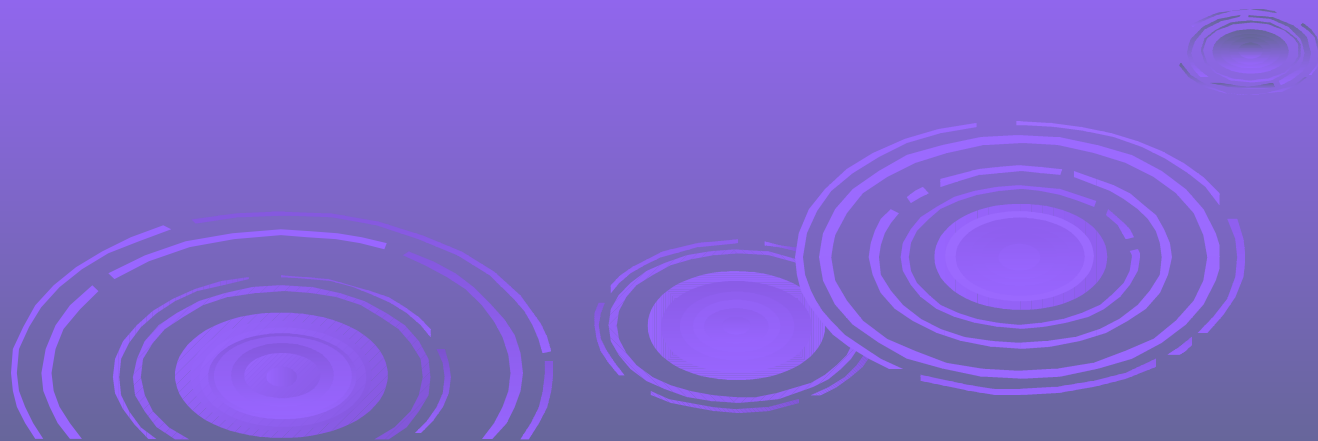


Cryptography and Network Security

Chapter 14



Authentication Applications

- will consider authentication functions
- developed to support application-level authentication & digital signatures
- will consider Kerberos – a private-key authentication service
- then X.509 - a public-key directory authentication service



Kerberos

- trusted key server system from MIT
- provides centralised private-key third-party authentication in a distributed network
 - allows users access to services distributed through network
 - without needing to trust all workstations
 - rather all trust a central authentication server
- two versions in use: 4 & 5

Kerberos Requirements

- its first report identified requirements as:
 - secure
 - reliable
 - transparent
 - scalable
- implemented using an authentication protocol based on Needham-Schroeder

