BSc in Computer Engineering
CMP4103
Computer Systems and Network Security

Lecture 6

Systems: Threats, Vulnerabilities and Risks

Eng Diarmuid O'Briain, CEng, CISSP

Department of Electrical and Computer Engineering,
College of Engineering, Design, Art and Technology,
Makerere University

Table of Contents

1. Application Issues

1.1 Non distributed Environment

This is the traditional computing environment where individual computers store programs and execute them for a local user. From this environment the user may use networked resources like printers, e-mail, web-servers. Key to this model is that all user executed code is either stored on the machine or on a file system accessible by the machine. Threats to this environment are:

1.1.1 Viruses

A computer virus is a computer program that can copy itself and infect a computer. The term "virus" is also commonly but erroneously used to refer to other types of malware, adware, and spyware programs that do not have the reproductive ability. A true virus can only spread from one computer to another (in some form of executable code) when its host is taken to the target computer; for instance because a user sent it over a network or the Internet, or carried it on a removable medium such as a floppy disk, CD, DVD, or USB drive. Viruses can increase their chances of spreading to other computers by infecting files on a network file system or a file system that is accessed by another computer.

Master Boot Record (MBR)/GUID Partition Table (GTP) Viruses

Many destructive viruses damage the MBR or GTP and make it impossible to start the computer from the hard disk. Because the code in the MBR or GPT executes before any Operating System (OS) is started, no OS can detect or recover from corruption of the MBR or GPT.

File Infector Viruses

Although there are many different kinds of file infector viruses, most of them operate the same and take the following course of actions:

- Once a user executes an infected file, the virus copies the file and places into an area where it can be executed. In most cases, this would be the RAM.
- The malicious code runs first while the infected file remains quiescent.
- The virus then copies itself in a location separate from where the infection occurred, allowing it to continuously infect files as the user functions other programs.
- When the initial process is set in to place, the virus grants control back to the infected file.
- When a user opens another application, the dormant virus proceeds to run again. It then inserts a copy of itself into files that were previously uninfected which enables the cycle to repeat consistently.

Macro Viruses

A macro virus is a virus that is written in a macro language: that is to say, a language built into a software application such as a word processor. Since some applications (notably, but not exclusively, the parts of Microsoft Office) allow macro programs to be embedded in documents, so that the programs may be run automatically when the document is opened, this provides a distinct mechanism by which viruses can be spread. This is why it may be dangerous to open unexpected attachments in e-mails.

Anti-virus software and other preventive measures

Many users install anti-virus software that can detect and eliminate known viruses after the computer downloads or runs the executable. There are two common methods that an anti-virus software application uses to detect viruses. The first, and by far the most common method of virus detection is using a list of virus signature definitions. This works by examining the content of the computer's memory (its RAM, and boot sectors) and the files stored on fixed or removable drives (hard drives, floppy drives), and comparing those files against a database of known virus "signatures". The disadvantage of this detection method is that users are only protected from viruses that pre-date their last virus definition update. The second method is to use a heuristic algorithm to find viruses based on common behaviours. This method has the ability to detect viruses that anti-virus security firms have yet to create a signature for.

Some anti-virus programs are able to scan opened files in addition to sent and received e-mails '*on the fly*' in a similar manner. This practice is known as "on-access scanning." Anti-virus software does not change the underlying capability of host software to transmit viruses. Users must update their software regularly to patch security holes. Anti-virus software also needs to be regularly updated in order to prevent the latest threats.

One may also minimise the damage done by viruses by making regular backups of data (and the OSs) on different media, that are either kept unconnected to the system (most of the time), read-only or not accessible for other reasons, such as using different file systems. This way, if data is lost through a virus, one can start again using the backup (which should preferably be recent).

If a backup session on optical media like CD and DVD is closed, it becomes read-only and can no longer be affected by a virus (so long as a virus or infected file was not copied onto the CD/DVD). Likewise, an OS on a bootable CD can be used to start the computer if the installed OSs become unusable. Backups on removable media must be carefully inspected before restoration.

Multipartite Virus

A virus that uses more than one technique to spread itself. i.e. A virus that initially writes itself into files like .com and .exe files (file infector) and then later when the opportunity presents itself it writes code to the MBR (boot sector virus).

Stealth Virus

These virus write themselves into the OS itself to avoid being detected by anti-virus software.

Polymorphic Virus

This is a virus that modifies its own code as it traverses systems. This allows new iterations of the virus to avoid having a known signature of earlier viruses that anti-virus software can detect.

Encrypted Virus

Encrypted viruses use cryptographic techniques in order to hide their signature from anti-virus software. They alter the way they are stored on disks by having a *Virus Decryption Routine* which contains the code necessary to decrypt and upload the virus code stored elsewhere on the disk, camouflaged by the cipher.

1.1.2 Hoax

A hoax is a spam e-mail that warns of a virus that is spread by friend to friend and warns of a virus (real though more often imaginary) that is very destructive.

1.1.3 Trojan horse

A Trojan horse, or trojan for short, is a term used to describe malware that appears, to the user, to perform a desirable function but, in fact, facilitates unauthorised access to the user's computer system. Trojan horses are not self replicating, which distinguishes them from viruses and worms.

Once a Trojan horse has been installed on a target computer system it is possible for a hacker to access it remotely and perform various operations. The operations that a hacker can perform are limited by user privileges on the target computer system and the design of the Trojan horse. They include:

- Use of the machine as part of a Botnet (e.g., to perform Distributed Denial-of-service (DDoS) attacks)
- Data Theft (e.g., passwords, security codes, credit card information)
- Installation of software (including other malware)
- Downloading of files
- Uploading of files
- Deletion of files
- Modification of files
- Keystroke logging
- Viewing the user's screen

- Wasting computer storage space

According to a survey conducted by BitDefender from January to June 2009, "Trojan-type malware is on the rise, accounting for 83-percent of the global malware detected in the wild".

1.1.4 Logic Bomb

A logic bomb is a piece of code intentionally inserted into a software system that will set off a malicious function when specified conditions are met. For example, a programmer may hide a piece of code that starts deleting files (such as a salary database trigger), should they ever be terminated from the company.

Software that is inherently malicious, such as viruses and worms, often contain logic bombs that execute a certain payload at a pre-defined time or when some other condition is met. This technique can be used by a virus or worm to gain momentum and spread before being noticed. Many viruses attack their host systems on specific dates, such as Friday the 13th or April Fool's Day. Trojans that activate on certain dates are often called "*time bombs*".

To be considered a logic bomb, the payload should be unwanted and unknown to the user of the software. As an example, trial programs with code that disables certain functionality after a set time are not normally regarded as logic bombs.

1.1.5 Worms

A computer worm is a self-replicating computer program. It uses a network to send copies of itself to other nodes (computers on the network) and it may do so without any user intervention. Unlike a virus, it does not need to attach itself to an existing program. Worms almost always cause at least some harm to the network, if only by consuming bandwidth, whereas viruses almost always corrupt or devour files on a targeted computer.

