UNIT IV

SECURITY PRACTICE & SYSTEM SECURITY

UNIT - IV

(2)

Security Practice & System Security.

Authentication Applications :-

* key concern of security are confidentiality and timeliness.

* To provide confidentiality one must encrypt identification field & session key.

* Developed to support application - level authentication and digital services.

* kerberos - a private key authentication sewice.

* X.509 - a public key directory authentication sewice.

kerberos :-

* kerberos is a centralized authentication sewer, whose fun is to authenticate the users to sewer and sewer to users.

Problem that kerberos address in this :-

Workstation cannot be trusted to identify its users correctly to nlw,

threats, > masquarade
> Eavesdropping
> Replay.

Requirements of kerberos :-

- > Secure
- > Reliable
- > Transparent
- > Scalable.

kerberos version 4 :-

* This make use of DES.

* Simple Authentication Dialogue :-

* To overcome unauthorized users to access, it means to use an authentication sever (AS) that stores password of all users and shows a unique secret key with each sever.

(1) $C \rightarrow AS : IP_C \,||\, P_C \,||\, ID_V$

(2) $AS \rightarrow C$: Ticket

(3) $C \rightarrow V : ID_C \,||$ Ticket

$$\text{Ticket} : E(k_V, [ID_C \,||\, AD_C \,||\, AD_V])$$

Steps:

* Users logs on to a workstation & request access to sever.

* Ticket is encypted, so it is not altered by C (or) opponent.

* The ticket is decypted by V and verify the user ID.

Adv:

* As ticket is encrypted, it prevents alteration by C. (4)
* Inclusion of ADc in the ticket, avoids attack by an opponent.

Disadv:

* Each ticket can be used only once
* Pwd is used as clear plain text Pc, opponent can misuse it.

* More Secure Authentication Dialogue:-
  * this provided by use of ticket granting Sever (TGS) instead of authentication Sever (AS).

Scenario:-

once per user logon section:

(1) C → As : IDc || IDtgs
1. tgs authenticate client & provide ticket
(2) AS → C : E(Kc, Ticket tgs)
2. ticket encrypted with session key, client will decrypt it

once per type of severs:

(3) C → TGS : IDc || IDv || Ticket tgs
(4) TGS → C : Ticket v.

Once per Sevice Session:

(5) C → V : I_{pc} || Ticket V.

Encrypted ticket shared by sever & TGS.

Ticket contains following information:

$Ticket_v = E(K_v, [ID_c || AD_c || ID_v || TS_2 || Lifetime\ 2])$.

$Ticket_{tgs} = E(K_{tgs}, [ID_c || AD_c || ID_{tgs} || TS_1 || Lifetime\ 1])$.

Adv:-

* Ticket reusability.
* Protection of user password.
* Timestamps indication of issuing tickets date and time.
* Encryption tgs and tv prevents forgery.

* V4 Authentication Dialogue:-

* Combination of simples & more secured authentication.

V4 Message Exchanges:-

a) Authentication sever exchange to obtain ticket-granting ticket.

(1) C → AS : ID C || ID_{tgs} || TS_1

(2) AS → C : E(Kc, [Kc, tgs || IDtgs || TS2 || Lifetime 2 || Ticket tgs]) (6)

3) Ticket - Granting sewice exchange to obtain sewice - granting ticket.

(3) C → TGS : IDv || Ticket tgs || Authenticator C

4) TGS → C : E(kc, tgs, [kc,v || IDv || TS4 || Ticketv])

Ticket tgs = E (ktgs, [k C, tgs [k C, tgs || IDc || ADc || IDtgs || TS2 || Lifetime 2)

Ticket v = E (kv, [kc,v || IDc || AD c || IDv || TS4 || Lifetime 4])

Authenticator c = E (Kc, tgs, [IDc || AD c || TS3])

5) Client / sewer authenticate exchange to obtain sewer:

(5) C → V : Ticket v || Authenticator c.

