SNS COLLEGE OF ENGINEERING
Kurumbapalayam (Po), Coimbatore – 641 107
AN AUTONOMOUS INSTITUTION

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

Department of Computer Science and Engineering

Cyber Security is the branch of computer technology that deals with the security of the virtual cloud and internet. Any information that is stored or transmitted through the cloud needs to be secure and safe. Cyber Networking plays a very important role in maintaining that the connection established is secured and content goes through a secured/ safe channel for transmission.

Security in the network is very important and can't be compromised in any situation. Security in Networking particularly in IP Sec or IP Network Security is significant and has some characteristics associated with it.

Characteristics Associated with IPSec:

1. The standardized algorithms present in IP Sec are SHA and MD5.
2. IPSec uniquely identifies every packet, and then authentication is carried out based on verifying the same uniqueness of the packet.
3. IP network or IPSec has an ESP present in it for security purposes.

Here, we will discuss ESP, the structure of ESP, and its importance in security.

Encapsulating security payload, also abbreviated as ESP plays a very important role in network security. ESP or Encapsulating security payload is an individual protocol in IPSec. ESP is responsible for the CIA triad of security (Confidentiality, Integrity, Availability), which is considered significant only when encryption is carried along with them. Securing all payload/ packets/ content in IPv4 and IPv6 is the responsibility of ESP.

As the name suggests, it involves encapsulation of the content/ payload encrypts it to suitable form and then there a security check or authentication takes place for payload in IP Network. Encryption/ encapsulation and security/ authentication make the payload extremely secure and safe from any kind of harm or threat to content/ data/ payload being stolen by any third party. The encryption process is performed by authenticated user,

similarly, the decryption process is carried out only when the receiver is verified, thus making the entire process very smooth and secure. The entire encryption that is performed by ESP is carried on the principle of the integrity of payload and not on the typical IP header.

Working of ESP:

1. Encapsulating Security Payload supports both main Network layer protocols: IPv4 and IPv6 protocols.
2. It performs the functioning of encryption in headers of Internet Protocol or in general say, it resides and performs functions in IP Header.
3. One important thing to note here is that the insertion of ESP is between Internet Protocol and other protocols such as UDP/ TCP/ ICMP.

Modes in ESP:

Encapsulating Security Payload supports two modes, i.e. Transport mode, and tunnel mode.

Tunnel mode:

1. Mandatory in Gateway, tunnel mode holds utmost importance.
2. Here, a new IP Header is created which is used as the outer IP Header followed by ESP.

Transport mode:

1. Here, IP Header is not protected via encryption or authentication, making it vulnerable to threats
2. Less processing is seen in this mode, so the inclusion of ESP is preferred

Advantages:

Below listed are the advantages of Encapsulating Security Payload:

1. Encrypting data to provide security
2. Maintaining a secure gateway for data/ message transmission
3. Properly authenticating the origin of data
4. Providing needed data integrity
5. Maintaining data confidentiality
6. Helping with antireplay service using authentication header

Disadvantages:

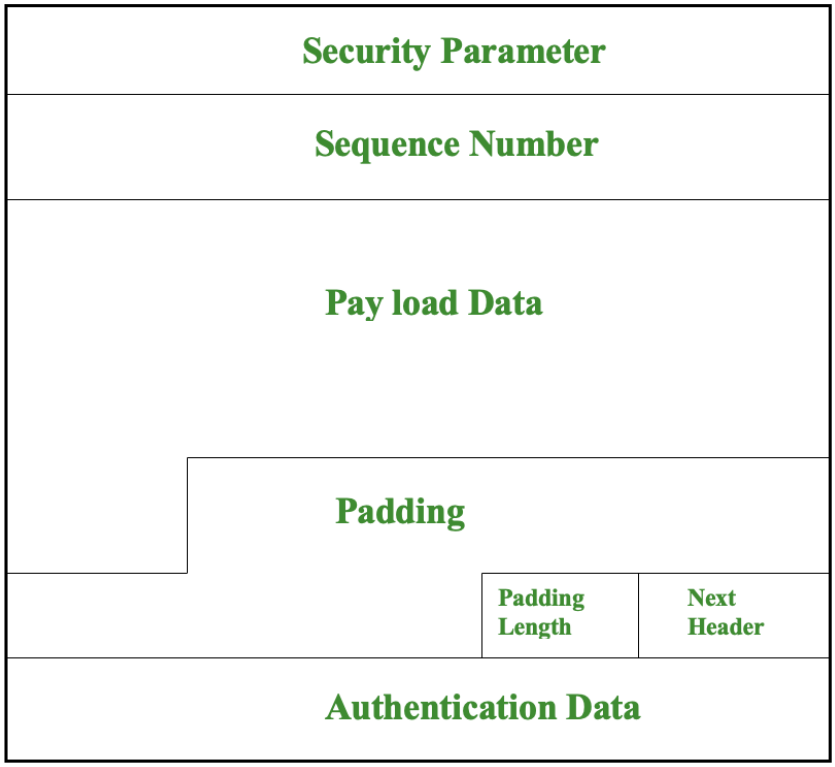
Below listed are the disadvantages of Encapsulating Security Payload:

1. There is a restriction on the encryption method to be used
2. For global use and implementation, weaker encryptions are mandatory to use

Components of ESP:

An important point to note is that authentication and security are not provided for the entire IP packet in transport mode. On the other hand for the tunnel mode, the entire IP packet along with the new packet header is encapsulated.

