MAWLANA BHASHANI SCIENCE&TECHNOLOGY UNIVERSITY

Assignment

Department of : Information & Communication Technology

Assignment No : 01

Name of the assignment : Questions of cryptograph

Course Title : Advanced Cryptograph

Course Code : ICT-6115

Submitted by

Name : Iftakher Ahamed Hasib

ID : IT23622 Session : 2022-23

Year : 5th Semester : 1st

Dept. of ICT MBSTU

Submitted to

Dr. Ziaur Rahman

Associate professor

Dept. of ICT

MBSTU.

Date of Performence :

Date of Submission :

IT-23622

NO-1

Ans:

Quantum computing poses a significant threat to traditional cryptographic protocols, particularly by public-key cryptosystems like RSA and ECC. The primary reason is Shor's algorithm which can efficiently factor large integers and solve the discrete algorithm problem in ploynomial time.

The implication includes:

1. Loss of confidentiality.
2. Compromised integrity.
3. Long-term security risk.

post- Quantum Cryptographic Algorithms.

To counter the quantum threat, researchers are developing post-quantum cryptography (PQ algorithms that are resistant to quantum attacks.

1. Lattice-based cryptograph:

→ Algorithms: crystal-kyber, CRYSTAL-DILI

→ Resistance: The security is based on hard lattice problems such as the Learning with errors problem.

→ Strengths: Efficient operations and well-understood security foundations.

2. Code based cryptography:

→ Algorithm: Classic McEliece

→ Resistance: Based on the difficulty

of decoding random linear codes.

3. Hash-based Cryptography:

→ Algorithm: Stateless Hash-based-signature

→ Resistance: Based on the security of cryptographic has functions which are resistant to quantum attacks.

→ Strengths: Stateless design ensures security without requiring state tracking.

Unlike RSA and ECC which relay to integer functions and discrete logarithms, ~~ECC~~ PQC algorithm rely on problems that are either not efficiently solvable by quantum algorithms or only offer a minimal advantages to quantum attackers.

For instance:

→ Lattice problems remain hard even with quantum speedups.

→ Code-based cryptograph relies on an error-correcting problem that quantum computers cannot solve efficiently

→ Hash-based cryptography is based on one-way functions that remain resistant to Grover's algorithm, which only provides a quardratic speedup for brute-force attack.

IT-23622

NO-2

Implementation:

```
import time
import os

class CustomPRNG:
    def __init__(self, seed=None, mod=100):
        if seed is None:
            self.seed = int(time.time()*1000000)^os.getpi
        else:
            self.seed = seed
        self.mod = mod

    def next(self):
        self.seed ^= (self.seed << 13) & 0xFFFFFFFF
        self.seed ^= (self.seed >> 7) & 0xFFFFFFFF
        self.seed ^= (self.seed << 17) & 0xFFFFFFFF
        return abs(self.seed) % self.mod
```

```
def random_list(self, size):
    return [self.next() for _ in range(size)]

# Example usage:
prng = CustomPRNG(mod=1000)
print(prng.next())
print(prng.random_list(5))
```

IT-23622

NO-3

Comparison between Traditional ciphers and modern symmetric ciphers:

Feature	Traditional ciphers	modern symmetric ciphers
Key legth	short (1-26 for Ceasar a few words for vigenere)	Long (56-bit to DES, 128/192/256 bit for AES)
Encryption speed	fast	fast but computationally heavier
Decryption speed	same as encryption	similar to encryption optimize for speed
security	weak against brute force, frequency analysis and pattern detection.	strong ayain brute force and statistical attack.
vulnerability to cryptanalysis	Highly vulnerable.	Resistant to linear and differential cryptanalysis

IT-23622

strength and weakness

Traditional ciphers:

① Caesar ciphers

strength: simple and easy to implement.

weaknesses: only 25 possible keys, vulnerable to frequency analysis.

