SET-1

1. Explain briefly about Security Attacks and Security Services

Ans

SECURITY ATTACK:

Any action that compromises the security of information owned by an organization. There are four general categories of attack which are listed below.

Interruption

An asset of the system is destroyed or becomes unavailable or unusable. This is an attack on availability.

e.g., destruction of piece of hardware, cutting of a communication line or disabling of file management system.

Figure.2a

Interception

An unauthorized party gains access to an asset. This is an attack on confidentiality. Unauthorized party could be a person, a program or a computer. e.g., wiretapping to capture data in the network, illicit copying of files.

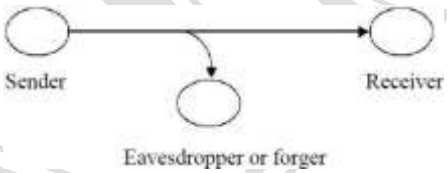


Figure. 2b

Modification

An unauthorized party not only gains access to but tampers with an asset. This is an attack on integrity.

e.g., changing values in data file, altering a program, modifying the contents of messages being transmitted in a network.

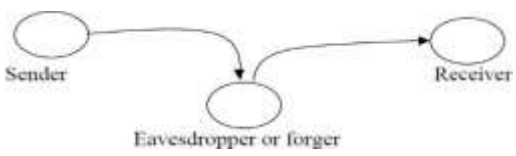


Figure. 2c

Fabrication

An unauthorized party inserts counterfeit objects into the system. This is an attack on authenticity.

e.g., insertion of spurious message in a network or addition of records to a file.

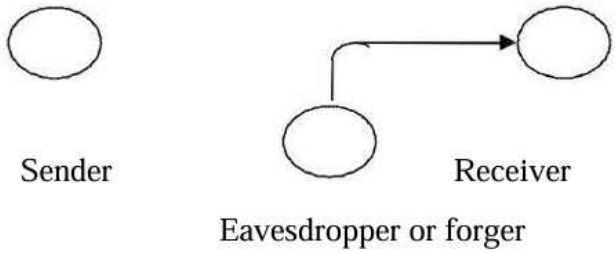


Figure.2d

The attack is majorly classified into two types:

- Active attack
- Passive Attack

PASSIVE ATTACK:

Passive attacks (Fig.3) are in the nature of eavesdropping on, or monitoring of, transmissions. The goal of the opponent is to obtain information that is being transmitted.

Passive attacks are of two types:

Release of message contents: A telephone conversation, an e-mail message and a transferred file may contain sensitive or confidential information. We would like to prevent the opponent from learning the contents of these transmissions.

Traffic analysis: If we had encryption protection in place, an opponent might still be able to observe the pattern of the message. The opponent could determine the location and identity of communication hosts and could observe the frequency and length of messages being exchanged. This information might be useful in guessing the nature of communication that was taking place.

Passive attacks are very difficult to detect because they do not involve any alteration of data. However, it is feasible to prevent the success of these attacks.

ACTIVE ATTACKS:

These attacks involve some modification of the data stream or the creation of a false stream.

These attacks can be classified in to four categories:

Masquerade – One entity pretends to be a different entity.

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

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

Denial of service – Prevents or inhibits the normal use or management of communication facilities. Another form of service denial is the disruption of an entire network, either by disabling the network or overloading it with messages so as to degrade performance. 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.

SECURITY SERVICE: A processing or communication service that enhances the security of the data processing systems and the information transfers of an organization. The services are intended to counter security attacks, and they make use of one or more security mechanisms to provide the service.

X.800 defines a security service as a service that is provided by a protocol layer of communicating open systems and that ensures adequate security of the systems or of data transfers. X.800 divides these services into five categories

Confidentiality: Ensures that the information in a computer system and transmitted information are accessible only for reading by authorized parties.

Eg., printing, displaying and other forms of disclosure.

Authentication: Ensures that the origin of a message or electronic document is correctly identified, with an assurance that the identity is not false.

Integrity: Ensures that only authorized parties are able to modify computer system assets and transmitted information. Modification includes writing, changing status, deleting, creating and delaying or replaying of transmitted messages.

Non repudiation: Requires that neither the sender nor the receiver of a message be able to deny the transmission.

Access control: Requires that access to information resources may be controlled by or the target system.

Availability: Requires that computer system assets be available to authorized parties when needed.

2. Explain in detail about Model for Network Security

Ans

NETWORK SECURITY MODEL:

A model for a network security is shown in the below figure. 5

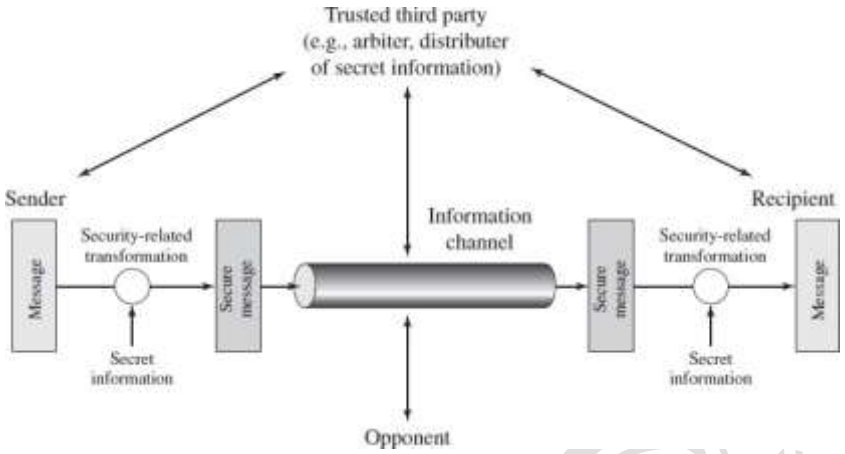


Figure.5 Network Security Model

A message is to be transferred from one party to another across some sort of Internet service. The two parties, who are the principals in this transaction, must cooperate for the exchange to

take place. A logical information channel is established by defining a route through the Internet from source to destination and by the cooperative use of communication protocols (e.g., TCP/IP) by the two principals.

Security aspects come into play when it is necessary or desirable to protect the information transmission from an opponent who may present a threat to confidentiality, authenticity, and so on. All the techniques for providing security have two components:

- A security-related transformation on the information to be sent. Examples include the encryption of the message, which scrambles the message so that it is unreadable by the opponent, and the addition of a code based on the contents of the message, which can be used to verify the identity of the sender.

- Some secret information shared by the two principals and, it is hoped, unknown to the opponent. An example is an encryption key used in conjunction with the transformation to scramble the message before transmission and unscramble it on reception.

A trusted third party may be needed to achieve secure transmission. For example, a third party may be responsible for distributing the secret information to the two principals while keeping it from any opponent. Or a third party may be needed to arbitrate disputes between the two principals concerning the authenticity of a message transmission.

This general model shows that there are four basic tasks in designing a particular security service:

1. Design an algorithm for performing the security-related transformation. The algorithm should be such that an opponent cannot defeat its purpose.

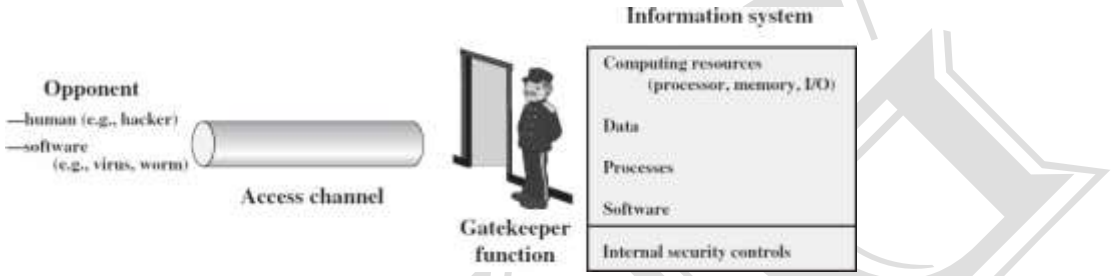
2. Generate the secret information to be used with the algorithm.

3. Develop methods for the distribution and sharing of the secret information.

4. Specify a protocol to be used by the two principals that makes use of the security algorithm and the secret information to achieve a particular security service.

However, there are other security-related situations of interest that do not neatly fit this model but are considered. A general model of these other situations is illustrated in Figure.6 which reflects a concern for protecting an information system from unwanted access.

