Secure Socket Layer

SSL provides **security** to the data transferred between web browser and the server. It **encrypts the link** to ensure that the data transferred remains **private** and **free from attack**.

SSL protocols

- SSL record protocol
- Handshake protocol
- Change-cipher spec protocol
- Alert protocol

SSL Record protocol

Provides

- Confidentiality
- Message Integrity

Steps involved

1. Data is divided into **fragments**.
2. Fragments are **compressed**.
3. Message Authentication Code(**MAC**) generated by algorithms like **SHA**(Secure Hash Protocol)and **MD5**(Message Digest) is appended.
4. Data is **encrypted**.
5. **SSL header** is appended to the data.

Handshake Protocol

It allows the client and server to **authenticate** each other by sending a series of messages to each other.

Phase-1:

- Both Client and Server send **hello-packets** to each other.
- IP session, cipher suite and protocol version are exchanged for security purpose.

Phase-2:

- Server send his **certificate** and **Server-key-exchange**.
- Server end the phase-2 by sending **Server-hello-end packet**.

Phase-3:

- Client replies to the server by sending **certificate** and **Client-exchange-key**.

Phase-4:

- Change-cipher suite occurs and after this Handshake Protocol ends.

Change-cipher Protocol

- Unless Handshake Protocol is completed, the SSL record Output will be in **pending state**.
- After handshake protocol the Pending state is converted into **Current state**.

- Change-cipher protocol consists of **single message** which is 1 byte in length with only **one value**.

Purpose: To cause pending state to be copied into current state.

Alert protocol

This protocol is used to convey SSL-related **alerts** to the peer entity. Each message in this protocol contain 2 bytes.

- **Warning**
  This Alert have no impact on the connection between sender and receiver.
- **Fatal Error**
  This Alert **breaks** the connection between sender and receiver.

Features of SSL

- The service can be customized to the specific needs of the given application.
- Secure Socket Layer was originated by **Netscape**.
- SSL is designed to make use of TCP to provide **reliable** end-to-end secure service.
- This is two-layered protocol.

How SSL works?

It involves

- Asymmetric Cryptography
- Symmetric Cryptography

Asymmetric Cryptography

Asymmetric cryptography uses mathematically-related key pair to encrypt and decrypt data.

The **key pair** has

- **Public Key**
  It is shared with anyone who is interested in a communication.
- **Private Key**
  It is kept secret.

Here, the keys referred to a **mathematical value** and were created using a mathematical algorithm which encrypts or decrypts the data.

SSL uses asymmetric cryptography to initiate the communication which is known as **SSL handshake.**

Symmetric Cryptography

In the symmetric cryptography, there is **only one key** which encrypts and decrypts the data. Both sender and receiver should have this key, which is only known to them.

SSL uses symmetric cryptography using the session key after the initial handshake is done.

Data transfer in SSL involves

- Asymmetric Cryptography
  SSL handshake
- Symmetric Cryptography
  Data transfer

SSL Communication

SSL Handshake

The SSL handshake is an asymmetric cryptography which allows the browser to **verify the web server**, get the public key and establish a **secure connection** before the beginning of the actual data transfer.

Steps involved

- **Client Hello Message**
  This includes the client's SSL version number, cipher settings, session-specific data and other information that the server needs to communicate with the client using SSL.

- **Server Hello message**
  The server responds with a "server hello" message. This includes the server's SSL version number, cipher settings, session-specific data and an SSL certificate with a public key and other information that the client needs to communicate with the server over SSL.

- **Authentication of server**
  The client **verifies** the server's SSL certificate and authenticates the server. If the authentication fails, then the client refuses the SSL connection and throws an exception. If the authentication succeeds, then proceed to next step.

- **Client creates and sends Encrypted session key**
  The client creates a **session key**, encrypts it with the server's public key and sends it to the server.

- **Server sends acknowledgement**
  The server decrypts the session key with its private key and sends the acknowledgement to the client encrypted with the session key.

Thus, at the end of the SSL handshake, both the client and the server have a **valid session key** which they will use to encrypt or decrypt actual data.

Actual Data Transfer

The client and the server now use a shared session key to encrypt and decrypt actual data and transfer it.

This is done using the same session key at both ends and so, it is a **symmetric cryptography.**

SSL Certificate

The SSL certificate is a data file issued by the authorised Certificate Authority (CA). The SSL certificate contains the **owner's public key** and other details. The web server sends a public key to the browser through an **SSL certificate** and the browser verifies it and authenticates the web server using the SSL certificate.

Secure Socket Layer(SSL)	**Transport Layer Security(TLS)**
Message digest is used to create master secret.	Pseudo-random function is used to create master secret.
Message Authentication Code protocol is used.	Hashed Message Authentication Code protocol is used.
SSL is complex than TLS.	TLS is simple.

IPsec	**SSL**
Internet protocol security (IPsec) is a set of protocols that provide security for Internet Protocol.	SSL is a secure protocol developed for sending information securely over the Internet.
It Work in Internet Layer of the OSI model.	It Work in Between the transport layer and application layer of the OSI model.
Configuration of IPsec is Complex	Configuration of SSL is Comparatively Simple
IPsec is used to secure a Virtual Private Network.	SSL is used to secure web transactions.
IPsec resides in operating system space.	SSL resides in user space