Computer Security All In One

Computer security Introduction

Computer security is the process of preventing and detecting unauthorized use of your computer. Privacy is the process of protecting he's or her own personal files against any intrusion. Prevention measures help you to stop unauthorized users (also known as "intruders") from accessing any part of your computer system. Detection helps you to determine whether or not someone attempted to break in to your system , if they were successful and what they may have done.

Data security is the practice of keeping data protected from corruption and unauthorized access. The focus behind data security is to ensure privacy while protecting personal or corporate data. Information Technology Security is the process of protecting computer networks , programs and data from unintended or unauthorized access , change or destruction.

Why do we care about computer security ? Our modern ways of communication provide a lot of examples of critical situations involving security issues. To illustrate , communication by phone , be email or by fax , getting connected to a bank via the internet and performing transactions , digital payment systems , e-voting systems , this all demands confidentiality and integrity of exchanged information.

Security Goals :- Confidentiality , Integrity , Availability

Confidentiality (secrecy or privacy) :- It ensures that computer related assets are accessed only by authorized parties. Confidentiality is sometimes called secrecy or privacy , only authorized entities are allowed to view , only sender and intended receiver should understand message contents.

Integrity :- Information needs to be changed constantly. Integrity means that changes need to be done only by authorized entities and through authorized mechanisms or assets can be modified only by authorized parties or only in authorized ways. Ensures the message was not altered by unauthorized individuals.

Availability :- It means that assets are accessible to authorized parties at appropriate times. The information created and stored by an organization needs to be constantly changed , which means it must be accessible to authorized entities. It assures that system works promptly and service is not denied for authorized users.

Vulnerability :- is a weakness In the security system. Weakness can appear in any element of a computer , both in the hardware , operating system and the software.

Hardware Vulnerabilities :- Hardware is more visible than software. It is rather simple to attack by adding devices , changing them , removing them , intercepting the traffic to them , or flooding them with traffic until they can no longer function. Computers have been drenched with water , burned , frozen , gassed and electrocuted with power surges.

Software Vulnerability :- Software can be replaced , changed or destroyed maliciously , or it can be modified , deleted or misplaced accidentally. Whether intentional or not , these attacks exploit the software's vulnerabilities.

Sometimes , the attacks are obvious , as when the software no longer runs. More subtle are attacks in which the software has been altered but seems to run normally.

Data vulnerability :- A data attack is more widespread and serious problem than either a hardware or a software attack. A data items have greater public value than hardware and software because more people know how to use or interpret data.

Policies and mechanisms :- Policy is a statement of what is, and what is not allowed by users of a system. Mechanisms is a method , tool or procedure for enforcing a security policy.

Security controls :- controls or counter measures that attempt to prevent exploiting a computing system's vulnerabilities.

A) Authentication :- is a process of binding an identity to a subject. Validates the source of a message , to ensure the sender is properly identified. Sender , receiver want to confirm the identity of each other.

Authentication in computer security is the process of verifying the identity of a user or a device before granting access to a system or resource. The purpose of authentication is to ensure that only authorized individuals or devices are granted access to sensitive information , applications or systems.

Authentication typically involves the user of credentials and password , a smart card or biometric factor like a finger print , that are verified by the system before granting access. This process can be performed locally on the device or through a remote authentication server.

B) Encryption

C) Auditing

D) Standards

CHAPTER TWO

Threats and Attacks :- is a potential violation of security , it is any person , act or object that poses a danger to computer security / privacy. The fact that the violation might occur means that those actions that could cause it to occur must be guarded against (or prepared for) and those actions are called attacks. Those who execute such actions , or cause them to be executed are called attackers.

Types of threats :- The four types of attacks that are commonly referred to as the “4D’s” of security are disclosure , deception , disruption , and usurpation. These categories are used to classify different types of attacks based on their objectives and methods.

1) Disclosure Attacks :- unauthorized access to information (also called snooping or interception) : they are aimed at stealing sensitive information or data.

2) Deception Attacks(spoffing) :- acceptance of false data , this attacks are aimed to misleading or tricking users in to taking actions that benefit the attacker , examples of deception attacks includes phishing scams , fake websites.

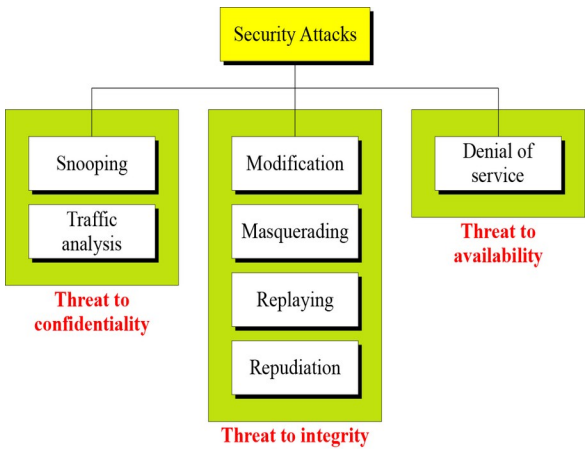
3) Disruption Attacks :- Interruption or prevention of correct operation (Modification , unauthorized change of information) :- this attacks are aimed at disrupting or disabling the normal operations of a system or a network.

4) Usurpation attacks :- unauthorized control of some part of a system , these attacks are aimed at gaining unauthorized access to a system network. Example :- stealing login credentials. Example :- deny of service , it is a process of blocking legitimate users from the system.

An attack is a security threat that involves an attempt to obtain , alter , destroy , remove , implant or reveal information without authorized access or permission. It happens for both individuals and organizations. The goal of security confidentiality , integrity , availability , authentication or non-repudiation can be threatened by security attacks.

Taxonomy of attacks with relation to security goals.

- Threat to confidentiality
- Threat to Integrity
- Threat to availability



Threat to confidentiality :-

snooping :- refers to unauthorized access to or interception of data

Traffic Analysis :- refers to obtaining some other type of information by monitoring online traffic.

Attacks threaten Integrity :-

Modification :- means that the attacker intercepts the message and change it.

Masquerading or spoofing :- happens when the attacker impersonates somebody else.

Replaying :- Replaying means the attacker obtain a copy of a message sent by a user and later tries to reply it.

Repudiation :- In computer security repudiation refers to the act of denying responsibility or involvement in a particular action or transaction. Specifically , repudiation refers to an attack in which an individual denies having performed a particular action or transaction that they actually did perform.

For example :- In the context of online transactions , repudiations may occur when a user denies having authorized a particular purchase or transaction , even though they actually did authorize it. This can occur if the transaction was not properly logged or recorded , or of the user’s credentials were stolen or compromised.

Repudiation attacks can have serious consequences , particularly in situations where financial transactions or legal agreements are involved. To prevent repudiation attacks , systems often use techniques such as digital signatures , transaction logs , and audit trails to provide strong evidence of who performed a particular action or transaction , and to prevent users from denying their involvement or responsibility.

Attacks threaten Availability :-

Denial of service (DOS):- is a very common attack , it may slow down or totally interrupt the service of a system.

WHAT IS SNOOPING IN COMPUTER SECURITY

Snooping in computer security refers to the unauthorized access of data transmitted over a network or stored on a computer system. It is an activity in which an individual or software intercepts and examines network traffic or data packets without the owner’s consent.

WHAT IS SPOOFING IN A COMPUTER SECURITY

spoofing in computer security is the act of falsifying information to deceive or trick a user , a network or a system in to believing that the attacker is someone or something else. It is a type of cyber-attack where the attacker creates a fake identity or impersonates a legitimate one to gain access to sensitive data or to perform unauthorized actions.

Spoofing can be done in various ways , including email spoofing , IP spoofing. In email spoofing , the attacker send an email that appears to come from a legitimate source , such as a bank or a government agency , to trick the recipient in to providing personal or financial information.

TYPES OF ATTACKS : ONE WAY OF CATEGORIZING ATTACKS IS AS PASSIVE AND ACTIVE

Passive Attacks :- A passive attack is a type of attack in which the attacker attempts to obtain information from a system without modifying or disrupting its operations.

There are two types of passive attacks : release of message contents (or sniffing) and traffic analysis

Release of message contents(sniffing) :- A telephone conversation , an electronic mail message , and a transferred file may contain sensitive or confidential information , we would like to prevent an opponent from learning the contents of these transmission. It is a type of passive network attack where an attacker uses a software tool or device to capture data packets transmitted over a network.

Traffic analysis :- to determine the location and identity of communicating hosts and to observe the frequency and length of messages being exchanged (even if the message is encrypted). This information might be useful in guessing in nature of the communication that was taking place.

Note :- Passive attacks are difficult to detect but easy to prevent.

Active Attack :- An active attack is a type of attack in which the attacker attempts to modify or disrupt the normal operations of a system. In active attacks , the attacker tries to alter , destroy , or steal data , or gain unauthorized access to a system.

The transmitted data is fully controlled by the intruder , the attacker can modify , extend , delete or play any data , modify messages in transmit , Add , delete messages , denial of service.

Categories of Active Attacks :-

1) Spoofing or Masquerading :- also called fabrication , an attack on authenticity.

2) Modification or Alteration :- An attack on integrity

3) Delay :- Could be classified as an attack on availability

4) Denial of service (DOS) :- or degrading of service or interruption :- An attack on availability

==========

1) Spoofing or Masquerading ;- situation in which one person or program successfully imitates another (impersonation) by falsifying data and thereby gaining an illegitimate advantages.

2) Modification or Alteration :- An authorized change of information :-

3) Delay :- A temporary inhibition of a service , is a form of usurpation. If an attacker can force the delivery to make more time for a message through manipulation of system control structures , such as network components or server components.

4) Denial of service (DOS) :- OR degrading of service attacks :-

Attackers make resources (server , bandwidth) unavailable to legitimate traffic by overwhelming with bogus traffic , it is blocking access (prevention) of legitimate users to a service / system , it is a form of usurpation.

Active attacks are easy to detect but difficult to prevent.

Attacks	Passive/Active	Threaten
Snooping, traffic analysis Release of message content	Passive	Confidentiality
Modification Masquerading Replaying Repudiation	Active	Integrity
Denial of Service Delay	Active	Availability

TYPES OF THREATS / ATTACKS – ANOTHER WAY OF CATEGORIZING ATTACKS

- Physical Attack :- stealing , breaking , or damaging of computing devices

Denial of service (DoS) Attack :-

Malware Attack :- Malware attack is a type of cyber attack that involves the user of malicious software to gain unauthorized access or cause damage to a computer system or network. Malware is a broad term that refers to any software designed to harm or exploit a computer system.

Hacking (Intrusion) Attack

Is any attempt to intrude or gain unauthorized access to your system either via some operating system flaw or other means. The purpose may or may not be malicious.

Hacker is any skilled computer expert that use their knowledge to overcome a problem. It can be expert programmer more commonly used to refer to someone who can gain unauthorized access to other computers.

Ethical Hacker (White Hat) :- A hacker who gains access to system with a view of to fix the identified weakness. They may also perform penetration , testing and vulnerability assessment.

Cracker (Black Hat) :- A hacker who gains unauthorized access to computer system for personal gain. The intent is usually to steal corporate data , violate privacy of rights , transfer funds from bank account etc ….

Grey Hat (both) :- A hacker is in between ethical and black hat hackers. She / he breaks in to computer system without authority with a view to identify weakness and reveal them to the system owner.

Malware Attack :-

Examples are :- Viruses , Worms , Trojan horses , Spywares , Login bombs

Computer Viruses :- A computer virus is a type of malicious software (malware) that is designed to replicate itself and spread from one computer to another , often without the user's knowledge or consent. Computer viruses are typically spread through email attachments , file downloads , infected websites and other types of malware. To protect against computer viruses , it is essential to use reliable antivirus software and keep software and operating systems up to date with security patches.

Worms :- A worm is a type of malicious software (malware) that is designed to spread across a network or the internet , often without the user's knowledge or consent. Unlike viruses worms do not need to attach themselves to a host file or program to spread. Instead , they can self – replicated and spread independently by exploiting vulnerabilities in computer networks and software.

The main difference between worms and viruses in the way they spread. Viruses require a host file or a program to attach themselves to spread , while worms can spread independently.

It often creates a denial of service.

Trojan Horses :-

A trojan horse or Trojan , is a type of malware that is disguised as a legitimate file or program , but once installed on a computer , it can perform malicious actions. Trojan can be designed to steal sensitive information , create back doors for hackers to gain access to a computer or network or cause damage to a computer system.
Trojans are different from viruses and worms in several ways , unlike viruses , Trojans do not self replicated or infect other files or programs instead they rely on users to download and install them , often by disguising themselves as a harmless or legitimate file or program.

Trojans are also different from worms in that they do not spread independently across network or the internet. Instead , they are typically spread through social engineering tactics , such as phishing emails , or by exploiting vulnerabilities in software or operating systems.

To protect against Trojans , it is important to use antivirus software and keep software and operating systems up to date with security patches. Additionally , it is essential to be cautious when downloading files or programs from unknown sources and to avoid opening email attachments or clicking on links from unknown senders.

Spyware

A software that literally spies on what you do on your computer.

Spyware is a type of malicious software (malware) that is designed to monitor a user's computer activity and gather sensitive information , often without the user's knowledge or consent. Spyware can track key strokes , capture screenshots , record web browsing history , and steal personal information such as login credentials and credit card numbers.

Cookies :- Any data that the cookie saves can be retrieved by any website , so your entire internet browsing history can be tracked.

Key Loggers :- Record all of your key strokes , the most common use of a key logger is to capture usernames and passwords.

A key logger , also known as keystroke or keystroke recorder , Is a type of software or hardware device that is designed to record every keystroke made on a computer or mobile device. This includes every letter , number and symbol typed on a keyboard , as well as mouse clicks and other input methods.

Hardware key logger are physical devices that can be attached to keyboard or USB port to record keystrokes while software key loggers are programs that run in the background of a computer or mobile device and record key strokes.

Spyware is different from viruses , worms , and Trojan horses in that it does not usually cause damage to a computer system. Instead it is focuses on gathering information from the user. While viruses , worms , and Trojan horses can cause damage to a computer system , spyware is designed to operate in the background and remain undetected for as long as possible.

Legal uses of spyware
Positive sides of spyware

Employers may use spyware as a means of monitoring employee use of company technology

Parents may use this type of software on their computer to monitor the activities their children on the internet to protect their children from online predators.

Adware

Adware is a type of software that displays unwanted advertisements on a user's computer or mobile device. Adware is often bundled with other software and installed without the user's knowledge or consent. The ads displayed by adware can take various forms , such as pop-ups , banners and sponsored search results.

Adware is different from viruses , worms and spyware in that it does not usually cause harm to a computer or steal sensitive information. Instead , it is designed to generate revenue for its creators by displaying ads and collecting user data , such as web browsing history and search queries.

Logic Bomb

Software that lays dormant until some condition is met ; that condition is usually a data and time , when the condition is met , the software does some malicious act such as deleting files , altering system configuration or perhaps releasing a virus.

A specific condition is a requirement for a logic bomb to be created , as It determines when the malicious code will be activated. The condition can be based on various factors , such as a particular date or time , the occurrence of a specific event or a certain user action.

For example , a logic bomb might be programmed to activate when a specific employee is terminated or when a particular file is accessed. This specific condition is included in the logic bomb's code and will cause it to execute the harmful action once the condition is met.

The specific condition is often chosen by the creator of the logic bomb to maximize its potential impact and to make It difficult to detect. By selecting a unique condition , the creator can ensure that the logic bomb remains dormant until the desired trigger event occurs , making it more difficult for security measures to detect and prevent its execution.

However , identifying the specific condition can also be a key factor in detecting and preventing the logic bomb. By understanding the potential trigger events for a logic bomb , security professionals can implement monitoring and alerting systems to identify and respond to any unusual or suspicious behavior that may indicate the presence of a logic bomb.

tips to avoid virus and spyware attacks ?

1. Keep your operating system and software up to date: Make sure that your computer's operating system and all installed software are up to date with the latest security patches and updates.
2. Be cautious when downloading files or software: Only download files and software from reputable sources, and avoid clicking on links or downloading attachments from unknown senders.
3. Be cautious when opening email attachments: Don't open email attachments from unknown or suspicious senders, as they may contain viruses or other malware.
4. Use strong passwords: Use complex passwords and two-factor authentication to protect your accounts from unauthorized access.
5. Be cautious when using public Wi-Fi: Avoid using public Wi-Fi for sensitive activities such as online banking or shopping, as these networks may be unsecured and vulnerable to attack.
6. Use a virtual private network (VPN): Use a VPN when using public Wi-Fi or accessing sensitive information online to add an extra layer of security.
7. Back up your data regularly: Back up important files and data regularly to protect against data loss in the event of a virus or spyware attack.

8. Use ad-blocking software: Use ad-blocking software or browser extensions to minimize the risk of unwanted ads, which may contain spyware or other types of malware.
9. Be vigilant and skeptical: Always be skeptical of emails, websites, and other online content that seems too good to be true, and be cautious of unsolicited offers or requests for information.

IDENTITY THEFT

PHISHING

Phishing is a type of cyber attack that involves the use of fraudulent emails or messages to trick individuals in to divulging sensitive information , such as login credentials , credit card numbers or personal data. Phishing attacks typically use social engineering tactics to create a sense of urgency or fear in the victim , in order to persuade them to click on a link or download an attachment that contains malware or directs them to a fake website that is designed to steal their information.

Phishing attacks can take many forms, including emails that appear to be from legitimate organizations, such as banks, social media platforms, or government agencies. These emails may contain convincing logos, graphics, and language that make them appear genuine, and may ask the recipient to update their account information, verify their identity, or take other actions that require the disclosure of personal information.

To protect against phishing attacks, it's important to be cautious when receiving emails or messages from unknown or suspicious sources, and to avoid clicking on links or downloading attachments from these sources. It's also essential to keep software and operating systems up to date with security patches and to use reliable antivirus software to protect against malware. Finally, it's important to educate yourself and others about the risks of phishing attacks and to remain vigilant and skeptical of unsolicited requests for information.

