Kerberos V4 Dialogues ← Timestamp ↑

1) $C \rightarrow AS : ID_c \| ID_{tgs} \| TS1$

AS: authenticates the user
TGS: Grants tickets
Server: The actual content user is trying to access.

TGS determines the amount of time a service can be used (after which ticket needs to be renewed) and also which services can be accessed.

2) $AS \rightarrow C : E_{K_{pw}}[K_{c,tgs} \| TS1 \| LT1 \| Ticket_{tgs}]$

$K_{c,tgs}$ ⇑ Secret key LT1 ↑ Lifetime

$Ticket_{tgs} = E_{K_{tgs}}[K_{c,tgs} \| TS1 \| LT1]$

3) $C \rightarrow TGS : ID_c \| Ticket_{tgs} \| Authenticator_c$ (recv.)

$Authenticator_c = E_{K_{c,tgs}}[ID_c \| TS4 \| AD_c]$

ID_c ↑ (sent by client) AD_c (Address of C)

4) $TGS \rightarrow C : E_{K_{c,tgs}}[K_{c,v} \| TS1 \| TS2 \| LT2 \| Ticket_v]$

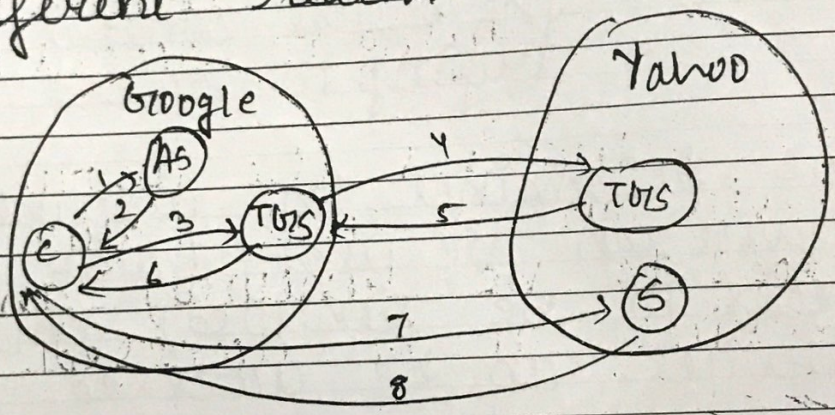
$Ticket_v = E_{K_v}[K_{c,v} \| TS3 \| LT3]$

5) $C \rightarrow V$: $ID_c \| Ticket_v \| TS3 \| Authenticator_c$

$Authenticator_c = E_{K_{c,v}}[ID_c \| TS3 \| AD_c]$ (Address of c)

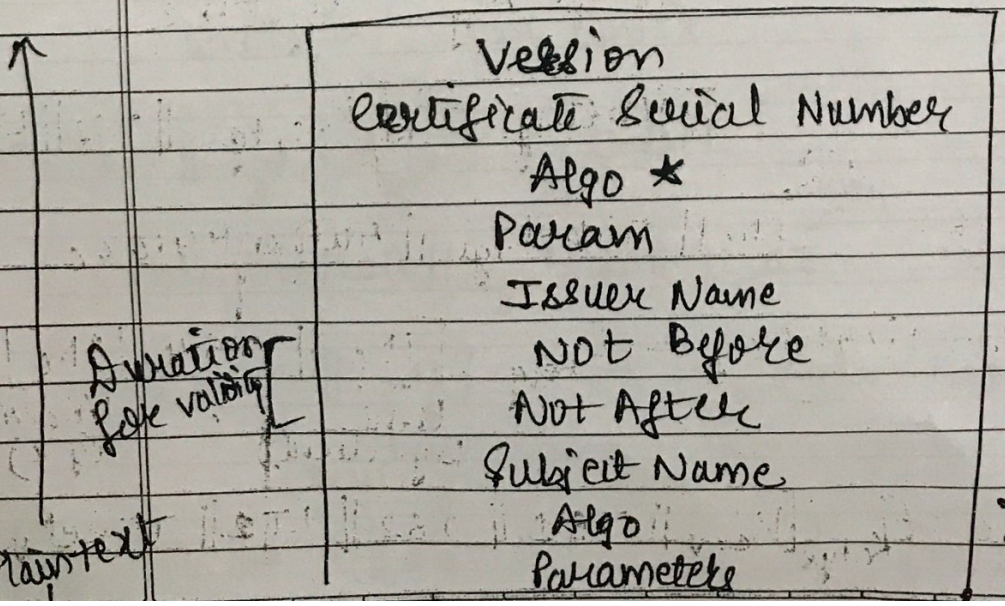
6) $C \rightarrow V$: $E_{K_{c,v}}[Data]$

