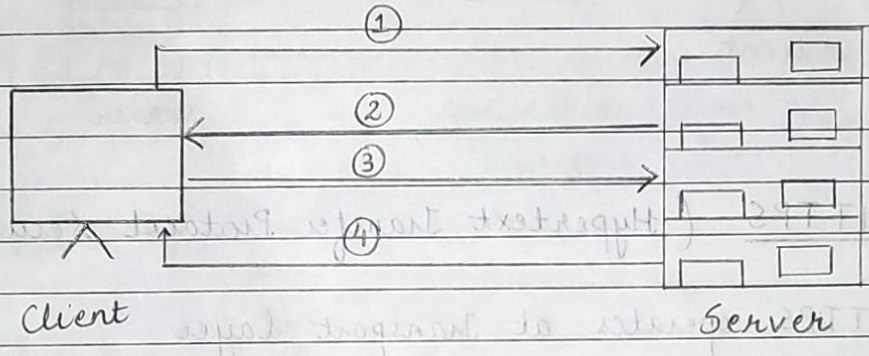
Page No.
Date

Unit - 4

Secure Socket Layer (SSL)

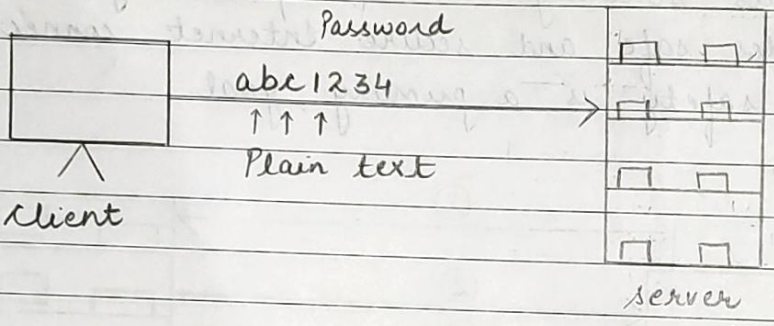
1. SSL was first developed by Netscape. Version 2 of SSL was released in 1995 and version 3 was released on 1999.
2. Internet Engineering Task Force (IETF) launched TLS (Transport layer Security) in 2015.
3. Internet Protocol for secure exchange of information between browser and server.
4. Provides security at transport layer.
5. Provides safe and secure Internet connection.
6. Data safety is a primary goal.



(1) The client sends a request to the server to set up a secure SSL session.
(2) The server responds by sending the certificate to the client.
(3) The client then generates its session key and sends it to the server.
(4) Secure SSL session is established.

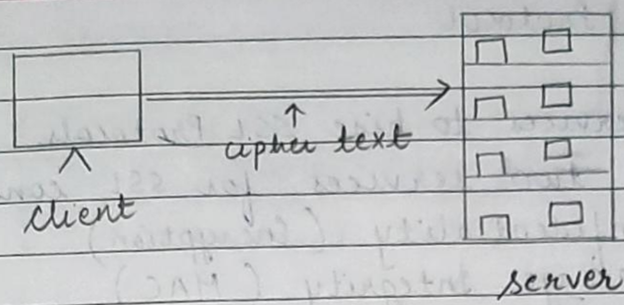
HTTP (Hypertext Transfer Protocol)

1. HTTP operates at Application layer inside TCP/IP
2. It lacks the security mechanism to encrypt the data.
3. Since there is no encryption of the data, it transfers data in the form of plain text.
4. It is good only for accessing the internet/ browser. It is an insecure method as no encryption methods / algorithms are used.



HTTPS (Hypertext Transfer Protocol Secure)

1. HTTPS operates at Transport layer.
2. HTTPS provides SSL to secure the communication between server and client.
3. It encrypts all the data and transfers data in the form of cipher text (encrypted text).
4. It is a combination of SSL protocol and HTTP.
5. Various web browsers and websites which need login credentials should use HTTPS protocol for sending the data.

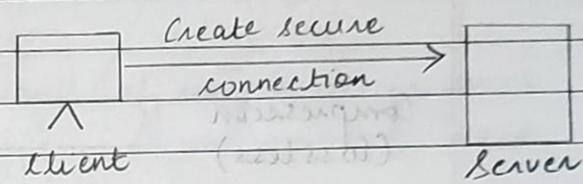


Position of SSL in TCP/IP

Application layer
SSL
Transport layer
Internet layer
Network interface

TCP/IP

⇒



SSL is located between the application layer and the Transport layer.

