# Computer System Security (KNC-401)

UNIT – I

**What is Security?**

Achieving some goal in the presence of an adversary.

Many systems are connected to the Internet, which has adversaries. Thus, design of many systems might need to address security, i.e. will the system work when there's an adversary?

**Aims of Security (Security Services)**

Generally the following are considered to be the aims of computer and information security:

- Confidentiality- Certain information must be kept secret from unauthorized access.

  Importance of confidentiality
  - ✓ Loss of revenue
  - ✓ Loss of reputation
  - ✓ Loss of clients/customer
  - ✓ Embarrassment

  Ensuring confidentiality
  - ✓ Encryption
  - ✓ Access Control
- Integrity- ensures that information and systems have not been altered in an unauthorized way.

  Ensuring integrity
  - ✓ Regular backups
- Availability- information or systems are accessible and modifiable in a timely fashion by those authorized to do so.
- Authenticity- Verification of claim
- Non-repudiation/Accountability- means to ensure that a transferred message has been sent and received by the parties claiming to have sent and received the message. Nonrepudiation is a way to guarantee that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message.

  Nonrepudiation can be obtained through the use of:
  - ✓ digital signatures-- function as a unique identifier for an individual, much like a written signature. A digital code that can be attached to an electronically transmitted message that uniquely identifies the sender.

Computer security, also known as cyber security or IT security, is the protection of information systems from theft or damage to the hardware, the software, and to the information on them, as well as from disruption or misdirection of the services they provide.

"**Computer security** can be defined as controls that are put in place to provide confidentiality, integrity, and availability for all components of computer systems."

**Information security (IS)** is designed to protect the confidentiality, integrity and availability of computer system data from those with malicious intentions.

**Cyber security** refers to the body of technologies, processes, and practices designed to protect networks, devices, programs, and data from attack, damage, or unauthorized access.

**Attack** is any attempt to expose, alter, disable, destroy, steal or gain unauthorized access to or make unauthorized use of an Asset.

**What is a Threat?**

A threat to a computing system is a set of circumstances that has the potential to cause loss or harm.

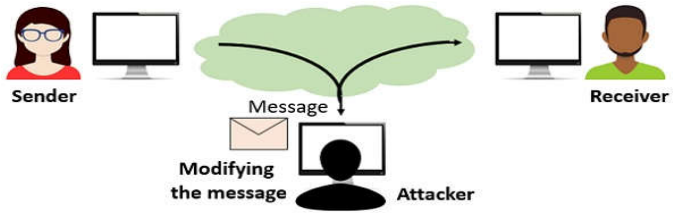
A threat can be either "intentional" (i.e. hacking: an individual cracker or a criminal organization) or "accidental" (e.g. the possibility of a computer malfunctioning, or the possibility of a natural disaster such as an earthquake, a fire, or a tornado) or otherwise a circumstance, capability, action.

**Types of Attack:**

There are two types of Attacks-

- Active Attacks
- Passive Attacks

**Active attacks** are the attacks in which the attacker tries to modify the information or creates a false message. Or An active attack attempts to alter system resources or effect their operations.




Active Attack

The active attacks are the different types

- **Masquerade-** A masquerade (is an attack that uses a fake identity) attack in which unauthorized attacker tries to pose as another entity.
- **Modification -** If an unauthorized party accesses and tampers with an asset, the threat is a **modification**. Modification can be done using two ways replay attack and alteration. In the replay attack, a sequence of events or some data units is captured and resent by them. While alteration of the message involves some change to the original message, either one of them can cause alteration.
- **Fabrication** causes Denial Of Service (DOS) attacks in which attacker prevents normal use of communication facilities or other services. This attack may have a specific target. For example, an entity may suppress all messages directed to a particular destination.

**(Fabrication:** An unauthorized party inserts counterfeit objects into the system and basically attacks the authenticity of the system.

**Modification:** An unauthorized party modifies the assets of the system and basically attacks the integrity of the system.**)**

**Passive Attack** - A passive attack, in computing security, is an attack characterized by the attacker monitoring communication or systems. This can take forms such as reading emails, tracking internet use, or using a system's microphone and camera to "spy" on an individual. In a passive attack, the intruder/hacker does not attempt to alter the system or change data. Even though a passive attack sounds less harmful, the damage in the end can be just as severe if the right type of information is obtained.

It is possible for passive attacks to be performed for non-malicious reasons, such as marketing research.

Types of Passive attacks are as following:

1. The release of message content can be expressed with an example, in which the sender wants to send a confidential message to the receiver. The sender doesn't want the contents of that message to be read by some interceptor.
2. By using encryption a message could be masked in order to prevent the extraction of the information from the message, even if the message is captured. Though still attacker can analyse the traffic and observe the pattern to retrieve the information. This type of passive attack refers to as **traffic analysis.**

**Key Differences between Active and Passive Attacks**

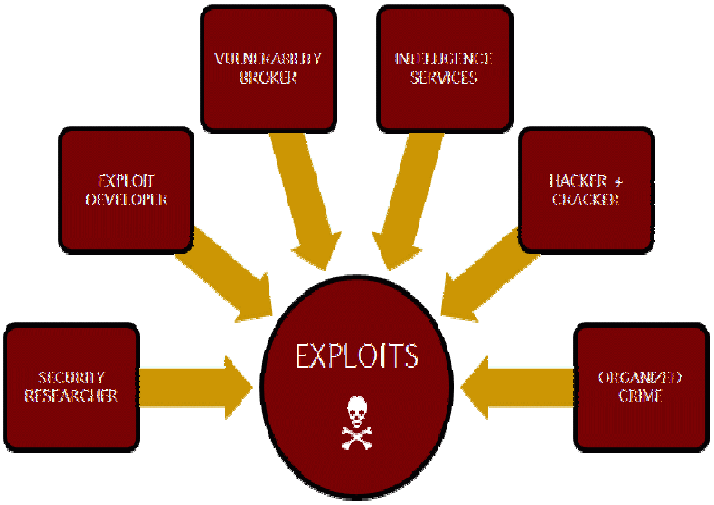
1. The active attack includes modification of the message. On the other hand, in passive attacks, the attacker doesn't commit any changes to the intercepted information.
2. The active attack causes a huge amount of harm to the system while the passive attack doesn't cause any harm to the system resources.

3. A passive attack is considered as a threat to data confidentiality. In contrast, an active attack is a threat to the integrity and availability of the data.
4. The attacked entity is aware of the attack in case of active attack. As against, the victim is unaware of the attack in the passive attack.
5. The active attack is accomplished by gaining the physical control over the communication link to capture and insert transmission. On the contrary, in a passive attack, the attacker just needs to observe the transmission.

**The Marketplace for vulnerabilities:-**

Vulnerability is a weakness in the security system.

**A vulnerability broker** is an organization or person who provides a link between a vulnerability discoverer and the highest bidder.




**Error 404**

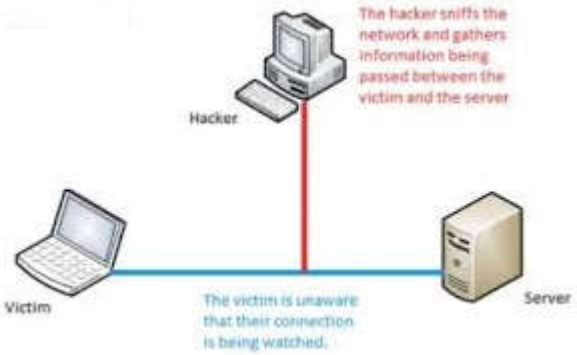
The error 404 indicates that the server where the page should reside has been contacted but that the page is does not exist at that address, at that time. It happens when pages or files are moved or deleted or when URLS are mistyped.


**Error 404 hacking** (as evidenced by the 404 Page Not Found error) are almost always blind attacks on standard web applications located in their default directories. A huge proportion of these attempts can be diverted simply by renaming or relocating your web applications.

Computer hacking refers to the practice of modifying or altering computer software and hardware to accomplish a goal that is considered to be outside of the creator's original objective.

**Hijack-** hijack refers to taking control over system and causing it to do something else.

The most common type of hijacking is when malware infects your computer and redirects your web browser, homepage, or search engine to a malicious site or somewhere you don't want to be.





**Control Hijacking-** Attacks that divert a program's control flow for malicious purposes are generally known as control-hijacking attacks.

(A **control**-**hijacking attack** overwrites some data structures in a victim program that affect its **control** flow, and eventually hijacks the **control** of the program and possibly the underlying system.)

What is needed?

- Understanding C functions and the stack
- Some familiarity with machine code
- Know how systems calls are made
- The exec() system call
- Attacker needs to know which CPU and OS are running on the target machine:

**Inter overflow attacks-**

An integer overflow occurs when you attempt to store inside an integer variable a value that is larger than the maximum value the variable can hold. The C standard defines this situation as undefined behavior.

The ISO C99 standard says that an integer overflow causes "undefined behaviour", meaning that compilers conforming to the standard may do anything they like from completely ignoring the overflow to aborting the program. Most compilers seem to ignore the overflow, resulting in an unexpected or erroneous result being stored. However, overflow vulnerabilities are especially dangerous because they may lead to program instability even under normal operation and in many cases, are exploitable as well.

**Example -**

print the value ***(****The range of int type is from -$2^{15}$ to +$2^{15}$-1 (-32768 to +32767))*

```
#include<stdio.h>
int main()
{
int x=32770;
printf("%d",x);
return 0;
}
```

Output:

-32766

*(While printing, the printf() treats it as a negative value by looking 1 at the first bit, so printf() reverts the 2's and 1's compliment of binary and prints equal decimal value with -ve sign)*

A typical control hijacking attack starts by corrupting a pointer to an attacker supplied malicious data, which we refer to as the payload. Then, the attack proceeds by modifying code pointers, which are objects that affect control flow of a program. In order to hijack control successfully, the attacker needs to know the correct target value.

**Format string vulnerabilities-**

1. A format string is an ASCII (American Standard Code for Information Interchange) string that contains text and format parameters.

   Example: printf("my name is:%s\n","Ram");

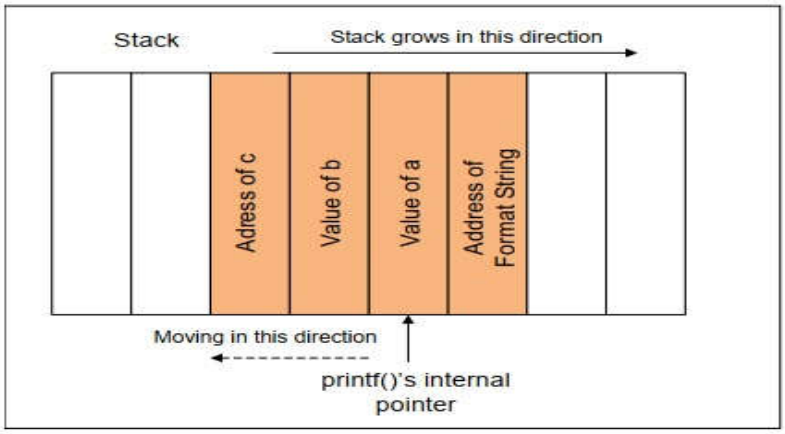
If a program containing the above example is run it will output: *My name is: Ram*

In addition to %d, there are several other format parameters, each having different meaning.

%d, %s, %n, %f, %o etc.

**The stack and its role at format strings-** The behavior of the format function is controlled by the format string. The function retrieves the parameters requested by the format string from the stack.

printf("a has value %d, b has value %d, c is at address: %08x\n",a, b, &c);




What if there is a miss-match between the format string and the actual arguments?

printf("a has value %d, b has value %d, c is at address: %08x\n",a, b);

- In the above example, the format string asks for 3 arguments, but the program actually provides only two (i.e. *a* and *b*). In this case there is no way for the compiler to find the miss-match in this case.
- The function printf() fetches the three arguments from the stack. Unless the stack is marked with a boundary, printf() does not know that it runs out of the arguments that are provided to it.
- Since there is no such a marking. printf() will continue fetching data from the stack. In a miss-match case, it will fetch some data that do not belong to this function call.

2. Attacks on Format strings vulnerability

A format string attack begins when an intruder takes aim at the Format Function. Format Function is designed to handle routine tasks. However, if these functions aren't adequately protected, the application can be at risk for a format string attack. The attacker may explore and test the software by inserting specific formatting characters via a form at a website or another input tool. When an application receives certain conversion characters, such as "%f", "%p", or "%n", the attacker is allowed to execute arbitrary code on a server, read values from an attack, or software crashes.

Once an attacker finds a successful intrusion point, he can read data from the stack, read character strings from the process memory, write an integer to locations in the process memory, change security controls, crash applications, or launch a denial of service (DoS) attack crashing the program.

Example

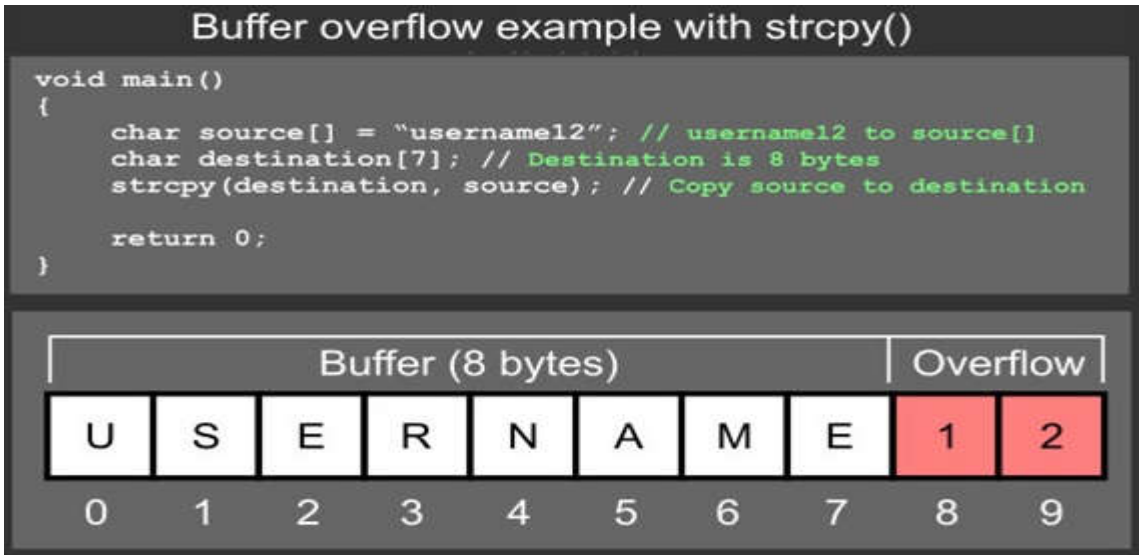
```
printf ("%s%s%s%s%s%s%s%s%s%s%s%s");
```

For each %s, printf() will fetch a address from the stack, and print out the memory contents pointed by this address as a string and in case the memory pointed by this number might not exist, the program will crash.
Using this attack, attackers can do the following:

* Overwrite important program flags that control access privileges
* Overwrite return addresses on the stack, function pointers, etc.

**Buffer Overflow Vulnerabilities-**

A buffer overflow is a common software coding mistake that an attacker could exploit to gain access to your system.
A buffer overflow occurs when more data is put into a fixed-length buffer than the buffer can handle. The extra information, which has to go somewhere, can overflow into adjacent memory space, corrupting or overwriting the data held in that space.




This overflow usually results in a system crash, but it also creates the opportunity for an attacker to run arbitrary code or manipulate the coding errors to prompt malicious actions.

Cybercriminals exploit buffer overflow problems to alter the execution path of the application by overwriting parts of its memory. The malicious extra data may contain code designed to trigger specific actions — in effect sending new instructions to the attacked application that could result in unauthorized access to the system. Hacker techniques that exploit a buffer overflow vulnerability vary per architecture and operating system.

Buffer Overflow Attack Example - In the code below, the correct password grants the user root privileges. If the password is incorrect, the program will not grant the user privileges.

```
int main(void)
{
  char buff[15];
  int pass = 1;
  printf("\n Enter the password : \n");
  gets(buff);
  if(strcmp(buff, "thegeekstuff"))
  {
    printf ("\n Wrong Password \n");
  }
  else
  {
    printf ("\n Correct Password \n");
    pass = 1;
  }
```

```
  if(pass)
  {
    /* Now Give root or admin rights to user*/
     printf ("\n Root privileges given to the user \n");
  }
  return 0;
}
```

However, there is a possibility of buffer overflow in this program because the gets() function does not check the array bounds. Here is an example of what an attacker could do with this coding error:

*Enter the password:*

*hhhhhhhhhhhhhhhhhhhh*

*Wrong Password*

*Root privileges given to the user*

In the above example, the program gives the user root privileges, even though the user entered an incorrect password. The gets() function does not check the array bounds and can even write string of length greater than the size of the buffer to which the string is written. Now, can you even imagine what can an attacker do with this kind of a loophole?

**Defenses against Control Flow Hijacking-**

A variety of defensive mechanisms have been proposed to mitigate (reduce) control-flow hijacking attacks. Some of them are as follows:

- **Platform Defenses -** In this case prevent attack code execution

Remote code execution (RCE) refers to the ability of a cyber attacker to access and make changes to a computer owned by another, without authority and regardless of where the computer is geographically located. Attacker to take over a computer or a server by running arbitrary malicious software (malware) which is most dangerous.

**Prevention**

- ✓ Timely installation of software update ranks as the top cyber security measure.
- ✓ To prevent attackers which trying to infect vulnerable servers with crypto currency mining malware, the initial attack must be blocked.
- ✓ **Data Execution Prevention** – It is security features that can help prevent damage to your computer from viruses and other security threats. Harmful programs can try to attack Windows by attempting to run code from your computer's memory reserved for Windows and other authorized programs. These types of attacks can harm your programs and files.

- **Run time Defenses -** Add runtime code to detect overflows exploits

Runtime Malware Defense (RMD) is endpoint protection software that acts as a last line of defense against damage from malicious attacks. Unlike antivirus and next-generation antivirus, which attempt to identify and stop malware pre execution, RMD blocks malware at the time of execution by recognizing malicious activity. Importantly, runtime malware defense is preventative; it blocks malware before damage is done, preventing data loss, data theft, and downtime.

**Types of Cyber Attacks-**

It is the illegal attempt to harm someone's computer system or the information on it, using the internet.

Cyber-attacks can be classified into the following categories:

- System - based attacks
- Web - based attacks

**System – Based Attacks-** These are the attacks which are intended to compromise a computer or a computer network. Some of the important system-based attacks are as follows-

1. **Virus-** A computer virus is a program or piece of code that can link itself to the executable files of a computer. It corrupt and erase a file or program.
2. **Worm –** It is a type of malware (**Malware**" is short for "**malicious software**") whose primary function is to replicate itself to spread to uninfected computers .It consumes system resources and slow down it, and halt the system completely. Worms often originate from email attachments that appear to be from trusted senders.
3. **Trojan horse- It** is the malware which misleads users of its true intent. It appears to be a normal application but when opened/executed some malicious code will run in the background.

**Web – Based Attacks-** These are the attacks which occur on a website or web applications. Some of the important web-based attacks are as follows-

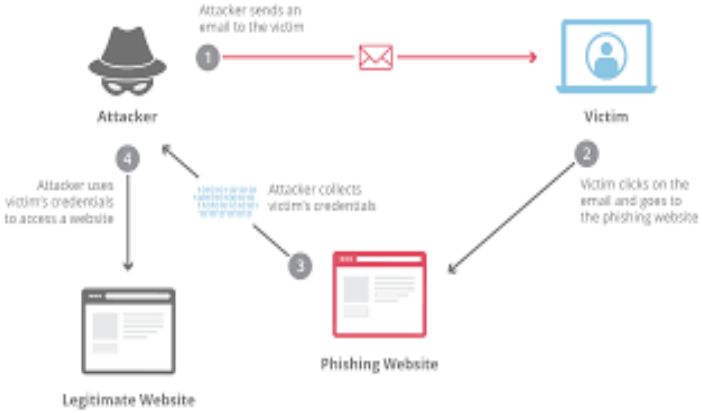
1. **Injection attacks-** It is the attack in which some data will be injected into a web application to manipulate the application and fetch the required information.

