Times asked: 6 times
5 times
4 times
3 times
2 times
1 time

# indicates 5-mark question

CSS Question Bank

1.Introduction - Number Theory and Basic Cryptography

1. Give examples of replay attacks. List three general approaches for dealing with replay attack. #
2. Encrypt given string using Playfair cipher.
   Previously asked strings:
   i. "ALL THE BEST" using "DOCUMENT" .
   ii. "The key is hidden under the door" using "domestic".
3. Define non-repudiation and authentication. Show with example how it can be achieved. #
4. State the rules for finding Euler's phi function. Calculate:
   a. ϕ(11)
   b. ϕ(49)
   c. ϕ(240)
   d. ϕ(10)
   e. ϕ(343)
5. Explain the relationship between Security Services and Mechanisms in detail. #

2. Symmetric and Asymmetric key Cryptography and key Management

1. Discuss DES with reference to following points:
   i. Block size and key size
   ii. Need of expansion permutation
   iii. Role of S-box
   iv. Weak keys and semi weak keys
   v. Possible attacks on DES
2. Explain Kerberos in detail.
3. Explain Diffie Hellman key agreement algorithm. Also discuss the possible attacks on it. #
4. Numerical on Diffie Hellman key exchange algorithm.
5. Elaborate the steps of key generation using the RSA algorithm #
6. Numerical on RSA algorithm.
7. Explain the different modes of block ciphers. (ECB and CBC asked twice) #
8. Explain Advanced Encrypted Standards (AES) in detail.

3. Cryptographic Hash Functions

1. Explain properties of secure hash function. #

2. Explain secure hash algorithm on 512 bits. #
3. Differentiate between SHA-1 and MD5. #
4. What is need for message authentication? List various techniques used for message authentication. Explain any one of them.

4. Authentication Protocols & Digital Signature Schemes

1. Why are digital certificates and signatures required? What is the role of digital signature in digital certificates?
2. Discuss RSA as a digital signature algorithm.
3. Discuss various attacks on Digital signatures.

5. Network Security and Applications

1. Explain the phases of handshake protocol in SSL. #
2. Enlist the various functions of the different protocols of SSL. #
3. How does PGP achieve confidentiality and authentication in emails?
4. Explain various types of firewall.
5. Differentiate between IDS and Firewall. #
6. Explain DDOS attack and how it is launched.
7. Write a short note on ARP spoofing. #
8. Explain TCP/IP vulnerabilities layer wise. #
9. Write a short note on: Packet Sniffing. #
10. IPSEC protocol: (asked 4 times, these 3 questions asked)
    i. How is security achieved in Transport and Tunnel modes of IPSEC? Explain the role of AH and ESP.
    ii. Explain IPSEC protocol in detail. Also write applications and advantages of IPSEC.
    iii. How does ESP header guarantee confidentiality and integrity of packet payload? What is an authentication header (AH)? How does it protect against replay attack?

6. System Security

1. Explain buffer overflow attack. #

	1	2	3	4	5	6
2024 Dec	15	45	15	10	35	0
2024 May	15	45	15	15	30	5
2023 Dec	15	45	30	0	35	0
2023 May	20	35	10	15	40	5
2022 Dec	15	35	5	20	40	10
Last 5 Avg	**15**	**40**	**15**	**15**	**35**	**5**
*2022 May	15	30	15	10	25	10
Total	95	235	70	90	205	30

Asked once:

1.Introduction - Number Theory and Basic Cryptography

1. List and explain various types of attacks on encrypted message. #
2. Explain Euclidian Algorithm. #
3. Use Hill cipher to encrypt the text “short”. The key to be used is hill.
4. Explain with examples keyed and keyless transposition cipher. #

2. Symmetric and Asymmetric key Cryptography and key Management

1. Explain algorithmic modes encryption process of symmetric key. #
2. Explain DES algorithm with flowcharts.
3. Explain RC4 stream cipher. #
4. Explain Public Key Distribution in detail.
5. Explain Needham-Schroeder authentication protocol.
6. Explain man in middle attack on Diffie Hellman. Explain how to overcome the same.
7. What is PKI? List its components.
8. What is digital certificate? How does it help to validate authenticity of a user. Explain X.509 certificate format.
9. Highlight the difference between AES and DES. #

3. Cryptographic Hash Functions

1. Provide a comparison between HMAC, CBC-MAC and CMAC
2. What goals are served using a message digest? Explain using MD5.

4. Authentication Protocols & Digital Signature Schemes

1. Explain challenge response-based authentication tokens. #

5. Network Security and Applications

1. What is ICMP flood attack? Explain in detail.
2. What are the different components of IDS? List and explain different approaches of IDS.
3. Explain key rings in PGP. #

6. System Security

1. Explain worms and viruses. #
2. Write a short note on: SQL injection. #
3. List various Software Vulnerabilities. How vulnerabilities are exploited to launch an attack. #

CSS Answer Bank

multiple times asked questions highlighted
# indicates 5-mark question

1.Introduction - Number Theory and Basic Cryptography

1. Give examples of replay attacks. List three general approaches for dealing with replay attack. #

Replay Attacks – Examples:

1. **Network Login Replay:**
   - An attacker captures a legitimate user's login request (e.g., username and encrypted password or token) and re-sends it later to gain unauthorized access.
2. **Secure Shell (SSH) Replay:**
   - A previously recorded SSH session is replayed by an attacker to execute commands on a server as if they were the original user.
3. **Payment Gateway Replay:**
   - An attacker intercepts a payment transaction (like a mobile payment request) and resends it to make unauthorized purchases.

Three General Approaches to Prevent Replay Attacks:

1. **Use of Nonces:**
   - A **nonce** is a unique, random number added to each message. Servers track recent nonces and reject reused ones.
2. **Timestamping:**
   - Each message is time-stamped, and the receiver checks whether the timestamp is within an acceptable time window.
3. **Session Tokens or One-Time Passwords (OTPs):**
   - Tokens or OTPs are generated for a single session or transaction and become invalid after one use.

2. Encrypt given string using Playfair cipher. PYQs:

i. "The key is hidden under the door" using "domestic".

ii. "ALL THE BEST" using "DOCUMENT" .

i.

Plain Text: The key is hidden under the door.

Key: Domestic

Steps:

1. Pair the plain text alphabets in two.

Th ek ey is hi dd en un de rt he do or

2. If any character in the plain text is 'J' then replace it with 'I'.

Th ek ey is hi dd en un de rt he do or

3. Double letter or consecutively repeated same letters are separated by x or z.

Th ek ey is hi dxd en un de rt he dox or

4. If an odd character is left out pair it with x or z.

Th ek ey is hi dx de nu nd er th ed ox or

5. Prepare a table same as Monoalphabetic table but this table will be 5 x 5 table because 'I' & 'J' will be merge together. (Using Key i.e. Domestic)

d	o	M	e	s
t	i/j	C	a	b
f	g	H	k	l
n	p	Q	r	u
v	w	X	y	z

6. Replace the pair of characters with the intersection, if the intersection is not found follow the rule.

Rules:

- If both letters are in the same column, take the letter below each one (going back to the top if at the bottom)
- If both letters are in the same row, take the letter to the right of each one (going back to the left if at the farthest right)
- If neither of the preceding two rules are true, form a rectangle with the two letters and take the letters on the horizontal opposite corner of the rectangle.

Th ek ey is hi dx de nu nd er th ed ox or

After Applying Rules: **cf ar ae bo gc mv os pn vt ay cf so mw ep**

Therefore, Cipher Text is: **cf ar ae bo gc mv os pn vt ay cf so mw ep**

3. Define non-repudiation and authentication. Show with example how it can be achieved. #

Non-Repudiation:

- Non-repudiation ensures that a sender cannot deny having sent a message and a receiver cannot deny having received it.
- It provides proof of origin and integrity of the data.

Authentication:

- Authentication is the process of verifying the identity of a user or system.
- It ensures that the entity involved is who it claims to be.

Example scenario: Sending a secure email using digital signatures

1. Non-Repudiation using Digital Signature:

- Alice sends a message to Bob.
- She signs the message using her **private key** (creating a digital signature).
- Bob verifies the signature using Alice's **public key**.

Result:

- If the signature is valid, Bob is sure that Alice sent it.
- Alice cannot deny having sent the message, because only her private key could have created that signature.

2. Authentication using Public Key Infrastructure (PKI):

- Bob receives the signed message from Alice.
- He uses **Alice's public key certificate** (issued by a trusted Certificate Authority, CA) to verify her identity.

Result:

- The certificate authenticates that the public key belongs to Alice.
- Bob **confirms the identity** of the sender.

4. State the rules for finding Euler's phi function. Calculate:
   a. ϕ(11)
   b. ϕ(49)
   c. ϕ(240)
   d. ϕ(10)
   e. ϕ(343)

Euler's Phi (ϕ) Function:

Euler's phi function ϕ(n) counts the number of positive integers less than n that are coprime to n (i.e., numbers that share no common factor with n other than 1).

Rules for Euler's ϕ Function:

1. **If n is a prime number:**

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

2. **If n = p^n (a power of a prime):**

$$\phi(p^k) = p^k - p^{k-1} = p^k \left(1 - \frac{1}{p}\right)$$

3. **If n = p × q × r... (product of distinct primes):**

$$\phi(n) = n \left(1 - \frac{1}{p}\right) \left(1 - \frac{1}{q}\right) \ldots$$

4. **For any positive integer n with prime factorization:**

$$\phi(n) = n \prod_{p|n} \left(1 - \frac{1}{p}\right)$$

where the product is over all **distinct** prime divisors of **n**.

---

a. φ(11)

- 11 is a **prime number**, so:

$$\phi(11) = 11 - 1 = \boxed{10}$$

b. φ(49)

- 49 = 7^2, and 7 is prime.

$$\phi(49) = 49 - 7 = \boxed{42}$$

(Using rule: φ(p^2) = p^2 - p)

c. φ(240)

- Prime factorization of 240:

$$240 = 2^4 \times 3 \times 5$$

$$\phi(240) = 240 \left(1 - \frac{1}{2}\right) \left(1 - \frac{1}{3}\right) \left(1 - \frac{1}{5}\right)$$

$$\phi(240) = 240 \times \frac{1}{2} \times \frac{2}{3} \times \frac{4}{5} = \boxed{64}$$

---

d. φ(10)

- 10 = 2 × 5

$$\phi(10) = 10 \left(1 - \frac{1}{2}\right) \left(1 - \frac{1}{5}\right)$$

$$\phi(10) = 10 \times \frac{1}{2} \times \frac{4}{5} = \boxed{4}$$

---

e. φ(343)

- 343 = 7^3

$$\phi(343) = 343 - 343/7 = 343 - 49 = \boxed{294}$$

Or:

$$\phi(343) = 343 \left(1 - \frac{1}{7}\right) = 343 \times \frac{6}{7} = \boxed{294}$$

5. Explain the relationship between Security Services and Mechanisms in detail. #

Relationship between Security Services and Mechanisms:

Security Service	*Security Mechanism*
Data confidentiality	Encipherment and routing control
Data integrity	Encipherment, digital signature, data integrity
Authentication	Encipherment, digital signature, authentication exchanges
Nonrepudiation	Digital signature, data integrity, and notarization
Access control	Access control mechanism

1. Data Confidentiality

- **Goal:** Ensure that data is only accessible to authorized users.
- **Mechanisms:**
  - **Encipherment:** Encrypts the data to make it unreadable to unauthorized parties.
  - **Routing Control:** Prevents data from taking insecure paths where it could be intercepted.

2. Data Integrity

- **Goal:** Ensure that data has not been altered in transit.
- **Mechanisms:**
  - **Encipherment:** Detects tampering by encrypting and checking for changes.
  - **Digital Signature:** Verifies sender identity and confirms data hasn't been altered.
  - **Data Integrity Mechanism:** Uses checksums or hashes to validate data accuracy.

3. Authentication

- **Goal:** Confirm the identity of communicating parties.
- **Mechanisms:**
  - **Encipherment:** Often combined with secret keys to authenticate.
  - **Digital Signature:** Ensures the sender is who they claim to be.
  - **Authentication Exchange:** Protocols that verify identity through challenges.

4. Non-repudiation

- **Goal:** Prevent sender or receiver from denying a transaction.
- **Mechanisms:**
  - **Digital Signature:** Provides legal proof of message origin.
  - **Data Integrity:** Ensures message was not changed.

- **Notarization:** Involves a third party to verify transactions.

5. Access Control

- **Goal:** Restrict access to resources only to authorized users.
- **Mechanism:**
  - **Access Control Mechanism:** Uses credentials, roles, and permissions to grant or deny access.

2. Symmetric and Asymmetric key Cryptography and key Management

6. Discuss DES with reference to following points:
   i. Block size and key size
   ii. Need of expansion permutation
   iii. Role of S-box
   iv. Weak keys and semi weak keys
   v. Possible attacks on DES

DES (Data Encryption Standard)

DES is a symmetric key encryption algorithm that processes data in fixed-size blocks using a series of transformations. Below are its key characteristics:

1. Block Size and Key Size

- **Block Size:** 64 bits (operates on 64-bit plaintext blocks).
- **Key Size:** 56 bits (although the input key is 64 bits, 8 bits are used for parity, leaving an effective key length of 56 bits).
- **Rounds:** 16 rounds of encryption.

2. Need for Expansion Permutation

- DES uses an Expansion Permutation (E-P box) to increase the 32-bit right half of the plaintext to 48 bits before applying the XOR operation with the subkey.
- **Purpose:**
  - Introduces diffusion by spreading bits across multiple S-Box inputs.
  - Helps in key mixing to make encryption more complex.

3. Role of S-Box

- The S-Box (Substitution Box) is the heart of DES.
- It takes a 6-bit input and produces a 4-bit output using predefined non-linear mappings.
- **Purpose:**
  - Provides confusion, making the relationship between the key and ciphertext complex.
  - Ensures non-linearity, which strengthens security by resisting linear cryptanalysis.