CHAPTER THREE

Cryptography is the practice of securing information by transforming it in to unreadable format using mathematical algorithms and methods. It involves techniques for confidentiality , integrity , and authentication of data , ensuring that only authorized individuals can access and use of the information.

The primary goal of cryptography is to protect sensitive information from being intercepted , read or modified by unauthorized individuals. This is accomplished by encoding the information using a key , which is a set of instructions used to transform the data in to a secure format. The key is kept secret and only those who have the key can decode the information back to its original form.

Cryptology :- It's name is derived from Greek word called "Kryptos" which means "Hidden Secrets". It's is an art and science of secret writing. Or it is the science of using mathematics for encrypting and decrypting data.

Encryption :- The process by which the plain text is converted in to cipher text

Decryption :- Recovering plain text from the cipher text

Secret Key :- In cryptography , a secret key (also know as symmetric key) is a shared secret between two or more parties that is used to encrypt and decrypt information. The secret key is a single key that is used both to encrypt and decrypt data , meaning that the same key is used for both processes.

What is secret key ?

In cryptography a secret key or shared key is a type of cryptographic key that is used fo both encryption and decryption of data. This means that the same secret key is used for both encrypting and decrypting the data.

The secret key is kept private and must be shared only between the sender and the receiver of the encrypted data. The security of the encrypted data depends on the security of the secret key.

When using symmetric key cryptography , the same secret key is used for both to encrypt and decrypt the data , this means that the sender and the receiver must have both access to the same key. This can be a challenge if the sender and the receiver are not in direct communication with each other.

One solution to this problem is to use public key cryptography , which uses a pair of keys , a public key for encryption and a private key for decryption , in this system the public key can be freely distributed while the private key is kept secret , this allows anyone to send encrypted messages to the owner of the private key , without the need for a shared secret key.

The process of using a secret key to encrypt and decrypt information is known as symmetric key cryptography. The encryption process takes the original message and transforms it in to cipher text using the secret key. The cipher text can only be decrypted back to the original message using the same secret key.

The strength of the encryption provided by a secret key depends on the length of the key and the complexity of the encryption algorithm used. Longer keys and more complex algorithms make it more difficult for an attacker to guess the key and decipher the message.

How are we going to use this secret key for both the encryption and decryption purposes ?

In a symmetric key cryptography system , the same secret key is used for both encryption and decryption of data. Here is how the key is used :-

1) Encryption :- To encrypt the data , the sender uses the secret key to scramble the plain text in to the cipher text , the cipher text is then sent to the receiver.

2) Decryption :- To decrypt the data , the receiver uses the same secret key to unscramble the cipher text back in to the plain text.

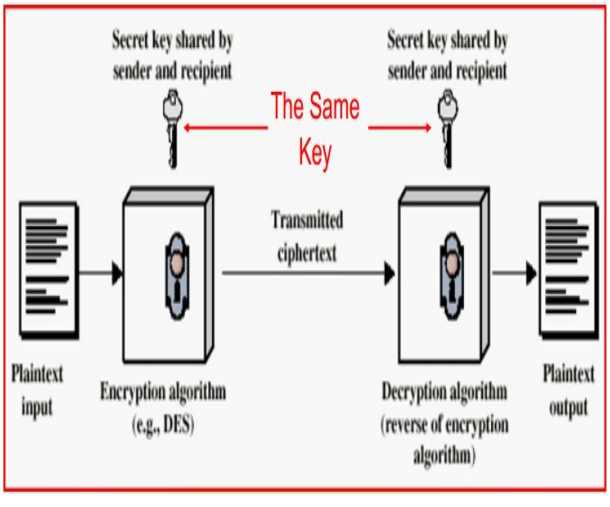
Cryptography has five ingredients :-

Plain Text :- The original message that is fed in to the algorithm as input.

Encryption Algorithm :- Performs various substitutions and transformations on the plain text.

Secret Key (Smmetrical Key) :- is also input to the algorithm : the exact substitutions and transformations performed by the algorithm depend on the key ; the larger key size means greater security but may decrease encryption / decryption speed.

Cipher Text :- the scrambled message produce as output. It depends on the plain text and the secret key : for a given message , two different keys will produce two different cipher texts.



THE NEED FOR CRYPTOGRAPHY

If you have the best firewall , very tight security policies , hardened operating systems , virus scanners , intrusion-detection software , anti spyware and every other computer security angle covered but send your data in raw , plain text , then you simply are not secure

Description :-

* A sender S wants to transmit message M to a receiver R
* To protect the message M , the sender first encrypts it in to unintelligible message M'
* After receipt of M' , R decrypts the message to obtain M
* M is called the plain text : what we want to encrypt
* M' is called the cipher text : the encrypted output

What is Steganography in cryptography ?

Steganography is the practice of hiding a message or information with in another object or a medium , such as an image , audio file , or text document , in order to keep the message secret. In the context of cryptograph , steganography can be used as a technique for secure communcation by embedding a secret message within a seemingly innocuous cover medium.

For example , In image steganography , a message can be hidden with in the pixels of an image file by slightly altering the values of the pixels to encode the message. The changes are usually small enough to be imperceptible to the human eye , but can still be detected and extracted by a recipient who knows how to decode the message.

Notation :- Given
P = Plain Text
C = Cipher Text

$C = E_k(P)$ Encryption
$P = D_k(C)$ Decryption

=> P = Dk(Ek(P))
=> C = Ek(Dk(C))

TYPES OF CRYPTOGRAPHY

1) SYMMETRIC KEY CRYPTOGRAPHY
2) ASYMMETRIC KEY CRYPTOGRAPHY

SYMMETRIC KEY CRYPTOGRAPHY

Symmetric encryption is a form of crypto system in which encryption and decryption are performed using the same key. It is also known as conventional encryption. Symmetric encryption transforms plain text in to cipher text using a secret key and an encryption algorithm. Using the same key and a decryption algorithm. Using the same key and a decryption algorithm , the plain text is recovered from the cipher text.

Symmetric ciphers are a type of cryptography that uses a shared secret key to encrypt and decrypt the messages. In traditional symmetric ciphers , such as the ones used

before the advent of computers , there were two main techniques used to perform the encryption , these are substitution and transposition.

Substitution techniques involve mapping plain text elements , which can be characters , bits or other symbols , in to cipher text elements using a predetermined substitution rule. For-example , a simple substitution cipher might replace each letter of the alphabet with a corresponding number or symbol according to a fixed pattern. The resulting cipher text would be a sequence of numbers or symbols that represent the original message.

Transposition techniques , on the other hand , involve systematically rearranging the positions of plain text elements to create the cipher text. For example , a transposition cipher might shift every third letter of the message to the front , then every fourth letter to the end , and so on , according to a predetermined pattern. The resulting cipher text would be a scrambled version of the original message that is difficult to read without the decryption key.

In practice , modern symmetric ciphers use much more complex techniques than simple substitution or transposition , but the basic concepts remain the same. These techniques are designed to be secure against attacks such as frequency analysis , where an attacker can analyze the frequency of certain letters or symbols in the cipher text to deduce the substitution rule or transposition pattern used to create it.

SUBSTITUTION CIPHER TECHNIQUES

Substitution cipher is a method of encryption by which units of plain text are substituted with cipher text according to a regular system.

SUBSTITUTION CIPHER TECHNIQUES ARE :-

- ➔ Caesar's Cipher
- ➔ Playfair cipher
- ➔ Monoalphabetic cipher
- ➔ Polyalphabetic cipher
- ➔ One time pad and Hill cipher

CAESAR'S CIPHER

Caesar's cipher is a simple substitution cipher that Is name after Julius Caesar , who is said to have used it to encrypt his private messages. In Caesar's cipher , each letter in the plain text is replaced by a letter some fixed number of positions down the alphabet. For example , if the shift is 3 , then A would be replaced by D , B would become E and so on. The key for this cipher is the number of positions to shift each letter.

Here is an example of how Caesar's cipher works with a shift of 3 :-

Plain text :- HELLO WORLD
Ciphertext :- KHOOR ZRUOG

In this example , the letter H is shifted three positions down to become K , E is shifted three positions to become H , L is shifted three positions to become O , and so on. The resulting cipher text appears random and is difficult to read without knowledge of the key.

Caesar's cipher is a simple and easy to use encryption technique , but it also very easy to crack using brute-force attacks or frequency analysis , especially if the key is known to be shift of 1-25 positions. However , it is still a useful introduction to the concepts of substitution and cryptography in general.

WHAT IS BRUTE-FORCE ATTACK

Brute force crypto analysis is a method of attempting to crack a cipher or encrypted message by trying every possible key or combination of keys until the correct one is found. In other words , it involves systematically trying all possible solutions , without any specialized knowledge of the cipher or encryption method being used.

For examples , if a message is encrypted with a Caesar cipher , which involves shifting each letter by a fixed number of positions in the alphabet , brute force attack would involve trying all 25 possible shift values until the correct one is found. This can be a time consuming and computationally expensive process , especially for more complex ciphers with larger key spaces.

Although brute force attacks are generally not practical for larger key sizes or complex ciphers , they can still be effective against weaker ciphers or in cases where the key is relatively short or predictable. Therefore , it is important to use strong cryptographic algorithms with sufficiently long keys to resist brute force attacks.

Encryption :- Suppose we want to encrypt the plain text message "HELLO WORLD" using Caesar's cipher with a shift of 3. The first step is to assign each letter a numerical value based on its position in the alphabet.

A = 0 , B = 1 , C = 2 , D = 3 , E = 4 , F = 5 , G = 6 , H = 7 , I = 8 , J = 9 , K = 10 , L = 11 , M = 12 , N = 13 , O = 14 , P = 15 , Q = 16 , R = 17 , S = 18 , T = 19 , U = 20 , V = 21 , W = 22 , X = 23 , Y = 24 , Z = 25

Using this mapping , we can then apply the caesar cipher encryption function to each letter in the plain text message.

Cipher text = (plain text + shift) mod 26

For example :- to encrypt the letter "H" with a shift of 3 , we would calculate
Cipher text = (7 + 3) mod 26 = 10

Therefore , "H" is encrypted to "K" using Caesar's cipher with a shift of 3 , repeating this process for each letter in the plain text message gives us the cipher text.

Plain text :- HELLO WORLD

NUMERICAL :- 7 4 11 11 14 22 14 17 11 3
SHIFT :- 3 3 3 3 3 3 3 3 3 3
CIPHER TEXT :- K H O O R Z R U O G

Therefore , the encrypted message is “KHOO RZUOG”

DECRYPTION

To decrypt the message , we use the reverse operation. We subtract the shift value from each letter in the cipher text message :-

Plain text = (Cipher text – shift) mod 26

For example , to decrypt the letter “K” with a shift of 3 , we would calculate

plain text = (10 – 3) mod26 = 7

Therefore , “K” is decrypted to “H” using Caesar’s cipher with a shift of 3 , repeating this process for each letter in the cipher text message gives us the plain text.

Cipher text = K H O O R Z R U O G
numerical = 10 7 14 14 17 25 17 20 14 6
Shift :- 3 3 3 3 3 3 3 3 3 3
plain text = H E L L O W O R L D

WHAT IS MOD REFERS TO ?

In the case of Caesar cipher , each letter of the plain text is shifted a certain number of places down the alphabet to create the corresponding cipher text. The shift is determined by the key , which is a number between 1 and 25. For example if the key is 3 , then each letter in the plain text is shifted 3 places down the alphabet to create the corresponding cipher text.

To add the calculation mod 26 in the Caesar cipher , we perform the shift modulo 26. This means that after we shift each letter by the key , we take the result modulo 26 to get a number between 0 and 25. This number represents the position of the new letter in the alphabet , starting with A = 0 , B = 1 , C = 2 and so on. If the result of that shift is greater than 25 , we subtract 26 from the result until it is between 0 and 25.

For example , if the key is 3 and we want to encrypt the plain text “HELLO” , we would perform the following steps :-

1) Convert each letter to its corresponding number in the alphabet H = 7 , E = 4 , L = 11 , 0 = 14

2) Add the key to each number : 7 + 3 = 10 , 4 + 3 = 7 , 11 + 3 = 14 , 14 + 3 = 17

3) Take each result modulo 26 :- 10 mod 26 = 10 , 7 mod 26 = 7 , 14 mod 26 = 14 , 17 mod 26 = 17

4) convert each number back to its corresponding letter in the alphabet :- 10 = K, 7 = H, 14 = O, 17 = R.

5) The cipher text is “KHOR”

By performing the shift modulo 26 , we ensure that the resulting cipher text only contains letters from the alphabet , and that the letters maintain their relative positions in the alphabet , this makes it more difficult for an attacker to decrypt the message without knowing the key.

When taking the modulus of a number , we find the remainder when the number is divided by the modulus. In the case of 10 mod 26 , we are finding the remainder when 10 is divided by 26 , in this case , 10 is less than 26 , so the remainder when 10 is divided by 26 is simply 10. Therefore , 10 mod 26 is equal to 10.

If there is a number greater than 26 , we can continue to take the modulus until we get a number between 0 and 25 , for example If we want to find 35 mod 26 :

1) 35 divided by 26 is 1 , with a remainder of 9
2) Therefore , 35 mod 26 is equivalent to 9 mod 26.

Monoalphabetic Cipher substitution technique:

Caesar cipher is far from secure , it can be easily break by brute – force cryptanalysis because of the key space are small , simply try all possible keys , all possible keys are 26.

A mono alphabetic substitution cipher is a type of cryptographic cipher where each letter of the plain text is replaces by a corresponding letter of the cipher text according to a fixed substitution rule. In other words , each letter of the alphabet is mapped to a different letter , so that the same letter in the plain text is always replaced by the same letter in the cipher text.

The main difference between monoalphabetic substitution and Caesar’s cipher is that in monoalphabetic substitution , each letter of the plain text is replace with a different letter or symbol based on a fixed substitution table , where as in Caesar’s cipher each letter of the plain text is replaced with a letter that is a fixed number of positions down the alphabet.

In monoalphabetic substitution , the substitution table can be any permutation of letters of the alphabet , meaning that each plain text letter could potentially be replaced with any letter in the alphabet. This makes mono alphabetic substitution much harder to crack than Caesar’s cipher , because there are many more possible combinations of letters. However , once the substitution

table is discovered , the cipher becomes very easy to decrypt.

On the other hand , Caesar's cipher is a very substitution technique , which makes it easy to implement and understand. However , it is very easy to crack using brute force attacks or frequency analysis , as there are only 25 possible shift values (excluding the case of no shift) which can be easily tested.

Playfair Cipher substitution technique

In the playfair cipher , a 5*5 grid of letters is used to encode plain text messages. The letters of the alphabet are arranged in grid , with the letter "I" typically being combined with the letter "J" In a single grid cell. To encode a message , the plain text is divided in to pairs of letters and each pair is encoded using a set of rules.

Example :- Solved by lord peter Wismsey in Dorthy Sayers's have his Carcase. In this case , the keyword is monarchy , the matrix is constructed by filling the letter of the keyword from left to right and from top to bottom.

M O N A R

C H Y B D

E F G I/J K

L P Q S T

U V W X Z

The rules for encoding pair of letters in the Playfair cipher are as follows :-

- If the two letters are in the same row in the grid , they are replaced by the letter to their immediate right , with the left most letter wrapping around the right side of the grid.

- If the two letters are in the same column of the grid , they are replaced by the letters immediately below them , with the top letter wrapping around to the bottom of the grid.

- If the two letters are not in the same row or column of the grid , they are replace by the two letters in the same row as the first letter, but in the column of the second letter and vice versa.

- If a pair of letters contains a repeated letter , a filler letter such as "X" is inserted between them before encoding.

Why are we putting i/j in the same element of the table ?

One factor is the limited number of elements in the 5*5 matrix used for the cipher. Since there are only 25 elements in the matrix , it is not possible to include all 26 letters of the alphabet without duplicating one of the letters. So in order to include all 26 letters , I and j are combined in to a single element.

Examples Of Play Fair

Let's say our key phrase is "SECRET MESSAGE". First we need to remove any duplicate letters from the key phrase , and then fill in the remaining letters of the alphabet in order , skipping over any letters that are already in the key phrase. Here's what the key phrase looks like after we've removed duplicates and added in the remaining letters.

S	E	C	R	T
M	A	G	B	D
F	H	IJ	K	L
N	O	P	Q	U
V	W	X	Y	Z

No we we have our 5*5 grid , to encrypt a message using the play fair cipher , we take pairs of letters from the plain text and convert them to pairs of letters using the following rules :

1) If the letters are in the same row of the grid , we replace them with the letters to their immediate right , wrapping around to the left side of the row If necessary.

2) If the letters are in the same column of the grid , we replace them with the letters immediately below , wrapping around the top of the column if necessary.

3) If the letters are not in the same row or column , we replace them with the letters in the same row but in the column of the other letter.

Let's say we want to encrypt the message "HELLO WORLD" first we need to split the message in to pairs of letters like this :-

HE LX LO WO RL DX

HE -- > OA
LX -- > IZ
LO -- > HU
WO -- > EW
RL -- > TK
DX -- > GX

OA IZ HU EW TK GX

HILL CIPHER

The Hill Cipher is a substitution technique used in computer security that employs linear algebra concept to perform encryption and decryption. The Hill Cipher works by breaking the plain text in to blocks of n letters and performing matrix multiplication on each block. The matrix used for multiplication is called key matrix and is typically a square matrix of size n*n. The key matrix must be chosen carefully to ensure that it is invertible , which allows for decryption ($D = K^{-1}C \bmod 26$).

To encrypt a message using the Hill Cipher , the following steps are typically taken ….

1) Choose a key matrix of size n * n
2) Divide the plain text in to blocks of n letters
3) Convert each block of the plain text in to a column vector of size n*1
4) Multiply each column vector by the key matrix to obtain the corresponding encrypted vector.
5) Convert each encrypted vector back in to a block of cipher text.

