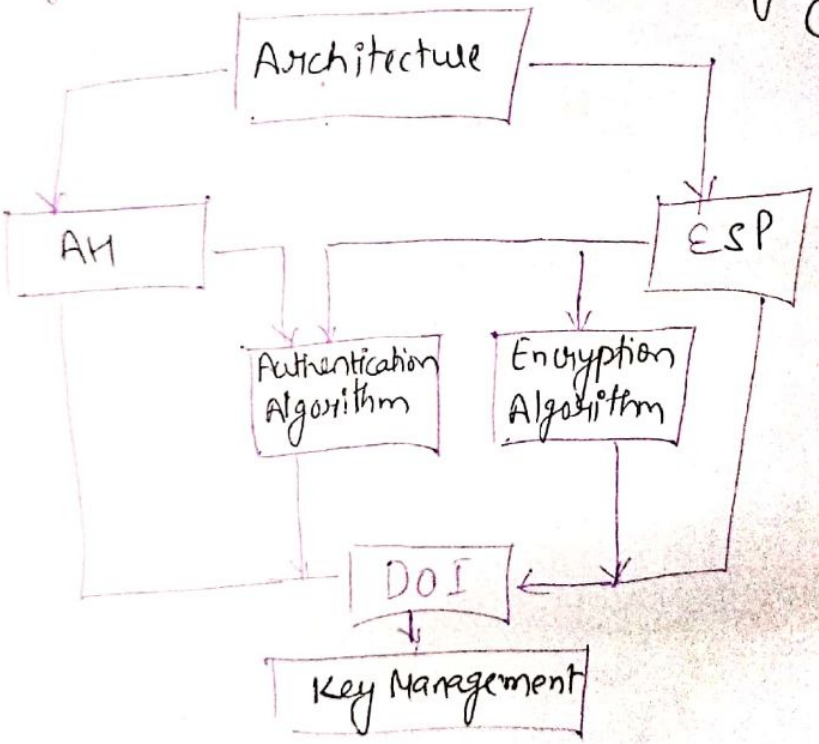
IP Security:-
(IPsec) Internet Protocol Security (IPsec) is a framework for protecting communication over IP.

Application of IPsec:

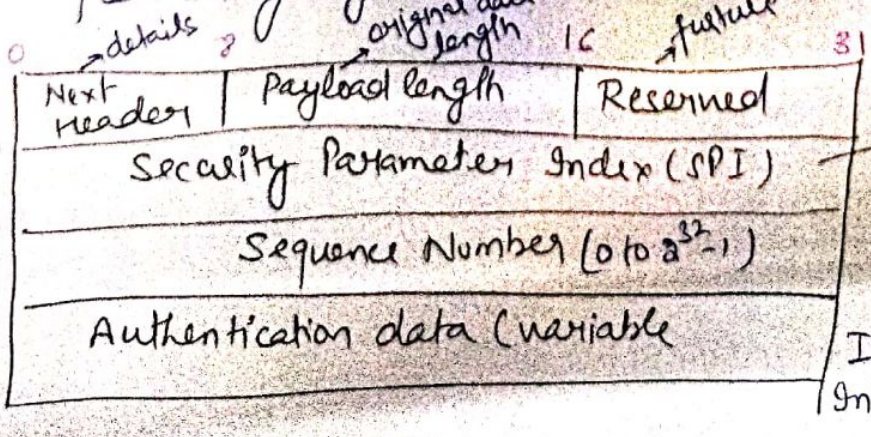
1. Secure branch office connectivity over the internet
2. Secure remote access over the internet
3. Enhancing electronic commerce security

IP Security Architecture:- — Authentication Header (AH)
— Escapsulating Security Payload (ESP)