4. Weak Keys and Semi-Weak Keys

- **Weak Keys:**
  - Certain keys cause the encryption function to behave identically across rounds, making DES vulnerable. Four out of 2^{56} possible keys are called weak keys
  - Example: If all bits in the key are 0s or 1s, encryption and decryption become identical.

- **Semi-Weak Keys:**
    - There are six key pairs that are called semi-weak keys.
    - A semi-weak key creates only two different round keys and each of them is repeated eight times.

5. Possible Attacks on DES

- **Brute Force Attack:**
    - Since DES has a 56-bit key, exhaustive search of all possible keys (2^{56}) is feasible with modern computing power.
- **Linear Cryptanalysis:**
    - Uses linear approximations of DES operations to find key bits.
- **Meet-in-the-Middle Attack (on 2DES):**
    - Reduces the effective security of Double DES (2DES) from 112-bit security to about 57 bits.

7. Explain Kerberos in detail.

Kerberos is an authentication protocol, and at the same time a KDC. Three servers are involved in the Kerberos protocol:

- An authentication server (AS),
- A ticket-granting server (TGS),
- A real (data) server that provides services to others.

In our examples and figures, **Bob is the real server** and **Alice is the user requesting service**.

Authentication Server (AS):

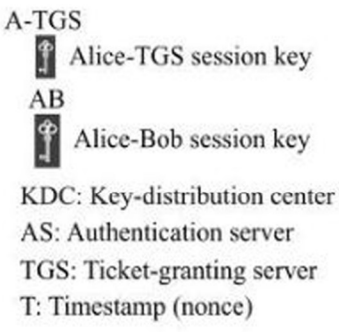
- The authentication server (AS) is the KDC in the Kerberos protocol.
- Each user registers with the AS and is granted a user identity and a password.
- The AS verifies the user, issues a session key to be used between Alice and the TGS, and sends a ticket for the TGS.

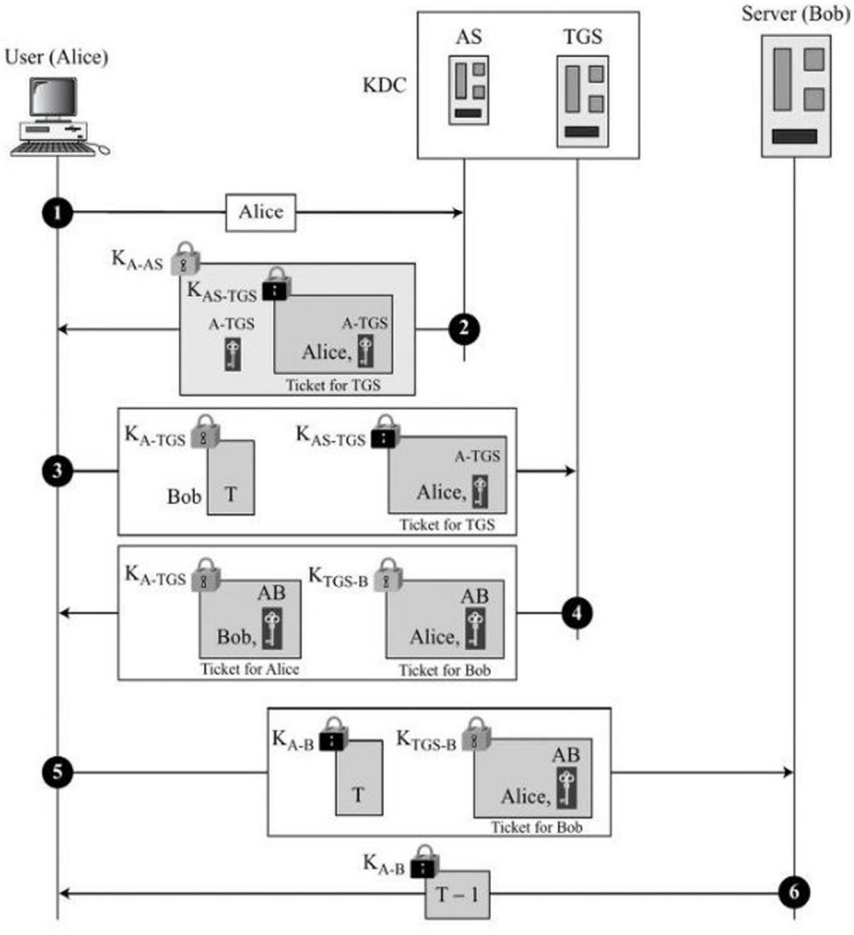
Ticket-Granting Server (TGS)

- The TGS issues a ticket for the real server (Bob).
- It also provides the session key (KAB) between Alice and Bob.
- Kerberos has separated user verification from the issuing of tickets.
- In this way, though Alice verifies her ID just once with the AS, she can contact the TGS multiple times to obtain tickets for different real servers.

Real Server:

- The real server (Bob) provides services for the user (Alice).
- Kerberos is designed for a client-server program, such as FTP, in which a user uses the client process to access the server process.
- Kerberos is not used for person-to-person authentication.





8. Explain Diffie Hellman key agreement algorithm. Also discuss the possible attacks on it. # (Asked along with numerical)

Algorithm

1. Consider a prime number q.
2. Select Primitive root 'g' and g<q.
3. Assume X_a (Private key of A) and X_a <q.

   Calculate (Public Key)$Y_a = g^{X}{}_a \bmod q$
4. Assume X_b (Private key of B) and X_b <q.

   Calculate (Public Key)$Y_b = g^{X}{}_b \bmod q$
5. Check whether key is exchanged successfully or not

   $K_a = Y_b \text{^} X_a \bmod q$

   $K_b = Y_a \text{^} X_b \bmod q$

   Now if $K_a = K_b$ then key exchange is successful.

Possible Attacks on Diffie-Hellman

1. **Man-in-the-Middle Attack (MITM):**
   - An attacker intercepts the keys exchanged and establishes two separate shared keys with each user.
   - The attacker can decrypt messages from one party, read or alter them, and send modified messages to the other.
2. **Small Subgroup Attack:**
   - Exploits poorly chosen parameters (especially g and p) to reduce the difficulty of computing discrete logs, allowing key recovery.
3. **Precomputation Attacks:**
   - For commonly used values of p and g, attackers can precompute data to speed up cracking of keys.

9. Numerical on Diffie Hellman key exchange algorithm.

Ex. 5.2.1

Solve if p = 7 and q = 17 using Diffie Hellman algorithm. Select a = 6, b = 4.

Soln. :

By using Diffie Hellman algorithm

1. Ramesh and Suresh are agree on two large prime numbers say p = 7 and q = 17.
2. Ramesh selects another secret large random number 6 i.e. a = 6 and calculate R such that

$$R = q^a \bmod p = 17^6 \bmod 7$$
$$= (17 \times 17 \times 17 \times 17 \times 17 \times 17) \bmod 7$$
$$R = 1$$

3. Ramesh sends R to Suresh.
4. Suresh selects another secret large number 4 i.e. b = 4 and calculate S such that

$$S = q^b \bmod p = 17^4 \bmod p$$
$$= (17 \times 17 \times 17 \times 17) \bmod 7$$
$$S = 4$$

5. Suresh sends number S to Ramesh
6. Ramesh now calculates it's secret key R_K as follows :

$$R_K = S^a \bmod p = S^6 \bmod p = 4^6 \bmod 7$$
$$= (4 \times 4 \times 4 \times 4 \times 4 \times 4) \bmod 7$$
$$R_K = 1$$

7. Suresh is calculating his secret S_K as follows :

$$S_K = R^b \bmod p = 1^4 \bmod 7$$
$$S_K = 1$$

8. If $R_K = S_K$ then Ramesh and Suresh can agree for future communication.

PYQs:

I] Consider the example where A and B decide to use the Diffie Hellman algorithm to share a key. They choose p=23 and g=5 as the public parameters. Their secret keys are 6 and 15 respectively. Compute the secret key that they share. (twice)

II] Apply Diffie Hellman key exchange algorithm, two users P & Q will agree on two numbers as n=11 common prime & g=7 is generator, x=3 , y=6 are private keys of P & Q respectively. What is shared secret key?

Solution for I]

🔢 Given:

- Prime number: $p = 23$
- Primitive root (base): $g = 5$
- A's secret key: $a = 6$
- B's secret key: $b = 15$

🔐 Step 1: Compute public keys

A computes:

$$A_{\text{pub}} = g^a \mod p = 5^6 \mod 23$$

$$5^6 = 15625 \quad \Rightarrow \quad 15625 \mod 23 = 8$$

So, **A sends 8** to B.

B computes:

$$B_{\text{pub}} = g^b \mod p = 5^{15} \mod 23$$

$$5^{15} = 30517578125 \quad \Rightarrow \quad 5^{15} \mod 23 = 2$$

So, **B sends 2** to A.

🔐 Step 2: Compute shared secret

Now each computes the shared secret using the other's public key.

A computes:

$$K = B_{\text{pub}}^a \mod p = 2^6 \mod 23 = 64 \mod 23 = 18$$

B computes:

$$K = A_{\text{pub}}^b \mod p = 8^{15} \mod 23 = 18$$

✅ **Shared Secret Key = 18**

Solution for II]

🔹 Given:

- Prime number $n = 11$
- Generator $g = 7$
- Private key of P: $x = 3$
- Private key of Q: $y = 6$

🔸 Step 1: Compute public keys

P computes:

$$A = g^x \mod n = 7^3 \mod 11 = 343 \mod 11 = 2$$

✅ Public key sent by P is **2**

Q computes:

$$B = g^y \mod n = 7^6 \mod 11$$

First compute 7^6:

$$7^2 = 49 \mod 11 = 5$$
$$7^4 = 5^2 = 25 \mod 11 = 3$$
$$7^6 = 7^2 \cdot 7^4 = 5 \cdot 3 = 15 \mod 11 = 4$$

✅ Public key sent by Q is **4**

🔸 Step 2: Compute the shared secret key

Now each side computes:

P computes shared key:

$$K = B^x \mod n = 4^3 \mod 11 = 64 \mod 11 = 9$$

Q computes shared key:

$$K = A^y \mod n = 2^6 \mod 11 = 64 \mod 11 = 9$$

✅ Final Answer:

The **shared secret key** is **9**.

10. Elaborate the steps of key generation using the RSA algorithm (asked along with numerical both times) #

RSA algorithm:

Key Generation:

1. Choose two large prime numbers (p and q)
2. Calculate n = p*q and Φ = (p-1)(q-1)
3. Choose a number e where 1 < e < Φ and it is co-prime to Φ
4. Calculate d = e^{-1} mod(p-1)(q-1) Or e*d =1 mod Φ
5. Bundle private key pair as (n,d)
6. Bundle public key pair as (n,e)

Encryption:

If the plaintext is m, ciphertext = m^e mod n.

Decryption:

If the ciphertext is c, plaintext = c^d mod n

11. Numerical on RSA algorithm.

Example 1:

1. Select two prime numbers a = 13, b = 11.
2. $n = a*b = 13 * 11 = 143$.
3. $\phi(n) = (13 - 1) * (11 - 1) = 12 * 10 = 120$.
4. Select e = 13, gcd (13, 120) = 1.
5. **Finding d :**

   $e*d \bmod \phi(n) = 1$

   $13 * d \bmod 120 = 1$

   Do the following procedure till you are not getting a integer numbers

$$d = \frac{(\phi(n) * i) + 1}{e}$$

$$d = \frac{(120 + 1)}{13} = \frac{121}{13} = 9.30 \ (i = 1) \qquad \text{where, } i = 1 \text{ to } 9$$

$$d = \frac{240 + 1}{13} = \frac{241}{13} = 18.53 \ (i = 2)$$

$$d = \frac{360 + 1}{13} = \frac{361}{13} = 27.76 \ (i = 3)$$

$$d = \frac{480 + 1}{13} = \frac{481}{13} = 37$$

   Hence **d = 37**

6. Hence public key = {13, 143} and Private key = {37, 143}
7. Encryption :

   Consider any integer as a plaintext (P)

   Such that P < n

   Example : 13 $\quad \because (13 < 143)$

Now, $C = P^e \bmod n$

$C = 13^{13} \bmod 143$

Here to find out $13^{13} \bmod 143$, use the following procedure

$13 \bmod 143 = 13$

$13^2 \bmod 143 = 169 \bmod 143 = 26$

$13^4 \bmod 143 = 26^2 \bmod 143 = 104$

$13^8 \bmod 143 = 104^2 \bmod 143 = 91$

$\therefore C = [(13^8 \bmod 143) * (13^4 \bmod 143) * (13 \bmod 143)] \bmod 143$

$= [91 * 104 * 13] \bmod 143$

$= 52$

8. **Decryption :**

$P = C^d \bmod n$

$= 52^{37} \bmod 143$

Again use above mentioned procedure to find out $52^{37} \bmod 143$. As

$52 \bmod 143 = 52$

$52^2 \bmod 143 = 130$

$52^4 \bmod 143 = (130)^2 \bmod 143 = 26$

$52^8 \bmod 143 = (26)^2 \bmod 143 = 104$

$52^{16} \bmod 143 = (104)^2 \bmod 143 = 91$

$52^{32} \bmod 143 = (91)^2 \bmod 143 = 130$

Hence,

$P = 52^{37} \bmod 143$

$= [(52^{32} \bmod 143) * (52^4 \bmod 143) * (52 \bmod 143)] \bmod 143$

$= [130 * 26 * 52] \bmod 143$

$= 13$

PYQs:

I] Elaborate the steps of key generation using the RSA algorithm. In RSA system the public key (E, N) of user A is defined as (7,187). Calculate Φ(N) and private key 'D'. What is the cipher text for M=10 using the public key. (twice)

II] Use RSA algorithm, user A has public key (17,321), B has public key (5,321). Calculate private keys of both the users. Encrypt m=7 by B's public keys. How B can decrypt the same.