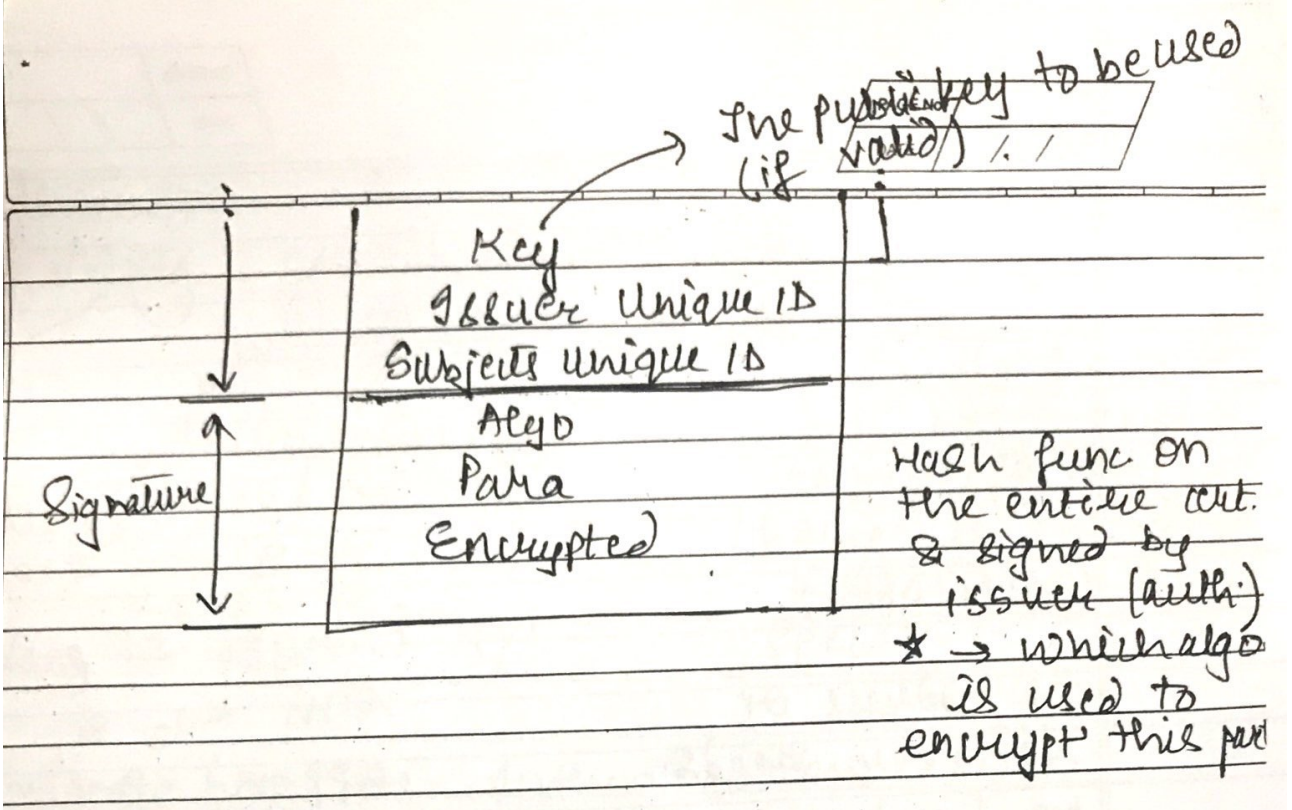
When client requests for a service on some server which exists in a different realm :



18/9/19

X.509 Authentication Service





IP Security

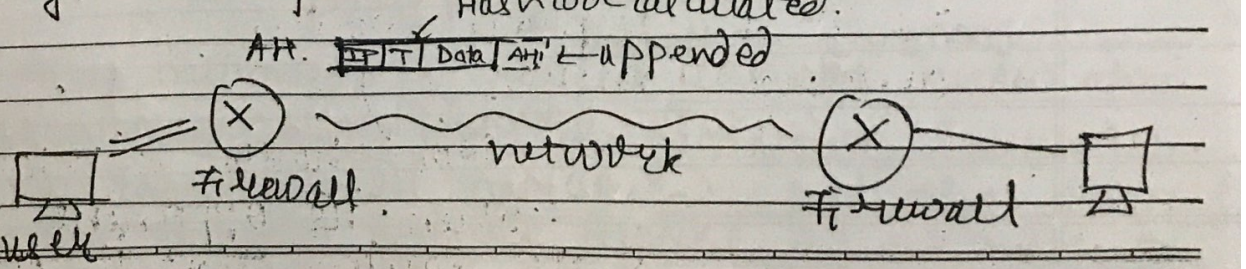
IPSec (prevents IP spoofing)

