

UNIT - V

E-Mail, IP & Web - security

(1)

E-mail security : - (2)

Security services for E-mail attacks

- (2)

Possible through Email

- (3)

Establishing key & privacy - (4)

Authentication of the source - (5)

Message Integrity - (6)

Non-repudiation - (7)

Pretty Good Privacy - (8)

S/MIME - (9)

IP Security: overview of IPsec - (10)

IP and IPsec, Authentication Header - (11)

Encapsulation Security Payload ESP - (12)

Internet Key Exchange (Phases of IKE,

ISAKMP / IKE Encoding) - (13)

Web security: SSL/TLS Basic protocol - (14)

computing the keys - (15)

client authentication - (16)

PKI as deployed by SSL - (17)

Attack fixed in v3

11

(18)

~~Attacks~~ to avoid attacks
possible through email - (2)
- (3) -

~~Ensuring~~ key & privacy - (4)

~~Authentication~~ of the source - (5)

~~Message Integrity~~ - (8)

~~Non-repudiation~~ - (9)

~~Pretty Good Privacy~~ - (10)

S/MIME - (15)

IP Security: overview of IPsec - (17)

IP and IPX, Authentication Header - (21)

Encapsulation Security Payload ESP - (24)
- (26)

Internet Key Exchange (phases of IKE,

TLSAKMP / IKE Encoding - (27)
- (29)

Web security: SSL/TLS Basic Protocol - (31)

computing the keys - (35)

client authentication - (36)

PKI as deployed by SSL - (36)

Attacks fixed in v3
Exportability - (38)
SET - (40) Encoding - (39)

UNIT-V

(2)

E-mail, IP and web-security

mail security : Security services for e-mail:-

*. For Security, some features are provided for the electronic mail systems, are as follows;

- * Privacy
- * Authentication
- * Integrity
- * Non-repudiation
- * Proof of submission
- * Proof -of - delivery
- * Msg flow of confidentiality
- * Anonymity
- * Containment
- * Audit
- * Accounting
- * Self-destruct
- * Msg sequence integrity

Attacks possible through E-mail :- ③

- * E-mail hacking is the illegal access to (or) manipulation of an e-mail account.
- * SPAM
- * VIRUS
- * PHISHING

SPAM:-

- * Spam is created by attackers who send bulk e-mail.
- * Spammers attempt to find new ways around the increased legislation and policies governing unsolicited emails.

VIRUS:-

- * A virus incorporates email as a means of transportation. This type of virus is called a worm - the sobig virus is an example.
- * This virus creates a spamming framework by taking over unwilling participants.

PHISHING:-

(4)

* It is a type of attack that involves e-mails that appear to be from legitimate businesses that are user associated with.

* Phishing msgs look authentic, with all the corporate logos & formats as per to that of official emails.
>> An account no
> A pwd (or)
> A date - of - birth.

Establishing keys privately :-

Establishing keys :-

* Security services are provided by cryptography requires keys.

Establishing secret keys :-

* Two Parties establish a shared secret key for e-mail or some other means of private comm.

* Electronic mail is private but there are many ways to read that msg.

End-to-End Privacy:-

* Alice want to send a msg to Bob that only Bob can read it. She can't depend on the net keeping the msg secret, but she can ensure that nobody but Bob can read the msg by using cryptography to encrypt the message.

Privacy with Distribution List Explodees:-

* If Alice is sending a msg to a distribution list which will be remotely exploded, and Bob is only one of the recipients.

→ Local exploding services diff mechanisms.

→ Alice has to trust the maintainer of the mailing list, since a user can insert extra names into the distribution list.

Authentication of the source (6)

* In an unsecured mail system,
it is possible for Carol to send a
msg to Bob where the FROM field
says Alice.

* This cause great harm if
Bob takes the msg seriously.

Source Authentication based on public-
key technology:-

* Bob knows Alice's public-
key, Alice digitally sign the msg,
using her private key, which will assure
Bob that Alice wrote the msg.

* Alice compute a hash
of the msg & then to sign the
msg digest, since computing a
msg digest.

* Since, computing a msg
digest is faster than public key opns
and the msg digest is usually a
smaller quantity to sign than the msg.

Source Authentication based on secret key:-

(7)

MIC (or) MAC :- [Message Integrity Code]

- * The MAC is the CBC residue of the msg computed with the shared secret key.
- * The MAC is the msg digest of the shared secret appended to the msg.
- * The MAC is the encrypted msg digest, where the 128-bit msg digest is encrypted with the shared secret key, for instance in ECB (or) CBC mode.

Source Authentication with Distribution list

- > Source Authentication is easy with public keys and distribution lists.
- > Source Authentication is complicated with secret keys.
- > Using mail exploders & secret-key technology.

Message Integrity (8)

- * Bob receives a msg from Alice, how does Bob know that Carol did not intercept the msg & modify it?

Message Integrity without source Authentication

Authentication :-

- * To provide msg integrity protection without source authentication is possible?

- * The msg anonymous so there is no source verification.

Integrity protection :-

- * To prevent someone from reversing that the first part of the msg was the proof that they were the kidnappers & substitute a diff second part of the msg, with diff direction for dropping off price money.

authentication by humans No experts
this odd combination & a kidnapper rep
such a small market that none is to be
developed.

Repudiation: Non-Repudiation
* It is the act of denying that
the particular person sent a msg.

- Non-Repudiation:
- * If Alice sends a msg to Bob, Alice
can't deny that she sent msg, Alice Bob
prove to a third party that Alice
sent the msg.
 - * Non-repudiation based on Public-
key technology.
 - * plausible Deniability Based on
public-key Technology.
 - * Non-Repudiation with secret keys.

PRETTY GOOD PRIVACY [PGP]:-

- * PGP is a secure mail protocol (10)
- * PGP performs encryption & integrity protection on files.

Process:- * one send a secure mail msg could first transform the file to be mailed using PGP & then mail the transformed file using traditional mailer.

by distribution:-

- * PGP uses public-key computation for personal keys.
- * public keys certifications and certificate chain verifications are done b/w PGP, PEM, S/MIME.
- * PGP Assumes Apachy.
- * With PGP, each user decides which keys to trust.

Efficient Encoding:-

* PEM

* PGP

(11)

PEN designed the work in the environment where the msg contents are restricted by two Pblms.

First Pblm:

- * The intermediate mail relays insist that the msg contain only Ascii characters & limited - length lines.

2nd Pblm:

- * The representation of text at the destination machine is different from the representation at the source machine.

PGP:

- * PGP process a file whether the file is text (or) binary.
- * If PGP knows that the file is binary, then PGP will not cannibalise it & PGP will mark the PGP processed msg as binary so that the PGP at the receiver

Certificate And Key Revocation :-

(12)

* A certificate is revoked by whoever signed the certificate, it does not mean that the key in the certificate is bad, it means that whoever signed the certificate no longer wants to vouch for its authenticity.

Key rings:-

* Key ring is a data structure that contains some public keys, some info abt people & some certificates.

