Network Security
أمنية الشبكات

أعداد
الأستاذ الدكتور
زياد طارق الطائي

المرحلة الرابعة
علوم الحاسبات

Network Security

Introduction:

The need for network security is a relatively new requirement. Prior to the 1980s most computers were not networked. It was not due to lack of desire to network them; it was more a result of the lack of technology. Most systems were mainframes or midrange systems that were centrally controlled and administered. Users interfaced with the mainframe through terminals. The terminals had limited capabilities.

In the 1980s, the combination of the development of the personal computer (PC), the development of network protocol standards, the decrease in the cost of hardware, and the development of new applications made networking a much more accepted practice. As a result, LANs, WANs, and distributed computing had a big growth during that period.

In this distributed environment the emphasis was on providing ease of access and connectivity. As a result, many systems were wide open and vulnerable to threats that previously had not existed. The Internet is the largest and best known of this type of network. The Internet utilizes TCP/IP and was primarily designed to connect computers regardless of their operating systems in an easy and efficient manner. Security was not part of the early design of TCP/IP, and there have been a number of widely publicized attacks that have exploited inherent weaknesses in its design. One well-known event was the Internet Worm that brought the Internet to its knees back in 1986. Today, security has to be more important than ease of access.

What is network security? To answer this question, it is necessary to determine what you are trying to protect. Network security is concerned with the security of company information

resources. It is also important to remember that network security is not absolute. All security is relative. Network security is a balancing act that requires the deployment of "proportionate defenses." The defenses that are deployed or implemented should be proportionate to the threat.

Basic Terminology:

1- Threats:
A threat is anything that can stop the operation, functioning, integrity, or availability of a network or system. There are different categories of threats. There are natural threats, occurrences such as floods, earthquakes, and storms. There are also unintended threats that are the result of accidents and stupidity. Finally, there are intended threats that are the result of malicious indent. Each type of threat can be deadly to a network.

2- Vulnerabilities:
A vulnerability is an inherent weakness in the design, configuration, or implementation of a network or system that makes it clear to a threat. Most vulnerabilities can usually be traced back to one of three sources:

1. Poor design:
Hardware and software systems that contain design faults that can be exploited. In essence, the systems are created with security holes. An example of this type of vulnerability would be the "sendmail" faults in early versions of Unix. The sendmail faults allowed hackers to gain privileged "root" access to Unix systems. These faults were exploited on numerous occasions.

2. Poor implementation:
Systems that are incorrectly configured, and therefore vulnerable to attack. This type of vulnerability usually results from inexperience, insufficient training, or incorrect work. An example of this type of vulnerability would be a system that

does not have restricted-access privileges on critical executable files, thereby allowing these files to be altered by unauthorized users.

3. Poor management*:*
Inadequate procedures and insufficient checks and balances. Security measures cannot operate in a vacuum; they need to be documented and monitored. Even something as simple as the daily backup of a system needs to be verified.

While there are only three sources of vulnerabilities, they can obvious themselves in many ways:

Physical Vulnerabilities
The first rule of security is to physically safeguard systems and networks. Are your systems, communications equipment, and media located in a secure environment? Central hosts and servers should be kept in secure rooms that can only be entered by authorized personnel.
Routers and communications equipment should also be kept in secure locations with restricted access. In addition, critical removable media, such as backups, should be stored in a secure area to which only authorized personnel have access.
As part of this process, organizations need to take into consideration the physical and natural environment in which they operate. They should consider the probability of earthquakes, fires, floods, and plan accordingly. Proper planning of physical facilities can reduce many of the effects of natural disasters.

Hardware and Software Vulnerabilities:
Design faults in hardware or software can make systems vulnerable to attack or affect the availability of systems. For example, the sendmail flaw in earlier versions of UNIX enabled hackers to gain privileged access to systems.

Media Vulnerabilities:
Disks, tapes, and other media can be stolen, lost, or damaged. Information can be copied and removed from an organization's environment without detection. Accordingly, companies need to ensure the safety of all media that contains or stores vital information resources.

Transmission Vulnerabilities:
Signal emissions from electrical equipment can be remotely intercepted and monitored using sophisticated devices. Organizations also need to be concerned about the interception of most forms of communication.
Communication is the sharing of information on a medium. As such, it is inherently vulnerable to interception, monitoring, forgery, alteration, and interruption. Every medium used for transmission of information can be "tapped." Network "sniffers" or packet sniffers are common hacker tools that can read traffic as it passes on a network.

Human Vulnerabilities
Human stupidity, carelessness, laziness, and anger represent the greatest threats to networks and systems and will do more damage than the rest of the others combined.
Moreover, human vulnerabilities and the risks associated with them are the most difficult to defend against.

It is important to keep in mind that every network or system designed, configured or implemented has vulnerabilities. There is no such thing as a totally secure network or system. It does not exist!

3- Countermeasures:
Countermeasures are the techniques or methods used to defend against attacks and to close or compensate for vulnerabilities in networks or systems.

4- Identification

Identification is simply the process of identifying one's self to another entity or determining the identity of the individual or entity with whom you are communicating.

5- Authentication

Authentication serves as proof that you are who you say you are or what you claim to be.

Authentication is critical if there is to be any trust between parties. Authentication is required when communicating over a network or logging onto a network. When communicating over a network you should ask yourself two questions: 1) With whom am I communicating? and 2)Why do I believe this person or entity is who he, she, or it claims to be? If you don't have a good answer for question 2, then chances are you are wrong on question 1.

6- Access Control (Authorization):

This refers to the ability to control the level of access that individuals or entities have to a network or system and how much information they can receive. Your level of authorization basically determines what you're allowed to do once you are authenticated and allowed access to a network, system, or some other resource such as data or information. Access control is the determination of the level of authorization to a system, network, or information (i.e., classified, secret, or top-secret).

7- Availability:

This refers to whether the network, system, hardware, and software are reliable and can recover quickly and completely in the event of an interruption in service. Ideally, these elements should not be susceptible to denial of service attacks.

8- Confidentiality:

This can also be called privacy or secrecy and refers to the protection of information from unauthorized detector. Usually

achieved either by restricting access to the information or by encrypting the information so that it is not meaningful to unauthorized individuals or entities.

9- Integrity:
This can be thought of as accuracy. This refers to the ability to protect information, data, or transmissions from unauthorized, uncontrolled, or accidental alterations. The term integrity can also be used in reference to the function of a network, system, or application.

10- Accountability:
This refers to the ability to track or audit what an individual or entity is doing on a network or system.

11- Nonrepudiation:
The ability to prevent individuals or entities from reject (repudiating) that information, data, or files were sent or received or that information or files were accessed or altered, when in fact they were.

Attacks:

An attack is a specific technique used to exploit a vulnerability. For example, a threat could be a denial of service. A vulnerability is in the design of the operating system, and an attack could be a "ping of death."
There are two general categories of attacks, passive and active. Passive attacks are very difficult to detect, because there is no activity that can be monitored or detected. Examples of passive attacks would be packet sniffing or traffic analysis. These types of attacks are designed to monitor and record traffic on the network.

