UNIT-3

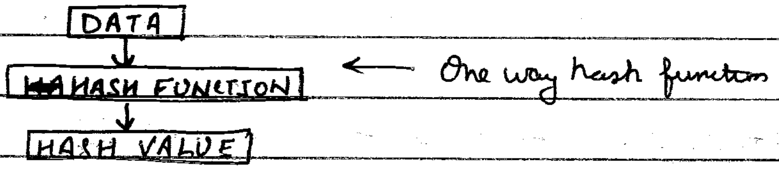
① Hash Functions -

It maps a variable-length message into a fixed-length hash value, or message digest.

The principle object of a hash function is data integrity. This kind of hash function needed for security applications is referred to as a cryptographic hash function.

② One-way Hash function -

It is an important building block to help achieve data integrity. It is function that is easy to compute but difficult to reverse. Also, it is difficult to find two values for which the function would compute the same output value.



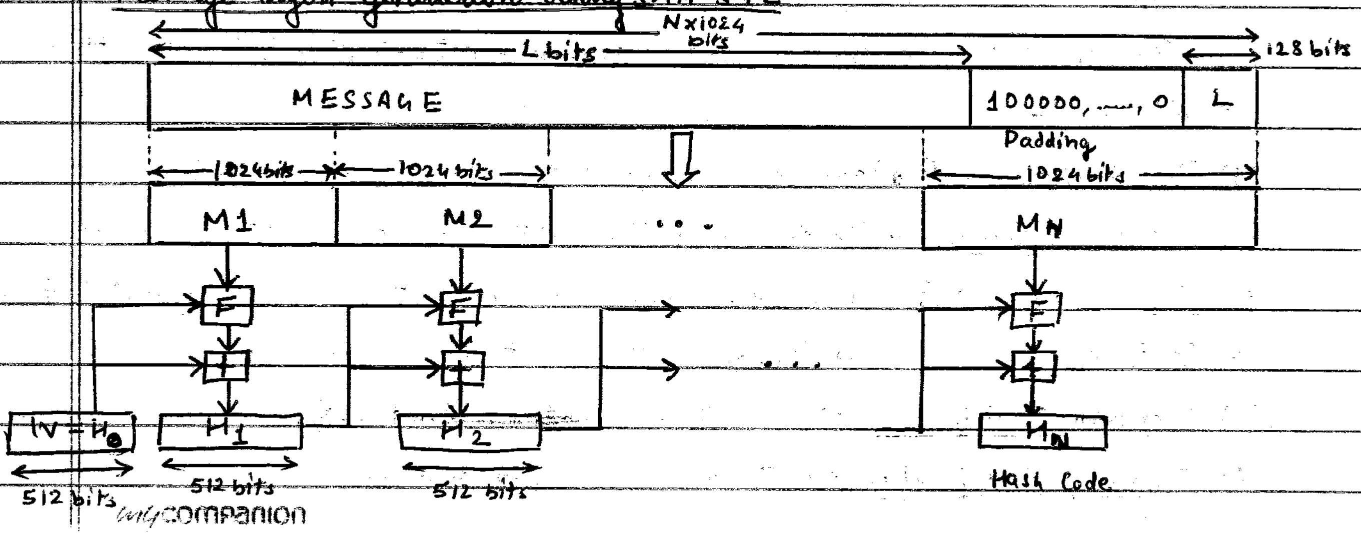
③ SHA (Secure Hash Algorithm) -

ll sizes ne measured in bits

	SHA-1	SHA224	SHA-256	SHA-384	SHA-512
Message Digest Size	160	224	256	384	512
Message Size	$<2^{64}$	$<2^{64}$	$<2^{64}$	$<2^{128}$	$<2^{128}$
Block Size	512	512	512	1024	1024
Word Size	32	32	32	64	64
Number of Steps	80	64	64	80	80

