Q1) Discuss authentication, header and ESP in detail with their packet format

Authentication is the process of verifying the identity of a user or information.

Authentication technology provides access control for systems by checking to see if a user's credentials match the credentials in a database of authorized users or in a data authentication server. In doing this, authentication assures secure systems, secure processes and enterprise information security.

There are several authentication types in which they are:

Single-Factor authentication:

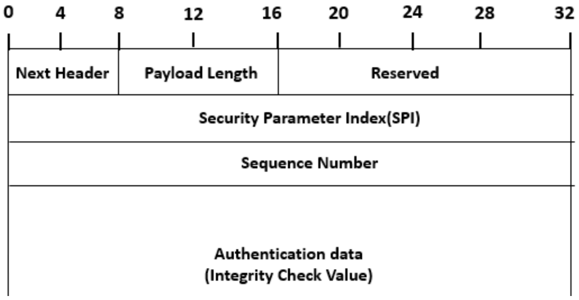
This was the first method of security that was developed. On this authentication system, the user has to enter the username and the password to confirm whether that user is logging in or not. Now if the username or password is wrong, then the user will not be allowed to log in or access the system.

Two-factor Authentication:

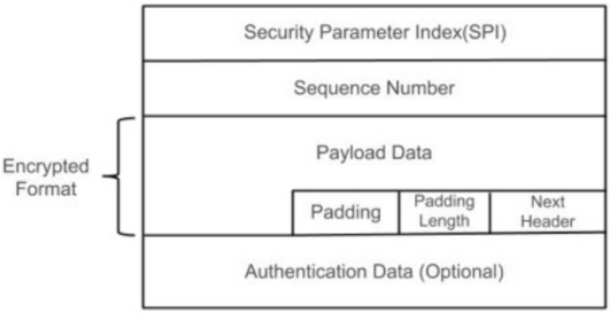
In this authentication system, the user has to give a username, password, and other information. There are various types of authentication systems that are used by the user for securing the system. Some of them are wireless tokens and virtual tokens, OTP and more.

Authentication header:

The AH protocol provides a mechanism for authentication only. AH provides data integrity, data origin authentication, and an optional replay protection service. Data integrity is ensured by using a message digest that is generated by an algorithm such as HMAC-MD5 or HMAC-SHA. Data origin authentication is ensured by using a shared secret key to create the message digest. Replay protection is provided by using a sequence number field with the AH header.



AH



ESP

1. Next Header: This field is 8 bit used to identify the header types that immediately follow the authentication header. For example, if the ESP header follows the AH, this field contains 50 as a value; otherwise, if another AH follows this AH, it contains 51 as a value.
2. Payload Length: This field is 8 bit. It contains the length of the Authentication header in 32-bit words minus.
3. Reserved: This field is 16 bit, which is reserved for future use.
4. Security Parameter Index (SPI): This field is 32 bit. It is used in combination with the source address and destination address and the IPsec protocol (Internet Protocol Security), uniquely identifying the security association (SA) for the traffic to which the datagram belongs.
5. Sequence Number: This field is 32 bit which is used for the replay attacks.
6. Authentication Data: This variable-length field contains the authentication data, called an Integrity Check Value of (ICV) for the datagram. This value is used for integrity, and authentication purpose is in MAC form.
7.

ESP protocol:

The ESP protocol provides data confidentiality (encryption) and authentication (data integrity, data origin authentication, and replay protection). ESP can be used with confidentiality only, authentication only, or both confidentiality and authentication. When ESP provides authentication functions, it uses the same algorithms as AH, but the coverage is different

Security Parameters Index (32 bits) − Identifies a security association. This field is mandatory. The value of zero is reserved for local, implementation- specific use and MUST NOT be sent on the wire.

Sequence Number (32 bits) − A monotonically increasing counter value; this provides an anti-replay function, as discussed for AH. The first packet sent using a given SA will have a Sequence number of 1.

Payload Data (variable) − This is a transport-level segment (transport mode) or IP packet (tunnel mode) that is protected by encryption. The type of content that was protected is indicated by the Next Header field.

Padding (0-255 bytes) − Padding for encryption, to extend the payload data to a size that fits the encryption's cipher block size, and to align the next field.

Pad Length (8 bits) − Indicates the number of pad bytes immediately preceding this field.

Next Header (8 bits) − Identifies the type of data contained in the payload data field by identifying the first header in that payload.

Authentication Data (variable) − A variable-length field (must be an integral number of 32-bit words) that contains the Integrity. Check Value computed over the ESP packet minus the Authentication Data field. This field is optional and is included only if the authentication service has been selected for the SA in question.

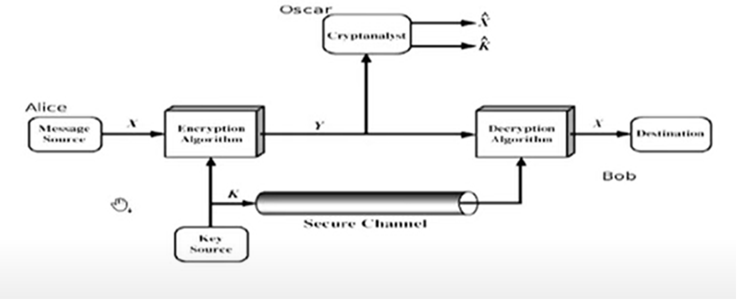
Q2) What is Cryptography? Explain the types and features of Cryptography

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

In Cryptography the techniques which are use to protect information are obtained from mathematical concepts and a set of rule based calculations known as algorithms to convert messages in ways that make it hard to decode it.

A cryptographic algorithm works in combination with a key—a word, number, or phrase—to encrypt the plaintext. The same plaintext encrypts to different ciphertext with different keys. The security of encrypted data is entirely dependent on two things: the strength of the cryptographic algorithm and the secrecy of the key. A cryptographic algorithm, plus all possible keys and all the protocols that make it work, comprise a cryptosystem.

Cryptography is technique of securing information and communications through use of codes so that only those person for whom the information is intended can understand it and process it. Thus preventing unauthorized access to information.



Cryptographic systems are generally classified along 3 independent dimensions:

1.Type of operations used for transforming plain text to cipher text

All the encryption algorithms are based on two general principles: substitution, in which each element in the plaintext is mapped into another element, and transposition, in which elements in the plaintext are rearranged.

2.The number of keys used

If the sender and receiver uses same key then it is said to be symmetric key (or) single key (or) conventional encryption.

If the sender and receiver use different keys then it is said to be public key encryption.

3.The way in which the plain text is processed

A block cipher processes the input and block of elements at a time, producing output block for each input block.

Types Of Cryptography:

Symmetric Key Cryptography:

It is an encryption system where the sender and receiver of a message use a single common key to encrypt and decrypt messages. Symmetric Key Systems are faster and simpler but the problem is that sender and receiver have to somehow exchange keys in a secure manner. The most popular symmetric-key cryptography system is Data Encryption System(DES).

Hash Functions:

There is no usage of any key in this algorithm. A hash value with a fixed length is calculated as per the plain text which makes it impossible for the contents of plain text to be recovered. Many operating systems use hash functions to encrypt passwords.

Asymmetric Key Cryptography:

Under this system, a pair of keys is used to encrypt and decrypt information. A public key is used for encryption and a private key is used for decryption. The public key and Private Key are different. Even if the public key is known by everyone the intended receiver can only decode it because he alone knows the private key.

Features Of Cryptography are as follows:

Confidentiality:

Information can only be accessed by the person for whom it is intended and no other person except him can access it.

Integrity:

Information cannot be modified in storage or transition between sender and intended receiver without any addition to information being detected.

Non-repudiation:

The creator/sender of information cannot deny his intention to send information at a later stage

Authentication:

The identities of the sender and receiver are confirmed. As well as the destination/origin of the information is confirmed.

Q3) Describe the SSL Architecture in detail

Secure Socket Layer (SSL) provides security to the data that is transferred between web browser and server. SSL encrypts the link between a web server and a browser which ensures that all data passed between them remain private and free from attack.

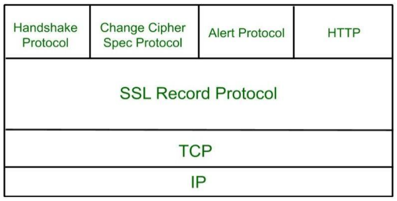
Working of SSL:

SSL encrypts data communicated across the web to guarantee a high level of privacy. Anyone attempting to intercept this data will meet a jumbled mess of characters nearly hard to decrypt.

SSL begins an authentication process known as a handshake between two communicating devices to confirm that both devices are who they say they are.SSL also digitally certifies data to ensure data integrity, ensuring that it has not been tampered with before reaching its intended receiver.

SSL has gone through multiple incarnations, each one more secure than the last. TLS (Transport Layer Security) was introduced in 1999, replacing SSL.

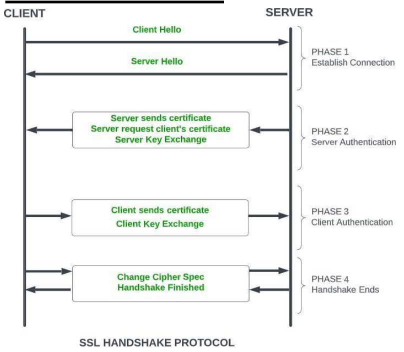
SSL Architecture:

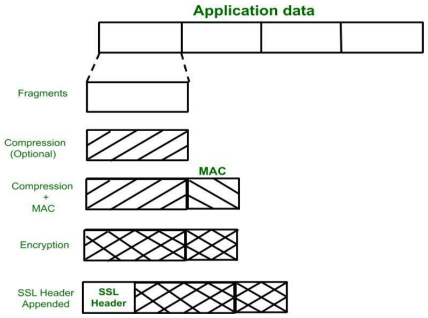


Secure Socket Layer (SSL) Protocols:

- SSL record protocol
- Handshake protocol
- Change-cipher spec protocol
- Alert protocol

SSL Record Protocol:





SSL Record provides two services to SSL connection.

