- Encrypting files
- Decrypting files
- Smart card authentication

SSL certificate: To create this secure connection, an **SSL certificate** (also referred to as a "digital certificate") is installed on a web server and serves two functions:

- It authenticates the identity of the website (this guarantees visitors that they're not on a bogus site)
- It encrypts the data that's being transmitted

There are many different types of SSL certificates based on the number of domain names or subdomains owned, such as:

- **Single** – secures one fully-qualified domain name or subdomain name
- **Wildcard** - covers one domain name and an unlimited number of its subdomains
- **Multi-Domain** – secures multiple domain names and the level of validation needed, such as:
  - **Domain Validation** – This level is the least expensive, and covers basic encryption and verification of the ownership of the domain name registration. This type of certificate usually takes a few minutes to several hours to receive.
  - **Organization Validation** – In addition to basic encryption and verification of ownership of the domain name registration, certain details of the owner (e.g., name and address) are authenticated. This type of certificate usually takes a few hours to several days to receive.
  - **Extended Validation (EV)** – This provides the highest degree of security because of the thorough examination that is conducted before this certificate is issued (and as strictly specified in guidelines set by the SSL certification industry's governing consortium). In addition to ownership of the domain name registration and entity authentication, the legal, physical and operational existence of the entity is verified. This type of certificate usually takes a few days to several weeks to receive.

Components Of Public Key Infrastructure: There are three key components: digital certificates, certificate authority, and registration authority.

1. **Digital Certificates:** A digital certificate is like a driver's license—it's a form of electronic identification for websites and organizations. Secure connections between two communicating machines are made available through PKI because the identities of the two parties can be verified by way of certificates.So how do devices get these certificates? You can create your own certificates for internal communications. If you would like certificates for a commercial site or something of a larger scale, you can obtain a PKI digital certificate through a trusted third party issuer, called a Certificate Authority.

2. **Certificate Authority**: A Certificate Authority (CA) is used to authenticate the digital identities of the users, which can range from individuals to computer systems to servers. Certificate Authorities prevent falsified entities and manage the life cycle of any given number of digital certificates within the system.Much like the state government issuing you a license, certificate authorities vet the organizations seeking certificates and issue one based on their findings. Just as someone trusts the validity of your license based on the authority of the government, devices trust digital certificates based on the authority of the issuing certificate authorities. This process is similar to how code signing works to verify programs and downloads.

3. **Registration Authority**: Registration Authority (RA), which is authorized by the Certificate Authority to provide digital certificates to users on a case-by-case basis. All of the certificates that are requested, received, and revoked by both the Certificate Authority and the Registration Authority are stored in an encrypted certificate database. Certificate history and information is also kept on what is called a certificate store, which is usually grounded on a specific computer and acts as a storage space for all memory relevant to the certificate history, including issued certificates and private encryption keys. Google Wallet is a great example of this.

PKI Security:PKI is best utilized for situations that require digital security, which is where encryption plays a vital role. PKI performs encryption directly through the keys that it generates. It works by using two different cryptographic keys: a public key and a private key. Whether these keys are public or private, they encrypt and decrypt secure data.

By using a two-key encryption system, PKI secures sensitive electronic information as it is passed back and forth between two parties, and provides each party with a key to encrypt and decrypt the digital data.

Popular Ways PKI Security Is Used

You might be thinking what PKI security might look like in your day to day. PKI security is used in many different ways. The main ways that PKI security can be used are:

authentication and access control also enables organizations to meet regulatory and privacy compliancy, as well as fulfill internal security policies using PKI-based two-factor authentication – 'something you have' (a Global Sign Digital Certificate) and 'something you know' (an internally managed password).

Benefits of client authentication

Client authentication has multiple benefits as an authentication method especially when compared to the basic username and password method:

- You can decide whether or not a user is required to enter a username and password
- Encrypts transactions over the network, identifies the server and validates any messages sent
- Validates the user identity using a trusted party (the Certificate Authority) and allows for centralized management of certificates which enables easy revocation
- Optional - you can configure the certificate so it cannot be exported to other devices, making it unique to the device it is installed on
- Restrict access by user, group, roles, or device based on Active Directory (using Global Sign's Auto Enrolment Gateway (AEG) solution)
- Serves more purposes than authentication such as integrity and confidentiality
- Prevents malicious attacks/problems, including but not limited to phishing, keystroke logging and man-in-the-middle (MITM) attacks

Public Key Infrastructure: PKI (or Public Key Infrastructure) is the framework of encryption and cyber security that protects communications between the server (your website) and the client (the users). PKI is essential in building a trusted and secure business environment by being able to verify and exchange data between various servers and users.
Through encryption and decryption, PKI is based on digital certificates that verify the identity of the machines and/or users that ultimately proves the integrity of the transaction. As the number of machines is increasing dramatically in today's digital age, it's important that our information is trusted and protected against attacks.