Figure.6 Network Access Security Model



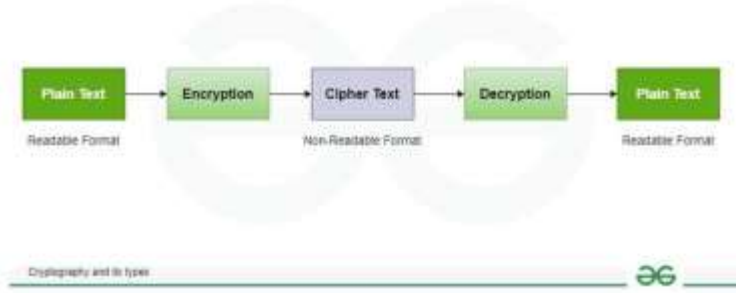
Another type of unwanted access is the placement in a computer system of logic that exploits vulnerabilities in the system and that can affect application programs as well as utility programs, such as editors and compilers. Programs can present two kinds of threats: Information access threats: Intercept or modify data on behalf of users who should not have access to that data.

Service threats: Exploit service flaws in computers to inhibit use by legitimate users

3a. Explain what is Cryptography

Ans

Cryptography is a technique of securing information and communications through the use of codes so that only those persons for whom the information is intended can understand and process it. Thus preventing unauthorized access to information. The prefix “crypt” means “hidden” and the suffix “graphy” means “writing”. In Cryptography, the techniques that are used to protect information are obtained from mathematical concepts and a set of rule-based calculations known as algorithms to convert messages in ways that make it hard to decode them. These algorithms are used for cryptographic key generation, digital signing, and verification to protect data privacy, web browsing on the internet and to protect confidential transactions such as credit card and debit card transactions.



Features Of Cryptography

- **Confidentiality:** Information can only be accessed by the person for whom it is intended and no other person except him can access it.
- **Integrity:** Information cannot be modified in storage or transition between sender and intended receiver without any addition to information being detected.
- **Non-repudiation:** The creator/sender of information cannot deny his intention to send information at a later stage.
- **Authentication:** The identities of the sender and receiver are confirmed. As well destination/origin of the information is confirmed.
- **Interoperability:** Cryptography allows for secure communication between different systems and platforms.
- **Adaptability:** Cryptography continuously evolves to stay ahead of security threats and technological advancements.

Types Of Cryptography

1. Symmetric Key Cryptography

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

Symmetric Key Cryptography

2. Hash Functions

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

3. Asymmetric Key Cryptography

In Asymmetric Key Cryptography, a pair of keys is used to encrypt and decrypt information. A receiver's public key is used for encryption and a receiver's private key is used for decryption. Public keys and Private keys are different. Even if the public key is known by everyone the intended receiver can only decode it because he alone knows his private key. The most popular asymmetric key cryptography algorithm is the RSA algorithm.

3b. Explain briefly about Substitution Techniques in Cryptography

Ans(answer is given in side by side pages)

SUBTITUTION TECHNIQUES

- The two basic building block of all the encryption techniques are substitution and transposition
- A substitution techniques is one in which the letter of plaintext are replaced by other letter or by number or symbols
- The subsitution techniques have a four techniques
  - caeser cipher
  - monoalphabetic cipher
  - play fair cipher
  - hill cipher
  - polyalphabetic cipher

Caeser cipher

2

- caeser cipher involves replacing each letter of the alphabet with the letter standing three place further down the alphabet

  plain : meet me after the to go party

  cipher: DREFR JUKHI TYRTY ELKJH VFDCB

  we can define transposition listing

  plain: a b c d e f g h i j k l m

  cipher :DE F G H I K L M N O P Q R S T U

- The algorithm can be expressed:

  plain text p,subsitution the cipher text c2

  $c=E(3,p)=(p+3)\bmod 26$

  gentral caeser algorithm:

  $c=E(k,p)=(p+k)\bmod 26$

3

three important characteristics of this problem

1.encryption and decryption algorithm

2.there are 25 key

3.the language are plaintext

the text file comprressed using algorithm called ZIP

EX:BRUTE FORCE CRYPTANALYSIS OF CAESER CIPHER

PHHW	PH	WKH
OGGU	OG	VJG

MONOALPHABETIC CIPHER

4

- The can 25 key possible key ,caeser cipher is far from secure,
- The are 26 alphabetic characters ,10 order of magnitude greater then key space DES
- this also called monoalphabetic subsitution cipher
- Ex:

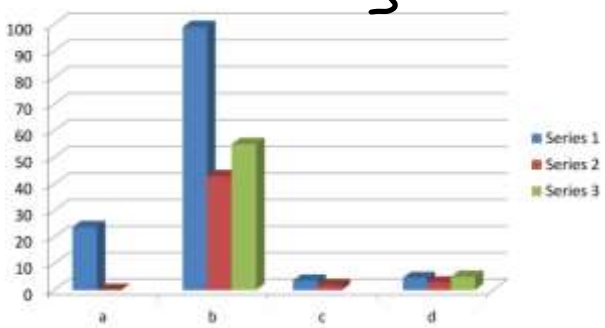
  NYUTGHREDSWACXZFGHJKLIOLPMNBGTYREFCVXD

  LOIKUJYTRGFDCVBHNUYTREWASEDXZCDSFREDFVBNMKOLP

The realtive frequency on cipher text in (percentage)

P 13.33	F 3.33
Z 11.67	W 3.33
S 8.33	Q 2.50
U 8.33	T 2.50
O 7.50	

5



6

- Monoalphabet cipher reficect frequency data original alphabet
- Multiple subsitution know as homophones
- EX: different cipher models

  16,17,35,21 homophone using rotation randomely

PLAYFAIR CIPHER 7

the multiple letter encryption is play Fair

The play fair algorithm is based on used construer using keyword

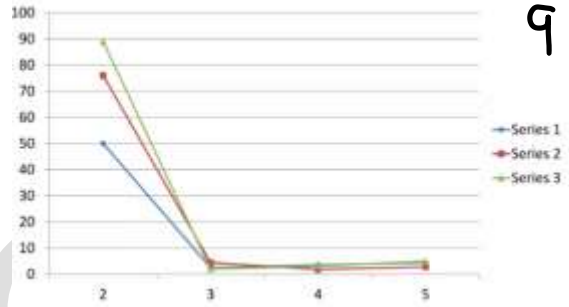
THE KEYWORD "MONOARCHY"

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

- THE FOLLOWING RUIE IN PLAY FAIR: 8
- 1. repeating plaintext letter are the same pair filltering letter
- 2. two plaintext letter in same row of the matrix are replaced the letter to the right
- 3.top the element colum circulary from the last

- Play fair text easy to break

- Ex:relative frequency of occurrence of letters:



9

HILL CIPHER 10

.The multiletter cipher hill cipher developed by the mathematician Lester hill in 1929.

. The algorithm m plaintext letter substitution cipher text m ,numerical value(a=0,b=1,....z=25)

c1=(k11p1+k12p2+k13p3)mod 26

c2=(k21p1+k22p2+k23p3)mod 26

c3=(k31p1+k32p2+k33p3)mod 26

the column of vectors:

$$\begin{pmatrix} c1 \\ c2 \\ c3 \end{pmatrix} \quad \begin{pmatrix} k11 & k12 & k13 \\ k21 & k22 & k23 \\ k31 & k32 & k33 \end{pmatrix} \quad \begin{pmatrix} p1 \\ p2 \\ p3 \end{pmatrix} \quad \text{mod } 26$$

- c=kp mod 26
- c and p colum of vectorsog length 3,plaintext cipher text and k represent encryption key
- Ex: 11

$$k=\begin{pmatrix} 17 & 17 & 5 \\ 21 & 18 & 21 \\ 2 & 2 & 19 \end{pmatrix} \quad \begin{pmatrix} 15 \\ 0 \\ 24 \end{pmatrix} \quad \begin{pmatrix} 375 \\ 819 \\ 486 \end{pmatrix} \quad \text{mod } 26 \quad \begin{pmatrix} 11 \\ 13 \\ 18 \end{pmatrix}$$

The square matrix determinate matrix equals sum of all the product can be one element each colum from each row on element .

