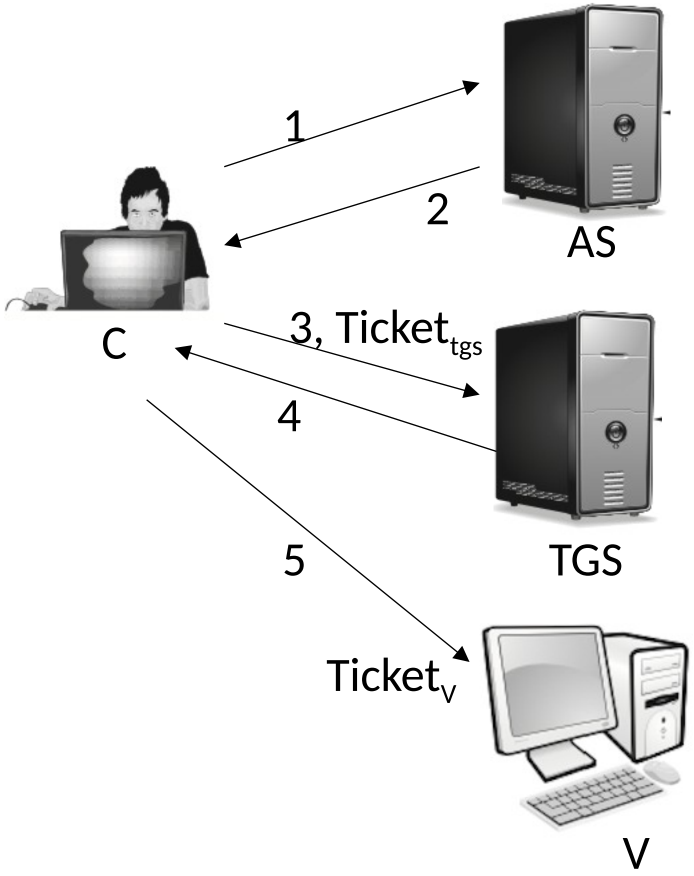
Lecture 25

A More Secure Authentication Dialogue

- Once per user logon session
  - (1) C —>AS: $ID_C || ID_{tgs}$
  - (2) AS —> C: $E(K_C, Ticket_{tgs})$
- Once per type of service:
  - (3) C —>TGS: $ID_C || ID_v ||$ $Ticket_{tgs}$
  - (4) TGS —> C: $Ticket_V$
- Once per service session:
  - (5) C —> V: ID_C || $Ticket_V$

$$Ticket_{tgs} = \mathrm{E}(K_{tgs}, [ID_C \| AD_C \| ID_{tgs} \| TS_1 \| Lifetime_1])$$

$$Ticket_v = \mathrm{E}(K_v, [ID_C \| AD_C \| ID_v \| TS_2 \| Lifetime_2])$$



1. C —>AS: $ID_C || P_C || ID_V$
2. AS —> C : Ticket = $E(K_V, [ID_C || AD_C || ID_V])$
3. C —> V: ID_C || Ticket

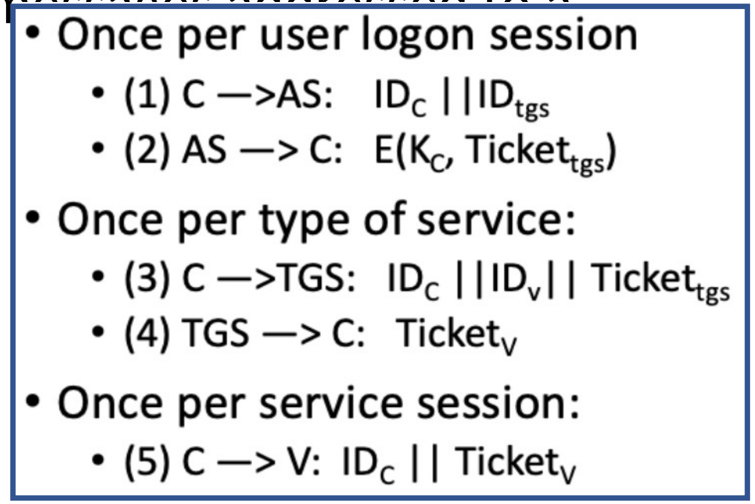
Advantage

- No password transmitted in plaintext
- Ticket is reusable. Timestamp is added to prevent reuse of ticket by an attacker

Secure?

