PNS SCHOOL OF ENGINEERING & TECHNOLOGY

Nishamani Vihar, Marshaghai, Kendrapara

LECTURE NOTES

ON

CRYPTOGRAPHY AND NETWORK SECURITY

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

6TH SEMESTER

PREPARED BY

MR. BISWARANJAN SWAIN

LECTURER IN COMPUTER SCIENCE & ENGINEERING

Syllabus

Unit 1. Possible attacks on Computers

1.1 The need for security

1.2 Security approach

1.3 Principles of security

1.4 Types of attacks

Unit 2: Cryptography Concepts

2.1 Plain text & Cipher Text

2.2 Substitution techniques

2.3 Transposition techniques

2.4 Encryption & Decryption

2.5 Symmetric & Asymmetric key cryptography

Unit 3: Symmetric & Asymmetric key algorithms

3.1 Symmetric key algorithm types

3.2 Overview of Symmetric key cryptography

3.3 Data encryption standards

3.4 Over view of Asymmetric key cryptography

3.5 The RSA algorithm

3.6 Symmetric & Asymmetric key cryptography

3.7 Digital signature

Unit 4: Digital certificate & Public key infrastructure

4.1 Digital certificates

4.2 Private key management

4.3 PKIX Model

4.4 Public key cryptography standards

Unit 5: Internet security protocols

5.1 Basic concept

5.2 Secure socket layer

5.3 Transport layer security

5.4 Secure Hyper text transfer protocol(SHTTP)

5.5 Time stamping protocol (TSP)

5.6 Secure electronic transaction (SET)

Unit 6: User authentication

6.1 Authentication basics

6.2 Password

6.3 Authentication Tokens

6.4 Certificate based authentication

6.5 Biometric authentication

Unit 7 . Network Security & VPN

7.1 Brief introduction of TCP/IP

7.2 Firewall

7.3 IP Security

7.4 Virtual Private Network (VPN)

UNIT-1

Possible Attacks on Computers

INTRODUCTION:

The computer security in the contemporary sense is that the security of the data over the network or stored in the computer. The different aspects of security are

- ➢ Computer Security - generic name for the collection of tools designed to protect data and to prevent 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.

1.1 THE NEED FOR SECURITY :

The computer security specifically relates to network security.

It is the security against attackers and hackers.

Network Security includes two basic securities.

The first is the security of data information i.e. to protect the information from Unauthorized access and loss.

Second is computer security i.e. to protect your computer system from unwanted damages caused due to network. One of the major reason for such damages are the viruses and spywares that can wipe off all the information from your hard disk or sometimes they may be enough destructive and may cause hardware problems too.

So there is a need to protect data and to thwart(prevent) hackers.

The security is needed for the following given reasons.

- ➢ To protect the secret information of users on the net only. No other person should see or access it.
- ➢ To protect the information from unwanted editing, accidently or intentionally by unauthorized users.
- ➢ To protect the information from loss and make it to be delivered to its destination properly.
- ➢ To manage for acknowledgement of message received by any node in order to protect from denial by sender in specific situations.
- ➢ To restrict a user to send some message to another user with name of a third one. For example a user X for his own interest makes a message containing some favourable instructions and sends it to user Y in such a manner that Y accepts the message as coming from Z, the manager of the organization.
- ➢ To protect the message from unwanted delay in the transmission lines/route in order to deliver it to required destination in time, in case of urgency.

1.2 SECURITY APPROACH

An organization can take several approaches to implement security model, so these are

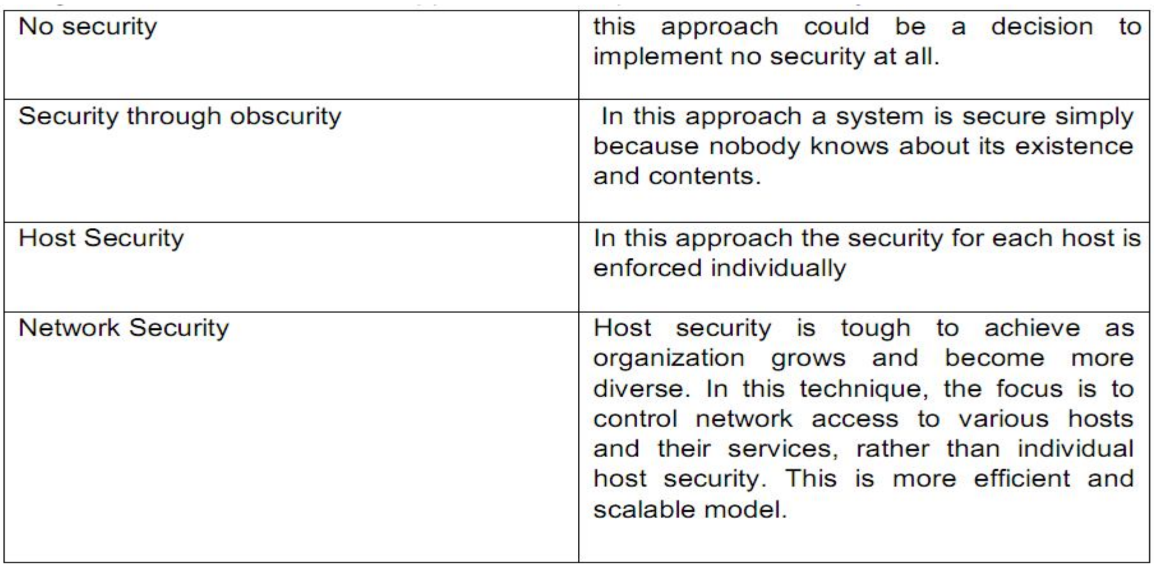
Trusted system

A trusted system is a system that is relied upon to a specified extent to enforce a specified security policy. As such, a trusted system is one whose failure may break a specified security policy.

Trusted systems are used for the processing, storage and retrieval of sensitive or classified information.

Security model

An organization can take several approaches to implements its security model.



No security	this approach could be a decision to implement no security at all.
Security through obscurity	In this approach a system is secure simply because nobody knows about its existence and contents.
Host Security	In this approach the security for each host is enforced individually
Network Security	Host security is tough to achieve as organization grows and become more diverse. In this technique, the focus is to control network access to various hosts and their services, rather than individual host security. This is more efficient and scalable model.

Security Management Practices

Good security management practices always talk of a security policy being in place.

A good security policy and its proper implementation ensures security management practices.

A good security policy generally takes care of four key aspects, as follows

Affordability : - Cost and effort in security implementation must be affordable (cost effective).

Functionality : - The mechanism of providing security must be simple.

Cultural issues: - The policy must withstand peoples' expectations, working style and beliefs.

Legality : - Policy must meet the legal requirements.

Once a security policy is in place, the following points should be ensured.

- The policy must be explained clearly to all the stake holders.
- Responsibility of everyone should be outlined.
- Simple language must be used in all the communications such that everybody could understand.
- Accountability must be fixed.
- There must be periodic reviews to verify that all the principles are being implemented or not. Also there should be provision for exception for smooth functioning of the system.

1.3 PRINCIPLES OF SECURITY

There are four chief principles of security. For a system to be secured these principles must be followed. Those are:

Confidentiality

Integrity

Authentication

Non- repudiation.

There are two more

access control

availability

Confidentiality :

- The principle of confidentiality specifies that only the sender and the intended recipients should be able to access the contents of a message.
- Confidentiality gets compromised if an unauthorized person is able to access a message.
- Example of compromising confidentiality is if user of computer A sends message to user of computer B, another user C get access to this message which is not desired .
- This type of attack is called interception.
- Interception causes loss of message confidentiality.

Integrity :

➢ When the contents of a message are changed after the sender sends it, but before it reaches the intended recipient, we say that the integrity of the message is lost. It is shown in figure.

➢ For example, consider that user A sends message to user B. User C tampers with a message originally sent by user A, which is actually meant for user B. User C change its contents and send the changed message to user B. User B has no way of knowing that the contents of the message changed after user A had sent it. User A also does not know about this change. This type of attack is called modification.

➢ Modification causes of loss of message integrity.

Authentication :

- Authentication is the process of determining the true identity of someone.
- Authentication is also used in other ways -- not just for identifying users, but also for identifying devices and data messages.
- For example suppose user C send an electronic documents to user B ,the trouble is that user C had posed as user A .
- How would user B know that the message has come from user C who is posing user A.
  This type of attack is called Fabrication.
- Fabrication is possible in absence of proper authentication mechanisms

Non-repudiation :

Non repudiation is the assurance that someone cannot deny something.

Typically, non repudiation refers to the ability to ensure that a party to a contract or a communication cannot deny the authenticity of their signature on a document or the sending of a message that they originated.

You might send registered mail, for example, so the recipient cannot deny that a letter was delivered.

A legal document typically requires witnesses to signing so that the person who signs cannot deny having done so.

Non repudiation does not allow the sender of a message to refute the claim of not sending that message

Access Control :

Access control is a security technique that can be used to determine who should be able to access what.

Physical access control limits access to campuses, buildings, rooms and physical IT assets.

Logical access limits connections to computer networks, system files and data.

Access control systems perform authorization identification, authentication, access approval, and accountability of entities through login credentials including passwords, personal identification numbers (PINs), biometric scans, and physical or electronic keys.

Access control specifies and control who can access what.

Availability :

Availability of information refers to ensuring that authorized parties are able to access the information when needed.

Information only has value if the right people can access it at the right times.

Denying access to information has become a very common attack nowadays, For example, due to the intentional actions of an unauthorized user C, an authorized user A may not be able to contact a server computer B.

This would defeat the principle of availability. Such an attack is called as **interruption.**

Interruption puts the availability of resources in danger.

1.4 TYPES OF ATTACK:

The attack on computer system are grouped into following two categories

1. Passive attacks
2. Active attacks

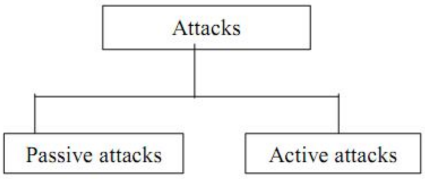


Fig. 1.1

a) Passive attacks:-

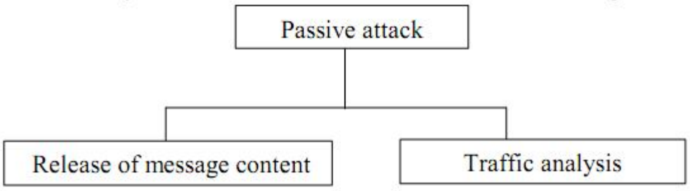
In passive attack, the attacker's goal is just to obtained information.

This means that the attack does not modify data or harm the system.

Passive attack do not involves any modifications to contents of an original message.

Passive attacks are difficult to detect.

Further the passive attacks are classified into two sub-categories.



(Types of passive attacks)

Fig. 1.2

Release of message content:-

Release of message content is quite simple to understand.

When we send a confidential email message to our friend, we desire that only she be able to access it.

Otherwise, the content of the message are released against our wishes to someone else.

Using certain security mechanisms, we can prevent release against contents.

Traffic analysis:-

If we had encryption protection, an attacker might still be able to observe the pattern of the messages . Such attempts of analyzing messages to come up with likely pattern are known as traffic analysis attacks.

Passive attacks are difficult to detect because they do not involves any alteration of the data.

b) Active attacks: -

Active attacks may change the data or harm the system. Attacks that threatens the integrity and availability are active attacks.

Active attacks are normally easier to detect than to prevent, because an attackers can launch them in a variety of ways.

In active attack, the content of the original message are modified in some way .

It is quite difficult to prevent active attacks absolutely, because to do so would require physical protection of all communication facilities and paths at all times. Instead, the goal is to detect them and to recover from any disruption or delays caused by them.

Active attacks are divided into three categories:

Masquerade –

One entity pretends to be a different entity.

In a masquerade attack, the intruder pretends to be a particular user of a system to gain access or to gain greater privileges than they are authorized for. A masquerade may be attempted through the use of stolen login IDs and passwords, through finding security gaps in programs or through bypassing the authentication mechanism.

Example user C might pose as user A and send a message to user B. User B might be led to believe that the message came from A.

Replay attack –

Involves passive capture of a data unit and its subsequent transmission to produce an unauthorized effect.

Example suppose user A wants to transfer some amount to user C .Both users A and C have accounts with bank B. User A might send an electronic message to bank B requesting for the fund transfer. User C could capture this message and send a second copy to bank B would have no idea that this is an unauthorised message and would treated as second and different. Therefore user C would get the benefit of the found transfer twice.

Alteration of messages–

Some portion of message is altered or the messages are delayed or recorded, to produce an unauthorized effect.

Suppose user A(Bob) send an electronic message transfer $1000 to D‟s account to bank B(alice). User C (Darth) might capture this message and change it to $10000.

Denial of service –

Denial of service (DOS) attacks make an attempt to prevent legitimate users from accessing some services, which they are eligible for. For instance, an unauthorized user send too many login requests to a sever using random user ids one after the other in quick succession, so as to flood the network and deny other legitimate users from using the network facilities.

Computer Security (Program that attacks):-

A few programs that attack computer system are:

Virus:-A virus is a computer program that attaches itself to another legitimate program and causes damage of the computer system or to the network. During the life time a virus goes through four phases

1. Dormant phases:-Here the virus is idle it gate activated on certain action or event
2. Propagation phase:-In this phase a virus copy itself and each copy start creating more copies of self .
3. Triggering phase:-A dormant virus move into this phase when the action event or event for which it was waiting is initiated.
4. Execution phase:-this is actual work of the virus.

Virus can be classified into

Parasitic virus

Memory resident virus

Boot sector virus

Stealth virus

Polymorphic virus

Metamorphic virus

Micro virus

Worm:- A worm does not perform any destructive action and instead only consumes system resources to bring it down. a virus modify a program but worm does not modify a program, it replicate itself again and again.

Trojan horse:-A Trojan horse allow an attacker to obtain some confidential information about a computer or network.

Applets and active AX:-Java applets and active AX control are small client side programs that might cause security problems , if used by attackers with a malicious intention .

Questions on Unit –I

1.Write down the principles of security and explain.

2. Explain the type of passive attacks.

3. Explain the type of active attacks.

4. What are the difference between active attack and passive attack.

UNIT-II

CRYPTOGRAPHY CONCEPTS

1.1 CRYPTOGRAPHY TECHNIQUES :

From the beginning any era, human being has two natural needs for communication:

(a) To communicate and share information 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.
- The word "cryptography‟ is the combination of two Greek words, "Krypto" meaning hidden or secret and "graphene" meaning writing.

Cryptography: It is the art of achieving security by encoding messages to make them non-readable format.

✓ The art or science encompassing the principles and methods of transforming an intelligible/understandable message into one that is unintelligible/non-understandable form, and then retransforming that message back to its original form.

Cryptanalysis: It is the technique of decoding messages from a non-readable format back to a readable format.

✓ It is done without knowing how they were initially converted from readable format to non-readable format.

✓ Also called code breaking.

Cryptology: is a combination of cryptography and cryptanalysis.

✓ Cryptography + Cryptanalysis = Cryptology.

Some Keywords:

Plain Text : Clear text or plain text signifies a message that can be understood by the sender, the recipient and also by anyone else who gets an access to that message.

Example: Hi Amit.

Cipher Text : When a plain text message is codified using any suitable scheme, the resulting message is called as cipher text.

Example: Plain Text: Hi Amit.

Cipher Text: Ki Dplw.

Encryption/ Encipher/Encrypt:

- Converting plaintext to cipher text.
- Encryption is the process of encoding a message or plain text so that cipher text can be produced.
- Plaintext is converted into cipher text by using encryption algorithm.
- Converting cipher text into plaintext.
- Decryption is the reverse process, transforming an encrypted message back into its normal text/plaintext.
- This is done by using decryption algorithm.

Cipher:

- Encryption and Decryption algorithms are together known as cipher.

Key:

- It is a number or set of numbers on which the cipher operates. Encryption Technique/ transforming a Plaintext into Ciphertext:
- Clear text, or plain text, signifies a message that can be understood by the sender, the recipient, and also by anyone else who gets access to that message. When a plain-text message is codified using any suitable scheme, the resulting message is called cipher text.

2.2 ENCODING TECHNIQUES:

- There are two primary ways in which a plain-text message can be codified/ transform to obtain the corresponding cipher text:

– Substitution technique

– Transposition technique.

1) **Substitution-cipher technique:**

In the substitution-cipher technique, each character of a plain-text message is replaced by other character, number or symbol.

There are several techniques. They are:

– Caesar Cipher

– Modified version of Caesar Cipher

– Mono-alphabetic Cipher

– Homophonic Substitution Cipher

– Polygram Substitution Cipher

– Polyalphabetic Cipher

i) <u>Caesar Cipher</u>

- Proposed by Julius Caesar.
- Mechanism to make a plaintext message into cipher text message.
- It replacing each letter of the alphabet with the letter standing n places further down the alphabet.
- Example: Replace each A with D, B with E, etc.

ABCDEFGHIJKLMNOPQRSTUVWXYZ

DEFGHIJKLMNOPQRSTUVWXYZABC

PT: KIIT

CT: NLLW

ii) ***<u>Modified Version of Caesar Cipher :</u>***

Modified version Caesar cipher is Caesar cipher but an alphabet A in plain text would not necessarily be replaced by D.

It can be replaced by an any valid alphabet, i .e. by E or by F or by G and so on .

Once replacement scheme is decided, it would be constant and will be used for all other alphabets in that message.

The English language contains 26 alphabets thus an alphabet A can be replaced by any alphabet in the English alphabet set (i.e. b to z) of course. it does not make sense to replace an alphabet by itself(means A is replaced by A) that means each alphabet has 25 possible of replacement.

A mechanism of encoding the message so that they can send securely is called cryptography.

Few terms are used in cryptography:-

Brute force attack:-

An attacks on a cipher text message, where they attacker attempt to use all possible permutation and combination is called as a brute force attack.

Cryptanalysis:-

The process of trying to break any text message to obtain the original plain text message itself is called cryptanalysis.

