Government polytechnic Bagepalli

TRANSPORT LAYER SECURITY

Name : Nitesh.p

Reg no:136CS21037

Program:3rd year ,cse

Course: **cyber security**

Definition of TLS ?

- Transport Layer Security, or TLS, is a widely adopted security protocol designed to facilitate privacy and data security for communications over the Internet.
- TLS is a security protocol that provides privacy and data integrity for Internet communications
- TLS was derived from a security protocol called secure sockets layer(SSL)
- Tls is a protocol that ensures privacy between communicating application and their users on the internet.

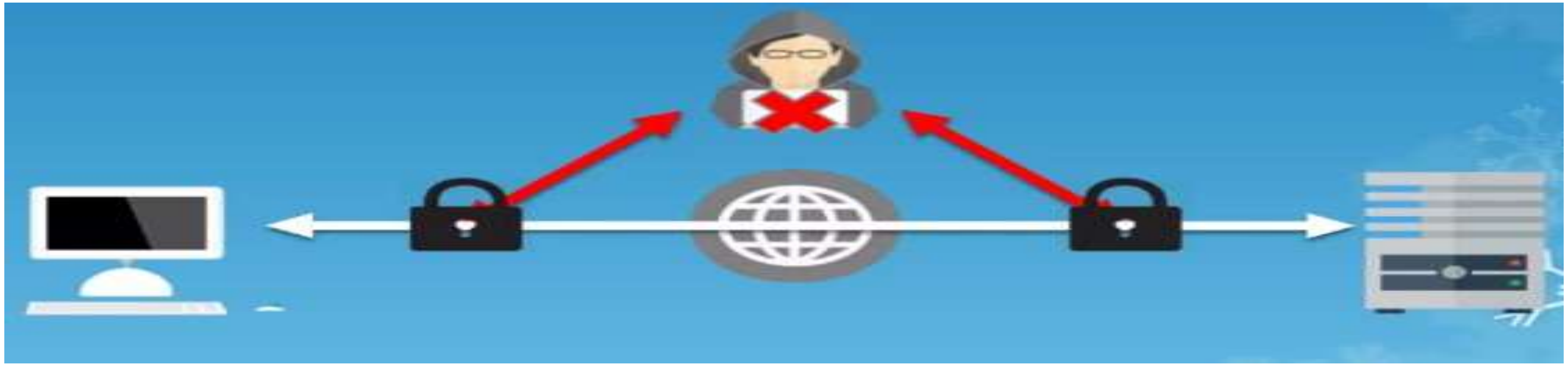
Importance of protocol

- **TLS (Transport Layer Security) protocol plays a crucial role in securing communication over the internet. Here are several key reasons why TLS is important**

- **Confidentiality:** TLS ensures that data transmitted between a client and a server remains private and cannot be intercepted or eavesdropped on by malicious actors. This is vital for protecting sensitive information such as passwords, financial details, and personal messages.
- **Data Integrity**: TLS uses cryptographic hash functions to verify that data has not been tampered with during transit. This ensures that the information received is exactly the same as what was sent, preventing unauthorized modifications.
- **Authentication**: TLS allows servers to prove their identity to clients using digital certificates. This helps in preventing impersonation and ensures that users are interacting with legitimate and trusted servers.
- **Protection Against Man-in-the-Middle Attacks**: Without TLS, attackers could potentially intercept and manipulate data in transit. TLS encryption safeguards against this, ensuring that the content remains confidential and unaltered.

Why do we need TLS ?

- TLS ensures that no third party may eavesdrop or tamper with any message.
- TLS encrypts data sent over the internet to ensure that hackers are unable to see what you transmit or transfer your private information.



