Unit II: Cryptographic Algorithms

Contents

2.1 Classical Cryptosystems: Ceasar, Vignere, Playfair, Rail Fence Ciphers
2.2 Modern Ciphers: Block vs. Stream Ciphers, Symmetric vs. Asymmetric Ciphers
2.3 Symmetric Encryption: Fiestel Cipher Structure, Data Encryption Standards (DES), Basic Concepts of Fields: Groups, Rings, Fields, Modular Arithmetic, Galois Fields, Polynomial Arithmetic, Advanced Encryption Standards (AES)
2.4 Number Theory: Prime Numbers, Fermat's Theorem, Primility Testing: Miller-Rabin Algorithm, Euclidean Theorem, Extended Euclidean Theorem, Euler Totient Function
2.5 Asymmetric Encryption: Diffie-Helman Key Exchange, RSA Algorithm

Symmetric encryption, also referred to as conventional encryption or single-key encryption, was the only type of encryption in use prior to the development of public-key encryption in the 1970s. It remains by far the most widely used of the two types of encryption.

An original message is known as the **plaintext**, while the coded message is called the **ciphertext**.

The process of converting from plaintext to ciphertext is known as **enciphering** or **encryption**; restoring the plaintext from the ciphertext is **deciphering** or **decryption**.

The many schemes used for encryption constitute the area of study known as **cryptography**. Such a scheme is known as a **cryptographic system** or a **cipher**.

Techniques used for deciphering a message without any knowledge of the enciphering details fall into the area of **cryptanalysis**. Cryptanalysis is what the layperson calls “breaking the code”. The areas of cryptography and cryptanalysis together are called **cryptology**.

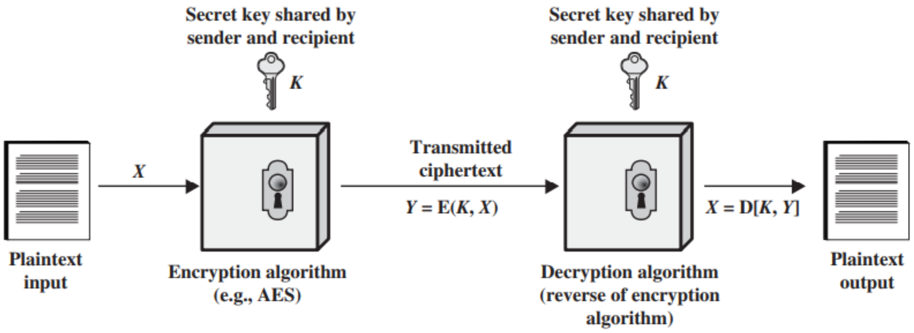

Secret key shared by sender and recipient
K
Secret key shared by sender and recipient
K
X
Transmitted ciphertext
Y = E(K, X)
X = D[K, Y]
Plaintext input
Encryption algorithm (e.g., AES)
Decryption algorithm (reverse of encryption algorithm)
Plaintext output

Figure: Simplified model of Symmetric Encryption

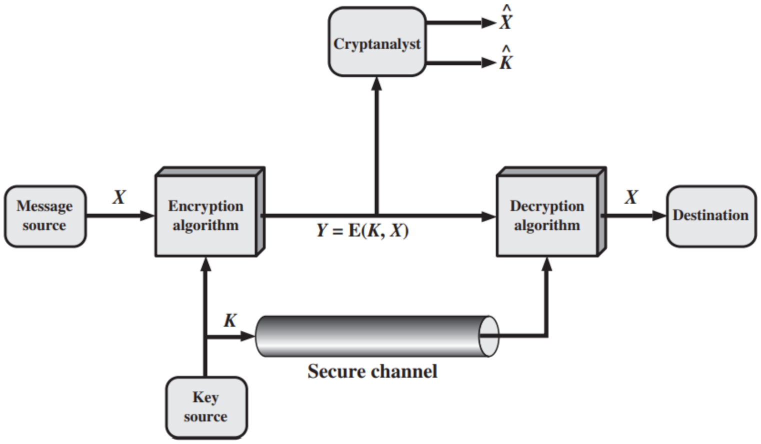


Figure: Model of Symmetric Cryptosystem

2.1 Classical Cryptosystems: Ceasar, Vigenere, Playfair, Rail Fence Ciphers

Classical cryptosystems are mainly based on substitution techniques. A substitution technique is one in which the letters of plaintext are replaced by other letters or by numbers or symbols. If the plaintext is viewed as a sequence of bits, then substitution involves replacing plaintext bit patterns with ciphertext bit patterns.

Ceasar Cipher

The earliest known, and the simplest, use of a substitution cipher was by Julius Caesar. The Caesar cipher involves replacing each letter of the alphabet with the letter standing three places further down the alphabet.

For example,

plain: meet me after the toga party

cipher: PHHW PH DIWHU WKH WRJD SDUWB

Note that the alphabet is wrapped around, so that the letter following Z is A. We can define the transformation by listing all possibilities, as follows:

plain: 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

cipher: 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

Then the algorithm can be expressed as follows. For each plaintext letter *p*, substitute the ciphertext letter C^2.

C = E(3, p) = (p + 3) mod 26

A shift may be of any amount, so that the general Caesar algorithm is

C = E(k, p) = (p + k) mod 26

Where, k takes on a value in the range 1 to 25. The decryption algorithm is simply

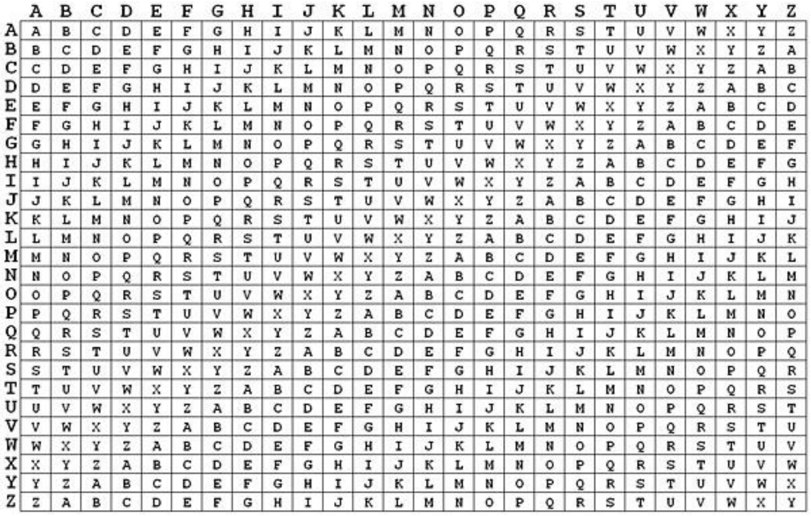
p = D(k, C) = (C - k) mod 26

If it is known that a given ciphertext is a Caesar cipher, then a brute-force cryptanalysis is easily performed: simply try all the 25 possible keys. Three important characteristics of this problem enabled us to use a brute-force cryptanalysis:

1. The encryption and decryption algorithms are known.
2. There are only 25 keys to try.
3. The language of the plaintext is known and easily recognizable.

