LECTURE NOTES

ON

Cryptography and network security

Diploma 6th semester

Branch-CSE

CHAPTER -1

Origin of Cryptography

Human being from ages had two inherent needs − (a) to communicate and share informa on and (b) to communicate selectively. These two needs gave rise to the art of coding the messages in such a way that only the intended people could have access to the information. Unauthorized people could not extract any information, even if the scrambled messages fell in their hand.

The art and science of concealing the messages to introduce secrecy in information security is recognized as cryptography.

The word ‗cryptography' was coined by combining two Greek words, ‗Krypto' meaning hidden and ‗graphene' meaning writing.

History of Cryptography

The art of cryptography is considered to be born along with the art of writing. As civilizations evolved, human beings got organized in tribes, groups, and kingdoms. This led to the emergence of ideas such as power, battles, supremacy, and politics. These ideas further fueled the natural need of people to communicate secretly with selective recipient which in turn ensured the continuous evolution of cryptography as well.

The roots of cryptography are found in Roman and Egyptian civilizations.

Hieroglyph − The Oldest Cryptographic Technique

The first known evidence of cryptography can be traced to the use of ‗hieroglyph'. Some 4000 years ago, the Egyptians used to communicate by messages written in hieroglyph. This code was the secret known only to the scribes who used to transmit messages on behalf of the kings. One such hieroglyph is shown below.

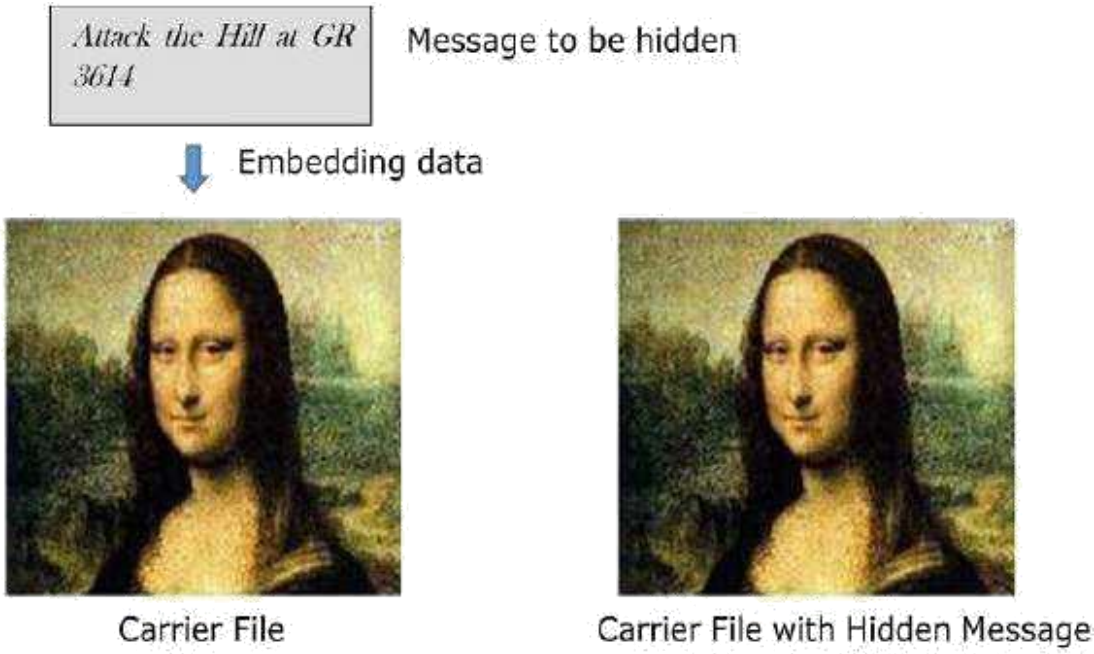
Later, the scholars moved on to using simple mono-alphabetic substitution ciphers during 500 to 600 BC. This involved replacing alphabets of message with other alphabets with some secret rule. This rule became a key to retrieve the message back from the garbled message.

The earlier Roman method of cryptography, popularly known as the Caesar Shift Cipher, relies on shifting the letters of a message by an agreed number (three was a common choice), the recipient of this message would then shift the letters back by the same number and obtain the original message.

Steganography

Steganography is similar but adds another dimension to Cryptography. In this method, people not only want to protect the secrecy of an information by concealing it, but they also want to make sure any unauthorized person gets no evidence that the information even exists. For example, invisible watermarking.

In steganography, an unintended recipient or an intruder is unaware of the fact that observed data contains hidden information. In cryptography, an intruder is normally aware that data is being communicated, because they can see the coded/scrambled message.



Evolution of Cryptography

It is during and after the European Renaissance, various Italian and Papal states led the rapid proliferation of cryptographic techniques. Various analysis and attack techniques were researched in this era to break the secret codes.

Improved coding techniques such as Vigenere Coding came into existence in the 15th century, which offered moving letters in the message with a number of variable places instead of moving them the same number of places.

Only after the 19th century, cryptography evolved from the ad hoc approaches to encryption to the more sophisticated art and science of information security.

In the early 20th century, the invention of mechanical and electromechanical machines, such as the Enigma rotor machine, provided more advanced and efficient means of coding the information.

During the period of World War II, both cryptography and cryptanalysis became excessively mathematical.

With the advances taking place in this field, government organizations, military units, and some corporate houses started adopting the applications of cryptography. They used cryptography to guard their secrets from others. Now, the arrival of computers and the Internet has brought effective cryptography within the reach of common people.

Modern Cryptography

Modern cryptography is the cornerstone of computer and communications security. Its foundation is based on various concepts of mathematics such as number theory, computational-complexity theory, and probability theory.

Characteristics of Modern Cryptography

There are three major characteristics that separate modern cryptography from the classical approach.

Classic Cryptography	Modern Cryptography
It manipulates traditional characters, i.e., letters and digits directly.	**It operates on binary bit sequences.**
It is mainly based on _security through obscurity'. The techniques employed for coding were kept secret and only the parties involved in communication knew about them.	**It relies on publicly known mathematical algorithms for coding the information. Secrecy is obtained through a secrete key which is used as the seed for the algorithms. The computational difficulty of algorithms, absence of secret key, etc., make it impossible for an attacker to obtain the original information even if he knows the algorithm used for coding.**
It requires the entire cryptosystem for communicating confidentially.	**Modern cryptography requires parties interested in secure communication to possess the secret key only.**

Need of cryptography

Cryptography is an essential way of preventing that from happening. It secures information and communications using a set of rules that allows only those intended—and no one else— to receive the information to access and process it.

Security Approach

Computer data often travels from one computer to another, leaving the safety of its protected physical surroundings. Once the data is out of hand, people with bad intention could modify or forge your data, either for amusement or for their own benefit.
Cryptography can reformat and transform our data, making it safer on its trip between computers. The technology is based on the essentials of secret codes, augmented by modern mathematics that protects our data in powerful ways.

- **Computer Security - generic name for the collection of tools designed to protect data and to thwart hackers**
- **Network Security - measures to protect data during their transmission**
- **Internet Security - measures to protect data during their transmission over a collection of interconnected networks**

Security Attacks, Services and Mechanisms
To assess the security needs of an organization effectively, the manager responsible for security needs some systematic way of defining the requirements for security and characterization
of approaches to satisfy those requirements. One approach is to consider three aspects of information security:

Security attack – Any action that compromises the security of information owned by an organization.
Security mechanism – A mechanism that is designed to detect, prevent or recover from a security attack.
Security service – A service that enhances the security of the data processing systems and the information transfers of an organization. The services are intended to counter security attacks and
they make use of one or more security mechanisms to provide the service.

Basic Concepts

Cryptography The art or science encompassing the principles and methods of transforming an intelligible message into one that is unintelligible, and then retransforming that message back to its original form

Plaintext The original intelligible message

Cipher text The transformed message

Cipher An algorithm for transforming an intelligible message into one that is unintelligible by transposition and/or substitution methods

Key Some critical information used by the cipher, known only to the sender& receiver

Encipher (encode) The process of converting plaintext to cipher text using a cipher and a key

Decipher (decode) the process of converting cipher text back into plaintext using a cipher and a key

Cryptanalysis The study of principles and methods of transforming an unintelligible message back into an intelligible message without knowledge of the key. Also called code breaking

Cryptology Both cryptography and cryptanalysis

Code An algorithm for transforming an intelligible message into an unintelligible one using a code-book

Cryptography

Cryptographic systems are generally classified along 3 independent dimensions:

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.

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. 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.
A stream cipher processes the input elements continuously, producing output element one at a time, as it goes along.

Cryptography and Network Security Principles

In present day scenario security of the system is the sole priority of any organisation. The main aim of any organisation is to protect their data from attackers. In cryptography, attacks are of two types such as Passive attacks and Active attacks.
Passive attacks are those that retrieve information from the system without affecting the system resources while active attacks are those that retrieve system information and make changes to the system resources and their operations.

The Principles of Security can be classified as follows:

1. **Confidentiality:**
   The degree of confidentiality determines the secrecy of the information. The principle specifies that only the sender and receiver will be able to access the information shared between them. Confidentiality compromises if an unauthorized person is able to access a message.
   For example, let us consider sender A wants to share some confidential information with receiver B and the information gets intercepted by the attacker C. Now the confidential information is in the hands of an intruder C.

2. **Authentication:**
   Authentication is the mechanism to identify the user or system or the entity. It ensures the identity of the person trying to access the information. The authentication is mostly secured

by using username and password. The authorized person whose identity is preregistered can prove his/her identity and can access the sensitive information.

3. **Integrity:**
   Integrity gives the assurance that the information received is exact and accurate. If the content of the message is changed after the sender sends it but before reaching the intended receiver, then it is said that the integrity of the message is lost.

4. **Non-Repudiation:**
   Non-repudiation is a mechanism that prevents the denial of the message content sent through a network. In some cases the sender sends the message and later denies it. But the non-repudiation does not allow the sender to refuse the receiver.

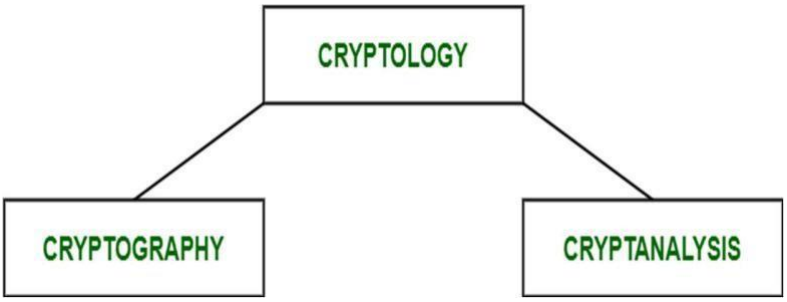
5. **Access control:**
   The principle of access control is determined by role management and rule management. Role management determines who should access the data while rule management determines up to what extent one can access the data. The information displayed is dependent on the person who is accessing it.

6. **Availability:**
   The principle of availability states that the resources will be available to authorize party at all times. Information will not be useful if it is not available to be accessed. Systems should have sufficient availability of information to satisfy the user request.

Cryptanalysis and Types of 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 practicing 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

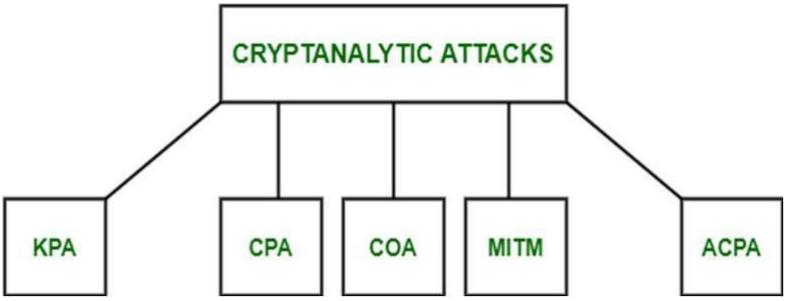
might try to decipher a ciphertext to derive the plaintext. It can help us to deduce the plaintext or the encryption key.



Parts Of Cryptology

To determine the weak points of a cryptographic system, it is important to attack the system. This attacks are called Cryptanalytic attacks. The attacks rely on nature of the algorithm and also knowledge of the general characteristics of the plaintext, i.e., plaintext can be a regular document written in English or it can be a code written in Java. Therefore, nature of the plaintext should be known before trying to use the attacks.

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. Its 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 CPA. Here, the attacker requests the cipher texts of additional plaintexts after they have ciphertexts for some texts.

CHAPTER-2

Cryptography concepts

Definition: Cryptography is associated with the process of converting ordinary plain text into unintelligible text and vice-versa. It is a method of storing and transmitting data in a particular form so that only those for whom it is intended can read and process it. Cryptography not only protects data from theft or alteration, but can also be used for user authentication.

The various components of a basic cryptosystem are as follows −

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 cipher text 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.

Simple Substitution Cipher

It is an improvement to the Caesar Cipher. Instead of shifting the alphabets by some number, this scheme uses some permutation of the letters in alphabet.

For example, A.B…..Y.Z and Z.Y……B.A are two obvious permutation of all the letters in alphabet.

Permutation is nothing but a jumbled up set of alphabets.

With 26 letters in alphabet, the possible permutations are 26! (Factorial of 26) which is equal to $4x10_{26}$. The sender and the receiver may choose any one of these possible permutation as a ciphertext alphabet. This permutation is the secret key of the scheme.

Process of Simple Substitution Cipher

Write the alphabets A, B, C,...,Z in the natural order.

The sender and the receiver decide on a randomly selected permutation of the letters of the alphabet.

Underneath the natural order alphabets, write out the chosen permutation of the letters of the alphabet. For encryption, sender replaces each plaintext letters by substituting the permutation letter that is directly beneath it in the table. This process is shown in the following illustration. In this example, the chosen permutation is K,D, G, ..., O. The plaintext _point' is encrypted to _MJBXZ'.

Here is a jumbled Ciphertext alphabet, where the order of the ciphertext letters is a key.

Plaintext Alphabet	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
Ciphertext Alphabet	K	D	G	F	N	S	L	V	B	W	A	H	E	X	J	M	Q	C	P	Z	R	T	Y	I	U	O

On receiving the ciphertext, the receiver, who also knows the randomly chosen permutation, replaces each ciphertext letter on the bottom row with the corresponding plaintext letter in the top row. The ciphertext _MJBXZ' is decrypted to _point'.

Security Value

Simple Substitution Cipher is a considerable improvement over the Caesar Cipher. The possible number of keys is large (26!) and even the modern computing systems are not yet powerful enough to comfortably launch a brute force attack to break the system. However, the Simple Substitution Cipher has a simple design and it is prone to design flaws, say choosing obvious permutation, this cryptosystem can be easily broken.

Transposition Cipher

Transposition Cipher is a cryptographic algorithm where the order of alphabets in the plaintext is rearranged to form a cipher text. In this process, the actual plain text alphabets are not included.

Example

A simple example for a transposition cipher is columnar transposition cipher where each character in the plain text is written horizontally with specified alphabet width. The cipher is written vertically, which creates an entirely different cipher text.

Consider the plain text hello world, and let us apply the simple columnar transposition technique as shown below

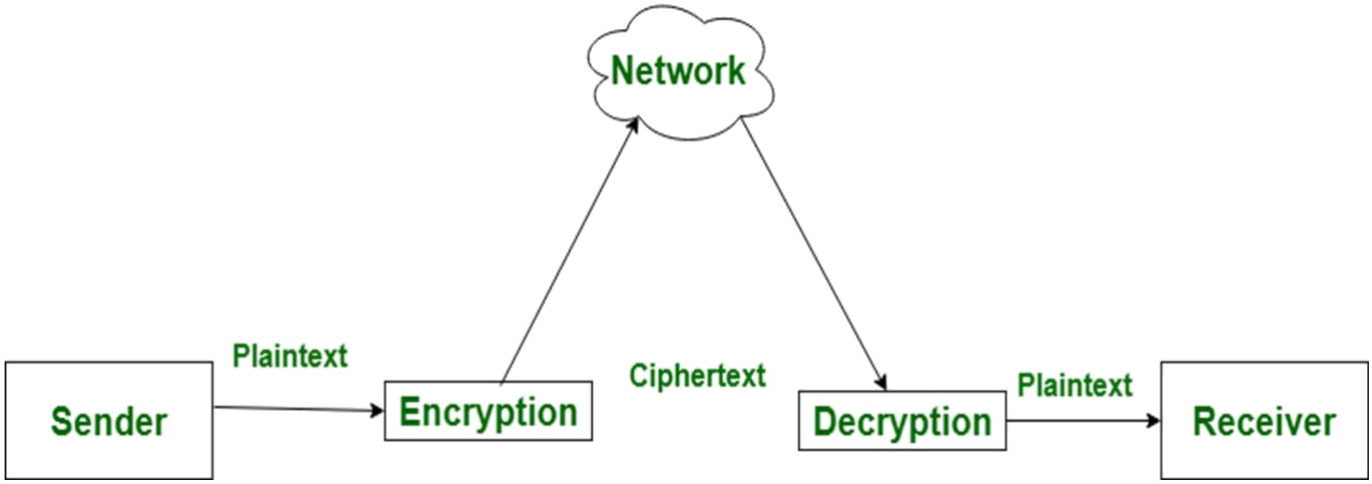
h	e	l	l
o	w	o	r
l	d		

The plain text characters are placed horizontally and the cipher text is created with vertical format as : holewdlo lr. Now, the receiver has to use the same table to decrypt the cipher text to plain text.

