

block cipher:

Encryption algorithms are divided into two categories based on the input type, as a block cipher and stream cipher. Block cipher is an encryption algorithm that takes a fixed size of input say b bits and produces a ciphertext of b bits again. If the input is larger than b bits it can be divided further. For different applications and uses, there are several modes of operations for a block cipher.

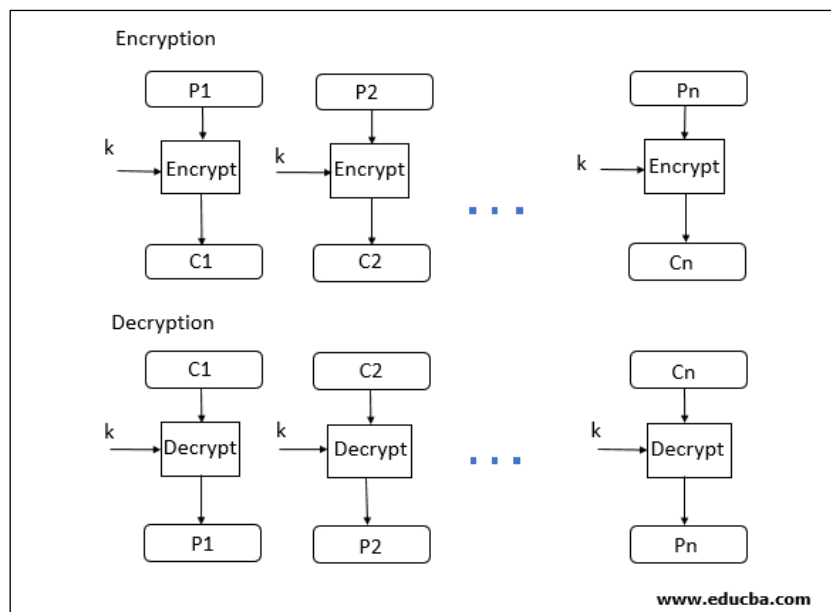
Electronic Code Block Mode:

ECB mode stands for Electronic Code Block Mode. It is one of the simplest modes of operation. In this mode, the plain text is divided into a block where each block is 64 bits.

Then each block is encrypted separately. The same key is used for the encryption of all blocks. Each block is encrypted using the key and makes the block of ciphertext.

At the receiver side, the data is divided into a block, each of 64 bits. The same key which is used for encryption is used for decryption. It takes the 64-bit ciphertext and, by using the key convert the ciphertext into plain text.

As the same key is used for all blocks' encryption, if the block of plain text is repeated in the original message, then the ciphertext's corresponding block will also repeat. As the same key used for all block, to avoid the repetition of block ECB mode is used for an only small message where the repetition of the plain text block is less.



Advantages of using ECB –

- Parallel encryption of blocks of bits is possible, thus it is a faster way of encryption.
- Simple way of the block cipher.

Disadvantages of using ECB –

- Prone to cryptanalysis since there is a direct relationship between plaintext and ciphertext.

Cipher Block Chaining:

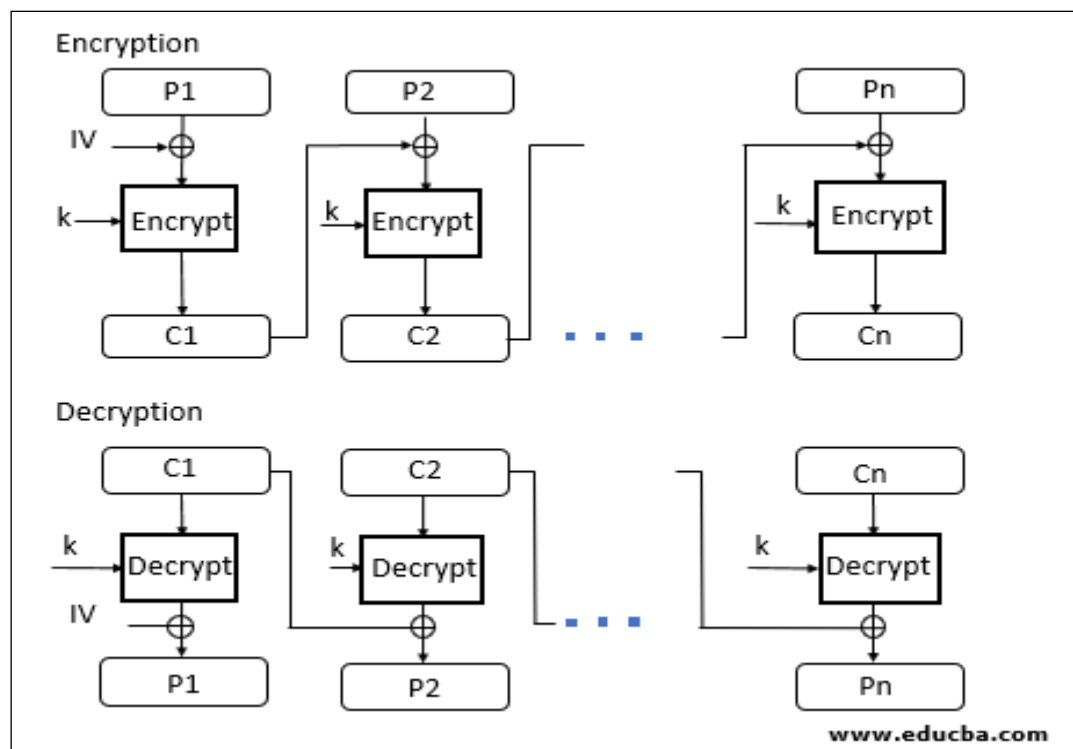
CBC Mode stands for Cipher block Mode at the sender side; the plain text is divided into blocks. In this mode, IV (Initialization Vector) is used, which can be a random block of text. IV is used to make the ciphertext of each block unique.

The first block of plain text and IV is combined using the XOR operation and then encrypted the resultant message using the key and form the first block of ciphertext. The first block of ciphertext is used as IV for the second block of plain text. The same procedure will be followed for all blocks of plain text.