Cryptanalyst:-

The person attempting a cryptanalysis is called cryptanalyst.

iii) ***Mono-alphabetic Cipher:***

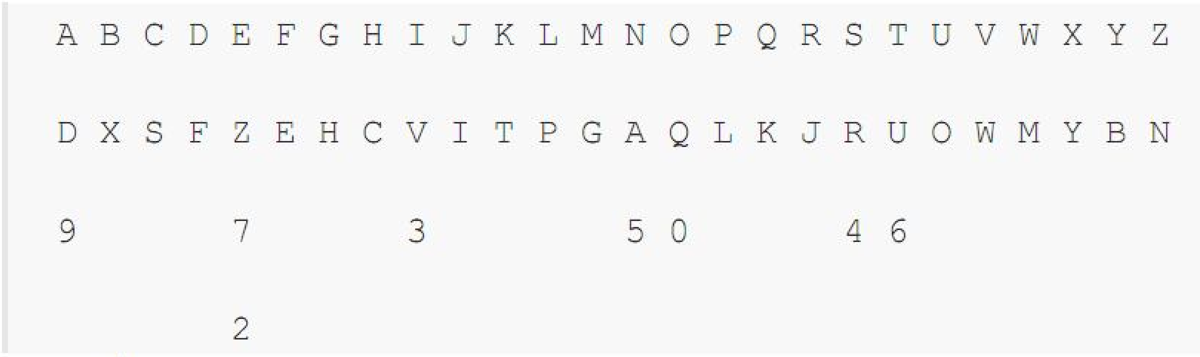
- A mono-alphabetic cipher is a substitution cipher where a symbol in the plaintext has a one-to-one relationship with a symbol in the cipher text.
- It means that a symbol in the plaintext is always replaced with the same symbol in the cipher text, irrespective of its position in the plaintext.
- It uses random substitution.
- This means that in a given plain-text message, each A can be replaced by any other alphabet (B through Z), each B can also be replaced by any other random alphabet (A or C through Z), and so on. The crucial difference being, there is no relation between the replacement of B and replacement of A. That is, if we have decided to replace each A with D, we need not necessarily replace each B with E—we can replace each B with any other character!
- To put it mathematically, we can now have any permutation or combination of the 26 alphabets, which means (26 x 25 x 24 x 23 x ... 2) or 4 x 1026

possibilities! This is extremely hard to crack.

iv) Homophonic Substitution Cipher

The Homophonic Substitution cipher is a substitution cipher in which single plaintext letters can be replaced by any of several different cipher text letters.

They are generally much more difficult to break than standard substitution ciphers. The number of characters each letter is replaced by is part of the key, e.g. the letter 'E' might be replaced by any of 4 different symbols(z,7,2,1), while the letter 'Q' may only be substituted by 1 symbol (k) as shown on Fig 2.4



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
D	X	S	F	Z	E	H	C	V	I	T	P	G	A	Q	L	K	J	R	U	O	W	M	Y	B	N
9				7				3					5	0				4	6						
				2																					

1

Example to encipher the message , DEFEND THE EAST WALL OF THE CASTLE, we find „D" in the top row, then replace it with the letter below it, „F". The second letter, „E" provides us with several choices; we could use any of „Z", „7", „2" or „1". We choose one of these at random, say „7". After continuing with this, we get the cipher text:

Plain text: DEFEND THE EAST WALL OF THE CASTLE

Cipher text: F7EZ5F UC2 1DR6 M9PP 0E 6CZ SD4UP1

v) *Polygram substitution :*

A simple substitution cipher substitutes for single plaintext letters. In contrast, polygram substitution ciphers involve groups of characters being substituted by other groups of characters.

In Polygram Substitution Cipher technique replaces one block of plain text with a block of cipher text-it does not work on character by character.

For example, HELLO can be replaced by YUQQW, but HELL could be replaced by a totally different cipher text block TEUI.

vi) *Poly-alphabetic Substitution Cipher :*

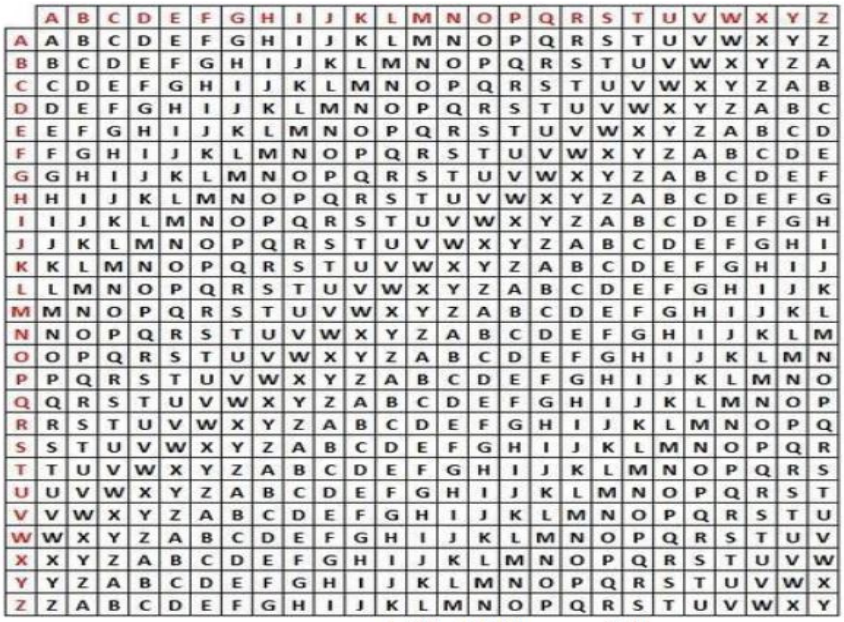
A poly-alphabetic substitution cipher involves the use of two or more cipher alphabets.

Instead of there being a one-to-one relationship between each letter and its substitute, there is a one-to-many relationship between each letter and its substitutes.

The Vigenere Cipher and Beaufort Cipher are example of Poly-alphabetic Substitution Cipher .

The Vigenere Table

The Vigenere Cipher, proposed by Blaise de Vigenere from the court of Henry III of France in the sixteenth century, is a poly-alphabetic substitution based on the following table.



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

Keyword:	RELAT IONSR ELATI ONSRE LATIO NSREL
Plaintext:	TOBEO RNOTT OBETH ATIST HEQUE STION
Cipher text:	KSMEH ZBBLK SMEMP OGAJX SEJCS FLZSY

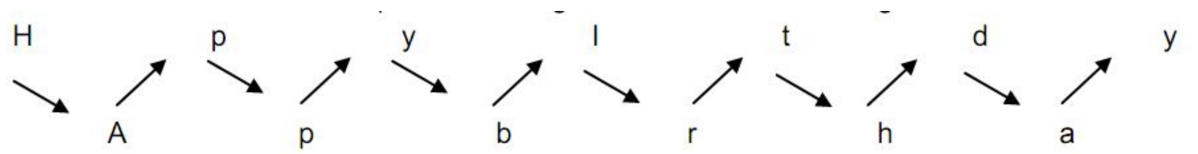
2.3 TRANSPOSITION TECHNIQUES :

As we have seen in substitution technique we replace each plain text with new alphabets in cipher text but in transposition, we don't replace one alphabet with another alphabet. We just rearrange the positions of plain text and get the cipher text. Techniques used in transposition:

i) *Rail fence Technique:*

It is a type of transposition technique which rotates the position of plain text message.

Example, suppose that we have a plain text message HAPPY BIRTHDAY we can convert this text in cipher text using rail fence as shown in following figure.



text: HPYITDYAPBRHA

Algorithm :

Arrange the plain text message in sequence of diagonals as shown above .

Read the text row by row and write it in sequence and thus we will get the cipher text.

ii) Simple columnar transposition technique:

It rotates the position of alphabets in plain text and then find out the cipher text.

Algorithm:

a) write the plain text message in a rectangle of pre defined size.

b) Read the message column by column in random order of columns.

c) The message obtained by doing so is the cipher text.

Example, suppose plain text that we have to encrypt is HAPPY BIRTHDAY. We can encrypt this as follows:

Consider a rectangle with four columns and write the plain text row by row.

Col1	col2	col3	col4
H	a	p	p
Y	b	I	r
T	h	d	a
y			

Now decide the order of columns as random order. Suppose order decided is 3,1,4,2 and read the text in this order.

Resulting text is the cipher text that is in this example cipher text is PIDHYTYPRAABH

iii) Simple columnar transposition technique with multiple rounds:

On improve the simple columnar transposition technique, we increase the complexity of this technique by implementing the same steps twice or thrice or depending upon the security of message.

Algorithm:

1. write the message row by row in a rectangle of pre defined size.
2. Read the message column by column in random order of columns.
3. The message thus obtained is cipher text.
4. Repeat steps a to c as many times as needed.

Example, consider the same PLAIN TEXT as above HAPPY BIRTHDAY.

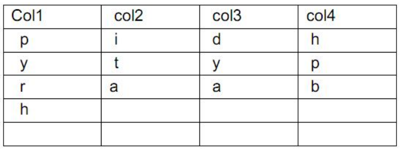
a) Consider a rectangle with four columns and write the plain text row by row as shown in table

Col1	col2	col3	col4
H	A	P	P
Y	B	I	R
T	H	D	A

b) Now decide the order of columns as random order. Suppose order decided is 3,1,4,2 and read the text in this order.

c) Resulting text is the cipher text that is in this example cipher text is PIDHYTYPRAABH

d)Perform step a to c once more.



Col1	col2	col3	col4
p	i	d	h
y	t	y	p
r	a	a	b
h			

iv) <u>*Vernam Cipher (One-Time-Pad)*</u>

The vernam Cipher, also called as One-Time Pad, is implemented using a random set of non-repeating characters as the input cipher text. The most significant point here is that once an input cipher text for transposition is used, it is never used again for any other message (hence the name one-time).

The length of the input cipher text is equal to the length of the original plain text. The algorithm used in Vernam Cipher

1. Treat each plain text alphabet as a number in an increasing sequence, i.e. A=0, B=1, ... Z=25.
2. Do the same for each character of the input cipher text.
3. Add each number corresponding to the plain text alphabet to the corresponding input cipher text alphabet number.
4. If the sum thus produced is greater than 26, subtract 26 from it.
5. Translate each number of the sum back to the corresponding alphabet. This gives the output cipher text.

Let us apply the Vernam Cipher algorithm to a plain text message HOW ARE YOU using a one-time pad NCBTZQARX to produce a cipher text message UQXTQUYFR.

It should be clear that since the one-time pad is discarded after a single use, this technique is highly secure and suitable for small plain text message, but is clearly impractical for large messages. The Vernam Cipher was first implemented at AT&T with the help of a device called as Vernam Machine.

Vernam Cipher uses a one-time pad, which is discarded after a single use and therefore, is suitable only for short messages.

2.4 ENCRYPTION & DECRYPTION

Encryption or Encoding or Encode:

➢ The process of converting or transforming plain text or original text into cipher text is called as encoding.

➢ This new form of the message is totally different from the initial message.

➢ It occurs at the sender’s side.

➢ The sender uses an encryption algorithm and a key to transform the original message into an encrypted message i.e., cipher text.

➢ Encryption is also called enciphering .

Decryption or Decoding :

➢ The process of converting cipher text into plain text is called as decoding.

➢ It occurs at the receiver’s end.

➢ The receiver uses decryption algorithms and a key to transform the cipher text back to original plaintext message.

➢ The decryption is also called deciphering or decipherment.

➢ Decryption is the reverse process of encryption.

Questions on Unit-II

1. What is substitution technique? Explain with example.
2. What is transposition technique ? Explain with example.
3. Write down the difference between encryption and decryption.

UNIT-III

SYMMETRIC & ASYMMETRIC KEY ALGORITHMS

ALGORITHM TYPES:

➢ It defines what size of plain text should be encrypted in each step of algorithm.

There are two aspects of algorithms: algorithm types and algorithms modes.

Algorithm Types:- An algorithms type defines what size of plain text should be encrypted in each step of algorithm. Based on this, algorithms are of two types:

Stream Cipher- In stream cipher the plain text is encrypted one byte at a time and the decryption happens one byte at a time.

Block Cipher-In block cipher the plain text is encrypted one block of text at a time and decryption also takes one block at a time.

ALGORITHM MODES:

The algorithm mode define the details of the cryptography algorithm, once the type is decided. An algorithm mode is combination of a series of the basic algorithm steps on the block cipher and some kind of feedback from the previous steps. There are 4 types of algorithm modes.

Electronic Code Book- ECB is the simplest mode of operation; the incoming plain text message is divided into blocks of 64 bits each. Each such block is then encrypted independently of the other blocks. For all blocks in a message, the same key is used for encryption.

Cipher Block Chaining- CBC mode ensures that even if a block of plain text repeats in the input, these two identical plain text yields totally different cipher text blocks in the output. For this a feedback mechanism is used.

Cipher Feedback- CFB mode encrypts data in units that's smaller e.g. they could be of size 8 bits than a defined block size.

Output Feedback- OFB mode is extremely similar to the CFB. The only difference is that is the case of CFB, the cipher text is fed into the next stage of encryption process. But in the case of OFB, the output of the Initial Vector (IV) encryption process is fed into the next stage of encryption process.

The important aspects of Encryption & Decryption process are:

Algorithm:

➢ The technique/ method used to encrypt or decrypt. Algorithm is generally not kept secret.

Key:

➢ A key is a character or a group of characters used to encrypt or decrypt the plain text. A key is generally kept secret.

Depending on what keys are used, there are two types of cryptography mechanisms/ types of cryptography:

Symmetric Key Cryptography:

➢ Symmetric key cryptography uses the same key for encryption and decryption.

Asymmetric Key Cryptography:

➢ Asymmetric key cryptography uses one key for encryption, and another different key for decryption.

3.1 OVERVIEW OF SYMMETRIC KEY CRYPTOGRAPHY

An encryption system in which the sender and receiver of a message share a single, common key that is used to encrypt and decrypt the message.

Symmetric-key systems are simpler and faster, but their main drawback is that the two parties must somehow exchange the key in a secure way.

With symmetric encryption, both parties use the same key for encryption and decryption purposes. Each user must possess the same key to send encrypted messages to each other.

The sender uses the key to encrypt their message, and then transmits it to the receiver. The receiver, who is in procession of the same key, uses it to decrypt the message.

The security of this encryption model relies on the end users to protect the secret key properly. If an unauthorized user were able to intercept the key, they would be able to read any encrypted messages sent by other users.

Conceptually it as similar to physical lock, perhaps a door lock. The same key is used to lock and unlock the door.

Some examples of symmetric-key algorithms include Data Encryption Standard (DES), double DES, triple DES, and Advanced Encryption Standard (AES).

Diffie-Hellman Key Exchange/Agreement Algorithm :

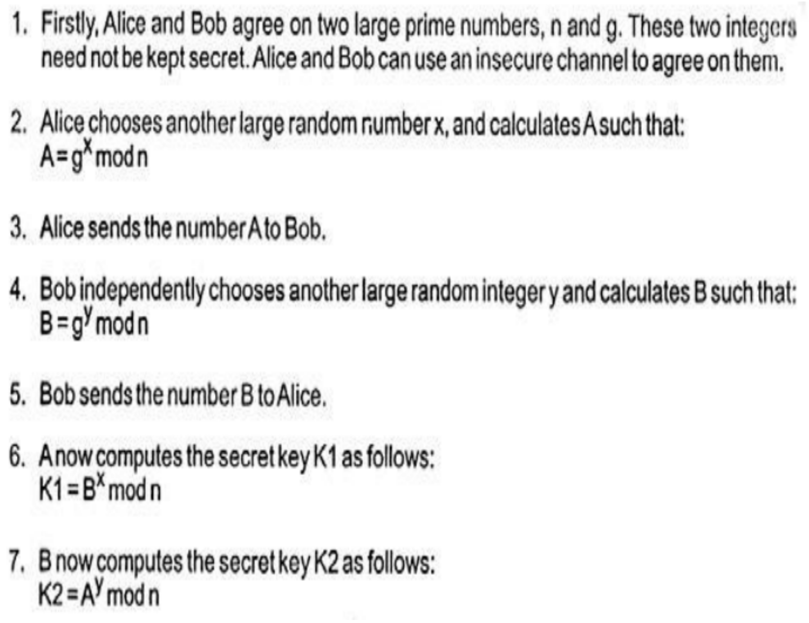
Whitefield Diffie and Martin Hellman devised an amazing solution to the problem of key agreement or key exchange in 1976.

This solution is called as the Diffie-Hellman Key Exchange/Agreement Algorithm.

The two parties, who want to communicate securely, can agree on a symmetric key using this technique.

This key can then be used for encryption / decryption. However, we must note that Diffie – Hellman key exchange algorithm can be used only for key agreement, but not for encryption or decryption of messages.

Description of the Algorithm: Let us assume that Alice and Bob want to agree upon a key to be used for encrypting / decrypting messages that would be exchanged between them. Then, the Diffie-Hellman key exchange algorithm works as follows.



1. Firstly, Alice and Bob agree on two large prime numbers, n and g. These two integers need not be kept secret. Alice and Bob can use an insecure channel to agree on them.
2. Alice chooses another large random number x, and calculates A such that:
   $A = g^x \bmod n$
3. Alice sends the number A to Bob.
4. Bob independently chooses another large random integer y and calculates B such that:
   $B = g^y \bmod n$
5. Bob sends the number B to Alice.
6. A now computes the secret key K1 as follows:
   $K1 = B^x \bmod n$
7. B now computes the secret key K2 as follows:
   $K2 = A^y \bmod n$

- Example of Algorithm

.

1. Firstly, Alice and Bob agree on two large prime numbers, n and g. These two integers need not be kept secret. Alice and Bob can use an insecure channel to agree on them.

   Let n = 11, g=7.

2. Alice choose another large random number x, and calculates A such that :
   $A = g^x \bmod n$

   Let x=3. Then, wwe have, $A=7^3 \bmod 11 = 343 \bmod 11$

   Let x=3. Then, wwe have, $A=7^3 \bmod 11 = 343 \bmod 11 = 2$

3. Alice sends the number A to Bob.

   Alice sends A=2 to Bob

4. Bob independently chooses another large random integer y and calculates B such that :
   $B = g^y \bmod n$

   Let y=6. Then we have,

   $B =7^6 \bmod 11 = 117649 \bmod 11 = 4$

5. Bob sends B=4 to Alice

6. A now computes the secret key K1 as follows :
   $K1 = B^x \bmod n$

   We have, $K1 = 4^3 \bmod 11 = 64 \bmod 11 =9$

7. B now computes the secret key K2 as follows :
   $K2 = A^y \bmod n$.

   We have, $K2 = 2^6 \bmod 11 = 64 \bmod 11 = 9$.

Asymmetric Key Operation:

The asymmetric-key encipherment also called public-key encipherment or public-key cryptography, was introduced by Diffie and Hellman in 1976 to overcome the problem found in symmetric key cryptography.

It uses two different keys for encryption and decryption.