Difference between Encryption and Decryption

Encryption is the process of converting normal message (plaintext) into meaningless message (Ciphertext). Whereas Decryption is the process of converting meaningless message (Ciphertext) into its original form (Plaintext).

The major distinction between secret writing associated secret writing is that secret writing is that the conversion of a message into an unintelligible kind that‘s undecipherable unless decrypted. whereas secret writing is that the recovery of the first message from the encrypted information.



Let‘s see that the difference between encryption and decryption:

S.NO	Encryption	Decryption
1.	**Encryption is the process of converting normal message into meaningless message.**	**While decryption is the process of converting meaningless message into its original form.**
2.	**Encryption is the process which take place at sender‘s end.**	**While decryption is the process which take place at receiver‘s end.**
3.	**Its major task is to convert the plain text into cipher text.**	**While its main task is to convert the cipher text into plain text.**
4.	**Any message can be encrypted with either secret key or public key.**	**Whereas the encrypted message can be decrypted with either secret key or private key.**
5.	**In encryption process, sender sends the data to receiver after encrypted it.**	**Whereas in decryption process, receiver receives the information(Cipher text) and convert into plain text.**

CHAPTER-3

Symmetric and Asymmetric Cryptography

Symmetric Cryptography

In this type, the encryption and decryption process uses the same key. It is also called as secret key cryptography. The main features of symmetric cryptography are as follows −

- **It is simpler and faster.**
- **The two parties exchange the key in a secure way.**

Drawback

The major drawback of symmetric cryptography is that if the key is leaked to the intruder, the message can be easily changed and this is considered as a risk factor.

Asymmetric Cryptography

It is also called as public key cryptography. It works in the reverse way of symmetric cryptography. This implies that it requires two keys: one for encryption and other for decryption. The public key is used for encrypting and the private key is used for decrypting.

Drawback

- **Due to its key length, it contributes lower encryption speed. Key management is crucial.**

Differentiator	Symmetric Key Encryption	Asymmetric Key Encryption
1. Symmetric Key vs Asymmetric key	Only one key (symmetric key) is used, and the same key is used to encrypt and decrypt the message.	Two different cryptographic keys (asymmetric keys), called the public and the private keys, are used for encryption and decryption.
2. Complexity and Speed of Execution	It‘s a simple technique, and because of this, the encryption process can be carried out quickly.	It‘s a much more complicated process than symmetric key encryption, and the process is slower.

3. Length of Keys	The length of the keys used is typically 128 or 256 bits, based on the security requirement.	The length of the keys is much larger, e.g., the recommended RSA key size is 2048 bits or higher.
4. Usage	It‘s mostly used when large chunks of data need to be transferred.	It‘s used in smaller transactions, primarily to authenticate and establish a secure communication channel prior to the actual data transfer.
5. Security	The secret key is shared. Consequently, the risk of compromise is higher.	The private key is not shared, and the overall process is more secure as compared to symmetric encryption.
Examples of Algorithms	Examples include RC4, AES, DES, 3DES, etc.	Examples include RSA, Diffie-Hellman, ECC, etc.

Symmetric and asymmetric key algorithm

In today‘s cyber-world there is an ever-present risk of unauthorized access to all forms of data. Most at risk is financial and payment system data that can expose the personal identifying information (PII) or payment card details of customers and clients. Encryption is crucial for protecting PII and mitigating the risks that businesses which conduct payment transactions face every minute of every day.

In this article we will talk about symmetric encryption in banking, its advantages and some challenges of managing the keys.

What is Symmetric Encryption?

Symmetric encryption is a type of encryption where only one key (a secret key) is used to both encrypt and decrypt electronic information. The entities communicating via symmetric encryption must exchange the key so that it can be used in the decryption process. This encryption method differs from asymmetric encryption where a pair of keys, one public and one private, is used to encrypt and decrypt messages.

By using symmetric encryption algorithms, data is converted to a form that cannot be understood by anyone who does not possess the secret key to decrypt it. Once the intended recipient who possesses the key has the message, the algorithm reverses its action so that the message is returned to its original and understandable form. The secret key that the sender and recipient both use could be a specific password/code and it can be random string of letters or numbers that have been generated by a secure random number generator (RNG). For banking-grade encryption, the symmetric keys must be created using an RNG that is certified according to industry standards, such as FIPS 140-2.

There are two types of symmetric encryption algorithms:

1. **Block algorithms. Set lengths of bits are encrypted in blocks of electronic data with the use of a specific secret key. As the data is being encrypted, the system holds the data in its memory as it waits for complete blocks.**
2. **Stream algorithms. Data is encrypted as it streams instead of being retained in the system‘s memory.**

Some examples of symmetric encryption algorithms include:

AES (Advanced Encryption Standard)

DES (Data Encryption Standard)

IDEA (International Data Encryption Algorithm) Blowfish (Drop-in replacement for DES or IDEA) RC4 (Rivest Cipher 4)

RC5 (Rivest Cipher 5)

RC6 (Rivest Cipher 6)

AES, DES, IDEA, Blowfish, RC5 and RC6 are block ciphers. RC4 is stream cipher.

DES

In ―modern‖ computing, DES was the first standardized cipher for securing electronic communications, and is used in variations (e.g. 2-key or 3-key 3DES). The original DES is not used anymore as it is considered too ―weak‖, due to the processing power of modern computers. Even 3DES is not recommended by NIST and PCI DSS 3.2, just like all 64-bit ciphers. However, 3DES is still widely used in EMV chip cards.

AES

The most commonly used symmetric algorithm is the Advanced Encryption Standard (AES), which was originally known as Rijndael. This is the standard set by the U.S. National Institute of Standards and Technology in 2001 for the encryption of electronic data announced in U.S. FIPS PUB 197. This standard supersedes DES, which had been in use since 1977. Under NIST, the AES cipher has a block size of 128 bits, but can have three different key lengths as shown with AES-128, AES-192 and AES-256.

What is Symmetric Encryption Used For?

While symmetric encryption is an older method of encryption, it is faster and more efficient than asymmetric encryption, which takes a toll on networks due to performance issues with data size and heavy CPU use. Due to the better performance and faster speed of symmetric encryption (compared to asymmetric), symmetric cryptography is typically used for bulk encryption / encrypting large amounts of data, e.g. for database encryption. In the case of a database, the secret key might only be available to the database itself to encrypt or decrypt.

Some examples of where symmetric cryptography is used are:

Payment applications, such as card transactions where PII needs to be protected to prevent identity theft or fraudulent charges

Validations to confirm that the sender of a message is who he claims to

be Random number generation or hashing

Key management for symmetric encryption - what we need to consider

Unfortunately, symmetric encryption does come with its own drawbacks. Its weakest point is its aspects of key management, including:

Key Exhaustion

Symmetric Encryption suffers from behavior where every use of a key _leaks‘ some information that can potentially be used by an attacker to reconstruct the key. The defenses against this behavior include using a key hierarchy to ensure that master or key-encryption keys are not over-used and the appropriate rotation of keys that do encrypt volumes of data. To be tractable, both these solutions require competent key-

management strategies as if (for example) a retired encryption key cannot be recovered the data is potentially lost.

Attribution data

Unlike asymmetric (public-key) Certificates, symmetric keys do not have embedded metadata to record information such as expiry date or an Access Control List to indicate the use the key may be put to - to Encrypt but not Decrypt for example.
The latter issue is somewhat addressed by standards such as ANSI X9 -31 where a key can be bound to information prescribing its usage. But for full control over what a key can be used for and when it can be used, a key-management system is required.

Key Management at large scale

Where only a few keys are involved in a scheme (tens to low hundreds), the management overhead is modest and can be handled through manual, human activity. However, with a large estate, tracking the expiration and arranging rotation of keys quickly becomes impractical.

Consider an EMV payment card deployment: millions of cards multiplied by several keys-per-card requires a dedicated provision and key-management system.

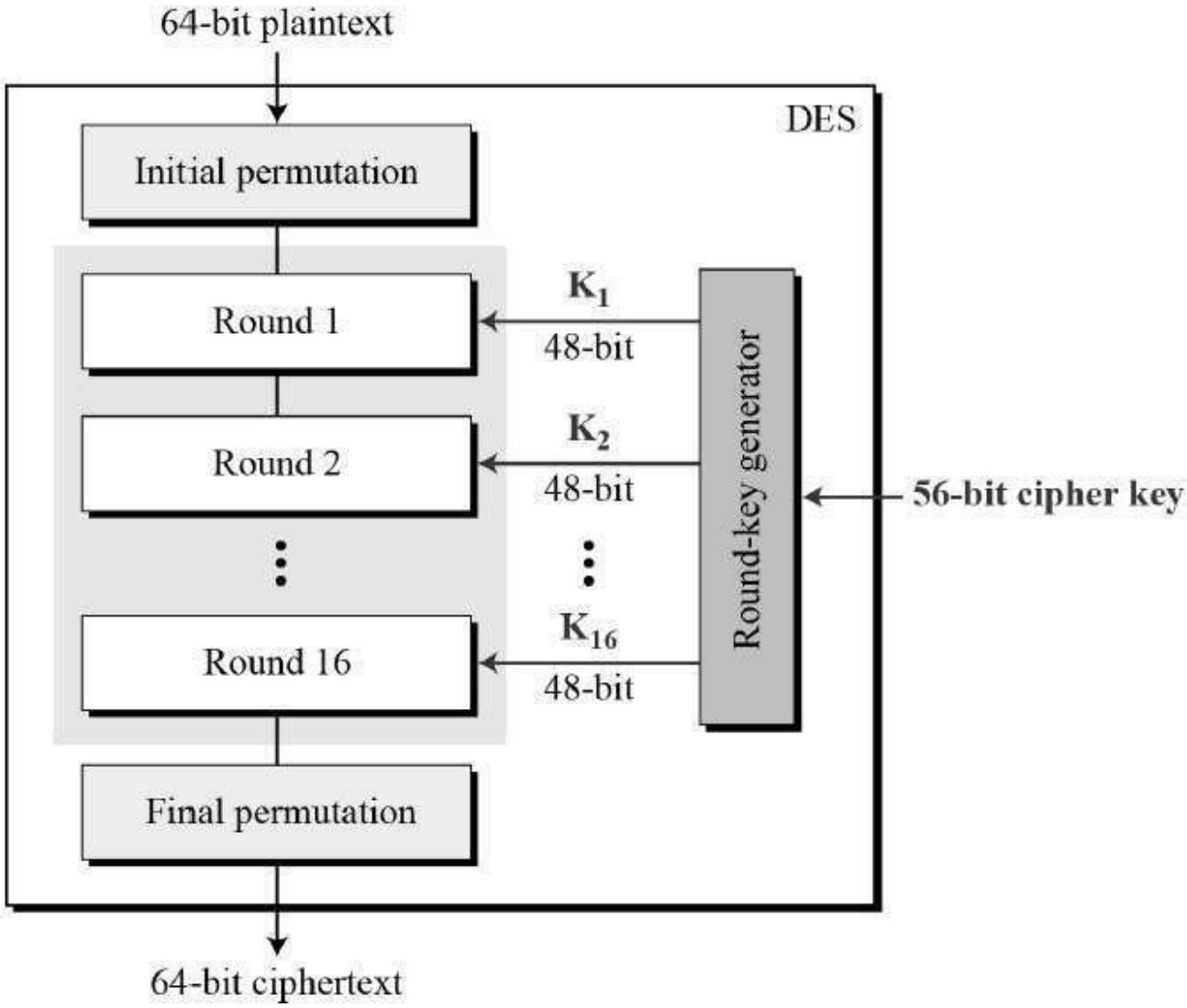
Symmetric Key Cryptography

Symmetric Key Cryptography also known as Symmetric Encryption is when a secret key is leveraged for both encryption and decryption functions. This method is the opposite of Asymmetric Encryption where one key is used to encrypt and another is used to decrypt. During this process, data is converted to a format that cannot be read or inspected by anyone who does not have the secret key that was used to encrypt it.
The success of this approach depends on the strength of the random number generator that is used to create the secret key. Symmetric Key Cryptography is widely used in today's Internet and primarily consists of two types of algorithms, Block and Stream. Some common encryption algorithms include the Advanced Encryption Standard (AES) and the Data Encryption Standard (DES). This form of encryption is traditionally much faster than Asymmetric however it requires both the sender and the recipient of the data to have the secret key.

Data Encryption Standard

The Data Encryption Standard (DES) is a symmetric-key block cipher published by the National Institute of Standards and Technology (NIST).

DES is an implementation of a Feistel Cipher. It uses 16 round Feistel structure. The block size is 64-bit. Though, key length is 64-bit, DES has an effective key length of 56 bits, since 8 of the 64 bits of the key are not used by the encryption algorithm (function as check bits only). General Structure of DES is depicted in the following illustration −



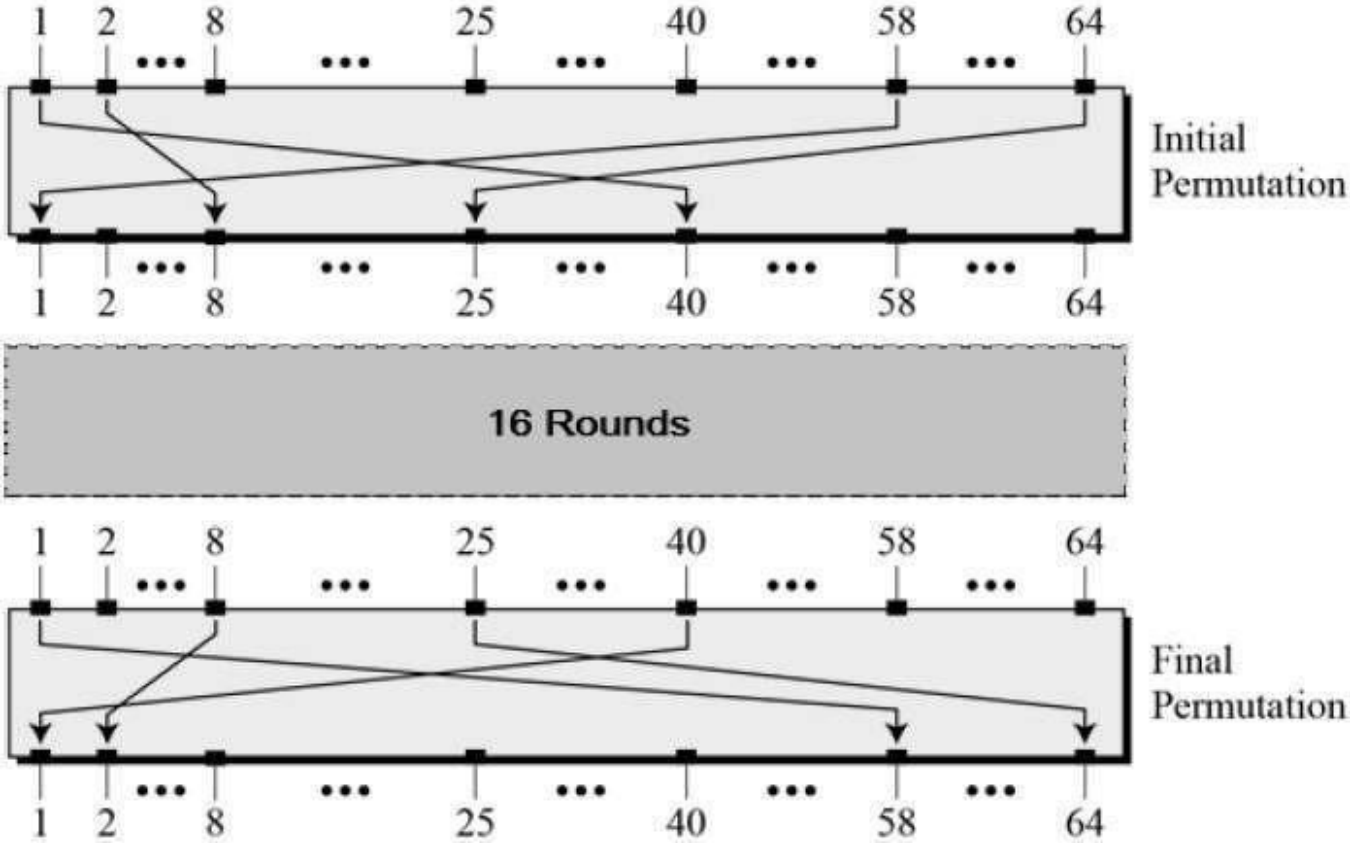
Since DES is based on the Feistel Cipher, all that is required to specify DES is −