3 levels:-

- > None
- > Partial (or)
- > Complete

Anomalies:-

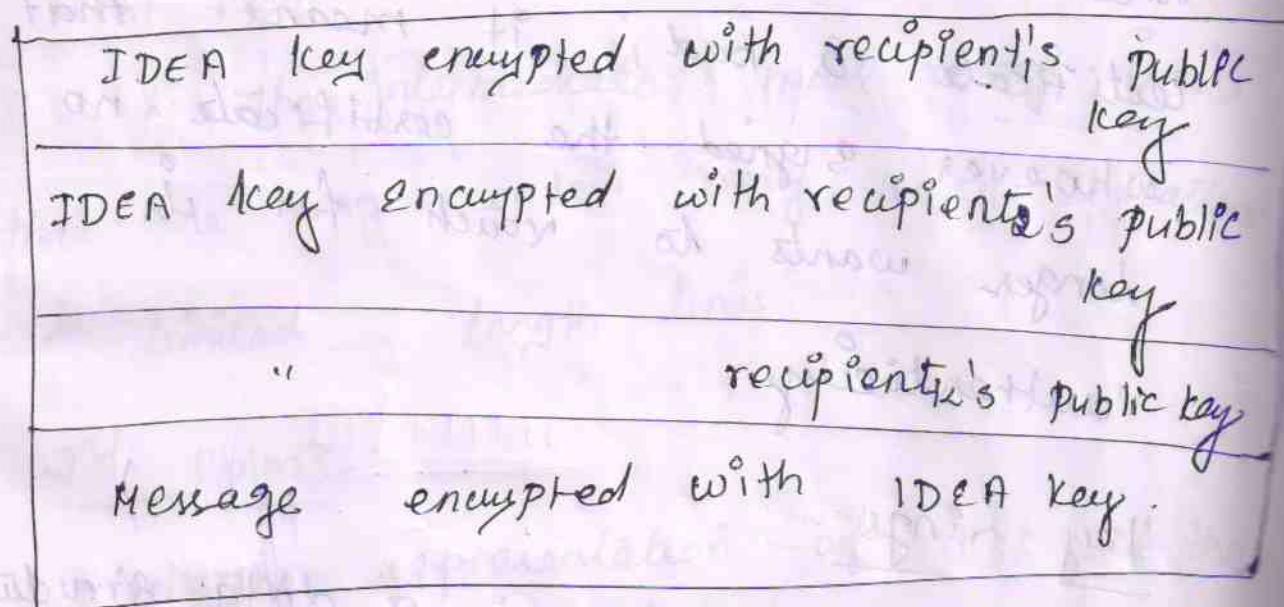
- > File Name
- > People
- > Obj Formats

Msg formats:-

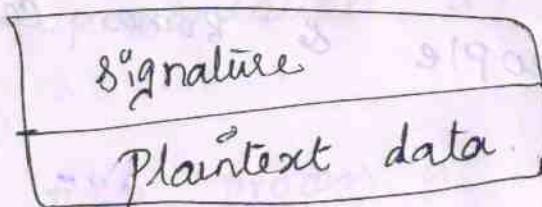
(13)

- * PGP msgs are composed of a sequence of primitive objs.

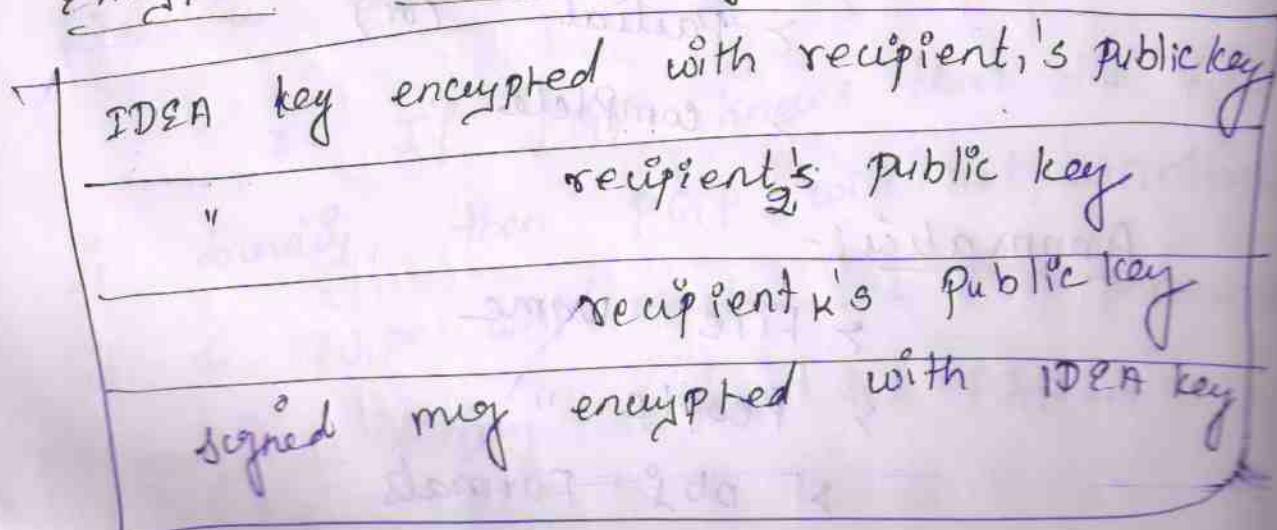
1) Encrypted msg:-



Signed msg:-



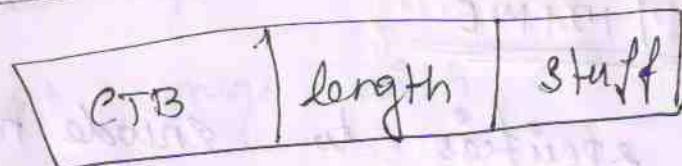
Encrypted signed msg:-



Signed Human - Readable message: (4)

----- Begin PGP signed message -----
clear text message
----- Begin PGP signature -----
version: 2.2
signature
----- End PGP signature -----

primitive object formats:-



key encrypted under a public key

0001 in CTB:-

Field description

| | |
|--------|--|
| octet | Version number (2) |
| " | key Id (low order 64-bits of rsa modules) |
| " | algorithm id (1 for rsa, no other defined) |
| higher | Idea key encrypted with rsa |

IDEA key before RSA- Encryption

(15)

| | | | | | | |
|---|---|-----------------------------|---|---|-------------|-----|
| 0 | 2 | N random non-zero octets | 0 | 1 | IDEA key | CRC |
|---|---|-----------------------------|---|---|-------------|-----|

Key Rings:-

| | | | | | |
|-----------|-------|--------|-------|-----------|-------|
| publickey | trust | userID | trust | signature | trust |
| | | | | signature | trust |

S/MIME

- * MIME specifies to encode non-text data and type labels into what will look like a text msg to intermediate mailers.

- * S/MIME specifies it's encrypted blobs are binary data & MIME takes care of encoding them.

- * S/MIME has some encoding issues.

- * PEM, signed msg are,
 - > clear - signed
 - > encoded

clear - signed :-

(16)

- * clear signed can be read by mail readers that don't understand S/MIME.

Encoded format :-

If the recipient has a mailer that doesn't understand S/MIME, the encoded format appear to it as an encrypt msg with a file attachment with the name S/mime, PTM.

S/MIME Specification :-

* explicitly allows a sender to encrypt a msg & then sign the result, (or) even sign the msg, encrypt the result & then sign the encrypted version again.