To decrypt a message that has been encrypted using the Hill Cipher , the following steps are typically taken :

1) Determine the inverse of the key matrix

2) Dive the cipher text in to blocks of n letters

3) Convert each block of the cipher text in to a column vector of size n*1

4) Multiply each column vector by the inverse of the key matrix to obtain the corresponding decrypted vector.

5) Convert each decrypted vector back in to a block of plain text.

The Hill cipher is considered to be relatively strong , as it is resistant to most types of attacks , including brute force attacks , frequency analysis attacks and known plain text attacks .

Poly alphabetic cipher

This method is built to improve the problem of mono alphabetic technique , what was the limitation in the mono alphabetic technique ?

The poly alphabetic cipher solves the problem of the mono alphabetic cipher by using multiple substitution alphabets instead of just one. In a mono alphabetic cipher , each letter of the plain text is replaced with a single corresponding letter in the cipher text using a fixed substitution alphabet. This makes the encryption vulnerable to frequency attacks , where the frequency of letters In the cipher text can be analyzed to infer the original plain text.

In a poly alphabetic cipher , each letter of the plain text is still replaced with a corresponding letter in the cipher text , but the substitution alphabet changes based on the position of the letter in the plain text and the key being used for encryption. This means the same plain text letter can be encrypted to different cipher text letters depending on the position in the message and the specific substitution alphabet used.

Vigenere Cipher

In the vigenere cipher , a popular poly alphabetic cipher , a key word is used to generate a series of substitution alphabets , with each letter of the key word representing a shift in the substitution alphabets , with each letter of the key word representing a shift in the substitution alphabet , this means the same plain text letter may be encrypted to different cipher text letters depending on its position in the message and the specific shift being used.

By using multiple substitution alphabets in this way , the poly alphabetic cipher makes frequency analysis attacks much more difficult , as the frequency of each letter in the cipher text will not match the frequency of that letter in the plain text. This makes it much more difficult for an attacker to determine the original plain text from the cipher text with out knowing the specific substitution alphabets being used.

Encryption process

Ei = (Pi + Ki) mod 26

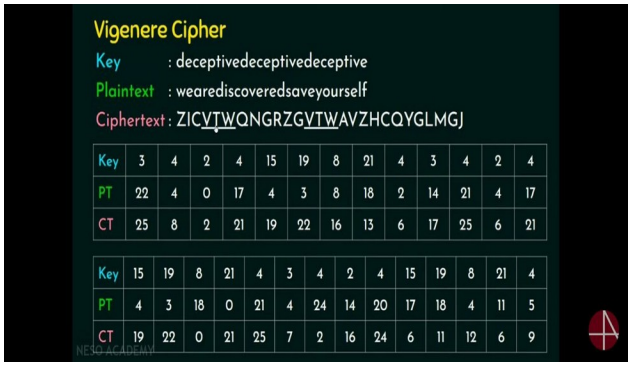
Decryption Process

Di = (Ci – Ki) mode 26

Poly alphabetic substitution cipher that I natural evolution of the caesar cipher , A key is needed that is along the message , usually the key is a repeating key word.

Example : - If the key is deceptive and the plain message is **“**we are discovered save yourself **”** is encrypted as follows

Key :- deceptivedeceptivedeceptive
plain:- wearediscoveredsaveyourself



ONE TIME PAD SUBSTITUTION CIPHER

This technique yields the ultimate in security , because it uses random key that is as long as the plain message. So since the key is random and has a length that is long as the plain message , so there is no need for a key to be repeated.

So since the key is random and has a length that is as long as the plain message so there is no need for a key to be repeated. (This gives additional security)

In addition , the key is to be used to encrypt and decrypt a single message and then discarded (key never reused).

Gives the best security in the history of the cryptography , this is due to the randomness and the key never reused nature of the algorithm. Each new message requires a new key of the same length as the new message (unbroken in nature.)

The fundamental difficulties of one time pad algorithm

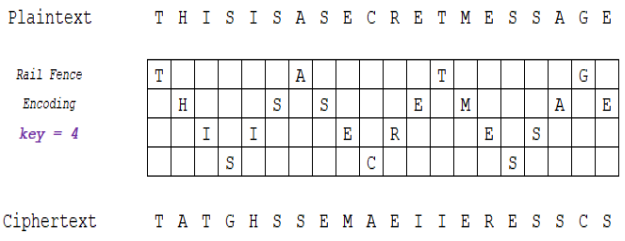
The practical problem of making large quantities of random keys , notice we are generating a random key for every message we want to encrypt , this creates a problem of key distribution and protection.

TRANSPOSITION TECHNIQUES

Transposition technique is a cryptographic technique that converts the plain text to cipher text by performing permutations on the plain text , that is change the position of each character of plain text for each round. It includes like Rail fence technique, simple columnar transposition technique with multiple rounds and book cipher to encrypt the plain text.

RAIL FENCE TECHNIQUE

The Rail Fence technique is a transposition cipher technique used in computer security. It is a type of transposition cipher , which means it involves rearranging the letters in a message without altering the letters themselves. In the Rail Fence technique , the plain text is written out diagonally on a set of number of “rails” or “lines” , then read off in a difference order to create the cipher text.



One weakness of the Rail Fence technique is that it is vulnerable to frequency analysis attacks , which involve analyzing the frequency of certain letters or groups of letters in the cipher text to determine the original message. In addition , the technique can be susceptible to brute force attacks , which involve , the rail fence technique is not generally considered a secure method of

encryption on its own , but it can be useful part of a more complex system.

SIMPLE COLUMNAR TRANSPOSITION TECHNIQUES

Example :- Let's assume that plain text is a corporate bridge and we need to calculate the cipher text using a simple columnar transposition technique. Let's take 6 columns and arrange the plain text in a row-wise manner.

Column 1	Column 2	Column 3	Column 4	Column 5	Column 6
c	o	r	p	o	r
a	t	e	b	r	i
d	g	e			

let's assume 1 , 3 , 5 , 2 , 4 , 6 is an order. Now read the message in a columnar manner using the decided order ---
cadreeorotgpbri cadreeorotgpbri is a cipher text.

The simple columnar transposition technique can be categorized in to two parts – Basic technique and multiple rounds. The simple columnar transposition technique – basic technique. The simple columnar transposition technique simply arranges the plain text in a sequence of rows of rectangle and reads it in a columnar manner.

How does this algorithm work ?

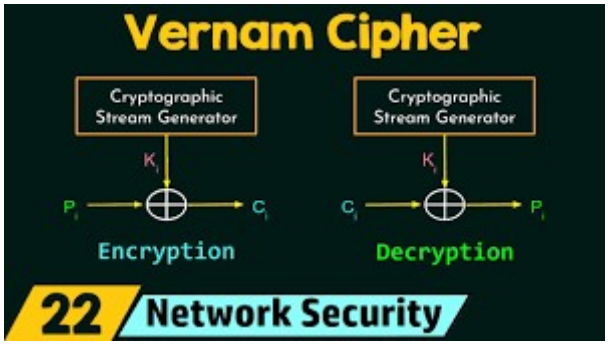
Step 1 :- Write all the character of plain text message row by row in a rectangle of a pre-defined size

Step 2 :- Read the message in a columnar manner , that is column by column

Note :- For reading the message , it needs not to be in the order of columns. It can be any random sequence.

Step 3 :- The resultant message is cipher text.

VERNAM CIPHER



A Vernam cipher is a type of symmetric encryption that uses a one-time pad, which is a random key that is as long as the plaintext message. The key is combined with the plaintext message using the XOR (exclusive or) operation to produce the ciphertext. The key is used only once and is never reused, hence the name "one-time pad".

Here's an example of a Vernam cipher encryption:

Suppose we want to encrypt the message "HELLO" using a one-time pad key of "XMCKL". First, we convert the plaintext message and the key to binary:

- Plaintext "HELLO" in ASCII: 01001000 01000101 01001100 01001100 01001111
- Key "XMCKL" in ASCII: 01011000 01001101 01000011 01001011 01001100

Next, we perform the XOR operation between the binary representations of the plaintext message and the key, bit by bit:

- Ciphertext: 00010000 00001010 00011100 00000111 00000011

Finally, we convert the binary ciphertext back to ASCII characters, which results in the encrypted message: "\n".

To decrypt the ciphertext, the recipient needs to have the same one-time pad key, which is used to perform the XOR operation with the ciphertext to recover the original plaintext message.

One of the main limitations of the Vernam cipher is the need for a secure key distribution channel. Since the key must be randomly generated and only used once, both the sender and the receiver must have access to a secure way of sharing the key. Any compromise in the security of the key compromises the security of the entire system.

Modern Cryptography

A stream cipher is an encryption algorithm that encrypts data one bit or one byte at a time , in a continuous stream. Stream ciphers are typically used to encrypt real time data such as voice and video transmission. It generates a key stream that is combines with the plain text to produce cipher text.

One of the advantages of stream ciphers is that they are typically faster and more efficient than block ciphers , as they can encrypt and decrypt data on the fly with out needing to buffer the entire message. However , stream ciphers are generally less secure than block ciphers , as they can be vulnerable to attacks such as known plain text attacks and related – key attacks.

Block cipher is a type of encryption algorithm that encrypts data in fixed length blocks or chunks , typically of 64 or 128 bits. The plain text is divided in to blocks , which are then encrypted using a cryptographic key and a specific encryption algorithm. Each block is independent of the other , and the same key and algorithm are used for each block.

Block ciphers are considered more secure than stream ciphers because they are less susceptible to known plain-text and other cryptographic attacks. However , block ciphers can be slower and less efficient than stream ciphers , especially when encrypting large amounts of data. Many modern encryption systems use a combination of both stream and block ciphers to provide strong and efficient encryption.

Data Encryption Standard (DES)

DES (Data Encryption Standard) is a symmetric key encryption algorithm that uses a 56 bit key to encrypt and decrypt data. It was developed in the 1970s by IBM and adopted by the US government as a standard for protecting sensitive data. DES operates on 64-bit blocks of data (block cipher) , which are divided in to two 32 bit halves. The algorithm uses a series of permutations and substitutions to transform the plain text in to cipher text and vice versa. The encryption process involves 16 rounds of these transformations , with a different sub key used in each round.

The encoding process of each 64-bit block of data in DES involves several steps :-

1) Initial Permutation (IP) :- The 64 bit plain text block is first subjected to an initial permutation (IP) which rearranges the bits according to a specific permutation table. The result of the IP is then divided in to two 32-bit halves.

2) Multiple rounds of encryption :- The IP output is then subjected to a series of 16 rounds of encryption. Each round uses a 48 bit sub key that is derived from the original 56-bit key using the key scheduling algorithm. In each round , the 32 bit right half of the input is expanded to 48 bits using another fixed permutation table. The resulting 48 bit block is then XORed with the 48 bit sub key. The XOR output is then divided in to eight 6 bit blocks , each of which is substituted using a fixed S-box table. The eight 4-bit outputs from the S-boxes are then combined to produce a 32 bit output. The 32 bit left half of the input is then XORed with the 32-bit output of the S-boxes. The resulting 32-bit halves are swapped to become the input for the next round.

3) Final permutation (FP) :- After the final round of encryption , the resulting 64-bit block is passed through a fixed permutation table known as the Final Permutation (FP).

4) Output :- The resulting 64-bit output Is the encrypted cipher text.

The decryption process of DES is simply the reverse of the encryption process. The cipher text block is first subjected to the IP , followed by the 16 rounds of decryption using the same sub keys in reverse order. Finally , the resulting 64 bit block is passed through the FP to produce the original plain text block.

The working principles of DES

What is the whole point of DES ?

The main point of DES is to provide confidentiality by encrypting data in a way that it can only be decrypted by someone who has the correct key. This is achieved by using a complex set of mathematical operations that scramble the data in to unintelligible form. Only someone who has the key can perform the reverse operations and recover the original data.

DES was designed to be a relatively fast and efficient encryption algorithm that could be implemented in hardware or software. However , it was eventually replaced by more secure and advanced encryption algorithms , such as the Advanced Encryption Standard (AES) , due to its vulnerability to brute force attacks.

Why are we dividing the plain text in to left and right half ?

In DES encryption , the plain text is divided in to left and right halves because it is a Feistel cipher. A Feistel cipher is a cryptographic algorithm that uses a combination of substitution and permutation to encrypt data. It operates on blocks of data , dividing them in two halves and processing them iteratively through a series of rounds.

In the case of DES , the plain text block is divided in two 32-bit halves , called the left half and the right half. These halves are processed independently through the Feistel network , which consists of 16 rounds of processing. During each round , the right half is fed through a series of substitution and permutations , using a sub key derived from the main encryption key. The output of these substitutions and permutations is then combines with the left half using an exclusive OR operation. The result of this operation becomes the right half of the next round , while the previous right half becomes the left half.

The process of dividing the plain text block in to two halves is a crucial step in the Feistel cipher design , as it allows for the use of the same processing functions and sub keys on both halves. This results in a highly efficient encryption algorithm that can be implemented in hardware and software with minimal computational resources.

What is a feistel cipher is mean by ?

A Feistel cipher is a symmetric encryption scheme used in many cryptographic applications , it works by dividing the plain text in to two halves and then performing multiple rounds of substitution and permutation operations on these halves. In each round , one half of the data is processed and the result is XORed with the other half to produce the new half. The halves are then swapped and the process is repeated for a fixed number of rounds. The output of the last round is the cipher text.

Why are we using initial permutation ?

The initial permutation in DES (Data Encryption Standard) cryptography is used to ensure that the input plain text bits are shuffled pr permuted before the actual encryption process. The main purpose of the initial permutation is to provide confusion and diffusion to the plain text , making it harder for attackers to decipher the original message.

The initial permutation swaps the positions of certain bits in the plain text according to a per-defined permutation table. The permutation table consists of 64-bit positions , where each position corresponds to a specific bit in the plain text. The permutation table is fixed and known to both the sender and receiver.

The initial permutation ensures that the plain text is transformed in to a new sequence of bits that are not easily recognizable. This helps to increase the security of the encryption process and prevent attackers from identifying patterns in the plain text that could be exploited to break the encryption.

Why are we rounding 16 times to generate the final key result ?

The DES algorithm uses 16 rounds to generate the final key result. This is because 16 rounds are considered to be the optimal number of rounds for achieving both security and efficiency.

During each round , the algorithm performs a series of operations , including permutation , substitution , and XOR operations , to create a new key from the previous key. These operations are designed to make it difficult for attackers to

determine the original key from the encrypted data , even if they have access to the algorithm and the cipher text.

If the number of rounds were increased to 17 , the algorithm may become more secure , but it may also become less efficient. Conversely , if the number of rounds were decreased to 15 ,the algorithm may become less secure. Therefore 16 rounds strike a balance between security and efficiency that is considered optimal for the DES algorithm.

Why are we using the substitution box or S-box ?

The substitution box or S-box is a crucial component in the DES encryption algorithm. It is used to add confusion and non linearity to the encryption process , making it more secure.

The s-box works by taking a 6-bit input and producing a 4-bit output. It achieves this through a series of substitution operations that are specified in the DES algorithm. The S-box is essentially a look up table that maps each possible 6-bit input to a corresponding 4-bit output.

The reason for using the S-box is to introduce non-linearity in to the encryption process , if only linear operations were used , then the encryption process could potentially be broken using linear algebra. However , by introducing non-linearity through the S-box , the encryption becomes much more resistant to attack.

Non-linearity means that the relationship between the input and output bits is not simple , predictable linear function. In other words , changing a single bit in the input can result in a completely different output , making it difficult for an attacker to analyze and break the encryption.

Another important feature of the S-box is that it provides confusion , this means that each output bit is influenced by multiple input bits , making it difficult for an attacker to determine the relationship between the input and output bits.

The DES algorithm uses eight S-boxes , each of which performs a unique substitution operation. The choice of these s-boxes was based on extensive testing and analysis to ensure their effectiveness in providing both confusion and non-linearity.

Why are we using a left circular shift in the key ?

Why are we using inverse initial permutation ?

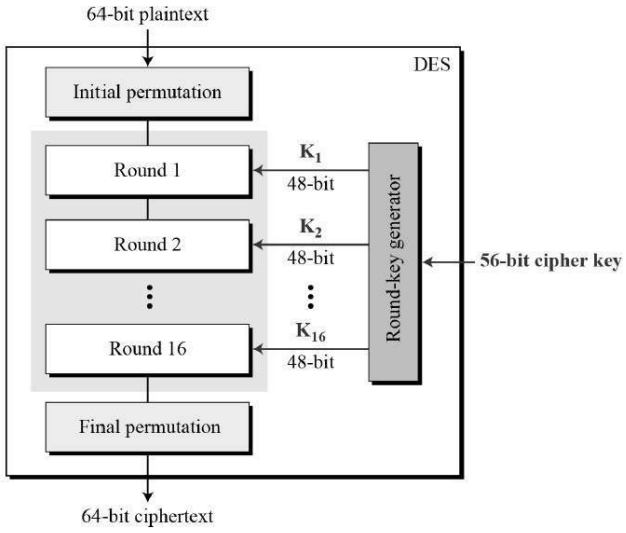
The inverse initial permutation is used in DES to provide additional security to the encrypted message. After the 16 rounds of encryption are completed , the resulting cipher text Is subjected to a final permutation known as the final permutation (IP-1) which is the inverse of the initial permutation (IP).

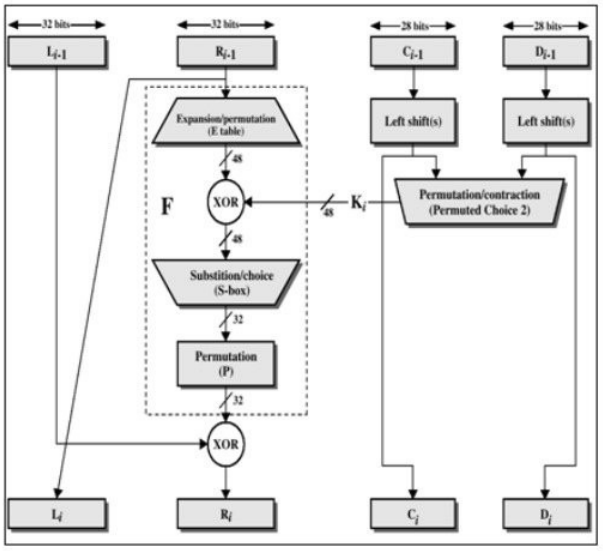
The purpose of the inverse initial permutation is to provide a final layer of diffusion to the encrypted message , making it more difficult for an attacker to decipher the message. The inverse initial permutation rearranges the positions of the bits in the cipher text so that adjacent bits in the input message are no longer adjacent in the output cipher text.