Components Of Public Key Infrastructure: There are three key components: digital certificates, certificate authority, and registration authority.

	In SSL(Secure Socket Layer), Message digest is used to create master secret.	In TLS(Transport Layer Security), Pseudo-random function is used to create master secret.
5.	In SSL(Secure Socket Layer), Message Authentication Code protocol is used.	In TLS(Transport Layer Security), Hashed Message Authentication Code protocol is used.
6.	SSL (Secure Socket Layer) is complex than TLS(Transport Layer Security).	TLS (Transport Layer Security) is simple.
7.	SSL (Secure Socket Layer) is less secured as compared to TLS(Transport Layer Security).	TLS (Transport Layer Security) provides high security.

Client Authentication: It is the process by which users **securely access a server or remote computer by exchanging a Digital Certificate**. The Digital Certificate is used to cryptographically bind a customer, employee, or partner's identity to a unique Digital Certificate (typically including the name, company name and location of the Digital Certificate owner). The Digital Certificate can then be mapped to a user account and used to provide access control to network resources, web services and websites.

Just as organizations need to control which individual users have access to corporate networks and resources, they also need to be able to identify and control which machines and servers have access. Implementing device authentication means only machines with the appropriate credentials can access, communicate, and operate on corporate networks.

The Digital Certificates used for client and device authentication may look the same as any other Digital Certificate that you may already be using within your organization, such as certificates for securing web services (SSL) or email/document signatures (digital signatures), but Digital Certificates are likely to have a few different properties depending on the use.

Client authentication can be used to prevent unauthorized access, or simply to add a second layer of security to your current username and password combination. Client

TLS/SSL works with most web browsers, including Microsoft Internet Explorer and on most operating systems and web servers.

- **Algorithm flexibility:** TLS/SSL provides operations for authentication mechanism, encryption algorithms and hashing algorithm that are used during the secure session.
- **Ease of Deployment:** Many applications TLS/SSL temporarily on a windows server 2003 operating systems.
- **Ease of Use:** Because we implement TLS/SSL beneath the application layer, most of its operations are completely invisible to client

Working of TLS:

The client connect to server (using TCP), the client will be something. The client sends number of specification: Version of SSL/TLS, which cipher suites, compression method it wants to use.

The server checks what the highest SSL/TLS version is that is supported by them both, picks a cipher suite from one of the clients option (if it supports one) and optionally picks a compression method. After this the basic setup is done, the server provides its certificate. This certificate must be trusted either by the client itself or a party that the client trusts. Having verified the certificate and being certain this server really is who he claims to be (and not a man in the middle), a key is exchanged. This can be a public key, "PreMasterSecret" or simply nothing depending upon cipher suite.

Both the server and client can now compute the key for symmetric encryption. The handshake is finished and the two hosts can communicate securely. To close a connection by finishing. TCP connection both sides will know the connection was improperly terminated. The connection cannot be compromised by this through, merely interrupted.

Difference between Secure Socket Layer (SSL) and Transport Layer Security (TLS)

S.NO	SSL	TLS
1.	SSL stands for Secure Socket Layer.	TLS stands for Transport Layer Security.
2.	SSL (Secure Socket Layer) supports **Fortezza** algorithm.	TLS (Transport Layer Security) does not Supports **Fortezza** algorithm.
3.	SSL (Secure Socket Layer) is the 3.0 version.	TLS (Transport Layer Security) is the 1.0 version.

- **Phase-2:** Server sends his certificate and Server-key-exchange. The server end phase-2 by sending the Server-hello-end packet.
- **Phase-3:** In this phase Client reply to the server by sending his certificate and Client-exchange-key.
- **.Phase-4:** In Phase-4 Change-cipher suite occurred and after this Handshake Protocol ends.

3. **Change-cipher Protocol:** This protocol uses the SSL record protocol. Unless Handshake Protocol is completed, the SSL record Output will be in a pending state. After handshake protocol, the Pending state is converted into the current state.