Active attacks, as the name implies, employ more actions on the network or system. As a result, they can be easier to detect, but at the same time they can be much more destructive to a network. Examples of this type of attack would be a denial-of-service attack or active probing of systems and networks.

Networks and systems face many types of threats. There are viruses, worms, Trojan horses, trap doors, spoofs, masquerades, replays, password cracking, social engineering, scanning, sniffing, war dialing, denial-of-service attacks, and other protocol-based attacks. It seems new types of threats are being developed every month.

The following sections review the general types of threats that network administrators face every day, including specific descriptions of a few of the more widely known attacks.

Viruses:

A virus, a parasitic (طفيلـي) program that cannot function independently, is a program or code fragment that is self-propagating. It is called a virus, because like its biological example, it requires a "host" to function. In the case of a computer virus the host is some other program to which the virus attaches itself. A virus is usually spread by executing an infected program or by sending an infected file to someone else, usually in the form of an e-mail attachment.

There are several virus scanning programs available on the market. Most are effective against known viruses. Unfortunately, however, they are incapable of recognizing and adapting to new viruses.

In general, virus scanning programs rely on recognizing the "signature" of known viruses, turning to a database of known virus signatures that they use to compare against scanning

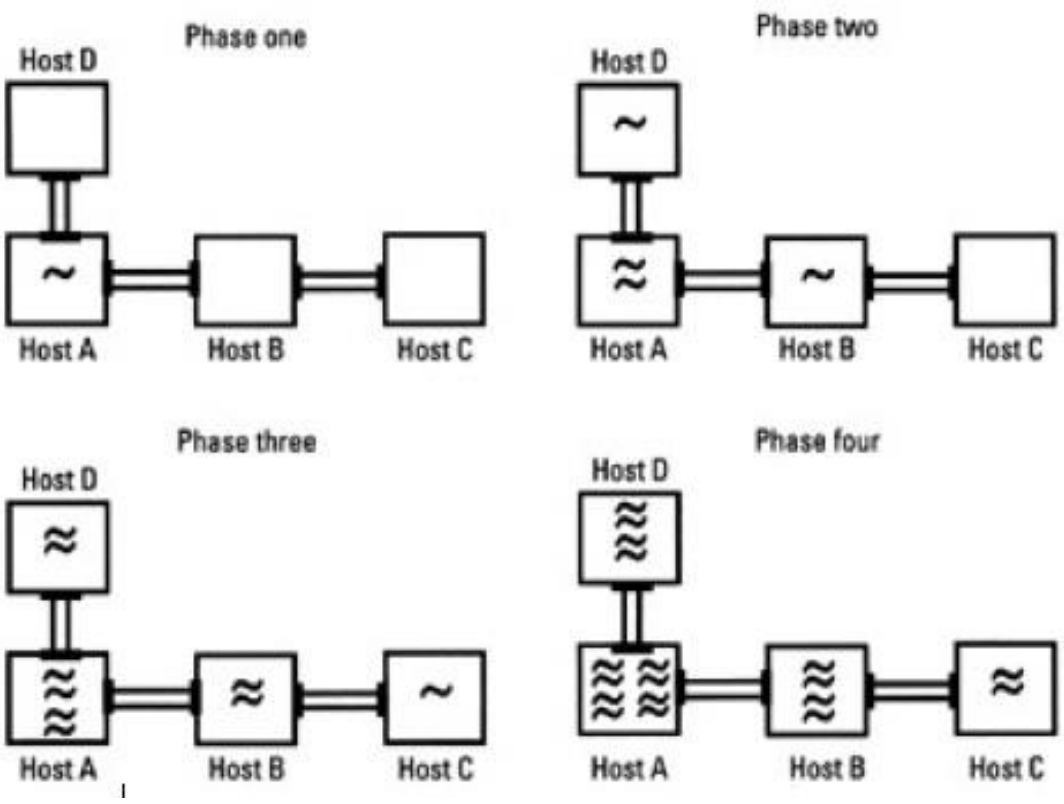
results. The program detects a virus when a match is found. If the database is not regularly updated the virus scanner can become obsolete quickly. As one would expect, there is usually some lag time between the introduction of a new virus and a vendor updating its database.

Worm:

A worm is a self-contained and independent program that is usually designed to propagate or breed itself on infected systems and to seek other systems via available networks. The main difference between a virus and a worm is that a virus is not an independent program.

One of the first and perhaps the most famous worms was the Internet Worm created and released by Robert Morris. In 1986, Morris wrote his worm program and released it onto the Internet. The worm's functioning was relatively kind, but it still had a destructive effect on the Internet. The worm was designed to simply reproduce and infect other systems. Once released, the program would breed another process. The other process was simply another running copy of the program. Then the program would search out other systems connected to the infected system and propagate itself onto the other systems on the network. The number of processes running grew geometrically. Figure below illustrates how the Internet worm grew and spread: One process breed to become two processes. Two processes breed to become four processes. Four processes breed to become eight. It didn't take very long for the breeding processes to consume all the CPU and memory resources until the system crashed.

In addition, each time the processes breed another, the processes would seek outside connections. The worm was designed to propagate, seek out other systems to infect them, and then repeat the process.

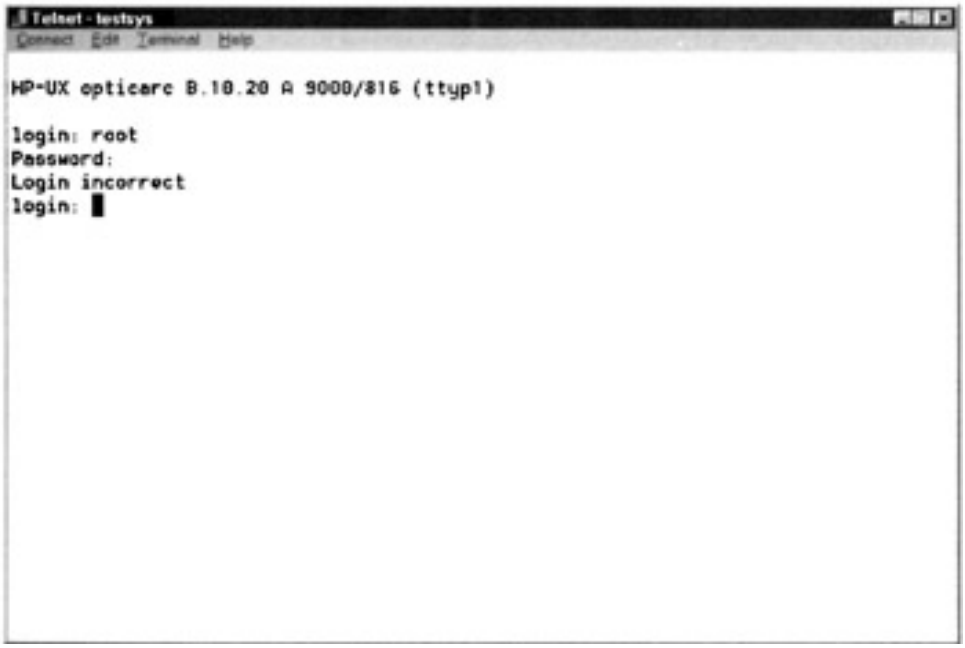


Stopping the processes from growing was a simple matter of rebooting the system. However, system administrators found that they would reboot their systems and get them functioning again only to find them being reinfected by another system on the Internet. To stop the worm from reinfecting systems on the network, all of the systems had to be shut down at the same time or taken off-line. The cost to clean up the Internet worm was estimated to be in the tens of millions of dollars. Morris was arrested, prosecuted, and convicted for his vandalism (ألقي القبض عليه ومحاكمته وأدانته لأعمال التخريب).

