CNS-2024-May-PYQ

Q1. [20 Marks]

a. Distinguish between passive and active security attacks

b. Differentiate between virus and worm.

c. Explain SSH protocol stack in brief

d. Write short note on: Email Security

Q2. [10 Marks]

a. Discuss classical encryption techniques with example

b. Explain different types of denial-of-service attacks

Q3. [10 Marks]

a. What are Block cipher modes? Describe any two in detail

b. Given modulus n = 221 and public key e = 7, find the values of p, q, phi(n) and d using RSA. Encrypt M = 5.

Q4. [10 Marks]

a. Discuss various NAC enforcement methods

b. Design sample digital certificate and explain each field of it

Q5. [10 Marks]

a. Show how a Kerberos protocol can be used to achieve single sign-on in distributed systems.

b. Explain the different types of protocol offered by SSL

Q6. [10 Marks]

a. Why is there a need for a firewall? Explain the different types of firewalls

b. How does IPSec help to achieve authentication and confidentiality? Justify the need of AH and ESP

Q1. [5 Marks] - Answers

a. Distinguish between passive and active security attacks

Aspect	Passive Attack	Active Attack
Definition	Attempts to **monitor or eavesdrop** on communication without altering the data.	Attempts to **modify, disrupt, or destroy** data or communication.
Objective	To **gain unauthorized information**.	To **alter system resources or data** and affect operations.
Effect on Data	**No change** to the original data; only observation occurs.	**Changes or damages** the data or system functionality.
Detection	**Difficult to detect**, as no visible alteration occurs.	**Easier to detect**, as it causes noticeable effects.
Examples	Eavesdropping, traffic analysis, password sniffing.	Masquerading, message modification, denial-of-service (DoS), replay attack.
Prevention Method	Use of **encryption** and secure communication channels.	Use of **authentication, integrity checks, and intrusion detection systems (IDS)**.
Goal	**Information gathering**.	**System manipulation or disruption**.

b. Differentiate between virus and worm.

Aspect	Virus	Worm
Definition	A **malicious program** that attaches itself to a legitimate file or program and spreads when the host is executed.	A **self-replicating malicious program** that spreads automatically across networks without attaching to other files.

Aspect	Virus	Worm
Dependency	Requires a **host program** to run and propagate.	**Independent program**; does not need a host to spread.
Spreading Method	Spreads through **infected files**, removable drives, or user actions (like opening attachments).	Spreads through **network connections**, exploiting vulnerabilities automatically.
Execution	Needs **user action** (e.g., running an infected file) to activate.	Spreads and executes **automatically**, without user intervention.
Impact	Usually **corrupts or modifies files** and slows down the system.	Consumes **network bandwidth**, causes **system slowdowns** or **network congestion**.
Example	Melissa Virus, ILOVEYOU Virus.	Conficker Worm, Mydoom Worm.
Prevention	Use **antivirus software** and avoid suspicious downloads.	Use **firewalls**, **patch system vulnerabilities**, and monitor network traffic.

c. Explain SSH protocol stack in brief

SSH Protocol Stack (Secure Shell Protocol)

- **Definition:**

  SSH (Secure Shell) is a **network protocol** used for **secure remote login and data transfer** between two networked devices over an unsecured network.

Layers of SSH Protocol Stack:

1. **Transport Layer (SSH-TRANS):**
   - Establishes a **secure and encrypted connection** between client and server.
   - Provides **confidentiality**, **integrity**, and **server authentication**.
   - Uses **algorithms like AES (encryption)** and **HMAC (integrity check)**.
   - Runs typically over **TCP port 22**.

2. **Authentication Layer (SSH-AUTH):**
    - Verifies the **identity of the user/client**.
    - Supports multiple authentication methods:
        - Password-based
        - Public key-based
        - Host-based authentication
    - Ensures only **authorized users** gain access.

3. **Connection Layer (SSH-CONN):**
    - Manages **multiple logical channels** over a single SSH session.
    - Each channel can carry different services like:
        - **Remote shell access** (command execution)
        - **File transfer** (SCP/SFTP)
        - **Port forwarding**
    - Ensures **session control and multiplexing**.

d. Write short note on: Email Security

Email Security

- **Definition:**

  Email security refers to the **measures and protocols** used to protect email communication from **unauthorized access, data theft, phishing, and malware attacks**.