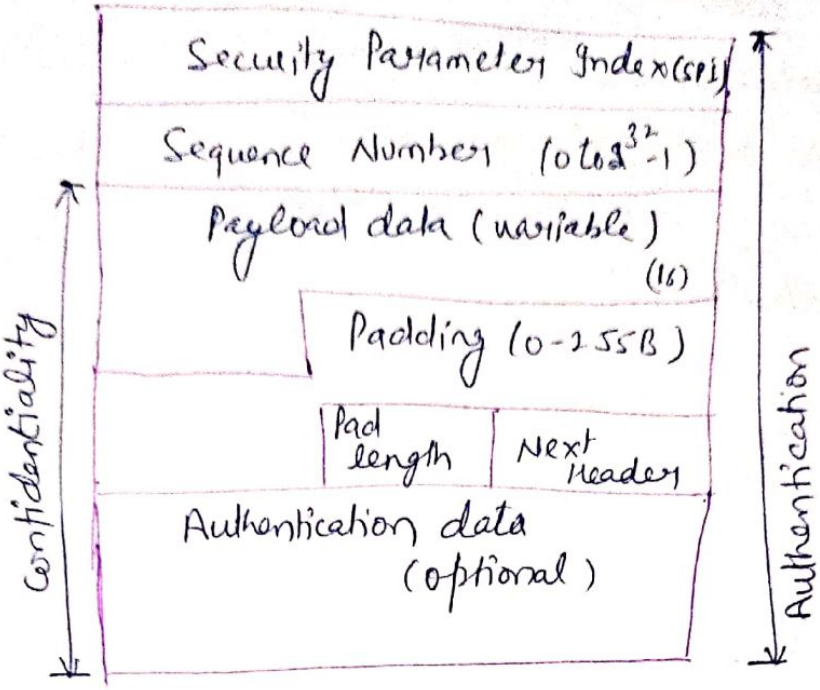
DOI - domain of Interpretation (ids) that support both AH and ESP

Authentication Header:-
for integrity and Authentication → 32 bit (0-31 bit)



details
original data length
future
Security Assoc of a packet
ICV or MAC Integrity check value

Encapsulating Security Payload



Security Parameter Index (SPI) : This parameter is used by Security Association. It is used to give a unique no. of to the connection built b/w client and server.

Sequence Number :- Unique Sequence no. are allowed to every packet so that on the receiver side packets can be arranged properly

Payload Data : ~~Payload~~ payload data means the actual data or original message The Payload data is an encrypted format to achieve confidentiality

Padding :- Extra bits of space are added to original message in order to ensure confidentiality. Padding length is the size of the added bits of space in the original message.

Next Header :- Next header means the next payload or next actual data

Authentication data :- This field is optional in ESP protocol packet format

Security Association (SA)

↳ SA describe a particular kind of secure connection b/w one device to another.

↳ SA are key to IPSec's authentication and confidentiality mechanisms.

↳ SA are needed to negotiate in the exchange of the "shared secret" process.

Uniquely identified by three parameters:

Security Parameter Index (SPI)

Security Protocol Identifier (AH/ESP)

IP Destination Address

Sequence number

Authentication Header Info

ESP info

IPsec Protocol Mode — 2 modes

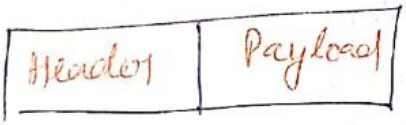
Transport Mode

Tunnel Mode

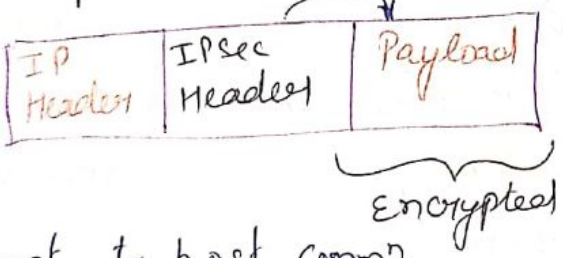
Transport Mode :-

- payload — Encrypted
- Header — Not Encrypted

Initially,



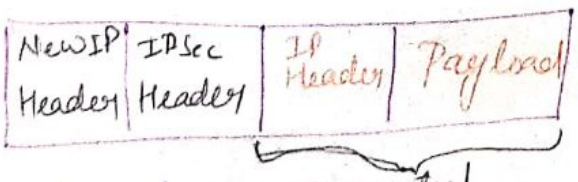
later, in transport mode, we insert IPsec header in b/w



- direct host to host comm^n.

Tunnel Mode :- payload & Header } Both are encrypted

(i.e. entire packet is encrypted as a result new IP Header is generated)



- direct gateway to gateway comm^n.

sender

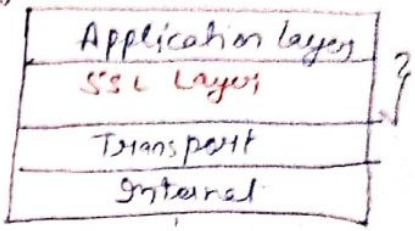
Receiver

SSL (Secure Socket Layer):