Goals of SSL

C - Confidentiality
I - Integrity
A - Authentication / Availability

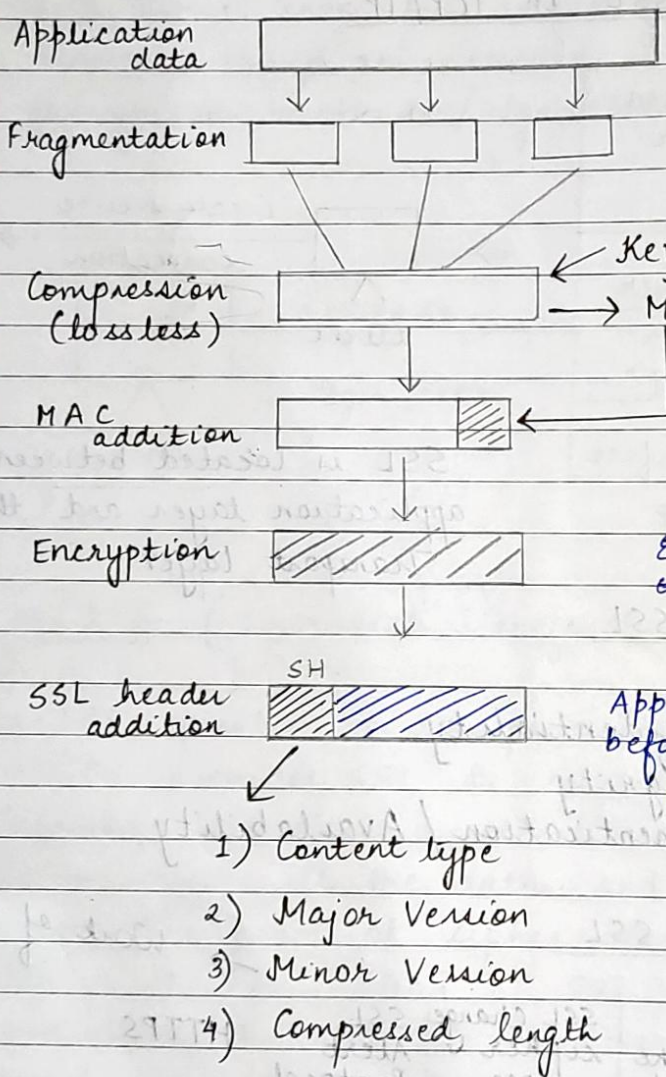
Working of SSL

→ Work of SSL

SSL handshake protocol	SSL Change cipher spec	SSL Alert Protocol	HTTPS
SSL Record Protocol			
TCP			
IP			

SSL Record Protocol

1. It offers services to hire SSL Protocols.
2. It provides two services for SSL connections.
   i) Confidentiality (Encryption)
   ii) Message Integrity (MAC)



* Backbone of SSL - Handshake protocol
Record protocol

Page No.
Date

SSL header

Content type	Major Version	Minor Version	Compressed length
Compressed data			
MAC			

SSL V3

Major version - 3

Minor version - 0

Compressed length - The length of compressed fragment.

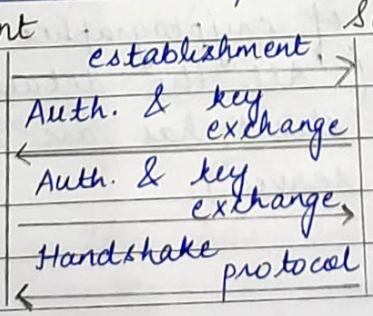
Content type - The higher layer protocol used to process the enclosed fragments.