Vigenere cipher

The Vigenère cipher uses a 26×26 table with A to Z as the row heading and column heading. This table is usually referred to as the Vigenère Tableau, Vigenère Table or Vigenère Square. The first row of this table has the 26 English letters. Starting with the second row, each row has the letters shifted to the left one position in a cyclic way. For example, when B is shifted to the first position on the second row, the letter A moves to the end.



	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

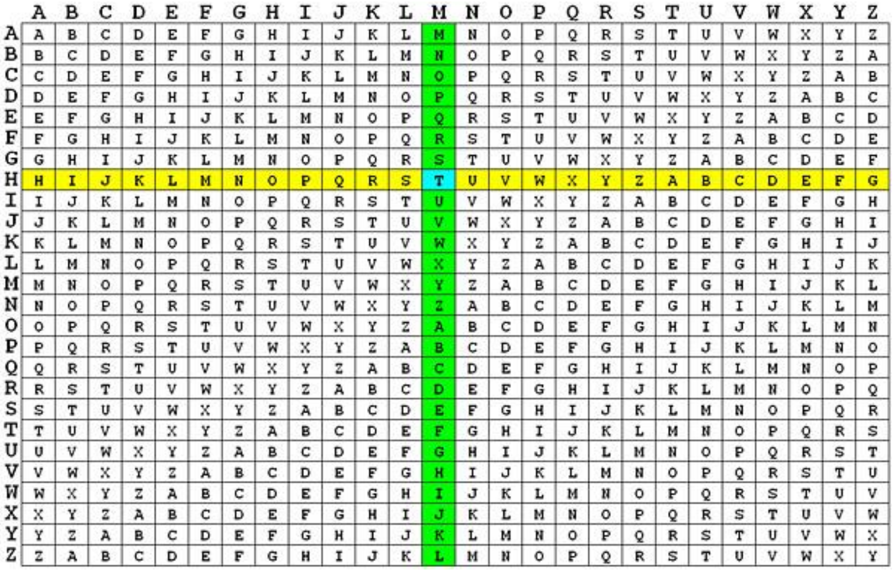
In addition to the plaintext, the Vigenère cipher also requires a keyword, which is repeated so that the total length is equal to that of the plaintext. For example, suppose the plaintext is MICHIGAN TECHNOLOGICAL UNIVERSITY and the keyword is HOUGHTON. Then, the keyword must be repeated as follows:

```
MICHIGAN TECHNOLOGICAL UNIVERSITY
HOUGHTON HOUGHTONHOUGH TONHOUGNTO
```

We follow the tradition by removing all spaces and punctuation, converting all letters to upper case, and dividing the result into 5-letter blocks. As a result, the above plaintext and keyword become the following:

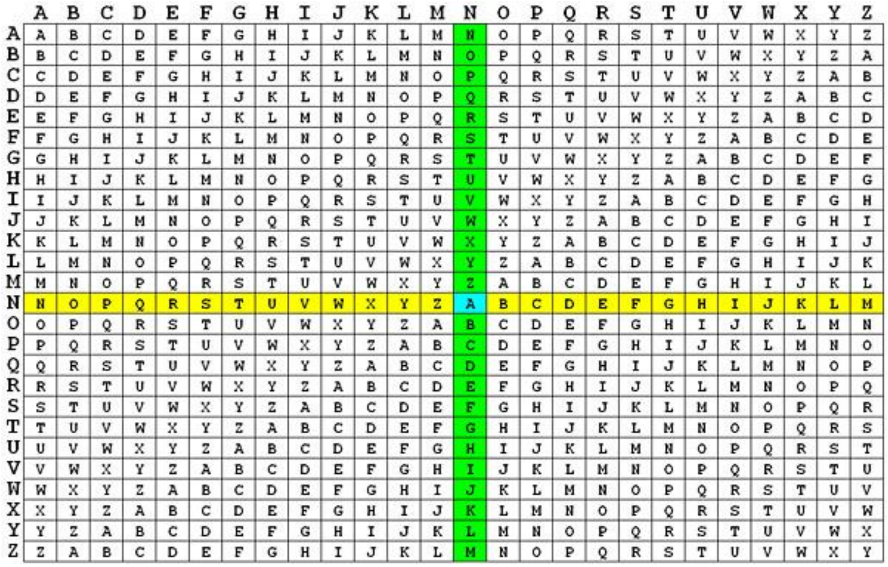
```
MICHI GANTE CHNOL OGICA LUNIV ERSIT Y
HOUGH TONHO UGHTO NHOUG HTONH OUGHT O
```

To encrypt, pick a letter in the plaintext and its corresponding letter in the keyword, use the keyword letter and the plaintext letter as the row index and column index, respectively, and the entry at the row-column intersection is the letter in the ciphertext. For example, the first letter in the plaintext is M and its corresponding keyword letter is H. This means that the row of H and the column of M are used, and the entry T at the intersection is the encrypted result.



	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

Similarly, since the letter N in MICHIGAN corresponds to the letter N in the keyword, the entry at the intersection of row N and column N is A which is the encrypted letter in the ciphertext.



	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

Repeating this process until all plaintext letters are processed, the ciphertext is TWWNPZOA ASWNUHZBNWWGS NBVCSLYPMM. The following has the plaintext, repeated keyword and ciphertext aligned together.

```
MICHI GANTE CHNOL OGICA LUNIV ERSIT Y
HOUGH TONHO UGHTO NHOUG HTONH OUGHT O
TWWNP ZOAAS WNUHZ BNWWG SNBVC SLYPM M
```

To decrypt, pick a letter in the ciphertext and its corresponding letter in the keyword, use the keyword letter to find the corresponding row, and the letter heading of the column that contains the ciphertext letter is the needed plaintext letter. For example, to decrypt the first letter T in the ciphertext, we find the corresponding letter H in the keyword. Then, the row of H is used to find the corresponding letter T and the column that contains T provides the plaintext letter M (see the above figures). Consider the fifth letter

P in the ciphertext. This letter corresponds to the keyword letter H and row H is used to find P. Since P is on column I, the corresponding plaintext letter is I.

Playfair cipher

The Playfair cipher was the first practical digraph substitution cipher. The scheme was invented in 1854 by Charles Wheatstone but was named after Lord Playfair who promoted the use of the cipher. In playfair cipher unlike traditional cipher we encrypt a pair of alphabets (digraphs) instead of a single alphabet.

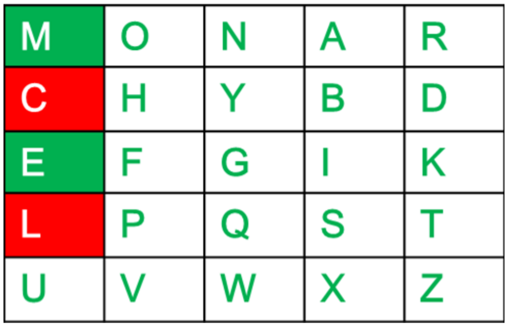
It was used for tactical purposes by British forces in the Second Boer War and in World War I and for the same purpose by the Australians during World War II. This was because Playfair is reasonably fast to use and requires no special equipment.

