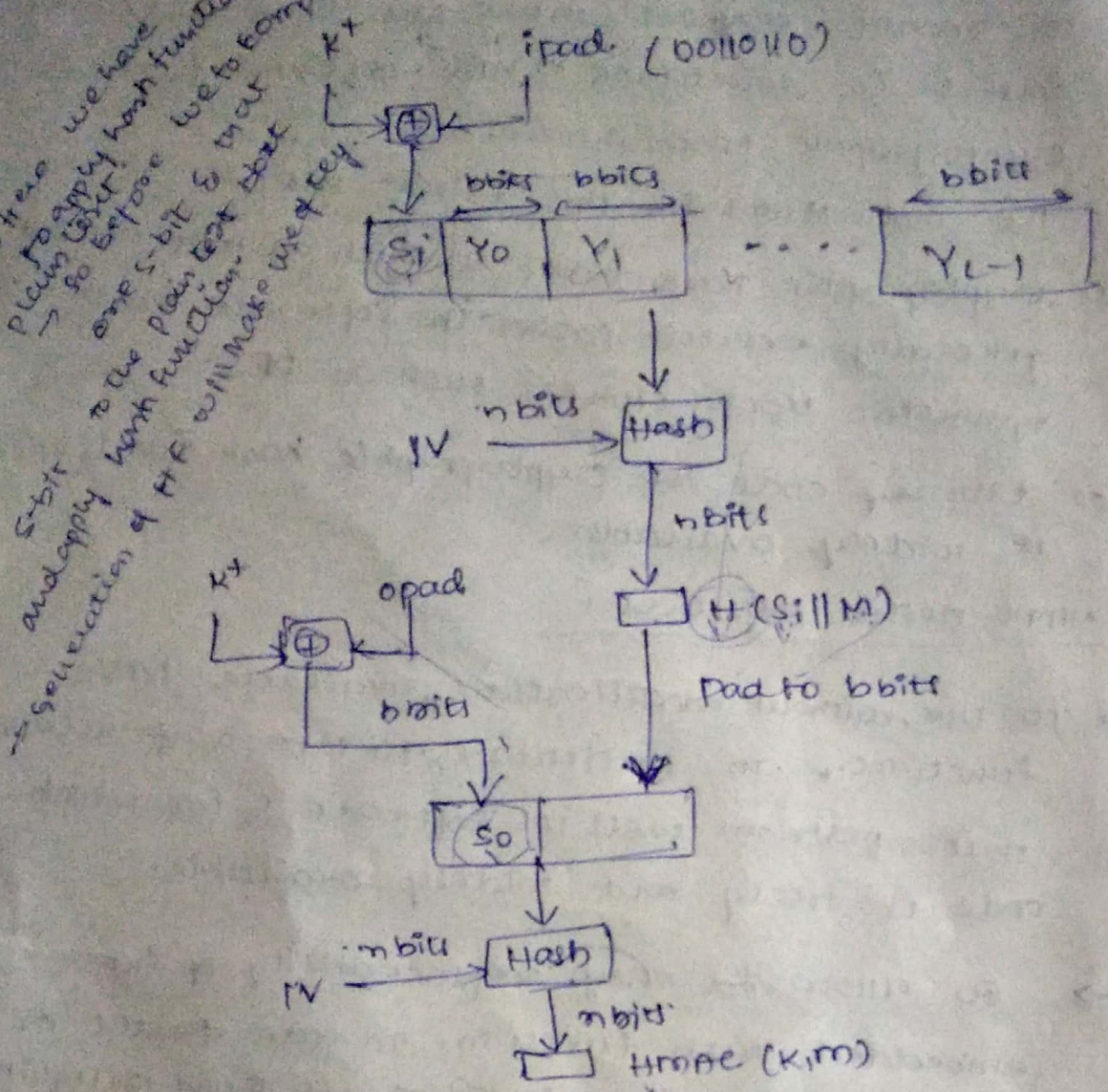
HMAC Algorithm:-



→ Here we have to apply hash function to plain text. So before we to compute one S-bit so that the plain text start and apply hash function. → Selection of HF will make we key.

HMAC Structure

Y_i = ith block of M.

H - embedded hash function

IV - initial value input to hash function.

M - menage input to HMAC

b - no. of bits in block.

n = length of hash code produced by embedded hash function

K = secret key

K^+ = K padded with zeros on left so the result is b bits in length.

ipad = 00110110 repeated b/8 times

opad = 01011100 repeated b/8 times.

HMAC can be expressed as.

$$HMAC(K,M) = H[(K^+ \oplus opad) \| H[(K^+ \oplus ipad) \| M]]$$

we can describe algorithm as follows.

→ Append 0's to left end of K to create a b-bit string K^+

→ XOR K^+ with ipad to produce b-bit block S_i

→ Append M to S_i

→ Apply H to the stream generated in step-3

→ XOR K^+ with opad to produce b-bit block S_o.

→ Append hash result from step 4 to S_o.