EX: matrix

$$\begin{pmatrix} k11 & k12 \\ k21 & k22 \end{pmatrix}$$

POLALPHABETIC CIPHER 12

- The general name approach is polyalphabetic subsitution cipher
- the feature:
  1. monoalphabetic rule is used
  2. choose given transformation

key:meetmemeetmemeetme

plaintext:ewrdsfhthyujkikllolol

ciphertext:ZCSDEFRGTHBVNJUIKMNK

The ciphertext row determined column and plain text top of the column

vigenere proposed is refereed to auto key system

a	b	c	d	e
A	B	C	D	E
B	C	D	E	F
C	D	E	F	G
D	E	F	G	H

13

- This system works on binary data ,the system expressed follows:

ci=pi+ki 14

Where:

pi=binary digit plaintext

ki=binary digit key

ci=binary digit cipher text

4a.Explain what is Transposition Techniques
4b.Explain any one Transposition Technique briefly

Ans :answer for both a and b

Transposition Cipher Technique

The Transposition Cipher Technique is an encryption method used to encrypt a message or information. This encryption method is done by playing with the position of letters of the plain text. The positions of the characters present in the plaintext are rearranged or shifted to form the ciphertext. It makes use of some kind of permutation function to achieve the encryption purpose. It is very easy to use and so simple to implement.

Types of Transposition Cipher Techniques

There are three types of transposition cipher techniques

- Rail Fence Transposition Cipher
- Block (Single Columnar) Transposition Cipher
- Double Columnar Transposition Cipher

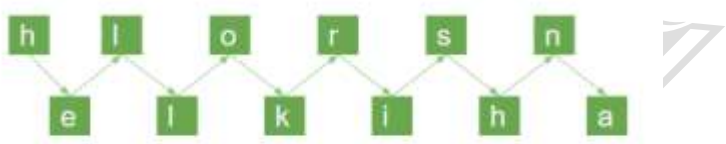
Rail Fence Transposition Cipher

Rail Fence Transposition cipher technique is the simplest transposition cipher technique**its**. It is also termed as a zigzag cipher. It gets its name from the way through which it performs encryption of plain text. The steps to get cipher text with the help of the Rail Fence Transposition cipher technique are as follow-

Technique of Rail Fence Transposition Cipher

Example: The plain text is “Hello Krishna”

Now, we will write this plain text in the diagonal form:



Rail Fence Transposition Cipher

Now, following the second step we get our cipher text.

Cipher Text = “rsnelkiha”

Block (Single Columnar) Transposition Cipher

Block Transposition Cipher is another form of Transposition Cipher which was used to encrypt the message or information. In this technique, first, we write the message or plaintext in rows. After that, we read the message column by column. In this technique, we use a keyword to determine the no of rows.

- Step 1: First we write the message in the form of rows and columns, and read the message column by column.
- Step 2: Given a keyword, which we will use to fix the number of rows.
- Step 3: If any space is spared, it is filled with null or left blank or in by (_).
- Step 4: The message is read in the order as specified by the keyword.

Block Columnar Transposition Cipher

For example: The plaintext is “KRISHNA RANJAN”
Now we will write the plaintext in the form of row and column.
Cipher Text = IAN_RNANS_J_KHRA

Double Columnar Transposition Cipher

Double Columnar Transposition Cipher is another form of Transposition Cipher Technique. It is just similar to the columnar transposition technique. The main objective of using a Double Columnar Transposition Cipher is to encrypt the message twice. It makes use of the Single Columnar Transposition technique but uses two times. It can use the same or different secret keys. The output obtained from the first encryption will be the input to the second encryption.

- Step 1: First we write the message in the form of rows and columns, and read the message column by column.
- Step 2: Given a keyword, which we will use to fix the number of rows.

Double Columnar Transposition Cipher:Step 1

- Step 3: If any space is spared, it is filled with null or left blank or in by (_).

Now applying keyword 2:

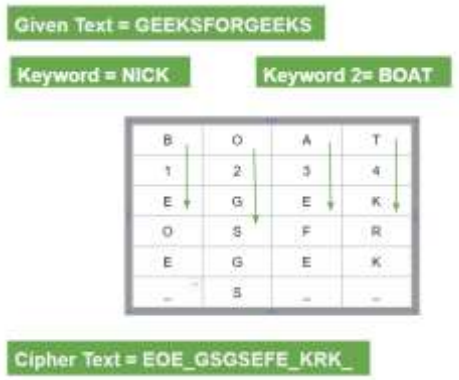
Double Columnar Transposition Cipher: Step 2

- Step 4: The message is read in the order in by the keyword.

Now apply step 3:

Double Columnar Transposition Cipher: Step 3

- Step 5: Then the output from the first encryption is input to the second.
- Step 6: Now the message is read in Technique in the order specified by the second keyword.



Double Columnar Transposition Cipher: Step 4

The Cipher Text is: **“S_J_IAN_RNANKHRA”**

5. Explain in detail about DES Symmetric encryption Algorithm

Ans

Youtube Explanations:

https://youtu.be/j53iXhTSi_s?si=mHXGVBIq2oOT4p1F

https://youtu.be/jTyilHC51_w?si=YWSxQSYp7oaNOmqk

The DES (Data Encryption Standard) algorithm is a symmetric-key block cipher created in the early 1970s by an IBM team and adopted by the National Institute of Standards and

Technology (NIST). The algorithm takes the plain text in 64-bit blocks and converts them into ciphertext using 48-bit keys.

Since it's a symmetric-key algorithm, it employs the same key in both encrypting and decrypting the data. If it were an asymmetrical algorithm, it would use different keys for encryption and decryption.

Initial Permutation (IP)
The plain text is divided into smaller chunks of 64-bit size. The IP is performed before the first round. This phInitial Permutation (IP)

The plain text is divided into smaller chunks of 64-bit size. The IP is performed before the first round. This phase describes the implementation of the transposition process. For example, the 58th bit replaces the first bit, the 50th bit replaces the second bit, and so on. The resultant 64-bit text is split into two equal halves of 32-bit each called Left Plain Text (LPT) and Right Plain Text (RPT).

Step 1: Key Transformation

We already know that the DES process uses a 56-bit key, which is obtained by eliminating all the bits present in every 8th position in a 64-bit key. In this step, a 48-bit key is generated. The 56-bit key is split into two equal halves and depending upon the number of rounds the bits are shifted to the left in a circular fashion.

Due to this, all the bits in the key are rearranged again. We can observe that some of the bits get eliminated during the shifting process, producing a 48-bit key. This process is known as compression permutation.

Step 2: Expansion Permutation

Let's consider an RPT of the 32-bit size that is created in the IP stage. In this step, it is expanded from 32-bit to 48-bit. The RPT of 32-bit size is broken down into 8 chunks of 4 bits each and extra two bits are added to every chunk, later on, the bits are permutated among themselves leading to 48-bit data. An XOR function is applied in between the 48-bit key obtained from step 1 and the 48-bit expanded RPT.

ase describes the implementation of the transposition process. For example, the 58th bit replaces the first bit, the 50th bit replaces the second bit, and so on. The resultant 64-bit text is split into two equal halves of 32-bit each called Left Plain Text (LPT) and Right Plain Text (RPT).

Step 1: Key Transformation

We already know that the DES process uses a 56-bit key, which is obtained by eliminating all the bits present in every 8th position in a 64-bit key. In this step, a 48-bit key isgenerated. The 56-bit key is split into two equal halves and depending upon the number of rounds the bits are shifted to the left in a circular fashion.

Due to this, all the bits in the key are rearranged again. We can observe that some of the bits get eliminated during the shifting process, producing a 48-bit key. This process is known as compression permutation.

Step 2: Expansion Permutation

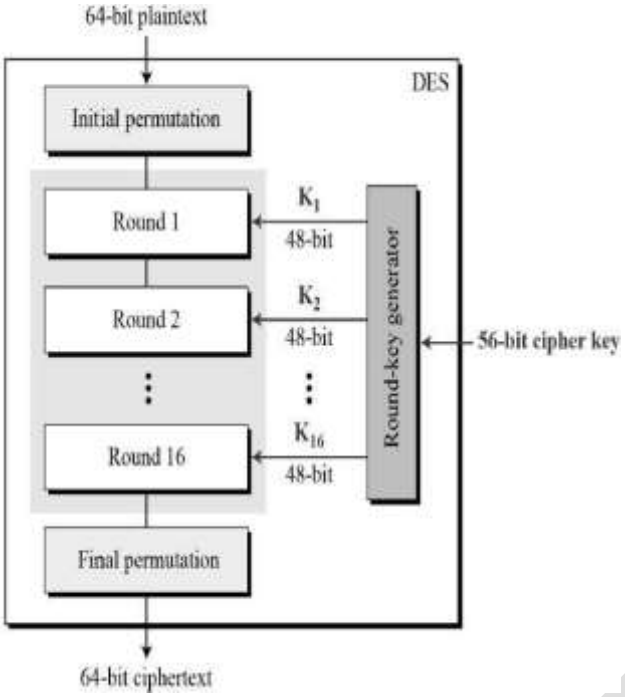
Let's consider an RPT of the 32-bit size that is created in the IP stage. In this step, it is expanded from 32-bit to 48-bit. The RPT of 32-bit size is broken down into 8 chunks of 4 bits each and extra two bits are added to every chunk, later on, the bits are permutated among themselves leading to 48-bit data. An XOR function is applied in between the 48-bit key obtained from step 1 and the 48-bit expanded RPT.