→ SHA-512 -

Message Digest Generation Using SHA-512 -



→ SHA-512 processing steps is given as,

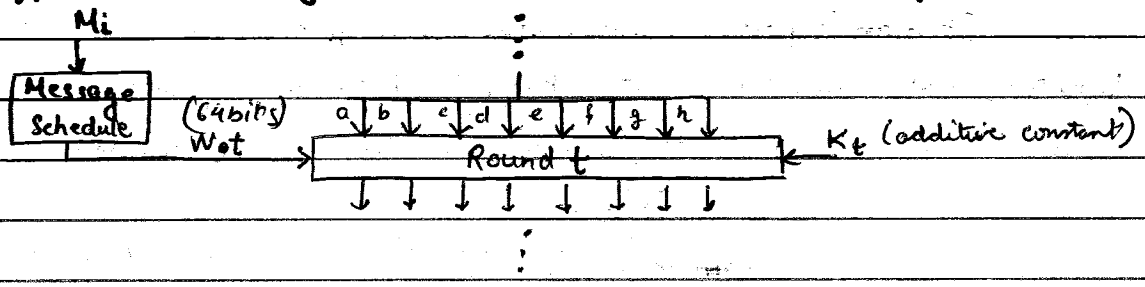
STEP-1 - Append Padding ~~bit~~ bits

STEP-2 - Append length L that 128 bits unsigned integer

STEP-3 - Initialize Hash buffer → 8, 64 bit registers (a, b, c, d, e, f, g, h)

STEP-4 - Process message in 1024-bit (128-word) blocks

Module F consists of 80 rounds. Each round takes as input the 512 bit buffer value, abcdefgh, and update the contents of the buffer.



STEP-5 - Output → From the N^{th} stage is the 512 bit message digest

→ We can summarize the behaviour of SHA-512 as follows -

$H_0 = IV$ IV → initial value of the abcdefgh buffer.

$H_i = SUM_{64}(H_{i-1}, abcdefgh_i)$ N → Numbers of blocks in the message

$MD = H_N$ $abcdefgh_i$ → output of the last round of processing i^{th} message block

SUM_{64} → addition modulo 2^{64} performed separately on each word of the pair of inputs

MD → final message digest value

**** The algorithm is slightly slower than MD5, but the larger message digest makes it more secure against brute force collision and inversion attacks.

(4) Message Authentication Requirements -

In the context of communication across a network, the following attacks can be identified -

(1) Disclosure → Release of message contents to any person or process.

(2) Traffic analysis → Discovery of the pattern of traffic between parties

(3) Masquerade → Insertion of messages into the network from a fraudulent source

(4) Content Modification → Changes to the contents of a message.

(5) Sequence Modification → Modification ~~between parties~~ of sequence of messages

(6) Timing Modification → Delay or replay of messages.

(7) Source repudiation - Denial of transmission of message by source.

(8) Destination repudiation - Denial of receipt of message by destination.

(5) Message authentication functions -

Types of functions that may be used to produce an authenticator are-

(1) Hash functions - A function that maps a message of any length into a fixed-length hash value, which serves as the authenticator.

(2) Message Encryption - The Ciphertext of the ~~entire as~~ entire message serves as its authenticator. Basic uses of Message encryption are-

(i) Symmetric Encryption → confidentiality and authentication

(ii) Public Key Encryption → confidentiality, authentication and signature

(3) Message Authentication Code (MAC) - A function of the message and a secret key that produces a fixed-length value that serves as the authenticator.

It is also known as a cryptographic checksum.

$$\boxed{MAC = C(K, M)}$$

C → MAC function, K → shared secret key, M → input message

Basic use of message authentication code are -

(i) Message authentication

(ii) Message authentication and confidentiality → authentication tied to plaintext or ciphertext. (two separate key are needed)

A MAC function is similar to encryption. One difference is that the MAC algorithm need not be reversible.

MAC attacks - Brute Force Attacks and Cryptanalysis

(6) Kerberos -

It is an authentication service developed as part of Project Athena at MIT. Kerberos provides a centralized authentication server whose function is to authenticate users to servers and servers to users. It relies ~~on~~ exclusively on symmetric encryption. It handles mainly three threats -

(1) A user may gain access to a particular workstation and pretend to be another user operating from that workstation.

(2) A user may alter the network address of a workstation so that the requests sent from the altered workstation appear to come from the impersonated workstation.

(3) A user may eavesdrop on exchanges and use a replay attack to gain entrance to a server or to disrupt operations.