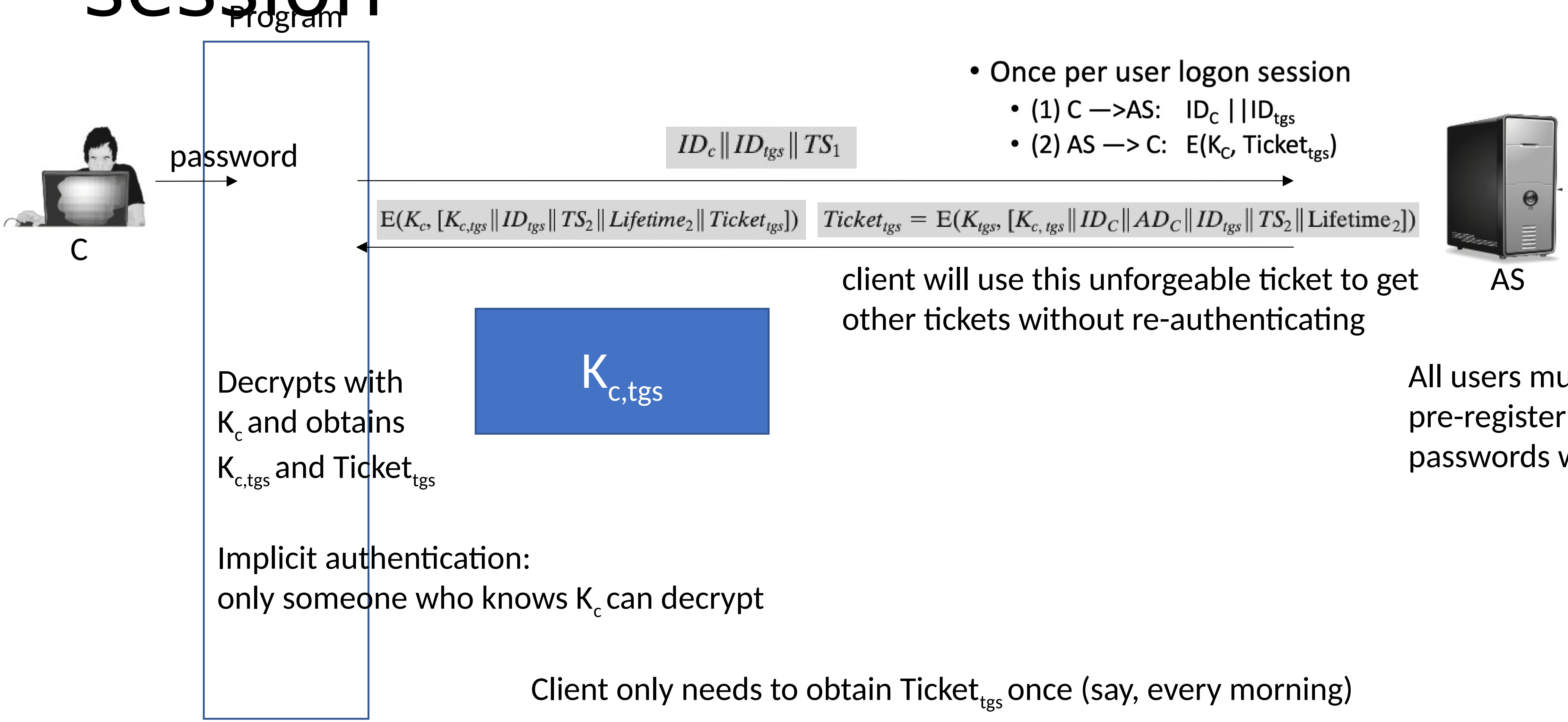
no user authentication

- Ticket hijacking
  - Malicious user may steal the service ticket of another user on the same workstation and try to use it
    - Network address verification does not help
  - Servers must verify that the user who is presenting the ticket is the same user to whom the ticket was issued
- No server authentication
  - Attacker may misconfigure the network so that he receives messages addressed to a legitimate server – man in the middle attack
    - Capture private information from users and/or deny service
  - Servers must prove their identity to users
- Solution: section key



- Once per user logon session
  - (1) C —>AS: ID_C ||ID_{tgs}
  - (2) AS —> C: E(K_C, $Ticket_{tgs}$)
- Once per type of service:
  - (3) C —>TGS: ID_C ||ID_v|| $Ticket_{tgs}$
  - (4) TGS —> C: $Ticket_V$
- Once per service session:
  - (5) C —> V: ID_C || $Ticket_V$

Kerberos v4. - once per user logon session



- Once per user logon session
  - (1) C —>AS: ID_C ||ID_{tgs}
  - (2) AS —> C: $E(K_C, Ticket_{tgs})$

$ID_c \| ID_{tgs} \| TS_1$

$E(K_c, [K_{c,tgs} \| ID_{tgs} \| TS_2 \| Lifetime_2 \| Ticket_{tgs}])$

$Ticket_{tgs} = E(K_{tgs}, [K_{c,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| \mathrm{Lifetime}_2])$

client will use this unforgeable ticket to get other tickets without re-authenticating