* S/MIME defined a large set of header files for both encrypted & signed msgs.

* S/MIME certificate Hierarchy. (17)

* S/MIME with a public certificate

* S/MIME with an organizational certificate

* S/MIME with certificates from any old CA.

IP Security

Overview of IPsec:

* IPSEC assumes that two nodes have a shared session key which is configured manually.

Security Associations:-

* An IPsec security Association SA is a protected connection.

* SA is uni-directional, so a conversation between Alice & Bob consists of two SA's one in each direction.

(18)

* IPSEC header

* Security Association Database

* Security Policy Database

* AH and ESP :-

* AH - ~~feature~~ Authentication Header.

* ESP - Encapsulating Security Payload.

* 2 types :-

> AH provides integrity protection only.

> ESP provides encryption (or)

Integrity protection

> ESP optionally provides integrity protection.

* AH also provides integrity protection for some of the fields inside the IP header.

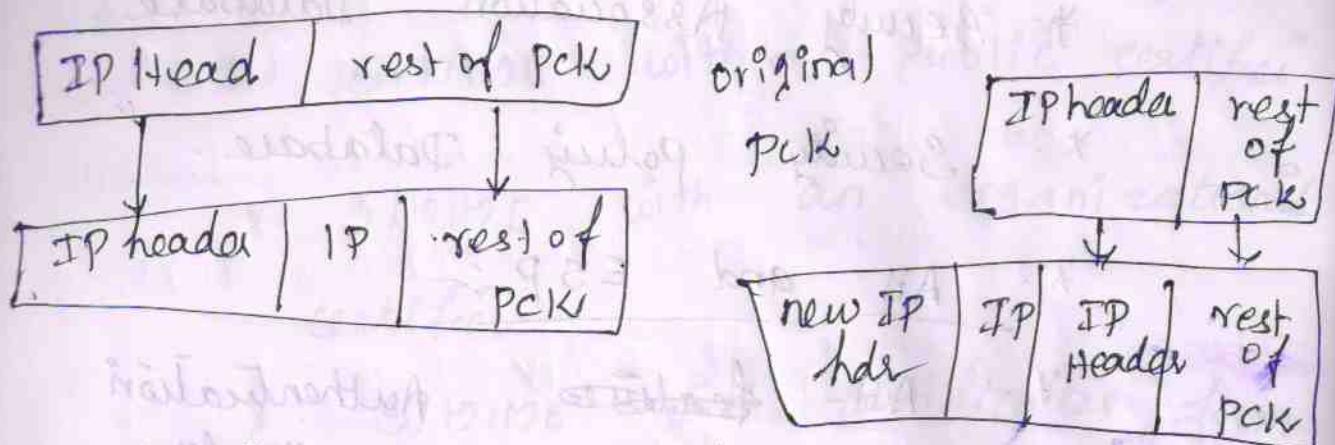
> AH Feature

> Firewall & Router - ESP

> Router & firewall - TCP ports.

Tunnel, Transport Mode:-

(19)

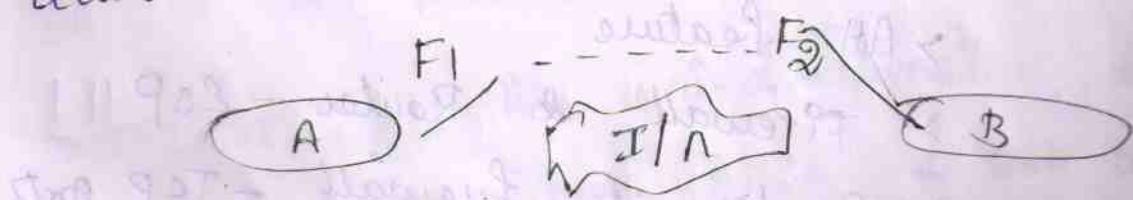


* "Transport mode" is most logical when IPsec is being applied end-to-end.

* "Tunnel mode" - The common use of tunnel mode is firewall to firewall

(or) end-node to firewall, where the data is only protected along part of the path b/w the end-points.

* If two firewalls establish an encrypted tunnel to each other across the internet, which is



IPsec, Tunnel mode b/w Firewalls.

13. IP : Src = F1... dst = F2 | ESP | IP : Src =

A, dst = B

(20)

Multiply encrypted IP Packet :-

original pck as
launched by A,

IP : Src = F1; dst = F2 | ESP | IP : Src = A,
dst = B | ESP |

Encrypted with
the F1-F2 key.

IP header Protection :-

- * AH is necessary bcoz it protects the IP header.
- * If AH is necessary, this is provided by ESP in tunnel mode.
- * Intermediate routers can't enforce AH's integrity protection, bcoz they don't know the session key for the Alice - Bob security association.
- * AH is used by Bob to check that the IP header was received as launched by Alice.

IP and IPv6

(21)

- * IP was designed with 32-bit addresses.
- Internet started to use larger addresses so IETF invent their own header format.
- * The "V6" in the name "IPv6" comes from the first four bits of an IP header.
- * IPv6 is not easily accepted by people.

NAT (Network Address translation):-

- * NAT translates an internal node's IP address into a globally unique address when that node is interacting with something on the internet.

IPsec with NAT:-

- * An IPsec tunnel can't go through a NAT box. Be'z, the NAT box wants to update the IP addresses inside the encrypted data and it doesn't have the key.

FTP:-

- * FTP is the protocol. It is encapsulated within both 3 & 4th layer.

Ex: 178.201.19.175

Firewalls:-

(22)

- * Network administrators like to have firewalls observe pass & discard (filter) packets based on characteristics which protocol is being used.
- * IPsec encrypts info on which firewalls like to base decisions, such as the port fields in the TCP header that to know whether the data is E-mail (or) telnet.