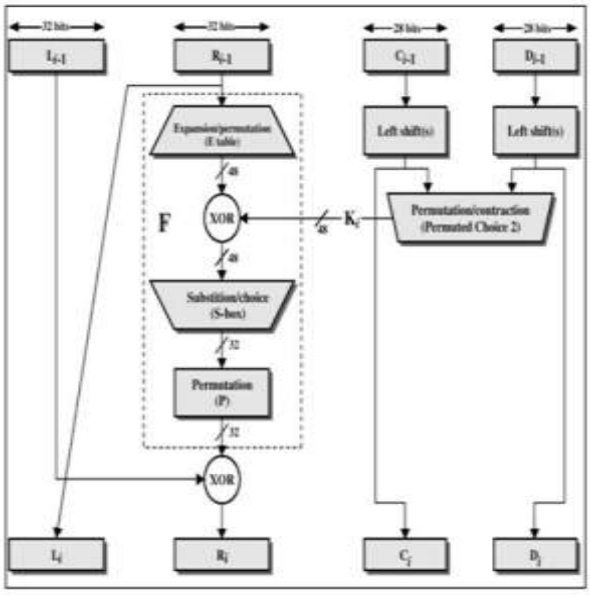
DES Algorithm steps:

The algorithm process breaks down into the following steps:

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

Single Round of Execution of DES Algorithm:





Orr

Data Encryption Standard (DES) is a block cipher with a 56-bit key length that has played a significant role in data security**.** Data encryption standard (DES) has been found vulnerable to very powerful attacks therefore, the popularity of DES has been found slightly on the decline. DES is a block cipher and encrypts data in blocks of size of **64 bits** each, which means 64 bits of plain text go as the input to DES, which produces 64 bits of ciphertext. The same algorithm and key are used for encryption and decryption, with minor differences. The key length is **56 bits**.

The basic idea is shown below:

We have mentioned that DES uses a 56-bit key. Actually, The initial key consists of 64 bits. However, before the DES process even starts, every 8th bit of the key is discarded to produce a 56-bit key. That is bit positions 8, 16, 24, 32, 40, 48, 56, and 64 are discarded.

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

Figure - discording of every 8th bit of original key

Thus, the discarding of every 8th bit of the key produces a **56-bit key** from the original **64-bit key**.

DES is based on the two fundamental attributes of cryptography: substitution (also called confusion) and transposition (also called diffusion). DES consists of 16 steps, each of which is called a round. Each round performs the steps of substitution and transposition. Let us now discuss the broad-level steps in DES.

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

Broad Level Steps in DES

Initial Permutation (IP)

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

This is nothing but jugglery of bit positions of the original plain text block. the same rule applies to all the other bit positions shown in the figure.

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	33	45	37	29	21	13	5	63	55	47	39	31	23	15	7

Figure - Initial permutation table

As we have noted after IP is done, the resulting 64-bit permuted text block is divided into two half blocks. Each half-block consists of 32 bits, and each of the 16 rounds, in turn, consists of the broad-level steps outlined in the figure.

Step 1: Key transformation

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

For example: if the round numbers 1, 2, 9, or 16 the shift is done by only one position for other rounds, the circular shift is done by two positions. The number of key bits shifted per round is shown in the figure.

Round	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
#key bits shifted	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Figure - number of key bits shifted per round

After an appropriate shift, 48 of the 56 bits are selected. From the 48 we might obtain 64 or 56 bits based on requirement which helps us to recognize that this model is very versatile and can handle any range of requirements needed or provided. for selecting 48 of the 56 bits the table is shown in the figure given below. For instance, after the shift, bit number 14 moves to the first position, bit number 17 moves to the second position, and so on. If we observe the table , we will realize that it contains only 48-bit positions. Bit number 18 is discarded (we will not find it in the table), like 7 others, to reduce a 56-bit key to a 48-bit key. Since the key transformation process involves permutation as well as a selection of a 48-bit subset of the original 56-bit key it is called Compression Permutation.

14	17	11	24	1	5	3	28	15	6	21	10
23	19	12	4	26	8	16	7	27	20	13	2
41	52	31	37	47	55	30	40	51	45	33	48
44	49	39	56	34	53	46	42	50	36	29	32

Figure - compression permutation

Because of this compression permutation technique, a different subset of key bits is used in each round. That makes DES not easy to crack.

Step 2: Expansion Permutation

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

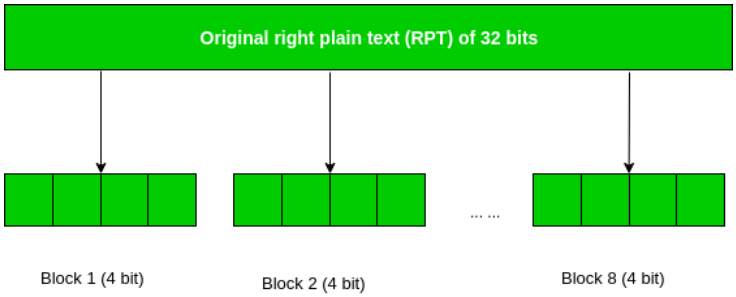


Figure - division of 32 bit RPT into 8 bit blocks

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

6.Explain about RSA Algorithm in detail?

Ans

Youtube Explanations: https://youtu.be/JFQAHDOHjfM?si=hGd5b46ZgRozMci0

RSA Encryption Algorithm

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

Public key encryption algorithm:

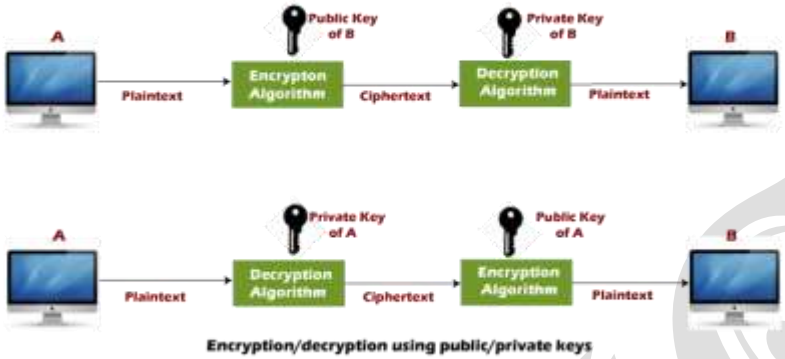
Public Key encryption algorithm is also called the Asymmetric algorithm. Asymmetric algorithms are those algorithms in which sender and receiver use different keys for encryption and decryption. Each sender is assigned a pair of keys:

- **Public key**
- **Private key**

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

derived from the public key. The public key is well known, but the private key is secret and it is known only to the user who owns the key. It means that everybody can send a message to the user using user's public key. But only the user can decrypt the message using his private key.

The Public key algorithm operates in the following manner:

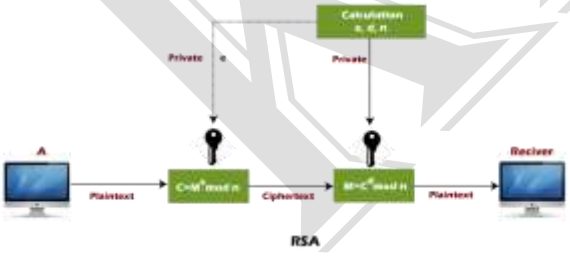


Encryption/decryption using public/private keys

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

RSA encryption algorithm:

RSA is the most common public-key algorithm, named after its inventors **Rivest, Shamir, and Adelman (RSA).**



RSA

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

- Select two large prime numbers, p and **q**.
- Multiply these numbers to find **n = p x q,** where **n** is called the modulus for encryption and decryption.
- Choose a number **e** less than **n**, such that n is relatively prime to **(p - 1) x (q -1).** It means that **e** and **(p - 1) x (q - 1)** have no common factor except 1. Choose "e" such

that 1<e < φ (n), e is prime to φ (n), **gcd (e,d(n)) =1**

- If **n = p x q,** then the public key is <e, n>. A plaintext message **m** is encrypted using public key <e, n>. To find ciphertext from the plain text following formula is used to getciphertextC.