② vigenere cipher:

strength: uses a key phrase, making frequency analysis harder than the caesar cipher.

weaknesses: If the key length is known, it can be broken using the kasiski examination.

③ playfair cipher

strengths: Encrypts two letters at a time, reducing frequency analysis effectiveness.

weaknesses: still vulnerable to frequency analysis if enough ciphertext is availabl

modern symmetric ciphers

① DES (Data Encryption Standard):

strengths: first widely adopted symmetric cipher, structure provides diffusion and confusion.

weaknesses: 56-bit key is too short, making it vulnerable to brute-force attack.

IT-23622

② AES (Advanced Encryption Standard):

Strengths: uses key length of 128, 192 or 256 bits, making brute-force impractical.

Weakness: Complex implementation, requires high computational power for embedded systems.

No-4 |

Defining the Action of S_4 on 2-Element Subsets

The symmetric group S_4 consists of all permutations of the set $X = \{1, 2, 3, 4\}$, we define an action of S_4 on the set of 2-element subsets of X as follows:

for any $\sigma \in S_4$ and any subset $\{a, b\}$ where $a, b \in X$ and $a \neq b$ define:

$$\sigma \cdot \{a, b\} = \{\sigma(a), \sigma(b)\}$$

proving the Action is well-Defined

To show that this action is well-defined, we must verify:

① The image of a 2-element subset under any permutation is still a 2-element subset

(2) The identity element of S_4 acts trivially.

(3) The composition of two permutations behaves as expected.

Closure: If $\{a, b\}$ is a 2-element subset X, then for any $\sigma \in S_4$ $\sigma(a) \neq \sigma(b)$ because σ is a bijection. Hence, $\sigma \cdot \{a, b\} = \{\sigma(a), \sigma(b)\}$ is still a 2-element subset.

Identity Action: The identity permutation e satisfies $e \cdot \{a, b\} = \{e(a), e(b)\} = \{a, b\}$.

Thus the action is well-defined.

Computing the orbit of $\{1,2\}$

The orbit of $\{1,2\}$ under the action consists of all 2-element subsets that can be reached by applying some permutation in S_4

Since S_4 acts transitively on the 2-element subsets of X, the orbit of $\{1,2\}$ includes all possible 2-element subsets of X, namely,

$$\{1,2\}, \{1,3\}, \{1,4\}, \{2,3\}, \{2,4\}, \{3,4\}$$

There are $\binom{4}{2} = 6$ such subsets, meaning the size of the orbit of $\{1,2\}$ is 6.

NO-5

Ans:

We are given the finite field $GF(2^2)$, which is constructed using the irreducible polynomial

$$x^2 + x + 1$$

Constructing $GF(2^2)$

Since $GF(2^2)$ is a degree-2 extension of $GF(2)$, we define an element α as a root of the irreducible polynomial:

$$\alpha^2 + \alpha + 1 = 0$$

$$\Rightarrow \alpha^2 = \alpha + 1 \quad [\text{rearranging}]$$

Since $GF(2) = \{0, 1\}$ we construct the elements of $GF(2^2)$ as:

$$GF(2^2) = \{0, 1, \alpha, \alpha+1\}.$$

IT-23622

we know consider the non zero elements:

$$E = \{1, \alpha, \alpha+1\}.$$

i) Showing that E forms a Group under multiplication

To verify that E forms a group under multiplicatio we check the group properties:

① Closure: we compute the product:

→ $1.\alpha = \alpha, 1 \cdot (\alpha+1) = \alpha+1$ and $1.1 = 1$

→ $\alpha \cdot (\alpha+1) = \alpha^2 + \alpha = (\alpha+1) + \alpha = 1$

→ $(\alpha+1) \cdot (\alpha+1) = \alpha^2 + 2\alpha + 1 = (\alpha+1) + 2\alpha + 1 = \alpha + 2\alpha + 2$

→ $\alpha . \alpha . \alpha^2 = \alpha + 1$

since all products remain in E, closure holds.