SSL Handshake protocol

Most

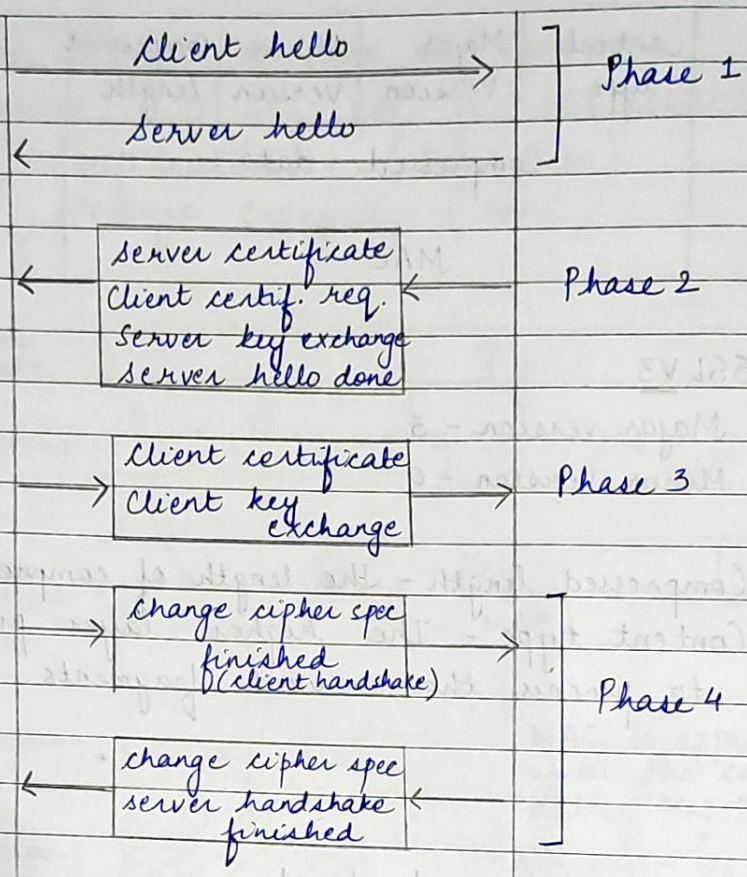
1. Complex protocol.
2. Establishment of secure connection between 2 entities.
3. Authentication between Client and Server.
4. Negotiation of encryption / MAC algorithm.
5. Exchange cryptographic keys.

Simple Handshake protocol



Page No.
Date

Complex Handshake Protocol



Phase 1 - 4 Parameters

1) SSL version (version 2 or 3)
2) Session id (The id which identifies the entire session)
3) Cipher suite (list of cryptographic algorithms)
4) Compression method (All the details that client has are sent to the server)

Handshake protocol uses four phases to complete its cycle -

> Phase 1 - In Phase 1, both client and server send hello-packets to each other.

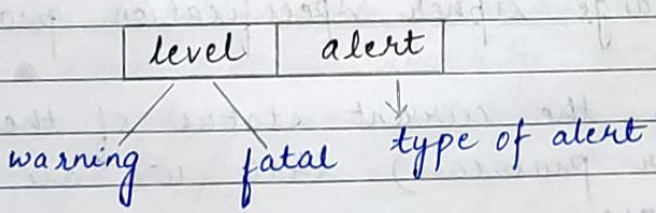
> Phase 2 - Server sends his certificate and server key exchange. It requests for the client certificate. The server ends phase-2 by sending the "server hello done".

> Phase 3 - In this phase, client replies to the server by sending his certificate and client-exchange-key.

> Phase 4 - In Phase-4, change cipher spec occurs and after this the handshake protocol ends, first from the client side and then from the server side.

SSL Alert Protocol

The primary job of the SSL Alert Protocol is to inform the other end (server or client) about the issues in the current session



Warning - This has no impact on the connection between sender and receiver. Only ~~the~~ a warning

would be received, the communication / program will continue.

Fatal - This immediately breaks the connection between sender and receiver. The communication will stop.

* Alert occurs when the two entities that are communicating face any kind of problem.
* An entity informs the other entity if it encounters any error.

Type of Alert	Alert Message	Description
(1)	close_notify	It notifies that the sender will no longer send any msgs
(2)	unexpected_message	incorrect message received.
(3)	bad_record_mac	wrong MAC received
(4)	bad_certificate	when the received certificate is corrupt
(5)	certificate_expired	when a certificate has expired.

SSL change cipher specification protocol

1. It keeps the current status of the protocol (cipher protocols) that we are using right now.

2. It has only one message (1 Byte)

3. This protocol's purpose is to cause the pending state to be copied into the current state.

4. One message of 1 Byte consists of value 1. (1-1-1)

TLS (Transport layer security)

> TLS was proposed by IETF (Internet Engineering Task Force)
> TLS is defined in RFC 2246] SSL 3.0
> It is a protocol that provides authentication, privacy and data integrity between two communicating computer applications.
> It is designed to provide security at transport layer.
> TLS was derived from SSL.
> Encrypts the communication between web app and server.
> It's the most widely deployed security protocol in use today and is best suited for web browsers and other applications that require data to be securely exchanged over a network.
> TLS is a cryptographic protocol that provides end to end communication security over network.
> It is a widely used protocol for internet communication / data sharing and online transactions.