  $C=m^e$ modn

  Here**, m** must be less than **n**. A larger message (>n) is treated as a concatenation of messages, each of which is encrypted separately.
- To determine the private key, we use the following formula to calculate the d such that:

  D_e mod{(p-1)x(q-1)}=1

  Or

  D_e mod φ (n) = 1
- The private key is <d, n>. A ciphertext message **c** is decrypted using private key <d, n>. To calculate plain text **m** from the ciphertext c following formula is used to get plaintextm.

  $m = c^d$ mod n

Let's take some example of RSA encryption algorithm:

Example 1:

This example shows how we can encrypt plaintext 9 using the RSA public-key encryption algorithm. This example uses prime numbers 7 and 11 to generate the public and private keys.

Explanation:

Step 1: Select two large prime numbers, p, and **q**.

p = 7

q = 11

Step 2: Multiply these numbers to find **n = p x q,** where **n** is called the modulus for encryption and decryption.

First, we calculate

n = p x q

n = 7 x 11

n = 77

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

Second, we calculate

φ (n) = (p - 1) x (q-1)

φ (n) = (7 - 1) x (11 - 1)

φ (n) = 6 x 10

φ (n) = 60

Let us now choose relative prime e of 60 as 7.

Thus the public key is <e, n> = (7, 77)

Step 4: A plaintext message **m** is encrypted using public key <e, n>. To find ciphertext from the plain text following formula is used to get ciphertext C.

To find ciphertext from the plain text following formula is used to get ciphertext C.

$C = m^e \bmod n$

$C = 9^7 \bmod 77$

C = 37

Step 5: The private key is <d, n>. To determine the private key, we use the following formula d such that:

$D_e \bmod \{(p - 1) \times (q - 1)\} = 1$

7d mod 60 = 1, which gives d = 43

The private key is <d, n> = (43, 77)

Step 6: A ciphertext message **c** is decrypted using private key <d, n>. To calculate plain text **m** from the ciphertext c following formula is used to get plain text m.

$m = c^d \bmod n$

$m = 37^{43} \bmod 77$

m = 9In this example, Plain text = 9 and the ciphertext = 37

SET-2

1.What is Need for Security and Principles of Security explain in detail

Ans

Need for Security

The "Need for Security" refers to the reasons and motivations behind implementing security measures to protect information systems. Several factors drive this need:

1. **Protection of Confidentiality**: Ensuring that sensitive information is only accessible to authorized individuals. For example, personal data, financial records, and proprietary business information must be kept secret to prevent unauthorized access.
2. **Integrity of Data**: Safeguarding data from being altered or tampered with by unauthorized parties. Maintaining data integrity ensures that information remains accurate and trustworthy over time.
3. **Availability**: Ensuring that information and resources are available to authorized users when needed. This involves protecting systems from disruptions, such as denial-of-service attacks or hardware failures.
4. **Compliance**: Meeting legal, regulatory, and contractual obligations regarding data protection. Various laws and regulations mandate how organizations must handle sensitive data, such as GDPR in the EU or HIPAA in the US.
5. **Protection Against Threats**: Addressing the risks posed by various threats, including cyberattacks, malware, and insider threats. Security measures help mitigate the impact of these threats.
6. **Preservation of Trust**: Maintaining the trust of customers, clients, and stakeholders. Effective security practices demonstrate a commitment to protecting sensitive information and can enhance an organization's reputation.

The Principles of Security can be classified as follows:

1. Confidentiality:

The degree of confidentiality determines the secrecy of the information. The principle specifies that only the sender and receiver will be able to access the information shared between them. Confidentiality compromises if an unauthorized person is able to access a message.

For example, let us consider sender A wants to share some confidential information with receiver B and the information gets intercepted by the attacker C. Now the confidential information is in the hands of an intruder C.

2. Authentication:

Authentication is the mechanism to identify the user or system or the entity. It ensures the identity of the person trying to access the information. The authentication is mostly secured by using username and password. The authorized person whose identity is preregistered can prove his/her identity and can access the sensitive information.

3. Integrity:

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

System Integrity: System Integrity assures that a system performs its intended function in an unimpaired manner, free from deliberate or inadvertent unauthorized manipulation of the system.

Data Integrity: Data Integrity assures that information (both stored and in transmitted packets) and programs are changed only in a specified and authorized manner.

4. Non-Repudiation:

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

5. Access control:

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

6. Availability:

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

7. Issues of ethics and law

The following categories are used to categorize ethical dilemmas in the security system.

Individuals' right to access personal information is referred to as privacy.

Property: It is concerned with the information's owner.

Accessibility is concerned with an organization's right to collect information.

Accuracy: It is concerned with the obligation of information authenticity, fidelity, and accuracy.

2. Explain briefly about Security Threats briefly.

Ans

Computer security threats are potential threats to your computer's efficient operation and performance. These could be harmless adware or dangerous trojan infection. As the world becomes more digital, computer security concerns are always developing. A threat in a computer system is a potential danger that could jeopardize your data security. At times, the damage is irreversible.

Types of Threats:

A security threat is a threat that has the potential to harm computer systems and organizations. The cause could be physical, such as a computer containing sensitive information being stolen. It's also possible that the cause isn't physical, such as a viral attack.

1. Physical Threats: A physical danger to computer systems is a potential cause of an occurrence/event that could result in data loss or physical damage. It can be classified as:

- **Internal:** Short circuit, fire, non-stable supply of power, hardware failure due to excess humidity, etc. cause it.
- **External:** Disasters such as floods, earthquakes, landscapes, etc. cause it.
- **Human:** Destroying of infrastructure and/or hardware, thefts, disruption, and unintentional/intentional errors are among the threats.

2. Non-physical threats: A non-physical threat is a potential source of an incident that could result in:

- Hampering of the business operations that depend on computer systems.
- Sensitive – data or information loss
- Keeping track of other's computer system activities illegally.
- Hacking id & passwords of the users, etc.

The non-physical threads can be commonly caused by:

(i) Malware: Malware ("malicious software") is a type of computer program that infiltrates and damages systems without the users' knowledge. Malware tries to go unnoticed by either hiding or not letting the user know about its presence on the system. You may notice that your system is processing at a slower rate than usual.

(ii) Virus: It is a program that replicates itself and infects your computer's files and programs, rendering them inoperable. It is a type of malware that spreads by inserting a copy of itself into and becoming part of another program. It spreads with the help of software or documents. They are embedded with software and documents and then transferred from one computer to another using the network, a disk, file sharing, or infected e-mail. They usually appear as an executable file.

(iii) Spyware: Spyware is a type of computer program that tracks, records, and reports a user's activity (offline and online) without their permission for the purpose of profit or data theft. Spyware can be acquired from a variety of sources, including websites, instant chats, and emails. A user may also unwittingly obtain spyware by adopting a software program's End User License Agreement.

Adware is a sort of spyware that is primarily utilized by advertising. When you go online, it keeps track of your web browsing patterns in order to compile data on the types of websites you visit.

(iv) Worms: Computer worms are similar to viruses in that they replicate themselves and can inflict similar damage. Unlike viruses, which spread by infecting a host file, worms are freestanding programs that do not require a host program or human assistance to proliferate. Worms don't change programs; instead, they replicate themselves over and over. They just eat resources to make the system down.
(v) Trojan: A Trojan horse is malicious software that is disguised as a useful host program. When the host program is run, the Trojan performs a harmful/unwanted action. A Trojan horse, often known as a Trojan, is malicious malware or software that appears to be legal yet has the ability to take control of your computer. A Trojan is a computer program that is designed to disrupt, steal, or otherwise harm your data or network.
(vi) Denial Of Service Attacks: A Denial of Service attack is one in which an attacker tries to prohibit legitimate users from obtaining information or services. An attacker tries to make a system or network resource unavailable to its intended users in this attack. The web servers of large organizations such as banking, commerce, trading organizations, etc. are the victims.
(vii) Phishing: Phishing is a type of attack that is frequently used to obtain sensitive information from users, such as login credentials and credit card details. They deceive users into giving critical information, such as bank and credit card information, or access to personal accounts, by sending spam, malicious Web sites, email messages, and instant chats.
(viii) Key-Loggers: Keyloggers can monitor a user's computer activity in real-time. Keylogger is a program that runs in the background and records every keystroke made by a user, then sends the data to a hacker with the intent of stealing passwords and financial information.