Key Objectives:

1. **Confidentiality:** Ensures only intended recipients can read the email (using **encryption**).

2. **Integrity:** Protects emails from **modification or tampering** during transmission.
3. **Authentication:** Verifies the **identity of the sender** to prevent spoofing.
4. **Non-repudiation:** Ensures the sender **cannot deny** sending the email.

Techniques and Tools:

- **Encryption:** Uses standards like **S/MIME** and **PGP** to secure message content.
- **Digital Signatures:** Ensure message **authenticity** and **integrity**.
- **Spam Filters:** Block **junk or phishing emails**.
- **Anti-Malware Protection:** Scans attachments for **viruses and malicious scripts**.
- **Authentication Protocols:**
  - **SPF (Sender Policy Framework)**
  - **DKIM (DomainKeys Identified Mail)**
  - **DMARC (Domain-based Message Authentication, Reporting & Conformance)**

Example:

Corporate email systems use **S/MIME encryption**, **digital signatures**, and **spam filters** to protect employees from phishing and data breaches.

Q2. [10 Marks] - Answers

a. Discuss classical encryption techniques with example

Classical Encryption Techniques

Classical encryption techniques are the **traditional methods of securing messages** before the advent of modern cryptography. They are broadly classified into **Substitution** and **Transposition** techniques.

---

1. Substitution Techniques

In these, **each element (letter/bit) of plaintext is replaced with another symbol**.

(a) **Caesar Cipher**

- Each letter is shifted by a fixed number of positions in the alphabet.
- Example: Shift = 3
  - Plaintext: `HELLO`
  - Ciphertext: `KHOOR`

(b) **Monoalphabetic Cipher**

- Each letter is substituted by another fixed letter (random substitution).
- Example:
  - Mapping: A→Q, B→W, C→E ...
  - Plaintext: `HELLO`
  - Ciphertext: `ITSSG`

(c) **Playfair Cipher**

- Uses a **5×5 matrix** of a keyword.
- Encrypts **pairs of letters (digraphs)**.
- Example with keyword **“PLAYFAIR”**:
  - Plaintext: `HI DE` → Ciphertext: `BM OD`

(d) **Vigenère Cipher**

- Uses a **keyword** to shift letters by varying amounts.
- Example:
  - Keyword: `KEY`

    - Plaintext: `HELLO`
    - Ciphertext: `RIJVS`

---

2. Transposition Techniques

In these, the **positions of characters are rearranged**, but the letters remain unchanged.

(a) **Rail Fence Cipher**

- Write plaintext in a zig-zag (rails), then read row by row.
- Example:
    - Plaintext: `HELLO WORLD`
    - Ciphertext: `HLOOL ELWRD`

(b) **Columnar Transposition**

- Write plaintext in a rectangle (rows), read column by column according to a key.
- Example:
    - Key: 3 1 2
    - Plaintext: `HELLO` → Arrange:

```
H E L
L O X
```

Read columns in order: **ELX HLO**

b. Explain different types of denial-of-service attacks

What is a DoS Attack?

- A **Denial-of-Service (DoS)** attack aims to make a system, service, or network **unavailable** to legitimate users.
- It overwhelms resources like bandwidth, CPU, or memory, causing **slowdowns or crashes**.

Types of DoS Attacks

1. **Volume-Based Attacks**

- Flood the network with massive traffic to exhaust bandwidth.
- Examples:
  - **UDP Flood**: Sends large numbers of UDP packets to random ports.
  - **ICMP Flood (Ping Flood)**: Overloads the target with ICMP Echo Requests.

2. **Protocol Attacks**

- Exploit weaknesses in network protocols to consume server resources.
- Examples:
  - **SYN Flood**: Sends many TCP SYN requests but never completes the handshake.
  - **Ping of Death**: Sends oversized ping packets that crash the system.

3. **Application Layer Attacks**

- Target specific applications or services (e.g., HTTP, DNS).
- Aim to exhaust server-side resources like threads or database connections.
- Example:
  - **HTTP GET/POST Flood**: Sends repeated requests to web servers.

4. **Distributed Denial-of-Service (DDoS)**

- Attack launched from **multiple compromised systems** (botnet).
- Harder to block due to traffic coming from many sources.
- Example:
  - DDoS on a banking website using thousands of infected devices.

5. **Slowloris Attack**

- Sends partial HTTP requests slowly to keep connections open.
- Exhausts web server's connection pool without high bandwidth usage.

