split into fixed size blocks.
Hash = block 1 ⊕ block 2 ⊕ block 3 ⊕ block 4
- This functions do not provide complexity to make it difficult

* Birthday Attack:
- It is phenomenon from probability theory that the probability of two people having same birthday is surprisingly high even in a small group of people
- It shows that certain hash functions with smaller output size (64 bit hashes) are vulnerable to birthday
- from this we conclude that we need to use large output

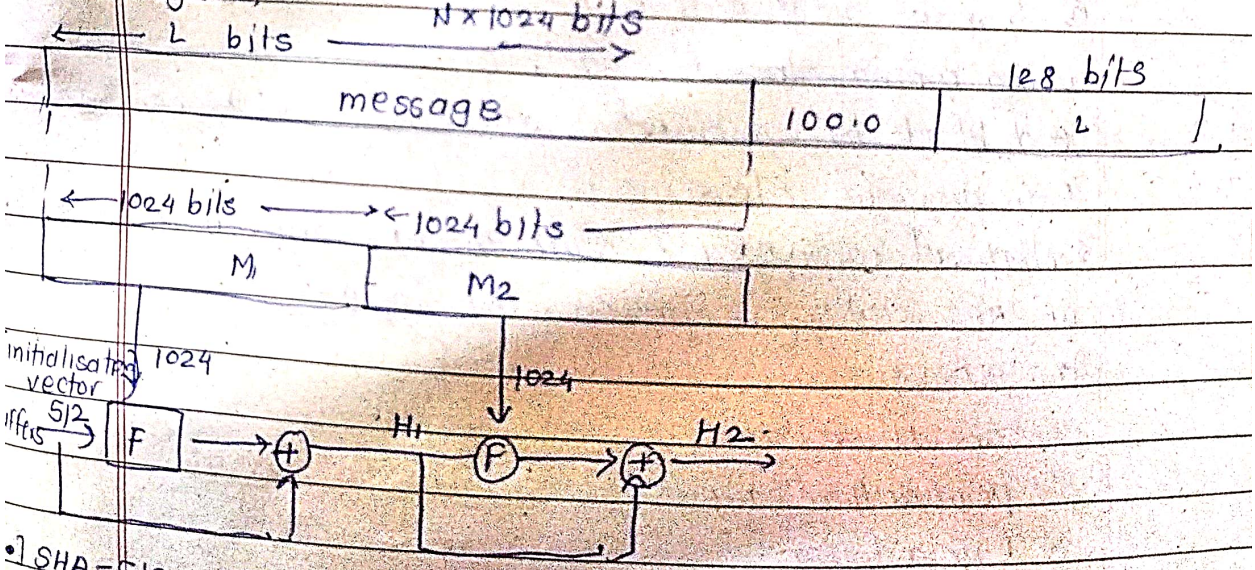
* Hash function and MAC security:
- This involve trying every possible input to find a hash that matches a given hash value
- 128 bits hash looks better, 160 bits better.

* SHA-512 overview:
- secure hash algorithm
- It will produce 512 bit output.
- size of plaintext block = 1024 bits
- No. of rounds = 80
- each round will produce a word (word(w) → 64 bits)
- on each round, we are using k-constant
- we also use Buffers → result
  ↓
  op of one block → input of next block

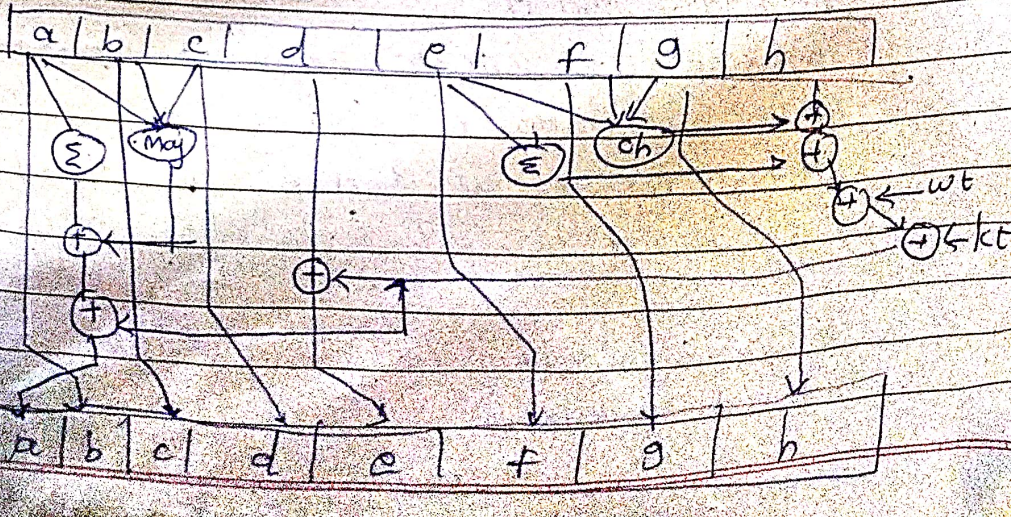
1. plaintext block size = 1024 bits
2. No of Rounds/step = 80
3. Each round → @word = 64 bit
   ↓ generated from plaintext
4. Each round → use constant k
5. Each round we use buffer to store intermediate result and store o/p (hash code)
6. Each buffer size = 64 bit, means we need 8 buffers

① Pad the bits such that 100.... so that length of plaintext is 1024 bit - 128 bits
② Append 128 bit representation of original plain text such that length = multiple of 1024 bits
③ Initialize the buffers (a,b,c,d,e,f,g,h) 64 bits in hexa-decimal format.
④ Process each block of plaintext in 80-rounds
⑤ output stored in buffers is hashcode (512 bits)

* Block diagram:



• SHA-512 function.