This shuffling of bits provides additional security it breaks up any patterns that may have been created during the encryption process , making it more difficult for an attacker to analyze the cipher text and determine the original message.

The Technical Working Of DES

- **Block size = 64 bit**
- **Key size = 64 bit**
- **Number of rounds = 16 rounds**
- **The plain text is processed in number of rounds , each one should have a separate independent key , so we have 16 sub keys for each rounds**
- **Sub key size = 48 bit sub key – we have to generate 48 bit sub keys for each of the 16 rounds**





1) DES KEY SCHEDULING :- CREATING 16 SUB KEYS , EACH OF WHICH IS 48 BITS LONG

In the DES key scheduling process, the original 64-bit key is first converted into a 56-bit permutation known as the permutated key, denoted as K+. This is achieved through a process called the initial permutation or IP-1, which rearranges the bits in the key according to a fixed table.

Original 64-bit key

PC-1

57	49	41	33	25	17	9
1	58	50	42	34	26	18
10	2	59	51	43	35	27
19	11	3	60	52	44	36
63	55	47	39	31	23	15
7	62	54	46	38	30	22
14	6	61	53	45	37	29
21	13	5	28	20	12	4

K=0001001100110100010101110111100110011101111100110110001111

K+=11110000110011001010101011110101010101100110011110001111

The PC-1 table, or permutation choice-1, is used for this initial permutation. The 64-bit key is divided into two halves, with 28 bits each. Each half is then subjected to a circular shift according to a predefined schedule. The shifted halves are then combined to form a 56-bit key, where each bit position is determined by the PC-1 table.

The PC-1 table consists of 56 entries, each corresponding to a specific bit position in the 64-bit key. The table is designed in such a way that it selects a subset of bits from the original key, discarding 8 of the 64 bits. The selected bits are then permuted according to the table to produce the 56-bit permutated key.

LEFT CIRCULAR SHIFTS

The DES key scheduling process is used to create 16 sub-keys, each of which is 48 bits long, from an original 64-bit key.

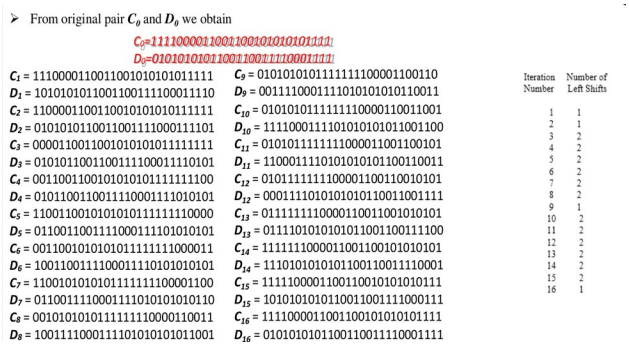
The first step in the key scheduling process is to permute the original key, K, using a permutation known as PC-1. This permutation takes the 64-bit key and transforms it into a 56-bit key known as K+.

Next, the 56-bit key is split into two halves, C0 and D0, each with 28 bits. These halves are used to generate 16 blocks, C1 through C16 and D1 through D16.

To generate each block, the previous block is shifted left by either one or two bits, depending on the block number, and the first bit of the block is cycled to the end. This process is done separately for both halves.

For example, C3 and D3 are obtained from C2 and D2, respectively, by two left shifts, and C16 and D16 are obtained from C15 and D15, respectively, by one left shift.

The resulting 16 blocks are then combined to form the sub-keys. The sub-keys are created by taking a 48-bit subset of the 56-bit key for each round of the encryption process.



A left circular shift is an operation that shifts all the bits in a binary number to the left by a certain number of positions, and wraps the shifted-out bits around to the beginning of the number. This operation is often used in cryptography and computer science.

For example, suppose we have the binary number 11001100 and we want to perform a left circular shift by 3 positions. The result would be 01100110, because the three leftmost bits have been shifted around to the right side of the number.

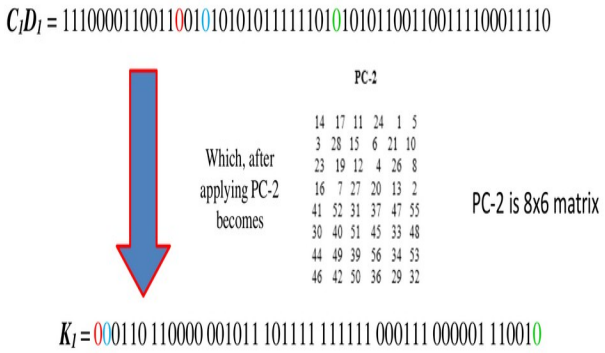
We need left circular shifts in the DES key scheduling algorithm to generate the 16 subkeys. These shifts are used to create new pairs of 28-bit blocks Cn and Dn by shifting the bits in the previous blocks to the left and wrapping the shifted-out bits around to the right side of the block.

Note :- Now the left circular shift depends on the round number , we will decide how many bits we have to shift , so for the round 1 , 2 , 9 , 16 → we will perform 1-bit circular shift and for the rest we will perform 2-bit circular shift.

Permutation choice 2

In the DES key scheduling process, after creating the 16 blocks of subkeys, each of which is 48 bits long, we need to apply a permutation table to each of the concatenated pairs CnDn to obtain the individual keys Kn. This permutation table is known as PC-2 and is an 8x6 matrix.

To obtain Kn, we take the concatenated pair CnDn, which is 56 bits long, and apply PC-2 to it. The first bit of Kn is the 14th bit of CnDn, the second bit is the 17th, and so on, ending with the 48th bit of Kn being the 32nd bit of CnDn.

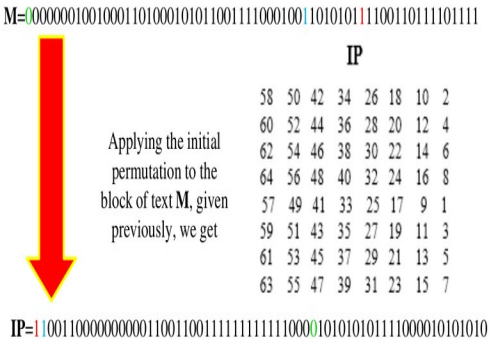


K_2 = 011110 011010 111011 011001 110110 111100 100111 100101
K_3 = 010101 011111 110010 001010 010000 101100 111110 011001
K_4 = 011100 101010 110111 010110 110110 110011 010100 011101
K_5 = 011111 001110 110000 000111 111010 110101 001110 101000
K_6 = 011000 111010 010100 111110 010100 000111 101100 101111
K_7 = 111011 001000 010010 110111 111101 100001 100010 111100
K_8 = 111101 111000 101000 111010 110000 010011 101111 111011
K_9 = 111000 001101 101111 101011 111011 011110 011110 000001
K_{10} = 101100 011111 001101 000111 101110 100100 011001 001111
K_{11} = 001000 010101 111111 010011 110111 101101 001110 000110
K_{12} = 011101 010111 000111 110101 100101 000110 011111 101001
K_{13} = 100101 111100 010111 010001 111110 101011 101001 000001
K_{14} = 010111 110100 001110 110111 111100 101110 011100 111010
K_{15} = 101111 111001 000110 001101 001111 010011 111100 001010
K_{16} = 110010 110011 110110 001011 000011 100001 011111 110101

So much for the subkeys. Next, we look at the message itself.

2) MESSAGE ENCODING

In the process of message encoding, each 64-bit block of data is first subjected to an initial permutation (IP) to rearrange the bits according to a predefined table. The IP table maps the bits from their initial order in the message to a new arrangement specified in the table. For instance, the 58th bit of the message becomes the first bit of IP, the 50th bit becomes the second bit of IP, and so on until the last bit of the message becomes the 64th bit of IP. This rearranged block of data is then ready to undergo further processing for encryption.



In DES encryption, the 64-bit message block is first permuted using an initial permutation (IP) table, which rearranges the bits according to a predefined pattern. The IP table specifies the new arrangement of bits, where the 58th bit of the message becomes the first bit of IP, the 50th bit becomes the second bit of IP, and so on.

For example, suppose the original message block is: M = 0000000100100011010001010110011111000100110101011100110111101111

After applying the initial permutation table, the new block becomes: IP = 1100110000000000110011001111111111110000101010111100001010101010

Next, the IP block is divided into two halves: a left half L0 of 32 bits and a right half R0 of 32 bits. The left half is formed from the first 32 bits of IP, while the right half is formed from the last 32 bits.

For example, in the above IP block, we have: L0 = 11001100000000001100110011111111 R0 = 11110000101010101111000010101010

Now, the DES algorithm proceeds through 16 rounds of encryption. In each round, the left half Ln of the previous round becomes the right half Rn-1 of the current round, and the right half Rn of the current round is obtained by XORing the left half Ln-1 of the previous round with a function f that operates on Rn-1 and a subkey Kn of 48 bits.

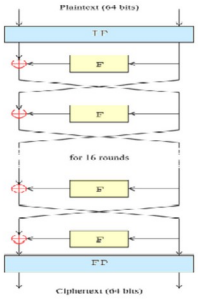
The subkeys Kn are generated from the original 64-bit key using a key schedule algorithm that generates 16 48-bit subkeys. In each round, the corresponding subkey Kn is used in the function f.

The function f operates on a 32-bit block of data and a 48-bit subkey Kn, and produces a 32-bit output. The function f includes several operations, such as expansion, permutation, substitution, and XORing. The details of the function f are beyond the scope of this explanation, but the result of the function f is XORed with Ln-1 to obtain Rn.

At the end of 16 rounds of encryption, the final block is obtained as L16R16. This final block undergoes a final permutation (FP) using a permutation table that is the inverse of the initial permutation table. The result is the encrypted message block.

$L_n = R_{n-1}$

$R_n = L_{n-1} \oplus f(R_{n-1}, K_n)$



In the DES algorithm, the function f operates on two blocks - a data block of 32 bits and a subkey of 48 bits - to produce a block of 32 bits. The function f consists of four stages: expansion, key mixing, substitution, and permutation.

1. Expansion: The 32-bit data block is first expanded to 48 bits using the expansion permutation, denoted as E in the diagram. The expansion permutation duplicates some of the bits to produce a 48-bit output.
2. Key mixing: The expanded data block is then combined with a subkey using an XOR operation. There are 16 subkeys in total, one for each round, which are derived from the main key using the key scheduling algorithm. The subkey is also expanded to 48 bits using a permutation, called PC-2.

3. Substitution: After the XOR operation with the subkey, the block is divided into eight 6-bit pieces before processing by the S-boxes (substitution boxes). Each of the eight S-boxes replaces its 6-bit input with a 4-bit output according to a non-linear transformation, provided in the form of a lookup table. The S-boxes provide the core of the security of DES, as they introduce non-linearity into the encryption process, which makes it more difficult to crack.

4. Permutation: Finally, the 32 outputs from the S-boxes are rearranged according to a fixed permutation, called the P-box, to produce the output of the function f.

The resulting 32-bit output is then XORed with the left half of the input block to produce the right half of the output block. The left half of the input block becomes the right half of the output block.

The Expansion

In the DES algorithm, the function f is used to mix the right half of the block (32 bits) with a 48-bit subkey. The function f consists of several operations, one of which is the expansion operation, which expands the 32 bits to 48 bits by duplicating some of the bits. This expansion operation is denoted by E and is done using a selection table that repeats some of the bits in the input block.

To calculate E(R0) from R0, we apply the selection table to R0 as follows. The first bit of E(R0) is the 32nd bit of R0, the second bit of E(R0) is the 1st bit of R0, and the third bit of E(R0) is the 2nd bit of R0. Similarly, we continue applying the selection table to obtain all 48 bits of E(R0).

- Let **E** be such that the 48 bits of its output, written as 8 blocks of 6 bits each, are obtained by selecting the bits in its inputs in order according to the following table:

E BIT-SELECTION TABLE

32	1	2	3	4	5
4	5	6	7	8	9
8	9	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	30	31	32	1

- Thus the first three bits of $\mathbf{E}(R_{n-1})$ are the bits in positions 32, 1 and 2 of R_{n-1} while the last 2 bits of $\mathbf{E}(R_{n-1})$ are the bits in positions 32 and 1.
- We calculate $\mathbf{E}(R_0)$ from R_0 as follows:

R_0 = 11110000101010101111000010101010
$\mathbf{E}(R_0)$ = 011110100001010101010101011110100001010101010101

- Note that each block of 4 original bits has been expanded to a block of 6 output bits.

KEY MIXING

Key mixing is the process of combining the output of the expansion function E and a subkey using an XOR operation to generate a 48-bit result. This operation is performed in each round of the DES encryption process. The subkeys are generated from the original 64-bit key using the key scheduling algorithm.

The key mixing step is important because it adds additional complexity to the encryption process, making it more difficult for an attacker to determine the key or the plaintext message.

Here's an example of key mixing in the first round of DES encryption:

- Original message block: 11110000101010101111000010101010
- Permutation choice 1 (PC-1) output: 00011011000000101110111111111110001110000001110010
- Left half L0: 11001100000000001100110011111111
- Right half R0: 11110000101010101111000010101010
- Expansion function output E(R0): 01111010000101010101010111101000010101010101
- Subkey K1: 00011011000000101110111111111110001110000001110010
- XOR of E(R0) and K1: 01100001000101110010101010000111011010101010111

In this example, we see that the 48-bit result of key mixing is generated by XORing the 48 bits of the output of the expansion function E with the 48 bits of the subkey K1. This result is then used in the next step of the DES encryption process, which involves substituting values using S-boxes.

A	B	A XOR B
0	0	0
0	1	1
1	0	1
1	1	0

SUBSTITUTION

In DES, substitution is performed using a set of eight S-boxes, each taking six input bits and producing four output bits. The S-boxes implement a nonlinear transformation, providing the core of the security of DES. Without them, the cipher would be linear and easily broken.

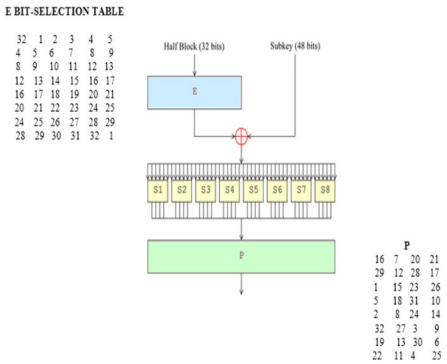
Each S-box is a table that maps 6-bit inputs to 4-bit outputs. The input to an S-box is a 6-bit block of the expanded right half of the previous step (i.e., Rn-1) XORed with the 48-bit subkey Kn. The output of each S-box is a 4-bit block.

The S-boxes are defined by a series of tables, where each table is a 4x16 matrix. For example, the first S-box takes the first six bits of the input and returns a 4-bit output. This is done by selecting a row based on the first and last bits of the input and a column based on the middle four bits of the input. The value found in the selected row and column is the 4-bit output of the S-box.

After all eight S-boxes have been applied to the input, the resulting 32-bit block is then passed through a permutation function, known as the P-box, to provide the output of the F-function. The P-box rearranges the bits in the 32-bit block according to a fixed permutation.

The output of the F-function is then XORed with the left half of the previous step (i.e., Ln-1) to produce the new right half (Rn) for the current step. The left half for the current step (Ln) is simply the right half from the previous step (i.e., Rn-1).

This process of applying the F-function to alternate halves of the message data, XORing the output with the other half, and swapping the halves is repeated for 16 rounds to produce the final output ciphertext.



We now calculate:
S1(B1)S2(B2)S3(B3)S4(B4)S5(B5)S6(B6)S7(B7)S8(B8)

How are we going to calculate it

S1(011011) :- The first and last bits are rows and the middle four bits represent the column number

01 :- a two bit :- what is the maximum value can a two bit represent (11) , which is three and what is the minimum value can a two bit represent , which is 00 , so the rows will be from 0-3 , which is four.

1101 :- a four bit , what Is the maximum value can a four bit represent (1111) , what is the minimum value can a four bit represent (0000) , so the number of columns will be from 0-15.

So in this typical example the first and the last two bits (01) means 1 , which is row 1 and the middle bits 1101 :- means 13 , which is the 13th column in our look up table.

determine S_1 is shown and explained below:

$$S_1(B) = S_1(011011) = 0101.$$

S1

Column Number

Row No.	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

If S1 is the function defined in this table and B is a block of 6 bits, then S1(B) is determined as follows: The first and last bits of B represent in base 2 a number in the decimal range 0 to 3 (or binary 00 to 11). Let that number be i. The middle 4 bits of B represent in base 2 a number in the decimal range 0 to 15 (binary 0000 to 1111). Let that number be j.

Look up in the table the number in the i-th row and j-th column. It is a number in the range 0 to 15 and is uniquely represented by a 4 bit block. That block is the output S1(B) of S1 for the input B. For example, for input block B = 011011 the first bit is "0" and the last bit "1" giving 01 as the row. This is row 1. The middle four bits are "1101". This is the binary equivalent of decimal 13, so the column is column number 13. In row 1, column 13 appears 5. This determines the output; 5 is binary 0101, so that the output is 0101. Hence S1(011011) = 0101.

S1

14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

S2

15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

S3

10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

S4

7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

S5

2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

S6

12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

S7

4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

S8

13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

Last Permutation

After applying the S-boxes to the input block, we get 8 blocks of 4 bits each. These blocks are then concatenated to form a single 32-bit block. The permutation P is then applied to this 32-bit block.