1.2 Spyware and Adware

1.2.1 Spyware

Spyware is a type of malware that is installed on computers and collects information about users without their knowledge. The presence of spyware is typically hidden from the user. Typically, spyware is secretly installed on the user's personal computer. Sometimes, however, spywares such as key-loggers are installed by the owner of a shared, corporate, or public computer on purpose in order to secretly monitor other users.

While the term spyware suggests software that secretly monitors the user's computing, the functions of spyware extend well beyond simple monitoring. Spyware programs can collect various types of personal information, such as Internet surfing habits and sites that have been visited, but can also interfere with user control of the computer in other ways, such as installing additional software and redirecting Web browser activity. Spyware is known to change computer settings, resulting in slow connection speeds, different home pages, and/or loss of Internet or functionality of other programs. In an attempt to increase the understanding of spyware, a more formal classification of its included software types is captured under the term privacy-invasive software.

In response to the emergence of spyware, a small industry has sprung up dealing in anti-spyware software. Running anti-spyware software has become a widely recognised element of computer security practices for computers, especially those running Microsoft Windows. A number of jurisdictions have passed anti-spyware laws, which usually target any software that is surreptitiously installed to control a user's computer.

1.2.2 Adware

Adware or advertising supported software is any software package which automatically plays, displays, or downloads advertisements to a computer after the software is installed on it or while the application is being used. Some types of adware are also spyware and can be classified as privacy invasive software.

1.3 Password Attacks

Password cracking is the process of recovering passwords from data that has been stored in or transmitted by a computer system. A common approach is to repeatedly try guesses for the password. The purpose of password cracking might be to help a user recover a forgotten password (though installing an entirely new password is less of a security risk, but involves system administration privileges), to gain unauthorised access to a system, or as a preventive measure by system administrators to check for easily crackable passwords. On a file-by file basis, password cracking is utilised to gain access to digital evidence for which a judge has allowed access but the particular file's access is restricted.

1.4 Dictionary Attacks

The distinction between guessing, dictionary and brute force attacks is not strict. They are similar in that an attacker goes through a list of candidate passwords one by one, the list may be explicitly enumerated or implicitly defined, can incorporate knowledge about the victim, and can be linguistically derived. Each of the three approaches, particularly 'dictionary attack', is frequently used as an umbrella term to denote all the three attacks and the spectrum of attacks encompassed by them.

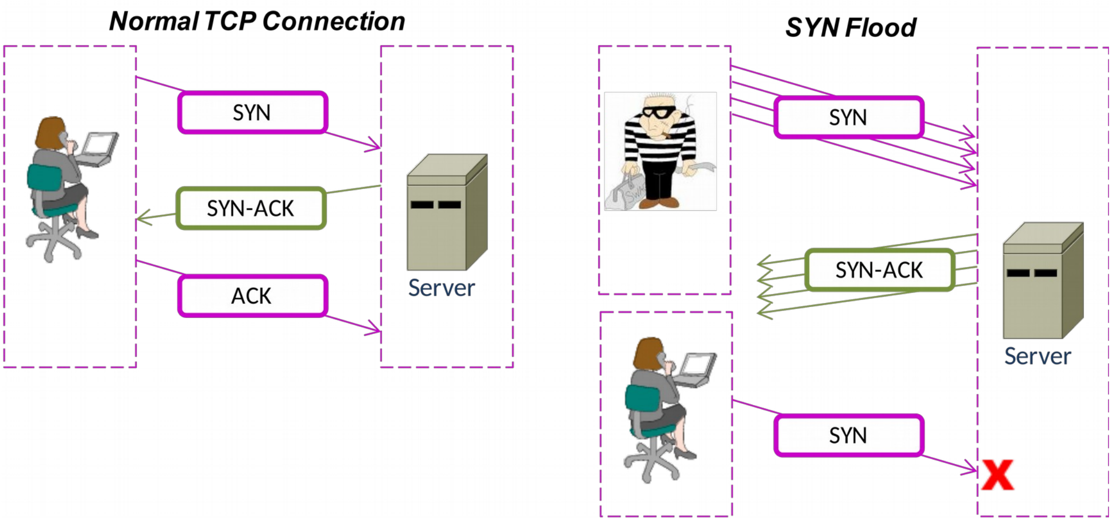
1.5 Social Engineering

Social engineering is the act of manipulating people into performing actions or divulging confidential information. While similar to a confidence trick or simple fraud, the term typically applies to trickery or deception for the purpose of information gathering, fraud, or computer system access; in most cases the attacker never comes face to face with the victim.

2. Denial of Service (DoS) Attacks

A DoS attack is an attempt to make a computer resource unavailable to its intended users. Although the means to carry out, motives for, and targets of a DoS attack may vary, it generally consists of the concerted efforts of a person or people to prevent an Internet site or service from functioning efficiently or at all, temporarily or indefinitely. Perpetrators of DoS attacks typically target sites or services hosted on high-profile web servers such as banks, credit card payment gateways, and even root nameservers.

2.1 SYN Flood Attack



A normal connection between a user and a server. The three-way handshake is correctly performed.

SYN Flood. The attacker sends several packets but does not send the "ACK" back to the server. The connections are hence half-opened and consuming server resources. Alice, a legitimate user, tries to connect but the server refuses to open a connection resulting in a denial of service.

2.1.1 Hping3

hping3 is a network tool capable of sending custom TCP/IP packets and to display target replies like ping program does with ICMP replies. hping3 handle fragmentation, arbitrary packets body and size and can be used in order to transfer files encapsulated under supported protocols.

```
Linux:~# sudo apt-get install hping3
```

```
Linux:~# hping3 --count 10000 --data 120 --syn --win 64
            --destport 21 --flood -rand-source  www.attacktarget.com

HPING www.attacktarget.com (lo 127.0.0.1): S set, 40 headers + 120
data bytes

hping in flood mode, no replies will be shown

--- www.hping3testsite.com hping statistic ---

1189112 packets transmitted, 0 packets received, 100% packet loss

round-trip min/avg/max = 0.0/0.0/0.0 ms
```

This attack was pretty simple to execute you must agree, the command sent 100000 SYN only packets with a TCP window size of 64 to port 21, the FTP port, could be any port however, 80 for example for webserver. The command floods packets as fast as possible, without showing incoming replies and uses a random source IP address to camouflage the source. The target in this instance being *www.attacktarget.com***.**

Note that in hping3 flood mode, replies are not received, why ? Well the use of --rand-souce means the source IP address is no longer the real source.

2.1.2 Nping

Nping, part of nmap is an open-source tool for network packet generation, response analysis and response time measurement. Nping allows users to generate network packets of a wide range of protocols, letting them tune virtually any field of the protocol headers. While Nping can be used as a simple ping utility to detect active hosts, it can also be used as a raw packet generator for network stack stress tests, ARP poisoning, Denial of Service attacks, route tracing, and other purposes.

```
Linux:~# sudo apt-get install nmap

Linux:~# nping --tcp-connect --rate=90000 --count 900000
               --reduce-verbosity www.attacktarget.com

Starting Nping 0.6.40 ( http://nmap.org/nping ) at 2016-01-21 12:08
EAT

Max rtt: N/A | Min rtt: N/A | Avg rtt: N/A

TCP connection attempts: 900000 | Successful connections: 0 | Failed:
900000 (100.00%)

Nping done: 1 IP address pinged in 260.25 seconds
```

This command asks the underlying operating system to establish a connection with the target machine and port by issuing a connect system call. It sends 900,000 packets at a rate of 90,000 packets per second. Reduced verbosity means that no debug information is to be fed back. Note there were no successful connections, this is expected.