- **Round function**
- **Key schedule**

Any additional processing − Ini al and final permuta on

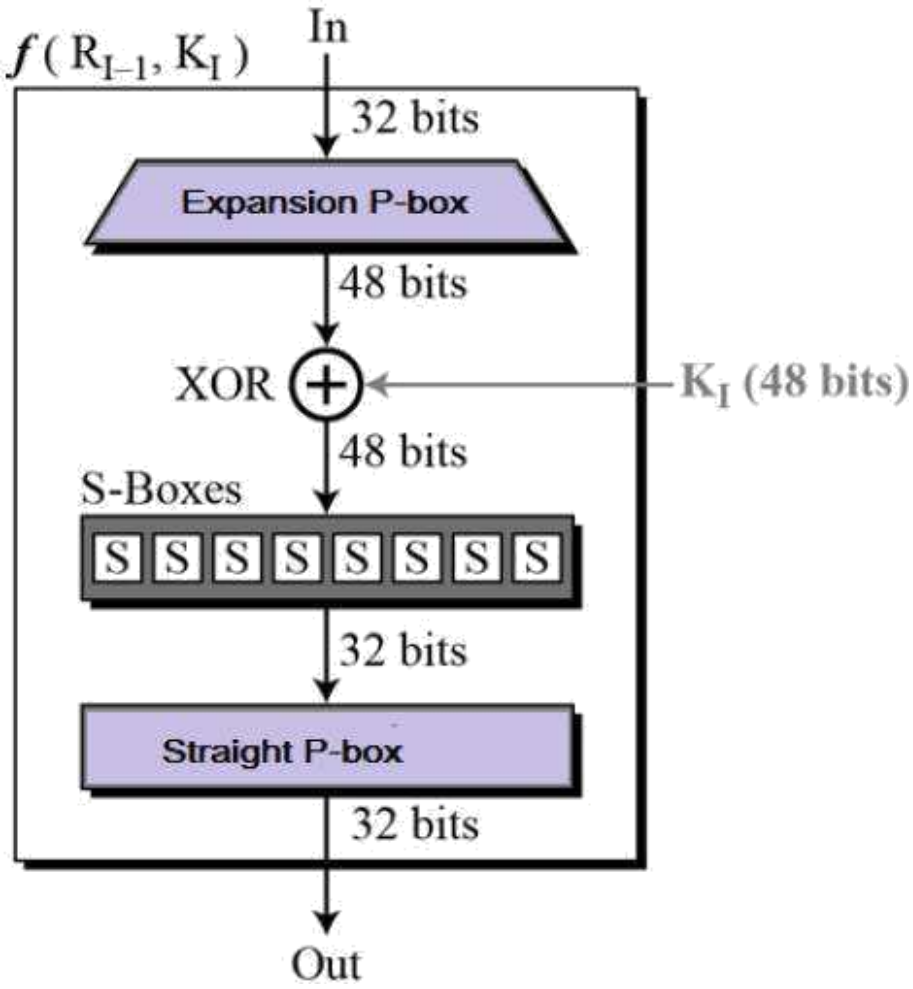
Initial and Final Permutation

The initial and final permutations are straight Permutation boxes (P-boxes) that are inverses of each other. They have no cryptography significance in DES. The initial and final permutations are shown as follows −

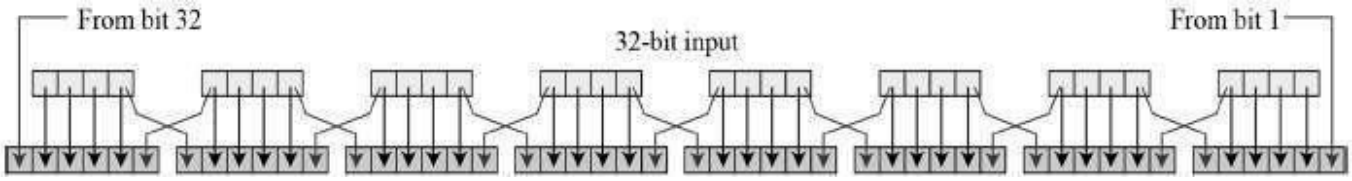


Round Function

The heart of this cipher is the DES function, f. The DES function applies a 48-bit key to the rightmost 32 bits to produce a 32-bit output.



Expansion Permutation Box − Since right input is 32-bit and round key is a 48-bit, we first need to expand right input to 48 bits. Permutation logic is graphically depicted in the following illustration −

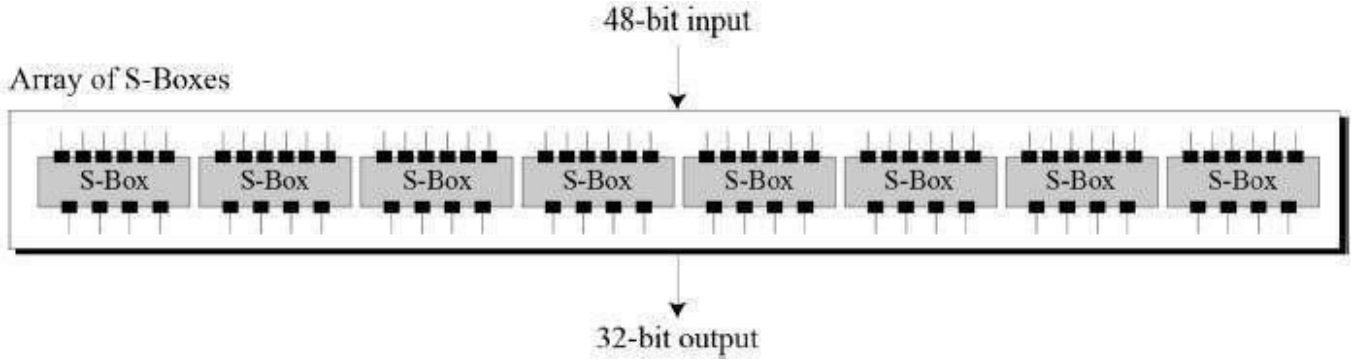


The graphically depicted permutation logic is generally described as table in DES specification illustrated as shown −

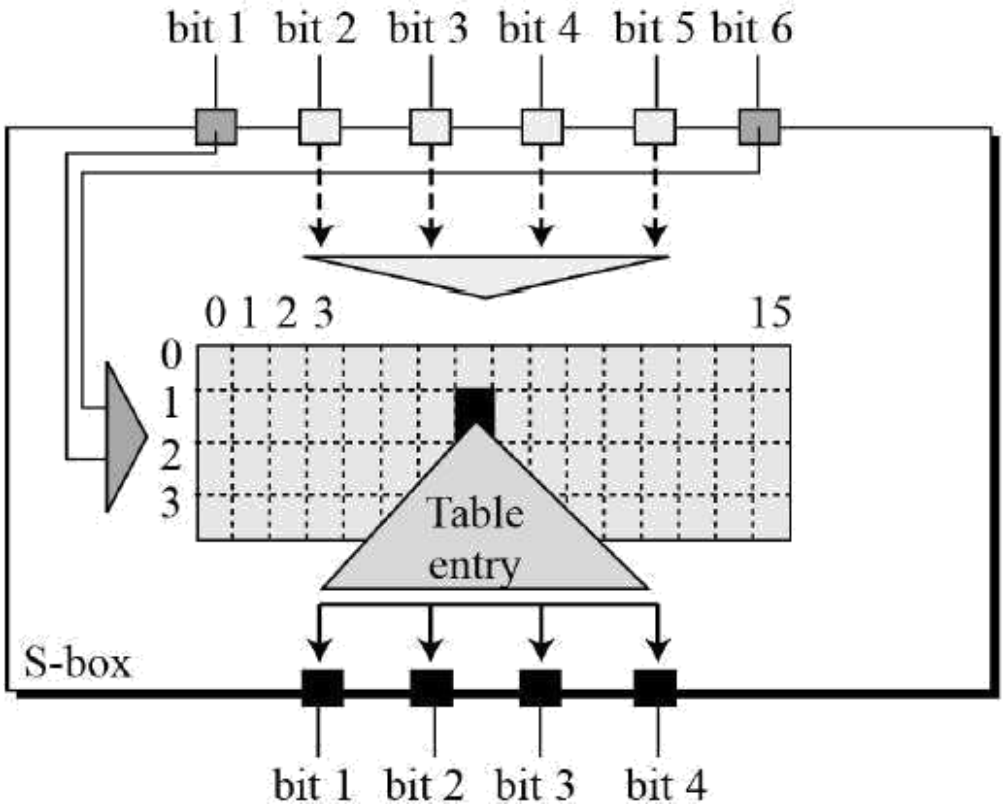
32	01	02	03	04	05
04	05	06	07	08	09
08	09	10	11	12	13
12	13	14	15	16	17
16	17	18	19	20	21
20	21	22	23	24	25
24	25	26	27	28	29
28	29	31	31	32	01

XOR (Whitener). − A er the expansion permuta on, DES does XOR opera on on the expanded right section and the round key. The round key is used only in this operation.

Substitution Boxes. − The S-boxes carry out the real mixing (confusion). DES uses 8 S-boxes, each with a 6-bit input and a 4-bit output. Refer the following illustration −



The S-box rule is illustrated below −



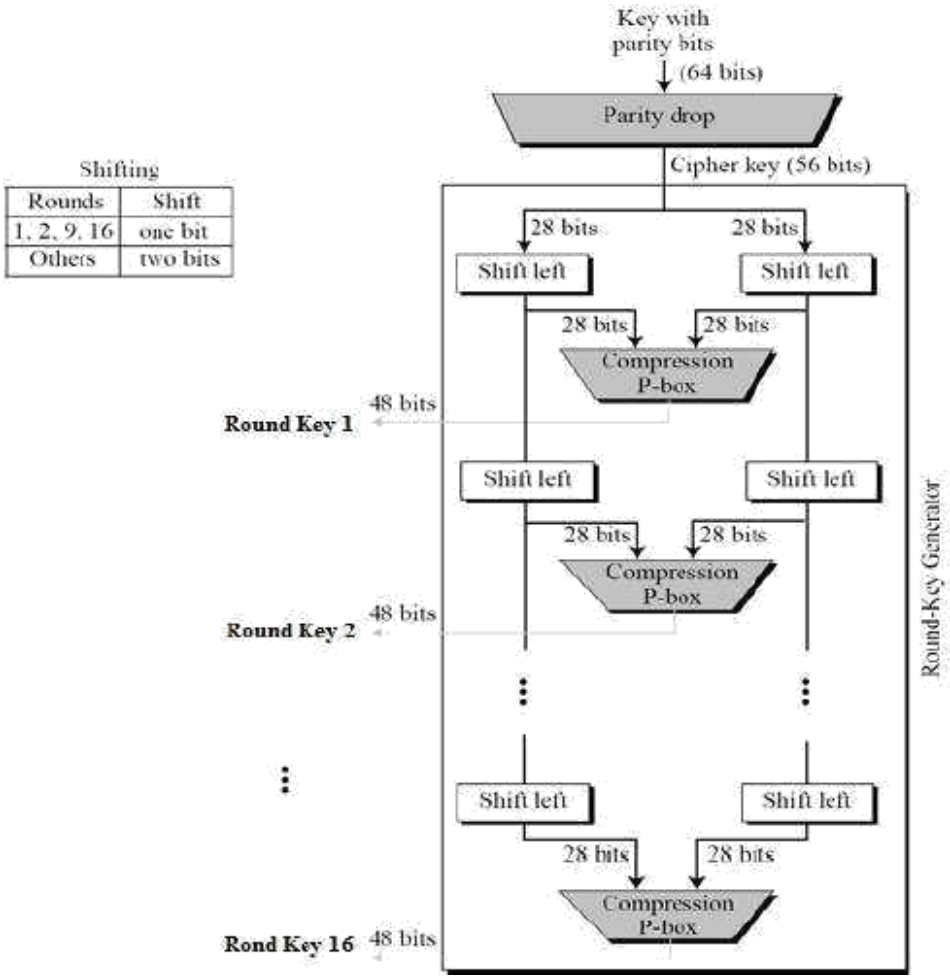
There are a total of eight S-box tables. The output of all eight s-boxes is then combined in to 32 bit section.

Straight Permutation − The 32 bit output of S-boxes is then subjected to the straight permutation with rule shown in the following illustration:

16	07	20	21	29	12	28	17
01	15	23	26	05	18	31	10
02	08	24	14	32	27	03	09
19	13	30	06	22	11	04	25

Key Generation

The round-key generator creates sixteen 48-bit keys out of a 56-bit cipher key. The process of key generation is depicted in the following illustration −



The logic for Parity drop, shifting, and Compression P-box is given in the DES description.

DES Analysis

The DES satisfies both the desired properties of block cipher. These two properties make cipher very strong.

Avalanche effect − A small change in plaintext results in the very great change in the cipher text.

Completeness − Each bit of cipher text depends on many bits of plaintext.

During the last few years, cryptanalysis have found some weaknesses in DES when key selected are weak keys. These keys shall be avoided.

DES has proved to be a very well designed block cipher. There have been no significant cryptanalytic attacks on DES other than exhaustive key search.

Asymmetric cryptography (public key cryptography)

Asymmetric cryptography, also known as public-key cryptography, is a process that uses a pair of related keys -- one public key and one private key -- to encrypt and decrypt a message and protect it from unauthorized access or use. A public key is a cryptographic key that can be used by any person to encrypt a message so that it can only be deciphered by the intended recipient with their private key. A private key -- also known as a secret key -- is shared only with key's initiator.

When someone wants to send an encrypted message, they can pull the intended recipient's public key from a public directory and use it to encrypt the message before sending it. The recipient of the message can then decrypt the message using their related private key. On the other hand, if the sender encrypts the message using their private key, then the message can be decrypted only using that sender's public key, thus authenticating the sender. These encryption and decryption processes happen automatically; users do not need to physically lock and unlock the message.

How asymmetric cryptography works

Asymmetric encryption uses a mathematically related pair of keys for encryption and decryption: a public key and a private key. If the public key is used for encryption, then the related private key is used for decryption; if the private key is used for encryption, then the related public key is used for decryption.

The two participants in the asymmetric encryption workflow are the sender and the receiver; each has its own pair of public and private keys. First, the sender obtains the receiver's public key. Next, the plaintext -- or ordinary, readable text -- is encrypted by the sender using the receiver's public key; this creates ciphertext. The ciphertext is then sent to the receiver, who decrypts the ciphertext with their private key and returns it to legible plaintext.

Because of the one-way nature of the encryption function, one sender is unable to read the messages of another sender, even though each has the public key of the receiver.

Uses of asymmetric cryptography

Asymmetric cryptography is typically used to authenticate data using digital signatures. A digital signature is a mathematical technique used to validate the authenticity and integrity of a message, software or digital document. It is the digital equivalent of a handwritten signature or stamped seal.

Based on asymmetric cryptography, digital signatures can provide assurances of evidence to the origin, identity and status of an electronic document, transaction or message, as well as acknowledge informed consent by the signer.

Asymmetric cryptography can also be applied to systems in which many users may need to encrypt and decrypt messages, including:

- **Encrypted email - a public key can be used to encrypt a message and a private key can be used to decrypt it.**
- **The SSL/TSL cryptographic protocols - establishing encrypted links between websites and browsers also makes use of asymmetric encryption.**
- **Bitcoin and other cryptocurrencies rely on asymmetric cryptography as users have public keys that everyone can see and private keys that are kept secret. Bitcoin uses a cryptographic algorithm to ensure that only the legitimate owners can spend the funds.**

In the case of the Bitcoin ledger, each unspent transaction output (UTXO) is typically associated with a public key. So if user X, who has an UTXO associated with his public key, wants to send the money to user Y, user X uses his private key to sign a transaction that spends the UTXO and creates a new UTXO that's associated with user Y's public key.

Benefits and disadvantages of asymmetric cryptography

The benefits of asymmetric cryptography include:

- **The key distribution problem is eliminated because there's no need for exchanging keys.**
- **Security is increased as the private keys don't ever have to be transmitted or revealed to anyone.**
- **The use of digital signatures is enabled so that a recipient can verify that a message comes from a particular sender.**

It allows for non-repudiation so the sender can't deny sending a message.

Disadvantages include:

- **it's a slow process compared to symmetric cryptography, so it's not appropriate for decrypting bulk messages.**

if an individual loses his private key, he can't decrypt the messages he receives.

- **since the public keys aren't authenticated, no one really knows if a public key belongs to the person specified. Consequently, users must verify that their public keys belong to them.**
- **if a hacker identifies a person's private key, the attacker can read all of that individual's messages.**

RSA Algorithm

RSA algorithm is a public key encryption technique and is considered as the most secure way of encryption. It was invented by Rivest, Shamir and Adleman in year 1978 and hence name RSA algorithm.