② Associativity:

Since $GF(2^2)$ is a field, multiplication is associative.

③ Identity elements:

The identity element in multiplication is 1, since for all $x \in E$.

$$x \cdot 1 = x.$$

Thus $E = \{1, \alpha, \alpha+1\}$ forms a group under multiplication.

(i) verifying if E is cyclic

A group is cyclic if there exists an element $g \in E$ such that all elements of E can be written as power of g.

we check if α can generate all elements:

$$\alpha^1 = \alpha, \alpha^2 = \alpha+1, \alpha^3 = (\alpha+1), \alpha = 1 :$$

Thus the powers of α cycle through all elements, meaning α is a generator of E. E is cyclic.

IT-23622

NO-6

Ans:

Define the General Linear Group GL(2,R)

The general linear group GL(2,R) consists of all 2×2 invertible matrices over R i.e.,

$$GL(2,R) = \{A \in M_{2\times 2}(R) \mid \det(A) \neq 0\}.$$

This is a group under matrix multiplication.

Define the set of scalar matrices

A scalar matrix is a multiple of the identity matrix:

$$S = \{\lambda I \mid \lambda \in R^*\} = \{\lambda \mid \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix} \mid \lambda \neq 0\}.$$

Since λI is invertible for all $\lambda \neq 0$.

Constructing the Factor Group:

The quotient group $GL(2,\mathbb{R})/S$ consists of cosets of the form:

$$[A] = AS = \{A(\lambda I) \mid \lambda \neq 0\}.$$

Since λI scales all element uniformly, two matrices A and B belong to the same coset if and only if they differ by a scalar multiple:

$$A \sim B \iff B = \lambda A \text{ for some } \lambda \neq 0$$

This means that the cosets represent equivalence classes of matrices under scalar multiplication.

IT-23622

NO-7

Ans:

Diffie-Hellman key Exchange protocol

The Diffie-Hellman (DH) key exchange is a cryptographic protocol that allows two parties to securely establish a shared secret over an insecure channel without directly transmitting the secret itself.

Steps of the protocol:

1. Public parameters selection
2. key exchange Between Two parties:
3. Shared secret Computation:

Since exponentiation is associative both Alice and Bob derive the same shared secret

$$S = g^{ab} \bmod P$$

IT-23622

Security of the Diffie-Hellman protocol

~~The security~~ It relies on the difficulty of solving the Discrete Logarithm problem

Potential Attacks and Defenses

1. Man-in-the-middle Attack:

Attack: An attacker intercepts messages and establishes separate key exchanges with Alice and Bob.

Defense: Use authenticated key exchange to verify identities.

2. Brute force or precomputation Attacks:

Attack: If the prime p is small, an attacker can precompute logarithms for all value

Defense: use large primes to prevent feasibility of such attack.

IT-23622

Impact of using a small prime modulus:

If the prime is not sufficient large, the following security risk arise;

1. Efficient Brute force and precomputation.
2. Faster computation Attacks;
3. Quantum Computing Threat.

NO-8

Ans:

proof:

Let G be a group and let H and K be two subgroups of G. we want to show th the intersection $H \cap K$ is also a subgrou

Step-1: Show $H \cap K$ is non-empty.

Since H and K are subgroups, they both contain the identity element e of G,

$e \in H$ and $e \in K$.

Thus, $e \in H \cap K$, meaning $H \cap K$ is non-empty

Step-2: Closure under multiplication

Let $a, b \in H \cap K$.

Since $H \cap K$ consists of element.

$a, b \in H$ and $a, b \in K$

Since both H and K are subgroups, they are closed under multiplication, so:

$ab \in H$ and $ab \in K$.

Thus, $ab \in H \cap K$, proving closure under multiplication.

So, we conclude that $H \cap K$ is a subgroup of G.