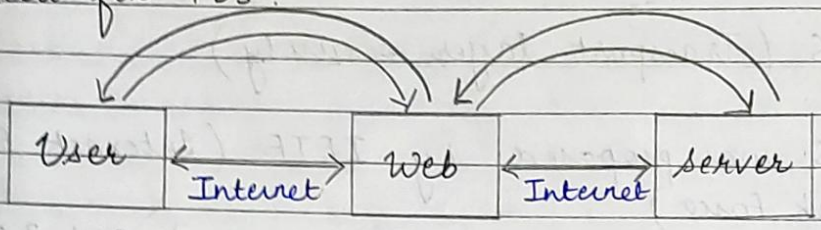
Page No.
Date

Goals of TLS

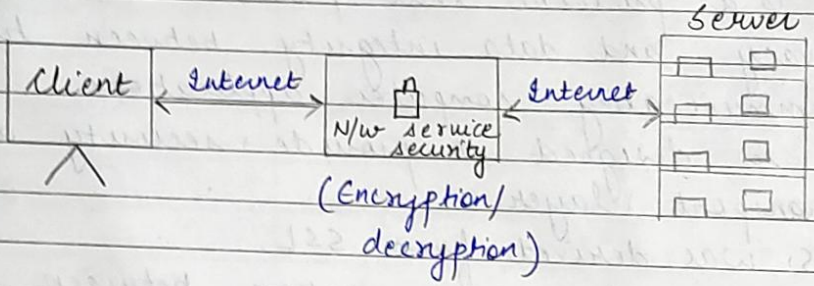
Record Protocol { C - Confidentiality
I - Integrity (HMAC)
Handshake { A - Authentication / availability

Need for TLS:

Simple internet communication

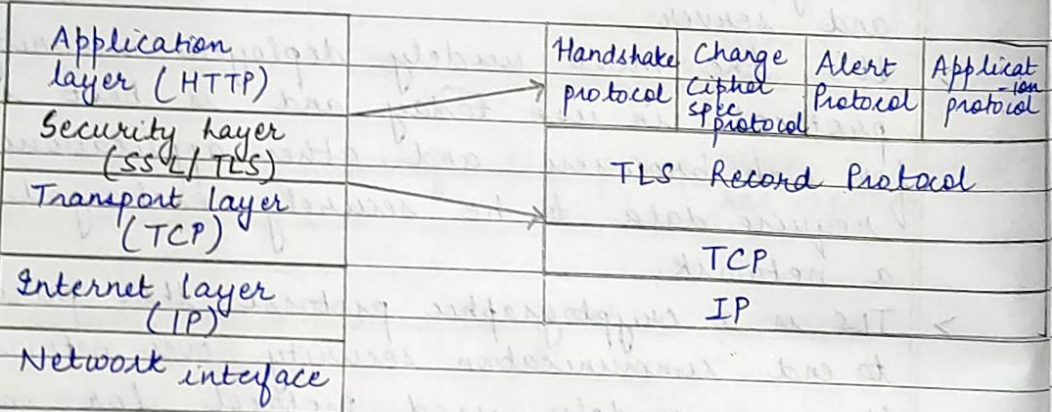


Online transaction



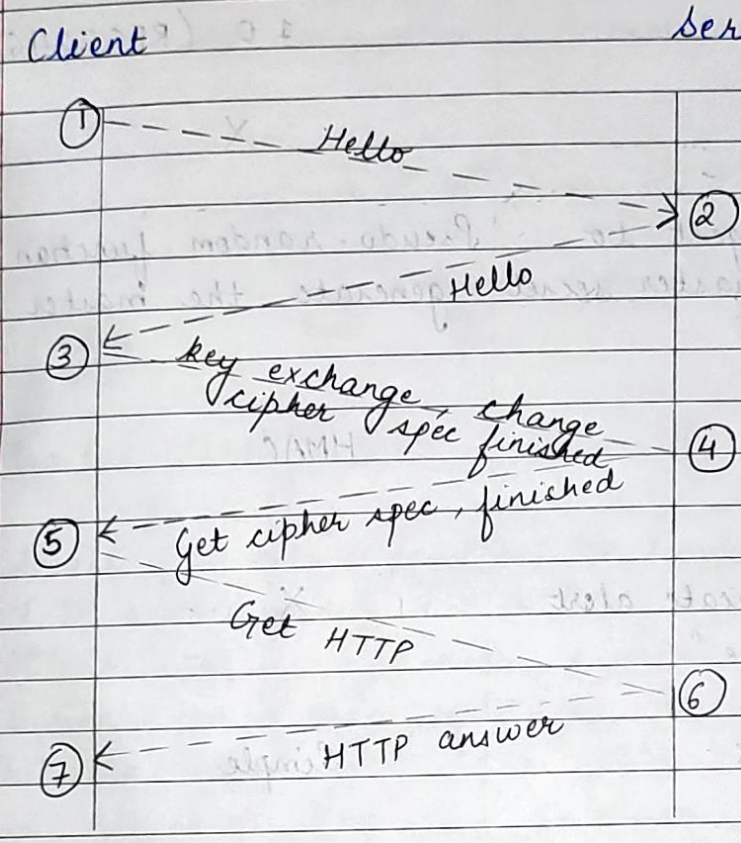
higher level protocol

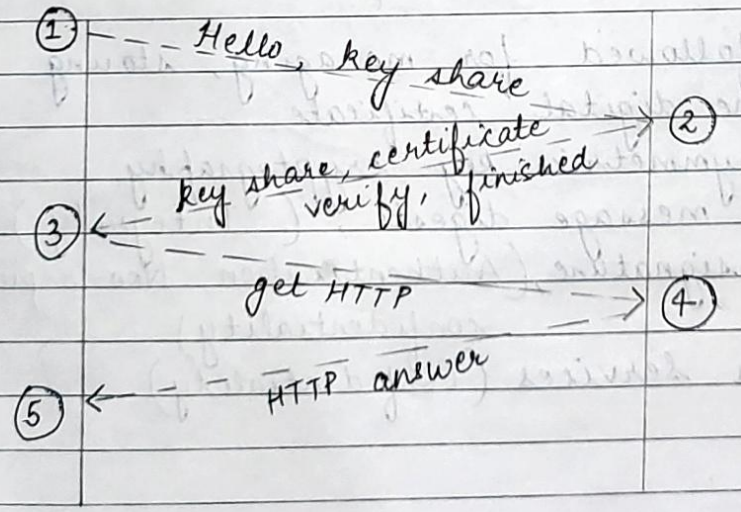
lower layer protocol



* TLS Record Protocol works exactly the same as in SSL, the only difference being that here HMAC (Hash Message Authentication code) is

used instead of MAC.





Page No.
Date

Difference between SSL & TLS

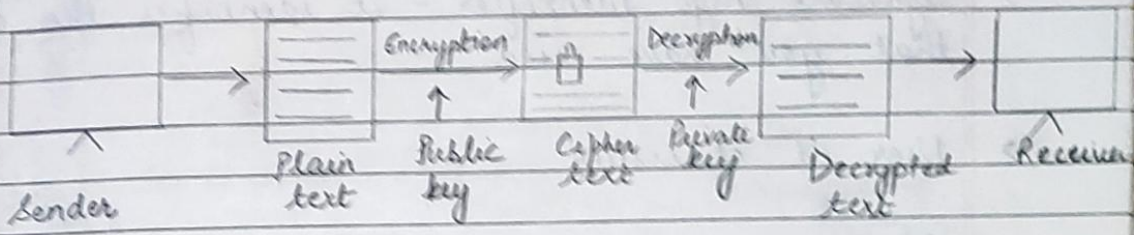
	SSL	TLS
Version:	3.0	1.0 (RFC 2246)
Cipher suite	Fortezza	X
Cryptography Secret	Message digest to generate master secret	Pseudo-random function to generate the master secret.
Record Protocol	MAC	HMAC
Alert Protocol	"No certificate alert Message"	X
Certificate verification	Complex	Simple

PKI (Public Key Infrastructure)

> Standard followed for managing, storing and revoking the digital certificate.
> follows asymmetric key cryptography.
> Includes message digests, (Integrity)
Digital signature, (Authentication, Non-repudiation, confidentiality)
Encryption services (Confidentiality)

> To enable all the services, digital certificates are required

Public key encryption (Asymmetric key Cryptography)



Digital Certificate

1. Small file on computer / Electronic device
2. File extension is (.cer)
3. It is issued by some trusted party / entity
4. Digital certificate establishes a relationship between the user and public key
5. It requires name of the user and his public key

Sample Digital Certificate

Username : abc
Public key : <abc &#12>
Serial no : 12345
Other information : email-id
Valid from : 23-04-2022
Valid to : 23-04-2026
Issuer Name : Entrust, verisign

Fields of Digital Certificate

> Version : X.509 - It defines the standard of digital certificate.