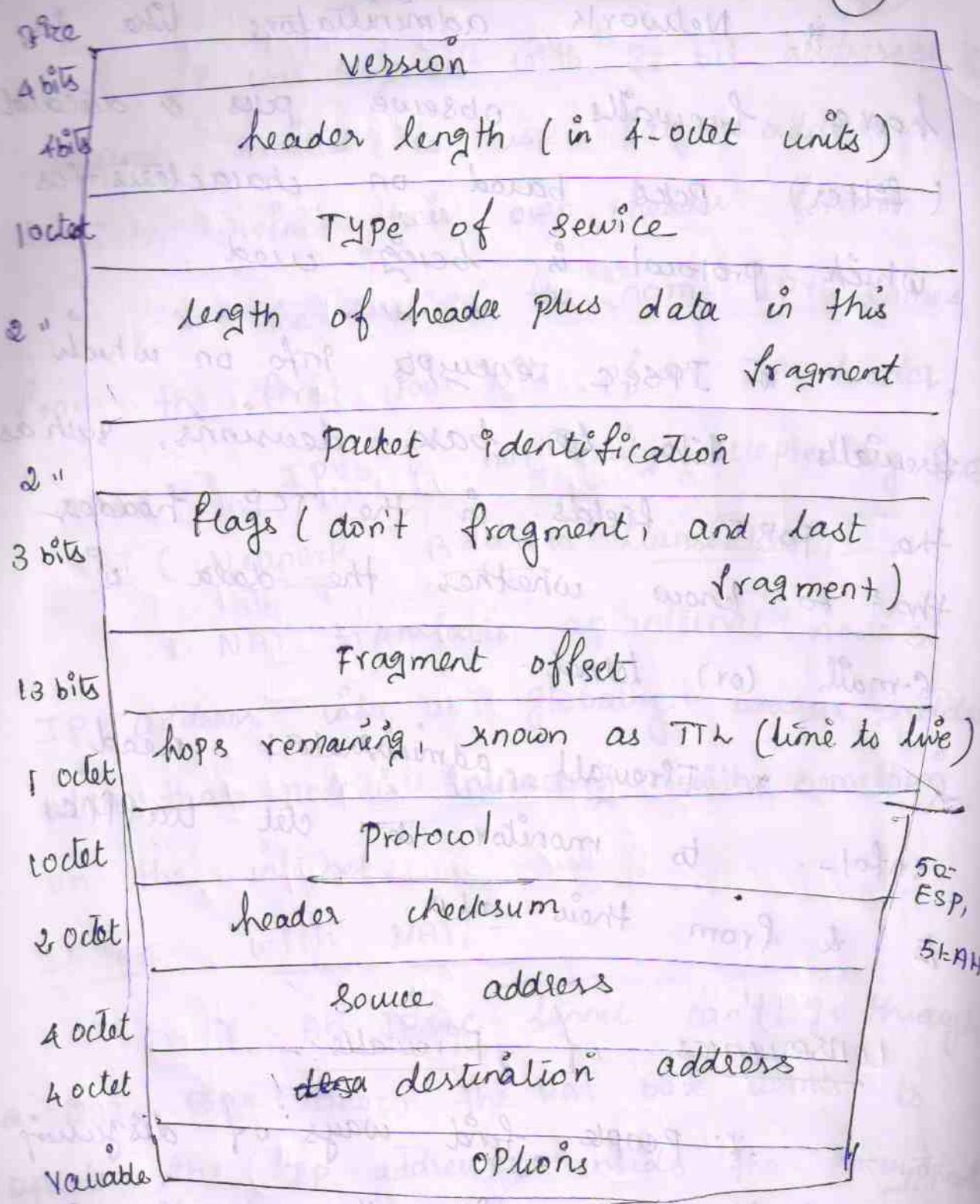
- * Firewall administrators need info. to monitor all traffic & from their n/w.

Consequences of Firewalls:-

- * People find ways of disguising the traffic that firewall administrators like to block so that it looks to the firewall like the kind of traffic the firewall is configured to allow.

IPV4 header:

(23)



- * protocol indicates IP, if it's a tunneled pck, (ie) IP header is another IP header.

IPv6 Header :-

octets

4

Version (4 bits) | types of service
flowlabel

2

Payload length

1

next header

1

hop remaining

1b

source address

1b

destination address

+ octets

Authentication Header (AH) :-

1

next header

1

Payload length

2

unused

4

SPI (Security Parameter Index)

4

sequence number

variable

authentication data

Mutable, Immutable :-

* Some fields in the IP header

modified by routers, so they can't be included in AH's end-to-end integrity check.

The IPX4 AH defines the mutable fields:-

* Type of Service

* Flags

* Fragment offset

* Time to Live

* Header checksum

IPX6, the mutable fields are:-

* Type of Service

↳ flow-label

↳ hop-limit

Mutable but predictable:-

* IP source routing

(S)

(D₁)

(D₂)

(D₁)

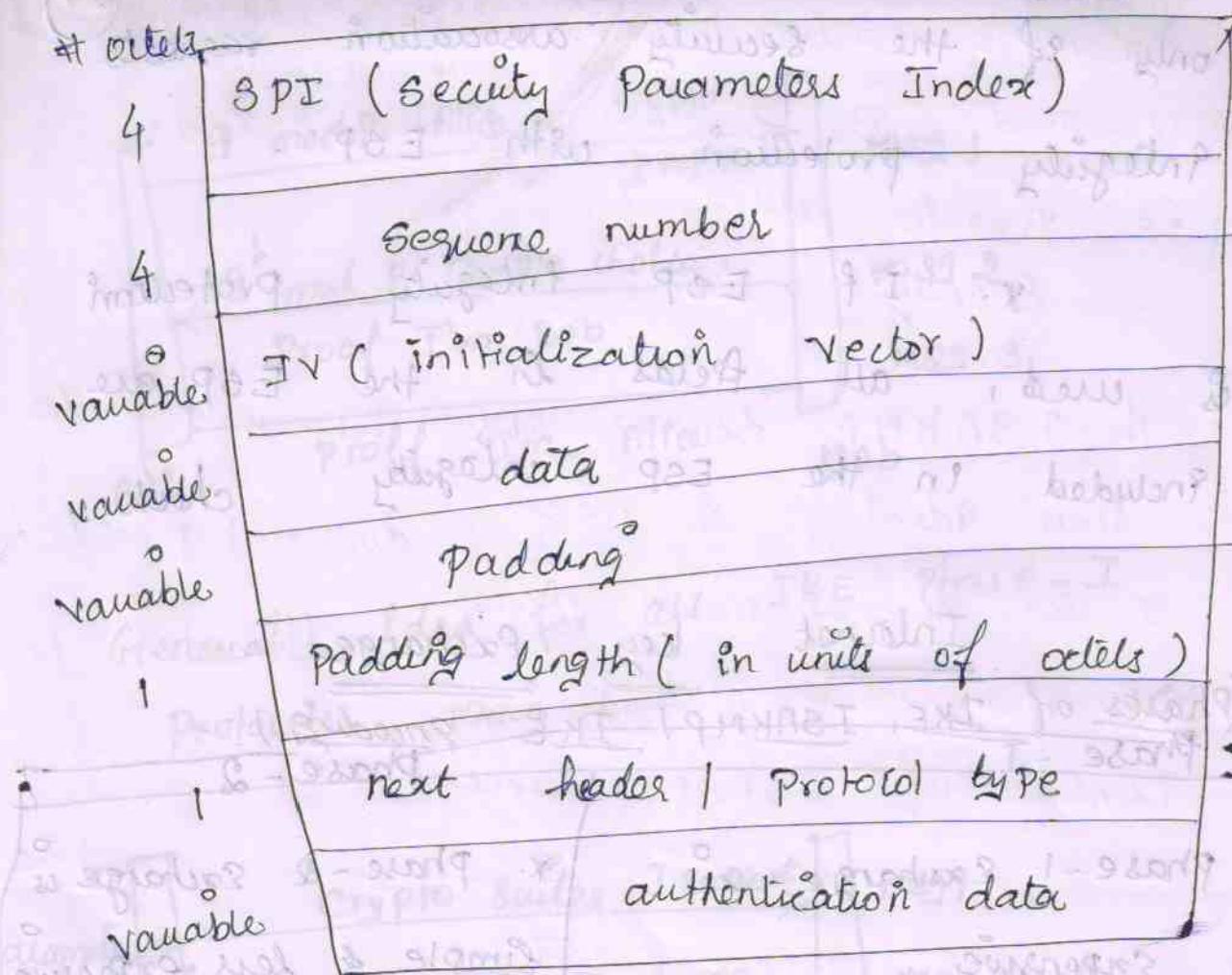
(D₃)

* Fields are mutable but predictable

included in the AH integrity check, but with the values they will have when received at the other end.

continuation in next note

Encapsulating Security Payload (ESP) (26)



* ESP allows for encryption (or)

integrity protection

* The security association database

tells what to use when transmitting to a particular IP address.