Algorithm

The RSA algorithm holds the following features −

RSA algorithm is a popular exponentiation in a finite field over integers including prime numbers.

The integers used by this method are sufficiently large making it difficult to solve. There are two sets of keys in this algorithm: private key and public key.

You will have to go through the following steps to work on RSA algorithm −

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

Creating RSA Keys

Generating RSA keys

The following steps are involved in generating RSA keys −

Create two large prime numbers namely p and q. The product of these numbers will be called n, where n= p*q

Generate a random number which is relatively prime with (p-1) and (q-1). Let the number be called as e.

Calculate the modular inverse of e. The calculated inverse will be called as d.

RSA Cipher Encryption

In this chapter, we will focus on different implementation of RSA cipher encryption and the functions involved for the same. You can refer or include this python file for implementing RSA cipher algorithm implementation.

The modules included for the encryption algorithm are as follows −

```
from Crypto.PublicKey import RSA
from Crypto.Cipher import PKCS1_OAEP
from Crypto.Signature import PKCS1_v1_5
from Crypto.Hash import SHA512, SHA384, SHA256, SHA,
MD5 from Crypto import Random
from base64 import b64encode, b64decode
hash = "SHA-256"
```

RSA Cipher Decryption

This chapter is a continuation of the previous chapter where we followed step wise implementation of encryption using RSA algorithm and discusses in detail about it.

The function used to decrypt cipher text is as follows −

```
def decrypt(ciphertext, priv_key):
    cipher = PKCS1_OAEP.new(priv_key)
    return cipher.decrypt(ciphertext)
```

For public key cryptography or asymmetric key cryptography, it is important to maintain two important features namely Authentication and Authorization.

Symmetric Key Encryption	Asymmetric Key Encryption
It only requires a single key for both encryption and decryption.	It requires two key one to encrypt and the other one to decrypt.
The size of cipher text is same or smaller than the original plain text.	The size of cipher text is same or larger than the original plain text.
The encryption process is very fast.	The encryption process is slow.
It is used when a large amount of data is required to transfer.	It is used to transfer small amount of data.
It only provides confidentiality.	It provides confidentiality, authenticity and non-repudiation.
Examples: 3DES, AES, DES and RC4	Examples: Diffie-Hellman, ECC, El Gamal, DSA and RSA
In symmetric key encryption, resource utilization is low as compared to asymmetric key encryption.	In asymmetric key encryption, resource utilization is high.

Digital signatures

Digital signatures are the public-key primitives of message authentication. In the physical world, it is common to use handwritten signatures on handwritten or typed messages. They are used to bind signatory to the message.

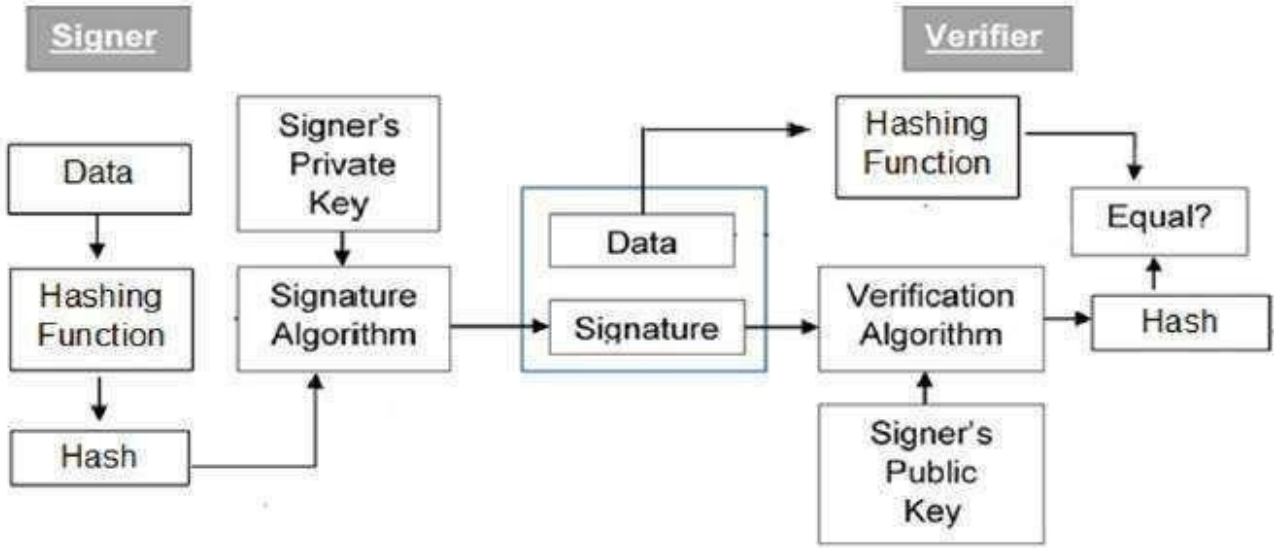
Similarly, a digital signature is a technique that binds a person/entity to the digital data. This binding can be independently verified by receiver as well as any third party.

Digital signature is a cryptographic value that is calculated from the data and a secret key known only by the signer.

In real world, the receiver of message needs assurance that the message belongs to the sender and he should not be able to repudiate the origination of that message. This requirement is very crucial in business applications, since likelihood of a dispute over exchanged data is very high.

Model of Digital Signature

As mentioned earlier, the digital signature scheme is based on public key cryptography. The model of digital signature scheme is depicted in the following illustration −



The following points explain the entire process in detail −

- **Each person adopting this scheme has a public-private key pair.**
- **Generally, the key pairs used for encryption/decryption and signing/verifying are different. The private key used for signing is referred to as the signature key and the public key as the verification key.**
- **Signer feeds data to the hash function and generates hash of data.**
- **Hash value and signature key are then fed to the signature algorithm which produces the digital signature on given hash. Signature is appended to the data and then both are sent to the verifier.**
- **Verifier feeds the digital signature and the verification key into the verification algorithm. The verification algorithm gives some value as output.**
- **Verifier also runs same hash function on received data to generate hash value.**

For verification, this hash value and output of verification algorithm are compared. Based on the comparison result, verifier decides whether the digital signature is valid.

Since digital signature is created by ‗private' key of signer and no one else can have this key; the signer cannot repudiate signing the data in future.

It should be noticed that instead of signing data directly by signing algorithm, usually a hash of data is created. Since the hash of data is a unique representation of data, it is sufficient to sign the hash in place of data. The most important reason of using hash instead of data directly for signing is efficiency of the scheme.

Let us assume RSA is used as the signing algorithm. As discussed in public key encryption chapter, the encryption/signing process using RSA involves modular exponentiation.

Signing large data through modular exponentiation is computationally expensive and time consuming. The hash of the data is a relatively small digest of the data, hence signing a hash is more efficient than signing the entire data.

Importance of Digital Signature

Out of all cryptographic primitives, the digital signature using public key cryptography is considered as very important and useful tool to achieve information security.

Apart from ability to provide non-repudiation of message, the digital signature also provides message authentication and data integrity. Let us briefly see how this is achieved by the 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 a acker 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 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.

By adding public-key encryption to digital signature scheme, we can create a cryptosystem that can provide the four essential elements of security namely − Privacy, Authen ca on, Integrity, and Non-repudiation.

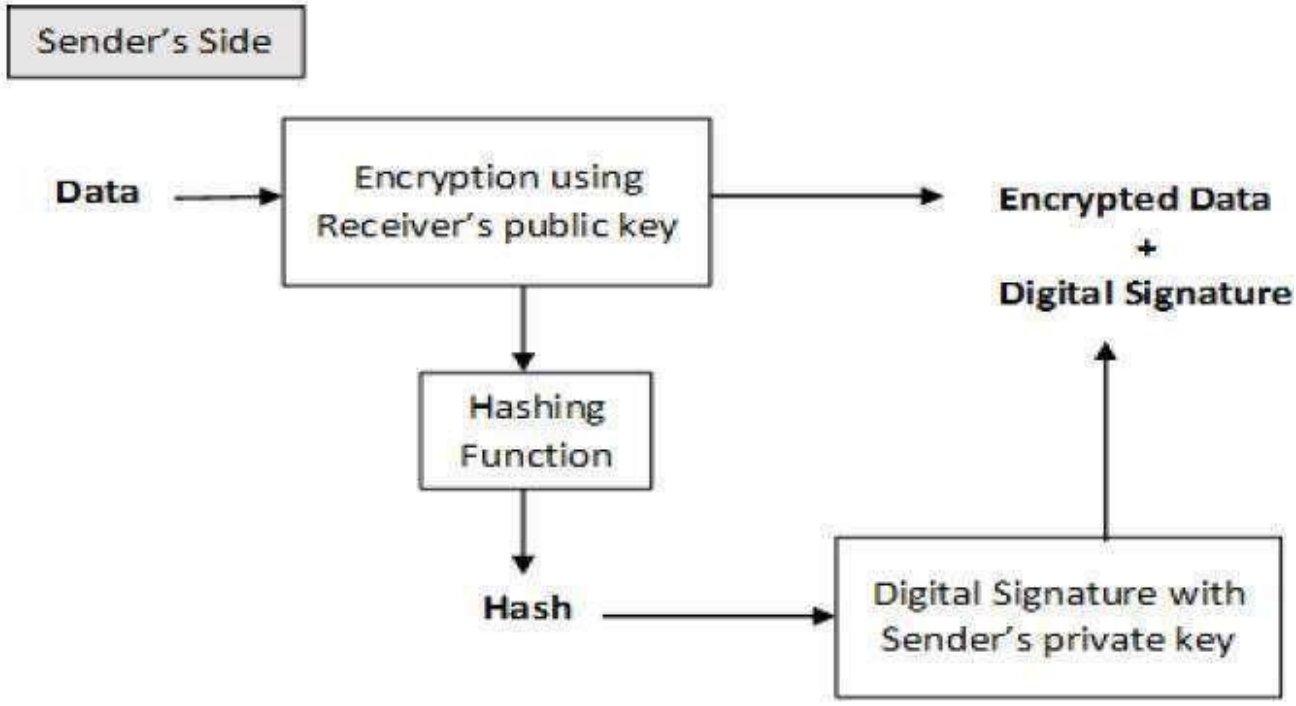
Encryption with Digital Signature

In many digital communications, it is desirable to exchange an encrypted messages than plaintext to achieve confidentiality. In public key encryption scheme, a public (encryption) key of sender is available in open domain, and hence anyone can spoof his identity and send any encrypted message to the receiver.

This makes it essential for users employing PKC for encryption to seek digital signatures along with encrypted data to be assured of message authentication and non-repudiation.

This can archived by combining digital signatures with encryption scheme. Let us briefly discuss how to achieve this requirement. There are two possibilities, sign-then-encrypt and encrypt-then-sign.

However, the crypto system based on sign-then-encrypt can be exploited by receiver to spoof identity of sender and sent that data to third party. Hence, this method is not preferred. The process of encrypt-then-sign is more reliable and widely adopted. This is depicted in the following illustration −



The receiver after receiving the encrypted data and signature on it, first verifies the signature using sender's public key. After ensuring the validity of the signature, he then retrieves the data through decryption using his private key.

CHAPTER-4

Digital Certificate

Digital certificate is issued by a trusted third party which proves sender's identity to the receiver and receiver‘s identity to the sender. A digital certificate is a certificate issued by a Certificate Authority (CA) to verify the identity of the certificate holder. The CA issues an encrypted digital certificate containing the applicant‘s public key and a variety of other identification information. Digital certificate is used to attach public key with a particular individual or an entity. 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.(used for decrypting messages and digital signatures)**
5. **Digital Signature of the certificate issuing authority.**

Digital ceritifcate is also sent with the digital signature and the message.

Digital certificate vs digital signature :

Digital signature is used to verify authenticity, integrity, non-repudiation ,i.e. it is assuring that the message is sent by the known user and not modified, while digital certificate is used to verify the identity of the user, maybe sender or receiver. Thus, digital signature and certificate are different kind of things but both are used for security. Most websites use digital certificate to enhance trust of their users.

Feature	Digital Signature	Digital Certificate
Basics / Definition	Digital signature is like a fingerprint or an attachment to a digital document that ensures its authenticity and integrity.	Digital certificate is a file that ensures holder‘s identity and provides security.
Process / Steps	Hashed value of original message is encrypted with sender‘s secret key to generate the digital signature.	It is generated by CA (Certifying Authority) that involves four steps: Key Generation, Registration, Verification, Creation.
Security Services	Authenticity of Sender, integrity of the document and non-repudiation.	It provides security and authenticity of certificate holder.

It follows Digital Signature Standard
Standard (DSS).

Key Management in Cryptography

Key Management:

In cryptography it is a very tedious task to distribute the public and private key between sender and receiver. If key is known to the third party (forger/eavesdropper) then the whole security mechanism becomes worthless. So, there comes the need to secure the exchange of keys.

There are 2 aspects for Key Management:

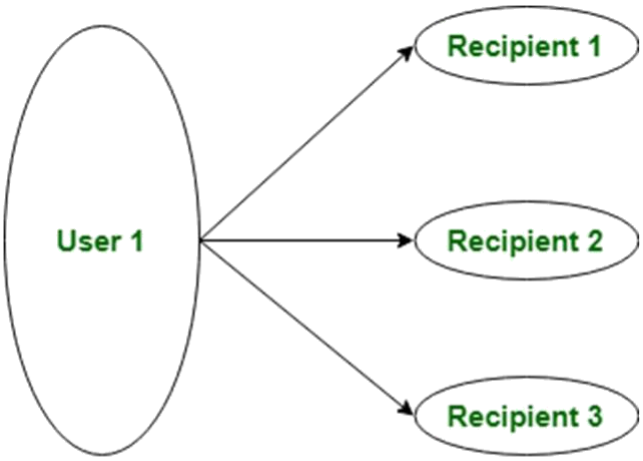
1. **Distribution of public keys.**
2. **Use of public-key encryption to distribute**

secret. Distribution of Public Key:

Public key can be distributed in 4 ways: Public announcement, publicly available directory, Public-key authority, and Public-key certificates.

These are explained as following below.

1. **Public Announcement:**
   Here the public key is broadcasted to everyone. Major weakness of this method is forgery. Anyone can create a key claiming to be someone else and broadcast it. Until forgery is discovered can masquerade as claimed user.



Public Key Announcement

2. **Publicly Available Directory:**
   In this type, the public key is stored at a public directory. Directories are trusted here, with properties like Participant Registration, access and allow to modify values at any **time, contains entries like {name, public-key}.**
   Directories can be accessed electronically still vulnerable to forgery or tampering.

3. **Public Key Authority:**
   It is similar to the directory but, improve security by tightening control over distribution of keys from directory. It requires users to know public key for the directory. Whenever the keys are needed, a real-time access to directory is made by the user to obtain any desired public key securely.

4. **Public Certification:**
   This time authority provides a certificate (which binds identity to the public key) to allow key exchange without real-time access to the public authority each time. The certificate is accompanied with some other info such as period of validity, rights of use etc. All of this content is signed by the trusted Public-Key or Certificate Authority

Public Key Infrastructure

The most distinct feature of Public Key Infrastructure (PKI) is that it uses a pair of keys to achieve the underlying security service. The key pair comprises of private key and public key.

Since the public keys are in open domain, they are likely to be abused. It is, thus, necessary to establish and maintain some kind of trusted infrastructure to manage these keys.

Key Management

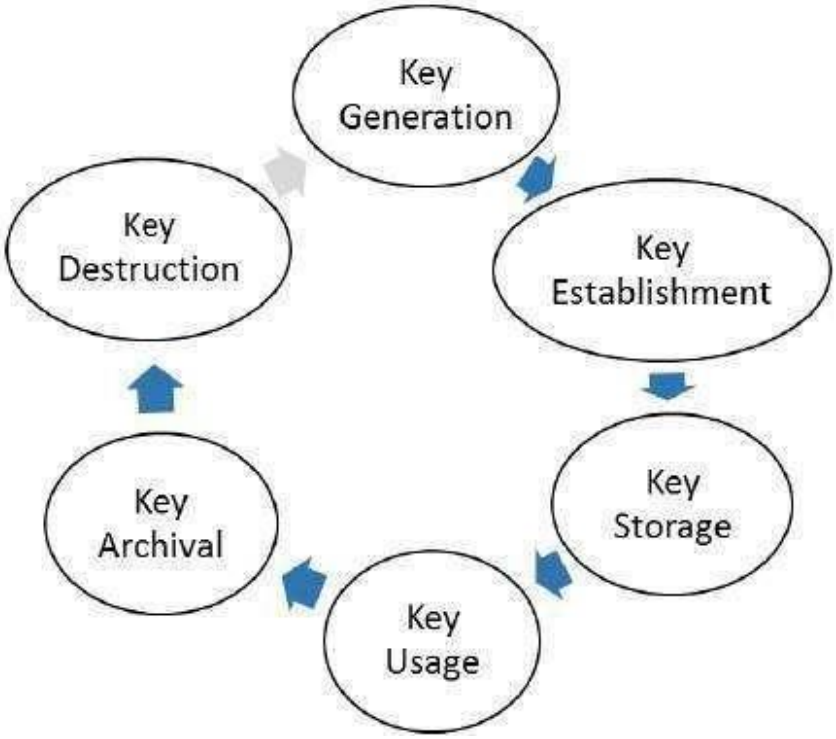
It goes without saying that the security of any cryptosystem depends upon how securely its keys are managed. Without secure procedures for the handling of cryptographic keys, the benefits of the use of strong cryptographic schemes are potentially lost.