The permutation P rearranges the bits in the 32-bit block according to the following table:

P

16	7	20	21
29	12	28	17
1	15	23	26
5	18	31	10
2	8	24	14
32	27	3	9
19	13	30	6
22	11	4	25

➢ From the output of the eight S boxes:

$S_1(B_1)S_2(B_2)S_3(B_3)S_4(B_4)S_5(B_5)S_6(B_6)S_7(B_7)S_8(B_8) = 0101\ 1100\ 1000\ 0010\ 1011\ 0101\ 1001\ 0111$

we get $f = P(S_1(B_1)S_2(B_2)...S_8(B_8)) = 0010\ 0011\ 0100\ 1010\ 1010\ 1001\ 1011\ 1011$

The permutation table specifies which bit of the 32-bit input block goes to which position in the output block. For example, the 16th bit of the input block goes to the first position of the output block, the 7th bit of the input block goes to the second position of the output block, and so on.

The output of the permutation P is the final output of the function f. It is a 32-bit block that is XORed with the left half of the input block to produce the right half of the output block.

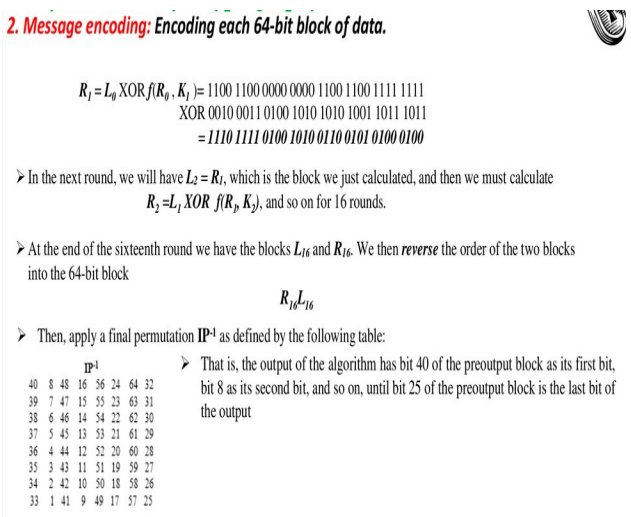
XORING WITH THE LEFT HALF

After completing the Feistel function for the right half of the block and obtaining the output from it, we need to combine it with the left half of the block to obtain the final output. This is done by performing an XOR operation between the output of the Feistel function and the left half of the block.

For example, let's say the left half of the block is L and the right half is R. After performing the Feistel function on R, we get the output f. The final output block is obtained by performing the XOR operation between L and f:

Output block = L XOR f

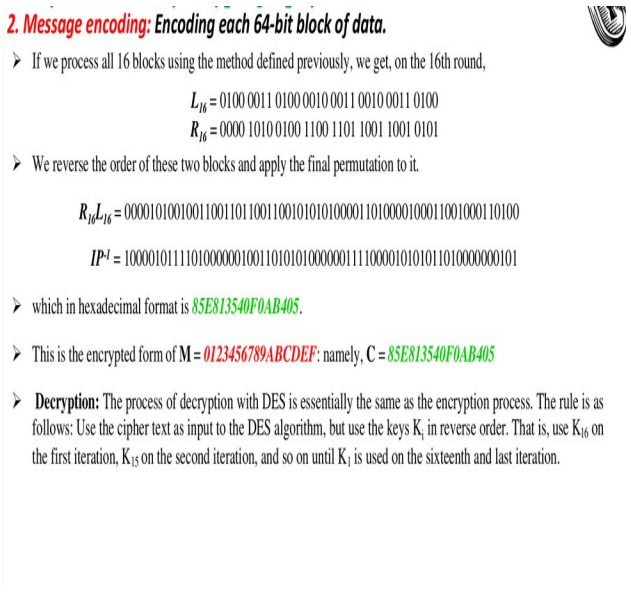
This process is repeated for each round of the DES algorithm until the final round, where the left and right halves of the block are swapped and combined to produce the final output.



This is the final step of the DES encryption process. After performing 16 rounds of encryption, the 32-bit halves L16 and R16 are obtained. The order of these halves is reversed, i.e., R16L16 is formed.

The final permutation IP-1 is applied to R16L16, which rearranges the bits according to the following table. The first bit of R16L16 becomes the 40th bit of IP-1. The second bit of R16L16 becomes the 8th bit of IP-1, and so on. The last bit of R16L16 becomes the 48th bit of IP-1.

The resulting 64-bit block in binary format is then converted to hexadecimal format to obtain the ciphertext. In this example, the ciphertext is 85E813540F0AB405. This is the encrypted form of the original message M = 0123456789ABCDEF.



Asymmetric Key Encryption

Asymmetric key cryptography , also known as public key cryptography , is a cryptographic system that uses two mathematically related keys , a public key and a private key , to secure communication between two parties.
The public key can be freely distributed to anyone who want to communicate with the owner of the private key. The private key , on the other hand , must be keep secret by the owner and should not be shared with anyone.

When someone want to send a secure message to the owner of the public key , they encrypt the message using the public key. The owner of the public key can then decrypt the message using their private key. This means that only the owner of the private key can read the message.

Symmetric key cryptography uses the same key for both encryption and decryption. While this approach is fast and efficient , it has significant limitation in terms of key distribution. If two parties want to communicate securely using symmetric key cryptography , they must both have the same key. This creates a problem of securely exchanging the key without anyone else intercepting it.

Asymmetric key cryptography solves this problem by using two mathematically related keys , a public key and a private key. The public key can be shared widely , while the private key is secret. This allows for secure communication between two parties without the need for a secure key exchange.

Another advantage of asymmetric key cryptography is that it can be used for digital signatures , which are used to verify the authenticity of messages and documents. The private key is used to generate the digital signature , while the public key is used to verify the signature. This allows for secure authentication and non-repudiation , meaning that the sender of a message can not deny having sent it.

Public Key Encryption :-

Both the sender and the receiver own a pair of keys , one public and the other a closely guarded private one. To encrypt a message from sender A

to receiver B , both A and B must create their own pairs of keys.

Then A and B publicize their public keys – any body can acquire them. When A is to send a message M to B , A uses B's public key to encrypt M. On receipt of M , B then uses his or her private key to decrypt the message M. As long as only B , the recipient , has access to the private key , then A , the sender , is assured that only B , the recipient , can decrypt the message.

This ensures data confidentiality. Data integrity is also ensured because for data to be modified by an attacker it requires the attacker to have B's the recipient private key. Data confidentiality and integrity in public key encryption is also guaranteed.
Public key encryption , also known as asymmetric key encryption , is used for a variety of reasons.

1) Secure Key Exchange :- Public key encryption allows for secure key exchange without the need for a secure channel. Each user has a public key that can be freely distributed , allowing others to encrypt messages that can only be decrypted by the owner of the private key.

2) Digital Signature :- Public key encryption can be used to create digital signature that verify the authenticity and integrity of messages and documents. The sender uses their private key to generate a digital signature , and the recipient can verify the signature using the sender's public key. This provides a secure method for authentication and non-repudiation , meaning that the sender of a message can not deny having sent it.

3) Secure communication :- Public key encryption can be used for secure communication between two parties , as each user has their own public and private key. This eliminates the need for a shared key , which can be compromised if intercepted by a third party.

4) Key management :- Public key encryption simplifies key management , as each user only needs to manage their own private key. This eliminates the risk of lost or stolen keys compromising the security of messages encrypted with that key.

THE RSA ALGORITHM

RSA is a widely used public key encryption algorithm in computer security. RSA is based on the mathematical concept of the difficulty of factoring large numbers in to their prime factors.

How does factoring large numbers in their prime factors , will increase the difficulty of breaking the algorithm ?

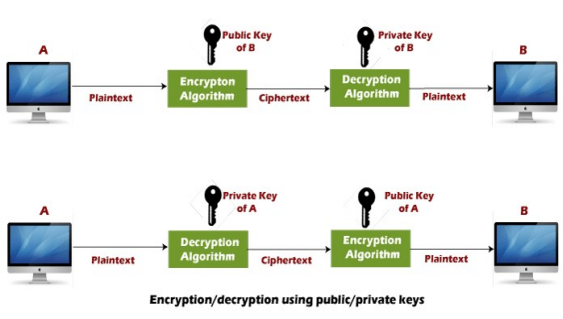
The RSA algorithm involves two keys , a public key and a private key. The public key can be freely distributed to any one who wants to send encrypted message to the owner of the private key. The private key , on the other hand , must be kept secret and is only known to the owner.

To better understand RSA , let first understand what is public-key encryption algorithm.

Public key encryption algorithm :- Public key encryption algorithm is also called Asymmetric algorithm. Asymmetric algorithms are those algorithms in which sender and receiver use different keys for encryption and decryption. Each sender is assigned a pair of keys :- Public Key and a private key.

The public key is used for encryption and the private key is used for decryption. Decryption can not be done using a public key. The two keys are linked , but the private key can not be derived from the public key. The public key is well known , but the private key is secret and it is known only to the user who owns the key. It means that everybody can send a message to the user using user's public key but only the user can decrypt the message using his private key.

The Public key algorithm operates in the following manner:

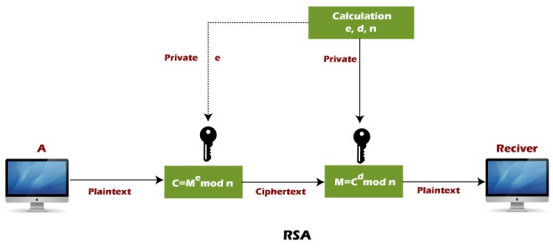


Encryption/decryption using public/private keys

The data to be sent is encrypted by sender A using the public key of the intended receiver. B decrypts the received cipher text using its private key , which is known only to B , B replies to A encrypting its message using A's public key.

A decrypts the received cipher text using its private key , which is known only to him. Encryption using the RSA algorithm involves representing the plain text message as a number m and raising it to the power of the public exponent e module n. The result , c , is the cipher text. Decryption involves raising the cipher text c to the power of the private exponent d modulo n , which yields the original plain text message m.

$C = P^e \mod(n)$
$P = C^d \mod(n)$



RSA

RSA algorithm uses the following procedure to generate public and private keys :-

Select two large prime numbers , p and q.

When it comes to the RSA algorithm , "selecting two large prime numbers , p and q" means that you need to choose two prime numbers that are sufficiently large. These two prime numbers will be used to calculate the modulus n , which is the basis for the public and private keys used in the RSA algorithm.

The choice of these prime numbers is critical for the security of the RSA algorithm. The larger the prime numbers , the more secure the algorithm will be. This is because factoring large numbers in to their prime factors is a difficult computational problem , and the security of the RSA algorithm is based on the assumption that factoring large numbers in to their prime factors is a hard problem.

For instance , if n = 55 , it can be factored in to p = 5 , and q = 11. This is relatively easy to do. However , if n = 8191 , it is much harder to factor it in to its prime factors , which makes the RSA encryption and decryption more secure.

The security of RSA depends on the fact that it is easy to calculate the modulus n from p and q , but it is very difficult to calculate p and q from n. p and q are kept secret in the RSA algorithm. They are used to calculate the modulus n , which is made public , but p and q themselves are not shared. The security of the RSA algorithm relies on the difficulty of factoring n in to p and q , so keeping p and q secret helps to maintain the security of the system.

Multiply these numbers to find n = p * q , where n is called the modulus for encryption and decryption.

In the RSA algorithm the modulus "n" is a product of two large numbers "p" and "q". The product "n" is used as the modulus for both encryption and decryption. The larger value of "n" the more secure the RSA algorithm , as it becomes increasingly difficult to factor "n" back in to its prime factors "p" and "q". To find "n" , we need to multiply the two prime numbers "p" and "q" For example , if we choose "p" to be 17 and "q" to be 11 , then the modulus "n"is n = p * q = 17 * 11 = 187 //

The modulus n is important because it serves as the basis for the public and private keys used in the RSA algorithm. The public keys is derived from the modulus and an encryption exponent , while the private key is derived from the modulus and a decryption component.

Example :- Now let's go through a simple worked example where the message / plain text is given to be m = 6 , p = 7 and q = 19.

A) Key Generation

1) Generate two large prime numbers , p and q

To make the example easy to follow small numbers are used , but this is not secure. To find random primes , we start at a random number and go up ascending odd numbers until we find a prime. Let's take the given two primes for the sake of simplicity.

P = 7 , q = 16

2) Let's compute the modulus , n = p * q
n = 7*19 = 133//

3) Let's compute the Euler Totient

$\Phi = (p - 1)(q - 1)$
$\Phi = (7 - 1)(19 - 1)$
$= 6 * 18$
$= 108$ //

What is Euler Totient ? It counts the number of numbers between 1 and n that have no common factors with n except 1. For example , if n = 10 , then the number less than or equal to 10 that are relatively prime to 10 are 1 , 3 7 , and 9 , therefore , $\varphi(10) = 4$. In cryptography , Euler's totient function is often used in RSA algorithm to generate public and private keys. The function is used to determine the number of possible values that can be used for the public and private keys , which is related to the security of the encryption.

4) Choose a small number , e co prime to Φ

E is co prime to Φ , means that the largest number that can exactly divide both e and Φ (Their greatest common divisor or gcd) is 1. Euclud's algorithm is used to find the gcd of two numbers , but the details are omitted.

When we say that E is co-prime to Φ , we mean that E and Φ do not have any common factor other than 1. This is important condition for the RSA algorithm because it ensures that the encryption and decryption keys are unique and that the system is secure.

e = 2 == > gcd (e , 108) = 2 [no]
e = 3 == > gcd (e , 108) = 3 [no]
e = 4 == > gcd (e , 108) = 4 [no]
e = 5 == > gcd (e , 108) = 1 [Yes !]

How to find the greatest common divisor :- The GCD of two integers is the largest positive integer that divides both numbers without leaving a remainder.

5) Find d , such that de % Φ = 1

This is equivalent to finding d which satisfies de = 1 + k*Φ , where k is any integer. We can rewrite this as d = (1 + k*Φ)/e. Now we work through values of k until an integer solution for e is found :-

k = 0 == > d = 1/ 5 [no]
k = 1 == > d = 109 / 5 [no]
k = 2 == > d = 217 / 5 [no]
k = 3 == > d = 325 / 5 = 65 (yes)

The public key is [n , e] and the private key is [n , d] , so the public key [133 , 5] and the private key is [133 , 65]
B) Encryption Process :- The message must be a number less than the smaller of p and q. However , at this point we don't know p or q , so in practice a lower bound on p and q must be published. This can be somewhat below their true value and so isn't major security concern. For this example , let's use the message "6".

$C = P^e \bmod n$
$= 6^5 \bmod 133$
$= 7776 \bmod 133$
$= 62$ //

C) Decryption Process :- This works very much like encryption , but involves a larger exponential which is broken down in to several steps.

$P = C^d \bmod n$
$= 62^{65} \bmod 133$
$= 62 * 62^{64} \bmod 133$
$= 62 * (62\ 2)32 \bmod 133$
$= 62 * (\ 3844\ ) 32 \bmod 133$
$= 6$ //

Example :- In an RSA cryptosystem , a particular A uses two prime numbers , 13 and 17 , to generate the public and private keys. If the public of A is 35 then the private key of A is …… ?

Step 1 :- In the first step , select two large prime numbers , p and q.

p = 13
q = 17

step 2 :- Multiply these numbers to find n = p * q , where n is called the modulus for encryption and decryption. First we calculate ……

n = p * q
n = 13 * 17
n = 221 //

Step 3 :- Choose a number e , such that n is relatively prime to (p-1) * (q – 1). It means that e and (p-1) * (q – 1) have no common factor except 1. Choose "e" such that $1 < e < \varphi(n)$, e is prime to φ (n) , gcd (e,φ (n))= 1

Second , we calculate
φ (n) = (p – 1) * (q – 1)
φ (n) = (13 – 1) * (17 – 1)
φ (n) = 192
g.c.d (35 , 192) = 1 //

Step 4 :- To determine the private key , we use the following to calculate the d such that ….

calculate :- d = (1 + k.φ (n))/e [let k =0,1, 2 , 3]
put k = 0 ==> (1 + 0 * 192) / 35 [No]
put k = 1 ==> (1 + 1 * 192) / 35 [No]
put k = 2 ==> (1 + 2 * 192) / 35 = 11 [Yes]
The private key is <d , n> = (11 , 221) , hence , private key that is d = 11

Note :- In the RSA encryption algorithm, the private key consists of two parts: the private exponent (d) and the modulus (n).

Example :- A RSA crypto system uses two prime numbers 3 and 13 to generate the public key = 3 and the private key = 7 , what is the value of cipher text for the plain text 5?

Step 1 :- In the firsts step , select two large prime numbers , p and q.

p = 3
q = 13

step 2 :- Multiply these numbers to find n = p * q , where n is called the modulus for encryption and decryption. First we calculate :-

n = p * q
n = 3 * 13
n = 39 //

step 3 :- If n = p * q , then the public key is <e , n>. A plain text message m is encrypted using public key <e , n>. Thus the public key is <e , n> = (3 , 39). To find cipher text from the plain text following formula is used to get cipher text C.

$C = m^e \bmod n$
$C = 5^3 \bmod 39$
C = 125 mod 39
C = 8 //

Example :- A RSA crypto system uses two prime numbers , 3 and 11 , to generate private key = 7 , what is the value of cipher text for a plain text 5, using the RSA public key encryption algorithm ?