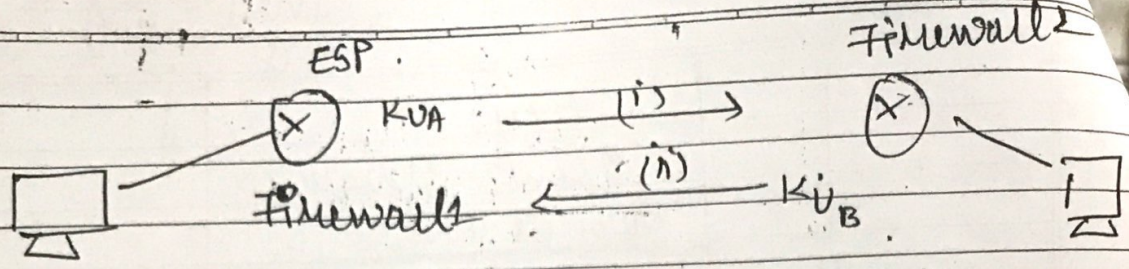
ESP protocol
- uses encryption + authentication

AH protocol
- uses only authentication

Using both these protocols, we can design DDI (for key management).

	ESP	AH	ESP + AH
Access control	✓	✓	✓
Connectionless Integrity	✓		✓
Data Origin authentication	✓		✓
Rejection of replay attack	✓	✓	✓
Confidentiality			✓





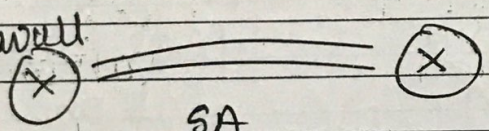
User.
Sender)

(VPN/DNS)

IP address of receiver needs to be known by the sender or someone who wants to spoof the IP.

iii) Change sender IP to Firewall's IP & Recv. IP to F2 IP.

iv) Encrypt IP packet with KU_B & append the new (Firewall's) IP with the packet.



SA
(Security association)

SA uses SPI (Security ~~Protocol Identifier~~ Parameter Index) which consists of the Destination IP address and security protocol ID (AH / ESP).

Other parameters:
i) Seq. no. counter
ii) Anti-replay policy
among others.

↑ How should a m/c react when its opponent sends a mess. saying that it did not [illegible]

→ Mode : i) Transport mode
ii) Tunnel mode

Intruder can neither see the contents & nor change them.

can see the packet but, not change it encrypted content (packet is encrypted upto trans. layer spec.)

Scope of AH Authentication

IPV4 | Original IP | TCP | Data |

IPV6. | Original IP | Ext. para | TCP | Data |

(Ext - Extensio

Before AH.

1) Transport mode

calculated Hash code on the entire packet.

| Orig. IP | AH | TCP | Data |

| Orig. IP | Ext. para. | AH | TCP | Data |

2) Tunnel Mode

Payload

| New IP header | AH | Orig. IP | TCP | Data |

| New IP header | Ext. para. | AH | Orig. IP | Ext. para. | TCP | Data |

Any intermediate router cannot know about the original IP (originating source). It can only be read by the two end points of a tunnel, i.e. Sender & Receiver.

Scope of ESP mode of IPSec

IPv4 | Orig. IP | TCP | Data |

IPv6 | Orig. IP | Ext. para. | TCP | Data |

Transport mode:

(Encrypted)

| Orig. IP | ESP | TCP | Data |

| Orig. IP | Ext. para. | ESP | TCP | Data |

Tunnel mode

v6 | New IP header | Ext. para | ESP | OG IP header | Ext para | TCP | Data |

v4 | New IP header | ESP | Orig. IP | TCP | Data |

⇒ VPN ≃ IPSec with ESP

9/19 Security Electronic Transaction (SET)

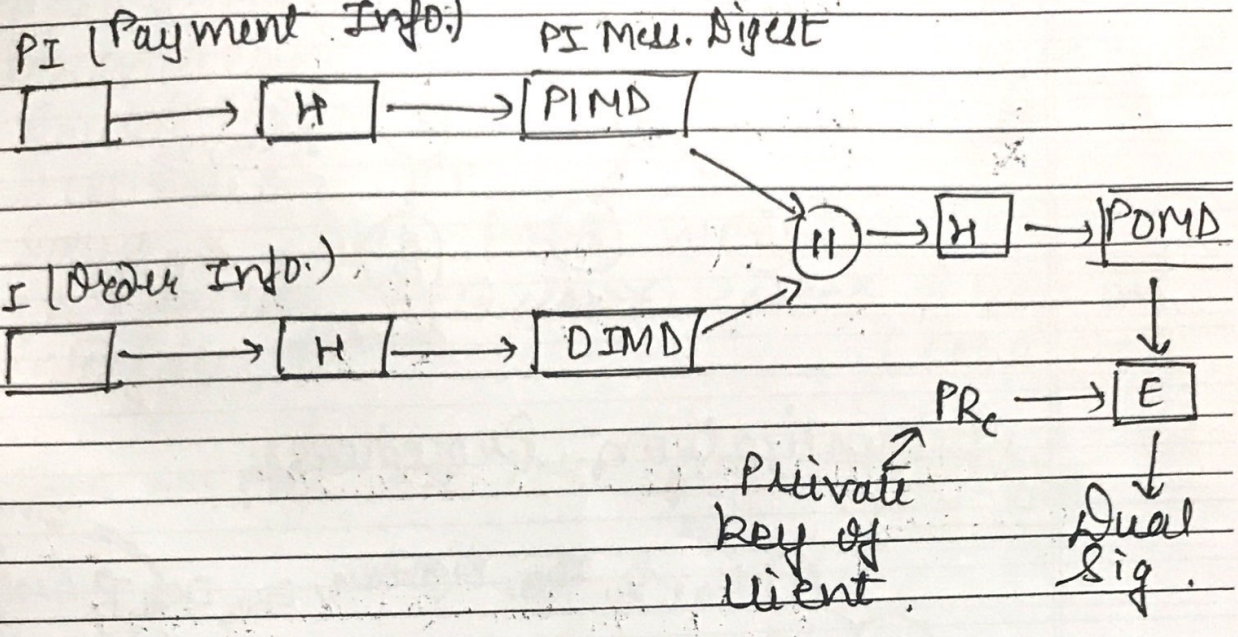
Participants:

i) Card Holder
ii) Merchant
iii) Issuer (VISA)
iv) Acquirer (Bank)
v) Certification Authority

Steps:

1) Cust. opens account & gets credit card
2) Cust. receives a certificate

Dual Signature

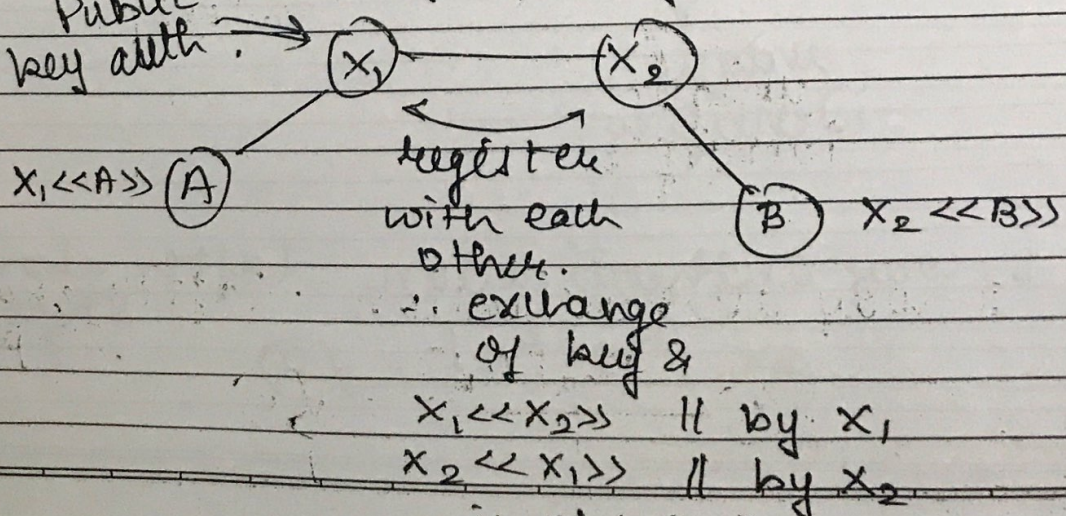


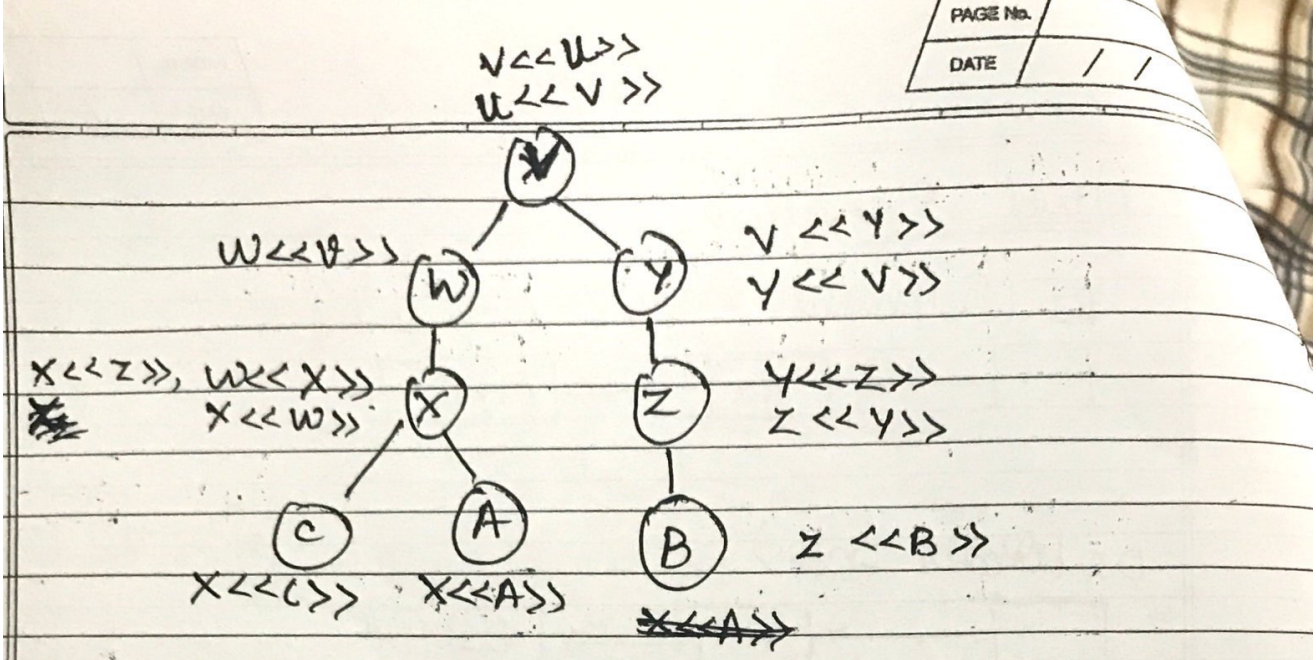
Secure Socket Layer (SSL)