6. **DNS Amplification**

- Exploits open DNS servers to reflect and amplify traffic to the victim.
- Small request → large response → overloads target.

Q3. [10 Marks] - Answers

a. What are Block cipher modes? Describe any two in detail

What are Block Cipher Modes?

- Block ciphers encrypt data in **fixed-size blocks** (e.g., 64 or 128 bits).
- **Block cipher modes of operation** define how these blocks are processed to securely encrypt larger messages.
- They determine how each block interacts with others and how patterns are avoided.

Common Block Cipher Modes

Some widely used modes include:

- ECB (Electronic Codebook)
- CBC (Cipher Block Chaining)
- CFB (Cipher Feedback)
- OFB (Output Feedback)
- CTR (Counter Mode)
- GCM (Galois/Counter Mode)

1. ECB (Electronic Codebook Mode)

Working:

- Each block is encrypted **independently** using the same key.
- No chaining or dependency between blocks.

Advantages:

- Simple and fast.
- Parallel encryption possible.

Disadvantages:

- **Pattern leakage**: identical plaintext blocks → identical ciphertext blocks.
- Not suitable for encrypting images or structured data.

Example:

Plaintext blocks: `A B A`
Ciphertext blocks: `X Y X` (same input → same output)

2. CBC (Cipher Block Chaining Mode)

Working:

- Each plaintext block is **XORed with the previous ciphertext block** before encryption.
- First block uses an **Initialization Vector (IV)**.

Advantages:

- Eliminates pattern repetition.
- More secure than ECB.

Disadvantages:

- Slower due to sequential dependency.
- Errors propagate: one corrupted block affects the next.

Example:

```
C1 = Encrypt(P1 ⊕ IV)
C2 = Encrypt(P2 ⊕ C1)
C3 = Encrypt(P3 ⊕ C2)
```

b. Given modulus n = 221 and public key e = 7, find the values of p, q, phi(n) and d using RSA. Encrypt M = 5.

CNS - 2024 - May - PYQ

Q.3] Ans - b

Given :

$n = 221$

public key $(e) = 7$

Encrypt $M/P = 5$

Values to find p, q, $\phi(n)$ and d using RSA.

i] Factoring "n" to get p & q

$\sqrt{221} = 14.8$

Now, we will be looking for integers that are divisible by "221" around the value of "14.8"

which are $221 \div 13 = 17$

$221 \div 17 = 13$

$\therefore 221 = 17 \times 13$

$\boxed{p = 13}$ & $\boxed{q = 17}$

ii] Computing $\phi(n)$

formula to find $\phi(n)$:

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

$$\phi(n) = (13-1) \times (17-1)$$

$$= 12 \times 16$$

$$\therefore \boxed{\phi(n) = 192}$$

iii) Finding the private key (d)

Formula: $\underset{\text{public key}}{d\,e} = 1 + \overset{\text{constant}}{k}\,\phi(n)$

$$d = \frac{1 + k\,\phi(n)}{e} \quad \text{for } [k = 0, 1, 2, 3 \ldots]$$

$$d = \frac{1 + 0 \times 192}{7} = \frac{1}{7} = 0.142 \quad \times$$

$$d = \frac{1 + (1 \times 192)}{7} = \frac{193}{7} = 27.57 \quad \times$$

$$d = \frac{1 + (2 \times 192)}{7} = \frac{385}{7} = 55 \quad \checkmark$$

Note: we have to take constants [k = 0, 1, 2, 3] until we get the value of "d" a whole number or "without decimal"

$$\therefore \boxed{\text{Private key } (d) = 55}$$

4] Encrypt $M=5$ or finding the value of ciphertext $(C) = ?$

formual : P $\underset{\text{ciphertext}}{C} = \underset{\text{plaintext}}{P}^{\overset{\text{public key}}{e}} \bmod n$

$\therefore C = 5^7 \bmod 221$ .... $(M = P = 5)$

$= 78125 \bmod 221$

$\therefore \boxed{C = \underline{\underline{112}}}$

Answer :- $P = 13$, $q = 17$, $\phi(n) = 192$

$d = 55$ & $C = \underline{\underline{112}}$

Q4. [10 Marks] - Answers

a. Discuss various NAC enforcement methods

What is NAC Enforcement?