↳ Internet Protocol for secure exchange of "data" b/w browser and server

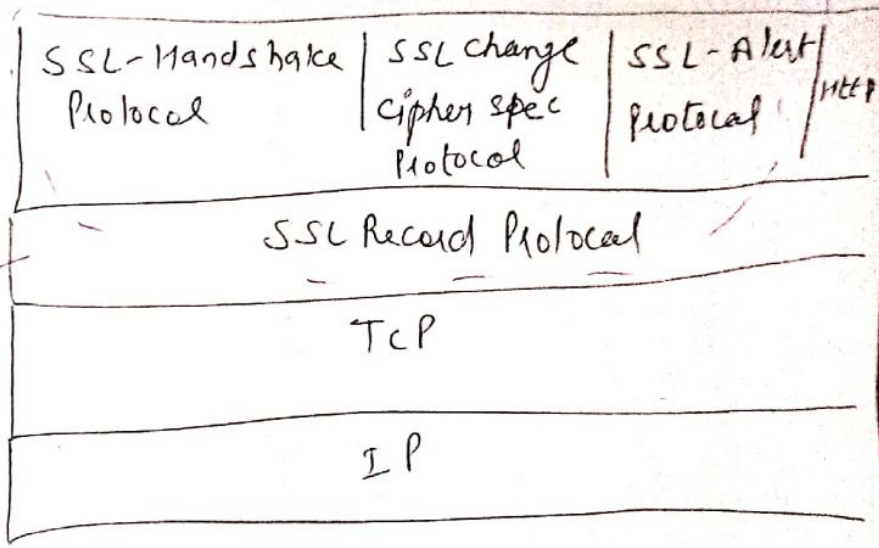
↳ Provides security at Transport layer

GOALS:
- i) Confidentiality [C]
- ii) Integrity [I]
- iii) Authentication [A]



Working of SSL

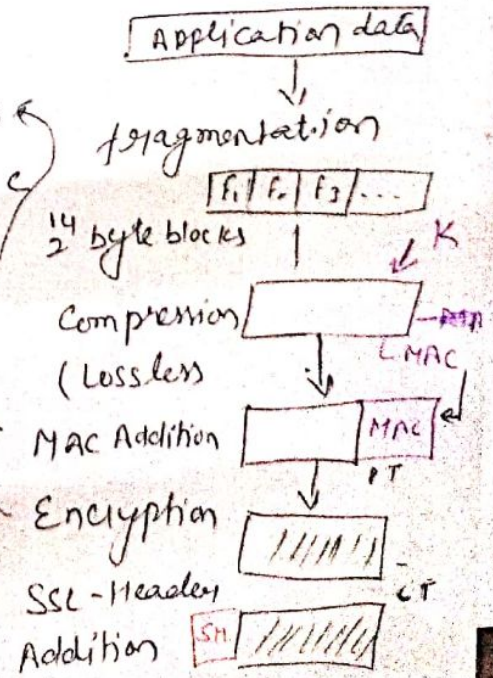
SSL Protocol stack:



SSL Record Protocol.

↳ Provides two services for SSL Connections
- ↳ (i) Confidentiality : Encryption
- ↳ (ii) Message Integrity : MAC

SSL Header consists of 4 fields
1. Content type
2. Version (major)
3. Version (minor)
4. Compressed length



Change cipher Spec Protocol

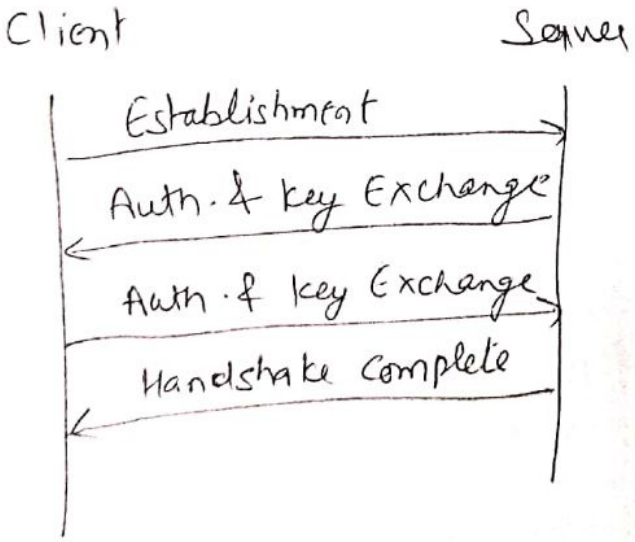
It consists of a single message consisting of single byte with value 1. It is used to cause pending state to be copied into current state which updates cipher suite

