password cracking tools as listed below, hacker might be able to recover the plaintext passwords.

Password Cracking Tool: John the Ripper is a password cracker available under Linux and Windows. DoS attacks have become more complicated, concealing malicious client requests as legitimate ones. Also, a distributed approach, the DDoS (distributed denial of service) is now being adopted, which involves generating multiple requests to create a flood scenario. One type of DDoS flood attack is the TCP SYN queue flood.

A SYN queue flood attack takes advantage of the TCP protocol's "three-way handshake." A client sends a TCP SYN (S flag) packet to begin a connection to the server. The target server replies with a TCP SYN-ACK (SA flag) packet, but the client does not respond to the SYN-ACK, leaving the TCP connection "half-open." In normal operations, the client should send an ACK (a flag) packet followed by the data to be transferred, or an RST reply to reset the connection. On the target server, the connection is kept open, in a "SYN_RECV" state, as the ACK packet may have been lost due to network problems.

In a DDoS, multiple attackers make many such half-connections to the target server, in a storm of requests. When the server's SYN buffer is full with half-open TCP connections, it stops accepting SYN connections, thus resulting in denial of service to legitimate clients.

Such DDoS attacks are generally carried out using "botnets" of other compromised systems across the Internet, which through backdoors and Trojans are directed to send artificial SYN flood traffic to targeted servers. To defend against such attacks, a strong monitoring system is required, as there is a very fine line between legitimate and fake clients. SYN queue flood attacks can be mitigated by tuning the kernel's TCP/IP parameters.

Sniffing

The most popular packet sniffer—Wireshark—is a computer program that can monitor the traffic passing over a network or part of a network. Wireshark is normally placed on a network node and configured to run in a promiscuous mode to capture every packet traversing on that network. Sniffer can be configured to capture traffic at any one of the Internet's models, such as layer 2 (Ethernet), layer 3 (IP), layer 3 (TCP or UDP), layer 4 (SMTP, HTTP, DNS, DHCP), or layer 5 (Applications).

Once the data traffic is captured, the hacker would have analyzed the contents of the packets and be able to draw inferences about what is being captured. Hackers would thus have access to port numbers, IP addresses, and application details.

Sniffing Tools

One popular sniffing tool is called Wireshark (http://www.wireshark.com/). The other popular sniffing tool is TCPDump, which can provide very detailed information on the captured data traffic (http://www.tcpdump.org/).

IP Address Spoofing

IP address spoofing normally involves what is known as IP packet crafting. Once again it is a computer program that allows the attacker to target a perimeter router into accepting the IP packet with a disguised IP source address. The real IP source address is spoofed. The purpose of IP spoofing is to make it difficult to trace back to the attacker's node. IP packet crafting is possible by overriding the function of the kernel of the operating system.

MAC Address Modifying Utility: SMAC

Similar to IP address spoofing is a utility that will allow the Ethernet address (MAC) to be spoofed (http://www.klcconsulting.net/smac). The attacker could use nmap to generate packets with the fake IP address in their headers. In this scenario, the target will send any response packets to the spoofed address, so its usefulness is limited to situations where the attacker needs to obscure the source of packets, such as in a denial-of-service attack.

DNS Spoofing

In this case, the domain name system server is spoofed to alter entries of domain names to reflect the attackers' IP address. This results in sending Web or email traffic to the attackers' machine. This attack is achieved by creating multiple forged packets wherein the IP, port, and service type entries are modified to serve the purpose.

Session Hijacking

Session hijacking is an act of taking over an ongoing active connection between two nodes on a network. Hijackers would have been monitoring an active session over a network, using a combination of sniffing and spoofing tools for a while. There is a TCP and UDP session hijacking. The hacker would have to continue to monitor the type of application layer protocol being used between two nodes, since the application layer protocol would decide the type of application being hijacked. We give examples of application layer protocols such as HTTP, SMTP, and DNS used to exchange data between any two active nodes on the network. We remind the reader that all three application layer protocols just stated use the TCP protocol at the layer 3 of the Internet model. Hence the hijacker would have to monitor the TCP port number 80 (HTTP), 25 (SMTP), and 57 (DNS) in order to hijack an active session.

