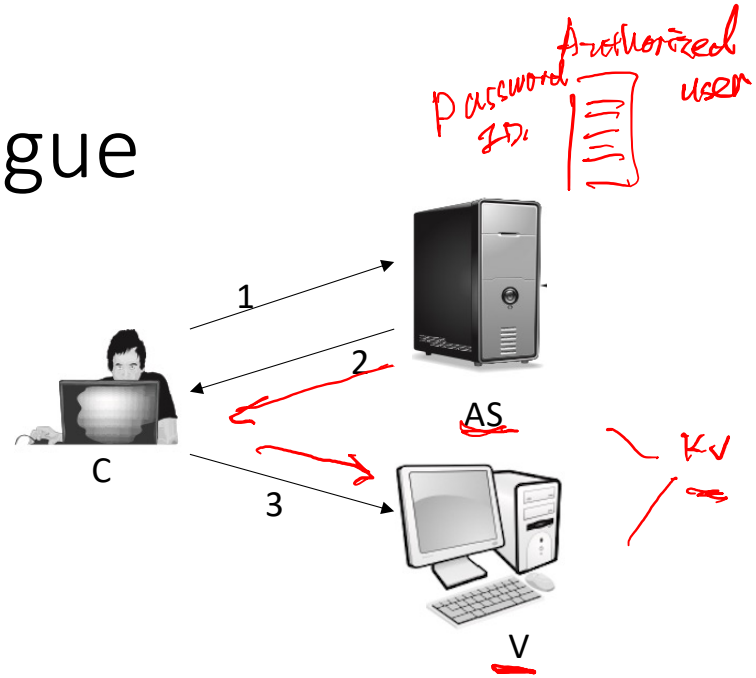
Kerberos

- a centralized authentication server provides mutual authentication between users and servers
  - a key distribution and user authentication service developed at MIT
  - works in an open distributed environment
- client-service model
- Kerberos protocol messages are protected against eavesdropping and replay attacks
- Kerberos v4 and v5 [RFC 4120]

A Simple Authentication Dialogue

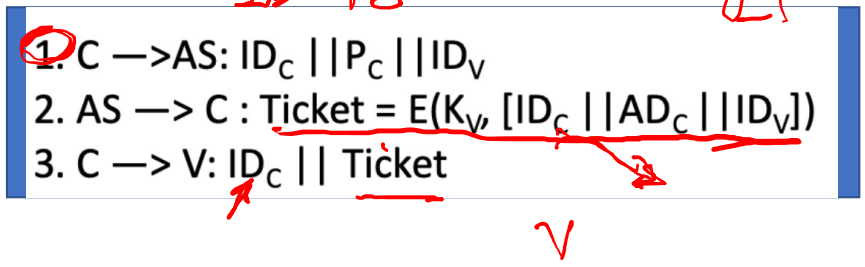
- 1. C —>AS: ID_C ||P_C ||ID_V
- 2. AS —> C : Ticket = $E(K_V, [ID_C ||AD_C ||ID_V])$
- 3. C —> V: ID_C || Ticket



- AS – authentication server
- ID_* - identifier
- P_C - password of user
- AD_C - network address of C
- K_V - secret encryption key shared by AS and V

Advantage

- Client and malicious attacker cannot alter ID_C (impersonate), AD_C(change of address) , ID_V
- server V can verify the user is authenticated through ID_C , and grants service to C
- guarantee the ticket is valid only if it is transmitted from the same client that initially requested the ticket



Secure?

- Insecure: password is transmitted openly and frequently
- Solution: no password transmitted by involving ticket-granting server (TGS)