- Confidentiality
- Message Integrity

In the SSL Record Protocol application data is divided into fragments. The fragment is compressed and then encrypted MAC (Message Authentication Code) generated by algorithms like SHA (Secure Hash Protocol) and MD5 (Message Digest) is appended. After that encryption of the data is done and in last SSL header is appended to the data.

Handshake protocol:

Handshake Protocol is used to establish sessions. This protocol allows the client and server to authenticate each other by sending a series of messages to each other. Handshake protocol uses four phases to complete its cycle.

Change-cipher Protocol:

This protocol uses the SSL record protocol. Unless Handshake Protocol is completed, the SSL record Output will be in a pending state. After the handshake protocol, the Pending state is converted into the current state.

1 byte

Change-cipher protocol consists of a single message which is 1 byte in length and can have only one value. This protocol's purpose is to cause the pending state to be copied into the current state.

Alert Protocol:

This protocol is used to convey SSL-related alerts to the peer entity. Each message in this protocol contains 2 bytes.

Level (1 byte)	Alert (1 byte)

The level is further classified into two parts:

Warning (level = 1):

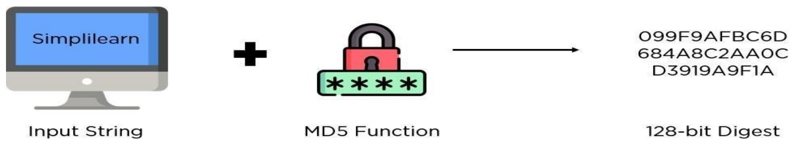
This Alert has no impact on the connection between sender and receiver.

Fatal Error (level = 2):

This Alert breaks the connection between sender and receiver. The connection will be stopped, cannot be resumed but can be restarted.

Q5) Explain MD5 in detail

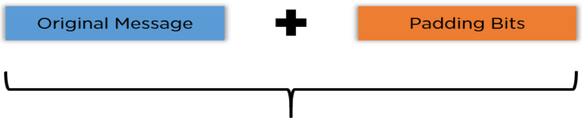
MD5 (Message Digest Method 5) is a cryptographic hash algorithm is used to generate a 128-bit digest from a string of any length. It represents the digests as 32-digit hexadecimal numbers. The hash algorithm MD5 is widely used to check the integrity of messages. Ronald Rivest designed this algorithm in 1991 to provide the means for digital signature verification.



Steps in MD5 Algorithm:

There are four major sections of the algorithm:

Padding Bits:

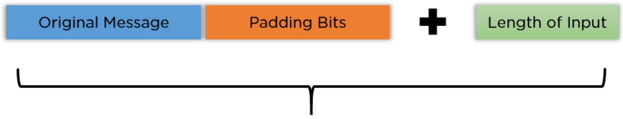


Total length to be 64 bits less than multiple of 512

When you receive the input string, you have to make sure the size is 64 bits short of a multiple of 512. When it comes to padding the bits, you must add one(1) first, followed by zeroes to round out the extra characters.

Padding Length:

You need to add a few more characters to make your final string a multiple of 512. To do so, take the length of the initial input and express it in the form of 64 bits. On combining the two, the final string is ready to be hashed.



Final Data to be Hashed as a multiple of 512

Initialize MD buffer

The entire string is converted into multiple blocks of 512 bits each. You also need to initialize four different buffers, namely A, B, C, and D. These buffers are 32 bits each and are initialized as follows:

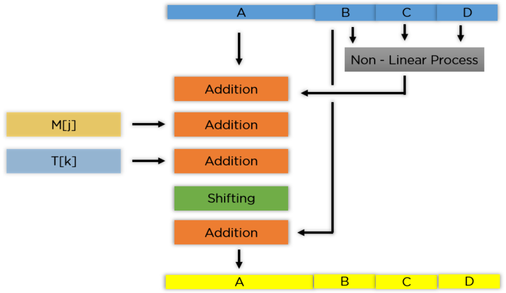
A = 01 23 45 67

B = 89 ab cd ef C = fe dc ba 98 D = 76 54 32 10

Process Each Block:

Each 512-bit block gets broken down further into 16 sub-blocks of 32 bits each. There are four rounds of operations, with each round utilizing all the sub-blocks, the buffers, and a constant array value.

This constant array can be denoted as T[1] -> T[64]. Each of the sub-blocks are denoted as M[0] -> M[15].



- It passes B, C, and D onto a non-linear process.
- The result is added with the value present at A.
- It adds the sub-block value to the result above.
- Then, it adds the constant value for that particular iteration.
- There is a circular shift applied to the string.
- As a final step, it adds the value of B to the string and is stored in buffer A.

The steps mentioned above are run for every buffer and every sub-block. When the last block's final buffer is complete, you will receive the MD5 digest.

The non-linear process above is different for each round of the sub-block. Round 1: (b AND c) OR ((NOT b) AND (d))

Round 2: (b AND d) OR (c AND (NOT d))

Round 3: b XOR c XOR d Round 4: c XOR (b OR (NOT d)

Q Write a note on Digital signature

A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software or digital document. It's the digital equivalent of a handwritten signature or stamped seal, but it offers far more inherent security. A digital signature is intended to solve the problem of tampering and impersonation in digital communications.

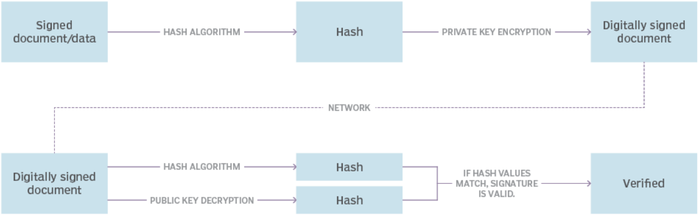
In many countries, including the United States, digital signatures are considered legally binding in the same way as traditional handwritten document signatures. Digital signatures increase the transparency of online interactions and develop trust between customers, business partners, and vendors.

There are three algorithms at work in Digital Signatures. They are as follows:

Key Generation Algorithms – Key Generation Algorithms help ensure authenticity and integrity or it would be very easy to tamper with the data. They also prevent anyone from pretending to be the sender.

Signing Algorithms – Signing Algorithms make one-way hashes of the data that has to be signed. Then they encrypt the hash value using the signature key. The encrypted hash along with the other information is the Digital Signature.

Signature Verification Algorithms – Signature Verification Algorithms help process the Digital Signature and the verification key to generate some values. The algorithm also processes the same hash function on the data received and creates a hash value.



Importance of Digital Signature:

Message authentication:

When the verifier validates the digital signature using public key of a sender, he is assured that signature has been created only by sender who possess the corresponding secret private key and no one else.

Data Integrity:

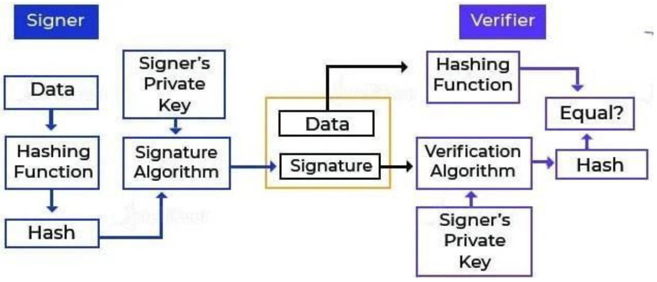
In case an attacker has access to the data and modifies it, the digital signature verification at receiver end fails. The hash of modified data and the output provided by the verification algorithm will not match. Hence, receiver can safely deny the message assuming that data integrity has been breached.

Non-repudiation:

Since it is assumed that only the signer has the knowledge of the signature key, he can only create a unique signature on a given data. Thus the receiver can present data and the digital signature to a third party as evidence if any dispute arises in the future.

Q7) What are the steps followed in creating a digital signature?

The steps followed in creating digital signature are :



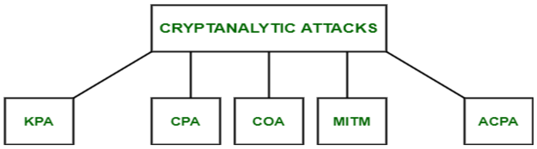
1. People who adopt the Digital Signature scheme have a public-private key pair.
2. The pairs for encryption/decryption and signing/verifying are usually different. The signature key is the private key that is used for signing and the verification key is the private key.
3. Signer feeds all data to the hashing function and in turn, generates a hash of data.
4. The signature key and hash value are fed into the signature algorithm to help produce a Digital Signature on the given hash. Once the signature is appended to the data, both are sent forward to the verifier.
5. The verifier will then feed the Digital Signature and the verification key into the verification algorithm. The verification algorithm will generate an output value.
6. Verifier is also responsible for running some hashing functions on the data received so it can generate a hash value.
7. Verification is processed by comparing the hash value generated by the verifier and the output of the verification algorithm. The result of this comparison helps the verifier decide if the Digital Signature is valid or not.
8. Nobody else has access to the private key of the signer and the Digital Signature is created using this key, so the signer cannot reject signing the document in the future

Q8) Explain Cryptanalytic attacks

Cryptology has two parts namely, Cryptography which focuses on creating secret codes and Cryptanalysis which is the study of the cryptographic algorithm and the breaking of those secret codes. The person practising Cryptanalysis is called a Cryptanalyst. It helps us to better understand the cryptosystems and also helps us improve the system by finding any weak point and thus work on the algorithm to create a more secure secret code.

For example, a Cryptanalyst might try to decipher a ciphertext to derive the plaintext. It can help us to deduce the plaintext or the encryption key.

CRYPTOLOGY
CRYPTOGRAPHY
CRYPTANALYSIS



Types of Cryptanalytic attacks :

The Five Types of Cryptanalytic Attacks

Known-Plaintext Analysis (KPA) :

In this type of attack, some plaintext-ciphertext pairs are already known. Attacker maps them in order to find the encryption key. This attack is easier to use as a lot of information is already available.

Chosen-Plaintext Analysis (CPA) :

In this type of attack, the attacker chooses random plaintexts and obtains the corresponding ciphertexts and tries to find the encryption key. It's very simple to implement like KPA but the success rate is quite low.

