Kerberos

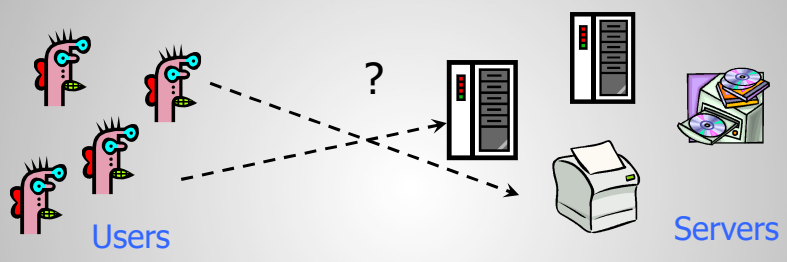
https://web.mit.edu/kerberos/

https://tools.ietf.org/html/rfc4120

slide 1

•1

Many-to-Many Authentication



How do users prove their identities when requesting services from machines on the network?

Naïve solution: every server knows every user's password

- Insecure: break-in of one server ⇒ compromise all users
- Inefficient: to change password, user must contact every server

slide 2

•2

Requirements

- Security
  - Must be secure against attacks by passive eavesdroppers and active attackers, including rogue insiders
- Reliability
  - Must be always available
- Transparency
  - Users should not notice authentication taking place
  - Entering password is OK, if done rarely enough
- Scalability
  - Must handle large numbers of users and servers

slide 3

•3

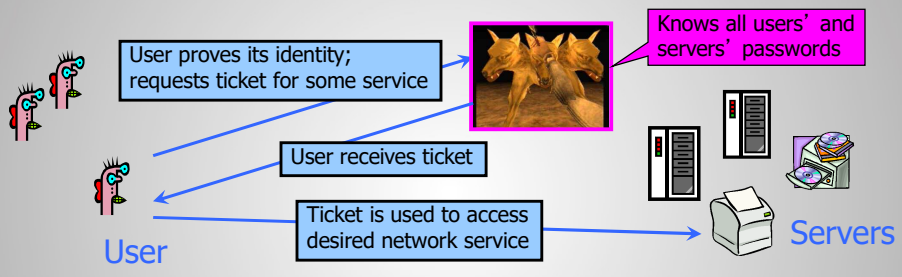
Threats

- User impersonation
  - Malicious user with access to a workstation pretends to be another user from the same workstation
    - Can't trust workstations to verify users' identities
- Network address impersonation
  - Malicious user changes network address of his workstation to impersonate another workstation
    - Can't trust network (IP and/or MAC) addresses
- Eavesdropping, tampering and replay
  - Malicious user eavesdrops on, tampers with, or replays, other users' conversations (messages) to gain unauthorized access to servers

slide 4

•4

Solution: Trusted Third Party

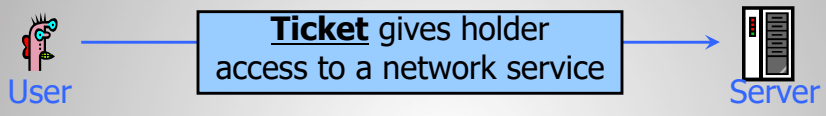


- Trusted authentication service
  - Knows all passwords, can grant access to any server
  - Convenient, but also single point of failure
  - Requires high level of physical security

slide 5

•5

What Should a Ticket Look Like?

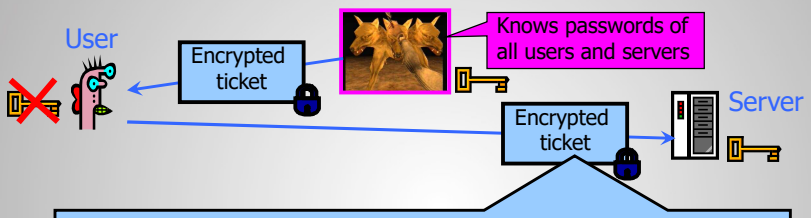


- Ticket cannot include server plaintext password
  - Otherwise, next time user will access server directly without proving its identity to authentication service
- Solution: encrypt some information with a key known to the server, but not to the user!
  - Server can decrypt ticket and verify information
  - User does not learn server key

slide 6

•6

What Should a Ticket Include?