Network Access Control (NAC) enforcement refers to **how access policies are applied** to devices attempting to connect to a network. It ensures that only **authorized and compliant devices** gain access, and others are blocked, quarantined, or restricted.

Types of NAC Enforcement Methods

1. **Inline Enforcement (In-Band)**

- NAC device sits **directly in the data path** between endpoints and the network.
- It actively monitors and controls traffic.
- **Pros**: Real-time control, strong security.
- **Cons**: Can introduce latency or become a single point of failure.

2. **Out-of-Band Enforcement**

- NAC device operates **outside the data path**.
- Uses network infrastructure (e.g., switches, routers) to enforce policies via protocols like SNMP or RADIUS.
- **Pros**: Scalable, less intrusive.
- **Cons**: Slower response, depends on third-party device support.

3. **802.1X Enforcement**

- Uses **port-based authentication** on switches and wireless access points.
- Devices must authenticate before gaining network access.

- Common in enterprise environments.
- **Pros**: Strong identity-based control.
- **Cons**: Requires compatible infrastructure and configuration.

4. **DHCP Enforcement**

- NAC controls access by managing **IP address assignment**.
- Non-compliant devices may be assigned to a restricted VLAN or denied IP.
- **Pros**: Easy to implement.
- **Cons**: Less secure, bypassable by static IPs.

5. **VPN Enforcement**

- Applies NAC policies to devices connecting via **Virtual Private Network**.
- Ensures remote users meet compliance before accessing internal resources.
- **Pros**: Secures remote access.
- **Cons**: Depends on VPN client integration.

6. **Virtual Firewall or Agent-Based Enforcement**

- Uses **software agents** on endpoints to enforce policies.
- Can restrict access based on device health, location, or user role.
- **Pros**: Granular control.
- **Cons**: Requires agent installation and maintenance.

b. Design sample digital certificate and explain each field of it

Sample Digital Certificate (X.509 Format)

```
Certificate:
    Version: 3
    Serial Number: 0456789A
    Signature Algorithm: sha256WithRSAEncryption
    Issuer: CN=CertAuthority, O=SecureOrg, C=IN
    Validity:
        Not Before: Nov 1 2025
        Not After : Nov 1 2026
    Subject: CN=www.example.com, O=ExampleCorp, C=IN
    Subject Public Key Info:
        Public Key Algorithm: RSA
        RSA Public-Key: (2048 bit)
        Modulus: ...
        Exponent: 65537 (0x10001)
    Extensions:
        Key Usage: Digital Signature, Key Encipherment
        Subject Alternative Name: DNS:www.example.com, DNS:example.net
    Signature:
        Signature Algorithm: sha256WithRSAEncryption
        Signature Value: ...
```

Explanation of Each Field

1. **Version**
    - Indicates the X.509 version (usually v3).
    - v3 supports extensions like SAN (Subject Alternative Name).

2. **Serial Number**
    - Unique identifier assigned by the Certificate Authority (CA).
    - Used to track and revoke certificates.

3. **Signature Algorithm**
   - Specifies the algorithm used to sign the certificate (e.g., SHA-256 with RSA).
   - Ensures authenticity and integrity.

4. **Issuer**
   - Identifies the CA that issued the certificate.
   - Includes Common Name (CN), Organization (O), Country (C).

5. **Validity**
   - Defines the time period during which the certificate is valid.
   - Includes "Not Before" and "Not After" dates.

6. **Subject**
   - Identifies the entity the certificate is issued to (e.g., website, user).
   - Includes CN, O, C similar to Issuer.

7. **Subject Public Key Info**
   - Contains the public key and algorithm (e.g., RSA).
   - Used by others to encrypt data or verify signatures.

8. **Extensions**
   - Additional attributes that enhance functionality:
     - **Key Usage**: Specifies allowed operations (e.g., signing, encryption).
     - **Subject Alternative Name (SAN)**: Lists additional domain names or IPs.

9. **Signature**
    - Digital signature created by the CA using its private key.
    - Verifies that the certificate hasn’t been tampered with.

Q5. [10 Marks] - Answers

a. Show how a Kerberos protocol can be used to achieve single sign-on in distributed systems.

What is Kerberos?

- **Kerberos** is a network authentication protocol that uses **secret-key cryptography** and a trusted third party (Key Distribution Center) to authenticate users securely.
- It enables **Single Sign-On (SSO)** by allowing users to authenticate once and access multiple services without re-entering credentials.