Ciphertext-Only Analysis (COA) :

In this type of attack, only some cipher-text is known and the attacker tries to find the corresponding encryption key and plaintext. Its the hardest to implement but is the most probable attack as only ciphertext is required

Man-In-The-Middle (MITM) attack :

In this type of attack, attacker intercepts the message/key between two communicating parties through a secured channel.

Adaptive Chosen-Plaintext Analysis (ACPA) :

This attack is similar to CPA. Here, the attacker requests the cipher texts of additional plaintexts after they have ciphertexts for some texts.

Q9) Illustrate the SHA algorithm and explain

SHA stands for secure hashing algorithm. SHA is a modified version of MD5 and is used for hashing data and certificates. SHA-1 or Secure Hash Algorithm 1 is a cryptographic hash function which takes an input and produces a 160-bit (20-byte) hash value. This hash value is known as a message digest. This message digest is usually then rendered as a hexadecimal number which is 40 digits long.

It following versions:

- SHA-O
- SHA-1
- SHA-224
- SHA-256
- SHA-512

Working of SHA-1:

Step 1: Padding of Bits

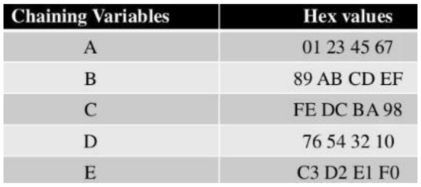
- Padding of Bits similar to MD5 in a way that the length of the message is 64 bits short of a multiple of 512.
- Ex: Length of the message = 1000 bits Padding bits = 472 (1536-1064) 1000+472 +64= 1536 That is Multiple of 512 (512 x 3 = 1536)

Step 2: Append Length

Step 3: Divide the input into 512-bit blocks

Step 4: Initialize chaining variables

Step 5: Process Blocks



Chaining Variables	Hex values
A	01 23 45 67
B	89 AB CD EF
C	FE DC BA 98
D	76 54 32 10
E	C3 D2 E1 F0

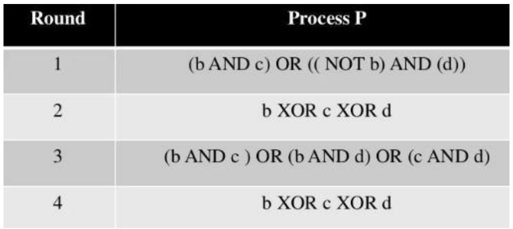
Step 5.1: Copy chaining variables A-E into variables a-e.

Step 5.2: Divide the current 512-bit block into 16 sub-blocks of 32-bits.

Step 5.3: SHA has 4 rounds, each consisting of 20 steps. Each round takes 3 inputs:

- 512-bit block
- The register abcde
- A constant K[J (where t= 0 to 79)

Step 5.4: SHA has a total of 80 iterations (4 rounds X 20 iterations). Each iteration consists of the following operations:



Round	Process P
1	(b AND c) OR ((NOT b) AND (d))
2	b XOR c XOR d
3	(b AND c) OR (b AND d) OR (c AND d)
4	b XOR c XOR d

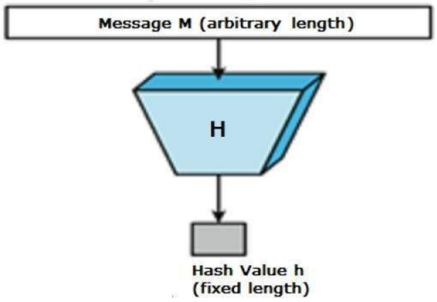
- Process P in each round
- Then at last values of W[t] is calculated.

A hash function is a mathematical function that converts a numerical input value into another compressed numerical value. The input to the hash function is of arbitrary length but the output is always of fixed length. Values returned by a hash function are called the message digest or simply hash values.

Q. Hash function design and properties

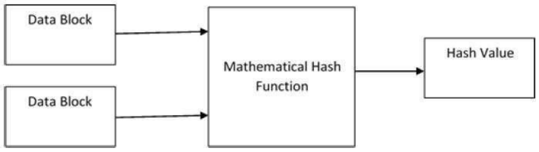
Design of Hashing Algorithms

At the heart of a hashing is a mathematical function that operates on two fixed-size blocks of data to create a hash code. This hash function forms part of the hashing algorithm.



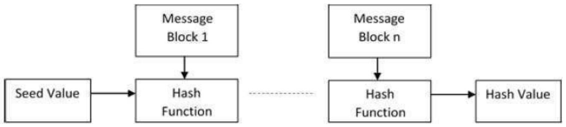
The size of each data block varies depending on the algorithm. Typically the block sizes are from 128 bits to 512 bits. The following illustration demonstrates the hash function −

<u>Hash Function Structure:</u>



Hashing algorithm involves rounds of above hash function like a block cipher. Each round takes an input of a fixed size, typically a combination of the most recent message block and the output of the last round.

This process is repeated for as many rounds as are required to hash the entire message. Schematic of hashing algorithm is depicted in the following illustration −



<u>Hashing Algorithm</u>

Since, the hash value of first message block becomes an input to the second hash operation, output of which alters the result of the third operation, and so on. This effect, known as an avalanche effect of hashing. The Avalanche effect results in substantially different hash values for two messages that differ by even a single bit of data.

Understand the difference between hash function and algorithm correctly. The hash function generates a hash code by operating on two blocks of fixed-length binary data.

Hashing algorithm is a process for using the hash function, specifying how the message will be broken up and how the results from previous message blocks are chained together.

Features of hash function

<u>Fixed Length Output (Hash Value):</u>

- Hash function coverts data of arbitrary length to a fixed length. This process is often referred to as hashing the data.
- In general, the hash is much smaller than the input data, hence hash functions are sometimes called compression functions.

<u>Efficiency of Operation:</u>

- Generally for any hash function h with input x, computation of h(x) is a fast operation.
- Computationally hash functions are much faster than a symmetric encryption.

Properties of a hash function:

- Deterministic - The output will be the same for a given outcome.
- Not reversible – We can’t reverse a hash function back to the original password.
- Collision resistant – Two inputs do not result in the same output.
- Non-predictable – A hash function randomly generates a unique hash value that is not predictable.
- Compression – The hash function’s output is much smaller than the input size
-

Q. RSA DSS 2 APPROACHES TO DIGITAL SIGN

RSA algorithm

RSA algorithm is an asymmetric cryptography algorithm. Asymmetric actually means that it works on two different keys i.e. Public Key and Private Key. As the name describes that the Public Key is given to everyone and Private key is kept private.

An example of asymmetric cryptography :

- A client (for example browser) sends its public key to the server and requests for some data.
- The server encrypts the data using client’s public key and sends the encrypted data.
- Client receives this data and decrypts it.
- Since this is asymmetric, nobody else except browser can decrypt the data even if a third party has public key of browser.

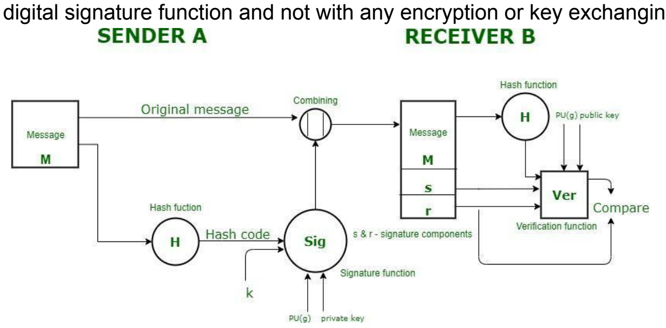
The idea! The idea of RSA is based on the fact that it is difficult to factorize a large integer. The public key consists of two numbers where one number is multiplication of two large prime numbers. And private key is also derived from the same two prime numbers. So if somebody can factorize the large number, the private key is compromised.

Therefore encryption strength totally lies on the key size and if we double or triple the key size, the strength of encryption increases exponentially. RSA keys can be typically 1024 or 2048 bits long, but experts believe that 1024-bit keys could be broken in the near future. But till now it seems to be an infeasible task.

DSS (Digital Signal Standard)

As we have studied, signature is a way of authenticating the data coming from a trusted individual. Similarly, digital signature is a way of authenticating a digital data coming from a trusted source.

Digital Signature Standard (DSS) is a Federal Information Processing Standard(FIPS) which defines algorithms that are used to generate digital signatures with the help of Secure Hash Algorithm(SHA) for the authentication of electronic documents. DSS only provides us with the digital signature function and not with any encryption or key exchanging strategies.



Sender Side :

In DSS Approach, a hash code is generated out of the message and following inputs are given to the signature function –

- The hash code.
- The random number ‘k’ generated for that particular signature.
- The private key of the sender i.e., PR(a).
- A global public key(which is a set of parameters for the communicating principles) i.e., PU(g).

These input to the function will provide us with the output signature containing two components – ‘s’ and ‘r’. Therefore, the original message concatenated with the signature is sent to the receiver.

Receiver Side :

At the receiver end, verification of the sender is done. The hash code of the sent message is generated. There is a verification function which takes the following inputs –

- The hash code generated by the receiver.
- Signature components ‘s’ and ‘r’.
- Public key of the sender.
- Global public key.

The output of the verification function is compared with the signature component ‘r’. Both the values will match if the sent signature is valid because only the sender with the help of it private key can generate a valid signature.

Therefore the two most popular and commonly used public-key system based digital signature schemes are the RSA (named after Rivest, Shamir, and Alderman, the inventors of the RSA public-key encryption scheme) and the digital signature algorithm (DSA) approaches. The DSA is incorporated into the Digital Signature Standard (DSS), which was published by the National Institute of Standards and Technology as the Federal Information Processing Standard.

Q12) Explain the attacks related to Digital Signature

There are three types of attacks on Digital Signatures:

1. Chosen-message Attack
2. Known-message Attack
3. Key-only Attack

Let us consider an example where c is the attacker and A is the victim whose message and signature are under attack.

1. <u>Chosen-message Attack :</u>

The chosen attack method is of two types:

- Generic chosen-method – In this method C tricks A to digitally sign the messages that A does not intend to do and without the knowledge about A's public key.
- Direct chosen-method – In this method C has the knowledge about A's public key and obtains A's signature on the messages and replaces the original message with the message C wants A to sign with having A's signature on them unchanged.