At the receiver side, the ciphertext is divided into blocks. The first block ciphertext is decrypted using the same key, which is used for encryption. The decrypted result will be XOR with the IV and form the first block of plain text.

The second block of ciphertext is also decrypted using the same key, and the result of the decryption will be XOR with the first block of ciphertext and form the second block of plain text. The same procedure is used for all the blocks.

CBC Mode ensures that if the block of plain text is repeated in the original message, it will produce a different ciphertext for corresponding blocks.



Advantages of CBC –

- CBC works well for input greater than b bits.
- CBC is a good authentication mechanism.
- Better resistive nature towards cryptanalysis than ECB.

Disadvantages of CBC –

- Parallel encryption is not possible since every encryption requires a previous cipher.

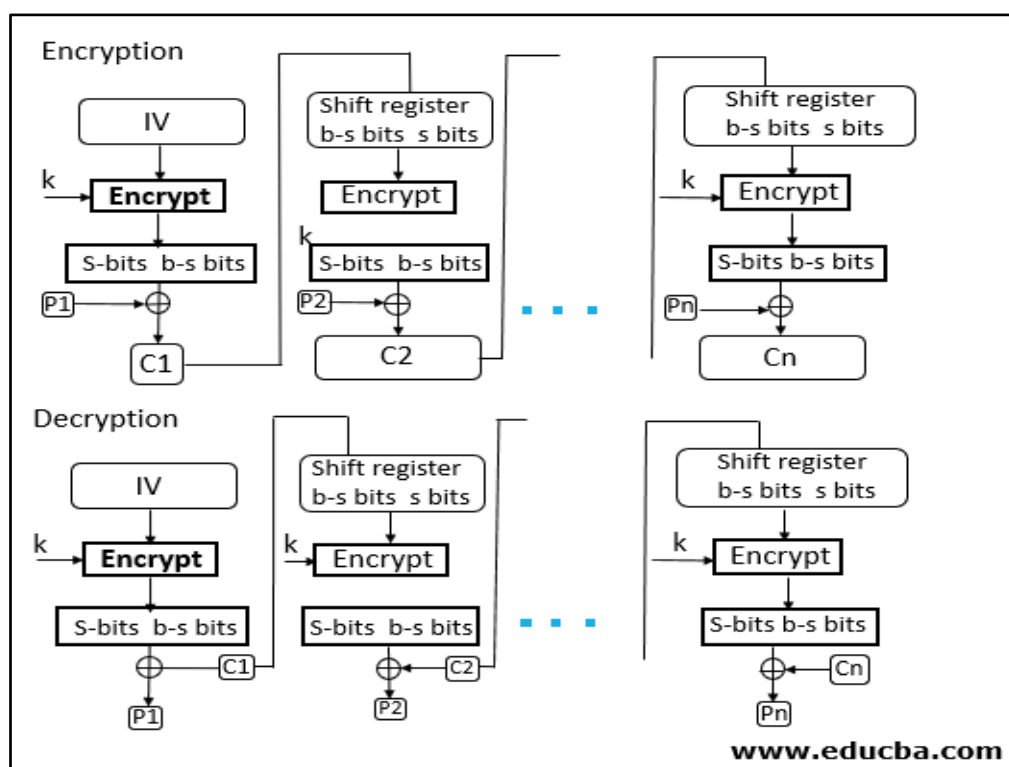
CFB Mode

CFB mode stands for Cipher Feedback Mode. In this mode, the data is encrypted in the form of units where each unit is of 8 bits.

Like cipher block chaining mode, IV is initialized. The IV is kept in the shift register. It is encrypted using the key and form the ciphertext.

Now the leftmost j bits of the encrypted IV is XOR with the plain text's first j bits. This process will form the first part of the ciphertext, and this ciphertext will be transmitted to the receiver.

Now the bits of IV is shifted left by j bit. Therefore, the rightmost j position of the shift register now has unpredictable data. These rightmost j positions are now filled with the ciphertext. The process will be repeated for all plain text units.



Advantages of CFB –

- Since, there is some data loss due to the use of shift register, thus it is difficult for applying cryptanalysis.

Disadvantages of using ECB –

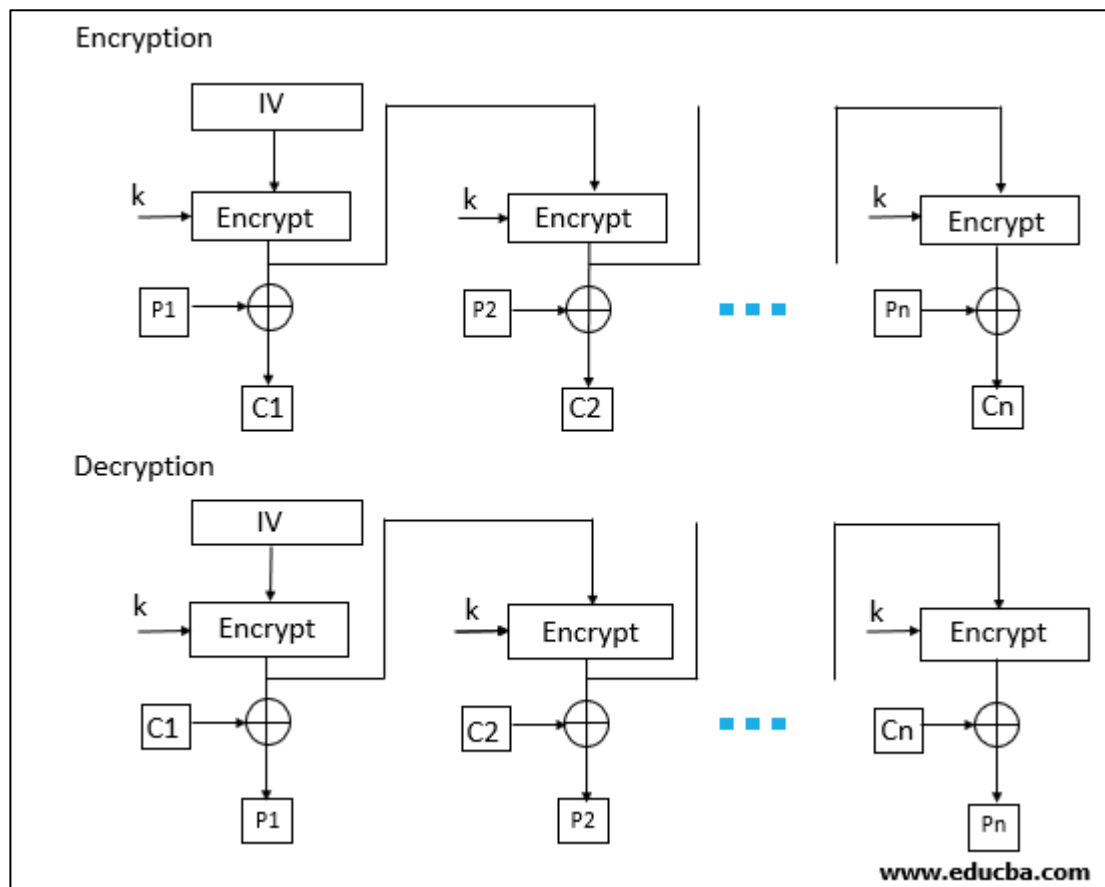
- The drawbacks of CFB are the same as those of CBC mode. Both block losses and concurrent encryption of several blocks are not supported by the encryption. Decryption, however, is parallelizable and loss-tolerant.