→ Apply H to stream generated in step 6. and output the result

A more efficient implementation is possible with 2 quantities:

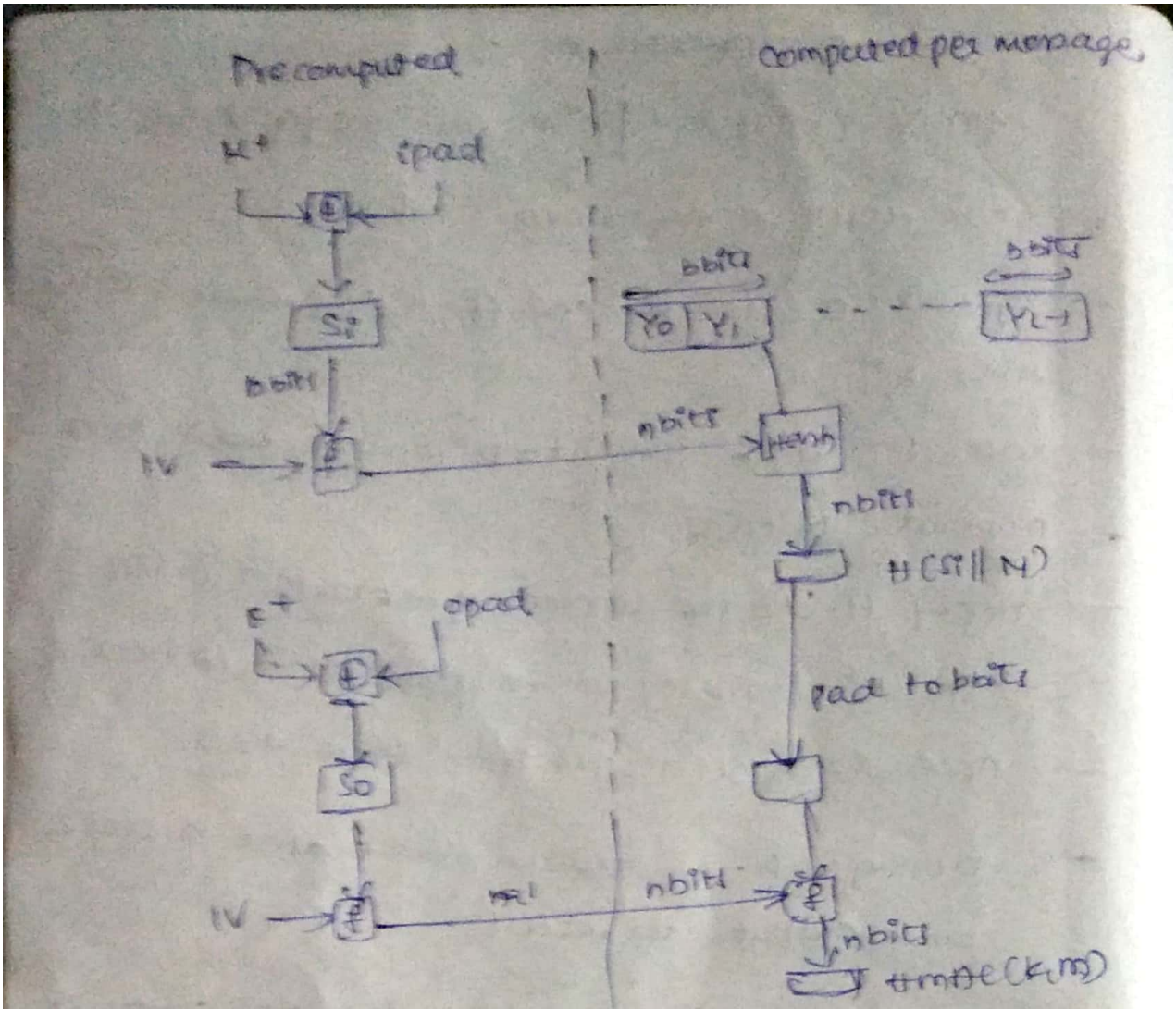
$$f(IV, (K^+ \oplus ipad))$$

$$f(IV, (K^+ \oplus opad))$$

Security of HMAC:-

→ The security of any MAC function based on an embedded hash function depends in some way on the cryptographic strength of underlying hash function.

→ The appeal of HMAC is that its designers have been able to prove an exact relationship b/n strength of embedded hash function & strength of HMAC.



Efficient implementation of HMAC.

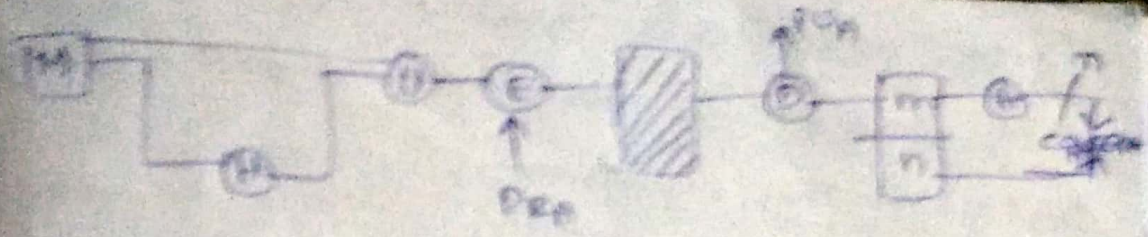
~~This makes use of the Secure Hash Algorithm (SHA) and the Digital Signature Standard~~ | Presents a new digital signature technique

Security services is Digital signature Algorithm

2 types of keys are there private key and publickey. Public key is available for all users and private key is not sharable only it will be with particular user.