Two version of Kerberos are in common use. Version 4 implementation still exist. Version 5 corrects some of the security defences of version 4 and has been issued as a proposed Internet Standard.

→ Kerberos Design -

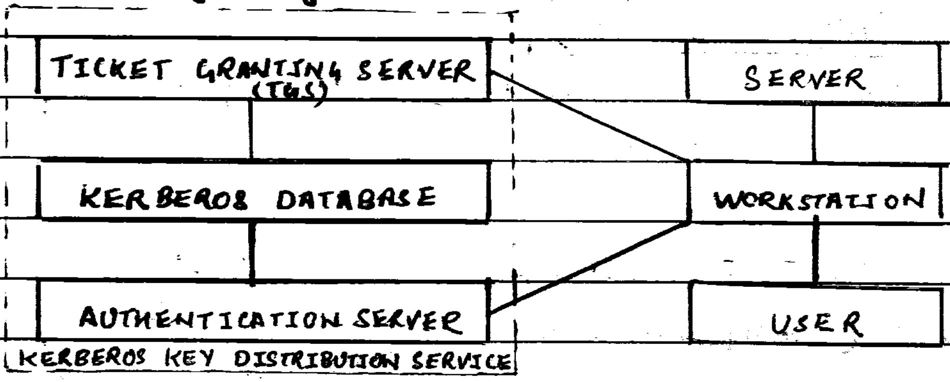
(1) User must identify itself once ~~the~~ at the beginning of a workstation session

(2) Passwords are never sent across the network in cleartext (or stored in memory).

(3) Every user has a password

(4) Every service has a password.

(5) The only entity that knows all the passwords is the Authentication Server.


TICKET GRANTING SERVER (TGS)
SERVER
KERBEROS DATABASE
WORKSTATION
AUTHENTICATION SERVER
USER
KERBEROS KEY DISTRIBUTION SERVICE

Kerberos uses secret key cryptography that is DES.

Tickets - Each request for a service requires a ticket. A ticket provides a single client with access to a single server.

The TGS seals (encrypt) each ticket with the secret encryption key of the server. Each ticket has a limited lifetime (a few hours)

Ticket Contents - Client Name (user login name), Server name, Client host network address, session key for client/server, ticket lifetime, creation timestamp.

Session Key - Random number that is specific to a session. It is used to seal client requests to server. and also responses.

Authenticator - Prove a client's identity. It includes client user name, client network address and Timestamp. Authenticators are sealed with a session key.

⑦ Message Digest functions –

It change the information contained in a file, (small or large) into a single large number, typically between 128 and 256 bits in length.

Every bit of message digest function is influenced by the function's input. If any bit of the function's input is changed, every output bit has a 50 percent chance of changing.

Given an input file and its corresponding message digest, it should be computationally infeasible to find another file with the same message digest value.

⑧ MD5 (Message Digest, Version 5) –

The MD5 algorithm takes as input, a message of arbitrary length, and outputs a 128-bit fingerprint or message digest of the input.

The MD5 algorithm is intended for digital signature applications, where a large file is compressed in a secure manner before being encrypted with a private (secret) key under a public-key cryptosystem, such as RSA.

The same text always produces the same MD5 code.

⑨ SSL (Secure Sockets Layer) –

It is a protocol developed by Netscape for transmitting private documents via the Internet. It works by using a private key to encrypt data which is transferred over the SSL connection.

SSL Protocol is an ~~tcln~~ independent application protocol.

SSL architecture –

<table>
<tr><td>SSL HANDSHAKE PROTOCOL</td><td>SSL CHANGE CIPHER SPEC PROTOCOL</td><td>SSL ALERT PROTOCOL</td><td>HTTP</td></tr>
<tr><td colspan="4">SSL RECORD PROTOCOL</td></tr>
<tr><td colspan="4">TCP</td></tr>
<tr><td colspan="4">IP</td></tr>
</table>

SSL Protocol Stack

SSL Record Protocol provides basic security services to various higher layer protocols. Services are confidentiality and message integrity.

→ Two important SSL concepts are -

(1) SSL Connection - Provide peer-to-peer relationships. The connection are transient. Every connection is associated with one session.

