UNIT-IV

Transport-Level Security

Wireless Network Security

UNIT-4

Part-1 Transport level security

→ Web Security considerations

The World wide Web is fundamentally a client/server application running over the Internet and TCP/IP intranets.

- The Internet is a two way. Unlike traditional publishing environments - even electronic publishing systems involving teletext, voice response the Web is vulnerable to attacks on the web servers over the Internet
- The Web is increasingly serving as a highly visible outlet for corporate and product information and as the platform for business transactions
- Although web browsers are very easy to use, They are relatively easy to configure and manage and web content is increasingly easy to develop, the underlying software is extraordinarily complex.
- A web server can be exploited as a launching pad into the corporation's or agency's entire computer complex.
- Casual and untrained users are common clients for web based services. Such users are not necessarily aware of the security risks that exist and donot have the tools or knowledge to take effective countermeasures.

→ Web Security threats

One way to group threats is in terms of passive and active attacks.

- Passive attacks include eavesdropping on network traffic between browser and server gaining access to information on a website that is supposed to be restricted
- Active attacks include impersonating another user, altering messages in transit between client and server

Another way to classify web security threats is in terms of location of the threat

web server, web browser and network traffic between browser and server

- Issues of server and browser security fall into the category of computer system security

→ Comparision of threats on the web

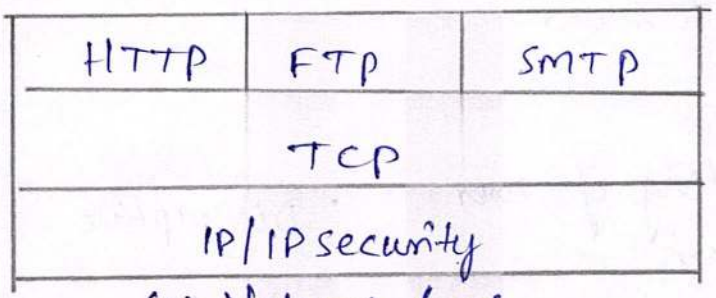
	Threats	Consequences	counter measures
Integrity	• Modification of user data • Trojan horse browser • Modification of memory • Modification of message traffic in transit	• Loss of information • compromise of machine • Vulnerability to all other threats	Cryptographic Checksum

	Threats	Consequences	Counter measures
Confidentiality	• Eavsdropping on the net • Theft of info from server • Theft of data from client • info about network configuration • info about which client talks to server	• Loss of info • Loss of privacy	• Encryption • Web proxies
Denial of service	• Killing of user thread • flooding machine with bogus requests • filling up disk or memory • Isolating machine by DNS	• Disruptive • Annoying • Prevent user from getting work done	• Difficult to prevent
Authentication	• Impersonation of legitimate users • Data forgery	• Mis representation of user • Belief that false info is valid	Cryptographic techniques

Comparision of threats on the Web

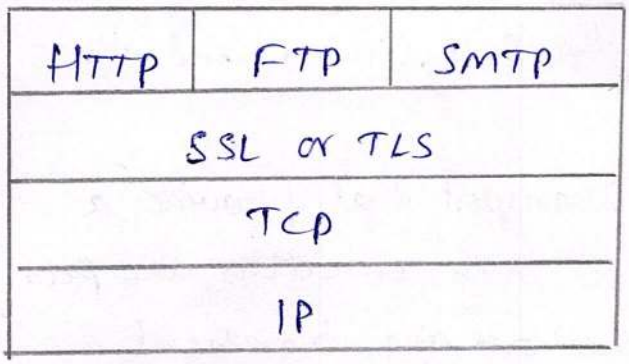
→ Web traffic security approaches

- One way to provide web security is to use IP security (IPsec). The advantage of using IPsec is that it is transparent to end user and applications and provide a general purpose solution and it includes a filtering capability so that only selected traffic need incur the overhead of IPsec processing

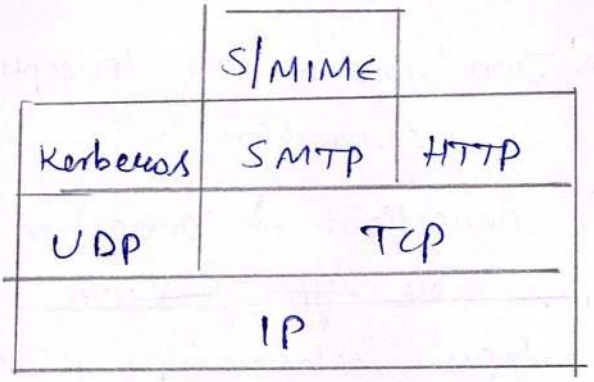


(a) Network Level

- Another way is to implement security just above TCP the foremost example of this approach is the secure socket layer (SSL) and the follow-on internet standard known as Transport layer security (TLS) At this level, there are 2 implementation choices. For full generality SSL (or TLS) could be provided as a part of the underlying protocol suite and therefore be transparent to applications. Alternatively, SSL can be embedded in specified packages for example Netscape and Microsoft explorer browsers come equipped with SSL and most web services have implemented the protocol



Transport level



Application level

- Application - Specific security services are embeded within a particular application. The advantage of this approach is that a server can be tailored to the specific needs of a given application

-> Secure Socket layer and transport layer security

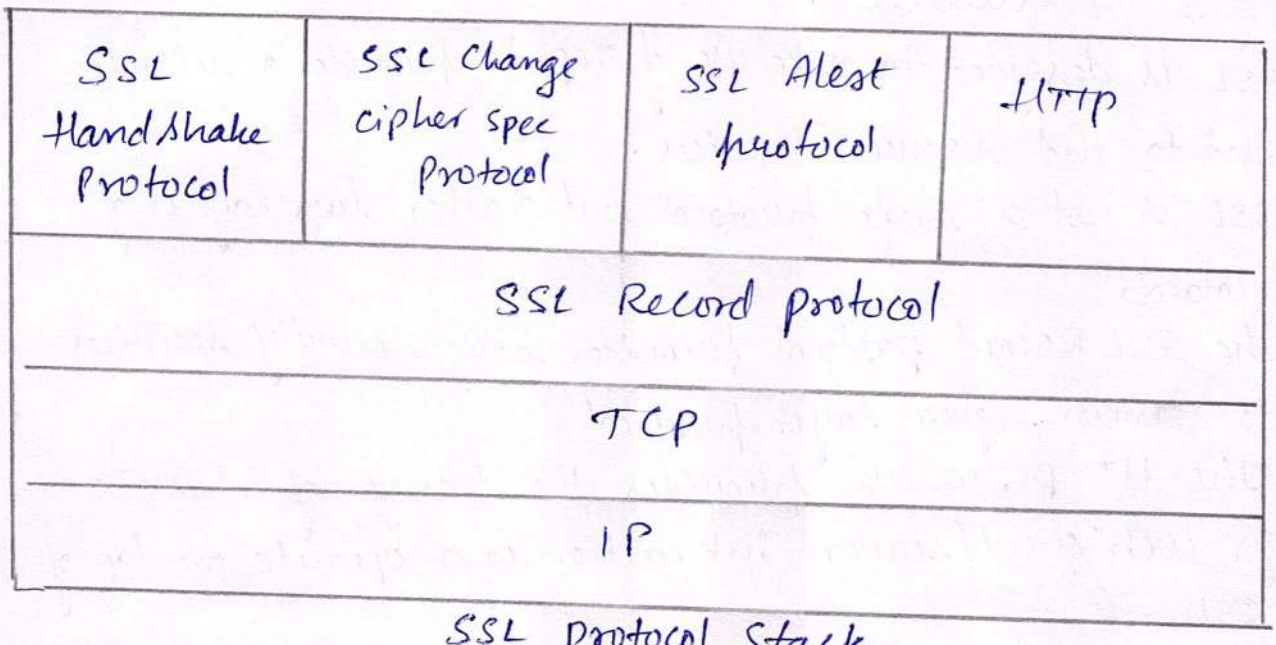
SSL Architecture

- SSL is designed to make use of TCP to provide a reliable end to end secure services.
- SSL is not a single protocol but rather two layers of protocols
- The SSL Record protocol provides basic security services to various higher layer protocol
- The HTTP, which provides the transfer of service for web client/server interaction can operate on top of SSL
- Three higher-layer protocols are defined as part of SSL the handshake protocol, The change cipher spec protocol and the Alert protocol

→ Two important SSL concepts are the SSL session and the SSL connection

i) Connection:- A connection is a transport that provides a suitable type of service. For SSL, such connections are peer to peer relationships. The connections are transient. Every connection is associated with one session

ii) Session:- An SSL session is an association between a client and a server. Sessions are created by the handshake protocol. Session define set of cryptographic security parameters which can be shared among multiple connections. They are used to avoid the expensive negotiation of new security parameters for each connection



SSL protocol Stack

• Between any pair of parties there may be multiple secure connections. There may also be multiple simultaneous sessions between parties

- There are no of states associated with each session. Once a session is established there is a current operating ~~state~~ state for both read and write in addition, during the hand shake protocol, pending read and write states are created upon successful conclusion of the hand shake protocol, the pending states become the current states

→ A session state is defined by the following parameters

- Session identifier :- An arbitary byte sequence chosen by the service to identify an active or resumable session state
- Peer certificate :- An X509.v3 certificate of the peer. This element of the state may be null
- compression method :- The alg used to compress data prior to encryption
- Cipher spec :- Specifies the bulk data encryption algorithm and a hash algorithm used for mac calculation
- Master secret :- 48 byte secret shared b/w client & server
- Is resumable :- A flag indicating whether the session can be used to initiate new connections

→ A connection state is defined by the following parameter

- Server and client random :- Byte sequence that are choosen by the server and client for each connection
- Server write MAC secret :- The secret key used in MAC operations on data sent by the server

- Client write MAC Secret :- The secret key used in MAC op on data sent by client
- Server write key :- The secret encryption key for data encrypted by the server and decrypted by the client
- Client write key :- The symmetric encryption key for data encrypted by the client and decrypted by the server
- Initialization Vector :- When a block cipher in CBC mode is used, an initialization vector (IV) is maintained for each key. This field is first initialized by the SSL Handshake protocol.
- Sequence numbers :- Each party maintains separate sequence numbers for transmitted and received message for each connection.

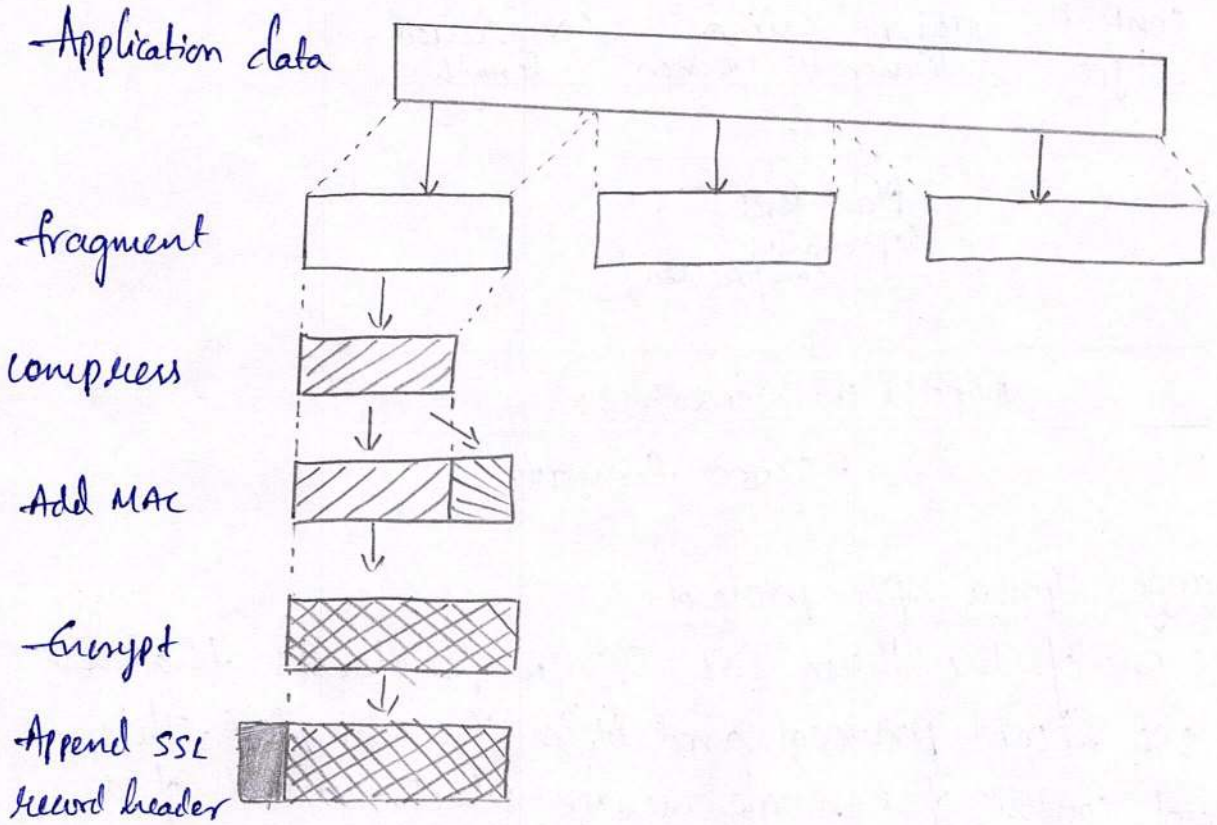
→ SSL Record protocol

It provides 2 services for SSL connections

- Confidentiality :- The handshake protocol defines a shared secret key that is used for conventional encryption of SSL payloads
- Message integrity :- The handshake protocol also defines a shared secret key that is used to form a message authentication code (MAC)

→ The record protocol takes an application message to be transmitted, fragments the data into manageble blocks, optionally compressed data, applies a mac, encrypts, adds a header and transmits the resulting unit in a TCP segment.