M → encrypted using sender private key then automatically we can say this process is a Digital signature process.

consider the plaintext message



2 approaches of digital signature are

(a) RSA approach

(b) DSS approach (or) DAS approach

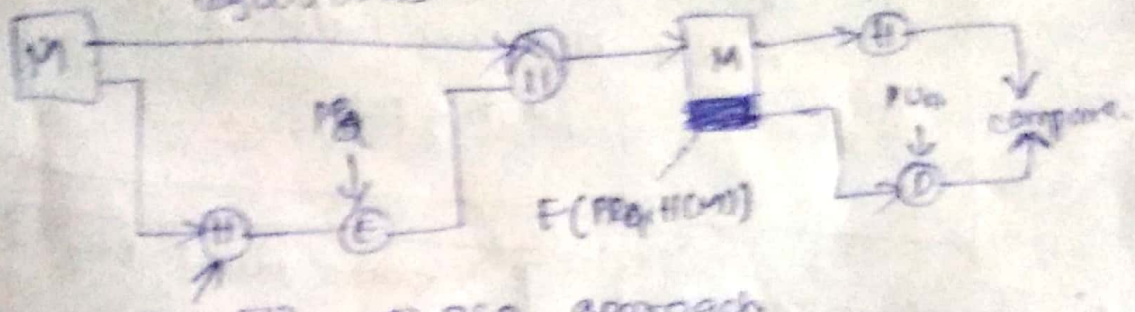
↓

Digital Signature Standard.

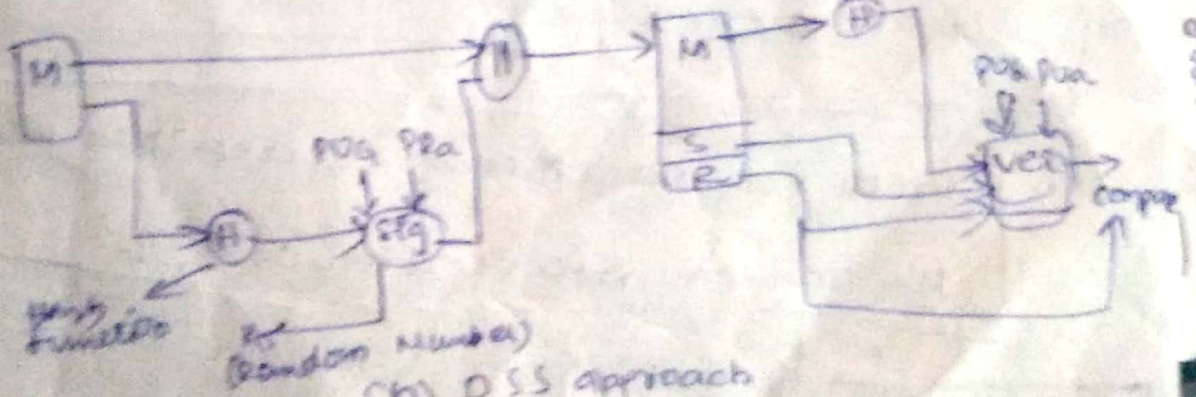
RSA approach :-

Digital signature by using RSA approach

→ DSS is based on difficulty of computing discrete logarithms



(a) RSA approach



(b) DSS approach

→ DSS is an algorithm that is designed to provide only digital signature function. Unlike RSA, it cannot be used for encryption of key exchange.

→ DSS approach also uses hash functions.

→ At sender side signature algorithm will happen
→ At receiver side verification algorithm will happen

Digital Signature algorithm:

Global Public-key components

p: prime number where $2^{L-1} < p < 2^L$ for $512 < p < 1024$ & L is a multiple of 64

q: prime divisor of (p-1)

g: $g = h^{(p-1)/q} \bmod p$ where h is any integer with $1 < h < p-1$

User's private key

x = random integer with $0 < x < q$

User's public key

$y = g^x \bmod p$

User's per message secret number

k = random integer with $0 < k < q$

Signing

$r = (g^k \bmod p) \bmod q$

$s = k^{-1}(H(m) + xr) \bmod q$

Signature = (r, s)

Verifying

$w = (s')^{-1} \bmod q$

$u_1 = [H(m')w] \bmod q$

$u_2 = (r')w \bmod q$

$v = [(g^{u_1} y^{u_2}) \bmod p] \bmod q$

Test: $v = r'$

Sender side is M, γ, S

receiver side is M', γ', S'

Attacks and Forgeries

Types of Attacks in in creating order. Here A denotes the user whose signature method is being attacked, and C denotes the attacker.

- **key-only attack**:- C only knows A's public key.
- **Known message attack**:-
  C is given access to set messages and their signatures
- **Generic chosen message attack**:-
  C chooses a list of messages before attempting to breaks A's signature scheme, independent of A's public key. The attack is Generic because it doesnot depend on the A's public key, the same attack is used against everyone.
- **Directed chosen message attack**:-
  Similar to the generic attack, except the list of messages to be signed is chosen after C knows A's public key but before any signature are seen.
- **Adaptive chosen message attack**:-
  C is allowed to use A as an "oracle". This means A may request signatures of messages that depend on previously obtained messages-signature pairs.

Forgeries are—

- Total break:- C determines the A's private key.
- Universal forgery:- C finds an efficient algorithm that provides an equivalent way of constructing signatures on arbitrary messages.
- Selective forgery:- C forges a signature for a particular message chosen by C.
- Existential forgery:- C forges a signature for at least one message. C has no control over the message.

Digital signature requirements—

- The signature must be a bit pattern that depends on the message being signed.
- The signature must use some information unique to the sender to prevent both forgery and denial.
- It must be relatively easy to produce the digital signature.
- It must be relatively easy to recognize and verify the digital signature.
- It must be practical to retain a copy of the digital signature in storage.

Direct Digital Signature :-

→ The direct digital signature involves only the communicating parties. It is assumed that the destination knows the public key of source.