> Signature Algo Identifier - It identifies the algo that you have used.

> User id of issuer

> CA digital signature (Certification Authority) : used during digital certificate verification.

* What is Certification Authority (CA) ?

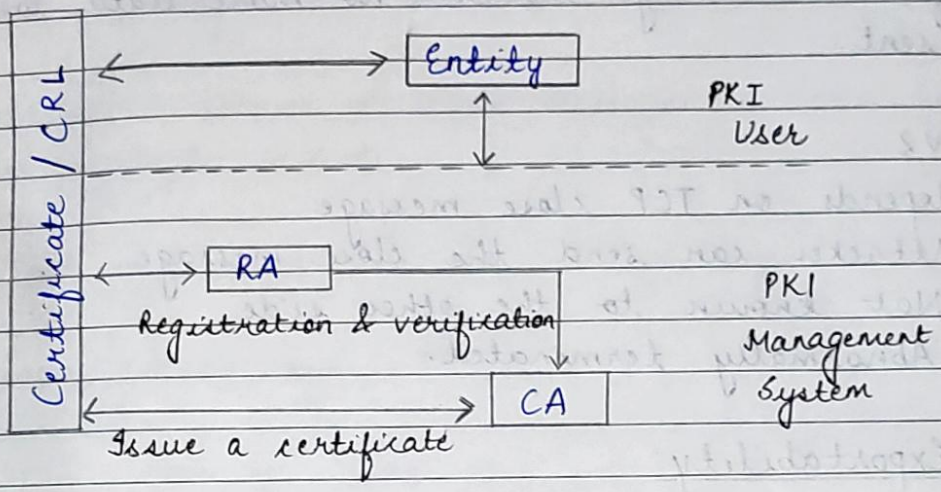
CA are trusted agency that can issue digital certificate.

Components of PKI

1. Certificate Management System
2. Digital Certificate
3. Validation Authority
4. Certification Authority
5. Registration Authority
6. End user

Page No.
Date

Architecture of PKI



CRL - Certificate Revocation list
It is a list of digital certificates that have been revoked by the issuing certificate authority (CA) before their scheduled expiration date and should no longer be trusted.

SSL Attacks in version 3 (V3)

Downgrade Attack

> V2
- No integrity checks
- Active attacker can remove cipher suite

> V3
- finished messages
- digest of all previous messages

Truncation Attack

> V3

- finished message indicate no more data to be sent

> V2

- depends on TCP close message.
- Attacker can send the close message.
- Not known to the other side.
- Abnormally terminated.

Exportability

1. Export controls - Weak crypto
2. Strong crypto - complex mechanism

Exportability in SSL V2 -

> limited to 40 bits
> Use 128 bit key
> 40 bit - secret encrypted with server's public key
> 88 bit - in clear (non-secret bit)
> Client Master key

Exportability in SSL V3 -

> Only 40 bits keys allowed.
> Servers can encrypt keys using 512 b RSA keys.
> Normally, RSA keys are 1024 b
> 512 b Ephemeral key

Page No.
Date

Encoding

> All exchanges are in records up to 2^{14} B or $2^{16}-1$ B.

> Standard allows multiple messages in one record or multiple records.

> Most implementations use one message per record.

Four Record types -

> 20 = Change cipher spec
> 21 = Alerts (1 = Warning, 2 = Fatal)
> 22 = Handshake
> 23 = Application data

Record header :

1	Record Type
2	Version Number
3	length

> Each message starts with a 1B message-type and 3B message length.