Step 1 :- In the first step , select two large prime numbers , p and q

p = 3
q = 11

step 2 :- Multiply these numbers to find n = p * q , where n Is called the modulus for encryption and decryption.
First we calculate ,
n = p * q
n = 3 * 11
n = 33 //

Step 3 :- Choose a number "e" less that n , such that n is relatively prime to (p – 1) * (q – 1). It means that e and (p – 1) * (q – 1) have no common factor except 1. Choose "e" such that 1 < e < φ (n) , e is prime to φ (n), gcd (e , d(n)) = 1

Second we calculate :-

φ (n) = (p – 1) * (q – 1)
φ (n) = (3 – 1) * (11 – 1)
φ (n) = 2 * 10
φ (n) = 20 //

Step 4 :- To determine the public key , we use the following formulat to calcuate the d such that :-

calculate d* e = (1 + k. φ (n)) [let k =0, 1, 2, 3]

put k = 0 ==> e = (1 + 0 * 20) / 7 == > 1 / 7 [No]
put k = 1 ==> e = (1 + 1 * 20) / 7 == > 21 / 7 , e = 3 //

The public key is <e , n> = (3 , 33) , hence , public key , I.e e = 3 //

CHAPTER FOUR
NETWORK SECURITY

WHAT CAN A BAD GUY DO ON A NETWORKING

Eavesdropping, also known as intercepting messages, is the act of secretly listening to or intercepting private communications such as emails, phone calls, or data transmissions. An attacker can eavesdrop on network traffic by intercepting packets and decoding the data contained within.

Inserting messages into a connection, also known as a man-in-the-middle (MITM) attack, is when an attacker intercepts a communication between two parties and injects their own data into the conversation. This can allow the attacker to modify or redirect the communication without either party knowing.

Impersonation, also known as spoofing, is the act of disguising oneself as someone or something else. In networking, an attacker can impersonate another user or device by spoofing their IP address or other identifying information. This can allow the attacker to gain unauthorized access or perform malicious actions.

Session hijacking is when an attacker takes over an ongoing communication between two parties by inserting themselves in place of one of the parties. This is typically done by stealing a valid session ID or session token, which allows the attacker to assume the identity of the legitimate user.

Denial of service (DoS) attacks are designed to prevent legitimate users from accessing a network, service, or resource. This is typically accomplished by flooding the target with traffic or overwhelming it with requests, causing it to become unavailable to other users. DoS attacks can be difficult to defend against as they often involve a large number of compromised systems.

Network protocols are sets of rules governing communication between devices in a network. They ensure the the data is transmitted reliably and securely between network devices. Some common network protocols include TCP/IP , HTTP , FTP , DNS and SMTP.

Computer security involves protecting networks , devices , and data from unauthorized access , theft or damage. Security measure include firewalls , intrusion detection systems , encryption and access control mechanisms.

Network protocols can be vulnerable to security threats such as eavesdropping , data tampering , and denial of service attacks. These threats can compromise the confidentiality , integrity , and availability of data on a network.

To protect against theses threats , network security protocols are designed to ensure the secure transmission of data over a network. For example , the Transport Layer Security (TLS) protocol provides end to end encryption for data transmitted over the internet. Similarly , the secure shell (SSH) protocol is used to securely log in to remote computers.

Security measures for network protocols also include access control mechanisms to ensure that only authorized uses can access data on a network. This can include user authentication , password detection and role based access control.

Network protocols and computer security are closely interwined. Network protocols provide the foundation for communication and data transfer between devices , while security measures ensure the confidentiality , integrity , and availability of data on a network. Effective network security measures include encryption , access control mechansims and intrusion detection systems to protect against security threats.

Attacks on the TCP/IP protocols ?

Attacks on TCP/IP protocols refer to the various techniques used by attackers to exploit vulnerabilities in the TCP/IP protocols. The TCP/IP protocols are a set of communication protocols used for transmitting data over networks , including the internet. Since they are the back bone of the internet and other computer networks , attackers constantly look for ways to

exploit vulnerabilities in these protocols to compromise network security.

Data - Link Layer → ARP Spoofing :-

The Address Resolution Protocol (ARP) is a protocol used to map a network address (such as an IP address) to a physical address (such as a MAC address). It's main job is to allow network devices to communicate with each other over Ethernet or other physical networks. When a device wants to communicate with another device on the network , it first checks its ARP cache to see if it already has the physical address of the device. If not , it sends an ARP request packet to the network , asking for the physical address associated with the IP address. The device with that IP address responds with its physical address , and the requesting device adds that information to its ARP cache. This process is known as ARP resolution.

ARP is a stateless protocol , meaning it does not require any prior communication or negotiation between devices before sending ARP packets. This simplicity makes it vulnerable to attacks such as ARP spoofing , where an attacker sends falsified ARP messages in order to associate their own MAC address with the IP address of a legitimate device on the network , once this association is made , the attacker can intercept , modify or redirect network traffic intended for the legitimate device.

In computer networking , every device on a network has a unique identifier known as a Media Access Control (MAC) address. This is a hardware address that identifies the device's network interface card (NIC). Similarly , every device on a network also has an IP (Internet Protocol) address , which is a logical address assigned to the device's network interface for communication on the network for communication on the network.

ARP spoofing is a type of cyber attack where an attacker sends fake ARP (Address Resolution Protocol) messages on a local area network (LAN) in order to associate their MAC (Media Access Network) address with the IP address of a another host in a legitimate network device. This allows the attacker to intercept and modify network traffic , perform a man in the middle attack and steal sensitive information.

In an ARP spoffing attack , the attacker typically sends a series of ARP messages to the target network , with the goal of associating their own MAC address with the IP addresses of the network's gateway or some other critical device such as a server. Once the attack is successful , the attacker can intercept and modify network traffic , steal data such as passwords or credit card numbers and potentially launch other attacks on the network.

How does ARP spoofing works ?

ARP Spoofing is a type of attack that exploits the weakness in the Address Resolution Protocol (ARP) to associate the attacker's Media Access Control (MAC) address with the IP address of a legitimate network device. This allows the attacker to intercept or modify the network traffic intended for the target device.

One way the attacker can use this information is to intercept the network traffic meant for the legitimate device by redirecting it to their own computer. The attacker can the modify the network traffic and send it on to its intended destination without the sender or receiver being aware of the interception.

Another way the attacker can use ARP spoofing is to launch a denial of service attack by flooding the network with ARP packets. This can cause the ARP cache of legitimate devices to become saturated with false entries , making It difficult for them to communicate on the network.

The weakness in the ARP protocol that allows ARP spoofing to occur is due to the stateless nature of the protocol. This means that ARP requests and replies are sent out without any authentication or verification of the source. Additionally , ARP caches are automatically updated with any replies received , regardless of weather the reply was solicited or not.

To protect against ARP spoofing attacks , network administrators can implement various measures such as using static ARP tables , implementing ARP spoofing detection software and configuring network devices to only accept ARP replies from trusted sources.
ARP spoofing is a type of data link layer threat , it involves manipulating the ARP protocol at the data link layer in order to perform a variety of

attacks such as Man in the Middle attacks or Denial of service attacks. The Spoofed ARP messages are used to deceive the victim devices in to sending traffic to the attacker , instead of the intended destination , allowing the attacker to intercept and potentially manipulate the traffic. Therefore , ARP spoofing is a serious security threat that needs to be addressed in network security.

NETWORK LAYER SECURITY :- IPSEC

Ipsec (Internet Protocol Security) is a set of protocols and standards that provides secure communication over IP networks , including the internet , it is used to ensure confidentiality , integrity , and authenticity of IP packets , and to protect against various network attacks such as eavesdropping , tampering and replay attacks.

Ipsec operates at the network layer of the OSI model and provides security services for IP packets , including encryption , authentication and key management. It works by creating a secure tunnel between two devices over an insecure network , such as the internet.

The three primary services provided by Ipsec are origin authentication , confidentiality and key management. **Origin authentication** guarantees that the received packet was transmitted by the party that claims to be the source of the packet has not been tampered with during transit. To provide origin authentication , IP sec inserts an authentication header (AH) in to the packet. AH provides message integrity and anti-replay services as well.

Confidentiality , on the other hand , encrypts messages to prevent unauthorized parties from reading them. To provide confidentiality , IP sec inserts an Encapsulated security payload (ESP) header in to the packet. ESP can also provide origin authentication and message integrity in addition to confidentiality.

Key management is the process of securely exchanging keys between communicating parties. Ipsec uses a protocol called Internet Key Exchange (IKE) to exchange keys. IKE provides a secure way for two parties to establish a shared secret key over an insecure network.

Ipsec (Internet Protocol Security) is a set of security algorithms and a framework that provide secure communication between two entities over the internet. The Ipsec protocol provides encryption and authentication services , which make it an ideal solution for securing communications across a LAN , WAN , or the internet.

One of the most significant benefits of Ipsec is its ability to provide secure branch office connectivity over the internet. This allows businesses to connect their remote offices securely over the internet , without the need for expensive leased lines or other costly infrastructure. By using IPsec , business can establish a secure virtual private network (VPN) over the internet or a public WAN , which enables secure communication between their remote offices.

Ipsec can also be used to provide secure remote access over the internet. This is particularly useful for businesses with remote employees who need to access their corporate network securely. With Ipsec , employees can securely access their organization's network from anywhere in the world , using a VPN client that encrypts their traffic and provides secure authentication.

Another use case for Ipsec is establishing intranet connectivity with partners. Ipsec can be used to secure communication with other organizations , ensuring authentication and confidentiality , and providing a key exchange mechanism. This makes it an idea solution for businesses that need to communicate securely with partners , suppliers or customers.

Finally , Ipsec is also useful for enhancing electronic commerce security , while some web and e-commerce applications have built in security protocols , the use of Ipsec can help prevent eavesdropping , tampering and other attacks that could compromise sensitive data during electronic transactions.

IPSec provides the capability to secure communications across a LAN , across private and public WANS and across the internet.

IPsec (Internet Protocol Security) is a set of protocols and standards that provide security for Internet Protocol (IP) communication by encrypting and authenticating IP packets.

IPsec provides the capability to secure communications across a LAN, across private and public WANs, and across the Internet.

IPsec is often used for secure branch office connectivity over the Internet, providing a secure virtual private network (VPN) over a public WAN. This allows branch offices to connect securely to the main office over the Internet, without the need for expensive dedicated lines or leased circuits.

IPsec can also be used for secure remote access over the Internet, enabling users to securely connect to a network from a remote location.

In addition, IPsec can be used for establishing intranet connectivity with partners. This enables organizations to securely communicate with other organizations, ensuring authentication and confidentiality, and providing a key exchange mechanism to prevent unauthorized access.

IPsec also enhances electronic commerce security by providing an additional layer of security to Web and electronic commerce applications that have built-in security protocols.

IPsec can be used to encrypt and authenticate IP packets, ensuring that data transmitted over the Internet is secure and cannot be intercepted or tampered with.

In addition to its role in providing security for end users and protecting premises systems and networks, IPsec also plays a role in routing. Specifically, it provides authentication and integrity protection for routing messages to ensure that they are not forged or tampered with.

When a new router advertises its presence on a network, it sends a router advertisement message to notify other devices on the network. IPsec ensures that this message comes from an authorized router, preventing unauthorized devices from claiming to be a router and potentially disrupting the network.

When a router seeks to establish or maintain a neighbor relationship with a router in another routing domain, it sends a neighbor advertisement message. Again, IPsec ensures that this message comes from an authorized router, preventing unauthorized devices from establishing unauthorized neighbor relationships and potentially disrupting the network.

When a router sends a redirect message to another device, it is directing the device to send its traffic to a different router. IPsec ensures that this message comes from the correct router and prevents unauthorized devices from sending fraudulent redirect messages that could redirect traffic to unauthorized destinations.

Finally, IPsec provides authentication and integrity protection for routing updates, ensuring that they are not forged or tampered with. This prevents attackers from disrupting the routing of traffic by sending fraudulent routing updates.

Security Gateway

When we send a message over the internet, it passes through many different computers before it reaches its final destination. Sometimes, there are special computers called security gateways that help protect our message from being seen or changed by other people.

A security gateway is like a gatekeeper that checks to make sure that our message is safe and secure as it passes through. It has special tools called IPsec mechanisms that help keep our message private and protected.

An IPsec mechanism is like a secret code that encrypts our message so that no one can read it except for the person we're sending it to. It also makes sure that our message hasn't been changed or tampered with during its journey.

A security gateway can be a router or gateway, which are special kinds of computers that help connect different networks together. When a message passes through a security gateway, the gateway checks to make sure that the message is safe and secure before allowing it to continue on its journey.

Transport Layer Security

A TCP SYN attack , also known as a SYN flood attack , is a type of denial of service (DOS) attack that exploits a weakness in the TCP/IP protocol. The attack works by sending a large number of TCP connection requests with spoofed IP addresses to a target server , flooding It with traffic and overwhelming its ability to respond to legitimate requests.

In a normal TCP three-way handshake , when a client sends a SYN message to initiate a connection , the server responds with a SYN-ACK message to acknowledge the request. The client then sends an ACK message to confirm the connection. In a SYN flood attack , the attacker sends a large number of SYN messages to those addresses. When the server does not receive a response to the SYN-ACK messages , it keeps the connection half-open , trying up system resources and eventually causing the server to become unresponsive.

A TCP SYN flood attack is a type of denial of service (DoS) attack that targets a server by exploiting the normal TCP three-way handshake process between a client and server. The goal of this attack is to consume the resources of the targeted server and make it unavailable to legitimate users.

In a normal TCP three-way-handshake , when a client sends a SYN (message) to initiate a connection , the server responds with a SYN-ACK message to acknowledge the request. The client then sends an ACK message to confirm the connection. In a SYN flood attack , the attacker sends a large number of SYN messages with spoofed IP addresses , causing the target server to send SYN-ACK messages to those addresses. When the server does not receive a response to the SYN-ACK messages , it keeps the connection half-open , tying up system resources and eventually causing the server to become unresponsive.

A TCP SYN flood attack is a type of DDoS attack in which an attacker sends repeated requests to connect to a server on every port , often using a fake IP address. The server receives these requests , thinking they are legitimate connection requests and responds with a SYN-ACK packet. However , the attacker does not respond with the expected ACK , or if the IP address is fake , the attacker never receives the SYN-ACK in the first place. This results in the server waiting for acknowledgment of its SYN-ACK packet for some time , which ties up its resources and can ultimately result in denial of service to legitimate users. The attacker keeps sending the requests faster than the server can process them , causing network overload and disrupting service.

A TCP SYN flood attack is a type of Distributed Denial Of Service (DDoS) attack that exploits a weakness in the way the Transmission Control Protocol (TCP) works. The attack floods the target server with a large number of TCP SYN packets , overwhelming the server and causing it to become unresponsive.

When two devices such as a client and server , establish a TCP connection , they use a three-way handshake process. The client sends a SYN packet to request a connection with the server , the server responds with a SYN-ACK packet to acknowledge the request , and the client sends an ACK packet to confirm the connection , during these process , the server set aside resources to handle the connection until it is closed.

In a SYN flood attack , the attacker sends a large number of SYN packets to the target sever, often using a fake IP address to mask their identity. The server , thinking that it is responding to legitimate connection requests , sends SYN-ACK packets back to the source IP address of each coming SYN packet. However , since the attacker either does not responds or never receives the SYN-ACK packets , the server is left waiting for a response that never comes.

The server sets aside resources to handle each incoming connection request , and in a SYN flood attack , these resources quickly become exhausted as the server is flooded with more connection requests than it can handle. As a result , the server becomes unresponsive and may even crash, making it impossible for legitimate users to connect to the server to access the services it provides.

TCP SYN attacks can be difficult to detect and prevent , as they appear to be legitimate traffic at first , however , network administrators can use a variety of strategies to mitigate the effects of SYN floods , such as implementing firewalls or

intrusion detection systems . Limiting the rate of incoming traffic , or using SYN cookies to prevent the server from keeping connection open for too long.

WEB SECURITY

The web , also known as the world wide web (www) is a client / server application running over the internet or TCP/IP intranet. It is a vast collection of web pages and other digital content that is accessed via the internet using web browsers such as Google Chrome , Mozilla Firefox , or Microsoft edge. The popularity of the web has grown exponentially since its inception , and it has become a vital tool for business and individuals alike. However , this popularity has made it target for attacker who seek to exploit vulnerabilities in web servers and the underlying software.

The web presents new challenges that are not well appreciated in the context of computer and network security. The web is a visible outlet for corporate and business transactions that can lead to damages and losses. If the web servers are subverted , reputations can be damaged , and money can be lost. Web servers are easy to configure and web content is easy to develop and manage , but the underlying software is getting extraordinarily complex , which may hide many potential security flaws.
Web servers can be exploited as a launching pad to attack corporate data systems , as users are usually not aware of the risks. Attackers can use the web to install malware on user systems , steal sensitive data or use the web server as a means to launch further attacks on another systems. This can lead to serious consequences for businesses , including financial losses , reputation damage and legal liability.

One of the primary challenges with web security is that it is a constantly evolving landscape , and attackers are always finding new vulnerabilities to exploit. Web developers must be vigilant in keeping up with the latest security threats and ensuring that their web applications are designed with security in mind. This includes using secure coding practices , encrypting sensitive data and regularly testing for vulnerabilities.

Web security also requires a multi-layered approach , including network security , server security , and application security. Network security involves securing the web server and other network devices to prevent unauthorized access. Server security involves securing the web server operating system , database server , and other server components. Application security involves ensuring that web applications are designed with security in mind and are tested thoroughly for vulnerabilities.

In summary , the popularity of the web has made it a prime target for attackers and web security is a complex and ever-evolving landscape. It requires a multi-layered approach including network security , server security and application security to mitigate the risks and ensure the safety of web servers and businesses.

The Web (WWW) is a client/server application that runs over the internet or TCP/IP intranet , it presents new challenges for computer and network security that are not well understood. There are several types of web threats that can compromise the integrity , confidentiality and authentication of web data.

Integrity :- Data , memory and/or message modification and Trojan horse browser are two common forms of attacks that can compromise the integrity of web data. Cryptographic check sums can be used to prevent these attacks by providing a digital signature of the web data. This signature can be compared to a previously calculated value to detect any modifications to the data.

Confidentiality :-, Theft of data from client and information from server , access to information about network configuration and access to information about which client is communicating are all examples of attacks that can compromise the confidentiality of web data. Encryption can be used to prevent these attacks by verifying the identity of the user or server and ensuring the integrity of the web data.