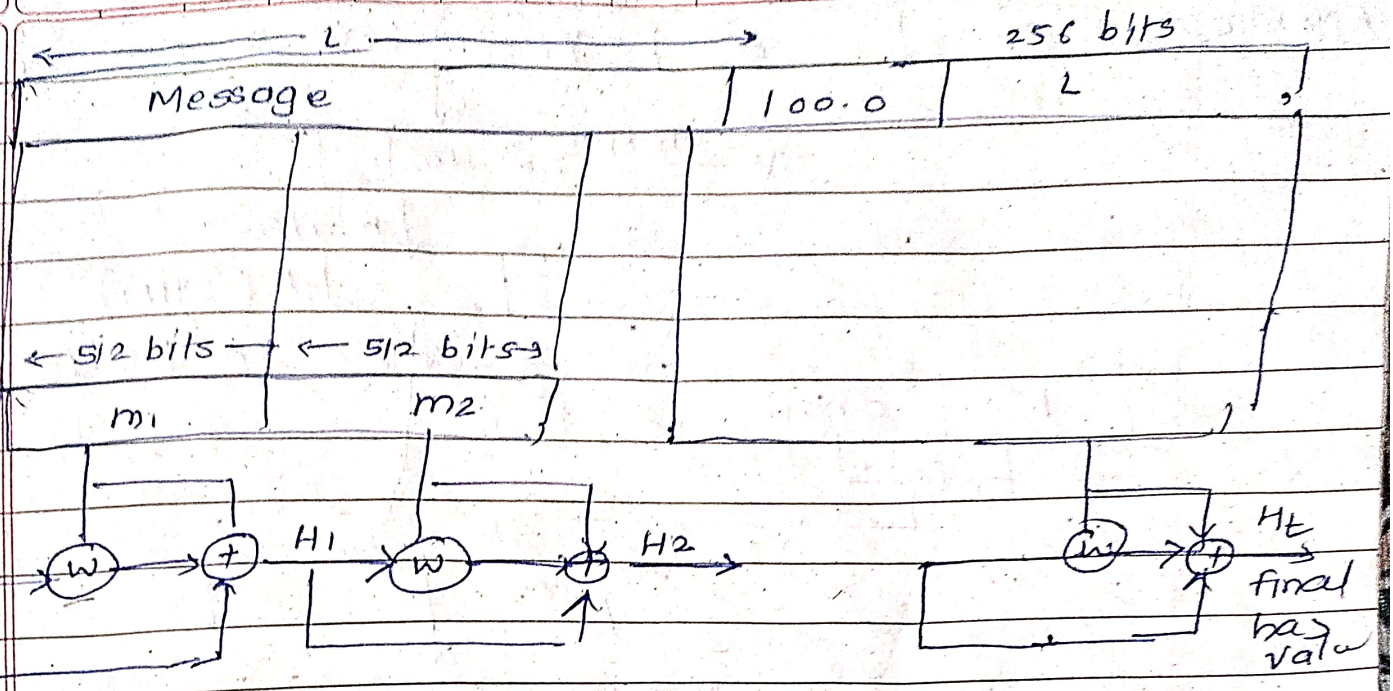
* Whirlpool hash function:
- cryptographic hash function that uses symmetric key block cipher

- uses AES block cipher
- Produces fixed length 512 bit hash
- can process message of any length
- Design to resist the collision attacks, pre-image attacks
- performance is comparable with algorithms like SHA.

working:-
① It basically uses structure around Merkle Damgard construction principle which is commonly used in design principles.
① Padding: The message is padded to ensure it is multiple of 512 bits
- Padding includes appending 1 bit followed by 0's
② whirlpool process the message in 512 bit blocks using compression function.
- Each block is combined with hash value constructed from previous block
③ whirlpool compression function uses modified version of AES cipher to process the input
key steps:
① substitution layer
② permutation layer
③ Mixing layer
④ Add Round key
- This steps are repeated for 10 rounds to ensure strong cryptographic security
- After processing all blocks the final hash value is produced

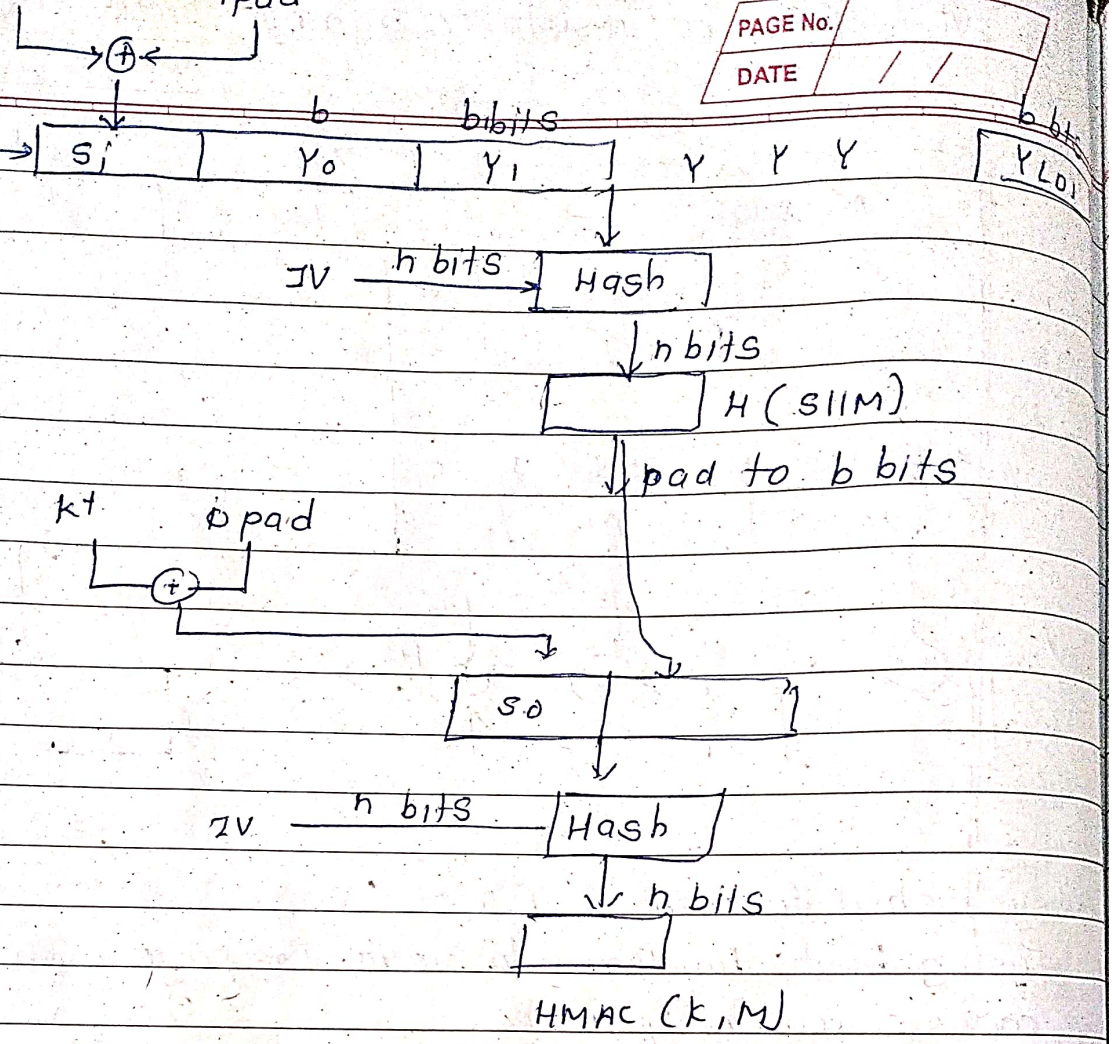
- security of whirlpool:
① Avalanche effect
② Pre-image resilance:
infeasible to find original message when we have