   Change-cipher protocol consists of a single message which is 1 byte in length and can have only one value. This protocol's purpose is to cause the pending state to be copied into the current state.

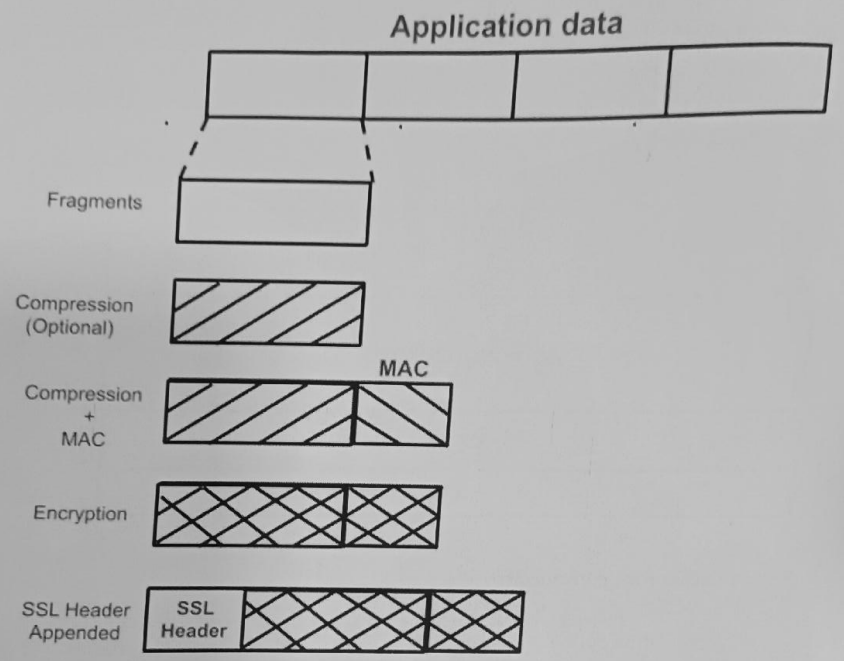
4. **Alert Protocol:**
   This protocol is used to convey SSL-related alerts to the peer entity. Each message in this protocol contain 2 bytes.

Silent Features of Secure Socket Layer:

- The advantage of this approach is that the service can be tailored 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 a two-layered protocol

Transport Layer Securities (TLS): It is designed to provide security at the transport layer. TLS was derived from a security protocol called Secure Socket Layer (SSL). TLS ensures that no third party may eavesdrop or tampers with any message. There are several benefits of TLS:

- **Encryption:**
  TLS/SSL can help to secure transmitted data using encryption.
- **Interoperability:**



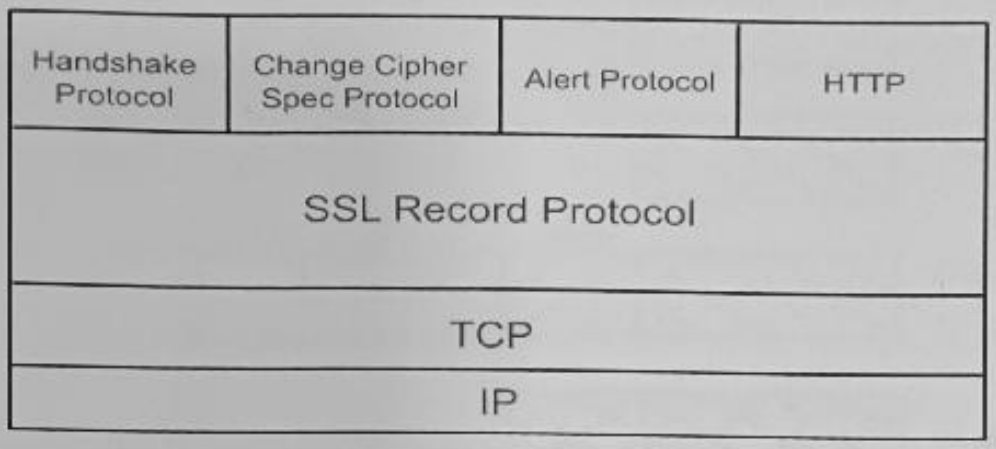
In the SSL Record Protocol application data is divided into fragments. The fragment is compressed and then encrypted MAC (Message Authentication Code) generated by algorithms like SHA (Secure Hash Protocol) and MD5 (Message Digest) is appended. After that encryption of the data is done and in last SSL header is appended to the data.