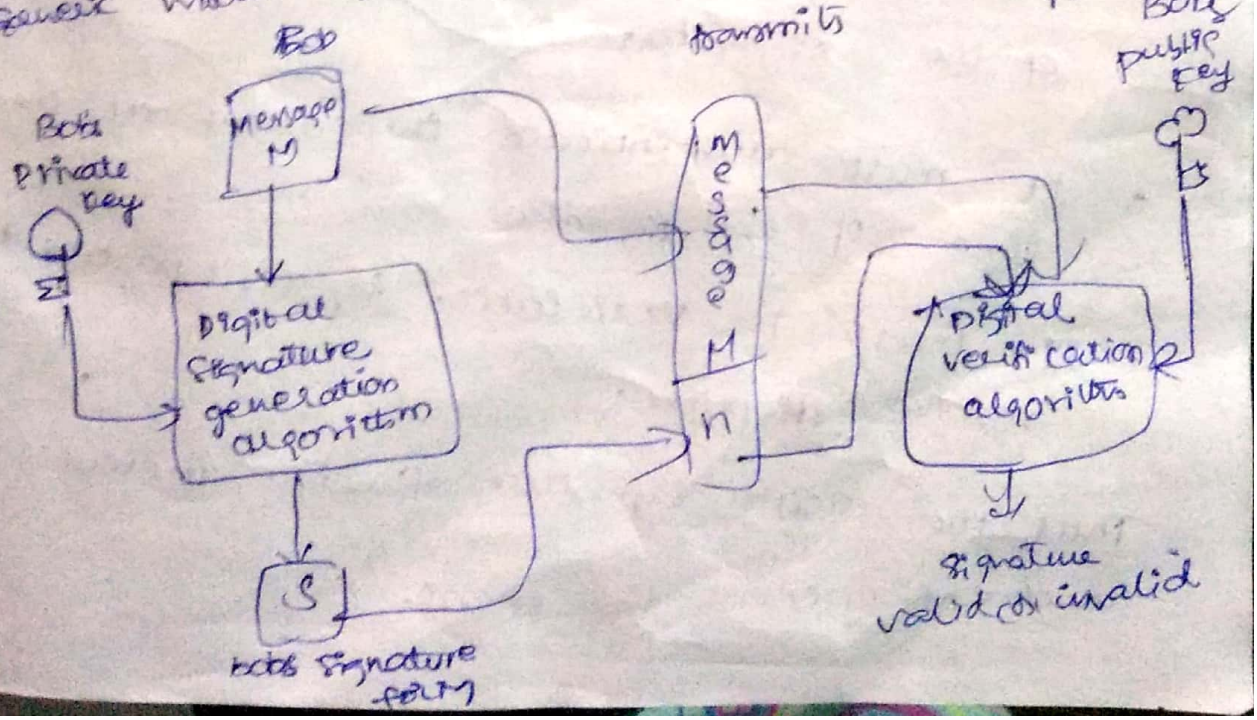
→ The digital signature may be formed by encrypting the entire message with senders private key. (or) by encrypting the hash code with the sender's private key.