(3) collision
(4) 512 bit output show resistance against



* key hashed function as MAC:
- using hash function to create message authentication code is common practice.
  - Faster than block cipher
  - widely available in libraries
- In this approach, key is concanted with message before applying hash function
  keyed Hash = Hash (key || message)
- disadvantages:
  ① length extension attack: attacker add or appen any additional data to compute MAC.

• HMAC Algorithm
- It is an authentication algorithm
- It is basically combination of hashing and MAC Algorithm
  ① compute s bits
  ② s || m (append s to message)
  ③ Hash function



$\therefore HMAC_K = Hash[(K^+ XOR opad) || Hash[(K^+ XOR ipad || M)]]$

- Here k^+ is the key padded out to size
- If the key is longer than hash block size B
  Hash the key $k = H(k)$
- If key k is shorter than B:
  - Pad the key with zeroes to make it B Byte long
- ipad - ~~block~~ B-byte block filled with 0x36
- opad: B byte filled with 0x5C

- Hmac security:
  1. Provides authentication
  2. Resistance to length extension attack
  3. flexibility
  4. Integrity protection

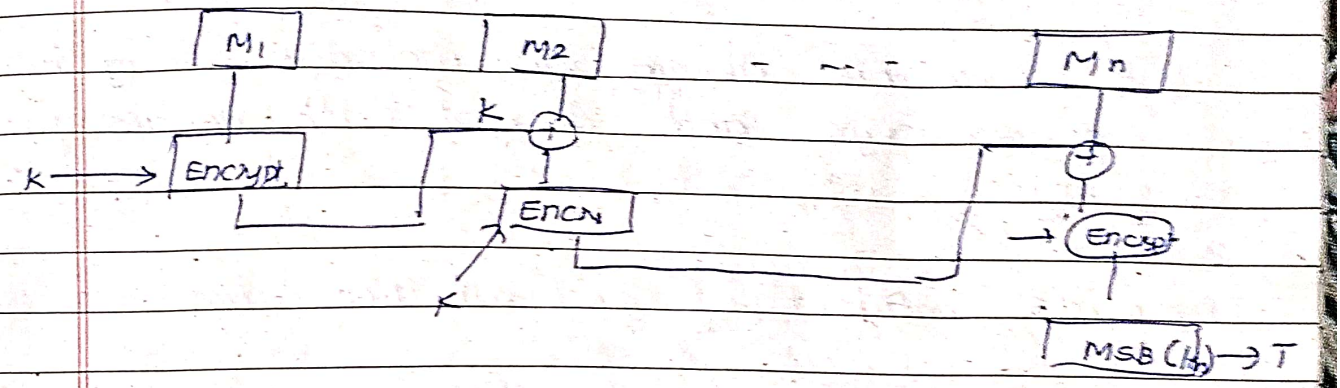
* CMAC - cipher based message Authentication code
- widely used in gov & industry
- it has some message size limitation
- ~~it is block cipher based means message is~~ divided into blocks and encryption is performed on the this block
- consider

A1, A2, A3 ....., An are blocks and k is key

$C_1 = E(k, A)$

$C_2 = E(k, A_2 \oplus C_1)$

$C_3 = E(k, A_3 \oplus C_2)$



PPT 13

* Digital signatures:
- cryptographic method that provide authentication and integrity with features non-repudition
- ① verification of author: confirms the identity of sender
② timestamping of when signature was created
③ ensure message content - tells that msg was not modified since signed
④ provide non-repudition

properties:
① must depend on message signed
② must use ~~information~~ unique to sender like private key of sender is used
③ digital signature generation should be computationally efficient
④ verifying digital signature is straightforward
⑤ computationally infeasible to forge
- without knowing private key it is impossible to generate valid ds.

* Direct digital signature:
- involve secure comm^{n} bet^{h} sender and receiver without involving third party

PAGE No.
DATE / /

- signature rely on sender private key and receiver public key

steps:

① sender compute hash message and encrypt the hash with private key of own. and then again encrypt signature using public key of receiver

$$C = E_{public}(M||S)$$

② Receive first decrypt the message using his private keys and decrypt with sender public key.