Solution for I]

Given:

- Public Key $(E, N) = (7, 187)$
- Message $M = 10$

Step 1: Factor $N = 187$

We need to find the two prime numbers p and q such that:

$$N = p \times q$$

Try factoring 187:

$$187 = 11 \times 17$$

So, $p = 11, q = 17$

Step 2: Calculate Euler's Totient Function $\phi(N)$

$$\phi(N) = (p-1)(q-1) = (11-1)(17-1) = 10 \times 16 = 160$$

Step 3: Find Private Key D

To find D, use the formula:

$$D = \frac{(\phi(N) \cdot i) + 1}{E}$$

Try successive values of i until the result is an integer:

- For $i = 1$:

$$D = \frac{160 \cdot 1 + 1}{7} = \frac{161}{7} = 23.00 \Rightarrow \checkmark \text{ Integer}$$

Hence,

$$\boxed{D = 23}$$

Final Key Pair for User A:

- Public Key (E, N): $(7, 187)$
- Private Key (D, N): $(23, 187)$

Step 4: Encryption — Find Cipher Text for $M = 10$

Use the encryption formula:

$$C = M^E \mod N = 10^7 \mod 187$$

Using modular exponentiation:

- $10^2 = 100$
- $10^4 = 100^2 = 10000 \mod 187 = 82$
- $10^7 = 10^4 \cdot 10^2 \cdot 10 = 82 \cdot 100 \cdot 10 \mod 187$

Step-by-step:

- $82 \cdot 100 = 8200 \mod 187 = 168$
- $168 \cdot 10 = 1680 \mod 187 = 179$

$$\boxed{\text{Cipher Text } C = 179}$$

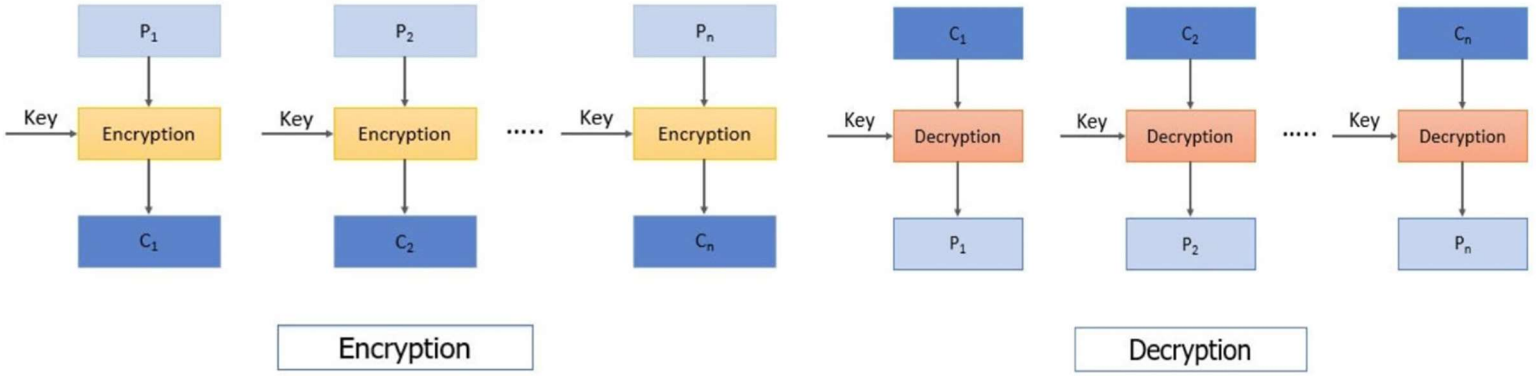
Final Answers:

- ϕ(N) = 160
- Private key D = 23
- Cipher Text C = 179

12. Explain the different modes of block ciphers. (ECB and CBC asked separately twice) #

1. Electronic Codebook (ECB) mode:

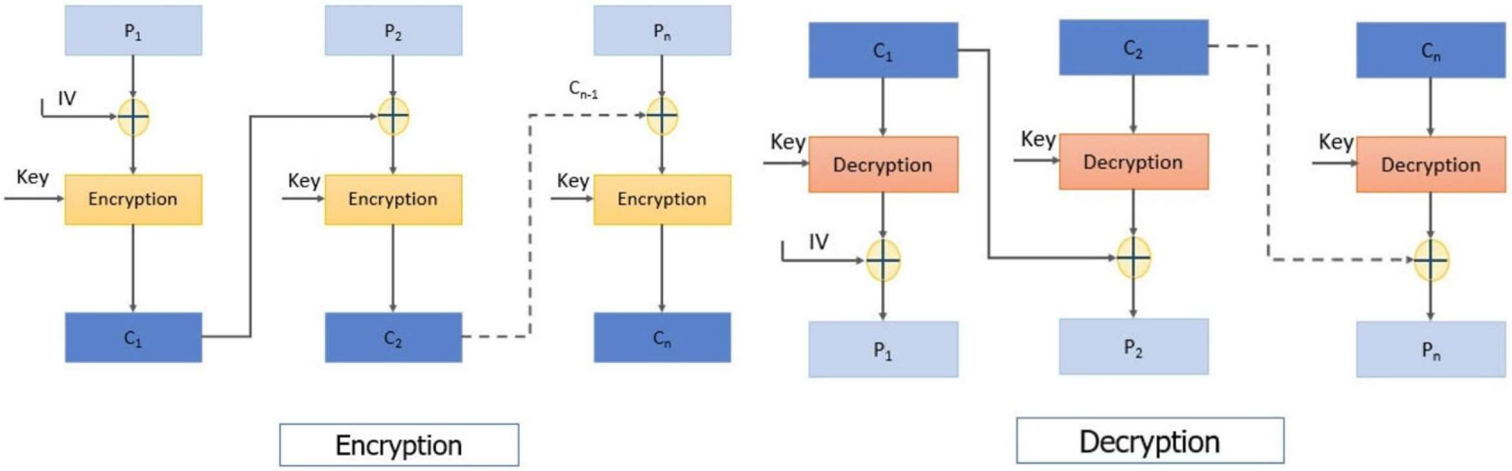
- Plaintext is divided into **64-bit blocks**.
- Each block is **independently encrypted** using the **same key**.
- Ciphertext is also divided into **64-bit blocks**, decrypted one at a time with the **same key** to recover plaintext.



2. Cipher Block Chaining (CBC) Mode

Encryption:

1. IV is XORed with the first plaintext block, then encrypted to get the first ciphertext block.
2. The first ciphertext block is fed to the encryption of the second plain text block. Similarly, each ciphertext block is XORed with the next plaintext block before encryption, continuing the chain.



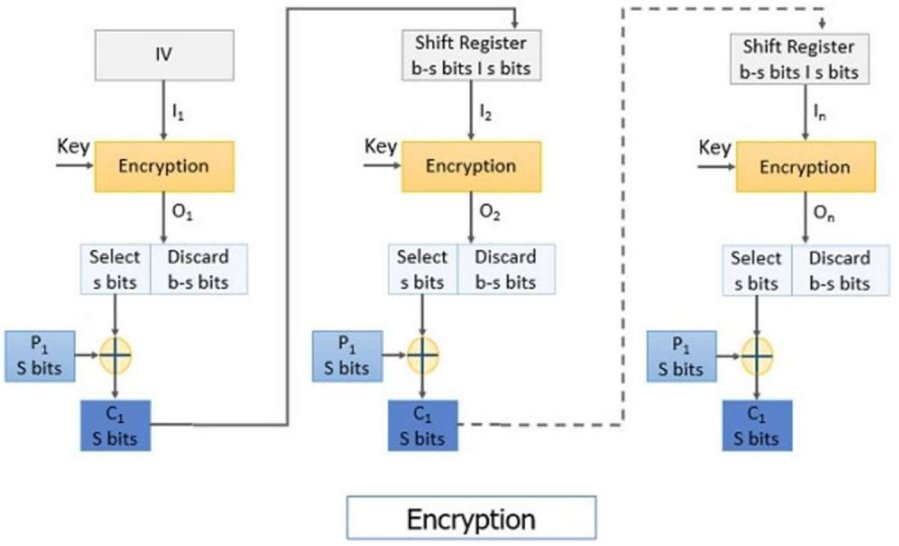
Decryption:

1. The first ciphertext block is decrypted using the same key that was used for encrypting all plain text blocks. The result of decryption is then XORed with the initialization vector (IV) to obtain the first plain text block.
2. Each ciphertext block is decrypted, then XORed with the previous ciphertext to recover plaintext. The process repeats for all blocks.

3. Cipher Feedback (CFB) Mode

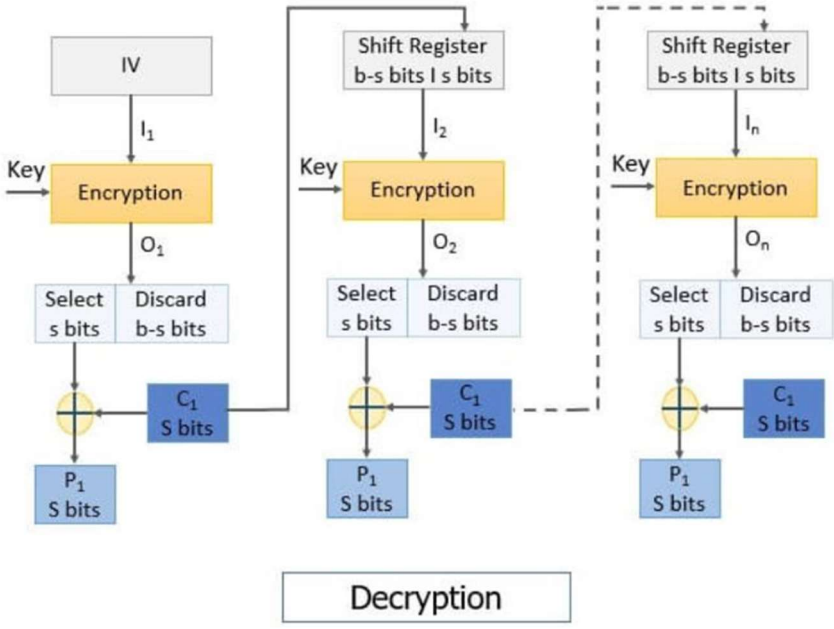
Encryption:

1. IV is encrypted using the key.
2. Leftmost s bits of encrypted IV are XORed with the first s-bit plaintext fragment to get ciphertext C1.
3. C1 is fed back into the IV, shifting left by s bits, and the process repeats for the next plaintext fragment.



Decryption:

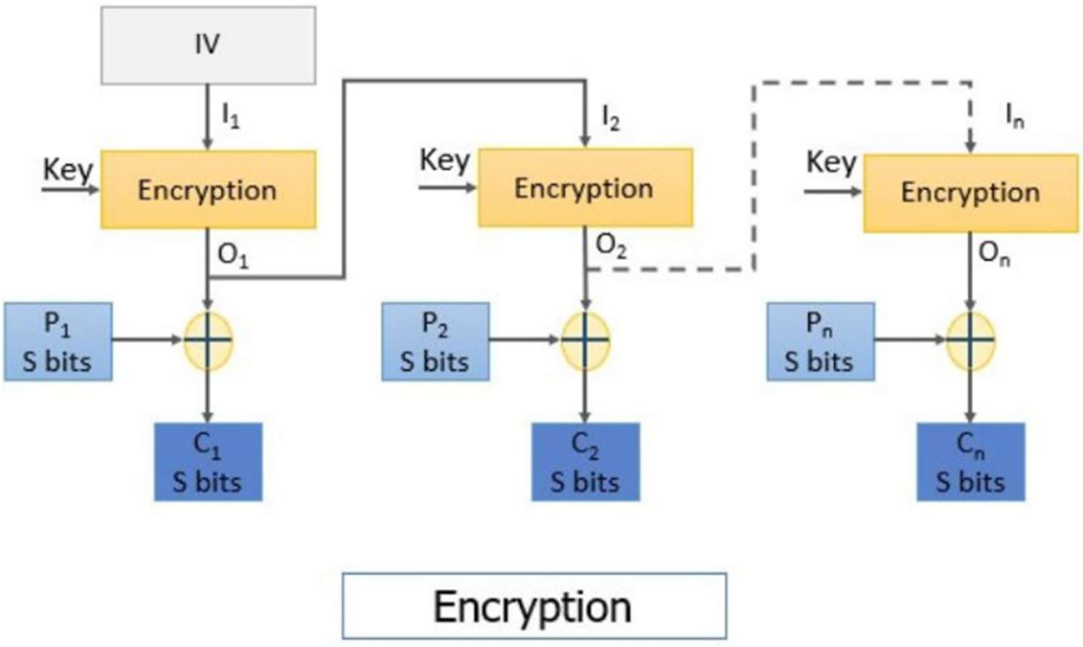
1. IV is encrypted using the same key (encryption is used for decryption).
2. Leftmost s bits of encrypted IV are XORed with C1 to retrieve plaintext P1.
3. C1 is fed back into the IV, and the process continues for all ciphertext fragments.