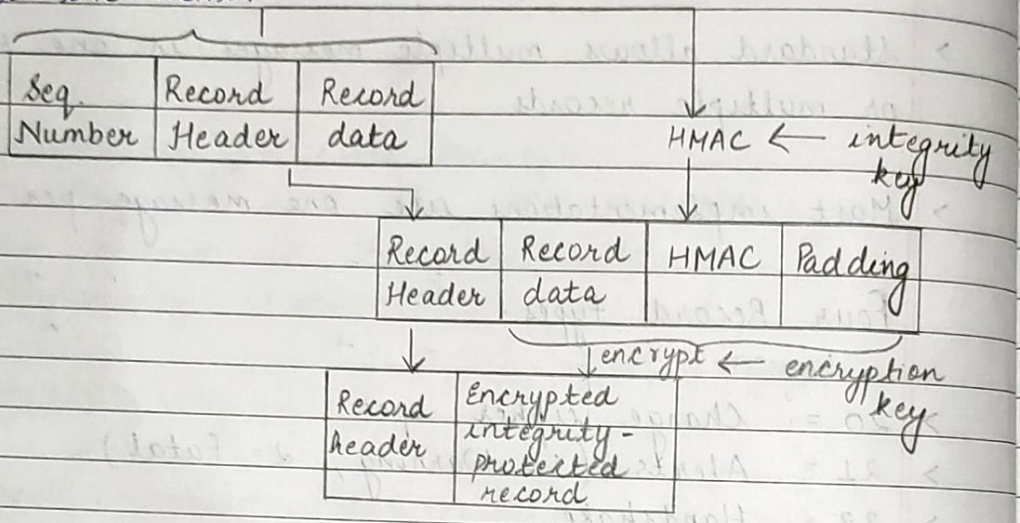
Encrypted Records

> Integrity is provided by HMAC using the integrity key.

> Data prefixed by 64b sequence.

> Block cipher = 40B padding in SSLV3,
44B in TLS

> Final block of each record is used as IV for the next.



Secure Electronic Transaction (SET)

> open encryption and security specification designed to protect credit-card and debit-card transactions on the Internet.

Services of SET:

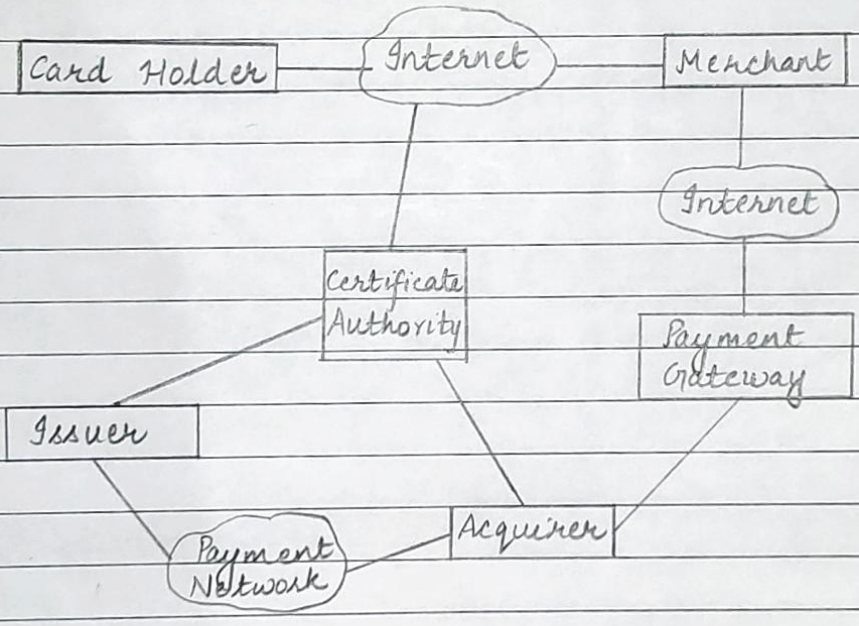
> Provides a secure communication channel
> Provides authentication by use of digital certificates.
> Ensures confidentiality (the information is available only to the parties involved in the transaction)

SET Participants:

i) Card holder (User)
ii) Merchant
iii) Issuer (Financial Institution - Bank which provides payment card)
iv) Acquirer (has relation with the merchant for processing of the payment card authorisation)

v) Payment Gateway - It is a third party organisation which exists between client and merchant. It processes the payment messages on behalf of the merchant)

vi) Certificate Authority - Provides and verifies the digital certificate.