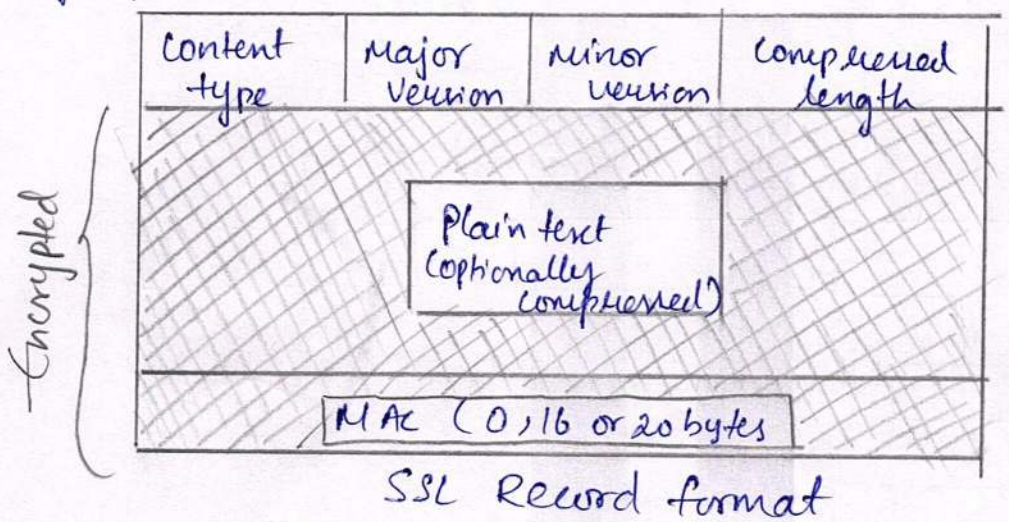
- Recieved data are decrypted, verified, decompressed and reassembled before being delivered to higher level users
- The first step is fragmentation. Each upper layer message is fragmented into blocks of 2^{14} bytes (16384 bytes) or less.
- Next, compression is optionally applied. It must be lossless and may not increase the content length more than 1024 bytes
- Next, message authentication code over the compressed data For this purpose, a shared secret key is used



SSL Record protocol Operation

- next the compressed message plus the MAC are encrypted using symmetric encryption.

- The final step of SSL Record protocol processing is to prepare a header consisting of the following fields
- Content type (8 bits) :- The higher layer protocol used to process the enclosed fragment
- Major version (8 bits) :- indicates major version of SSL in use. For SSL V_3 the value is 3
- Minor version (8 bits) : indicates minor version of SSL for SSL V_3 the value is 0
- Compressed length (16 bits) :- The length in bytes of plaintext fragment. The max value is $2^{14} + 2048$.

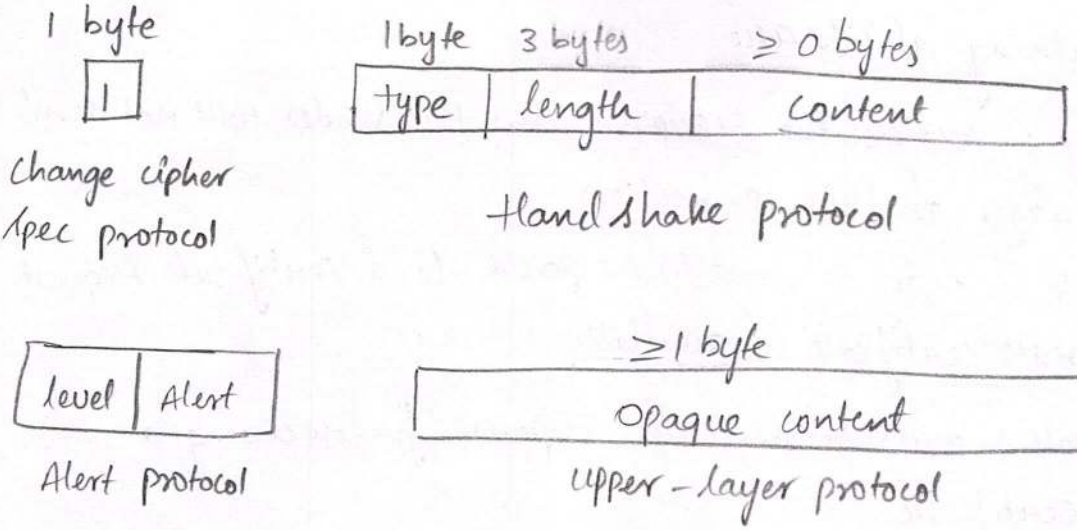


SSL Record format

→ Change cipher spec protocol

- It's one of the three SSL-specific protocols that use the SSL Record protocol and it is the simplest. This protocol consists of a single message which consists of a single byte with the value 1

The sole purpose of this message is to cause the pending state to be copied into the current state, which updates the cipher suite to be used on this connection



→ Alert protocol

The alert protocol is used to convey SSL-related alerts to the peer entity. As with other applications that use SSL, alert message are compressed and encrypted, as specified by the current state

Each message in this protocol consists of 2 bytes. The 1st byte takes the value warning (1) or fatal (2) to convey the severity of the message

If the level is fatal SSL immediately terminates the connection. Other connection on the same session may continue, but no new connections on this session may be established

→ List of fatal errors

i) unexpected_message :- An inappropriate msg recieved

ii) bad_record_mac :- An incorrect MAC recieved

iii) decompression_failure :- The decompression fn recieved improper input

iv) hand shake failure :- Sender was unable to negotiate an acceptable set of security parameter given the options available

v) illegal_parameter :- message out of range or inconsistent with other fields

-> The remaining alerts are

i) close_notify :- Notifies the recipient that the sender will not send any more messages on this connection

ii) no_certificate :- may be sent in response to a certificate request if no appropriate certificate is available

iii) Bad_certificate :- may be sent in response for recieving a corrupt certificate

iv) unsupported_certificate :- Type of certificate not supported

v) certificate_revoked :- revoked by its signer

vi) certificate_expired :- It's expired

-> Hand shake protocol

- Most complex protocol
- It allows the server and client to authenticate each other and to negotiate an encryption and MAC algorithm and cryptographic keys to be used to protect data sent in an SSL record.
- It is used before any application data is transmitted
- The hand shake protocol consists of a series of messages exchanged by client and server

Each message has 3 fields

i) Type (1 byte) :- indicates one of the 10 messages.

ii) length (3 bytes) :- The length of message in bytes

iii) content (≥0 bytes) :- There parameters are associated with the message

Message type	Parameters
Hello_request	null
client_hello	version, random, session id, cipher suite, compression method
Server_hello	version, random, session id, cipher suite, compression method
certificate	chain of X.509 V_3 certificate
server_key exchange	Parameters, signature
certificate_request	type, authorities
Server_done	null
finished	Hash Value

→ The exchange can be viewed in 4 phases

Phase 1 :- Establish security capabilities

This is used to initiate a logical connection and to establish the security capabilities that will be associated with it. The exchange is initiated by the client, which sends client_hello message with all the parameters that are required

- Version :- highest SSL version understood by client
- random :- client generated random structure consisting of 32 bit timestamp and 28 byte generated by a secure random no generator

- Session-ID :- A Variable length session identifier
- Cipher suite :- Combination of cryptographic alg supported by client
- Compression method :- list of compression methods that client support

→ After sending the client_hello message, the client waits for server_hello message with same parameters as client_hello but version contains lower version supported by server, the randomgenerator is generated by the server and is independent of the clients random field, session id field contains the value for a new session

→ Some of the key exchange methods which are supported are

- RSA, fixed diffie-Hellman, ephemeral diffie-hellman, Anonymous Diffie-hellman, fortezza

→ Key exchange method fields

- cipher algorithm
- MAC algorithm
- cipher type
- Is exportable
- Hash size
- Key material
- IV size

→ Phase-2 Server authentication and Key exchange

The Server begins this phase by sending its certificate if it needs to be authenticated.

The message contains one or a chain of X.509 certificates

The certificate message is required for any agreed-on key exchange method except anonymous diffie-Hellman

- Server_Key-exchange message may be sent if it's required. It's not required in 2 instances

1) The server has sent a certificate with fixed diffie-Hellman parameters

2) a RSA Key exchange is to be used. The

- The Server_Key-Exchange message is needed for the following
  - Anonymous diffie Hellman
  - Ephemeral diffie-Hellman
  - RSA key exchange
  - Fortezza

- Next non-anonymous Server can request a certificate from the client. The certificate_request message includes 2 parameters: certificate_type and certificate_authority

The certificate_type indicates the public key algorithm and its use:

- RSA, Signature Only
- RSA for fixed diffie Hellman
- RSA for ephemeral diffie Hellman
- DSS, Signature only
- DSS for fixed diffie Hellman
- DSS for ephemeral diffie Hellman
- fortezza

The second parameter in the certificate_request message is a list of the distinguished names of acceptable certificate authorities

→ Phase 3 :- Client Authentication and Key Exchange
Upon recieving the server_done message the client should verify that the server provided a valid certificate and check the server_hello parameters are acceptable. If all is satisfactory, the client sends the one or more messages back to server
If the server has requested a certificate, the client begins this phase by sending a certificate message. If no suitable certificate is available, the client sends a no_certification alert..

- Next is the client key_exchange message, The content of the message depends on the type of key exchange as follows
  - RSA :- client generates a 48 bit pre master secret key and encrypt with the public key from the server's certificate
  - Ephemeral or Anonymous diffie-Hellman
  - fixed diffie Hellman
  - Fortezza

- finally the client may send a certificate_verify message to provide explicit verification of a client certificate

Phase 4 : FINISH

This phase completes the setting up of a secure connection

The client sends a change_cipher_spec message and copies the pending cipher spec into the current cipher spec