* Encryption is used in the fields;

> Data

> Padding

> Padding length

> Next header are encrypted.

- * The Authentication Data appears only if the security association requests integrity protection with ESP.

- * If ESP integrity protection is used, all fields in the ESP are included in the ESP integrity check.

Internet key Exchange :-

(Phases of IKE, ISAKMP / IKE Encoding) :-

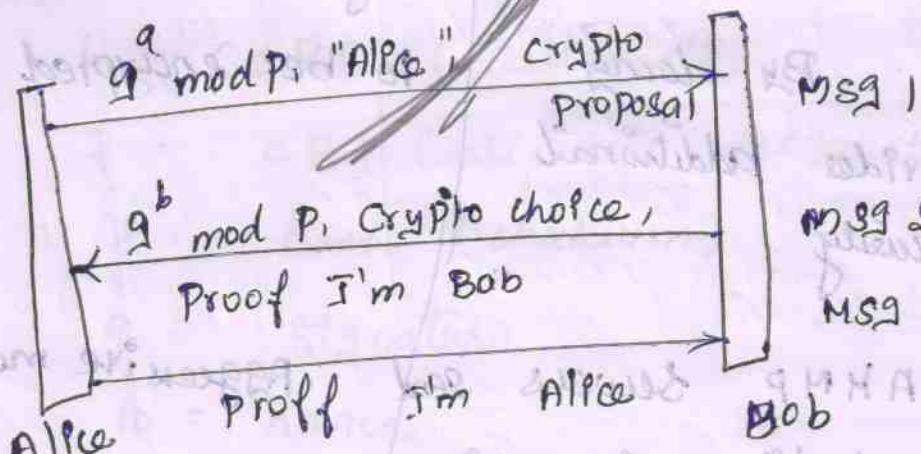
- | Phase - I | Phase - 2 |
|---|---|
| * Phase - 1 Exchange is expensive. | * Phase - 2 Exchange is simple & less expensive bcoz, they can use session key created out of the phase - I Exchange. |
| * There will be multiple Phase - 2 setups inside the same phase - I exchange. | |

Security Weakness :-

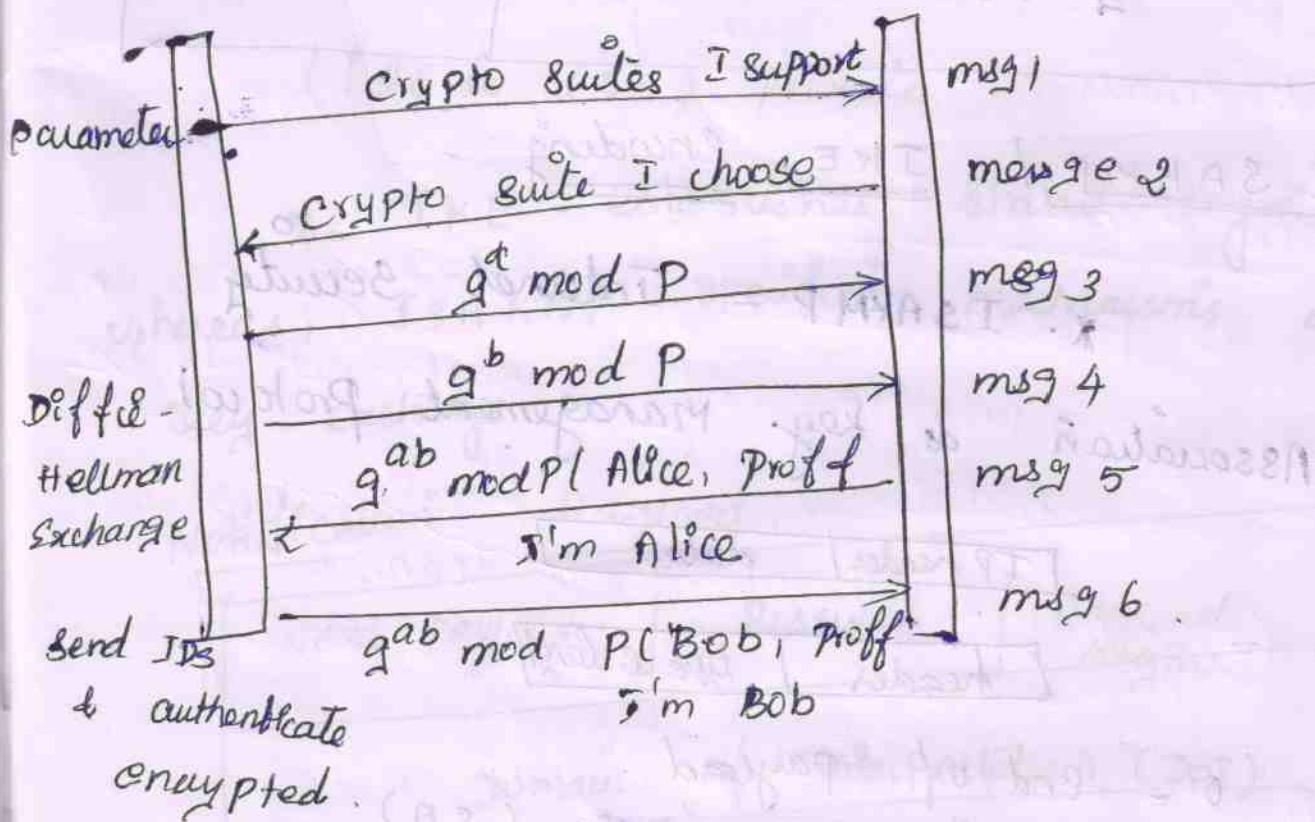
- > there is weakness if different flows used the same key.
- > If the SA used encryption only, then there exists weakness.

General Idea for all IKE Phase-I
Protocols, Aggressive mode.

(28)



General Idea for all IKE Phase-I
Protocols, Main Mode



key types

- Pre-shared secret key
- Public Encryption key
- Public Signature key

| Aggressive Mode | Main Mode (2) |
|---|---|
| * No msgs needed to be encrypted. By doing so provides additional security. | msg 5 & 6 are needed to be encrypted. |
| * The ISAKMP servers send their identity in msg 5 (or) 6 of main mode (Q). main mode protects identity of ISAKMP servers. | Aggressive mode does not provide certificate. |

ISAKMP / IKE Encoding

* ISAKMP - Internet Security Association & key management protocol.



0 = end of payload

1 = Security Association (SA)

2 = P (proposal) : proposed SPI

3 = T (transform) : cryptographic cha-

4 = KE (key exchange)

(30)

5 = ID (Identifier phase 1, Phase 2)

6 = CERT (certificate)

7 = CR (Certificate request)

8 = hash (checksum)

9 = signature

10 = nonce

11 = notification

12 = delete

13 = Vendor ID

14 to 127 (Reserved)

(128 to 255) private

* IKE establishes shared key whereas, ISAKMP specifies mechanisms of key-exchange.