ESP structure is composed of the following parts as shown below :



ESP Structure

The diagrammatic representation of ESP has the below-mentioned components :

1. Security Parameter :

- Security parameters are assigned a size of 32 bits for use
- Security Parameter is mandatory to security parameter in ESP for security links and associations

2. Sequence Number:

- The sequence number is 32 bits in size and works as an incremental counter.
- The first packet has a sequence number 1 assigned to it whenever sent through SA

3. Payload Data:

- Payload data don’t have fixed size and are variable in size to use
- It refers to the data/ content that is provided security by the method of encryption

4. Padding:

- Padding has an assigned size of 0-255 bytes assigned to it.

- Padding is done to ensure that the payload data which needs to be sent securely fits into the cipher block correctly, so for this padding payloads come to the rescue.

5. Pad Length:

- Pad Length is assigned the size of 8 bits to use
- It is a measure of pad bytes that are preceding

6. Next Header:

- The next header is associated with a size of 8 bits to use
- It is responsible for determining the data type of payload by studying the first header of the payload

7. Authentication Data:

- The size associated with authentication data is variable and never fixed for use-case
- Authentication data is an optional field that is applicable only when SA is selected. It serves the purpose of providing integrity

KEY MANAGEMENT

The key management portion of IPSec involves the determination and distribution of secret keys. The IPSec Architecture document mandates support for two types of key management: • Manual: A system administrator manually configures each system with its own keys and with the keys of other communicating systems. This is practical for small, relatively static environments. • Automated: An automated system enables the on-demand creation of keys for SAs and facilitates the use of keys in a large distributed system with an evolving configuration. The default automated key management protocol for IPSec is referred to as ISAKMP/Oakley and consists of the following elements:

• Oakley Key Determination Protocol: Oakley is a key exchange protocol based on the Diffie-Hellman algorithm but providing added security. Oakley is generic in that it does not dictate specific formats. • Internet Security Association and Key Management Protocol (ISAKMP): ISAKMP provides a framework for Internet key management and provides the specific protocol support, including formats, for negotiation of security attributes.

Oakley Key Determination Protocol Oakley is a refinement of the Diffie-Hellman key exchange algorithm. The DiffieHellman algorithm has two attractive features: • Secret keys are created only when needed. There is no need to store secret keys for a long period of time, exposing them to increased vulnerability. •

The exchange requires no pre-existing infrastructure other than an agreement on the global parameters. However, Diffie-Hellman has got some weaknesses: • No identity information about the parties is provided. • It is possible for a man-in-the-middle attack • It is computationally intensive. As a result, it is vulnerable to a clogging attack, in which an opponent requests a high number of keys. Oakley is designed to retain the advantages of Diffie-Hellman while countering its weaknesses.

Features of Oakley The Oakley algorithm is characterized by five important features:

1. It employs a mechanism known as cookies to thwart clogging attacks.

2. It enables the two parties to negotiate a group; this, in essence, specifies the global parameters of the Diffie-Hellman key exchange. 107

3. It uses nonces to ensure against replay attacks.

4. It enables the exchange of Diffie-Hellman public key values.

5. It authenticates the Diffie-Hellman exchange to thwart man-in-the-middle attacks.

Here are some notes on combining security associations, key management, and network security:

- **Security associations**

A security association (SA) is a set of information that ensures secure communication between two entities in a network. An SA includes an SA Identifier (SAID) and is applied to service flows.

- **Internet Security Association and Key Management Protocol (ISAKMP)**

ISAKMP is a protocol that defines how to create, manage, and authenticate security associations, generate keys, and mitigate threats. ISAKMP SAs are bidirectional, meaning both the initiator and responder can initiate a phase 2 negotiation.

- **Combining security associations**

The IPsec Architecture document lists four examples of SA combinations that must be supported by compliant IPsec hosts and security gateways.

- **Polyalphabetic ciphers**

A polyalphabetic cipher is a substitution cipher that uses more than one alphabet to make the cipher more secure. This makes it more difficult to crack the code using frequency analysis.

- **Oakley Key Determination Protocol**

The Oakley Key Determination Protocol is a key-agreement protocol that allows authenticated parties to exchange keying material across an insecure connection.