Asymmetric Encryption is a form of Encryption where keys come in pairs. What one key encrypts, only the other can decrypt. In the sense that if key A encrypts a message, and then B can decrypt it, and if key B encrypts a message, then key A can decrypt it.

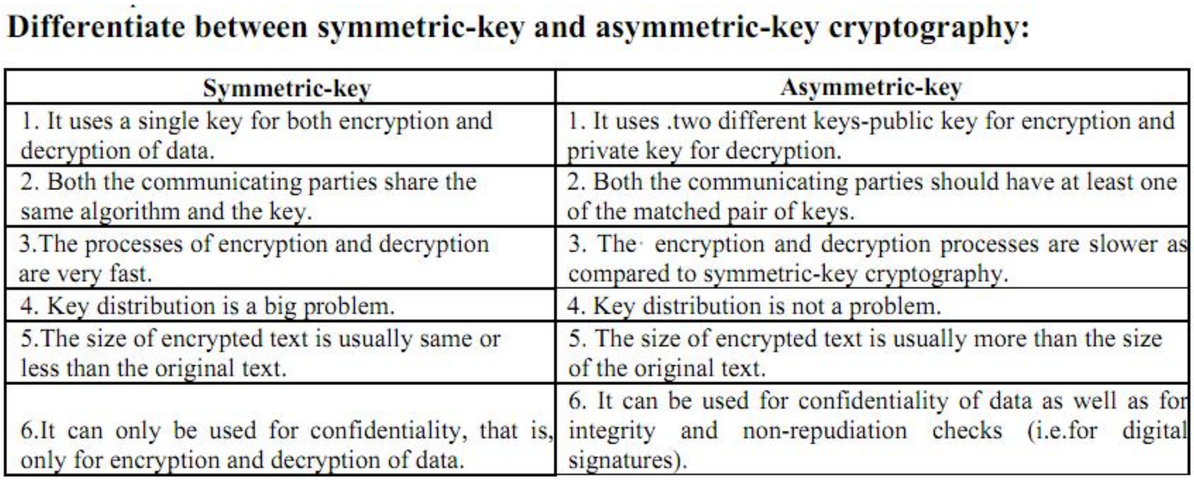
These two keys are referred to as the public key (used for encryption) and the private key (used for decryption).

Each authorized user has a pair of public and private keys. The public key of each user is known to everyone, whereas the private key is known to its owner only.

Asymmetric Encryption is also known as Public Key Cryptography, since users typically create a matching key pair, and make one public while keeping the other secret.

RSA is a well-known example of asymmetric-key algorithm.

The main advantage of public-key cryptography is that the sender and the receiver need not have to share the secret key. All communication involves only public keys.
Thus, the private key is never transmitted or shared. Anyone can send a confidential message using a public key, but the message can only be decrypted with a private key, which is kept by the intended recipient.



Differentiate between symmetric-key and asymmetric-key cryptography:

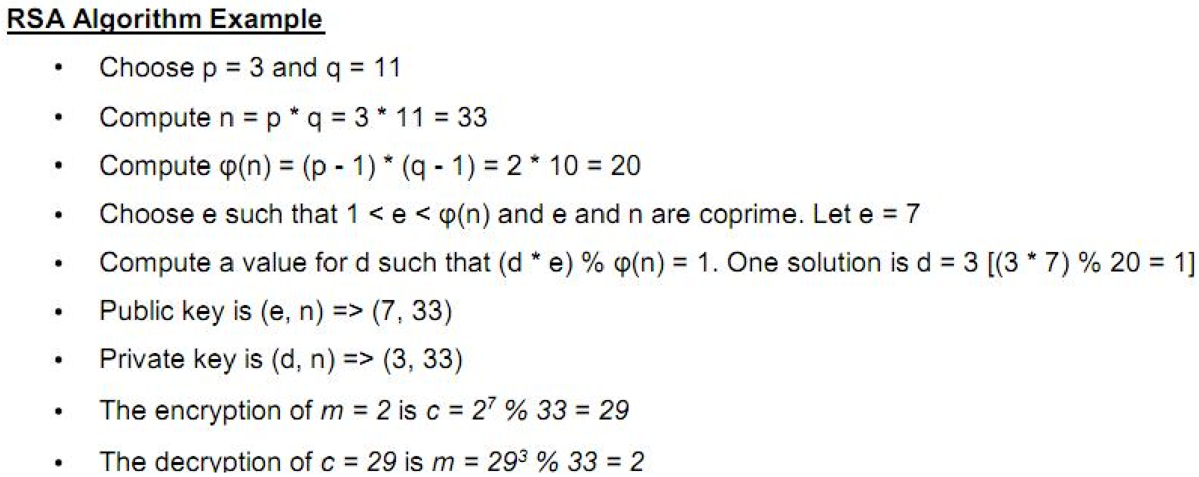
Symmetric-key	Asymmetric-key
1. It uses a single key for both encryption and decryption of data.	1. It uses .two different keys-public key for encryption and private key for decryption.
2. Both the communicating parties share the same algorithm and the key.	2. Both the communicating parties should have at least one of the matched pair of keys.
3.The processes of encryption and decryption are very fast.	3. The· encryption and decryption processes are slower as compared to symmetric-key cryptography.
4. Key distribution is a big problem.	4. Key distribution is not a problem.
5.The size of encrypted text is usually same or less than the original text.	5. The size of encrypted text is usually more than the size of the original text.
6.It can only be used for confidentiality, that is, only for encryption and decryption of data.	6. It can be used for confidentiality of data as well as for integrity and non-repudiation checks (i.e.for digital signatures).

3.5 THE RSA ALGORITHM

The RSA algorithm is named after Ron Rivest, Adi Shamir and Len Adleman, who invented it in 1977 [RIVE78].
The RSA cryptosystem is the most widely-used public key cryptography algorithm in the world. It can be used to encrypt a message without the need to exchange a secret key separately.
The RSA algorithm can be used for both public key encryption and digital signatures.
Its security is based on the difficulty of factoring large integers.
Party A can send an encrypted message to party B without any prior exchange of secret keys.
A just uses B's public key to encrypt the message and B decrypts it using the private key, which only he knows. RSA can also be used to sign a message, so A can sign a message using their private key and B can verify it using A's public key

RSA algorithm

1. Choose two large prime number P and Q .
2. Calculate N =P x Q .
3. Select the public key (i.e the encryption key) E such that it is not a factor of (P-1) and (Q-1).
4. Select the private key (i.e the decryption key) D such that the following equation is true:
   (D*E) mod (P-1) *(Q-1) = 1
5. For encryption, calculate the cipher text CT from the plain text PT as follows:
   CT=PTE mod N.
6. Send CT as the cipher text to the receiver.
7. For decryption, calculate the plain text PT from the cipher text CT as follows:
   PT=CTD mod N.



RSA Algorithm Example

- Choose p = 3 and q = 11
- Compute n = p * q = 3 * 11 = 33
- Compute φ(n) = (p - 1) * (q - 1) = 2 * 10 = 20
- Choose e such that 1 < e < φ(n) and e and n are coprime. Let e = 7
- Compute a value for d such that (d * e) % φ(n) = 1. One solution is d = 3 [(3 * 7) % 20 = 1]
- Public key is (e, n) => (7, 33)
- Private key is (d, n) => (3, 33)
- The encryption of $m = 2$ is $c = 2^7 \% 33 = 29$
- The decryption of $c = 29$ is $m = 29^3 \% 33 = 2$

3.3 DATA ENCRYPTION STANDARDS

The data encryption standard (DES) is a common standard for data encryption and a form of secret key cryptography (SKC), which uses only one key for encryption and decryption.

In 1972, the National Bureau of Standards (NBS) approached the Institute for Computer Sciences and Technology (ICST) to devise an encryption algorithm to secure, stored and transmitted data. The algorithm would be publicly available, but its key would be top secret.

The National Security Agency (NSA) assisted with the cryptographic algorithm evaluation processes, and in 1973, submission invitations were posted in the Federal Register.

In 1977, NBS issued the algorithm, i.e., DES.

DES WORKING PRINCIPLE

DES is a block cipher. It encrypts data in block of size 64 bits. That is 64 bits of plain text goes as the input to DES, which produce 64 bits of cipher text. The same algorithm and Key are used for encryption and decryption.

Actually the initial key consists of 64 bits. However before DES process even starts every eight bit of the key is discarded to produce 56 key. The bit position 8,16,24,32,56,64 are discarded.

DES is based on the two fundamental attributes, Substitution (also called as confusion) And transposition (also called as Diffusion).

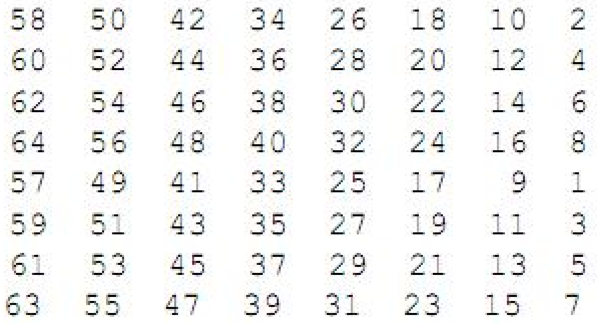
Steps of DES

1. The 64-bit plain text block is handed over to an initial Permutation (IP) function.
2. The Initial Permutation produces 2 halves of permuted block .let Left Plain Text (LPT) and Right Plain Text (RPT).
3. Each LPT and RPT go through 16 rounds of encryption process.
4. In the end LPT and RPT are rejoined and Final Permutation (FP) is performed on the combined block.
5. The result of the process 64 bit cipher text.

Initial Permutation

The 64 bits of the input block to be enciphered are first subjected to the following permutation, called the initial permutation IP:

IP



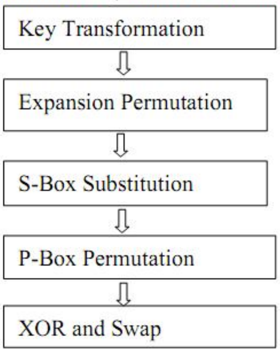
58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

That is the permuted input has bit 58 of the input as its first bit, bit 50 as its second bit, and so on with bit 7 as its last bit. The computation which uses the permuted input block as its input to produce the pre-output block consists, but for a final interchange of blocks, of 16 iterations of a calculation that is described below in terms of the cipher function f which operates on two blocks, one of 32 bits and one of 48 bits, and produces a block of 32 bits.

Let the 64 bits of the input block to an iteration consist of a 32 bit block LPT followed by a 32 bit block RPT. Using the notation defined in the introduction, the input block is then LR.

Rounds

Each of the 16 rounds in turn , consists of the following steps



Key Transformation

Expansion Permutation

S-Box Substitution

P-Box Permutation

XOR and Swap

Key Transformation

Let K be a block of 48 bits chosen from the 56-bit key. A different 48 bit sub key is generated during each round using a process called as Key transformation. For this the 56-bit key is divided into 2 halves each of 28 bits. These halves are circularly shifted left by one or two positions, depending on the round. For example, if the round number is 1, 2, 9 or 16, the shift is done by only positions. For other rounds, the circular shift is done by two positions.

Expansion Permutation

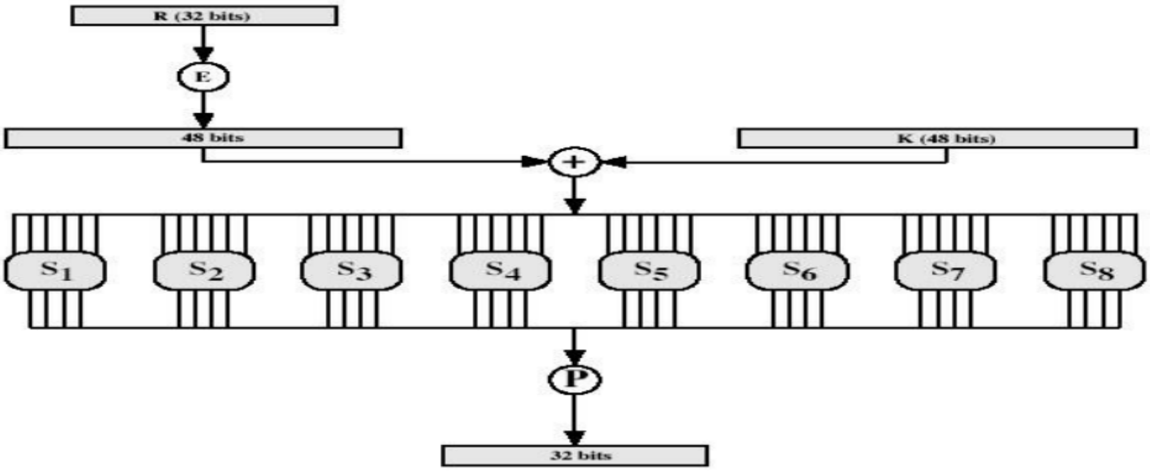
We had two 32 bit plain text called LPT and RPT. During expansion permutation the RPT is expanded from 32 bits to 48 bits. This happens as follows The 32-bit RPT is divided into 8 blocks, having 4 bits in each block.

Each 4 bit blocks of the previous step is then expanded to a corresponding 6 blocks, 2 more bits are added they are actually the repeated first and fourth of the 4 bit block.

The Key transformation processes compress the 56 bit key to 48 bits. Then the Expansion permutation process expands the 32 bit RPT to 48 bit. Now the 48 bit key and 48 bit RPT is XORed and the resulting is given to the S-box Substitution.

S-Box Substitution:

It is the process that accepts 48-bit input from the XOR operations and produces a 32 bit output using substitution technique. The Substitution is performed by 8 substitution boxes called as S-boxes. Each of 8 boxes has a 6 –bit input and 4 bit output.



P-Box Permutation

The out put of S-box consists of 32 bits. these 32 bits are permuted using a P-box.

XOR and Swap

The 32 bit RPT and 32 bit LPT is XORed and swap means the LPT becomes and RPT and vice versa.

Final Permutation

At the end of the 16 rounds the final permutation is performed (only once).

The nature of DES algorithm: of more concern is that cryptanalysis is possible by exploiting the characteristics of DES. The focus is the eight S-boxes used in each iteration.

The design criteria for the complete algorithm has never been published and there has been speculation that the boxes were constructed in such a way that cryptanalysis is possible by an opponent who knows the weakness in the S-boxes. Although this has not been established, the US governments' "clipper project" raises many questions. These are the main reasons DES is now being replaced by the AES standard.

Using a brute-force attack by simply searching for a key is possible. However, for 56-bit key, there are 256 possible key combinations, if we could search one key in 1 µs, then we need 2283 years to try all keys. (Distributed.net broke a DES-56 within 22 hours and 15 minutes, by using 100,000 PCs).

DES decryption

The decryption process with DES is essentially the same as the encryption process and is as follows:

Use the cipher text as the input to the DES algorithm but use the keys K in reverse order.

That is, use K16 on the first iteration, K15 on the second until K1Which is used on the 16th and last iteration .

Variation of DES:

In spite of its strength it is felt that with the tremendous advance in computer hard ware, DES is susceptible to possible attack. However because DES is already proven to be a very competent algorithm, it would be nice to reuse DES by making it stronger by some means, rather than writing a new cryptography algorithm. Two main variation of DES are Double DES and Triple DES.

Double DES

Double DES is quite simple ,it does twice what DES normally does only once. Double DES uses two keys K1 and K2. The final output is encryption of encrypted text.

Triple DES

Triple DES means DES three times. It comes in 2 flavour : one that uses 3 keys and second that uses 2 keys.

Triple DES with 3 keys – the plain text encrypted with K1 , then encrypted with K2,and finally with K3.where K1 ,K2, K3 are all different from each other.

Triple DES with 2 keys – the plain text encrypted with K1 , then encrypted with K2,and finally with K1.where K1 ,K2, are used.

3.7 DIGITAL SIGNATURE

Digital signature:

➢ It is an authentication mechanism that allows the sender to attach an electronic code with the message. This electronic code acts as the signature of the sender and hence, is named digital signature.

➢ It is done to ensure its authenticity and integrity.

➢ Digital signature uses the public-key cryptography technique. The sender uses his or her private key and a signing algorithm to create a digital signature and the signed document can be made public.

The receiver, uses the public key of the sender and a verifying algorithm to verify the digital signature.

➢ A normal message authentication scheme protects the two communicating parties against attacks from a third party (intruder). However, a secure digital signature scheme protects the two parties against each other also.

➢ Suppose A wants to send a signed message (message with A's digital signature) to B through a network. For this, A encrypts the message using his or her private key, which results in a signed message. The signed message is then sent through the network to B.

➢ Now, B attempts to decrypt the received message using A's public key in order to verify that the received message has really come from A.

➢ If the message gets decrypted, B can believe that the message is from A. However, if the message or the digital signature has been modified during transmission, it cannot be decrypted using A's public key. From this, B can conclude that either the message transmission has tampered with, or that the message has not been generated by A.

Message integrity:

➢ Digital signatures also provide message integrity.

➢ If a message has a digital signature, then any change in the message after the signature is attached will invalidate the signature.

➢ That is, it is not possible to get the same signature if the message is changed. Moreover, there is no efficient way to modify a message and its signature such that a new message with a valid signature is produced.

Non-repudiation:

➢ Digital signatures also ensure non-repudiation.

➢ For example, if A has sent a signed message to B, then in future A cannot deny about the sending of the message. B can keep a copy of the message along with A's signature.

➢ In case A denies, B can use A’s public key to generate the original message. If the newly created message is the same as that initially sent by A, it is proved that the message has been sent by A only.

➢ In the same way, B can never create a forged message bearing A's digital signature, because only A can create his or her digital signatures with the help of that private key.

Message confidentiality:

➢ Digital signatures do not provide message confidentiality, because anyone knowing the sender's public key can decrypt the message.

Digital signature process:

The digital signature process is shown in Figure. Suppose user A wants to send a signed message to B through a network. To achieve this communication, these steps are followed:

➢ A uses his private key (EA), applied to a signing algorithm, to sign the message (M).

➢ The message (M) along with A's digital signature (S) is sent to B.

➢ On receiving the message (M) and the signature (S), B uses A's public key (DA), applied to the verifying algorithm, to verify the authenticity of the message. If the message is authentic, B accepts the message, otherwise it is rejected.

Questions on Unit- III

1. What are the algorithm modes?
2. Explain Diffie-Hellman algorithm with example.
3. Explain RSA algorithm with example.
4. What is the difference between symmetric key cryptography and asymmetric key cryptography?
5. Write a short note on digital signature.