Trojan Horses:

A Trojan horse is a program or code fragment that hides inside a program and performs a disguised function. This type of threat gets its name from Greek mythology and the story of the siege of Troy (حصار طروادة). The story tells of how Odysseus and his men conquered Troy by hiding within a giant wooden horse. A Trojan horse program hides within another program or disguises itself as a legal program. This can be accomplished by modifying the existing program or by simply replacing the existing program with a new one. The Trojan horse program functions much the same way as the legal program, but usually it also performs some other function, such as recording sensitive information or providing a trap door.

An example would be a password grabber (أختطاف) program. A password grabber is a program designed to look and function like the normal login prompt that a user sees when first accessing a system. For example, in the screen depicted in Figure below, the user has entered the username john and the correct password. However, the system tells the user that the login is incorrect. When the user tries again it works and he or she is able to log on.



In this example a Trojan horse designed to steal passwords is actually controlling the interaction. The standard login.exe has

been replaced with a Trojan horse program. It looks like the standard login prompt, but what is actually occurring is that the first login prompt is the Trojan horse. When the username and password is entered that information is recorded and stored. Then the Trojan horse program displays the "login incorrect" message and passes the user off to the real login program, so that he or she can actually log on to the system. The user simply assumes that he or she mistyped the password the first time never knowing that her or his username and password have just been stolen.

Trap Doors

A trap door or back door is an undocumented way of gaining access to a system that is built into the system by its designer(s). It can also be a program that has been altered to allow someone to gain privileged access to a system or process.

There have been numerous stories of vendors utilizing trap doors in disputes (خلافات) with customers. One example is the story of a consultant (المستشار) who was contracted to build a system for a company. The consultant designed a trap door into the delivered system. When the consultant and the company got into a dispute over payment, the consultant used the trap door to gain access to the system and disable the system. The company was forced to pay the consultant to get its system turned back on again.

Logic Bombs

A logic bomb is a program or subsection of a program designed with malevolent intent (نوايا سيئة). It is referred to as a logic bomb, because the program is triggered when certain logical conditions are met. This type of attack is almost always perpetrated (يُرتكب) by an insider with privileged access to the network. The perpetrator could be a programmer or a vendor that supplies software. As an example, I once heard a story about a programmer at a large corporation who engineered this type of attack. Apparently, the programmer had been having some trouble at the company at which he worked and was on

trial period . Fearing that he might be fired and with revenge in mind, he added a subroutine to another program. The subroutine was added to a program that ran once a month and was designed to scan the company's human resources employee database to determine if a termination date had been loaded for his employee record. If the subroutine found that a termination date had been loaded, then it was designed to wipe out (delete) the entire system by deleting all files on the disk drives. The program ran every month and so long as his employee record did not have a termination date then nothing would happen. In other words, if he were not fired the program would do no damage. Sure enough this excellent employee was fired, and the next time the logic bomb that he created ran it found a termination date in his employee record and wiped out the system. This is an example of how simple it can be, for one with privileged access to a system, to set up this type of attack.

Port Scanning

A hacker will often case a system to gather information that can later be used to attack the system. One of the tools that hackers often use for this case is a port scanner. A port scanner is a program that listens to well-known port numbers to detect services running on a system that can be exploited to break into the system. There are several port-scanning programs available on the Internet at various sites. They are not difficult to find. Organizations can monitor their system log files to detect port scanning as an introduction to an attack. Most intrusion detection software monitors for port scanning. If you find that your system is being scanned you can trace the scan back to its origination point and perhaps take some pre-emptive (أستباقي) action. However, some scanning programs take a more stealthy approach to scanning that is very difficult to detect. For example, some programs use a SYN scan, which employs a SYN packet to create a half-open connection that doesn't get logged.

Spoofs

Spoofs cover a broad category of threats. In general terms, a spoof entails falsifying one's identity or masquerading as some other individual or entity to gain access to a system or network or to gain information for some other unauthorized purpose. There are many different kinds of spoofs, including, among many others, IP address spoofing, session highjacking (أختطاف), domain name service (DNS) spoofing, sequence number spoofing, and replay attacks.

IP Address Spoofing

Every device on a TCP/IP network has a unique IP address. The IP address is a unique identification of the device, and no two devices on the network can have the same IP address. IP addresses are formatted as four decimal numbers separated by dots (e.g., 147.34.28.103).

IP address spoofing takes advantage of systems and networks that rely on the IP address of the connecting system or device for authentication. For example, packet-filtering routers are sometimes used to protect an internal network from an external untrusted network. These routers will only allow specified IP addresses to pass from the external network to the internal network. If a hacker is able to determine an IP address that is permitted access through the router, he or she can spoof the address on the external network to gain access to the internal network. The hacker in effect masquerades as someone else.

Sequence Number Spoofing

TCP/IP network connections use sequence numbers. The sequence numbers are part of each transmission and are exchanged with each transaction. The sequence number is based upon each computer's internal clock, and the number is predictable because it is based on a set algorithm.

By monitoring a network connection, a hacker can record the exchange of sequence numbers and predict the next set of sequence numbers. With this information, a hacker can insert

himself or herself into the network connection and, effectively, take over the connection or insert misinformation.
The best defense against sequence number spoofing is to encrypt a connection. Encrypting a connection prevents anyone who may be monitoring the network from being able to determine the sequence numbers or any other useful information.

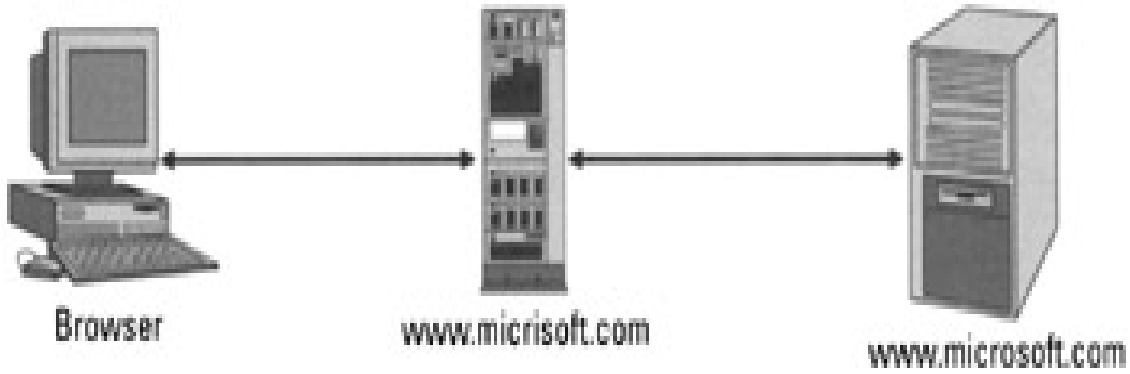
Session Highjacking
Session highjacking is similar to sequence number spoofing. In this process, a hacker takes over a connection session, usually between a client user and a server. This is generally done by gaining access to a router or some other network device acting as a gateway between the legitimate user and the server and utilizing IP spoofing. Since session highjacking usually requires the hacker to gain privileged access to a network device, the best defense to take is to properly secure all devices on the network.

