Nija Khavangar
219302126
IT-6B

Date : _____

CRYPTOGRAPHY & INFORMATION SECURITY

ASSIGNMENT-1

Q1.

(a)

Q	A	B	R	T_1	T_2	T
1	11	10	1	0	1	-1
10	10	1	0	1	-1	11
	1	0		-1	11	

$T_1 = 0 \quad T_2 = 1$

$T = T_1 - T_2 \times Q$

$T_1 = -1$

To make multiplicative inverse positive : $-1 + 11 = 10$

$\therefore$ M.I = 10

(b)

Validity : $10 \equiv 3 \bmod 12$

$$12 \overline{)\,10\,} \quad 0 \quad \text{remainder } 10$$

$$12 \overline{)\,10\,} \quad 1 \quad \text{remainder } -2$$

R can be 10 or -2 but never 3

$\therefore 10 \equiv 3 \pmod{12}$ is not valid statement

Date :

(c) $P.T = \text{"POH"} = \begin{bmatrix} 15 \\ 14 \\ 7 \end{bmatrix}$ $\quad key = GYBNQKURP = \begin{bmatrix} G & Y & B \\ N & Q & K \\ U & R & P \end{bmatrix} = \begin{bmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{bmatrix}$

For encryption:

$$C = KP \bmod 26$$

$$C = \begin{bmatrix} 6 & 24 & 1 \\ 13 & 16 & 10 \\ 20 & 17 & 15 \end{bmatrix}_{3\times3} \begin{bmatrix} 15 \\ 14 \\ 7 \end{bmatrix}_{3\times1} = \begin{bmatrix} 15\times6+14\times24+7\times1 \\ 13\times15+16\times14+10\times7 \\ 20\times15+17\times14+15\times7 \end{bmatrix} = \begin{bmatrix} 433 \\ 489 \\ 643 \end{bmatrix} \bmod 26$$

$$C = \begin{bmatrix} 17 \\ 21 \\ 19 \end{bmatrix} = \begin{bmatrix} R \\ V \\ T \end{bmatrix}$$

$CT = RVT$

(d) $CT = \text{"ymnsp dtz hfs"} \quad key = 5$

For decryption: $P = (C - K) \bmod 26$

For y: $P = (24 - 5) \bmod 26 = 19 \bmod 26 = 19 = T$

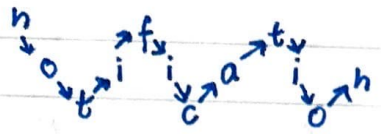
Similarly;

y	m	n	s	p	d	t	z	hfs
↓	↓	↓	↓	↓	↓	↓	↓	↓↓↓
t	h	i	n	k	y	o	u	can

PT: think you can

Date : _____

(e) PT: "notification" Column = 3



CT: nftoiiaintco

(f) ~~PT: message~~ PT: discovery

~~Key: google~~ key: googlegoo

key:	g	o	o	g	l	e	g	o	o
key No:	6	14	14	6	11	4	6	14	14
PT:	d	i	s	c	o	v	e	r	y
PT No:	3	8	18	2	14	21	4	17	24
CT No:	9	22	6	8	25	25	10	5	12
CT:	J	W	G	I	Z	Z	K	F	M

$C = (P + k) \bmod 26$

CT: JWGIZZKFM

(g) key: security

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

PT: "this is secret key"

↓

th is is se cr et ke yx

↓ ↓

CT: AF DI DI EC US TF FR AW

Date :

Q2. RSA algorithm is an asymmetric cryptography algorithm. It works on block cipher. It uses two different keys i.e. Public key and Private key.

Eg: A client request some data from server. The server encrypts the data using client's public key and sends the encrypted data. The client receives data and decrypts it using it's private key.