3. Discuss about Encryption and Decryption and Types of Encryption with examples

Ans

Encryption and Decryption

- **Encryption**:
    - **Definition**: The process of converting plaintext (readable data) into ciphertext (unreadable data) using an algorithm and a key. The purpose is to protect the data's confidentiality by making it unreadable to unauthorized users.
    - **Process**: An encryption algorithm takes the plaintext and a key as input and produces ciphertext. The key is a piece of information used by the algorithm to perform the transformation.
- **Decryption**:
    - **Definition**: The process of converting ciphertext back into plaintext using an algorithm and a key. This process is the reverse of encryption and makes the data readable again to authorized users.
    - **Process**: A decryption algorithm takes the ciphertext and a key as input and restores the original plaintext. The key used for decryption is either the same as or different from the encryption key, depending on the encryption method.

Types of Data Encryption

There are multiple encryption techniques, each of which have been developed with various security requirements in mind. Symmetric and Asymmetric encryption are the two types of data encryption.

1. Symmetric Key Encryption

There are a few strategies used in cryptography algorithms. For encryption and decryption processes, some algorithms employ a unique key. In such operations, the unique key must be secured since the system or person who knows the key has complete authentication to decode the message for reading. This approach is known as “symmetric encryption” in the field of network encryption.

Symmetric Encryption

Examples:

- **AES (Advanced Encryption Standard)**: A widely used symmetric encryption algorithm that provides a high level of security and efficiency. For example, encrypting the message "HELLO" with AES using a specific key will produce ciphertext that can only be decrypted by someone with the same key.
- **DES (Data Encryption Standard)**: An older symmetric encryption algorithm that was widely used but is now considered less secure due to advances in computational power. For example, DES encrypts "HELLO" into a fixed-length block of ciphertext using a 56-bit key

2. Asymmetric Key Encryption

Some cryptography methods employ one key for data encryption and another key for data decryption. As a result, anyone who has access to such a public communication will be unable to decode or read it. This type of cryptography, known as “public-key” encryption, is used in the majority of internet security protocols. The term “asymmetric encryption” is used to describe this type of encryption.

Examples:

- **RSA (Rivest-Shamir-Adleman)**: One of the most widely used asymmetric encryption algorithms. For example, encrypting a message with a public RSA key ensures that only the holder of the corresponding private RSA key can decrypt and read the message.
- **ECC (Elliptic Curve Cryptography)**: Provides similar security to RSA but with shorter key lengths, making it more efficient. For example, ECC can encrypt "HELLO" with an elliptic curve public key, and only someone with the corresponding private key can decrypt it.

3. Hybrid Encryption

- **Definition**: Hybrid encryption combines both symmetric and asymmetric encryption to leverage the strengths of both methods. Typically, asymmetric encryption is used to securely exchange a symmetric key, which is then used for encrypting the actual data.
- **Characteristics**:
    - **Efficiency**: Provides the security benefits of asymmetric encryption for key exchange and the performance benefits of symmetric encryption for data encryption.
- **Examples**:
    - **TLS/SSL (Transport Layer Security/Secure Sockets Layer)**: These protocols use hybrid encryption to secure internet communications. They use asymmetric encryption to establish a secure connection and exchange a symmetric key, which is then used to encrypt the actual data transmitted over the connection.
    -

4.Discuss in detail about Block ciphers and Stream ciphers.

Ans

What is a Block Cipher?

A block cipher is a cryptographic algorithm that encrypts fixed-size data blocks, commonly 128 bits. It operates with a fixed key and encrypts data groups to ensure maximum confidentiality and security. Modern block ciphers are based on the iterated product cipher design, using a fixed key for encrypting data groups.

How does Block Cipher work?

In a block cipher, encryption occurs in fixed-length blocks, typically chunks of 128 bits. Block ciphers operate on larger data units than stream ciphers, which encrypt one byte at a time. A key, denoted as K, transforms each plaintext block into a corresponding ciphertext block.

Key features and benefits of the block cipher:

- **Block Size**

  Encryption is performed on fixed-size blocks, often consisting of 128 bits. Each block undergoes a series of transformations using the encryption key.

- **Key Expansion**

  The encryption key, K, is expanded to produce subkeys for each round of encryption, enhancing the security of the block cipher.

- **Confusion and Diffusion**

  Confusion involves complexing the relationship between the key and ciphertext. Diffusion ensures that a change in one part of the plaintext affects a large portion of the ciphertext.

- **Rounds of Transformation**

  Each block undergoes multiple rounds of transformation, each involving substitution, permutation, and mixing operations.

- **Cryptographic Security**

  Block ciphers are designed to resist various cryptographic attacks, including differential and linear cryptanalysis. The careful design of the cipher's structure achieves robust security.

- **Electronic Codebook (ECB) Mode**

  Each block is independently encrypted in ECB mode. Identical plaintext blocks yield identical ciphertext blocks.

- **Cipher Block Chaining (CBC) Mode**

  CBC mode introduces an initialization vector (IV) to enhance security. Before encryption, each block is XORed with the previous ciphertext block.

Example Encryption:

- The encryption process involves multiple rounds of substitution and permutation.
- Each block of plaintext is transformed into a corresponding block of ciphertext.

Example Decryption:

- Decryption reverses the encryption process, applying inverse transformations.
- Each block of ciphertext is decrypted to reveal the original plaintext.

Block ciphers are foundational in various cryptographic protocols, ensuring secure communication and data protection across diverse applications. Their fixed block size and complex encryption mechanisms contribute to their resilience against attacks.

Tailored Encryption Services

We assess, strategize & implement encryption strategies and solutions.

What is Stream Cipher?

Encryption is performed one byte at a time in a stream cipher, providing a continuous stream of pseudorandom bits for increased security. The process begins with initializing a key, denoted as k, which is fed into a pseudorandom bit generator. This generator produces an 8-bit keystream, serving as the foundation for encryption.

Key features and benefits of the stream cipher:

1. **Keystream Generation**
    - A key, k, is input into a pseudorandom bit generator.
    - The generator produces an 8-bit keystream, forming the basis for encryption.
2. **Cryptanalysis Resistance**
    - The sequential nature of stream ciphers enhances resistance against cryptanalysis.
    - Increasing the length of the keystream makes cryptanalysis more challenging.
3. **Brute Force Protection**
    - Longer keys contribute to resistance against brute force attacks.
    - Strengthening security is achieved by employing longer key lengths.
4. **Efficient Keystream Design**
    - Keystreams are designed for optimal efficiency, incorporating a balanced mix of 1s and 0s.
    - This design choice aims to heighten the complexity of cryptanalysis.
5. **Stream Cipher Operation**
    - **The encryption process**

      It involves XORing each plaintext bit with the corresponding bit in the keystream

      Example: Plain Text: 10011001, Keystream: 11000011, Cipher Text: 01011010

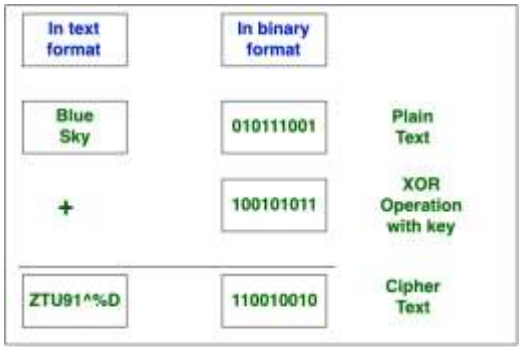
    - **Decryption process**

      Decryption mirrors the encryption process using the same keystream. XORing the ciphertext with the keystream yields the original plaintext.

      Example: Cipher Text: 01011010, Keystream: 11000011, Plain Text: 10011001

6. The stream cipher's simplicity, efficiency, and cryptographic strength make it suitable for various applications where real-time encryption and decryption are crucial.

Block Cipher and **Stream Cipher** belongs to the symmetric key cipher. These two block ciphers and stream cipher are the methods used for converting the plain text into ciphertext. The main difference between a **Block cipher** and a **Stream cipher** is that a block cipher converts the plain text into cipher text by taking plain text's block at a time. While stream cipher Converts the plain text into cipher text by taking 1 byte of plain text at a time.



7. Stream Cipher

5a. Explain in detail about IDEA Algorithm
5b. Explain the Single round of IDEA Algorithm.
Ans Answer for both a and b

Youtube Explanation: https://youtu.be/909pwcyqVFQ?si=k2v3F0pvqyjMVLvZ

IDEA (International Data Encryption Algorithm) is an encryption algorithm. It is a symmetric block cipher that takes 64 bit as an input, 28-bit key and performs 8 identical rounds for encryption in which 6 different subkeys are used, and four keys are used for output transformation.