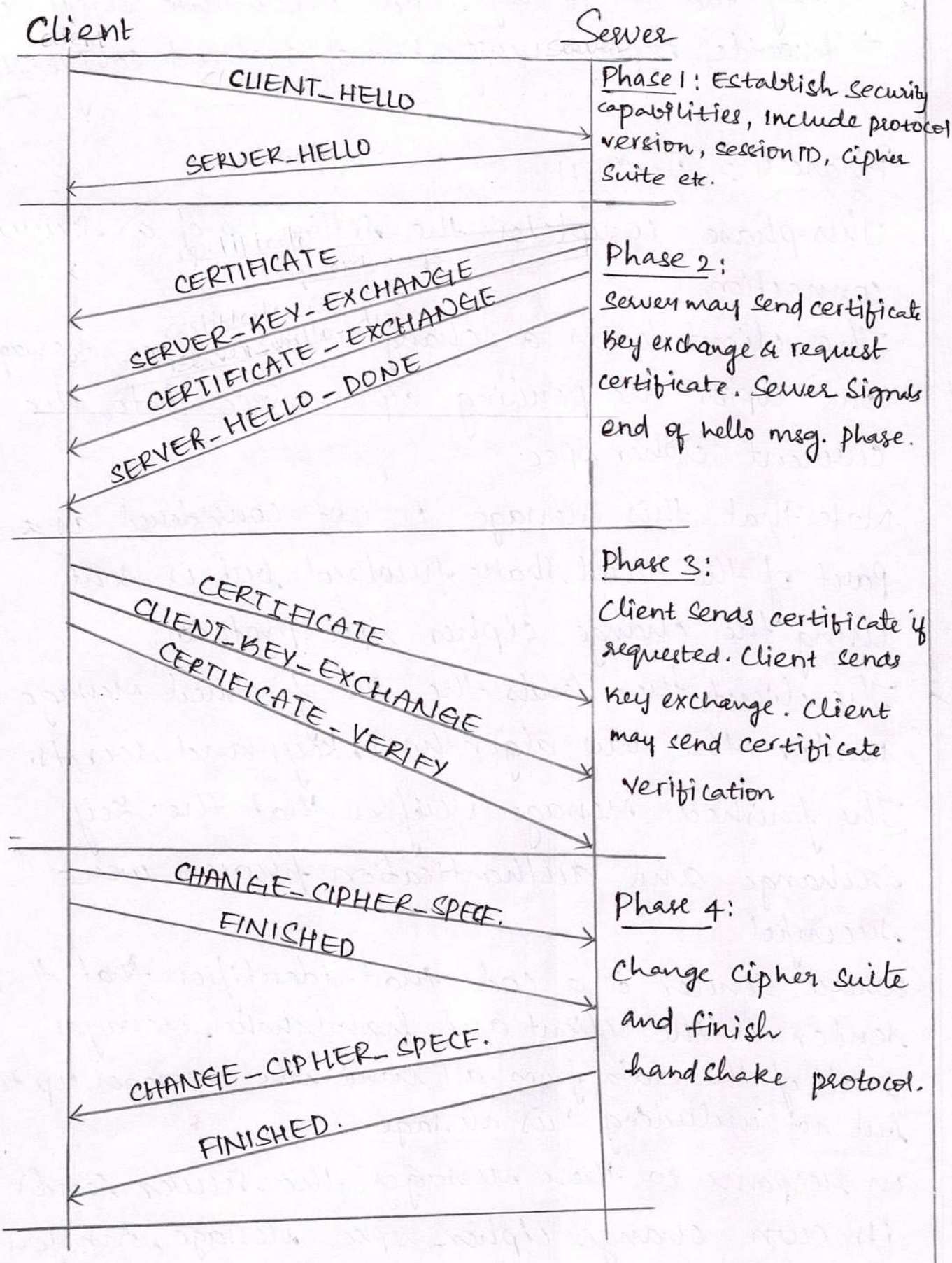
Note that this message is not considered as a part of the Hand shake protocol but is sent using the change cipher spec protocol.

The client then sends the the finished message under the new algorithms, keys and secrets.

The finished message verifies that the key exchange and authentication process were succesful

where sender is a code that identifies that the sender is the client and handshake_messages is all of the data from all handshake messages up to but not including this message

in response to these messages, the server sends its own change_cipher_spec message, transfers the pending to the current cipherspec, and sends its finished message



Hand Shake protocol action

→ Cryptographic Computations

i) Master Secret creation :-

The shared master ~~key~~ secret is a one time 48 byte value generated for this session by means of secure key exchange

The creation is in two stages

* Pre-master-secret is exchanged
* Master-secret is calculated by both parties

For pre-master-secret exchange, there are 2 possibilities

- RSA
- Diffie Hellman

ii) Generation of cryptographic parameters

Cipher specs require a client write MAC secret, a server write MAC secret, a client write key, a server write key, a client write IV and a server write IV which are generated from the master key

→ Transport layer Security (TLS)

- TLS is an IETF standardization initiative whose goal is to produce an internet standard version of SSL
- TLS is defined as a proposed internet standard in RFC 5246.

→ Version Number

The major version is 3 and minor version is 3

→ Message authentication code

There are 2 differences between the SSL V_3 and TLS MAC schemes:. The actual algorithm and the scope of the MAC calculation.

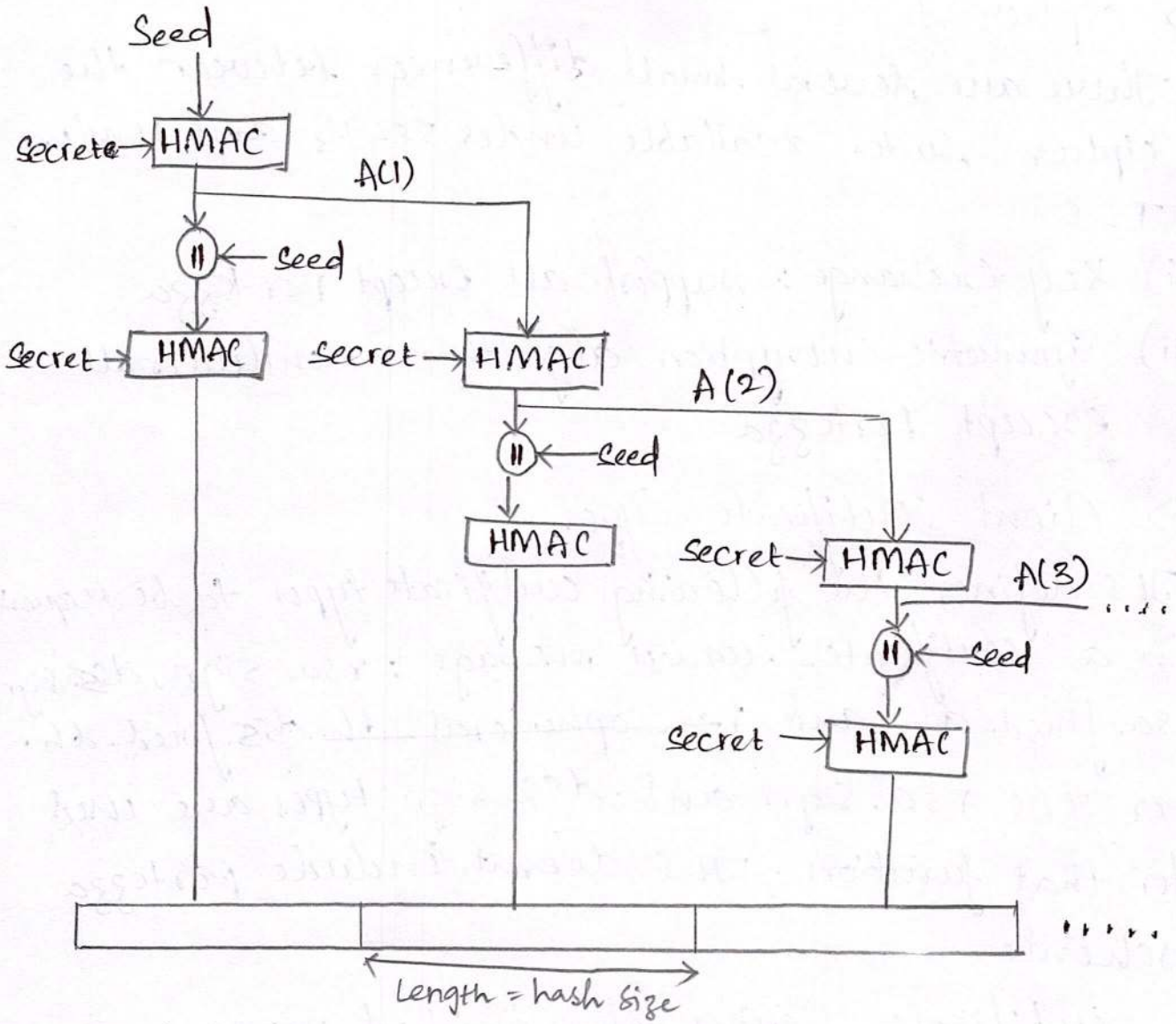
TLS makes use of the HMAC algorithm defined in RFC 2014.

SSL V_3 uses the same algorithm, except that the padding bytes are concatenated with the secret key rather than being XORed with the secret key padded to the block length

→ Pesudorandom functions

TLS makes use of a pesudorandom function referred to as PRF to expand secrets into blocks of data for purposes of key generation or validation

The objective is to make use of a relatively small shared secret value but to generate longer blocks of data in a way that is secure from the kinds of attacks made on Hash function



TLS function P_hash (Secret, Seed).

→ Alert codes

TLS supports all of the alert codes defined in SSLV3 with the exception of no_certificate. A no of additional codes are defined in TLS

i) record_overflow
ii) unknown_ca
iii) access_denied
iv) decode_error
v) protocol_version
vi) insufficient_security
vii) unsupported_extention
viii) internal_error
ix) decrypt_error
x) user_cancelled
xi) no_renegotiation

→ Cipher suites

There are several small differences between the cipher suites available under SSL V_3 and under TLS

i) Key Exchange :- supports all except Fortezza

ii) Symmetric encryption algorithms :- includes all except Fortezza

→ Client certificate types

TLS defines the following certificate types to be requested in a certificate_request message : rsa_sign, dss_sign, rsa_fixed_dh, and ~~rsa_ephemeral_dh, dss~~ fixed_dh. For TLS rsa_sign and dss_sign types are used for that function. TLS doesn't include Fortezza scheme

→ certificate_verify and finished messages

- In the TLS certificate_verify message, the MD5 and SHA-1 hashes are calculated only over handshake messages
- As with the finished messages in SSL V_3, the finished message in TLS is a hash based on the shared master_secret, the previous handshake messages and a label that identifies client or server.

→ Padding

The padding can be amount that results in total that is a multiple of the cipher's block length upto a maximum of 255 bytes

→ HTTPS

HTTP over SSL | TLS (HTTPS) refers to the combination of HTTP and SSL to implement secure communication between a web browser and a web server

The principal difference seen by user of a web browser is that URL address begin with https:// and with the port no 443

When HTTPS is used, the following elements are encrypted

- URL of the requested document
- contents of the document
- contents of the browser forms
- cookies sent from browser to server and from server to browser
- contents of HTTP header

→ connection initiation

- HTTPS, the agent acting as the HTTP client also acts as TLS client. The client initiates a connection to the server on the appropriate port and then

sends the TLS client_hello message to begin the TLS handshake. When TLS handshake is finished the client may initiate the first HTTP request. All HTTP data is to be sent as TLS application data.

There are three levels of awareness of a connection in HTTPS, an HTTP client requests a connection to an HTTPS server by sending a connection request to the next lowest level, next lowest layer is TCP, but it also may be TLS/SSL. At the level of TLS, a session is established between a TLS client and TLS server. This session can support one or more connections at a time

-> Connection closure

- It have "connection: close" in HTTP record
- At TLS level exchange it uses close_notify alerts
- we must handle TCP close connection before alert exchange

-> Secure shell (SSH)

It's a protocol for secure network communications designed to be relatively simple and inexpensive to implement

- The initial version, SSH1 was focused on providing a secure remote logon facility to replace TELNET and other remote logon schemes that provide no security.

SSH Protocol Stack:

SSH user Authentication Protocol: Authenticate client side user to server	SSH Connection Protocol: Multiplexes encrypted tunnel into several logical channel
SSH Transport layer Protocol: Provide server authentication, confidentiality, integrity. It may optionally provide compression	
TCP: It provide reliable connection-oriented end to end delivery.	
IP: It provide datagram delivery across multiple networks.	

- SSH also provides a more general client server capability and can be used for n/w functions as File transfer and Email.
- SSH is organised as 3 protocols that typically run on top of TCP.
  1. Transport layer protocol
  2. User Authentication Protocol
  3. Connection Protocol.