→ It is designed to make use of TCP to provide a reliable end-to-end secure service.

→ It is a two layer protocol → lies between network and application layer.

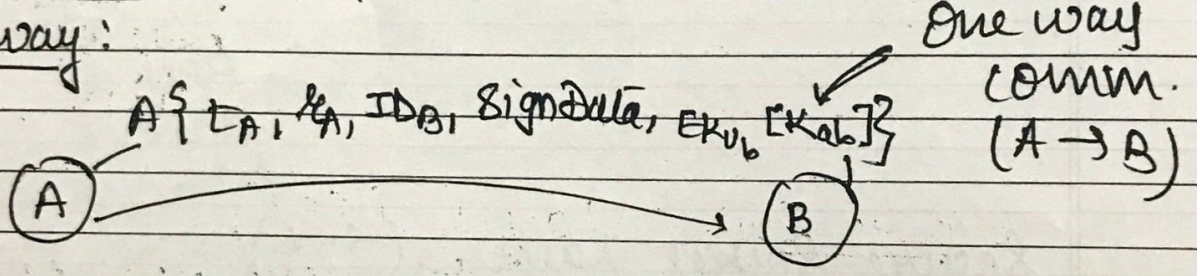
Y <<X>> ⇒ means certificate of user X signed by Y





Authentication Procedures

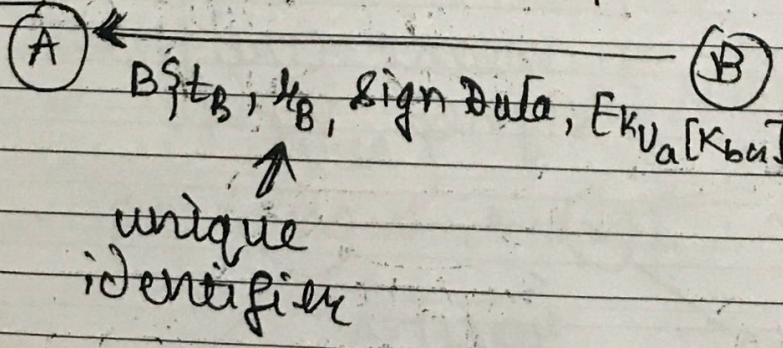
One-way:



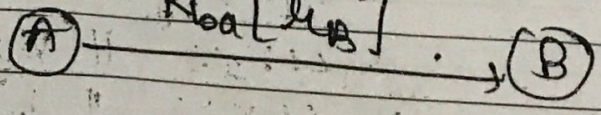
Further comm. messages are encrypted using K_{ab}.

Further, if B wants to communicate with A:

Two way: (after above step is complete)



3-way authentication: (after above 2 steps)



& on Windows OS - enabled.

PAGE No. / DATE / /

Types of Firewall :

Packet filtering firewall (Works on N/W layer

Parameters:

Source IP
Destination IP
Soure & Dest. Port numbers
IP Protocol Field // version.
Interface. (etho etc.)

action	src	dest.	Dest. Port	flags	command
Allow	*				
Allow	*				
Block	*	*	*	25	

↑ Blocks all requests to dest. port 25 (let's say where mail server is running)

Application Level Gateway.

Circuit Level Gateway

PAGE No.
DATE / /

Birthday attack.

1. The source, A prepares to sign a mess. by appending the appro. m-bit hash code and encrypting it with A's private key.

2. Opponent generates $2^{m/2}$ variations on the mess, all of which convey the same meaning (almost).

3. The two sets of mess are compared to find a pair of mess. that produces the same hash code. Prob. of success > 0.5.

The opp. offers the valid variation to A for sig. This sig. can then be attached to the fraudulent var. for trans. to the intended recipient. Because both the variations have same hash code, they produce same sig. (Success even though encryption key is not known).