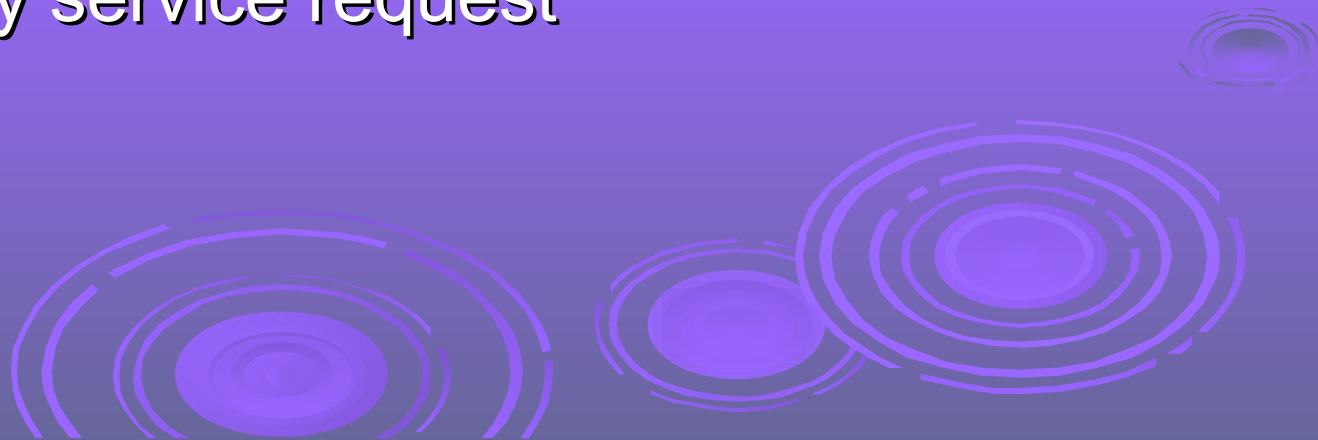


Kerberos v4 Overview

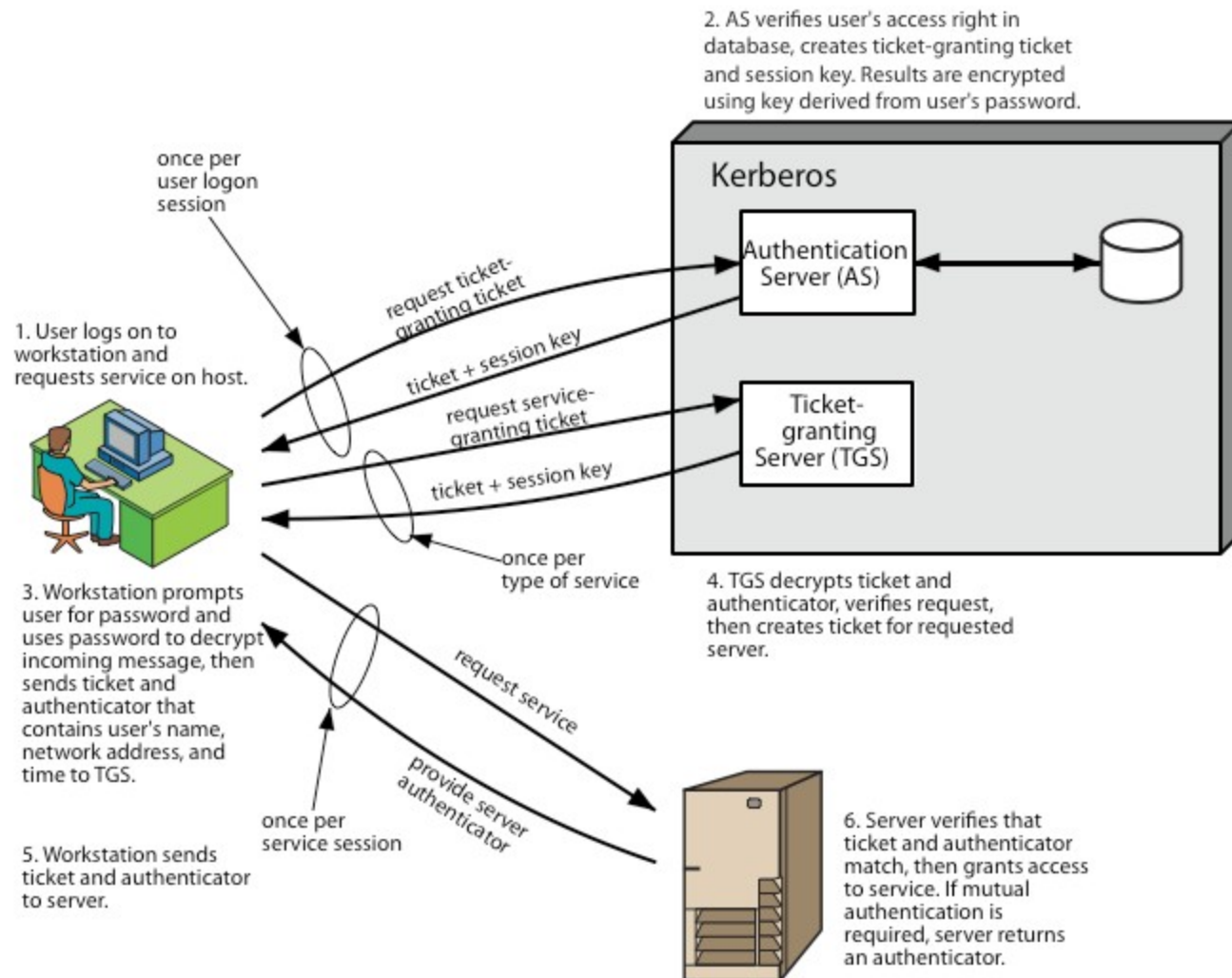
- a basic third-party authentication scheme
- have an Authentication Server (AS)
 - users initially negotiate with AS to identify self
 - AS provides a non-corruptible authentication credential (ticket granting ticket TGT)
- have a Ticket Granting server (TGS)
 - users subsequently request access to other services from TGS on basis of users TGT