   **Example-** SQL Injection, code Injection; log Injection, XML Injection etc.
2. **Spoofing-** A technique used to gain unauthorized access to computers, whereby the intruder sends messages to a computer with an IP address indicating that the message is coming from a trusted host.

   (In simple words,

   I. Spoofing means to pretend (deceive) to be someone else.

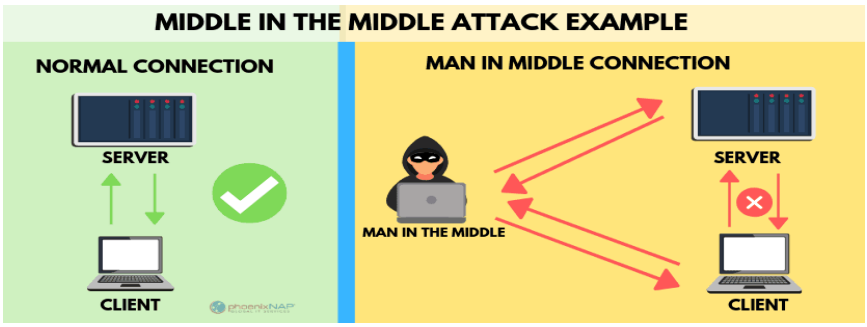
   II. Sniffing means to illegally listen into another's conversation )
3. **Phishing-** Phishing is the fraudulent use of electronic communications that appear to come from a reputable source (to deceive) and take advantage of users. Phishing attacks attempt to gain sensitive, confidential information such as usernames, passwords, credit card information.



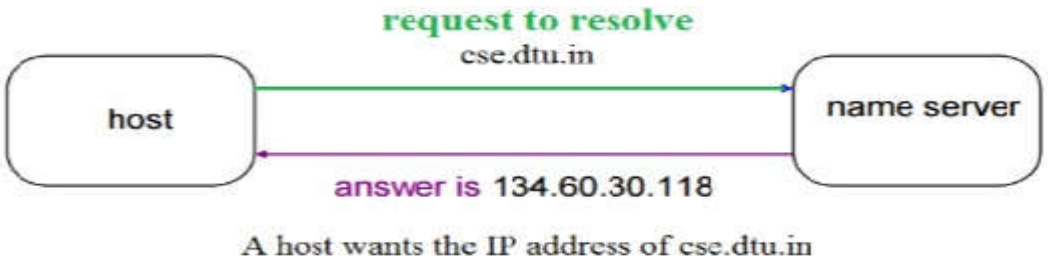

4. **Denial of Service -** A denial-of-service attack is a event that occurs when an attacker prevents legitimate (legal) users from accessing specific computer systems, devices, services or other IT resources.
5. **Man in the middle attacks-** A MitM attack occurs when a hacker inserts itself between the communications of a client and a server. Some common types of man-in-the-middle attacks are :

   I. Session hijacking

   II. DNS hijacking

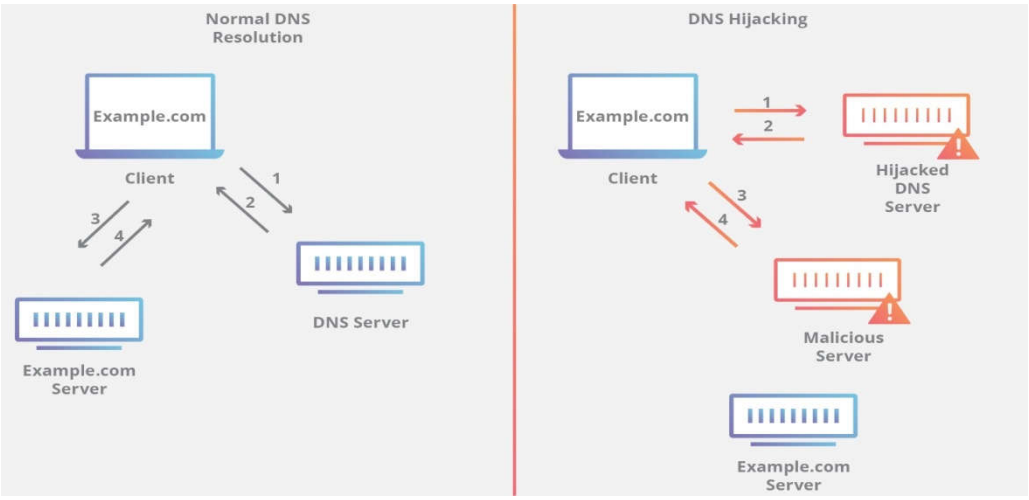


❖ **DNS hijack-** Domain name system, or DNS, is the protocol that translates human-friendly URLs, such as cse.dtu.in, into machine-friendly IP addresses, such as 134.60.30.118**.**




A host wants the IP address of cse.dtu.in

DNS hijacking is when a cybercriminal hijacks a user's DNS traffic. Generally, a compromised DNS server will be used to return fake IP addresses when a user's device asks for a specific website's address.




Malware attacks can infect your router, and change its DNS settings so that it uses hacker-owned DNS servers instead of legit ones. DNS hijacking can be used in phishing attacks with the intent to steal personal and financial information from online users.

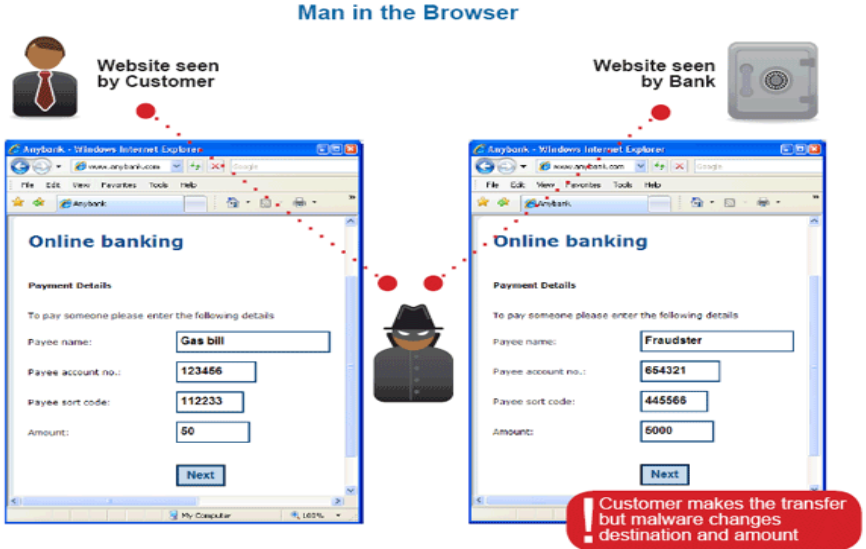
Types of DNS attacks include:

**Zero day attack –** A zero day attack (also referred to as Day Zero) is an attack that exploits a potentially serious software security weakness that the vendor or developer may be unaware of.

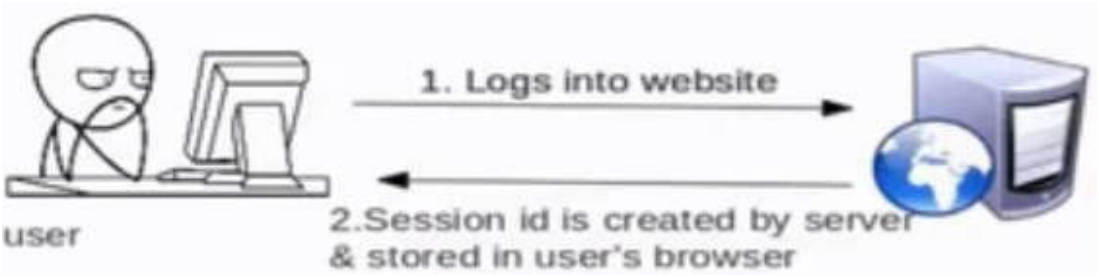
**Cache poisoning –** the attacker corrupts a DSN server by replacing a legitimate IP address in the server's cache with that of another, in order to redirect traffic to a malicious website, collect information or initiate another attack. Cache poisoning may also be referred to as DNS poisoning.

❖ **MAN-IN-THE-BROWSER ATTACK (MITB) -** Man-in-the-browser is a type of man-in-the-middle. It specifically involving a browser infected with some type of proxy malware. This malware allows an attacker to intercept or modify information sent from a user's browser to a server. Such attacks are often carried out in an attempt to steal financial information by intercepting a user's traffic to a banking site.

For example, if the customer is trying to make a transfer to an account, the malware might alter the end user account number to the fraudsters account number, and then altering the amount.



❖ **Session Hijacking –** A session refers to a limited time of communication between two systems**.** The session ID is shared between the browser and web server at every request helps to indentify authenticated user.




**For example,** the time between you first log into your bank account, and then log off after your operation, is a session.

During a session hijacking, a malicious hacker places himself in between your computer and the website's server (Facebook for instance), while you are engaged in an active session.

At this point, the malicious hacker actively monitors everything that happens on your account, and can even kick you out and take control of it.

**Prevention**

Use Secure connection (HTTPS)

Don't log in on open wireless networks

Don't click directly links received in your mail. Copy and paste

Use a good antivirus

Log out at the end of every session

Keep your browser updated and other software updated at all times

**Session Hijacking attack methods**

- Man is the middle attack
- Session Sniffing- capturing network traffic between victim and website using sniffing tools
- Cross Site Scripting

❖ **SQL Injection –** SQL Injection (SQLi) is a type of an injection attack that makes it possible to execute malicious SQL statements by embedded in poorly designed application. These statements access the information that was not intended to be displayed.

Criminals may use it to gain unauthorized access to your sensitive data: customer information, personal data, trade secrets, intellectual property, and more.

Therefore, a successful SQL Injection attack can have very serious consequences.

- To gain complete access to all data in a database server.
- Alter data in a database and add new data. Like alter balances, void transactions, or transfer money to their account.
- Delete records from a database, even drop tables.

**Computer System Security (KNC-401)**

**UNIT – II**

**Confidentiality Policies**

- Prevention of unauthorized disclosure of information.

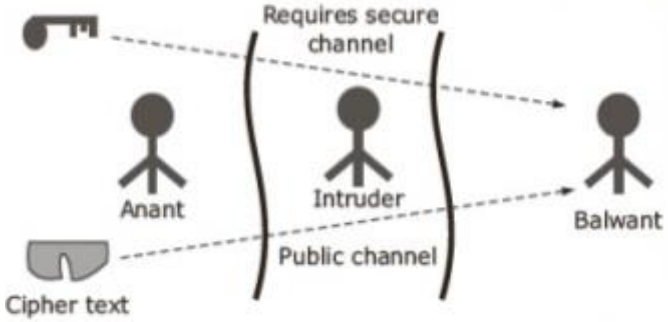
**Confinement Problem**

The confinement problem is the problem of preventing a server from leaking information that the user of the service considers confidential.

**Example-** Server balances bank accounts for clients.

- In this situation client send request.
- Server uses this data and performs some function and find out the result & then send this result to the client

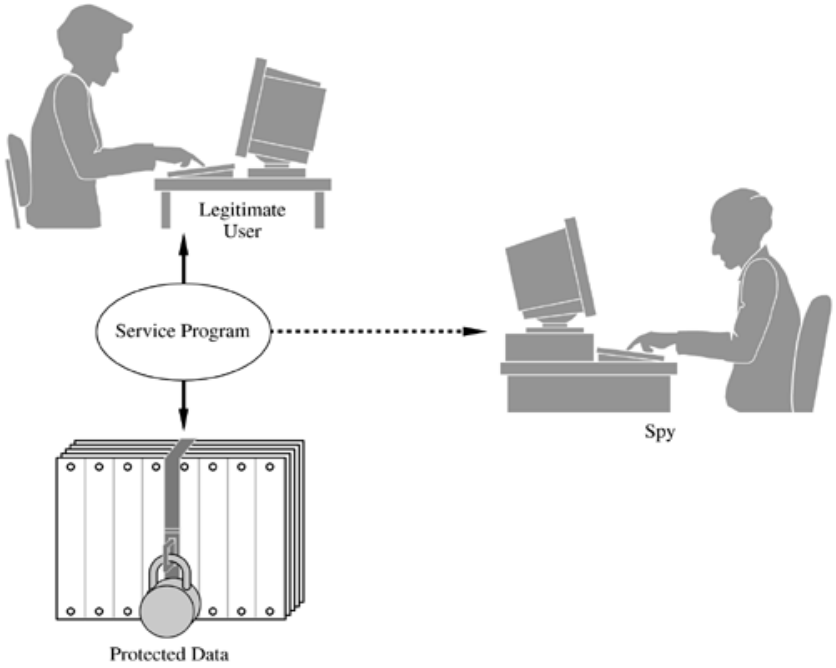
In this case confinement problem deals with preventing the server from leaking the confidential information of user.




The following types of channel can be used by a program to leak information.

1. **Covert Channels:** A communication channel (attack techniques) whose existence is hidden or covert.

"A covert channel is any communication channel that can be exploited by a process to transfer information in a manner that violates the system's security policy".

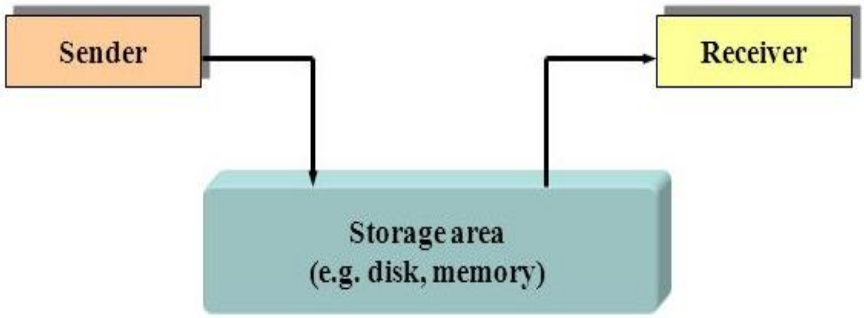



(Suppose a group of students is preparing for an exam for which each question has four choices (a, b, c, d); one student in the group, Sophie, understands the material perfectly and she agrees to help the others. She says she will reveal the answers to the questions, in order, by coughing once for answer "a," sighing for answer "b," and so forth. Sophie uses a communications channel that outsiders may not notice; her communications are hidden in an open channel. This communication is a human example of a covert channel.)

The following types of channels can be used by a program to leak information:

- Storage covert channel
- Timing Covert channel

**Storage covert channel - It** Involves the direct or indirect writing to storage location by one process and direct or indirect reading of the storage by another process. Example disk space, file lock

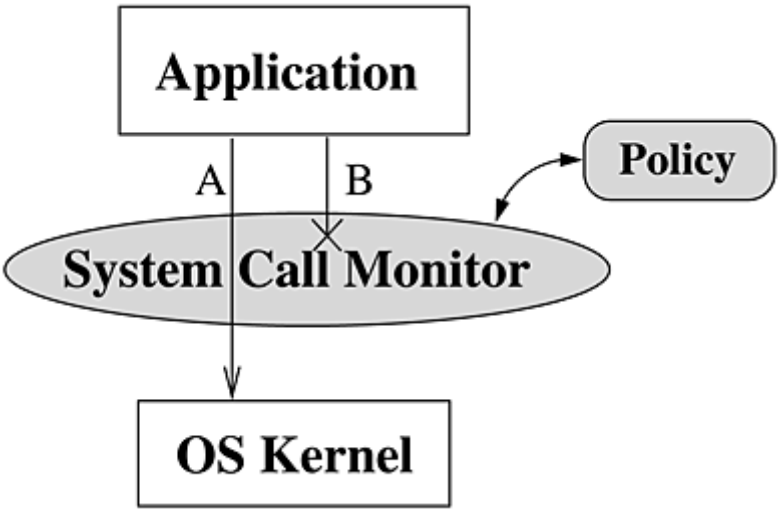



**Timing covert channel** – The use of delays between packets transmitted over computer networks i.e. one process relay information to another process by modulating its own use of system resources.

**Confinement Problem can implemented at many levels:**

1. **System call interposition**

It is a powerful technique for regulating and monitoring program behaviors. It gives security systems the ability to monitor all of the application's interaction with network, file system and other sensitive system resources. Many security systems, such as host intrusion detection systems (HIDS), leverage (to take advantage) system call interposition to detect anomalous program behaviors. The discrimination between normal and abnormal behavior is based on what system calls are normally invoked by a running program.



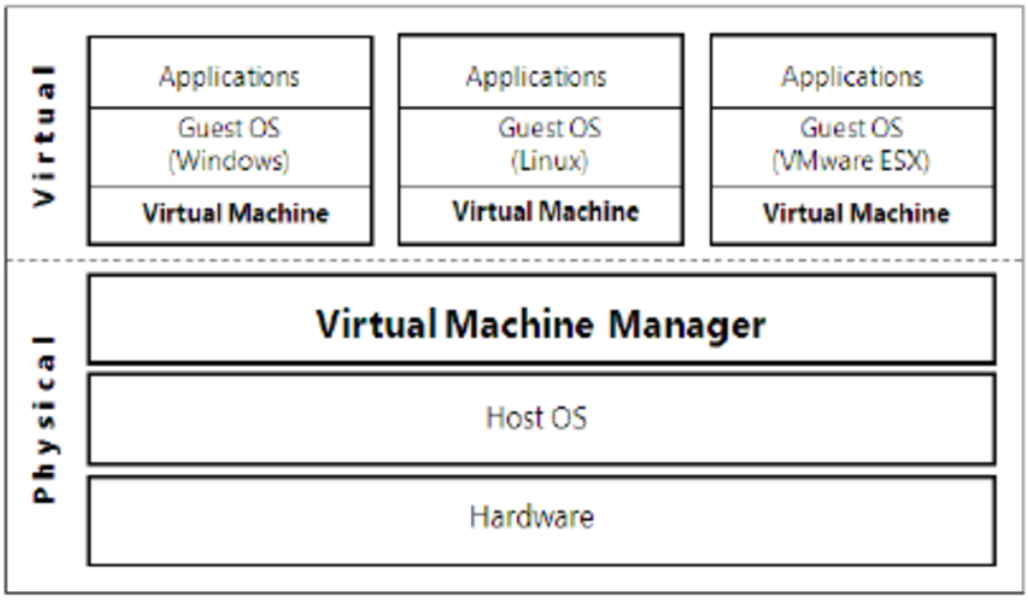

It is possible to identify and prevent damage if we can monitor every system call made by every process, and launch actions to preempt any damage. To guarantee the effectiveness and security of these security systems, system calls must be intercepted and handled safely and completely.

In addition to the preventive approaches, system call interception can significantly enhance the power and effectiveness of most offline intrusion detection techniques.

2. **Virtual Machine based isolation**

A virtual machine (VM) is a virtual environment that functions as a virtual computer system created on a physical hardware system. A VM behaves like an actual computer i.e. it can have an OS and application programs running on it. A VM is an isolated environment with access to a subset of physical resources of the computer system. By design, all VMware virtual machines are isolated from one another. This isolation enables multiple virtual machines to run securely while sharing hardware and ensures both their ability to access hardware and their uninterrupted performance.

The national security agency of USA tried to create an environment called NetTOP as follows:




In 1960's few computers and lots of users were these and VMs allows many users to share a single system so VMs were popular. From 1070's to 2000, due to fall in hardware price the VMs become non-existence. Now a day's VMs are heavily in cloud computing.

**VMM Security Assumption-** The popularity of VMs is based on the following VMM security assumption:

- Malware can infect guest OS and guest applications.
- But malware can't infect host OS and also can't infect other VM on same hardware.

However this requires that VMM protect itself from malicious and it not buggy. This assumption is not very unrealistic because VMM is much simpler than full OS and device drivers run in host OS therefore VMM can be checked and tested so that one can be more assure that the VMM is not buggy and it does not have security flaws.

### 3. Software Fault Isolation

It is a method to modify the programs so that they behave only in safe ways. This is embodied by a recent approach to security known as *software-based fault isolation* (SFI).

Now a day's mostly software has multiple threads that share same address space. If one thread is doing something dangerous then other should not be affected with that. So we need to isolate them as well.

Example - Device drivers should not corrupt kernel.