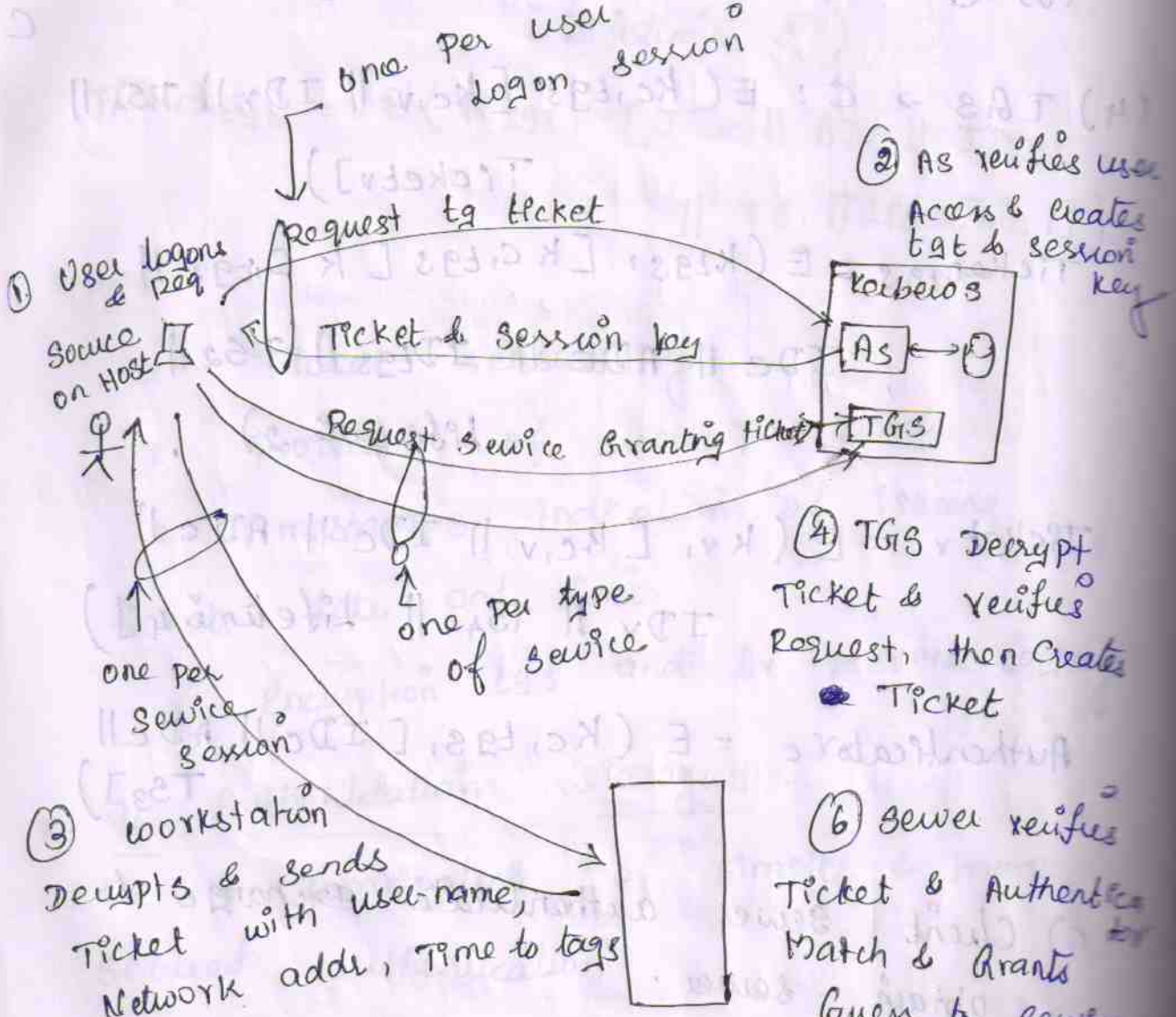
(6) V → C : E(Kc,v, [TS5 + 1)

(for mutual authentication)

$$Ticket_v = E(K_v, [K_{c,v} \| ID_c \| AD_c \| ID_v \| TS_4 \| Lifetime_4])$$

$$Authenticator_c = E(K_{c,v}, [ID_c \| AD_c \| TS_5])$$

Overview of kerberos:-



(5) Workstation sends Ticket and authenticator to server.

Kerberos Version 5:-

* Encryption System dependence
* Internet protocol "
* Msg byte ordering
* Ticket lifetime
* Authetication Forwarding in V5 not in V4.
* Inter-realm authentication.

V5 Message Exchanges:-

a) Authentication souxe exchange to obtain tags:

(1) C → AS: Options || IDc || Realm c ||
IDtgs || Times || Nonce1

(2) AS → C: Realmc || IDc || Tickettgs ||
E(Kc, [Kc, [Kc, tgs || Times || Nonce1 ||
Realm tgs || IDtgs]).

Ticket tgs = E(Ktgs, [Flags || Kc, tgs ||
Realm c || IDc || ADc || Times])

b) Ticket-granting source Exchange to obtain source-granting ticket:-

(3) C → TGS: Options || IDv || Times ||
Nonce 2 || Ticket tgs || Authenticator C.