4. Output Feedback (OFB) Mode

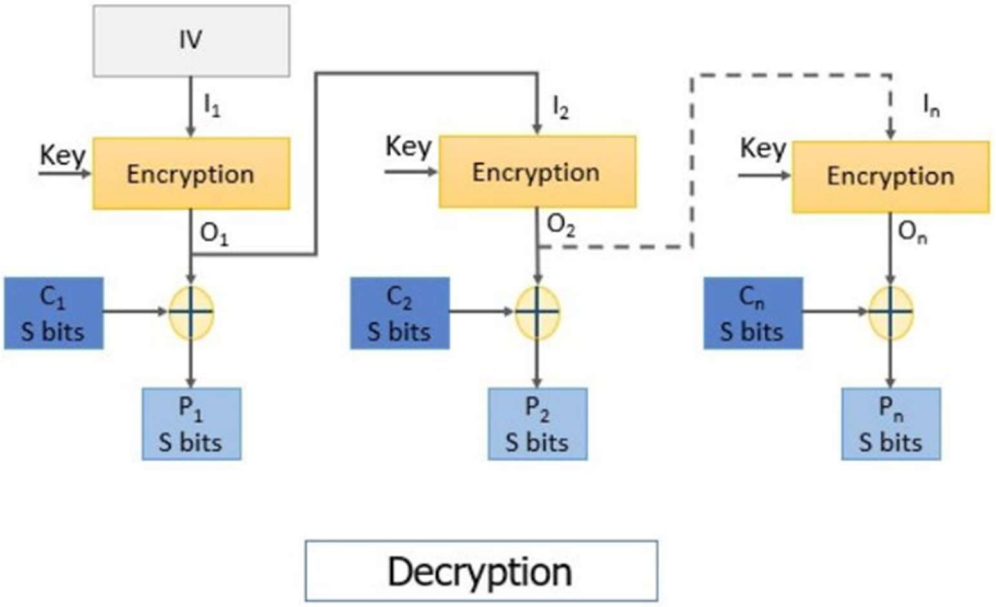
Encryption:

1. IV is encrypted using the key.
2. The encrypted IV is XORed with the plaintext block to produce ciphertext.



Decryption:

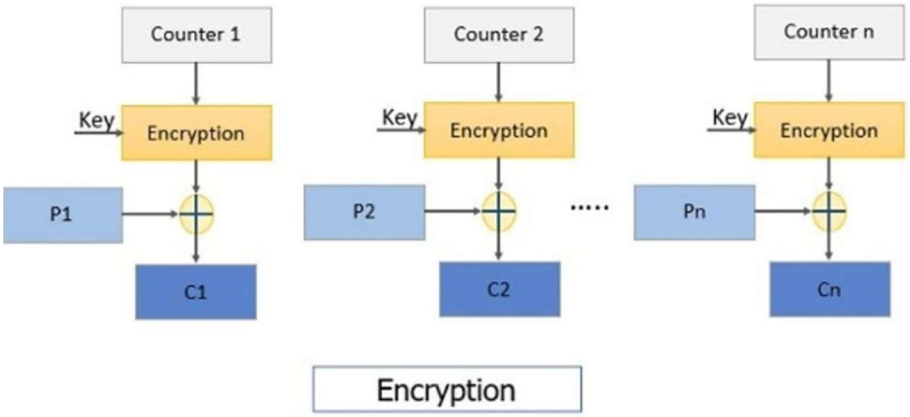
1. IV is encrypted using the same key.
2. The encrypted IV is XORed with the ciphertext to retrieve plaintext.
3. The encrypted IV is used for the next block.



5. Counter Mode

Encryption:

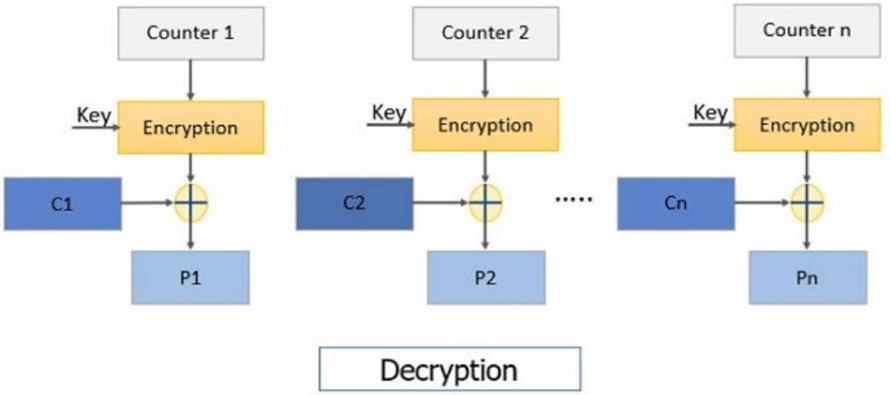
1. The Counter value is encrypted using the key.
2. The encrypted counter is XORed with plaintext to get ciphertext.
3. The counter increments for each block, and the steps repeat.



Decryption:

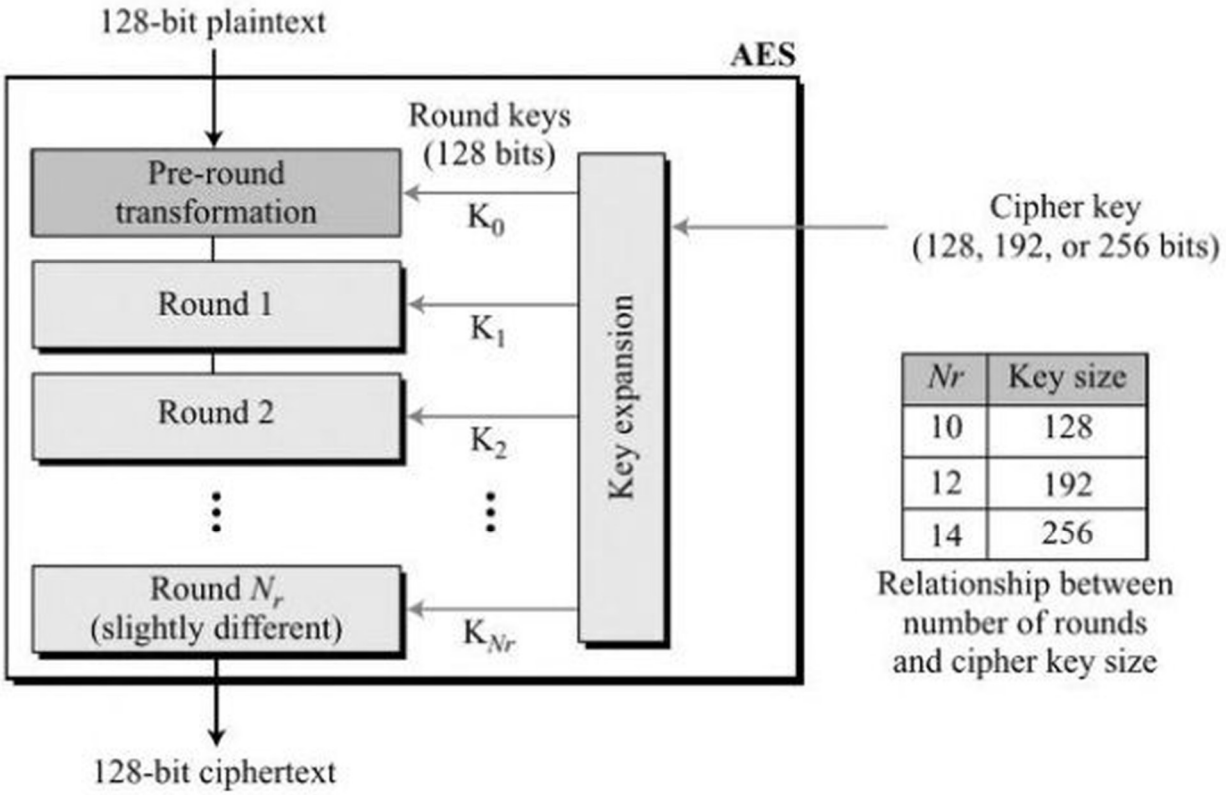
1. The Counter value is encrypted using the same key.
2. The encrypted counter is XORed with ciphertext to retrieve plaintext.

The counter increments, and steps repeat until all blocks are



13. Explain Advanced Encrypted Standards (AES) in detail.

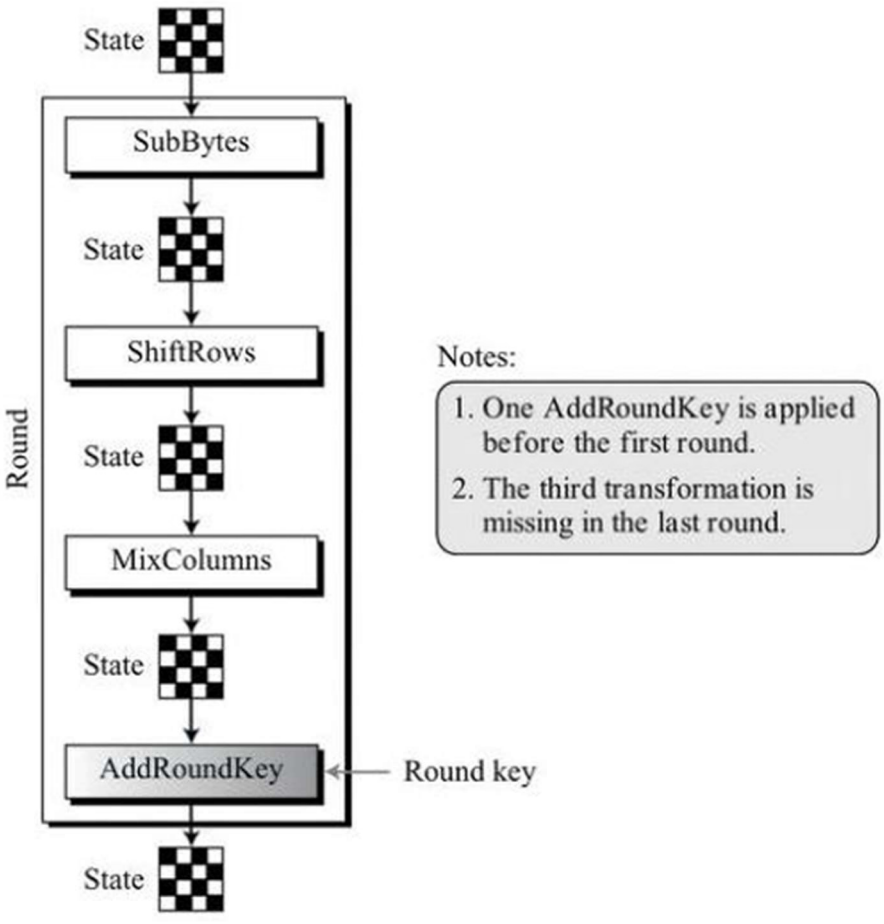
- It requires a block size of 128 bits and three different key sizes of 128, 192, and 256 bits.
- AES is a non-Feistel cipher that encrypts and decrypts a data block of 128 bits.
- It uses 10, 12, or 14 rounds.
- The key size, which can be 128, 192, or 256 bits, depends on the number of rounds.
- The number of round keys generated by the key-expansion algorithm is always one more than the number of rounds.



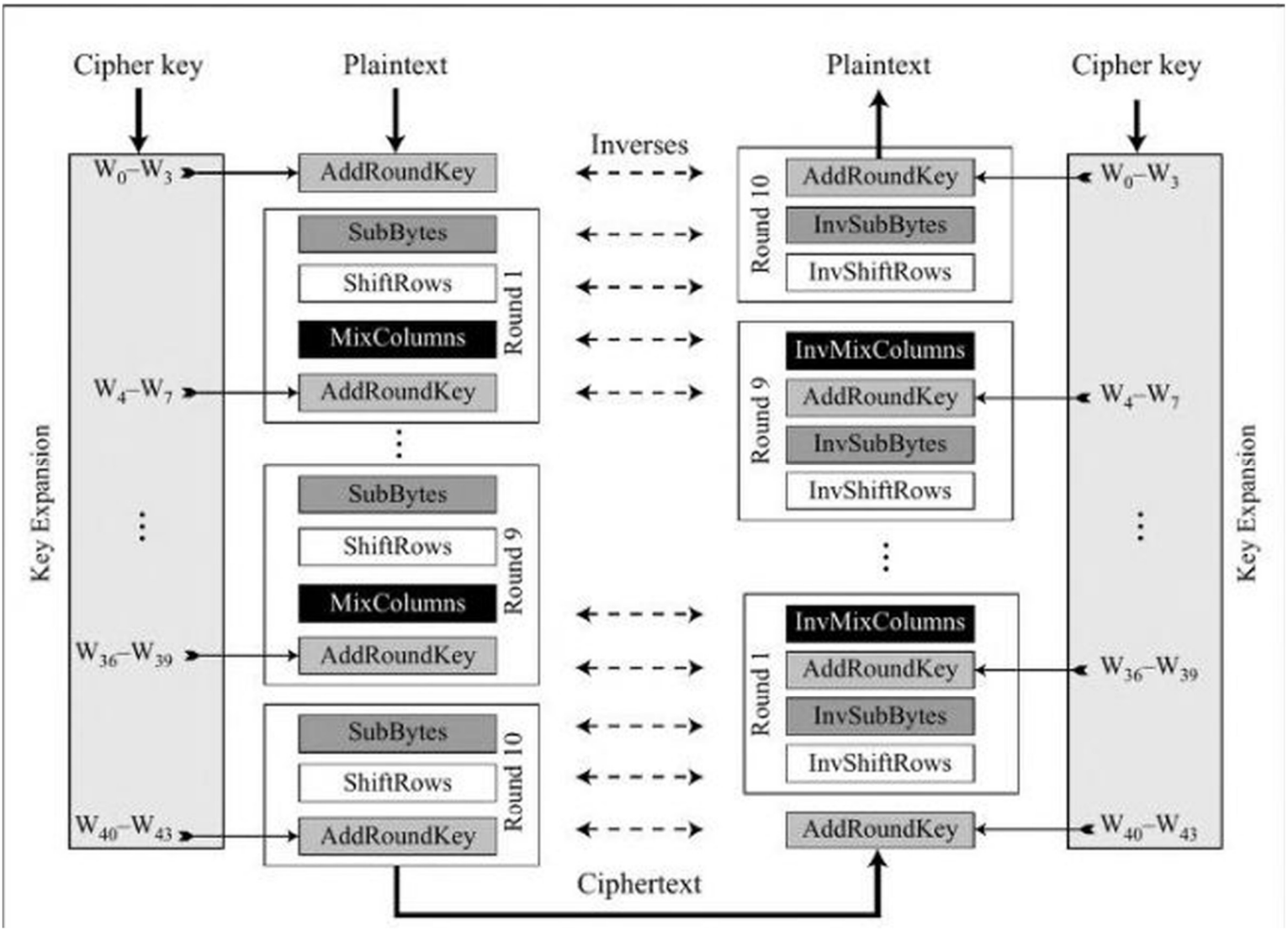
Nr	Key size
10	128
12	192
14	256

Relationship between number of rounds and cipher key size

Structure of Each Round:



Encryption and Decryption process in AES:



Encryption Process

1. **Key Expansion**:
    - The input cipher key (128 bits) is expanded into multiple round keys using a key schedule.
    - Keys W_0 to W_{43} are derived for 11 rounds (0–10).
2. **Initial Round**:
    - **AddRoundKey**: XOR the plaintext with the first-round key.
3. **Rounds 1 to 9** (Each round consists of):

- **SubBytes (Substitution):**
  Replaces each byte using an S-Box to add confusion and make patterns harder to detect.
- **ShiftRows (Permutation):**
  Shifts rows of the matrix to the left to mix data and spread byte influence.
- **MixColumns (Mixing):**
  Mixes data within each column using math in a special field (GF 2^8) to increase security.
- **AddRoundKey (Key Addition):**
  XORs the state with a round-specific key, making encryption depend on the key.