2. **Handshake Protocol:** 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:** In Phase-1 both Client and Server send hello-packets to each other. In this IP session, cipher suite and protocol version are exchanged for security purposes.

Secure Socket Layer (SSL): It provides security to the data that is transferred between web browser and server. SSL encrypts the link between a web server and a browser which ensures that all data passed between them remain private and free from attack.

SSL Protocol Stack:



Secure Socket Layer Protocols:

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

1. **SSL Record Protocol:** SSL Record provides two services to SSL connection.

- Confidentiality
- Message Integrity

SSL Attacks

(1) Downgrade attack – In SSLv2 there is no integrity protection for the initial handshake, where active attacker can remove strong crypto algorithm from proposed cipher suite by A, forcing A and B to agree on a weak cipher. It can be fixed by adding a finished message containing a hash of previous messages.

(2) Truncation attack – Without a finished message an attacker can send a TCP a FIN message and close the connection without communicating nodes detecting it.

Exportability Issues

Exportable suites in SSLv2 has
- 40 secret bits out of 128 in symmetric keys
- 512-bits RSA keys

Exportability suites in SSLv3
- Integrity key computed
- 40 bits secret Encryption keys
- IV non-secret
- Server creates an ephemeral key of 512-bits and signs it with 1024-bit key when communicates with external client.

Computing the keys

S: pre-master secret

$K = f(S, R_A, R_B)$

6 keys $= g_i(K, R_A, R_B)$

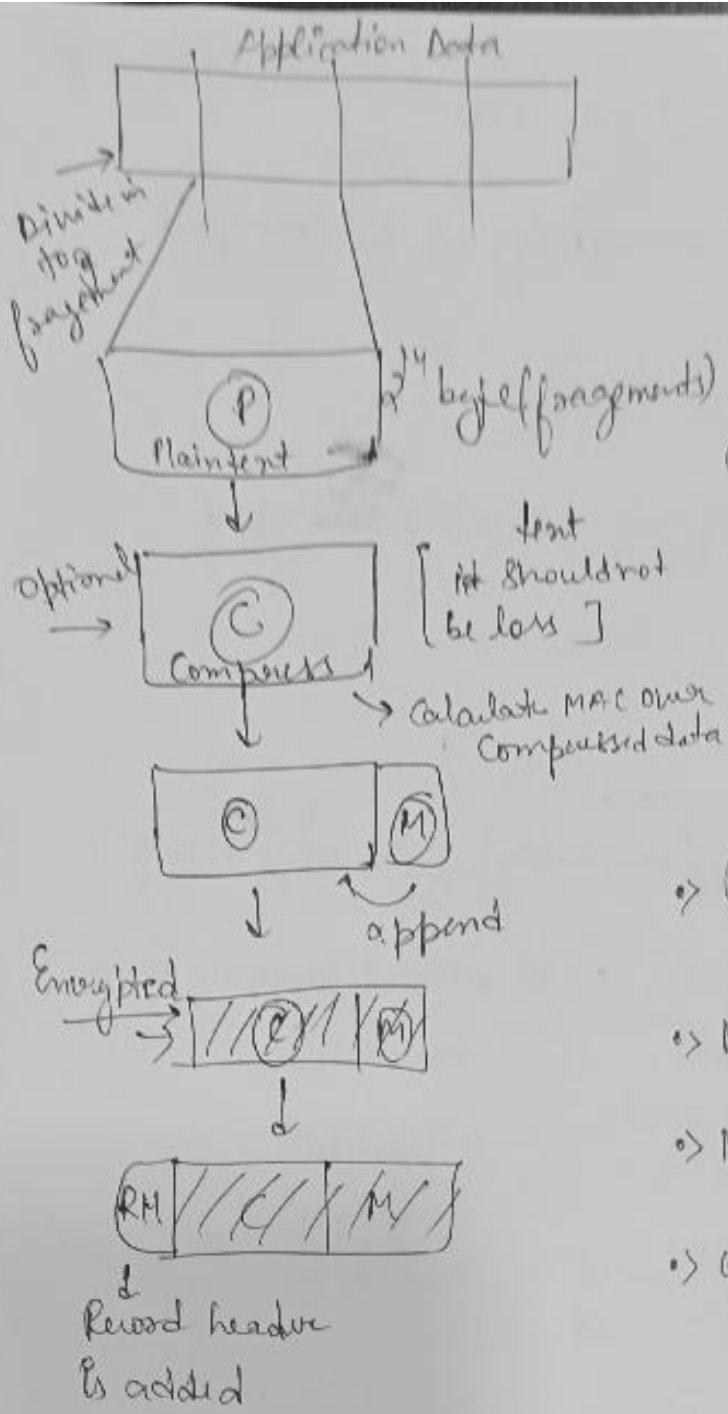
R_S : 32 bytes (usually the first 4 bytes are Unix time)