(4) TGS → C : $Realm_C$ || ID_C || $Ticket_v$ ||

E($K_{C,tgs}$, [$K_{c,v}$ || Times || $Nonce_2$ ||

$Realm_v$ || ID_v])

$Ticket_{tgs}$ = E(K_{tgs}, [flags || $K_{c,tgs}$ || $Realm_c$

|| ID_C || AD_C || times])

$Ticket_v$ = E(K_v, [flags || $K_{c,v}$ || $Realm_C$ ||

ID_C || AD_C || Times])

Authenticator C = E($K_{c,tgs}$ (ID_C ||

$Realm_C$ || TS_1]).

c) Client | Server Authenticate exchange to obtain service :-

(5) C → V : Options || $Ticket_v$ || authenticator

(6) V → C : $E_{K_{C,V}}$ [TS_2 || subkey || seq #]

$Ticket_v$ = E(K_v, [flags || $K_{c,v}$ || $Realm_C$ ||

ID_C || AD_C || Times])

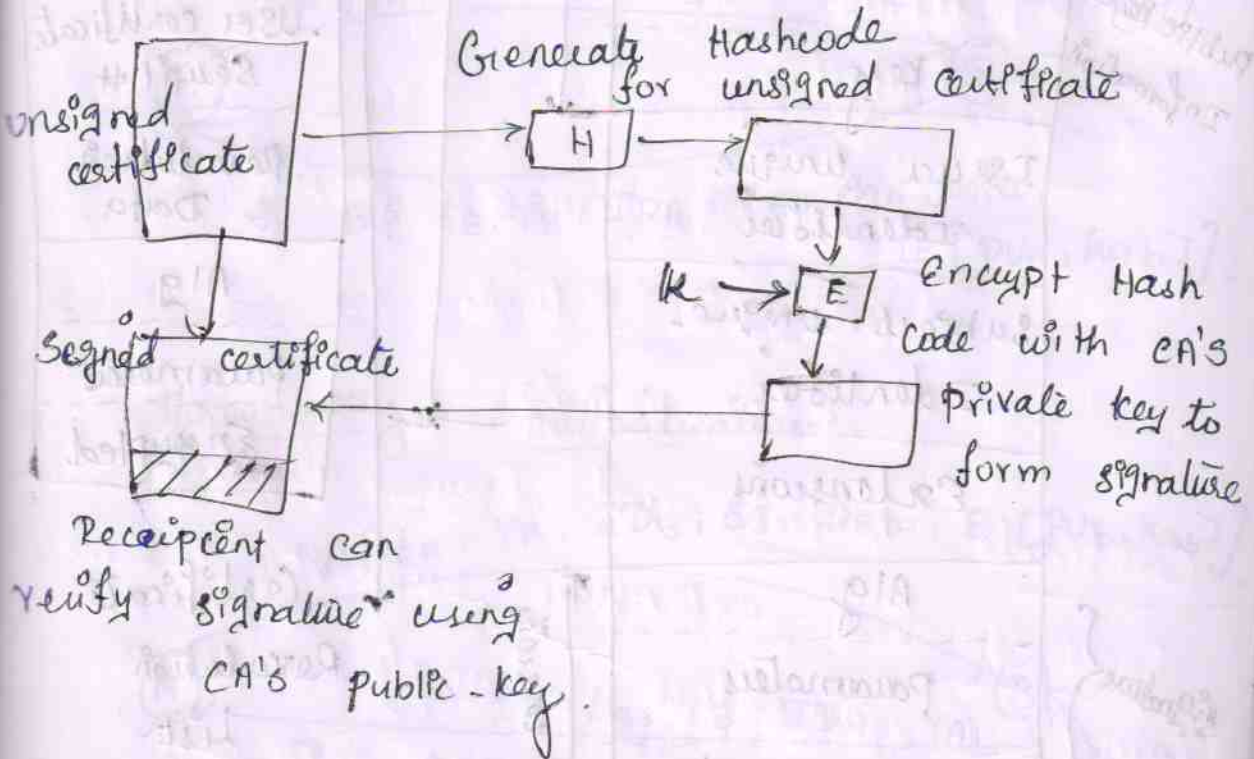
Authenticator C = E($K_{c,v}$ [ID_C || $Realm_C$ ||

TS_2 || subkey || seq #])

X.509 Authentication Service :- (10)

* X.509 defines format for Public-key certificates. This format is widely used in variety of applications. that are,
  * S/MIME
  * SSL/TLS
  * IP Security
  * SET.

Public-key certificate :-



Certificate

$$CA\langle\langle A\rangle\rangle = CA\{V, SN, AI, CA, TA, A, Ap\}$$