OFB mode:

OFB Mode stands for output feedback Mode. OFB mode is similar to CFB mode; the only difference is in CFB, the ciphertext is used for the next stage of the encryption process, whereas in OFB, the output of the IV encryption is used for the next stage of the encryption process.

The IV is encrypted using the key and forms encrypted IV. Plain text and leftmost 8 bits of encrypted IV are combined using XOR and produce the ciphertext.

For the next stage, the ciphertext, which is the form in the previous stage, is used as an IV for the next iteration. The same procedure is followed for all blocks.



Advantages of OFB –

- In the case of CFB, a single bit error in a block is propagated to all subsequent blocks. This problem is solved by OFB as it is free from bit errors in the plaintext block.

Disadvantages of OFB-

- The drawback of OFB is that, because to its operational modes, it is more susceptible to a message stream modification attack than CFB.

CTR Mode

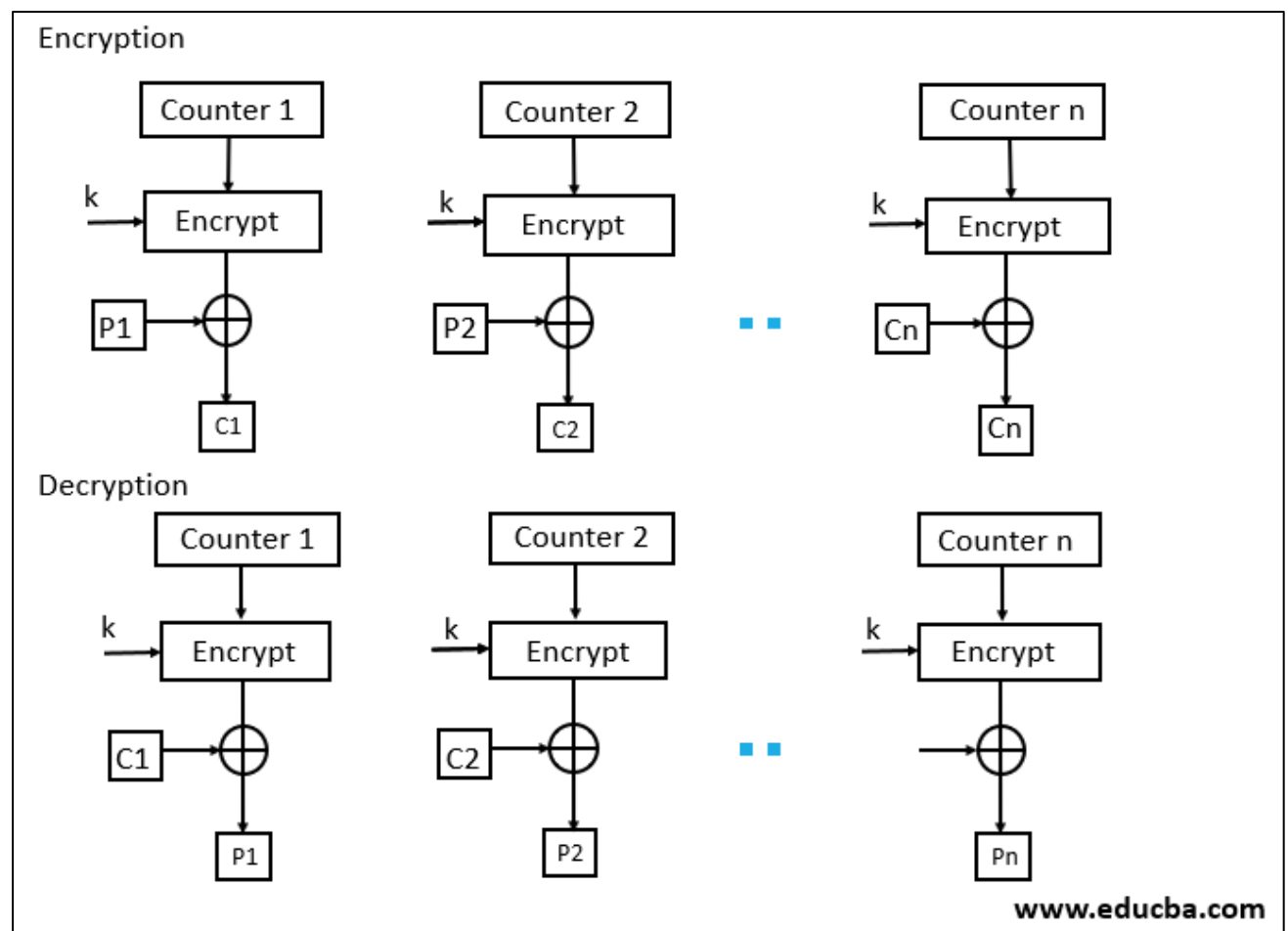
CTR Mode stands for counter mode. As the name is counter, it uses the sequence of numbers as an input for the algorithm. When the block is encrypted, to fill the next register next counter value is used.