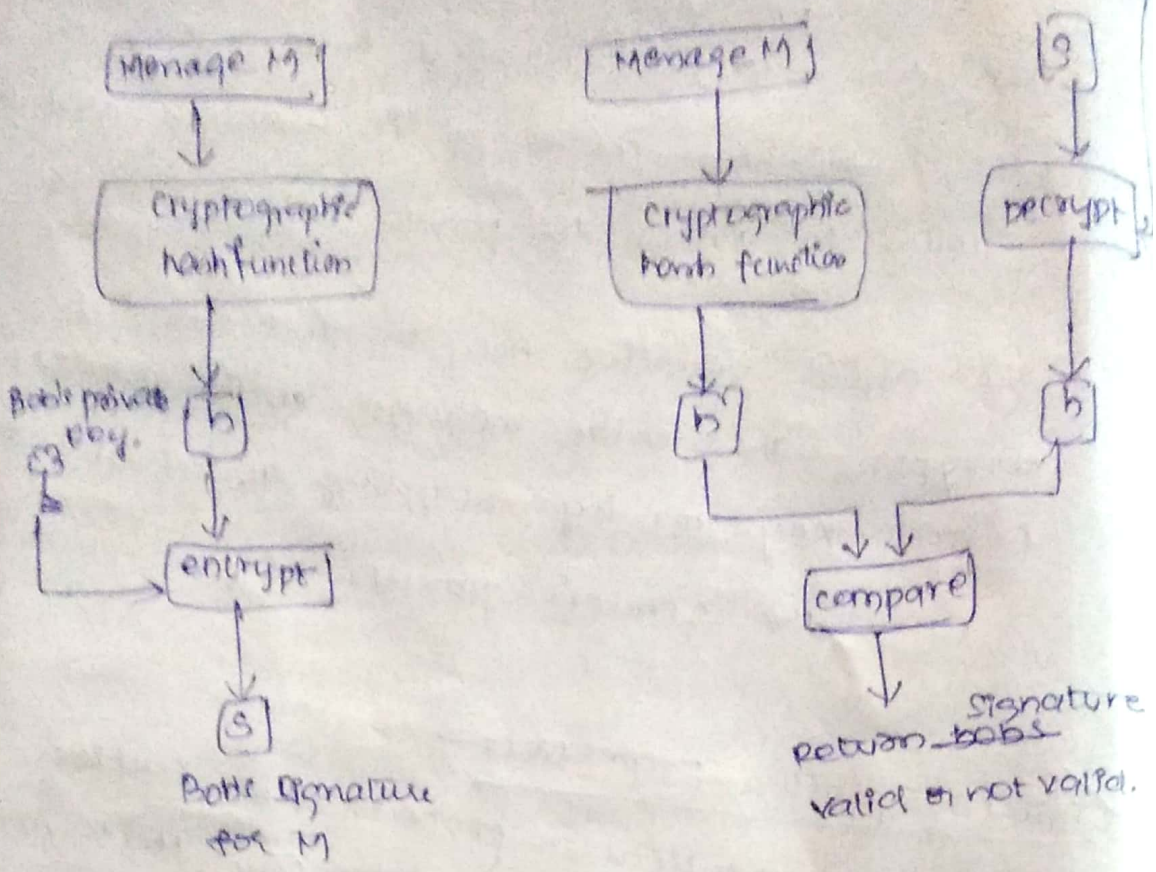
Digital signature properties:-

Message authentication protects two parties who exchange messages from any third party however it doesnot protect the two parties against each other.

Several forms of dispute b/n 2 possible.

Generic model of digital signature:-





→ In these situations where there is not complete trust between receiver and sender.

→ The digital signature must have the following properties.

- It must verify the author and the date & time of the signature
- It must authenticate the content at the time of the signature
- It must be verifiable by third parties to resolve disputes.

Thus, the digital signature function includes the authentication functions.

Message authentication requirements:-

1. Disclosure - release of message contents to any person
2. Traffic Analysis
3. Masquerade
4. content modification
5. sequence modification
6. Timing modification
7. source repudiation
8. destination repudiation

Disclosure:-

release of message content to any person (or) process on possessing by appropriate cryptographic key.

Traffic Analysis:

Discovery of pattern of traffic b/w 2 parties. In a connection-oriented application, the frequency & duration of connections would be determined. In connectionless oriented the number of length of messages between parties would be determined.

Masquerade:

Insertion of message into network from a fraudulent source. This includes creation of message by opponent is come from an unauthorised entity.

Content modification:-

changes to the content of message, including insertion, deletion, transposition & modification.

Sequence modification:-

Any modifications in the sequence of message between parties including insertion, deletion & reordering.

Timing modification:-

Delay or replay of messages. In a connection oriented application, an entire sequence of messages could be replayed or delayed. In connectionless-oriented, an individual message will be delayed or replayed.

Source Repudiation:-

Denial of transmission of message from source

Destination Repudiation:-

Denial of reciept of message from destination.

Security of MACs

We can attacks on MACs into two categories:

(1) Brute Force attack

(2) Cryptanalysis.

Brute Force attack:

A brute-force attack on MAC is difficult undertaking than the brute-force attack on Hash function because it knows the message-tag pairs.

→ To attack on a hash code we have a fixed message x and with n-bits hashcode h=H(x).

→ A brute-force attack of finding collision is to pick a random bit string y then H(y)=H(x) to be checked.

→ Attacker can do this repeadly offline.

→ offline attack can be used on MAC algorithm whether depends on the relative size of key and tag.

→ **computation resistance :-**

Given one or more MAC pairs [xi, MAC(K,xi)] it is computationally infeasible to compute any MAC-text pair [x, MAC(K,x)] for any new input x≠xi;

Cryptanalysis :-

→ The way to measure the resistance of an MAC algorithm to cryptanalysis is to compare the strength of effort required by a brute force attack.

→ That is, an ideal MAC algorithm will requires a cryptanalytic effort greater than or equal to brute force attack.

→ There is much more variety in the structure of MAC's than in hash function, so it is difficult to generalize the crypt-analysis.

Message Authentication Functions

Any message Authentication or digital signature mechanism has two levels of functionality.

-> lower-level
-> higher-level

At lower, there must be some sort of function that produce authenticators.

At higher-level protocol, that enables a receiver to verify the authenticity of message.

There are three classes of different type of functions:-

(1) Hash function
(2) Message encryption
(3) Message Authentication code (MAC).

Hash function:-

A function that maps message of any lengths to fixed-length hash values, which serves as authenticator.

Message Encryption:-

The ciphertext of entire message serves as an authenticator.

Message Authentication code (MAC):-

A function of message and secret key that produces a fixed-length value which serves as authenticator.

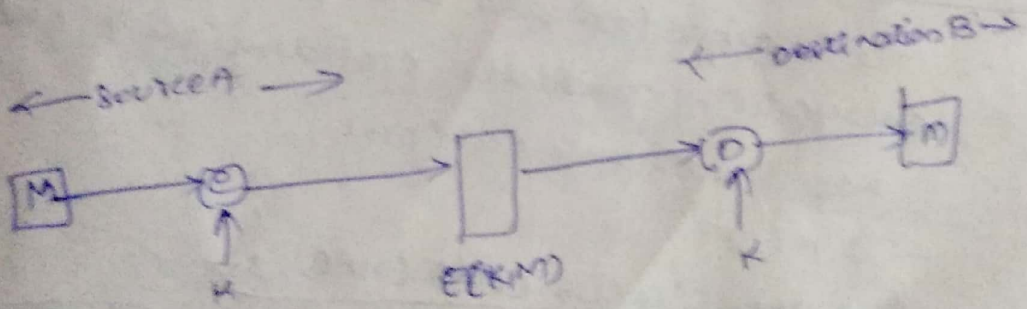
Message encryption:-

It provides a measure of authentication by itself. It has two types.