<u>The Playfair Cipher Encryption Algorithm:</u>

The Algorithm consists of 2 steps:

1. Generate the key Square(5×5): The key square is a 5×5 grid of alphabets that acts as the key for encrypting the plaintext. Each of the 25 alphabets must be unique and one letter of the alphabet (usually J) is omitted from the table (as the table can hold only 25 alphabets). If the plaintext contains J, then it is replaced by I.
   The initial alphabets in the key square are the unique alphabets of the key in the order in which they appear followed by the remaining letters of the alphabet in order.

   Example: Plaintext: “instrument”   key: ”monarchy”



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

2. Algorithm to encrypt the plain text: The plaintext is split into pairs of two letters (digraphs). If there is an odd number of letters, a Z is added to the last letter.

Example:

Plaintext: “instruments”   After split: ‘in’ ‘st’ ‘ru’ ‘me’ ‘nt’ ‘sz’   Here, z is added at the last.

Pair cannot be made with same letter. Break the letter in single and add a bogus letter to the previous letter. Plain Text: “hello”   After Split: ‘he’ ‘lx’ ‘lo’   Here, x is the bogus letter.

<u>Rules for Encryption</u>

1. If both the letters are in the same column: Take the letter below each one (going back to the top if at the bottom).
   For example: diagraph: “me”
   Encryption: m -> c , e -> l
   Encrypted text: cl

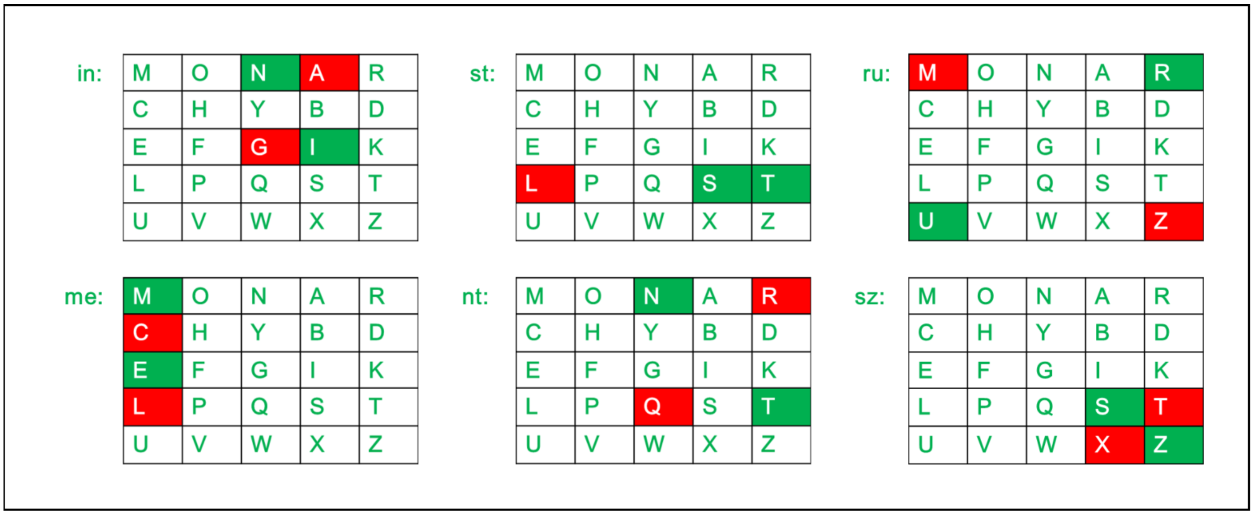
2. If both the letters are in the same row: Take the letter to the right of each one (going back to the leftmost if at the rightmost position).
   For example: diagraph: “st”
   Encryption: s -> t , t -> l
   Encrypted text: tl
3. If neither of the above rules is true: Form a rectangle with the two letters and take the letters on the horizontal opposite corner of the rectangle.
   For example: diagraph: “nt”
   Encryption: n -> r , t -> q
   Encrypted text: rq

Therefore,

Plaintext: “instrumentsz”

Encryption: i -> g , n -> a , s -> t , t -> l , r -> m , u -> z , m -> c , e -> l, n -> r , t -> q , s -> t , z -> x

Encrypted text: gatlmzclrqtx



Rail Fence Ciphers

The rail fence cipher (also called a zigzag cipher) is a form of transposition cipher. It derives its name from the way in which it is encoded.

Example 1:

Encryption Input : "GeeksforGeeks" Key = 3 Output : GsGsekfrek eoe

Decryption Input : GsGsekfrek eoe Key = 3 Output : "GeeksforGeeks"

Example 2:

Encryption Input : "defend the east wall" Key = 3 Output : dnhaweedtees alf tl

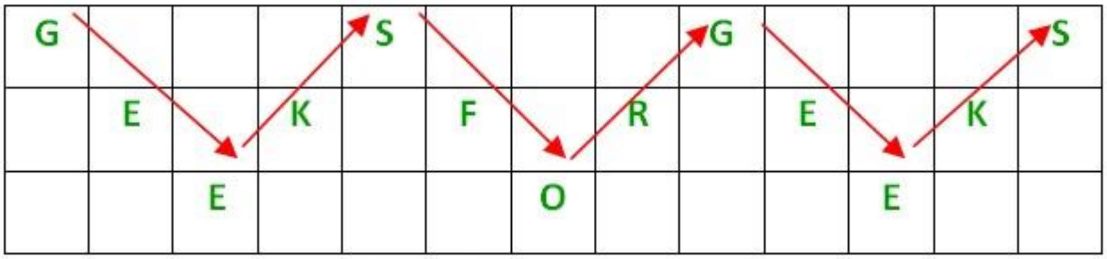
Decryption Input : dnhaweedtees alf tl Key = 3 Output : defend the east wall

Encryption

In a transposition cipher, the order of the alphabets is re-arranged to obtain the cipher-text.

- In the rail fence cipher, the plain-text is written downwards and diagonally on successive rails of an imaginary fence.
- When we reach the bottom rail, we traverse upwards moving diagonally, after reaching the top rail, the direction is changed again. Thus the alphabets of the message are written in a zig-zag manner.
- After each alphabet has been written, the individual rows are combined to obtain the cipher-text.

For example, if the message is “GeeksforGeeks” and the number of rails = 3 then cipher is prepared as:



G				S				G				S
	E		K		F		R		E		K	
		E				O				E		

Ciphertext : GSGSEKFREKEOE

Decryption

As we’ve seen earlier, the number of columns in rail fence cipher remains equal to the length of plain-text message. And the key corresponds to the number of rails.

- Hence, rail matrix can be constructed accordingly. Once we’ve got the matrix we can figure-out the spots where texts should be placed (using the same way of moving diagonally up and down alternatively).
- Then, we fill the cipher-text row wise. After filling it, we traverse the matrix in zig-zag manner to obtain the original text.