2. C(K,M) should be uniformly distributed

Prob. $[C(K,M) = C(K,M')] = \frac{1}{2^n}$

n → no. of bits in MAC

M & M' chosen at random

3. If $M' = f(M)$ [some known transformation]

then, Prob. $[C(K,M) = C(K,M')] = \frac{1}{2^n}$

Requirements for Hash func:

i) H can be applied to a block of data of any size

ii) H produces a fixed-length output.

iii) H(x) is relatively easy to compute for any given x, making both H/w & S/w implementations practical.

iv) For any given h, it is computationally infeasible to find x such that:

$H(x) = h$ [One-way]

For any given block x, it is comp. infea. to find $y \neq x$ such that

$H(y) = H(x)$ [Weak collision]

v) It is comp. infea. to find a pair (x, y) such that $H(x) = H(y)$. [Strong collision]

Round 1: $\frac{2^k}{2^n} = 2^{k-n}$

Round 2: $\frac{2^{k-n}}{2^n} = 2^{k-2n}$

On avg. α rounds needed if $k = \alpha \times n$

If $k \leq n$,
then it is likely that first round will produce a singe match. (More than one key will produce a match).

Thus, brute force may require more effort to discover a decryption key of same length

$M = (X_1 \| X_2 \cdots \| X_m)$ X_i = 64-bit block

$\Delta M = X_1 \oplus X_2 \oplus \cdots \oplus X_m$

$C(K, M) = E(K, \Delta(M))$ Key-length = 56 bit

$\{ M \| C(K, M) \}$ requires atleast 2^{56} enery.

But, replace $X_1 - X_{m-1}$ with $Y_1 - Y_{m-1}$
& X_m with $Y_m = Y_1 \oplus Y_2 \oplus \cdots \oplus \Delta(M)$

Thus, entire mess. $X_1 - X_m$ is replaced with $Y_1 - Y_m$.

Assume opponent knows C but not K.
Requirements of MAC:
$C(K, M') = C(K, M)$
M' should be infeasible to construct on knowledge of M & $C(K, M)$.

Reasons for avoiding encryption:

i) En. s/w is relatively slow. Amt of data to be processed per mess. small but, a steady stream of mess.

ii) En. H/w costs are not negligible.

iii) En. H/w is optimized for large data sizes for small blocks, initialization/invocation overhead is time consuming.

iv) En. algo. may be covered by patents. (Additional cost)

MAC

⇒ Security depends on bit length of key.

⇒ Brute-force requires $2^{(k-1)}$ attempts. (k-bit key)

⇒ If confidentiality not applied, opponent has access to plaintext & covers. MAC.

If $k > n$ (key len. > MAC size)

then, $MAC_1 = C(K, M_1)$

cryptanalyst can perform $MAC_2 = C(K_i, M_1)$

Atleast one key gives $MAC_i = MAC_1$

2^k MACs produced but, only 2^n are unique

∴ No. of keys will produce correct MAC & opponent cannot know which is correct

On avg., $\frac{2^k}{2^n} = 2^{k-n}$ keys will produce a match.

PAGE No.
DATE / /

and cannot afford the time to decrypt all incoming mess. (Mess. are picked at random and checked).

iii) Comp. prog. can be executed without decrypti (every time decrypting is wasteful use of resources). If MAC is attached, it could be checked whenever assurance was req. of the integrity of prog.

iv) Some app. → imp. to authenticate mess. than to keep secret. (Mess. contains and to change parameters at the managed system)

v) Separation of auth. & confi. affords archi. flexibility. (Auth - at app. level while confi. at lower level).

vi) Period of protection > Time of reception; is a req.

Hash func.

⇒ Produces hash code (Mess. digest/ hash value)
⇒ Func. of input mess. - Only (does not use key)
⇒ Hash code is a func. of all the bits of mess. & provides error-detection capability (A change in any bit results in a change in hash code).

|| Diagram (Pg-329)