3. Distributed Environment

The Distributed Computing Environment (DCE) is a software system developed in the early 1990s by a consortium that included Apollo Computer (later part of Hewlett-Packard), IBM, Digital Equipment Corporation, and others. The DCE supplies a framework and toolkit for developing client/server applications. The framework includes a Remote Procedure Call (RPC) mechanism known as DCE/RPC, a naming (directory) service, a time service, an authentication service and a Distributed File System (DFS) known as DCE/DFS.

The largest unit of management in DCE is a cell. Major components of DCE within every cell are:

- The Security Server that is responsible for authentication
- The Cell Directory Server (CDS) that is the repository of resources and ACLs
- The Distributed Time Server (DTS) that provides an accurate clock for proper functioning of the entire cell.

Modern DCE implementations such as IBM's are fully capable of interoperating with Kerberos as the security server, LDAP for the CDS and the Network Time Protocol (NTP) implementations for the time server.

3.1.1 Agents (bots)

A software agent is a piece of software that acts for a user or other program in a relationship of agency. Such "*action on behalf of*" implies the authority to decide which (and if) action is appropriate. The idea is that agents are not strictly invoked for a task, but activate themselves.

The largest use of bots is in web spidering, in which an automated script fetches, analyses and files information from web servers at many times the speed of a human. Each server can have a file called *robots.txt*, containing rules for the spidering of that server that the bot is supposed to obey.

A malicious use of bots is the coordination and operation of an automated attack on networked computers, such as a DoS attack by a botnet. Internet bots can also be used to commit click fraud. A spambot is an Internet bot that attempts to spam large amounts of content on the Internet, usually adding advertising links.

There are malicious bots (and botnets) of the following types:

- Spambots that harvest email addresses from Internet forums, contact forms or guestbook pages
- Downloader programs that suck bandwidth by downloading entire web sites
- Web site scrapers that grab the content of web sites and re-use it without permission on automatically generated doorway pages
- Viruses and worms
- DDoS attacks
- Botnets / zombie computers; etc.
- File-name modifiers on peer-to-peer file-sharing networks. These change the names of files (often containing malware) to match user search queries.

3.1.2 Applets

An applet is any small application that performs one specific task, sometimes running within the context a larger program perhaps as a plug-in. However, the term typically also refers to programs written in the Java programming language which are included in an HTML page.

Java Applets are used to provide interactive features to web applications that cannot be provided by HTML. Since Java's byte-code is platform independent, Java applets can be executed by browsers for many platforms, including Windows, Unix, Mac OS and Linux. When a Java technology-enabled web browser views a page that contains an applet, the applet's code is transferred to the clients system and executed by the browser's Java Virtual Machine (JVM).

Obviously the Security concern is around the fact that a remote machine is sending code to another machine for execution. Sun Microsystems created the concept of a Sandbox for the JVM where the sandbox provides a tightly controlled set of resources for guest programs to run in, such as scratch space on disk and memory. Network access, the ability to inspect the host system or read from input devices are usually disallowed or heavily restricted.

3.1.3 ActiveX

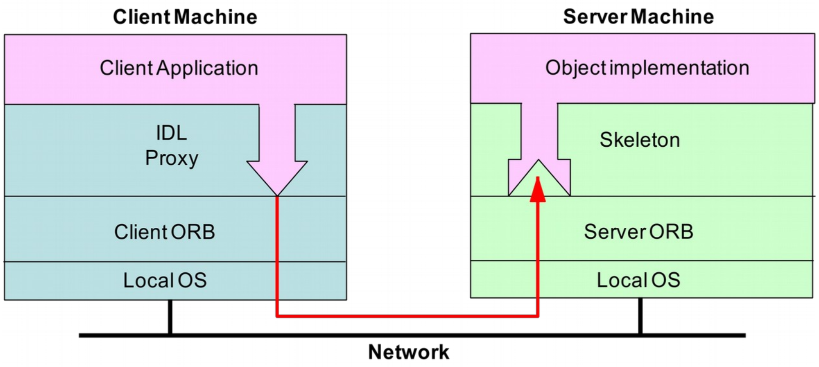
Microsoft’s answer to Applets. ActiveX is a framework for defining reusable software components (known as controls) that perform a particular function or a set of functions in Microsoft Windows in a way that is independent of the programming language used to implement them. ActiveX controls serve to create distributed applications that work over the Internet through web browsers. Examples include customised applications for gathering data, viewing certain kinds of files, and displaying animation.

Malwares, such as computer viruses and spywares, can be accidentally installed from malicious websites using ActiveX controls (drive-by downloads).

3.1.4 Object Request Broker (ORB)

This is a piece of middleware software that allows programmers to make program calls from one computer to another via a network. ORBs promote interoperability of distributed object systems because they enable users to build systems by piecing together objects from different vendors that communicate with each other via the ORB.

ORBs handle the transformation of in-process data structures to and from the byte sequence, which is transmitted over the network. This is called marshalling or serialisation.



Some ORBs, such as Common Object Request Broker Architecture (CORBA) compliant systems, use an Interface Description Language (IDL) to describe the data which is to be transmitted on remote calls.

In object-oriented languages, the ORB takes the form of an object with methods enabling connection to the objects being served. After an object connects to the ORB, the methods of that object become accessible for remote invocations. The ORB requires some means of obtaining the network address of the object that has now become remote. The typical ORB also has many other methods.

3.1.5 Distributed Common Object Model (DCOM)

DCOM is a proprietary Microsoft technology for communication among software components distributed across networked computers. DCOM, which originally was called 'Network Object Linking and Embedding (Network OLE)', extends Microsoft's Common Object Model COM, and provides the communication substrate under Microsoft's COM+ application server infrastructure. It has been deprecated in favour of the Microsoft .NET Framework.

3.1.6 .NET Framework

This is a software framework that can be installed on computers running Microsoft Windows OS. It includes a large library of coded solutions to common programming problems and a Virtual Machine (VM) that manages the execution of programs written specifically for the framework. The .NET Framework is intended to be used by most new applications created for the Windows platform. It also replaces DCOM.

3.2 Distributed DoS (DDoS) Attacks

3.2.1 Smurf Attack

In such an attack, a perpetrator sends a large volume of ICMP echo request (ping) traffic to IP broadcast addresses, all of which have a spoofed source IP address of the intended victim. If the routing device delivering traffic to those broadcast addresses delivers the IP broadcast to all hosts (for example via a layer 2 broadcast), most hosts on that IP network will take the ICMP echo request and reply to it with an echo reply, multiplying the traffic by the number of hosts responding. On a multi-access broadcast network, hundreds of machines might reply to each packet.