Note: the counter value will be incremented by 1.

For encryption, the first counter is encrypted using a key, and then the plain text is XOR with the encrypted result to form the ciphertext.

The counter will be incremented by 1 for the next stage, and the same procedure will be followed for all blocks. For decryption, the same sequence will be used.

Here to convert ciphertext into plain text, each ciphertext is XOR with the encrypted counter. For the next stage, the counter will be incremented by the same will be repeated for all Ciphertext blocks.



Advantages of Counter –

- Since there is a different counter value for each block, the direct plaintext and ciphertext relationship is avoided. This means that the same plain text can map to different ciphertext.
- Parallel execution of encryption is possible as outputs from previous stages are not chained as in the case of CBC.

Disadvantages of Counter-

- The fact that CTR mode requires a synchronous counter at both the transmitter and the receiver is a severe drawback. The recovery of plaintext is erroneous when synchronisation is lost.

Data Encryption Standard (DES)

DES stands for Data Encryption Standard. There are certain machines that can be used to crack the DES algorithm. The DES algorithm uses a key of 56-bit size. Using this key, the DES takes a block of 64-bit plain text as input and generates a block of 64-bit cipher text.

The DES process has several steps involved in it, where each step is called a round. Depending upon the size of the key being used, the number of rounds varies. For example, a 128-bit key requires 10 rounds, a 192-bit key requires 12 rounds, and so on.

DES Algorithm Steps

To put it in simple terms, DES takes 64-bit plain text and turns it into a 64-bit ciphertext. And since we're talking about asymmetric algorithms, the same key is used when it's time to decrypt the text.

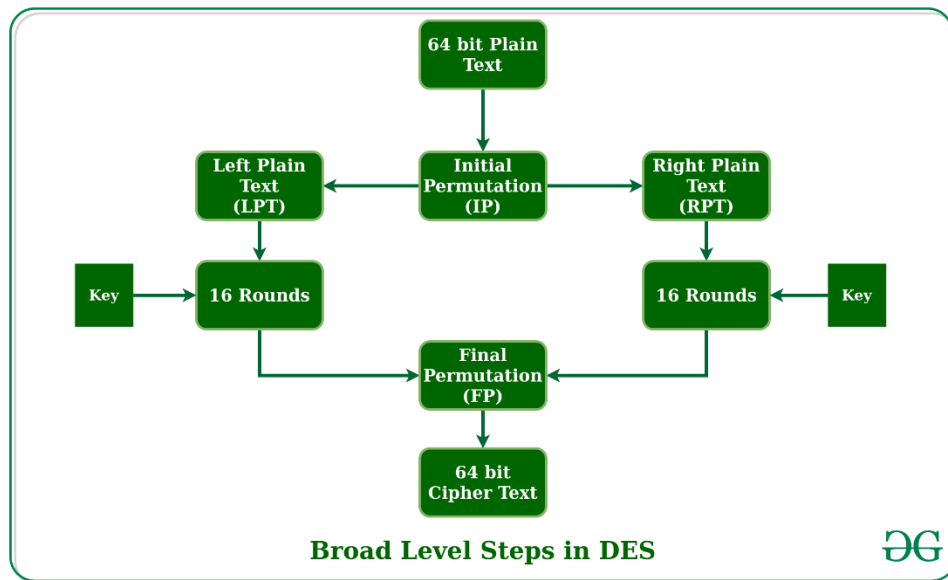
The algorithm process breaks down into the following steps:

- 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 (RPT).
- 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.
- The encryption process step (step 4, above) is further broken down into five stages:
 1. Key transformation
 2. Expansion permutation
 3. S-Box permutation
 4. P-Box permutation
 5. XOR and swap
- For decryption, we use the same algorithm, and we reverse the order of the 16 round keys.

History of DES Algorithm

DES is based on the Feistel block cipher, called LUCIFER, developed in 1971 by IBM cryptography researcher Horst Feistel. DES uses 16 rounds of the Feistel structure, using a different key for each round.

DES became the approved federal encryption standard in November 1976 and was subsequently reaffirmed as the standard in 1983, 1988, and 1999.



let us now discuss the broad-level steps in DES.

- In the first step, the 64-bit plain text block is handed over to an initial Permutation (IP) function.
- The initial permutation is performed on plain text.
- Next, the initial permutation (IP) produces two halves of the permuted block; saying Left Plain Text (LPT) and Right Plain Text (RPT).
- Now each LPT and RPT go through 16 rounds of the encryption process.
- In the end, LPT and RPT are rejoined and a Final Permutation (FP) is performed on the combined block
- The result of this process produces 64-bit ciphertext.

Initial Permutation (IP):

As we have noted, the initial permutation (IP) happens only once and it happens before the first round. It suggests how the transposition in IP should proceed, as shown in the figure. For example, it says that the IP replaces the first bit of the original plain text block with the 58th bit of the original plain text, the second bit with the 50th bit of the original plain text block, and so on.

Step-1: Key transformation:

We have noted initial 64-bit key is transformed into a 56-bit key by discarding every 8th bit of the initial key. Thus, for each a 56-bit key is available. From this 56-bit key, a different 48-bit Sub Key is generated during each round using a process called key transformation. For this, the 56-bit key is divided into two halves, each of 28 bits. These halves are circularly shifted left by one or two positions, depending on the round.