Here, Ciphertext = GSGSEKFREKEOE and Key = 3

Number of columns = length of ciphertext = 13

Number of rows = Key = 3

Hence the matrix will be 3*13, now marking places with text as “*”, we get

*				*				*				*
	*		*		*		*		*		*	
		*				*				*		

2.2 Modern Ciphers: Block vs. Stream Ciphers, Symmetric vs. Asymmetric Ciphers

Block vs. Stream Ciphers

Both Block and Stream cipher are the methods of Encryptions which are primarily used for converting the plain text into cipher text directly and belong to the family of symmetric key ciphers.

Block Cipher takes a message and break it into a fixed size of blocks and converts one block of the message at an instant. For example, we have a message in plain text "STREET_BY_STREET" required to be encrypted. Using bock cipher, "STREET" must be encrypted at first, followed by "_BY_" and finally at last "STREET".

Stream Cipher typically encrypts one byte of the message at that moment instead of using blocks. Let's take an example, suppose the original message (plain text) is "blue sky" in ASCII (i.e. text format). When you convert these ASCII into equivalent binary values, it will give the output in 0's and 1's form. Let it be translated in 010111001.

Differences between Block Cipher and Stream Cipher

SN	Key	Block Cipher	Stream Cipher
1	Definition	Block Cipher is the type of encryption where the conversion of plain text performed by taking its block at a time.	On other hand Stream Cipher is the type of encryption where the conversion of plain text performed by taking one byte of the plain text at a time.
2	Conversion of bits	As Block Cipher takes block at a time so comparatively more bits get converted as compared to in Stream Cipher specifically 64 bits or more could get converted at a time.	On other hand in case of Stream Cipher at most 8 bits could get converted at a time.
3	Principle	Block Cipher uses both confusion and diffusion principle for the conversion required for encryption.	On other hand Stream Cipher uses only confusion principle for the conversion.
4	Algorithm	For encryption of plain text Block Cipher uses Electronic Code Book (ECB) and Cipher Block Chaining (CBC) algorithm.	On other hand Stream Cipher uses CFB (Cipher Feedback) and OFB (Output Feedback) algorithm.
5	Decryption	As combination of more bits get encrypted in case of Block Cipher so the reverse encryption or decryption is comparatively complex as compared to that of Stream Cipher.	On other hand Stream Cipher uses XOR for the encryption which can be easily reversed to the plain text.
6	Implementation	The main implementation of Block Cipher is Feistel Cipher.	On other hand the main implementation of Stream Cipher is Vernam Cipher.
7	Complexity	Simple design	Comparatively complex.
8	No. of bits used	64 bits or more	8 bits

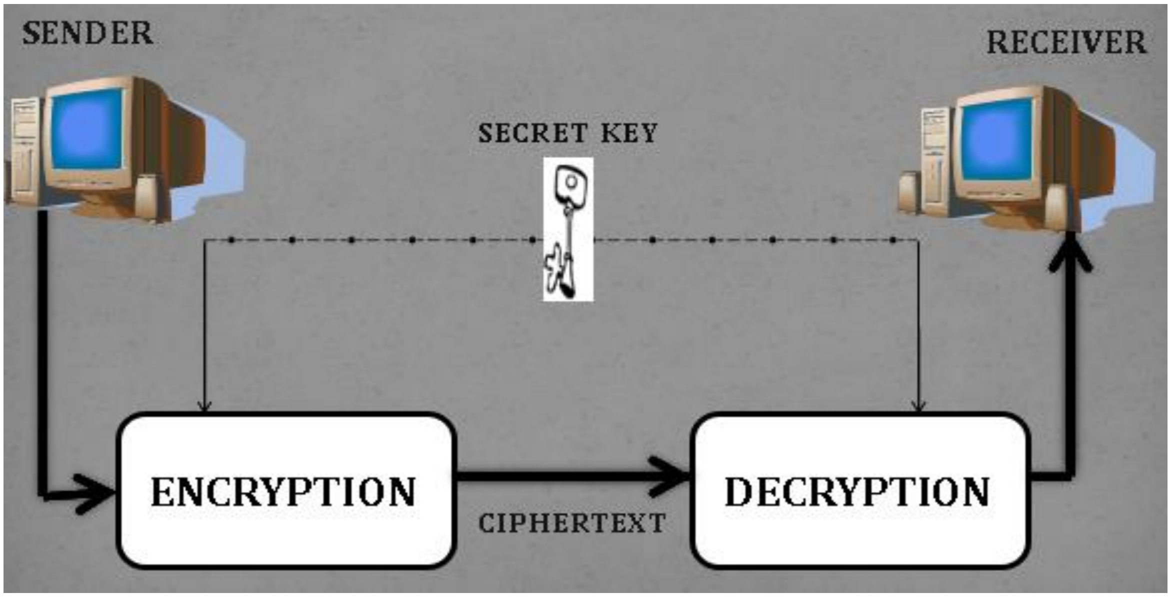
Symmetric vs. Asymmetric Ciphers

Secret/Symmetric Cipher uses a single key for both encryption and decryption.

Public/Asymmetric Cipher uses one key for encryption and another for decryption.

Secret/Symmetric key Cryptography

- The message (plaintext) is encrypted into ciphertext using a key.
- The resulting ciphertext is sent to the recipient, who will decrypt it using the **same key.**
- Hence, the same key must be known to both parties.
- The best known secret-key system is the Data Encryption Standard (DES).
- This method is easy and fast to implement but has weaknesses;
- The algorithm that is used to encode the message is easier for attackers to understand, enabling them to more easily decode the message.



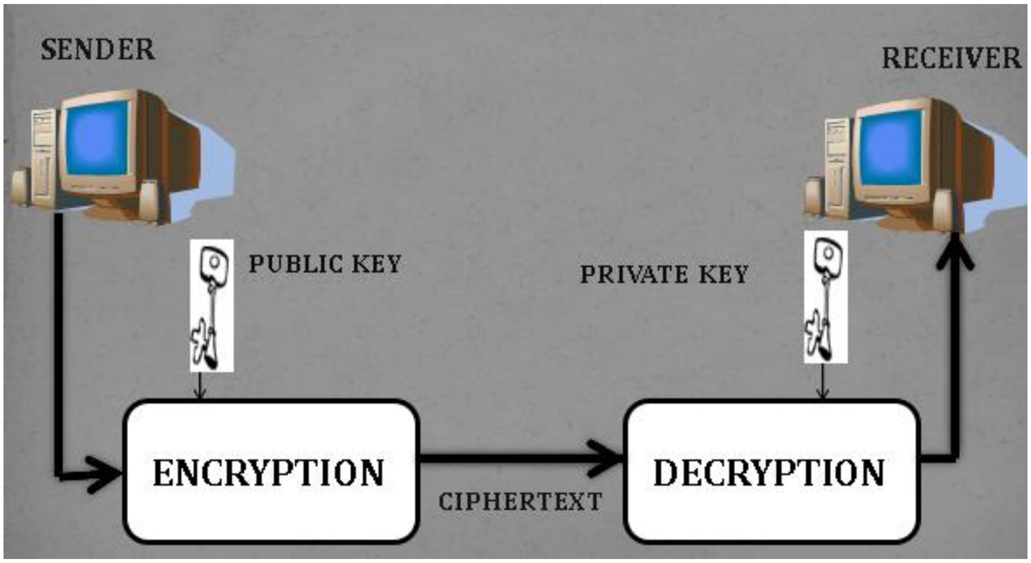
Advanced Encryption Standard (AES) is the most widely used symmetric algorithm. Wireless Protected Access 2 (WPA2), which is the most commonly used security protocol for wireless networks today, employs the AES encryption algorithm.