DNS
Domain Name Service (DNS) is a hierarchical name service used with TCP/IP hosts that is distributed and replicated on servers across the Internet. It is used on the Internet and on intranets for translating IP addresses into host names. The host names can be used in URLs.
DNS can be thought of as a lookup table that allows users to specify remote computers by host names rather than their IP addresses. The advantage of DNS is that you don't have to know the IP addresses for all the Internet sites to access the sites. DNS can be configured to use a sequence of name servers, based on the domains in the name being sought, until a match is found. The most commonly deployed DNS server software on the Internet is BIND. DNS is subject to several different spoofs. Two common ones are the man in the middle (MIM) and DNS poisoning. Redirects, another less common attack, rely on the manipulation of the domain name registry itself to redirect a URL.

Man in the Middle Attack (MIM)

In a MIM attack, a hacker inserts himself or herself between a client program and a server on a network. By doing so the hacker can intercept information entered by the client, such as credit card numbers, passwords, and account information. Under one execution of this scheme, a hacker would place himself or herself between a browser and a Web server. The MIM attack, which is also sometimes called Web spoofing, is usually achieved by DNS or hyperlink spoofing.

There are several ways a hacker can launch a MIM attack. One way is to register a URL that is very similar to an existing URL. For example, a hacker could register a URL like www.microsoft.com. When someone who wants to go to the Microsoft Web site at www.microsoft.com mistakenly types in www.microsoft.com they would be brought to a Web site set up by the hacker to look like the Microsoft Web site. Figure below illustrates how:



To Web surfers everything would look normal. They would interact with the counterfeit Web site just as they would with the real site. As the Web surfer enters in choices and information the hacker's Web site can even pass it onto the real site and pass back to the Web surfer the screens that the real site returns.

DNS Poisoning

DNS poisoning exploits a vulnerability in early versions of the Berkeley Internet Name Daemon (BIND). BIND, the most commonly deployed DNS software on the Internet, was developed for BSD UNIX. A network of Internet BIND servers translates native Internet IP addresses to the commonly used names. Prior to version 8.1 of BIND, it was possible to "poison" the table entries of a DNS server with false information.

The information could include a false IP address for a DNS entry in the server's table. The result could be that when someone used that DNS server to "resolve" the URL name, he or she would be directed to the incorrect IP address.

By compromising a DNS server, a hacker can make a legitimate URL point to the hacker's Web site. The Web surfer might enter in www.amazon.com expecting to go to the Amazon.com Web site to purchase a book. The URL www.amazon.com normally points to xxx.xxx.xxx.xxx, but the hacker has compromised a DNS server to point that URL to his or her server. As a result, the Web surfer is brought to the hacker's site and not to Amazon.com.

Redirects

Under another method of DNS attack, hackers compromise a link on someone else's page or set up their own page with false links. In either case, the link could state that it is for a legitimate site, but in reality the link brings the Web surfer to a site set up and controlled by the hacker that looks like the site the Web surfer was expecting.

Replay Attack

A hacker executes a replay attack by intercepting and storing a legitimate transmission between two systems and retransmitting it at a later time. Theoretically, this attack can even be successful against encrypted transmissions. The best defense to this attack is to use session keys, check the time stamp on all transmissions, and employ time-dependent message digests.

Password Cracking

Password cracking is sometimes called a dictionary-based attack. Password crackers are programs that decipher password files. Password-cracking programs are available for most network and computer operating systems. They are able to decipher password files by utilizing the same algorithm used to create the encrypted password. They generally employ a dictionary of known words or phrases, which are also encrypted with the password algorithm.

The password crackers compare each record in the password file against each record in the dictionary file to find a match. When a match is found, a password is found.

Social Engineering

Social engineering, which refers to the non technical methods hackers employ to gain access to systems. Social engineering usually refers to the process of convincing a person to reveal information (such as a password) that enables the hacker to gain access to a system or network.

It is important for every organization to have a policy regarding reveal of passwords. Generally that policy should state that passwords are not to be revealed to anyone.

Another method commonly employed by hackers is referred to as dumpster diving. Dumpster diving may not officially fall under the category of social engineering, but it certainly is lowtech. Dumpster diving refers to the process of gathering information by going through garbage. Computer printout is of particular value in dumpster diving. Hackers look for information such as system account names, source code (particularly if it has passwords), or customer account numbers (for financial institutions). It is important that an organization has proper controls for the disposal of hardcopy records and files. The controls should be codified in a formal policy.

Sniffing

Network sniffing or packet sniffing is the process of monitoring a network in an attempt to gather information that may be useful in an attack. With the proper tools a hacker can monitor the network packets to obtain passwords or IP addresses.

Password sniffing is particularly a threat for users who log into Unix systems over a network. Telnet or remote login is usually employed when logging onto a Unix systems over a network. Telnet and rlogin do not encrypt passwords. As a result, when a user enters in his or her password, it is transmitted in the clear, meaning anyone monitoring the network can read it. In contrast, both Novel and Windows NT workstations encrypt passwords for transmission. There are many tools available to reduce the risk of packet sniffing. Employing network switches instead of traditional hubs is another method to reduce the risk of network sniffing.

Web Site Defacement

Web site defacements are usually achieved by exploiting some incorrect configuration or known vulnerability of the Web server software, or by exploiting some other protocol-based vulnerability of the server's operating system.

An organization's best defense against Web site defacement is to maintain the most recent versions of its Web server software and the server's operating system. Also, an organization should ensure that its Web administrator is properly trained to install and maintain the software.

War Dialing

War dialing is a brute-force method of finding a back door into an organization's network. Most organizations have telephone numbers that are within a specified range and begin with the same prefix. For example, let's consider a company called Acme Networks. The range of telephone numbers for Acme Networks begins at 595-1000 and ends at 595-5000. War dialing usually employs an automated dialing system (a program) to call every

telephone number for the organization, searching for modem connections. The program logs a telephone number whenever it finds a modem. Later after the program has called every extension, the hacker can review the log for modems and go back and attempt to break into the system to which the modem is connected to gain access to the network. This method almost always works for large organizations.

Denial of Service

Denial-of-service attacks are designed to shut down or render inoperable a system or network. The goal of the denial-of-service attack is not to gain access or information but to make a network or system unavailable for use by other users. It is called a denial-of-service attack, because the end result is to deny legitimate users access to network services. Such attacks are often used to exact revenge or to punish some individual or entity. Unlike real hacking, denial-of-service attacks do not require a great deal of experience, skill, or intelligence to succeed. There are many different types of denial-of-service attacks. The following sections present four examples: ping of death, "synchronize sequence number" (SYN) flooding, spamming, and smurfing. These are examples only and are not necessarily the most frequently used forms of denial-of-service attacks.