Components of TLS with explanation

- **Handshake Protocol**:The TLS handshake protocol is the initial step in establishing a secure connection between a client and a server. It involves a series of messages exchanged between the two parties to negotiate encryption parameters and establish cryptographic keys.
- **Key Exchange**:During the handshake, the client and server negotiate and agree upon a shared secret key. This key is crucial for encrypting and decrypting data during the session.
- **Authentication and Digital Certificates**:TLS uses digital certificates to authenticate the identity of servers. These certificates are issued by trusted Certificate Authorities (CAs) and serve as a means of verifying the legitimacy of the server.
- **Encryption and Data Privacy**:Once the handshake is complete, actual data transmission begins. The data exchanged between the client and server is encrypted, ensuring that it is indecipherable to anyone intercepting it.
- **Data Integrity**:TLS employs cryptographic hash functions to verify that the transmitted data has not been tampered with during transit. This ensures the integrity

Types of TLS ?

- **Domain Validation:** A domain validated certificate (DV) is an X. 509 public key certificate typically used for Transport Layer Security (TLS) where the domain name of the applicant is validated by proving some control over a DNS domain.
- **Organization Validation**: Organization Validation: A higher assurance TLS/SSL that strengthens security by verifying the organization's information in the certificate against an independent CA; this tells visitors to the site that it is associated with a real organization, which deters fraudulent activities such as phishing**.**
- **Extended Validation** :Extended Validation (EV) is an advanced level of SSL/TLS certificate that requires a thorough validation process by a Certificate Authority (CA). This process verifies the legal and physical existence of the entity requesting the certificate

History of the protocol

SSL 1.0

– Internal Netscape design, 1994

– Not publicly released

- **SSL 2.0**
- – Published by Netscape, 1995

– Several weaknesses

- **SSL 3.0**

– Designed by Netscape and Paul Kocher, 1996

- **TLS 1.0**

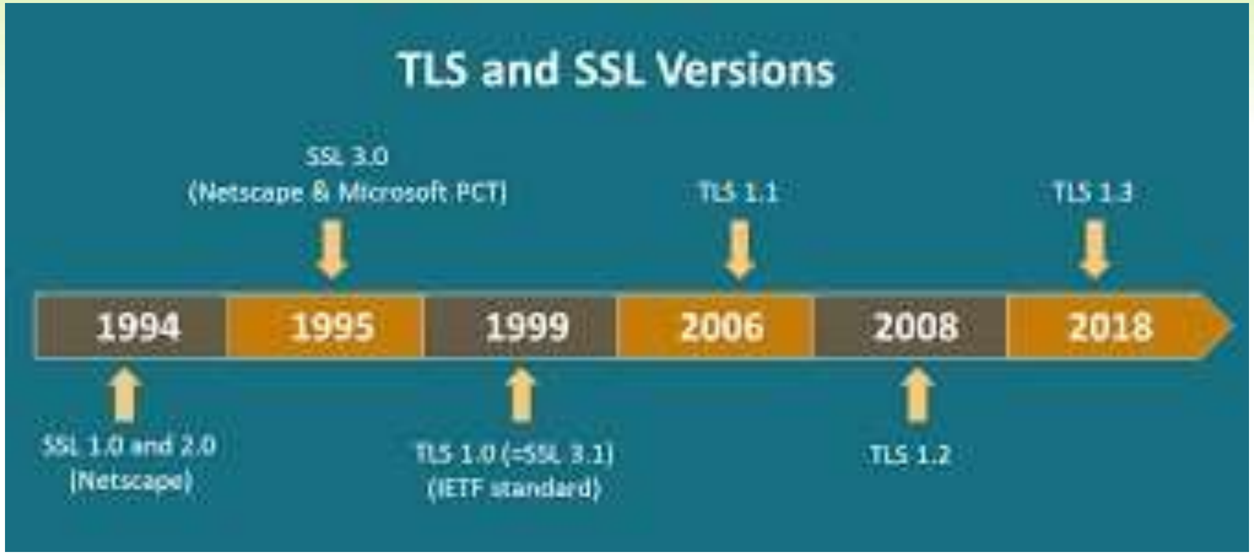
– IETF makes RFC 2246 based on SSL 3.0, 1999

– Not interoperable with SSL 3.0

TLS uses HMAC instead of MAC; can run on any port

History of TLS

- **TLS 1.1, 2006**
  - RFC 4346
  - Protection against cipher-block chaining (CBC) attacks
- **TLS 1.2, 2008**
  - RFC 5246
  - More options in cipher suite
- **TLS 1.3, 2018**
  - Published as RFC 8446
  - Some insecure ciphers removed (RC4, DES,...)
  - streamline RTT handshakes (e.g. 0-RTT mode)