Understanding IDEA Algorithm

- The typical block size is 16 bytes of 128 bits. A block cipher will typically operate in round blocks where part of the key is applied to the round, and then other operations are performed on it. After a certain number of rounds, say between 10 to 16, we end up with our ciphertext for that block.

- The ciphertext block is exactly the same size as the plain text block, 16 bytes. We operate on the block for each round using a part of the encryption key that we call as the round key. We derive the multiple round keys from the encryption key using a key schedule.
- The key schedule is an **algorithm that Shifts**, XORs, Multiplies and performs other types of operation on the original encryption key in order to come up with these round keys. Well, if I have a 16-byte block and I have a 128-bit key, which is also 16 bytes,

Confusion and Diffusion

Let's understand the difference between Confusion and Diffusion.

Confusion

- Confusion has to do with the relation between key and ciphertext.
- We ensure that a small change in the key leads to a large change in the ciphertext.
- XOR is not sufficient; one-to-one.
- Key Schedule.

Diffusion

- Diffusion has to do with the relation between the message and ciphertext.
- A small change in the message -> large change in the ciphertext.
- Hides patterns within the message.

So Electronic Code Book mode of operation, we will usually run a **block cipher in Cipher** Block Chaining mode of operation or CBC. With cipher block chaining, you will XOR the previous block's ciphertext with the previous block's with the next block's plain text

before you were encrypted. In that way, every block in the message depends upon all of the blocks that came before.

Data Encryption Standard (DES)

Let's look at some of the aspects of the Data Encryption Standard (DES).

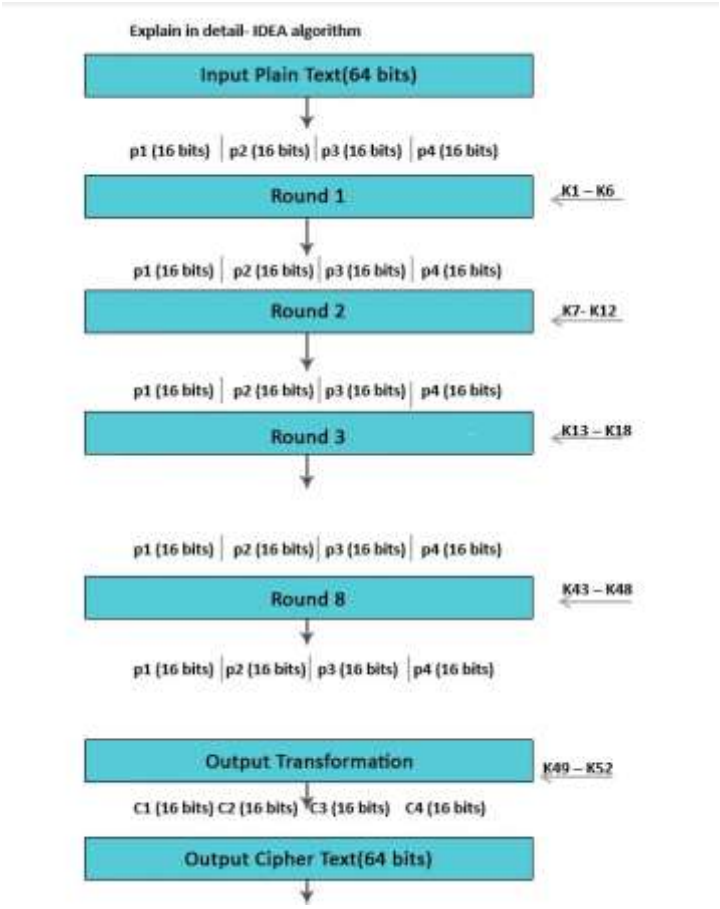
Key Length

- 64-bit input
- 8-bit parity check
- 56-bit effective key

Weakness

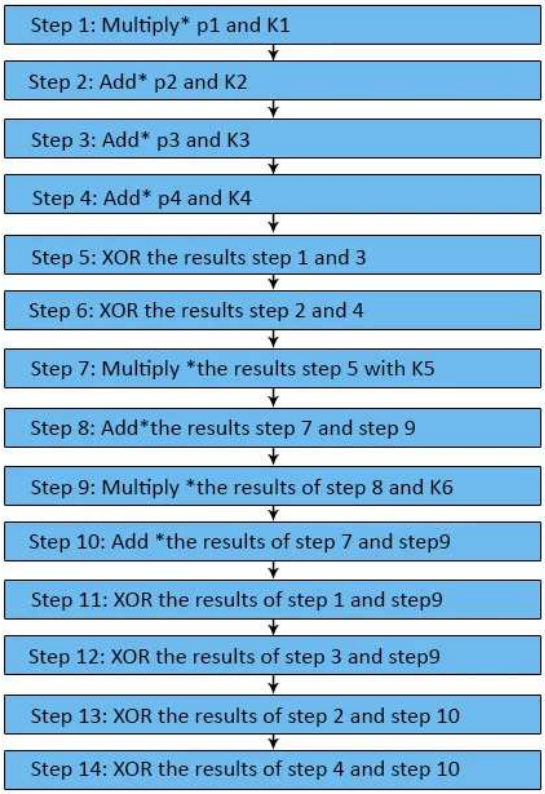
- Theoretical
- Short key

Understanding IDEA Algorithm in Detail



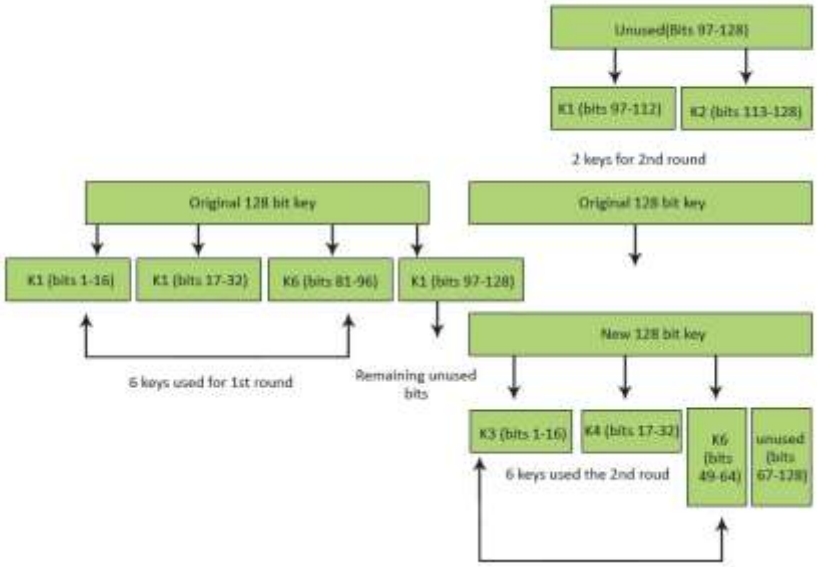
The 64-bit input plain text block-divided into 4 part (16 bits each) Declare p1 to p4.

- Therefore, p1 to p4 will be the inputs for the initial round of the algorithm.
- There are 8 such rounds.
- The key is made up of 128 bits.
- In each round, 6 sub-keys will be produced.
- Each one of the sub-keys includes 16 bits.
- All these sub-keys will be put on the 4 input blocks p1 to p4.
- The last actions include Output Transformation, which usually benefits simply 4 sub-Keys.
- The last result is 4 blocks of ciphertext C1 to C4 (each of 16 bits).
- They are mixed to create the last 64-bit ciphertext block.



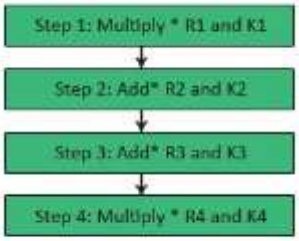
Single Round Information

- There are 8 rounds in IDEA.
- Every single requires several operations around the four data blocks applying 6 keys.
- These steps work in numerous mathematical activities.
- There are multiple *, add * & XOR procedures.
- Multiply * means multiplication modulo.
- Add* requires addition modulo.



Output Transformation

- It can be a one-time procedure.
- It requires places by the end of the 8th round.
- The Output transformation input is a 64-bit value divided into 4 sub-blocks (state R1 to R4 every among 16 bits).
- The four 16 bits Sub-keys (K1 to K4) are used here.
- The process of the outcome transformation can be as follows.