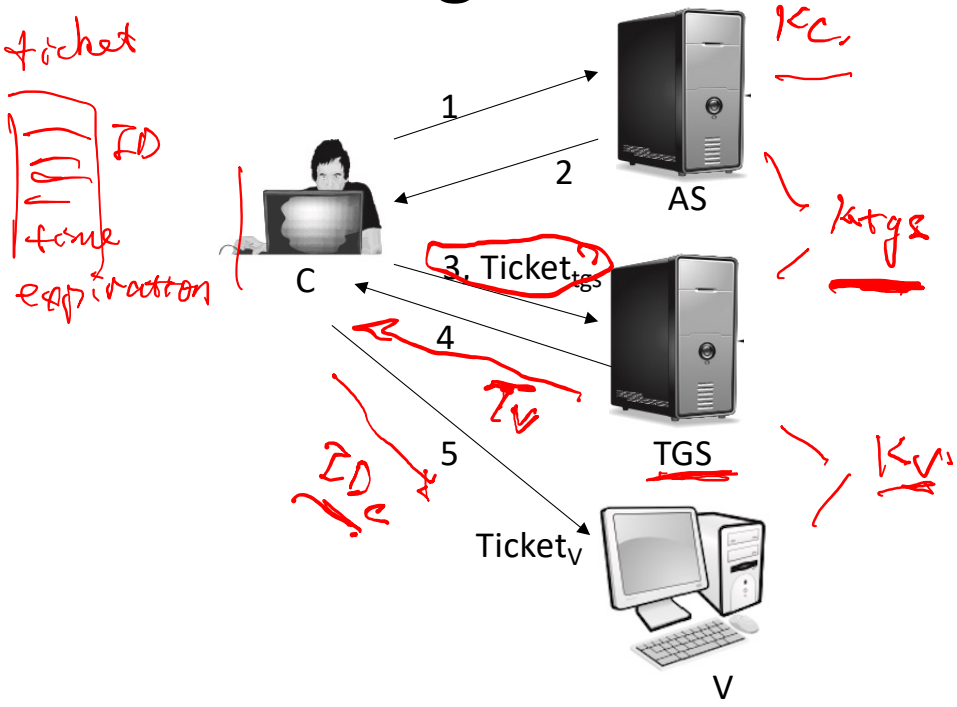
1. C —>AS: $ID_C || P_C || ID_V$
2. AS —> C : Ticket = $E(K_V, [ID_C || AD_C || ID_V])$
3. C —> V: ID_C || Ticket

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: $ID_C || Ticket_V$

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

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



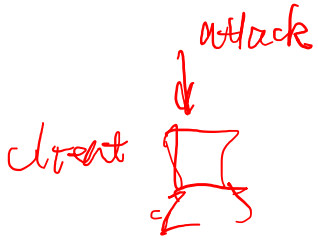
1. C —>AS: $ID_C || P_C || ID_V$
2. AS —> C : Ticket = $E(K_V, [ID_C || AD_C || ID_V])$
3. C —> V: $ID_C || Ticket$

Advantage

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

Secure?