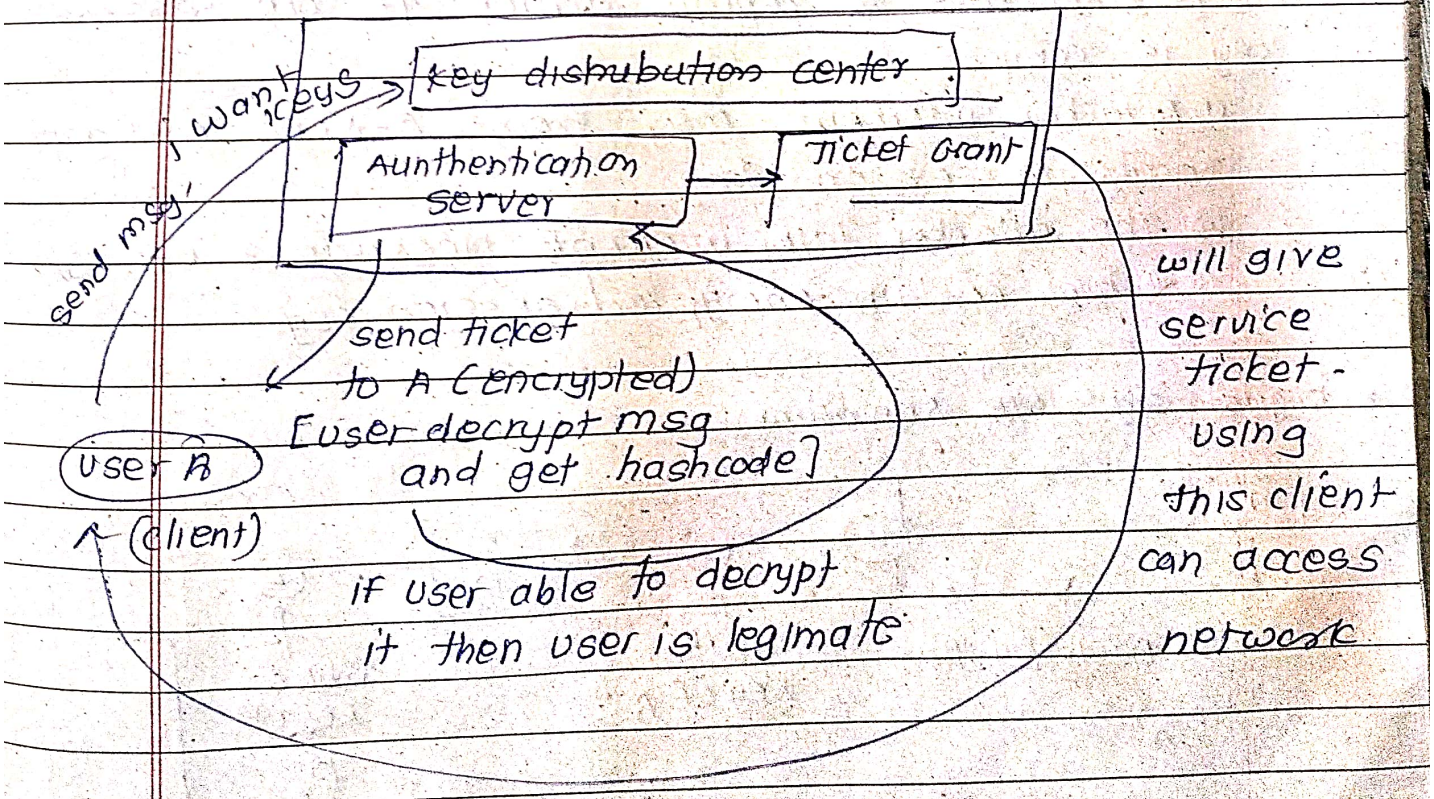
① Simple
② provide confidentiality and authentication

* Arbitrated digital signatures:
- involve use of trusted third party called arbiter
  - who validate signed messag
  - Then dated and sent to recipient
- Requires suitable level of trust in arbiter.
- arbiter may or may not see message.
- can be implemented with either private or public key algorithms.

* Authentication protocols:-
- used to verify the identity of communicating parties and securing exchanging session keys.
- may be one way or mutual
- Issues:
  ① confidentiality: To protect session keys
  ② Timeliness: To prevent replay attacks
- published protocols are often found to have flaws.

* Replay attack:
- A replay attack occur when attacker intercept a valid signed message and then re-send it to gain unauthorized access
- Types of replay attack:
① simple replay :- A hacker intercept login request and replays to the server
② Repetition that can be logged.
③ Repetition that cannot be detected
④ backward replay without modification
- countermeasures:-
① timestamps (use synchronised times)
② use of sequence numbers (usually impractical)
③ challenge response protocol

* kerberos authentication protocol:
- It is network authentication protocol
- client server architecture
- symmetric key (same key for enc and decryption)
- Requires third party for key (KDC)
database of all secret keys



* Needham and Schroeder protocol:

- Uses third party key distribution protocol
- Uses symmetric key encryption
- KDC shares a symmetric key between all users
- This protocol uses for authentication also
- Purpose is to share session key
- There are 5 steps.