Step-2: Expansion Permutation:

Recall that after the initial permutation, we had two 32-bit plain text areas called Left Plain Text(LPT) and Right Plain Text(RPT). During the expansion permutation, the RPT is expanded from 32 bits to 48 bits. Bits are permuted as well hence called expansion permutation. This happens as the 32-bit RPT is divided into 8 blocks, with each block consisting of 4 bits. Then, each 4-bit block of the previous step is then expanded to a corresponding 6-bit block, i.e., per 4-bit block, 2 more bits are added.

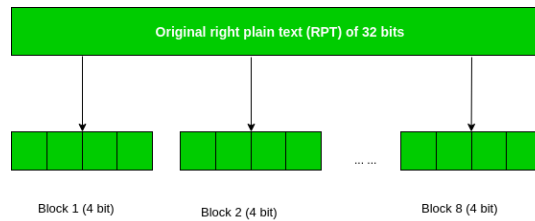


Figure - division of 32 bit RPT into 8 bit blocks

This process results in expansion as well as a permutation of the input bit while creating output. The key transformation process compresses the 56-bit key to 48 bits. Then the expansion permutation process expands the 32-bit RPT to 48-bits. Now the 48-bit key is XOR with 48-bit RPT and the resulting output is given to the next step, which is the S-Box substitution.

Advantages and Disadvantages of DES Algorithm

The advantages of the DES algorithm:

- It is set as a standard by the US government.
- When compared to the software, it works faster on hardware.
- Triple DES, used a 168-bit key which is very hard to crack.

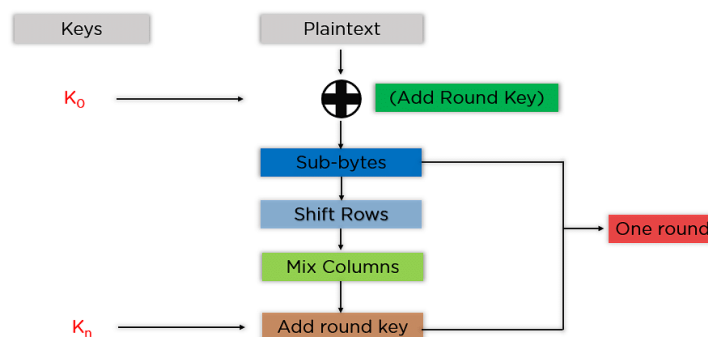
The **disadvantages** of the DES algorithm:

- Weakly secured algorithm.
- There is a threat from Brute force attacks.
- A DES cracker machine known as Deep Crack is available in the market.

What is the Advanced Encryption Standard?

The AES Encryption algorithm (also known as the Rijndael algorithm) is a symmetric block cipher algorithm with a block/chunk size of 128 bits. It converts these individual blocks using keys of 128, 192, and 256 bits. Once it encrypts these blocks, it joins them together to form the ciphertext.

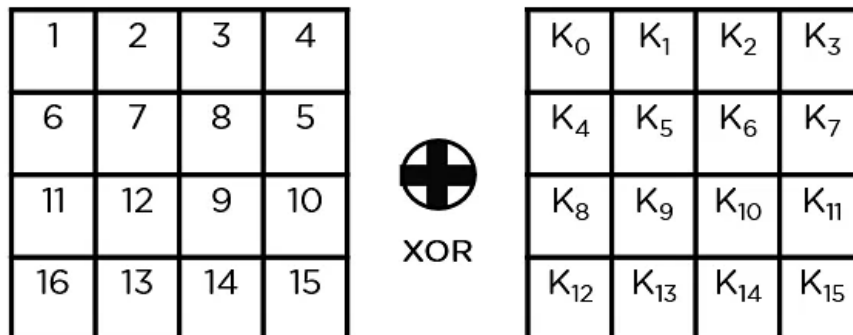
It is based on a substitution-permutation network, also known as an SP network. It consists of a series of linked operations, including replacing inputs with specific outputs (substitutions) and others involving bit shuffling (permutations).



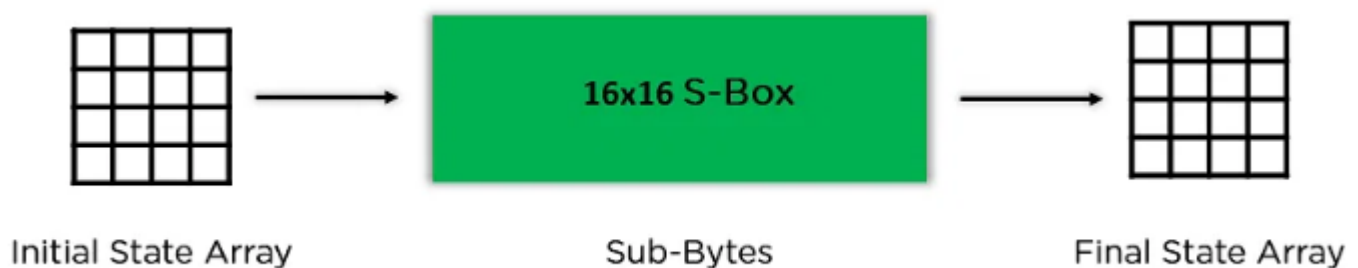
Steps to be followed in AES:

The mentioned steps are to be followed for every block sequentially. Upon successfully encrypting the individual blocks, it joins them together to form the final ciphertext. The steps are as follows:

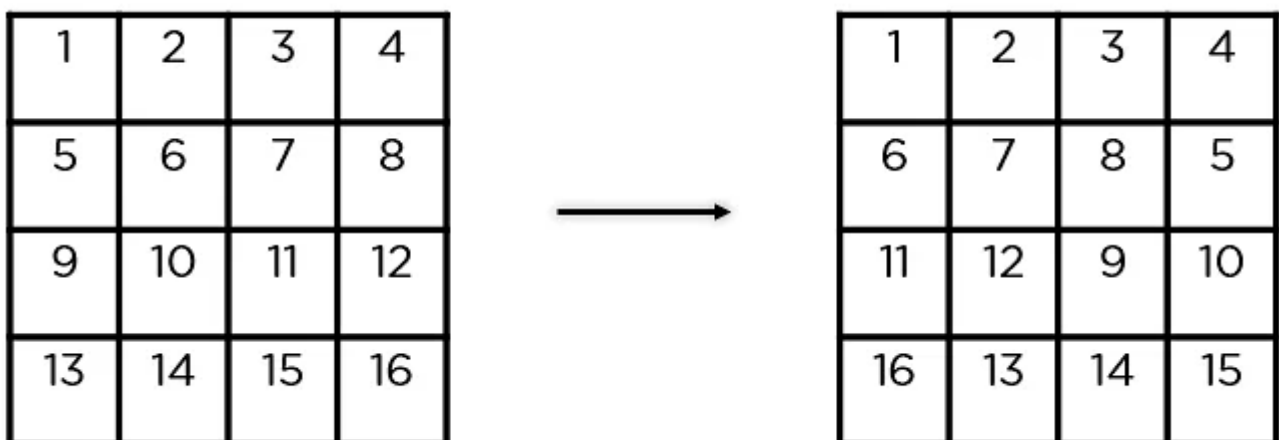
Add Round Key: You pass the block data stored in the state array through an XOR function with the first key generated (K_0). It passes the resultant state array on as input to the next step.



Sub-Bytes: In this step, it converts each byte of the state array into hexadecimal, divided into two equal parts. These parts are the rows and columns, mapped with a substitution box (S-Box) to generate new values for the final state array.



Shift Rows: It swaps the row elements among each other. It skips the first row. It shifts the elements in the second row, one position to the left. It also shifts the elements from the third row two consecutive positions to the left, and it shifts the last row three positions to the left.



Mix Columns: It multiplies a constant matrix with each column in the state array to get a new column for the subsequent state array. Once all the columns are multiplied with the same constant matrix, you get your state array for the next step. This particular step is not to be done in the last round.

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

 \times

C_0
C_1
C_2
C_3


 $=$

NC_0
NC_1
NC_2
NC_3

Constant Matrix Old Column New Column

Add Round Key: The respective key for the round is XOR'd with the state array is obtained in the previous step. If this is the last round, the resultant state array becomes the ciphertext for the specific block; else, it passes as the new state array input for the next round.

1	2	3	4
6	7	8	5
11	12	9	10
16	13	14	15


XOR

K_0	K_1	K_2	K_3
K_4	K_5	K_6	K_7
K_8	K_9	K_{10}	K_{11}
K_{12}	K_{13}	K_{14}	K_{15}

RSA Encryption Algorithm:

RSA encryption algorithm is a type of public-key encryption algorithm. To better understand RSA, let's first understand what is public-key encryption algorithm.

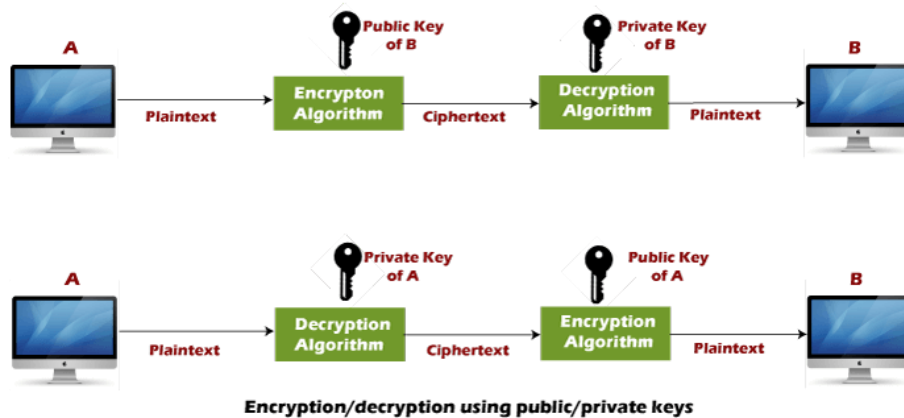
Public key encryption algorithm:

Public Key encryption algorithm is also called the 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
- Private key

The Public key is used for encryption, and the Private Key is used for decryption. Decryption cannot be done using a public key. The two keys are linked, but the private key cannot 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:



The algorithm works as Follows

1. Select two prime numbers a and b where $a \neq b$.
2. Calculate $n = a * b$
3. Calculate $\phi(n) = (a - 1) * (b - 1)$.
4. Select e such that, e is relatively prime to $\phi(n)$ i.e. $\gcd(e, \phi(n)) = 1$ and $1 < e < \phi(n)$.
5. Calculate d such that $d = e^{-1} \bmod \phi(n)$ or $ed \bmod \phi(n) = 1$.
6. Public key = $\{e, n\}$, private key = $\{d, n\}$.
7. Find out ciphertext using the formula,
 $C = P^e \bmod n$ where, $P < n$ and
 $C = \text{Ciphertext}$, $P = \text{Plaintext}$, $e = \text{Encryption key}$ and $n = \text{Block size}$.
8. $P = C^d \bmod n$. Plaintext P can be obtain using the given formula.

Where, $d = \text{decryption key}$.

Both sender and receiver know the value of n . In addition, the sender must know encryption key ' e ' and receiver must know decryption key ' d '.