UNIT - IV

DIGITAL CERTIFICATE & PUBLIC KEY INFRASTRUCTURE

Digital Certificate

- To solve the man-in-the-middle attack, Digital Certificates were introduced.
- A digital certificate is simply a small computer file. For example, my digital certificate would actually be a computer file with a file name such as name .cer.
- The digital certificate is actually quite similar to a passport. As we know every passport has a unique passport number, similarly every digital certificate has a unique serial number. Also gives information of the issuer's name, serial number, public key, validity period, etc.
- Digital Certificate is issued by a trusted agency called as CA (Certification Authority).
- Another third party called as RA (Registration Authority) acts as a intermediate entity between CA and end user.
- Satisfies the principle of Authentication, non-repudiation.

Who can be a CA?

- CA has to be someone, who everybody trusts. Consequently, the governments in various countries decide who can and who cannot be a CA.
- Usually, a CA is a reputed organization, such as a post office, financial institution, software company, etc. Two of the world's most famous CAs are VeriSign and Entrust.
- Safes crypt Limited is the first Indian CA.
- Thus, a CA has the authority to issue digital certificates to individuals and organizations, who want to use those certificates in asymmetric-key cryptographic applications.

Technical Details of a Digital Certificate

A standard called X.509 defines the structure of a digital certificate. The International Telecommunication Union (ITU) designs this standard. At that time, it was a part of another standard called X.500. The current version of the standard is Version 3, called X.509V3.

Contents of Digital Certificate:

Version: Version of X.509 protocol. Version can be 1,2 or 3

Certificate Serial No.: Contains unique integer which is generated by CA

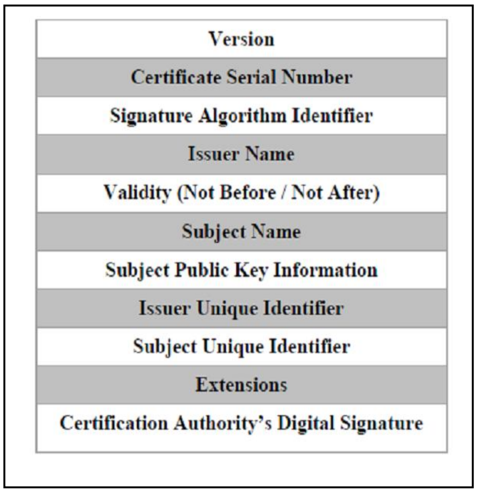
Signature Algorithm Identifier: Identifies the algorithm used by CA to sign the certificate.

Issuer Name: Identifies the Distinguished Name that created & signed the certificate

Validity: (not before/not after) Contains two date-time values. This value generally specifies the date & time up to seconds or milliseconds.

Subject name: Distinguished Name of the end user (user or organization)

Subject Public key info.: This field can never be blank. Contains public key & algorithm related.



Version
Certificate Serial Number
Signature Algorithm Identifier
Issuer Name
Validity (Not Before / Not After)
Subject Name
Subject Public Key Information
Issuer Unique Identifier
Subject Unique Identifier
Extensions
Certification Authority's Digital Signature

Issuer Unique Identifier: Helps identify a CA uniquely if two or more CAs have used the same Issuer Name over time.

Subject Unique Identifier: Helps identify a subject uniquely if two or more subjects have used the same Subject Name over time.

Digital-Certificate Creation:

1. Parties Involved

- end user (may be a single user or organization),
- issuer (CA),
- third party is also (optionally) called a Registration Authority (RA), involved in the ertificate creation and management.

The RA commonly provides the following services

Accepting and verifying registration information about new users.

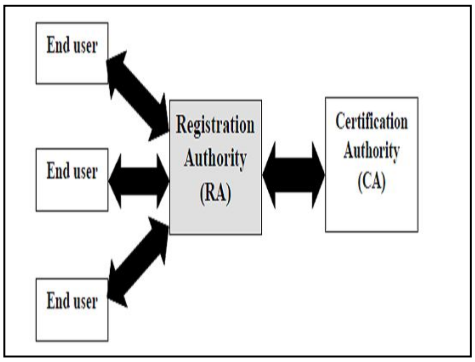
Generating keys on behalf of the end users.

Accepting and authorizing requests for key backups and recovery.

Accepting and authorizing the requests for certificate revocation.

RA is mainly set up for facilitating the interaction between the end users and the CA.

The RA cannot issue digital certificates.



The CA must handle this. Additionally, after a certificate is issued, the CA is responsible for all the certificate management aspects, such as tracking its status, issuing revocation notices if the certificate needs to be invalidated for some reason, etc.

2. Certificate Creation Steps

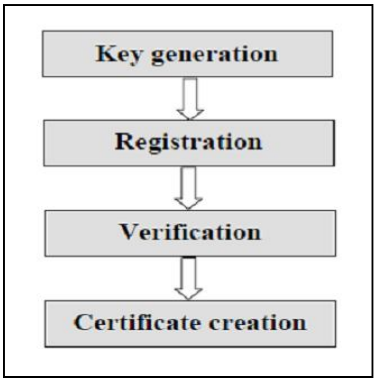
Step 1: Key Generation:

The action begins with the subject (i.e. the user/organization) who wants to obtain a certificate.

There are two different approaches for this purpose:

Firstly, the subject can create a private key and public key pair using some software.

The subject must keep the private key which is generated, keep it secret. The subject then sends the public key along with other information to the RA.



Secondly, the RA can generate a key pair on-behalf the subject.

This can happen in cases where either the user is not aware of the technicalities involved in the generation of a key pair.

The RA sends the private key which is generated, to the subject. The RA keeps the public key.

Step 2: Registration:

This step is required only if the user generates the key pair in the first step. If the RA generates the key pair on the user's behalf, this step will also be a part of the first step itself.

Assuming that the user has generated the key pair, the user now sends the public key and the associated registration information (e.g. subject name, as it is desired to appear in the digital certificate) and all the required evidence about himself/herself to the RA.

For this, the software provides a wizard in which the user enters all the data then submits it. This data then travels over the network/Internet to the RA. This format for the certificate requests has been is called Certificate Signing Request (CSR). This is one of the Public Key Cryptography Standards (PKCS).

Note that the user must not send the private key to the RA—the user must keep it securely.

Step 3: Verification:

After the registration process is complete, the RA has to verify the user's credentials.

This verification is in two respects, as follows.

1. Firstly, the RA needs to verify the user's credentials which are provided by the user.
   - If the user were actually an organization then the RA would perhaps like to check the business records, historical documents and credibility proofs.
   - If it is an individual user then simpler checks are in call, such as verifying the postal address, email id, phone number, passport or driving-license details can be sufficient.
2. Secondly, check is to ensure that the user who is requesting for the certificate, whether he/she possesses the private key or not corresponding to the public key that is sent to the RA.

This is very important, because there must be a record that the user possesses the private key corresponding to the given public key. Otherwise, this can create legal problems. This check is called the Proof Of Possession (POP) of the private key.

How can the RA perform this check? There are many approaches to this, the chief ones being as follows.

- The RA can demand that the user must digitally sign his/her Certificate Signing Request (CSR) using his/her private key. If the RA can verify the signature (i.e. de-sign the CSR) correctly using the public key of the user, the RA can believe that the user indeed possesses the private key.
- Alternatively, the RA can create a random number challenge; encrypt it with the user's public key and send the encrypted challenge to the user. If the user can successfully decrypt the challenge using his/her private key, the RA can assume that the user possesses the right private key.
- Thirdly, the RA can actually generate a dummy certificate for the user, encrypt it using the user's public key and send it to the user. The user can decrypt it only if he/she can decrypt the encrypted certificate, and obtain the plain-text certificate.

Step 4: Certificate Creation:

Assuming that all the steps so far have been successfully done, and then RA passes on all the details of the user to the CA.

The CA does its own verification (if required) and creates a digital certificate for the user.

The creation of certificate as per the X.509 standard.

The CA sends the certificate to the user, and also retains a copy of the certificate for its own record.

The CA's copy of the certificate is maintained in a certificate directory. This is a central storage location maintained by the CA.

4.2 Private key management

<u>Protecting private key</u>

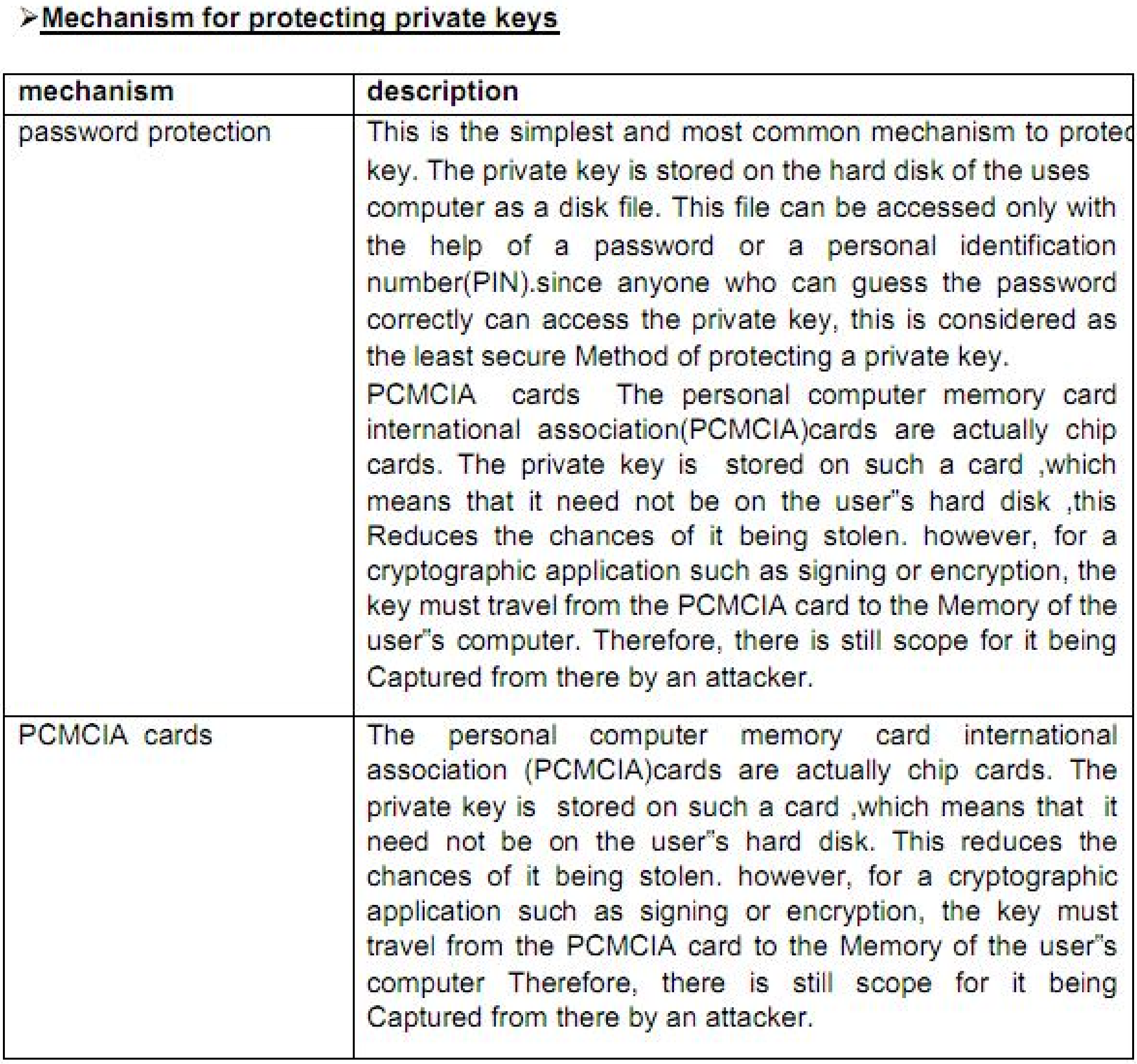
In many situations, the private key of the user might be required to be transported from one location to another. For instance, suppose that the user wants to change her PC. To handle these situations, there is a cryptography standard by the name PKCS#12.This allows a user to export her digital certificate and private key in the form of a computer file.

Obviously, the certificate and the private key must be protected as they are moved to another location. for this ,the PKCS#12 standard ensures that they are encrypted using a symmetric key which is derived from the user's private key protection password.

Multiple key pairs

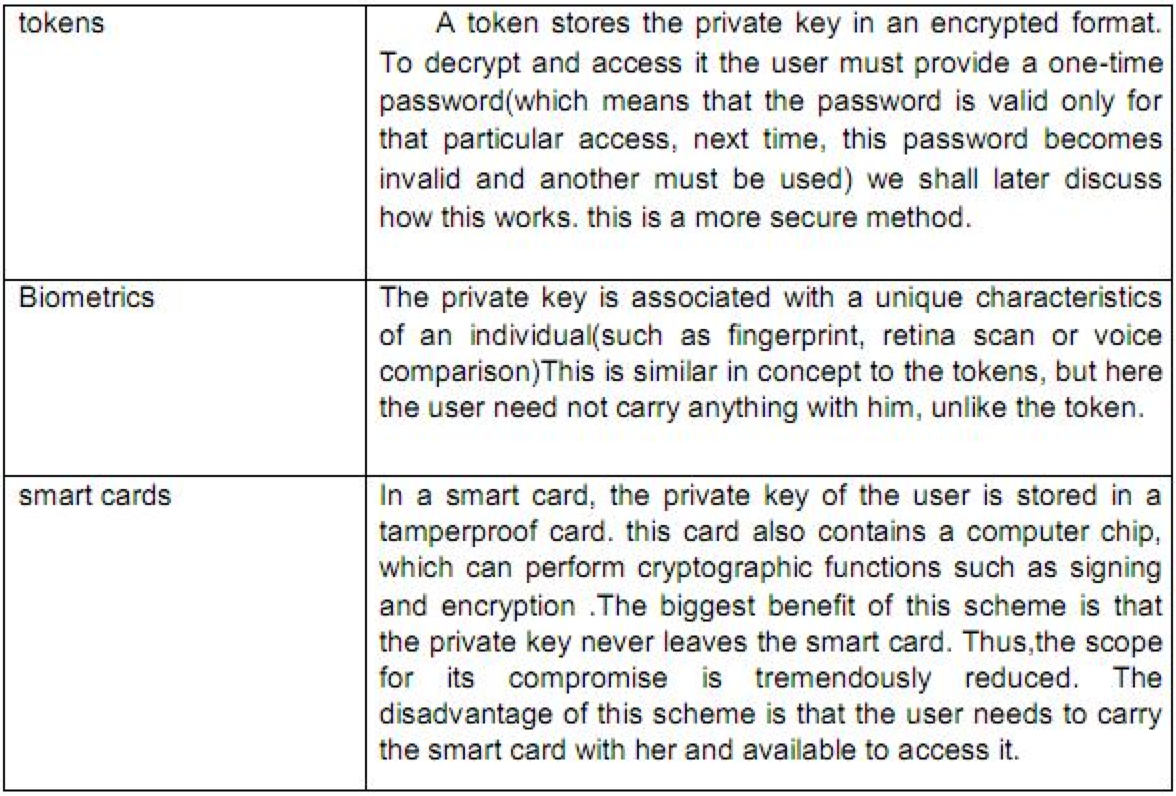
The PKI approach also recommends that in serious business applications, user should posses multiple digital certificates, which also means multiple key pairs. The need for this is that one certificate could be strictly used for signing anther for encryption. This ensures that the loss of one of the private keys does not affect the complete operations of the user. The following guidelines are generally helpful:

The private key that is used for digital signing(non-repudiation)must be backed up or archived after it expires. it must be destroyed. This ensures that it is not used by someone else for signing on behalf of the person at a future date(although chances are that this will be detected by CRL/OCSP checks or certificate expiry date checks, you cannot say this with a 100% guarantee).



➢**Mechanism for protecting private keys**

mechanism	description
password protection	This is the simplest and most common mechanism to protec key. The private key is stored on the hard disk of the uses computer as a disk file. This file can be accessed only with the help of a password or a personal identification number(PIN).since anyone who can guess the password correctly can access the private key, this is considered as the least secure Method of protecting a private key. PCMCIA cards The personal computer memory card international association(PCMCIA)cards are actually chip cards. The private key is stored on such a card ,which means that it need not be on the user"s hard disk ,this Reduces the chances of it being stolen. however, for a cryptographic application such as signing or encryption, the key must travel from the PCMCIA card to the Memory of the user"s computer. Therefore, there is still scope for it being Captured from there by an attacker.
PCMCIA cards	The personal computer memory card international association (PCMCIA)cards are actually chip cards. The private key is stored on such a card ,which means that it need not be on the user"s hard disk. This reduces the chances of it being stolen. however, for a cryptographic application such as signing or encryption, the key must travel from the PCMCIA card to the Memory of the user"s computer Therefore, there is still scope for it being Captured from there by an attacker.



tokens	A token stores the private key in an encrypted format. To decrypt and access it the user must provide a one-time password(which means that the password is valid only for that particular access, next time, this password becomes invalid and another must be used) we shall later discuss how this works. this is a more secure method.
Biometrics	The private key is associated with a unique characteristics of an individual(such as fingerprint, retina scan or voice comparison)This is similar in concept to the tokens, but here the user need not carry anything with him, unlike the token.
smart cards	In a smart card, the private key of the user is stored in a tamperproof card. this card also contains a computer chip, which can perform cryptographic functions such as signing and encryption .The biggest benefit of this scheme is that the private key never leaves the smart card. Thus,the scope for its compromise is tremendously reduced. The disadvantage of this scheme is that the user needs to carry the smart card with her and available to access it.

KEY UPDATE

Good security practices demand that the key pairs should be updated periodically. This is because over time, keys become susceptible to analysis attacks. Causing a digital certificate to expire after a certain date ensures this. This requires an update to the key pair. The expiry of a certificate can be dealt with in one of the two following ways:

- The CA reissues a new certificate based on the original key pair (of course, this is not recommended unless there is an all-around confidence in the strength of the original key pair).
- A fresh key pair is generated and the CA issues a new certificate based on that the new pair. The key update process itself can be handled in two ways, as follows:
- In the first approach, the end user has to detect that certificate is about to expire and request the CA to issue a new one.
- In the other approach, the expiry date of the certificate is automatically checked every time it is used and as soon as it is about to expire, its renewal request is sent to the CA. For his, special systems need to be in place.

KEY ARCHIVAL

The CA must plan for and maintain the history of the certificate and the keys of its users. For instance, suppose that someone approaches the CA of Alice, requesting the CA to make Alice's digital certificate available, as was used three years back to sign a legal document for verification purposes. If the CA has not archived the certificates, how can the CA provide this information? This can cause serious legal problems. Therefore, key archival is a very significant aspect of any PKI solution.