Simple solution to avoid this problem is to run different application in separate address spaces. But that make it very slow if applications communicate frequently (Known as (interprocess communication) which requires context switching per message. One more practical approach is the SFI.

**SFI Approach –**

- Although address space is same (shared) but it is divided into segment. These segments are used to contain the various threads. App1 can access only part 1 and app2 can access only part 2 and so on.


- Locate unsafe instructions: jmp, load, store

i.e. jump from one address to another address then it must be checked that it is not going into other app code or data segment. Therefore at compile time add guards before unsafe instructions when loading code ensure all guards are present.

**4. Rootkit**

It is a program (collection of programs) that hides in a computer and allows someone from a remote location to take full control of the computer. The rootkit author can execute programs, change settings, monitor activity, and access files on the remote computer. The word rootkits comes from the root user, which is the administrator account on LINUX systems. The kit refers to a toolkit or a set of tools.

Today rootkits are generally associated with malware – such as Trojans, worms, viruses – that conceal their existence and actions from users and other system processes.

Rootkits can be installed in a number of ways, including phishing attacks or social engineering tactics to trick users into giving the rootkit permission to be installed on the victim system, often giving remote cyber criminals administrator access to the system.

**Types of rootkits -** there are five types of rootkits.

1. **Hardware or firmware rootkit -** Hardware/Firmware rootkits are actually embedded (hide itself) within the firmware of devices such a network card, system BIOS etc. A firmware rootkit is activated if a BIOS function is called or when the machine is booted. The rootkit would always be available as long as the device is. And can be harder to detect. That's why it's good to let your anti-virus scan every device that your plugin.
2. **Boot loader rootkit -** Your computer's boot loader is an important tool. It loads your computer's operating system when you turn the machine on. It replaces or modifies the legitimate boot loader with another one thus enabling the Boot loader Level (Boot kit) to be activated even before the operating system is started. Boot loader Level (Boot kit) Rootkits are serious threat to security because they can be used to hack the encryption keys and passwords.
3. **Memory rootkit -** A memory rootkit is the one which hides itself in the memory (RAM). These rootkits have a short lifespan. They only live in your computer's RAM and will disappear once you reboot your system — though sometimes further work is required to get rid of them.
4. **User or Application rootkit -** Application rootkits replace standard files in your computer with rootkit files. They might also change the way standard applications work. These rootkits might infect programs such as Word, Paint, or Notepad. Every time you run these programs, you will give hackers access to your computer. The challenge here is that the infected programs will still run normally, making it difficult for users to detect the rootkit.
5. **Kernel mode rootkits -** The kernel is the core of the Operating System and Kernel Level Rootkits are created by adding additional code or replacing portions of the core operating system, with modified code via device drivers (in Windows) or Loadable Kernel Modules (Linux). This can give them easy access to your computer and make it easy for them to steal your personal information.

**Rootkit Detection -** Once an infection takes place, things get tricky. It is difficult to detect rootkits as this kind of malware is designed to stay hidden and do its business in the background. There are utilities designed to look for known and unknown types of rootkits through various methods, including behavioral-based methods (e.g., looking for strange behavior on a computer system), signature scanning and memory dump analysis.

There is no way to magically protect yourself from all rootkits. Fortunately, you can avoid these attacks by following the same common-sense strategies you take to avoid all computer viruses, including these.

- **Don't ignore updates -** Keeping your operating systems, antivirus software, and other applications updated is the best way to protect yourself from rootkits.
- **Watch out for phishing emails -** Phishing emails are sent by scammers who want to trick you into providing them your financial information or downloading malicious software, such as rootkits, onto your computer. Often, these emails will look like they come from a legitimate bank or credit card provider. These messages may state that your account is about to be frozen or that you need to verify your identity. The messages will also ask that you click on a link.

  If you do, you'll be taken to a fake website. Once there, you might accidentally download a rootkit to your computer.

- **Don't download files sent by people you don't know -** Be careful, too, when opening attachments. Don't open attachments sent to you by people you don't know. Doing so could cause a rootkit to be installed in your computer. If you receive a suspicious attachment? Delete the email message immediately.

Removing a rootkit is a complex process and typically requires the use of specialized tools. Major security firms, such as Symantec, Kaspersky Lab and Intel Security (McAfee), offer rootkit scanners to enterprise customers.

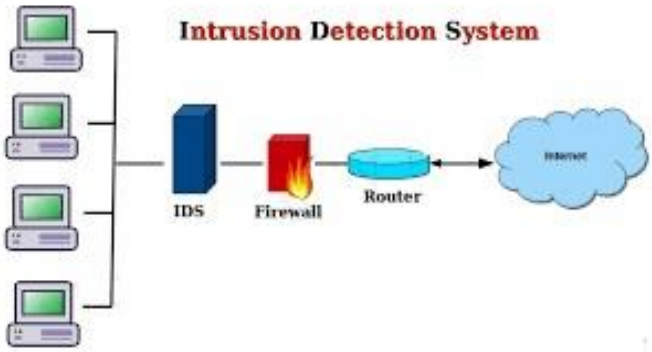
### 5. Intrusion Detection System

**Intrusion** can be defined as a subversion of security to gain access to a system.

- These unauthorized accesses to computer or network systems are often designed to study the system's weakness for future attacks.
- Other forms of intrusions are aimed at limiting access or even preventing access to computer systems or networks.

**What is an IDS?**

An intrusion detection system (IDS) is a software application or hardware appliance (Device) that monitors traffic moving on networks and through systems to search for suspicious activity or policy violation and known threats, sending up alerts when it finds such items.




**What are we protecting?**

- Data
- Availability
- Privacy

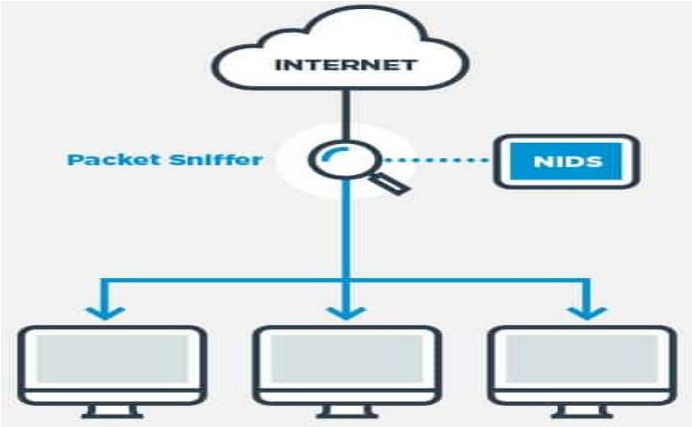
**Who are the intruders?**

- Hackers
- Thieves

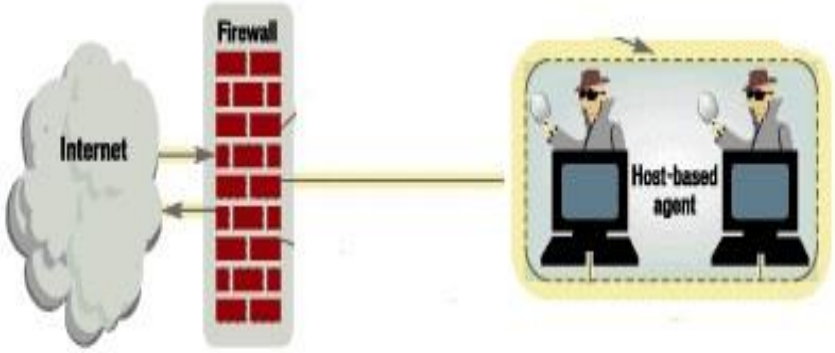
**Types of IDS -** Intrusion detection systems can be broken into two broad categories:

- Network-based: A system that analyzes the stream of packets which travel across the network traffic.
- Host-based: A system that monitors important operating system and application files.

**Network-based -** A Network based Intrusion Detection System (NIDS) is one common type of IDS that analyzes network traffic at all layers of the Open Systems Interconnection (OSI) model and makes decisions about the purpose of the traffic, analyzing for suspicious activity. Once a suspicious activity (attack) is identified or abnormal behavior is observed, the alert can be sent to the administrator.



**Host - based -** Host-based intrusion detection systems (HIDS) analyze network traffic and system-specific settings such as software calls, local security policy, local log audits, and more. A HIDS must be installed on each machine and requires configuration specific to that operating system and software. Host-Based, which monitors the characteristics of a single host and the events occurring within that host for suspicious activity. Host intrusion detection systems (HIDS) run on independent hosts or devices on the network. A HIDS monitors the incoming and outgoing packets from the device only and will alert the administrator if suspicious or malicious activity is detected.




**Detection Method of IDS:**

1. **Signature-based Method:** Signature-based IDS detects the attacks on the basis of the specific patterns such as number of bytes or number of 1's or number of 0's in the network traffic. It also detects on the basis of the already known malicious instruction sequence that is used by the malware. The detected patterns in the IDS are known as signatures.

Signature-based IDS can easily detect the attacks whose pattern (signature) already exists in system but it is quite difficult to detect the new malware attacks as their pattern (signature) is not known.

**2. Anomaly-based Method:** Anomaly-based IDS was introduced to detect the unknown malware attacks as new malware are developed rapidly. In anomaly-based IDS there is use of machine learning to create a trustful activity model and anything coming is compared with that model and it is declared suspicious if it is not found in model. Machine learning based method has a better generalized property in comparison to signature-based IDS as these models can be trained according to the applications and hardware configurations.

**A *user ID*** (UID) is a unique positive integer assigned by a Unix-like operating system to each user. Each user is identified to the system by its UID, and user names are generally used only as an interface for humans.
UIDs are stored, along with their corresponding user names and other user-specific information, in the */etc/passwd* file, which can be read with the *cat* command as follows:

cat /etc/passwd

The third field contains the UID, and the fourth field contains the *group ID* (GID), which by default is equal to the UID for all ordinary users.
The UID of 0 has a special role: it is always the *root account* (i.e., the omnipotent administrative user).
**Process ID -** Each Process is uniquely identified by an unique integer is known as PID. Each process has process ID (PID), parent Process ID (PPID), start time, etc.

**Ownership of Linux files -** Every file and directory on your Unix/Linux system is assigned 3 types of owner, given below.

- **User -** A user is the owner of the file. By default, the person who created a file becomes its owner. Hence, a user is also sometimes called an owner.
- **Group -** A User - group can contain multiple users. All users belonging to a group will have the same access permissions to the file.
- **Other -** Any other user who has access to a file. This person has neither created the file, nor does he belong to a user group who could own the file.

**Permissions -** Every file and directory in your UNIX/Linux system has following three permissions defined for all the three owners discussed above.

- **Read (r) -** This permission give you the authority to open and read a file. Read permission on a directory gives you the ability to lists it's content.
- **Write (w) -** The write permission gives you the authority to modify the contents of a file. The write permission on a directory gives you the authority to add, remove and rename files stored in the directory. Consider a scenario where you have to write permission on file but do not have write permission on the directory where the file is stored. You will be able to modify the file contents. But you will not be able to rename, move or remove the file from the directory.
- **Execute (x) -** In Windows, an executable program usually has an extension ".exe" and which you can easily run. In Unix/Linux, you cannot run a program unless the execute permission is set. If the execute permission is not set, you might still be able to see/modify the program code (provided read & write permissions are set), but not run it.
- **No permission (-**) - means no permission

| Permission Type | Symbol |
|---|---|
| No Permission | --- |
| Execute | --x |
| Write | -w- |
| Execute + Write | -wx |
| Read | r-- |
| Read + Execute | r-x |
| Read +Write | rw- |
| Read + Write +Execute | rwx |

**Computer System Security (KNC-401)**

**UNIT – III**

**Security Policies** are a formal set of rules which is issued by an organization to ensure that the user who are authorized to access company technology and information assets comply with rules and guidelines related to the security of information. It is a written document in the organization which is responsible for how to protect the organizations from threats and how to handles them when they will occur.

### 1. Access Control Concepts

Access control is a process by which resources or services are granted or denied on a computer system or network.

In access control systems, users must present credentials before they can be granted access.

There are three factors that can be used for authentication:

- Something only known to the user, such as a password or PIN
- Something that is part of the user, such as a fingerprint, retina scan or another biometric measurement
- Something that belongs to the user, such as a card or a key

It is a fundamental concept in security that minimizes risk to the business or organization.

**Types of Access Control -** Organizations must determine the appropriate access control model to adopt based on the type and sensitivity of data they're processing. There are five types of access control:

- Discretionary Access Control (DAC)
- Mandatory Access Control (MAC)
- Role Based Access Control (RBAC)
- Rule Based Access Control
- Attribute Based Access Control (ABAC)

**Discretionary Access Control (DAC) -** An access control method in which owners or administrators of the protected system, data or resource set the policies defining who or what is authorized to access the resource.

**Mandatory Access Control (MAC) -** A security model in which access rights are regulated by a central authority based on multiple levels of security. Often used in government and military environments, classifications are assigned to system resources and the operating system or security kernel, grants or denies access to those resource objects based on the information security clearance of the user or device. For example, Security Enhanced Linux is an implementation of MAC on the Linux operating system.

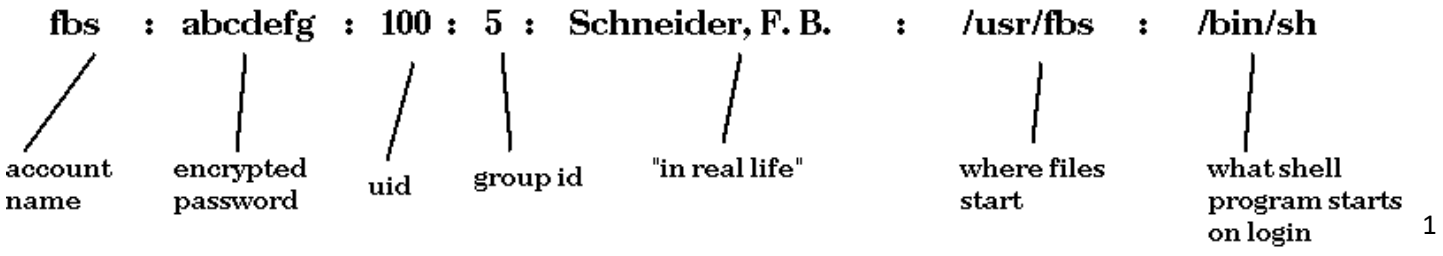
**Role Based Access Control (RBAC) -** RBAC grants access based on a user's role and implements key security principles, such as "least privilege" and "separation of privilege." Thus, someone attempting to access information can only access data that's deemed necessary for their role.

**Rule Based Access Control -** A security model in which the system administrator defines the rules that to govern access to resource objects. Often these rules are based on conditions, such as time of day or location.

**Attribute Based Access Control (ABAC) -** A methodology that manages access rights by evaluating a set of rules, policies and relationships using the attributes of users, systems and environmental conditions.

### 2. UNIX --WINDOWS Access Control

UNIX uses access control lists. A user logs into UNIX and has a right to start processes that make requests. A process is "bigger" than a subject; many domains may correspond to a single process. Each process has an identity (uid). This uid is obtained from the file that stores user passwords: /etc/passwd. An entry in /etc/passwd may look like:



Every process inherits its uid based on which user starts the process. Every process also has an *effective* uid, also a number, which may be different from the uid.

**Ownership of Linux files -** Every file and directory on your Unix/Linux system is assigned 3 types of owner, given below.

- **User -** A user is the owner of the file. By default, the person who created a file becomes its owner. Hence, a user is also sometimes called an owner.
- **Group -** A User - group can contain multiple users. All users belonging to a group will have the same access permissions to the file.
- **Other -** Any other user who has access to a file. This person has neither created the file, nor does he belong to a user group who could own the file.

**Permissions -** Every file and directory in your UNIX/Linux system has following three permissions defined for all the three owners discussed above.

- **Read (r) -** This permission give you the authority to open and read a file. Read permission on a directory gives you the ability to lists it's content.
- **Write (w) -** The write permission gives you the authority to modify the contents of a file. The write permission on a directory gives you the authority to add, remove and rename files stored in the directory. Consider a scenario where you have to write permission on file but do not have write permission on the directory where the file is stored. You will be able to modify the file contents. But you will not be able to rename, move or remove the file from the directory.
- **Execute (x) -** In Windows, an executable program usually has an extension ".exe" and which you can easily run. In Unix/Linux, you cannot run a program unless the execute permission is set. If the execute permission is not set, you might still be able to see/modify the program code (provided read & write permissions are set), but not run it.
- **No permission (-**) - means no permission

| Permission Type | Symbol |
|---|---|
| No Permission | --- |
| Execute | --x |
| Write | -w- |
| Execute + Write | -wx |
| Read | r-- |
| Read + Execute | r-w |
| Read + Write | Rw- |
| Read + Write + Execute | rwx |

**Windows NT Access Control -** Windows NT supports multiple file systems, but the protection issues we will consider are only associated with one: NTFS. NTFS is structured so that a file is a set of properties, the contents of the file being just one of those properties. An ACL is a property of an item. The ACL itself is a list of entries: (user or group, permissions). NTFS permissions are closer to extended permissions in UNIX than to the 9 mode bits. The permission offer a rich set of possibilities:

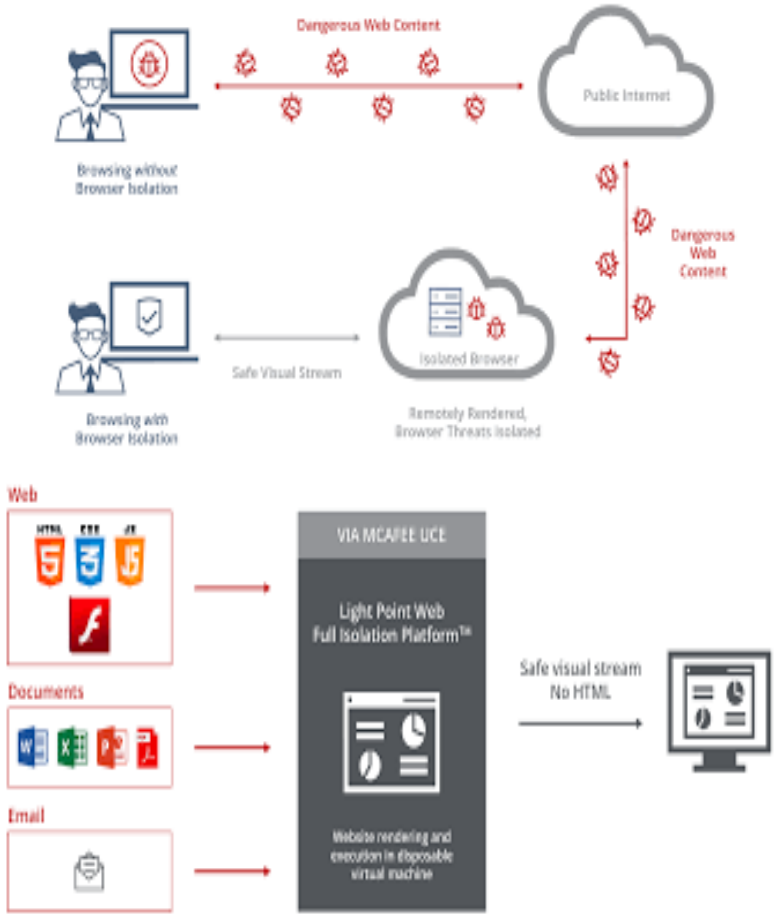
- R -- read
- W -- write
- X -- execute
- D -- delete
- P -- modify the ACL
- O -- make current account the new owner ("take ownership")

The owner is allowed to change the ACL. A user with permission P can also change the ACL. A user with permission O can take ownership. There is also a packaging of privileges known as permissions sets:

- no access
- read -- RX
- change -- RWXO
- full control -- RWXDPO

### 3. Browser Isolation

Browser isolation is an approach to cyber security that separates an end user's web browsing activity from the local network and its infrastructure,(maintaining browser operations away from a bare-metal environment or intermediate server hardware system) in order to provide barriers against malware, viruses and other threats.




**The benefits -** Browser isolation is an effective solution against many types of web based threats, such as ransom ware and malvertising.