Kerberos v4 Dialogue

1. obtain ticket granting ticket from AS
 - once per session
1. obtain service granting ticket from TGT
 - for each distinct service required
1. client/server exchange to obtain service
 - on every service request



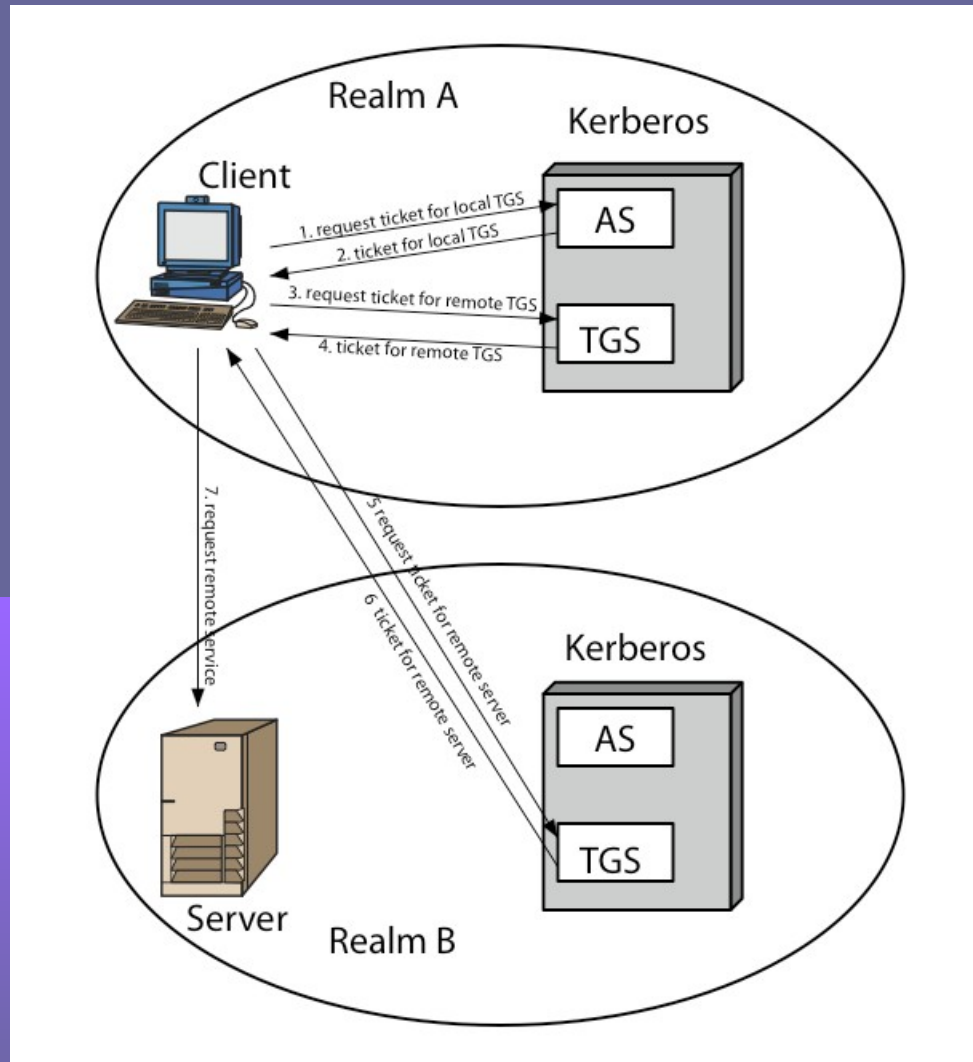
Kerberos 4 Overview



Kerberos Realms

- a Kerberos environment consists of:
 - a Kerberos server
 - a number of clients, all registered with server
 - application servers, sharing keys with server
- this is termed a realm
 - typically a single administrative domain
- if have multiple realms, their Kerberos servers must share keys and trust

Kerberos Realms



Kerberos Version 5

- developed in mid 1990's
- specified as Internet standard RFC 1510
- provides improvements over v4
 - addresses environmental shortcomings
 - encryption alg, network protocol, byte order, ticket lifetime, authentication forwarding, interrealm auth
 - and technical deficiencies
 - double encryption, non-std mode of use, session keys, password attacks