Example:

Consider the group of integers under addition $G = \mathbb{Z}$ and let:

- $H = 2\mathbb{Z} = \{\ldots -4, -2, 0, 2, 4 \ldots\}$
- $K = 3\mathbb{Z} = \{\ldots, -6, -3, 0, 3, 6 \ldots\}$

The intersection $H \cap K$ consists of all integers that are both even and divisible by 3 and 6.

$H \cap K = 6\mathbb{Z} = \{\ldots, -12, -6, 0, 6, 12 \ldots\}$

$6\mathbb{Z}$ is a valid subgroup of $\mathbb{Z}$.

IT-23622

NO-10:

Ans:

Vulnerabilities of the DES cipher:

The Data Encryption standard, developed in the 1970s, was one of the most widely used symmetric encryption algorithms. However due to advancement in computing power and cryptanalysis, DES is now considered insecure for modern applications. The main vulnerabilities of DES includes:

1. Short key length.
2. Brute-force-Attacks
3. cryptanalytic weakness
4. Small Block size.

Brute force Attack Break DES:

A brute-force attack systematically tries all possible keys until the correct is found.

→ with 56-bit key, there are $2^{56} \approx 72 \times 10^{15}$ possible keys.

→ modern hardware, such as ASICs, FPGAs, and cloud based parallel processing can exhusted this key space quickly.

→ Example: A modern high-performance cluster with specialized can test hundreds of billions of keys per sec

IT-23622

AES Addressed the shortcomings of DES:

The Advanced Encryption Standard (AES) was introduced in 2001 to replace DES and overcome its weakness:

→ Increased the key size

→ Resistance to cryptanalytic Attacks

→ Large block size.

IT-23622

NO-11

Ans:

(i) Differential cryptanalysis is a chosen-plaintext attack that analyze how difference in plaintext propagate through a cipher to predict differences in ciphertext.

Defense mechanisms in DES against DC:

1. S-Box Design to resist DC

The s-boxes in DES were carefully designe to minimize differential probabilities.

2. Feistal structure provides

In this feistal network of DES, the right half of the block is expanded, mixed with the round key, and substituted via s-boxes,

3. Key scheduling prevents simple DC attack

The key schedule in DES ensures that subkeys change across rounds, making it harder for an attacker to track.

(ii)

Unlike DES, AES is not a feistal cipher but follows a substitution-permutation network structure to DC.

Key features that improve DC Resistance

(i) Sub-bites (Non-linear substitution using S-boxes)

(ii) Shiftrows (Row-wise permutation for Diffusion)

(iii) mix columns for strong Diffusion.

(iv) Addroundkey.

(v) more rounds in AES as AES-128 has 10 rounds.

IT-23622

NO-12

Ans:

Finding the modular Inverse using the Extended Euclidean Algorithm:

The modular inverse of an integer a modulo n is an integer x such that:

$$a \cdot x \equiv 1 \pmod{n}$$

This means that x is the multiplicative inverse of a modulo n, provided that a and n are coprime (i.e. gcd(a,n)=1).

we use the Extended Euclidean Algorithm (EEA) to compute x.

Step:-1: Apply the Euclidean Algorithm

The Euclidian Algorithm finds the greatest common divisor (gcd) of a and n using

the division algorithm:

$$gcd(a,n) = gcd(n, a \bmod n)$$

we continue until we reach gcd = 1.

Step-2: Apply the Extended Euclidean algorithm.

The EEA express gcd(a,n) as a linear combination:

$$gcd(a,n) = ax + ny$$

since gcd(a,n) = 1, we can rewrite this as

$$1 = ax + ny$$

Reducing modulo n:

$$ax \equiv 1 \pmod{n}$$

Thus, x is the modular inverse of a modulo n.