Ping of Death

The ping-of-death attack, with its melodramatic name, is an example of how simple it can be to launch a denial-of-service attack once a vulnerability has been discovered.

Ping is a TCP/IP command that simply sends out an IP packet to a specified IP address or host name to see if there is a response from the address or host. It is often used to determine if a host is on the network or alive. The typical ping command syntax would be

• ping 145.34.35.56

or

• ping www.acme.net

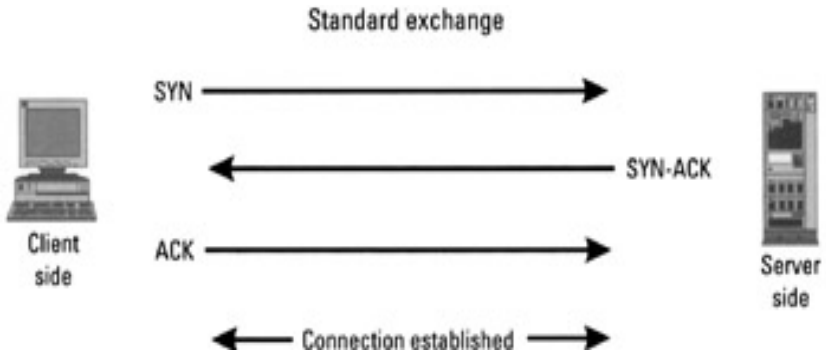
Specifying a large packet in a ping command can cause an overflow in some systems' internals that can result in system crashes. The command syntax would vary depending on the operating system you were using. Below are two examples, one for Windows and the other for Sun Solaris.

• Windows: ping-165527-s 1 hostname
• Solaris: ping -s hostname 65527

Normally it requires a flood of pings to crash a system. Moreover, from firsthand experience I have found that you are just as likely to crash the system from which you are launching the attack as you are to crash the system you are targeting. Nevertheless, the ping-of-death approach may still constitute an effective denial-of-service attack.

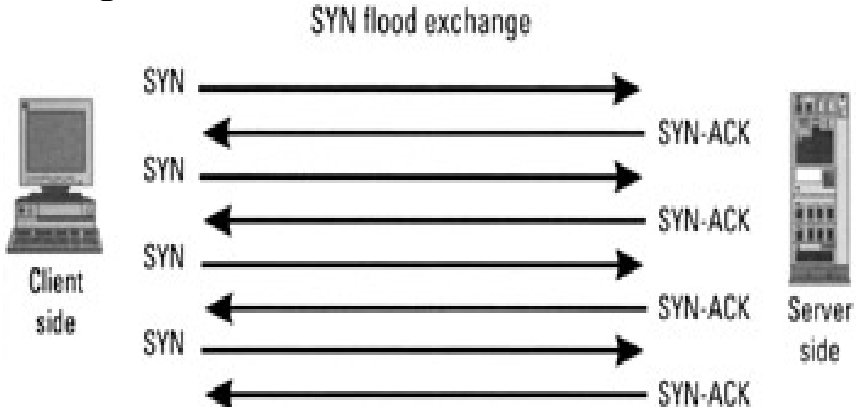
SYN Flooding

SYN flooding is a denial-of-service attack that exploits the three-way handshake that TCP/IP uses to establish a connection. Basically, SYN flooding disables a targeted system by creating many half-open connections. Figure below illustrates how a typical TCP/IP connection is established.



In above figure , the client transmits to the server the SYN bit set. This tells the server that the client wishes to establish a connection and what the starting sequence number will be for the client. The server sends back to the client an acknowledgment (SYN-ACK) and confirms its starting sequence number. The client acknowledges (ACK) receipt of the server's transmission and begins the transfer of data. With SYN flooding a hacker creates many half-open connections by

initiating the connections to a server with the SYN number bit. However, the return address that is associated with the SYN would not be a valid address. The server would send a SYN-ACK back to an invalid address that would not exist or respond. Using available programs, the hacker would transmit many SYN packets with false return addresses to the server. The server would respond to each SYN with an acknowledgment and then sit there with the connection half-open waiting for the final acknowledgment to come back. Figure below illustrates how SYN flooding works.



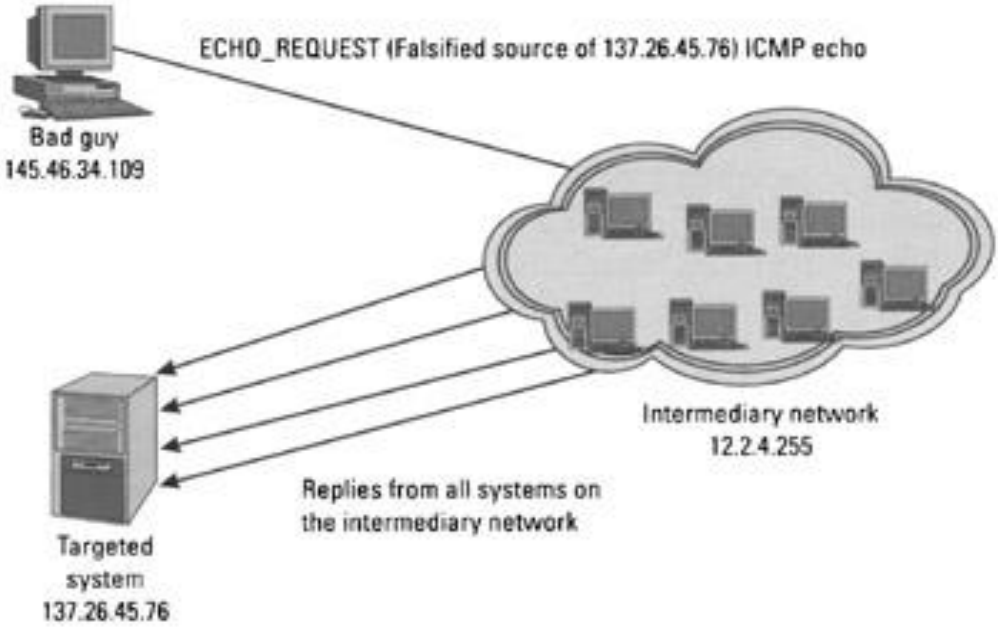
The result from this type of attack can be that the system under attack may not be able to accept legitimate incoming network connections so that users cannot log onto the system. Each operating system has a limit on the number of connections it can accept. In addition, the SYN flood may exhaust system memory, resulting in a system crash. The net result is that the system is unavailable or nonfunctional.

SPAM

SPAM is unwanted e-mail. Anyone who has an e-mail account has received SPAM. Usually it takes the form of a marketing offers from some company trying to sell something we don't want or need. To most of us it is just nothing, but to a server it can also be used as a denial-of-service attack. By falling a targeted system with thousands of e-mail messages, SPAM can eat available network bandwidth, overload CPUs, cause log files to grow very large, and consume all available disk space on a system. Ultimately, it can cause a system to crash.