Alert Protocol: Convey SSL-Protocol related alerts to the peer-entity

Byte 1 → 1 (warning) / 2 (fatal) | Byte 2 → Type of alert

Handshake Protocol

- most complex part of SSL
- Authentication b/w client and server
- negotiation of Encryption / MAC Algorithm
- exchange / Negotiation of cryptographic keys



TLS

Transport Layer Security

↳ It is an IETF standardization initiative

↳ Goal is to ~~provide~~ produce Internet standard version of SSL

↳ TLS is defined in RFC 2246 (Req. for ~~Comm~~ Comments)

↳ provide security in transport layer

↳ provide a secured connection b/w client and server (i.e. no third parties)

↳ TLS is used by http, smtp.

Working :-

- uses client server handshake mechanism

1. key exchange b/w client and server (by diffie-Hellman key Exchange Alg)
2. Now TLS protocol will open an Encryption channel (by RC4 / IDEA / DES Algorithm)
3. It also ensures that the messages are not altered (by MD5/ SHA algorithm)

* RFC 2246 is ~~sim~~ similar to SSLV3 (SSL version 3)

SSH Protocol :- (Secure Shell Protocol) Difference b/w SSL and ~~TLS~~ TLS

	SSL	TLS
Version	3.0	1.0
Cipher Suite	Fortezza	x
Key. Secret	M.D to generate master secret	Pseudo-Random func^n to generate master secret
Record Protocol:	MAC	HMAC
Alert Protocol:	"No Certificate"	x
Certific verification	Complex	Simple

Secure Shell Protocol (SSH)

↳ protocol for operating Network services over an insecured n/w (Internet)

↳ alternative to Telnet, FTP etc (unsecured)

↳ It uses client server Architecture

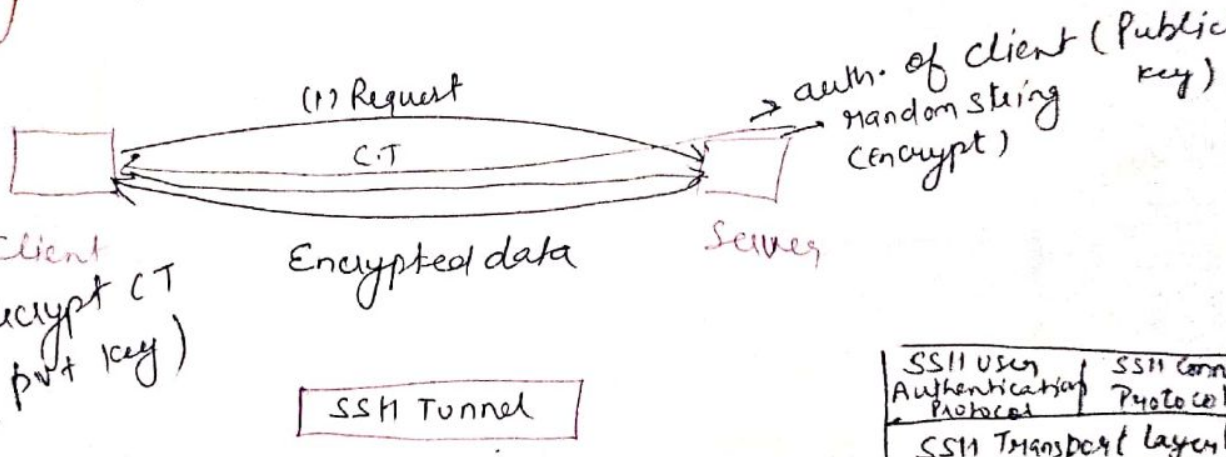
↳ follows asymmetric-key cryptography

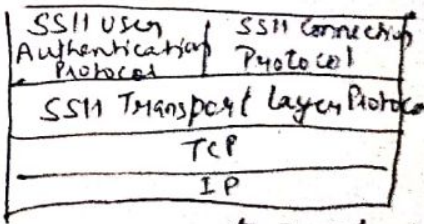
Encryption - Public key

Decryption - Private key

- provides confidentiality and integrity

Working





SSH Protocol stack

↳ is organized as three protocol that typically run on top of TCP

↳ SSH user Authentication Protocol :- Authenticates client (user) to the server

SSH Connection Protocol :- Multiplexes the encrypted tunnel into several logical channel.

SSH Transport layer Protocol :- Server authentication, confidentiality and integrity, compression.