It is observed that cryptographic schemes are rarely compromised through weaknesses in their design. However, they are often compromised through poor key management.

There are some important aspects of key management which are as follows −

Cryptographic keys are nothing but special pieces of data. Key management refers to the secure administration of cryptographic keys.

Key management deals with entire key lifecycle as depicted in the following illustration −



There are two specific requirements of key management for public key cryptography.

- **Secrecy of private keys. Throughout the key lifecycle, secret keys must remain secret from all parties except those who are owner and are authorized to use them.**
- **Assurance of public keys. In public key cryptography, the public keys are in open domain and seen as public pieces of data. By default there are no assurances of whether a public key is correct, with whom it can be associated, or what it can be used for. Thus key management of public keys needs to focus much more explicitly on assurance of purpose of public keys.**

The most crucial requirement of _assurance of public key‘ can be achieved through the public-key infrastructure (PKI), a key management systems for supporting public-key cryptography.

Public Key Infrastructure (PKI)

PKI provides assurance of public key. It provides the identification of public keys and their distribution. An anatomy of PKI comprises of the following components.

Public Key Certificate, commonly referred to as _digital certificate‘. Private Key tokens.

Certification Authority.

Registration Authority.

Certificate Management System.

Digital Certificate

For analogy, a certificate can be considered as the ID card issued to the person. People use ID cards such as a driver's license, passport to prove their identity. A digital certificate does the same basic thing in the electronic world, but with one difference.

Digital Certificates are not only issued to people but they can be issued to computers, software packages or anything else that need to prove the identity in the electronic world.

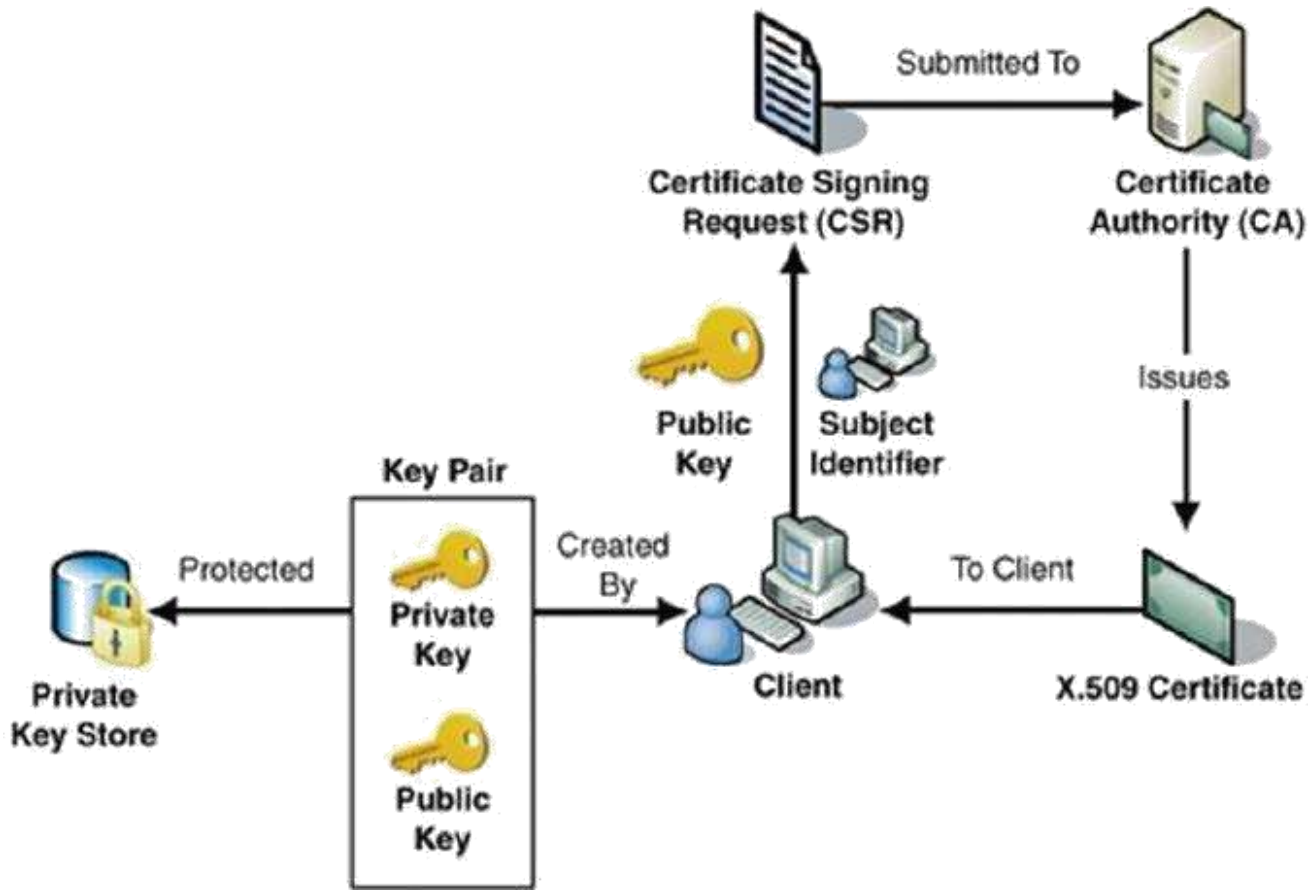
Digital certificates are based on the ITU standard X.509 which defines a standard certificate format for public key certificates and certification validation. Hence digital certificates are sometimes also referred to as X.509 certificates.

Public key pertaining to the user client is stored in digital certificates by The Certification Authority (CA) along with other relevant information such as client information, expiration date, usage, issuer etc.

CA digitally signs this entire information and includes digital signature in the certificate.

Anyone who needs the assurance about the public key and associated information of client, he carries out the signature validation process using CA‘s public key. Successful validation assures that the public key given in the certificate belongs to the person whose details are given in the certificate.

The process of obtaining Digital Certificate by a person/entity is depicted in the following illustration.



As shown in the illustration, the CA accepts the application from a client to certify his public key.

The CA, after duly verifying identity of client, issues a digital certificate to that client.

Certifying Authority (CA)

As discussed above, the CA issues certificate to a client and assist other users to verify the certificate. The CA takes responsibility for identifying correctly the identity of the client asking for a certificate to be issued, and ensures that the information contained within the certificate is correct and digitally signs it.

Key Functions of CA

The key functions of a CA are as follows −

Generating key pairs − The CA may generate a key pair independently or jointly with the client.

Issuing digital certificates − The CA could be thought of as the PKI equivalent of a passport agency − the CA issues a cer ficate a er client provides the creden als to confirm his identity. The CA then signs the certificate to prevent modification of the details contained in the certificate.

Publishing Certificates − The CA need to publish cer ficates so that users can find them. There are two ways of achieving this. One is to publish certificates in the equivalent of an electronic telephone directory. The other is to send your certificate out to those people you think might need it by one means or another.

Verifying Certificates − The CA makes its public key available in environment to assist verification of his signature on clients' digital certificate.

Revocation of Certificates − At mes, CA revokes the cer ficate issued due to some reason such as compromise of private key by user or loss of trust in the client. After revocation, CA maintains the list of all revoked certificate that is available to the environment.

Classes of Certificates

There are four typical classes of certificate −

Class 1 − these cer ficates can be easily acquired by supplying an email address.

Class 2 − these cer ficates require addi onal personal informa on to be supplied.

Class 3 − these cer ficates can only be purchased a er checks have been made about the requestor's identity.

Class 4 − they may be used by governments and financial organiza ons needing very high levels of trust.

Registration Authority (RA)

CA may use a third-party Registration Authority (RA) to perform the necessary checks on the person or company requesting the certificate to confirm their identity. The RA may appear to the client as a CA, but they do not actually sign the certificate that is issued.

Certificate Management System (CMS)

It is the management system through which certificates are published, temporarily or permanently suspended, renewed, or revoked. Certificate management systems do not normally delete certificates because it may be necessary to prove their status at a point in time, perhaps for legal reasons. A CA along with associated RA runs certificate management systems to be able to track their responsibilities and liabilities.

Private Key Tokens

While the public key of a client is stored on the certificate, the associated secret private key can be stored on the key owner's computer. This method is generally not adopted. If an attacker gains access to the computer, he can easily gain access to private key. For this reason, a private key is stored on secure removable storage token access to which is protected through a password.

Different vendors often use different and sometimes proprietary storage formats for storing keys. For example, Entrust uses the proprietary .epf format, while VeriSign, Global Sign, and Baltimore use the standard .p12 format.

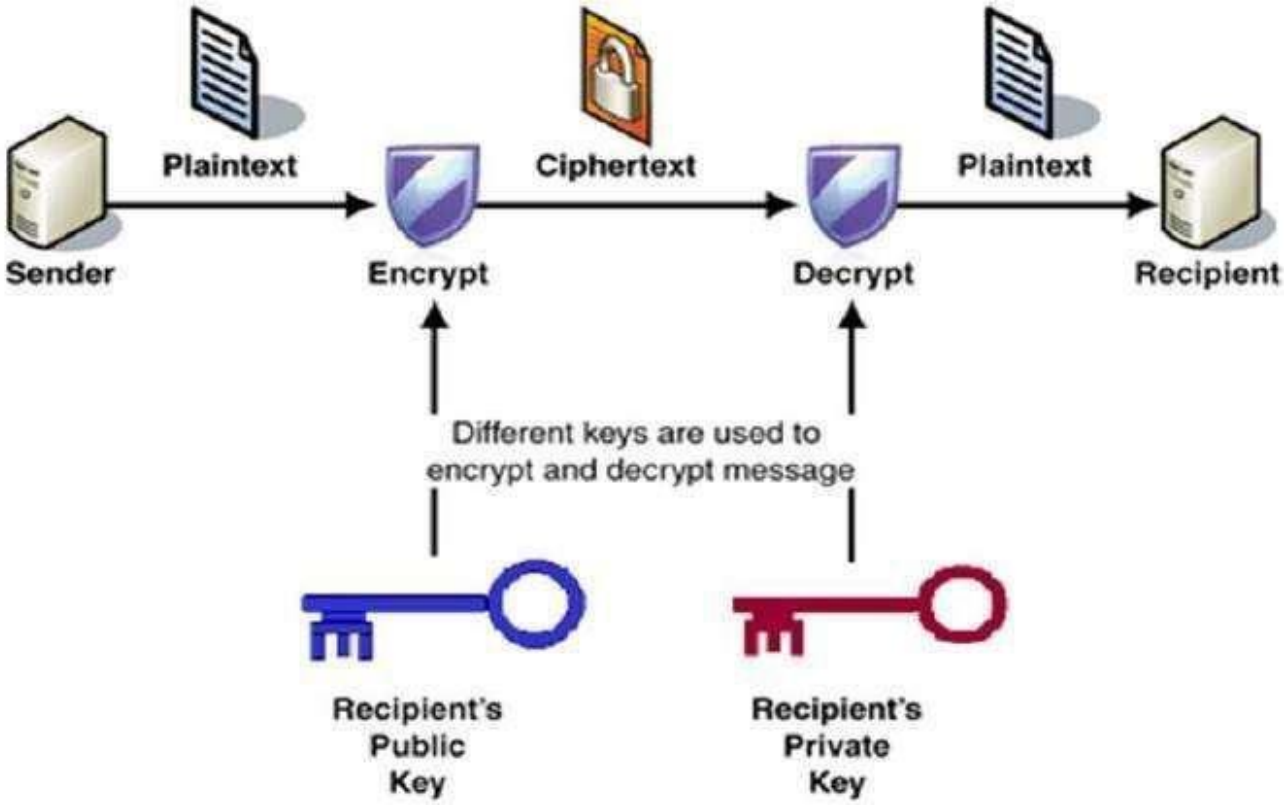
Public Key Cryptography

Unlike symmetric key cryptography, we do not find historical use of public-key cryptography. It is a relatively new concept.

Symmetric cryptography was well suited for organizations such as governments, military, and big financial corporations were involved in the classified communication.

With the spread of more unsecure computer networks in last few decades, a genuine need was felt to use cryptography at larger scale. The symmetric key was found to be non-practical due to challenges it faced for key management. This gave rise to the public key cryptosystems.

The process of encryption and decryption is depicted in the following illustration −



Public-Key Cryptography Standards (PKCS)

Public key cryptography standards (PKCS) are a group of specifications developed with the aim of accelerating the deployment of algorithms featuring two separate keys - one private and one public.

PKCS were first developed by RSA Laboratories with the cooperation of security developers from

around the world.

The first published release of PKCS was in 1991 as a result of the cooperation of early adaptors. The standards promote the use of cryptography techniques such as the RSA algorithm and the Scnorr signature.

The Public-Key Cryptography Standards (PKCS) are a set of inter vendor standard protocols for making possible secure information exchange on the Internet using a public key infrastructure (PKI). The standards include RSA encryption, password-based encryption, extended certificate syntax, and cryptographic message syntax for S/MIME, RSA's proposed standard for secure e-mail. The standards were developed by RSA Laboratories in cooperation with a consortium that included Apple, Microsoft, DEC, Lotus, Sun, and MIT.

CHAPTER-5

Internet Security

Introduction

Internet security refers to securing communication over the internet. It includes specific security protocols such as:

Internet Security Protocol (IP Sec)

Secure Socket Layer (SSL)

Internet Security Protocol (IP Sec)

It consists of a set of protocols designed by Internet Engineering Task Force (IETF). It provides security at network level and helps to create authenticated and confidential packets for IP layer.

Secure Socket Layer (SSL)

It is a security protocol developed by Netscape Communications Corporation. It provides security at transport layer. It addresses the following security issues:

Privacy

Integrity

Authentication

Threats

Internet security threats impact the network, data security and other internet connected systems. Cyber criminals have evolved several techniques to threat privacy and integrity of bank accounts, businesses, and organizations.

Following are some of the internet security threats:

Mobile worms

Malware

PC and Mobile ransomware

Large scale attacks like Stuxnet that attempts to destroy

infrastructure. Hacking as a Service

Spam

Phishing

Email Phishing

Email phishing is an activity of sending emails to a user claiming to be a legitimate enterprise. Its main purpose is to steal sensitive information such as usernames, passwords, and credit card details.

Such emails contains link to websites that are infected with malware and direct the user to enter details at a fake website whose look and feels are same to legitimate one.

What a phishing email may contain?

Following are the symptoms of a phishing email:

Spelling and bad grammar

Most often such emails contain grammatically incorrect text. Ignore such emails, since it can be a spam.

Beware of links in email

Don‘t click on any links in suspicious emails.

Threats

Such emails contain threat like —your account will be closed if you didn‘t respond to an email message‖.

Spoofing popular websites or companies

These emails contain graphics that appear to be connected to legitimate website but they actually are connected to fake websites.

Network Security – Transport Layer

Network security entails securing data against attacks while it is in transit on a network. To achieve this goal, many real-time security protocols have been designed. There are popular standards for real-time network security protocols such as S/MIME, SSL/TLS, SSH, and IPsec. As mentioned earlier, these protocols work at different layers of networking model.

For TCP/IP protocol based network, physical and data link layers are typically implemented in the user terminal and network card hardware. TCP and IP layers are implemented in the operating system. Anything above TCP/IP is implemented as user process.

Need for Transport Layer Security

Let's discuss a typical Internet-based business transaction.

Bob visits Alice‘s website for selling goods. In a form on the website, Bob enters the type of good and quantity desired, his address and payment card details. Bob clicks on Submit and waits for

delivery of goods with debit of price amount from his account. All this sounds good, but in absence of network security, Bob could be in for a few surprises.

If transactions did not use confidentiality (encryption), an attacker could obtain his payment card information. The attacker can then make purchases at Bob's expense.

If no data integrity measure is used, an attacker could modify Bob's order in terms of type or quantity of goods.

Lastly, if no server authentication is used, a server could display Alice's famous logo but the site could be a malicious site maintained by an attacker, who is masquerading as Alice. After receiving Bob's order, he could take Bob's money and flee. Or he could carry out an identity theft by collecting Bob's name and credit card details.