Transport Layer Protocol:

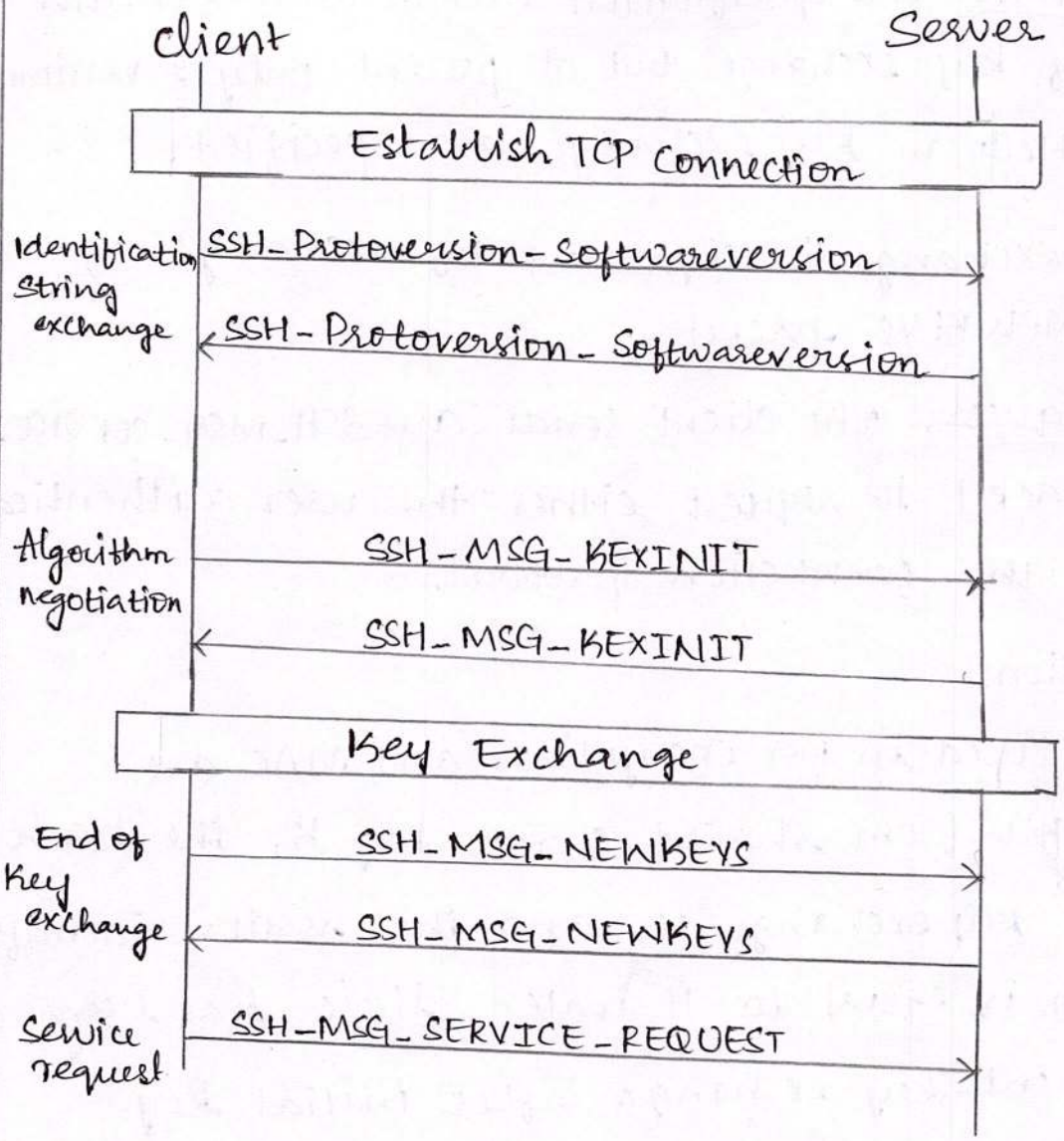
- Host keys Server authentication occurs at transport layer based on the server possessing public/private key pair.
- A Server may have multiple host keys using multiple different assymmetric encryption algorithms. Multiple host may share same host key.
- The two alternative trust model that can be used;
  ① Client has a local DB that associates each host name with corresp. public host key.
  ② The host name-to-key association is certified by a trusted CA. The client only knows CA root and can verify the validity of all host keys certified by accepted CA's.

packet exchange:

First the client establishes a TCP connection to the server. This is done by a TCP protocol and is not a part of transport layer protocol.

Each packet is in the following format:

1) Packet Length : Length of packet in bytes
2) Padding Length: Length of random padding field.
3) Payload : Useful contents of packet
4) Random Padding : Once an encryption alg. has been negotiated this field is added.
5) Message Authentication code: Contains MAC value (MAC)



The SSH transport layer packet exchange consist of a sequence of steps.

1. The Identification String exchange, begins with client sending a packet with identification string
2. Algorithm Negotiation, Each side sends an SSH-MSG-KEXINIT containing list of supported algo. in order of preference to the sender. There is one list for each type of cryptographic algorithm, which include key exchange encryption, MAC Algo., Compression algo.

3. Key exchange, The specification allows for alternative methods of key exchange but at present only 2 versions of Diffie-Hellman key exchange are specified.

*End of key exchange is signalled by exchange of SSH_MSG_NEWKEYS packet.

4. Service Request, The client sends an SSH_MSG_SERVICE_REQUEST packet to request either the user authentica-tion or the connection protocol.

Key Generation:

The keys used in for encryption and MAC are generated ~~for~~ from shared secret key K, the hash value from key exchange H, and the session identifi-er, which is equal to H unless there has been a subsequent key exchange after initial key exchange. The values are computed as,

* Initial IV client to server: HASH (K || H || "A" || session_id)
* Initial IV server to client: HASH [K || H || "B" || session_id)
* Encryption key client to server: HASH (K || H || "C" || session_id)
* Encryption Key Server to client: HASH (K || H || "D" || session_id)
* Integrity Key client to Server: HASH (K || H || "E" || session_id)
* Integrity Key server to client: HASH (K || H || "F" || session_id)

→ User authentication protocol

It provides the means by which the client is authenticated to the server

Message types and formats

Three types of messages are always used in the User authentication protocol. Authentication requests from the client have the format

byte	SSH_MSG_USERAUTH_REQUEST (50)
string	user name
string	service name
string	method name
....	Method specific fields

→ Message exchange

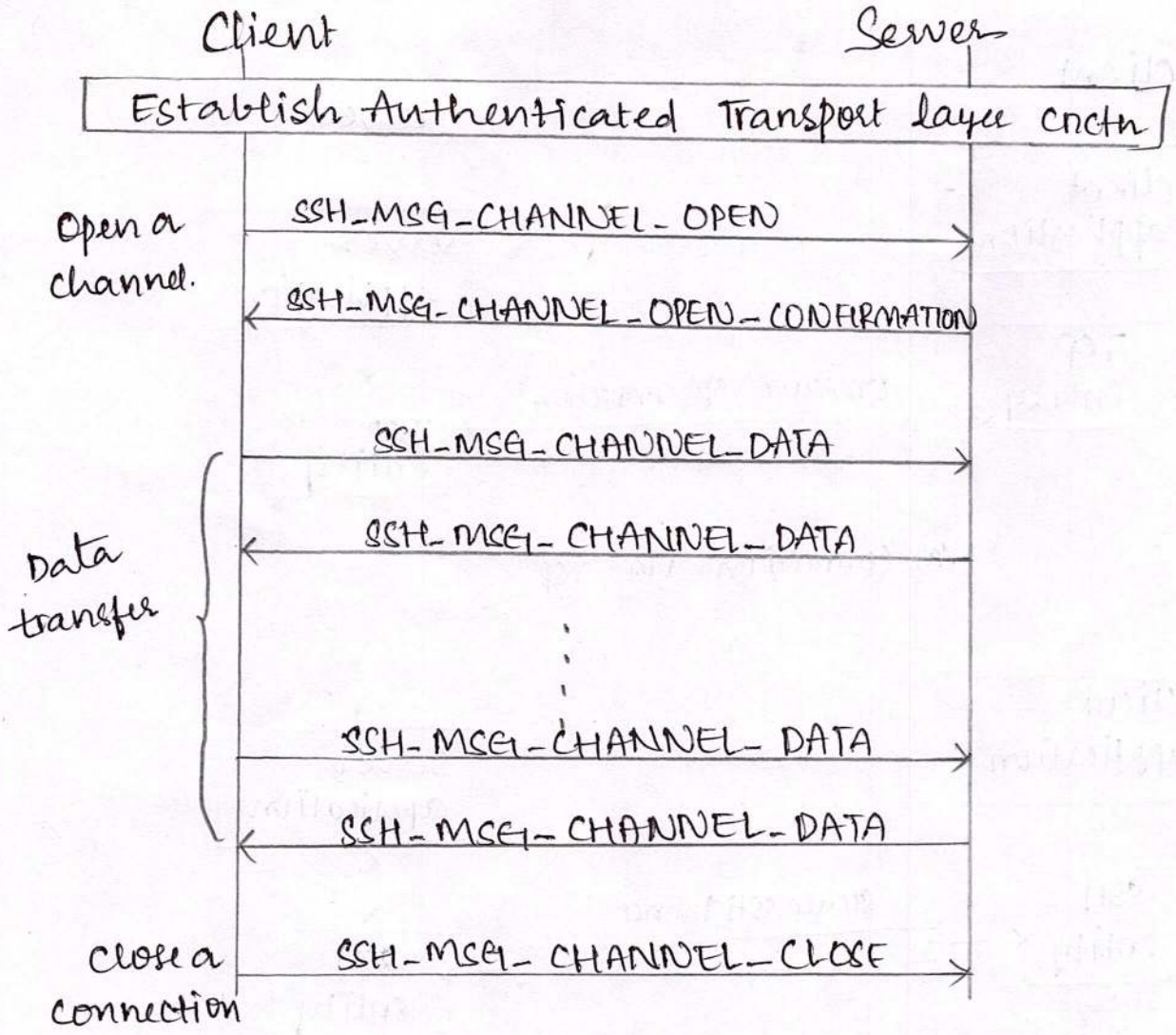
1. The client sends a SSH_MSG_USERAUTH_REQUEST with a requested method of none
2. The server checks to determine if the user name is valid if not sends the failure message
3. The server returns failure message with a list of one or more authentication methods to be used
4. The client selects one of the acceptable authentication methods and sends a SSH_MSG_USERAUTH_REQUEST with that method name and required method_spec fields.
5. If the authentication succeeds and more authentication methods are required
6. When all required authentication methods succeed, the server sends the success message and protocol is over

→ Authentication Methods
- Public Key
- Password
- Host based

→ Connection protocol
SSH connection protocol runs on top of the SSH Transport layer protocol and assumes that a secure authentication connection is in use
Secure authentication is reffered as a tunnel is used by the connection protocol to multiplex a number of logical channels

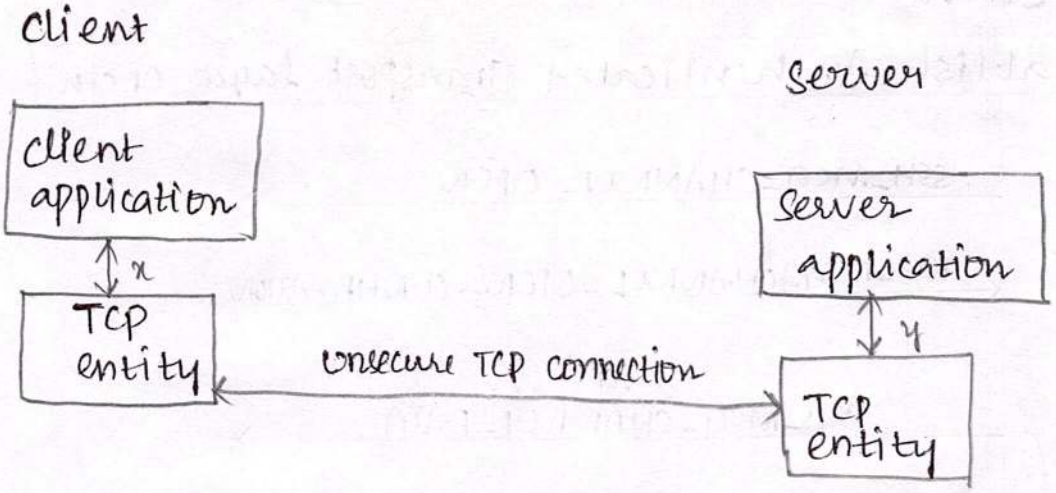
→ Channel mechanism : All types of communication using SSH, such as terminal session, are supported using seperate channels
- when either side wishes to open a connection / channel it allocates a local number for the channel and then sends Msg
- once a channel is open, data transfer is performed
- when either side wishes to close a channel it sends a close message

→ Channel types
- Session
- X11
- forwarded-tcpip
- direct-tcpip

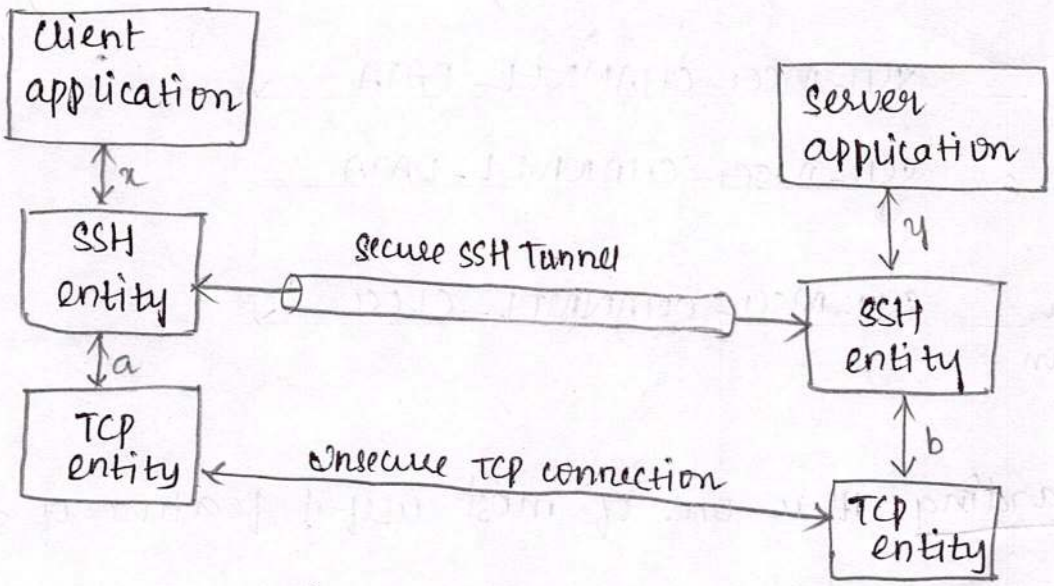


Port forwarding: It is one of most useful feature of SSH.