ID of A, ID of B, Nonce

1. Alice → KDC : $(ID_A \| ID_B \| N_1)$
2. KDC → Alice → $E_{K_a}[K_s \| ID_B \| N_1 \| E_{K_b}[K_s \| ID_A]]$

K_a is secret key shared between alice and KDC

K_s session key

K_b is shared key between bob and KDC

3. $A \rightarrow B : E_{K_b}[K_s \| ID_A]$

Nonce

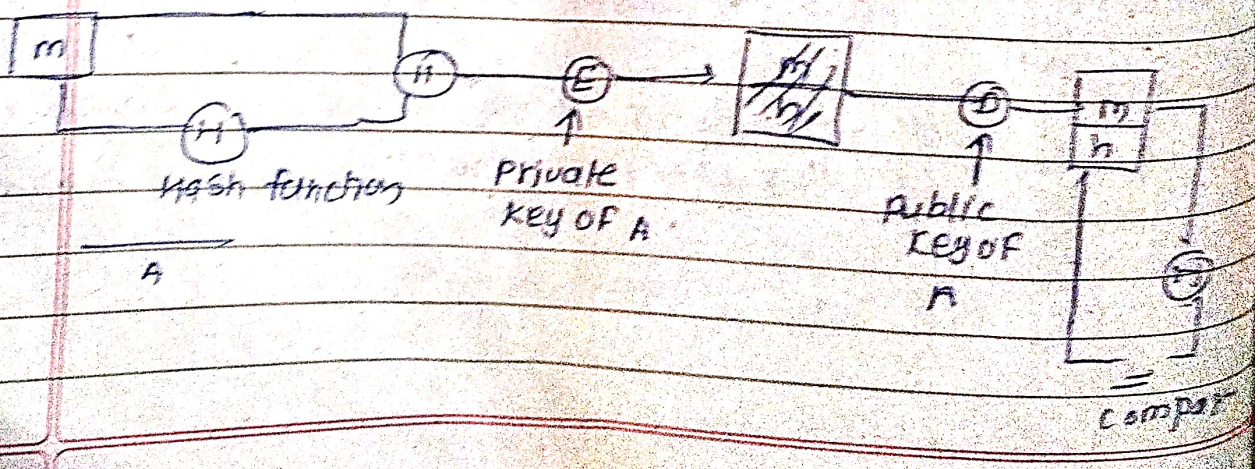
4. $B \rightarrow A : E_{K_s}[N_2]$

↳ session key

5. $A \rightarrow B : E_{K_s}[f(N_2)]$

- purpose is to distribute securely a session key b/w A and B
- but is vulnerable to replay attack if old session key is comprised.
  1. Attacker intercept msg from A → KDC and gain access to session key.
  2. The attacker later intercept message 3 (A → B) which contain encrypted session key.

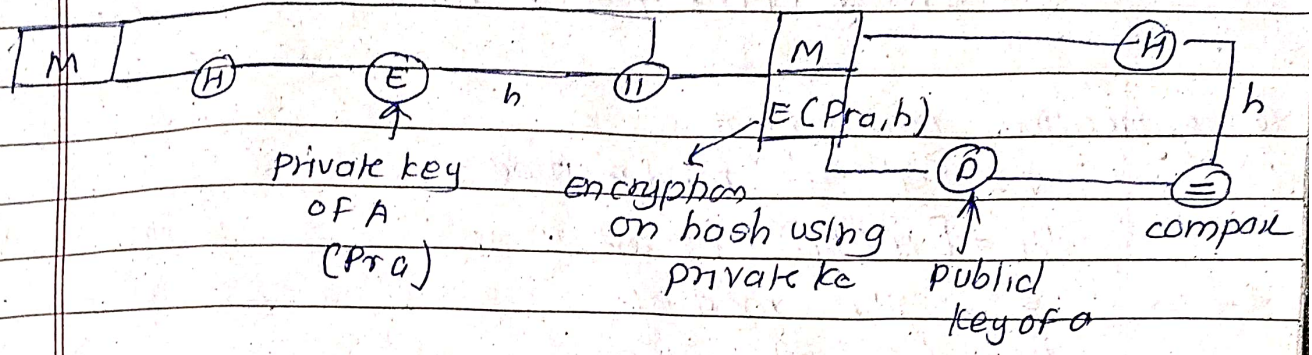
* Digital signature Algorithm



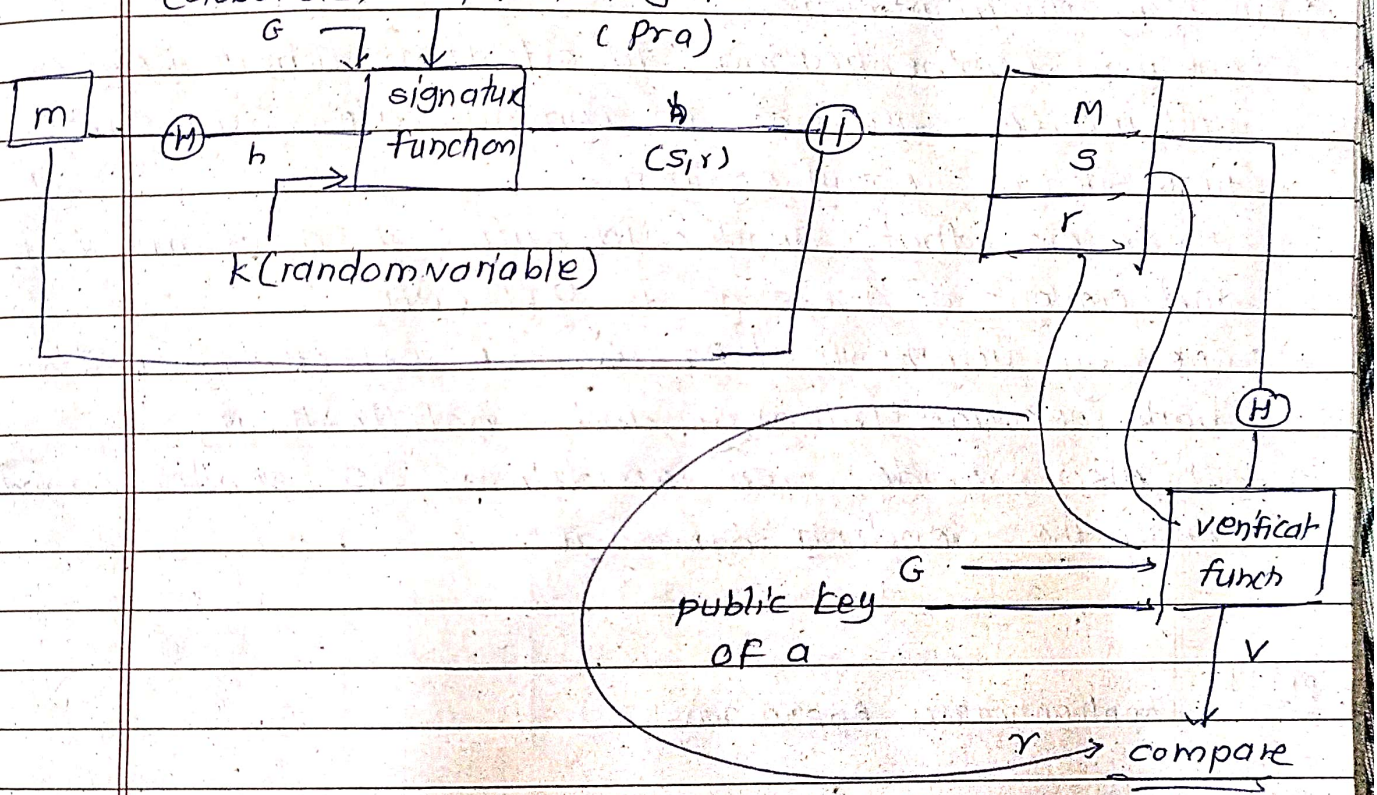
- digital signature
  ① RSA
  ② DSA / DSS