Authentication :- Impersonation of legitimate users and data forgery are two common forms of attacks that can compromise the authentication of web data. Cryptographic techniques such as digital certificates and public key infrastructure (PKI) can be used to prevent these attacks by verifying the identity of the user or server and ensuring the integrity of the web data.

Counter measures :- To protect against these web threats , several counter measures can be employed , including firewalls , intrusion detection systems (IDS) , intrusion prevention systems (IPS) , anti-virus software and access control mechanisms. It is also important to keep software up to date and apply security patches as soon as they become available.

In summary , web threats are significant risk to the integrity , confidentiality , and authentication of web data. Cryptographic techniques such as encryption , digital signatures , and PKI , as well as monitoring and detection tools , can be used to prevent these threats and protect against attacks. It Is important to remain vigilant and implement best practices for web security to ensure the safety for sensitive data and prevent financial and reputational damages.

When it comes to web security, there are various types of threats that can be faced by users. These threats can be classified based on their location in the web architecture. The three main areas of focus for web security are the web server, the web browser, and the network traffic between the browser and the server.

1. **Web Server Security:** The web server is responsible for hosting web pages and serving them to users who access them via the internet. Therefore, web server security is an important aspect of web security. The various types of threats that can be faced by a web server include attacks such as denial of service attacks, injection attacks.
2. **Web Browser Security:** Web browsers are the primary means by which users interact with the web. Therefore, web browser security is an essential aspect of web security. The various types of threats that can be faced by a web browser include attacks such as malware downloads, phishing attacks, man-in-the-middle attacks.
3. **Network Traffic Security:** Network traffic security is essential because it ensures the secure transmission of data between the web server and the browser. The various types of threats that can be faced by network traffic include eavesdropping, packet sniffing, man-in-the-middle attacks, and many more. Network traffic security is usually managed by implementing security protocols such as SSL, TLS, or HTTPS. These protocols ensure secure communication between the server and the client, and prevent unauthorized access or tampering with the data.

There are three standardized schemes that are becoming increasingly important as part of Web commerce and that focus on security at the transport layer: SSL/TLS, HTTPS, and SSH

When we use the internet, we often share information like our passwords, credit card numbers, and other personal details. To keep this information safe, we use special tools and technologies to encrypt, or scramble, our data so that no one else can read it.

One such technology is SSL/TLS. SSL (Secure Sockets Layer) and its successor, TLS (Transport Layer Security), are protocols that provide secure communication between a client and a server over the internet. These protocols are used to encrypt data sent between a client (like a web browser) and a server (like a website).

SSL/TLS provides several security services to protect your data. For example, it uses symmetric encryption to scramble your data so that it cannot be read by anyone except the intended recipient. It also uses a message authentication code to ensure that the data has not been tampered with during transmission.

In addition to these security services, SSL/TLS includes protocol mechanisms that allow two parties to establish a secure communication channel and agree on the specific security mechanisms and services they will use.

HTTPS (HTTP over SSL) is a combination of HTTP (Hypertext Transfer Protocol), which is the standard protocol used for browsing the web, and SSL/TLS. When you visit a website that uses

HTTPS, your browser and the website use SSL/TLS to encrypt your data and protect your privacy.

When you see "HTTPS" in the web address of a website, it means that the website is using SSL/TLS to encrypt your data and protect your privacy. HTTPS is widely used for secure online transactions, such as online banking and shopping, and is becoming increasingly important for protecting privacy and security on the web.

Secure Shell (SSH) is a technology that provides secure remote access to another computer or server over the internet. It allows you to securely connect to a remote computer or server and perform various tasks, such as file transfers, remote login, and running remote applications.

SSH is like a secret password that allows us to safely and securely connect to another computer or server over the internet. When we use SSH, we enter a special password that only we know, and this password allows us to access the other computer or server.

Once we're connected, we can do things like transfer files, run programs, and even log into the other computer or server as if we were sitting right in front of it!

But why is SSH so important? Well, the internet can be a dangerous place, and there are many people who want to steal our information or damage our computer. SSH helps protect us from these bad guys by using special tools to keep our connection safe and secure.

So, in short, SSH is a secret password that allows us to safely and securely connect to another computer or server over the internet. It's like having a secret key to unlock the door to another computer or server, and it helps keep us safe from bad guys who might try to steal our information or damage our computer

SECURITY SOCKET LAYER (SEL)

When we use the internet to visit websites, we want to make sure that our information and communication is safe and private. That's where SSL comes in!

SSL is like a secret code that protects our information when we send it over the internet. It has two different layers of protocols that work together to keep our information safe.

The first layer is called the SSL Record Protocol Layer. This layer is like a special envelope that holds our information and makes sure that it's protected as it travels over the internet.

The second layer is made up of three different parts: the SSL Handshake, the SSL Change Cipher Spec, and the SSL Alert.

The SSL Handshake is like a secret handshake that happens between our computer and the website we're visiting. It makes sure that we're talking to the right website and that the website is talking to the right computer. It also helps us agree on how we're going to protect our information with a secret code called a cryptographic key.

One of these parts is called the SSL Change Cipher Spec. This part of SSL is like a special signal that tells our computer to start using a secret code called a cryptographic key to protect our information.

Sometimes, when we're using SSL to visit a website, there might be a delay or a problem with the connection. This delay or problem can cause a backlog of information that needs to be protected with the cryptographic key.

The SSL Alert is like a special message that tells us if something goes wrong with our SSL connection. For example, if someone tries to steal our information or if there's a problem with the SSL code, the SSL Alert will warn us and tell us what to do next.

SSL Handshake Protocol Action

The SSL Handshake Protocol is like a special conversation that happens between our computer and the website we're visiting. It makes sure that

we're talking to the right website and that the website is talking to the right computer. It also helps us agree on how we're going to protect our information with a secret code called a cryptographic key.

During the SSL Handshake Protocol, there are a few different actions that happen. First, the server and the client authenticate each other. This means that they check to make sure that they're both who they say they are. This helps protect us from bad guys who might try to pretend to be someone else and steal our information.

Next, the server and the client negotiate on encryption, MAC (Message Authentication Code) algorithm, and cryptographic keys. This means that they agree on how they're going to protect our information with a secret code that only they know. This helps make sure that our information stays safe and private as it travels over the internet.

Finally, the SSL Handshake Protocol is used before any application data is transmitted. This means that the SSL Handshake Protocol happens before any of our personal information, like our passwords or credit card numbers, are sent over the internet. This helps make sure that our information stays protected right from the very beginning of our interaction with the website.

Security Enhanced Application Protocols

When we use the internet to send and receive information, we want to make sure that our information stays safe and private. That's why we use different security protocols to protect our information.

One type of security protocol is called a Security-Enhanced Application Protocol (SEAP). SEAPs are special protocols that are designed to enhance the security of different application layer protocols. They add extra security features to these protocols to help protect our information when we use them.

There are several examples of SEAPs that are commonly used on the internet:

- FTPS: This is a SEAP for FTP (File Transfer Protocol). It adds extra security features to FTP, such as encryption and authentication, to help protect our files when we transfer them over the internet.
- HTTPS: This is a SEAP for HTTP (Hypertext Transfer Protocol). It adds extra security features to HTTP, such as encryption and authentication, to help protect our information when we browse websites or submit forms online.
- SMTPS: This is a SEAP for SMTP (Simple Mail Transfer Protocol). It adds extra security features to SMTP, such as encryption and authentication, to help protect our emails when we send them over the internet.
- DNSSEC: This is a SEAP for DNS (Domain Name System). It adds extra security features to DNS, such as digital signatures and validation, to help protect our DNS queries and responses from being intercepted or modified by bad actors.

SECURITY ELECTRONIC TRANSACTION (SET)

Security in E-commerece (Electronic Payment)

When we buy things online, we need a way to pay for them. This is where electronic payment systems come in.

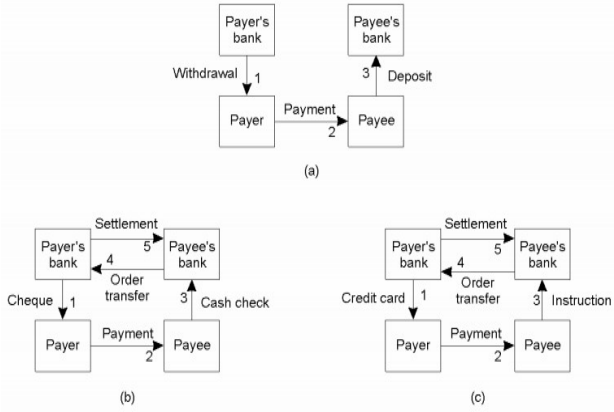
Electronic payment systems are like special computer programs that allow us to pay for things online. When we use an electronic payment system, there are usually three main players involved: the customer (that's us), the merchant (that's the online store we're buying from), and often banks (that help process the payment).

There are different types of electronic payment systems that we can use to pay for things online. These include:

- **Cash**: Some electronic payment systems allow us to pay for things online using

cash. For example, we might be able to use a service like Western Union to send cash to the merchant, who will then send us the product.

- **Check**: Some electronic payment systems allow us to pay for things online using a check. For example, we might be able to use a service like PayPal to send a check to the merchant, who will then send us the product.
- **Credit card:** One of the most common electronic payment systems is the use of credit cards. When we use a credit card to pay for something online, we enter our credit card information into a secure form on the merchant's website. The merchant then sends the credit card information to a bank, which checks to make sure we have enough money in our account to pay for the purchase. If everything checks out, the bank sends the money to the merchant, who then sends us the product.



When we use electronic payment systems to buy things online, we want to make sure that our payment information stays safe and secure.

In cash based systems (using ATM), the main issue is authentication

- Use of magnetic card
- PIN

Credit card or check based system

- No tampering/alteration
- Protection against repudiation (the buyer denies having made the order)

There are several security requirements that must be in place in order to ensure safe electronic payments. These are:

1. **Authentication**: This means verifying the identity of the person making the payment. In cash-based systems, this might involve using an ATM card and PIN to authenticate the user. In credit card-based systems, this might involve verifying the user's identity with their credit card number and billing address.
2. **Encryption**: This means protecting the payment information by using encryption algorithms to scramble the data so that it can only be read by authorized parties. Encryption helps to prevent eavesdropping and other forms of data theft.
3. **Integrity**: This means making sure that the payment information has not been tampered with or altered in any way. Integrity checks are typically done using digital signatures or other cryptographic techniques.
4. **Non-repudiation**: This means ensuring that the person making the payment cannot later deny that they made the payment. Non-repudiation measures typically involve using digital signatures or other forms of authentication to prove that the user authorized the payment.

Overall, electronic payment systems are critical to the success of e-commerce, and security is a key factor in ensuring their success. By implementing authentication, encryption, integrity, and non-repudiation measures, we can help to ensure that our payment information stays safe and secure when we buy things online.

When we use the internet to buy things, we want to make sure that our payment information stays safe and private. One way to do this is to use a security protocol called Secure Sockets Layer (SSL).

SSL is a protocol that is used by most major web browsers to create a secure channel between the consumer (that's us) and the merchant (that's the online store we're buying from). This secure channel helps to protect our payment information from eavesdroppers and other bad actors who might try to steal it.

However, SSL is not always enough to protect us from all types of online fraud. For example, some dishonest merchants might set up illegal websites and claim to be a legitimate business in order to collect our credit card numbers for personal use. Alternatively, some customers might try to use invalid credit card numbers to buy things online, which can cause problems for the merchant.

SET (Secure Electronic Transaction) is an example of an application of cryptography. It was developed by Visa and MasterCard, with involvement from other companies such as IBM, Microsoft, Netscape, RSA, Terisa, and Verisign. SET is designed to protect credit card transactions on the internet.

SET is not a payment system itself, but rather a security protocol that enables users to securely make credit card transactions over an open network like the internet. It uses encryption and other security techniques to ensure that credit card information is protected from unauthorized access or tampering.

One of the key features of SET is that it is an open encryption and security specification. This means that the entire protocol is published and available for anyone to see and analyze. This makes it easier for security experts to review the protocol and identify any potential vulnerabilities or weaknesses.

SET FEATURE AND BUSINESS REQUIREMENT

- Provide confidentiality of payment and ordering information
- Information made available only when and where necessary (privacy)
-)Ensure the integrity of all transmitted data
-)Provide authentication that a cardholder is a legitimate user of a credit card account
- Provide authentication that a merchant can accept credit card transactions through its relationship with a financial institution
- All parties must have digital certificates (trust)
-)Provides a secure communication channel in a transaction

SET PARTICIPANTS

In the SET (Secure Electronic Transaction) protocol, there are several different participants who play a role in the transaction:

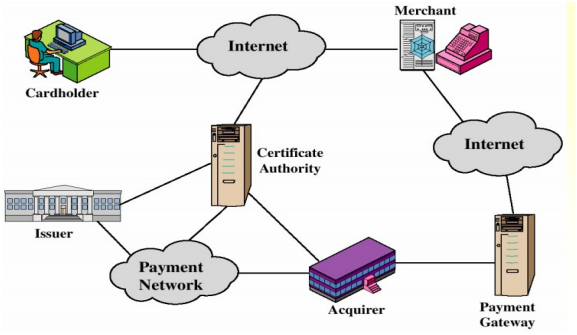
1. **Cardholder**: This is the authorized holder of the payment card, also known as the customer. The cardholder is the individual who wants to purchase goods or services from the merchant.
2. **Merchant**: The merchant is the individual or organization that has goods or services to sell to the cardholder. In the context of SET, the merchant is also known as the web server.
3. **Issuer**: The issuer is the financial institution that issued the payment card to the cardholder. This is typically the cardholder's bank.
4. **Acquirer**: The acquirer is the financial institution that verifies that a card account is active and that the proposed purchase does not exceed the credit limit. The acquirer is connected with the merchant.
5. **Payment gateway**: The payment gateway is operated by the acquirer or a designated third party. The payment gateway processes merchant payment

messages in order to facilitate the transaction.

6. **Certificate Authority (CA)**: The certificate authority is a trusted entity that issues X.509v3 public key certificates for cardholders, merchants, and payment gateways. **The success of SET depends on the CA, as these certificates are used to establish trust between the different participants in the transaction.**

These different participants work together to facilitate secure electronic transactions. The cardholder wants to purchase goods or services from the merchant, and the issuer and acquirer help to verify that the transaction is legitimate and within the cardholder's credit limit. The payment gateway processes the transaction, and the certificate authority issues the necessary certificates to establish trust between the different participants.

Both cardholders and merchants must register with CA first, before they can buy or sell on the Internet, i.e., The customer opens an account and receives a certificate; the Merchants have their own certificates



SEQUENCE EVENTS FOR TRANSACTION IN SET

Customer browses a website and decides what to purchase: The customer visits a website, selects the items they want to purchase, and proceeds to the checkout page to enter their payment information.

1. Customer sends order and payment information: The customer enters their payment information, which includes two parts in one message. The first part is the purchase order, which is for the merchant. The second part is the card information, which is for the merchant's bank only. The message is sent securely to the merchant.
2. Merchant forwards card information to its bank: The merchant separates the card information from the purchase order and forwards it to its bank. The purchase order is kept by the merchant.
3. Merchant's bank checks with issuer for payment authorization: The merchant's bank checks with the issuer (the financial institution that issued the customer's credit card) to verify that the card is valid and has sufficient funds for the purchase.
4. Issuer sends authorization to merchant's bank: If the issuer approves the transaction, it sends an authorization to the merchant's bank.
5. Merchant's bank sends authorization to merchant: The merchant's bank sends the authorization to the merchant, allowing them to complete the order.
6. Merchant completes the order and sends confirmation to the customer: The merchant completes the purchase order, packages and ships the items, and sends a confirmation message to the customer.
7. Merchant captures the transaction from its bank: The merchant captures the transaction from its bank, meaning that the funds are transferred from the customer's account to the merchant's account.
8. Issuer prints credit card bill (invoice) to customer: The issuer generates a credit card bill (invoice) for the customer, which includes the transaction details and the total amount due. The customer receives the bill and must pay the balance by the due date to avoid interest charges.

This sequence of events provides a secure and reliable way for customers to purchase goods and services online while protecting their sensitive payment information. SET employs cryptographic techniques to ensure the confidentiality, integrity, and authenticity of the payment transaction.

Application Layer Security

The application layer is responsible for providing services to the end user. In the context of security , application layer security is concerned with protecting applications that users interact with from various types of attacks.

Application layer security refers to the measures and techniques used to secure the application layer of a computer network , which is the layer responsible for providing end user services and applications. This layer includes web applications , email clients , file sharing applications , and other software that allows users to interact with the network.

DNS spoofing is a type of attack that can be used to redirect users to fake websites. If e attacker gains access to a name server , they can modify it so that it gives false information. This can be used to redirect users to the attackers own website or to steal their login credentials.

Web browsers can also pose a threat to application layer security. Most browsers are obtained online and can potentially contain malicious code that can compromise the security of the user. For example , the attacker can be informed of the activities of the user of passwords typed by the user. Browsers can also have their security downgraded , which can reduce the key length used in SSL.

What is DNS

DNS(Domain Name System) helps us to translate human readable domain names (such as www.google.com) in to IP addresses (such as 172.217.1.4) that machines can understand and use to connect to the appropriate web server. This makes it easier for users to navigate the internet using domain names instead of memorizing long strings of numbers.

What is DNS Spoofing ?

It is a type of cyber attack where an attacker modifies the DNS records in the Domain Name System (DNS) cache or on a DNS server to redirect traffic to a malicious website or IP address. This can be used to redirect users to fake websites that look like legitimate ones in order to steal sensitive information such as usernames , passwords , and credit card numbers.

When we say an attacker gains access to a name server , it means that the attacker has found a way to control or manipulate the DNS records in the Domain Name System cache or DNS server.

DNS records are used to map human readable domain names to their corresponding IP addresses. They are essentially information stored in a DNS server that contains information about a particular domain name , including IP addresses associated with it.