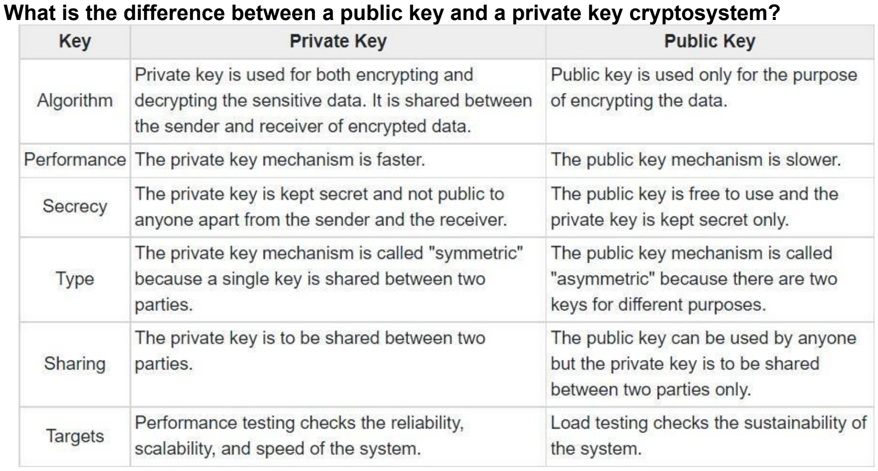
2. <u>Known-message Attack :</u>

In the known message attack, C has a few previous messages and signatures of A. Now C tries to forge the signature of A on to the documents that A does not intend to sign by using the brute force method by analyzing the previous data to recreate the signature of A. This attack is similar to known-plain text attack in encryption.

3. <u>Key-only Attack :</u>

In key-only attack, the public key of A is available to every one and C makes use of this fact and try to recreate the signature of A and digitally sign the documents or messages that A does not intend to do. This would cause a great threat to authentication of the message which is non-repudiated as A cannot deny signing it.

Q13**) What is the difference between a public key and a private key cryptosystem?**



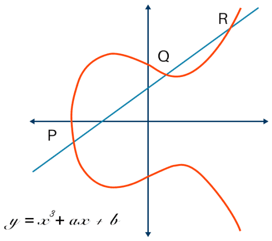
Key	Private Key	Public Key
Algorithm	Private key is used for both encrypting and decrypting the sensitive data. It is shared between the sender and receiver of encrypted data.	Public key is used only for the purpose of encrypting the data.
Performance	The private key mechanism is faster.	The public key mechanism is slower.
Secrecy	The private key is kept secret and not public to anyone apart from the sender and the receiver.	The public key is free to use and the private key is kept secret only.
Type	The private key mechanism is called "symmetric" because a single key is shared between two parties.	The public key mechanism is called "asymmetric" because there are two keys for different purposes.
Sharing	The private key is to be shared between two parties.	The public key can be used by anyone but the private key is to be shared between two parties only.
Targets	Performance testing checks the reliability, scalability, and speed of the system.	Load testing checks the sustainability of the system.

Q. Elliptical curve cryptography

Elliptic Curve Cryptography (ECC) is a key-based technique for encrypting data. ECC focuses on pairs of public and private keys for decryption and encryption of web traffic.

ECC is frequently discussed in the context of the Rivest–Shamir–Adleman (RSA) cryptographic algorithm. RSA achieves one-way encryption of things like emails, data, and software using prime factorization.

ECC, an alternative technique to RSA, is a powerful cryptography approach. It generates security between key pairs for public key encryption by using the mathematics of elliptic curves.



An elliptic curve for current ECC purposes is a plane curve over a finite field which is made up of the points satisfying the equation: $y^2=x^3 + ax + b$.

In this elliptic curve cryptography example, any point on the curve can be mirrored over the x-axis and the curve will stay the same. Any non-vertical line will intersect the curve in three places or fewer.

<u>Uses:</u>

- Websites make extensive use of ECC to secure customers' hypertext transfer protocol connections.
- It is used for encryption by combining the key agreement with a symmetric encryption scheme.
- It is also used in several integer factorization algorithms like Lenstra elliptic-curve factorization.
- Time stamping uses an encryption model called a blind signature scheme. It is possible using Elliptic Curve Cryptography.

Q16) What is the Advanced Encryption Standard?

The more popular and widely adopted symmetric encryption algorithm likely to be encountered nowadays is the Advanced Encryption Standard (AES). It is found at least six time faster than triple DES.

A replacement for DES was needed as its key size was too small. With increasing computing power, it was considered vulnerable against exhaustive key search attack. Triple DES was designed to overcome this drawback but it was found slow.

The features of AES are as follows:

1. Symmetric key symmetric block cipher 2. 128-bit data, 128/192/256-bit keys
2. Stronger and faster than Triple-DES
3. Provide full specification and design details
4.Software implementable in C and Java

Operation of AES

AES is an iterative rather than Feistel cipher. It is based on 'substitution–permutation network'. It comprises of a series of linked operations, some of which involve replacing inputs by specific outputs (substitutions) and others involve shuffling bits around (permutations).

Interestingly, AES performs all its computations on bytes rather than bits. Hence, AES treats the 128 bits of a plaintext block as 16 bytes. These 16 bytes are arranged in four columns and four rows for processing as a matrix −

UnIlke DES, the number of rounds in AES is variable and depends on the length of the key. AES uses 10 rounds for 128-bit keys, 12 rounds for 192-bit keys and 14 rounds for 256-bit keys. Each of these rounds uses a different 128-bit round key, which is calculated from the original AES key.

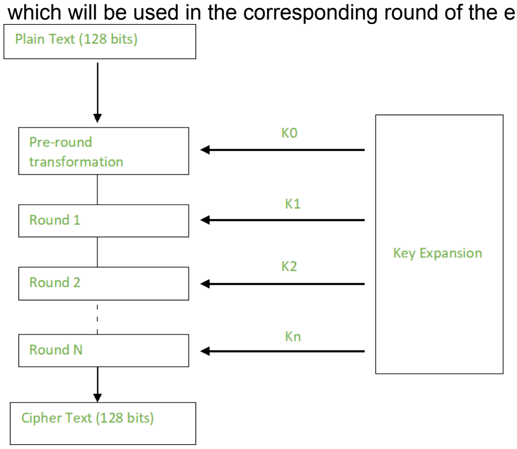
Working of the cipher :

AES performs operations on bytes of data rather than in bits. Since the block size is 128 bits, the cipher processes 128 bits (or 16 bytes) of the input data at a time.

The number of rounds depends on the key length as follows

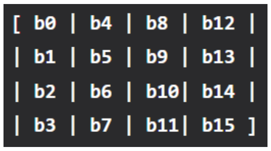
- 128-bit key – 10 rounds
- 192-bit key – 12 rounds
- 256-bit key – 14 rounds Creation of Round keys :

A Key Schedule algorithm is used to calculate all the round keys from the key. So the initial key is used to create many different round keys which will be used in the corresponding round of the encryption.



Encryption :

AES considers each block as a 16 byte (4 byte x 4 byte = 128) grid in a column major arrangement.



```
[ b0 | b4 | b8 | b12 |
| b1 | b5 | b9 | b13 |
| b2 | b6 | b10| b14 |
| b3 | b7 | b11| b15 ]
```

Each round comprises of 4 steps :

- SubBytes
- ShiftRows
- MixColumns
- Add Round Key

The last round doesn't have the MixColumns round.

The SubBytes does the substitution and ShiftRows and MixColumns performs the permutation in the algorithm.

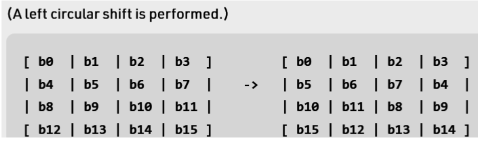
SubBytes

:This step implements the substitution.

- In this step each byte is substituted by another byte. Its performed using a lookup table also called the S-box. This substitution is done in a way that a byte is never substituted by itself and also not substituted by another byte which is a compliment of the current byte. The result of this step is a 16 byte (4 x 4) matrix like before.
- The next two steps implement the permutation.

<u>ShiftRows:</u>

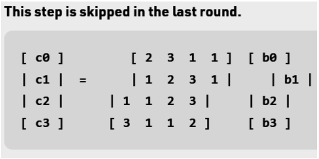
- This step is just as it sounds. Each row is shifted a particular number of times.
- The first row is not shifted
- The second row is shifted once to the left.
- The third row is shifted twice to the left.
- The fourth row is shifted to the thrice to left



(A left circular shift is performed.)

```
[ b0  | b1  | b2  | b3  ]        [ b0  | b1  | b2  | b3  ]
| b4  | b5  | b6  | b7  |   ->   | b5  | b6  | b7  | b4  |
| b8  | b9  | b10 | b11 |        | b10 | b11 | b8  | b9  |
[ b12 | b13 | b14 | b15 ]        [ b15 | b12 | b13 | b14 ]
```

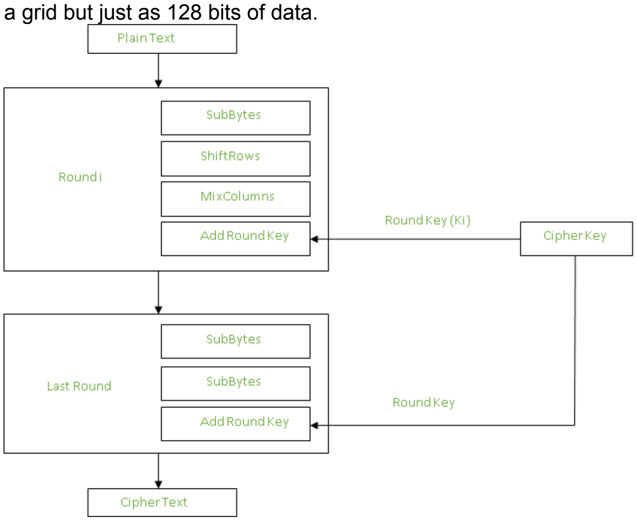
<u>MixColumns:</u>This step is basically a matrix multiplication. Each column is multiplied with a specific matrix and thus the position of each byte in the column is changed as a result.