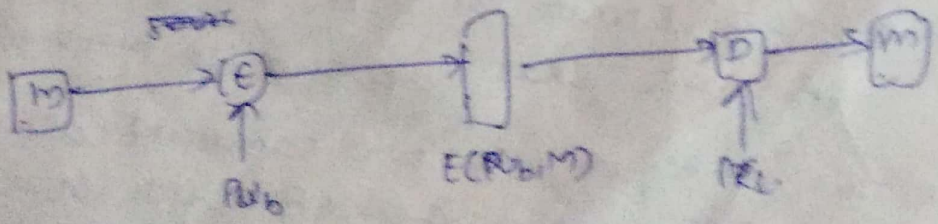
(1) Symmetric encryption
(2) public-key encryption

Symmetric encryption:-

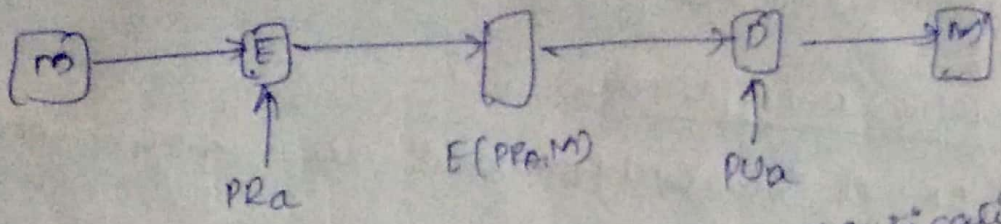
→ It is straightforward use of symmetric encryption.
→ A message M is transmitted from source A to destination B is encrypted by using a secret key K shared by A and B.
→ If no party knows the key, then the confidentiality is provided.
→ No other party can recover the plaintext message.
→ In addition, B assured that the message was generated by A.



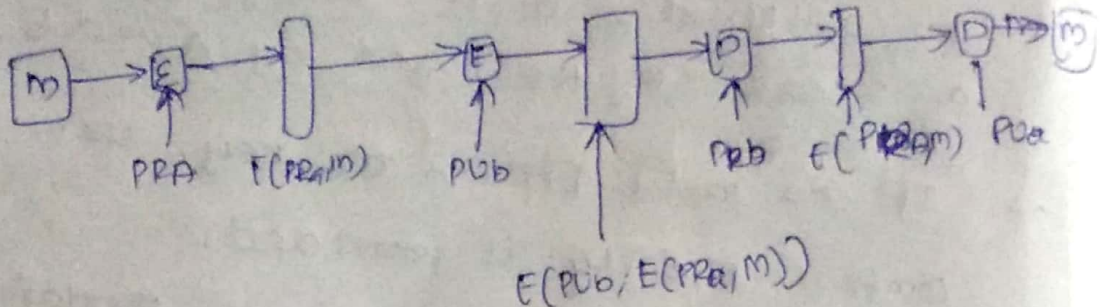
(a) Symmetric encryption: confidentiality & authentication



(b) public-key encryption: confidentiality.



(b) public-key encryption: authentication & signature.



(c) public-key encryption: confidentiality, authentication and signature

public-key encryption:-

→ The straight forward use of public-key encryption provides confidentiality but not authentication.

Message Authentication code (MAC) :-

Def: An alternative authentication technique involves the use of secret key to generate a small fixed-size block of data, known as cryptographic checksum (or) MAC. that append the message.

→ This technique use two communicating parties say A and B which shares the common secrete key K.

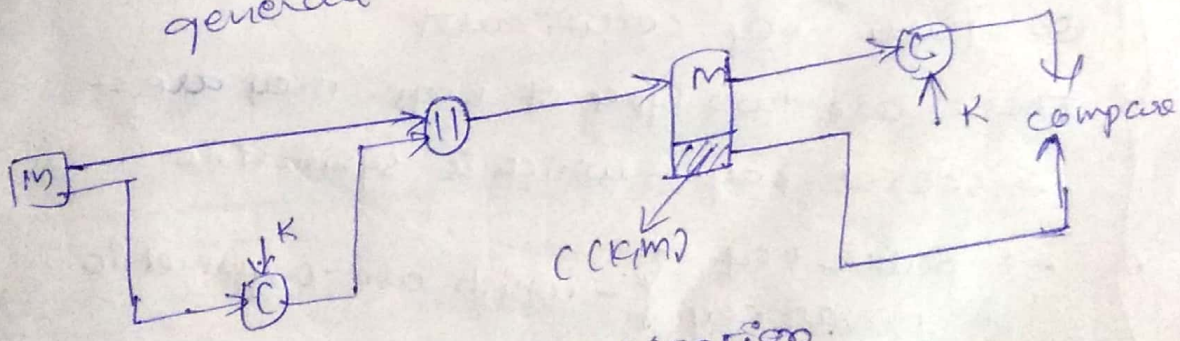
→ When A send message to B, it calculates the MAC as a function of message and the keys.