4.3 PKIX Model :

Management protocols are the protocols that are required to support on–line interactions between PKI user and management entities. The possible set of functions that can be supported by management protocols is registration of entity, that takes place prior to issuing the certificate . initialisation, for example generation of key–pair.

- certification, the issuance of the certificate .
- key–pair recovery, the ability to recover lost keys .
- key–pair update, when the certificate expires and a new key–pair and certificate have to be generated.
- revocation request, when an authorised person advices the CA to include a specific certificate into the revocation list cross-certification, when two CAs exchange information in order to generate a cross– certificate .

The Certificate Policies and the Certificate Practice Statements are recommendations of documents that will describe the obligations and other rules with regard the usage of the Certificate.

PKIX Architectural Model:

PKIX is working on the following five areas.

Profiles of X.509 v3 Certificate & v2 Certificate Revocation List profiles:

Lists the use of various options while describing extensions of a digital certificate.

Operational Protocol:

Defines the underlying protocols that provide the transport mechanism.

Management Protocol:

Enables exchange of information between the various PKI entities and specifies the structure & details of PKI messages.

Policy outlines:

Defines policies for the creation of Certificate Policies & Certificate Practice Statements.

Timestamp & Data Certification Services:

Both are the trusted third parties that provide services to guarantee the existence of certificate & DCS verifies the correctness of data that it receives.

4.4 Public key cryptography standards

In cryptography, PKCS is a group of public-key cryptography standards devised and published by RSA Security Inc, starting in the early 1990s. The company published the standards to promote the use of the cryptography techniques to which they had patents, such as the RSA algorithm, the Schnorr signature algorithm and several others. Though not industry standards(because the company retained control over them), some of the standards in recent years[when?] have begun to move into the "standards-track" processes of relevant standards organizations such as the IETF and the PKIX working-group.

Questions:

No. 1 Explain PKIX model in detail.

2. What is digital certificate(DC) ?Explain the procedure for creation of DC.

3. What is digital certificate and write different steps used in obtaining a digital certificate?

UNIT-5

INTERNET SECURITY PROTOCOLS

5.1 BASIC CONCEPT

Static Web Page:

A web page created by an application developer and stored on a web server.

Whenever any user request for that page, the web server sends back the same page without performing any additional processing.

All it does is to locate that page on its hard disk and add HTTP headers, and Send back an HTTP response.

Thus the contents of the Static web page do not change depending upon the request.

Dynamic Web Page :

The contents of a dynamic web page can vary depending on a number of parameters.

When a user request for a dynamic web page, the web server cannot simply send back an HTML page as in case of a static web page.

Here the web server actually invokes a program that resides on its hard disk. The program might run database perform transaction processing etc. then send the HTTP response back to the web server.

Dynamic web pages involve server side programming.

Active Web Page:

With the arrival of the programming language java, active web page became quite popular.

When a client sends an HTTP request for an active web page, the web page server sends back an HTTP response that contains an HTML page as usual.

In addition, the HTML page also contains a small program that executes on the client computer inside the web browser.

TCP/IP protocol suite:

Communications between computers on a network is done through protocol suits. The most widely used and most widely available protocol suite is TCP/IP protocol suite. Each layer of the TCP/IP has a particular function to perform and each layer is completely separate from the layer(s) next to it.

The communication process that takes place, at its simplest between two computers, is that the data moves from layer 5 to 3 to 2 then to 1 and the information sent arrives at the second system and moves from 1 to 2 to 3 to 4 and then finally to layer 5. The 5 layers are as follows :-

1. Application layer
2. Transport layer
3. Network layer
4. Data link layer
5. Physical Layer

1. Application layer

This is the top layer of TCP/IP protocol suite. This layer includes applications or processes that use transport layer protocols to deliver the data to destination computers.

At each layer there are certain protocol options to carry out the task designated to that particular layer. So, application layer also has various protocols that applications use to communicate with the second layer, the transport layer. Some of the popular application layer protocols are :

HTTP (Hypertext transfer protocol)
FTP (File transfer protocol)
SMTP (Simple mail transfer protocol)
SNMP (Simple network management protocol) etc

2. Transport Layer:

This layer provides backbone to data flow between two hosts. This layer receives data from the application layer above it. There are many protocols that work at this layer but the two most commonly used protocols at transport layer are TCP and UDP.

TCP is used where a reliable connection is required while UDP is used in case of unreliable connections.

3. Network Layer :

This layer is also known as Internet layer.

The main purpose of this layer is to organize or handle the movement of data on network. By movement of data, we generally mean routing of data over the network.

The main protocol used at this layer is IP.

While ICMP (used by popular "ping" command) and IGMP are also used at this layer.

4. Data Link Layer :

This layer is also known as network interface layer.

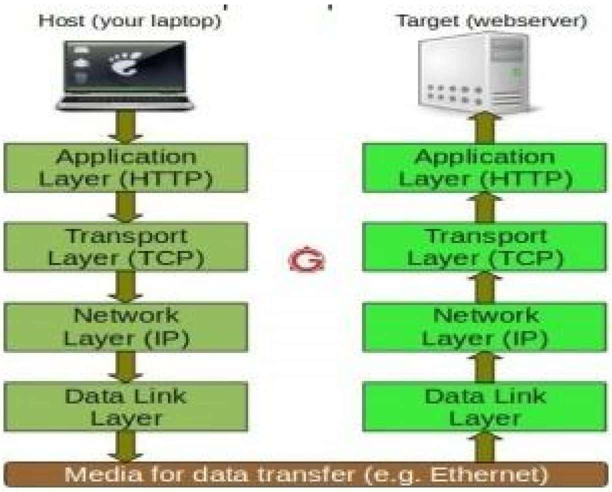
It provides error control and framing.

Some of the famous protocols that are used at this layer include ARP (Address resolution protocol), PPP(Point to point protocol) etc.

5. Physical Layer:

This layer specifies the characteristics of the hardware to be used in the network.

For example it specifies the characteristic of communication media.



Secure socket layer

The Secure Sockets Layer (SSL) is a computer networking protocol that manages server authentication, client authentication and encrypted communication between servers and clients.

SSL uses a combination of public-key and symmetric-key encryption to secure a connection between two machines, typically a Web or mail server and a client machine, communicating over the Internet or an internal network.

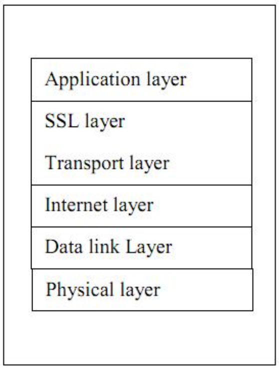
Using the OSI reference model as context, SSL runs above the TCP/IP protocol, which is responsible for the transport and routing of data over a network, and below higher-level protocols such as HTTP and IMAP, encrypting the data of network connections in the application layer of the Internet Protocol suite.

The Transport Layer Security (TLS)protocol evolved from SSL and has largely superseded it, although the terms SSL or SSL/TLS are still commonly used; SSL is often used to refer to what is actually TLS.

The combination of SSL/TLS is the most widely deployed security protocol used today and is found in applications such as Web browsers, email and basically any situation where data needs to be securely exchanged over a network, like file transfers, VPN connections, instant messaging and voice over IP.

The Position of SSL in TCP/IP Protocol Suite

SSL can be conceptually considered as an additional layer in the TCP/IP protocol suite.
The SSL layer located between the application layer and transport layer.



How ssl works

As mentioned, the Secure Sockets Layer (SSL) is a method for providing security for web based applications.

It is designed to make use of TCP to provide a reliable end to-end secure service. SSL has 3 sub protocols

1. Hand-shake Protocol.
2. Record Protocol.
3. Alert Protocol.

Hand-shake Protocol :

This is the most complex part of SSL and allows the server and client to authenticate each other and to negotiate an encryption and MAC algorithm and cryptographic keys to be used to protect data sent in an SSL record.

This protocol is used before any application data is sent. It consists of a series of messages exchanged by the client and server. Each message has three fields:

Type : Indicates one of 10 messages such as “hello request”

Length : The length of the message in bytes.

Content : The parameters associated with this message such as version of SSL being used.

The Handshake Protocol consists of four phases:

1. This phase consists of the client hello and server hello messages which contain the following

- Version: The highest SSL version understood by client SSL record protocol payload.
- Random: 32-bit timestamp and 28 byte nonce.
- Session ID: A variable length session identifier.
- Cipher Suite: List of crypto algorithms supported by client in decreasing order of preference.

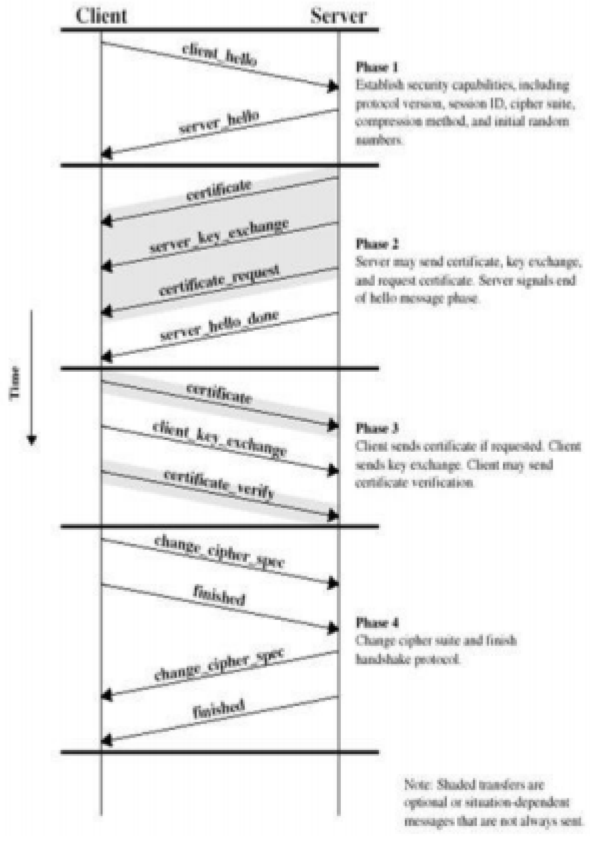
Both key exchange and Cipher Spec (this includes fields such as Cipher Algorithm, Mac Algorithm, Cipher Type, Hash Size, Key Material and IV Size) are defined.

- Compression Method: List of methods supported by client.

2. Server may send certificate, key exchange, and request certificate, it also signals end of hello message phase. The server key exchange is sent only if required. A certificate may be requested from the client if needs be by certificate request.

3. Client authentication and Key exchange upon receipt of the server done message, the client should verify that the server provided a valid certificate, if required, and check that the server hello parameters are acceptable. If all is satisfactory, the client sends one or more messages back to the server. The client sends certificate if requested. Next the client sends client key exchange message . Finally, the client may send certificate verification.

4. Finish Change cipher suite and finish handshake protocol. The secure connection is now setup and the client and server may begin to exchange application layer data.


Client
Server
client_hello
server_hello
Phase 1
Establish security capabilities, including protocol version, session ID, cipher suite, compression method, and initial random numbers.
certificate
server_key_exchange
certificate_request
server_hello_done
Phase 2
Server may send certificate, key exchange, and request certificate. Server signals end of hello message phase.
Time
certificate
client_key_exchange
certificate_verify
Phase 3
Client sends certificate if requested. Client sends key exchange. Client may send certificate verification.
change_cipher_spec
finished
change_cipher_spec
finished
Phase 4
Change cipher suite and finish handshake protocol.
Note: Shaded transfers are optional or situation-dependent messages that are not always sent.

Record Protocol

This protocol provides two services for SSL connections:

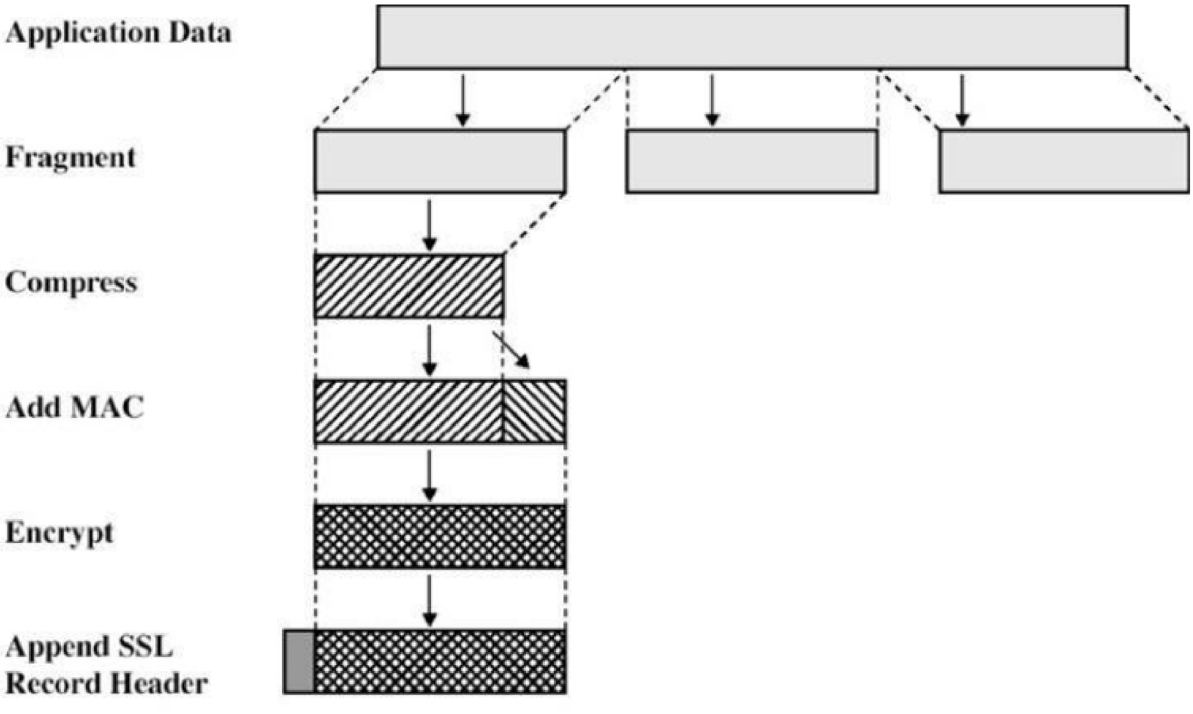
1. Confidentiality - using conventional encryption.
2. Message Integrity - using a Message Authentication Code (MAC).

In order to operate on data the protocol performs the following

- It takes an application message to be transmitted and fragments it into manageable blocks. These blocks are 214 = 16, 384 bytes or less.
- These blocks are then optionally compressed which must be lossless and may not increase the content length by more than 1024 bytes.
- A message authentication code is then computed over the compressed data using a shared secret key. This is then appended to the compressed (or plaintext) block.
- The compressed message plus MAC are then encrypted using symmetric encryption.

Encryption may not increase the content length by more than 1024 bytes, so that the total length may not exceed 214 + 2048. A number of different encryption algorithms are permitted.

- The final step is to prepped a header, consisting of the following fields:



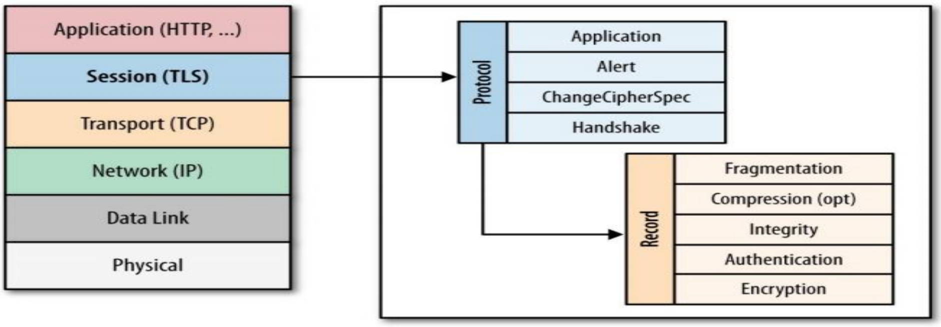
Alert Protocol

This protocol is used to convey SSL-related alerts to the peer entity. It consists of two bytes the first of which takes the values 1 (warning) or 2 (fatal). If the level is fatal SSL immediately terminates the connection. The second byte contains a code that indicates the error.

5.3 TRANSPORT LAYER SECURITY

The SSL protocol was originally developed at Netscape to enable ecommerce transaction securely on the Web, which required encryption to protect customers' personal data, as well as authentication and integrity guarantees to ensure a safe transaction.

When SSL is used correctly, a third-party observer can only infer the connection endpoints, type of encryption, as well as the frequency and an approximate amount of data sent, but cannot read or modify any of the actual data.



Transport Layer Security (TLS)

When the SSL protocol was standardized by the IETF, it was renamed to Transport Layer Security (TLS). Many use the TLS and SSL names interchangeably, but technically, they are different, since each describes a different version of the protocol .

DIFFERENCE BETWEEN SSL AND TLS

The differences between the two protocols are very minor and technical, but they are different standards. TLS uses stronger encryption algorithms and has the ability to work on different ports. Additionally, TLS version 1.0 does not interoperate with SSL version 3.0.

Netscape originally developed the SSL(Secure Sockets Layer) protocol to transmit information privately, ensure message integrity, and guarantee the server identity. SSL works mainly through using public/private key encryption on data.

It is commonly used on web browsers, but SSL can also be used with email servers or any kind of client-server transaction. For example, some instant messaging servers use SSL to protect conversations.

The Internet Engineering Task Force (IETF) created TLS (Transport Layer Security) as the successor to SSL. It is most often used as a setting in email programs, but, like SSL, TLS can have a role in any client-server transaction.

5.4 SECURE HYPER TEXT TRANSFER PROTOCOL (SHTTP)

Protocol (HTTP) that allows the secure exchange of files on the World Wide Web. Each S-HTTP file is either encrypted, contains a digital certificate, or both.

For a given document, S-HTTP is an alternative to another well-known security protocol, Secure Sockets Layer (SSL). A major difference is that S-HTTP allows the client to send a certificate to authenticate the user whereas, using SSL, only the server can be authenticated.

S-HTTP is more likely to be used in situations where the server represents a bank and requires authentication from the user that is more secure than a user id and password.