TCP Session Hijacking

Let us recall that a TCP session starts out with a three-way handshake between the two nodes (one node is a client, and the other node is a server) that would like to establish a session between them. The nodes would exchange a sequence of TCP segments with well-defined sequence numbers to establish an active session. This active session is normally terminated by an exchange of FIN (finish) packet or abruptly with RST (reset) packets.

If a would-be hijacker were to correctly guess the sequence number of TCP segments between the two nodes, then it is quite possible that the hijacker could hijack the session before that session gets established between the original TCP client and the server. The original client would still send an ACK segment to the server, but the server would assume that it has received a duplicate segment with a matching sequence number, and thus ignore, as this happens quite a lot of times on the network. This scenario is not a complete description of session hijacking, but just an overview.

Route Table Modification

In this scenario, the attacker would block the packets by modifying the routing tables so that the packets flow through the network that the attacker has the control over. This is known as redirection of traffic and is normally achieved using ICMP (Internet Control Message Protocols) packets.

UDP Hijacking

The DNS protocol would need to be hijacked, if the attacker would want to pretend to be a Web server. The attacker would grab a copy of the HTTP request packet originating from a client to a Web server. Then the attacker would extract the request for a HTTP session from the packet and insert the attacker's IP address, and forward the packet to the client. The client would then establish the HTTP session with the attacker's node, unless the client verified the IP address to confirm that the session has not been hijacked.

Session-Hijacking Tool: Hunt

Hunt (http://packetstormsecurity.org/sniffers/hunt) has sniffing and session hijacking modes. Hunt uses ARP spoofing to establish the attacker's machine as a relay between, say Alice and Bob. When a system prepares to send a packet over a LAN, it first sends out an Address Resolution (ARP) query to all the other systems on the LAN, asking which of them has the Medium Access Control (MAC) address that corresponds to the IP address in the packet's header. The destination system replies with its MAC address, which the source system stores in its ARP cache for a certain period of time. For that period, the source system uses the data from its ARP

cache to send transmissions to that destination. The attacker subverts this process by sending an unsolicited ARP response to Alice that maps Bob's IP address to a fake MAC address, and by sending an unsolicited ARP response to Bob that maps Alice's IP address to a fake MAC address. Both Bob's and Alice's systems overwrite these fake MAC addresses into their ARP caches, so that the packets they send to each other will go to fake addresses. They now cannot send packets to each other for the lifetime of the ARP cache.

Web Hijacking

Hackers may cause serious damage by either defacing the site or using the Web server to spread a virus. Unlike most other attacks, the techniques used in Web attacks range from layer 2 to layer 7 attacks, thus making the Web server susceptible to a wider variety of possible hacking attempts. Since the firewall port must be opened for the Web service (by default, port 80), it cannot help in preventing layer 7 attacks, which makes the detection of Web attacks difficult.

SQL Injection

As we saw earlier, Web portals use database servers in the backend, whereby the Web page connects to the database, queries for data, and presents the fetched data in a Web format to the browser. SQL injection attacks can occur if the input on the client side is not filtered appropriately before it is sent to the database in a query form. This can result in the possibility of manipulating SQL statements in order to perform invalid operations on the database.

A common example for this attack would be that of an SQL server, which is accessed by a Web application, wherein the SQL statements are not filtered by middleware or validation code components. This can lead to the attacker being able to craft and execute his own SQL statements on the backend database server, which could be simple SELECT statements to fetch and steal data, or could be as serious as dropping an entire data table.

5. MAINTAIN ACCESS

All types of service providers on the Internet that hold their clients' sensitive information, are required to notify their clients if a network breach has occurred. This has been mandated by every state in the country. Attackers must remove any evidence of intrusion associated with establishing access, modifying privileges, installing rootkits, and injecting backdoors.