This step is skipped in the last round.

```
[ c0 ]        [ 2  3  1  1 ]  [ b0 ]
| c1 |  =     | 1  2  3  1 |     | b1 |
| c2 |      | 1  1  2  3 |     | b2 |
[ c3 ]      [ 3  1  1  2 ]     [ b3 ]
```

<u>Add Round Keys:</u>Now the resultant output of the previous stage is XOR-ed with the corresponding round key. Here, the 16 bytes is not considered as a grid but just as 128 bits of data.



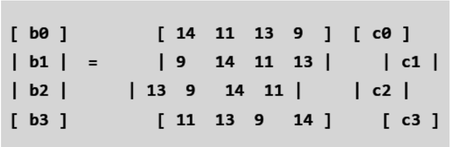
<u>Decryption:</u>The stages in the rounds can be easily undone as these stages have an opposite to it which when performed reverts the changes.Each 128 blocks goes through the 10,12 or 14 rounds depending on the key size.

The stages of each round in decryption is as follows :

- Add round key
- Inverse MixColumns
- ShiftRows
- Inverse SubByte
- The decryption process is the encryption process done in reverse so i will explain the steps with notable differences.

<u>Inverse MixColumns</u>

This step is similar to the MixColumns step in encryption, but differs in the matrix used to carry out the operation.

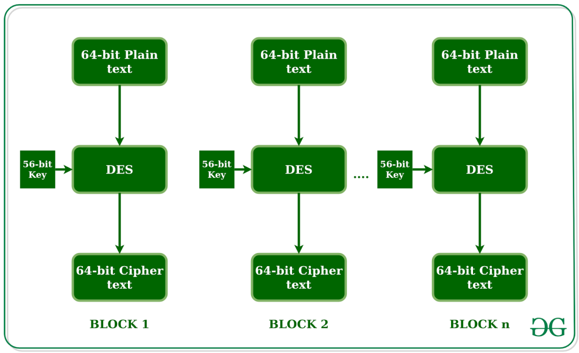


```
[ b0 ]        [ 14  11  13  9  ]  [ c0 ]
| b1 |  =     | 9   14  11  13 |     | c1 |
| b2 |      | 13  9   14  11 |     | c2 |
[ b3 ]        [ 11  13  9   14 ]     [ c3 ]
```

<u>Inverse SubBytes:</u>Inverse S-box is used as a lookup table and using which the bytes are substituted during decryption.

Q. DES triple des

DES: Data encryption standard (DES) has been found vulnerable to very powerful attacks and therefore, the popularity of DES has been found slightly on the decline. DES is a block cipher and encrypts data in blocks of size of 64 bits each, which means 64 bits of plain text go as the input to DES, which produces 64 bits of ciphertext. The same algorithm and key are used for encryption and decryption, with minor differences. The key length is 56 bits. The basic idea is shown in the figure.

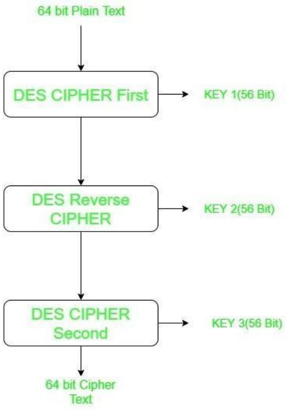


DES Algorithm Steps

The algorithm process breaks down into the following steps:

- The process begins with the 64-bit plain text block getting handed over to an initial permutation (IP) function.
- The initial permutation (IP) is then performed on the plain text.
- Next, the initial permutation (IP) creates two halves of the permuted block, referred to as Left Plain Text (LPT) and Right Plain Text T).
- Each LPT and RPT goes through 16 rounds of the encryption process.
- Finally, the LPT and RPT are rejoined, and a Final Permutation (FP) is performed on the newly combined block.
- The result of this process produces the desired 64-bit ciphertext.

Triple DES: Triple DES is a encryption technique which uses three instance of DES on same plain text. It uses there different types of key choosing technique in first all used keys are different and in second two keys are same and one is different and in third all keys are same.



Triple DES is also vulnerable to meet-in-the middle attack because of which it give total security level of 2^112 instead of using 168 bit of key. The block collision attack can also be done because of short block size and using same key to encrypt large size of text. It is also vulnerable to sweet32 attack.

Q18) **Explain the following operations used in AES**

1. Substitute bytes
2. Shift Rows
3. Mix Columns
4. Add Round Key

Byte Substitution (SubBytes):The 16 input bytes are substituted by looking up a fixed table (S-box) given in design. The result is in a matrix of four rows and four columns.

Shift rows: Each of the four rows of the matrix is shifted to the left. Any entries that 'fall off' are re-inserted on the right side of row. Shift is carried out as follows :

- The first row is not shifted.
- The second row is shifted one (byte) position to the left.
- The third row is shifted two positions to the left.
- The fourth row is shifted three positions to the left.

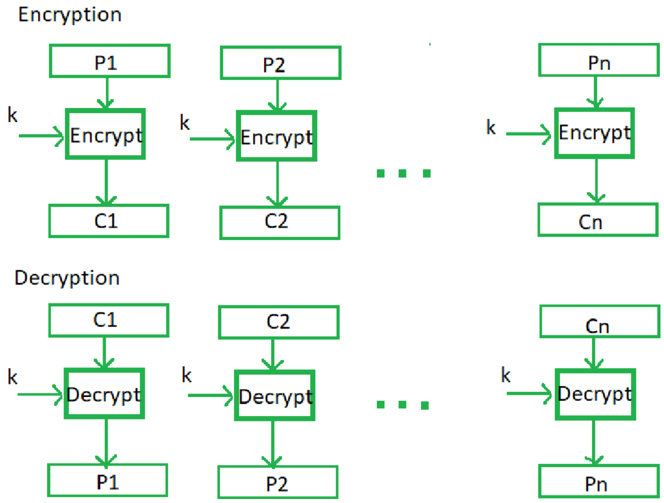
MixColumns: Each column of four bytes is now transformed using a special mathematical function. This function takes as input the four bytes of one column and outputs four completely new bytes, which replace the original column. The result is another new matrix consisting of 16 new bytes. It should be noted that this step is not performed in the last round.

Addroundkey: The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the ciphertext. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round

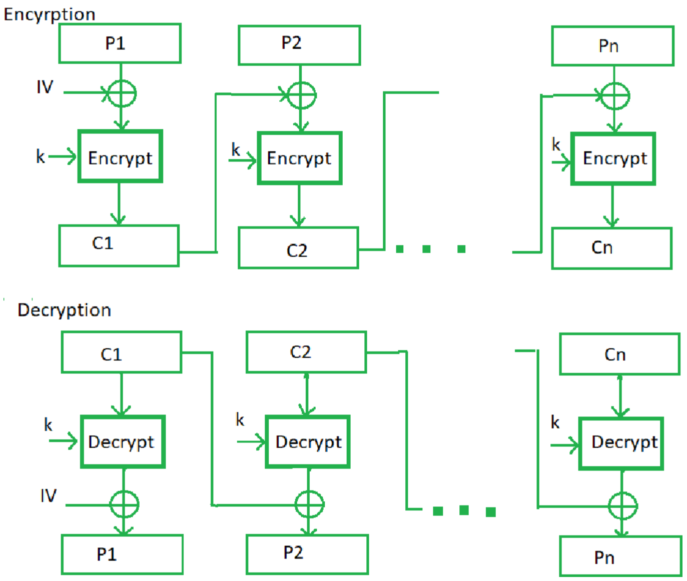
Q20) Explain are the following different modes of operation in DES

1. Electronic Code Book (ECB)
2. Cipher Block Chaining (CBC)
3. Cipher Feedback (CFB)
4. Output Feedback (OFB)
5. Counter Mode

Electronic code book: Electronic code book is the easiest block cipher mode of functioning. It is easier because of direct encryption of each block of input plaintext and output is in form of blocks of encrypted ciphertext. Generally, if a message is larger than b bits in size, it can be broken down into a bunch of blocks and the procedure is repeated.The process is illustrated below:



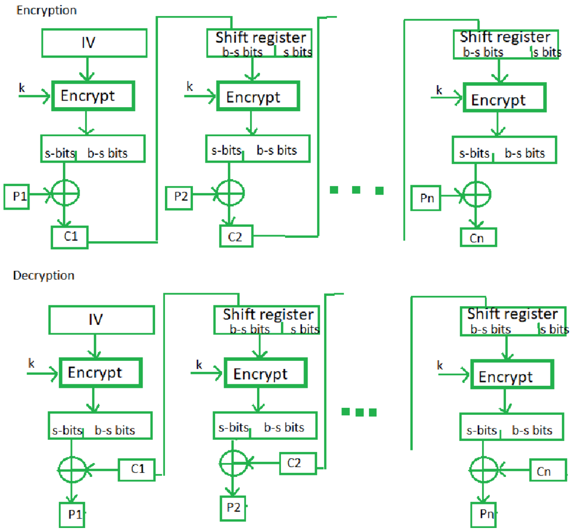
Cipher Block Chaining: Cipher block chaining or CBC is an advancement made on ECB since ECB compromises some security requirements. In CBC, the previous cipher block is given as input to the next encryption algorithm after XOR with the original plaintext block. In a nutshell here, a cipher block is produced by encrypting an XOR output of the previous cipher block and present plaintext block.The process is illustrated here:



Cipher feedback mode

In this mode the cipher is given as feedback to the next block of encryption with some new specifications: first, an initial vector IV is used for first encryption and output bits are divided as a set of s and b-s bits.The left-hand side s bits are selected along with plaintext bits to which an XOR operation is applied. The result is given as input to a shift register having b-s bits to lhs,s bits to rhs and the process continues. The encryption and decryption process for the same is shown below, both of them use encryption algorithms.