IT-23622

Use of the Extended Euclidean Algorithm in RSA key Generation:

In RSA encryption, two large prime numbers p and q are chosen to form:

$$n = p \times q$$

The Euler totient function is computed as:

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

A public exponent e is chosen such that:

$$\gcd(e, \phi(n)) = 1$$

To find the private key d, we compute:

$$d \equiv e^{-1} \pmod{\phi(n)}$$

This requires finding the modular inverse of e modulo $\phi(n)$, which is efficiently computed using the Extended Euclidean Algorithm.

IT-23622

Importance of the algorithm's efficie in cryptography is:

→ Large prime moduli

→ Polynomial time complexity O(log n

→ Avoiding Brute-force methods.

NO-13

(i) ECB mode is insecure for highly redundant Data

In Electronic Codebook (ECB) mode, a plain message P is divided into fixed size blo and each block is independently encryp using the same key K:

$$C_i = E_K(P_i)$$

Mathematical proof of ECB weakness

1. Lack of Diffusion:

Indentical plaintext Blocks produce identical ciphertext Blocks.

suppose we have two plaintext block P_i and P_j, such that:

$$P_i = P_j$$

since encryption in ECB is deterministic:

$$C_i = E_K(P_i) = E_K(P_j) = C_j$$

This means that identical plaintext blocks always produce indentical ciphertext block which leaks information about the structure of the plaintext.

NO-14

Ans:

A linear feedback shift register generates a repeating sequence of bits using a linear formula.

$$S_n = c_1 S_{n-1} \oplus c_2 S_{n-2} \oplus \cdots \oplus c_m S_{n-m}$$

where,

→ S_n are the output bits.

→ c_i are fixed number.

→ $\oplus$ is XOR.

This linear means an attacker can set up simple equations and solve them to find the LFSR's structure.

IT.23622

A hacker Breaks LFSR Encryption:

A stream cipher using an LFSR encrypts data like this:

$$c_i = P_i \oplus k_i$$

→ P_i = Plaintext bit.

→ k_i = LFSR generated keystream bit.

→ c_i = cipher text bit.

If a hacker know that both P_i and c_i they can recover k_i,

$$k_i = c_i \oplus P_i$$

Since the keystream follows a linear ru the hacker can use math tries to find all future k_i, This breaks the encryption.

IT-23622

NO-15]

Ans:

1) Claude Shannon defined perfect secrecy mathematically as:

$$P(m|c) = P(m)$$

for all plaintext m and ciphertext c, where:

→ $P(m)$ is the probability of choosing a plaintext m.

→ $P(m|c)$ is the probability of m given that we observe c.

This means that knowing the ciphertext gives no additional information about the plaintext.

Using Bayes theorem, we can rewrite the condition as:

$$P(c|m) = P(c)$$

(ii) proof that the one-Time pad (OTP) :

Definition:

- Let m be a plaintext message from the set M
- Let K be a key chosen uniformly at random from the keyspace K.
- Encryption is defined as:

$$C = m \oplus K$$

- Decryption work as:

$$m = c \oplus K$$

The OTP satisfies shannon's Definition:

we need to prove that Knowing c does not reveal any information about m.

1. Since K is chosen uniformly at random, for any given plaintext m, the ciphertext is.

$$C = m \oplus K$$

2. The key is equally likely to be any value in K, and since $|K| \geq |M|$, every plaintext has an equal chance of producing any ciphertext.

3. Thus, for any given ciphertext C, every plaintext m is equally probable:

$$P(m|c) = P(m)$$

which satisfies Shannon's definition of perfect secrecy.

IT-23622

No-16

Let's choose specific values for the LCG parameters

→ multiplier: $a = 5$

→ Increment $c = 3$

→ modulus $m = 16$

→ seed $x_0 = 7$

The recurrence relation is:

$$x_{n+1} = (a x_n + c) \bmod m$$

substituting our values:

$$x_{n+1} = (5x_n + 3) \bmod 16$$

Now, we compute the first 5 values.