4. **Final Round (Round 10)**:
    - Same as other rounds but no MixColumns.
    - Operations: SubBytes → ShiftRows → AddRoundKey.

Decryption Process

1. **Key Expansion**:
    - Uses the same expanded keys in reverse order.
2. **Initial Round**:
    - **AddRoundKey** using the last round key W_{40} to W_{43}.
3. **Rounds 1 to 9**:
    - **InvShiftRows**: Reverse row shifting.
    - **InvSubBytes**: Reverse byte substitution.
    - **AddRoundKey:** XOR the ciphertext with the round key
    - **InvMixColumns**: Reverse column mixing.
4. **Final Round**:
    - Only InvShiftRows, InvSubBytes, and AddRoundKey.

Advantages of AES

1. **Strong Security**
   AES uses key sizes of 128, 192, or 256 bits, making brute-force attacks infeasible.
2. **Fast and Efficient**
   AES is designed to be fast in both hardware and software implementations.

3. Cryptographic Hash Functions

14. Explain properties of secure hash function. #

A cryptographic hash function is a special type of hash function used in security applications. It takes an input (or message) and returns a fixed-size string of bytes, typically a digest that appears random. These functions are widely used in data integrity checks, digital signatures, password storage, and more.

Properties of a Secure Hash Function

1. **Deterministic**
   - The same input always produces the same hash output.
2. **Pre-Image Resistance** (One-way property)
   - Given a hash value *h*, it should be computationally infeasible to find any input *x* such that hash(x) = h.
3. **Second Pre-Image Resistance**
   - Given an input x_1, it should be infeasible to find another input x_2 such that hash(x_1) = hash(x_2).
4. **Collision Resistance**
   - It should be hard to find *any* two distinct inputs x_1 and x_2 such that hash(x_1) = hash(x_2).
5. **Avalanche Effect**
   - A small change in input should drastically change the output hash (bitwise), making patterns hard to detect.
6. **Fast Computation**
   - The function should quickly compute the hash for any input.
7. **Fixed Output Length**
   - Regardless of input size, the output should be a fixed length (e.g., 128, 256, or 512 bits).

15. Explain secure hash algorithm on 512 bits. #

SHA-512 is part of the SHA-2 family, designed by the NSA and standardized by NIST.

Key Features:

- **Output Length**: 512 bits (64 bytes)
- **Block Size**: 1024 bits
- **Word Size**: 64 bits
- **Number of Rounds**: 80

Working Steps of SHA-512:

1. **Padding the Message**
    - The message is padded to make its length a multiple of 1024 bits (a block size). Padding includes a 1 bit followed by 0s and the original message length.
2. **Parsing the Message**
    - The padded message is divided into 1024-bit blocks.
3. **Initialize Hash Values**
    - Eight 64-bit words are used as initial hash values (constants defined in the standard).
4. **Message Schedule Preparation**
    - For each block, 80 64-bit words are prepared using the original message block and bitwise operations.
5. **Compression Function**
    - Each message block goes through 80 rounds of processing using logical functions (e.g., AND, XOR), modular additions, and shifts.
6. **Update Hash Values**
    - Intermediate hash values are updated in each round and used for the next block.
7. **Final Output**
    - After processing all blocks, the final 512-bit hash value is produced.

16. Differentiate between SHA-1 and MD5. #

Points	MD-5	SHA
Message Digest Length	128 Bits.	160 Bits.
Security	Less Secure than SHA.	Considered more secure than MD-5.
Speed	Faster, only 64 Iterations.	Slower than MD-5, Required 80 Iterations.
Format	Little endian format used to store values.	Big endian format used to store values.
Buffers Used	4 buffers of 32 bits each.	5 buffers of 32 bits each.
Attack required to find out original message	2^{128} bit operations required to break.	2^{160} bit operations required to break.
Collision	Collision attack exist	Collision ratio is less than MD-5
Pass	It requires 4 passes.	It requires 5 passes.
Rounds	64 Rounds.	20 Rounds.
Cryptanalytic Attack	Vulnerable to cryptanalysis attack.	Non-Vulnerable to cryptanalysis attack.

17. What is need for message authentication? List various techniques used for message authentication. Explain any one of them.

Need for message authentication:

Message authentication is crucial in communication to ensure that a message:

1. Comes from a legitimate source (authentication)
2. Has not been altered in transit (integrity)
3. Has not been duplicated or replayed

Without message authentication:

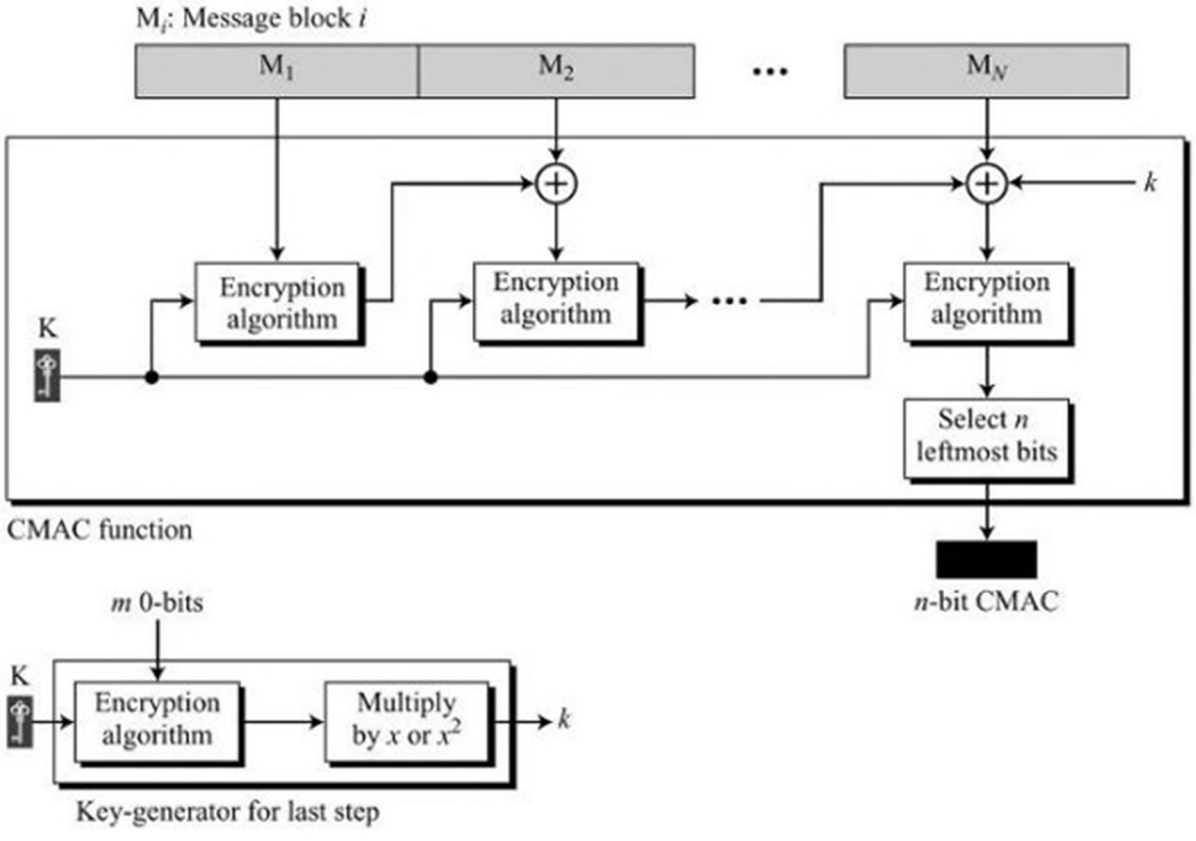
- Attackers can modify messages without detection (tampering).
- Impersonation becomes possible (spoofing).
- Messages can be re-sent maliciously (replay attacks).

Message authentication techniques:

- Message Authentication Code (MAC)
- Hash-based Message Authentication Code (HMAC)
- Cipher-based Message Authentication Code (CMAC)
- Cipher Block Chaining MAC (CBC-MAC)
- Digital Signatures

Technique for message authentication(can do any of the above):

CMAC (Cipher-based Message Authentication Code)



1. The message is divided into N blocks, each m bits long.
2. The size of the CMAC is n bits. If the last block is not m bits, it is padded with a 1-bit followed by enough 0-bits to make it m bits.
3. The first block of the message is encrypted with the symmetric key to create an m-bit block of encrypted data.
4. This block is XORed with the next block and the result is encrypted again to create a new m-bit block.
5. The process continues until the last block of the message is encrypted.
6. The n leftmost bit from the last block is the CMAC.
7. In addition to the symmetric key K, CMAC also uses another key k, which is applied only at the last step. This key is derived from the encryption algorithm with plaintext of m 0-bits using the cipher key K.
8. The result is then multiplied by x if no padding is applied and multiplied by x^2 if padding is applied.

4. Authentication Protocols & Digital Signature Schemes

18. Why are digital certificates and signatures required? What is the role of digital signature in digital certificates?

Need for Digital Certificates:

1. **Public Key Verification**:
   They verify that a public key truly belongs to the individual or organization claiming it.
2. **Trusted Communication**:
   Issued by a Certificate Authority (CA), they help establish trust between parties who have never met before.
3. **Prevent Man-in-the-Middle Attacks**:
   Without certificates, attackers could pose as someone else by sending a fake public key.

Need for Digital Signatures:

1. **Authentication**:
   Confirms the identity of the sender — the signature proves the message came from the private key holder.
2. **Data Integrity**:
   Ensures the message was not altered in transit. Even a small change would make the signature invalid.
3. **Non-Repudiation**:
   The sender cannot deny sending the message since only they possess the private key used to sign it.

Role of digital signature in digital certificates:

The digital signature plays a crucial role in the functioning and trustworthiness of a digital certificate. A digital certificate is an electronic document that binds a public key with the identity of its owner. However, for others to trust that this certificate truly belongs to the claimed entity, it must be verified and vouched for by a trusted third party—this is where the digital signature comes in.

When a Certificate Authority (CA) issues a digital certificate, it signs the certificate using its private key. This digital signature acts as proof that the certificate was indeed issued by the CA and that the information contained in it—such as the public key, the identity of the owner, and the validity period—has not been tampered with. Anyone receiving the certificate can verify the digital signature using the CA's public key. If the verification succeeds, the recipient knows that the certificate is authentic and trustworthy.

Without the digital signature, there would be no reliable way to trust that the certificate is legitimate or has not been altered.

19. Discuss RSA as a digital signature algorithm.

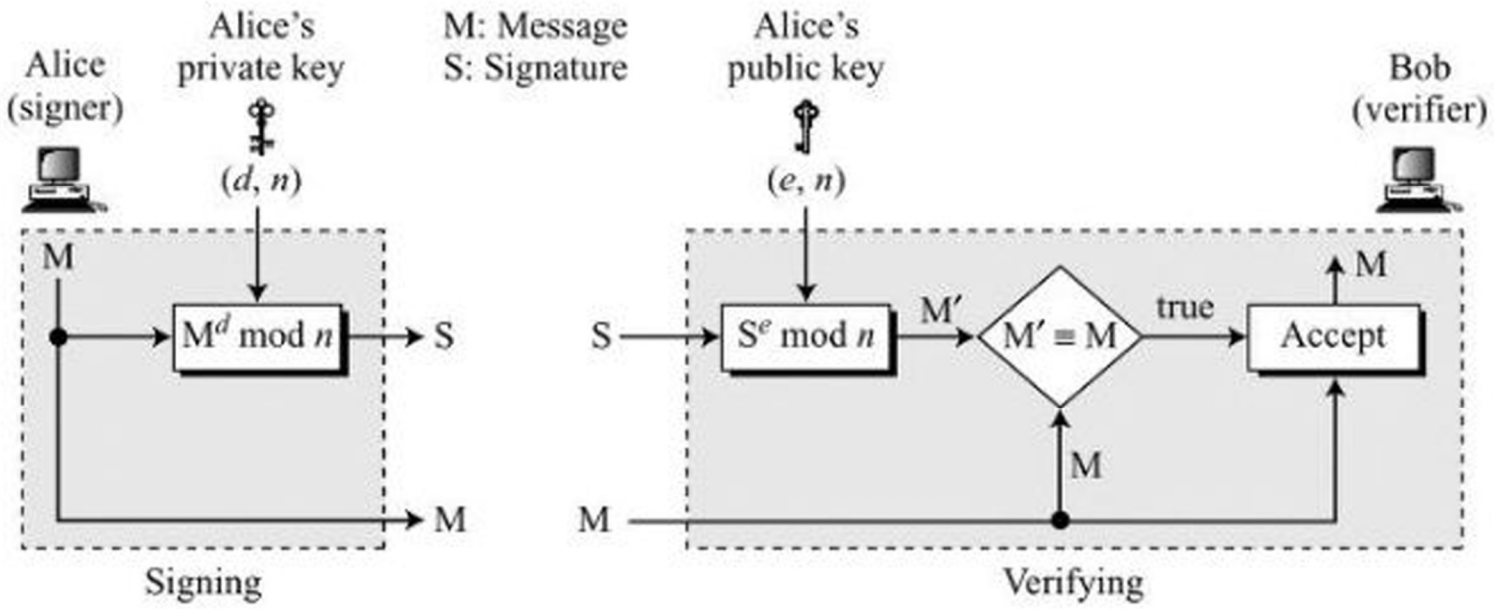
- The RSA idea can also be used for signing and verifying a message.
- The digital signature scheme changes the roles of the private and public keys.
- First, the private and public keys of the sender, not the receiver, are used.
- Second, the sender uses her own private key to sign the document; the receiver uses the sender's public key to verify it.
- The signing and verifying sites use the same function, but with different parameters.
- The verifier compares the message and the output of the function for congruence. If the result is true, the message is accepted.