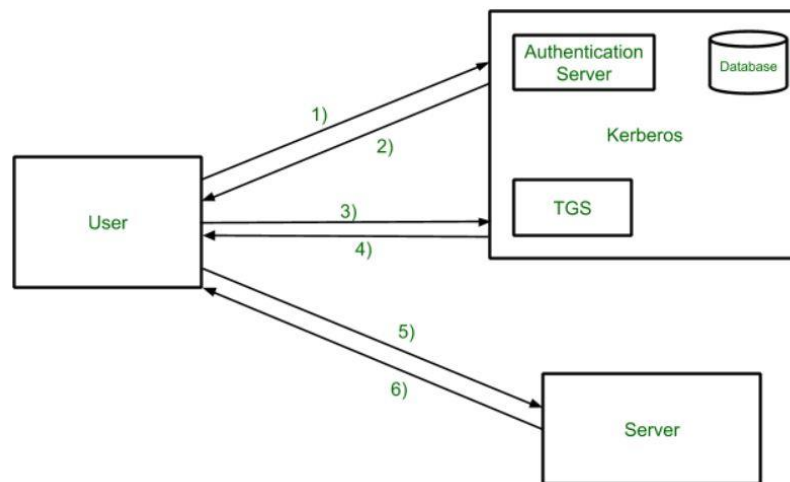
Kerberos

- Kerberos is a computer network security protocol that authenticates service requests between two or more trusted hosts across an untrusted network, like the internet.
- It uses secret-key cryptography and a trusted third party for authenticating client-server applications and verifying users' identities.
- Initially developed by the Massachusetts Institute of Technology (MIT) for Project Athena in the late '80s, Kerberos is now the default authorization technology used by Microsoft Windows.
- Kerberos implementations also exist for other operating systems such as Apple OS, FreeBSD, UNIX, and Linux.
- Microsoft rolled out its version of Kerberos in Windows 2000, and it's become the go-to protocol for websites and single sign-on implementations over different platforms.
- The Kerberos Consortium maintains the Kerberos as an open-source project.
- The protocol derives its name from the legendary three-headed dog Kerberos (also known as Cerberus) from Greek myths, the canine guardian to the entrance to the underworld.
- Kerberos had a snake tail and a particularly bad temper and, despite one notable exception, was a very useful guardian.
- But in the protocol's case, the three heads of Kerberos represent the client, the server, and the Key Distribution Center (KDC). The latter functions as the trusted third-party authentication service.

Components of Kerberos

- **Client (user)** : The client initiates communication for a service request.
- **Server** : The server hosts the service the user wants to access.
- **Authentication Server (AS)** : The AS performs the desired client authentication. If the authentication happens successfully, the AS issues the client a ticket called TGT (Ticket Granting Ticket). This ticket assures the other servers that the client is authenticated.
- **Key Distribution Center (KDC)** : In a Kerberos environment, the authentication server logically separated into three parts: A database (db), the Authentication Server (AS), and the Ticket Granting Server (TGS). These three parts, in turn, exist in a single server called the Key Distribution Center.
- **Ticket Granting Server (TGS)** : The TGS is an application server that issues the ticket for the server.

Kerberos Overview:



Step-1:

User login and request services on the host. Thus user requests for ticket-granting service.

Step-2:

Authentication Server verifies user's access right using database and then gives ticket-granting-ticket and session key. Results are encrypted using the Password of the user.

Step-3:

The decryption of the message is done using the password then send the ticket to Ticket Granting Server. The Ticket contains authenticators like user names and network addresses.

Step-4:

Ticket Granting Server decrypts the ticket sent by User and authenticator verifies the request then creates the ticket for requesting services from the Server.

Step-5:

The user sends the Ticket and Authenticator to the Server.

Step-6:

The server verifies the Ticket and authenticators then generate access to the service. After this User can access the services.

Kerberos Limitations

- Each network service must be modified individually for use with Kerberos
- It doesn't work well in a timeshare environment
- Secured Kerberos Server
- Requires an always-on Kerberos server
- Stores all passwords are encrypted with a single key
- Assumes workstations are secure
- May result in cascading loss of trust.
- Scalability

Applications

User Authentication: User Authentication is one of the main applications of Kerberos. Users only have to input their username and password once with Kerberos to gain access to the network. The Kerberos server subsequently receives the encrypted authentication data and issues a ticket granting ticket (TGT).

Single Sign-On (SSO): Kerberos offers a Single Sign-On (SSO) solution that enables users to log in once to access a variety of network resources. A user can access any network resource they have been authorized to use after being authenticated by the Kerberos server without having to provide their credentials again.

3.3 DIGITAL CERTIFICATE

- A Digital certificate, also known as a **public key certificate**, is a signed document used to bind ownership of a public key with the entity that owns it. Digital certificates are used for sharing public keys to be used for encryption and authentication. The public key is included in the certificate, while the private key is kept secure.
- The owner of the certificate (who has the corresponding private key) can then use it to sign documents, and the public key can be used to verify the validity of those signatures. Digital certificate is also sent with the digital signature and the message. Third parties can also use the public key to send encrypted information, which only the owner of the private key can encrypt.
- These certificates are issued by a trusted entity called as **Certificate Authority (CA)**. CAs are often selected government agencies or banks whose prime function is to verify the identity of the certificate holder.
- When an individual, website, or organization wish to obtain a digital certificate, they submit a **certificate signing request (CSR)** with the public key and information to be validated.
- CA then validates the information and sign it with an intermediate key that chains to a trusted root certificate. If validation is successful, the certificate is issued.
- Every Digital certificate contains:
 - (1) Name of certificate holder.
 - (2) Serial number which is used to uniquely identify a certificate, the individual or the entity identified by the certificate.
 - (3) Expiration dates.
 - (4) Copy of certificate holder's public key.
 - (5) Digital Signature of the certificate issuing authority.

Types of Digital Certificates

There are three types of digital certificates that a CA may issue and used by web servers and web browsers to authenticate over the internet.

(1) Client Certificates or Digital IDs

- This is the most basic type of the certificate which are used to identify one person to another, a person to a device or gateway or one device to another device.
- The client may apply through a regular e-mail stating his/her public key, name etc.
- Client Certificates are issued in their thousands and millions each year as CA requires no credentials from the applicant.

(2) Secure Socket Layer (SSL) server Certificates

- These certificates are installed on a server. This can be a server that hosts some website, a mail server or any other type of server that needs to be authenticated.
- These certificates can be Domain Validated, Organization Validated or Extended Validation certificates.

(3) Code Signing Certificates

- These certificates are used to sign software or programmed code that is downloaded over the Internet.

- It is the digital equivalent of the hologram seal used in the real world to authenticate software and assure the code is genuine and comes from the software publisher that it claims.