In the late 1990s, many IP networks would participate in Smurf attacks (that is, they would respond to pings to broadcast addresses). Today, thanks largely to the ease with which administrators can make a network immune to this abuse, very few networks remain vulnerable to Smurf attacks.

The fix is two-fold:

- Configure individual hosts and routers not to respond to ping requests or broadcasts.
- Configure routers not to forward packets directed to broadcast addresses. Until 1999, standards required routers to forward such packets by default, but in that year, the standard was changed to require the default to be not to forward.

Another proposed solution, to fix this as well as other problems, is network ingress filtering which rejects the attacking packets on the basis of the forged source address.

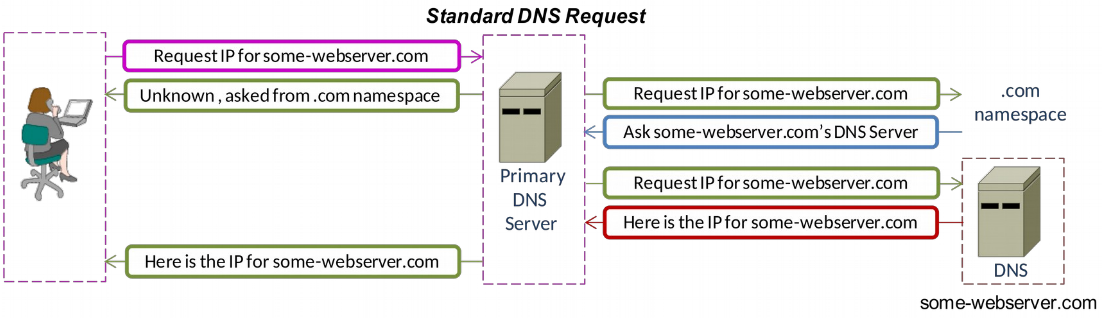
To prevent such attacks on a Cisco router add the command:

```
Router(config-if)# no ip directed-broadcast
```

This example does not prevent a network from becoming the target of Smurf attack, it merely prevents the network from "attacking" other networks, or better said, taking part in a Smurf attack.

3.2.2 DNS Amplification Attacks

Normal DNS



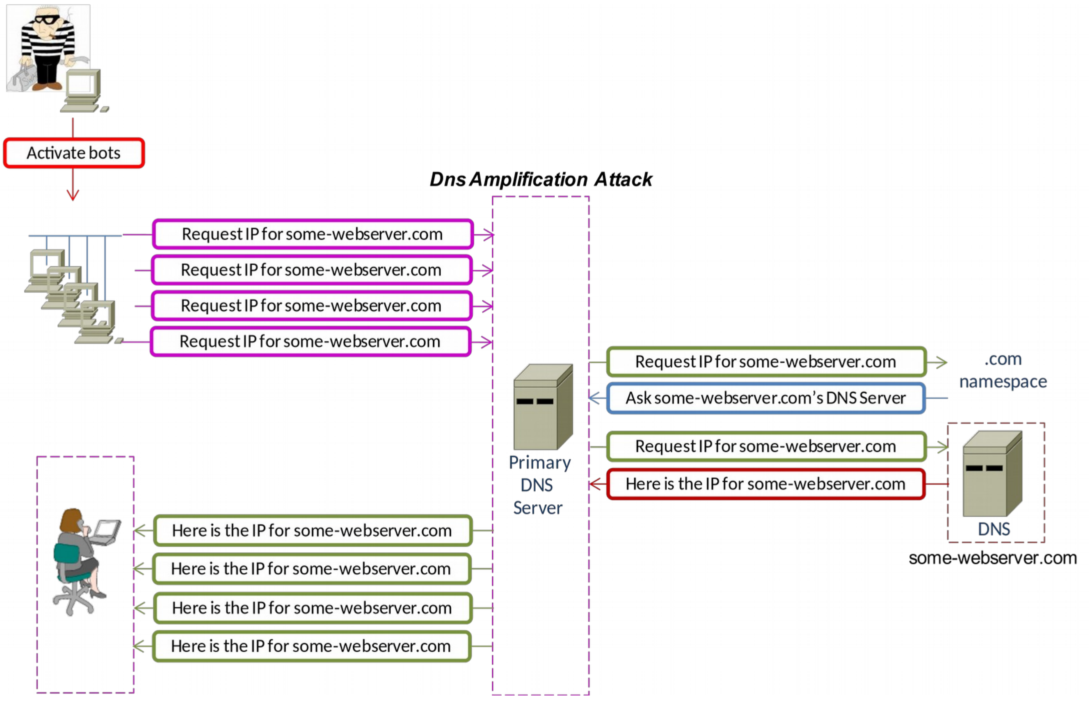
Standard DNS Request
Request IP for some-webserver.com
Unknown , asked from .com namespace
Here is the IP for some-webserver.com
Primary DNS Server
Request IP for some-webserver.com
Ask some-webserver.com's DNS Server
Request IP for some-webserver.com
Here is the IP for some-webserver.com
.com namespace
DNS
some-webserver.com

The User's PC with IP address "My IP Address" makes a DNS query to the Primary DNS Server configured in it's TCP/IP properties, asking to resolve the IP address for *some-webserver.com*.

Step 2 to Step 7 (Recursive Query): User's Primary DNS Server is not authoritative for the domain *some-webserver.com*. So, it asks the Root Servers which then points it to .com Namespace from where it learns about the Primary DNS Server of *some-webserver.com*, which replies with the IP Address of *some-webserver.com*.

Step 8: The IP Address of *some-webserver.com* is cached in the User's Primary DNS Server and it replies to the User's PC with the IP Address for *some-webserver.com*.

DNS Amplification Attack



Activate bots
Dns Amplification Attack
Request IP for some-webserver.com
Request IP for some-webserver.com
Request IP for some-webserver.com
Request IP for some-webserver.com
Primary DNS Server
Request IP for some-webserver.com
Ask some-webserver.com's DNS Server
Request IP for some-webserver.com
Here is the IP for some-webserver.com
.com namespace
DNS
some-webserver.com
Here is the IP for some-webserver.com
Here is the IP for some-webserver.com
Here is the IP for some-webserver.com
Here is the IP for some-webserver.com

Step 1: The attacker sends a signal to the compromised PCs to start DNS queries.

Step 2: All compromised PCs with spoofed IP address "Victim IP Address" make a DNS query to the Primary DNS Servers configured in their TCP/IP properties, asking to resolve the IP address for *some-webserver.com*.

Step 3 to Step 8 (Recursive Query): User's Primary DNS Servers are not authoritative for the domain *some-webserver.com*. So, they ask the Root Servers which then points them to .com Namespace from where they learn about the Primary DNS Server of *some-webserver.com*, which replies with the IP Address of *some-webserver.com*.

Step 9: The IP Address of *some-webserver.com* is cached in the User's Primary DNS Servers and they reply to the Victim's Server (Victim IP Address) with the IP Address for *some-webserver.com*. The reply goes to Victim's Server because the attacker has used this Spoofed Source IP address. The matter is made worse because this reply can be amplified up to factor of 73.