PAGE No.:
DATE : / /

① using RSA:



② using digital signature standard:



① Global components:
p: any prime number in range $2^{l-1} < p < 2^l$
q: Prime divisor of p-1
G: global component $(h^{(p-1)/q} \bmod q)$ ↴ h is any integer in range $1 < h < p-1$
where l is length of bit

② User private key:
x → random no. $0 < x < q$

③ user public key:
$y \rightarrow g^x \bmod p$

④ k is any integer
$\Rightarrow 0 < k < q$

signature component:

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

$s = (k^{-1}(H(M) + x \cdot r)) \bmod q$

⑥ verification component:

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

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

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

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

* one way authentication
- one way authentication is situation where sender and receiver are not in communication at same time such as email system.
- It ensure that sender identity is authenticated and content of messages is protected.
- one way authentication can be effectively with both message integrity and confidentiality.
- It also provide non-repudition as sender cannot deny the sending message

PPT14

Authentication Applications.

① Kerberos:
- developed by MIT
- provide strong authentication for client server applica by using secret-key cryptography and a trusted third party.
- It enable secure communication and access to distributed network services without needing to trust all workstations
- Two version in use: 4 and 5.

- Kerberos uses central authentication server (AS) which act as third party. All user and services on network trust this central server for authenticating identities.

PAGE No.
DATE / /

- Kerberos relies on symmetric cryptography to ensure all communication between client and server are tamperproof

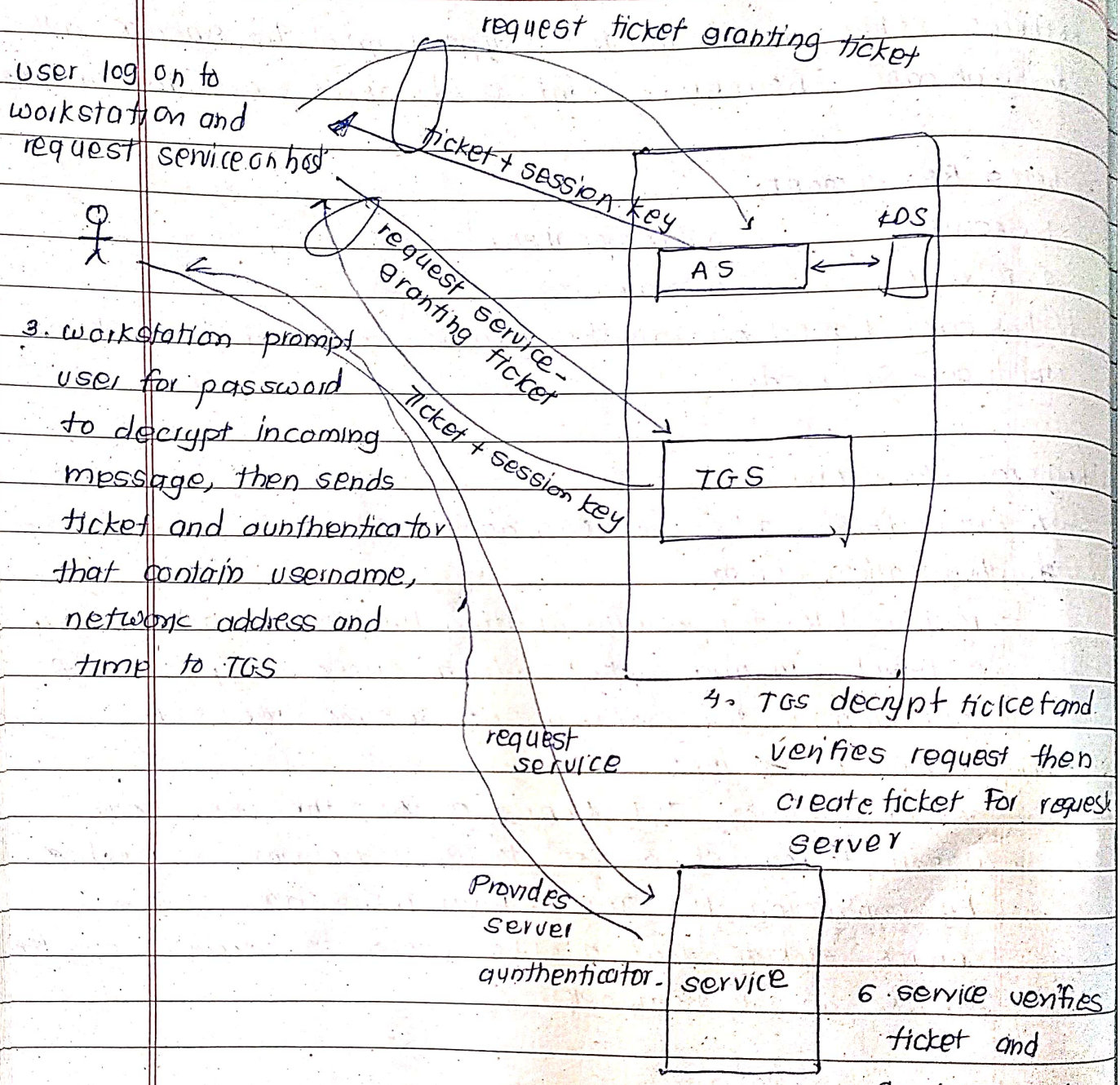
1) Kerberos Requirement:
1. Security
2. reliable
3. transparent
4. scalable