Given, $p = 3$, $q = 11$

$$n = p * q = 3 \times 11 = 33$$

$$\phi(n) = (p-1) \times (q-1)$$
$$\phi(n) = (3-1) \times (11-1)$$
$$\phi(n) = 2 \times 10$$
$$\phi(n) = 20$$

Let $e = 7$

Now, $d \equiv e^{-1} \bmod \phi(n)$

$$de \equiv 1 \bmod \phi(n)$$
$$de \bmod \phi(n) = 1$$
$$d \times 7 \bmod 20 = 1$$

$$\underline{d = 3}$$

Public key = $\{e, n\} = \{7, 33\}$

Private key = $\{d, n\} = \{3, 33\}$

Date : _____

Message = "01110101 11" M = 9

Encryption:

$C = M^e \bmod n$

$C = 9^7 \bmod 33$

$C = 15$

Decryption:

$M = C^d \bmod n$

$M = 15^3 \bmod 33.$

$M = 9$

Q3. Given; $q = 11$ $X_A = 97$ $X_B = 233$

primitive root; $\alpha = 2$

- **Key Generation of A:**

$Y_A = \alpha^{X_A} \bmod q$

$Y_A = 2^{97} \bmod 11$

$Y_A = 2$

- **Key Generation of B:**

$Y_B = \alpha^{X_B} \bmod q$

$Y_B = 2^{233} \bmod 11$

$Y_B = 8$

Secret key:

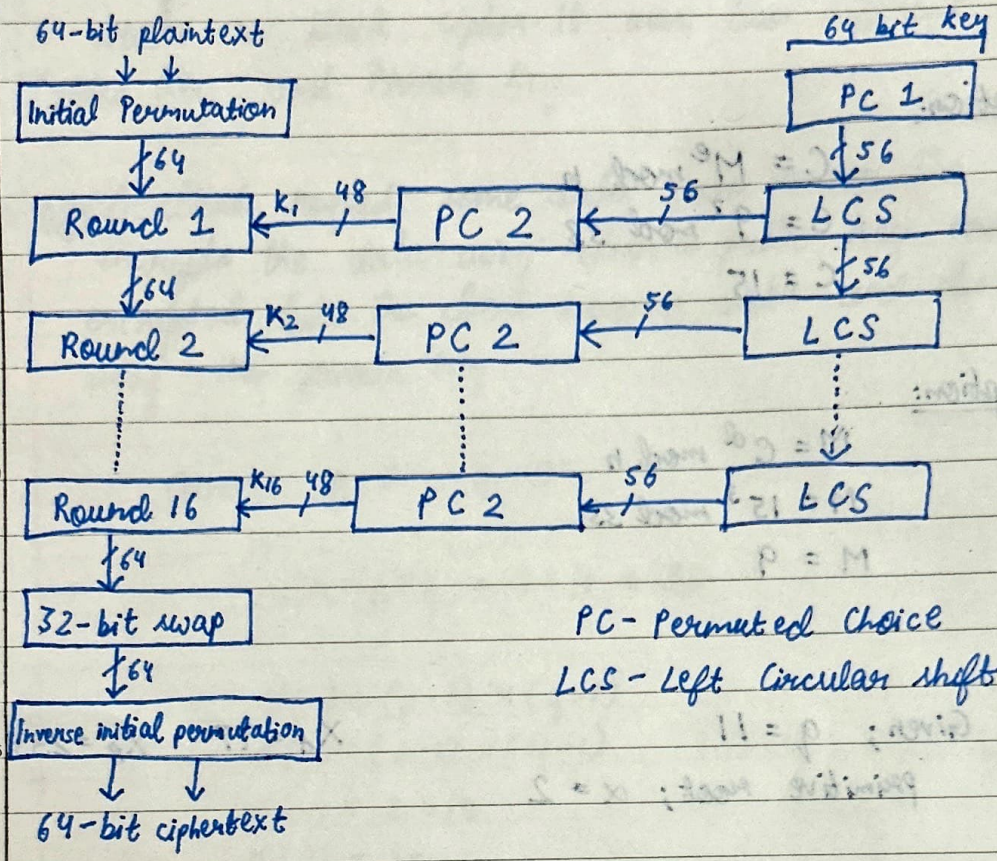
$K_A = (Y_B)^{X_A} \bmod q$ $K_B = (Y_A)^{X_B} \bmod q$

$K_A = (8)^{97} \bmod 11$ $K_B = (2)^{233} \bmod 11$

$K_A = 8$ $K_B = 8$

Thus, keys are exchanged.