3.2.3 Teardrop Attack

Teardrop is an application which sends Forged IP fragmented Packets that overlap each other and makes it difficult for the receiving host to reassemble them and usually causes a Kernel Panic in the target host.

Teardrop exploits an overlapping IP fragment which causes the TCP/IP fragmentation re-assembly code to improperly handle overlapping IP fragments. Teardrop Attack is not considered to be a severe DoS attack and therefore doesn't cause significant damage to the host systems. In most cases a simple reboot can be best solution but restarting the OS might cause the loss of unsaved data in running applications.

For machines that run Microsoft Windows

When a Teardrop attack is run against a machine, it will crash or reboot (on Windows machines, a user might experience the Blue Screen of Death).

3.2.4 LAND Attack

A LAND attack is a DoS attack that consists of sending a special poison spoofed packet to a computer, causing it to lock up. The security flaw was actually first discovered in 1997 by someone using the alias "m3lt", and has resurfaced many years later in OSs such as Windows Server 2003 and Windows XP SP2.

The attack involves sending a spoofed TCP SYN packet (connection initiation) with the target host's IP address and an open port as both source and destination.

The reason a LAND attack works is because it causes the machine to reply to itself continuously.

Definition: "A LAND attack involves IP packets where the source and destination address are set to address the same device."

Other LAND attacks have since been found in services like SNMP and Windows 88/tcp (kerberos/global services) which were caused by design flaws where the devices accepted requests on the wire appearing to be from themselves and causing replies repeatedly.

3.2.5 DNS cache poisoning

DNS cache poisoning is a maliciously created or unintended situation that provides data to a caching name server that did not originate from authoritative DNS sources. This can happen through improper software design, misconfiguration of name servers, and maliciously designed scenarios exploiting the traditionally open-architecture of the DNS system. Once a DNS server has received such non-authentic data and caches it for future performance increase, it is considered poisoned, supplying the non-authentic data to the clients of the server.

A domain name server translates a domain name (such as www.example.com) into an IP Address that Internet hosts use to contact Internet resources. If a DNS server is poisoned, it may return an incorrect IP Address, diverting traffic to another computer.

3.2.6 Ping of death (POD)

A POD is a type of attack on a computer that involves sending a malformed or otherwise malicious ping to a computer. A ping is normally 56 bytes in size (or 84 bytes when IP header is considered); historically, many computer systems could not handle a ping packet larger than the maximum IP packet size, which is 65,535 bytes. Sending a ping of this size could crash the target computer.

Traditionally, this bug has been relatively easy to exploit. Generally, sending a 65,536 byte ping packet is illegal according to networking protocol, but a packet of such a size can be sent if it is fragmented; when the target computer reassembles the packet, a buffer overflow can occur, which often causes a system crash.

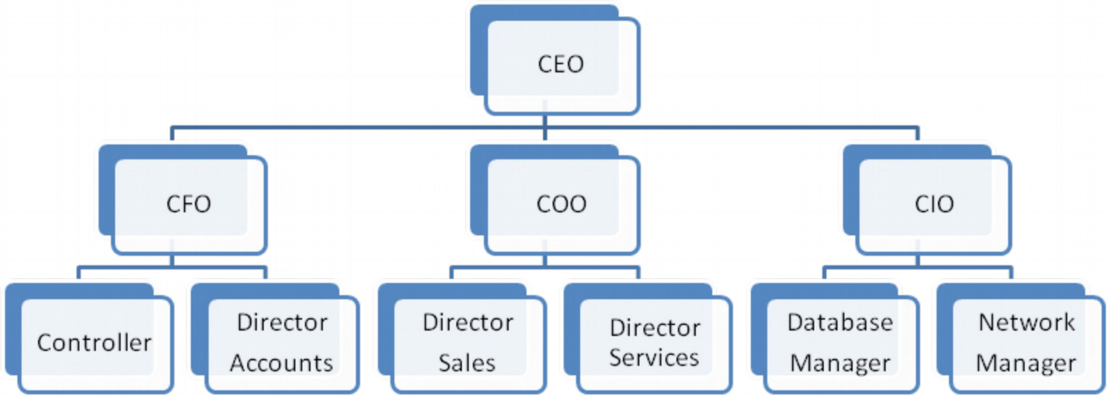
This exploit has affected a wide variety of systems, including UNIX, Linux, Mac, Windows, printers, and routers. However, most systems since 1997-1998 have been fixed, so this bug is mostly historical.

In recent years, a different kind of ping attack has become wide-spread - ping flooding simply floods the victim with so much ping traffic that normal traffic fails to reach the system (a basic DoS attack).

4. Databases and Data Warehousing

A Database Management System (DBMS) is a set of computer programs that controls the creation, maintenance, and the use of the database in a computer platform or of an organisation and its end users. It allows organisations to place control of organisation-wide database development in the hands of DataBase Administrators (DBAs) and other specialists. A DBMS is a system software package that helps the use of integrated collection of data records and files known as databases. It allows different user application programs to easily access the same database. DBMSs may use any of a variety of database models, such as the network model or relational model. In large systems, a DBMS allows users and other software to store and retrieve data in a structured way. Instead of having to write computer programs to extract information, user can ask simple questions in a Structured Query Language (SQL). Thus, many DBMS packages provide Fourth-generation programming language (4GLs) and other application development features. It helps to specify the logical organisation for a database and access and use the information within a database. It provides facilities for controlling data access, enforcing data integrity, managing concurrency controlled, restoring database.

4.1 Hierarchical Database



A hierarchical data model is a data model in which the data is organised into a tree-like structure. The structure allows repeating information using parent/child relationships: each parent can have many children but each child only has one parent. All attributes of a specific record are listed under an entity type.

4.2 Distributed Database

A distributed database is a database that is under the control of a central DBMS in which storage devices are not all attached to a common CPU. Data maybe dispersed across multiple computers located in the same physical location, or may be dispersed over a network of interconnected computers.

Collections of data can be distributed across multiple physical locations. A distributed database can reside on network servers on the Internet, on corporate intranets or extranets, or on other company networks. Replication and distribution of databases improve database performance at end-user worksites.

To ensure that the distributive databases are up to date and current, there are two processes: replication and duplication.

4.2.1 Replication

Replication involves using specialised software that looks for changes in the distributive database. Once the changes have been identified, the replication process makes all the databases look the same. The replication process can be very complex and time consuming depending on the size and number of the distributive databases. This process can also require a lot of time and computer resources.