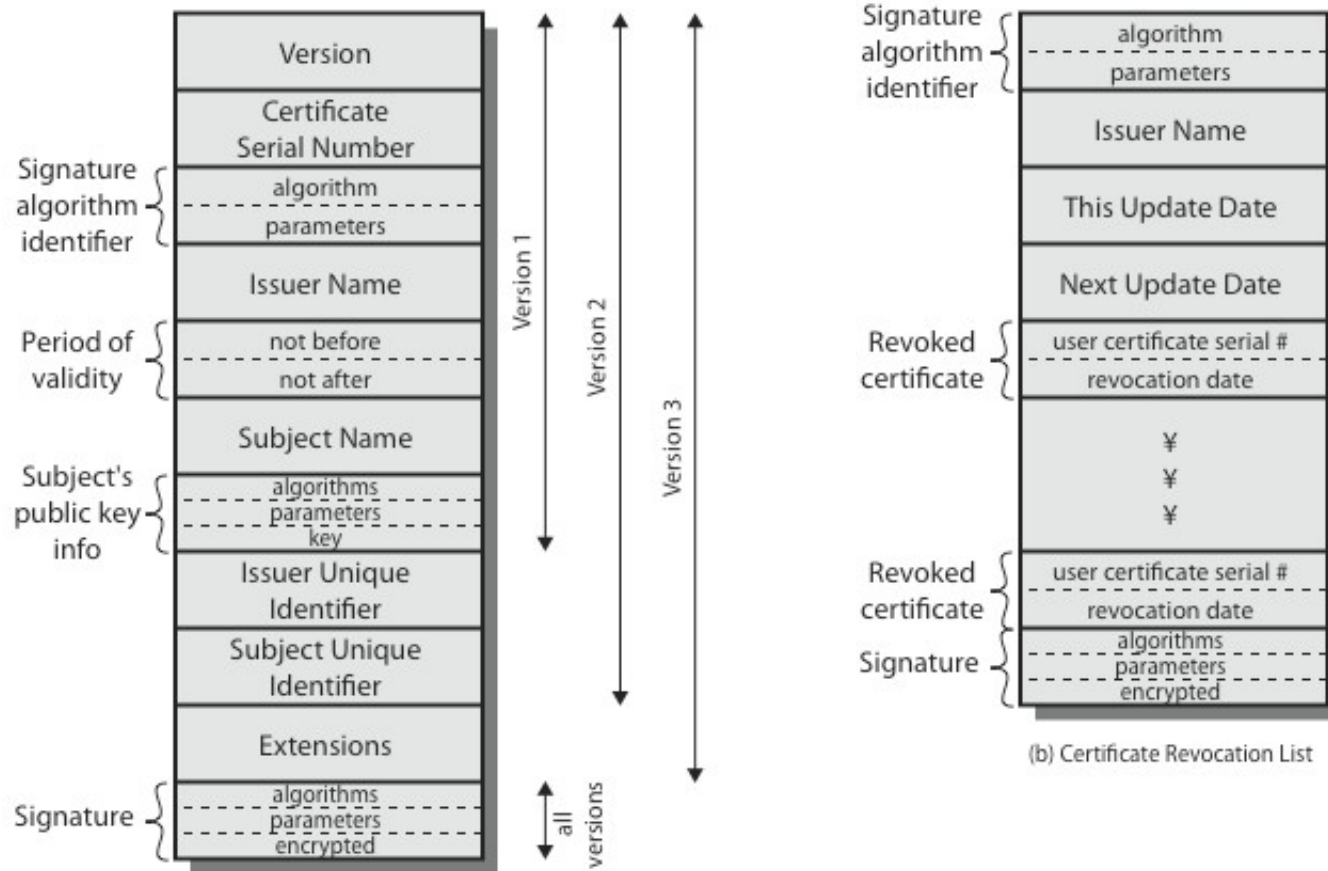
X.509 Authentication Service

- part of CCITT X.500 directory service standards
 - distributed servers maintaining user info database
- defines framework for authentication services
 - directory may store public-key certificates
 - with public key of user signed by certification authority
- also defines authentication protocols
- uses public-key crypto & digital signatures
 - algorithms not standardised, but RSA recommended
- X.509 certificates are widely used