1. $x_1 = (5 \times 7 + 3) \bmod 16 = (35 + 3) \bmod 16 = 38 \bmod 16 = 6$

2. $x_2 = (5 \times 6 + 3) \bmod 16 = (30 + 3) \bmod 16 = 33 \bmod 16 = 1$

3. $x_3 = (5 \times 1 + 3) \bmod 16 = (5 + 3) \bmod 16 = 8 \bmod 16 = 8$

4. $x_4 = (5 \times 8 + 3) \bmod 16 = (40 + 3) \bmod 16 = 43 \bmod 16 = 11$

5. $x_5 = (5 \times 11 + 3) \bmod 16 = (55 + 3) \bmod 16 = 58 \bmod 16 = 10$

The sequence is: 6, 1, 8, 11, 10 Ans:

NO-18

Ans:

RSA Encryption & Decryption:

Given values:

$p = 5, q = 11,$

$n = p \times q = 5 \times 11 = 55$

Compute Euler's totient function $\phi(n)$:

$\phi(n) = (p-1)(q-1) = (5-1)(11-1) = 40$

Step-1 Choose public key e

e must be co prime to $\phi(n) = 40$.

Choose $e = 3$ (since $\gcd(3, 40) = 1$).

Step-2 Compute private key d

d is the modulus inverse of e modulo $\phi(n)$, satisfying

$d \times e \equiv 1 \bmod 40$

Using the Extended Euclidean Algorithm,

$d = 27$, since $(3 \times 27) \bmod 40 = 81 \bmod 40 = 1$

Thus, our RSA key pair;

public key $(e, n) = (3, 55)$

private key $d = 27$

Step-3 Encrypt message $m = 2$

using RSA encryption formula

$$C = m^e \pmod n$$

$$C = 2^3 \bmod 55 = 8$$

The ciphertext is $e = 8$

Step-4: Decrypt ciphertext $e = 8$

using RSA decryption formula:

$$m = e^d \bmod n$$

$$m = 8^{27} \bmod 55$$

$$= 2$$

Ans

IT-23622

RSA Digital Signature

Given values,

$p = 7, q = 3, n = p \times q = 7 \times 3 = 21$

Compute $\phi(n)$:

$$\phi(n) = (p-1)(q-1) = (7-1)(3-1) = 12$$

Choose e such that $\gcd(e, 12) = 1$ Let's take $e = 5$

Compute private key d as the modular inverse of e modulo $\phi(n)$:

$$d \times 5 \equiv 1 \bmod 12$$

Using the Extended Euclidean Algorithm,

$$d = 5$$

Thus, our RSA key pair:

Public key $(e, n) = (5, 21)$

private key $d = 5$

IT-23622

Step-1: Sign hash of message H(m)=3 using RSA signature formula:

$$S = H(m)^d \bmod n$$

$$S = 3^5 \bmod 21$$

Computing: $3^5 = 243$

$$243 \bmod 21 = 243 - (21 \times 11) = 243 - 231 = 12$$

The signature is S=12

Step2: verify the signature

To verify, we compute:

$$H'(m) = S^e \bmod n$$

$$H'(m) = 12^5 \bmod n$$

$$= 3$$

since $H'(m) = H(m)$ the signature is verified.

IT-23622

NO:9

Ans:

we are given the elliptic curve equation:

$$y^2 \equiv x^3 + ax + b \bmod p$$

with parameters:

$p = 23$, $a = 1$, $b = 1$

(i) verify if $P = (3, 10)$ lies on the curve

To check if the point $P = (3, 10)$ lies on the curve, substitute $x = 3$, $y = 10$ into the equation. $10^2 \equiv 3^3 + 1(3) + 1 \bmod 23$

$$100 \equiv 27 + 3 + 1 \bmod 23$$

$$100 \equiv 31 \bmod 23$$

Since, $31 \bmod 23 = 8$ and $100 \bmod 23 = 8$

both sides are equal. Thus, P lies on the curve.

ii) Doubling the point P (Computing 2P)