Smurf Attack

The smurf attack is named after the source code employed to launch the attack (smurf.c). The smurf attack employs forged ICMP echo request packets (A type of hello message to check if the other end is alive) and the direction of those packets to IP network broadcast addresses. The attack issues the ICMP ECHO_REQUEST to the broadcast address of another network. The attack spoofs as the source address the IP address of the system it wishes to target. Figure below illustrates how a smurf attack works.



When the systems on the network to whose broadcast address the ECHO_REQUEST is sent receive the packet with the falsified source address (i.e., the return address), they respond, flooding the targeted victim with the echo replies. This flood can overwhelm the targeted victim's network. Both the intermediate and victim's networks will see degraded performance. The attack can eventually result in the inoperability of both networks.

Denial-of-service attacks are the most difficult to defend against, and, of the possible attacks, they require the least amount of expertise to launch. In general, organization should monitor for anomalous traffic patterns, such as SYN-ACK but no return ACKs. Since most routers filter incoming and

outgoing packets, router-based filtering is the best defense against denial-of-service attacks. Organizations should use packet filters that filter based on destination and sender address.

Encryption on the World Wide Web

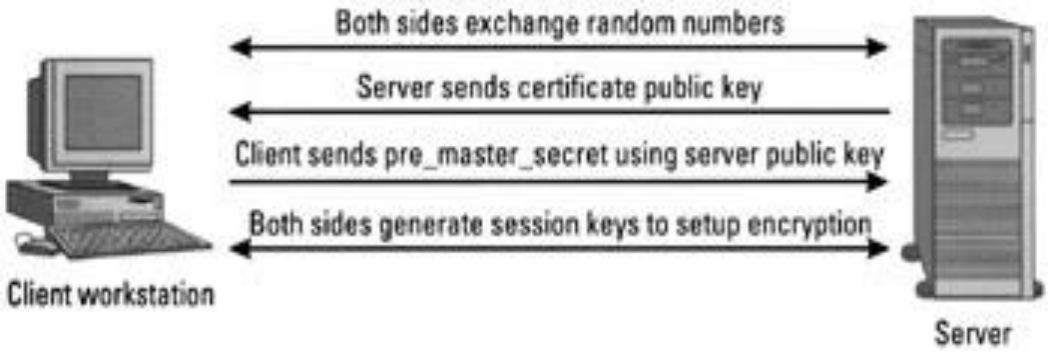
Another area where encryption has been widely deployed is on the Internet or the Web as it has come to be known. Much of the Internet's success and popularity lies in the fact that it is an open global network. At the same time, the fact that it is open and global makes it not very secure. The unique nature of the Internet makes exchanging information and transacting business over it inherently dangerous. The faceless, voiceless, unknown entities and individuals that share the Internet may or may not be who or what they declare to be. In addition, because the Internet is a global network, it does not recognize national borders. As a result, the transacting parties may not be where they say they are and may not be subject to the same laws or regulations.

As stated in earlier, for the exchange of information and for commerce to be secure on any network, especially the Internet, a system or process must be put in place that satisfies requirements for confidentiality, access control, authentication, integrity, and nonrepudiation (The ability to prevent entities from reject information were accessed or altered). These requirements are achieved on the Web through the use of encryption and by employing digital signature technology. There are many examples on the Web of the practical application of encryption. One of the most important is the Secure Sockets Layer (SSL) protocol.

Secure Sockets Layer:

SSL was developed by Netscape (an American computer services company) to provide security when transmitting information on the Internet. Without such a process very few individuals would feel comfortable entering information like credit card numbers on a Web site. SSL was developed to address the security needs of Web surfers.

The risk that a credit card number will be stolen in transit on the Internet is very small. A greater risk is that the credit card number will be stolen from a system on which it is stored. Like most companies, the ISP stored the user account information, including credit card numbers, in a database on a network. That is where the real risk lies. SSL utilizes both asymmetric and symmetric key encryption to set up and transfer data in a secure mode over an unsecured network. When used with a browser client, SSL establishes a secure connection between the client browser and the server. It sets up an encrypted tunnel between a browser and a Web server over which data packets can travel. No one tapping into the connection between the browser and the server can decipher the information passing between the two. Integrity of the information is established by hashing algorithms. Confidentiality of the information is ensured with encryption. Figure below illustrates basically how the process works.



To set up an SSL session both sides exchange random numbers. The server sends its public key with a digital certificate signed by a recognized CA (Certificate Authenticity) confirming to the authenticity of the sender's identity and binding the sender to the public key. The server also sends a session ID. The browser client creates a pre_master_secret key. The client browser encrypts the pre_master_secret key using the server's public key and transmits the encrypted pre_master_secret key to the server. Then both sides generate a session key using the

pre_master_secret and random numbers. The SSL session set-up begins with asymmetric encryption. The server presents the browser client with its public key, which the client uses to encrypt the pre_master_secret. However, once the client sends the encrypted pre_master_secret key back to the server, it employs a session key to establish a secure connection. The initial setup uses asymmetric encryption, but the two parties switch over to symmetric encryption. This is done because symmetric encryption creates much less overhead. Less overhead means better throughput and a faster response time. Asymmetric cryptosystems are much more CPU-dense and would significantly slow the exchange of information. As a result, for automatic exchanges, asymmetric encryption is used initially to establish a secure connection and to authenticate identities (using digital certificates). Once identities are established and public keys are exchanged, the communicating entities switch to symmetric encryption for efficiency. Even with the use of symmetric encryption, network throughput is significantly diminished with SSL. Cryptographic processing is extremely CPU-intensive. Web servers that would normally be able to handle hundreds of connections may only be able to handle a fraction of that when employing SSL. There are SSL accelerators available that can enhance the performance of Web servers that employ SSL. Products from Hewlett-Packard, and others offer solutions that speed up the cryptographic processing. Usually, these products are separate boxes that interface with a server and off-load the SSL process from the server's CPU.

Secure HTTP (SHTTP):

SHTTP is an extension of the HTTP protocol developed by Enterprise Integration Technologies. SHTTP is similar in function to HTTPS (HTTP over SSL) in that it is designed to secure transactions and messages on the Web. There are,

however, several differences: SSL is connection-oriented and operates at the transport level. SSL creates a secure connection over which transactions are transmitted. SHTTP, on the other hand, is transaction-oriented and operates at the application level. Each individual message is encrypted to be transmitted securely. SSL can be used for other TCP/IP protocols such as FTP and TELNET. SHTTP is specifically designed for HTTP and not for other protocols. HTTPS enjoys wide acceptance, while SHTTP's use is very limited. In fact, not all Web browsers support SHTTP. Meanwhile, both Netscape Navigator and Internet Explorer support HTTPS. Most Web server software supports HTTPS, and most e-commerce Web sites use the protocol when obtaining confidential user information. The server is usually authenticated to the client through a digital certificate. The strength of the encryption employed can be set by the server but is usually chosen based on the capability of the client browser. Until relatively recently, there were two types of encryption employed in browsers, depending on whether the browser would be sold in the United States or overseas. The overseas version used weak encryption, while the domestic (home) version used strong encryption. When one refers to weak encryption with SSL and browsers, it usually means 40-bit or 56-bit encryption. Strong encryption refers to 128-bit encryption. The difference in strength between 40-bit encryption and 128-bit encryption is not just 88 bits. In other words, 128-bit encryption is not just 88 times stronger than 40-bit encryption. In fact, 128-bit encryption is more than 300×10^{24} times stronger than 40-bit encryption.