- If a user lands on a site that intends to steal user data, browser isolations keeps the browsing data safe since the browsing session does not contact the system's file storage.
- If a user lands on a site that contains malware, browser isolation prevents the malware from reaching the system. The session is opened in a virtual browser and all browsing data gets erased once the user quits the session.
- Browser isolation solutions reduce the complexity and costs associated with protecting individual endpoint devices from web based cyber attacks, as there is no need for endpoint software installation.

### 4. Web Security

The CIA Triad refers to the 3 **goals** of **cyber security** Confidentiality, Integrity, and Availability of the organizations systems, network and data. ... Encryption services can protect your data at rest or in transit and prevent unauthorized access to protected data.

Web security is also known as “Cyber security”. It basically means protecting a website or web application by detecting, preventing and responding to cyber threats.



**Web Security Goals -** The objective of Web Security is to protect information from being stolen, compromised or attacked. Web security can be measured by at least one of three goals-

1. Protect the confidentiality of data.
2. Preserve the integrity of data.
3. Promote the availability of data for authorized users.

**Confidentiality** – Any important data you have should only be accessible to people or by systems to who you have given permission. It prevents essential information from reaching the wrong people while making sure that the right people can get it. Data encryption is a good example to ensure confidentiality.

**Integrity** – Integrity refers to the methods for ensuring that data is real, accurate and safeguarded from unauthorized user modification. It is the property that information has not be altered in an unauthorized way, and that source of the information is genuine.

**Availability** – all systems, services and information must be accessible when required by the business or its clients.

**Threat Model -** Threat modeling is the practice of identifying and prioritizing potential threats and security mitigations to protect something of value, such as confidential data or intellectual property. Threat modeling helps to define valuable assets and the possible attacks that they are likely to face. The purpose of threat modeling is to determine where the most effort should be applied to keep a system secure.

**What Methodologies to Use for Threat Modeling?**

There are several methodologies that you can use for threat modeling. The most popular one is STRIDE created by Microsoft in 1999. The name stands for six key aspects that you should consider when threat modeling: Spoofing, Tampering, Repudiation, Information disclosure, Denial of service, and Elevation of privileges.

## 5. HTTP content rendering

Rendering is something about displaying fetched content from the internet. Server-side rendering (SSR) is a method of providing pre-generated HTML as a response to an HTTP request.

## 6. Security Interface

In computing, an interface is a shared boundary across which two or more separate components of computer system exchange information. The exchange can be between software, computer hardware, peripheral devices, humans, and combinations of these.

Security interface provide the features for security such as authorization, access to digital certificates, and access to items in key chains etc.

## 7. Cookies, frames and frame busting

**Cookies** are a small piece of data sent from a website and stored on the user's computer by the user's web browser while the user is browsing. Cookies were designed to be a reliable mechanism for websites to remember stateful information (such as items added in the shopping cart in an online store) or to record the user's browsing activity. Cookies only contain bits of text. The text can be a user ID, session ID, or any other text.

**HTML frame** defines the particular area within an HTML file where another HTML web page can be displayed.

A frame is used with frameset, and it divides a webpage into multiple sections or frames, and each frame can contain different web pages. A collection of frames in the browser window is known as a frameset. The window is divided into frames in a similar way the tables are organized: into rows and columns.

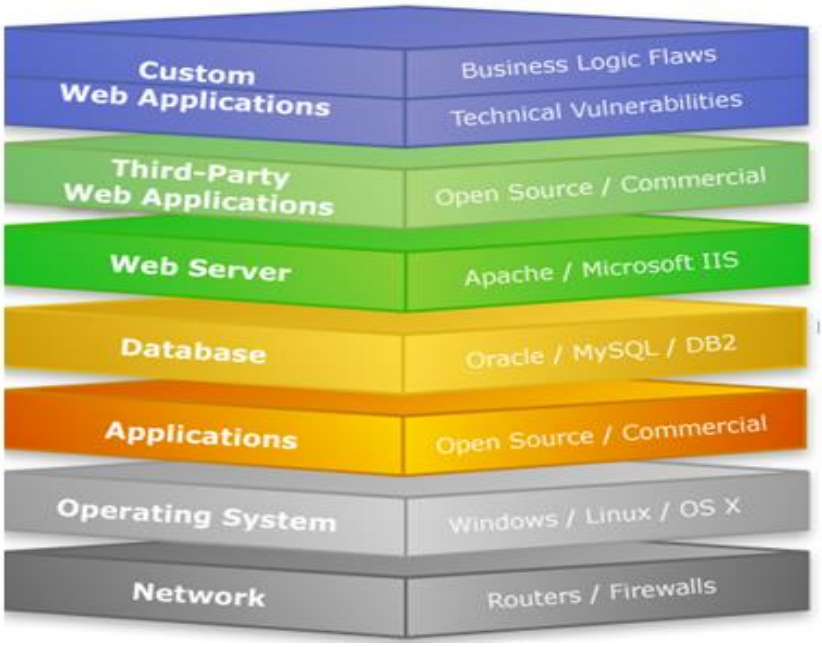
**Frame buster** is a code of annotation used by web applications to prevent their web pages from being displayed within a sub frame. A frame is a subdivision of a Web browser window and can act like a smaller window. It's usually deployed to prevent a frame from an external Web site being loaded from within a frameset without permission often as part of clickjacking attack. Frame busting is the recommended defense against click-jacking.

## 8. Major web server threats

Websites are hosted on web servers. Web servers are themselves computers running an operating system; connected to the back-end database, running various applications. Any vulnerability in the applications, Database, Operating system or in the network will lead to an attack on the web server. Vulnerability stack of a web server is given below in (fig.):

These threats are common ones that attackers like to use to either gain access to your server or bring it to its knees.

**Brute Force Attack -** In a brute force attack, the intruder attempts to gain access to a server by guessing a user password (usually the root administrator) through the SSH server, Mail server, or other service running on your system. The attacker will normally use software that will check every possible combination to find the one that works. Brute force detection software will alert you when multiple failed attempts to gain access are in progress and disable access from the offending IP address.




**Open Relay -** A Mail Transfer Agent (MTA) normally uses an SMTP server to send email from your server's users to people around the world. With an open relay, anyone can use your SMTP server, including spammers. Not only is it bad to give access to people who send spam, it could very well get your server placed on a DNS blacklist that some ISPs will use to block mail from your IP. It is very easy to close an open relay.

**Botnet -** Attackers use botnets to automatically run and distribute malicious software on "agent" servers. They then use the agent machines to attack or infect others. Because all of this can be done automatically without user intervention, botnets can spread very quickly and be deadly for large networks. They are commonly used in DDoS attacks and spam campaigns.

**DoS -** DoS stands for Denial of Service, and is a technique attackers will use to effectively shut off access to your site. They accomplish this by increasing traffic on your site so much that the victim's server becomes unresponsive. While some DoS attacks come from single attackers, others are coordinated and are called Distributed Denial of Service (DDoS) attacks. Often times, the users of computers executing a DDoS do not even know their computers are being used as agents.

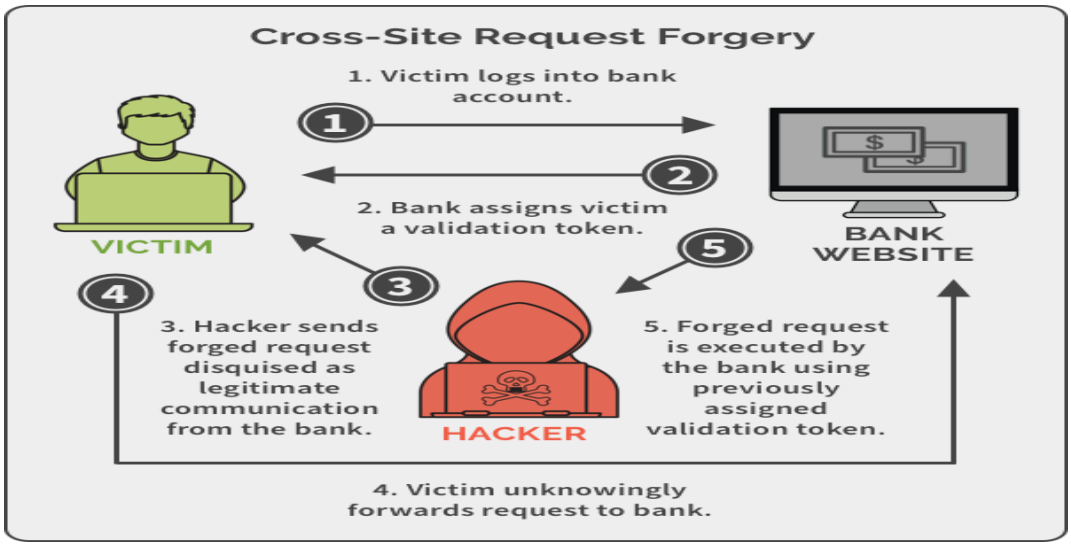
**Cross-site Scripting -** Cross-site scripting or XSS is a technique that makes use of vulnerabilities in web applications. According to UK dedicated hosting server specialists at 34SP.com, the vulnerability allows the attacker to inject code in a server-side script that they will use to execute malicious client-side scripts or gather sensitive data from the user. You can fix most XSS problems by using scanner software to detect vulnerabilities and then fix whatever you find.

**SQL Injection -** Like XSS, SQL injection requires a vulnerability to be present in the database associated with a web application. The malicious code is inserted into strings that are later passed to the SQL server, parsed, and executed. As with other vulnerability-dependent attacks, you can prevent it by scanning for problem code and fixing it

**Malware -** Malware can take many forms, but as the name implies, it is malicious software. It can take the form of viruses, bots, spyware, worms, trojans, rootkits, and any other software intended to cause harm. In most cases, malware is installed without the user's direct consent. It may attack the user's computer and/or attack other computers through the user's own system. Having proper firewall and security software protection can usually prevent malware from spreading.

## 9. Cross site request forgery (CSRF)

Cross-site request forgery (also known as one click attack) is a web security vulnerability that allows an attacker to induce users to perform actions that they do not intend to perform.




CSRFs are typically conducted using malicious social engineering, such as an email or link that tricks the victim into sending a forged request to a server. As the unsuspecting user is authenticated by their application at the time of the attack, it's impossible to distinguish a legitimate request from a forged one.

let's assume that I'm logged into my account on *examplebank.com,* which allows for online banking features, including transferring funds to another account etc.

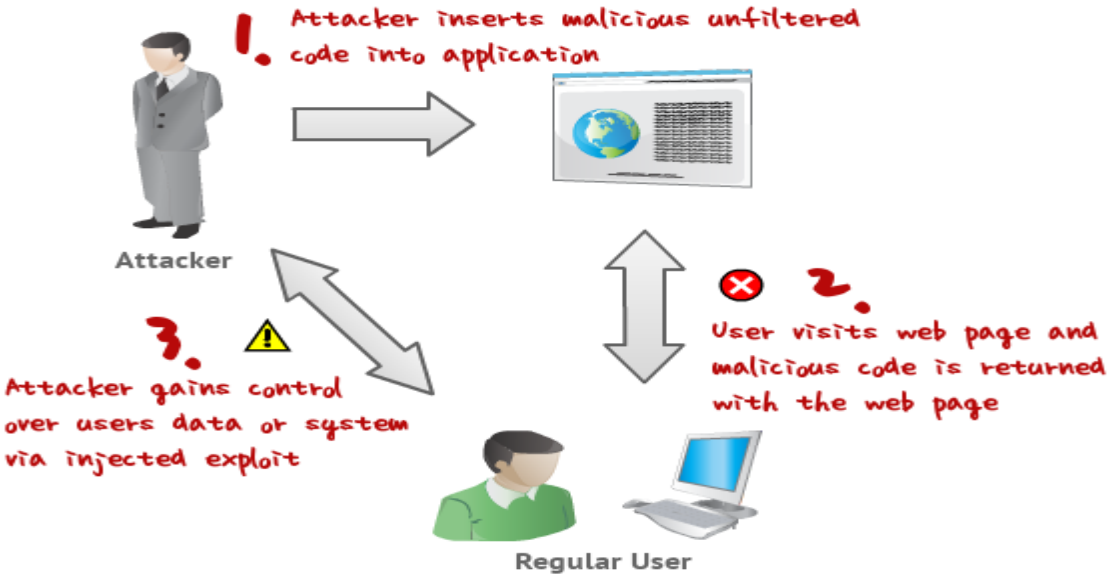
Now let's say I happen to visit *somemalicioussite.com*. It just so happens that this site is trying to attack people who bank with *examplebank.com* and has set up a CSRF attack on its site. The attack will transfer $1,500.00 to account number 123456789. Somewhere on *somemalicioussite.com,* attackers have added some line of code. Upon loading that iframe, my browser will send that request to *examplebank.com,* which my browser has already logged in as me. The request will be processed and send $1,500.00 to account 123456789.

**Preventing Cross-Site Request Forgery (CSRF) Vulnerabilities -** The most common method to prevent Cross-Site Request Forgery (CSRF) attacks is to append CSRF tokens to each request and associate them with the user's session. Such tokens should at a minimum be unique per user session, but can also be unique per request.

## 10.Cross site scripting

Cross-site scripting (XSS) is one of the most common application-layer web attacks. It is a type of computer security vulnerability typically found in web applications. XSS attacks enable attackers to inject code in a server-side script that they will use to execute malicious client-side scripts or gather sensitive data from the user.

Malicious scripts are often delivered in the form of bits of JavaScript code executed by the victim's browser, but exploits can incorporate malicious executable code in many other languages, including Java, Ajax, Flash and HTML.




Cross-site scripting allows an attacker to execute malicious scripts in another user's browser. However, the attacker doesn't attack the victim directly; rather, the attacker exploits a vulnerability in a website the victim visits and gets the website to deliver the malicious script for the attacker.

**What are the types of XSS attacks? -** There are three main types of XSS attacks. These are:

- **Reflected XSS i**s the simplest variety of cross-site scripting. It arises when an application receives data in an HTTP request and includes that data within the immediate response in an unsafe way. The attacker uses phishing emails and other social engineering methods to lure the victim to inadvertently make a request to the server that includes the XSS payload.
- **Stored XSS** is the most damaging type of cross-site scripting attack**.** It arises when an application receives data from an untrusted source and includes that data within its later HTTP responses in an unsafe way. Data might arrive from other untrusted sources; for example, a webmail application displaying messages received over SMTP, a marketing application displaying social media posts, or a network monitoring application displaying packet data from network traffic.
- **DOM-based XSS** arises when an application contains some client-side JavaScript that processes data from an untrusted source in an unsafe way, usually by writing the data back to the Document Object Model (DOM). The web application then reads the data from the DOM and delivers it to the browser. If the data isn't handled correctly, the attacker is able to inject a payload that will be stored as part of the DOM. The payload is then executed when the data is read back from the DOM.

### 11. Defenses and protections against XSS

Ultimately, XSS is a type of code injection very similar in nature to SQL injection. Like protecting against any code injection attack, the best defense is thorough and well-tested sanitization of any and all user input.

Site owners need to determine every input path by which their web site accepts incoming data. Each path must be hardened against malicious data that can represent executable code. Often this requires implementing multiple filters along the communication pathway – for example, a web application firewall such as ModSecurity plus input sanitization within server-side input processing code. Developers should also use tools such as XSSMe for Firefox or domsnitch for Google Chrome to test their own sites for XSS vulnerabilities.

As a secondary defense, a site could link browser cookie credentials to the user's IP address. While not a perfect defense, this would prevent easy abuse of users' cookies. An attacker could engineer a system to lift the user's IP

address and spoof their own actions under that address but this degree of attack will be far less widespread than simple cookie theft.

### 12. Finding Cross-Site Scripting Vulnerabilities

XSS vulnerabilities may occur if:

- Input coming into web applications is not validated
- Output to the browser is not HTML encoded

**XSS Examples**

Suppose there's a URL on Google's site, ***http://www.google.com/search?q=flowers***, which returns HTML documents containing the fragment

*** Your search for 'flowers' returned the following results: ***

i.e., the value of the query parameter q is inserted into the page returned by Google.

Suppose further that the data is not validated, filtered or escaped. Evil.org could put up a page that causes the following URL to be loaded in the browser (e.g., in an invisible*<iframe>*):

***http://www.google.com/search?q=flowers+%3Cscript%3Eevil_script()%3C/script%3E***

When a victim loads this page from www.evil.org, the browser will load the iframe from the URL above. The document loaded into the iframe will now contain the fragment.

*** Your search for 'flowers <script>evil_script()</script>'***

Loading this page will cause the browser to execute *evil_script()*. Furthermore, this script will execute in the context of a page loaded from *www.google.com.*

**Preventive measures/Secure development**

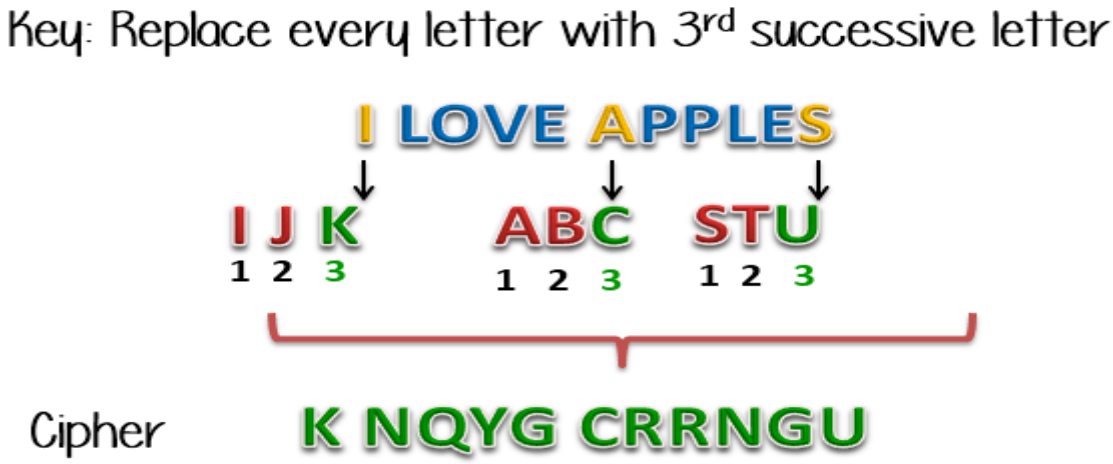
- Contextual output encoding/escaping of string input
- Safely validating untrusted HTML input
- Cookie security
- Disabling scripts

**UNIT – IV**

**CSS - (Basic cryptography)**

## 1. Cryptography

Cryptography is the science of using mathematics to encrypt and decrypt data. Cryptography enables you to store sensitive information or transmit it across insecure networks (like the Internet) so that it cannot be read by anyone except the intended recipient.




**Basic Terminology**

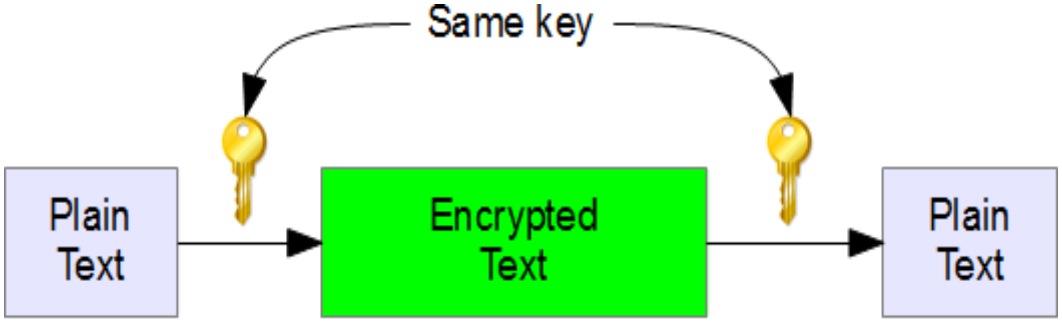
- **Plain text -** Plaintext is ordinary readable text before being encrypted into cipher text.
- **Cipher text -** Cipher text is the data that has been encrypted. In other word it is the encrypted text.
- **Encryption -** It is the process to convert the data (plain text) in some unreadable form (cipher text) in such a way that only authorized person can access it.
- **Decryption -** The conversion of encrypted data into its original form is called decryption. In the other word the reverse of encryption is called as decryption**.**

The concept of encryption and decryption requires some extra information for encrypting and decrypting the data. This information is known as ***key***. There may be cases when same key can be used for both encryption and decryption while in certain cases, encryption and decryption may require different keys.