* It provide an ability to convert any insecure TCP connection into a secure SSH connection.
* This is also called as SSH tunnelling.
* A Port is an identifier of a user of TCP. So, any application that runs on top of TCP has port number.
* Incoming TCP traffic is delivered to application on basis of port number, Application may have multiple port numbers.



a) Connection via TCP



b) Connection via SSH tunnel.

→ Local forwarding :- allows the client to set up a "hijacker" process. This will interrupt selected app-level traffic and redirect it from an unsecured TCP connection to a secure SSH tunnel

→ Remote forwarding :- the user's SSH client acts on the server's behalf. The client receives traffic with a given destination port number, places the traffic on the correct port and sends it to the destination the user chooses.

Part-2 Wireless network security

→ Wireless security

Wireless networks, and the wireless devices that use them introduce a host of security problems over and above those found in wired networks

- Key factors contributing to the higher security risk of wireless networks are

i) Channel :- Wireless networking typically involves broadcast communications which is far more susceptible to eavesdropping and jamming than wired networks

ii) Mobility :- Wireless devices are portable and mobile

iii) Resources :- Some wireless devices such as smart phones have sophisticated OS but limited memory & processing resources

iv) accessibility :- Some wireless devices such as robots, may be left unattended in remote locations

→ Wireless network threats

i) Accidental association :- Company wireless LAN's may create overlapping transmission ranges

ii) Malicious association :- Wireless device is configured to apear to be a legitimate access point, enabling the operator to steal passwords from legitimate users

iii) Ad hoc networks :- These are peer-to peer n/w between wireless computers with no access point between them

iv) Non traditional networks :- They posses a security risk in terms of both eavesdropping and spoofing

v) Identity theft (MAC spoofing) :- This occurs when an attacker is able to eavsdrop on network traffic and identify the MAC address

vi) Man-in-the middle attacks :- It involves persuading a user and an access point to belive that they are talking to each other

vii) Denial-of-service (DoS) :- when attacker continuously bombards a wireless access point with protocol msg design to consume system resources

viii) Network injuction :- It attacks targets wireless access points that are exposed to non filtered network traffic such as routing protocol message.

→ Wireless Security measures

i) Securing Wireless transmission :- To deal with eavsdro-pping, 2 types of counter measures are appropriate

- Signal-hiding techniques :- we can make it more difficult for an attacker to locate their wireless access points by reducing signal strength to the lowest level
- Encryption :- Encryption of all wireless transmission is effective against eavesdropping to the extent that the encryption keys are secured

ii) Securing Wireless access points
The main threat in this is unauthorized access to the network. The principal approach for preventing such access is the IEEE 802.1x standard for port based n/w access control

iii) Securing wireless networks :-
1. Use encryption
2. Use antivirus and anti-spy software
3. Turn off identifier broadcasting
4. Change the identifier on your router from the default
5. Change your router's pre-set password for administration
6. Allow only specific computers to access your wireless network

→ Mobile device security

Network security was based upon clearly defined perimeters that seperated trusted internal networks from the untrusted internet.

- Networks must accomodate the following in the Organisation

i) Growing use of new devices :- Employees are allowed to use a combination of end point devices as a part of their day to day activity

ii) cloud-based appliciation :- applications can run anywhere on traditional physical servers on mobile virtual servers, or in the cloud

iii) De-perimeterization :- multitude of network parameters have become dynamic as they must adapt to various environmental conditions

iv) External business requirements :- It must provide business partners network access using various devices from a multitude of locations

-> Security threats

Mobile devices need additional, specialized protection measures

- Security concerns for mobile devices

i) Lack of physical security controls : The security policy for mobile devices may be stolen or atleast accessed by malicious party. The threat is twofold : A malicious party may attempt to recover sensitive data from the device or may use it to gain access to org resource

ii) Use of untrusted mobile devices :- The org must assume that the personal phones of employees are not trustworthy

iii) Use of untrusted Networks :- Traffic that includes an off-premises segment is potentially susceptible to eavesdroping or man-in the middle types of attacks

iv) Use of applications created by unknown parties :- They may posses the obvious risk of installing malicious software

v) interaction with other systems :- There is a chance of ~~stor~~ risk of the org data.

vi) Use of untrusted content :- They may direct to malicious web sites

vii) Use of location service :- It creates security risks. An attacker can use the location info to determine the device.

→ Mobile device security strategy

Mobile device security strategies are divided into 3 categories

i) device security
ii) client/server traffic security
iii) Barrier security

→ Device security :- A no of organizations will supply mobile devices for employee use and preconfigure those devices to conform to the enterprise security policy The organization should configure the device with security controls they are

- Enable auto-lock
- Enable password or PIN protection
- Avoid using auto complete features
- Ensure that SSL protection is enabled
- Make sure S/w and OS are up to date
- install antivirus software
- sensitive data should be prohibited from storage
- staff should have the ability to access devices
- The org may prohibit all installation of third party applications
- deal with the threats of untrusted content
- To counter the threats of malicious use of location services

→ Traffic Security :- It's based on the usual mechanism for encryption and authentication

- All traffic should be encrypted and travel by secure means such as SSL or IPV6
- Virtual private n/w (VPNs) can be configured so that all traffic between the mobile device and the org n/w is via VPN

→ Barrier Security :- The org should have a security mechanism to protect the n/w from unauthorized access

The security strategy can also include firewall policies specific to mobile device traffic

→ IEEE 802.11 Wireless Lan Overview

IEEE 802 is a committee that has developed standards for a wide range of LAN.

- Terminology

i) access point (AP) :- Any entity that has station functionality and provides access to the distribution via wireless medium

ii) Basic Services (BSS) :- Set of stations controlled by single coordinate function

iii) coordination function :- logical fn that determines when a station operating within a BSS is permitted to transfer & receive

iv) Distribution System :- A system used to interconnect a set of BSS s and integrated LAN to create an ESS

v) Extended Service Set (ESS) :- A set of one or more interconnected BSSs and integrated LANs that appear as a single BSS to the LLC layer at any station

vi) MAC protocol data unit (MPDU) :- The unit of data exchanged b/w 2 peer MAC entities using the services of physical layer

vii) MAC service data unit :- info that is delivered as a unit b/w MAC users

viii) Station :- Any device that contains an IEEE 802.11 conformant MAC & physical layer

-> The Wi-Fi Alliance

- 802.11b first broadly accepted standard
- Wireless ethernet compatibility Alliance (WECA) industry consortium formed 1999
  - To assist interoperability of products
  - renamed Wi-Fi Alliance
  - created a test suite to certify interoperability
  - initially for 802.11b, later extended to 802.11g
  - concerned with a range of WLAN s markets, including entriprise, home, and hotspots.

→ IEEE 802 protocol architecture

	General fn	Specific fn
logical link control	flow control error control	
Medium access control	assemble data into frame addressing, access error detection	Reliable data delivery wireless access protocol
Physical	Encoding/decoding of signals bit transmission/ reception Transmission Medium	frequency band. detection wireless signal encoding

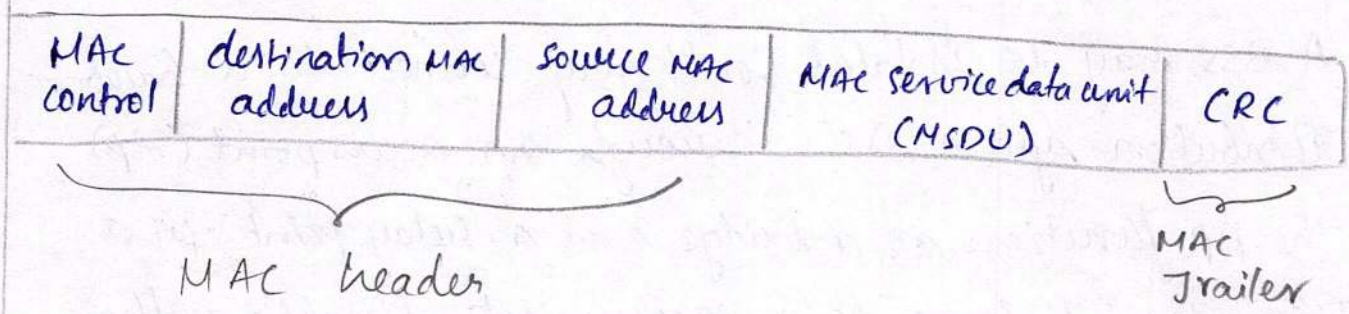
→ Physical layer

- Lowest layer of the IEEE 802 reference model
- It includes fn as encoding / decoding of signals and bit transmission / reception . It includes specification of transmission medium . It also defines frequency band and antenna characteristics

→ Media Access control :- All LANs consist of collections of devices that share the network transmission capacity
The function of media access control is controlling access to the transmission medium is needed to provide an orderly and efficient use of that capacity
- The MAC layer receives data from a higher-layer protocol, typically the logical link control (LLC) layer in the form of data known as MAC service data unit.

- MAC layer performs following functions
- On transmission, assemble data into a frame, known as a MAC protocol data unit with add and error detection fields
- On reception, disassemble frame and perform add recognition and error detection
- Govern acces to the LAN transmission medium

General MPDU Format



- MAC control:- This field contains any protocol control information needed for functioning of the MAC protocol
- Destination MAC Address :- The destination add on the LAN for this MPDU
- Source MAC address :- The source physical add on the LAN for this MPDU
- MAC Service data unit :- The data from the next higher layer
- CRC :- The cyclic redundancy check field also known as the frame check sequence (FCS) field. This is an error-detecting code, such as that which is used in other data link control protocols. The CRC is calculated based on the bits in the entire MPDU.