Some of the common secret key cryptography methods used are:

1. Advanced Encryption Standard (AES) – 128, 192, 256 bits
2. Data Encryption Standard (DES)
3. Triple Data Encryption Standard (TripleDES) – advanced form of DES
4. Twofish - 128 bits – successor of Blowfish
5. Rivest Cipher 4 (RC4)
6. QUAD (Cipher)

Public/Asymmetric key Cryptography

- The **public key** can only be used to encrypt the message and the **private key** can only be used to decrypt it.
- This allows a user to freely distribute his or her **public key** to people who are likely to want to communicate with him or her without worry of compromise because only someone with the **private key** can decrypt a message.
- To secure information between two users, the sender encrypts the message using the **public key** of the receiver. The receiver then uses the **private key** to decrypt the message.
- The best-known public-key cryptosystem is RSA, named after its inventors: Rivest, Shamir, and Adleman.



Some of the common public key cryptography methods used are:

1. Rivest-Shamir-Adleman (RSA)
2. Elliptic Curve Cryptography (ECC)
3. ElGamal Encryption

Differences between symmetric and asymmetric key cryptography

Characteristics	**Symmetric Key Cryptography**	**Asymmetric Key Cryptography**
Key used for encryption/decryption	Same key is used for encryption and decryption	One key is used for encryption and another, different key is used for decryption
Complexity	Simple	Complex, due to use of two different keys
Speed of encryption/decryption	Very fast	Slower
Size of the resulting encrypted text	Usually same as or less than the original plain text size	More than the original plain text size
Key length	Smaller key lengths are used to encrypt the data (e.g. 128 - 256 bits)	Usually uses longer keys lengths (e.g. 1024 – 4096 bits)
Key agreement/exchange	A big issue	No problem at all
Number of keys required as compared to the number of participants in the message exchange	Equals about the square of the number of the participants, so scalability is an issue	Same as the number of participants, so scales up quite well
Usage	Mainly for encryption/decryption (confidentiality). Cannot be used for digital signatures.	Can be used for encryption/ decryption as well as for digital signatures (integrity and non-repudiation)
Ideal use	Applications where a large number of data are to be encrypted	Applications where small amount of data are to be encrypted, digital signatures.

Examples	DES, AES, TDES, RC4, QUAD	RSA, ECC, ElGamal, Digital Signature algorithms

2.3 Symmetric Encryption: Fiestel Cipher Structure, Data Encryption Standards (DES), Basic Concepts of Fields: Groups, Rings, Fields, Modular Arithmetic, Galois Fields, Polynomial Arithmetic, Advanced Encryption Standards (AES)

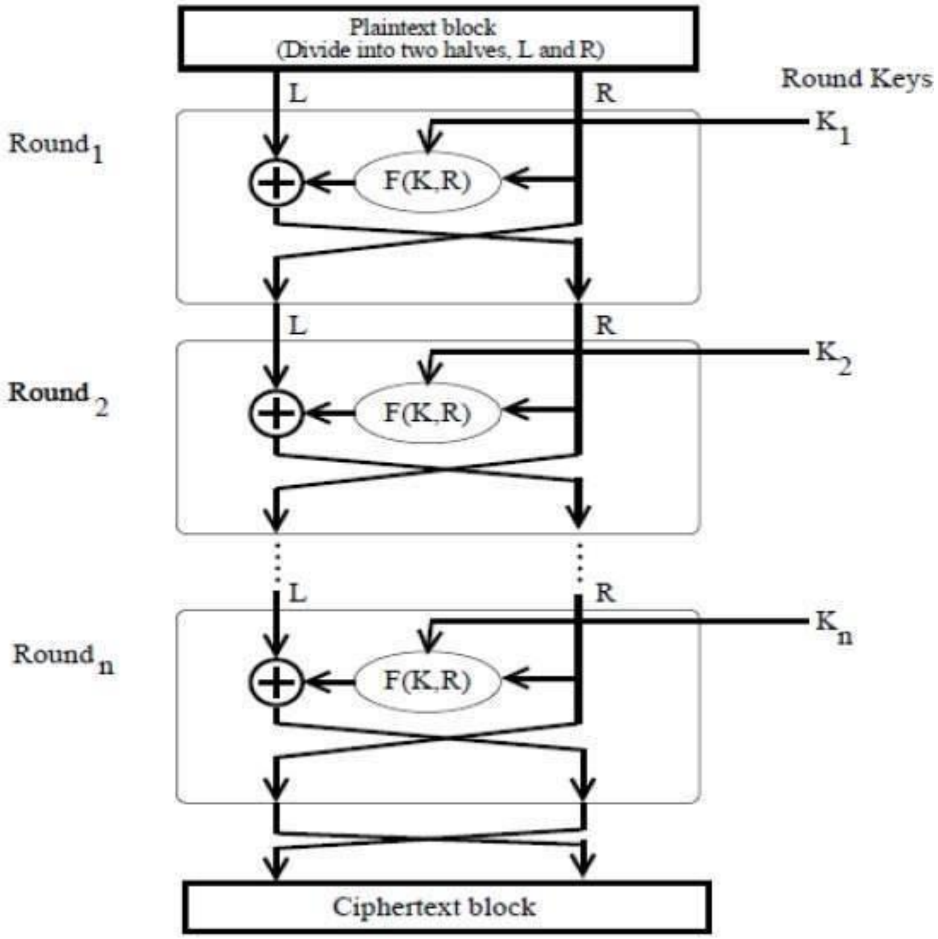
Fiestel Cipher Structure

It is a design model from which many different block ciphers are derived. DES is just one example of a Feistel Cipher. A cryptographic system based on Feistel cipher structure uses the same algorithm for both encryption and decryption.

Encryption Process

The encryption process uses the Feistel structure consisting multiple rounds of processing of the plaintext, each round consisting of a “substitution” step followed by a permutation step.

Feistel Structure is shown in the following illustration –



1. The input block to each round is divided into two halves that can be denoted as L and R for the left half and the right half.