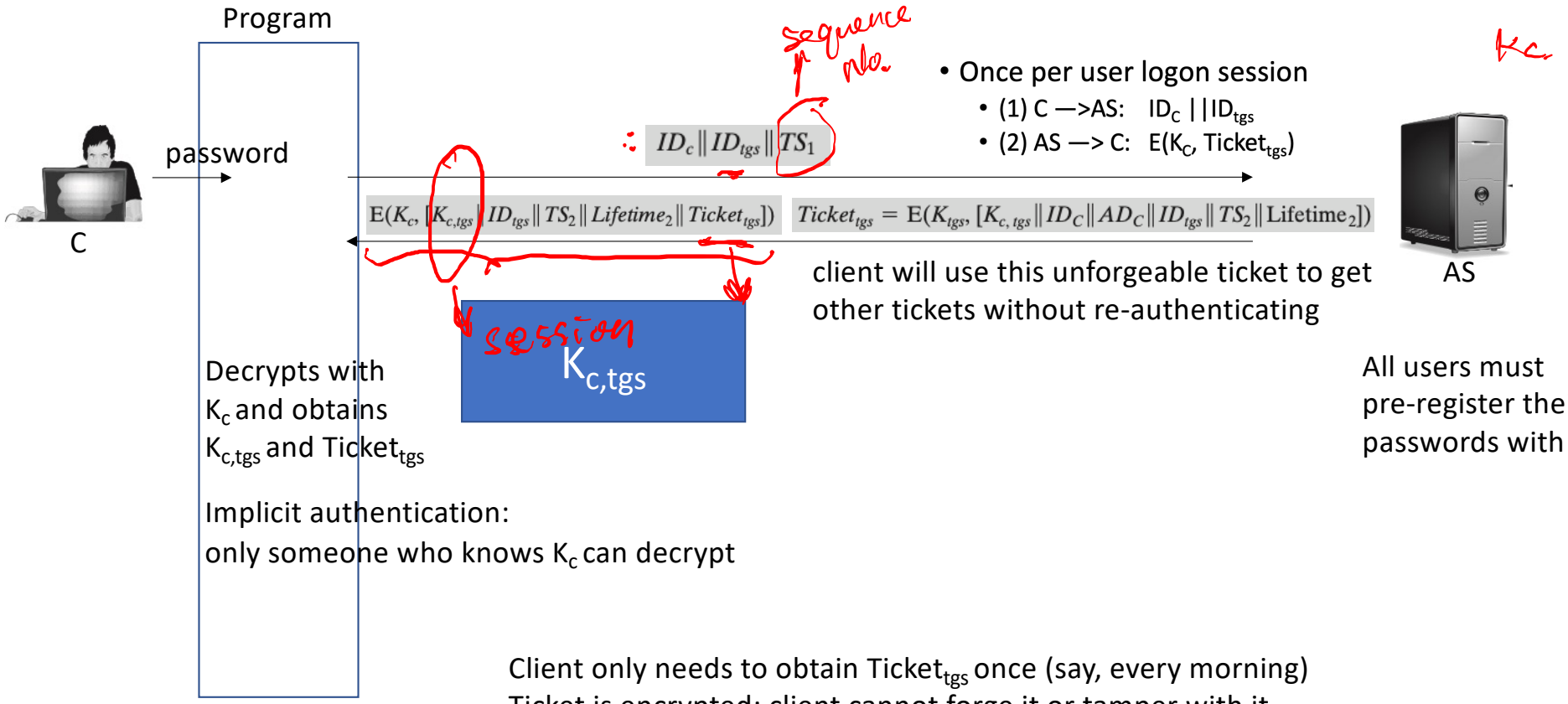
no user authentication



- 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: session key

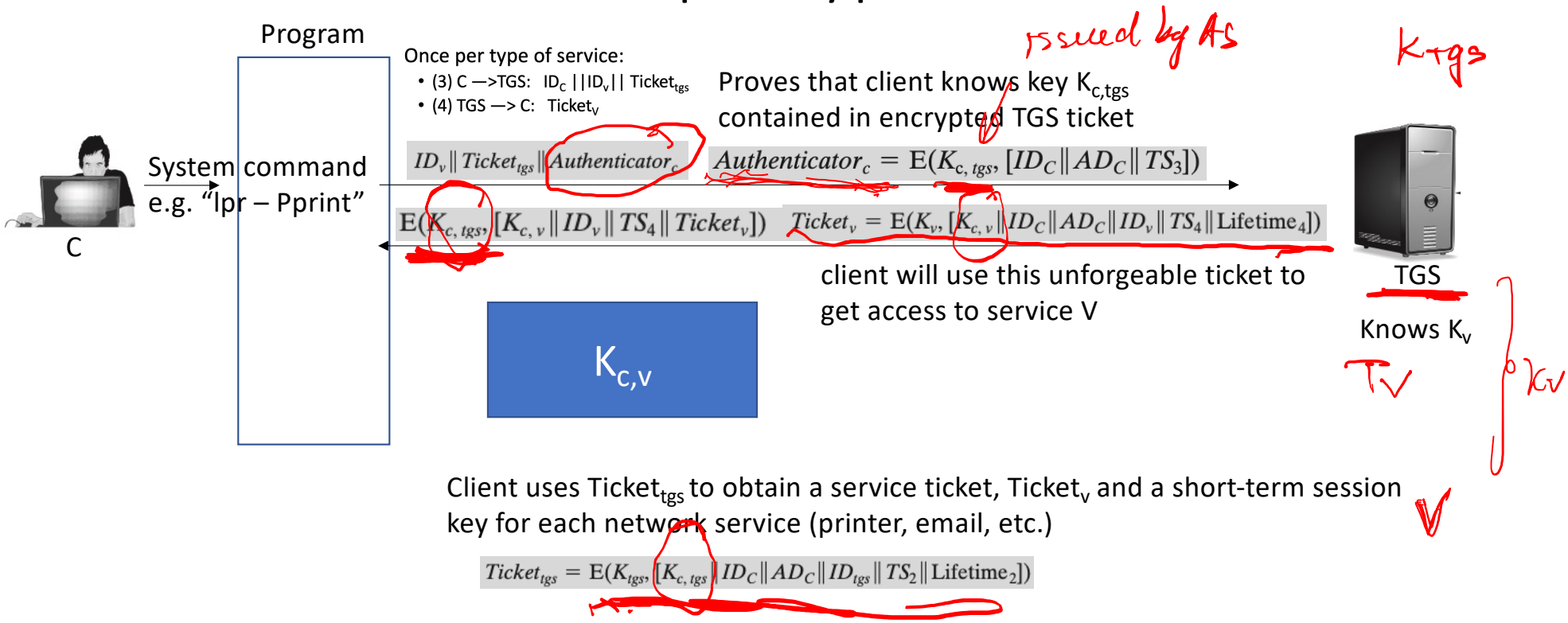
- Once per user logon session
  - (1) C —>AS: $ID_C || ID_{tgs}$
  - (2) AS —> C: $E(K_C, Ticket_{tgs})$
- Once per type of service:
  - (3) C —>TGS: $ID_C || ID_V || Ticket_{tgs}$
  - (4) TGS —> C: $Ticket_V$
- Once per service session:
  - (5) C —> V: $ID_C || Ticket_V$

Kerberos v4. - once per user logon session



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



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

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