$$MAC = MAC(K, M)$$

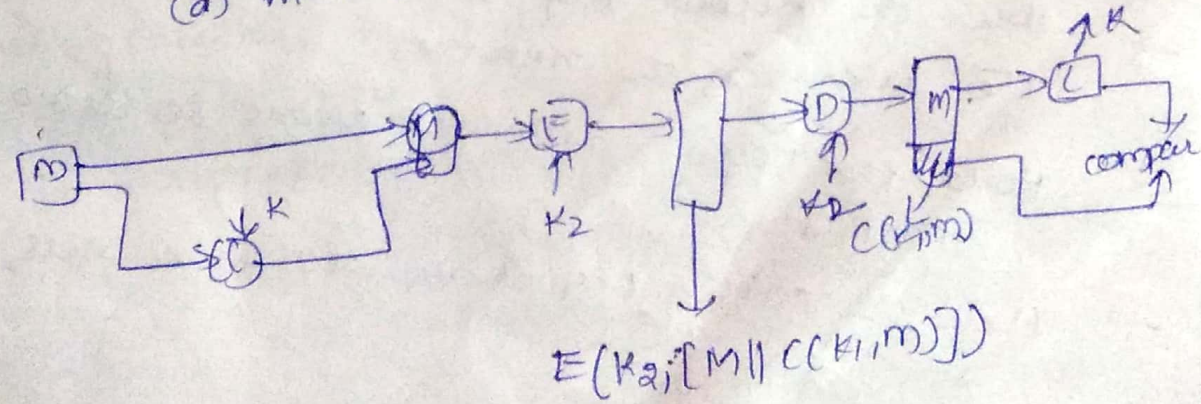
where C = MAC function
K = secrete key.
M = input message
MAC = message authentication code.

→ The message plus MAC are transmitted to a intended recipient.

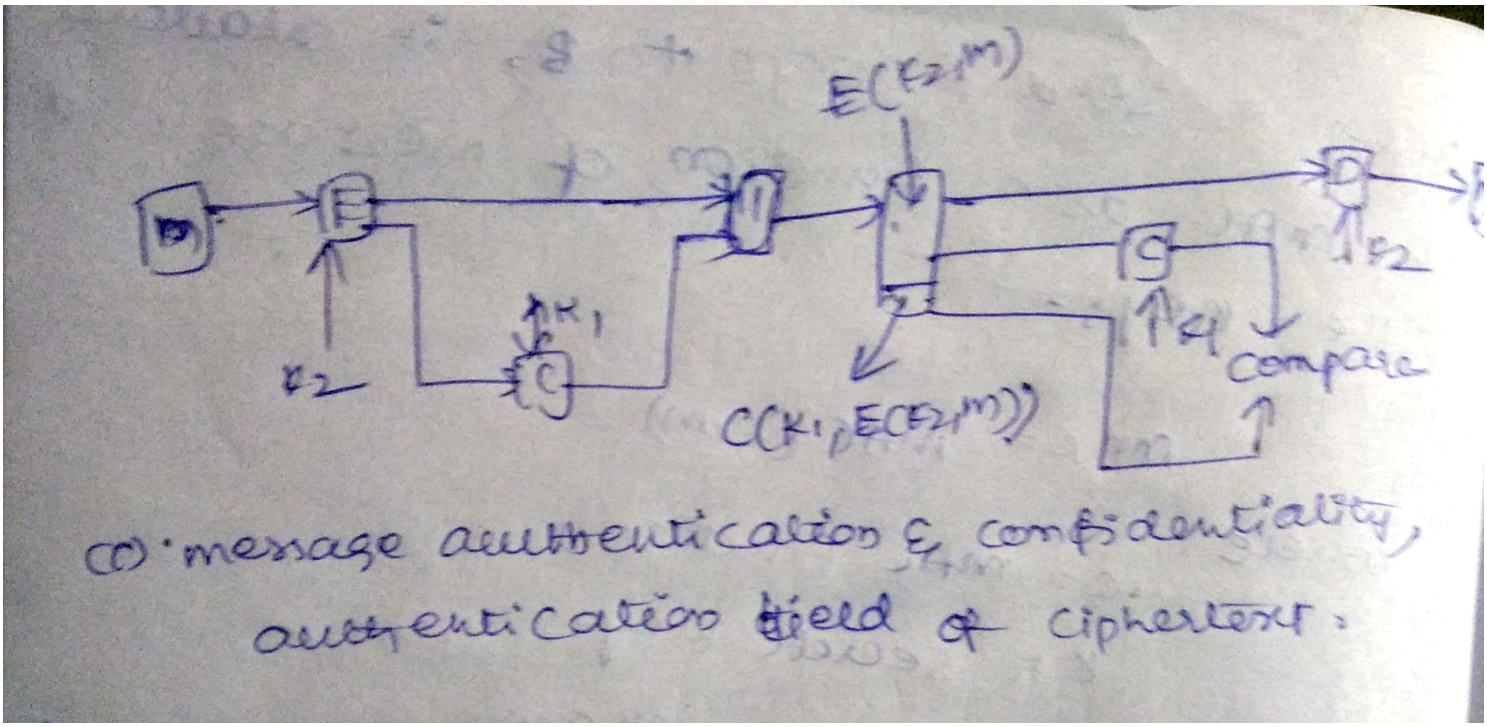
→ It performs same calculation for the received message, using same key, to generate new MAC.