Covering Tracks

Once a network has been compromised, the attacker must make sure that the attacker did not leave footprints behind. Every network runs some sort of security software such as the network intrusion detection system (NDIS), and the intrusion prevention system (IPS). The NDIS detects an intrusion and then reports it so that an appropriate response can be directed to the intrusion.

Backdoors and Trojan Horses

Trojan horses are code disguised as a benign program, but behave in an unexpected manner, usually a malicious manner. Trojan horses are normally injected into a foreign host while that host is browsing the Internet or downloading free utilities from the Internet. The host is normally quite unaware that a malicious program has been injected. This malicious program could hijack future HTTP sessions, monitor the activities on that host, and then relay that information back to the attacker's host and much more. Some noteworthy Trojans are ZeuS, ZeroAccess, TDSS Downloader, Alureon, Gbot, Butterfly bot, and BO2K.

Backdoors and Trojan horses have several things in common. They both come with two pieces of software, the client and the server. The server is the piece that the "remote administrator" will use to infect the victim's computer. The client is the piece that the attacker will use to monitor the victim's computer. Both programs allow for complete access to the victim's files. The hacker can copy, move, rename, delete, and even change any file or folder in the victim's computer.

Backdoor Tool: Netcat

Netcat (http://netcat.sourceforge.net) is a multipurpose networking tool capable of a variety of functions ranging from port scanning and opening connections to remote ports to creating backdoor shells for root access. It runs in either client mode or listening (server) mode.

Rootkits

One way an intruder can maintain access to a compromised system is by installing a rootkit as a Loadable Kernel Module in Linux or as a driver in MS-Windows. Furthermore, rootkits may be injected as user mode, in which case it might be detectable by a virus checker. Kernel-mode rootkits would run with the system privileges by adding a code or replacing portions of the core operating system. Kernel-mode rootkits are difficult to detect and remove, as they have the same level of security as the operating system. A rootkit contains a set of tools

and replacement executables for many of the operating system's critical components, used to hide evidence of the attacker's presence and to give the attacker backdoor access to the system. Rootkits require root access to install, but once they are set up, the attacker can get root access back at any time. A rootkit may also consist of spyware and other programs that monitor traffic and keystrokes and can create a "backdoor" into the system for the hacker to gain access to the hacked node. Third-party software to detect and remove rootkits, Trojans, and Malware can be found at the following URLs:

- http://www.gmer.net
- http://usa.kaspersky.com
- http://www.spamfighter.com

6. DEFEND NETWORK AGAINST UNAUTHORIZED ACCESS

Now that we have completed the discussion on how the hacker might gain control of the target network, we will briefly discuss network perimeter defense (see checklist: An Agenda for Action for Network Security Self-Assessment), known as a firewall, as illustrated in Figure 28.5. Most firewalls are both a hardware and software integrated into one device. The firewall sits on the perimeter that defines the inside of the network from the un-trusted outside. A firewall should provide protection against intruders while allowing trusted users to connect to the network and use the resources therein. To set this scenario, first an access policy has to be defined. This policy is then turned into a set of security rules and is implemented as scripts on the firewall. Hence, firewall rules are defined. Thus, firewalls will examine the packets on the basis of the security policy and will either permit or deny the traffic. The security policy would be made up of a range of IP addresses, port numbers, network protocols (TCP, UDP, IP), and application protocols (HTTP, SMTP, FTP, Telnet). A firewall would have some of the ports open for both authorized inbound and outbound traffic, but the rest of the ports would be closed. Open ports remain a necessary vulnerability; they allow connections to applications, but they may also turn into open doors for attack. In the end, as long as ports remain open, network applications are susceptible to attack. Use of intrusion detection systems (IDS) may certainly help in detecting would-be attackers and thus provide some sense of security.