Browsers used to employ two different strengths of encryption because of federal regulations. There were export restrictions on most software, hardware or firmware (is a type of software that provides control, monitoring and data manipulation of engineered products and systems) that included encryption technology. While the export restrictions have been relaxed somewhat, there are still significant rules in place. To export their browsers, companies such as Microsoft and Netscape had to offer versions of their software that employed weak

encryption. Even with the recent changes to U.S. laws regulating the export of cryptographic technology many of the browsers installed today use weak encryption.

Web server software can also be set to use 40-bit or 128-bit encryption. A Web server can be configured to reject browser clients that use a browser set for weak encryption. Web servers can also be configured for strong encryption but still be able to accept browsers that use weak encryption. Therefore, there is really no reason to configure the Web server software to default to 40-bit encryption.

There are several ways to tell if a site uses encryption and the strength of the encryption employed. The things to look for vary depending on whether you are using Netscape's Navigator or Microsoft's Internet Explorer and which version of either software you are using.

E-Mail Security

The most important thing to remember about standard e-mail is that it is very unsecure. Very often, e-mail by necessity must traverse many networks to reach its destination. During transit, an e-mail message may pass through many mail servers. As a result, it is vulnerable to interception, replication, disclosure (declare) , or modification anywhere along its prescribed path. It can be copied and stored to be retrieved at a later date. In fact, each mail server an e-mail message passes through may be making a copy of the message before it is forwarded. Whether you use your corporation's e-mail system (نظـام بريد المؤسسـة) or an e-mail account provided by an ISP, your e-mail messages reside on an e-mail server. Even if you download your e-mail to your local disk drive, those messages probably have already been backed up and stored on some other media. If your corporation uses an e-mail portal (بوابـة) to the Internet, then your e-mails are being copied there before being forwarded onto the appropriate Internet address. The point that I'm trying to make is that if you think your e-mail is secure and confidential, then you are greatly mistaken.

In addition to disclosure (declaration), e-mail messages are vulnerable to alteration. Anywhere along the path, an e-mail message can be intercepted and modified before being forwarded. Common email provides no method of detecting messages that have been modified in transit. Messages that have been copied and stored can also be modified and retransmitted at a later time.

Another vulnerability lies in the fact that e-mail identities are very easy to forge. With common e-mail, there is no built-in process to ensure that the sender of a message is who he or she claims to be. E-mail headers are easy enough to spoof (falsifying one's identity), and most individuals do not know what to look for when receiving an e-mail. One solution to these problems is to use secure e-mail. The basic requirements of secure email are described as follows:

• *Nondisclosure of the contents of the e-mail message:* This is usually achieved by employing some encryption technology.

• *Message integrity:* In other words, secure e-mail ensures that the message has not been altered during transit and provides a method to certify the message's integrity. This is usually achieved by employing some hashing or message digest algorithm.
• *Verification of sender:* Secure e-mail provides some method to ensure the identity of the sender with a high degree of confidence. This is usually achieved by employing digital signature technology.
• *Verification of recipient:* This can be achieved by employing public key encryption.

The following sections offer a very brief overview of just some of the options available for secure e-mail.

Secure E-Mail Protocols
When it comes to secure e-mail standards there is no lack of standards. In fact, that is the problem. There are several competing standards and products from which you can choose. Some of the standards and products that are available are listed as follows:
• PGP;
• PEM;
• Secure multipurpose Internet mail extension (MIME) (S/MIME);
• MIME object security service (MOSS);
• Message security protocol (MSP).

These competing standards and products are one of the primary reasons that secure e-mail has not been widely implemented. The standards are not interoperable. If you use PGP to send someone a secure e-mail, but the recipient employs S/MIME then the recipient will not be able to open and read the message, let alone authenticate the sender of the message.

Pretty Good Privacy (PGP):

PGP is a remarkable phenomenon. Largely the effort of a single person, Phil Zimmermann, PGP provides a confidentiality and authentication service that can be used for electronic mail and file storage applications. In essence, Zimmermann has done the following:

1. Selected the best available cryptographic algorithms as building blocks.

2. Integrated these algorithms into a general-purpose application that is independent of operating system and processor and that is based on a small set of easy-to-use commands.

3. Made the package and its documentation, including the source code, freely available via the Internet.

4. Provide a fully compatible, low-cost commercial version of PGP.

PGP has grown explosively and is now widely used. A number of reasons can be cited for this growth:

1. It is available free worldwide in versions that run on a variety of platforms, including Windows, UNIX, Macintosh, and many more.

2. It is based on algorithms that have survived extensive public review and are considered extremely secure. Specifically, the package includes RSA, DSS, and Diffie-Hellman for public-key encryption;CAST-128, IDEA, and 3DES for symmetric encryption; and SHA-1 for hash coding.

3. It has a wide range of applicability, that encrypting files and messages to individuals who wish to communicate securely with others worldwide over the Internet and other networks.

4. It was not developed by, nor is it controlled by, any governmental or standards organization. This makes PGP attractive.

5. PGP is now on an Internet standards track (RFC 3156; MIME Security with Open PGP).

IP Security:

In 1994, the Internet Architecture Board (IAB) issued a report titled "Security in the Internet Architecture". The report identified key areas for security mechanisms. Among these were the need to secure the network infrastructure from unauthorized monitoring and control of network traffic and the need to secure end-user-to-end-user traffic using authentication and encryption mechanisms.

To provide security, the IAB included authentication and encryption as necessary security features in the next-generation IP, which has been issued as IPv6. Fortunately, these security capabilities were designed to be usable both with the current IPv4 and the future IPv6. This means that vendors can begin offering these features now, and many vendors now do have some IPsec capability in their products. The IPsec specification now exists as a set of Internet standards.

Applications of IPsec:

IPsec provides the capability to secure communications across a LAN, across private and public WANs, and across the Internet. Examples of its use include:

• **Secure branch office connectivity over the Internet:**

A company can build a secure virtual private network over the Internet or over a public WAN. This enables a business to rely heavily on the Internet and reduce its need for private networks, saving costs and network management overhead.

• **Secure remote access over the Internet:**

An end user whose system is equipped with IP security protocols can make a local call to an Internet Service Provider (ISP) and gain secure access to a company network. This reduces the cost of charges for traveling employees.

• **Establishing extranet and intranet connectivity with partners:**
IPsec can be used to secure communication with other organizations, ensuring authentication and confidentiality and providing a key exchange mechanism.

• **Enhancing electronic commerce (تجارة) security:**
Even though some Web and electronic applications have built-in security protocols, the use of IPsec enhances that security. IPsec guarantees that all traffic designated by the network administrator is both encrypted and authenticated, adding an additional layer of security to whatever is provided at the application layer.

The principal feature of IPsec that enables it to support these varied applications is that it can encrypt and/or authenticate all traffic at the IP level. Thus, all distributed applications (including remote logon, client/server, e-mail, file transfer, Web access, and so on) can be secured.