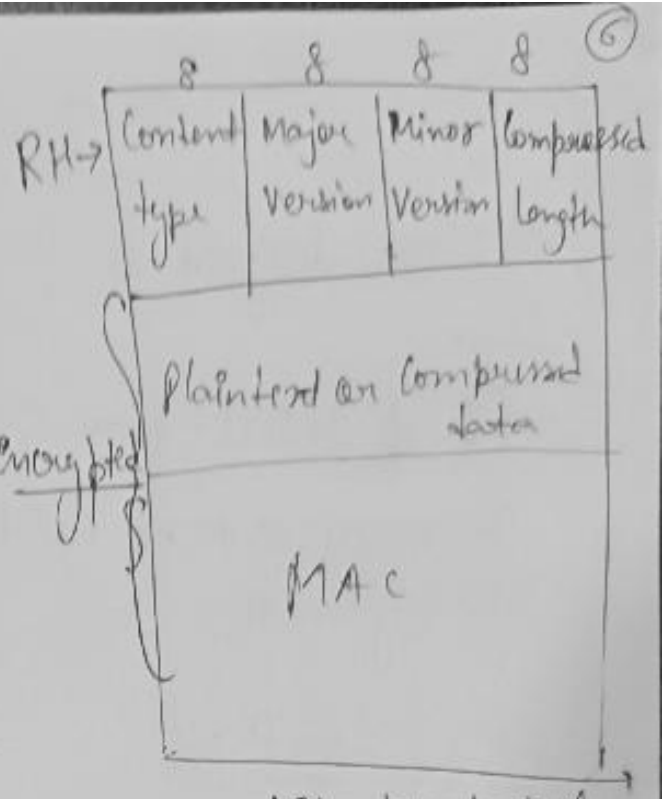
Cipher Suites

- Cipher suite is a complete package of encryption algorithm, key length, integrity checksum algorithm etc.
- Pre-defined, assigned a unique value (contrast wit IKE)
- → eg.→ SSL_RSA_EXPORT_WITH_DES40_CBC_SHA
  SSL2_RC4_128_WITH_MD5

PKI in SSL

When the server wishes to authenticate the client, server sends a list of CA it trusts and types of keys it can handle. A chain of certificates can be send.





- Content type :- Higher layer protocol used to process the enclosed fragment
- Major version :- For SSL 3 → value is 3
- Minor version :- Value is '0'
- Compressed length :- Length of compressed fragment in bytes.

SSL Record Protocol

SSL Record provides two services to SSL Connection

① Confidentiality →

- Using Symmetric encryption with a shared secret key defined by handshake protocol.
- IDEA, RC2-40, DES-40, 3DES.
- Message is compressed before encryption.

② Message Integrity

- Using a MAC with shared secret key.
- Based on HMAC & MD5.

→ In the SSL Record Protocol application data is divided into fragements. The fragement is compressed and then encrypted MAC generated by Algo like SHA (Secure Hash Protocol) & MD5 (message digest) is appended.

→ After that encryption of the data is done & in last SSL header is appended to the data.

Record header :-

[record type, version no., length]

ChangeCipherSpec = 20, Alert = 21, Handshake = 22, Application_data = 23

SSL Record Protocol

SSL Record provides two services to SSL Connection

① Confidentiality →

- Using Symmetric encryption with a shared secret key defined by handshake protocol.
- IDEA, RC2-40, DES-40, 3DES.
- Message is compressed before encryption.

② Message Integrity

- Using a MAC with shared secret key.
- Based on HMAC & MD5.

→ In the SSL Record Protocol application data is divided into fragements. The fragement is compressed and then encrypted MAC generated by Algo like SHA (Secure Hash Protocol) & MD5 (message digest) is appended.

→ After that encryption of the data is done & in last SSL header is appended to the data.