A connection state is defined by the following parameters -

(i) Server and client random byte sequences
(ii) Server write MAC secret
(iii) Client write MAC secret.
(iv) Server write key
(v) Client write key
(vi) Initialization vectors
(vii) Sequence numbers

(2) SSL Session - It is an association between a client and server. Sessions are created by the Handshake Protocol. Session define a set of cryptographic security parameters which can be shared among multiple connections.

A session state is defined by the following parameters -

(i) Session identifier
(ii) Peer certificate
(iii) Compression method
(iv) Cipher specification
(v) Master secret
(vi) Is resumable.

→ Hypertext transfer Protocol (HTTP) - provides the transfer service for Web client/server interaction, can operate on top of SSL.

→ SSL Change cipher Spec Protocol - It consists of a single 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.

1 byte
1

→ Alert Protocol - Convey SSL-related alerts to the peer entity. They are compressed and encrypted, as specified by the current state.

1 byte	1 byte
Level	Alert

Some of the alerts are unexpected message, bad record mac, handshake failure illegal parameter, close notify, no certificate, bad certificate etc.

→ Handshake Protocol - This protocol 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.

Type	Length	Content
1byte	3bytes	≥0 bytes

Other Upper-level layer Protocol (eg-HTTP) →

≥1byte
OPAQUE CONTENT

→ Four Phases to establish a logical connection between client and server are -

PHASE 1 - Establish security capabilities including protocol version, session ID, cipher suite, compression method, and initial random numbers.

PHASE2 - Server authentication and key exchange - Server may send certificate, key exchange, and request certificate. Server signals end of hello message phase.

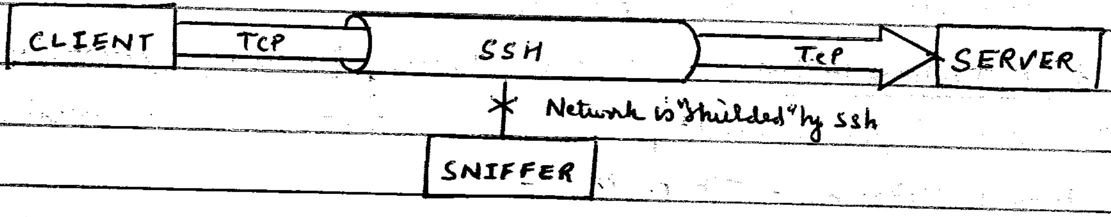
PHASE3 - Client authentication and key exchange - Client send certificate if requested. Client sends key exchange. Client may send certification verification.

PHASE4 - Finish - Change cipher suite and finish handshake protocol.

(10) SSH (Secure Shell) -

It is a protocol for secure network communications designed to be relatively simple and inexpensive to implement.

SSH2 is a more secure, efficient, and portable version of SSH that includes SFTP, an SSH2 tunnelled FTP.



SSH is organized as three protocols that typically run on top of TCP -

SSH USER AUTHENTICATION PROTOCOL	SSH CONNECTION PROTOCOL
SSH TRANSPORT LAYER PROTOCOL	
TCP	
IP	

SSH Protocol Stack

→ SSH Transport layer protocol - It provides server authentication, confidentiality, and integrity. It may optionally also provide compression.

→ HostKeys - A server may have multiple host keys using multiple different asymmetric encryption algorithms. Multiple hosts may share the same host key.

→ Package Exchange - Each packet has - packet length, padding length, payload, random padding, and Message authentication Code (MAC).

Steps of package exchange are -

(i) Identification string exchange

(ii) Algorithm negotiation.

(iii) Key exchange

(iv) End of the exchange

(v) Service Request

→ User Authentication Protocol - Authenticates the client-side user to the server

→ Message types and Formats - Three types of message -

(i) Authentication request from the client have the format -

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

(ii) Server sends the message with the format -

SSH_MSG_USERAUTH_FAILURE (51) (byte), authentications that can continue (name-list) and partial success (boolean)

(iii) If server accepts authentication, it sends a single byte message -

SSH_MSG_USERAUTH_SUCCESS (52)

→ Authentication methods - It can be done by using public key, password or host based

→ Connection Protocol - Multiplexes the encrypted tunnel into several logical channels

→ Channel Mechanism - Three stages -

(i) Open a new channel - allocates a local no. of the channel and send message as -

SSH_MSG_CHANNEL_OPEN (byte), Channel type (string), Sender channel (uint32), initial window size (uint32), maximum packet size (uint32) & channel type specific data flows

(ii) Data transfer - performed using SSH_MSG_CHANNEL_DATA message.