2. In each round, the right half of the block, R, goes through unchanged. But the left half, L, goes through an operation that depends on R and the encryption key. First, we apply an encrypting function ‘f’ that takes two input − the key K and R. The function produces the output f(R,K). Then, we XOR the output of the mathematical function with L.
3. In real implementation of the Feistel Cipher, such as DES, instead of using the whole encryption key during each round, a round-dependent key (a subkey) is derived from the encryption key. This means that each round uses a different key, although all these subkeys are related to the original key.
4. The permutation step at the end of each round swaps the modified L and unmodified R. Therefore, the L for the next round would be R of the current round. And R for the next round be the output L of the current round.
5. Above substitution and permutation steps form a ‘round’. The number of rounds are specified by the algorithm design.
6. Once the last round is completed then the two sub blocks, ‘R’ and ‘L’ are concatenated in this order to form the ciphertext block.

The difficult part of designing a Feistel Cipher is selection of round function ‘f’.

Decryption Process

The process of decryption in Feistel cipher is almost similar. Instead of starting with a block of plaintext, the ciphertext block is fed into the start of the Feistel structure and then the process thereafter is exactly the same as described in the given illustration.

The process is said to be almost similar and not exactly same. In the case of decryption, the only difference is that the subkeys used in encryption are used in the reverse order.

The final swapping of ‘L’ and ‘R’ in last step of the Feistel Cipher is essential. If these are not swapped then the resulting ciphertext could not be decrypted using the same algorithm.

Number of Rounds

The number of rounds used in a Feistel Cipher depends on desired security from the system. More number of rounds provide more secure system. But at the same time, more rounds mean the inefficient slow encryption and decryption processes. Number of rounds in the systems thus depend upon efficiency–security tradeoff.

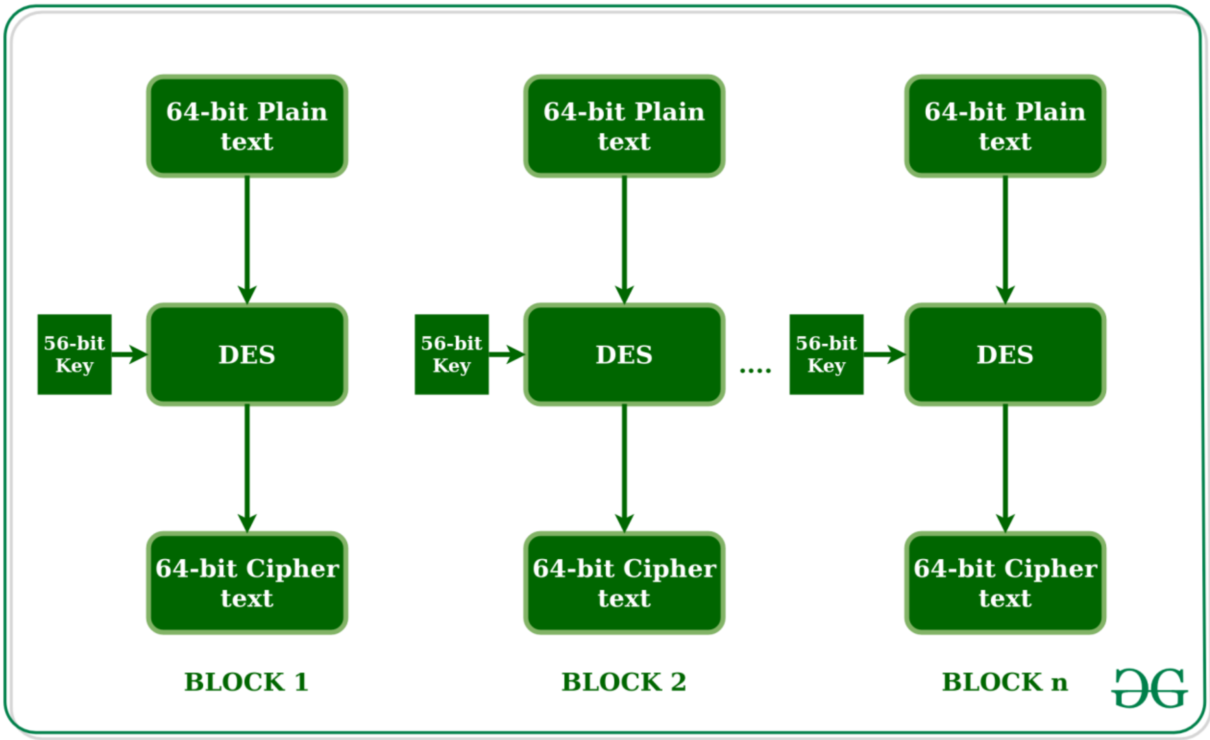
Data Encryption Standards (DES)

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

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

Data encryption standard (DES) has been found vulnerable against very powerful attacks and therefore, the popularity of DES has been found slightly on the decline.

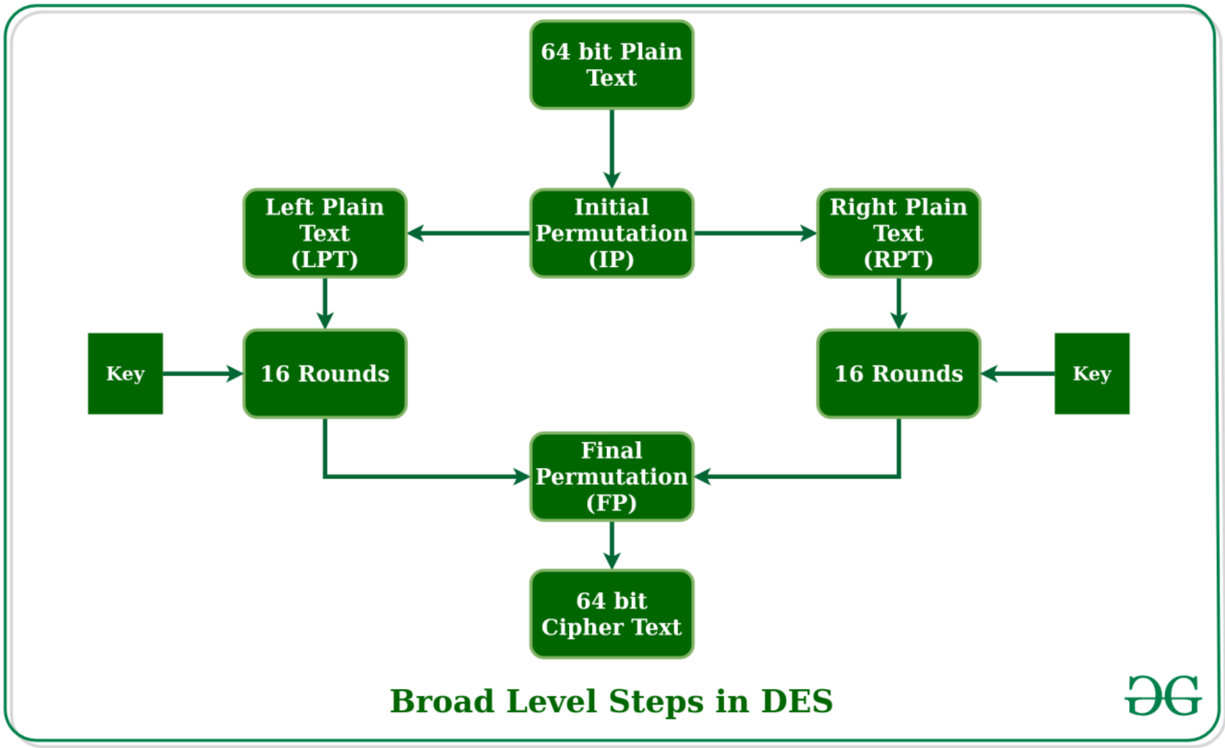
DES is a block cipher and encrypts data in blocks of size of 64 bits each, which means 64 bits of plain text goes as the input to DES, which produces 64 bits of ciphertext. The same algorithm and key are used for encryption and decryption, with minor differences. The key length is 56 bits. The basic idea is shown in the figure.