Q4. General structure of DES:



PC - Permuted Choice
LCS - Left Circular shift

ENCRYPTION:

I Initial Permutation:

→ The 64-bit plaintext blocks undergoes an initial permutation to shuffle the bits according to a predefined permutation table.

II 16 Rounds:

→ The 64-bit plaintext block ~~under~~ is divided into 2 32-bit halves: left (L_0) and right (R_0)

→ Each round applies a round function to the right half of the block using a round key derived from a main key.

→ The output of the round function is 'XORed' with the left half of the block.

→ The left and right halves are swapped before proceeding to the

Date : _____

next ~~pou~~ round.

III Final Permutation:
→ After the 16 rounds, left and right halves ~~used~~ are combined and undergo a final permutation which is inverse of the initial permutation.

IV Cipher Text:
The resulting 64 bit block after final permutation is ciphertext.

DECRYPTION:
It is similar to encryption, except that the round keys are used in reverse order.

I. Initial Permutation:
Same as in encryption

II 16 Rounds (with round keys in reverse order)
→ Each round applies the round function to the right half of the block using the round key in reverse order.
→ The output of the round function is XORed with the left half of the block.
→ The left and right halves are swapped before proceeding to the next round.

III Final Permutation:
same as encryption.

IV Plaintext:
The resulting 64 bit block after the permutation is plain text.

Date :

Q5. (a)

Message Authentication code – MAC	Hash-based Message Authentication code – HMAC
• A mac is a fixed-length cryptographic tag generated by applying a secret key and a cryptographic hash function to to the message. The key is shared b/w sender and receiver.	• HMAC is a specific construction for generating MAC's using a cryptographic hash function and in combination with a secret key. HMAC uses 2 passes of hash function to incorporate key securely into calculation.
• It depends on the strength of the cryptographic hash function and secrecy of the key. Weaker hash function or insufficiently long keys can lead to vulnerabilities.	• It is designed to provide additional security guarantees compared to generic MAC algorithms. It is resistant to certain classes of attacks due to its specific construction.
• Can be constructed in various ways and there are several standardized MAC algorithms, HMAC, CBC-MAC, CMAC.	• It is specific construction for generating MACs and is standardised.
• It is variously used in crypto protocols and applications such as SSL/TLS, IPs, message auth in network protocols.	• It is commonly used in security protocols and applications where message integrity and vulnerability are crucial, like VPN, SSH, email.

(b)

Date :

(b)

AES key generation

Key Expansion:

- AES operates on three key lengths 128, 192 and 256.
- Key expansion transforms the original key into set of round keys.
- For 128 AES, 128 bit key is divided into 4-byte words. Additional round keys are generated using a key schedule algo that involves operations like subBytes, Shift rows, Mix columns and Round constants.
- For AES-192 and 256, process is similar but it involves more rounds & larger key schedule.

Sub keys:

- They are derived from the key expansion and used in each round of encryption & decryption.
- Each round key is used to perform operations on plain text plain such as substitution permutation and mixing.
- AES employs key expansion to generate round keys.

DES key Generation

Key Generation:

- It operates with a fixed length of 56 bits.
- The 56-bit key is permuted and split into two 28-bit halves which are used in key schedule algorithm.
- The key algorithm generates the 16 round keys, one for each round of encryption and decryption.
- Each round key is derived from the 56-bit key using a combination of shifting permutation and compression operations.

Round keys:

- The round keys are used in each round of DES algorithm.
- Each round key is generated from the original key using the key schedule algorithm and is specific to a particular round of encryption & decryption.
- DES user key scheduling algorithm.

Date :

(c)

Mono Alphabetic Cipher	Poly Alphabetic Cipher
→ A mono alphabetic cipher is a substitution cipher where each letter in the plain text is replaced by the same corresponding letter or symbol in the cipher text.	→ A polyalphabetic cipher is also a substitution cipher where different substitution alphabets are used in encryption process.
→ Example: Ceasar cipher, where each letter in plain text is shifted by a fixed number of positions in the alphabet.	→ Example: The vigenere cipher is a well known example of polyalphabetic where each letter in the plaintext is shifted by a different amount on depending with the position of the key.
→ Each letter in plaintext is replaced by its correspondent letter in the cipher text acc. to fixed substitution table.	→ The plaintext is combined with the repeating key phrase and each letter in plaintext is shifted by the corresponding letter in key.
→ It can be cracked easily because of frequency.	→ It is more insecure. They introduce variability and make frequency attacks different.