Benefits of IPsec:
Some of the benefits of IPsec:

• When IPsec is implemented in a firewall or router, it provides strong security that can be applied to all traffic.

• IPsec in a firewall is resistant to bypass if all traffic from the outside must use IP and the firewall is the only means of entrance from the Internet into the organization.

• IPsec is below the transport layer (TCP, UDP) and so is transparent to applications. There is no need to change software on a user or server system when IPsec is implemented in the firewall or router.

• IPsec can be transparent to end users. There is no need to train users on security mechanisms, issue keying material on a per-user basis, or revoke keying material when users leave the organization.

• IPsec can provide security for individual users if needed. This is useful for offsite workers and for setting up a secure virtual subnetwork within an organization for sensitive applications.

Intruders (الدخلاء):

One of the two most publicized threats to security is the intruder (the other is viruses), often referred to as a hacker (القرصان) or cracker (المكسر). There are three classes of intruders:

• **Masquerader (المتخفي):** An individual who is not authorized to use the computer and who penetrates a system's access controls to exploit a legitimate user's account.

• **Misfeasor:** A legitimate user who accesses data, programs, or resources for which such access is not authorized, or who is authorized for such access but misuses his or her privileges.

• **Clandestine (سري) user:** An individual who seizes (يستولي) supervisory control (السيطرة الأشرافية) of the system and uses this control to evade (يتهرب) auditing and access controls or to suppress (قمع) audit collection.

The masquerader is likely to be an outsider; the misfeasor generally is an insider; and the clandestine user can be either an outsider or an insider.

Intruder attacks range from the benign (حميدة) to the serious. At the benign end of the scale, there are many people who simply wish to explore internets and see what is out there. At the serious end are individuals who are attempting to read privileged data, perform unauthorized modifications to data, or disrupt the system.

Firewalls:
Firewalls can be an effective means of protecting a local system or network of systems from network-based security threats while at the same time affording access to the outside world via wide area networks and the Internet.

Firewalls Characteristics:
1. All traffic from inside to outside, and vice versa, must pass through the firewall. This is achieved by physically blocking all access to the local network except via the firewall. Various configurations are possible.

2. Only authorized traffic, as defined by the local security policy, will be allowed to pass. Various types of firewalls are used, which implement various types of security policies.

3. The firewall itself is immune to penetration. This implies the use of a hardened system with a secured operating system. Trusted computer systems are suitable for hosting a firewall.

Limitations of Firewalls:
1. The firewall cannot protect against attacks that bypass the firewall. Internal systems may have dial-out capability to connect to an ISP. An internal LAN may support a modem pool (تجمع المودم) that provides dial-in capability.

2. The firewall may not protect fully against internal threats, such as a disgruntled employee (الموظف الناقم)or an employee who unwittingly(عن غير قصد)cooperates with an external attacker.

3. An improperly secured wireless LAN may be accessed from outside the organization. An internal firewall that separates portions of an enterprise network (مشروع الشبكة) cannot guard against wireless communications between local systems on different sides of the internal firewall.

4. A laptop, or portable storage device may be used and infected outside the corporate network, and then attached and used internally.

Types of Firewalls:

A firewall may act as a packet filter. It can operate as a positive filter, allowing to pass only packets that meet specific criteria, or as a negative filter, rejecting any packet that meets certain criteria. Depending on the type of firewall, it may examine one or more protocol headers in each packet, the payload of each packet, or the pattern generated by a sequence of packets.

A packet filtering firewall applies a set of rules to each incoming and outgoing IP packet and then forwards or discards the packet. The firewall is typically configured to filter packets going in both directions (from and to the internal network). Filtering rules are based on information contained in a network packet:

• **Source IP address:** The IP address of the system that originated the IP packet (e.g., 192.178.1.1)

• **Destination IP address:** The IP address of the system the IP packet is trying to reach (e.g., 192.168.1.2)

• **Source and destination transport-level address:** The transport-level (e.g., TCP or UDP) port number, which defines applications such as SNMP or TELNET.

• **IP protocol field:** Defines the transport protocol.

• **Interface:** For a firewall with three or more ports, which interface of the firewall the packet is destined for (معد لهذا الغرض).

Biometrics: *Identification and Authentication*

When we talk about an identification and authentication scheme that relies on "something you are," we mean biometrics. Biometric authentication is the process of using some physical characteristic, trait, aspect of physical being, or behavior to authenticate one's identity. The most commonly known example is the process of employing fingerprints to identify an individual.

Biometric authentication usually fits into one of two general categories. The first is physical characteristic recognition (PCR), which relies upon a physical characteristic such as a fingerprint, retina or iris scan, voiceprint, or facial geometry for identification and authentication.

The second category is behavioral characteristic recognition (BCR). BCR relies on a behavioral characteristics such as how a person types at a keyboard, writes, or signs his or her name. In general, PCR is much more widely implemented than BCR.

Biometric Identification Reliability:

When considering a biometric authentication system, there are two critical characteristics that you should review before deploying any system. They are listed as follows.

- False acceptance rate (FAR);
- False rejection rate (FRR).

The FAR is the rate at which a system incorrectly accepts or recognizes a would-be user as authorized to access the system when in fact he or she are not. In other words, how often does the system let someone in that it should keep out? Most manufacturers of biometric authentication devices list the FAR for their products. If not, you should be able to request it from the manufacturer. Very often the FAR is listed as a percentage.

The FAR for any biometric identification and authentication system should be closely scrutinized (تدقيق). A manufacturer may list a FAR that appears to be very small, but the numbers can be deceiving. For example, a FAR of only 1% means that

one time out of 100 a system will incorrectly accept an unauthorized user. That false acceptance rate percentage is much too high. A FAR of 1% means that if a hacker makes 100 attempts he or she will be successful at least one time. Even a FAR of 0.1% is too high to be acceptable. That means that one in 10,000 attempts will be incorrectly accepted.
Another important characteristic of any biometric identification and authentication system is the FRR, the rate at which a system incorrectly rejects a legitimate user. While it is not as critical as FAR, the FRR is important to the successful deployment of any biometric authentication system. If the FRR of a system is too high, it can cause end-user frustration (أحباط).
The frustration can lead users to circumvent (تحايل) proper authentication procedures to avoid the biometric system.

When evaluating any biometric authentication scheme you need to take into account how it will handle the natural changes people experience.

Intrusion Detection
Competent system administrators have always monitored their systems for intrusions. The process usually entailed reviewing logs on a daily basis. Intrusions were sufficiently rare that after-the-fact reviews were usually adequate to address any possible problems.
In general terms an "intrusion" can be defined as an unauthorized attempt or achievement to access, alter, render unavailable, or destroy information on a system or the system itself.
Basically, an intrusion is somebody attempting to break into or misuse a system. Some observers differentiate misuse and intrusion. The term intrusion is usually used in reference to attacks that originate from outside an organization. Misuse is usually used to describe an attack that originates from the internal network. However, not everyone makes this differentiation.

Intrusion detection is the art of detecting unauthorized, inappropriate, or anomalous (شاذ) activity.