4.2.2 Duplication

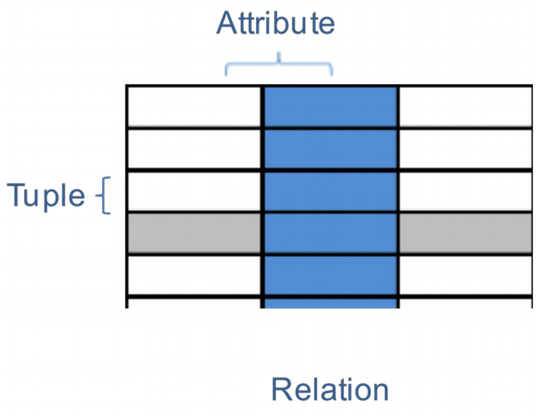
Duplication is not as complicated. It basically identifies one database as a master and then duplicates that database. The duplication process is normally done at a set time after hours. This is to ensure that each distributed location has the same data. During the duplication process, changes to the master database only are allowed. This is to ensure that local data will not be overwritten. Both of the processes can keep the data current in all distributive locations.

4.3 Relational database

A relational database matches data using common characteristics found within the data set. The resulting groups of data are organised and are much easier for people to understand.

Relational databases, as implemented in Relational DBMS (RDBMS), have become a predominant choice for the storage of information in new databases used for financial records, manufacturing and logistical information, personnel data and much more. Relational databases have often replaced legacy hierarchical databases and network databases because they are easier to understand and use, even though they are much less efficient. As computer power has increased, the inefficiencies of relational databases, which made them impractical in earlier times, have been outweighed by their ease of use.

4.3.1 Terminology



A relation is defined as a set of tuples that have the same attributes. A tuple usually represents an object and information about that object. Objects are typically physical objects or concepts. A relation is usually described as a table, which is organised into rows and columns. All the data referenced by an attribute are in the same domain and conform to the same constraints.

The relational model specifies that the tuples of a relation have no specific order and that the tuples, in turn, impose no order on the attributes. Applications access data by specifying queries (SQL see below), which use operations such as *select* to identify tuples, *project* to identify attributes, and *join* to combine relations. Relations can be modified using the *insert*, *delete*, and *update* operators. New tuples can supply explicit values or be derived from a query. Similarly, queries identify tuples for updating or deleting. It is necessary for each tuple of a relation to be uniquely identifiable by some combination of its attribute values. This combination is referred to as the primary key.

4.3.2 Candidate Keys

A set of attributes where in all the relations assigned to that variable there are no two distinct tuples (rows) that have the same values for the attributes in this set.

4.3.3 Primary Keys

Primary Key is a Candidate Key to uniquely identify each tuple in a table. It is selected by the DBA. A Primary Key comprises a single attribute or set of attributes. No two distinct tuples in a table can have the same value in those attributes. Depending on its design, a table may have arbitrarily many unique keys but at most one Primary Key.

4.3.4 Foreign Keys

A foreign key is a reference to a key in another relation, meaning that the referencing tuple has, as one of its attributes, the values of a key in the referenced tuple.

Look at this example:

FN	LN	AGE	CITY	SPORT
Conor	Ryan	14	LK	RUGBY
Cian	Ryan	18	LK	RUGBY
Brian	Tobin	13	CK	SOCCER
Aoife	Doherty	11	DU	SOCCER
Ian	Davies	12	GW	GAA
Sinéad	O'Meara	9	GW	GAA

In the relational database example, you can quickly compare sports and ages because of the arrangement of data in columns. The relational database model takes advantage of this uniformity to build completely new tables out of required information from existing tables. In other words, it uses the relationship of similar data to increase the speed and versatility of the database.

CITY	CITY NAME
LK	Limerick
CK	Cork
DU	Dublin
GW	Galway

Looking at the table above we can see it maps city codes to city names. By storing this information in another table, the database can create a single small table with the locations that can then be used for a variety of purposes by other tables in the database.

4.3.5 Structured Query Language

SQL is a database language designed for managing data in relational database management systems (RDBMS), and originally based upon Relational Algebra. Its scope includes data query and update, schema creation and modification. Using an SQL query it is possible to retrieve the names of all members 12 and over plus what actual city they are from.

```
mysql> SELECT a.FN, a.LN, b.CITY_NAME
            -> FROM  Table1 a INNER JOIN Table2 b
            -> ON a.CITY = b.CITY
       -> WHERE a.AGE > 11;

+--------+--------+-----------+
| LN     | FN     | CITY_NAME |
+--------+--------+-----------+
| Ryan   | Conor  | Limerick  |
| Ryan   | Cian   | Limerick  |
| Tobin  | Brian  | Cork      |
| Davies | Ian    | Galway    |
+--------+--------+-----------+
4 rows in set (0.00 sec)
```

4.4 Database Transactions

Data Integrity is very important in database transactions and all database transactions have four characteristics called the ACID model.

4.4.1 Atomicity

Atomicity refers to the ability of the DBMS to guarantee that either all of the tasks of a transaction are performed or none of them are. For example, the transfer of funds from one account to another can be completed or it can fail for a multitude of reasons, but atomicity guarantees that one account won't be debited if the other is not credited.

Atomicity states that database modifications must follow an “all or nothing” rule. Each transaction is said to be “atomic” if when one part of the transaction fails, the entire transaction fails. It is critical that the database management system maintain the atomic nature of transactions in spite of any DBMS, OS or hardware failure.

4.4.2 Consistency

The consistency property ensures that the DBMS remains in a consistent state before the start of the transaction and after the transaction is over (whether successful or not).

Consistency states that only valid data will be written to the database. If, for some reason, a transaction is executed that violates the database’s consistency rules, the entire transaction will be rolled back and the database will be restored to a state consistent with those rules. On the other hand, if a transaction successfully executes, it will take the database from one state that is consistent with the rules to another state that is also consistent with the rules.

4.4.3 Isolation

Isolation refers to the requirement that other operations cannot access or see the data in an intermediate state during a transaction. This constraint is required to maintain the performance as well as the consistency between transactions in a DBMS. Thus, each transaction is unaware of other transactions executing concurrently in the system.

4.4.4 Durability

Durability refers to the guarantee that once the user has been notified of success, the transaction will persist, and not be undone. This means it will survive system failure, and that the database system has checked the integrity constraints and won't need to abort the transaction. Many databases implement durability by writing all transactions into a transaction log that can be played back to recreate the system state right before a failure. A transaction can only be deemed committed after it is safely in the log.

Durability does not imply a permanent state of the database. Another transaction may overwrite any changes made by the current transaction without hindering durability.

4.5 Multilevel Database security

There are three characteristics Multilevel Database security:

- The security of a single element is different from the security of other elements of the same type.
- Several grades of sensitivity are needed (not only sensitive/non-sensitive).
- Sensitivity of an aggregate is different from the sensitivity of the sum of elements.

4.5.1 Polyinstantiation

• A “low” level user could accidentally try to update a missing field (containing a “high” value). How must the DBMS react:

- Refuse to update (reveals that there is sensitive info)
- Overwrite the data (compromises integrity)
- Keep both values, i.e. polyinstantiation.

4.5.2 Database Views

A Database View is a subset of the database sorted and displayed in a particular way. For example, in a tools database, perhaps you only wish to display the spanners stored in the database. To do that you would create a Spanners view. The equipment database templates has a view for each equipment type, sorted by the name of the equipment.

For each view, you can control which columns are displayed, what order they are displayed in, how wide each column is, how the data is sorted, and what types of records to display.