100-bit msgs. 10-bit MAC

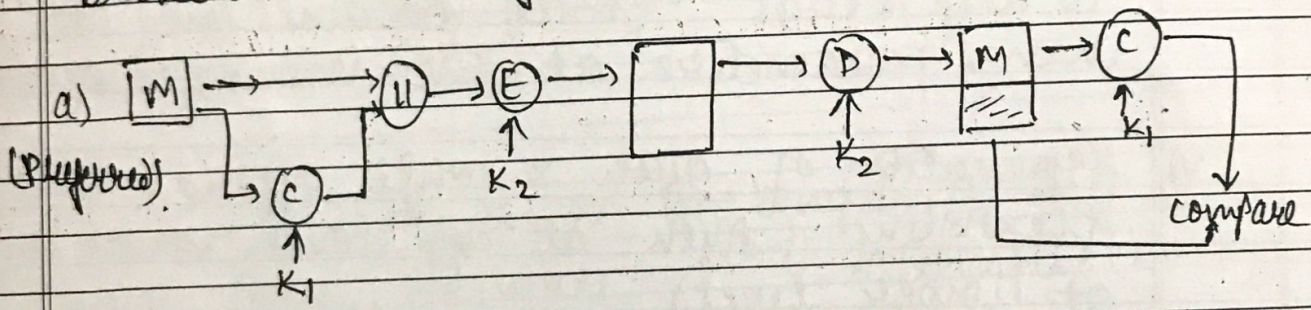
No. of msgs. $= 2^{100}$ possible

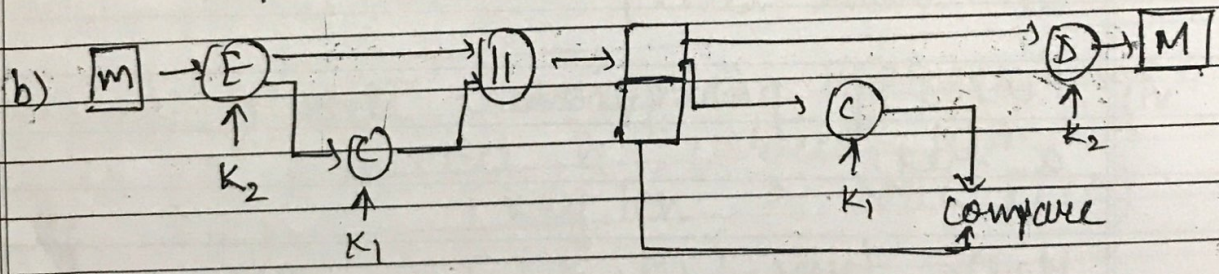
No. of MAC possible $= 2^{10}$

On avg., each MAC value is generated by 2^{90} $\left(\frac{2^{100}}{2^{10}}\right)$ msgs.

with a 5-bit key, then 32 (2^5) different mappings are possible from the set of msgs to the set of MAC values.

⇒ Less vulnerable than encryption to being broken because of mathematical properties.





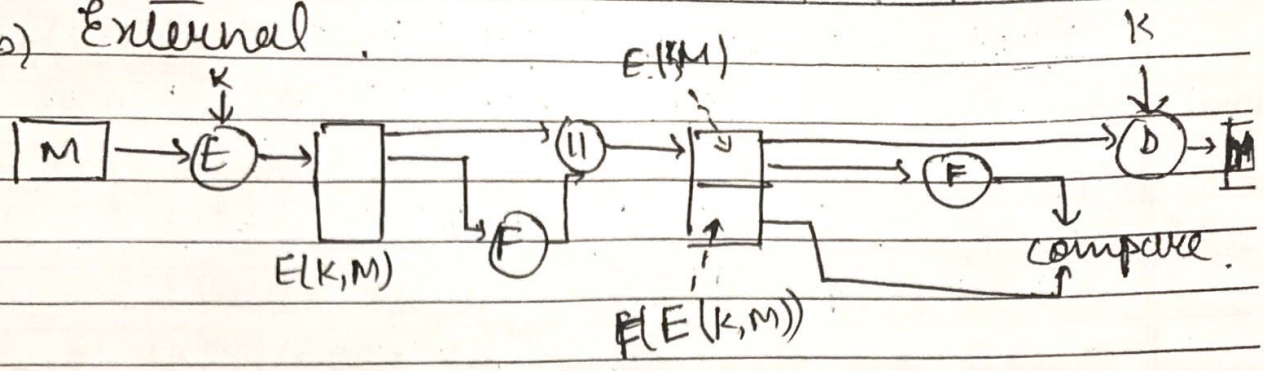
⇒ MAC does not provide digital sig. as both, S & R share the same key.

Uses of MAC:

i) Broadcast - It is cheaper & more reliable to have only one destination responsible for monitoring authenticity.

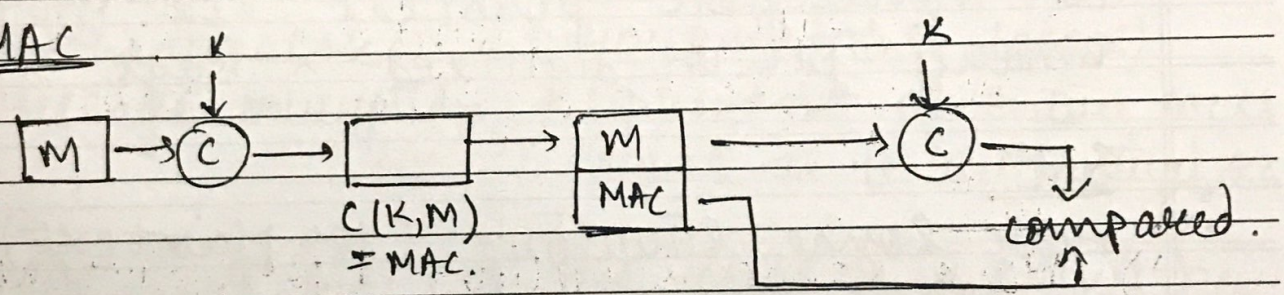
ii) Exchange in which one side has a heavy load

b) External.