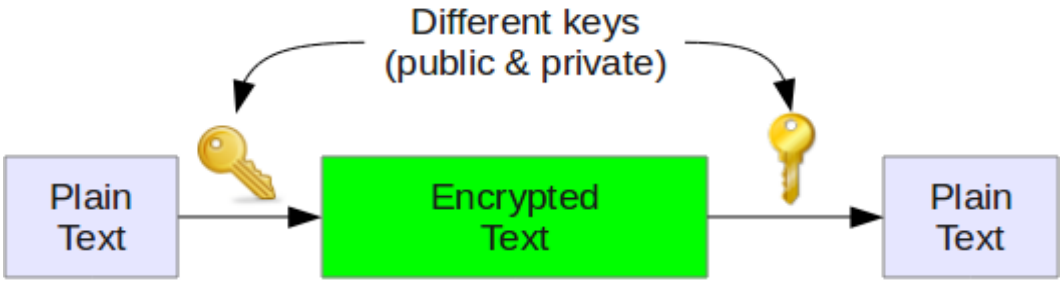
**Types of Cryptography -** Cryptography also allows senders and receivers to authenticate each other through the use of key pairs. There are various types of Cryptography (algorithms) for encryption, some common algorithms include:

- Secret (symmetric) key Cryptography (example DES, Triple DES, AES, RC5)
- Public (asymmetric) key cryptography (RSA, Elliptic Curve)

**Secret Key Cryptography -** Both the sender and receiver share a single key. The sender uses this key to encrypt plaintext and send the cipher text to the receiver. On the other side the receiver applies the same key to decrypt the message and recover the plain text. Since only single key is used so we say that this is a symmetric encryption.




**Public-Key Cryptography -** In this method, each party has a private key and a public key. The private is secret and is not revealed while the public key is shared with all those whom you want to communicate with. The public key is used for encryption and for decryption private key is used.




**2. RSA public key cryptography**

The RSA algorithm is named after Ron Rivest, Adi Shamir and Len Adleman, who invented it in 1977.

The RSA cryptosystem is the most widely-used public key cryptography algorithm in the world. It can be used to encrypt a message without the need to exchange a secret key separately. Messages encrypted using the public key can only be decrypted with the private key. The steps for the RSA algorithm are the following way:

**Key Generation**

1. Choose two different large random prime numbers *p* and *q*.
2. Calculate $n = p \times q$.
3. Calculate the $\Phi(n) = (p - 1) \times (q - 1)$ .
4. Choose an integer *e* such that $1 < e < \Phi(n)$, and *e* is co-prime to $\Phi(n)$ **i.e.** *e* and $\Phi(n)$ share no factors other than 1{ gcd $(e, \Phi(n)) = 1$}.
5. Select *d* such that $e \times d \equiv 1 \bmod (\phi(n))$.
6. Choose (*e, n*) as the public key.
7. Choose (*d, n*) as the private key.

**Encryption**

Suppose the sender wishes to send some text message (plain text) "M" to someone whose public key is *(e, n).* The sender then represents the plaintext as a series of numbers less than n. The encryption process is simple mathematical step as -

$$\textit{Cipher text } (C) = P^e \bmod n$$

**Decryption**

Receiver can recover text message "M" from by using her private key *(d, n)* in the following procedure:

$$\text{Plain text } (P) = C^d \bmod n$$

Example -

**I.** Choose p = 3 and q = 11.

**II.** Compute $n = p \times q = 3 \times 11 = 33$

**III.** Compute $\Phi(n) = (p - 1) \times (q - 1) = 2 \times 10 = 20$

**IV.** Choose *e* such that $1 < e < \Phi(n)$, and *e* is co-prime to $\Phi(n)$, let e = 7.

**V.** Select *d* such that $e \times d \equiv 1 \bmod (\phi(n))$. one solution is d = 3.

**VI.** Public key is (e, n) →(7, 33).

**VII.** Private key is (d, n) →(3, 33).

**VIII.** The encryption of M = 2 is $C = 2^7 \bmod 33 = 29$.

**IX.** The decryption of C = 29 is $M = 29^3 \bmod 33 = 2$.

**3. Digital signature**

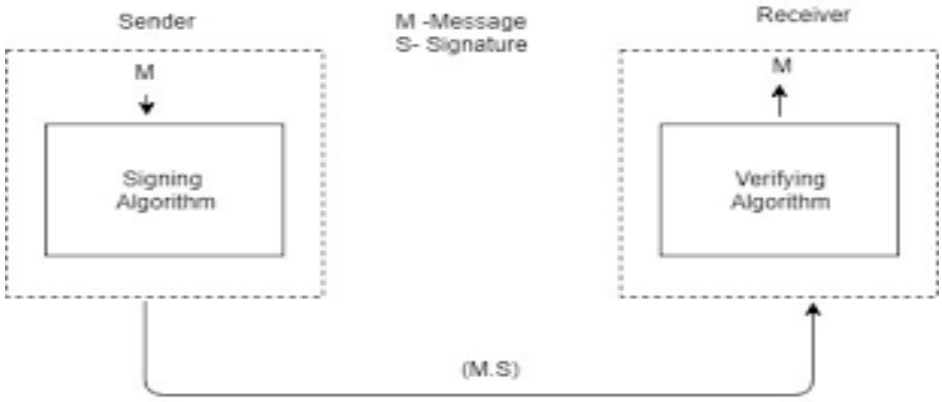
A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software or digital document. Digital signatures create a virtual fingerprint that is unique to a person or entity and are used to identify users and protect information in digital messages or documents. It provide the authenticity, integrity and non - repudiation of a message.

*(The terms digital signature and electronic signature are sometimes confused or used interchangeably. While digital signatures are a form of electronic signature, not all electronic signatures are digital signatures. Electronic signatures—also called e-signatures—are any sound, symbol, or process that shows the intent to sign something. This*

*could be a scan of your hand-written signature, a stamp, or a recorded verbal confirmation. An electronic signature could even be your typed name on the signature line of a document.)* - ***Only for knowledge purpose***

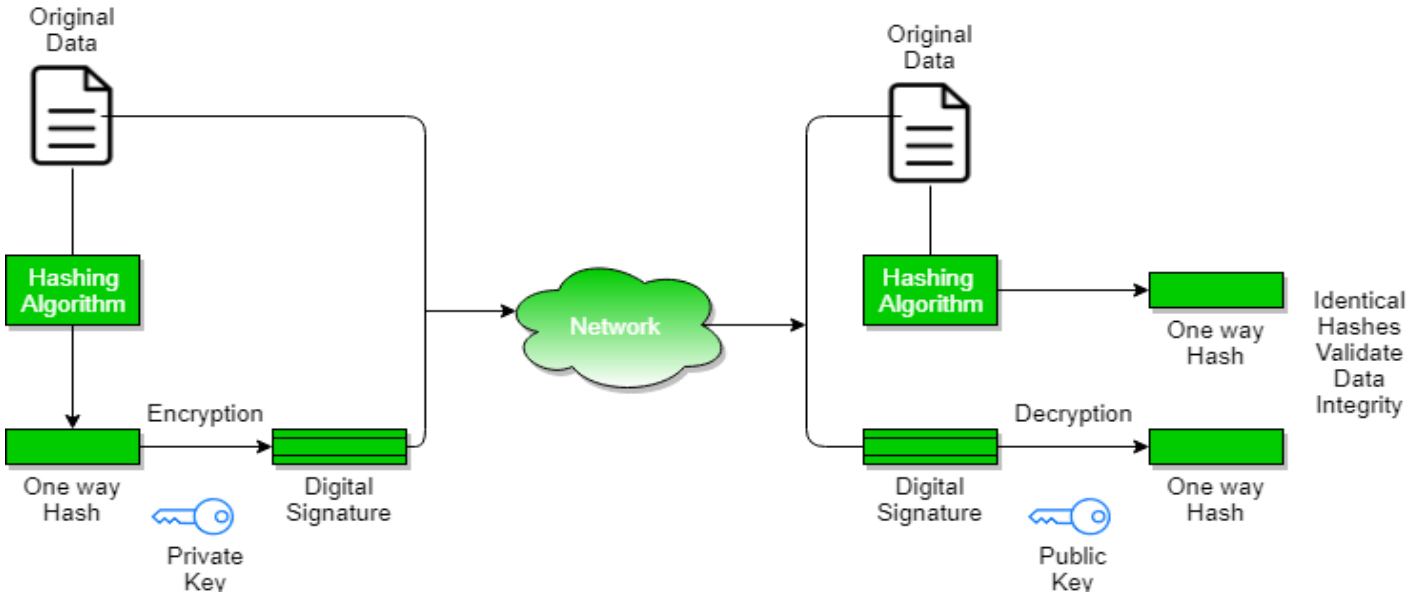
**How do digital signatures work?**

Digital signatures rely on certain types of encryption to ensure authentication. Authentication is the process of verifying that information is coming from a trusted source. Digital signatures are based on public key cryptography, such as RSA. Digital signatures work because public key cryptography depends on two mutually authenticating cryptographic keys. In a digital signature the signer uses her private key, applied to a signing algorithm, to sign the document. The verifier, on the other hand, uses the public key of the signer, applied to the verifying algorithm, to verify the document. This is how digital signatures are authenticated.




*Digital signature process*

**How to create a digital signature -** To create a digital signature, the hash value of a message is encrypted with a use's private key. The encrypted hash along with other information is the digital signature. Anyone who knows the user's public key can verify the integrity of the message that is associated with the digital signature. In this case, an attacker who wishes to alter the message would need to know the user's private key.




**Uses of digital signatures -** Industries use digital signature technology to streamline processes and improve document integrity. It increases the transparency of online interactions and develops trust between customers, business partners, and vendors. Industries that use digital signatures include:

- **Government -** Digital signatures are used by governments worldwide for a variety of uses, including processing tax returns, verifying business-to-government (B2G) transactions, ratifying laws and managing contracts.

- **Healthcare -** Digital signatures are used in the healthcare industry to improve the efficiency of treatment and administrative processes, to strengthen data security, for e-prescribing and hospital admissions.
- **Manufacturing -** Manufacturing companies use digital signatures to speed up processes, including product design, quality assurance (QA), manufacturing enhancements, marketing and sales.
- **Finance Services -** The U.S. financial sector uses digital signatures for contracts, paperless banking, loan processing, insurance documentation, mortgages, and more.

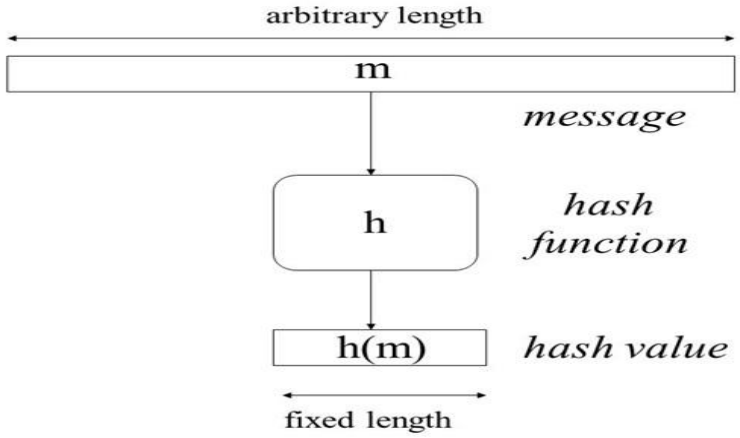
**Digital signature security features and benefits -** Security features embedded in digital signatures ensure that a document is not altered and that signatures are legitimate.

- **PINs, passwords and codes** - Used to authenticate and verify a signee's identity and approve their signature. Email, username and password are most common.
- **Time stamping** - Provides the date and time of a signature. Time stamping is useful when the timing of a digital signature is critical, such as stock trades, lottery ticket issuance and legal proceedings.
- **Trust Service Provider validation -** A TSP is a person or legal entity that performs validation of a digital signature on a company's behalf and offers signature validation reports.
- **Certificate authority validation**
- **Checksum**
- **Cyclic Redundancy Checking**

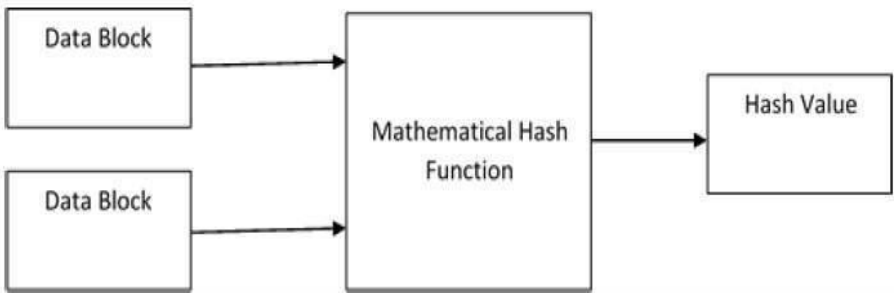
### 4. Hash function

A hash function is a mathematical function that converts a numerical input value into another compressed numerical value. The input to the hash function is of arbitrary length but output is always of fixed length.

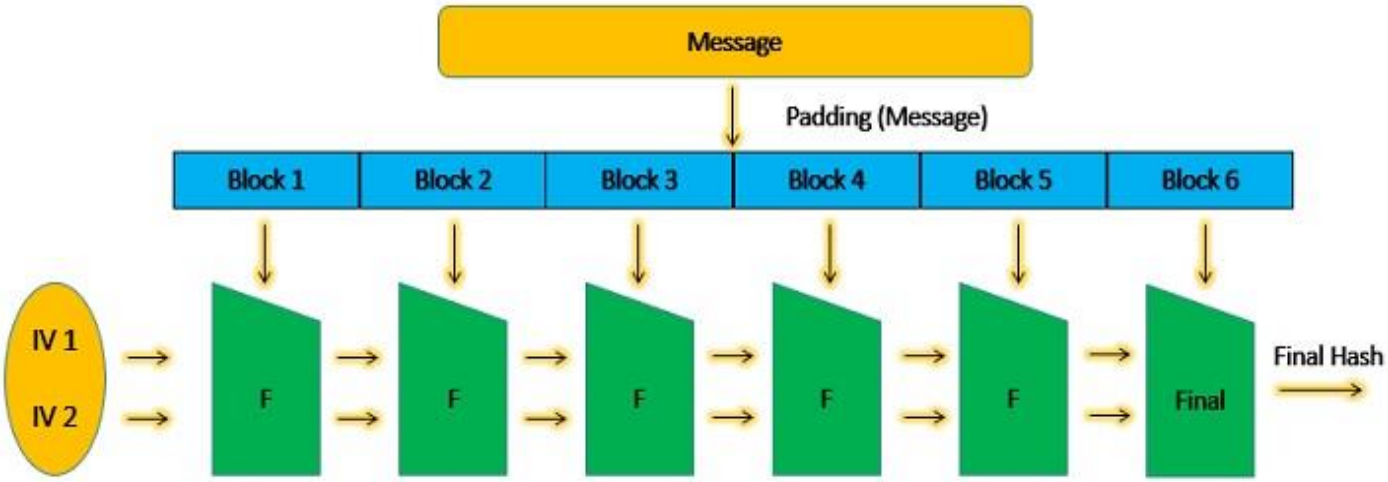
Values returned by a hash function are called message digest or simply hash values. Hash function with n bit output is referred to as an n-bit hash function. The following picture illustrated hash function −




**Design of Hashing Algorithms -** Every Hash algorithm has a set of hash functions. It operates on two fixed-size blocks of data to create a hash code. The size of each data block differs from one algorithm to another. Typically the block sizes are from 128 bits to 512 bits. The following illustration demonstrates hash function −



For example, SHA-1 takes in the message/data in blocks of 512-bit only. So, if the message is exactly of 512-bit length, the hash function runs only once (80 rounds in case of SHA-1). Similarly, if the message is 1024-bit, it's divided into two blocks of 512-bit and the hash function is run twice. However, 99% of the time, the message won't be in the multiples of 512-bit. For such cases (almost all cases), a technique called Padding is used. Using a padding technique, the entire message is divided into fixed-size data blocks. The hash function is repeated as many times as the number of data blocks. This is how it's done:




The output of the first data block is fed as input along with the second data block. Consequently, the output of the second is fed along with the third block and so on. Thus, making the final output the combined value of all the blocks.

**Popular Hash Functions - some popular hash functions are -**

- **Message Digest (MD)** - MD5 was most popular and widely used hash function for quite some years. The MD family comprises of hash functions MD2, MD4, MD5 and MD6.
- **Secure Hash Function (SHA) -** Family of SHA comprise of four SHA algorithms; SHA-0, SHA-1, SHA-2, and SHA-3. Though from same family, there are structurally different.
- **RIPEMD -** The RIPEND is an acronym for RACE Integrity Primitives Evaluation Message Digest. This set of hash functions was designed by open research community and generally known as a family of European hash functions.
- **Whirlpool -** This is a 512-bit hash function. It is derived from the modified version of Advanced Encryption Standard (AES). Three versions of Whirlpool have been released; namely WHIRLPOOL-0, WHIRLPOOL-T, and WHIRLPOOL.

**Applications -** Hash functions are extremely useful and appear in almost all information security applications like message authentication, digital signature, virus detection, intrusion detection etc.

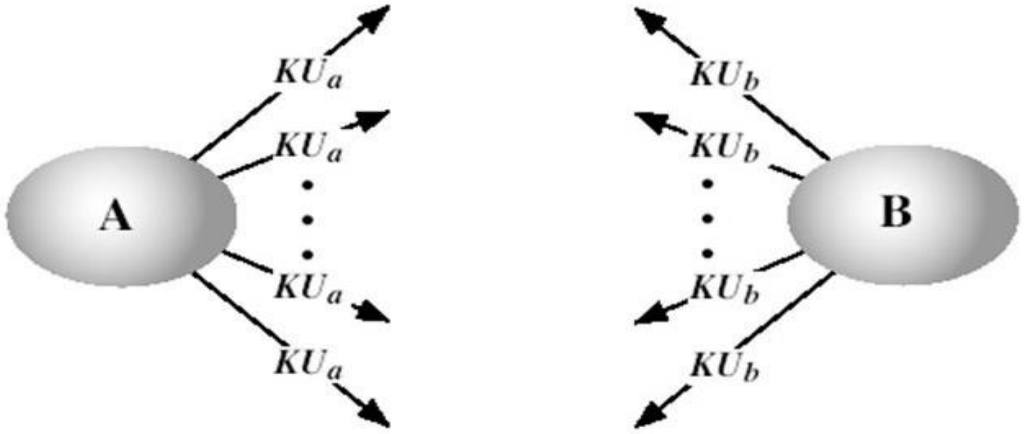
### 5. Public Key Distribution

In public key cryptography, everyone has access to everyone's public key, public keys are available to the public. Several techniques have been proposed for the distribution of public keys. Virtually all of these proposals can be grouped into the following general schemes:

- Public announcement
- Publicly available directory (trusted center)
- Public-key authority (controlled trusted center)
- Public-key certificates

**Public Announcement -** In the public-key encryption, public key is available for all i.e. public. Thus, if there is some broadly accepted public-key algorithm, such as RSA, any participant can send his or her public key to any other

participant or broadcast the key to the community at large. Although this approach is convenient, it has a major weakness. Anyone can forge such a public announcement. That is, some user could pretend to be user A and send a public key to another participant or broadcast such a public key.