4.5.3 Concurrency

Concurrency control in DBMS ensures that database transactions are performed concurrently without the concurrency violating the data integrity of a database. The DBMS uses a “LOCK” to allow an authorised user make a change while preventing any other write access to that piece of data until the change has been made.

4.6 Open Database Connectivity (ODBC)

ODBC provides a standard software API method for using DBMS. The designers of ODBC aimed to make it independent of programming languages, database systems, and OSs.

ODBC uses as its basis the various Call Level Interface (CLI) specifications from the SQL Access Group, X/Open (now part of The Open Group), and the ISO/IEC.

4.7 Java Database Connectivity (JDBC)

JDBC is an API for the Java programming language that defines how a client may access a database. It provides methods for querying and updating data in a database. JDBC is oriented towards relational databases.

4.8 Data Mining

Data mining is the process of extracting patterns from data. As more data are gathered, with the amount of data doubling every three years, data mining is becoming an increasingly important tool to transform these data into information. It is commonly used in a wide range of profiling practices, such as marketing, surveillance, fraud detection and scientific discovery.

5. Data Storage

5.1 Types of Storage

5.1.1 Primary Memory

This is the Random Access Memory (RAM) on a computer system.

5.1.2 Secondary Storage

This describes Hard-drives, USB Sticks, DVD/CD Disks, Tape Drives, Optical Drives etc...

5.1.3 Virtual Memory

A system can simulate primary memory on a secondary storage device. The system reserves an area of say a hard-disk and uses it as if it were additional RAM.

5.1.4 Virtual Storage

A simulation of Secondary Storage in Primary Memory. This is generally called a RAM Disk.

5.1.5 Random Access Storage

RAM and Hard-drives are considered Random Access Storage. Such devices are defined as such because any item of stored data can be accessed in the same timeframe as any other item of stored data.

5.1.6 Sequential Access Storage

CDs and Tapes fall into this category. It is any device where the disk or tape must be scanned from beginning to end to find items of data.

5.1.7 Volatile Storage

Storage where data is lost on power being removed. i.e. RAM

5.1.8 Non-volatile Storage

The removal of power from such a device does not result in the loss of data.

6. Knowledge Based Systems

6.1 Expert System

An expert system is software that attempts to provide an answer to a problem, or clarify uncertainties where normally one or more human experts would need to be consulted.

Expert systems are most common in a specific problem domain, and is a traditional application and/or subfield of Artificial Intelligence (AI). A wide variety of methods can be used to simulate the performance of the expert however common to most or all are:

- The creation of a so-called '*knowledgebase*' which uses some knowledge representation formalism to capture the Subject Matter Expert's (SMEx) knowledge
- A process of gathering that knowledge from the SMEx and codifying it according to the formalism, which is called knowledge engineering. Expert systems may or may not have learning components
- Once the system is developed it is proven by being placed in the same real world problem solving situation as the human SMEx, typically as an aid to human workers or a supplement to some information system.

6.2 Neural Networks

Neuron's are programming constructs that mimic the properties of biological neurons.

Neural networks are made up of interconnecting artificial neurons. Artificial neural networks may either be used to gain an understanding of biological neural networks, or for solving AI problems without necessarily creating a model of a real biological system. The real, biological nervous system is highly complex and includes some features that may seem superfluous based on an understanding of artificial networks.

AI and cognitive modelling try to simulate some properties of neural networks. While similar in their techniques, the former has the aim of solving particular tasks, while the latter aims to build mathematical models of biological neural systems.

In the AI field, artificial neural networks have been applied successfully to speech recognition, image analysis and adaptive control, in order to construct software agents (in computer and video games) or autonomous robots. Most of the currently employed artificial neural networks for AI are based on statistical estimation, optimisation and control theory.

6.3 Decision Support System (DSS)

A DSS is a class of information systems that support business and organisational decision-making activities. A properly designed DSS is an interactive software-based system intended to help decision makers compile useful information from a combination of raw data, documents, personal knowledge, or business models to identify and solve problems and make decisions.

Typical information that a decision support application might gather and present are:

- An inventory of all of your current information assets (including legacy and relational data sources, cubes, data warehouses, and data marts),
- Comparative sales figures between one week and the next,
- Projected revenue figures based on new product sales assumptions.

7. Application Attacks

7.1 Buffer Overflows

A buffer overflow, or buffer overrun, is an anomaly where a process stores data in a buffer outside the memory the programmer set aside for it. The extra data overwrites adjacent memory, which may contain other data, including program variables and program flow control data. This may result in erratic program behaviour, including memory access errors, incorrect results, program termination (a crash), or a breach of system security.

Buffer overflows can be triggered by inputs that are designed to execute code, or alter the way the program operates. They are thus the basis of many software vulnerabilities and can be maliciously exploited. Bounds checking can prevent buffer overflows.

Programming languages commonly associated with buffer overflows include C and C++, which provide no built-in protection against accessing or overwriting data in any part of memory and do not automatically check that data written to an array (the built-in buffer type) is within the boundaries of that array.

7.2 Time of check to time of use (TOCTTOU)

TOCTTOU (pronounced "TOCK too") is a software bug caused by changes in a system between the checking of a condition (such as a security credential) and the use of the results of that check. It is a kind of race condition.

A simple example is as follows: Consider a Web application that allows a user to edit pages, but allows administrators to lock pages to prevent editing. A user requests to edit a page, getting a form by which they can alter its content. Before the user submits the form, an administrator locks the page, which should prevent editing. However, since the user has already begun editing, when they submit the form, their edits are accepted. When the user began editing, their authorisation was checked, and they were indeed allowed to edit. However, the authorisation was used later, after they should no longer have been allowed.

7.3 Trap Doors

These are code sequences that permit access for developers during the write stage. If these are not removed before code release they can offer a means of access for an attacker.

7.4 Rootkits

A rootkit is a software system that consists of one or more programs designed to obscure the fact that a system has been compromised. Contrary to what its name may imply, a rootkit does not grant a user administrator privileges, as it requires prior access to execute and tamper with system files and processes. An attacker may use a rootkit to replace vital system executables, which may then be used to hide processes and files the attacker has installed, along with the presence of the rootkit. Access to the hardware, e.g., the reset switch, is rarely required, as a rootkit is intended to seize control of the OS. Typically, rootkits act to obscure their presence on the system through subversion or evasion of standard OS security scan and surveillance mechanisms such as anti-virus or anti-spyware scan. Often, they are Trojans as well, thus fooling users into believing they are safe to run on their systems. Techniques used to accomplish this can include concealing running processes from monitoring programs, or hiding files or system data from the OS. Rootkits may also install a "back door" in a system by replacing the login mechanism (such as /bin/login) with an executable that accepts a secret login combination, which, in turn, allows an attacker to access the system, regardless of the changes to the actual accounts on the system.

Rootkits may have originated as regular applications, intended to take control of a failing or unresponsive system, but in recent years have been largely malware to help intruders gain access to systems while avoiding detection. Rootkits exist for a variety of OSs, such as GNU/Linux, UNIX, Mac OS, Solaris and Microsoft Windows as well as mobile OS like Android. Rootkits often modify parts of the OS or install themselves as drivers or kernel modules, depending on the internal details of an OS's mechanisms.