(iii) Close a channel - it sends a SSH_MSG_CHANNEL_CLOSE message.

→ Channel types - Four types -

(i) Session - Remote execution of a program

(ii) X11 - X windows system which provides GUI and application

(iii) Forwarded-tcpip - Remote port forwarding (iv) direct tcpip - local port forwarding.

(11) Algorithms and Security -

(1) 40 bit key algorithms are of no use

(2) 56 bit key algorithms offer privacy, but are vulnerable

(3) 64 bit key algorithms are safe today but will be soon threatened as the technology evolves.

(4) 128 bit key and over algorithms are almost unbreakable

(5) 256 bit key and above are impossible.

(12) Disk Encryption -

It works similarly to text message encryption. With the use of an encryption program for your disk, you can safeguard any information to burn onto the disk, and keep it from falling into the wrong hands

Encryption of disks is useful ~~to send~~ when you need to send sensitive information through the mail.

(13) Government Access to Keys (GAK) -

It is also known as Key escrow. It means that software companies will give copies of all keys, (or at least enough of the key that the remainder could be cracked) to the government.

The government promises that they will hold on to the key in secure way, and will only use them when a court issues a warrant to do so.

To the government, this issue is similar to the ability to wiretap phones.

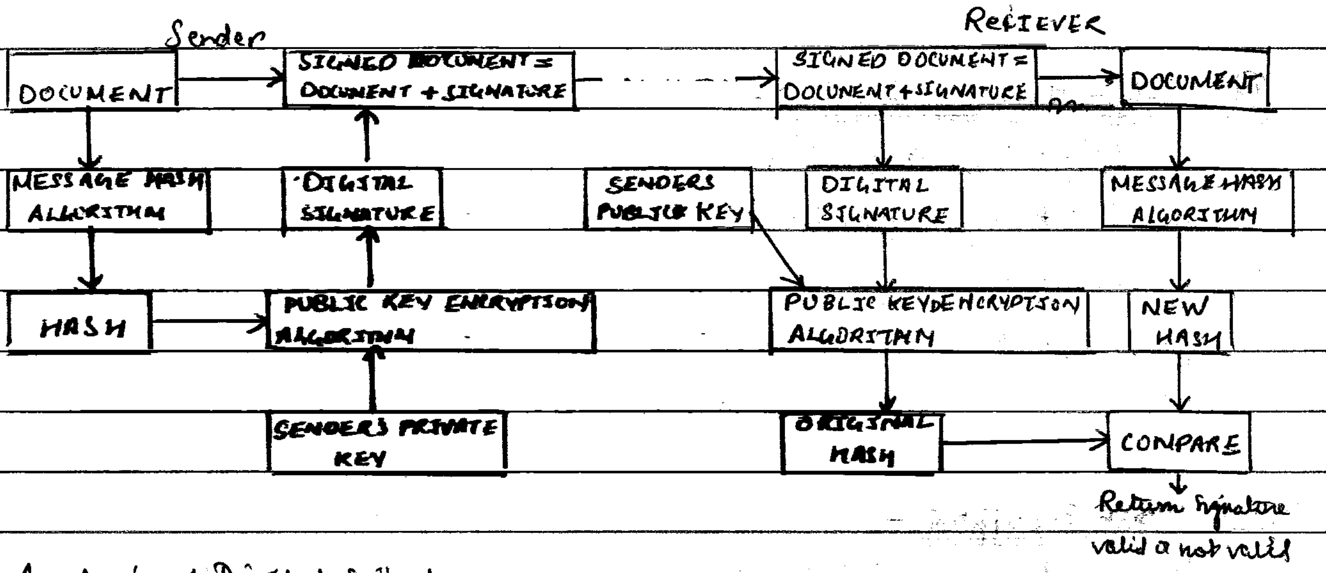
DIGITAL SIGNATURE -

(1) Digital signature is a type of asymmetric cryptography used to stimulate the security properties of a signature in digital, rather than written form.