S-HTTP does not use any single encryption system, but it does support the Rivest-Shamir-Adleman public key infrastructure encryption system. SSL works at a program layer slightly higher than the Transmission Control Protocol (TCP) level. S-HTTP works at the even higher level of the HTTP application. Both security protocols can be used by a browser user, but only one can be used with a given document. Terisa Systems includes both SSL and S-HTTP in their Internet security tool kits.

5.5 TIME STAMPING PROTOCOL (TSP)

The time stamping protocol (TSP) provides proof that a certain piece of data existed at a particular time .

This service is provided by an authority called as Time stamping authority (TSA). TSP is currently under development of the PKIX working group.

Using the time stamping technique ,we can ascertain whether an electronic document was created or signed at or before a particular date and time .

This can have serious legal implications, now that digital signatures are almost as good as pen-and-paper signatures. The TSA acts like a trusted third –party notary in this scheme.

Step1: Message digest calculation : firstly, the entity (client) requiring a time stamp calculates a message digest of the original message, which needs a timestamp from the TSA. The client should use a standard message digest algorithm, such as MD5 or SHA-1 for this purpose.

Step 2: Time stamping request : Now ,the client sends the message digest calculated in step 1 to the time stamp Authority (TSA) for getting it time stamped. This is called as a time stamping request.

Step 3: Time stamping response: In response to the client's request , the TSA might decide to grant or reject the time stamp. If it decides to accept the request and process it , it signs the client's request together with the time stamp by the TSA private key. Regardless, it returns a time stamping response back to the client.

5.6 SECURE ELECTRONIC TRANSACTION (SET)

The secure electronic transaction (SET) protocol is the protocol used to facilitate the secure transmission of consumer credit card information over insecure networks, such as the Internet.

SET blocks out the details of credit card information, thus preventing merchants, hackers and electronic thieves from accessing this information.

SET was developed by SETco, led by VISA and MasterCard starting in 1996.

SET was based on X.509 certificates with several extensions. The first version was finalised in May 1997 and a pilot test was announced in July 1998.

SET makes use of Netscape's Secure Sockets Layer (SSL), Microsoft's Secure Transaction Technology (STT), and Secure Hypertext Transfer Protocol (S-HTTP). SET uses some but not all aspects of a public key infrastructure (PKI).SET allowed parties to identify themselves to each other and exchange information securely.

SET used a cryptographic blinding algorithm that, in effect, would have let merchants substitute a certificate for a user's credit-card number.

SET Participants

The following are the participants in the SET system:

- Cardholder
- Acquirer
- Merchant
- Issuer
- Acquirer
- Certificate
- Authority

Cardholder

In the electronic environment, consumers and corporate purchasers interact with merchants from personal computers over the Internet. A cardholder is an authorized holder of a payment card that has been issued by an issuer.

Merchant

A merchant is a person or organization that has goods and services to sell to the cardholder. Typically, these goods and services are offered via a Web site or by electronic mail. A merchant that accepts payment cards must have a relationship with an acquirer.

Issuer

This is a financial institution, such as a bank, that provides the cardholder with the payment card.

Acquirer

This is a financial institution that establishes an account with a merchant and processes payment card authorizations and payments. Merchants will usually accept more that one credit card brand but do not want to deal with multiple bankcard associations or with multiple individual issuers. The acquirer provides authorization to the merchant that a given card account is active and that the proposed purchase does not exceed the credit limit. The acquirer also provides electronic transfer of payments to the merchant's account.

Certification Authority (CA)

This is an entity that is trusted to issue X509v3 public-key certificates for cardholders, merchants, and payment gateways. The success of SET will depend on the existence of a CA infrastructure available for this purpose.

THE SET PROCESS :-

1. The customer opens an account – the customer opens a credit card account (such as master card or visa) with a bank (issuer) that supports electronic payment mechanisms and the SET protocol.

2. The customer receives a certificate –After the customer's identity is verified (with the help of details such as passport, business documents etc .) , the customer receives a digital certificate from a CA. The certificate also contains details such as the customer's public key and its expiry date .

3. The merchant receives a certificate- A merchant that wants to accept a certain brand of credit cards must possess a digital certificate .

4. The customer places an order-this is a typical shopping card process where in the customer browses the list of items available, searches as for specific items , selects one or more of them and places the order. The merchant, in turn , sends back details such as the list of item selected, their quantities , prices, total bill, etc . Back to the customer for his record , with the help of an order form.

5. The merchant is verified. In addition to the order form, the merchant sends a copy of its certificate, so that the customer can verify that he or she is dealing with a valid store.

6. The order and payment is verified. The customer sends both order and payment information to the merchant, along with the customer's certificate. The order confirms the purchase of the items in the order form. The payment contains credit card details. The payment information is encrypted in such a way that it cannot be read by the merchant.

The customer's certificate enables the merchant to verify the customer.

7. The merchant requests payment authorization. The merchant sends the payment information to the payment gateway, requesting authorization that the customer's available credit is sufficient for this purchase.

8. The merchant confirm the order. The merchant sends confirmation of the order to the customer.

9. The merchant provides the goods or service. The merchant ships the goods or provides the service to the customer.

10. The merchant request payment. This request is sent to the payment gateway, which handles all of the payment processing.

SET Internals

The major transaction supported by SET are

1. Purchase Request
2. Payment Authorization

3. Payment Capture

1 Purchase Request

The purchase request exchange consists of four messages:

Initiate Request
Initiate Response
Purchase Request
Purchase Response

Initiate Request :

To send SET messages to the merchant, the cardholder must have a copy of the certificates of the merchant and the payment gateway.

The customer requests the certificates in the Initiate Request message, sent to the merchant. This message includes the brand of the credit card that the customer is using the message also includes an ID assigned to this request/response pair by the customer.

Initiate Response :

The merchant generates a response and signs it with its private key.

The response includes a transaction ID for this purchase transaction. In addition to the signed response, the Initiate Response message includes the merchant's certificate and the payment gateway's certificate.

Purchase Request:

The cardholder verifies the merchant and gateway certificates by means of their respective CA signatures and then creates the order information (OI) and payment information (PI). The transaction ID assigned by the merchant is placed in both the OI and
PI.

The OI doesn't contain explicit order data such as the number and price of items. Rather, it contains an order reference generated in the exchange between merchant and customer during the shopping phase before the first SET message.

Next, the cardholder prepares the Purchase Request message .For this purpose, the cardholder generates a one-time DES encryption key, known as a session key. The message includes the following:

Purchase-related information.:

This information will be forwarded to the payment gateway by the merchant and consists of the PI and a dual signature. The dual signature is a signature that covers both the PI and the OI. It's constructed in such a way that both the merchant and the payment gateway can verify the signature, even though the merchant only sees the OI and the payment gateway only sees the PI. Both the PI and the dual signature are encrypted using the one-time session key.

Finally, the session key is encrypted with the public key of the payment gateway and added to the message; only the payment gateway will be able to decrypt and read the session key and therefore only the payment gateway will be able to recover the PI.

Cardholder's Purchase Request will be forwarded to the Payment gateway by the merchant consisting of the PI

The dual signature, calculated over the PI and OI, signed with the customer's private Signature key the OI message digest (OIMD)

Order-related information. This information is needed by the merchant and consists of the OI and the dual signature. The merchant uses the dual signature to verify that the OI is valid.

Cardholder certificate. This contains the cardholder's public key. It's needed by both the merchant and the payment gateway.

Purchase Response

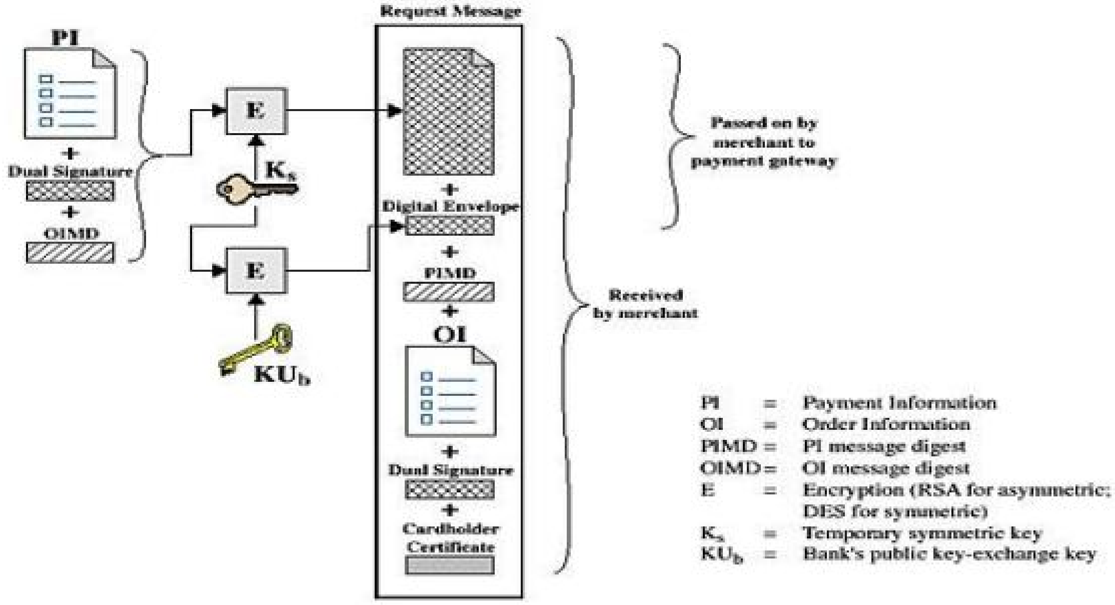
When the merchant receives the Purchase Request message, it performs the following actions:

1. Verifies the cardholder certificates by means of its CA signatures.
2. Verifies the dual signature using the customer's public signature key. This ensures that the order has not been tampered with in transit and that it was signed using the cardholder's private signature key.

Processes the order and forwards the payment information to the payment gateway for authorization.

Sends a purchase response to the cardholder.

Purchase Request



2 . Payment Authorization :

During the processing of an order from a cardholder, the merchant authorizes the transaction with the payment gateway.

The payment authorization ensures that the transaction was approved by the issuer. This authorization guarantees that the merchant will receive payment, the merchant can therefore provide the services or goods to the customer. The payment authorization exchange consists of two messages: Authorization Request and Authorization response. The merchant sends an Authorization Request message to the payment gateway, which consisting Purchase Related Information, Authorization related information and certificates.

Authorization Request

a. Purchase-Related information. This information was obtained from the customer and consists of:

The PI

The dual signature, calculated over he PI and OI, signed with the customer's private signature key

The OI message digest (OIMD)

The digital envelope

b. Authorization-related information.

This information is generated by the merchant and consists of

An authorization block that includes the transaction ID, signed with the merchant's private signature key and encrypted with a one-time symmetric key generated by the merchant Digital envelope. This is formed by encrypting the one-time key with the payment gateway's public key-exchange key.

c. Certificates.

The merchant includes the cardholder's signature key certificate (used to verify the dual signature), the merchant's signature key certificate (needed in the payment gateway's response).

The payment gateway performs the following tasks:

1. Verifies all certificates
2. Decrypts the digital
3. Verifies the merchant's signature on the authorization block
4. Decrypts the digital envelope of the payment block to obtain the symmetric key and then decrypts the payment block
5. Verifies the dual signature on the payment block
6. Verifies that the transaction ID received from the merchant matches that in the PI received (indirectly) from the customer
7. Requests and receives an authorization from the issuer having obtained authorization from the issuer, the payment gateway returns an

Authorization Response message to the merchant. It includes the following elements:

1. Authorization- related information. Includes an authorization block, signed with the gateway's private signature key and encrypted with a one-time symmetric key generated by the gateway. Also includes a digital envelope that contains one-time key encrypted with the merchant public key-exchange key.

2. Capture token information. This information will be used to effect payment later. This block is of the same form as (1)-namely, assigned, encrypted capture token together with a digital envelope. This token is not processed by the merchant. Rather, it must be returned, as is, with a payment request.

3. Certificate. The gateway's signature key certificate. With the authorization from the gateway, the merchant can provide the goods or service to the customer.

3. Payment Capture

To obtain payment, the merchant engages the payment gateway in a payment capture transaction, consisting of a capture request and a capture response message.

For the Capture Request message, the merchant generates, signs, and encrypts a capture request block, which includes the payment amount and the transaction ID. The message also includes the encrypted capture token received earlier for this transaction, as well as the merchant's signature key and key-exchange key certificates.

When the payment gateway receives the capture request message, it decrypts and verifies the capture request block and decrypts and verifies the capture token block. It then checks for consistency between the capture request and capture token. It then creates a clearing request that is sent to the issuer over the private payment network.

This request causes funds to be transferred to the merchant’s account. The gateway then notifies the merchant of payment in a Capture Response message.

The message includes a capture response block that the gateway signs and encrypts.

The message also includes the gateway’s signature key certificate. The merchant software stores the capture response to be used for reconciliation with payment received from the acquirer .

Questions:

No.1 What is tcp/ip ? Explain the function of each layer in tcp/ip suite. (2017)

2. What is secure electronic transaction (SET)?Describe process involved in SET.
3. What is role of SHTTP in cryptography?
4. What is static and dynamic page?

UNIT-6

USER AUTHENTICATION

6.1 Authentication basics

Authentication means verifying the identity of someone (a user, device, or an entity) who wants to access data, resources, or applications. Validating that identity establishes a trust relationship for further interactions.

Authentication also enables accountability by making it possible to link access and actions to specific identities. After authentication, authorization processes can allow or limit the levels of access and action permitted to that entity.

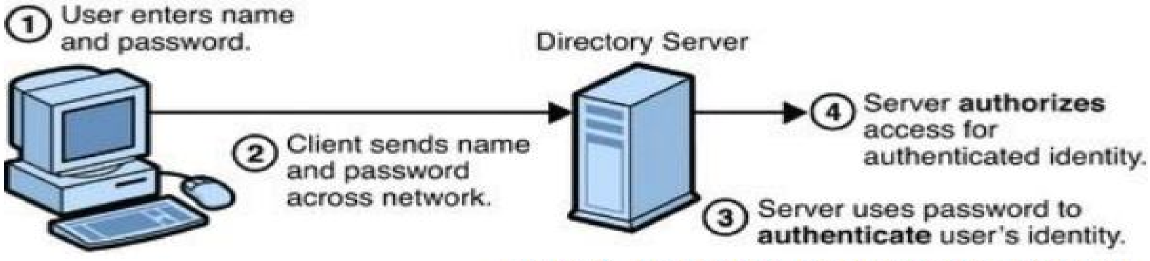
6.2 Password

Passwords are the most common form of authentication. A password is a string of alphabets, number and special character, which is supposed to be known only to the entity that is being authenticated.

Simple password authentication offers an easy way of authenticating users. In password authentication, the user must supply a password for each server, and the administrator must keep track of the name and password for each user, typically on separate servers.

Steps in Password-Based Authentication

Every user in the system is assigned a user id and initial password. The password is stored in clear text in the user database against the user id. Figure shows the steps involved in authenticating a client by using a user id (name) and password.



1 User enters name and password.
Directory Server
2 Client sends name and password across network.
4 Server **authorizes** access for authenticated identity.
3 Server uses password to **authenticate** user's identity.

Fig 6.1 Password-Based Authentication

InFig6.1 , password authentication is performed in the following steps.

1. *Prompt For user id and Password -* The application sends a screen to the user, prompting user id and password.
2. *User enters user id and password-* The user enters her id and password and presses the ok button. This causes the user id and password to travel in clear text to the server.
3. *User id and pass word validation-* The server consults the user database to validate the user id and password. This job is done by user authentication program.
4. *Authentication Results-* Depending on the failure or success of the validation of the user id and password, the user authentication program return an appropriate result back to the server.
5. *Inform user accordingly-* Depending on the outcome (success or failure), the server sends back an appropriate screen to the user. If the user authentication is successful, the serve resends a menu option, which lists the action user is allowed to perform. If the user authentication is failure, the server sends an error screen to the user.

Problem with the scheme

Database contains passwords in clear text.

Passwords travels in clear text from the user computer to the server.

Password encryption :

The transmission of clear text password, the password is encrypted on the user computer and the send it to the server for authentication. This means that there must be some sort of cryptographic functionality on the users computers (i.e. the client side).

There are two encryption processes:-

The first encryption happens before a password is stored in the user Database.

The other encryption is performed on the user's computer to encrypt the password before it is transmitted to the server.

These two encryption operations are no way directly related to the each other. They may even be using totally different approaches to encryption(for example ,the user computer would use the symmetric key shared between the user and the server first for encryption and then the SSL session key for secure transmission, where as the server could only use the shared symmetric key ,as it does not have to perform any transmission)

Therefore , the encrypted password in the database would not actually be same as the encrypted password coming from the user's computer . However, the main idea here is that both the encrypted passwords – neither of them is in clear text.

The fact that the encrypted versions of these two passwords may not be the same and that the server – side application logic would perform the necessary conversions between the two for verification is a minor technical variation.

6.3 Authentication Tokens

An authentication token is an extremely useful alternative to a password. An authentication token is a small device that generates a new random value every time it is used.

This random value becomes the basis for authentication. The small devices are typically of the size of small key chains, calculators of credit cards. Usually an authentication token has the following features:

Processor

Liquid crystal display(LCD)for display outputs

Battery

(optionally)a small keypad for entering information

(optionally)a real-time clock

Each authentication token (i.e. device) is pre-programmed with a unique number, called as random seed, or just seed. The seed forms the basis for ensuring the uniqueness of the output produced by the token.

Step 1: creation of a token

Whenever an authentication token is created, the corresponding random seed is generated for the token by the authentication server (a special server that is configured to work with authentication tokens).this set is stored or pre-programmed inside token, as well as its entry is made against that user's record in the user database.

Conceptually this seed is as the user's password (although this is technically completely different from a password), the user does not known about the value of the seed, unlike a password. This is because the seed is used automatically by the authentication token.

Step 2: use of token

An authentication token automatically generates pseudorandom numbers, called is onetime password or one-time pass codes are generated randomly by an authentication token, based on the seed value that they are pre-programmed with.

They are one time because they are generated, used once, and discarded forever. When a user wants to be authenticated, the user will get a screen to enter the user id and the latest one time password for this, the user will want enter the user id and the one time password obtained from the authentication token.

The user id and password travel to the server as a part of the login request. The server obtains the seed corresponding the user id from the user database, as seed retrieval program .
It then calls another program called as password validation program, to which the server gives the seed and the one time password.

If a user loses an authentication token? Can another user simply grab use it? To deal with such situation, the authentication token be generally protected by a password or a 4 digit pin.

Only when this PIN is entered can the one time password be generated . This also basis for what is called as multi-factor authentication.