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.

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.

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



Broad Level Steps in DES

Basic Concepts of Fields: Groups, Rings, Fields, Modular Arithmetic, Galois Fields, Polynomial Arithmetic

Group

- a set of elements or "numbers" (A generalization of usual arithmetic)
- obeys:

  – closure: a.b also in G

  – associative law: (a.b).c = a.(b.c)

  – has identity e: e.a = a.e = a

  – has inverses a^{-1}: $a.a^{-1} = e$

- if commutative a.b = b.a
- then forms an abelian group
- Examples in P.105

Ring

- a set of "numbers" with two operations (addition and multiplication) which are:
- an abelian group with addition operation
- multiplication:
  – has closure
  – is associative
  – distributive over addition: a(b+c) = ab + ac
- In essence, a ring is a set in which we can do addition, subtraction [a – b = a + (–b)], and multiplication without leaving the set.

- With respect to addition and multiplication, the set of all n-square matrices over the real numbers form a ring.
- if multiplication operation is commutative, it forms a commutative ring
- if multiplication operation has an identity element and no zero divisors (ab=0 means either a=0 or b=0), it forms an integral domain
- The set of Integers with usual + and x is an integral domain

Field

- a set of numbers with two operations:
  – Addition and multiplication
  – F is an integral domain
  – F has multiplicative reverse
  • For each a in F other than 0, there is an element b such that ab=ba=1
- In essence, a field is a set in which we can do addition, subtraction, multiplication, and division without leaving the set.
  – Division is defined with the following rule: a/b = a (b –1)
- Examples of fields: rational numbers, real numbers, complex numbers. Integers are NOT a field.

Modular Arithmetic

- Define modulo operator a mod n to be remainder when a is divided by n
  – e.g. 1 = 7 mod 3 ; 4 = 9 mod 5
- Use the term congruence for: a ≡ b (mod n)
  – when divided by n, a & b have same remainder
  – eg. 100 ≡ 34 (mod 11)
- b is called the residue of a mod n
  – since with integers can always write: a = qn + b
- usually have 0 <= b <= n-1 -12 mod 7 = -5 mod 7 = 2 mod 7 = 9 mod 73

Galois Fields

- finite fields play a key role in many cryptography algorithms
- can show number of elements in any finite field must be a power of a prime number p^n
- known as Galois fields
- denoted GF(p^n)
- in particular often use the fields:
  – GF(p)
  – GF(2^n)

Polynomial Arithmetic

- can compute using polynomials
- S€ $f(x) = a_nx^n + a_{n-1}x^{n-1} + ... + a_1x + a_0 = \sum_{i=0}^{n} a_i x^i$
  – poly arithmetic with coefficients mod p
  – poly arithmetic with coefficients mod p and polynomials mod another polynomial M(x)
- Motivation: use polynomials to model Shift and XOR operations

Advanced Encryption Standards (AES)

Advanced Encryption Standard (AES) is a specification for the encryption of electronic data established by the U.S National Institute of Standards and Technology (NIST) in 2001. AES is widely used today as it is a much stronger than DES and triple DES despite being harder to implement.

- AES is a block cipher.
- The key size can be 128/192/256 bits.
- Encrypts data in blocks of 128 bits each.

That means it takes 128 bits as input and outputs 128 bits of encrypted cipher text as output. AES relies on substitution-permutation network principle which means it is performed using a series of linked operations which involves replacing and shuffling of the input data.

Features of AES:

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

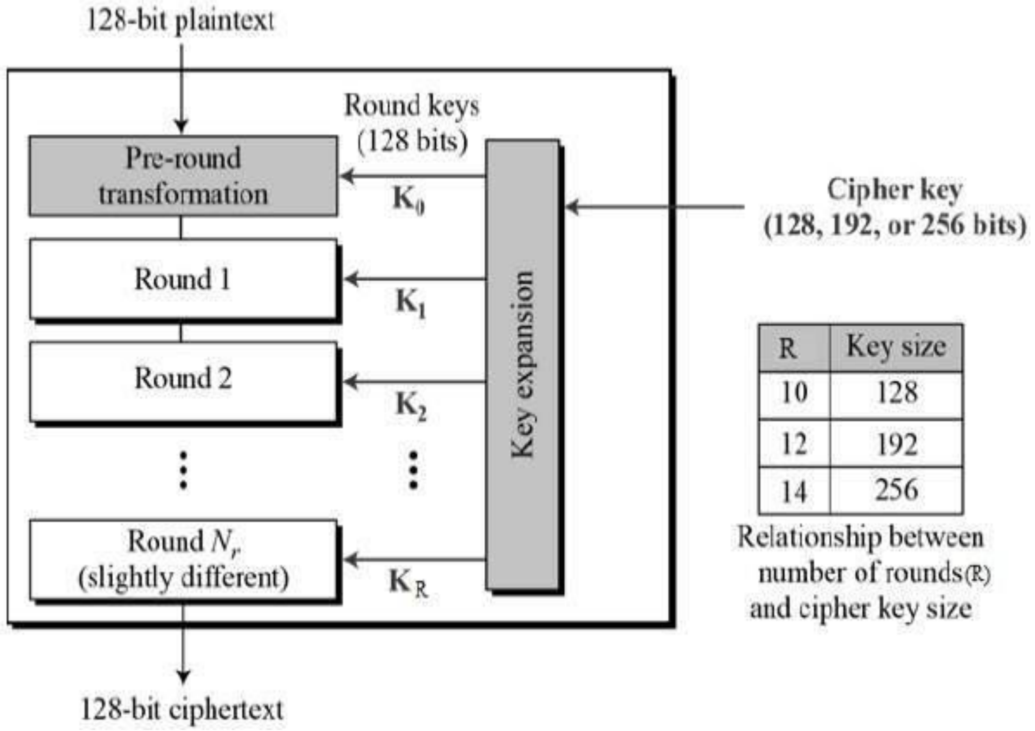
Operation of AES

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

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

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

The schematic of AES structure is given in the following illustration –



R	Key size
10	128
12	192
14	256

Relationship between number of rounds(R) and cipher key size

Decryption Process

The process of decryption of an AES ciphertext is similar to the encryption process in the reverse order. Each round consists of the four processes conducted in the reverse order −

1. Add round key
2. Mix columns
3. Shift rows
4. Byte substitution

Since sub-processes in each round are in reverse manner, unlike for a Feistel Cipher, the encryption and decryption algorithms needs to be separately implemented, although they are very closely related.