Page No.
Date

Requirements in SET :

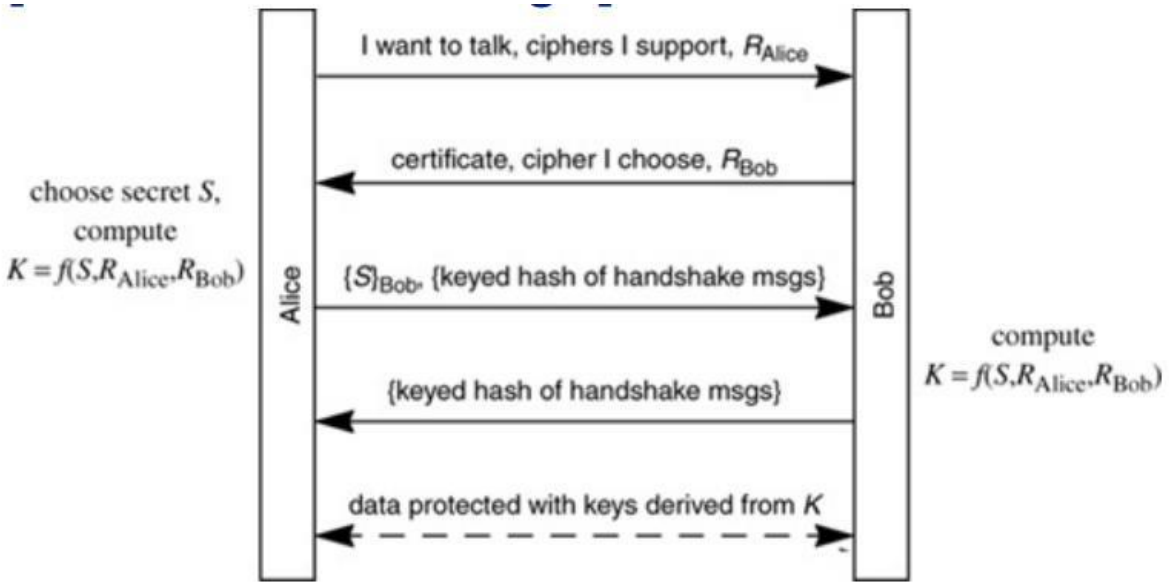
i) Mutual authentication
ii) Payment / order information confidentiality
iii) No message modification
iv) Interoperability

Computing the Keys:-

• The secret S sent in the first exchange is the pre master secret (Alice chooses a random number S (known as the pre master secret) and sends it, encrypted with Bob's public key)

She also sends a hash of the master secret K and the hand shake messages, both to prove she knows the key and to ensure that tampering of the handshake messages would be detected)

The notation f($K_{\text{Alice Bob}}$, R) means that R is cryptographically transformed somehow, with Alice and Bob's shared secret $K_{\text{Alice Bob}}$



• It is shuffled with the two Rs to produce the master secret K In other words K =f(S,R Alice ,R Bob

• For each connection (including the first) the master secret is shuffled with the two Rs to produce the six keys used for that connection (for each side encryption, integrity, and IV), i.e. each of the keys is g_i (K,R_{Alice} ,R_{Bob}

• Note that this is unnecessarily complex in the case of using RSA since on the first connection, the Rs get shuffled in twice

• It would have been fine with RSA to just use S the way K is used for each connection, combine S with that connection's Rs to produce the six keys used for that connection

• However, there are other authentication methods that would wind up always computing the same pre master secret S

• For instance, with Diffie Hellman using a fixed Diffie Hellman number as your public key, the same two parties (Alice and Bob) will always compute the same S

• If S were kept around in memory it's possible that malicious software might steal it

• So, it's safer to hash it with some nonces to produce K (the master secret), since if K is stolen, it will only affect communication between Alice and Bob during the single session

• The keyed hashes are keyed with K, and are sent protected with the data keys (i e encrypted using the encryption key and IV, and integrity protected with the integrity key

• The Rs are 32 octets long and it is recommended (but not enforced) that the first 4 octets not be randomly chosen, but instead be the Unix time (seconds since January 1 1970 when the message was generated

• This ensures (assuming that at least a second has elapsed) that Alice and Bob will choose different 32 octet Rs each time a session under the same master secret is resumed, even if their random number generators are really bad or they're really unlucky

• If the same Rs were chosen with the same master secret, an attacker might be able to successfully replay packets

Client Authentication :-

• As deployed today it is unusual for clients to have certificates

• However, it is possible in SSL/TLS for the server (to request that the client (authenticate herself

• This is done by having Bob send a certificate request in message 2

• Alice, upon seeing the request, sends her certificate, and her signature on a hash of the handshake messages, proving she knows the private key associated with the public key in the certificate