Record header :-

[record type, version no., length]

ChangeCipherSpec = 20, Alert = 21, Handshake = 22, Application_data = 23

(2) Change Cipher protocol :-

→ This protocol uses the SSL record protocol. Unless handshake protocol is completed, the SSL record output will be in pending state.

→ After handshake protocol, the pending state is converted into the current state.

→ Change cipher protocol consists of a single message which is 1 byte in length and can have only one value.

→ This protocol purpose is to cause the pending state to be copied into the current state.

B

(3) Alert Protocol :-

→ This protocol is used to convey SSL-related alert to the peer entity. Each message in this protocol contain 2 byte.

Level (1 byte)	Alert (1 byte)

The level is further classified into 2 parts :-

(i) Warning :- This alert has no impact on the connection b/w sender and receiver.

(ii) Fatal error :- This alert breaks the connection b/w sender & receiver

(2) Change Cipher protocol :-

→ This protocol uses the SSL record protocol. Unless handshake protocol is completed, the SSL record output will be in pending state.

→ After handshake protocol, the pending state is converted into the current state.

→ Change cipher protocol consists of a single message which is 1 byte in length and can have only one value.

→ This protocol purpose is to cause the pending state to be copied into the current state.

B

(3) Alert Protocol :-

→ This protocol is used to convey SSL-related alert to the peer entity. Each message in this protocol contain 2 byte.

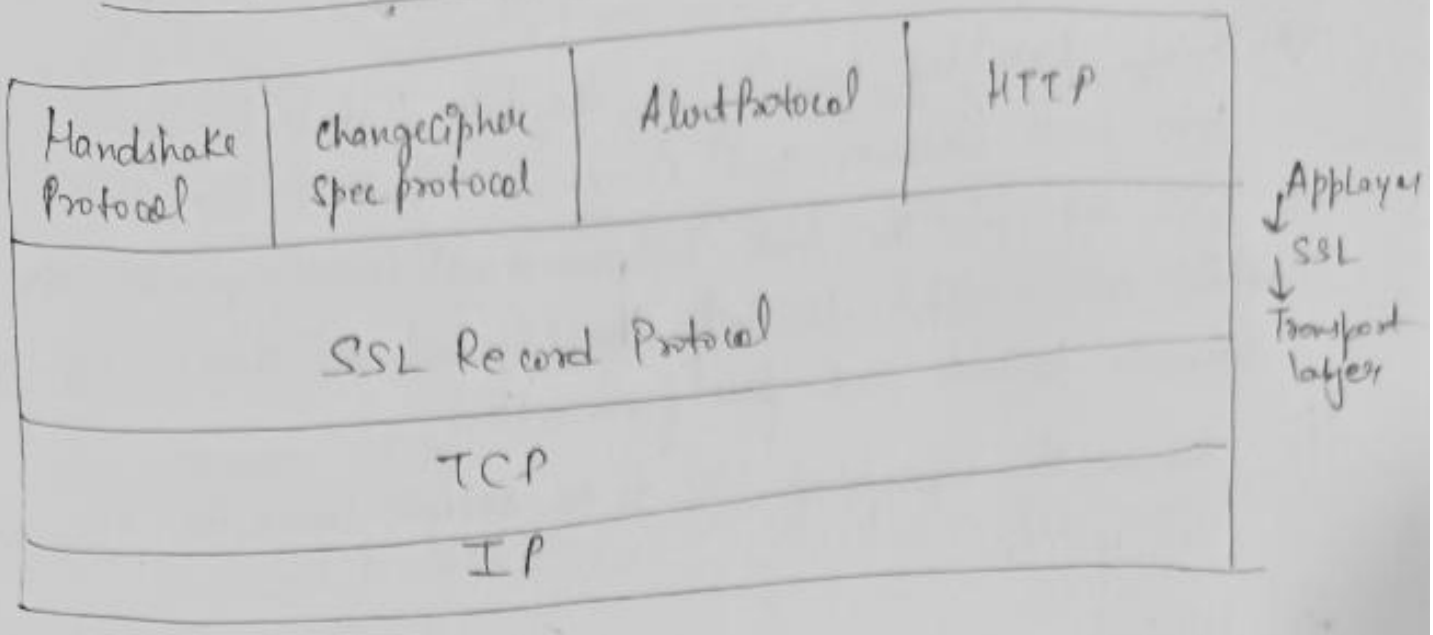
Level (1 byte)	Alert (1 byte)

The level is further classified into 2 parts :-

(i) Warning :- This alert has no impact on the connection b/w sender and receiver.