TLS Handshake protocol

Two parties: client and server

- Negotiate version of the protocol and the set of

  cryptographic algorithms to be used
- – Interoperability between different implementations of

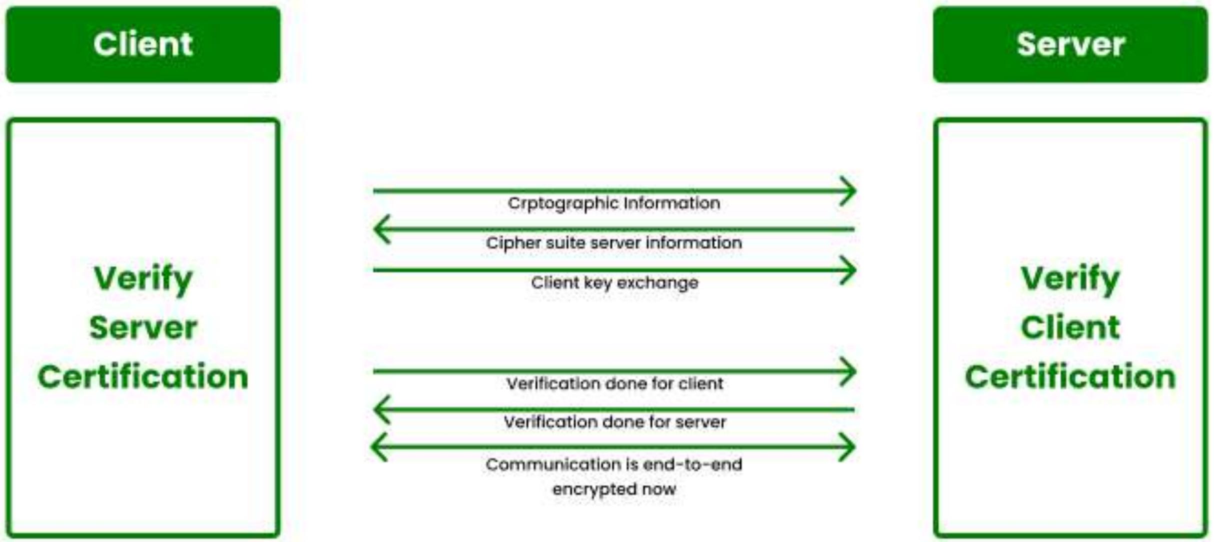
  the protocol

Authenticate server and client (optional)

- – Use digital certificates to learn each other’s public

  keys and verify each other’s identity
- Use public keys to establish a shared secret

  Symmetric key is generated from the secret
- The following is based on TLS 1.1 & 1.2


Client
Server
Verify
Server
Certification
Crptographic Information
Cipher suite server information
Client key exchange
Verification done for client
Verification done for server
Communication is end-to-end encrypted now
Verify
Client
Certification

Encryption and decryption

- **Encryption** is the process by which a readable message is converted to an unreadable form to prevent unauthorized parties from reading it.
  - It encrypts communications between a client and server ,primarily web browers and web sites/appications.
  - Once the session keys are established, they are used for encrypting the data being transmitted. TLS uses symmetric-key cryptography for actual data encryption. This means that the same key is used for both encryption and decryption.
- **Decryption** is the process of converting an encrypted message back to its original (readable) format
  - On the receiving end (server side), the encrypted data is received. The server uses the session keys (which were established during the handshake) to decrypt the data.

Difference's between SSL&TLS

SSL:

- SECURE SOCKE LAYER(SSL)
- First version developed by Netscape in 1995
- It uses explicit connections to setup secure channel
- SSL is obsolete(all versions),no more recommended for use

TLS:

- Transport Layer security(TLS)
- First version by IETF in 1999
- TLS begins its connection via protocol known as implicit connection
- TLS 1.3 is the latest version – faster and secure

THANK YOU..!!