(a) message authentication

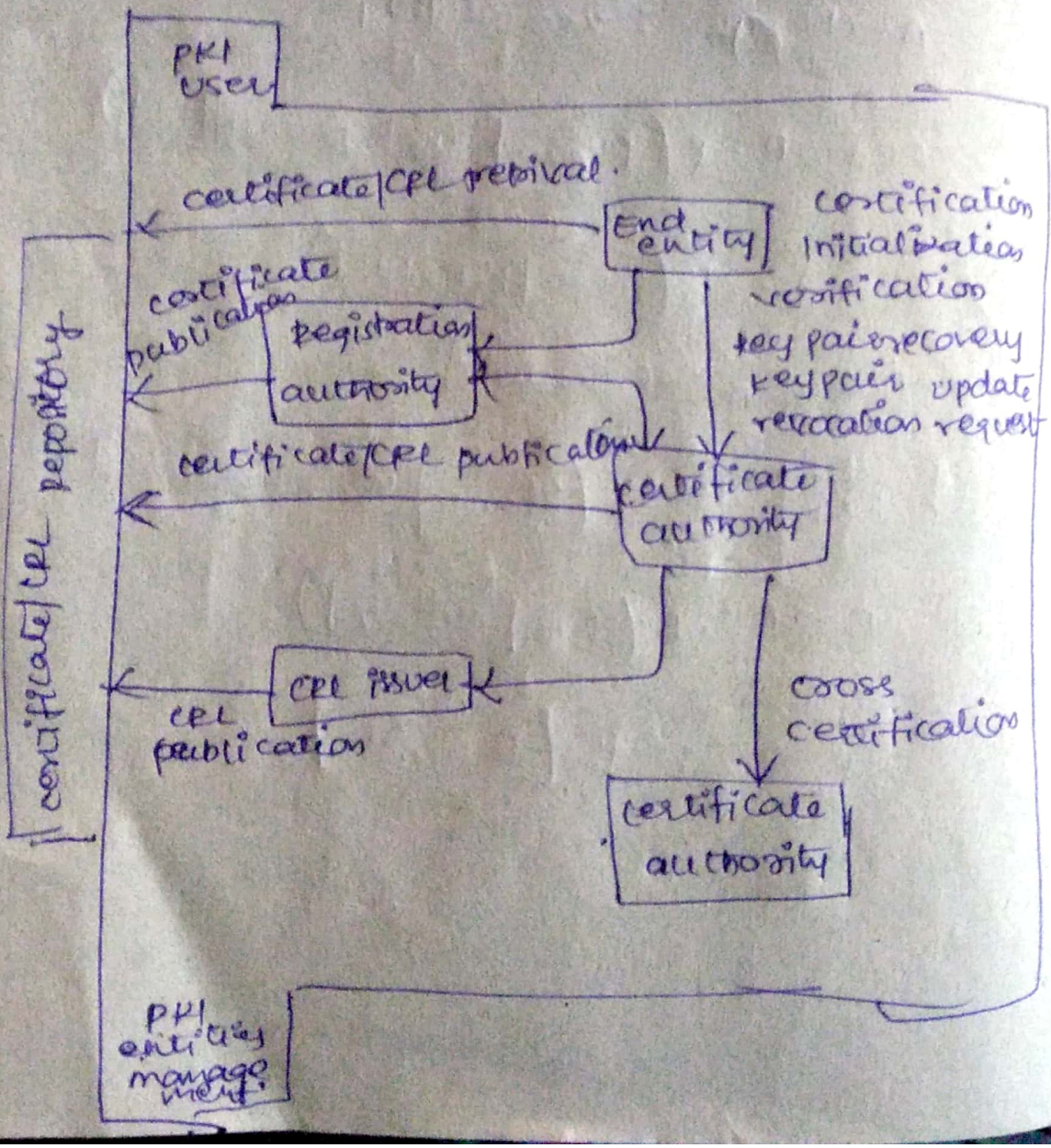


(b) message authentication & confidentiality; authentication of hashed plaintext.



(c) · message authentication & confidentiality, authentication field of ciphertext.

Infrastructure :



Def:
public key infrastructure is a set of hardware, software, people, policies and procedures needed to create, manage, store, distribute and revoke digital certificates based on asymmetric cryptography.

→ The principal objective for developing a PKI is enables secure, convinient and efficient acquisition of public keys.

The above diagram shows the interrelationship among the key elements of the PKIX model. These elements are:-

End Entity:
A generic term used to denote end users, devices (or) any other entitys that can be identified by subject field in public key certificates.

→ It typically consume and support PKI-related services.

Certificate Authority (CA):-
The issuer of certificates and certificate revocation lists. It also may support a variety of administrative functions. Although, these are often delegated with one (or) more Registration authorities.

Registration Authority (RA):-
An optional component that can assume number of administrative functions from the CA. RA often associated with end entity registration process.

CRL issuer:-

An optional component that a CA can can publish the CRL's.

Repository:-

A generic term used to denote any method for storing certificates & CRLs so they can retrieve by end entities.

PKIX Management Functions:-

PKIX identifies a number of management function that are potentially needed to supported by PKIX management protocols.

Registration:

this is the process where user first makes itself known to a CA.

Registration begins the process of enrolling in a PKI.

Registration usually involves in some online offline procedures of mutual authentication.

Initialization:-

Before client system can operate securely, it is necessarily to install key materials that have appropriate relationship with keys stored elsewhere, in infrastructure.

Key pair update:-

All the key pairs needed to be updated regularly and new certificates are issued. Update is required when certificate lifetime expires.

Certification:-

- The process in which a CA issues a certificate for user's public key, returns the certificate to user's client systems, and post that certificate in the repository.

Key pair recovery:

- It is used to support digital signature creation and verification, encryption & decryption or both.
- → It allows end entities to restore their encryption/decryption key pair from an authorized key backup facility.

Revocation request:

Reason for revocation include private key comprises, name exchanges and change in affiliation.

Cross certificate:

Cross certification is a certificate issued by one CA to another CA that contains a CA signature key for issuing.

PKIX Management Protocols:

2 alternatives protocols.

1) RFC 2510 - certificate management protocol
2) RFC 2797 - certificate management message