→ Logical link control :- The data link protocol entity is responsible not only for detecting errors using the CRC, but for recovering from those errors by retransmitting damaged frames

→ IEEE 802.11 Network components and Architecture model

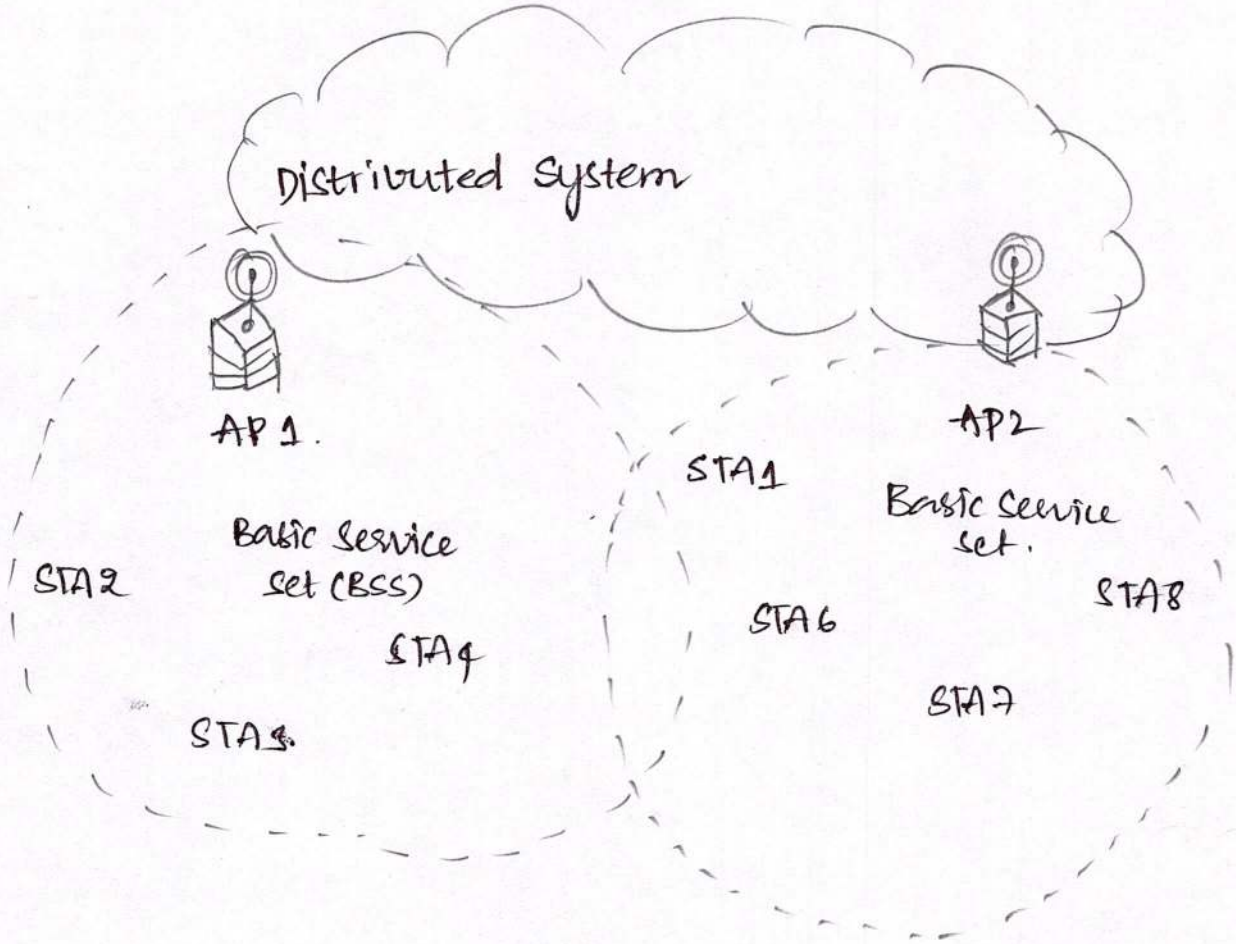
The smallest building block of a wireless LAN is a BSS (basic service set), which consists of wireless stations executing the same MAC protocol and competing for access to the same shared wireless medium

A BSS may be isolated, or it may connect to a backbone distribution system (DS) through an access point (AP)

The AP functions as a bridge and a relay point. In a BSS, client stations don't communicate directly with another, rather if one station want to communicate with another in the same BSS, the MAC frame is sent from the originating station to the AP and then from the AP to the destination station.

When all the stations in the BSS are mobile stations that communicate directly with one another, the BSS is called Independent BSS (IBSS) it's a ~~basic~~ adhoc n/w

An Extended service set (ESS) consists of two or more basic service sets interconnected by a distribution system. It appears as a single logical LAN to logical link control level



IEEE 802.11 Extended Service set.

→ IEEE 802.11 Services

IEEE 802.11 defines 9 services that need to be provided by the wireless LAN to achieve functionality equivalent to that which inherits to wired LAN

Service	Provider	Used to support
Association	distribution system	MSDU delivery
Authentication	Station	LAN access & security
De authentication	station	LAN access & security
Disassociation	distribution system	MSDU delivery
distribution	distribution system	MSDU delivery
integration	distribution system	MSDU delivery
MSDU delivery	Station	LAN access and security
reassociation	distribution system	MSDU delivery

1. The service provider can be either the station or the DS. Station services are implemented in every 802.11 station, including AP stations. Distribution services are provided b/w BSS these services may be implemented in an AP or in another special-purpose device attached to the distribution system
2. Three of the services are used to controll IEEE 802.11 LAN access and confidentiality. Six of the services are used to support delivery of MSDU's b/w stations

-> Distribution of messages within a DS

The 2 services involved with the distribution of messages within a DS are distribution and integration. Distribution is the primary service used by stations to exchange MPDU's when the MPDU must traverse the DS to get from a station in one BSS to a station in another BSS

The integration service enables transfer of data b/w a station on an IEEE 802.11 LAN and a station on an integrated IEEE 802.x LAN. The term integrated refers to a wired LAN ~~and a station on an integrated IEEE 802.x LAN~~ That is physically connected to the DS and whose stations may be logically connected to an IEEE 802.11 LAN via the integration service

-> Authentication-related services

The primary purpose of the MAC layer is to transfer MSDU b/w MAC entities; this purpose if fullfilled by the distributed service For that service to function, it requires information about stations within the ESS that is provided by the association-related services

The standard defines 3 transition types based on mobility

1) No transition :- A station of this type is either stationary or moves only within the direct comm range of

communicating stations of a single BSS

ii) BSS transition :- This is defined as a station movement from one BSS to an other BSS within the same ESS. In this case, delivery of data to the station requires that the addressing capability be able to recognize the new loc of the station

iii) ESS transition :- This is defined as a station movement from a BSS in one ESS to a BSS within another ESS. This case is supported only in the sense that the station can move

- To deliver a Msg within a DS, the distribution server must know where the destination station is located
DS need to know the identity of the AP to which the Msg should be delivered in order for the Msg to reach destination
To meet this requirement there are 3 services

i) Association :- Establish an initial association btw a station & an AP Before a station can transmit or recieve frames on a wireless LAN, its identity and address must be know. For this purpose a station must establish an association with an AP within a particular BSS

ii) Re association :- Enables an association to be transferred from one AP to another allowing a mobile station frm one BSS to another

iii) Dis association :- A notification from either station or an AP that an existing association is terminated

→ IEEE 802.11i Wireless lan security

There are 2 characteristics of a wired LAN that are not inherit in a wireless LAN

1. In order to transmit over a wired LAN, a station must be physically connected to the LAN.
   in wireless LAN, any station within radio range of the other devices on the LAN can transmit
2. Similarly, in order to receive a transmission frm a station that is part of a wired LAN, the receiving station also must be attached to the wired LAN
   In wireless LAN any station within audio range can receive.

→ IEEE 802.11i services.

- authentication :- A protocol is used to define an exchange b/w a user and an AS that provides mutual authentication and generates temporary keys to be used b/w the client and the AP over the wireless link
- access control : This fn enforces the use of the authentication fn, routes the message properly and facilitates key exchange it can work with authentication protocols
- privacy with msg integrity :- MAC-level data are encrypted along with a message integrity code that ensures that have not been altered

Robust Security Network (RSN)

Access Control	Authentication and key generation	Confidentiality, Data origin authentication & Integrity & Replay protection	
IEEE 802.1 Port based Access control	Extensible Authentication Protocol (EAP)	TKIP	CCMP

a) Services and Protocols.

Robust Security Network (RSN)

Confidentiality			Integrity and data origin Authentication				Key Generation	
TKIP (RC4)	CCM (AES-CTR)	NIST key wrap	HMAC-SHA1	HMAC MD5	TKIP (Michael MIC)	CCM (AES-CBC-MAC)	HMAC SHA1	RFC 1750

b) Cryptographic algorithms.

IEEE 802.11i phases of Operation:

This is broken down into 5 phases:

1. Two wireless stations in same BSS communicating via Access point for that BSS
2. Two wireless stations in different BSSs communicating via their resp. APs across distribution System

3. Two wireless stations in same ad hoc IBSS communi-cating directly with each other.

4. A wireless station communicating with an end station on wired n/w via its AP and the distribution system

* IEEE 802.11i security is concerned with secure communication between STA and its AP.

The five phases of operation for an RSN and maps them to network components involved. One new component is authentication server (AS). The rectangles indicate exchange of sequences of MPDUs. The 5 phases are defined as follows.

* Discovery: An AP uses messages called Beacons and Probe Responses to advertise its IEEE 802.11i security policy. The STA uses these to identify an AP for a WLAN with which it wishes to communicate.

* Authentication:

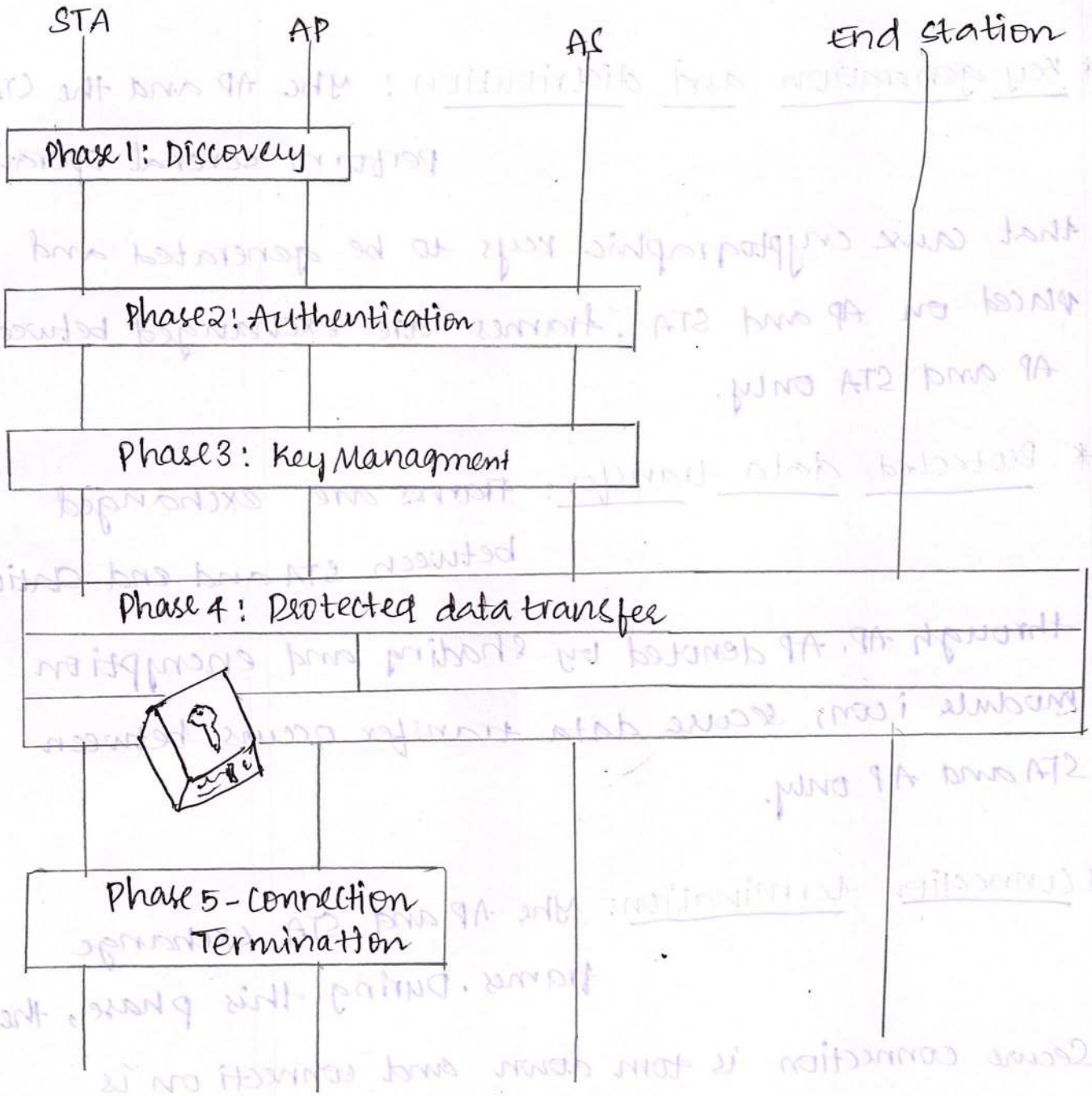
During this phase, the STA and AS prove their identities to each other. The AP blocks non-authentication traffic between STA and AS until

the authention transaction is sucessfull. AP doesn't participate in authentication transaction.

* Key generation and distribution : The AP and the STA perform several operations that cause cryptographic keys to be generated and placed on AP and STA . frames are exchanged between AP and STA only.

* Protected data transfer: Frames are exchanged between STA and end station through AP. AP denoted by shading and encryption module icon, secure data transfer occurs between STA and AP only.

* Connection termination: The AP and STA exchange frames. During this phase, the secure connection is torn down and connection is restored to original state.



IEEE 802.11i phases of operation.

*Discovery phase

The purpose of discovery phase is for an STA and an AP to recognise each other, agree on a set of security capabilities and establish an association for future communication using those security capabilities.