Key Components

- **Client**: User or device requesting access.
- **Authentication Server (AS)**: Verifies user identity.
- **Ticket Granting Server (TGS)**: Issues service tickets.
- **Service Server (SS)**: Hosts the requested service.
- **Key Distribution Center (KDC)**: Combines AS and TGS.

Kerberos Workflow for SSO

1. **Initial Login**
    - User logs in and sends a request to the **Authentication Server (AS)**.

- AS verifies credentials and issues a **Ticket Granting Ticket (TGT)** encrypted with the user's secret key.

2. **Requesting Service Access**
   - Client sends the TGT to the **Ticket Granting Server (TGS)** along with the service request.
   - TGS verifies the TGT and issues a **Service Ticket** for the requested service.
3. **Accessing the Service**
   - Client presents the Service Ticket to the **Service Server (SS)**.
   - SS validates the ticket and grants access without requiring re-authentication.

How It Achieves SSO

- User authenticates **once** to get the TGT.
- Subsequent access to services uses **tickets**, not passwords.
- Reduces password exposure and improves user experience across distributed systems.

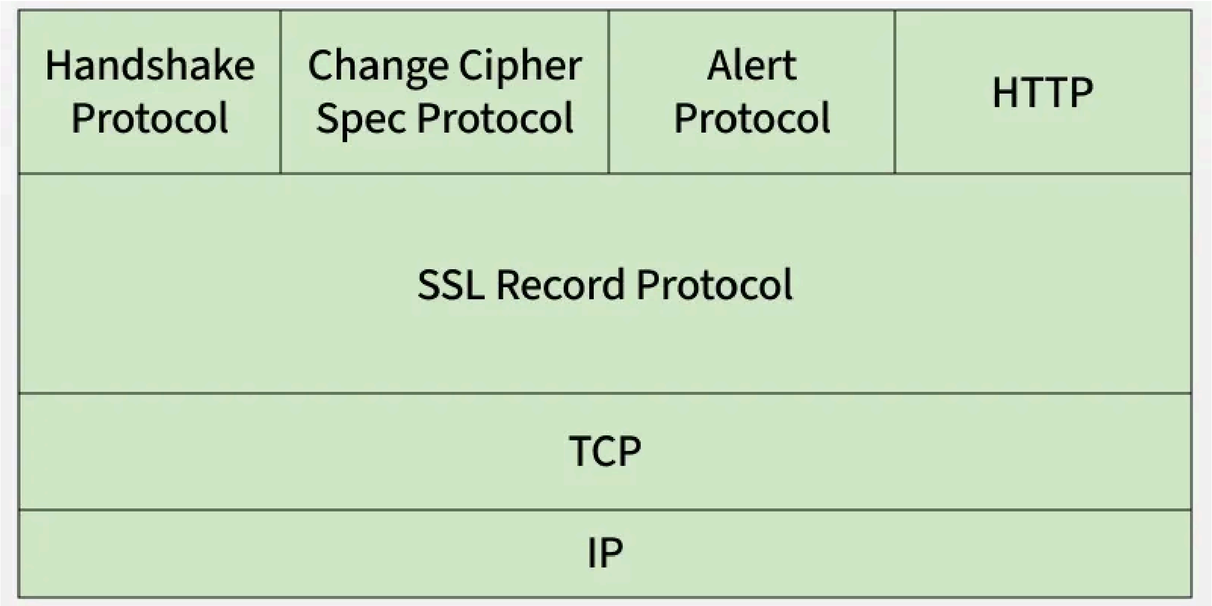
Example Scenario

- Sameer logs into his organization’s portal.
- Kerberos authenticates him and issues a TGT.
- He accesses email, file server, and internal apps — all using service tickets without re-entering his password.

b. Explain the different types of protocol offered by SSL

What is SSL?

- **SSL** is a cryptographic protocol designed to provide **secure communication** over networks.
- It ensures **confidentiality, integrity, and authentication** between client and server.
- SSL has evolved into **TLS (Transport Layer Security)**, but the term SSL is still widely used.



Secure Socket Layer (SSL)

Types of Protocols Offered by SSL

SSL uses several sub-protocols to manage secure communication:

1. **Handshake Protocol**

- Establishes the secure session between client and server.
- Performs:
  - Authentication (using certificates)
  - Key exchange (e.g., RSA, Diffie-Hellman)

  - Agreement on encryption and MAC algorithms

Example: When you visit `https://example.com`, the browser and server perform a handshake to agree on security settings.