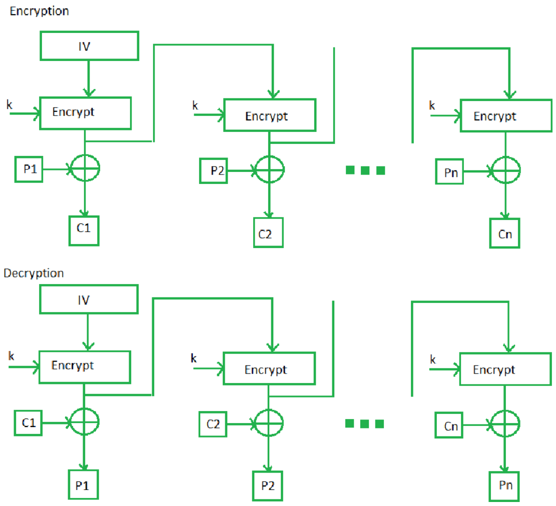
The process is illustrated below:



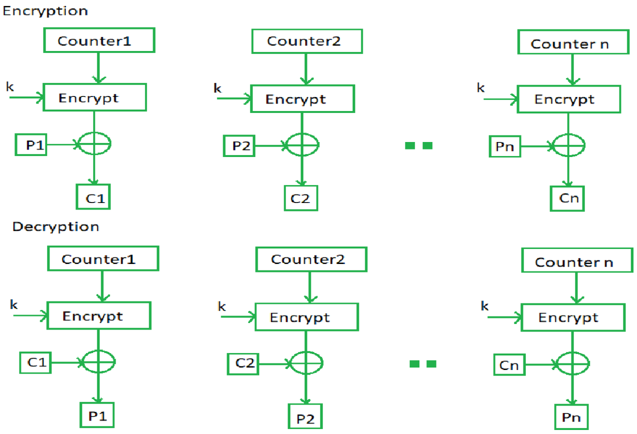
Output feedback mode:

The output feedback mode follows nearly the same process as the Cipher Feedback mode except that it sends the encrypted output as feedback instead of the actual cipher which's XOR output. In this output feedback mode, all bits of the block are sent instead of sending selected s bits. The Output Feedback mode of block cipher holds great resistance towards bit transmission errors. It also decreases the dependency or relationship of the cipher on the plaintext.

The process is illustrated below:



Counter mode: The Counter Mode or CTR is a simple counter-based block cipher implementation. Every time a counter-initiated value is encrypted and given as input to XOR with plaintext which results in ciphertext block. The CTR mode is independent of feedback use and thus can be implemented in parallel.The process is illustrated below:



Q. CLassical cryptosystem and types

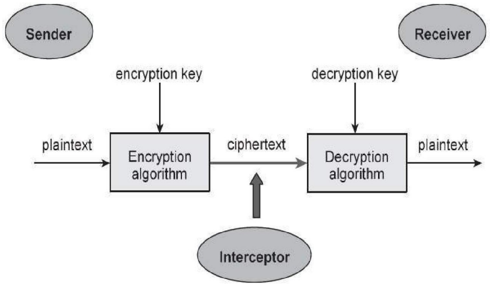
A cryptosystem is an implementation of cryptographic techniques and their accompanying infrastructure to provide information security services. A cryptosystem is also referred to as a cipher system. Cryptosystem takes sole responsibility to deliver the message to the authorized receiver only. It protects information from any leakage by protecting with encrypted codes.

Working of a cryptosystem:

There are two terminals, one is Sender end and another one is Receiver end.

step 1: At sender's end, Encryption system generates Cipher text as output on getting message (plain text) and encryption key as input.

Step 2: At receiver's end, Decryption system gets Cipher text as input from sender's terminal and it gets mixed with Decryption key. After further processing authorized user at this terminal receives the original message sent by sender.



The objective of this simple cryptosystem is that at the end of the process, only the sender and the receiver will know the plaintext.

Components :

Plaintext: It is the data to be protected during transmission.

Encryption Algorithm: It is a mathematical process that produces a ciphertext for any given plaintext and encryption key. It is a cryptographic algorithm that takes plaintext and an encryption key as input and produces a ciphertext.

Ciphertext: It is the scrambled version of the plaintext produced by the encryption algorithm using a specific the encryption key. The ciphertext is not guarded. It flows on public channel. It can be intercepted or compromised by anyone who has access to the communication channel.

Decryption Algorithm: It is a mathematical process, that produces a unique plaintext for any given ciphertext and decryption key. It is a cryptographic algorithm that takes a ciphertext and a decryption key as input, and outputs a plaintext. The decryption algorithm essentially reverses the encryption algorithm and is thus closely related to it.

Encryption Key:It is a value that is known to the sender. The sender inputs the encryption key into the encryption algorithm along with the plaintext in order to compute the ciphertext.

Decryption Key:It is a value that is known to the receiver. The decryption key is related to the encryption key, but is not always identical to it. The receiver inputs the decryption key into the decryption algorithm along with the ciphertext in order to compute the plaintext.

Types of cryptosystem :

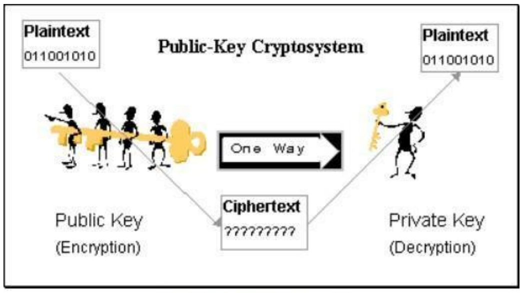
Symmetric key encryption

Asymmetric key encryption

Symmetric key encryption: In this process of cryptosystem, both sender and receiver use the same key for the encryption as well as decryption of the cipher text. This system is also referred as Symmetric key cryptosystem and Symmetric cryptographic. Symmetric key encryption system is highly used because of certain importance in Cryptography since early years . Each sender and receiver has to establish a secret symmetry key before any communication with this system. Apart from this, if receiver loses the key to any interceptor then he must prior inform this incident to sender to avoid leakage of plain text. Ex: BLOWFISH, Triple-DES, Digital Encryption Standard (DES) and IDEA.

Asymmetric key encryption:In this process of Cryptosystem, both sender and receiver use the different key for the encryption and decryption of the information. Concept of Public key and Private key comes into picture for Encryption and Decryption of the information.

Example: There are two hosts, Host 1 and Host 2. At sender's terminal, if host 1 wants to send information to host 2, then Host 1 will use Public key of Host 2 to encrypt the plain text. At receiver's terminal, host 2 will decrypt the cipher text by using its own Private key.



Asymmetric Cryptosystem

Challenges of symmetric

Key establishment:Before any communication, both the sender and the receiver need to agree on a secret symmetric key. It requires a secure key establishment mechanism in place.

Trust Issue:Since the sender and the receiver use the same symmetric key, there is an implicit requirement that the sender and the receiver 'trust' each other. For example, it may happen that the receiver has lost the key to an attacker and the sender is not informed.

Q. Firewalls

Firewall defined

A firewall is a network security device that monitors incoming and outgoing network traffic and permits or blocks data packets based on a set of security rules. Its purpose is to establish a barrier between your internal network and incoming traffic from external sources (such as the internet) in order to block malicious traffic like viruses and hackers.

How does a firewall work?

Firewalls carefully analyze incoming traffic based on pre-established rules and filter traffic coming from unsecured or suspicious sources to prevent attacks. Firewalls guard traffic at a computer's entry point, called ports, which is where information is exchanged with external devices. For example, "Source address 172.18.1.1 is allowed to reach destination 172.18.2.1 over port 22."

Think of IP addresses as houses, and port numbers as rooms within the house. Only trusted people (source addresses) are allowed to enter the house (destination address) at all—then it's further filtered so that people within the house are only allowed to access certain rooms (destination ports), depending on if they're the owner, a child, or a guest. The owner is allowed to any room (any port), while children and guests are allowed into a certain set of rooms (specific ports).

Types of firewalls

Firewalls can either be software or hardware, though it's best to have both. A software firewall is a program installed on each computer and regulates traffic through port numbers and applications, while a physical firewall is a piece of equipment installed between your network and gateway.

Next-generation firewalls (NGFW) combine traditional firewall technology with additional functionality, such as encrypted traffic inspection, intrusion prevention systems, anti-virus, and more. Most notably, it includes deep packet inspection (DPI). While basic firewalls only look at packet headers, deep packet inspection examines the data within the packet itself, enabling users to more effectively identify, categorize, or stop packets with malicious data

Proxy firewalls filter network traffic at the application level. Unlike basic firewalls, the proxy acts an intermediary between two end systems. The client must send a request to the firewall, where it is then evaluated against a set of security rules and then permitted or blocked. Most notably, proxy firewalls monitor traffic for layer 7 protocols such as HTTP and FTP, and use both stateful and deep packet inspection to detect malicious traffic.

Network address translation (NAT) firewalls allow multiple devices with independent network addresses to connect to the internet using a single IP address, keeping individual IP addresses hidden. As a result, attackers scanning a network for IP addresses can't capture specific details, providing greater security against attacks. NAT firewalls are similar to proxy firewalls in that they act as an intermediary between a group of computers and outside traffic.

Stateful multilayer inspection (SMLI) firewalls filter packets at the network, transport, and application layers, comparing them against known trusted packets. Like NGFW firewalls, SMLI also examine the entire packet and only allow them to pass if they pass each layer individually. These firewalls examine packets to determine the state of the communication (thus the name) to ensure all initiated communication is only taking place with trusted sources.

INTRUDERS

Intruders are often referred to as hackers and are the most harmful factors contributing to the vulnerability of security. They have immense knowledge and an in-depth understanding of technology and security. Intruders breach the privacy of users and aim at stealing the confidential information of the users. The stolen information is then sold to third-party, which aim at misusing the information for their own personal or professional gains.

Intruders are divided into three categories:

- **Masquerader:** The category of individuals that are not authorized to use the system but still exploit user's privacy and confidential information by possessing techniques that give them control over the system, such category of intruders is referred to as Masquerader. Masqueraders are outsiders and hence they don't have direct access to the system, their aim is to attack unethically to steal data/ information.
- **Misfeasor:** The category of individuals that are authorized to use the system, but misuse the granted access and privilege. These are individuals that take undue advantage of the permissions and access given to them, such category of intruders is referred to as Misfeasor. Misfeasors are insiders and they have direct access to the system, which they aim to attack unethically for stealing data/ information.
- **Clandestine User:** The category of individuals those have supervision/administrative control over the system and misuse the authoritative power given to them. The misconduct of power is often done by superlative authorities for financial gains, such a category of intruders is referred to as Clandestine User. A Clandestine User can be any of the two, insiders or outsiders, and accordingly, they can have direct/ indirect access to the system, which they aim to attack unethically by stealing data/ information.