Key Generation

Key generation in the RSA digital signature scheme is exactly the same as key generation in the RSA cryptosystem.

1. Choose two large prime numbers (p and q)
2. Calculate n = p*q and Φ = (p-1)(q-1)
3. Choose a number e where 1 < e < Φ and it is co-prime to Φ
4. Calculate d = e^{-1} mod(p-1)(q-1) Or e*d =1 mod Φ
5. Bundle private key pair as (n,d)
6. Bundle public key pair as (n,e)

Signing and Verifying



For signing, Alice creates a signature out of the message using her private exponent, S = M^d mod n and sends the message and the signature to Bob.

For verifying, Bob receives M and S. Bob applies Alice's public exponent to the signature to create a copy of the message M' = S^e mod n. Bob compares the value of M' with the value of M. If the two values are congruent, Bob accepts the message.

20. Discuss various attacks on Digital signatures.

There are 3 kinds of attacks on digital signatures:

- Key-only
- Known-message
- Chosen-message.

• Key-Only Attack

In the key-only attack, Eve has access only to the public information released by Alice. To forge a message, Eve needs to create Alice's signature to convince Bob that the message is coming from Alice.

• Known-Message Attack

In the known-message attack, Eve has access to one or more message-signature pairs. In other words, she has access to some documents previously signed by Alice. Eve tries to create another message and forge Alice's signature on it.

• Chosen-Message Attack

In the chosen-message attack, Eve somehow makes Alice sign one or more messages for her. Eve now has a chosen-message/signature pair. Eve later creates another message, with the content she wants, and forges Alice's signature on it.

Forgery Types

If the attack is successful, the result is a forgery. There can be two types of forgery: existential and selective.

• Existential Forgery

Eve creates a valid message-signature pair, but the message content is random or useless. Although this forgery is more likely, Eve can't gain much from it since the message has no real value.

• Selective Forgery

Eve is able to successfully forge Alice's signature on a message that she has chosen. The content of this message is specific and meaningful to Eve. This type of forgery can be very harmful to Alice because the forged message may contain important or sensitive information. Although it's less likely to occur, it's more dangerous if successful.

5. Network Security and Applications

21. Explain the phases of handshake protocol in SSL. #

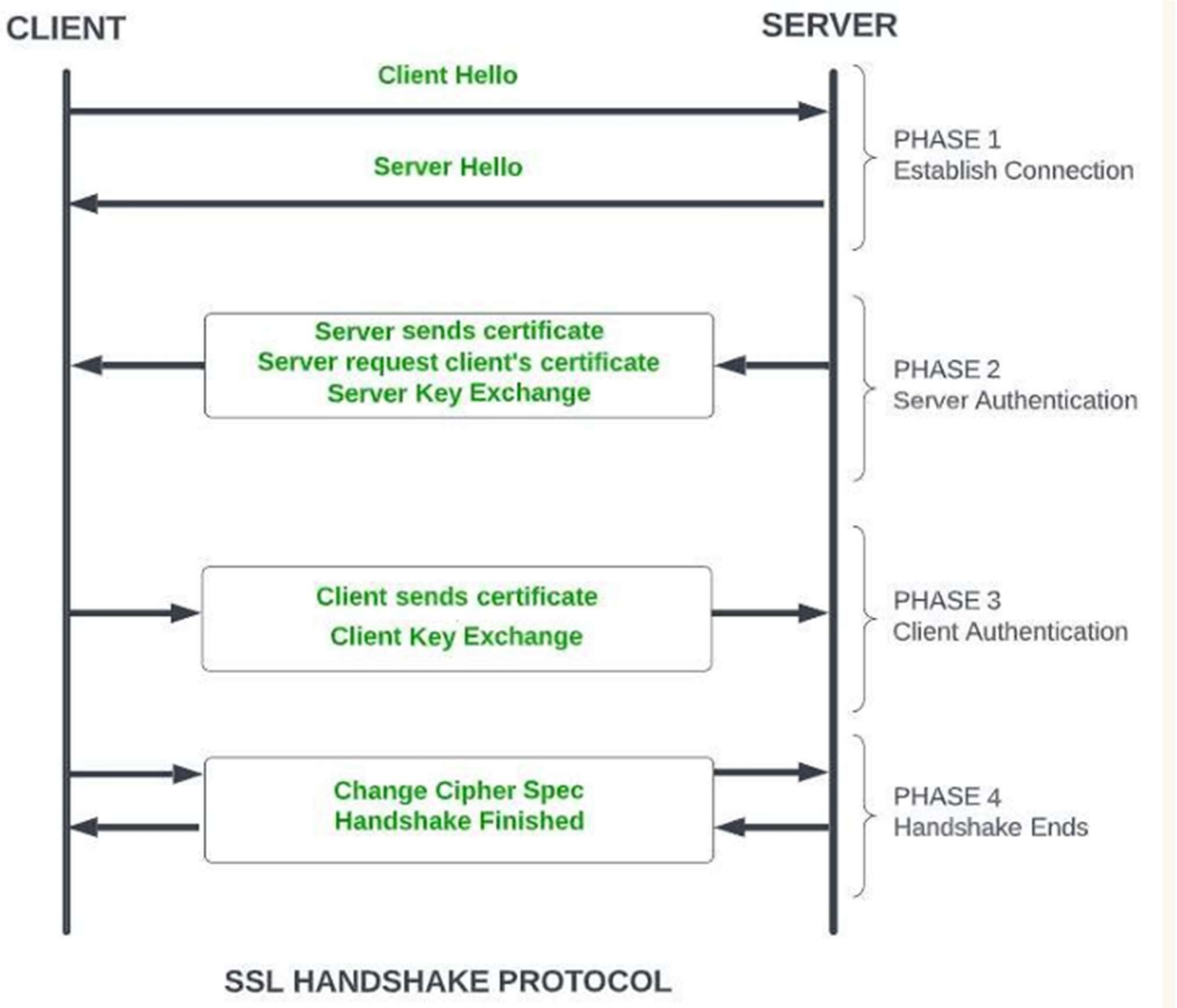
Handshake Protocol is used to establish sessions. This protocol allows the client and server to authenticate each other by sending a series of messages to each other. Handshake protocol uses four phases to complete its cycle.

Phase 1: Both the Client and Server send hello packets to each other. In this IP session, cipher suite and protocol version are exchanged for security purposes.

Phase 2: Server sends its certificate and Server-key-exchange message. The server ends phase-2 by sending the Server-hello-end packet.

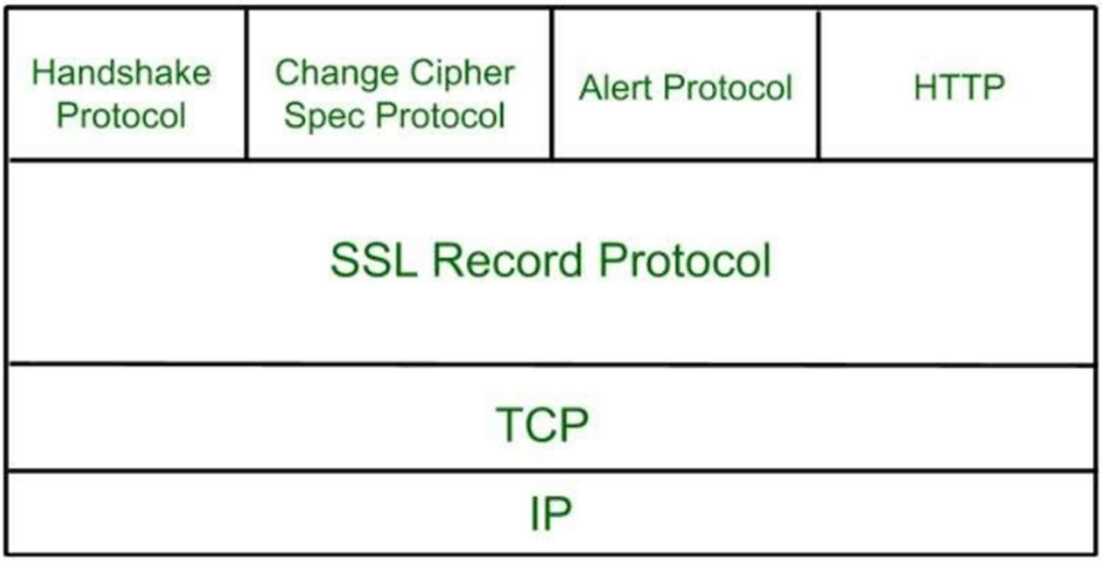
Phase 3: Client replies to the server by sending his certificate and Client-exchange-key.

Phase 4: Change-cipher suite occurs and after this the Handshake Protocol ends.



SSL HANDSHAKE PROTOCOL

22. Enlist the various functions of the different protocols of SSL. #



Functions of SSL Protocols

1. Handshake Protocol

- Negotiates cipher suite, SSL/TLS version, and session parameters.
- Authenticates client and server using digital certificates.

2. Change Cipher Spec Protocol

- Signals both parties to switch to the negotiated cipher suite for encryption.
- Indicates the end of key exchange and start of secure communication.

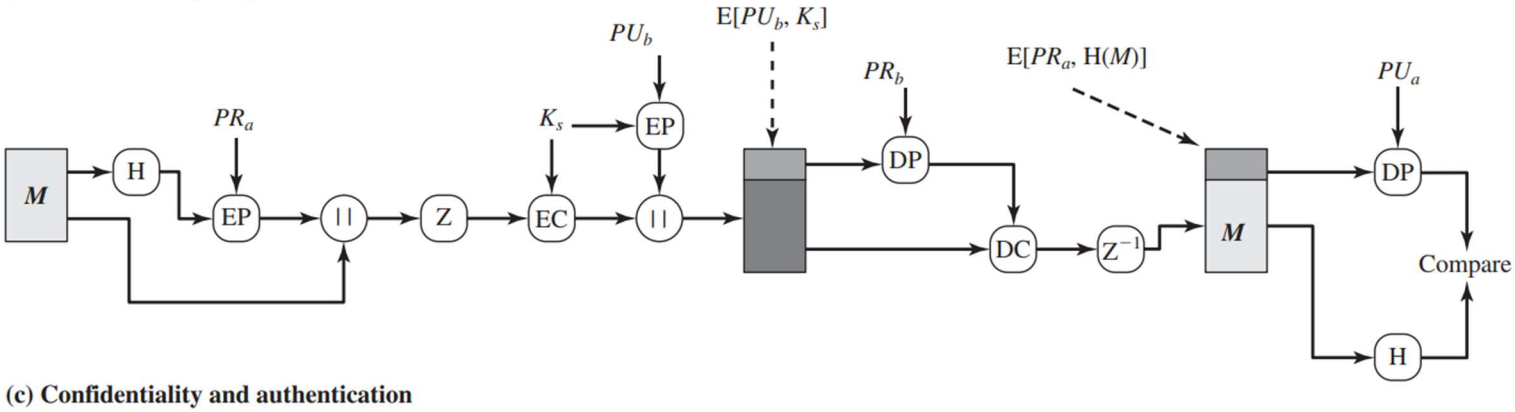
3. Alert Protocol

- Sends warning and fatal error messages during the session.
- Notifies closure of SSL session using close_notify.

4. Record Protocol

- Performs fragmentation, optional compression, encryption, and MAC generation.
- Ensures data confidentiality and integrity during transmission.

23. How does PGP achieve confidentiality and authentication in emails?



(c) Confidentiality and authentication

Notations:

- M = Message
- PR_a, PR_b = Private Keys of Sender (A) and Receiver (B)
- PU_a , PU_b = Public Keys of Sender (A) and Receiver (B)
- EP / DP = Public Key Encryption & Decryption Algorithms
- EC / DC = Symmetric Encryption & Decryption Algorithms
- Z/Z^{-1} = Compression & Decompression functions.
- H = Hash function
- K_s = Session key
- | | = Concatenation

Authentication:

To achieve authentication, PGP ensures that the message truly comes from the claimed sender and hasn't been altered during transmission. This is done using **digital signatures**, which allow the receiver to verify the sender's identity and the integrity of the message.

1. The sender creates a message.
2. SHA-1 is used to generate a 160-bit hash code of the message.
3. The hash code is encrypted with RSA using the sender's private key, and the result is prepended to the message.
4. The receiver uses RSA with the sender's public key to decrypt and recover the hash code.
5. The receiver generates a new hash code for the message and compares it with the decrypted hash code. If the two match, the message is accepted as authentic.