Step 3: server returns an appropriate message back to the user

Finally, the server sends an appropriate message back to the user, depending on whether the previous operations yielded success or failure.

Authentication token types :

There are two main types of authentication tokens.

1. Challenge/Response Tokens
2. Time based Tokens

1. Challenge/response:-

Challenge-response authentication is a family of protocols in which one party presents a question ("challenge") and another party must provide a valid answer ("response") to be authenticated. The simplest example of a challenge-response protocol is password authentication, where the challenge is asking for the password and the valid response is the correct password.

Steps Of Challenge/response works

Step1: User sends a login request

In this technique, the user sends the login request only with her user id.

Step2: Server creates a random challenge

When the server receives the user's login request containing the user id alone it first checks to see if the user id is a valid one. If the user id is valid, the server now creates random challenge (a random number, generated using a pseudo-random number generation technique) and sends it back to the user.

The random challenge can travel as plain text from the server to the user's computer.

Step3: User signs the random challenge with the message digest of the password

The user gets a screen, which displays the user id, the random challenge received from the server, and data entry field, with the label password. Let us assume that the random challenge sent by the user was 8102811291012.

At this stage, the user reads the random challenge displayed on the screen. It first opens the token using her PIN and the keys in the random challenge received from the server inside the token. In order to do this, the token has a small keypad.

The token accepts the random challenge, and encrypt it with seed value, which is known only to itself. The result is the random challenge encrypted with the seed. This result is displayed on the display (LCD) of the token.

The user reads this value and types it in the password field on the screen. This request is then sent to the server as the login request.

Step3: server verifies the encrypted random challenge received from the user. The server receives the random challenge, which was encrypted with the seed by the user's authentication token. In order to verify that the random challenge was indeed encrypted by the correct seed, the server must perform an identical operation.

The server can decrypt the encrypted random challenge received from the user with the seed value for the user. As we know; the seed for the user is available to the server via the user database. If this decryption matches the original random challenge available on the server, the server can be assured that the random challenge was indeed encrypted by the correct seed of the user's authentication token.

Alternatively, the server can simply encrypt its own version of the random challenge (i.e. the one which was sent earlier to the user) with the seed for the user. If this encryption produces an encrypted random challenge, which matches with the encrypted random challenge received from the user, the server can be assured that the random challenge was indeed signed by the correct seed.

Step4: server returns an appropriate message back to the user

Finally, the server sends an appropriate message back to the user, depending on whether the previous operations yielded success or failure.

2. Time based tokens

In the challenge /response mechanism discussed earlier the user has to make three entries:

firstly the user has to enter the PIN to access the token;

secondly, the user has to read the random challenge from the screen and key in the random number challenge into the token.

thirdly, the user has to read the encrypted random challenge from the LCD of the token and enter it into the password field.

Users generally make quite a few mistakes in all this process, resulting into a lot of flow of wasteful information between the user's computer, the server and the authentication token.

In time based token the server need not send any random challenge to the user. The theory behind this is usage of time as variable input to the authentication process.

Steps of Time based tokens

Step1: password generation and login request

The token is pre-performed with seed. These tokens do not require any user inputs. Instead, these tokens automatically generate a password every 60 second and display the latest password on the LCD output for the user to read and use it.

For generation the password, the time-based tokens use two parameters: the seed and the system time. It performs some cryptographic function on these two input parameters to produce the password automatically.

The token then displays it onto the LCD. Whenever a user wishes to log on she takes a look at the LCD display, reads the password from there and uses her id and password for login.

Step2: server-side verification

The server receives the password .It also performs an independent cryptographic function on the user‟s seed value and the current system time to generate its version of the password. If the two values match, it considers the user as a valid one.

Step3: server returns an appropriate message back to the user

Finally, the server sends an appropriate message back to the user, depending on whether the previous operations yielded success or failure.

6.4 Certificate based authentication :

This is based on the digital certificate of a user. FIPS-196 is a standard that specifies the operation of this mechanism.

Certificate based authentication is a stronger mechanism as compared to a password based authentication mechanism, because here the user is expected to have something (certificate) and not known something (password).

At the time of login the user is requested to send her certificate to the server over the network as part of the login request. A copy of the certificate exists on the server, which can be used to verify that the certificate is indeed a valid one.

Working of Certificate based authentication

Step 1 Creation, storage and distribution of digital certificates-

The digital certificates are created by CA for each user and the certificate are send to the respective users. The copy of a certificate is stored by the server in its data base, in order to verify the certificate during the user's certificate based authentication.

Step 2 Login request-

During login request the user sends her user id to the server.

Step 3 Server creates a random challenge-

When the server receives the user's login request, it validates the user id. If the user id is valid, the server now creates a random number challenge and sends it back to the user.

Step 4 User signs the random challenge-

The user has to sign the random challenge with her private key. The private key stored in a disk file on the user computer. The private key is used to encrypt the random challenge received from server to create users digital signature.

This is done by two steps:

first a message digest of the random challenge is created and the message digest is then encrypted with the users private key and send to the server.

The server then verifies the user's signature by obtaining the public key from the user database. The public key is used to decrypt the signed random challenge received from the user. After that it compares this decrypted random challenge with its original random challenge.

Step 3 Server returns an appropriate message back to the users -

Finally the server sends an appropriate message back to the user, wheatear the previous operation is success or failure.

6.5 Biometric Authentication:

Biometrics refers to the automatic identification of a person based on his or her physiological or behavioural characteristics.

A biometric device works on the basis of some human characteristics, such as fingerprints, voice or the pattern of lines in the iris of eye.

The user database contains a sample of user's biometric characteristics

During the authentication, the user is required to provide another sample of the users' biometric characteristic.

This is matched with the one in the database, and if the two samples are same, the user is considered to be a valid one.

The samples produced during every authentication process can vary slightly. (e.g. cuts on the finger)

An approximate match can be acceptable.

Any Biometric Authentication System defines two configurable parameters:

False Accept Ratio (FAR):

It is a measurement of the chance that a user who should be rejected is actually accepted by a system as good enough.

The probability that the system incorrectly matches the input pattern to a non-matching template in the database. It measures the percent of invalid inputs which are incorrectly accepted. In case of similarity scale, if the person is an imposter in reality, but the matching score is higher than the threshold, then he is treated as genuine. This increases the FAR, which thus also depends upon the threshold value.

False Reject Ratio (FRR):

It is a measurement of the chance that a user who should be accepted as valid is actually rejected by a system as not good enough. The probability that the system fails to detect a match between the input pattern and a matching template in the database. It measures the percent of valid inputs which are incorrectly rejected.

Thus FAR and FRR are exactly opposite to each other.

Biometric characteristics:

1) Physiological
2) Behavioural

Physical biometrics:

Fingerprint
Facial recognition/face location
Hand geometry
Iris scan
Retina scan

Fingerprint recognition

A live acquisition of a person's fingerprint.

Dots (very small ridges), Space between two temporarily divergent ridges), Spurs (a notch protruding from a ridge), Bridges (small ridges joining two longer adjacent ridges), crossovers (two ridges that cross each other).

Facial Recognition

1. Capture image
2. Find face in image
3. Extract features (store template)
4. Compare templates
5. Declare matches

Hand Geometry

Hand or finger geometry is an automated measurement of many dimensions of the hand and fingers.

Iris recognition

Iris scanning measures the iris pattern in the coloured part of the eye.

Retina recognition

Images back of the eye and compares blood vessels with existing data.

Behavioural biometrics:

Speaker/ voice recognition.
Signature/ handwriting.
Keystroke/ patterning.

Speaker / Voice Recognition

Voice or speaker recognition uses vocal characteristics to identify individuals using a pass-phrase.

A telephone or microphone can serve as a sensor.

Signature Verification

An automated method of measuring an individual's signature.

This technology examines speed, direction, and pressure of writing; the time that the stylus is in and out of contact with the "paper''.

Keystroke dynamics

It is an automated method of examining an individual's keystrokes on a keyboard.

This technology examines such dynamics as speed and pressure, the total time taken to type particular words, and the time elapsed between hitting certain keys.

APPLICATIONS:

Prevent unauthorized access to ATMs, Cellular phones Desktop PCs.
Criminal identification.
In automobiles biometrics can replace keys with keyless entry devices.
Airport security.

UNIT:7

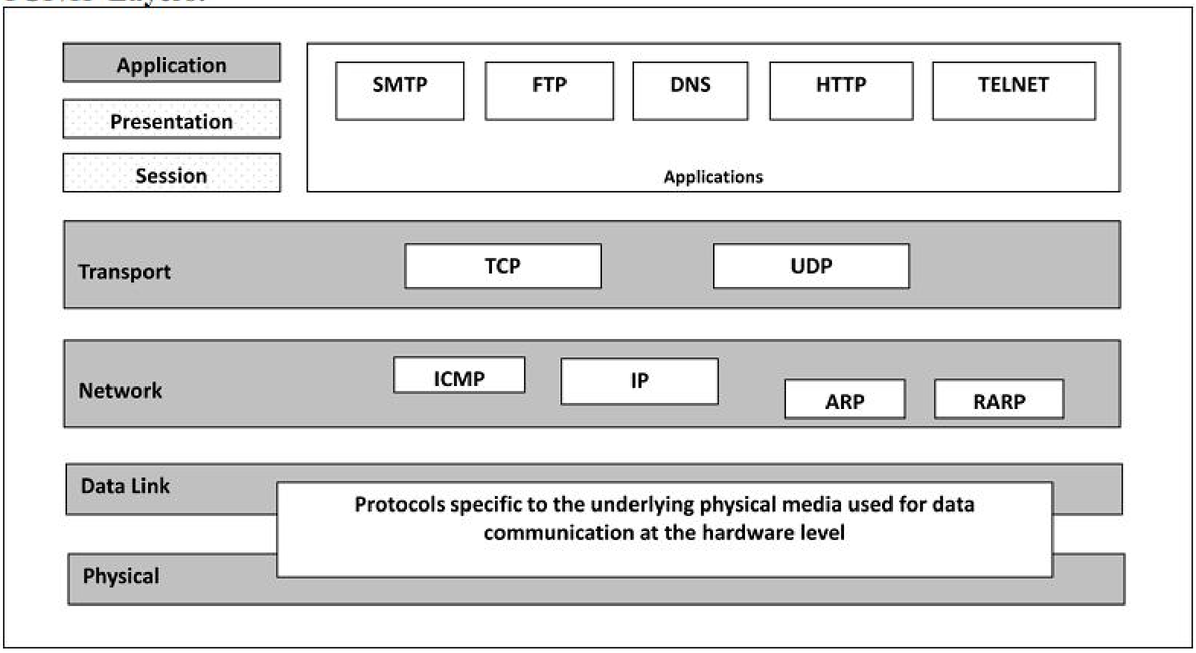
NETWORK SECURITY AND VPN

TCP/IP:

TCP/IP Protocol Suite:

- The TCP/IP protocol suite was developed prior to the OSI model. Therefore, the layers in the TCP/IP protocol suite do not exactly match those in the OSI model.
- TCP/IP protocol suite is made of five layers: Application Layer, Transport Layer, Internet Layer, Network Access Layer
- TCP/IP is a hierarchical protocol made up of interactive modules, each of which provides a specific functionality; however, the modules are not necessarily interdependent.
- At the transport layer, TCP/IP defines three protocols: Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Stream Control Transmission Protocol (SCTP).
- At the Internet layer, the main protocol defined by TCP/IP is the Internet Protocol (IP); there are also some other protocols that support data movement in this layer.

TCP/IP Layers:



TCP segment format:

A packet in TCP is called a *segment*. The segment consists of a header of 20 to 60 bytes, followed by data from the application program. The header is 20 bytes if there are no options and up to 60 bytes if it contains options.

Source port address:

This is a 16-bit field that defines the port number of the application program in the host that is sending the segment. This serves the same purpose as the source port address in the UDP.

Destination port address:

This is a 16-bit field that defines the port number of the application program in the host that is receiving the segment. This serves the same purpose as the destination port address in the UDP.

Sequence number:

This 32-bit field defines the number assigned to the first byte of data contained in this segment. As TCP is a stream transport protocol. To ensure connectivity, each byte to be transmitted is numbered.

The sequence number tells the destination which byte in this sequence is the first byte in the segment.

During connection establishment each party uses a random number generator to create an initial sequence number (ISN), which is usually different in each direction.

Acknowledgment number:

This 32-bit field defines the byte number that the receiver of the segment is expecting to receive from the other party. If the receiver of the segment has successfully received byte number x from the other party, it Returns x+1 as the acknowledgment number.

Header length:

This 4-bit field indicates the number of 4-byte words in the TCP header. The length of the header can be between 20 and 60 bytes. Therefore, the value of this field is always between 5 (5 *4=20) and 15 (15*4=60).

Reserved: This is a 6-bit field reserved for future use.

Control:

This field defines 6 different control bits or flags . One or more of these bits can be set at a time. These bits enable flow control, connection establishment and termination, connection abortion, and the mode of Flags from left to right.

Window size:

This field defines the window size of the sending TCP in bytes. Note that the length of this field is 16 bits, which means that the maximum size of the window is 65,535 bytes.

Checksum:

The 16-bit checksum field is used for error-checking of the header and data.

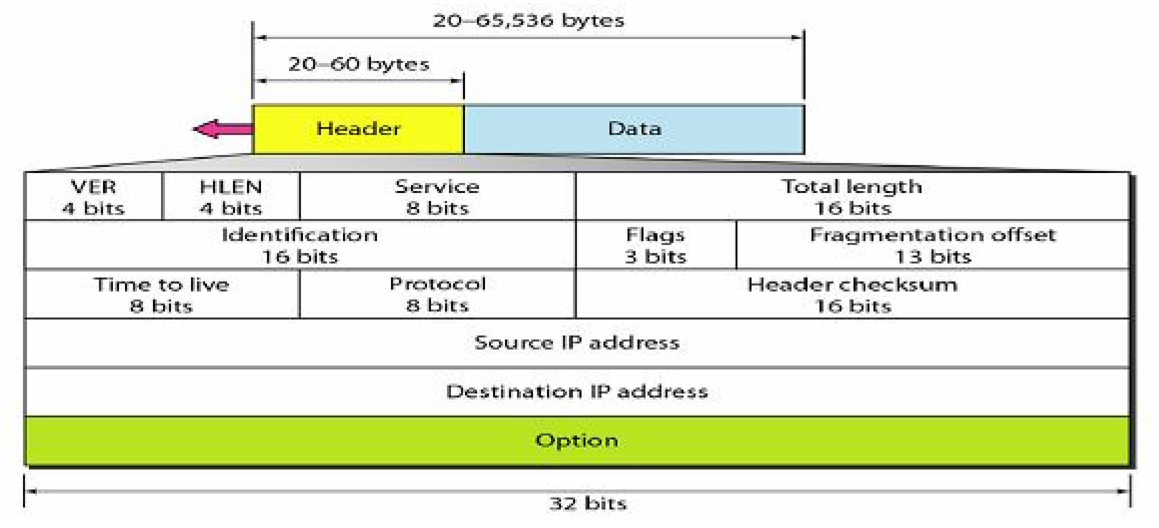
Urgent pointer:

if the URG flag is set, then this 16-bit field is an offset from the sequence number indicating the last urgent data byte.

IP DATAGRAM FORMAT:

- Packets in the network (internet) layer are called datagram.
- A datagram is a variable-length packet consisting of two parts: header and data.
- The header is 20 to 60 bytes in length and contains information essential to routing and delivery.

IP header format:



Version (VER):

This 4-bit field defines the version of the IP protocol. Currently the version is 4(IPv4).

Header length (HLEN):

This 4-bit field defines the total length of the datagram header in 4-byte words. This field is needed because the length of the header is variable (between 20 and 60 bytes). When there are no options, the header length is 20 bytes, When the option field is at its maximum size(i.e. 60) .

Service type (TOS):

It defines how the datagram should be handled. Part of the field was used to define the precedence of the datagram; the rest defined the type of service (low delay, high throughput, and so on).

Total length:

It defines the total length of the datagram including the header in bytes. It is a 16-bit number so maximum IP size is 216 bytes.

Identification:

This 16-bit field identifies a datagram originating from the source host. The combination of the identification and source IP address must uniquely define a datagram as it leaves the source host.

Flags:

This is a three-bit field. The first bit is reserved (not used). The second bit is called the do not fragment bit. If its value is 1, the machine must not fragment the datagram. If its value is 0, the datagram can be fragmented if necessary. The third bit is called the more fragment bit. If its value is 1, it means the datagram is not the last fragment; there are more fragments after this one. If its value is 0, it means this is the last or only fragment.

Fragmentation offset:

This 13-bit field shows the relative position of this fragment with respect to the whole datagram.

Time to live:

A datagram has a limited lifetime in its travel through an internet. This field was originally designed to hold a timestamp, which was decremented by each visited router. The datagram was discarded when the value became zero.

Protocol:

This 8-bit field defines the higher-level protocol that uses the services of the IP layer. An IP datagram can encapsulate data from several higher level protocols such as TCP, UDP, ICMP, and IGMP. This field specifies the final destination protocol to which the IP datagram should be delivered.

Header Checksum:

This fields represents a value that is calculated using an algorithm covering all the fields in header. This field is used to check the integrity of an IP datagram.

Source address:

This 32-bit field defines the IP address of the source. This field must remain unchanged during the time the IP datagram travels from the source host to the destination host.

Destination address:

This 32-bit field defines the IP address of the destination. This field must remain unchanged during the time the IP datagram travels from the source host to the destination host.

Firewall:

Firewalls can be used to protect a local system or network of systems (Internal Network) from Outside networks (Internet) from security threats.

➢ Special type of router.

➢ Frequently used to prevent unauthorized internet users from accessing private networks connected to the internet, especially intranets.

➢ Controls transmission between internal and external networks. i.e. All messages entering or leaving the intranet pass through the firewall, which examines each message and blocks those that do not meet the specified security criteria.

➢ It is essentially a barrier between two networks that evaluates all incoming or outgoing traffic to determine whether or not it should be permitted to pass to the other network. i.e. decides what to allow/disallow.

➢ Can be implemented in both hardware and software, or a combination of both.

➢ At broad level, there are two kind of attacks:

- Most corporations have large amounts of valuable and confidential data in their networks. Leaking of this critical information to competitors can be a great setback.

- Apart from the danger of the insider information leaking out, there is a great danger of the outside elements (such as viruses and worms) entering a corporate network to create disaster.