- It basically based on authentication protocol based on Needham-Schroeder

* Kerberos V4 overview:
- A basic third party authentication scheme
- Authentication server:
  - responsible for authenticating the user and issuing a ticket granting ticket which serve as proof of user identity for subsequent service request.
- Ticket granting server:-
  - After obtaining TGT from the AS, the user can use it to request access to other network services by contracting the TGS. Then TGS then issues service tickets that allows users to access specific services on the network.

* Kerberos Diologue:
① obtain TGT from authetication server. (once per session)
② obtain service granting ticket from TGT
③ client / server exchange to obtain the service
  - on every service request

2. AS verifie user's access right in database, create ticket granting ticket and session key.



3. workstation prompt user for password to decrypt incoming message, then sends ticket and authenticator that contain username, network address and time to TGS

4. TGS decrypt ticket and verifies request then create ticket for requested server

6. service verifies ticket and grant access to service

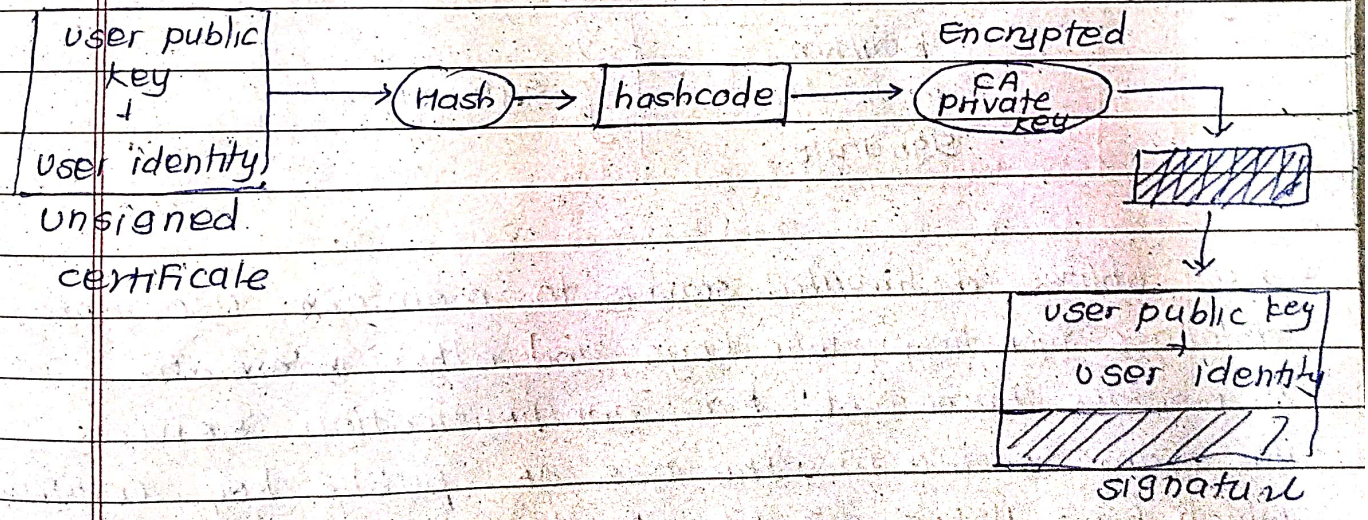
* Kerberos realms:
  - Kerberos environment consist of
    1. Kerberos server
    2. no. of client, all registered with server.
    3. application servers, sharing keys with server
  - Realm is distinct administrative domain in which a kerberos server manages authentication. It consist of
    1. one or more KDC
    2. client that registeresered with KDC
    3. Application servers

- developed in mid 1990.
- It is specified under internet standard under RFC 1510.

- Kerberos allow use multiple encryption algorithms providing flexibility.
- Kerberos V5 standardizes the network protocol ensuring better compatibility
- Allow flexibility in ticket lifetime

Disadvantages:
① double encryption
② password attacks
③ non-standard mode of use

* X.509 Authentication services:
- we are using asymmetric cryptography (using public and private keys)
- certificate authority → It is third party and provides an easy access to public keys.
- certificate authority → generate signature → that signature is attach with user data.



2) fields in X.509 certificate:-

version — 1, 2, 3
serial No — unique sr. no

Algorithm Parameter	generate signature
Issuer name	Name of CA
Not Before, Not after	→ period of validity
Subject Name	user
public key info	Public key + user identity
Issuer unique identity	unique id of CA
subject unique identity	
extensions	→ further extensions
Signature	encrypted fields

- After expiry, we need to renew the certificate
- Parameters of renewated list:

Algorithm parameter	} signature algorithm identifier
Issuer name	
update date	→ renewed certificate date
next update date	→ expiry date
user public key	
Revocation certicate No.	} Revocate certificate
⋮	
Algorithm parameter	} signature.
Signature	

- X.509 utilizes distributed servers to maintain user info database including public keys and other information.
- It provides framework for authentication services
- The main feature is the use of public key certificates which bind user's identity to their public key
- They are widely used
- X.509 does not does not mandate a specific public key algorithm but RSA is recommended

- X.509 have well defined format so they are compatible with almost all systems.