Digital signature schemes normally give two algorithms. One for signing which involves the user's secret or private key, and one for verifying signatures which involves the user's public key.

The output of the signature process is called the 'digital signature'.

→ Simplified Depiction of essential elements of Digital signature process is given as,



② **Analysis of Digital Signature –**

(1) Ensure validity of the message

(2) Prove that it was sent by the party believed to have sent it

(3) Prove that only that party has access to the private key.

③ **Components of a Digital Signature –**

(1) Public key

(2) Name and E-mail of sender

(3) Key expiry date

(4) Company name that sends the information

(5) Serial number of Digital Signature

(6) Digital Signature of certification authority (CA).

④ **Method of Digital Signature Technology –**

Two stages of Digital Signature i.e. Creation and Verification

Two keys used in Cryptography i.e. Public key which is available to everyone and private key which is known only to the sender.

Digital Signature uses public key cryptography to encrypt & decrypt messages. ~~Has~~ Hash Encrypted message is used. Hash function produces and checks the digital signatures.

⑤

(5) Digital Signature Applications -

They are used to check -

(1) Identity of the sender

(2) Dependability of the message

(3) Whether message sent is genuine

(4) For risk of frauds

(5) Whether message is illegally produced

(6) Fulfillment of lawful requirements

(7) For security of open systems

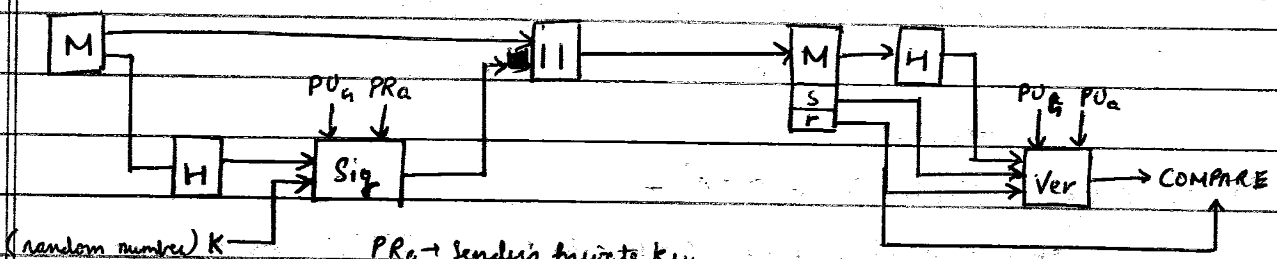
(6) Digital Signature Standard (DSS) -

DSS makes use of the Secure Hash Algorithm (SHA) and presents a new digital signature technique, called ~~Digigital~~ Digital signature algorithm (DSA).

→ DSS approach -

It uses an algorithm that is designed to provide only the digital signature function. It cannot be used for encryption or key exchange. It is a public key technique.

s & r → components of the signature.



PR_a → Sender's private key

PU_G → Global public key PU_a → Sender's public key.

(7) Algorithm : Signature Generation / Verification -

→ Digital signature algorithm -

→ Global Public key components -

p prime number where $2^{L-1} < p < 2^{L}$ for $512 \le L \le 1024$ and L is a multiple of 64; i.e, bit length of between 512 and 1024 bits in increments of 64 bits

q prime divisor of (p-1), where $2^{159} < q < 2^{160}$, i.e. bit length of 160 bits

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

→ User's Private Key — x random or pseudorandom integer with $0<x<q$

→ User's Public key — y $= g^x \bmod p$

→ User's Per-Message Secret Number —

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

→ Signing (Generation) —

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

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

Signature $=(r,s)$

→ Verifying —

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

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

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

$v = [(g^{u1} y^{u2}) \bmod p] \bmod q$

Test: $v = r'$

M', r', s' → received versions of M, r, s.

$H(M)$ → hash of M using SHA-1.

⑧ ECDSA (Elliptic Curve DSA) —

It is a variant of the DSA which uses Elliptic curve cryptography. Size of an ECDSA public key would be 160 bits. On the other hand, the signature size is the same for both DSA and ECDSA.