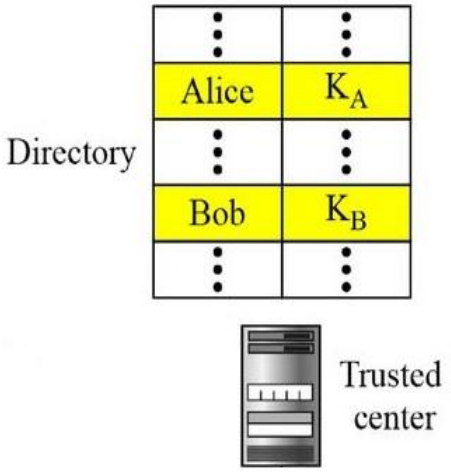

**Publicly available directory (trusted center) -** A more secure approach is to have a trusted center retain directory of public keys. The directory, like the one used in a telephone system, is dynamically updated. Each user can select a private and public key, keep the private key, and deliver the public key for insertion into the directory. The center requires that each user register in the center and prove his/her identity. The directory can be publicly advertised by the trusted center. The center can also respond to any inquiry about the public key. Such a scheme would include the following elements:

**I.** The authority maintains a directory with a {name, public key} entry for each participant.

**II.** Each participant registers a public key with the directory authority.

**III.** A participant may replace the existing key with a new one at any time, either because of the desire to replace a public key that has already been used for a large amount of data, or because the corresponding private key has been com- promised in some way.

**IV.** Participants could also access the directory electronically. For this purpose, secure, authenticated communication from the authority to the participant is mandatory.



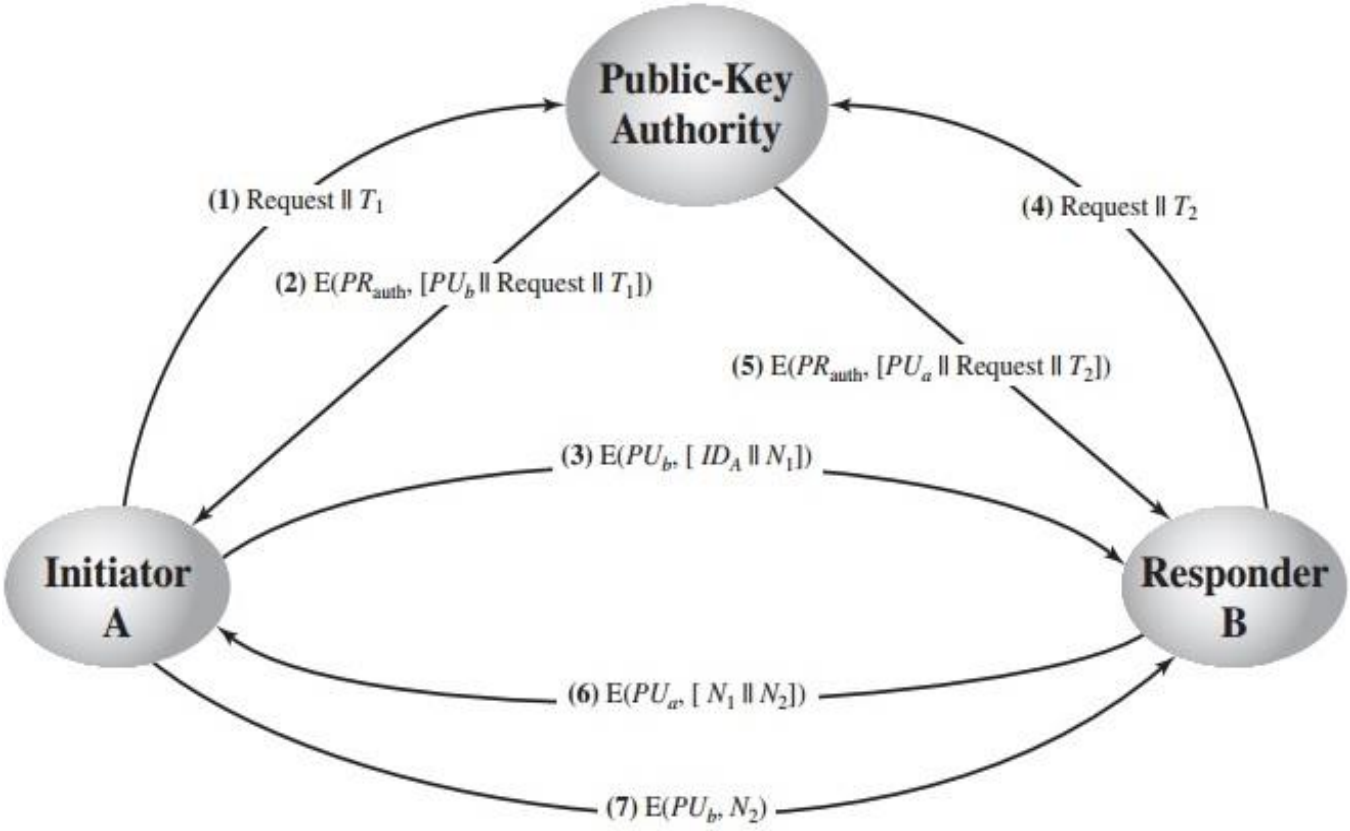

This scheme is clearly more secure than individual public announcements, but still has vulnerabilities. If an adversary succeeds in obtaining or computing the private key of the directory authority, the adversary could authoritatively pass out counterfeit public keys and subsequently impersonate any participant and eavesdrop on messages sent to any participant. Another way to achieve the same end is for the adversary to tamper with the records kept by the authority.

**Public-key authority (controlled trusted center) -** Stronger security for public-key distribution can be achieved by providing tighter control over the distribution of public keys from the directory. As before, the scenario assumes that a central authority maintains a dynamic directory of public keys of all participants. In addition, each participant reliably knows a public key for the authority, with only the authority knowing the corresponding private key. The following steps occur:

1. A sends a time stamped message to the public-key authority containing a request for the current public key of B.
2. The authority responds with a message that is encrypted using the authority's private key, $PR_{auth}$.Thus, A is able to decrypt the message using the authority's public key.Therefore,A is assured that the message originated with the authority.The message includes the following:
   - B's public key, $PU_b$, which A can use to encrypt messages destined for B.

- The original request used to enable A to match this response with the corresponding earlier request and to verify that the original request was not altered before reception by the authority.
- The original timestamp given so A can determine that this is not an old this is not an old message from the authority containing a key other than B's current public key.

**3.** A stores B's public key and also uses it to encrypt a message to B containing an identifier of A ($ID_A$) and a nonce ($N_1$), which is used to identify this transaction uniquely**.**




**4. & 5.** B retrieves A's public key from the authority in the same manner as A retrieved B's public key.

At this point, public keys have been securely delivered to A and B, and they may begin their protected exchange. However, two additional steps are desirable:

**6.** B sends a message to A encrypted with $PU_a$ and containing A's nonce ($N_1$) as well as a new nonce generated by B ($N_2$). Because only B could have decrypted message (3), the presence of $N_1$ in message (6) assures A that the correspondent is B.

**7.** A returns $N_2$, which is encrypted using B's public key, to assure B that its correspondent is A.

**Public Key Certificates -** public-key authority has some drawbacks. It could be somewhat of a bottleneck in the system, for a user must appeal to the authority for a public key for every other user that it wishes to contact. As before, the directory of names and public keys maintained by the authority is vulnerable to tampering. An alternative approach to the public key authority is the use of certificates that can be used by participants to exchange keys without contacting a public-key authority. Each certificate, containing a public key and other information, is created by a certificate authority and is given to the participant. Typically, the third party is a certificate authority, such as a government agency or a financial institution, which is trusted by the user community. A user can present his or her public key to the authority in a secure manner and obtain a certificate. The user can then publish the certificate. Anyone needing this user's public key can obtain the certificate and verify that it is valid by way of the attached trusted signature. A participant can also convey its key information to another by transmitting its certificate. Other participants can verify that the certificate was created by the authority. We can place the following requirements on this scheme:

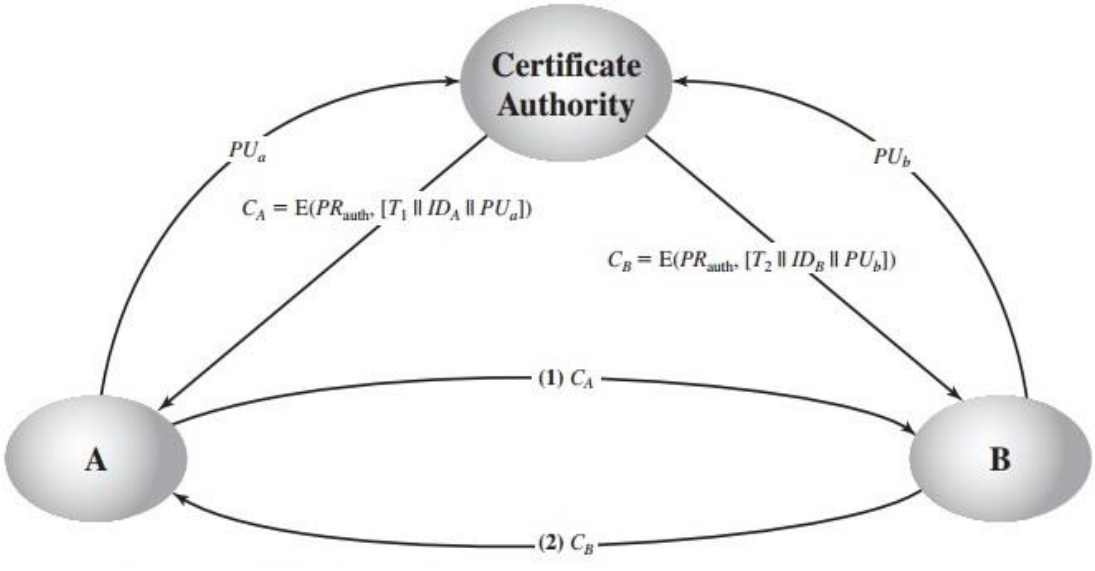
I. Any participant can read a certificate to determine the name and public key of the certificate's owner.

II. Any participant can verify that the certificate originated from the certificate authority and is not counterfeit.

III. Only the certificate authority can create and update certificates.

IV. Any participant can verify the currency of the certificate.

A certificate scheme is illustrated in Figure. Each participant applies to the certificate authority, supplying a public key and requesting a certificate.




An example of this scheme can be seen using the following transaction:

$C_A = \mathrm{E}(PR_{\text{auth}}, [T_1 \,||\, ID_A \,||\, PU_a])$

where $C_A$ is A's certificate, $PR_{\text{auth}}$ is the private key used by the authority $ID_A$ is A's identification and $PU_A$ is A's public key and $T$ is a timestamp.

A can then pass this certificate $C_A$ to any other participant, who reads and verifies the certificate as follows:

$\mathrm{D}(PU_{\text{auth}}, C_A) = \mathrm{D}(PU_{\text{auth}}, \mathrm{E}(PR_{\text{auth}}, [T_1 \,||\, ID_A \,||\, PU_a])) = (T_1 \,||\, ID_A \,||\, PU_a)$

An example of a certification service is the X.509. X.509 certificates are used in most network security applications, including IP security, transport layer security (TLS), and S/MIME.

### 6. Real World Protocols

A cryptographic protocol is a procedure carried out between two parties which is used to perform some security task. We need to discuss several widely used real world security protocols. Next, we look at real protocols:

- IPSec(IP Security) - Security at the IP layer
- TLS - provide privacy and integrity
- SSH - A simple and useful security protocol
- Kerberos - Symmetric key, single sign on etc.

Each has advantages and disadvantages; many of them overlap somewhat in functionality, but each tends to be used in different areas:

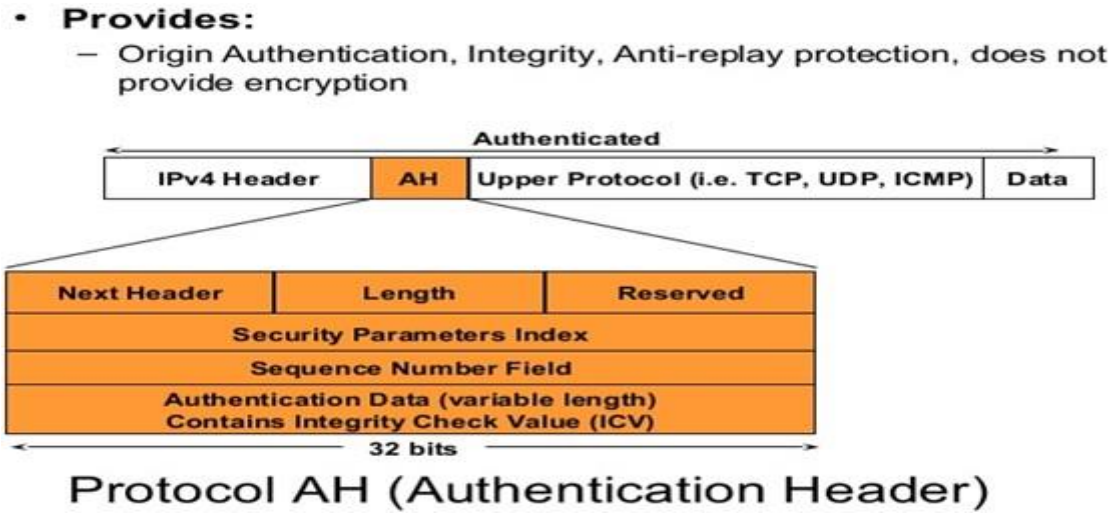
⇒ **IPSec -** The IP security (IPSec) is an Internet Engineering Task Force (IETF) standard suite of protocols between two communication points across the IP network that provide data authentication, integrity, and confidentiality. It also defines the encrypted, decrypted and authenticated packets.

IPSec (Internet Protocol Security) is made up of a number of different security protocols, and designed to ensure data packets (integrity) sent over an IP network remain unseen and inaccessible by third parties. IPSec provides high levels of security for Internet Protocol. Encryption is used to ensure confidentiality, and for authentication.

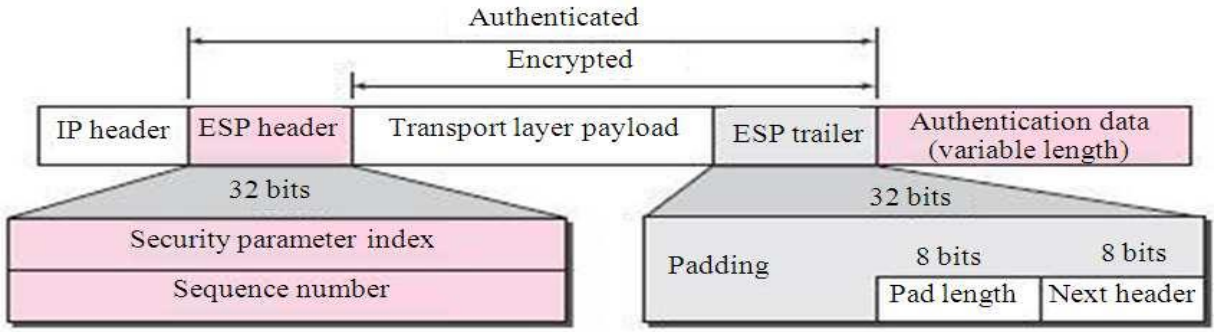
**IPSec Mechanisms -** Most other security protocols function at the application layer of network communication. A major advantage of IPSec is that, because it operates at network rather than application level, it is able to encrypt an

entire IP packet. IPSec is defined for use with both versions of the Internet Protocol, IPv4 and IPv6. There are three protocols/mechanism for imposing security on IP packets or IPSec implementation.

- **Authentication header (AH)** – Authentication Header (AH) is a protocol and part of the Internet Protocol Security (IPsec) protocol suite, which authenticates the origin of IP packets (datagrams) and guarantees the integrity of the data. The AH confirms the originating source of a packet and ensures that its contents (both the header and payload) have not been changed since transmission. AH cannot protect fields that change nondeterministically between sender and receiver. For example, the IP TTL field is not a predictable field and, consequently, not protected by AH.It also provides data integrity, authentication and anti replay and it does not provide encryption. The anti replay protection, protects against unauthorized transmission of packets. It does not protect data's confidentiality.

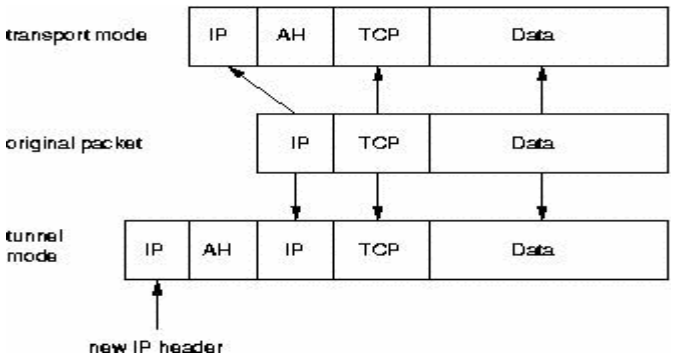



- **Encapsulating Security Payload (ESP)** – An Encapsulating Security Payload (ESP) is a protocol within the IPSec for providing authentication, integrity and confidentially of network packets data/payload in IPv4 and IPv6 networks. ESP provides message/payload encryption and the authentication of a payload and its origin within the IPSec protocol suite. ESP doesn't protect the packet header.




The IPSec standards define two modes of IPSec operation.

- **Tunnel Mode:** In this mode the whole IP packet forms secure communication between two places, or gateways. The IP header is placed in front of the original IP header. A new IP header added in front. The IPSec header, the preserved IP header, and the rest of the packet are treated as the payload.
- **Transport Mode:** In this mode, the IPSec header is added between the IP header and the rest of the packet.



**Advantages**

- As IPSec operates on the network layer, changes only have to be made to the operating system rather than individual applications.
- IPSec is completely invisible in its operation, making it the ideal choice for VPNs.
- Use of AH and ESP guarantees the highest possible levels of security and privacy.

**Disadvantages**

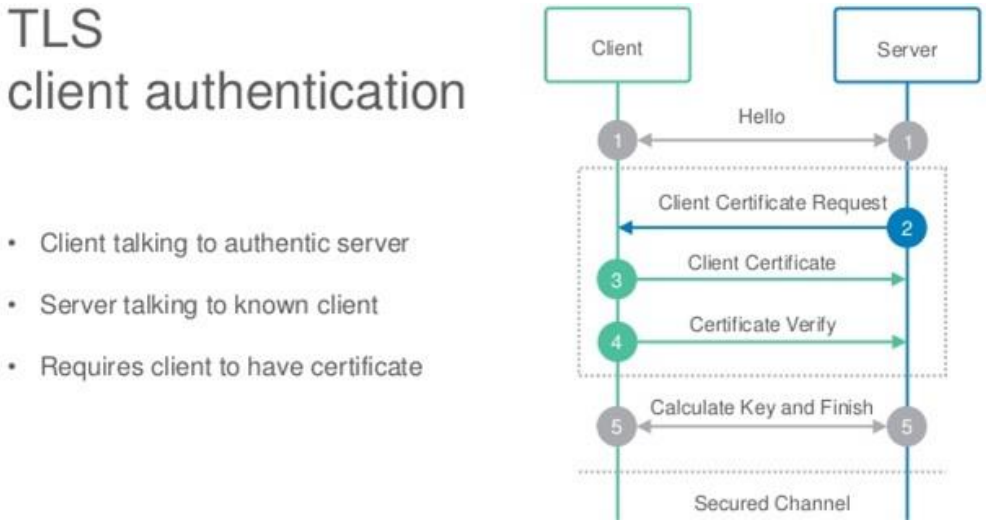
- IPSec is more complicated than alternative security protocols and harder to configure.
- Secure public keys are required for IPSec. If your key is compromised or you have poor key management, you may experience problems.
- For small size packet transmission, IPSec can be an inefficient way to encrypt data.
- IPSec cannot provide the same end-to-end security as systems working at higher levels. IPSec encrypts an IP connection between two machines.
- IPSec does not stop denial of service attacks and traffic analysis.

⇒ **Transport Layer Security -** TLS is a cryptographic protocol (encrypts) that provides end-to-end communications security over networks and is widely used for internet communications and online transactions. It is particularly useful for private and sensitive information such as passwords, credit card numbers, and personal correspondence. It is prevent eavesdropping, tampering and message forgery. However, it can be used for other applications such as e-mail, file transfers, video/audio conferencing, instant messaging and voice-over-IP, as well as Internet services such as DNS and NTP.