SECURITY CAPABILITIES : During this phase, the STA and AP decide on specific techniques in the following areas:

- confidentiality and MPDU integrity protocols for protecting unicast traffic
- Authentication method
- Cryptography key management approach.

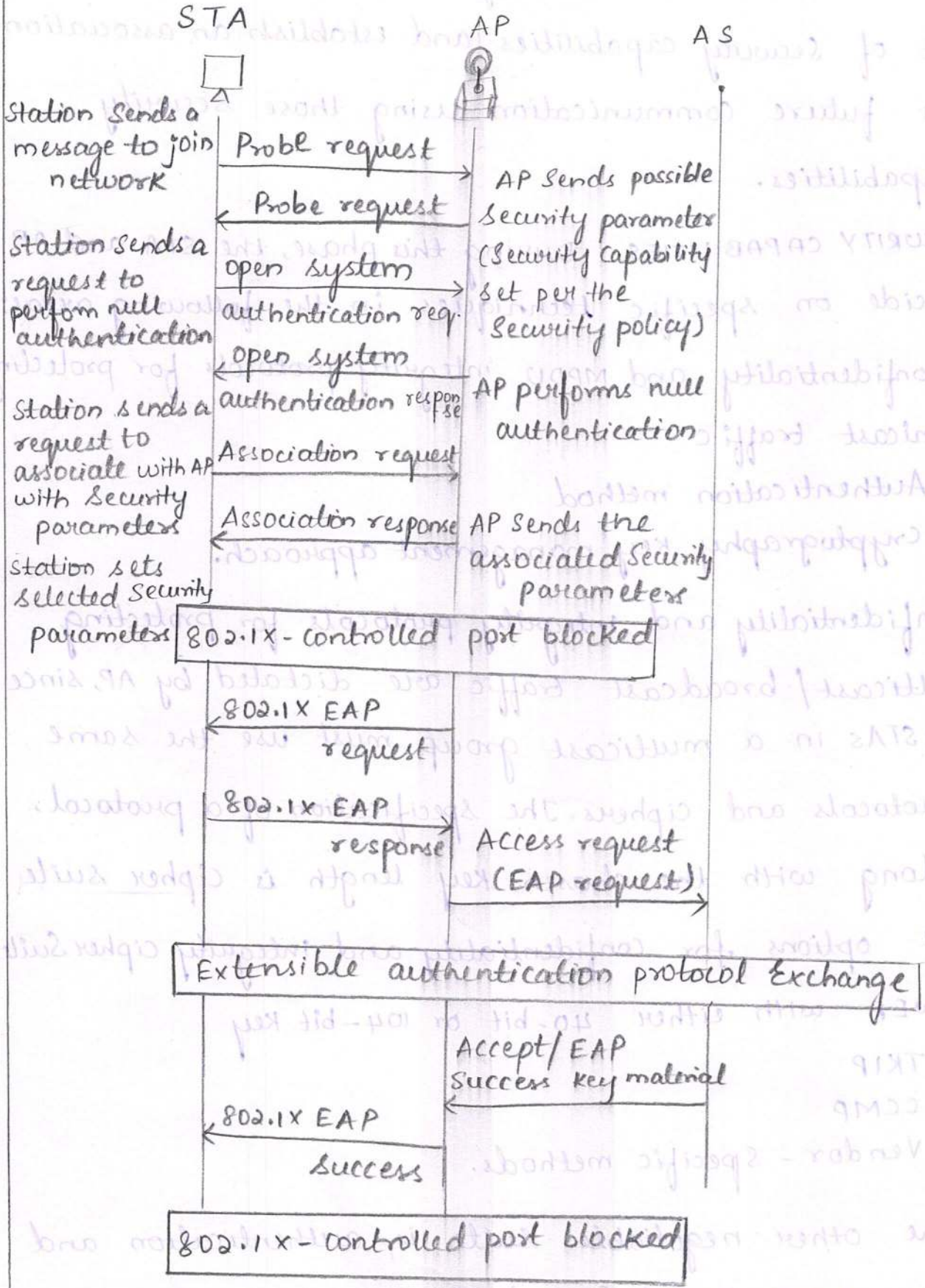
Confidentiality and integrity protocols for protecting multicast/broadcast traffic are dictated by AP, since all STAs in a multicast group must use the same protocols and ciphers. The specification of a protocol, along with the chosen key length is cipher suite

The options for confidentiality and integrity cipher Suite:

- WEP with either 40-bit or 104-bit key
- TKIP
- CCMP
- Vendor-specific methods.

The other negotiable suite is authentication and key management (AKM). The possible AKM suites are

- IEEE 802.1X
- Pre-shared Key
- Vendor-specific methods



IEEE 802.11i phases of operation: capability discovery, ~~Authentication~~ and ~~Association~~

MPDU Exchange: The discovery phase consists of three Exchanges.

Network and Security capability discovery: During this exchange, STA s discover the existence of a network with which to communicate. The AP either periodically broadcasts its Security capabilities, indicated by RSN IE in a specific channel through Beacon frame; or responds to a station's probe request through a probe Response frame.

Open System authentication: The purpose of this frame sequence, which provides no security, it simply maintains backward compatibility with IEEE 802.11 state machine. In essence, 2 devices (STA and AP) simply exchange identifiers.

Association: The purpose of this stage is to agree on a set of Security capabilities to be used. The STA then sends an Association Request frame to AP. In this frame, the STA specifiers one set of matching capabilities [one authentication and Key management Suite, one pairwise cipher suite, and one group-Key cipher Suite] from among those advertised by the AP. If there is no match in capabilities between AP and STA, the AP refuses the Association Request. The STA blocks it too, in case it has associated with a rogue AP or Someone is inserting frames illicitly.

*Authentication Phase

Authentication phase enables mutual authentication between an STA and an authentication Server (AS) located in DS. It is designed to allow only authorized stations to use the network and to provide the STA with assurance that it is communicating with a legitimate network.

ACCESS CONTROL APPROACH: The authentication protocol that is used, the Extensible Authentication Protocol (EAP) is defined in the IEEE 802.1X standard. IEEE 802.1X uses the term supplicant, authenticator and authentication Server AS is typically a separate device on the wired side of the network but could also reside directly on the authenticator.

→ 802.1X uses the concept of controlled and uncontrolled ports. Ports are logical entities defined within the authenticator and refer to physical network connections.

- An uncontrolled port allows the exchange of PDUs between the supplicant and other AS, regardless of the supplicant.
- A controlled port allows the exchange of PDUs between a supplicant and other systems on LAN only if the current state of the supplicant authorizes such an exchange.

MPDU Exchange:

- Connect to AS : The STA sends a request to its AP for connection to the AS. The AP acknowledges this request and sends an access request to AS.
- EAP exchange : This exchange authenticates the STA and AS to each other. A number of alternative exchanges are possible
- Secure Key delivery : Once authentication is established, AS generates a master session Key (MSK) also known as Authentication, Authorization and Accounting (AAA) Key and sends it to STA. All cryptographic keys needed by STA for secure communication with its AP are generated from this MSK.

EAP exchange : The message flow between STA and AP employs the EAP over LAN (EAPOL) protocal, and the message flow between AP and AS uses the Remote authentication Dial in user Service (RADIUS) Protocal. Authentication exchange using EAPOL and RADIUS are as follows:

1. The EAP exchanges begins with AP issuing an EAP-Request / Identity frame to STA.
2. The STA replies with an EAP-Response / Identity frame, which AP receives over the uncontrolled port. The packet is then encapsulated in RADIUS over EAP and passed on to RADIUS server as a RADIUS-Access-Request packet.

3. The AAA Server replies with a RADIUS-Access-challenge packet, which is passed on to the STA as an EAP-Request. This request is of the appropriate authentication type and contains relevant challenge information.
4. The STA formulates an EAP-Response message and sends it to AS. The response is transmitted by AP into a Radius-Access-Request with the response to the challenge as a data field. Steps 3 and 4 may be repeated multiple times, depending on EAP method in use. For TLS tunneling methods, it is common for authentication to require 10 to 20 round trips.
5. The AAA server grants access with a Radius-Access-Accept packet. The AP issues an EAP-success frame. The controlled port is authorized, and user may begin to access the network.

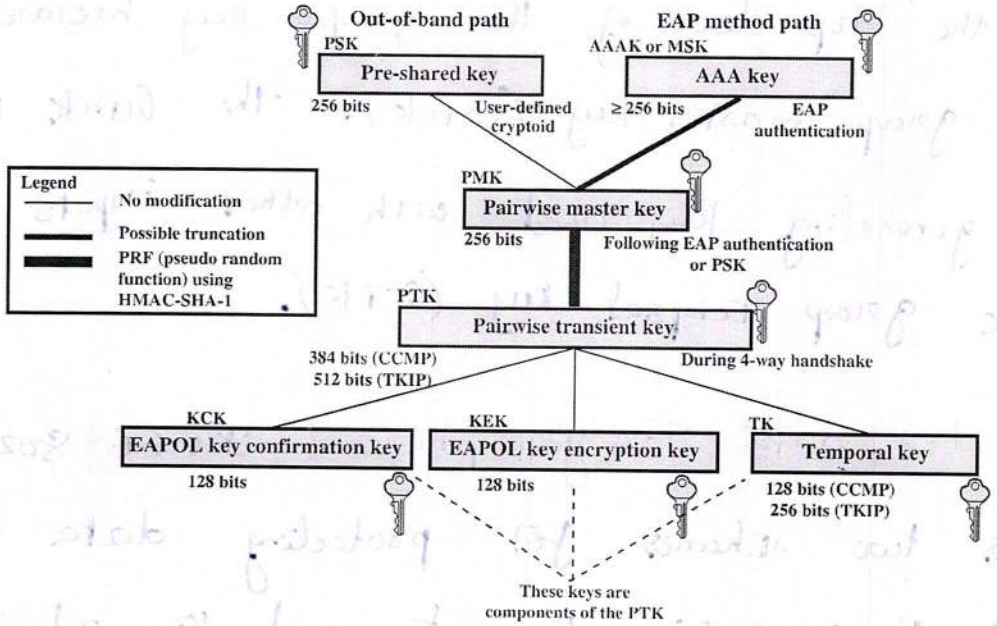
Key Management phase: During the key management phase, a variety of cryptographic keys are generated and distributed to STAs. There are two types of keys: pairwise keys used for communication between an STA and an AP and group keys for multicast communication.

pairwise keys: At the top level of the hierarchy are two possibilities. A preshared key (PSK) is a secret key shared by AP and STA and installed in some fashion outside the scope of IEEE 802.11i. The other alternative is the master session key (MSK) also known as AAAK, which is generated using the IEEE 802.1x protocol.

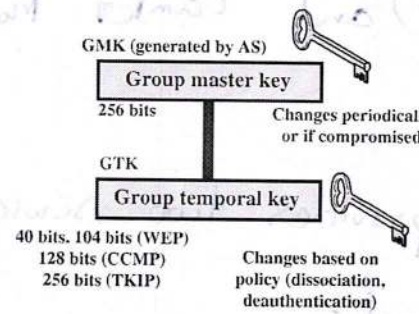
The pairwise master key (PMK) is derived from the master key. The PMK is used to generate the pairwise transient key (PTK) which in fact consists of three keys to be used for communication between an STA and AP after they have been mutually authenticated. To derive PTK, HMAC-SHA-1 function is applied to PMK, the MAC addresses of STA & AP and nonces generated when needed.

the three parts of PTK are as follows

1. EAP over LAN (EAPOL) key confirmation key (EAPOL-KCK): supports integrity & data origin authenticity of STA to AP control frames during operational setup of an RSN. It also performs an access control function
2. EAPOL-key Encryption key (EAPOL-KEK): protects the confidentiality of keys and other data during some RSN association procedures.
3. Temporal key (TK): provides the actual protection for user traffic.



(a) Pairwise key hierarchy



(b) Group key hierarchy

Figure 18.9 IEEE 802.11i Key Hierarchies

operational setup of an RSN. It also performs an access control function: proof-of-possession of the PMK. An entity that possesses the PMK is authorized to use the link.

- **EAPOL Key Encryption Key (EAPOL-KEK):** Protects the confidentiality of keys and other data during some RSN association procedures.
- **Temporal Key (TK):** Provides the actual protection for user traffic.

GROUP KEYS Group keys are used for multicast communication in which one STA sends MPDU's to multiple STAs. At the top level of the group key hierarchy is the **group master key (GMK)**. The GMK is a key-generating key used with other inputs

Group Keys : these are used for multicast communication in which one STA sends MPDU's to multiple STAs. At the top level of the group key hierarchy is the group master key (GMK). the GMK is a key generating key used with other inputs to derive group temporal key (GTK).