X.509 Certificates

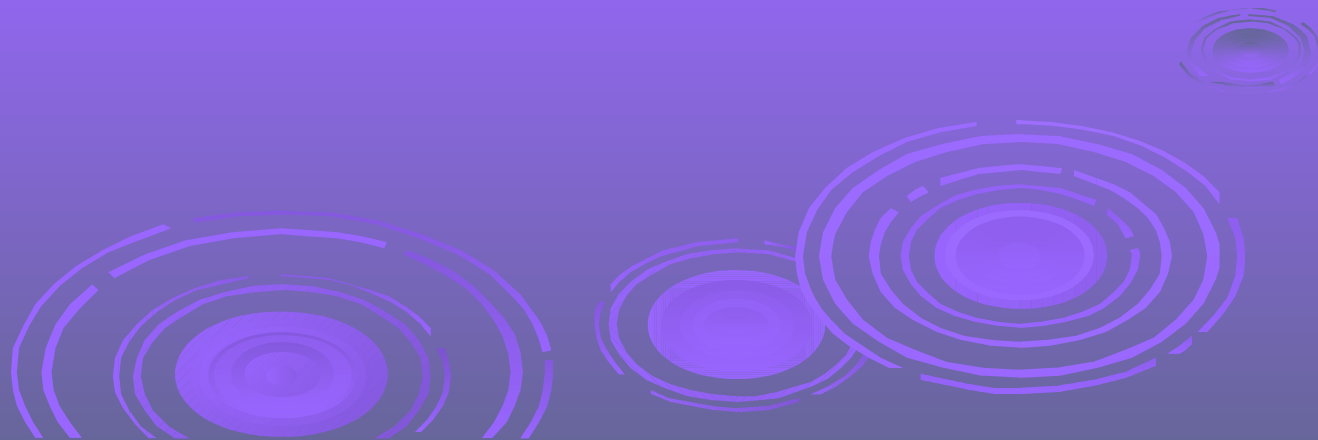
- issued by a Certification Authority (CA), containing:
 - version (1, 2, or 3)
 - serial number (unique within CA) identifying certificate
 - signature algorithm identifier
 - issuer X.500 name (CA)
 - period of validity (from - to dates)
 - subject X.500 name (name of owner)
 - subject public-key info (algorithm, parameters, key)
 - issuer unique identifier (v2+)
 - subject unique identifier (v2+)
 - extension fields (v3)
 - signature (of hash of all fields in certificate)
- notation $CA\langle\langle A \rangle\rangle$ denotes certificate for A signed by CA

X.509 Certificates



Obtaining a Certificate

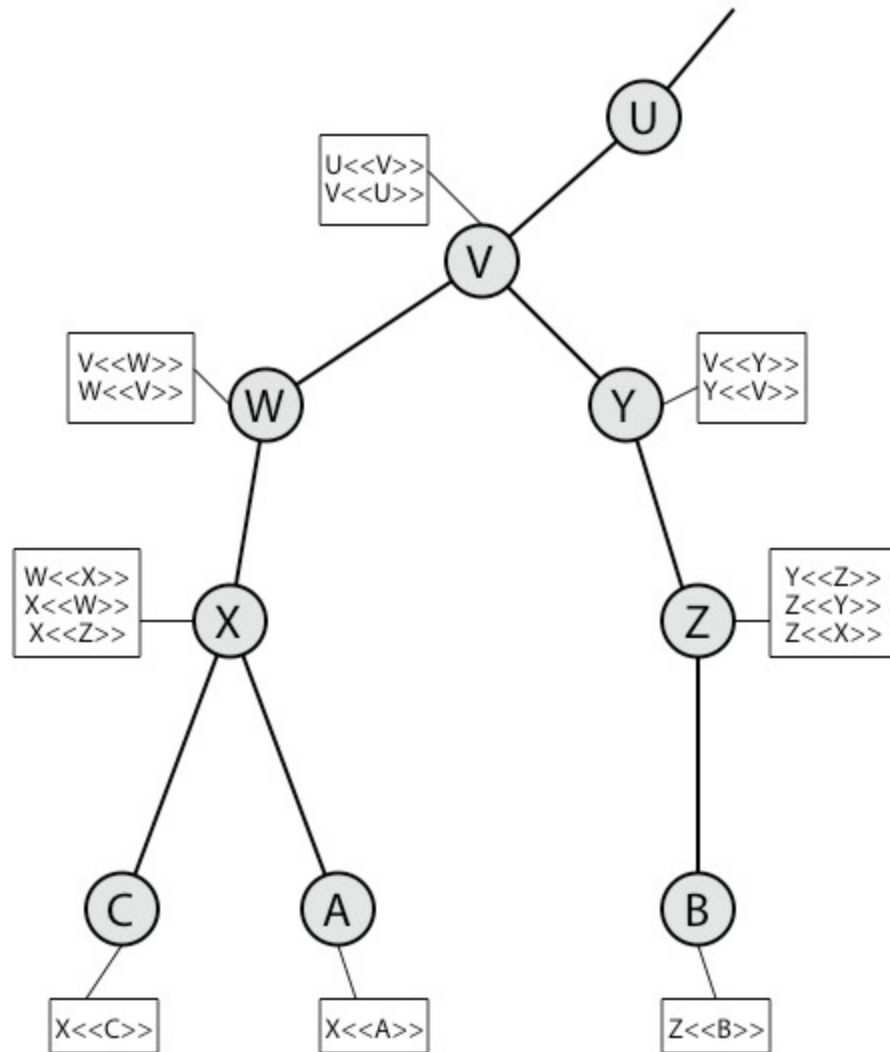
- any user with access to CA can get any certificate from it
- only the CA can modify a certificate
- because cannot be forged, certificates can be placed in a public directory