PAGE No. / DATE ../../

- CA containing: certificate fields
  1. version
  2. serial no
  3. signature algorithm identifier → Algorithm, parameter
  4. issuer X.500 name (CA)
  5. Period of validity → not before, not after
  6. subject X.500 name (owner)
  7. subject public key info → algorithm, parameter, key
  8. issue unique identifier
  9. subject unique identifier
  10. extension date
  11. signature (of hash of all fields in certificate) → Algorithm, parameter, encrypted

* obtaining certificate
  1. any user with access to CA can get any certificate from it
  2. only the CA can modify a certificate
  3. because they are not forged, certificates can be placed in public directory.

* CA hierarchy:
  - when multiple CA's are involved, a CA hierarchy is used to establish trust between CA's and user's entities within the system.
  - If two users shares a common CA, both users will trust that CA's public key. As both parties validate each other certificates as both trust same CA.
  - In CA hierarcy, multiple CA's are organized in a tree like structure with each CA having trust on the parent CA, Root CA sits at the top of hierarc

* X.509 include three alternative authentication procedures:

PAGE No.
DATE

① one-way authentication:
- only one party proves its identity to another.
(client)
- one party proves its identity to another party (server) by providing digital signatured on a message.
- The server verifies the signature using the client public key.

② Two-way authentication:
- more secure authentication process where both parties verify each other identity
- Both parties can exchange and verify digital signatures ensuring that each party is authen to other.

③ Three way authentication:
- build upon the two way authentication and additional steps are introduced to make it robust
- Step 1: client send authentication request to the server and server respond with challenge
Step 2: The client signs the challenge with private key and it back to server
Step 3: server verifies the client response and may send its own challenge to client to authenticate itself. Then client verifies server authenticity.
- provides non-repudition and additional secur

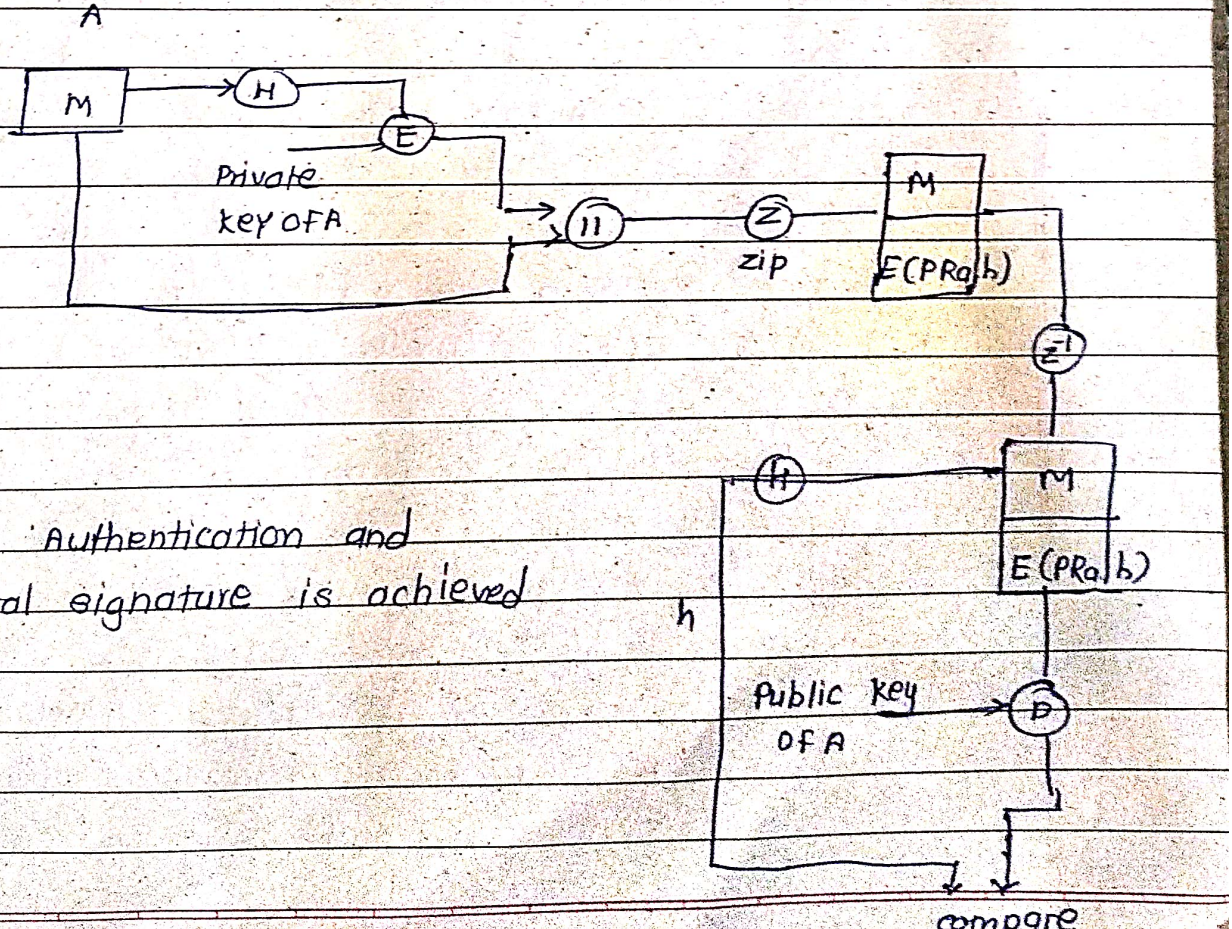
* Module 5 *

PAGE No.
DATE

① Email security.
- email is most widely used and regarded network services
- But lack of inbuilt mechanism for ensuring integrity, confidentiality and authencity of messages
- email messages are usually plaintext so it is easy to read during transmission

• enhancement in email security
① confidentiality: protection from disclosure
② Authentication: verify the identity of sender of msg
③ message integrity: protection from modification
④ non-repudiation: protection from denial by sender

① PGP: Pretty good privacy
- confidentiality, authentication and digital signature servi are provided
- Additional services like email compatibility and email compression is provided



∴ Here Authentication and digital signature is achieved