**How does SSL/TLS work?**

These are the essential principles to grasp for understanding how SSL/TLS works:

- Secure communication begins with a TLS handshake, in which the two communicating parties open a secure connection and exchange the public key
- During the TLS handshake, the two parties generate session keys, and the session keys encrypt and decrypt all communications after the TLS handshake
- Different session keys are used to encrypt communications in each new session
- TLS ensures that the party on the server side, or the website the user is interacting with, is actually who they claim to be
- TLS also ensures that data has not been altered, since a message authentication code (MAC) is included with transmissions




***(What's the difference between TLS and SSL?***

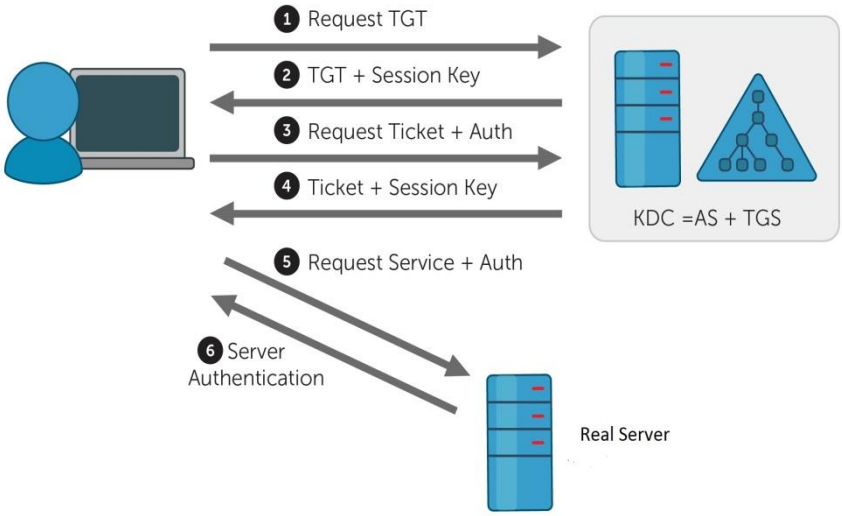
*TLS developed from a previous encryption protocol called SSL, which was developed by Netscape. TLS version 1.0 actually began development as SSL version 3.1, but the name of the protocol was changed before publication in order to indicate that it was no longer associated with Netscape. Because of this history, the terms TLS and SSL are sometimes used interchangeably.)*

⇒ **Kerberos -** Kerberos is a network authentication protocol. It is designed to provide strong authentication for client/server applications by using secret-key cryptography. Kerberos run as a third-party trusted server known as the Key Distribution Center (KDC). The main components of Kerberos are:

- **Client:** Initiates the communication for a service request. Acts on behalf of the user.
- **Authentication Server (AS):** The Authentication Server is the KDC in the Kerberos protocol. Each user registers with the AS and is granted a user identity and a password. The AS has a database with these identities and the corresponding passwords. AS verify the user, issues a session key to be used between client and the Ticket Granting Server (TGS), and send a ticket for the TGS.
- **Ticket Granting Server (TGS):** The TGS issues a ticket for the real server. It also provides the session key between client and server. The users verify her ID just once with the AS and contact the TGS multiple times to obtain tickets for different real servers.
- **Real Server:** It provides services for the client. Kerberos is designed for the client - server model.

**Operation -** Client process can access a process running on the real server in six steps as shown in the fig.

1. This is the initial authentication request. The client requests AS for a Ticket Granting Ticket (TGT). The client ID is sent in the request.
2. Authentication Server verifies user's access right using database and then gives ticket-granting-ticket and session key (SK1). Results are encrypted using Password of user.




3. Decryption of message is done using the password then sends the ticket, name of the real server and time stamp encrypted by secret key.
4. Ticket Granting Server decrypts the ticket send by User and authenticator verifies the request then creates the ticket and session key (SK2) for requesting services from the Server.
5. User sends the Ticket and Authenticator to the Server.
6. Server verifies the Ticket and authenticators then generate the access to the service. After this User can access the services.

Kerberos is built in to all major operating systems, including Microsoft Windows, Apple OS X, FreeBSD and Linux.

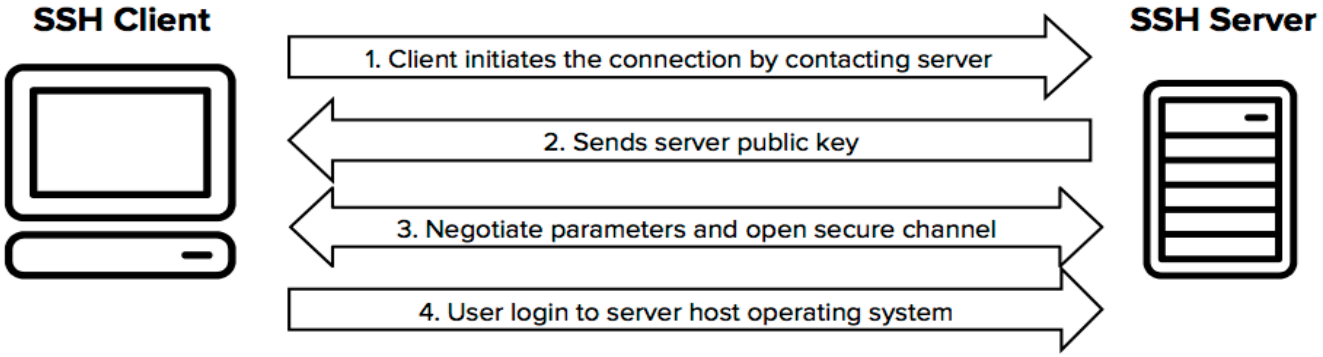
⇒ **Secure Shell (SSH) Protocol** - The SSH protocol uses encryption to secure the connection between a client and a server. It makes possible for a client (a user or even a machine) to open an interactive session on a remote machine (server) to send commands or files over a secure channel:

- The data circulating between the client and the server are encrypted, which guarantees their confidentiality (nobody other than the server and the client can read the information being sent on the network). As a result, it is not possible to monitor the network with a sniffer.
- The client and server authenticate one another in order to make sure the two communicating

machines are indeed those the parties believe them to be. A hacker can no longer take on the identity of the client or server (spoofing).

**How SSH works -** An SSH connection is established in several phases:

- Firstly, the server and client identify one another in order to establish a secure channel (secure transport layer).
- Secondly, the client logs in to the server to obtain a session.




7. **E mail Security Certificate**

An email certificate is a digital file that is installed to your email application to enable secure email communication. These certificates are known by many names — email security certificates, email encryption certificates, S/MIME certificates, etc. S/MIME, which stands for "secure/multipurpose internet mail extension", is a certificate that allows users to digitally sign their email communications as well as encrypt the content and attachments included in them. Not only does this authenticate the identity of the sender to the recipient, but it also protects the integrity of the email data before it is transmitted across the internet.

The way that an email encryption certificate works is by using asymmetric encryption.

Unfortunately, most webmail clients (OWA [Outlook Web App], Gmail, Hotmail, and Yahoo), do not currently support SMIME certificates, but most desktop email clients, including the following, do support email certificates:

- Microsoft Outlook & mail
- Outlook Express
- Mozilla Thunderbird
- Apple Mail
- Netscape Messenger
- Qualcomm Eudora

**Problems with Email certificates -**

- Email certificates aren't normally considered practical for webmail clients because the private key would need to be kept on the server, preventing end-to-end encryption.
- Malware can be sent to in an encrypted email without being stopped by a company gateway.
- The private key of the SMIME certificate could be lost and the messages would not be readable.

**Why Email Signing SSL Certificate?**

- Issued and installed within minutes
- Compatible with every email clients and web browsers
- Compatible with all mobile and desktop devices & Operating systems
- Available within minutes and easily accessible

8. **DNS Security -** DNS security is a process of securing the Domain Name System of your enterprise, protects internet clients from counterfeit DNS data**.** Some most of the common attacks are:
    - DNS spoofing/cache poisoning
    - DNS tunneling

- DNS hijacking
- Phantom domain attack
- Domain lock - up attack

Internet users rely on the DNS to identify the names of websites they want to visit, but browsers communicate with websites via their IP addresses. DNS is important because it links the domain name to the IP. As the Internet has grown, malicious actors have found weaknesses in the DNS system. Internet criminals can exploit these weaknesses and are capable of creating false DNS records. These fake records can trick users into visiting fake websites, downloading malicious software, or worse. Thus, DNS security was created to save the DNS.

**Prevention of DNS attacks**

**DNS firewall -** A DNS firewall is a tool that can provide a number of security and performance services for DNS servers. A DNS firewall sits between a user's recursive resolver and the authoritative name server of the website or service they are trying to reach. The firewall can provide rate limiting services to shut down attackers trying to overwhelm the server.

**DNS as a security tool -** DNS resolvers can also be configured to provide security solutions for their end users (people browsing the Internet). Some DNS resolvers provide features such as content filtering, which can block sites known to distribute malware and spam, and botnet protection, which blocks communication with known botnets. Many of these secured DNS resolvers are free to use and a user can switch to one of these recursive DNS services by changing a single setting in their local router.

**Update your DNS servers regularly**

**Buy DDOS protection services**

**UNIT-5**

**Internet Infrastructure:**

1. Basic security problems
2. Routing security
3. DNS revisited
4. Summary of weaknesses of internet security
5. Link layer connectivity and TCP IP connectivity
6. Packet filtering firewall
7. Intrusion detection

**INTERNET INFRASTRUCTURE:**

Generally speaking, infrastructures are the frameworks or architectures that systems are made of. For example, a nation's transportation infrastructure consists of roadways, railroads, airports, ocean ports, and rivers.

The Internet also has an infrastructure consisting of many different elements, each of which plays a critical role in the delivery of information/data from one point to another.

**ELEMENTS OF THE INTERNET INFRASTRUCTURE**

At the most rudimentary (basic) levels of the Internet infrastructure are endless miles of telephone lines and fiber optic cable. These cables connect millions of individual users and businesses to other parties, transmitting data at varying speeds, depending on the types of cabling used.

Here we discuss three important internet infrastructure components are:

- **TCP/IP-**Used for routing & messaging
- **BGP (Border Gateway Protocol) –**Used for routing announcement
- **DNS (Domain Name System):** Translating a host name like google.com into network address that can be used to actually connect to the host.
-

- **TCP/IP (Transmission Control Protocol/Internet Protocol)**

The Internet works by using a protocol called TCP/IP, or Transmission Control Protocol/Internet Protocol. TCP/IP is the underlying communication language of the Internet. In base terms, TCP/IP allows one computer to talk to another computer via the Internet through compiling packets of data and sending them to right location.

**Defining TCP**

There are two layers to TCP/IP. The top layer, TCP, is responsible for taking large amounts of data, compiling it into packets and sending them on their way to be received by a fellow TCP layer, which turns the packets into useful information/data.

**Defining IP**

The bottom layer, IP, is the location aspect of the pair allowing the packets of information to be sent and received to the correct location. If you think about IP in terms of a map, the IP layer serves as the packet GPS to find the correct destination. Much like a car driving on a highway, each packet passes through a gateway computer (signs on the road), which serve to forward the packets to the right destination.

***"In summary, TCP is the data. IP is the Internet location GPS."***

The Four Layers Embedded in TCP/IP

The four abstraction layers are the link layer (lowest layer), the Internet layer, the transport layer and the application layer (top layer).

The Link Layer is the physical network equipment used to interconnect nodes and servers.

The Internet Layer connects hosts to one another across networks.

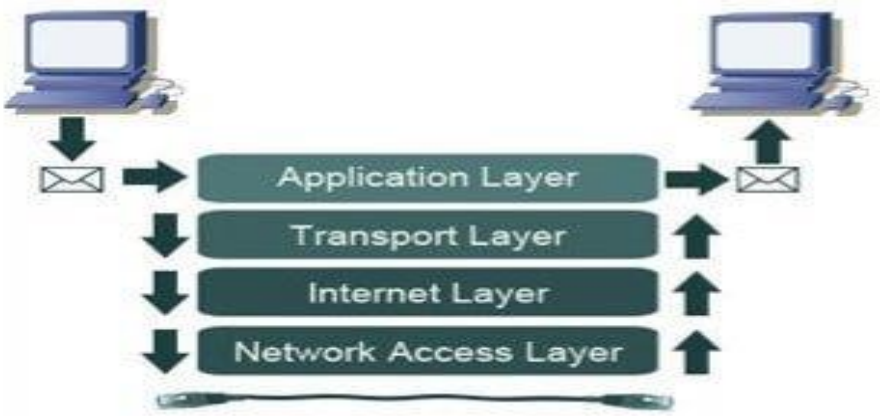
The Transport Layer resolves all host-to-host communication.

The Application Layer is utilized to ensure communication between applications on a network.

In English, the four abstraction layers embedded in TCP/IP allow packets of data, application programs and physical network equipment to communicate with one another over the Internet to ensure packets are sent intact and to the correct location.

They work in the following fashion:

In English, the four abstraction layers embedded in TCP/IP allow packets of data, application programs and physical network equipment to communicate with one another over the Internet to ensure packets are sent intact and to the correct location.



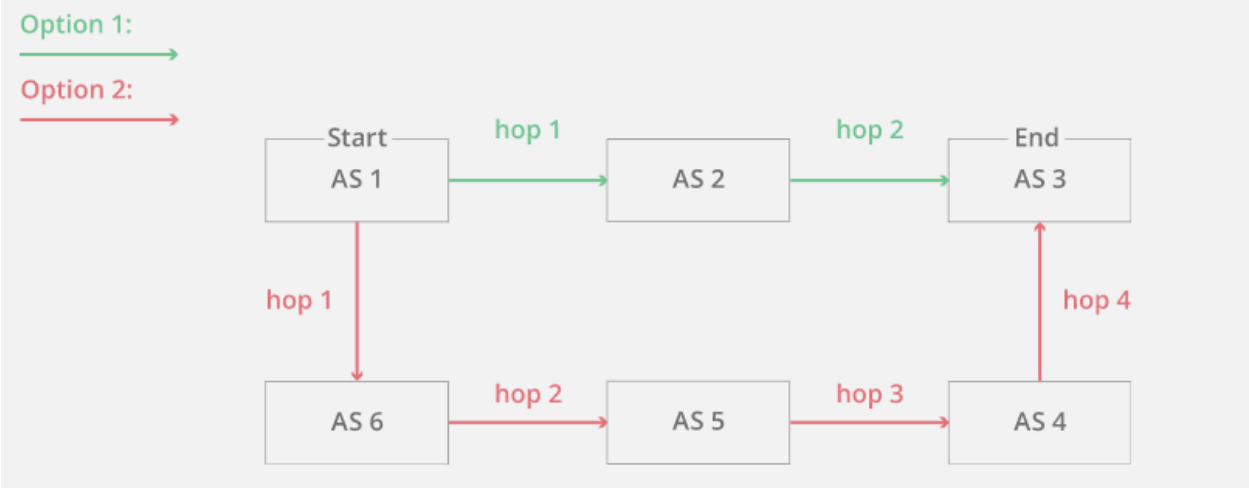

- **What is BGP?**

Border Gateway Protocol (BGP) is the postal service of the Internet. When someone drops a letter into a mailbox, the postal service processes that piece of mail and chooses a fast, efficient route to deliver that letter to its recipient. Similarly, when someone submits data across the Internet, BGP is responsible for looking at all of the available paths that data could travel and picking the best route, which usually means hopping between autonomous systems.

*What is an autonomous system?*

*The Internet is a network of networks; it's broken up into hundreds of thousands of smaller networks known as autonomous systems (AS). Each of these networks is essentially a large pool of routers run by a single organization. For example if we send message from our mobile to other mobile then it follow different -2 network from source to destination.*

If we continue to think of BGP as the postal service of the Internet, AS's are like individual post office branches. A town may have hundreds of mailboxes, but the mail in those boxes must go through the local postal branch before being routed to another destination. The internal routers within an AS are like mailboxes, they forward their outbound transmissions to the AS, which then uses BGP routing to get these transmissions to their destinations.

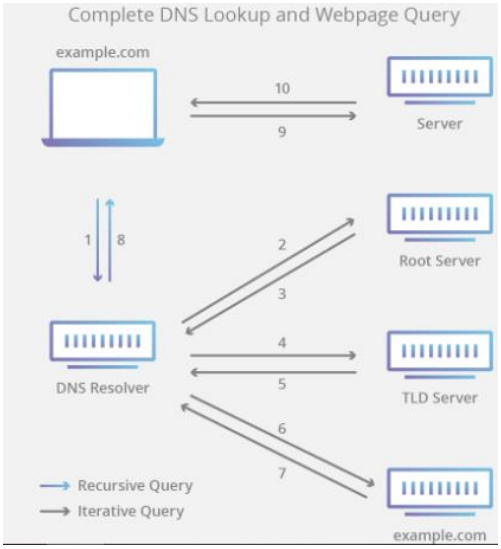



In the above figure starting node is AS1 & end Node is AS3, there are two paths possible BGP decide which path must be follow for message transmission.

- **DNS (Domain Name System)**

The Domain Name System (DNS) is the phonebook of the Internet. Humans access information online through domain names, like nytimes.com or espn.com. Web browsers interact through Internet Protocol (IP) addresses. DNS translates domain names to IP addresses so browsers can load Internet resources.

Each device connected to the Internet has a unique IP address which other machines use to find the device. DNS servers eliminate the need for humans to memorize IP addresses such as 192.168.1.1 (in IPv4), or more complex newer alphanumeric IP addresses such as 2400:cb00:2048:1:: c629:d7a2 (in IPv6).




The 10 steps in a DNS lookup:

A user types ‘example.com’ into a web browser and the query travels into the Internet and is received by a DNS recursive resolver.

i. The resolver then queries a DNS root nameserver (.).

ii. The root server then responds to the resolver with the address of a Top Level Domain (TLD) DNS server (such as .com or .net), which stores the information for its domains. When searching for example.com, our request is pointed toward the .com TLD.

iii. The resolver then makes a request to the .com TLD.

iv. The TLD server then responds with the IP address of the domain's nameserver, example.com.

v. Lastly, the recursive resolver sends a query to the domain's nameserver.

vi. The IP address for example.com is then returned to the resolver from the nameserver.

vii. The DNS resolver then responds to the web browser with the IP address of the domain requested initially.

viii. Once the 8 steps of the DNS lookup have returned the IP address for example.com, the browser is able to make the request for the web page:

ix. The browser makes a HTTP request to the IP address.

x. The server at that IP returns the webpage to be rendered in the browser (step 10).

## 1. BASIC SECURITY PROBLEMS

**i. Network packets pass through the untrusted host**(Because they follow different-2 network during transmission of message from source to destination )

- Eavesdropping, Packet sniffing
- Especially easy when attacker control a machine(computer) closed to victim(Wi-Fi Router)

**ii. TCP state easily obtained by Eavesdropping**

- Enable spoofing and session hijacking

**iii. Distributed Denial of Services(DDoS) Vulnerability**

- A Distributed Denial of Service (DDoS) attack generally involves a group of computers being harnessed together by a hacker to flood the target with traffic. A NETSCAPE Arbor report suggested there were 7.5 million DDoS attacks in 2017, so while many target IT service providers, they are still more prevalent than many people realize. One of the most worrying aspects of DDoS attacks for businesses is that without even being targeted, the business can be affected just by using the same server, service provider, or even network infrastructure.

## 2. ROUTING SECURITY

Routing is fundamental to how the Internet works. Routing protocols direct the movement of packets between your computer and any other computers it is communicating with. The Internet's routing protocol (Border Gateway Protocol or BGP) is considered as very sensitive for attacking. These problems can literally knock entire networks off the Internet or divert traffic to an unintended party.

Routing security has received varying levels of attention over the past several years and has recently begun to attract more attention specifically around Border Gateway Protocol (BGP) on the public Internet. Despite this new attention, however, the area most open to attack is often not the Internet's BGP tables but the routing systems within your own enterprise network. Because of some of the sniffing-based

attacks, an enterprise routing infrastructure can easily be attacked with man-in-the-middle and other attacks designed to corrupt or change the routing tables with the following results:

- **Traffic redirection**—In this attack, the adversary is able to redirect traffic, enabling the attacker to modify traffic in transit or simply sniff packets.
- **Traffic sent to a routing black hole**—Here the attacker is able to send specific routes to null0, effectively kicking IP addresses off of the network.
- **Router denial-of-service (DoS)**—Attacking the routing process can result in a crash of the router or a severe degradation of service.
- **Routing protocol DoS**—Similar to the attack previously described against a whole router, a routing protocol attack could be launched to stop the routing process from functioning properly.
- **Unauthorized route prefix origination**—This attack aims to introduce a new prefix into the route table that shouldn't be there. The attacker might do this to get a covert attack network to be routable throughout the victim network.