7. SUMMARY

In this chapter we presented a series of steps undertaken by the intruder to gain unlawful access to networks. The intruder first has to scan the network, thus obtaining the

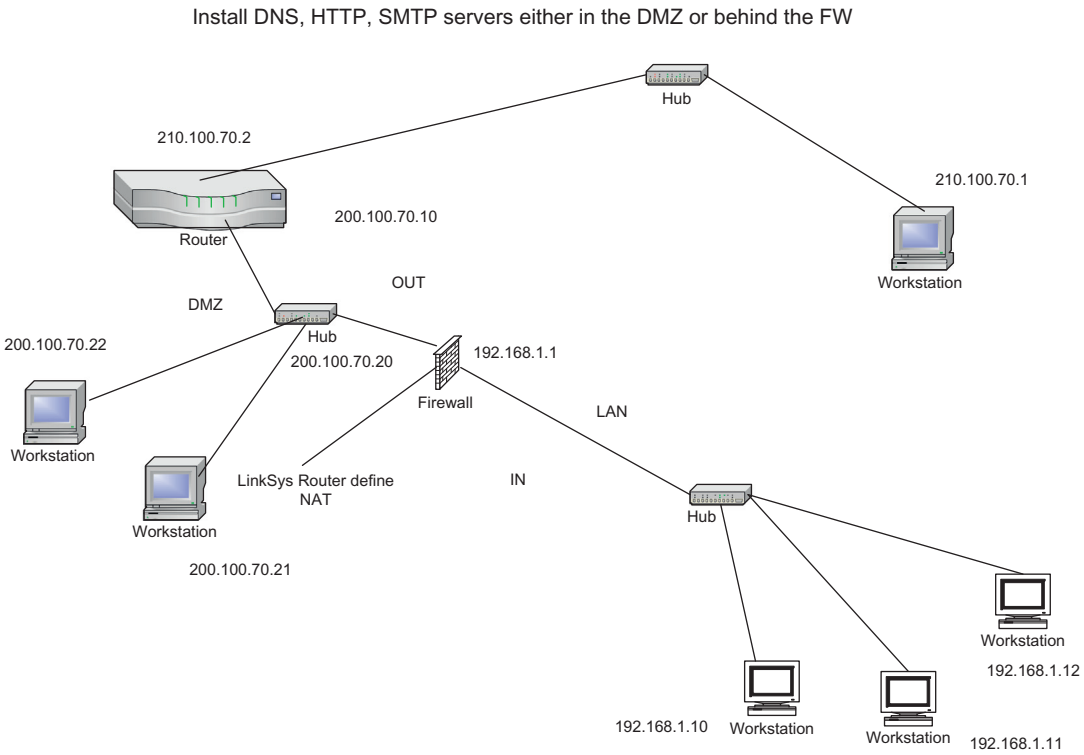

FIGURE 28.5 Network perimeter defense, known as a firewall.

An Agenda for Action for Network Security Self-Assessment

The first step to a self-defending network includes a careful and complete assessment of your network. The Network Security Self-Assessment Checklist shown here can help you quickly assess whether you have network security that is proactive, reactive, or open. The network security practices on this checklist will help ensure that your network is as secure as it can be. It will help you develop proactive, rather than reactive, security and will significantly limit your exposure to threats and the associated liabilities (check all tasks completed).