VIRUSES

What is a Computer Virus?

A computer virus is a piece of code embedded in a legitimate program and is created with the ability to self-replicate infecting other programs on a computer. Just like how humans catch a cold or flu, it can remain dormant inside the system and gets activated when you least expect it. A computer virus is developed to spread from one host to another and there are numerous ways on how your computer catches it. It can be through email attachments, file downloads, software installations, or unsecured links.

These viruses can steal your data such as passwords, hacked into your social media accounts or online banking accounts, and even wiped out all your data.

Common Types Of Computer Viruses

Cybercriminals are getting better and better at stealing our confidential data and viruses that are being created are evolving rapidly. There are millions of viruses around the world, but here are some common types you should be aware of:

1. File-infecting Virus

A virus that attached itself to an executable program. It is also called a parasitic virus which typically infects files with .exe or .com extensions. Some file infectors can overwrite host files and others can damage your hard drive's formatting.

2. Macro Virus

This type of virus is commonly found in programs such as Microsoft Word or Excel. These viruses are usually stored as part of a document and can spread when the files are transmitted to other computers, often through email attachments.

3. Browser Hijacker

This virus targets and alters your browser setting. It is often called a browser redirect virus because it redirects your browser to other malicious websites that you don't have any intention of visiting. This virus can pose other threats such as changing the default home page of your browser.

4. Web Scripting Virus

A very sneaky virus that targets popular websites. What this virus does is overwrite code on a website and insert links that can install malicious software on your device. Web scripting viruses can steal your cookies and use the information to post on your behalf on the infected website.

5. Boot Sector Virus

These viruses are once common back when computers are booted from floppy disks. Today, these viruses are found distributed in forms of physical media such as external hard drives or USB. If the computer is infected with a boot sector virus, it automatically loads into the memory enabling control of your computer.

6. Polymorphic Virus

This virus has the capability to evade anti-virus programs since it can change codes every time an infected file is performed.

7. Resident Virus

A resident virus stores itself on your computer's memory which allows it to infect files on your computer. This virus can interfere with your operating system leading to file and program corruption.

8. Multipartite Virus

A type of virus that is very infectious and can easily spread on your computer system. It can infect multiple parts of a system including memory, files, and boot sector which makes it difficult to contain.

Q. Pretty Good Privacy

ELECTRONIC MAIL SECURITY PRETTY GOOD PRIVACY (PGP)

PGP provides the confidentiality and authentication service that can be used for electronic mail and file storage applications.

The steps involved in PGP are Select the best available cryptographic algorithms as building blocks.

Integrate these algorithms into a general purpose application that is independent of operating system and processor and that is based on a small set of easy-to-use commands. Make the package and its documentation, including the source code, freely available via the internet, bulletin boards and commercial networks. Enter into an agreement with a company to provide a fully compatible, low cost commercial version of PGP.

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

· It is available free worldwide in versions that run on a variety of platform.

· It is based on algorithms that have survived extensive public review and are considered extremely secure.

· e.g., RSA, DSS and Diffie Hellman for public key encryption CAST-128, IDEA and 3DES for conventional encryption SHA-1 for hash coding. · it has a wide range of applicability.

· It was not developed by, nor it is controlled by, any governmental or standards organization

. **Operational description**

The actual operation of PGP consists of five services: authentication, confidentiality, compression, e-mail compatibility and segmentation.

1. Authentication The sequence for authentication is as follows: The sender creates the message SHA-1 is used to generate a 160-bit hash code of the message The hash code is encrypted with RSA using the sender's private key and the result is prepended to the message The receiver uses RSA with the sender's public key to decrypt and recover the hash code. The receiver generates a new hash code for the message and compares it with the decrypted hash code. If the two match, the message is accepted as authentic.

2. Confidentiality Confidentiality is provided by encrypting messages to be transmitted or to be stored locally as files. In both cases, the conventional encryption algorithm CAST-128 may be used. The 64-bit cipher feedback (CFB) mode is used. In PGP, each conventional key is used only once. That is, a new key is generated as a random 128- bit number for each message. Thus although this is referred to as a session key, it is in reality a one time key. To protect the key, it is encrypted with the receiver's public key.

The sequence for confidentiality is as follows:

The sender generates a message and a random 128-bit number to be used as a session key for this message only. · The message is encrypted using CAST-128 with the session key. · The session key is encrypted with RSA, using the receiver's public key and is prepended to the message. · The receiver uses RSA with its private key to decrypt and recover the session key. · The session key is used to decrypt the message.

Confidentiality and authentication

Here both services may be used for the same message. First, a signature is generated for the plaintext message and prepended to the message. Then the plaintext plus the signature is encrypted using CAST-128 and the session key is encrypted using RSA

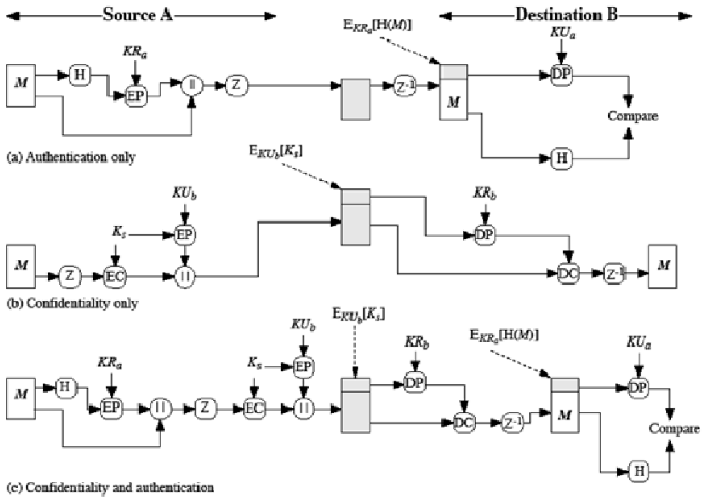


Figure 15.1 PGP Cryptographic Functions

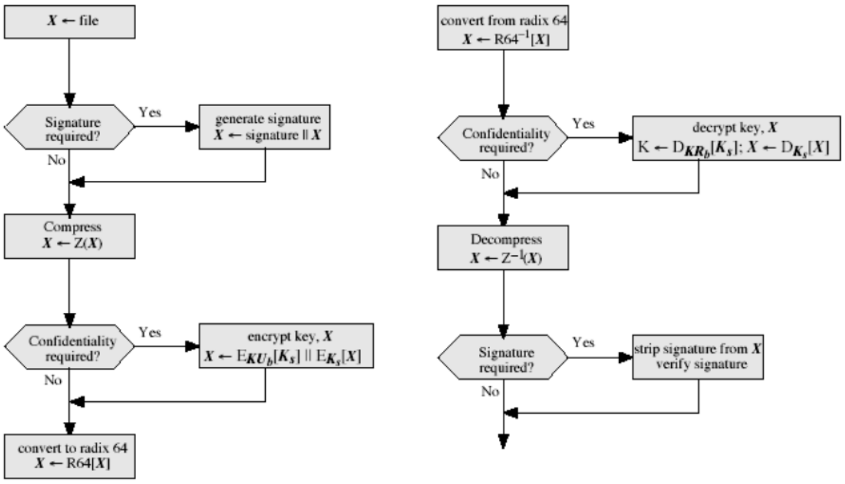
3. Compression As a default, PGP compresses the message after applying the signature but before encryption. This has the benefit of saving space for both e-mail transmission and for file storage. The signature is generated before compression for two reasons: It is preferable to sign an uncompressed message so that one can store only the uncompressed message together with the signature for future verification. If one signed a compressed document, then it would be necessary either to store a compressed version of the message for later verification or to recompress the message when verification is required. Even if one were willing to generate dynamically a recompressed message fro verification, PGP's compression algorithm presents a difficulty. The algorithm is not deterministic; various implementations of the algorithm achieve different tradeoffs in running speed versus compression ratio and as a result, produce different compression forms. Message encryption is applied after compression to strengthen cryptographic security. Because the compressed message has less redundancy than the original plaintext, cryptanalysis is more difficult. The compression algorithm used is ZIP.

4. e-mail compatibility Many electronic mail systems only permit the use of blocks consisting of ASCII texts. To accommodate this restriction, PGP provides the service of converting the raw 8-bit binary stream to a stream of printable ASCII characters. The scheme used for this purpose is radix-64 conversion. Each group of three octets of binary data is mapped into four ASCII characters. e.g., consider the 24-bit (3 octets) raw text sequence 00100011 01011100 10010001, we can express this input in block of 6-bits to produce 4 ASCII characters.

5. Segmentation and reassembly E-mail facilities often are restricted to a maximum length. E.g., many of the facilities accessible through the internet impose a maximum length of 50,000 octets. Any message longer than that must be broken up into smaller segments, each of which is mailed separately. To accommodate this restriction, PGP automatically subdivides a message that is too large into segments that are small enough to send via e-mail. The segmentation is done after all the other processing, including the radix-64 conversion. At the receiving end, PGP must strip off all e-mail headers and reassemble the entire original block before performing the other steps.