Conclusion

- IDEA may be a recognized cipher that many experts have examined for the previous 10 Sub-key creation for the round, each one of the 8 rounds utilizes 6 sub-keys (hence 8 * 6 = 48 sub-keys are essential for the rounds). The last result transformation benefits 4 sub-keys (i.e. 48 + 4 = 52 sub-keys total). From an input key 128 bits, all these 52 sub-keys will be produced years, as well as; however, no strike against five or higher of its 8.5 rounds has been found.
- Because of its toughness against cryptanalytic attacks and because of its inclusion in several well-known cryptographic deals, IDEA can be trusted. The Basic IDEA algorithm is definitely not, which can be likened for effectiveness or security with simple DES or AES versions. The Basic IDEA algorithm is intended to assist learners in being familiar with the IDEA algorithm by giving a version of IDEA that enables instances to get worked well manually and to offer a comparison of the technique of IDEA and the ways of DES and AES.

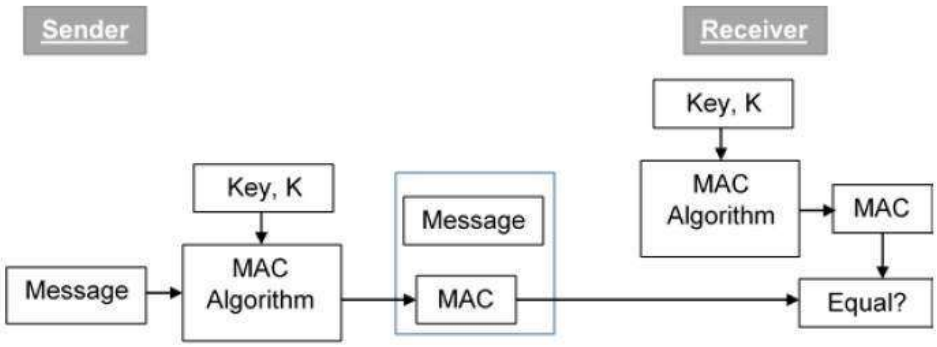
6a. Explain about Message Authentication
Ans
Youtube Explanation: https://youtu.be/Xb_84P8Y7JI?si=QZulZyzz_oAVT6t0

Message Authentication Code (MAC)

MAC algorithm is a symmetric key cryptographic technique to provide message authentication. For establishing MAC process, the sender and receiver share a symmetric key K.

Essentially, a MAC is an encrypted checksum generated on the underlying message that is sent along with a message to ensure message authentication.

The process of using MAC for authentication is depicted in the following illustration –



Let us now try to understand the entire process in detail –

- The sender uses some publicly known MAC algorithm, inputs the message and the secret key K and produces a MAC value.
- Similar to hash, MAC function also compresses an arbitrary long input into a fixed length output. The major difference between hash and MAC is that MAC uses secret key during the compression.
- The sender forwards the message along with the MAC. Here, we assume that the message is sent in the clear, as we are concerned of providing message origin authentication, not confidentiality. If confidentiality is required then the message needs encryption.
- On receipt of the message and the MAC, the receiver feeds the received message and the shared secret key K into the MAC algorithm and re-computes the MAC value.
- The receiver now checks equality of freshly computed MAC with the MAC received from the sender. If they match, then the receiver accepts the message and assures himself that the message has been sent by the intended sender.
- If the computed MAC does not match the MAC sent by the sender, the receiver cannot determine whether it is the message that has been altered or it is the origin that has been falsified. As a bottom-line, a receiver safely assumes that the message is not the genuine.

6b. Explain about HMAC algorithm
Ans
Youtube Explanation: https://youtu.be/HjgDdTKtohM?si=zPxPZVd6wl0qhPGW

HMAC (Hash-based Message Authentication Code) is a type of message authentication code (MAC) that is acquired by executing a cryptographic hash function on the data that is to be authenticated and a secret shared key. Like any of the MACs, it is used for both data integrity and authentication. In this article, we will discuss every point about HMAC.
What is HMAC?
HMAC algorithm stands for Hashed or Hash-based Message Authentication Code. It is a result of work done on developing a MAC derived from cryptographic hash functions. HMAC is a great resistance to cryptanalysis attacks as it uses the Hashing concept twice. HMAC consists of twin benefits of Hashing and MAC and thus is more secure than any other authentication code. RFC 2104 has issued HMAC, and HMAC has been made

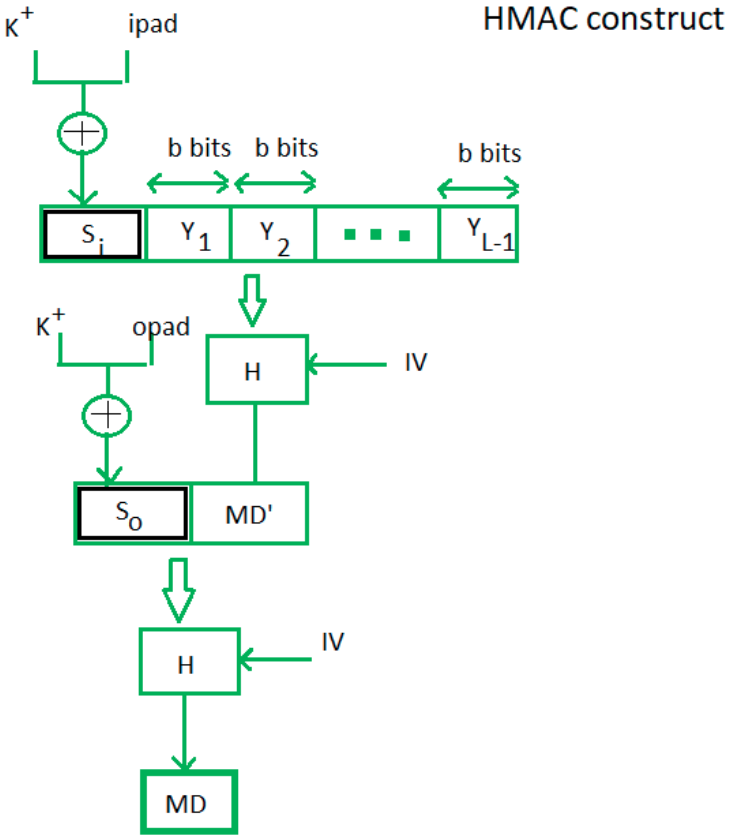
compulsory to implement in IP security. The FIPS 198 NIST standard has also been issued by HMAC.

Objectives of HMAC

- As the Hash Function, HMAC is also aimed to be one way, i.e., easy to generate output from input but complex the other way around.
- It aims at being less affected by collisions than the hash functions.
- HMAC reuses algorithms like MD5 and SHA-1 and checks to replace the embedded hash functions with more secure hash functions, in case found.
- HMAC tries to handle the Keys in a more simple manner.

Working of HMAC Algorithm

The working of HMAC starts with taking a message M containing blocks of length *b* bits. An input signature is padded to the left of the message and the whole is given as input to a hash function which gives us a temporary message-digest MD'. MD' again is appended to an output signature and the whole is applied a hash function again, the result is our final message digest MD. Here is a simple structure of HMAC:



- Here, H stands for Hashing function,
- M is the original message
- Si and So are input and output signatures respectively,
- Yi is the ith block in original message M, where I ranges from [1, L)
- L = the count of blocks in M
- K is the secret key used for hashing
- IV is an initial vector (some constant)
- The generation of input signature and output signature *Si* and *So* respectively.

$S_i = K^+ \oplus$ ipad — where K^+ is nothing but K padded with zeros on the left so that the result is b bits in length

$S_o = K^+ \oplus$ opad — where ipad and opad are 00110110 and 01011100 respectively taken b/8 times repeatedly.

$MD' = H(S_i || M)$

$MD = H(S_o || MD')$ or $MD = H(S_o || H(S_i || M))$

To a normal hash function, HMAC adds a compression instance to the processing. This structural implementation holds efficiency for shorter MAC values.

Security in Hash-based Message Authentication Code

HMAC is more secure than MAC since the key and message are hashed in different steps: *HMAC(key, message) = H(mod1(key) || H(mod2(key) || message)*.

The data is initially hashed by the client using a private key before being sent to the server as part of the request. The server then creates its own HMAC. This assures that the process is not vulnerable to attacks, which could result in crucial data being disclosed as subsequent MACs are generated. Additionally, once the procedure is completed, the delivered message becomes irreversible and resistant to hackers. Even if a malicious party attempts to steal the communication, they will be unable to determine its length or decrypt it because they do not have the decryption key.

Advantages of HMAC

- HMACs are ideal for high-performance systems like routers due to the use of hash functions which are calculated and verified quickly unlike the public key systems.
- Digital signatures are larger than HMACs, yet the HMACs provide comparably higher security.
- HMACs are used in administrations where public key systems are prohibited.