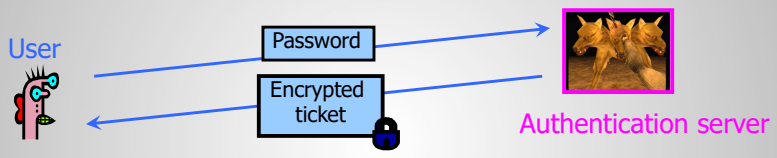


- User name
- Server name
- Address of user's workstation
  - Otherwise, a user on another workstation can steal the ticket and use it to gain access to the server
- Ticket lifetime
- A few other things (e.g., session key)

slide 7

•7

How to authenticate initially?



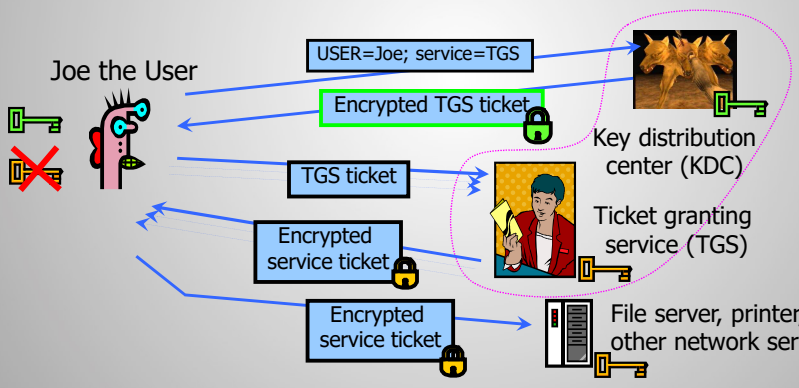
- Insecure: passwords are sent in plaintext
  - Eavesdropper can steal password and impersonate user
- Inconvenient: need to send the password each time to obtain the ticket for any network service
  - Separate authentication for email, printing, etc.

slide 8

•8

Two-Step Authentication

- Prove identity **once** to obtain special TGS ticket
- Use TGS to get tickets for any network service



slide 9

•9

Still Not Good Enough

- Ticket hijacking
  - Malicious user may steal the service ticket of another user on the same workstation and use it
    - IP 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 mis-configure the network so that it (attacker) receives messages addressed to a legitimate server
    - Capture private information from users and/or deny service
  - Servers must prove their identity to users