Date : _____

Q6. $n = 3233$

For prime no. p and q such that $3233 = p \times q$
value should be $p = 53$
$q = 61$

$\phi(n) = (p-1) * (q-1)$
$\phi(n) = 52 * 60$
$\phi(n) = 3120$

Let $e = 17$

$d \equiv e^{-1} \bmod \phi(n)$
$de \equiv 1 \bmod \phi(n)$
$de \bmod \phi(n) = 1$
$d \times 17 \bmod 3120 = 1$

$d = 2753$

Q7. $p = 41$
$g = 7$

Secret no (private key) : Alice : $a = 10$
Bob : $b = 20$

Public key generation:

Alice
$Y_A = 7^{10} \bmod 41$
$Y_A = 9$

Bob
$Y_B = 7^{20} \bmod 41$
$Y_B = 40$

Secret key: $K_A = (40)^{10} \bmod 41$
$K_A = 1$

$K_B = (9)^{20} \bmod 41$
$K_B = 1$

Date :

Q8 $p = 17 \quad q = 11$

$n = p \times q$

$n = 17 \times 11$

$n = 187$

$\phi(n) = (p-1) * (q-1)$

$\phi(n) = (17-1) * (11-1)$

$\phi(n) = 16 \times 10$

$\phi(n) = 160$

given, $e = 17$

$ed \bmod \phi(n) = 1$

$7d \bmod 160 = 1$

$d = 23$

Q	A	B	R	T_1	T_2	T
22	160	7	6	0	1	-22
1	7	6	1	1	-22	23
6	6	1	0	-22	23	x
	1	0		23		

23 is M.I

Date : _____

Q9. $391x + 299y = \gcd(391, 299)$

Q	A	B	R	S_1	S_2	S	T_1	T_2	T
1	391	299	92	1	0	1	0	1	-1
3	299	92	23	0	1	-3	1	-1	4
4	92	23	0	1	-3	13	-1	4	-17
	23	0		-3	13		4	-17	

GCD = 23, x = -3, y = 4

$\Rightarrow 391(-3) + 299(4)$

$\Rightarrow 23$

$\Rightarrow \gcd(391, 299)$

Q10.

Q	A	B	R	S_1	S_2	S	T_1	T_2	T
1	26	17	9	1	0	1	0	1	-1
1	17	9	8	0	1	-1	1	-1	2
1	9	8	1	1	-1	2	-1	2	-3
8	8	1	0	-1	2	-17	2	-3	22
	1	0		2	-17		-3	22	

$\Rightarrow 17x + 26y = \gcd(17x + 26y)$

LHS: $17x + 26y$

$\Rightarrow 17(-3) + 26(2)$

$\Rightarrow 1$

$\rightarrow$ RHS

Q11. $w_0 = 2b7e1516$

$w_1 = 28aed2ab$

$w_2 = abf71588$

$w_3 = 09cf4f3c$

$w_0 = 00101011 \quad 01111110 \quad 00010101 \quad 00010110$

$w_3 = 00001001 \quad 11001111 \quad 01001111 \quad 00111100$

Inside g-function:

⇒ Left shifting w_3 :

11001111 01001111 00111100 00001001

→ Look up in S-table:

ae b2 e2 30

→ adding constant R

$$\begin{array}{cccc} ae & b2 & e2 & 30 \\ \oplus\ 00000001 & 0 & 0 & 0 \\ \hline af & b2 & e2 & 30 \end{array}$$

→ Performing XOR with w_0:

$$\begin{array}{cccc} af & b2 & e2 & 30 \\ \oplus\ 7b & 7e & 15 & 16 \\ \hline 84 & cc & f7 & 26 \end{array} = w_5$$

$w_5 = 84\ cc\ f7\ 26$

Q12.

Date : _____

Q12. i/p = 101111

row = 11 = 3

column = 0111 = 7

S-value in table for (3, 7) = 7

= 0111