Internet Authentication

Types of Environments:
- PC or workstation
- Cloud
- A bunch of servers

A bunch of servers: Separation of concern, redundant connections

Problem: Enforcing least privilege in the system. Do I have to authenticate myself every time I query the database server? We want a way for authentication.

Kerberos:
- Most common internet authentication system
- Do authentication without sending a password every time

Authentication Server (AS):
- Has a list of usernames and passwords
- Ticket granting ticket (encrypted with password)
- Password not sent across wire
- Password used to encrypt/decrypt ticket

Ticket Granting Server (TGS):
- Other servers need shared key with AS
- Service granting ticket requested (encrypted with shared key)

Kerberos Realm
- Interrealm: AS must have trust (shared secrets)

X.509: Certificate:
- Use 2 key system (public/private) to prove I am who I say I am
- Certifying authority certifies public key (someone I trust)

Certifying Authority Issues:
- Do I trust them?
- Is someone masquerading? (Bigger problem)

Use CA’s public key steps:
- I want a certificate
- Ask CA for certificate
- Provide CA with my public key (and $$)
- CA can sign certificate with their private key and my public key
- I can distribute certificate

CA needs to change key: Certificate revocation list. Easier to ask if key is valid. (Online certificate status protocol)