**3. DNS REVISITED**

The Domain Name System resolves the names of internet sites with their underlying IP addresses adding efficiency and even security in the process.

At its most basic, DNS is a directory of names that match with numbers. The numbers, in this case are IP addresses, which computers use to communicate with each other.

When the internet was very, very small, it was easier for people to correspond specific IP addresses with specific computers, but that didn't last for long as more devices and people joined the growing network. In addition to creating a directory for all of these devices, words were used to let people connect to different sites; for most people, remembering words is easier than remembering specific sets of numbers. It is still possible to type in a specific IP address into a browser to reach a website.

**How DNS adds efficiency**

DNS is organized in a hierarchy that helps keep things running quickly and smoothly. To illustrate, let's pretend that you wanted to visit networkworld.com.

The initial request for the IP address is made to a recursive resolver, a server that is usually operated by an ISP or other third-party provider. The recursive resolver knows which other DNS servers it needs to ask to resolve the name of a site (networkworld.com) with its IP address. This search leads to a root server, which knows all the information about top-level domains, such as .com, .net, .org and all of those country domains like .cn (China) and .uk (United Kingdom). Root servers are located all around the world, so the system usually directs you to the closest one geographically.

Once the request reaches the correct root server, it goes to a top-level domain (TLD) name server, which stores the information for the second-level domain, the words used before you get to the .com, ,org, .net (for example, that information for networkworld.com is "network world"). The request then goes to the Domain Name Server,

which holds the information about the site and its IP address. Once the IP address is discovered, it is sent back to the client, which can now use it to visit the website. All of this takes mere milliseconds.

**DNS reflection attacks**

DNS reflection attacks can swamp victims with high-volume messages from DNS resolver servers. Attackers request large DNS files from all the open DNS resolvers they can find and do so using the spoofed IP address of the victim. When the resolvers respond, the victim receives a flood of unrequested DNS data that over whelms their machines.

**DNS cache poisoning**

DNS cache poisoning can divert users to malicious Web sites, Attackers manage to insert false address records into the DNS so when a potential victim requests an address resolution for one of the poisoned sites, the DNS responds with the IP address for a different site, one controlled by the attacker. Once on these phony sites, victims may be tricked into giving up passwords or suffer malware downloads.

**DNS resource exhaustion**

DNS resource exhaustion attacks can clog the DNS infrastructure of ISPs, blocking the ISP's customers from reaching sites on the internet. This can be done by attackers registering a domain name and using the victim's name server as the domain's authoritative server. So if a recursive resolver can't supply the IP address associated with the site name, it will ask the name server of the victim Attackers generate large numbers of requests for their domain and toss in non-existent sub domains to boot, which toads to a torrent of resolution requests fired at the victim's name server, overwhelming it.

**What is DNS Cache Poisoning?**

DNS cache poisoning, also known as DNS spoofing, is a type of attack that exploits vulnerabilities in the domain name system (DNS) to divert Internet away from legitimate servers and towards fake ones.

One of the reasons DNS poisoning is so dangerous is because it can spread from DNS server to DNS server. In 2010, a DNS poisoning event resulted in the Great Firewall of China temporarily escaping China's national borders, censoring the Internet in the USA until the problem was fixed .

**DNS rebinding Attack**

DNS rebinding is a method of manipulating resolution of domain names that is commonly used as a form of computer attack. In this attack, a malicious web page causes visitors to run a clientside script that attacks machines elsewhere on the network. In theory, the same-origin policy prevents this from happening: client-side scripts are only allowed to access content on the same host that served the script. Comparing domain names is an essential part of enforcing this policy, so DNS rebinding circumvents this protection by abusing the Domain Name System (DNS).

This attack can be used to breach a private network by causing the victim's web browser to access computers at private IP addresses and return the results to the attacker It can also be employed to use the victim machine for spamming, distributed denial-of-service attacks, or other malicious activities

**How DNS rebinding works**

The attacker registers a domain (such as attacker.com) and delegates it to a DNS server that is under the attacker's control. The server is configured to respond with a very short time to live (TTL) record, preventing the DNS response from being cached. When the victim browses to the malicious domain, the attacker's DNS server first responds with the IP address of a server hosting the malicious client-side code. For instance, they could point the victim's browser to a website that contains malicious JavaScript or Flash scripts that are intended to execute on the victim's computer.

The malicious client-side code makes additional accesses to the original domain name (such as attacker.com). These are permitted by the same-origin policy. However, when the victim's browser runs the script it makes a new DNS request for the domain, and the attacker replies with a new IP address. For instance, they could reply with an internal IP address or the IP address of a target somewhere else on the Internet Protection

**4. SUMMARY OF WEAKNESSES OF INTERNET SECURITY**

i. **Unauthenticated protocols**

When protocol lacks authentication, any computer on the network can send commands that alter the physical process. This may lead to incorrect process operation, which damages goods, destroys plant equipment, harms personnel, or degrades the environment.

**ii. Outdated hardware**

Hardware can be operational for decades. This hardware may operate too simplistically or lack the processing power and memory to handle the threat environment presented by modern network technology.

**iii. Weak user authentication**

User authentication weaknesses in legacy control systems often include hard-coded passwords, easily cracked passwords, passwords stored in easily recoverable formats, and passwords sent in clear text. An attacker who obtains these passwords can often interact with the controlled process at will, the report said.

**iv. Weak file integrity checks**

Lack of software signing that confirms the software author and guarantee that the code has not been altered or corrupted allows attackers to mislead users into installing software that did not originate from the vendor. It also allows attackers to replace legitimate files with malicious ones.

**v. Vulnerable Windows operating systems**

Industrial systems often run unpatched Microsoft Windows operating systems, leaving them exposed to known vulnerabilities.

**The Weaknesses of the TCP/IP model are**

1. It is not generic in nature. So, it fails to represent any protocol stack other than theTCP/IP suite. For example, it cannot describe the Bluetooth connection.
2. It does not clearly separate the concepts of services, interfaces, and protocols. So, it is not suitable to describe new technologies in new networks.

3. It does not distinguish between the data link and the physical layers, which has very different functionalities. The data link layer should concern with the transmission of frames. On the other hand, the physical layer should lay down the physical characteristics of transmission. A proper model should segregate the two layers.
4. It was originally designed and implemented for wide area networks. It is not optimized for small networks like LAN (local area network) and PAN (personal area network).
5. Among its suite of protocols, TCP and IP were carefully designed and well implemented. Some of the other protocols were developed ad hoc and so proved to be unsuitable in long run. However, due to the popularity of the model, these protocols are being used even 30 40 years after their introduction.

## 5. LINK LAYER CONNECTIVITY AND TCP IP CONNECTIVITY

**Link Layer**

- The link layer is responsible for transporting information from one host (or router) to another over a *single* link
- Each network-layer datagram is encapsulated in a link-layer *frame*
- Two fundamentally different types of link-layer channels:

  *a) Broadcast* channels

  - common in local area networks (LANs), wireless LANs, etc.
  - many hosts connected to the same communications channel
  - *medium access protocol* is needed to coordinate transmissions

  *b)Point-to-point* communications link

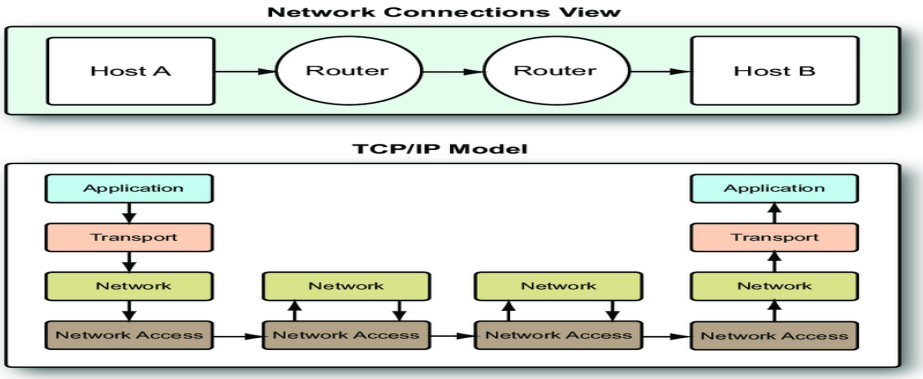
  - used between two routers or home dial-up modem and ISP router
  - coordination is trivial
  - still issues around framing, reliable transfer etc.

**Link Layer Services**

- *framing*: encapsulation of network datagram within a link-layer frame
- *link access*: a medium access (MAC) protocol specifies the rules by which a frame is transmitted onto the link
- *reliable delivery*: useful for links prone to high error rates; avoids cost of end-to-end retransmission at transport or application layer
- *flow control*: frames can be lost if buffering capacity is exceeded
- *error detection*: usually more sophisticated than Internet checksum and implemented in hardware
- *error correction*: possible to correct errors as well as detect them

**TCP/IP model**

- The TCP/IP model consists of five layers: the application layer, transport layer, network layer, data link layer and physical layer.
- The first four layers provide physical standards, network interface, internetworking, and transport functions that correspond to the first four layers of the OSI model and these four layers are represented in TCP/IP model by a single layer called the application layer.
- TCP/IP is a hierarchical protocol made up of interactive modules, and each of them provides specific functionality.




**Transmission Control Protocol (TCP):**It provides a full transport layer services to applications.

It creates a virtual circuit between the sender and receiver, and it is active for the duration of the transmission.

TCP is a reliable protocol as it detects the error and retransmits the damaged frames. Therefore, it ensures all the segments must be received and acknowledged before the transmission is considered to be completed and a virtual circuit is discarded.

At the sending end, TCP divides the whole message into smaller units known as segment, and each segment contains a sequence number which is required for reordering the frames to form an original message.At the receiving end, TCP collects all the segments and reorders them based on sequence numbers.

**IP Protocol:** IP protocol is used in this layer, and it is the most significant part of the entire TCP/IP suite. Following are the responsibilities of this protocol:

- **IP Addressing:** This protocol implements logical host addresses known as IP addresses. The IP addresses are used by the internet and higher layers to identify the device and to provide internetwork routing.
- **Host-to-host communication:** It determines the path through which the data is to be transmitted.
- **Data Encapsulation and Formatting:** An IP protocol accepts the data from the transport layer protocol. An IP protocol ensures that the data is sent and received securely, it encapsulates the data into message known as IP datagram.
- **Fragmentation and Reassembly:** The limit imposed on the size of the IP datagram by data link layer protocol is known as Maximum Transmission unit (MTU). If the size of IP datagram is greater than the MTU unit, then the IP protocol splits the datagram into smaller units so that they can travel over the

local network. Fragmentation can be done by the sender or intermediate router. At the receiver side, all the fragments are reassembled to form an original message.

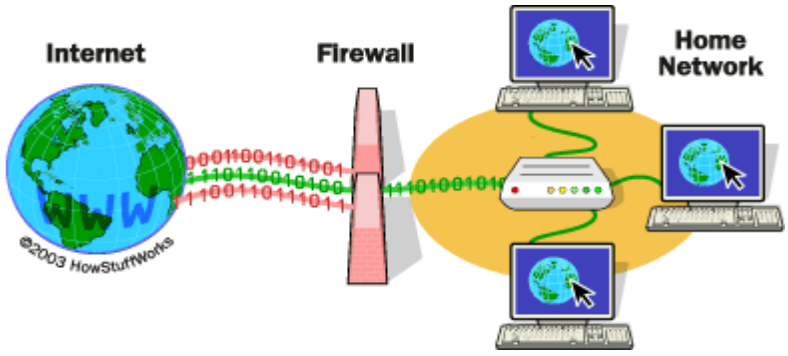
- **Routing:** When IP datagram is sent over the same local network such as LAN, MAN, WAN, it is known as direct delivery. When source and destination are on the distant network, then the IP datagram is sent indirectly. This can be accomplished by routing the IP datagram through various devices such as routers.

## 6. PACKET FILTERING FIREWALL

**Introduction of Firewall**: A firewall is a network security device, which isolates organization's internal network from larger outside network/Internet. It can be a hardware or software-based or combination of both ,which monitors all incoming and outgoing traffic based on a predefined set of security rules it accepts, rejects that specific traffic.

*"A **firewall** is a network security system that monitors and controls incoming and outgoing network traffic based on predetermined security rules. A **firewall** typically establishes a barrier between a trusted internal network and untrusted external network, such as the Internet."*

A firewall establishes a barrier between secured internal networks and outside untrusted network, such as the Internet.




**Types of Firewall -** There are three basic types of firewalls that are used by companies to protect their data & devices to keep destructive elements out of network, viz. Proxy Server, Stateful Inspection and Packet Filters Firewalls**.**

- **Proxy firewall:** Also called the application level gateways. Proxy Server Firewalls are the most secured type of firewalls that effectively protect the network resources by filtering messages at the application layer. It works only for the protocols which are configured such as HTTP and FTP. Application level firewalls can also be configured as Caching Servers which in turn increase the network performance and makes it easier to log traffic.
- **Stateful inspection firewall:** Stateful inspection, also known as dynamic packet filtering. Stateful inspection firewalls are often thought of as a "traditional" firewall as it allows or blocks traffic based on state, port, and protocol. These firewalls work to monitor all activity from the moment a connection is opened until it's fully closed.
- **Packet Filtering Firewall**: As the most "basic" and oldest type of firewall architecture, packet-filtering firewalls basically create a checkpoint at a traffic router or switch. Packet filtering firewall is used to control network access by monitoring outgoing and incoming packet and allowing them to pass or stop based on source and destination IP address, protocols and ports. It analyses traffic at the transport

protocol layer (but mainly uses first 3 layers). Filtering rules are based on information contained in a network packet like source IP address, destination IP address etc.

Packet firewalls treat each packet in isolation. They have no ability to tell whether a packet is part of an existing stream of traffic. Only it can allow or deny the packets based on unique packet headers.

| | Source IP | Dest. IP | Source Port | Dest. Port | Action |
|---|---|---|---|---|---|
| 1 | 192.168.21.0 | -- | -- | -- | deny |
| 2 | -- | -- | -- | 23 | deny |
| 3 | -- | 192.168.21.3 | -- | -- | deny |
| 4 | -- | 192.168.21.0 | -- | >1023 | Allow |

Sample Packet Filter Firewall Rule

i. Incoming packets from network 192.168.21.0 are blocked.
ii. Incoming packets destined for internal TELNET server (port 23) are blocked.
iii. Incoming packets destined for host 192.168.21.3 are blocked.
iv. All well-known services to the network 192.168.21.0 are allowed.

## 7. INTRUSION DETECTION

**Intrusion:** It is any unauthorized access to the system. Intrusion is the one that try to intrude into the privacy of the network

**Types of Intrusion:**

i. **Masquerader Intrusion:** User with authority to use the system but penetrate the security as the legitimate user.
ii. **Misfeasor Intrusion:** Legitimate user with no permission to access the application but misuse the privileges.
iii. **Clandestine Intrusion:** It may be internal or external. They try to steal & use the credential of their supervisor.

**Intrusion Detection:**

A. **Statistical Anomaly Detection:** Behavior of user is analyzed over a period of time & rules are created to differentiate between legitimate and illegal user.
   a) **Threshold Based Detection:** Certain threshold is defined for each user, if that threshold is cross it considers as intrusion.
   b) **Profile Based detection:** Profile is created for each user & they are match for any illegal activity.

   For example Ram often uses 2 hours of internet (**Threshold**) & he visited educational & engineering website (**Profile Based**).

   *Let us suppose that Mohan Steal the identity of Ram & use 4 hours of internet (here it is detected because Ram uses only 2 hours of internet)-**THRESHOLD BASED DETECTION***

   *Let us suppose that Mohan Steal the identity of Ram & visited other than educational & engineering website (here it is detected because Ram uses only educational & engineering website)-**PROFILE BASED DETECTION***

**B. Rule Based Detection**

**a) Anomaly Based Detection:** Here we detect the unauthorized use who break the rule

**Rule:** User uses only 3 hours of internet

*If someone is use the internet more than three hours they will be in category of anomaly detection.*

b) **Penetration Identification Based Detection:** Here we use Expert intelligence system for monitoring the data packet over the network on the behalf of that it decide which one is legitimate user and which one is bad user.

**IDS (Intrusion Detection Systems):** An intrusion detection system (IDS) examines system or network activity to find possible Intrusions or attacks. Intrusion detection systems are either network-based or host-based; vendors are only beginning to integrate the two technologies.

**Types of IDS**

**Network-based IDS:** A network-based IDS usually consists of a network appliance (or sensor) with a Network Interface Card (NIC) operating in promiscuous mode and a separate management interface.

**Host-based IDS**: A host-based IDS requires small programs (or agents) to be installed on individual systems to be monitored. The agents monitor the operating system and write data to log files and/or

trigger alarms.

A host-based ID can only monitor the individual host systems on which Knowledge-based and behavior-based IDS A knowledge-based (or signature-based) IDS references a database of previous attack profiles and known system vulnerabilities to identify active intrusion attempts.

Knowledge-based IDS is currently more common than behavior-based IDS.

**Advantages of knowledge-based systems include the following:**

- It has lower false alarm rates than behavior-based IDS.
- Alarms are more standardized and more easily understood than behavior-based IDS.

**Disadvantages of knowledge-based systems include these:**

- Signature database must be continually updated and maintained.
- New, unique, or original attacks may not be detected or may be improperly classified.

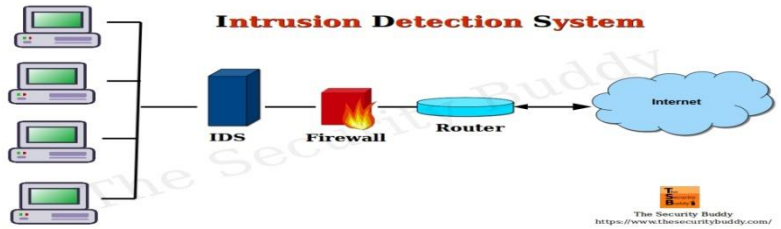
A behavior-based (or statistical anomaly based) IDS references a baseline or learned pattern of normal system activity to identify active intrusion attempts. Deviations from this baseline or pattern cause an alarm to be triggered.

**Advantages of behavior-based systems include that they:**

- Dynamically adapt to new, unique, or original attacks.
- Are less dependent on identifying specific operating system vulnerabilities.

**Disadvantages of behavior-based systems include**

- Higher false alarm rates than knowledge-based IDSes.
- Usage patterns that may change often and may not be static enough to implement an effective behavior-based IDS




**Categories of IDS**

- **Signature-Based IDS:** This IDS verifies signatures of data packets in the network traffic. Basically, it finds the data packets and uses their signatures to confirm whether they are a threat or not. Such signatures are commonly known for intrusion-related signatures or anomalies related to internet protocol. Intruders such as computer viruses, etc, always have a signature, therefore, it can be easily detected by software IDS. As it uses signatures to identify the threats.
- **Anomaly Based IDS**: This IDS usually detects if a data packet behaves anomaly. It issues an alert if packet anomalies are present in protocol header parts. This system produces better results in some cases than signature-based IDS. Normally such IDS captures data from the network and on these packets, it then applies the rules to it in order to detect anomalies.

**Types of IDS**

- **NIDS:**NIDS stand for Network Intrusion Detection System. These types of IDS will capture data packets that were received and sent in the network and tally such packets from the database of signatures. if the packet is a match then no alert will be issued otherwise it will issue an alert letting everyone know of a malicious attack. Snort is an excellent example of a NIDS.
- **HIDS:**HIDS stands for Host Intrusion Detection System which, obviously, acts as a host. Such types of IDS monitor system and application logs to detect intruder activity. Some IDS reacts when some malicious activity takes place, others monitor all the traffics coming to the host where IDS is installed and give alerts in real time.