AES Analysis

In present day cryptography, AES is widely adopted and supported in both hardware and software. Till date, no practical cryptanalytic attacks against AES has been discovered. Additionally, AES has built-in flexibility of key length, which allows a degree of 'future-proofing' against progress in the ability to perform exhaustive key searches.

However, just as for DES, the AES security is assured only if it is correctly implemented and good key management is employed.

2.4 Number Theory: Prime Numbers, Fermat's Theorem, Primility Testing: Miller-Rabin Algorithm, Euclidean Theorem, Extended Euclidean Theorem, Euler Totient Function

Overview

- **Modular Arithmetic**: We begin by defining how to perform basic arithmetic modulo n, where n is a positive integer. Addition, subtraction, and multiplication follow naturally from their integer counterparts, but we have complications with division.

- **Euclid's Algorithm**: We will need this algorithm to fix our problems with division. It was originally designed to find the greatest common divisor of two numbers.
- **Divison**: Once armed with Euclid's algorithm, we can easily compute divisions modulo n.
- **The Chinese Remainder Theorem:** We find we only need to study Z_p^k where p is a prime, because once we have a result about the prime powers, we can use the Chinese Remainder Theorem to generalize for all n.
- **Units**: While studying division, we encounter the problem of inversion. Units are numbers with inverses.
- **Exponentiation**: The behaviour of units when they are exponentiated is difficult to study. Modern cryptography exploits this.
- **Order of a Unit:** If we start with a unit and keep multiplying it by itself, we wind up with 1 eventually. The order of a unit is the number of steps this takes.
- **The Miller-Rabin Test**: It is a fast way of telling if a given number is prime that works with high probability.
- **Generators**: Sometimes powering up a unit will generate all the other units.
- **Cyclic Groups**: We focus only on multiplication and see if we can still say anything interesting.
- **Quadratic Residues**: Elements of Z_n that are perfect squares are called quadratic residues.

Prime Numbers

Prime numbers are extremely important to computer security. Public-key cryptography would not be possible without prime number. An important concern in public-key cryptography is to test a randomly selected integer for its primality. That is, we first generate a random number and then try to figure out whether it is prime.

An integer is prime if it has exactly two distinct divisors, the integer 1 and itself. That makes the integer 2 the first prime.

2.5 Asymmetric Encryption: Diffie-Helman Key Exchange, RSA Algorithm

Diffie-Helman Key Exchange Algorithm

Diffe Hellman cryptography is based on key exchange. The both parties need to exchange secrets key to encrypt message. It is based on difficulty of computing discrete logarithms. Diffie-Hellman is based on symmetric key exchange for both encryption and decryption.

- Suppose the two parties Alice and Bob would to communicate and they have decided to Diffe Hellman cryptography.
- Alice select prime modulus (p) as 23 and generator (g) as 7. The two parties publishes p and g public meaning p and q are known to both parties.
- Alice randomly selects number (x) as 3 and Bob randomly selects number (y) as 6. Alice will calculate the secret number (r1) as r1 = g x mod p (7 3 mod 23= 21) and sends to Bob.
- Bob on the other hand calculate r2 = g y mod p, 76 mod 23 which is 4. So r2 = 4. Now Bob will send r2 = 4 to Alice.
- Now they will both calculate the share or symmetric key (k). Alice will calculate k = r2x mod p, 4 3 mod 23 and k = 18. Bob will calculate k = r1y mod p, 216 mod 23 and k = 18. The key k = 18 is same and both will use this key to send and receive data.

Use of Diffie-Helman

1. Sharing of new DH key is very fast and safe
2. The sender and the receiver may not have prior knowledge of each other.
3. Communication can take place through an insecure channel.
4. Used in SSL (Secure Socket Layer), TLS (Transport layer security), SSH (Secure Shell), IPSec (Internet Protocol Security), PKI (Public Key Infrastructure).
5. Used in card payment system.
6. Used in POS and ATM network management.

Limitations of Diffie-Helman

1. Cannot be used for asymmetric key exchange.
2. Cannot be used for digital signature.
3. Vulnerable to man-in-the-middle attacks since it doesn’t authenticate either party involved in exchange.
4. Could be used in Denial of service attack easily.
5. Cannot be used to encrypt message.

RSA Algorithm

RSA algorithm is called a public key algorithm. RSA uses one key for encryption and the second key for decryption. The key used for encryption is called public key and the key used for decryption is called private key. RSA is invited by three most popular people. Rivest, Shamir and Adleman. Hence the algorithm is called RSA. The encryption and decryption uses modular exponentiation. Public key is more advanced and being used for many years. RSA uses mathematical computation. The Public key encryption has six components: Plaintext, Encryption Algorithm, Public and Private Key, Cipher text, Decryption Algorithm.

RSA encryption uses a block cipher in which the plaintext and cipher text are integers between 0 and and n-1 for some n.

RSA Algorithm breakdown

Key Generation

Step 1: Choose two large prime numbers called p and q (where p <> q)

Step 2: Calculate the value of n, n=p*q

Step 3: Choose public key as e, such e is not a factor of (p-1) and (q-1)

Step 4: Choose private key as d, such that d=e mod (p-1)*(q-1)=1

Encryption

Step 5: ct = (pt)*mod n

Plaintext= pt, Ciphertext = ct

Decryption

Step 6: pt = $(ct)^d$ mod n

Plaintext = pt, Ciphertext = ct

Uses of RSA Algorithm

RSA is a widely used cryptography in network environment and it supports the software and hardware as mentioned in the list below:

1. Provide a method of assuring confidentiality integrity.
2. Securing electronic communication and online data storage.
3. Use to secure Internet, social media, online shopping, and secure personal information such as credit cards.
4. RSA is used in security protocols such as IPSEC/IKE, TSL/SSL, PGP, SSH, SILC.
5. Used in military and government to secure communication.
6. Used in website and web based applications, internet browsers.
7. Used for signing digital signatures.
8. Very fast and simple encryption.
9. Easier to understand and implement.
10. Used in DRM (Digital Right Management).
11. Used in securedID token.
12. Widely deployed, better industry support.
13. Confidentiality, integrity, and authentication of electronic communication.
14. Prevents third party from intercepting message.

Limitations of RSA algorithm

1. Very slow key generation.
2. Slow signing and decryption while are slightly tricky to implement securely.
3. Key is vulnerable to various attacks if poorly implemented.

Practice Questions

1. What is a cryptosystem? Explain different classical cryptosystems in detail.
2. How does playfair cipher encryption algorithm works?
3. Explain about rail fence ciphers.
4. Differentiate between block and stream cipher.
5. Differentiate between symmetric key cryptography and asymmetric key cryptography.
6. What is fiestel cipher structure? Explain
7. Explain DES (Data Encryption Standards) in detail.
8. What is AES? Explain in detail.
9. Write a short note on Diffie-Helman Key Exchange.
10. Explain about RSA algorithm along with its uses and limitations.