Notification Payload :-

| Next payload | Reserved | Payload length |
|--------------------------------|----------|-----------------|
| Domain of Interpretation (DOI) | | |
| Protocol - ID | SPI size | Notify msg-type |
| Security Parameter Index (SPI) | | |
| Notification Data | | |

Notify msg type & values:-

(31)

- | | |
|----------------------------|---------------------------|
| 1 - Invalid - Payload type | 6 - Invalid minor version |
| 2 - DOI not supported | 7 - " exchange type |
| 3 - Situation not " | 8 - " flag |
| 4 - Invalid cookie | 9 - " msg ID |
| 5 - .. major version | 10 - " protocol ID |

Web Security

* The In. web is vulnerable to attacks
 So, it needs security to overcome attacks on
 web-servers over Internet.

Web security threats:

- ① Passive Threat → Accessing info on a website that is supposed to be restricted
- ② Active threat {
 → Attacking msg in transit
 → Impersonating another user.

Threats & classified as,

- > Web server
- > Web browser
- > N/w traffic

Comparison of threats on Web

(B2)

Threats consequences

- | | | |
|--------------------|--|--|
| 1. Integrity | 1. Modification of data 2. " of money 3. " of msg 4. Trojan horse browser. | 1. Loss of info 2. compromise of machine 3. Vulnerability to all other threats |
| 2. Confidentiality | 1. Eaves dropping on net 2. Theft of info from server 3. Theft of data from client | 1. Loss of info. 2. Loss of privacy. |
| 3. Denial Service | 1. killing of user threads 2. Flooding machines with bogus request 3. filling up disk or memory. | 1. Disruptive 2. Annoying 3. Prevent user from getting work done. |
| 4. Authentication | 1. Impersonation of legitimate users. 2. Data forgery | 1. Misrepresentation of user 2. Belief of false info is valid. |

Secure Socket Layer and Transport Layer Security

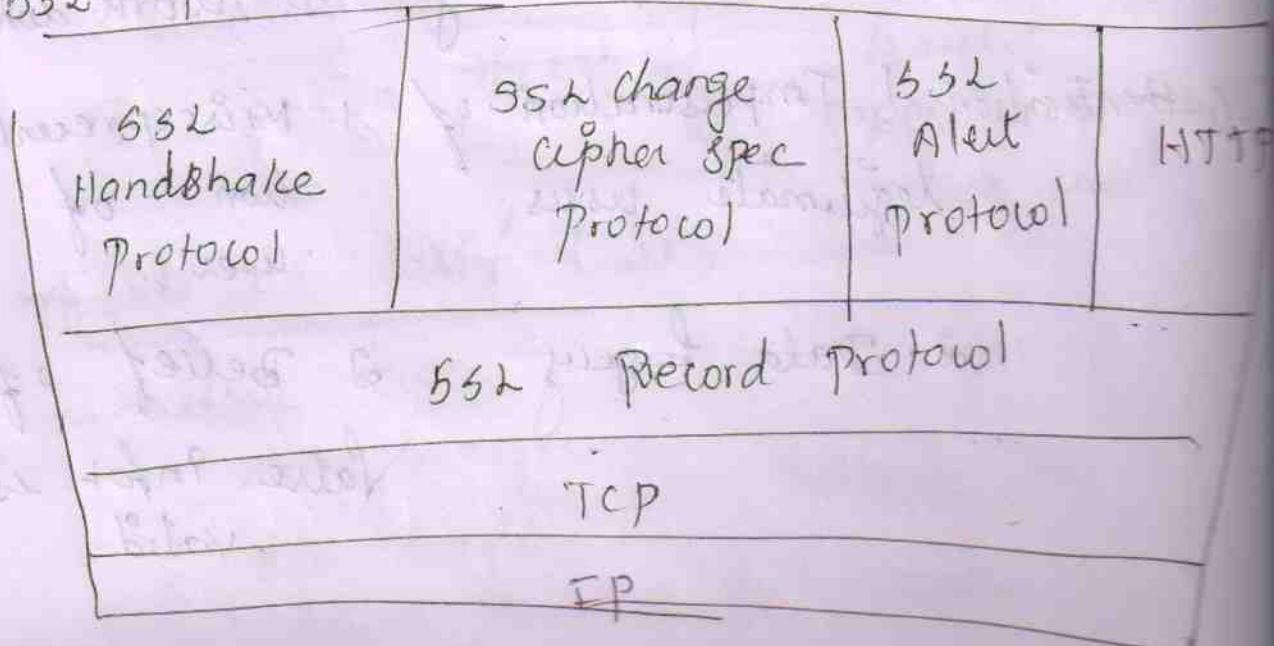
(33)

- * SSL provides security services b/w TCP & appn that use TCP
- * the 1st std version is called TLS
- * SET is open encryption & security specification designed to protect credit card transaction on Internet.

SSL Architecture:-

- * TCP, reliable, end-to-end security service. The version 3 of SSL is used in draft document by the public & industry.