In clogging attacks, an opponent forges the source address of a legitimate user and sends a public Diffie-Hellman key to the victim. The victim then performs a modular exponentiation to compute the secret key. Repeated messages of this type can clog the victim's system with useless work. The cookie exchange requires that each side send a pseudorandom number, the cookie, in the initial message, which the other side acknowledges. This acknowledgment must be repeated in the first message of the Diffie- Hellman key exchange. The recommended method for creating the cookie is to perform a fast hash (e.g., MD5) over the IP Source and Destination addresses, the UDP Source and Destination ports, and a locally generated secret value. Oakley supports the use of different groups for the Diffie-Hellman key exchange. Each group includes the definition of the two global parameters and the identity of the algorithm. Oakley employs nonces to ensure against replay attacks. Each nonce is a locally generated pseudorandom number. Nonces appear in responses and are encrypted during certain portions of the exchange to secure their use

Types of Key Management

There are two aspects of Key Management:

1. Distribution of public keys.
2. Use of public-key encryption to distribute secrets.

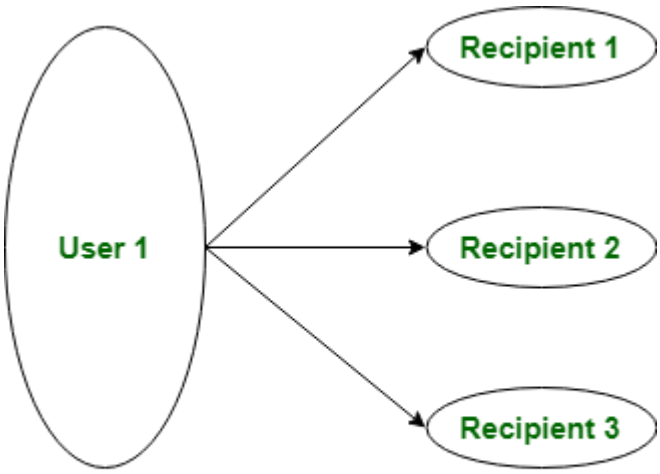
Distribution of Public Key

The public key can be distributed in four ways:

1. Public announcement
2. Publicly available directory
3. Public-key authority
4. Public-key certificates.

These are explained as following below:

1. Public Announcement: Here the public key is broadcast to everyone. The major weakness of this method is a forgery. Anyone can create a key claiming to be someone else and broadcast it. Until forgery is discovered can masquerade as claimed user.



Public Key Announcement

2. Publicly Available Directory: In this type, the public key is stored in a public directory. Directories are trusted here, with properties like Participant Registration, access and allow to modify values at any time, contains entries like {name, public-key}. Directories can be accessed electronically still vulnerable to forgery or tampering.

3. Public Key Authority: It is similar to the directory but, improves security by tightening control over the distribution of keys from the directory. It requires users to know the public key for the directory. Whenever the keys are needed, real-time access to the directory is made by the user to obtain any desired public key securely.

4. Public Certification: This time authority provides a certificate (which binds an identity to the public key) to allow key exchange without real-time access to the public authority each time. The certificate is accompanied by some other info such as period of validity, rights of use, etc. All of this content is signed by the private key of the certificate authority and it can be verified by anyone possessing the authority's public key.
First sender and receiver both request CA for a certificate which contains a public key and other information and then they can exchange these certificates and can start communication.

Key Management Lifecycle

The **key management lifecycle** outlines the stages through which cryptographic keys are generated, used, and eventually retired or destroyed. Proper management of these keys is critical to ensuring the security of cryptographic systems. Here's an overview of each stage:

1. Key Generation:

- **Creation**: Keys are created using secure algorithms to ensure randomness and strength.

- **Initialization**: Keys are initialized with specific parameters required for their intended use (e.g., length, algorithm).

2. Key Distribution:

- **Sharing**: For symmetric keys, secure methods must be used to share the key between parties.
- **Publication**: For asymmetric keys, the public key is shared openly, while the private key remains confidential.

3. Key Storage:

- **Protection**: Keys must be stored securely, typically in hardware security modules (HSMs) or encrypted key stores, to prevent unauthorized access.
- **Access Control**: Only authorized users or systems should be able to access keys.

4. Key Usage:

- **Application**: Keys are used for their intended cryptographic functions, such as encrypting/decrypting data or signing/verifying messages.
- **Monitoring**: Usage is monitored to detect any unusual or unauthorized activities.

Key Management in Cryptography

5. Key Rotation:

- **Updating**: Keys are periodically updated to reduce the risk of exposure or compromise.
- **Re-Keying**: New keys are generated and distributed, replacing old ones while ensuring continuity of service.

6. Key Revocation:

- **Invalidation**: Keys that are no longer secure or needed are invalidated.
- **Revocation Notices**: For public keys, revocation certificates or notices are distributed to inform others that the key should no longer be trusted.

7. Key Archival:

- **Storage**: Old keys are securely archived for future reference or compliance purposes.
- **Access Restrictions**: Archived keys are kept in a secure location with restricted access.

8. Key Destruction:

- **Erasure**: When keys are no longer needed, they are securely destroyed to prevent any possibility of recovery.
- **Verification**: The destruction process is verified to ensure that no copies remain.

Conclusion

Managing cryptographic keys is crucial for keeping data secure. It involves creating, distributing, storing, using, updating, and eventually destroying keys properly. Good key management ensures that keys are safe from unauthorized access and can be trusted throughout their life. By doing this, organizations protect sensitive information and maintain the security of their digital communications. In short, effective key management is essential for making encryption work and keeping information systems secure.