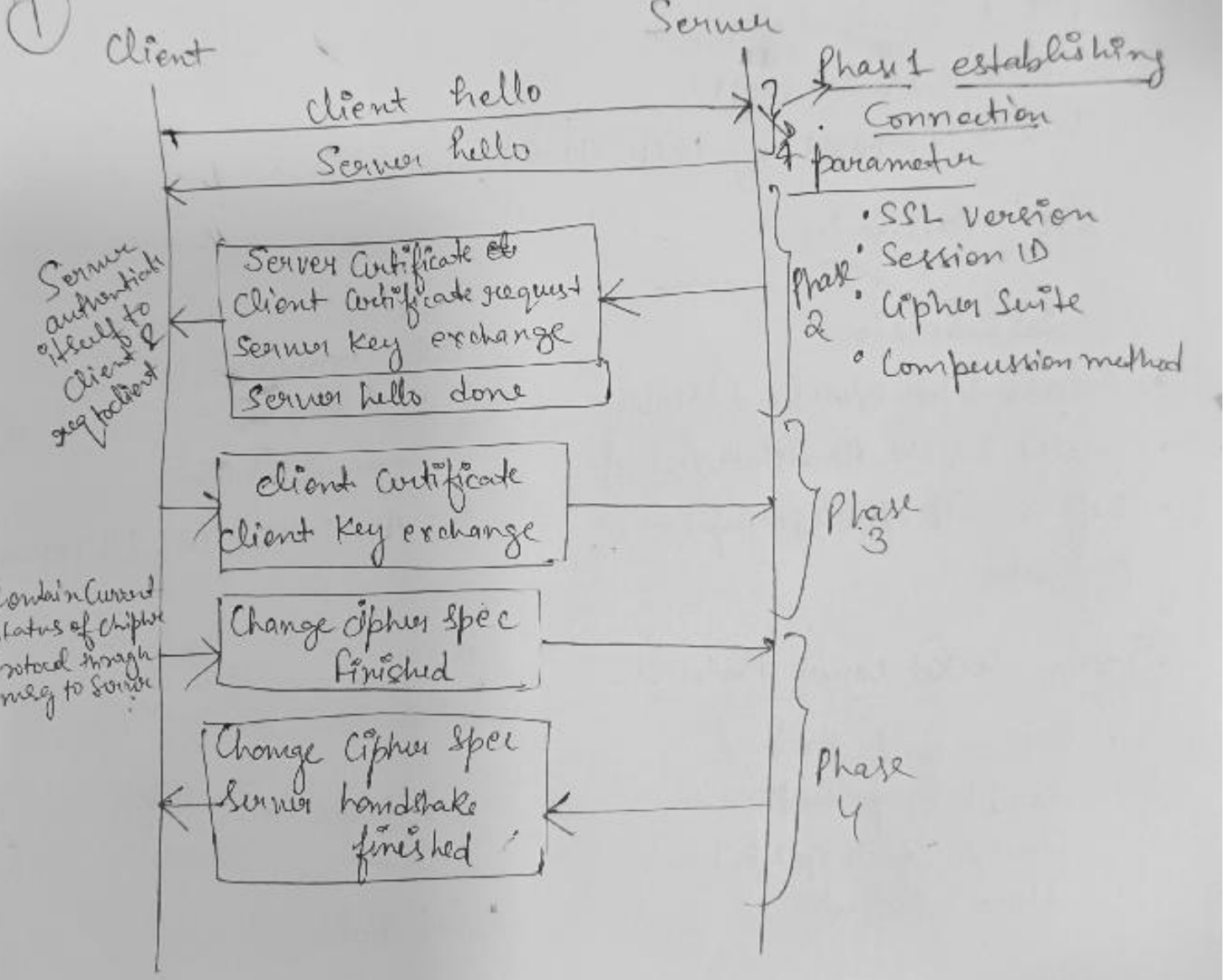
(ii) Fatal error :- This alert breaks the connection b/w sender & receiver

SSL Protocol Stack



1 SSL Handshake Protocol

① Provide Secure Connection b/w 2 entity
② 2 entity must authenticate each other



Secure Socket Layer (SSL)

⇒ SSL provides security to the data that is transfered b/w web browser and server.

⇒ SSL encrypts the link between a web server and a browser which ensure that all data passed between them remain private and free from attack.