ii) Use of comm. architecture consisting of layered protocols. (Ex- TCP/IP)
- Encrypt entire datagram except IP header
If tampered with, the segment holds no significance. (no meaningful header)

MAC



Assume only sender & receiver know about K.
If the received MAC & calculated MAC are equal, then:

i) R is assured that msg. is unaltered
ii) R is assured that msg. is from alleged sender. (NO body else knows the key → no one can prepare proper MAC)
iii) If msg. includes seq. no., then the R can be assured of proper seq. because attacker cannot successfully alter the seq. no.

⇒ MAC ≈ encryption but, MAC need not be reversible. (no decryption)

DATE / /

Problem: Any input x at B is accepted and $Y = D(K, X)$ calculated.

Y may be a meaningless seq. of bits if x is not legitimate

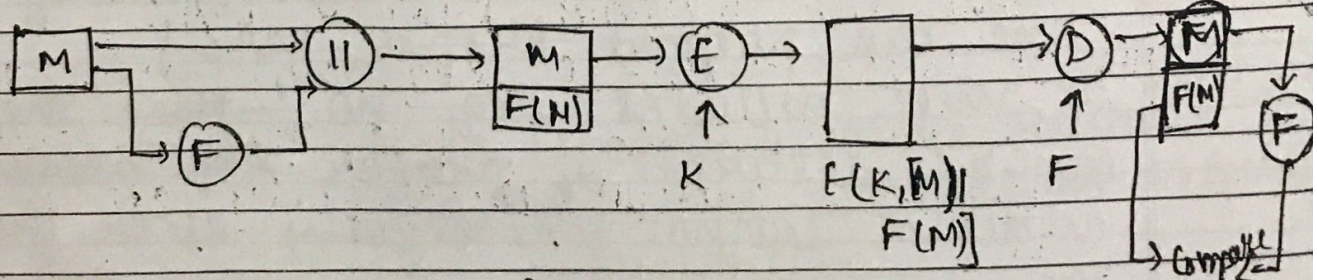
- Based on the freq. analysis of letters of Eng. alphabet in a meaningful text, an acceptable guess can be made if the deciphered text is really meaningful or not.
- Still is difficult to automatically determine whether incoming cipher text will decrypt to intelligible plaintext Ex- in case of binary file or digitized X-Rays

Soln:

i) Have some structure (for plaintext) that is easily recognized but that cannot be replicated without recourse to the encryption func. Ex- Checksum

// Internal & External error control

a) Internal



Authentication provided because opponent would have difficulty in generating ciphertext that, when decrypted would have valid error control bits

PAGE No.
DATE / /

Message authentication functions

Two levels of functionality (for mess. authen. used)

i) Lower level - some sort of func. which produces an authenticator (value to be used to authenticate a message).

ii) Higher level - use the lower level func as a primitive to enable receiver to verify the authenticity of a message.

Types of functions used to produce an authenticator: (grouped into 3 classes)

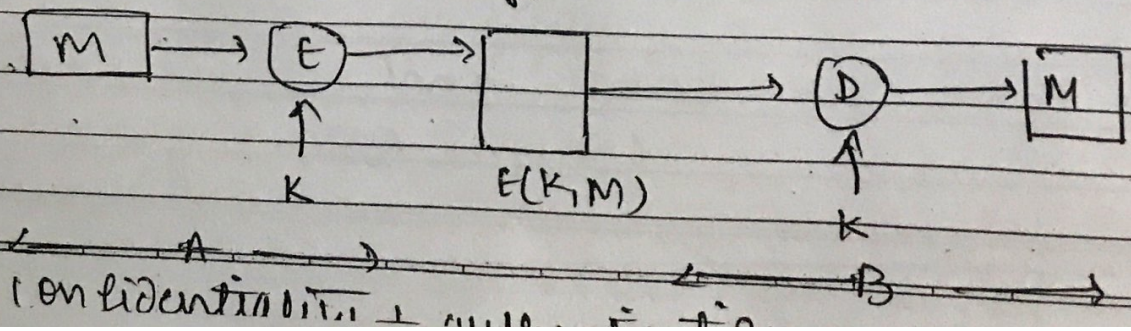
a) Message encryption - ciphertext of entire mess serves as its authenticator

b) MAC: A func. of the mess & a secret key that produces a fixed-length value that serves as the authenticator

c) Hash func.: A func. that maps a mess. of any length into a fixed-length hash value which serves as the authenticator.

Message Encryption

- Symmetric encryption.



(confidentiality + auth...