$K_{c,tgs}$

Decrypts with K_c and obtains $K_{c,tgs}$ and $Ticket_{tgs}$

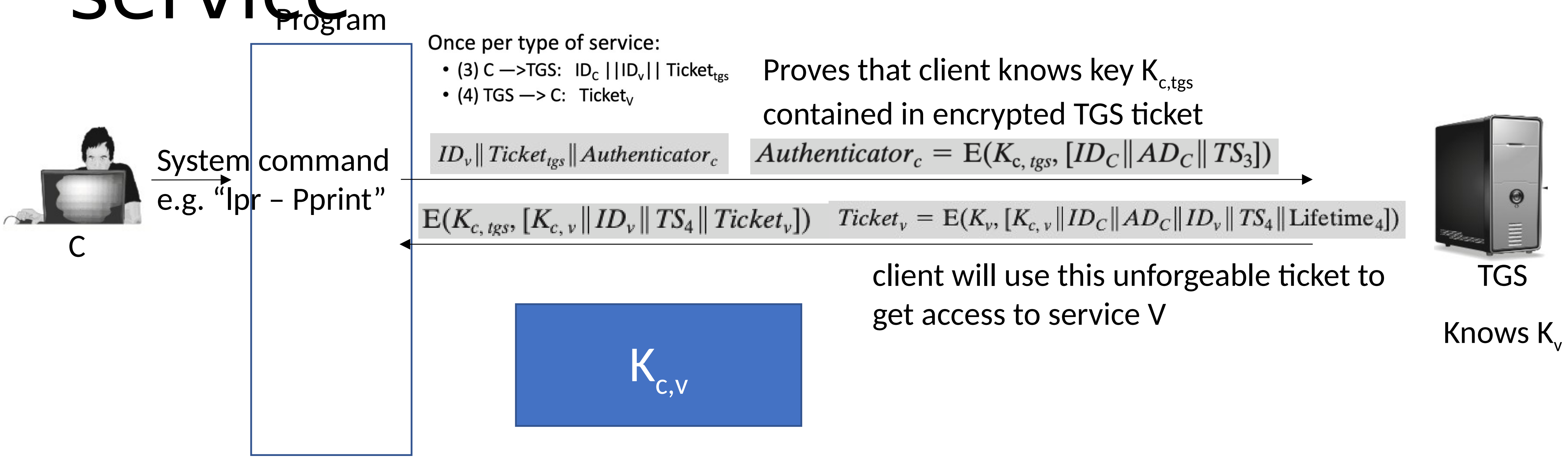
All users must pre-register their passwords with AS

Implicit authentication:
only someone who knows K_c can decrypt

Client only needs to obtain $Ticket_{tgs}$ once (say, every morning)

Ticket is encrypted; client cannot forge it or tamper with it

Kerberos v4. - once per type of service



Program

Once per type of service:

- (3) C —>TGS: $ID_C || ID_v || Ticket_{tgs}$
- (4) TGS —> C: $Ticket_V$

System command e.g. “lpr – Pprint”

C

$ID_v \| Ticket_{tgs} \| Authenticator_c$

Proves that client knows key $K_{c,tgs}$ contained in encrypted TGS ticket

$Authenticator_c = \mathrm{E}(K_{c,\,tgs}, [ID_C \| AD_C \| TS_3])$

$\mathrm{E}(K_{c,\,tgs}, [K_{c,\,v} \| ID_v \| TS_4 \| Ticket_v])$

$Ticket_v = \mathrm{E}(K_v, [K_{c,\,v} \| ID_C \| AD_C \| ID_v \| TS_4 \| \mathrm{Lifetime}_4])$

client will use this unforgeable ticket to get access to service V

TGS

Knows K_v

$K_{c,v}$

Client uses $Ticket_{tgs}$ to obtain a service ticket, $Ticket_v$ and a short-term session key for each network service (printer, email, etc.)

$Ticket_{tgs} = \mathrm{E}(K_{tgs}, [K_{c,\,tgs} \| ID_C \| AD_C \| ID_{tgs} \| TS_2 \| \mathrm{Lifetime}_2])$

Kerberos v4. - once per service session

Program

Once per service session:

- (5) C —> V: ID_C || $Ticket_V$

C

System command e.g. “lpr – Pprint”

$Ticket_v \| Authenticator_c$

Proves that client knows key $K_{c,v}$ contained in encrypted ticket

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

$E(K_{c,v}, [TS_5 + 1])$(for mutual authentication)

V

Authenticates server to client

Chain of Reasoning:

Server can produce this message only if he knows $K_{c,v}$

Server can learn key $K_{c,v}$ only if he can decrypt service ticket

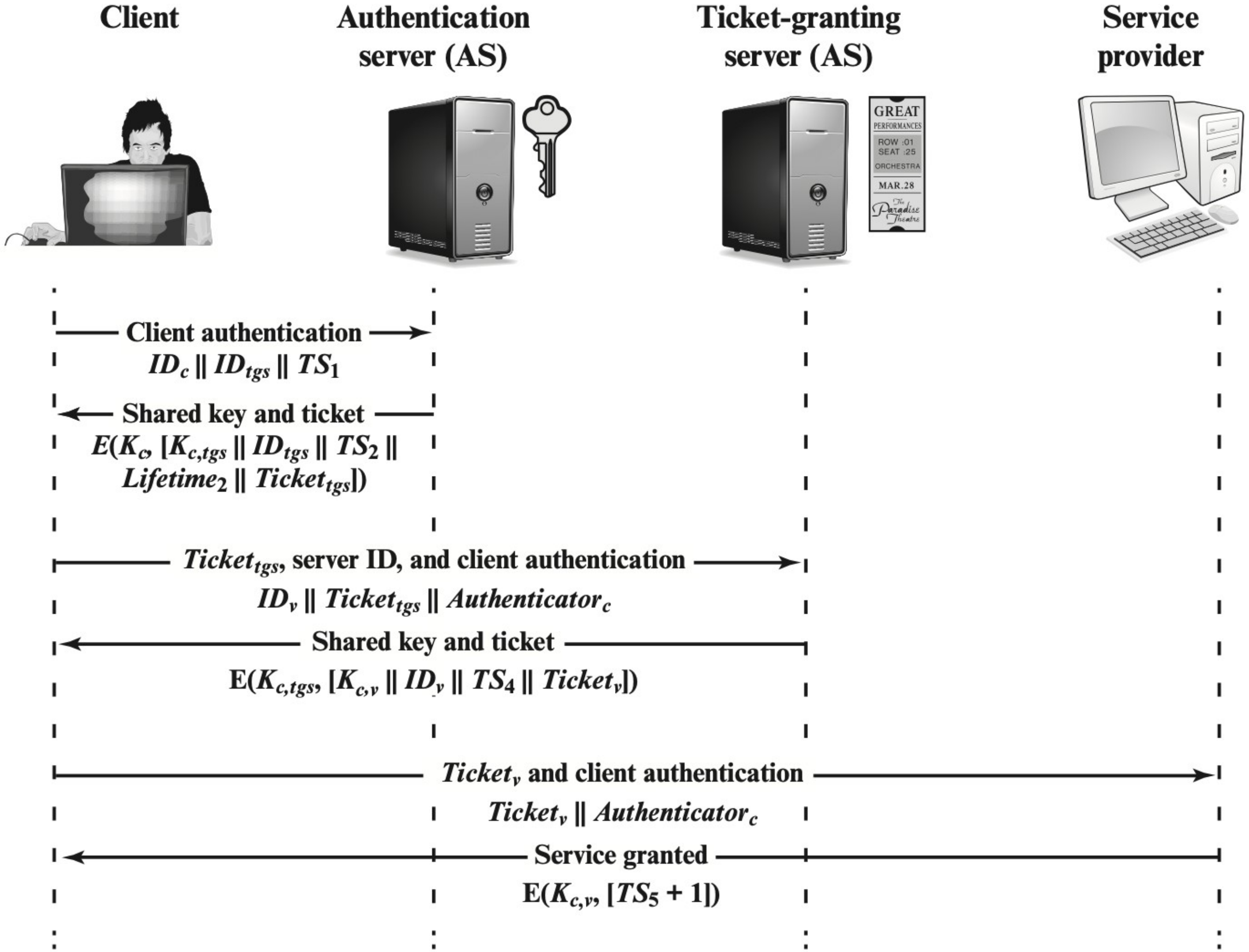
Server can decrypt service ticket only if he knows correct key K_V

If server knows correct key K_V, then he is the right server

For each service request, client uses the short-term key, $K_{c,v}$, for that service and the ticket he received from TGS

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

Overview of Kerberos



Client

Authentication server (AS)

Ticket-granting server (AS)

Service provider

Client authentication

$ID_c \parallel ID_{tgs} \parallel TS_1$

Shared key and ticket

$E(K_c, [K_{c,tgs} \parallel ID_{tgs} \parallel TS_2 \parallel Lifetime_2 \parallel Ticket_{tgs}])$

$Ticket_{tgs}$, server ID, and client authentication

$ID_v \parallel Ticket_{tgs} \parallel Authenticator_c$

Shared key and ticket

$\mathrm{E}(K_{c,tgs}, [K_{c,v} \parallel ID_v \parallel TS_4 \parallel Ticket_v])$

$Ticket_v$ and client authentication

$Ticket_v \parallel Authenticator_c$

Service granted

$\mathrm{E}(K_{c,v}, [TS_5 + 1])$