Firewall characteristics/ Design Goals of Firewalls:

A firewall is defined as collection of components placed between two networks that collectively have Following characteristics:

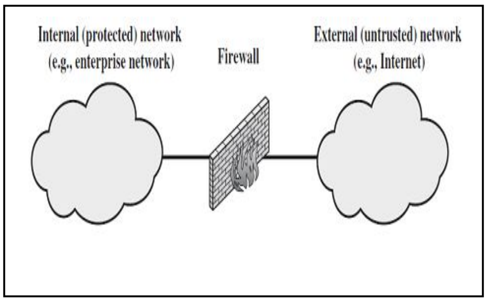
1. All traffic from inside to outside, and vice versa, must pass through the firewall.

– This is achieved by physically blocking all access to the local network except via the firewall.

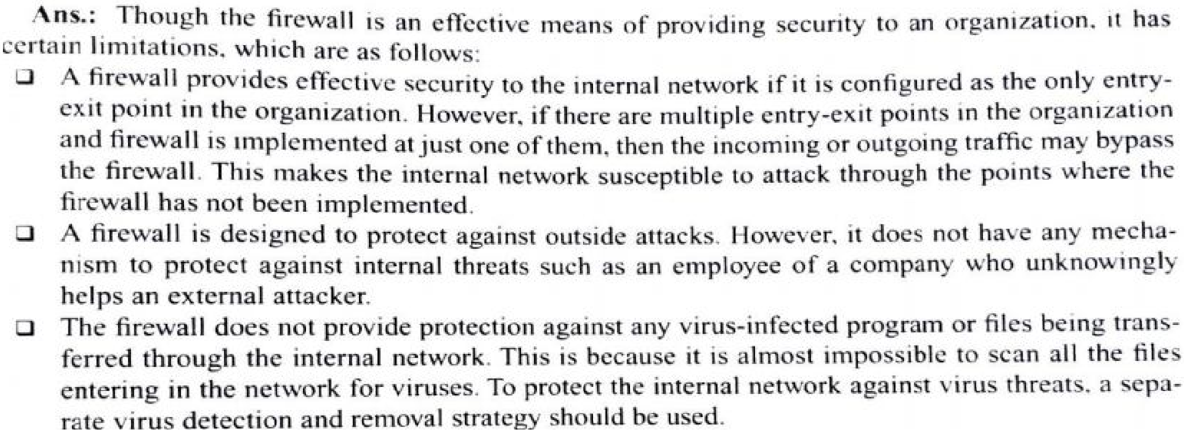
2. Only authorized traffic, as defined by the local security policy, will be allowed to pass.

– Various types of firewalls are used, which implement various types of security policies.

The firewall itself must be strong enough, so as to render attacks on it useless.


Internal (protected) network
(e.g., enterprise network)
Firewall
External (untrusted) network
(e.g., Internet)

Limitations of Firewalls:



Ans.: Though the firewall is an effective means of providing security to an organization, it has certain limitations, which are as follows:

❑ A firewall provides effective security to the internal network if it is configured as the only entry-exit point in the organization. However, if there are multiple entry-exit points in the organization and firewall is implemented at just one of them, then the incoming or outgoing traffic may bypass the firewall. This makes the internal network susceptible to attack through the points where the firewall has not been implemented.

❑ A firewall is designed to protect against outside attacks. However, it does not have any mechanism to protect against internal threats such as an employee of a company who unknowingly helps an external attacker.

❑ The firewall does not provide protection against any virus-infected program or files being transferred through the internal network. This is because it is almost impossible to scan all the files entering in the network for viruses. To protect the internal network against virus threats, a separate virus detection and removal strategy should be used.

Types of Firewalls:

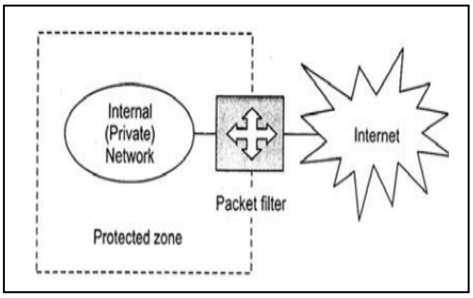
➢ Packet Filters.

➢ Application Level Filtering.

➢ Circuit Level Gateways.

Packet Filtering Firewall:

- A firewall may act as a packet filter.
- It can act as a positive filter, if pass only packets that meet specific criteria, or as a negative filter, if rejecting any packet that not meets certain criteria.



- A packet filtering firewall applies a set of rules to each incoming and outgoing IP packet and then forwards or discards the packet.
- It is also called as screening router or screening filter.

A packet filter performs the following functions/operations:

Conceptually, a packet filter can be considered as a router that perform 3 main actions, that is shown in figure

(a) Receive each packet as it arrives.

(b) Pass the packet through a set of rules,

based on the contents of the IP and transport header fields of the packet. If there is a match with one of the set rules, decide whether to accept or discard the packet based on that rule.

For example: a rule could specify: disallow all incoming traffic from an IP address 157.29.19.10 (this

IP address is taken just as an example)
(c) If there is no match with any rule, take the default action. The default can be discard all packets or accept all packets.

2. Application Gateways:

An application gateway is known as application proxy or application-level proxy, an application gateway is an application program that runs on a firewall system between two networks. When a client program establishes a connection to a destination service, it connects to an application gateway, or proxy.

The client then negotiates with the proxy server in order to communicate with the destination service. In effect, the proxy establishes the connection with the destination behind the firewall and acts on behalf of the client, hiding and protecting individual computers on the network behind the firewall.

This creates two connections: one between the client and the proxy server and one between the proxy server and the destination. Once connected, the proxy makes all packet-forwarding decisions. Since all communication is conducted through the proxy server, computers behind the firewall are protected.

It filters incoming node traffic to certain specifications which mean that only transmitted network application data is filtered. Such network applications include File Transfer Protocol (FTP), Telnet, Real Time Streaming Protocol (RTSP) and BitTorrent.

While this is considered a highly secure method of firewall protection, application gateways require great memory and processor resources compared to other firewall technologies, such as stateful inspection.

Virtual Private Network (VPN):

➢ A VPN is thus a mechanism to simulate a private network over a public network, such as the Internet.

➢ The term virtual signifies that it depends on the use of virtual connections.

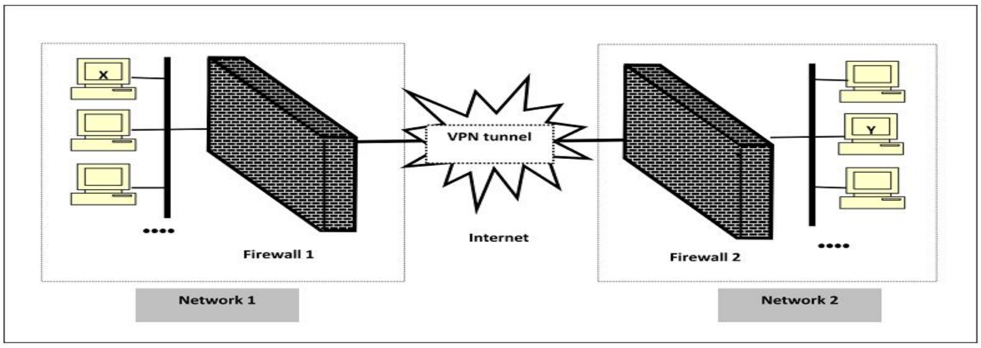
➢ These connections are temporary and do not have any Physica1 presence. They are made up of packets.

➢ Uses the Internet as if it is a private network.

➢ Far less expensive than a leased line.

➢ Uses IPSec protocol.

VPN Architecture:

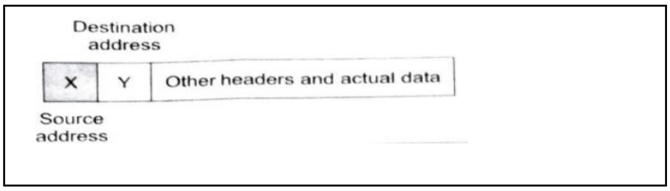


➢ We have shown two networks, Network I and Network 2. Network I connects to the Internet via a firewall named Firewall I. Similarly, Network 2 connects to the Internet with its own firewall 2.

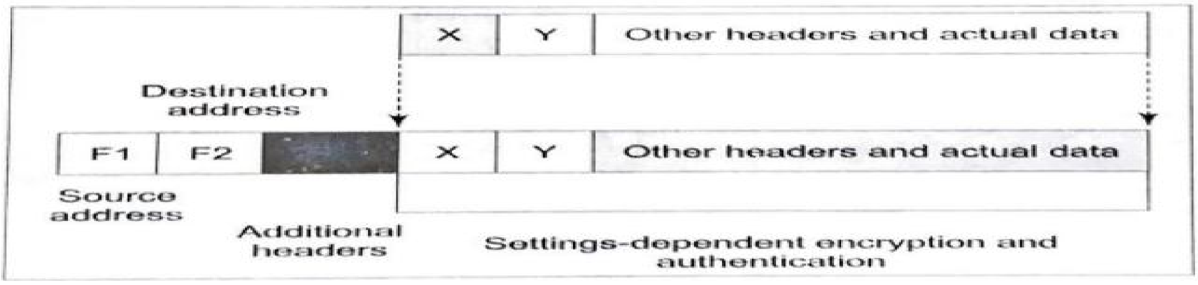
➢ The two firewalls are virtually connected to each other via the Internet. We have shown this with the help of a VPN tunnel between the two firewalls.

Let us understand how the VPN protects the traffic passing between any two hosts on the two different networks. For this, let us assume that host X on Network 1 wants to send a data packet to host Y on Network 2. This transmission would work as follows.

I. Host X creates the packet, inserts its own IP address as the source address and the IP address of host Y as the destination address. This is shown in figure. It sends the packet using the appropriate mechanism.

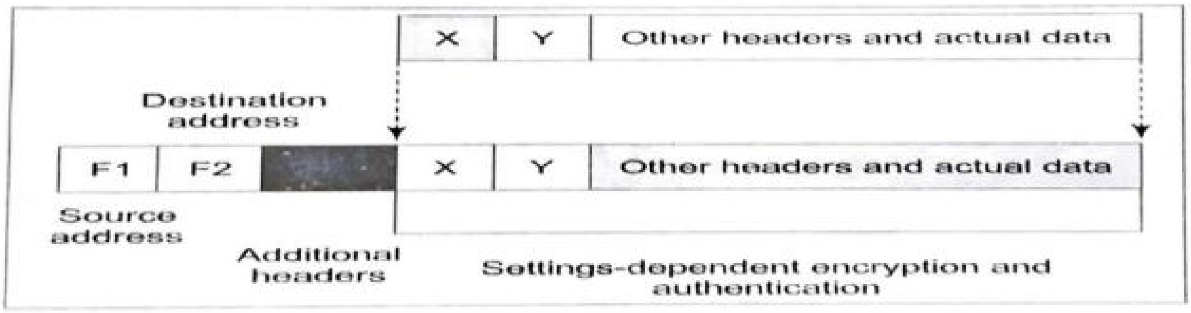


2. The packet reaches firewall 1. As we know, firewall 1 now adds new headers to the packet. In these new headers, it changes the source IP address or the packet from that of host X to its own address (i.e. the IP address of Firewall 1, say F1). It also changes the destination IP address of the packet from that of host Y to the IP address of Firewall 2. say F2). This is shown in Fig. It also performs the packet encryption and authentication, depending on the settings and sends the modified packet over the Internet.



X
Y
Other headers and actual data
Destination address
F1
F2
X
Y
Other headers and actual data
Source address
Additional headers
Settings-dependent encryption and authentication

3.The packet reaches firewall1 2 over the internet, via one more routers, as usual, Firewall 2 discards the outer header and performs the outer header and performs appropriates decryption and other cryptographic functions as necessary. This yields the original packets, as was created by host X in step 1. This is shown in fig. It then takes a look the plain text contents of the packets and realizes that the packet is meant for host Y. Therefore, it delivers the packet to host Y.



X
Y
Other headers and actual data
Destination address
F1
F2
X
Y
Other headers and actual data
Source address
Additional headers
Settings-dependent encryption and authentication

Main Network Protocols:

There are three network protocols.

IPSec:(Internet Protocol Security): It is a framework for uses cryptographic security services developed by the IETF to protect secure exchange communications over Internet Protocol (IP).

PPTP(Point-to-Point Tunneling Protocol): It is a network protocol. It mainly support the vpn connectivity bet a single user and a LAN.

L2TP:(Layer Two Tunneling Protocol): It is an extension of the Point-to-Point Tunneling Protocol (PPTP) used by Internet service providers (ISPs) to operate Virtual Private Networks (VPNs).

IP Security (IPSec) Protocols:

➢ Before IPSec was initiated, the IP packets were prone to security failure.

➢ The technology that brings secure communications to the internet protocol layer or network layer is called IP Security, commonly abbreviated IPSec.

➢ IPSec is a set of services and protocols that provide a complete security solution for an IP networks.

➢ It is a collection of protocols designed by the Internet Engineering Task Force (IETF) to provide security in the internet layer.

➢ It can be used in protecting data flows between a pair of host(host-to-host), between a pair of security gateways(network-to-network), or between a security and a host(network-to-host).

General IP Security mechanisms:

It provides:

Authentication: Ensures that packets are arriving from the actual source.

Confidentiality: It allows two communicating nodes to transfer msg in an encrypted form in order to prevent third party.

Key management: Provide platform to key exchange in a secured manner.

Applications of IP security: (Important)

➢ IPSec provides the capability to secure communications across a LAN, across private and public WANs, and across the Internet. Examples of its use include:

Secure remote access over the Internet:

➢ An end user whose system is equipped with IP security protocols can make a local call to an Internet Service Provider (ISP) and gain secure access to a company network. This reduces travelling cost and time wastage of employees and telecommuters.

Secure branch office connectivity over the Internet:

➢ A company can build a secure virtual private network over the Internet or over a public WAN. This enables connecting all the branches of company. That will save the costs of creating a private network and network management overhead.

Establishing extranet and intranet connectivity with partners:

➢ IPSec can be used to secure communication with other organizations, ensuring authentication and confidentiality and providing a key exchange mechanism.

Enhancing electronic commerce security:

➢ Even though some Web and electronic commerce applications have built-in security protocols, the use of IPSec enhances that security.

Benefits of IP security: (Important)

➢ IPSec can be transparent to end users.

- There is no need to train users on security mechanisms.
- No need to issue or cancel keys to and from the users.

➢ When IPSec is implemented in a firewall or router, it provides strong security that can be applied to all traffic crossing the perimeter.

- Traffic within a company or workgroup does not have to use IPSec, thus it minimize the overhead of security-related processing.

➢ IPSec in a firewall is resistant to bypass if all traffic from the outside must use IP and the firewall is the only means of entrance from the Internet into the organization.

➢ Since IPSec is implemented at network layer, there is no need to make any changes at the upper layers such as transport layer (TCP, UDP) and application layer.

➢ IPSec can provide security for individual users if needed. Individuals can set up a secure virtual sub-network within an organization for sensitive applications.

IP security services: (Important)

➢ IPSec provides security services at the IP layer.

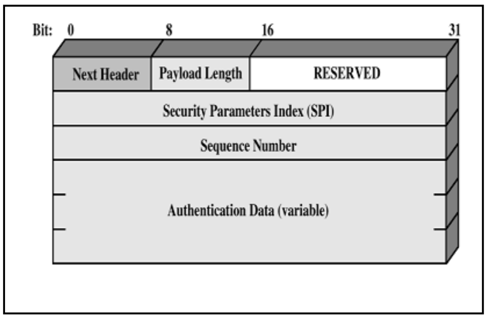
➢ Two protocols are used to provide security:

▪ An authentication protocol designated by the header of the protocol, Authentication Header (AH).

▪ And a combined encryption/ authentication protocol designated by the format of the packet for that protocol, Encapsulating Security Payload (ESP).

➢ Lists the following services:

1. Access control
2. Connectionless integrity
3. Data origin authentication
4. Rejection of replayed packets (a form of partial sequence integrity)
5. Confidentiality (encryption)
6. Limited traffic flow confidentiality


Bit: 0
8
16
31
Next Header
Payload Length
RESERVED
Security Parameters Index (SPI)
Sequence Number
Authentication Data (variable)

IPSec Architecture/ Protocol:

The IPSec architecture comprises of different protocols like:

1. Authentication Header (AH) protocol.
2. Encapsulating Security Payload (ESP).

1. Authentication Header (AH) protocol:

➢ Provides support for data integrity and authentication (MAC code) of IP packets.

➢ Guards against replay attacks.

A brief description each field:

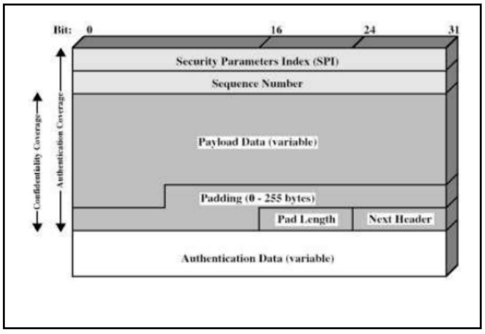
Next header: The 8-bit next-header field defines the type of payload carried by the IP datagram (such as TCP, UDP, ICMP).

Payload length: this is 8 bit field is misleading. It defines the length of the authentication header.

Security parameter index: this is 32bit security parameter index (SPI) field plays the role of a virtual-circuit identifier and is the same for all packets sent during a connection called a security association.

Sequence Number: a 32-bit sequence number provides ordering information for a sequence of datagram's. The sequence number is not repeated even if a packet is retransmitted.

Authentication data: finally, the authentication data field is the result of applying a hash function to the entire IP datagram except for the fields that are changed during transmit.



ESP protocol:

Due to the limitations of the authentication header IPSec defined an alternative protocol that provides source authentication and integrity and privacy called Encapsulating Security Payload (ESP).

ESP fields:

Security Parameter Index (SPI):

➢ It is a 32 bit field. It is used in combination with source and destination address

Sequence number

➢ It is a 32 bit field used to prevent replay attack.

Payload data:

➢ It is variable length field. It contains the transport layer segment or ip packet.

Padding:

➢ This field contains padding bits (if any).

➢ These bits are mainly used in encryption algorithm.

Pad length:

- ➢ It is an 8 bit field. It indicates the number of bytes padded in the previous field.
- ➢ It is reserved bits for next header.

Next header:

- ➢ It is an 8 bit field. It indicates the type of data content in the payload data field.

Authentication Data:

- ➢ It is a variable length field. It contains the ICV(integrity check value)