Diagram hai iske baad

PGP Operation Summary:



(a) Generic Transmission Diagram (from A)

(b) Generic Reception Diagram (to B)

Q. KERBEROS

AUTHENTICATION SERVICES KERBEROS Kerberos provides a centralized authentication server whose function is to authenticate users to servers and servers to users. Kerberos relies exclusively on conventional encryption, making no use of public-key encryption.
The following are the requirements for Kerberos: · **Secure**: A network eavesdropper should not be able to obtain the necessary information to impersonate a user. More generally, Kerberos should be strong enough that a potential opponent does not find it to be the weak link. · **Reliable:** For all services that rely on Kerberos for access control, lack of availability of the Kerberos service means lack of availability of the supported services. Hence, Kerberos should be highly reliable and should employ a distributed server architecture, with one system able to back up another. · **Transparent:** Ideally, the user should not be aware that authentication is taking place, beyond the requirement to enter a password. · **Scalable**: The system should be capable of supporting large numbers of clients and servers. This suggests a modular, distributed architecture.
To support these requirements, the overall scheme of Kerberos is that of a trusted third- party authentication service that uses a protocol based on that proposed by Needham and Schroeder [NEED78] It is trusted in the sense that clients and servers trust Kerberos to mediate their mutual authentication. Assuming the Kerberos protocol is well designed, then the authentication service is secure if the Kerberos server itself is secure

Q.Primality test on rsa

Primality Testing and RSA · The first stage of key-generation for RSA involves finding two large primes p, q · Because of the size of numbers used, must find primes by trial and error · Modern primality tests utilize properties of primes eg:
a^n-1 = 1 mod n where GCD(a,n)=1
all primes numbers 'n' will satisfy this equation
some composite numbers will also satisfy the equation, and are called pseudoprimes.
· Most modern tests guess at a prime number 'n', then take a large number (eg 100) of numbers 'a', and apply this test to each. If it fails the number is composite, otherwise it is is probably prime.
· There are a number of stronger tests which will accept fewer composites as prime than the above test. eg:

$$\mathrm{GCD}(a,n) = 1, \quad \text{and} \quad \left(\frac{a}{n}\right) (\mathrm{mod}\ n) = a^{\frac{(n-1)}{2}} (\mathrm{mod}\ n)$$

$$\text{where} \left(\frac{a}{n}\right) \text{is the Jacobi symbol}$$

Q. ELGAMAL

ElGamal · A variant of the Diffie-Hellman key distribution scheme, allowing secure exchange of messages · published in 1985 by ElGamal in T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms", IEEE Trans. Information Theory, vol IT-31(4), pp469-472, July 1985. · like Diffie-Hellman its security depends on the difficulty of factoring logarithms
Key Generation o select a large prime p (~200 digit), and o [[alpha]] a primitive element mod p o A has a secret number xA o B has a secret numbr xB o A and B compute yA and yB respectively, which are then made public ß yA = [[alpha]]xA mod p ß yB = [[alpha]]xB mod p
to encrypt a message M into ciphertext C, o selects a random number k, 0 <= k <= p-1 o computes the message key K ß K = yB k mod p o computes the ciphertext pair: C = {c1,c2} ß C1 = [[alpha]]k mod p C2 = K.M mod p
· to decrypt the message o extracts the message key K ß K = C1 xB mod p = [[alpha]]k.xB mod p o extracts M by solving for M in the following equation: ß C2 = K.M mod p

Q.S-box (substitution-box)

In cryptography, an S-box (substitution-box) is a basic component of symmetric key algorithms which performs substitution. an S-box takes some number of input bits, m, and transforms them into some number of output bits, n, where n is not necessarily equal to m. An m×n S-box can be implemented as a lookup table with 2m words of n bits each. Fixed tables are normally used, as in the Data Encryption Standard (DES), but in some ciphers the tables are generated dynamically from the key .
S-Boxes are Boolean mappings from {0,1}mÆ{0,1}n – m x n mappings • Thus there are n component functions each being a map from m bits to 1 bit – in other words, each component function is a Boolean function in m Boolean variables
• S-P networks are based on the two primitive cryptographic operations we have seen before: • confusion and diffusion of message • diffusion – dissipates statistical structure of plaintext over bulk of ciphertext • confusion – makes relationship between ciphertext and key as complex as possible.S-Boxes provide confusion of input bits P-Boxes provide diffusion across S-box inputs

Q.What is a stream cipher?

A stream cipher is a method of encrypting text (to produce ciphertext) in which a cryptographic key and algorithm are applied to each binary digit in a data stream, one bit at a time. The main alternative method to stream cipher is, in fact, the block cipher, where a key and algorithm are applied to blocks of data rather than individual bits in a stream.
How does a stream cipher work?
A stream cipher is an encryption algorithm that uses a symmetric key to encrypt and decrypt a given amount of data. A symmetric cipher key, as opposed to an asymmetric cipher key, is an encryption tool that is used in both encryption and decryption. Asymmetric keys will sometimes use one key to encrypt a message and another to decrypt the respective ciphertext.
What makes stream ciphers particularly unique is that they encrypt data one bit, or byte, at a time. This makes for a fast and relatively simple encryption process.
Basic encryption requires three main components:
a message, document or piece of data
a key
an encryption algorithm
The key typically used with a stream cipher is known as a one-time pad. Mathematically, a one-time pad is unbreakable because it's always at least the exact same size as the message it is encrypting.
Stream Modes
Cipher Feedback (CFB) - Where the message is treated as a stream of bits, added to the output of the DES, with the result being feedback for the next stage C_(i) = P_(i)(+) DES_(K1) (C_(i-1)) C_(-1)=IV

Output Feedback (OFB) - Where the message is treated as a stream of bits, added to the message, but with the feedback being independent of the message C_(i) = P_(i)(+) O_(i) O_(i) = DES_(K1)(O_(i-1)) O_(-1)=IV · each mode has its advantages and disadvantages

Limitations of Various Modes

ECB · repetitions in message can be reflected in ciphertext o if aligned with message block o particularly with data such graphics o or with messages that change very little, which become a code-book analysis problem · weakness is because enciphered message blocks are independent of each other

CBC · use result of one encryption to modify input of next o hence each ciphertext block is dependent on all message blocks before it o thus a change in the message affects the ciphertext block after the change as well as the original block to start need an Initial Value (IV) which must be known by both sender and receiver o however if IV is sent in the clear, an attacker can change bits of the first block, and change IV to compensate o hence either IV must be a fixed value (as in EFTPOS) or it must be sent encrypted in ECB mode before rest of message · also at the end of the message, have to handle a possible last short block o either pad last block (possible with count of pad size),

CFB · when data is bit or byte oriented, want to operate on it at that level, so use a stream mode · the block cipher is use in encryption mode at both ends, with input being a feed-back copy of the ciphertext · can vary the number of bits feed back, trading off efficiency for ease of use · again errors propogate for several blocks after the error

Q.Security threat

Threat can be anything that can take advantage of a vulnerability to breach security and negatively alter, erase, harm object or objects of interest

Software attacks means attack by Viruses, Worms, Trojan Horses etc. Many users believe that malware, virus, worms, bots are all same things. But they are not same, only similarity is that they all are malicious software that behaves differently.

Malware is a combination of 2 terms- Malicious and Software. So Malware basically means malicious software that can be an intrusive program code or anything that is designed to perform malicious operations on system. Malware can be divided in 2 categories:

Infection Methods

Malware Actions

Malware on the basis of Infection Method are following:

Virus – They have the ability to replicate themselves by hooking them to the program on the host computer like songs, videos etc and then they travel all over the Internet. The Creeper Virus was first detected on ARPANET. Examples include File Virus, Macro Virus, Boot Sector Virus, Stealth Virus etc.

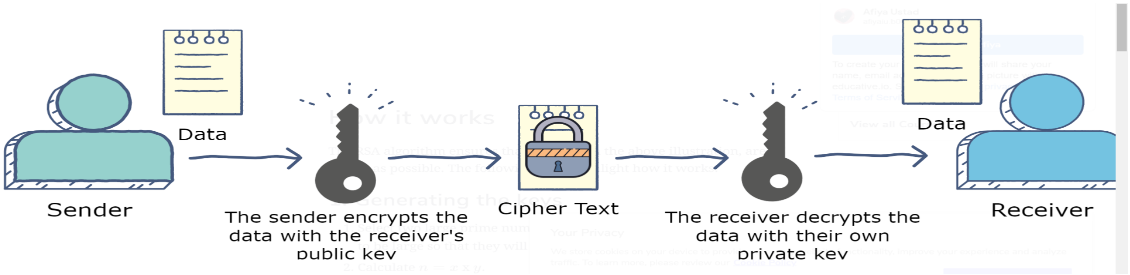
Worms – Worms are also self-replicating in nature but they don't hook themselves to the program on host computer. Biggest difference between virus and worms is that worms are network-aware. They can easily travel from one computer to another if network is available and on the target machine they will not do much harm, they will, for example, consume hard disk space thus slowing down the computer.

Q.RSA Rivest-Shamir-Adleman

he RSA algorithm is an asymmetric cryptography algorithm; this means that it uses a public key and a private key. As their names suggest, a public key is shared publicly, while a private key is secret and must not be shared with anyone.

The RSA algorithm is named after those who invented it in 1978: Ron Rivest, Adi Shamir, and Leonard Adleman.

The RSA scheme is a block cipher in which the plaintext and ciphertext are integers between 0 and n-1 for some n. A typical size for n is 1024 bits, or 309 decimal digits.


Data
Sender
The sender encrypts the data with the receiver's public key
Cipher Text
The receiver decrypts the data with their own private key
Data
Receiver

Step 1: Generate the RSA modulus

The initial procedure begins with selection of two prime numbers namely p and q, and then calculating their product N, as shown −

N=p*q Here, let N be the specified large number.

Step 2: Derived Number (e)

Consider number e as a derived number which should be greater than 1 and less than (p-1) and (q-1). The primary condition will be that there should be no common factor of (p-1) and (q-1) except 1

Step 3: Public key

The specified pair of numbers n and e forms the RSA public key and it is made public.

Step 4: Private Key

Private Key d is calculated from the numbers p, q and e. The mathematical relationship between the numbers is as follows −

ed = 1 mod (p-1) (q-1)

The above formula is the basic formula for Extended Euclidean Algorithm, which takes p and q as the input parameters.

Encryption Formula

Consider a sender who sends the plain text message to someone whose public key is (n,e). To encrypt the plain text message in the given scenario, use the following syntax −

C = Pe mod n

Decryption Formula

The decryption process is very straightforward and includes analytics for calculation in a systematic approach. Considering receiver C has the private key d, the result modulus will be calculated as −

Plaintext = Cd mod n