The formula for point doubling is:

$$\lambda = \frac{3x_1^2 + a}{2y_1} \bmod p$$

$$x_2 = \lambda^2 - 2x_1 \bmod p$$

$$y_2 = \lambda(x_1 - x_2) - y_1 \bmod p$$

Substituting, P = (3,10):

$$\lambda = \frac{3(3^2)+1}{2(10)} \bmod 23$$

$$= \frac{3(9)+1}{20} \bmod 23$$

$$= \frac{28}{20} \bmod 23$$

since division in modular arithmatic requires the modular inverse of 20 modulo 23, we compute

$$20^{-1} \bmod 23$$

using the Extended Euclidean Algorithm we find

$$20^{-1} \equiv 7 \bmod 23 \text{ so:}$$

$$\lambda = 28 \times 7 \bmod 23$$

$$= 196 \bmod 23 = 12$$

Now Compute x_2:

$$x_2 = \lambda^2 - 2x_1 \bmod 23$$

$$= 12^2 - 2(3) \bmod 23$$

$$= 144 - 6 \bmod 23$$

$$= 138 \bmod 23$$

$$= 138 - (23 \times 6) = 138 - 138 = 0$$

Compute y_2:

$$y_2 = \lambda(x_1 - x_2) - y_1 \bmod 23$$

$$= 12(3 - 0) - 10 \bmod 23$$

$$= 36 - 10 \bmod 23$$

$$= 26 \bmod 23$$

$$= 3$$

Thus $2P = (0, 3)$

Ans

NO-20 |

Ans:

The curve equation is: $y^2 \equiv x^3+7x+10 \bmod 37$

$G = (2,5)$, $n = 19$, $d = 9$, $k = 3$, $H(m) = 8$

Step-1 : Compute the public key $Q = dG$

Since the public key is obtained by scalar multiplication of the best point:

$$Q = d \cdot G = 9 \cdot (2,5)$$

we compute $9G$ using double and-add.

Compute $2G$ (Point doubling)

formula:

$$\lambda = \frac{3x_1^2 + a}{2y_1} \bmod p$$

$$x_2 = \lambda^2 - 2x_1 \bmod p$$

$$y_2 = \lambda(x_1 - x_2) - y_1 \bmod p$$

For $G = (2,5)$, we have:

$$\lambda = \frac{3(2^2)+7}{2(5)} \bmod 37$$

$$= \frac{19}{10} \bmod 37$$

We compute the modular inverse of 10 mod 37:

$10^{-1} \equiv 26 \bmod 37$

$\lambda = 19 \times 26 \bmod 37$

$= 494 \bmod 37$

$= 494 - (37 \times 13) = 494 - 481 = 13$

Now,

$x_2 = 13^2 - (2)2 \bmod 37$

$= 165 \bmod 37$

$= 165 - (37 \times 4) = 165 - 148 = 17$

$y_2 = 13(2-17) - 5 \bmod 37$

$= -200 \bmod 37$

$= -200 + (37 \times 6) = -200 + 222 =$

$= 22$

$\therefore 2G = (17, 22)$

Continuing this process, we compute $9G$ to get,

$Q = (x_9, y_9)$

STEP-2: Compute the signature (r, s)

Compute r:

$$r = x_1 \bmod n$$

Compute s:

$$s = k^{-1}(H(m) + dr) \bmod n$$

where k^{-1} is the modular inverse of $k \bmod n$.

Using Extended Euclidean Algorithm compute $k^{-1} \bmod n$

then, compute s.

To verify the signature check:

$$r \equiv (u_1 G + u_2 Q)_x \bmod n$$

where:

$$u_1 = H(m) s^{-1} \bmod n$$

$$u_2 = r s^{-1} \bmod n$$

Compute $u_1 G + u_2 Q$ and confirm that its x-coordinate matches r.

NO-22

Ans:

A Galois field (GF(q)) is a finite set of elements where arithmetic operations are defined.