8. Web Application Security

8.1 Cross Site Scripting (XSS)

XSS is a type of computer security vulnerability typically found in web applications which enable malicious attackers to inject client-side script into web pages viewed by other users. An exploited XSS vulnerability can be used by attackers to bypass access controls such as the same origin policy. XSS carried out on websites were roughly 80% of all documented security vulnerabilities as of 2007. Their impact may range from a petty nuisance to a significant security risk, depending on the sensitivity of the data handled by the vulnerable site, and the nature of any security mitigations implemented by site owner.

8.2 Cross Site Request Forgery (CSRF)

Unauthorised commands are transmitted from a user that the website trusts.

Unlike XSS, which exploits the trust a user has for a particular site, CSRF exploits the trust that a site has in a user's browser.

8.3 SQL Injection

SQL injection is a code injection technique that exploits a security vulnerability occurring in the database layer of an application. The vulnerability is present when user input is either incorrectly filtered for string literal escape characters embedded in SQL statements or user input is not strongly typed and thereby unexpectedly executed. It is an instance of a more general class of vulnerabilities that can occur whenever one programming or scripting language is embedded inside another. SQL injection attacks are also known as SQL insertion attacks.

Example;

User is asked to enter their account number: **123456789**

The account number is entered in an SQL statement by the system.

```
SELECT * FROM transaction WHERE account = ‘<A/C Number>’;
```

This would result in:

```
SQL> SELECT * FROM transaction WHERE account = ‘123456789’;
A/C 123456789
Date        Sort Code   Account     Value       Balance
12/11/2009        99-45-22    98234567    € 2,500           € 8,340
12/11/2009        99-45-22    99876543    € 1,000           € 7,340
```

If the user had entered the following instead of the A/C number.

123456789; DELETE * FROM transaction WHERE account = ‘123456789’

i.e. It would be entered in the SQL statement like this.

```
SQL> SELECT * FROM transaction WHERE account = ‘123456789; DELETE *
FROM transaction WHERE account = ‘123456789’’;
```

Now the following could be the result as the SQL queries executed:

```
SQL> SELECT * FROM transaction WHERE account = ‘123456789’;
A/C 123456789
Date        Sort Code   Account     Value       Balance
12/11/2009        99-45-22    98234567    € 2,500          € 8,340
12/11/2009        99-45-22    99876543    € 1,000          € 7,340

SQL> DELETE * FROM transaction WHERE account = ‘123456789’;
A/C 123456789
All transactions deleted !!
```

It is therefore very important that input validation is performed in the code. The second input should have been rejected in the first place.

9. Reconnaissance Attacks

9.1 IP Probes & Port Scans

IP Probes are the initial sweep of a network carried out on a target network. These probes can ping a target network looking for a response. This is usually followed by a Port Scan of ‘interesting’ networks.

An example is Nmap. Nmap is a security scanner originally written by Gordon Lyon. Nmap is a "Network Mapper", used to discover computers and services on a computer network, thus creating a "map" of the network. Just like many simple port scanners, Nmap is capable of discovering passive services on a network despite the fact that such services aren't advertising themselves with a service discovery protocol. In addition Nmap may be able to determine various details about the remote computers. These include OS, device type, uptime, software product used to run a service, exact version number of that product, presence of some firewall techniques and, on a local area network, even vendor of the remote network card.

9.2 Vulnerability Scan

A vulnerability scanner is a program designed to search for and map systems for weaknesses in an application, computer or network. It is usually deployed on systems that a port scanner have identified as vulnerable. It attempts to determine the target OS, version and patch level and working of a database of known vulnerabilities it can suggest devices for attack and the type of attack.

9.3 Dumpster Diving

Dumpster diving is the practice of sifting through commercial or residential trash to find items that have been discarded by their owners, but which may be useful to the dumpster diver. It is amazing the information that can be found that would aid with a computer hack. Organisations should ensure that all waste is disposed of in a secure manner.

10. Masquerade Attacks

10.1 IP Spoofing

IP Spoofing is the creation of IP packets with a forged source IP address, with the purpose of concealing the identity of the sender or impersonating another computing system. IP spoofing is most frequently used in DoS attacks. In such attacks, the goal is to flood the victim with overwhelming amounts of traffic, and the attacker does not care about receiving responses to the attack packets. Packets with spoofed addresses are thus suitable for such attacks. Ingress filtering is necessary on the gateway to a network to block packets from outside the network with a source address inside the network. This prevents an outside attacker spoofing the address of an internal machine. The gateway should also perform egress filtering on outgoing packets, which is blocking of packets from inside the network with a source address that is not inside. This prevents an attacker within the network performing filtering from launching IP spoofing attacks against external machines. It is also recommended to design network protocols and services so that they do not rely on the IP source address for authentication.

10.2 Session Hijacking

Session hijacking refers to the exploitation of a valid computer session to gain unauthorised access to information or services in a computer system. In particular, it is used to refer to the theft of a magic cookie used to authenticate a user to a remote server. It has particular relevance to web developers, as the HTTP cookies used to maintain a session on many web sites can be easily stolen by an attacker using an intermediary computer or with access to the saved cookies on the victim's computer, called HTTP cookie theft.

TCP session hijacking is when a hacker takes over a TCP session between two machines. Since most authentication only occurs at the start of a TCP session, this allows the hacker to gain access to a machine.

A popular method is using source-routed IP packets. This allows a hacker at point A on the network to participate in a conversation between B and C by encouraging the IP packets to pass through its machine.

If source-routing is turned off, the hacker can use "blind" hijacking, whereby it guesses the responses of the two machines. Thus, the hacker can send a command, but can never see the response. However, a common command would be to set a password allowing access from somewhere else on the net.

A hacker can also be "inline" between B and C using a sniffing program to watch the conversation. This is known as a "man-in-the-middle attack".

A common component of such an attack is to execute a DoS attack against one end-point to stop it from responding. This attack can be either against the machine to force it to crash, or against the network connection to force heavy packet loss.

11. Decoy Techniques

11.1 Honey Pots

A honeypot is a trap set to detect, deflect, or in some manner counteract attempts at unauthorised use of information systems. Generally it consists of a computer, data, or a network site that appears to be part of a network, but is actually isolated, (un)protected, and monitored, and which seems to contain information or a resource of value to attackers.

These are valuable as a surveillance and early-warning tool. While it is often a computer, a honeypot can take other forms, such as files or data records, or even unused IP address space. A honeypot that masquerades as an open proxy to monitor and record those using the system is a sugarcane. Honeypots should have no production value, and hence should not see any legitimate traffic or activity. Whatever they capture can then be surmised as malicious or unauthorised. One practical implication of this is honeypots that thwart spam by masquerading as the type of systems abused by spammers. They categorise trapped material 100% accurately, it is all illicit.

11.2 Pseudo-flaw

An apparent loophole or trapdoor that has been inserted into an OS in order to trap unauthorised intruders who access a network.

This page is intentionally blank