Transport layer security schemes can address these problems by enhancing TCP/IP based network communication with confidentiality, data integrity, server authentication, and client authentication.

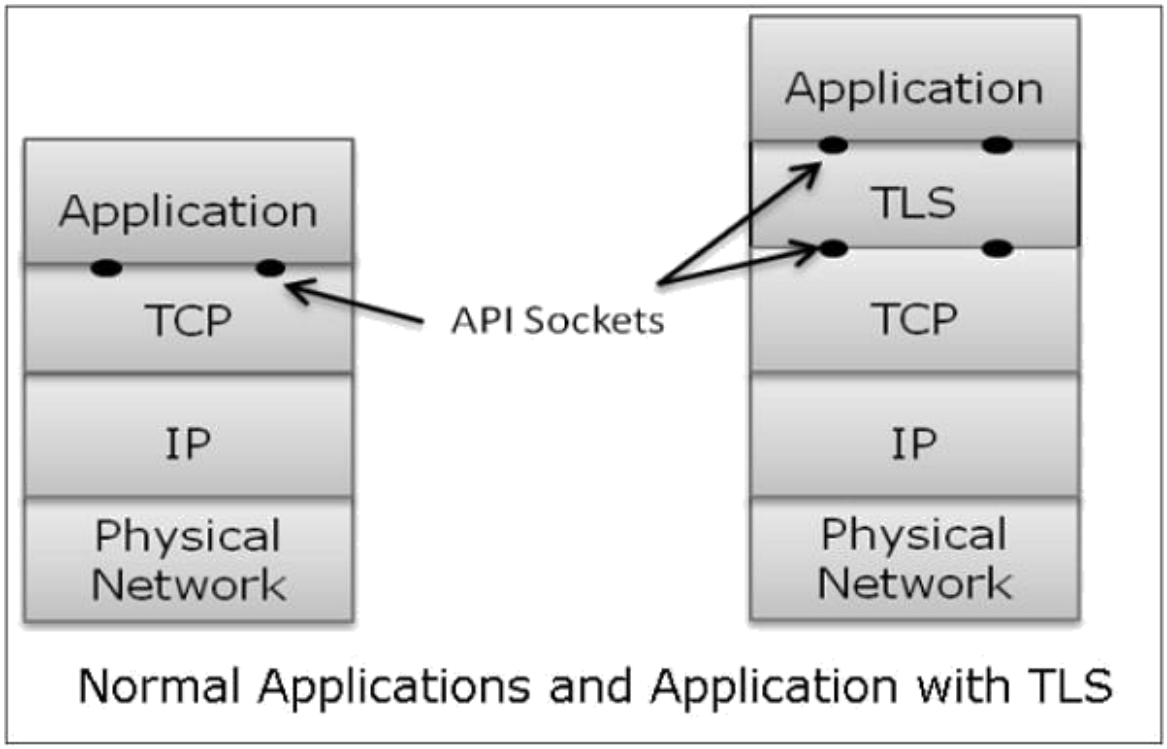
The security at this layer is mostly used to secure HTTP based web transactions on a network.

However, it can be employed by any application running over TCP.

Philosophy of TLS Design

Transport Layer Security (TLS) protocols operate above the TCP layer. Design of these protocols use popular Application Program Interfaces (API) to TCP, called —sockets" for interfacing with TCP layer.

Applications are now interfaced to Transport Security Layer instead of TCP directly. Transport Security Layer provides a simple API with sockets, which is similar and analogous to TCP's API.



In the above diagram, although TLS technically resides between application and transport layer, from the common perspective it is a transport protocol that acts as TCP layer enhanced with security services.

TLS is designed to operate over TCP, the reliable layer 4 protocol (not on UDP protocol), to make design of TLS much simpler, because it doesn't have to worry about _timing out‘ and _retransmitting lost data‘. The TCP layer continues doing that as usual which serves the need of TLS.

Why TLS is Popular?

The reason for popularity of using a security at Transport Layer is simplicity. Design and deployment of security at this layer does not require any change in TCP/IP protocols that are implemented in an operating system. Only user processes and applications needs to be designed/modified which is less complex.

Secure Socket Layer (SSL)

In this section, we discuss the family of protocols designed for TLS. The family includes SSL versions 2 and 3 and TLS protocol. SSLv2 has been now replaced by SSLv3, so we will focus on SSL v3 and TLS.

Brief History of SSL

In year 1995, Netscape developed SSLv2 and used in Netscape Navigator 1.1. The SSL version1 was never published and used. Later, Microsoft improved upon SSLv2 and introduced another similar protocol named Private Communications Technology (PCT).

Netscape substantially improved SSLv2 on various security issues and deployed SSLv3 in 1999. The Internet Engineering Task Force (IETF) subsequently, introduced a similar TLS (Transport Layer Security) protocol as an open standard. TLS protocol is non-interoperable with SSLv3.

TLS modified the cryptographic algorithms for key expansion and authentication. Also, TLS suggested use of open crypto Diffie-Hellman (DH) and Digital Signature Standard (DSS) in place of patented RSA crypto used in SSL. But due to expiry of RSA patent in 2000, there existed no strong reasons for users to shift away from the widely deployed SSLv3 to TLS.

Salient Features of SSL

The salient features of SSL protocol are as follows − SSL

provides network connection security through −

- **Confidentiality − Informa on is exchanged in an encrypted form.**
- **Authentication − Communica on en es iden fy each other through the use of digital certificates. Web-server authentication is mandatory whereas client authentication is kept optional.**
- **Reliability − Maintains message integrity checks.**

SSL is available for all TCP applications.

Supported by almost all web browsers.

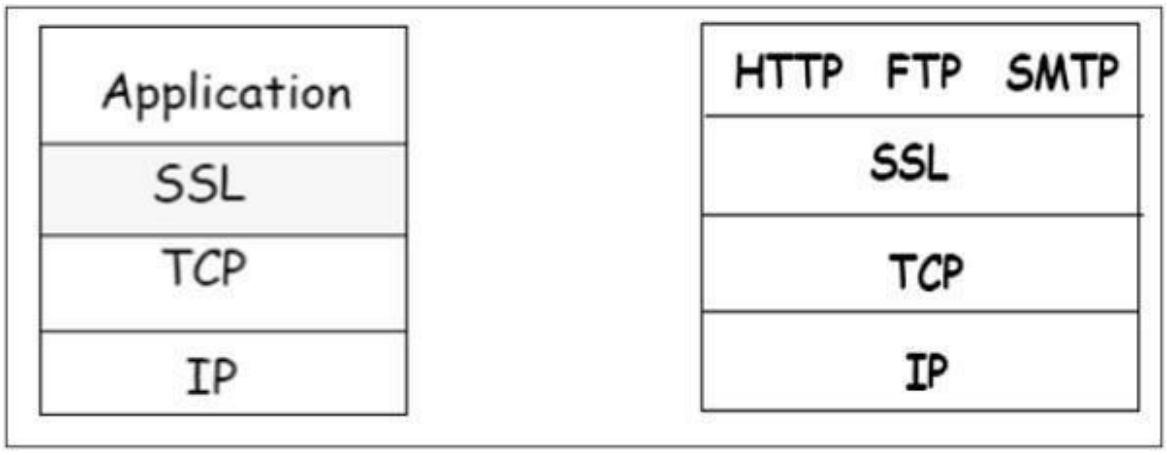
Provides ease in doing business with new online entities.

Developed primarily for Web e-commerce.

Architecture of SSL

SSL is specific to TCP and it does not work with UDP. SSL provides Application Programming Interface (API) to applications. C and Java SSL libraries/classes are readily available.

SSL protocol is designed to interwork between application and transport layer as shown in the following image −

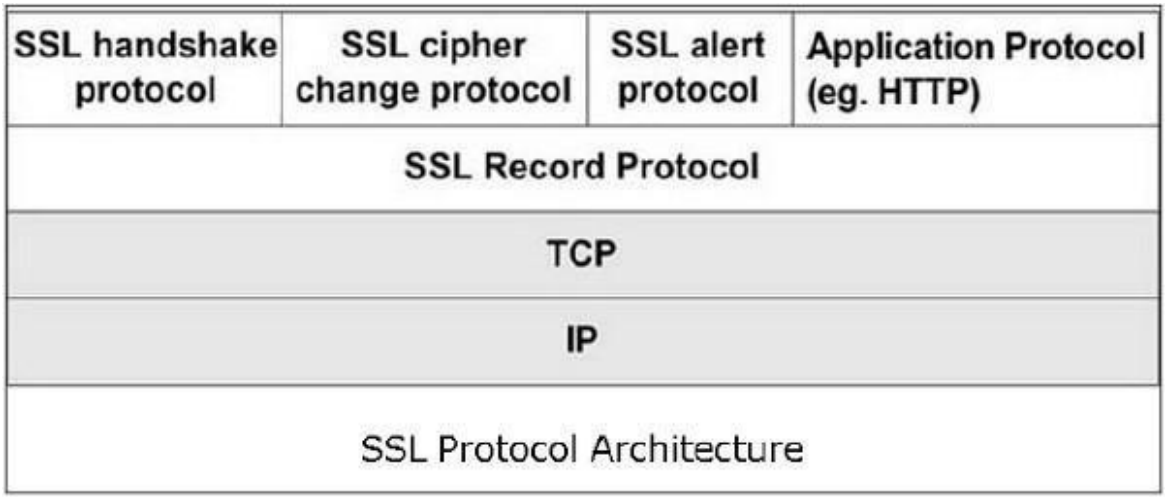


SSL itself is not a single layer protocol as depicted in the image; in fact it is composed of two sub-layers.

Lower sub-layer comprises of the one component of SSL protocol called as SSL Record Protocol. This component provides integrity and confidentiality services.

Upper sub-layer comprises of three SSL-related protocol components and an application protocol. Application component provides the information transfer service between client/server interactions. Technically, it can operate on top of SSL layer as well. Three SSL related protocol components are −

- **SSL Handshake Protocol**
- **Change Cipher Spec Protocol**
- **Alert Protocol.**

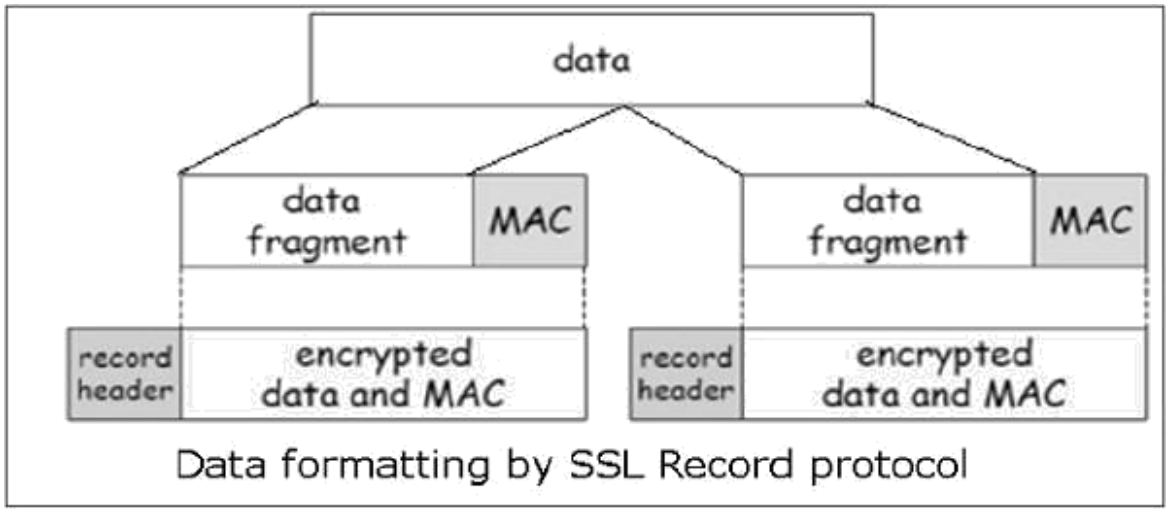


Functions of SSL Protocol Components

The four sub-components of the SSL protocol handle various tasks for secure communication between the client machine and the server.

Record Protocol

- **The record layer formats the upper layer protocol messages.**
- **It fragments the data into manageable blocks (max length 16 KB). It optionally compresses the data.**
- **Encrypts the data.**
- **Provides a header for each message and a hash (Message Authentication Code (MAC)) at the end.**
- **Hands over the formatted blocks to TCP layer for transmission.**



SSL Handshake Protocol

- **It is the most complex part of SSL. It is invoked before any application data is transmitted. It creates SSL sessions between the client and the server.**
- **Establishment of session involves Server authentication, Key and algorithm negotiation, Establishing keys and Client authentication (optional).**
- **A session is identified by unique set of cryptographic security parameters.**
- **Multiple secure TCP connections between a client and a server can share the same session.**
- **Handshake protocol actions through four**

phases. ChangeCipherSpec Protocol

- **Simplest part of SSL protocol. It comprises of a single message exchanged between two communicating entities, the client and the server.**
- **As each entity sends the ChangeCipherSpec message, it changes its side of the connection into the secure state as agreed upon.**
- **The cipher parameters pending state is copied into the current state.**
- **Exchange of this Message indicates all future data exchanges are encrypted and integrity is protected.**

SSL Alert Protocol

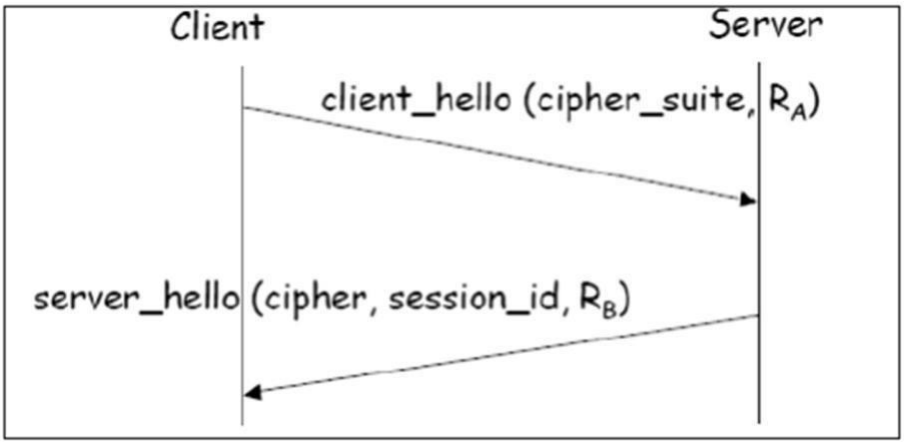
- **This protocol is used to report errors – such as unexpected message, bad record MAC, security parameters negotiation failed, etc.**
- **It is also used for other purposes – such as notify closure of the TCP connection, notify receipt of bad or unknown certificate, etc.**

Establishment of SSL Session

As discussed above, there are four phases of SSL session establishment. These are mainly handled by SSL Handshake protocol.

Phase 1 − Establishing security capabili es.

This phase comprises of exchange of two messages – Client_hello and Server_hello.



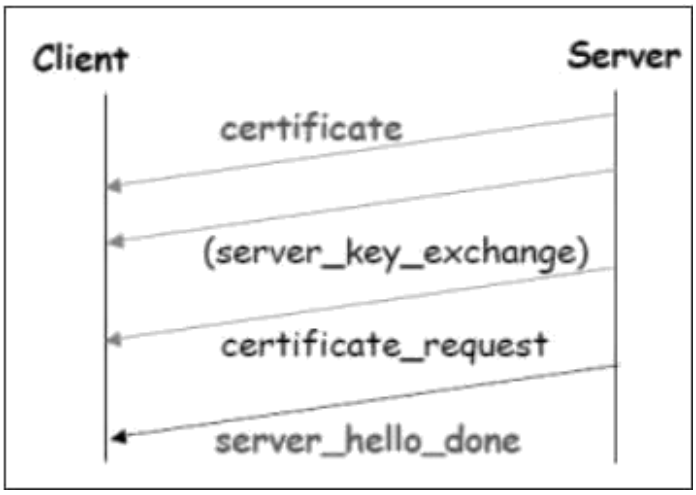
Client_hello contains of list of cryptographic algorithms supported by the client, in decreasing order of preference.

Server_hello contains the selected Cipher Specification (CipherSpec) and a new session_id.

The CipherSpec contains fields like −

- **Cipher Algorithm (DES, 3DES, RC2, and RC4)**
- **MAC Algorithm (based on MD5, SHA-1)**
- **Public-key algorithm (RSA)**
- **Both messages have —nonce‖ to prevent replay attack.**

Phase 2 − Server authen ca on and key exchange.