Types of Galois field's:

1. GF(P) (prime fields):

→ elements: {0, 1, 2 ..., p-1}, where P is prime

→ Used in Elliptic curve cryptography for secure encryption, digital signatures and key exchange.

2. GF(2^n) (Binary fields):

→ Elements are binary polynomials modulo an irreducible polynomial.

→ Used in AES encryption, error correction and post-quantum cryptography.

IT-23622

Importance in cryptography:

→ Efficient computation: Fast arithmetic in finite fields improves performance.

→ Security: Provides resistance to attack to attack like discrete logarithm problems.

→ Compact Representation: useful for hardware and embedded system.

→ Mathematical Rigor: Ensure cryptographic algorithms function correctly.

NO-23]

Ans:

① The shortest vector problem (SVP) is a fundamental hard problem in lattice-based cryptography. Given a lattice, the SVP asks for the shortest non-zero vector in the lattice under a chosen norm. The problem is computationally hard, meaning even the best known algorithms require exponential time for large lattices.

Role in security:

→ many lattice-based cryptographic schemes rely on the difficulty of approximating SVP.

IT-23622

→ learning with error and Ring-LWE key problem in lattice cryptography are reducible to SVP.

→ even quantum computers are not known to solve SVP efficiently, making lattice-based cryptography post-quantum secure.

ii)

cryptographic scheme	security Basis	vulnerability to Shor's Algorithm
RSA	Integer factorization problem	Broken by shor's algorithm
Elliptic curve cryptography	Elliptic curve Discrete log problem	Broken By shor's algorithm
Lattice based cryptography	Lattice problems. (SVP, LWE)	not efficiently solvable by quantum algorithms.

IT-23622

NO-24

Ans:

Step-1:

A Linear ~~shift~~ feedback shift register is defined by the recurrence relation:

$$K_t = c_1 K_{t-1} \oplus c_2 K_{t-2} \oplus \cdots \oplus c_m K_{t-m}$$

where,

→ The co-efficient $c_1, c_2 \cdots c_m$ belong to GF(2)

→ The operations are performed modulo 2.

→ The sequence of keystream bits $\{K_t\}$ is periodic, meaning it eventually repeats.

This recurrence relation corresponds to a characteristic polynomial:

$$P(x) = x^m - c_1 x^{m-1} - c_2 x^{m-2} - \cdots - c_m$$

Step-2:

The state of an LFSR at any time is completely determined by the m bit sequence. since there are m bits, the total number is

$$2^m$$

However the all zero state (000...000) is not allowed in a maximal length LFSR, since it would be produce an output stream of all zero indefinitely. Therefore the maximum possible non zero state is:

$$2^m - 1$$

Thus, the maximum possible period of the key stream is $2^m - 1$.

IT-23622

NU-25

Ans:

(1) An LWE-based signature scheme consists of three main steps:

1. Key Generation:

→ Generate a private key SK.

→ compute the public key PK using a matrix A and the LWE problem structure.

2. Signing:

→ Hash the message m to create a challenge.

→ use the private key SK and a trapdoor function to produce a short lattice vector.

3. verification:

→ use the public key P_K to check

→ If the verification equation holds, the signature is valid.

(ii)

step-1:

→ Choose a random matrix $A \in \mathbb{Z}_q^{n \times m}$

→ Generate a secret key S_K (a short vector)

→ Compute the public key:

$$P_K = A \cdot S_K + e$$

where e is a small error vector sampled from a noise distribution.

step-2:

→ Hash the message to obtain a challenge:

$$H(m) \in \mathbb{Z}_q^n$$

→ use a trapdoor function to find a short vector z such that:

$$A \cdot z \equiv H(m) \bmod q$$

→ The signature is the short vector z

Step-3

→ The verifier checks if:

$$A \cdot z \equiv H(m) \bmod q$$

→ If z is a short vector, the signature is valid.