So in DNS spoofing , refers to the act of falsifying information in order to deceive or trick a DNS server in to believing that a domain name or IP address corresponds to a different one. This can be done by manipulating DNS records or by poisoning the DNS cache with false information. The goal of DNS spoofing is often to redirect traffic to a fake website or to intercept and manipulate communication.

Cookies

Cookies are small text files that a website saves on a user's computer or mobile device when the user visits the site. Cookies help the website to remember information about the user's visit, such as their preferred language and other settings, making the website more user-friendly and personalized.

However, cookies can also be used to track a user's online activity across different websites, which can lead to privacy violations. For example, advertisers can use cookies to track a user's browsing habits and display targeted ads based on their interests.

Server-side risks:

Interactive web sites that rely on forms and scripts can be vulnerable to attacks. By writing malicious scripts, a client (i.e., user) can exploit vulnerabilities in the server software and gain unauthorized access to the server or crash it by causing a buffer overflow.

To mitigate these risks, web developers can use secure coding practices and implement security measures such as input validation and sanitization, access control, and encryption. It's also important to keep server software and security patches up-to-date to prevent known vulnerabilities from being exploited. Additionally, web servers can use firewalls and intrusion detection/prevention systems to monitor and block suspicious traffic.

Web browsers as threats

Web browsers are essential software tools for accessing the Internet and online services. However, web browsers themselves can be a threat to user security.

When a user obtains a browser from the Internet, there is a potential for the browser to contain malicious code that can compromise the user's system. Malicious code within the browser can inform an attacker about the user's activities and passwords, leading to serious privacy violations. Additionally, the malicious code can downgrade browser security, for example, reducing the key length used in SSL, making it easier for attackers to intercept and read sensitive information.

Helper applications are used by browsers to view content retrieved from the web. These applications are external viewer programs, such as Windows Media Player, QuickTime Player, or Adobe Reader. However, these helpers can also contain Trojan horse code that can exploit vulnerabilities in the user's system. For example, downloaded data can exploit vulnerabilities of helpers, leading to the execution of malicious code on the user's computer.

Overall, users should be cautious when obtaining and using web browsers and helper applications. They should ensure that they obtain software from trusted sources and keep their software up to date with the latest security patches.

Email Security

Emails are electronic messages that are sent and received over the internet. However, during transit, emails pass through various servers, making them visible to anyone who has access to the servers. This can pose a security risk, as emails may contain sensitive or confidential information.

The Simple Mail Transfer Protocol (SMTP) is the standard protocol used for sending and receiving emails over the internet. However, SMTP has some security holes and operational limitations that can be exploited by attackers to intercept and read emails.

To address these security issues, several tools and protocols have been developed to enhance email security. Two of the most commonly used tools for email encryption and authentication are Pretty Good Privacy (PGP) and Secure Multi-Purpose Internet Mail Extension (S/MIME).

PGP is a software program that uses encryption to protect email messages and attachments from unauthorized access. PGP uses a combination of symmetric-key and public-key cryptography to encrypt and decrypt messages. With PGP, users can create their own public and private keys, which they can use to encrypt and decrypt messages. Only the intended recipient, who possesses the corresponding private key, can decrypt and read the message.

PGP (Pretty Good Privacy) is an encryption program that provides cryptographic privacy and authentication for email messages and data files. It was developed by Phil Zimmermann in 1991 and was initially distributed as freeware. PGP provides various security services, including confidentiality, integrity, authentication, and non-repudiation, which are achieved through the use of encryption and digital signatures.

PGP uses a public-key encryption method that involves two keys: a public key that can be distributed freely to anyone, and a private key that is kept secret by the owner. The public key can be used to encrypt messages that can only be decrypted by the private key owner, providing confidentiality. Similarly, a digital signature can be created using the private key that can be verified using the public key, providing authentication and integrity.

PGP also incorporates tools for developing a public-key trust model and public-key certificate management. Users can create and manage their own public key certificates or use a third-party certificate authority.

SMTP (Simple Mail Transfer Protocol) is a protocol used for sending email messages between servers. It is a basic and widely used protocol for email transmission. However, there are several limitations with SMTP that can cause problems in sending email messages.

One limitation is the inability to transmit executable files or other binary files like JPEG images. This is due to security concerns and the potential for these files to contain malicious code that can harm the recipient's computer.

SMTP also has problems with handling "national language" characters, which are non-ASCII characters. This can cause issues with email messages that contain special characters from other languages.

Messages over a certain size may also have problems being transmitted using SMTP. This is due to the limitations of email servers and the potential for large messages to cause server overload.

ASCII to EBCDIC translation problems can also occur when using SMTP, as the protocol does not support non-ASCII character sets.

Lines longer than a certain length (72 to 254 characters)

To address some of these limitations, Multipurpose Internet Mail Extension (MIME) was developed. MIME is an extension to SMTP that allows for the transmission of non-ASCII characters, binary files, and messages over a certain size. MIME also allows for the encoding of non-textual data into ASCII format for transmission over SMTP.

S/MIME is a protocol that provides encryption and digital signatures for email messages. It uses a public key infrastructure (PKI) to provide digital certificates that can be used to authenticate the sender and encrypt the message. With S/MIME, users can sign and encrypt messages, providing an additional layer of security to their email communications.

S/MIME (Secure/Multipurpose Internet Mail Extensions) is a protocol that provides security features for email messages. It is an Internet standard approach to email security that incorporates the same functionality as PGP (Pretty Good Privacy). S/MIME uses cryptographic techniques to protect email messages, such as encryption and digital signatures.

The S/MIME protocol provides several functions to secure email messages, including:

1. **Enveloped Data:** This function encrypts the content of the message and also encrypts a session key for each recipient. The session key is then used to decrypt the message.
2. **Signed Data:** This function creates a message digest (a hash of the message) and encrypts it with the sender's private key. The recipient can then use the sender's public key to verify the signature and ensure the message has not been tampered with.
3. **Clear-Signed Data:** This function signs the message but does not encrypt it. This allows the recipient to verify the signature and ensure the message has not been tampered with.
4. **Signed and Enveloped Data:** This function combines the previous two functions, signing the message digest and then

encrypting both the content of the message and the signed digest.

S/MIME provides a secure and efficient way of sending email messages, ensuring confidentiality, integrity, and authentication. It is widely used in organizations that require secure email communication, such as government agencies and financial institutions.

CHAPTER – 5
Fire Wall

A Firewall is a security device used to monitor and control traffic between a computer network and the internet or other external networks. Its primary function is to protect a network by analyzing incoming and outgoing traffic and deciding whether to allow or block it based on a set of predetermined security rules.

Firewalls are essential for network security because they help prevent unauthorized access to a network, protect against malicious traffic such as viruses and malware, and safeguard sensitive data from theft or exposure. They act as a barrier between a trusted internal network and an untrusted external network, such as the internet, and can also help prevent unauthorized access to specific services and applications running on a network.

A firewall is a security device that monitors and controls incoming and outgoing network traffic based on predetermined security rules. It acts as a barrier between a trusted, secure internal network and an untrusted external network, such as the internet.

Firewalls were developed in response to the growing need to protect computer networks from unauthorized access and malicious attacks. They use a combination of hardware and software components to inspect traffic and determine whether to allow or block it based on a set of predetermined rules.

Firewalls can be configured to filter traffic based on various criteria, such as IP address, port number, protocol type, and application type. They can also be configured to block traffic from specific geographic regions, as well as to detect and block known malware and viruses.

A firewall is a hardware or software device that is used to protect a network or computer system from unauthorized access and malicious activity. It acts as a gatekeeper between an internal network and the internet or other external networks.

Firewalls are implemented with a set of security policies that determine what traffic is allowed to pass through and what traffic is blocked.

A firewall is an example of **a reference monitor** , which means it should have three characteristics

- ✔ Always invoked
- ✔ Tamperproof
- ✔ small and simple enough for rigorous analysis

Types of Firewalls

- ➔ Packet filtering gateways or screening routers
- ➔ Stateful inspection firewalls
- ➔ Application-level gateways , also known as proxies
- ➔ Circuit – level gateways
- ➔ Guards
- ➔ Personal or host-based firewalls

Packet filtering gateways or screening routers

These firewalls examine each packet that flows through them and compare it against a set of rules or criteria to determine whether the packet should be allowed or blocked.

Packet filtering gateways work by analyzing the source and destination IP addresses, as well as the protocol and port numbers of each packet. Based on these criteria, the firewall decides whether to allow or block the packet.

These firewalls are typically implemented using routers that are configured with ACLs (access control lists) to control the flow of traffic. ACLs contain a set of rules that define which packets are allowed and which are blocked. The rules can be based on a variety of criteria, including the source and destination IP addresses, the protocol, and the port numbers.

When we say "Packet-filtering gateways do not maintain any information about the state of a connection", it means that the packet-filtering firewall or gateway only looks at each packet in isolation and makes decisions based on its content (such as the source and destination IP addresses, port numbers, and protocol type) without taking into account the context of the connection.

In other words, packet-filtering gateways do not keep track of the state of a connection, such as whether it is an ongoing session or whether it has been established or terminated. They treat each packet as a separate, standalone entity and evaluate it based on the rules defined in their access control lists or policy rules.

Stateful inspection firewalls

Packet filtering gateways maintain no state from one packer to the next. They simply look at each packet's IP address and port and compare them to the configured policies.

Stateful Inspection Firewall, also known as dynamic packet filtering, is a type of firewall that goes beyond the basic packet filtering approach used by traditional packet filtering firewalls.

Instead of just examining each packet individually, stateful inspection firewalls maintain a record, or state table, of the TCP connections passing through them. This allows the firewall to recognize whether a particular packet belongs to an existing connection or not.

Stateful inspection firewalls can then apply more advanced filters to the traffic, such as filtering based on the state of the connection (e.g., established, new, or closed) or inspecting application-layer data in the packet payload.

Application-level gateways , also known as proxies

An application-level gateway, also known as a proxy firewall, is a type of firewall that operates at the application layer of the OSI (Open Systems Interconnection) reference model. It provides an additional layer of security by acting as an intermediary between client applications and servers.

In order to accomplish this, an application-level gateway essentially acts as a proxy for each connection it handles. When a client application attempts to connect to a server through the firewall, it first sends a request to the gateway. The gateway then establishes its own connection to the server on behalf of the client, and passes data back and forth between the two endpoints.

An application proxy acts as an intermediary between a user and a server, and it simulates the behavior of an application at the application layer of the OSI model. This ensures that the real application receives only requests that are valid and appropriate.

- ✔ Application proxies can be used to filter out dangerous application-layer requests,
- ✔ keep logs of requests and accesses,
- ✔ and cache results to save bandwidth.

In practice, the most commonly used type of application proxy is a **web proxy,** which companies often use to monitor and filter their employees' Internet use.

Circuit Level Gateway

A circuit-level gateway, also known as a circuit-level proxy or stateful protocol analysis firewall, operates at the session layer (Layer 5) of the OSI model. It establishes a circuit, or virtual connection, between two networks or hosts, and inspects the session setup messages that pass between them to determine whether to allow or deny access.

A circuit-level gateway works by establishing a virtual circuit, which acts as a tunnel between two networks. This allows the networks to

communicate as if they were directly connected to each other.

A circuit-level gateway, also known as a "proxy gateway," works at the session layer (Layer 5) of the OSI model, allowing one network to act as an extension of another. It establishes a virtual circuit between the client and server, which means that it sets up and manages the connection between them, ensuring that the communication is secure and private. When a circuit-level gateway receives a request from a client, it authenticates the request and opens a connection to the server on behalf of the client. It then mediates the communication between the two parties,

A circuit-level gateway is like a secret door between two secret clubs. It helps one club to become a part of another club. It works by creating a special path between the two clubs that only they can use. This special path is like a secret tunnel that only the members of both clubs can use to communicate with each other. One way people use this secret path is to create something called a VPN, which helps them use the Internet more safely and privately.

A circuit-level gateway is a type of firewall that allows two separate computer networks to communicate with each other securely by creating a virtual "circuit" between them. It operates at the session layer of the network stack, which is responsible for managing the connections between applications on different machines.

For example, if a company has a private network for its employees and wants to allow remote workers to securely access that network over the internet, they could use a circuit-level gateway to establish a secure "tunnel" between the two networks. This tunnel encrypts all data that is transmitted between the two networks, protecting it from eavesdropping or tampering.

Guard

A guard firewall is a type of firewall that is designed to protect sensitive information and assets from unauthorized access or external threats. It is typically used in high-security environments where information must be protected at all costs.

The main function of a guard firewall is to restrict access between two networks, typically an unsecured network and a secured network. The firewall operates at the OSI Layer 2, also known as the data link layer, and can filter incoming and outgoing traffic based on predetermined security policies.

Guard firewalls use a combination of hardware and software-based technologies to provide a high level of security. They can monitor all traffic passing between the two networks and filter out unauthorized or malicious traffic. In addition, guard firewalls can also detect and prevent any attempts to bypass or tamper with the firewall.

Guards are a type of firewall that implements a set of programmable rules to protect a network or system. These rules are designed to limit access to or from specific resources based on predefined criteria. Guards can be configured to monitor user activity, network traffic, or system processes, and take appropriate actions when necessary to prevent unauthorized access or damage to the system.

One of the key features of guards is their flexibility in terms of the rules they can implement. For example, guards can be configured to limit the number of email messages a user can receive or to restrict a user's web bandwidth. They can also be programmed to filter documents containing specific keywords or phrases like for example , Filtering documents containing specific keywords, such as "Secret," is a way to prevent sensitive or confidential information from being leaked or transmitted outside of a secure network. , and to pass downloaded files through a virus scanner to prevent malware infections.

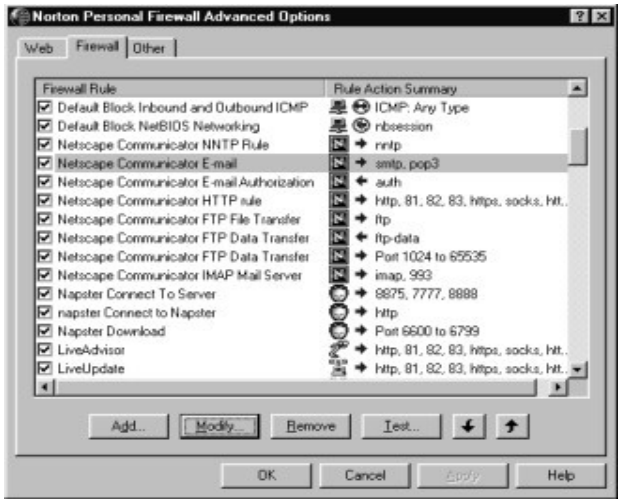
Guards can also be used to enforce security policies, such as blocking access to certain websites or enforcing password complexity requirements. They can be deployed at the network perimeter or within the internal network,

depending on the specific security requirements and the resources being protected.

Personal Firewalls

personal firewall is a type of firewall that is installed on a personal computer and provides protection for individual users. It typically monitors and controls incoming and outgoing network traffic based on predefined security rules.

Personal firewalls can block incoming traffic that is not authorized by the user, preventing unauthorized access to the computer, and can also prevent the transmission of sensitive information from the computer to the Internet. They can be configured to allow or block specific applications or services from accessing the network, and can also alert the user when suspicious activity is detected.



WHAT FIREWALLS CAN AND CAN NOT DO ?

Firewalls can protect an environment only if they control the entire perimeter: This means that firewalls must be deployed to cover all entry and exit points to the network or system they are meant to protect. If there are any unprotected entry or exit points, attackers can bypass the firewall and gain unauthorized access to the system or network.

Firewalls do not protect data outside the perimeter: This means that if an attacker gains access to data outside of the firewall-protected network, the firewall cannot prevent the attacker from exploiting or exfiltrating the data. This highlights the importance of other security measures, such as access controls and encryption.

Firewalls are the most visible part of an installation to the outside, so they are an attractive target for attack: This means that attackers often focus their efforts on bypassing or disabling firewalls to gain access to the network or system behind them. This makes it important to have strong and updated firewall policies and configurations, as well as implementing additional security measures to mitigate any potential risks.

Firewalls must be correctly configured, that configuration must be updated as the environment changes, and firewall activity reports must be reviewed periodically for evidence of attempted or successful intrusion: This means that proper configuration and management of firewalls is crucial to maintaining their effectiveness. Firewall policies and configurations must be reviewed and updated regularly to address any potential vulnerabilities or changes in the network environment. Firewall logs must be monitored to detect any unusual or suspicious activity.

Firewalls exercise only minor control over the content admitted to the inside, meaning that inaccurate or malicious code must be controlled by means inside the perimeter: This means that while firewalls can control and filter network traffic based on certain criteria, such as IP addresses and ports, they cannot fully protect against malware or other malicious content that is introduced through other means, such as email or USB drives. Therefore, additional security measures such as antivirus software and user education are necessary to mitigate these risks.

Intrusion Detection System (IDS)

An intrusion detection system (IDS) is a type of security software or hardware that monitors network traffic, system events, and user behaviors on a computer network or system. The primary goal of an IDS is to identify unauthorized or

suspicious activity that could indicate an ongoing or potential security threat.

IDSs come in two main types: signature-based and anomaly-based. Signature-based IDSs identify known attack patterns by comparing network traffic and system events against a database of signatures of known attacks. Anomaly-based IDSs, on the other hand, use statistical methods to establish a baseline of normal behavior for a system, and then alert security personnel when deviations from that baseline are detected.

When suspicious activity or potential security threats are detected, an IDS generates alerts to security personnel, allowing them to take action to investigate and respond to the issue. IDSs can also be configured to take automated actions when a threat is detected, such as blocking traffic from a particular IP address.

IDSs can be deployed as standalone systems or as part of a larger security infrastructure, including firewalls and intrusion prevention systems (IPSs). The primary difference between IDSs and IPSs is that IDSs focus on detecting and alerting on potential threats, while IPSs take proactive measures to prevent those threats from occurring in the first place.