⇒ Generally concept of SSL is to secure connection on top of TCP

⇒ History:-

SSL V2 → Netscape 1.1 (1995)
PCT → by microsoft
SSL V3 → used in (1995)
TLS → Proposed by IETF (1996)

SSL Architecture

SSL Session

- An association b/w client & server
- Created by the Handshake Protocol
- Defines a set of cryptographic parameter

SSL Connection

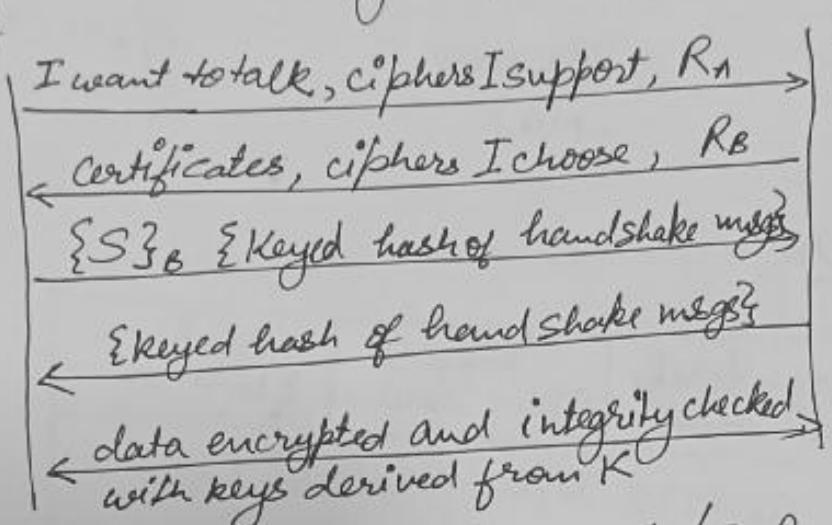
- A transient, peer to peer, communication link
- Associated with 1 SSL Session

Secure Socket Layer Protocol:-

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

SSL/TLS Basic Protocols

- SSL/TLS partitions TCP byte stream into records that has header, cryptographic protection
- provide reliable encrypted, integrity protected stream of octets
- Record type – userdata, handshake messages, Alerts, Change cipher spec

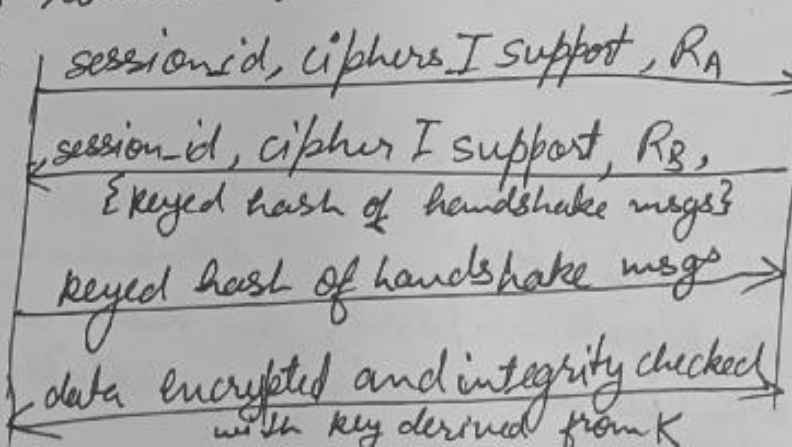


- Keyed hashes use $K = f(S, R_A, R_B)$
- SSL/TLS allows clients to authenticate using certificates
- Many secure connections can be derived from the session (Session Resumption)

- Session initiation: modify message 2

A ← session_id, certificate, cipher, R_B — B

- A & B remember (session id, master key)



Unit-4

Secure Socket Layer (SSL)

★ Where to put the Security in a protocol Stack ?

Application Layer :- https, mail: PGP
Transport Layer :- SSL/TLS
Network Layer :- IPSec, IKE
Link layer :- IEEE 802.1x / IEEE802.10
Physical layer :- Spread spectrum, quantum, crypto etc.

Difference between SSL & IPSec

→ SSL :- Avoid modifying "TCP stack" and require min. changes to the application and mostly used to authenticate servers.

→ It work on transport layer, safeguard sensitive data that is being sent b/w two system, preventing Criminals from reading and modifying any information transferred.

IPSec :- Transparent to the application and requires modification of the n/w stack.

→ Authenticates network nodes and establishes a secure channel between nodes.

→ Application still needs to authenticate the users.