Protected data transfer phase : IEEE 802.11i defines two schemes for protecting data transmitted in 802.11 MPDUs: the temporal key integrity protocol (TKIP) and counter mode-CBC MAC protocol (CCMP).

TKIP : it provides two services.

message integrity
Data confidentiality.

CCMP : it also provides two services, message integrity and data confidentiality.

Table 18.3 IEEE 802.11i Keys for Data Confidentiality and Integrity Protocols

Abbreviation	Name	Description / Purpose	Size (bits)	Type
AAA Key	Authentication, Accounting, and Authorization Key	Used to derive the PMK. Used with the IEEE 802.1X authentication and key management approach. Same as MMSK.	≥ 256	Key generation key, root key
PSK	Pre-shared Key	Becomes the PMK in pre-shared key environments.	256	Key generation key, root key
PMK	Pairwise Master Key	Used with other inputs to derive the PTK.	256	Key generation key
GMK	Group Master Key	Used with other inputs to derive the GTK.	128	Key generation key
PTK	Pair-wise Transient Key	Derived from the PMK. Comprises the EAPOL-KCK, EAPOL-KEK, and TK and (for TKIP) the MIC key.	512 (TKIP) 384 (CCMP)	Composite key
TK	Temporal Key	Used with TKIP or CCMP to provide confidentiality and integrity protection for unicast user traffic.	256 (TKIP) 128 (CCMP)	Traffic key
GTK	Group Temporal Key	Derived from the GMK. Used to provide confidentiality and integrity protection for multicast/ broadcast user traffic.	256 (TKIP) 128 (CCMP) 40,104 (WEP)	Traffic key
MIC Key	Message Integrity Code Key	Used by TKIP's Michael MIC to provide integrity protection of messages.	64	Message integrity key
EAPOL-KCK	EAPOL-Key Confirmation Key	Used to provide integrity protection for key material distributed during the 4-Way Handshake.	128	Message integrity key
EAPOL-KEK	EAPOL-Key Encryption Key	Used to ensure the confidentiality of the GTK and other key material in the 4-Way Handshake.	128	Traffic key / key encryption key
WEP Key	Wired Equivalent Privacy Key	Used with WEP.	40,104	Traffic key

to derive the **group temporal key (GTK)**. Unlike the PTK, which is generated using material from both AP and STA, the GTK is generated by the AP and transmitted to its associated STAs. Exactly how this GTK is generated is undefined. IEEE 802.11i, however, requires that its value is computationally indistinguishable from random. The GTK is distributed securely using the pairwise keys that are already established. The GTK is changed every time a device leaves the network.

Pairwise Key Distribution The upper part of Figure 18.10 shows the MPDU exchange for distributing pairwise keys. This exchange is known as the **4-way handshake**. The STA and AP use this handshake to confirm the existence of the PMK, verify the selection of the cipher suite, and derive a fresh PTK for the following data session. The four parts of the exchange are as follows.

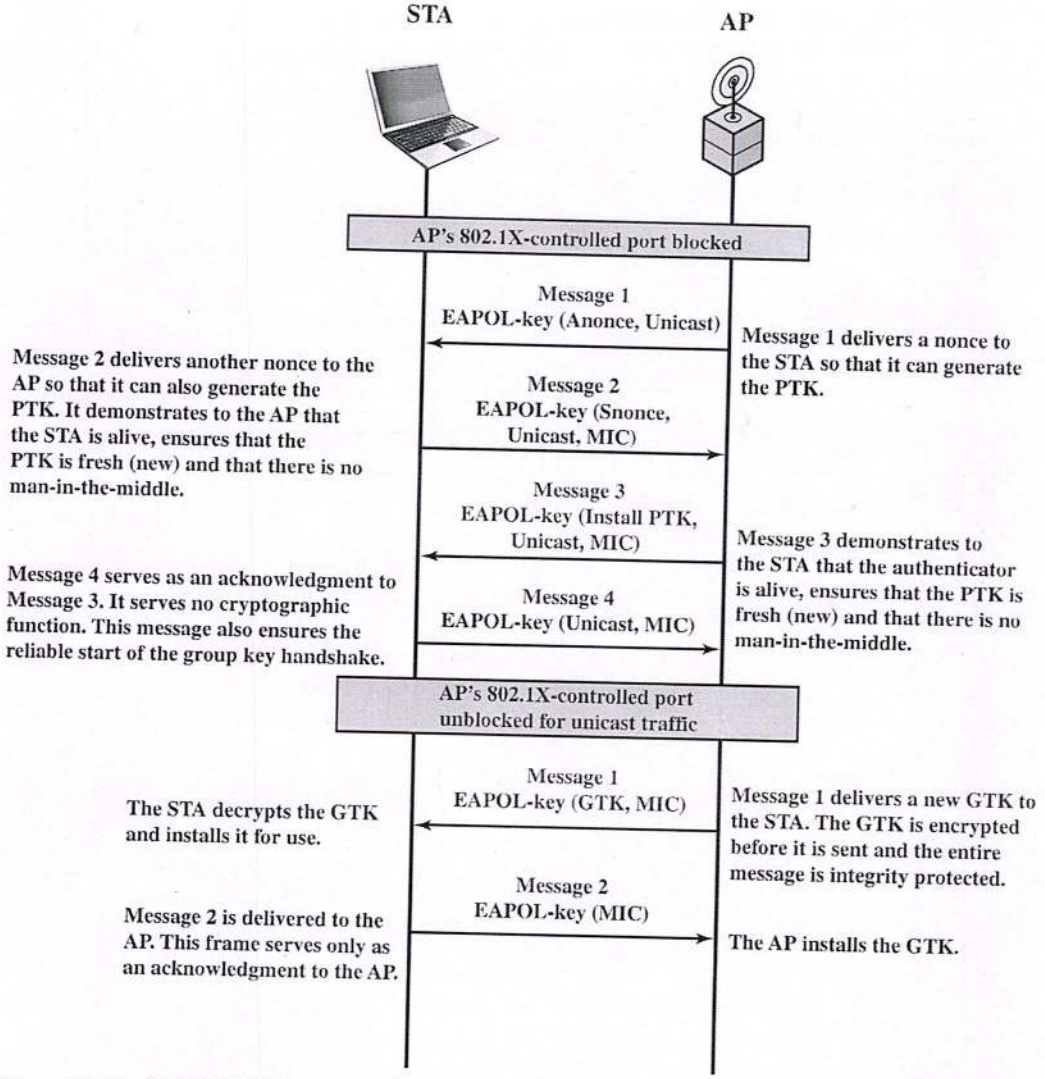


Figure 18.10 IEEE 802.11i Phases of Operation: Four-Way Handshake and Group Key Handshake

- **AP → STA:** Message includes the MAC address of the AP and a nonce (Anonce)
- **STA → AP:** The STA generates its own nonce (Snonce) and uses both nonces and both MAC addresses, plus the PMK, to generate a PTK. The STA then sends a message containing its MAC address and Snonce, enabling the AP to generate the same PTK. This message includes a message integrity code (MIC)[2] using HMAC-MD5 or HMAC-SHA-1-128. The key used with the MIC is KCK.

[2]While *MAC* is commonly used in cryptography to refer to a Message Authentication Code, the term *MIC* is used instead in connection with 802.11i because *MAC* has another standard meaning, Media Access Control, in networking.

- **AP → STA:** The AP is now able to generate the PTK. The AP then sends a message to the STA, containing the same information as in the first message, but this time including a MIC.
- **STA → AP:** This is merely an acknowledgment message, again protected by a MIC.

Group Key Distribution For group key distribution, the AP generates a GTK and distributes it to each STA in a multicast group. The two-message exchange with each STA consists of the following:

- **AP → STA:** This message includes the GTK, encrypted either with RC4 or with AES. The key used for encryption is KEK, using a key wrapping algorithm (as discussed in Chapter 12). A MIC value is appended.
- **STA → AP:** The STA acknowledges receipt of the GTK. This message includes a MIC value.

Protected Data Transfer Phase

IEEE 802.11i defines two schemes for protecting data transmitted in 802.11 MPDUs: the Temporal Key Integrity Protocol (TKIP), and the Counter Mode-CBC MAC Protocol (CCMP).

TKIP TKIP is designed to require only software changes to devices that are implemented with the older wireless LAN security approach called Wired Equivalent Privacy (WEP). TKIP provides two services:

- **Message integrity:** TKIP adds a message integrity code (MIC) to the 802.11 MAC frame after the data field. The MIC is generated by an algorithm, called Michael, that computes a 64-bit value using as input the source and destination MAC address values and the Data field, plus key material.
- **Data confidentiality:** Data confidentiality is provided by encrypting the MPDU plus MIC value using RC4.

The 256-bit TK (Figure 18.9) is employed as follows. Two 64-bit keys are used with the Michael message digest algorithm to produce a message integrity code. One key is used to protect STA-to-AP messages, and the other key is used to protect AP-to-STA messages. The remaining 128 bits are truncated to generate the RC4 key used to encrypt the transmitted data.

For additional protection, a monotonically increasing TKIP sequence counter (TSC) is assigned to each frame. The TSC serves two purposes. First, the TSC is included with each MPDU and is protected by the MIC to protect against replay attacks. Second, the TSC is combined with the session TK to produce a dynamic encryption key that changes with each transmitted MPDU, thus making cryptanalysis more difficult.

CCMP CCMP is intended for newer IEEE 802.11 devices that are equipped with the hardware to support this scheme. As with TKIP, CCMP provides two services:

- **Message integrity:** CCMP uses the cipher block chaining message authentication code (CBC-MAC), described in Chapter 12.

- Data confidentiality : CCMP uses the CTR block cipher mode of operation with AES for encryption.

*IEEE 802.11i Pseudorandom Function

At a number of places in IEEE 802.11i scheme, a pseudorandom function (PRF) is used. Best security practice dictates that different pseudorandom number streams be used for these different purposes like to generate nonces, to expand pairwise keys and to generate GTK.

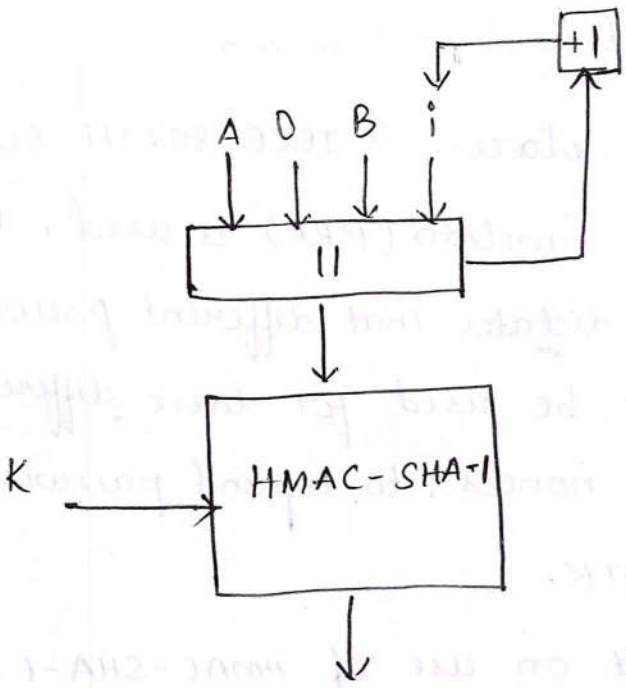
→ The PRF is built on use of HMAC-SHA-1 to generate a pseudorandom bit stream. SHA-1 has the property that the change of a single bit of input produces a new hash value with no apparent connection to preceding hash value, this is based on pseudo random number generation.

→ 802.11i PRF takes four parameters as input and produces desired number of random bits. It is of form PRF (K, A, B, len), where

K = a secret key
A = a text string specific to application
B = some data specific to each case
Len = desired number of pseudorandom bits.

The group temporal key is generated by

GTK = PRF (GMK, "Group key expansion", MAC || GNonce, 256)



$R = HMAC\text{-}SHA\text{-}1\ (K \| A \| 0 \| B \| i)$

The parameter K serves as key input to HMAC. The message i/p consists of four items concatenated together: the parameter A, a byte with value 0, the parameter B and a counter i. The counter is initialised to 0.

→ The HMAC algorithm is run once, producing a 160 bit hash value.

→ If more bits are required, HMAC is run again with same inputs, except that i is incremented each time until the necessary number of bits is generated.