→ Signature generation algorithm —

(1) Calculate $e = HASH(m)$, where HASH is a cryptographic hash function and let z be the L_n leftmost bits of e

(2) Select a random integer k from $[1, n-1]$.

(3) Calculate $r = x_1 \pmod n$, where $(x_1, y_1) = kG$. If $r=0$, go back to step 2

(4) Calculate $s = k^{-1}(z + r d_A) \pmod n$. If $s=0$, go back to step 2, d_A → sender private key

(5) The signature is the pair (r,s)

When computing s, the string z resulting from $HASH(m)$ shall be converted to an integer. Note that z can be greater than n but not longer.

Date __/__/__
Page ____

Signature verification algorithm -

(1) Check that $Q_A \neq O$ (identity element) and Q_A lies on the curve & $nQ_A = O$

(2) Verify that r and s are integers in $[1, n-1]$. If not, the signature is invalid

(3) Calculate $e = HASM(m)$, where HASH is the same function used in the signature generation. Let z be the L_n leftmost bits of e.

(4) Calculate $w = s^{-1} \pmod n$.

(5) Calculate $u_1 = zw \pmod n$ and $u_2 = rw \pmod n$

(6) Calculate $(x_1, y_1) = u_1 G + u_2 Q_A$

(7) The signature is valid if $r = x_1 \pmod n$, invalid otherwise

(9) Elgamal Signature Scheme -

It involves the use of the private key for encryption and the public key for decryption.

The global elements of ElGamal digital signature are a prime number q and α, which is a primitive root of q.

User A generates a private/public key pair as follows -

(1) Generate a random integer X_A, such that $1 < X_A < q-1$

(2) Compute $Y_A = \alpha^{X_A} \bmod q$.

(3) A's private key is X_A; A's public key is $\{q, a, Y_A\}$

To sign a message M, user A first computes the hash $m = H(M)$, such that m is an integer in the range $0 \le m \le q-1$. A then forms a digital signature as follows-

(1) Choose a random integer K such that $1 \le K \le q-1$ and $\gcd(K, q-1) = 1$.

(2) Compute $S_1 = \alpha^K \bmod q$.

(3) Compute $K^{-1} \bmod (q-1)$.

(4) Compute $S_2 = K^{-1}(m - X_A S_1) \bmod (q-1)$

(5) The signature consists of the pair (S_1, S_2).

Any user B can verify the signature as follows -

(1) Compute $V_1 = \alpha^m \bmod q$.

(2) Compute $V_2 = (Y_A)^{S_1}(\cancel{Y}S_1)^{S_2} \bmod q$.

The signature is valid if $V_1 = V_2$.

(10) **Digital Certificates** -

It verify the uniqueness of the principles and entities over networks as electronic documents. Unique identity to the owner of the digital certificate is defined by both public key and private keys.

Widely accepted format for digital certificates is defined by the ITU-T X.509 international standard. It is issued by a Certification Authority (CA).

→ Digital Certificates include a ~~widely~~ variety of information such as -

(1) Name of the subject
(2) Subjects public key
(3) Certification authority's name
(4) Serial number
(5) Lifetime period of the digital certificate right from the start date.

→ Four main types of digital certificates are -

(1) Server Certificates
(2) Personal Certificates
(3) Organization Certificates
(4) Developer Certificates.

→ Digital Certificates are used for -

(1) Proving the identity of the sender of a transaction
(2) Non Repudiation
(3) Encryption and checking the integrity of data
(4) Single Sign-On

→ It is used in SSL ~~and~~, Secure Multipurpose Internet Mail Extension (S/MI Secure Electronic Transactions (SET) and Internet Protocol Secure Standard (IPSec)