slide 10

•10

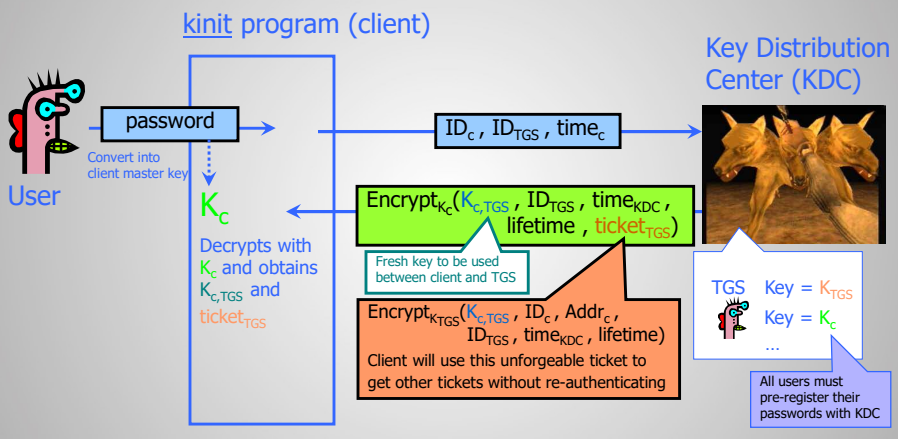
Symmetric Keys in Kerberos

- K_c -- long-term key of client C
  - Derived from users password
  - Known to client and key distribution center (KDC)
- K_{TGS} -- long-term key of TGS
  - Known to KDC and ticket granting service (TGS)
- K_v -- long-term key of network service V
  - Known to V and TGS; separate key for each service
- $K_{C,TGS}$ -- is short-term key between C and TGS
  - Created by KDC, known to C and TGS
- $K_{C,V}$ -- short-term key between C and V
  - Created by TGS, known to C and V'

slide 11

•11

"Single Logon" Authentication

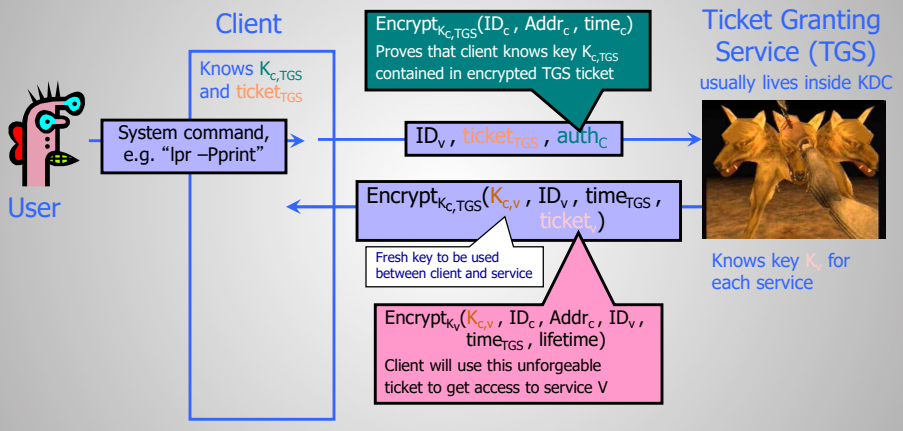


- Client only needs to obtain TGS ticket **once** (say, every morning)
  - Ticket is encrypted; client cannot forge it or tamper with it

slide 12

•12

Obtaining a Service Ticket

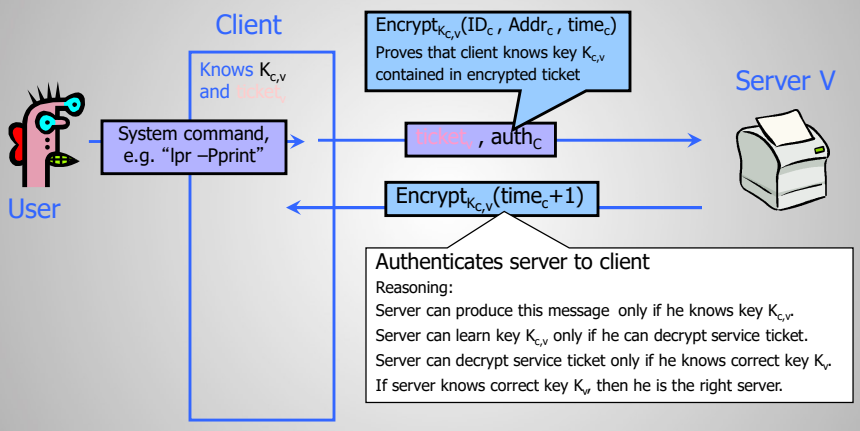


- Client uses TGS ticket to obtain a service ticket and a short-term key for each network service
  - One encrypted, unforgeable ticket per service (printer, email, etc.)