Confidentiality:

To ensure confidentiality, PGP encrypts the message so that only the intended recipient can read it. This is done by combining **symmetric encryption** (for speed) with **public key encryption** (for secure key exchange), protecting the message from unauthorized access.

1. The sender generates a message and a random 128-bit number to be used as a session key for this message only.
2. The message is encrypted using CAST-128 (or IDEA or 3DES) with the session key.
3. The session key is encrypted with RSA using the recipient's public key and is prepended to the message.
4. The receiver uses RSA with its private key to decrypt and recover the session key.
5. The session key is used to decrypt the message.

This is how PGP achieves both confidentiality and authentication in emails—by using a combination of **digital signatures** for verifying sender identity and **hybrid encryption** (symmetric + public key) for secure message transmission.

24. Explain various types of firewall.

1. PACKET FILTERING FIREWALLS

Definition: Packet filtering firewalls inspect individual packets of data as they pass through the network and make decisions to allow or block them based on predefined rules at the network layer (Layer 3) of the OSI model.

Operation: These firewalls examine attributes of packets such as source and destination IP addresses, ports, and protocol types to determine whether to permit or deny the traffic.

Example Rules:

- Allow incoming traffic on port 80 (HTTP) for web browsing.
- Block incoming traffic with a source IP address from a specific blacklist

Example: Windows Firewall in Microsoft Windows.

2. CIRCUIT-LEVEL GATEWAYS

Definition: Circuit-level gateways operate at the session layer (Layer 5) of the OSI model. They don't inspect packet contents. Instead, they monitor TCP handshakes to ensure proper connection establishment, follows the proper sequence of steps.

Operation: When a connection attempt is made, the circuit-level gateway acts as an intermediary, establishing a virtual circuit between the client and the server. It verifies that the TCP handshake (SYN, SYN-ACK, ACK) is completed successfully before allowing traffic to pass through.

Example Rules:

- Verify that the TCP handshake follows the SYN, SYN-ACK, ACK sequence.

Example: Microsoft Forefront Threat Management Gateway (TMG).

3. STATEFUL INSPECTION FIREWALLS

Definition: Stateful inspection firewalls, also known as dynamic packet filtering firewalls, operate at the network layer (Layer 3) of the OSI model. Unlike traditional packet filtering firewalls, they keep track of the state of active connections and make decisions based on the context of the traffic.

Operation: These firewalls not only inspect individual packets but also monitor the state of connections over time. They maintain a state table that tracks the state of each connection, including information such as source and destination IP addresses, ports, and sequence numbers.

Example Rules:

- Allow incoming traffic on port 80 (HTTP) only if it is part of an established connection.

- Block incoming traffic from IP addresses that have attempted suspicious activities within a defined timeframe.

Example: Cisco ASA

4. APPLICATION OR PROXY FIREWALLS

Definition: Application or proxy firewalls operate at the application layer (Layer 7) of the OSI model. They act as intermediaries between clients and servers, intercepting and inspecting traffic at the application level.

Operation: Instead of merely examining packet headers, these firewalls analyse the entire contents of network traffic, allowing for granular control and advanced security features.

Example Rules:

- Monitor and control specific applications or protocols, such as HTTP, FTP, or SMTP.
- Authenticate users before allowing access to certain applications or services

Example: McAfee Web Gateway

5. NEXT-GENERATION FIREWALLS

Definition: Next-generation firewalls integrate traditional firewall features with advanced security capabilities, such as application awareness, intrusion prevention, and user identity awareness, to provide enhanced protection against modern threats.

Operation: NGFWs inspect and control traffic at both the network and application layers (Layer 3 to Layer 7), allowing for more granular security policies based on application, user, and content.

Example Rules:

- Allow access to social media applications during non-business hours only.
- Block file uploads to cloud storage services from unapproved user groups.

Example: Check Point NGFW

25. Differentiate between IDS and Firewall. #

Parameter	Firewall	IDS
Purpose	Blocks unauthorized access to or from a network	Detects suspicious or malicious activity
Function	Filters and controls incoming/outgoing traffic	Monitors, analyzes, and alerts on traffic
Traffic Handling	Active – blocks or allows traffic based on rules	Passive – does not block traffic, only alerts
Position in Network	Positioned at the network perimeter	Placed inside the network
Response Type	Prevents threats by enforcing security policies	Detects and alerts, may trigger responses
Awareness Level	Rules-based filtering (IP, port, protocol)	Deep packet inspection and behavior analysis
Threat Detection	Defends against external threats	Detects internal and external attacks
Examples	pfSense, iptables, Cisco ASA	Snort, Suricata, OSSEC

26. Explain DDOS attack and how it is launched.

A Distributed Denial-of-Service (DDoS) Attack aims to disrupt a server, service, or network by flooding it with excessive traffic from multiple compromised devices. Attackers use botnets—networks of infected computers and IoT devices—to overwhelm targets, making it difficult to distinguish attack traffic from legitimate users.

How a DDoS Attack Works

1. Attackers infect multiple devices (bots) with malware.
2. These bots form a botnet, controlled remotely.
3. The attacker commands the botnet to flood the target with traffic, exhausting its resources and causing service disruption.

Types of DDOS attacks:

1. Volumetric Attacks

These are the most common type of DDoS attacks. They aim to consume all available bandwidth between the target and the internet by sending massive volumes of fake traffic. The goal is to clog the network pipes, making legitimate traffic unable to get through.

- **Example:** *UDP Flood* – attackers send a large number of UDP packets to random ports on a target server, overwhelming its ability to process and respond.

2. Protocol Attacks

Also known as state-exhaustion attacks, these exploit vulnerabilities in network protocols to exhaust server resources like firewalls, load balancers, and connection tables.

- **Example:** *SYN Flood* – the attacker sends repeated TCP connection requests (SYN packets) without completing the handshake, leaving the server waiting and tying up resources.

3. Application Layer Attacks

These target the top layer of the OSI model (Layer 7) and aim to crash web applications or servers by sending seemingly legitimate, but malicious requests that are computationally expensive to handle.

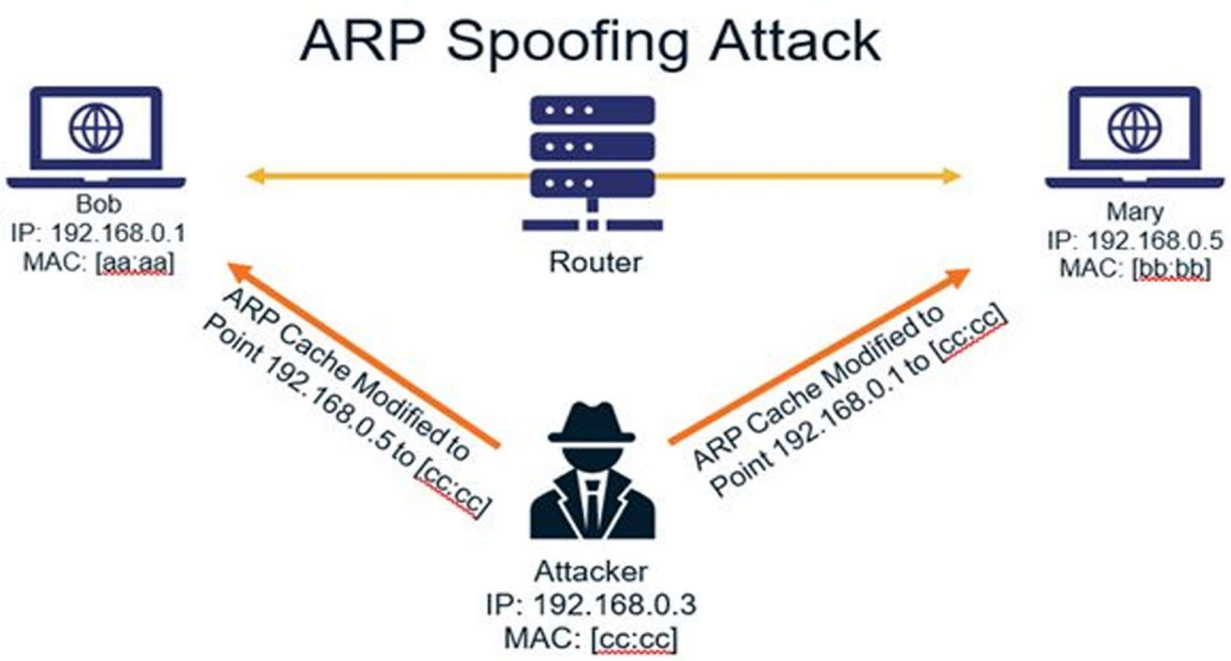
- **Example:** *HTTP GET/POST Flood* – attackers mimic normal users by sending many HTTP requests to a website, causing slowdowns or crashes.

27. Write a short note on ARP spoofing. #

ARP Spoofing is a type of cyberattack where an attacker sends fake ARP messages on a local network. The goal is to associate their own MAC address with the IP address of another device, such as a router or gateway.

As a result, traffic meant for the legitimate device is unintentionally sent to the attacker, enabling them to:

- Intercept sensitive information (e.g., passwords, emails)
- Modify or corrupt data (Man-in-the-Middle attack)
- Launch Denial-of-Service (DoS) attacks



Prevention Techniques

- **Network Segmentation**:
  - Divide the network into smaller segments using VLANs (Virtual LANs) to isolate devices and reduce the impact scope of ARP spoofing.
- **Intrusion Detection/Prevention Systems (IDS/IPS)**:
  - Deploy IDS/IPS to monitor and alert for unusual ARP traffic patterns, helping to detect or block spoofing attempts.

28. Explain TCP/IP vulnerabilities layer wise. #

The TCP/IP model has four layers, each with its own set of vulnerabilities:

1. Application Layer

- **Buffer overflows in services like HTTP, FTP, SMTP:** Attacker sends more data than a buffer can handle, overwriting memory and potentially executing malicious code.
- **Injection attacks (e.g., SQL Injection, Cross-Site Scripting):** Malicious input is inserted into a program to manipulate queries or scripts and gain unauthorized access or control..

2. Transport Layer

- **TCP session hijacking:** Attacker takes over an existing session.
- **SYN flood (DoS attack):** Exploits TCP 3-way handshake by sending multiple SYN requests.

3. Network Layer (Internet Layer)

- **IP spoofing**: Attacker fakes source IP address to disguise identity.
- **ICMP attacks (e.g., Ping of Death):** Attacker sends oversized or malformed ICMP packets to crash or freeze the target system.

4. Data Link Layer

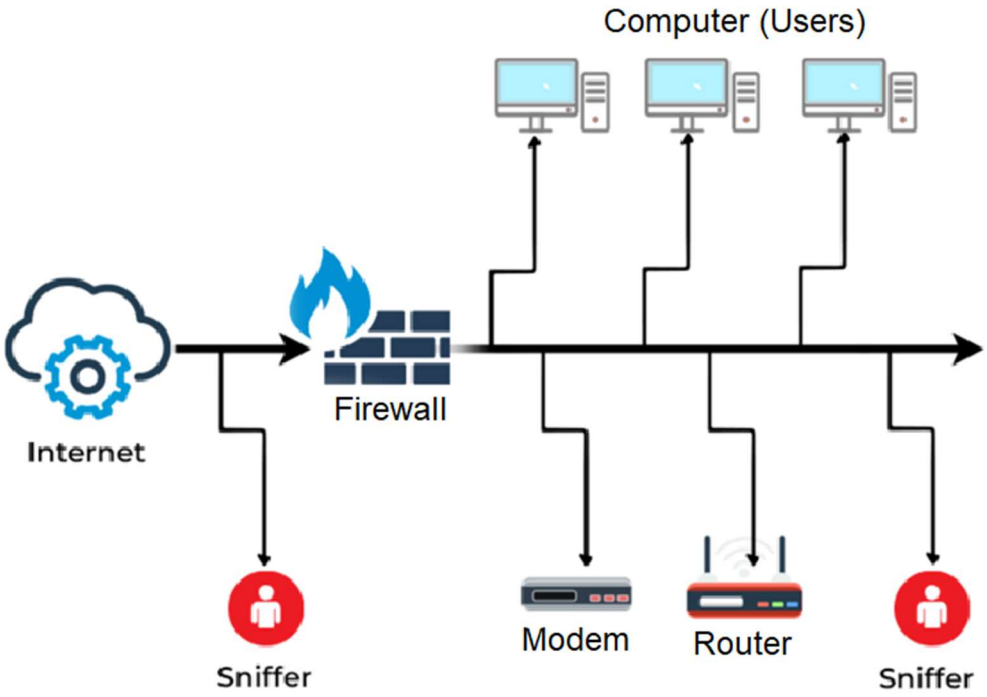
- **ARP spoofing:** Attacker associates their MAC address with another host's IP.
- **MAC flooding:** Overwhelms a switch's MAC table, forcing it to act like a hub.

29. Write a short note on: Packet Sniffing. #

Packet sniffing is a network monitoring technique where a program or device captures and analyses data packets as they travel across a network. It is commonly used by network administrators for troubleshooting and performance monitoring but can also be exploited by attackers for malicious purposes.

How Packet sniffing works:

Packet sniffing is done by using tools called packet sniffer. It can be either filtered or unfiltered. Filtered is used when only specific data packets have to be captured and Unfiltered is used when all the packets have to be captured. Wireshark, SmartSniff are examples of packet-sniffing tools.



Prevention Measures

- Use encryption protocols like HTTPS, SSH, and VPNs.
- Implement switch-based networks to limit broadcast traffic exposure.
- Use Intrusion Detection Systems (IDS) and firewalls.

30. IPSEC protocol: (asked 4 times, 3 different questions asked)

i. How is security achieved in Transport and Tunnel modes of IPSEC? Explain the role of AH and ESP.

ii. Explain IPSEC protocol in detail. Also write applications and advantages of IPSEC.

iii. How does ESP header guarantee confidentiality and integrity of packet payload? What is an authentication header (AH)? How does it protect against replay attack?