2. **Record Protocol**

- Handles the actual **data transmission** after the handshake.
- Provides:
  - Fragmentation
  - Compression (optional)
  - Encryption
  - Message Authentication (MAC)

Example: Encrypts HTTP data before sending it over the network.

3. **Alert Protocol**

- Communicates **error messages or warnings** between client and server.
- Alerts include:
  - Unexpected message
  - Bad certificate
  - Decryption failure
  - Close notify (session termination)

Example: If a certificate is invalid, an alert is sent to terminate the session.

4. **Change Cipher Spec Protocol**

- Signals that the parties should start using the newly negotiated **cipher suite**.
- Sent after the handshake and before encrypted communication begins.

Example: After agreeing on AES encryption, this protocol activates it.

Q6. [10 Marks] - Answers

a. Why is there a need for a firewall? Explain the different types of firewalls

Why Is There a Need for a Firewall?

Firewalls are essential for protecting networks and systems from unauthorized access and cyber threats. They act as a **security barrier** between trusted internal networks and untrusted external sources (like the internet).

Key Reasons:

- **Access Control**: Blocks unauthorized users and devices.
- **Threat Prevention**: Detects and prevents malware, DoS attacks, and intrusions.
- **Traffic Filtering**: Allows only legitimate traffic based on rules.
- **Monitoring and Logging**: Tracks network activity for audits and incident response.
- **Policy Enforcement**: Applies organizational security policies consistently.

Types of Firewalls

1. **Packet-Filtering Firewall**

- Filters traffic based on IP addresses, ports, and protocols.
- Operates at the **network layer**.
- **Pros**: Simple and fast.
- **Cons**: No deep inspection; vulnerable to spoofing.

2. **Stateful Inspection Firewall**

- Tracks the **state of active connections**.
- Allows packets that are part of a valid session.
- **Pros**: More secure than packet filtering.
- **Cons**: Higher resource usage.

3. **Application-Level Firewall (Proxy Firewall)**

- Operates at the **application layer**.
- Intercepts and inspects traffic for specific applications (e.g., HTTP, FTP).
- **Pros**: Deep packet inspection.
- **Cons**: Slower performance, complex setup.

4. **Next-Generation Firewall (NGFW)**

- Combines traditional firewall features with **advanced security**:
  - Intrusion Prevention System (IPS)
  - Deep packet inspection
  - Application awareness
- **Pros**: Comprehensive protection.
- **Cons**: Expensive and resource-intensive.

5. **Hardware vs Software Firewalls**

- **Hardware Firewall**: Dedicated device; used at network perimeter.
- **Software Firewall**: Installed on individual systems; protects host-level traffic.

b. How does IPSec help to achieve authentication and confidentiality? Justify the need of AH and ESP

How IPSec Achieves Authentication and Confidentiality

IPSec is a protocol suite that secures IP communications by providing:

- **Authentication**: Verifies the identity of the sender and ensures data integrity.
- **Confidentiality**: Encrypts data to prevent unauthorized access.

It operates at the **network layer**, securing all IP traffic regardless of the application.

Core Protocols in IPSec

IPSec uses two main protocols to achieve its goals:

1. **Authentication Header (AH)**

- Provides **data integrity** and **origin authentication**.
- Uses cryptographic hash functions (e.g., HMAC-SHA).
- **Does not encrypt** the payload — no confidentiality.
- Protects against **tampering and spoofing**.

Use Case: When encryption is not required but integrity and authenticity are critical (e.g., internal control systems).

2. **Encapsulating Security Payload (ESP)**

- Provides **confidentiality** by encrypting the payload.
- Also supports **authentication and integrity** (optional).
- Uses encryption algorithms like AES, DES.

Use Case: When secure transmission of sensitive data is needed (e.g., VPN traffic, remote access).

Justification for AH and ESP

Protocol	Provides Authentication	Provides Confidentiality	Use Case
AH	Yes	No	Integrity-only scenarios
ESP	Optional	Yes	Secure data transmission

- **AH is needed** when encryption is not allowed or necessary, but verifying sender and data integrity is crucial.
- **ESP is needed** when protecting data from eavesdropping is essential, especially over public networks.