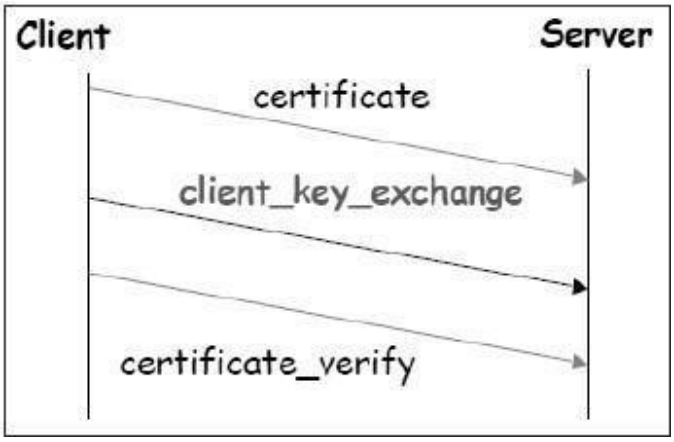
Server sends certificate. Client software comes configured with public keys of various —trusted‖ organizations (CAs) to check certificate.

Server sends chosen cipher suite.

Server may request client certificate. Usually it is not

done. Server indicates end of Server_hello.

Phase 3 − Client authentication and key exchange.

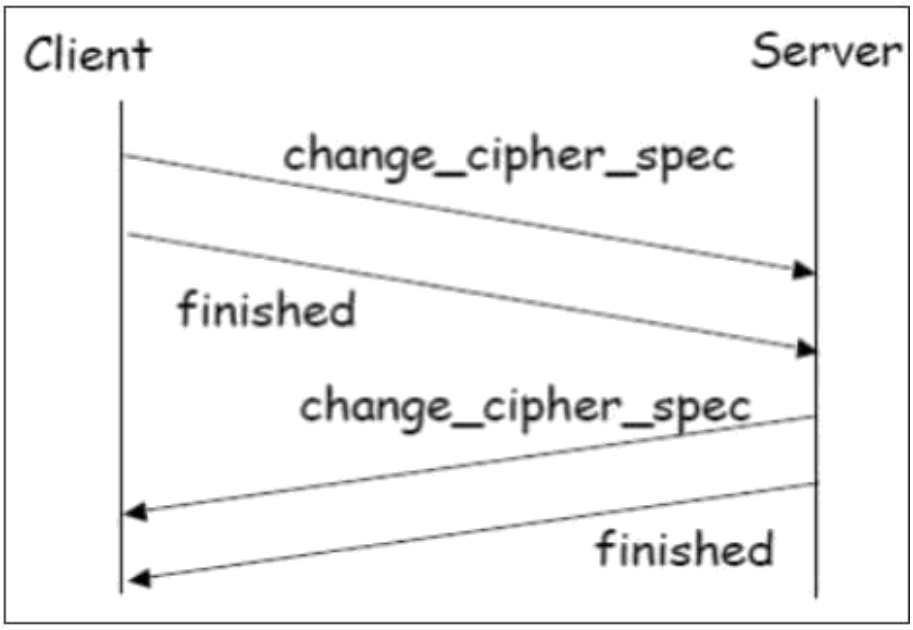


Client sends certificate, only if requested by the server.

It also sends the Pre-master Secret (PMS) encrypted with the server‘s public key.

Client also sends Certificate_verify message if certificate is sent by him to prove he has the private key associated with this certificate. Basically, the client signs a hash of the previous messages.

Phase 4 − Finish.



Client and server send Change_cipher_spec messages to each other to cause the pending cipher state to be copied into the current state.

From now on, all data is encrypted and integrity protected.

Message —Finished‖ from each end verifies that the key exchange and authentication processes were successful.

All four phases, discussed above, happen within the establishment of TCP session. SSL session establishment starts after TCP SYN/ SYNACK and finishes before TCP Fin.

Resuming a Disconnected Session

It is possible to resume a disconnected session (through Alert message), if the client sends a hello_request to the server with the encrypted session_id information.

The server then determines if the session_id is valid. If validated, it exchanges ChangeCipherSpec and finished messages with the client and secure communications resume.

This avoids recalculating of session cipher parameters and saves computing at the server and the client end.

SSL Session Keys

We have seen that during Phase 3 of SSL session establishment, a pre-master secret is sent by the client to the server encrypted using server's public key. The master secret and various session keys are generated as follows −

The master secret is generated (via pseudo random number generator)

using − o The pre-master secret.

- Two nonces (RA and RB) exchanged in the client_hello and server_hello messages.

Six secret values are then derived from this master secret as −

- Secret key used with MAC (for data sent by server)
- Secret key used with MAC (for data sent by client)
- Secret key and IV used for encryption (by server)
- Secret key and IV used for encryption (by client)

TLS Protocol

In order to provide an open Internet standard of SSL, IETF released The Transport Layer Security (TLS) protocol in January 1999. TLS is defined as a proposed Internet Standard in RFC 5246.

Salient Features

TLS protocol has same objectives as SSL.

It enables client/server applications to communicate in a secure manner by authenticating, preventing eavesdropping and resisting message modification.

TLS protocol sits above the reliable connection-oriented transport TCP layer in the networking layers stack.

The architecture of TLS protocol is similar to SSLv3 protocol. It has two sub protocols: the TLS Record protocol and the TLS Handshake protocol.

Though SSLv3 and TLS protocol have similar architecture, several changes were made in architecture and functioning particularly for the handshake protocol.

Comparison of TLS and SSL Protocols

There are main eight differences between TLS and SSLv3 protocols. These are as follows −

Protocol Version − The header of TLS protocol segment carries the version number 3.1 to differentiate between number 3 carried by SSL protocol segment header.

Message Authentication − TLS employs a keyed-hash message authentication code (H-MAC). Benefit is that H-MAC operates with any hash function, not just MD5 or SHA, as explicitly stated by the SSL protocol.

Session Key Generation − There are two differences between TLS and SSL protocol for generation of key material.

- Method of computing pre-master and master secrets is similar. But in TLS protocol, computation of master secret uses the HMAC standard and pseudorandom function (PRF) output instead of ad-hoc MAC.

- **The algorithm for computing session keys and initiation values (IV) is different in TLS than SSL protocol.**

Alert Protocol Message −

- **TLS protocol supports all the messages used by the Alert protocol of SSL, except No certificate alert message being made redundant. The client sends empty certificate in case client authentication is not required.**
- **Many additional Alert messages are included in TLS protocol for other error conditions such as record_overflow, decode_error etc.**

Supported Cipher Suites − SSL supports RSA, Diffie-Hellman and Fortezza cipher suites. TLS protocol supports all suits except Fortezza.

Client Certificate Types − TLS defines certificate types to be requested in a certificate_request message. SSLv3 support all of these. Additionally, SSL support certain other types of certificate such as Fortezza.

CertificateVerify and Finished Messages −

- **In SSL, complex message procedure is used for the certificate_verify message. With TLS, the verified information is contained in the handshake messages itself thus avoiding this complex procedure.**
- **Finished message is computed in different manners in TLS and SSLv3.**

Padding of Data − In SSL protocol, the padding added to user data before encryption is the minimum amount required to make the total data-size equal to a multiple of the cipher's block length. In TLS, the padding can be any amount that results in data-size that is a multiple of the cipher's block length, up to a maximum of 255 bytes.

The above differences between TLS and SSLv3 protocols are summarized in the following table.

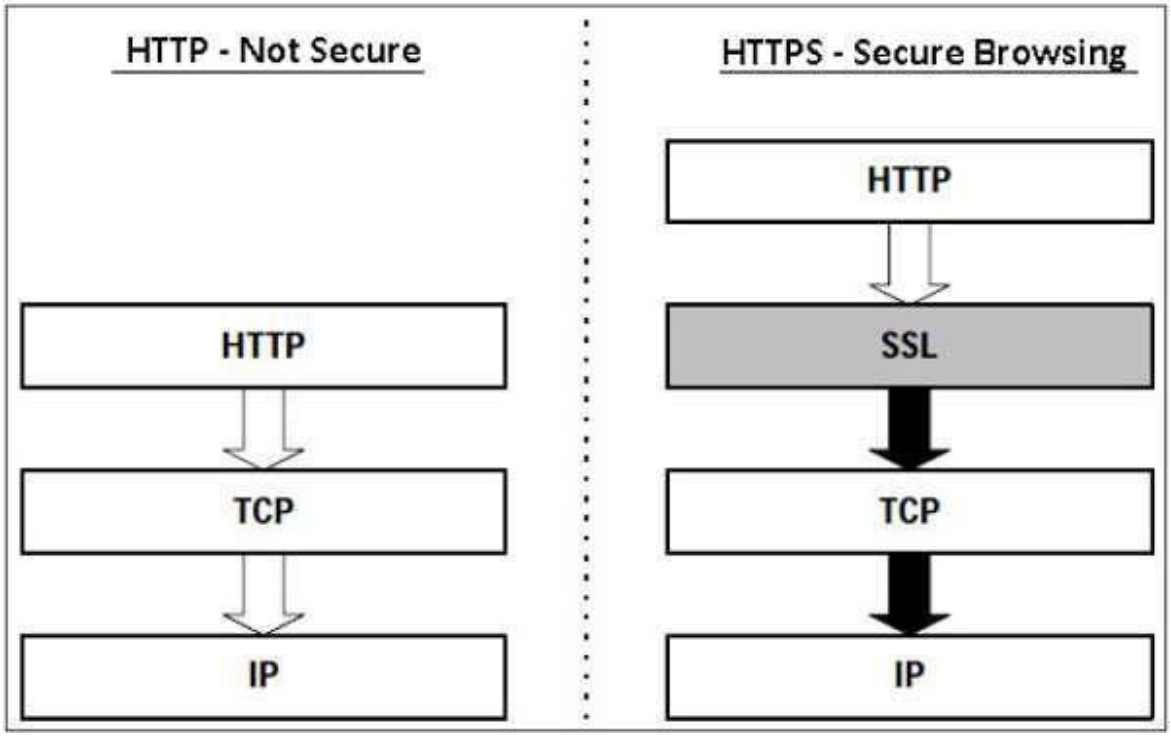
	SSL v3.0	TLS v1.0
Protocol version in messages	3.0	3.1
Alert protocol message types	12	23
Message authentication	ad hoc	standard
Key material generation	ad hoc	PRF
CertificateVerify	complex	simple
Finished	ad hoc	PRF
Baseline cipher suites	includes Fortezza	no Fortezza

Secure Browsing - HTTPS

In this section, we will discuss the use of SSL/TLS protocol for performing secure web browsing.

HTTPS Defined

Hyper Text Transfer Protocol (HTTP) protocol is used for web browsing. The function of HTTPS is similar to HTTP. The only difference is that HTTPS provides —secure‖ web browsing. HTTPS stands for HTTP over SSL. This protocol is used to provide the encrypted and authenticated connection between the client web browser and the website server.



The secure browsing through HTTPS ensures that the following content are encrypted −

URL of the requested web page.

Web page contents provided by the server to the user

client. Contents of forms filled in by user.

Cookies established in both directions.

Working of HTTPS

HTTPS application protocol typically uses one of two popular transport layer security protocols

- SSL or TLS. The process of secure browsing is described in the following points.

You request a HTTPS connection to a webpage by entering https:// followed by URL in the browser address bar.

Web browser initiates a connection to the web server. Use of https invokes the use of SSL protocol.

An application, browser in this case, uses the system port 443 instead of port 80 (used in case of http).

The SSL protocol goes through a handshake protocol for establishing a secure session as discussed in earlier sections.

The website initially sends its SSL Digital certificate to your browser. On verification of certificate, the SSL handshake progresses to exchange the shared secrets for the session.

When a trusted SSL Digital Certificate is used by the server, users get to see a padlock icon in the browser address bar. When an Extended Validation Certificate is installed on a website, the address bar turns green.

Once established, this session consists of many secure connections between the web server and the browser.

Use of HTTPS

Use of HTTPS provides confidentiality, server authentication and message integrity to the user. It enables safe conduct of e-commerce on the Internet.

Prevents data from eavesdropping and denies identity theft which are common attacks on HTTP.

Present day web browsers and web servers are equipped with HTTPS support. The use of HTTPS over HTTP, however, requires more computing power at the client and the server end to carry out encryption and SSL handshake.

Time-Stamp Protocol

The Time-Stamp Protocol, or TSP is a cryptographic protocol for certifying timestamps using X. 509 certificates and public key infrastructure. The timestamp is the signer's assertion that a piece of electronic data existed at or before a particular time.

A timestamp is the current time of an event that is recorded by a computer. Through mechanisms such as the Network Time Protocol (NTP), a computer maintains accurate current time, calibrated to minute fractions of a second.

One may also ask, what are the types of time stamping? Timestamping is different types those are digital timestamping, network timestamping, trusted timestamping and etc. Timestamps are a postmark on a letter or 'in' and 'out' times on a time record.

timestamp in transcription

In audio transcription, time stamping refers to aligning the transcript's texts to the recording by inserting timestamps at specific intervals. It is usually an add-on service designed to add value to transcripts, particularly to various types of market research transcriptions.

What is timestamp example?

The TIMESTAMP data type is used for values that contain both date and time parts. TIMESTAMP has a range of '1970-01-01 00:00:01' UTC to '2038-01-19 03:14:07' UTC. A DATETIME or TIMESTAMP value can include a trailing fractional second‘s part in up to microseconds (6 digits) precision.

When the date and time of an event is recorded, we say that it is time stamped. A digital camera will record the time and date of a photo being taken, a computer will record the time and date of a document being saved and edited. A social media post may have date and time recorded. These are all examples of a timestamp.

Timestamps are important for keeping records of when information is being exchanged or created or deleted online. In many cases, these records are simply useful for us to know about. But in some cases, a timestamp is more valuable.

Imagine this scenario: your organization electronically signs a legal agreement or NDA with another organization or contractor. Later down the line, it is discovered that the contractor has leaked information about the project under which the NDA was signed. The contractor disputes the NDA, arguing that information was shared prior to the signing of the NDA. Knowing when that document was actually signed is essential here.

In a legal setting like this, it‘s not enough to just have a timestamp. If your argument comes down to when the NDA was signed, you need to be able to prove that the timestamp of the signature is valid, that it says the document was signed when it was actually signed. Timestamps that rely on system clocks are not enough, since it‘s not difficult to alter the date and time locally on a machine.

Secure Electronic Transaction (SET) Protocol

Secure Electronic Transaction or SET is a system which ensures security and integrity of electronic transactions done using credit cards in a scenario. SET is not some system that enables payment but it is a security protocol applied on those payments. It uses different encryption and hashing techniques to secure payments over internet done through credit cards. SET protocol was supported in development by major organizations like Visa, Mastercard, Microsoft which provided its Secure Transaction Technology (STT) and NetScape which provided technology of Secure Socket Layer (SSL).

SET protocol restricts revealing of credit card details to merchants thus keeping hackers and thieves at bay. SET protocol includes Certification Authorities for making use of standard Digital Certificates like X.509 Certificate.

Before discussing SET further, let's see a general scenario of electronic transaction, which includes client, payment gateway, client financial institution, merchant and merchant financial institution.

Requirements in SET :

SET protocol has some requirements to meet, some of the important requirements are :

- **It has to provide mutual authentication i.e., customer (or cardholder) authentication by confirming if the customer is intended user or not and merchant authentication.**
- **It has to keep the PI (Payment Information) and OI (Order Information) confidential by appropriate encryptions.**
- **It has to be resistive against message modifications i.e., no changes should be allowed in the content being transmitted.**

SET also needs to provide interoperability and make use of best security mechanisms.

CHAPTER-6

Authentication in Computer Network

Authentication is the process of verifying the identity of user or information. User authentication is the process of verifying the identity of user when that user logs into a computer system.
The main objective of authentication is to allow authorized users to access the computer and to deny access to the unauthorized users. Operating Systems generally identifies/authenticates users using following 3 ways : Passwords, Physical identification, and Biometrics. These are explained as following below.

1. **Passwords :**
   Passwords verification is the most popular and commonly used authentication technique. A password is a secret text that is supposed to be known only to a user. In password based system, each user is assigned a valid username and password by the system administrator.
   System stores all username and Passwords. When a user logs in, its user name and password is verified by comparing it with stored login name and password. If the contents are same then the user is allowed to access the system otherwise it is rejected.

2. **Physical Identification :**
   This technique include machine readable badges(symbols), card or smart cards. In some companies, badges are required for employees to gain access to the organization's gate. In many system, identification is combined with the use of password i.e the user must insert the card and then supply his /her password. This kind of authentication is commonly used with ATM. Smart card can enhance this scheme by keeping the user password within the card itself. This allow the authentication without storage of password in the computer system. The loss of such card can be dangerous.

3. **Biometrics :**
   This method of authentication is based on the unique biological characteristics of each user such as finger prints, voice or face recognition, signatures and eyes.

Biometric devices often consist of –

A scanner or other devices to gather the necessary data about user.

Software to convert the data into a form that can be compared and stored.

A database that stores information for all authorized users.

A number of different types of physical characteristics are –

Facial Characteristics – Humans are differentiated on the basis of facial characteristics such as eyes, nose, lips, eyebrows and chin shape.