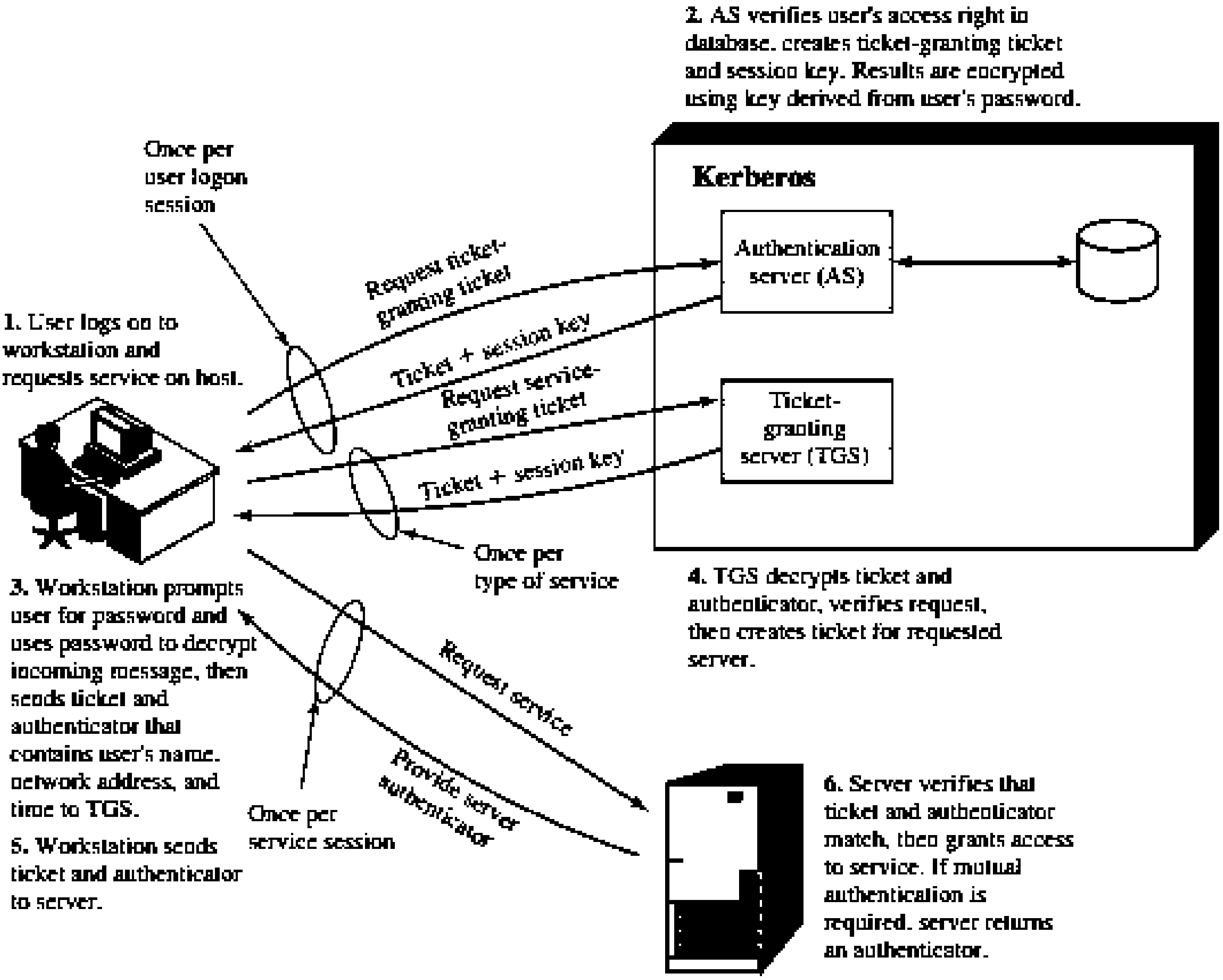

2. AS verifies user's access right in database, creates ticket-granting ticket and session key. Results are encrypted using key derived from user's password.
Kerberos
Authentication server (AS)
Ticket-granting server (TGS)
Once per user logon session
Request ticket-granting ticket
Ticket + session key
Request service-granting ticket
Ticket + session key
1. User logs on to workstation and requests service on host.
Once per type of service
4. TGS decrypts ticket and authenticator, verifies request, then creates ticket for requested server.
3. Workstation prompts user for password and uses password to decrypt incoming message, then sends ticket and authenticator that contains user's name, network address, and time to TGS.
Request service
Provide server authenticator
Once per service session
5. Workstation sends ticket and authenticator to server.
6. Server verifies that ticket and authenticator match, then grants access to service. If mutual authentication is required, server returns an authenticator.

Figure 15.1 Overview of Kerberos