CA Hierarchy

- if both users share a common CA then they are assumed to know its public key
- otherwise CA's must form a hierarchy
- use certificates linking members of hierarchy to validate other CA's
 - each CA has certificates for clients (forward) and parent (backward)
- each client trusts parents certificates
- enable verification of any certificate from one CA by users of all other CAs in hierarchy

CA Hierarchy Use

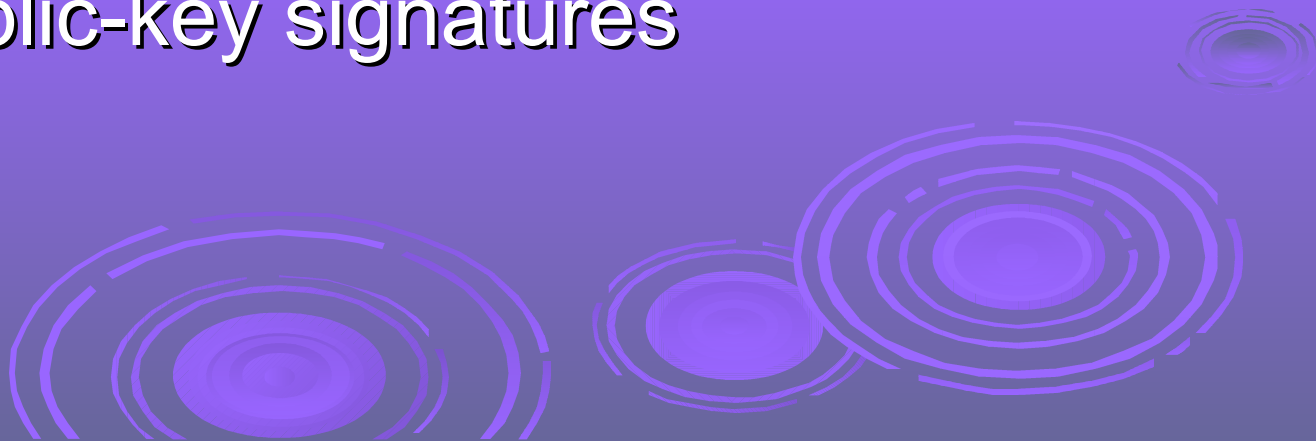


Certificate Revocation

- certificates have a period of validity
- may need to revoke before expiry, eg:
 1. user's private key is compromised
 2. user is no longer certified by this CA
 3. CA's certificate is compromised
- CA's maintain list of revoked certificates
 - the Certificate Revocation List (CRL)
- users should check certificates with CA's CRL

Authentication Procedures

- X.509 includes three alternative authentication procedures:
- One-Way Authentication
- Two-Way Authentication
- Three-Way Authentication
- all use public-key signatures