SSL Protocol Stack



(1) Record Protocol takes an appln msg to be transmitted as

(2) Fragmentation to 2^{14} bytes

(3) Compress it

(4) Adds MAC

(5) Encrypt

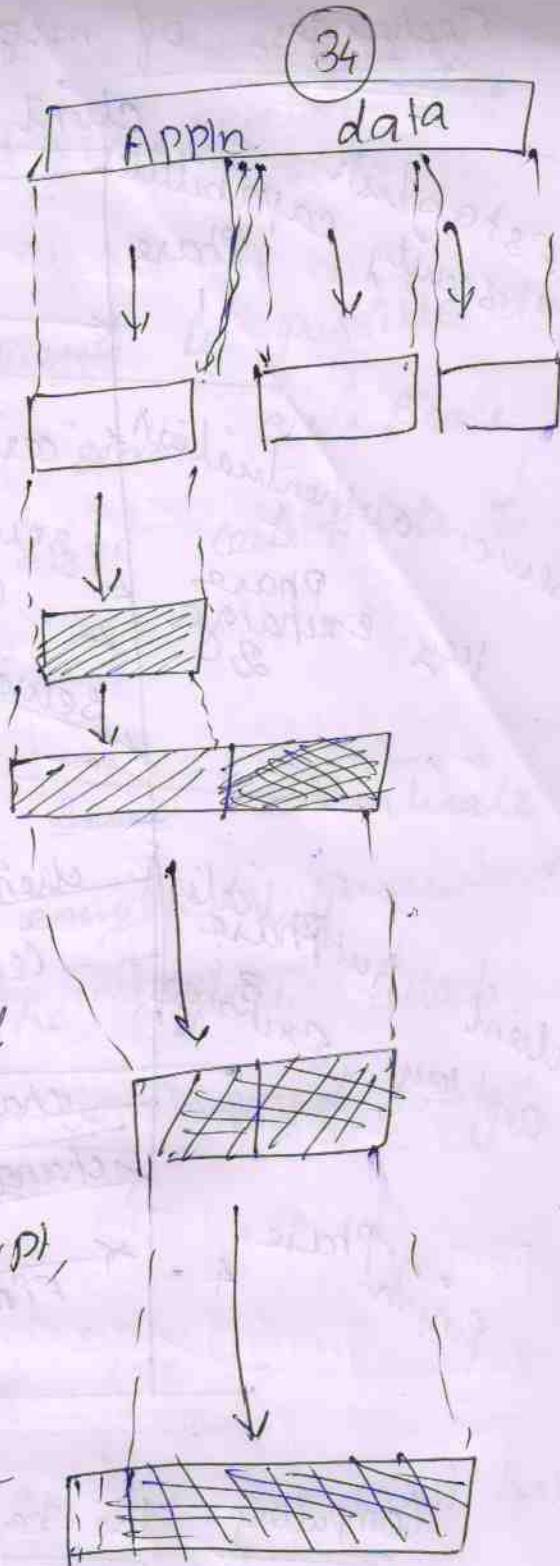
(6) APPend & Reuse header

(7) on receiving, Decrypt,

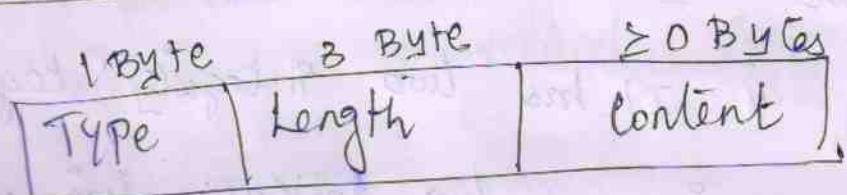
Verify, Decompress,

Reassemble and then

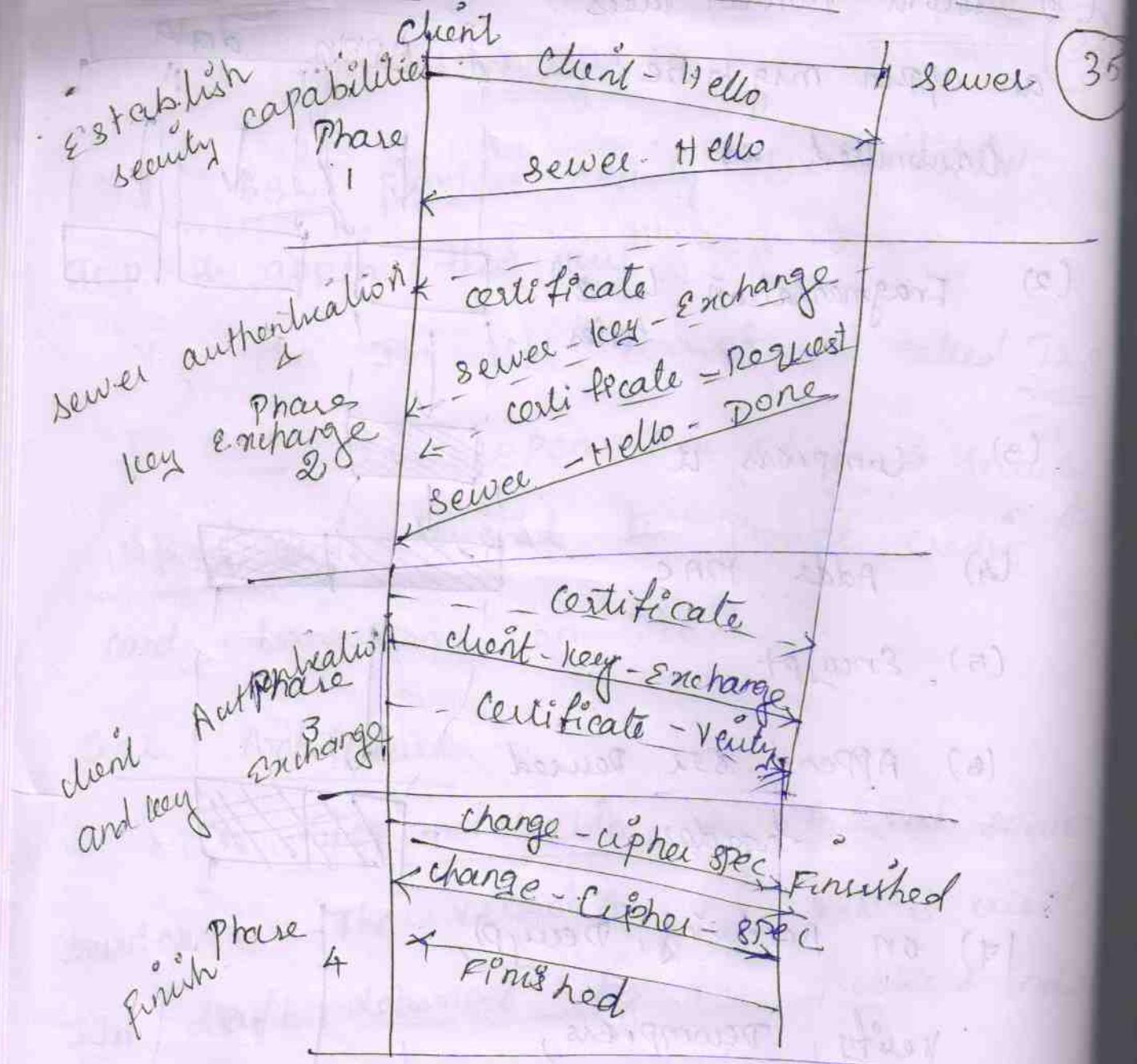
Delivers to higher
level users



Handshake Protocol :-



Exchanges of msgs b/w client & sever



Computing the keys

• Six keys $k = \text{hash}(S, R_A, R_B)$. Here two encryption keys one for send and other for receive.

• It has two integrity keys for send & receive and two initialization vector (IV) for send & receive.

client Authentication :-

(36)

- * Sender authenticates receiver but receiver) sender do not
- * But mutual exclusion is possible b/w sender & receiver. Receiver sends certificate to request, so client must have certificate to perform mutual exclusion.

* In-order to server authenticate client, server requires encrypted password to prevent man in the middle attack, receiver certificate must be signed by certificate authority.

PKI as Deployed by SSL :-

* When deploying SSL, we must have CA. CA provides certificate discovery, management, responses CRL and responses.

> private key, certificate

> Server protection, performance

> protecting appln

SSL 3.0 :-

- * In SSL 3.0; it separates transportation of data b/w transport layer & others.
- > Provides client ability to send certificate.
- > Implements key exchange protocol.
- > Allow compression.

Attacks in SSL v3 :-

- > Down-Grade attack
- > Truncation "
- * Down-Grade attack is a consistent problem in SSLv3. Avoiding backward compatibility can prevent downgrade attacks.
- * Truncating Attacks aims the receiver not to receive's complete msg from sender.
- * This is prevented in SSL by closing handshake msg. Until receiver receives closing handshake msg, receiver acts active to receive msgs.

Exportability :-

* SSL v2 supports 128-bit encryption.

After that exportable keys were limited to 40 bits.

* In Exportable Suite, now there were 40 secret bits and 88 non-secret bits. This 128 bits acts as SSL v2 client master key.

* In SSL v3, integrity keys were computed as in SSL v2. It uses 40 bits for secret encryption keys, and 14 non-secret.

* If a domestic sever (ex: 1024-bit RSA key) comm with other sever or client creates 512 bits of ephemeral key and signs it with its 1024-bit key.

Encoding :-

SSL/TLS runs on top of TCP, so it can send chunk of data. TCP handles chunk of data by breaking and re-assembling.