_____**1.** Conducting network and endpoint security assessments.
_____**2.** Classifying all network and information assets.
_____**3.** Deploying integrated security solutions with intelligent self-defending capabilities.
_____**4.** Identifying areas of regulatory similarities to minimize overhead and avoid duplicate investments in network security. For example, GLBA, the USA Patriot Act, and SOX all require consideration of capabilities for:
  _____**a.** Firewalls
  _____**b.** Encryption
  _____**c.** Access Controls
  _____**d.** Virtual Private Networks
  _____**e.** Intrusion Detection and Prevention
  _____**f.** Anti-virus Software
  _____**g.** Monitoring, Auditing, and Reporting
_____**5.** Aligning your people, processes, and technology to protect your institution.
_____**6.** Educating each employee on his or her security duties and responsibilities.
_____**7.** Managing security as an essential, dynamic, and ongoing project.
_____**8.** Regularly testing your network and endpoint security to identify weaknesses.
_____**9.** Responding immediately and appropriately to known and unknown or emerging security threats.
_____**10.** Updating your security practices to comply with new laws, rules, and guidelines and protect against new threats.
_____**11.** Identifying and reporting security-related events to executive management and the board of directors.

information regarding the resources available on the network. Once the intruder has completed the profile of the network and identified the "holes" in the network, he or she is ready to launch the attack. Next, the intruder would attempt to gain access to the network with the tools that are freely accessible on the Internet. If the intruder is successful in hacking into the network, then he or she could establish a backdoor entrance to the network. In later chapters we will introduce the topic of how network penetration can be achieved, with the tools that are freely available on the Internet. The reader should be warned that network intrusion is a punishable offense.

Now, let's move on to the real interactive part of this chapter: review questions/exercises, hands-on projects, case projects, and optional team case project. The answers and/or solutions by chapter can be found in the Online Instructor's Solutions Manual.

CHAPTER REVIEW QUESTIONS/EXERCISES

True/False

1. True or False? Network mapping is the process of discovering information about the topology of the target network, thus finding the IP addresses of gateways, routers, email, Web, FTP servers, and database servers.
2. True or False? The Nmap main page is described as an exploration tool and port-scanner.
3. True or False? Zombie hosts are those controlled by others on the network.
4. True or False? The decoy host command is not especially useful while testing IDS/IPS.
5. True or False? The standard use of a FIN packet is to not terminate the TCP connection — typically after the data transfer is complete.

Multiple Choice

1. Discover network interconnection and configuration, and look for network vulnerabilities:
   A. DOS
   B. Sniffing
   C. SYN flooding
   D. Reconnaissance
   E. All of the above
2. Removal and/or alteration of data, installing "backdoors," and hiding the tracks of attack activities is known as
   A. enumeration
   B. scanning
   C. DoS
   D. operational attacks
   E. all of the above
3. The port scanning technique is used to discover open _____ ports.
   A. TCP
   B. NetBIOS

C. PDP
D. HTTP
E. All of the above

4. The three-way TCP handshake is established during which of the TCP scanning sessions?
   A. TCP connect()
   B. TCP SYN
   C. TCP FIN
   D. TCP Open
   E. All of the above
5. TCP SYN scanning is also known as
   A. full open
   B. half open
   C. full close
   D. half close
   E. All of the above

EXERCISE

Problem

Download IP-tools and install the software:

1. Visit the Web site, http://www.ks-soft.net.
2. Download the software "IP-Tools."
3. Install IP-tools on your computer.

Hands-On Projects

Project

Download LANguard N.S.S. (Commercial grade Network Security Scanner, N.S.S.):

1. Visit the Web site, http://www.gfi.com.
2. Download LANguard N.S.S., a trial version.
3. Install the software on your computer.
4. Under Tools Explorer, select Whois.
5. Enter the Internet domain such as cox.net to discover the name servers and the IP addresses of the cox.net domain.

Case Projects

Problem

Using LANguard software to rnumerate the computers in your Windows domain:

1. Open LANguard application.
2. Under Tools Explorer, select Tools, and then Enumerate Computers.
3. Enter the domain in which you wish to enumerate.
4. Click the Retrieve TAB.
5. Copy the list of the computers thus displayed.

Optional Team Case Project

Problem

In this case study, you are to learn about RAW Sockets and to compare them with standard TCP/IP or Winsocks. This should prepare you to understand how TCP/IP packets are normally generated by the kernel, and to learn how a hacker bypassed the kernel to inject custom packets to attack a network.