3.3.1 X.509 Certificate

RQ. Design sample Digital Certificate and explain each field of it.

Ref.

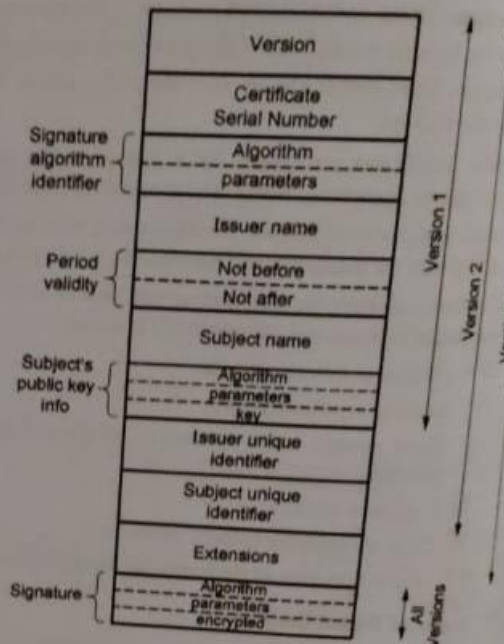
RQ. What is the significance of a digital signature on a certificate? Justify.

Ref.

- X.509 is a standard format for public key certificates, digital documents to verify that a public key belongs to computer or service identity contained within the certificate. X.509 has been adapted for internet use by the Public-Key Infrastructure (X.509) (PKIX) working group.

- Common fields in X.509 certificates are:

- Version number** : This field defines which X.509 version applies to the certificate. The version number started at 0 and currently it is version 2.
- Serial number** : This field defines serial number assigned to the certificate that distinguishes it from other certificates.
- Signature Algorithm Information** : This field identifies the algorithm used by the issuer to sign the certificate.
- Issuer name** : This field defines the name of the entity issuing the certificate (usually a certificate authority).
- Validity period of the certificate** : This field defines start/end date and time the certificate is valid.
- Subject name** : This field defines the name of the identity the certificate is issued to, the entity to which the public key belongs.
- Subject public key information** : This field defines the public key associated with the identity (heart of the certificate) as well as the corresponding algorithm.



(IB50)Fig. 3.3.1 : Format of X.509 Digital Certificate

- Issuer unique identifier** : This is an optional field which allows two issuers to have same issuer field value.
 - Subject unique identifier** : This is an optional field which allows two subjects to have same subject field value.
 - Extensions** : This is an optional field which allows issuers to add more private information to the certificate.
 - Signature** : This field is comprised of three sub-fields: algorithms, parameters and encrypted.
- Every certificate can be renewed after period of validity. The CA generally issues a new certificate if there is no problem, before the old certificate expires.

3.4 PUBLIC KEY INFRASTRUCTURE (PKI)

RQ. What is PKI? Explain different PKI architectures in detail.

Ref. May 19

- Public Key Infrastructure (PKI)** is the framework of encryption and cybersecurity that protects communications between the server (your website) and the client (the users).

- The distribution, authentication and revocation of digital certificates based on X.509 are the primary purposes of the PKI, the system by which public keys are distributed and authenticated.

Important duties of PKI

- Issuing, renewal and revocation of digital certificates.
- Storage and update of private keys.
- Providing services to protocols like IPSec and TLS.
- Providing different levels of access to the information stored in the database.

Components Of PKI

- There are three key components: digital certificates, certificate authority, and registration authority. By hosting these elements on a secure framework, PKI can protect the identities involved as well as the private information used in situations where digital security is necessary, such as smart card logins, SSL signatures, encrypted documents, and more.
- These elements are vital in securing and communicating digital information and electronic transactions.

(1) Digital Certificates

- A digital certificate is a form of electronic identification for websites and organizations.
- Secure connections between two communicating machines are made available through PKI because the identities of the two parties can be verified by with the help of certificates.

(2) Certificate Authority

- A Certificate Authority (CA) is used to authenticate the digital identities of the users, which can range from individuals to computer systems to servers.
- Certificate Authorities prevent fake entities and manage the life cycle of any given number of digital certificates within the system.

(3) Registration Authority

- Registration Authority (RA) is authorized by the Certificate Authority to provide digital certificates to users on a case-by-case basis.
- All of the certificates that are requested, received, and revoked by both the Certificate Authority and the Registration Authority are stored in an encrypted certificate database.

PKI Architecture

[Applications]		
System Security Enabling Services	Secure Protocols	Security Policy Services
	Protocol Security Services	
	Long-Term Key Services	Supporting Services
	Cryptographic Services	
	Cryptographic Primitives	

(1851) Fig. 3.4.1 : Overview of PKI Architecture

The PKI Architecture components are grouped into the following broad functional categories :

- (1) **System Security-enabling Services :** These services provide the functionality which allows a user's or other principal's identity to be established and associated with their actions in the system.
- (2) **Cryptographic Primitives and Services :** These services provide the cryptographic functions on which public-key security is based (including secret-key primitives, such as DES).
- (3) **Long-term Key Services :** These service permit users and other principals to manage their own long-term keys and certificates and to retrieve and check the validity of other principals' certificates.
- (4) **Protocol Security Services :** These services provide security functionality (data origin authentication, data integrity, data privacy, non-repudiation) suitable for use by implementors of security-aware applications, such as secure protocols.
- (5) **Secure Protocols :** These provide secure inter-application communications for security-unaware and "mildly" security-aware applications.
- (6) **Security Policy Services :** These services provide the policy-related information which must be carried in secure protocols to enable access control and provide access control checking facilities to security-aware applications which must enforce policy.
- (7) **Supporting Services :** These services provide functionality, which is required for secure operation, but is not directly involved in security policy enforcement.

- The root CA
- Multiple CAs
- The issuer of the middle root CA
- This arrangement implementation allow tight control
- The benefit of the CAs and their use the hierarchy.
- This CA, in turn, This is like cutting
- However, the drawback of the root CA
- **Mesh CA Trust Model**
- A mesh model is a CAs are not related the CA