(i) Record

(ii) Messages

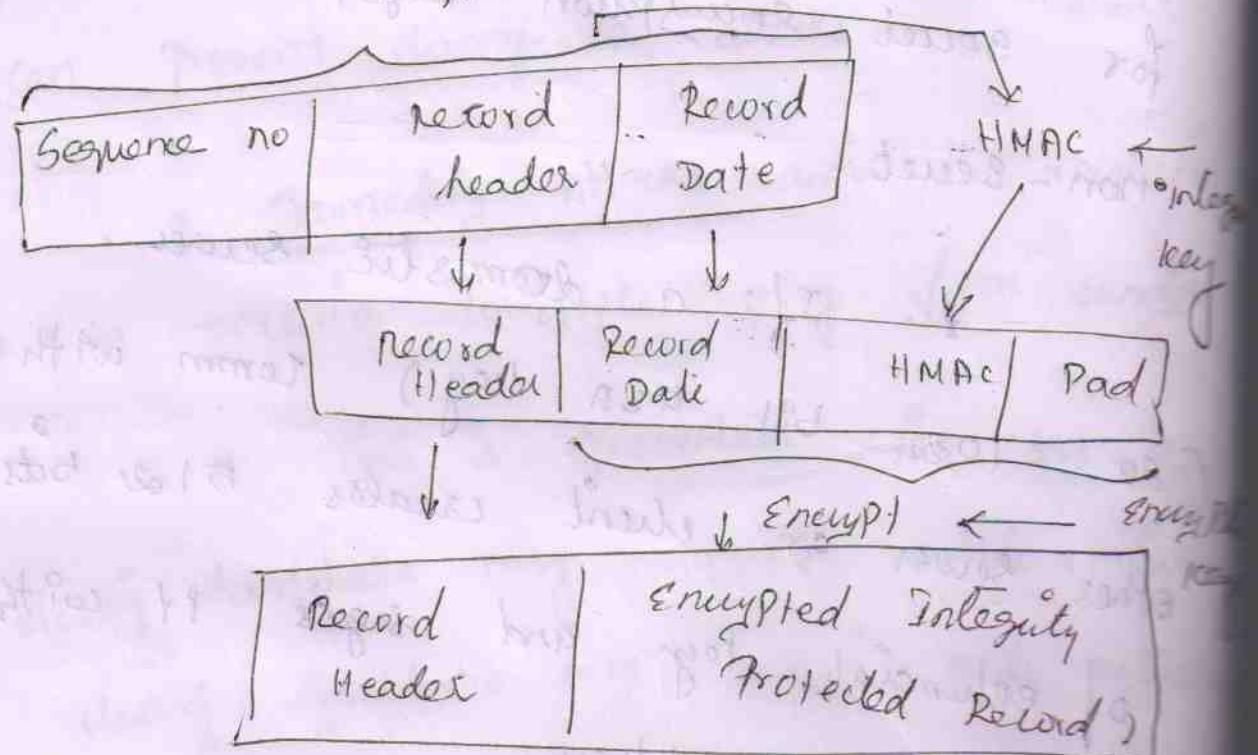
Encrypted Records :-

> 20 (change cipher spec)

> 21 (alert)

> 22 (handshake)

> 23 (appn data)



Secure Electronic Transaction (SET)

secure electronic transaction

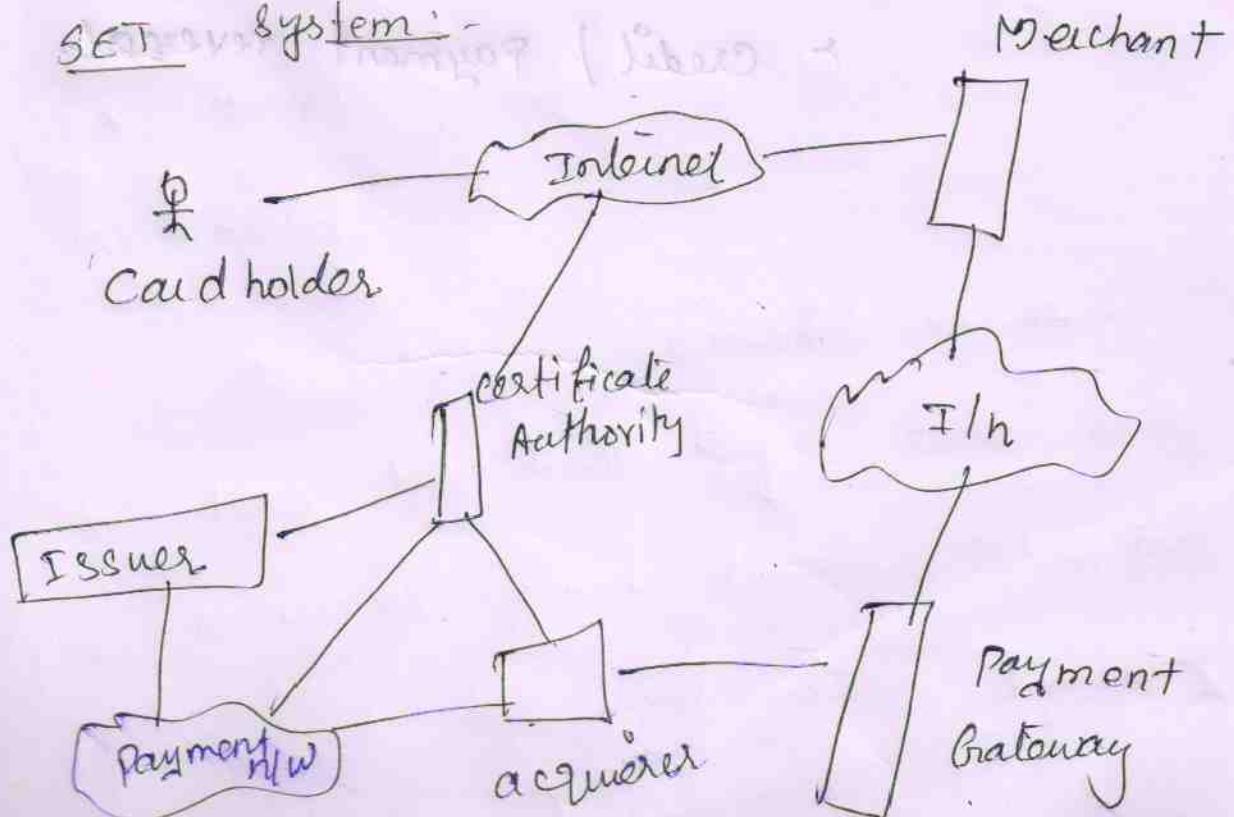
(49)

- * The ~~SET~~ Protocol known as SET which for providing secure credit card transactions on the I/n.

SET Qualities :-

- ii) Confidentiality
- iii) Integrity
- iv) Authenticity
- v) Non-Repudiability

SET System :-



SET supported technologies :-

(41)

- > card holder registration
- > Merchant "
- > Purchase request
- > payment authorization
- > " capture
- > certificate query
- > Purchase inquiry
- > " notification
- > Sale transaction
- > Authorization reversal
- > Capture reversal
- > Credit / Payment reversal