slide 13

•13

Obtaining Service



- For each service request, client uses the short-term key for that service and the ticket he received from TGS

slide 14

•14

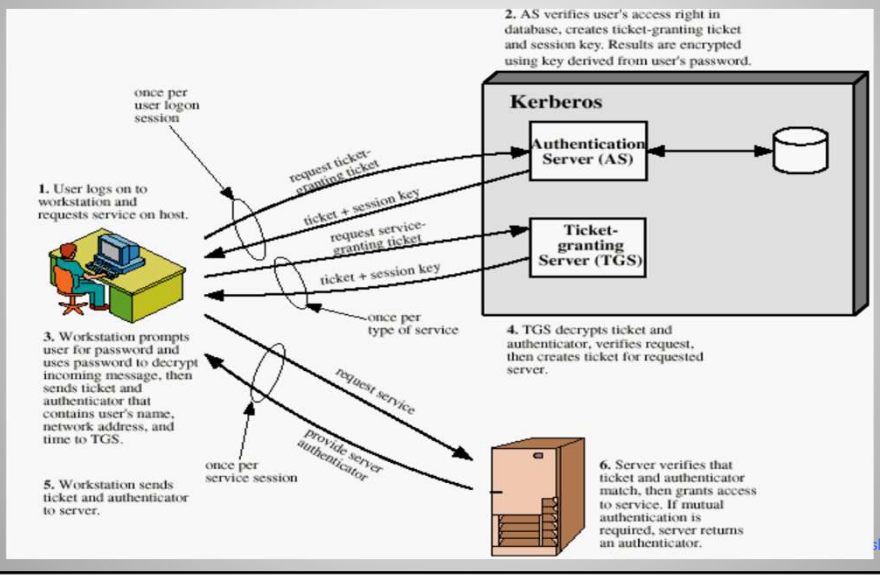
Kerberos in Large Networks

- One KDC isn't enough for large networks (why?)
- Network is divided into realms
  - KDCs in different realms have different key databases
- To access a service in another realm, users must…
  - Get ticket for home-realm TGS from home-realm KDC
  - Get ticket for remote-realm TGS from home-realm TGS
    - As if remote-realm TGS were just another network service
  - Get ticket for remote service from that realm's TGS
  - Use remote-realm ticket to access service
  - N(N-1)/2 key exchanges for full N-realm interoperation

slide 15

•15

Summary of Kerberos



slide 16

•16

Important Ideas in Kerberos

- Short-term session keys
  - Long-term secrets used only to derive short-term keys
  - Separate session key for each user-server pair
    - … but multiple user-server sessions re-use the same key
- Proofs of identity are based on authenticators
  - Client encrypts his identity, address and current time using a short-term session key
    - Also prevents replays (if clocks are globally synchronized)
  - Server learns this key separately (via encrypted ticket that client can't decrypt) and verifies user's identity
- Symmetric cryptography only (with some exceptions)

slide 17

•17

Problematic Issues

- Dictionary attacks on client master keys
- Replay of authenticators
  - 5-minute lifetimes long enough for replay
  - Timestamps assume global, secure synchronized clocks
  - Challenge-response would have been better
- Same user-server key used for all sessions
- "Homebrewed" PCBC mode of encryption
- Superfluous double encryption of tickets
- No ticket delegation
  - Printer can't fetch email from server on your behalf

slide 18

•18

Kerberos Version 5

- Better user-server authentication
  - Separate subkey for each user-server session instead of re-using the session key contained in the ticket
  - Authentication via subkeys, not timestamp increments
- Authentication forwarding
  - Servers can access other servers on user's behalf
- Realm hierarchies for inter-realm authentication
- Richer ticket functionality
- Explicit integrity checking + standard CBC mode
- Multiple encryption schemes, not just DES

slide 19

•19

Practical Uses of Kerberos

- Email, FTP, network file systems and many other applications have been kerberized
  - Use of Kerberos is transparent for the end user
  - Transparency is important for usability!
- MS Windows Networking, Active Directory, SAP…
- Local authentication
  - login and su in OpenBSD
- Authentication for network protocols
  - rlogin, rsh, telnet
- Secure windowing systems
  - xdm, kx

slide 20

•20