Fingerprints –

Fingerprints are believed to he unique across the entire human population.

Hand Geometry –

Hand geometry systems identify features of hand that includes shape, length and width of fingers.

Retinal pattern –

It is concerned with the detailed structure of the eye.

Signature –

Every individual has a unique style of handwriting, and this feature is reflected in the signatures of a person.

Voice –

This method records the frequency pattern of the voice of an individual speaker.

One Time passwords :

One-time passwords provide additional security along with normal authentication. In One-Time Password system, a unique password is required every time user tries to login into the system. Once a one-time password is used, then it cannot be used again. One- time password are implemented in various ways. Some commercial applications send one-time passwords to user on registered mobile/ email which is required to be entered prior to login.

4: Token-based authentication

Token-based authentication is a protocol which allows users to verify their identity, and in return receive a unique access token. During the life of the token, users then access the website or app that the token has been issued for, rather than having to re-enter credentials each time they go back to the same webpage, app, or any resource protected with that same token.

Auth tokens work like a stamped ticket. The user retains access as long as the token remains valid. Once the user logs out or quits an app, the token is invalidated.

Token-based authentication is different from traditional password-based or server-based authentication techniques. Tokens offer a second layer of security, and administrators have detailed control over each action and transaction.

But using tokens requires a bit of coding know-how. Most developers pick up the techniques quickly, but there is a learning curve.

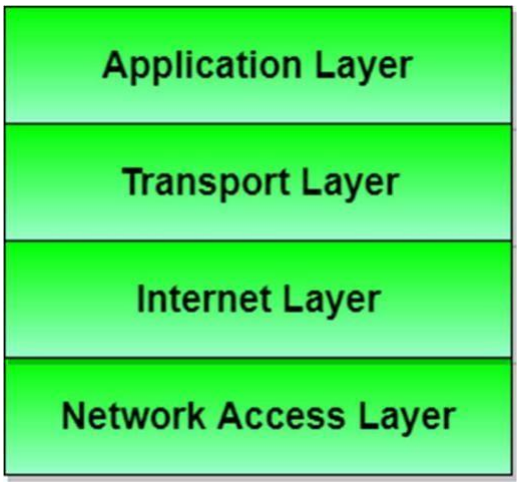
5 : Certificate-based authentication

Certificate-based authentication is based on what the user has, which is the user's private key, and what the user knows, which is the password that protects the private key (if the key is not located in a secure KeyStore. However, both of these assumptions are true only if unauthorized personnel have not gained access to the user's workstation or password, the password for the client's private key database has been set, and the client is set up to request the password at reasonably frequent intervals. Although certificate- based authentication addresses security, it does not address issues related to the physical access of individual workstations or passwords. Public key cryptography only verifies that a private key that is used to sign some information corresponds to the public key in a certificate. It is your responsibility to protect the physical security of a workstation and to keep the password for the private key a secret.

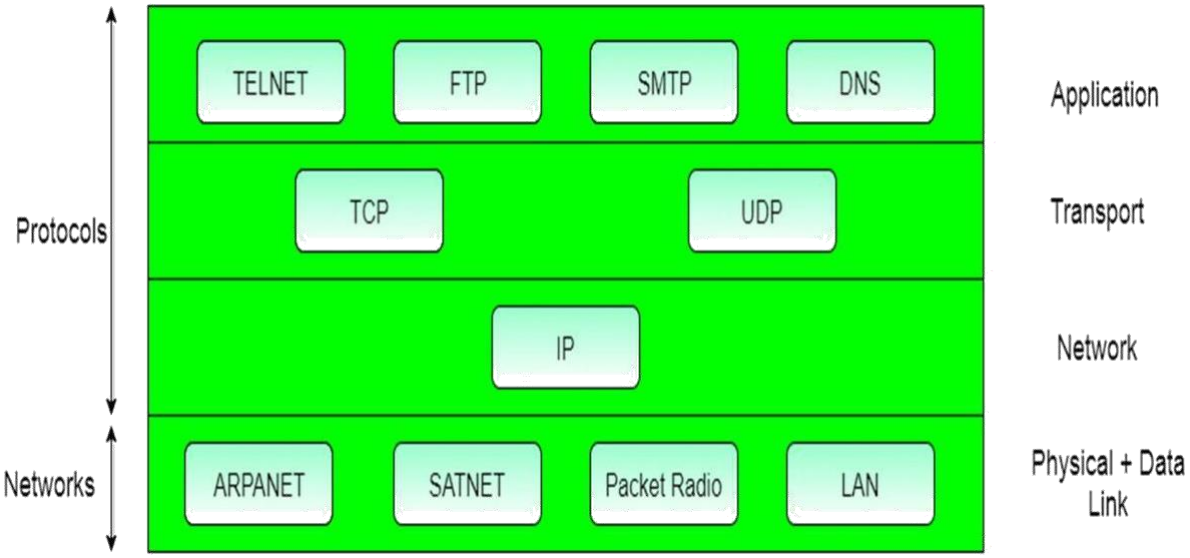
CHAPTER-7

The TCP/IP Reference Model

TCP/IP means Transmission Control Protocol and Internet Protocol. It is the network model used in the current Internet architecture as well. Protocols are set of rules which govern every possible communication over a network. These protocols describe the movement of data between the source and destination or the internet. They also offer simple naming and addressing schemes.



Protocols and networks in the TCP/IP model:



Overview of TCP/IP reference model

TCP/IP that is Transmission Control Protocol and Internet Protocol was developed by Department of Defence's Project Research Agency (ARPA, later DARPA) as a part of a research project of network interconnection to connect remote machines.

The features that stood out during the research, which led to making the TCP/IP reference model were:

- **Support for a flexible architecture. Adding more machines to a network was easy.**
- **The network was robust, and connections remained intact untill the source and destination machines were functioning.**

The overall idea was to allow one application on one computer to talk to(send data packets) another application running on different computer.

Different Layers of TCP/IP Reference Model

Below we have discussed the 4 layers that form the TCP/IP reference model:

Layer 1: Host-to-network Layer

1. **Lowest layer of the all.**
2. **Protocol is used to connect to the host, so that the packets can be sent over it.**
3. **Varies from host to host and network to network.**

Layer 2: Internet layer

1. **Selection of a packet switching network which is based on a connectionless internetwork layer is called a internet layer.**
2. **It is the layer which holds the whole architecture together.**
3. **It helps the packet to travel independently to the destination.**
4. **Order in which packets are received is different from the way they are sent.**
5. **IP (Internet Protocol) is used in this layer.**
6. **The various functions performed by the Internet Layer are:**
    - **Delivering IP packets**
    - **Performing routing**
    - **Avoiding congestion**

Layer 3: Transport Layer

1. **It decides if data transmission should be on parallel path or single path.**
2. **Functions such as multiplexing, segmenting or splitting on the data is done by transport layer.**
3. **The applications can read and write to the transport layer.**
4. **Transport layer adds header information to the data.**
5. **Transport layer breaks the message (data) into small units so that they are handled more efficiently by the network layer.**
6. **Transport layer also arrange the packets to be sent, in sequence.**

Layer 4: Application Layer

The TCP/IP specifications described a lot of applications that were at the top of the protocol stack. Some of them were TELNET, FTP, SMTP, DNS etc.

1. **TELNET is a two-way communication protocol which allows connecting to a remote machine and run applications on it.**
2. **FTP(File Transfer Protocol) is a protocol, that allows File transfer amongst computer users connected over a network. It is reliable, simple and efficient.**
3. **SMTP(Simple Mail Transport Protocol) is a protocol, which is used to transport electronic mail between a source and destination, directed via a route.**
4. **DNS(Domain Name Server) resolves an IP address into a textual address for Hosts connected over a network.**
5. **It allows peer entities to carry conversation.**

6. **It defines two end-to-end protocols: TCP and UDP**
    - **TCP(Transmission Control Protocol): It is a reliable connection-oriented protocol which handles byte-stream from source to destination without error and flow control.**
    - **UDP(User-Datagram Protocol): It is an unreliable connection-less protocol that do not want TCPs, sequencing and flow control. Eg: One-shot request-reply kind of service.**

Merits of TCP/IP model

1. **It operated independently.**
2. **It is scalable.**
3. **Client/server architecture.**
4. **Supports a number of routing protocols.**
5. **Can be used to establish a connection between two computers.**

Demerits of TCP/IP

1. **In this, the transport layer does not guarantee delivery of packets.**
2. **The model cannot be used in any other application.**
3. **Replacing protocol is not easy.**
4. **It has not clearly separated its services, interfaces and protocols.**

Firewall Tutorial

Nowadays, it is a big challenge to protect our sensitive data from unwanted and unauthorized sources. There are various tools and devices that can provide different security levels and help keep our private data secure. One such tool is a 'firewall' that prevents unauthorized access and keeps our computers and data safe and secure.

In this article, we have talked about firewalls as well as other related topics, such as why we need firewalls, functions of firewalls, limitations of firewalls, working of firewalls, etc.

What is a Firewall?

A firewall can be defined as a special type of network security device or a software program that monitors and filters incoming and outgoing network traffic based on a defined set of security rules. It acts as a barrier between internal private networks and external sources (such as the public Internet).

The primary purpose of a firewall is to allow non-threatening traffic and prevent malicious or unwanted data traffic for protecting the computer from viruses and attacks. A firewall is a cybersecurity tool that filters network traffic and helps users block malicious software from accessing the Internet in infected computers.

Firewall: Hardware or Software

This is one of the most problematic questions whether a firewall is a hardware or software. As stated above, a firewall can be a network security device or a software program on a computer. This means that the firewall comes at both levels, i.e., hardware and software, though it's best to have both.

Each format (a firewall implemented as hardware or software) has different functionality but the same purpose. A hardware firewall is a physical device that attaches between a computer network and a gateway. For example, a broadband router. On the other hand, a software firewall is a simple program installed on a computer that works through port numbers and other installed software.

Apart from that, there are cloud-based firewalls. They are commonly referred to as FaaS (firewall as a service). A primary advantage of using cloud-based firewalls is that they can be managed centrally. Like hardware firewalls, cloud-based firewalls are best known for providing perimeter security.

Why Firewall

Firewalls are primarily used to prevent malware and network-based attacks. Additionally, they can help in blocking application-layer attacks. These firewalls act as a gatekeeper or a barrier. They monitor every attempt between our computer and another network. They do not allow data packets to be transferred through them unless the data is coming or going from a user-specified trusted source.

Firewalls are designed in such a way that they can react quickly to detect and counter- attacks throughout the network. They can work with rules configured to protect the network and perform quick assessments to find any suspicious activity. In short, we can point to the firewall as a traffic controller.

IP security (IPSec)

The IP security (IPSec) is an Internet Engineering Task Force (IETF) standard suite of protocols between 2 communication points across the IP network that provide data authentication, integrity, and confidentiality. It also defines the encrypted, decrypted and authenticated packets. The protocols needed for secure key exchange and key management are defined in it.

Uses of IP Security –

IPsec can be used to do the following things:

To encrypt application layer data.

To provide security for routers sending routing data across the public internet.

To provide authentication without encryption, like to authenticate that the data originates from a known sender.

To protect network data by setting up circuits using IPsec tunneling in which all data is being sent between the two endpoints is encrypted, as with a Virtual Private

Components of IP Security –

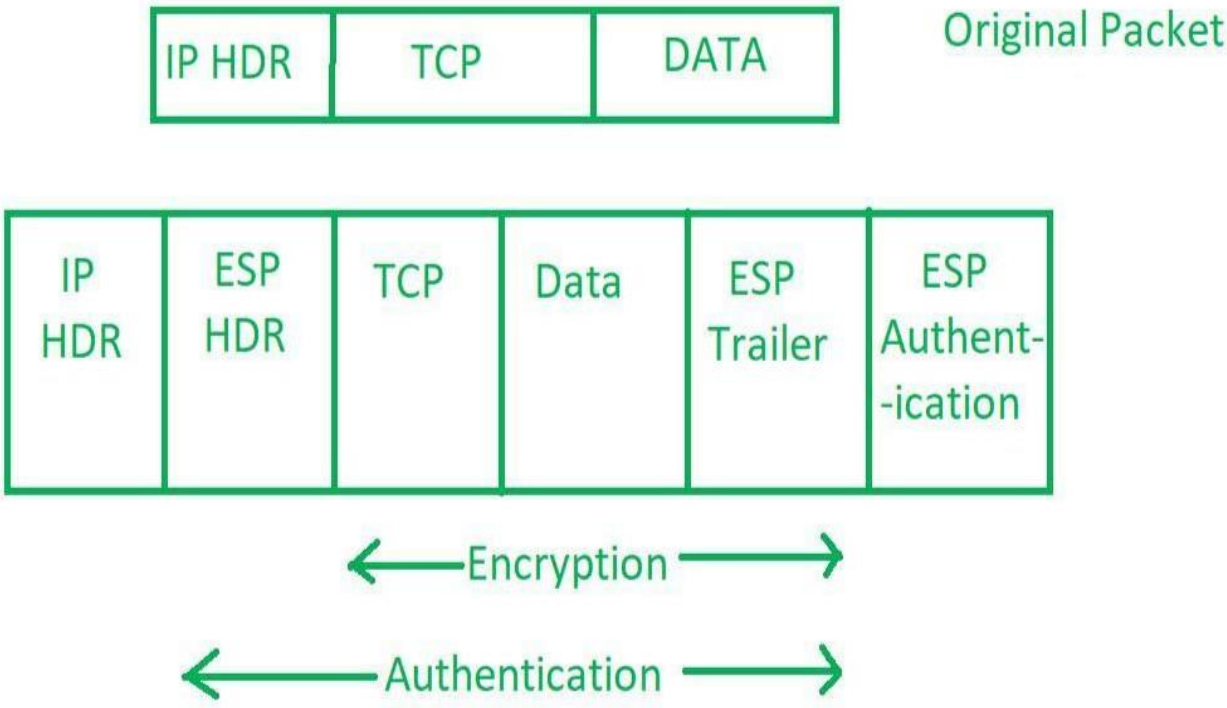
It has the following components:

1. **Encapsulating Security Payload (ESP) –**
   It provides data integrity, encryption, authentication and anti replay. It also provides authentication for payload.
2. **Authentication Header (AH) –**
   It also provides data integrity, authentication and anti replay and it does not provide encryption. The anti replay protection, protects against unauthorized transmission of packets. It does not protect data‘s confidentiality.

3. **Internet Key Exchange (IKE) –**

It is a network security protocol designed to dynamically exchange encryption keys and find a way over Security Association (SA) between 2 devices. The Security Association (SA) establishes shared security attributes between 2 network entities to support secure communication. The Key Management Protocol (ISAKMP) and Internet Security Association which provides a framework for authentication and key exchange. ISAKMP tells how the set up of the Security Associations (SAs) and how direct connections between two hosts that are using IPsec.

Internet Key Exchange (IKE) provides message content protection and also an open frame for implementing standard algorithms such as SHA and MD5. The algorithm‘s IP sec users produces a unique identifier for each packet. This identifier then allows a device to determine whether a packet has been correct or not. Packets which are not authorized are discarded and not given to receiver.



Working of IP Security –

1. **The host checks if the packet should be transmitted using IPsec or not. These packet traffic triggers the security policy for themselves. This is done when the system sending the packet apply an appropriate encryption. The incoming packets are also checked by the host that they are encrypted properly or not.**
2. **Then the IKE Phase 1 starts in which the 2 hosts(using IPsec) authenticate themselves to each other to start a secure channel. It has 2 modes. The Main mode which provides the greater security and the Aggressive mode which enables the host to establish an IPsec circuit more quickly.**
3. **The channel created in the last step is then used to securely negotiate the way the IP circuit will encrypt data across the IP circuit.**
4. **Now, the IKE Phase 2 is conducted over the secure channel in which the two hosts negotiate the type of cryptographic algorithms to use on the session and agreeing on secret keying material to be used with those algorithms.**
5. **Then the data is exchanged across the newly created IPsec encrypted tunnel. These packets are encrypted and decrypted by the hosts using IPsec SAs.**
6. **When the communication between the hosts is completed or the session times out then the IPsec tunnel is terminated by discarding the keys by both the hosts.**

Virtual Private Network (VPN)

VPN stands for virtual private network. A virtual private network (VPN) is a technology that creates a safe and encrypted connection over a less secure network, such as the internet. Virtual Private network is a way to extend a private network using a public network such as internet. The name only suggests that it is Virtual —private network‖ i.e. user can be the part of local network sitting at a remote location. It makes use of tunneling protocols to establish a secure connection.

Lets understand VPN by an example:

Think of a situation where corporate office of a bank is situated in Washington,USA.This office has a local network consisting of say 100 computers. Suppose another branches of bank are in Mumbai, India and Tokyo, Japan. The traditional method of establishing a secure connection between head office and branch was to have a leased line between the branches and head office which was very costly as well as troublesome job. VPN let us overcome this issue in an effective manner.