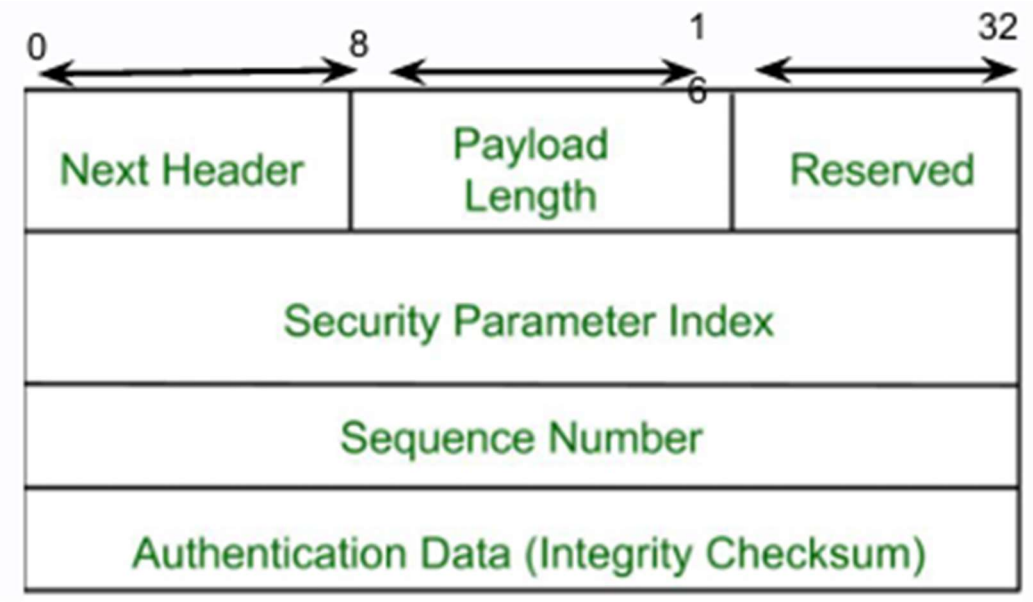
IPSec is a set of protocols that secure internet communications by providing encryption and authentication at the network layer (Layer 3) of the OSI model. It is used to establish secure connections between network devices, such as routers, firewalls, and hosts.

IPSec ensures secure data transmission over networks by providing three key security functions:

1. **Confidentiality** – Encrypts data to prevent unauthorized access or eavesdropping.
2. **Integrity** – Ensures that data remains unaltered during transmission.
3. **Authentication** – Verifies the identity of the communicating devices or users.

Authentication Header(AH):

- Provides authentication and integrity protection for IP packets.
- AH doesn't provide confidentiality as it doesn't encrypt but provides assurance that the packet has not been altered in transit.
- It uses Hash Algorithms like MD5, SHA1.
- Protects against replay attacks.
- AH supports two modes of operation: Transport Mode and Tunnel Mode.



Next Header(8 bits): Identify the type of Next payload or next actual data.

Payload Length(8 bits): It measures the length of the authentication header itself excluding the payload.

Reserved(16 bits): It is for future use and always set to zero.

Security Parameters Index (SPI) (32 bits): Unique identifier used to associate the packet with a particular security association.

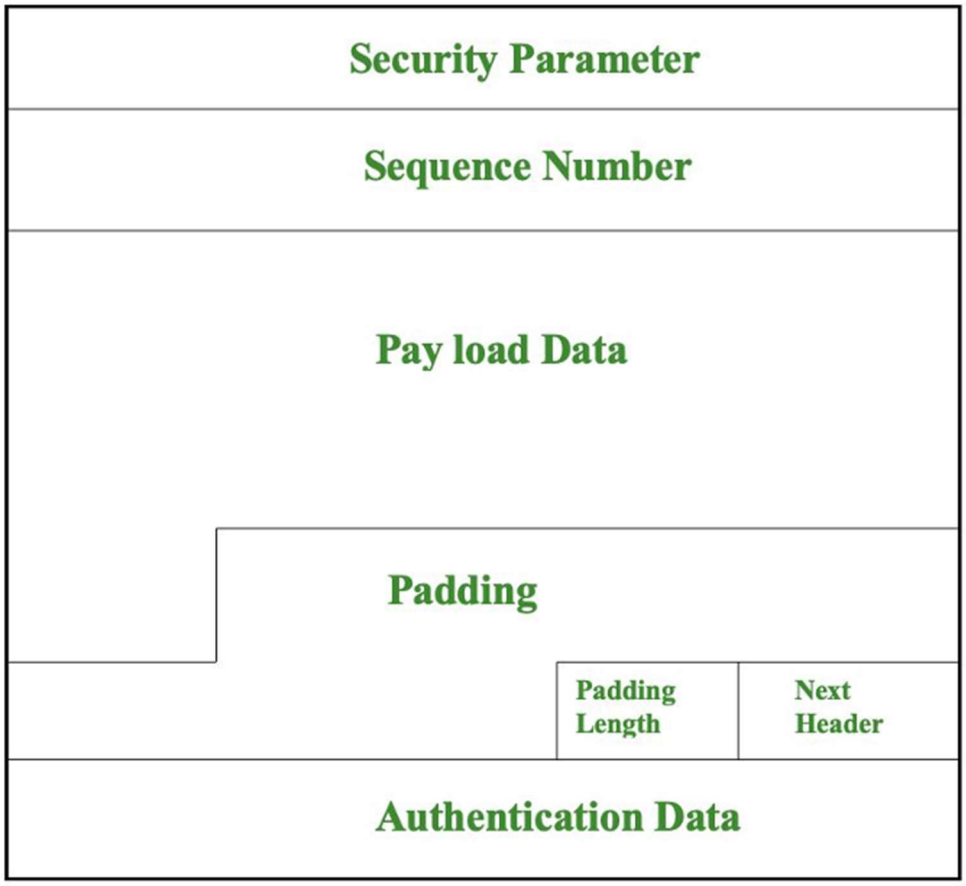
Sequence Number(32 bits): A monotonically increasing counter used to prevent replay attacks.

How AH protects against replay attacks:

- **AH**: Uses **sequence numbers** in the header, which are incremented with each packet. The receiver can check the sequence number to detect if a packet has been replayed.

Encapsulating Security Payload (ESP):

- Provides authentication, integrity, and confidentiality for IP packets
- The main functionality of ESP is to provide confidentiality to IP packet by encrypting them.
- Common encryption algorithms used with ESP include DES (Data Encryption Standard), 3DES (Triple DES), AES (Advanced Encryption Standard), and others.
- ESP provides data authentication, data integration and confidentiality by adding ESP Header, ESP trailer and MAC to the packet.
- ESP supports both Transport Mode and Tunnel Mode.



Security Parameters Index (SPI): A unique identifier used to associate the packet with a particular security association.

Sequence Number: A monotonically increasing counter used for anti-replay protection.

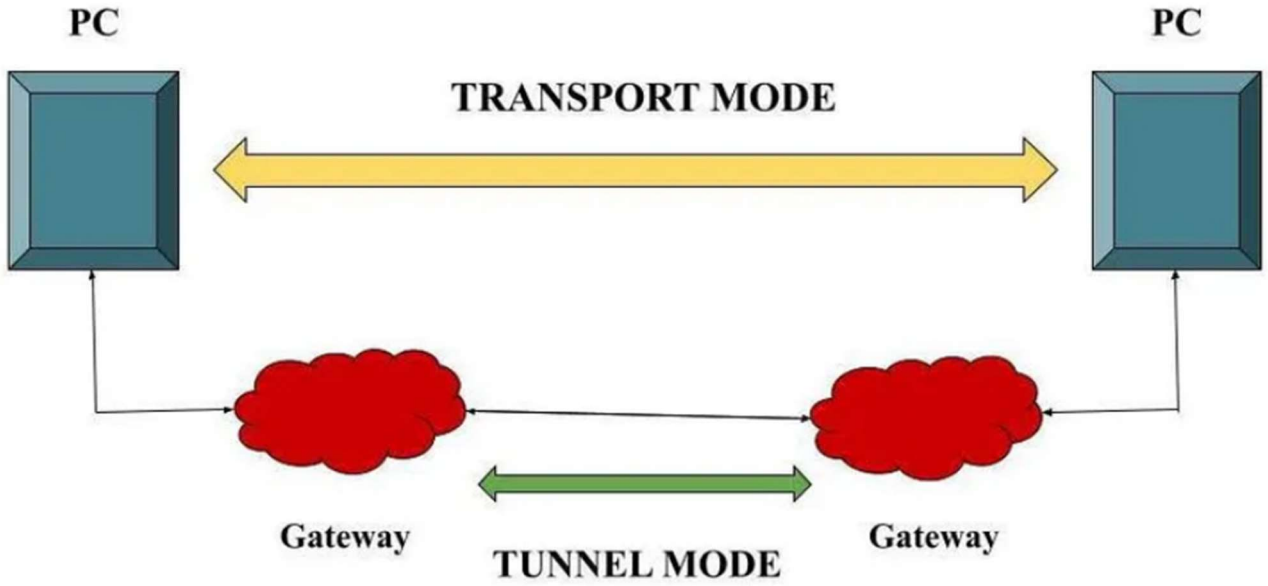
Payload Data: The encrypted and/or authenticated payload of the packet.

Padding: Additional padding may be added to ensure alignment with encryption block sizes(0-255 octets)

Padding Length: Mandatory field in ESP used to indicate no. of padding (protection) added in the packet.

Next Header: Indicates the protocol of the payload (e.g., TCP, UDP)

IPSEC Modes of Operation:



Transport Mode --- Only payload encrypted

In transport mode, only the payload of the IP packet is usually encrypted and authenticated. The routing is intact because the IP header is neither modified nor encrypted

Used to secure communication between two hosts (end-to-end) or between a host and a security gateway.

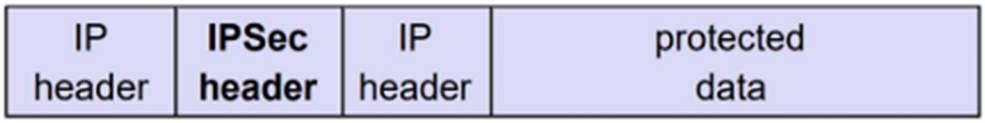
Security achieved by:

- **AH (Authentication Header):** Provides authentication and integrity but no encryption.
- **ESP (Encapsulating Security Payload):** Encrypts and optionally authenticates the data.

Tunnel Mode --- Entire IP packet encrypted

The original packet is encapsulated in a new IP packet (both its IP header and its payload).

Used to establish secure communication between two networks or between a network and a remote user.



Security achieved by:

- **AH:** Authenticates the entire original packet (including original header).
- **ESP:** Encrypts the whole original IP packet for confidentiality and optionally authenticates it.

Applications of IPsec:

- **VPNs (Virtual Private Networks):** IPsec is widely used in VPNs for secure communication between remote users and networks.
- **Site-to-Site Connectivity:** IPsec is used for securing communication between different network gateways.

Advantages of IPsec:

- **End-to-End Security:** Provides secure communication over untrusted networks.
- **Transparency:** Works at the IP layer, so applications don't need to be aware of the security features.
- **Flexibility:** Can be used in Transport or Tunnel mode depending on the specific requirements.

6. System Security

31. Explain buffer overflow attack. #

Buffers are memory storage regions that temporarily hold data while it is being transferred from one location to another. A buffer overflow (or buffer overrun) occurs when the volume of data exceeds the storage capacity of the memory buffer. As a result, the program attempting to write the data to the buffer overwrites adjacent memory locations.

Types of Buffer Overflow Attacks:

- **Stack-based Overflow**:
  Happens in the temporary memory used by functions; attackers can overwrite function return addresses.
- **Heap-based Overflow**:
  Happens in the memory used for dynamic (runtime) data; harder to exploit but still dangerous.
- **Integer Overflow**:
  Occurs when a calculation goes beyond the maximum value an integer can hold, leading to memory issues and overflows.

How to Prevent Buffer Overflows

- **ASLR (Address Space Layout Randomization)**:
  Randomly changes memory locations to make it harder for attackers to guess where to inject malicious code.
- **Data Execution Prevention (DEP)**:
  Blocks certain areas of memory from running code, stopping attackers from executing injected code.

~ AJ

Asked once:

1.Introduction - Number Theory and Basic Cryptography

1. List and explain various types of attacks on encrypted message. #
2. Explain Euclidian Algorithm. #
3. Use Hill cipher to encrypt the text “short.” The key to be used is hill.
4. Explain with examples keyed and keyless transposition cipher. #

2. Symmetric and Asymmetric key Cryptography and key Management

5. Explain algorithmic modes encryption process of symmetric key. #
6. Explain DES algorithm with flowcharts.
7. Explain RC4 stream cipher. #
8. Explain Public Key Distribution in detail.
9. Explain Needham-Schroeder authentication protocol.
10. Explain man in middle attack on Diffie Hellman. Explain how to overcome the same.
11. What is PKI? List its components.
12. What is digital certificate? How does it help to validate authenticity of a user. Explain X.509 certificate format.
13. Highlight the difference between AES and DES. #

3. Cryptographic Hash Functions

14. Provide a comparison between HMAC, CBC-MAC and CMAC
15. What goals are served using a message digest? Explain using MD5.

4. Authentication Protocols & Digital Signature Schemes

16. Explain challenge response-based authentication tokens. #

5. Network Security and Applications

17. What is ICMP flood attack? Explain in detail.
18. What are the different components of IDS? List and explain different approaches of IDS.
19. Explain key rings in PGP. #

6. System Security

20. Explain worms and viruses. #
21. Write a short note on: SQL injection. #
22. List various Software Vulnerabilities. How vulnerabilities are exploited to launch an attack. #

Not enough time to study all, will make later if there’s time