One-Way Authentication

- 1 message (A→B) used to establish
 - the identity of A and that message is from A
 - message was intended for B
 - integrity & originality of message
- message must include timestamp, nonce, B's identity and is signed by A
- may include additional info for B
 - eg session key

Two-Way Authentication

- 2 messages (A→B, B→A) which also establishes in addition:
 - the identity of B and that reply is from B
 - that reply is intended for A
 - integrity & originality of reply
- reply includes original nonce from A, also timestamp and nonce from B
- may include additional info for A

Three-Way Authentication

- 3 messages (A→B, B→A, A→B) which enables above authentication without synchronized clocks
- has reply from A back to B containing signed copy of nonce from B
- means that timestamps need not be checked or relied upon



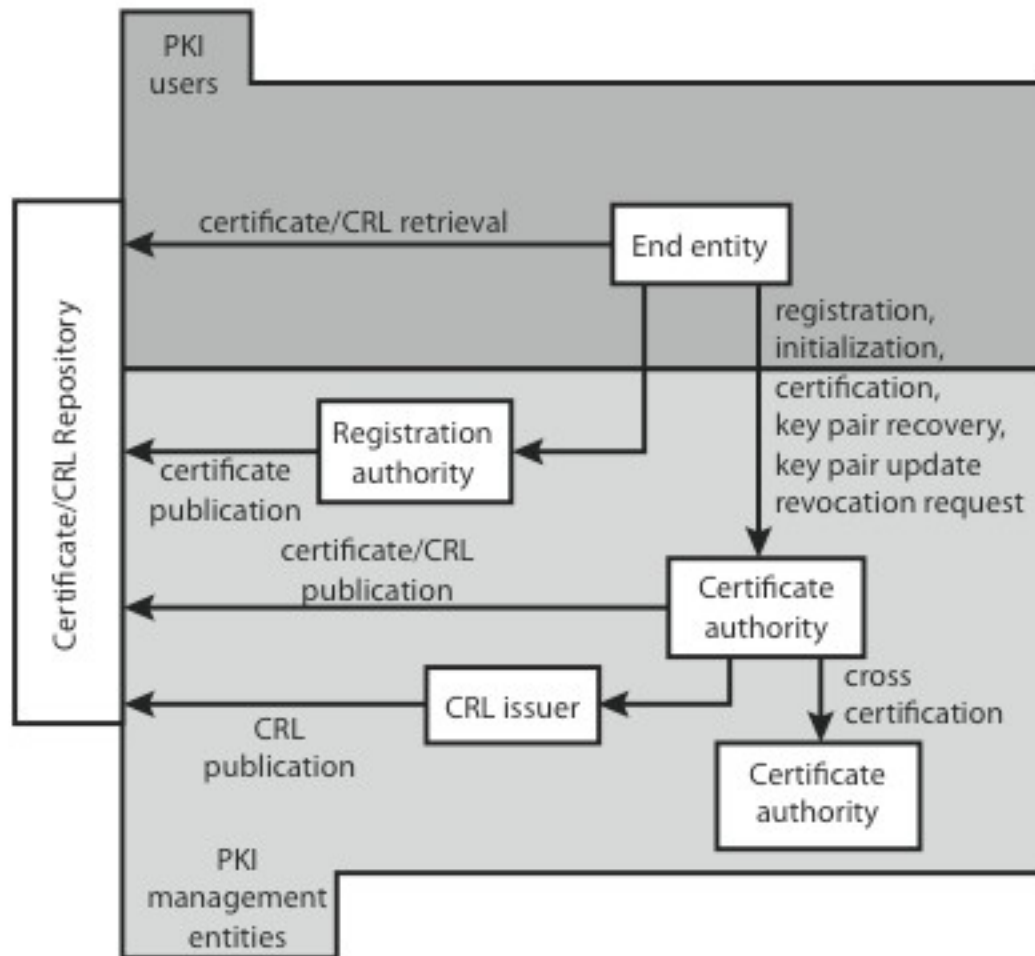
X.509 Version 3

- has been recognised that additional information is needed in a certificate
 - email/URL, policy details, usage constraints
- rather than explicitly naming new fields defined a general extension method
- extensions consist of:
 - extension identifier
 - criticality indicator
 - extension value

Certificate Extensions

- key and policy information
 - convey info about subject & issuer keys, plus indicators of certificate policy
- certificate subject and issuer attributes
 - support alternative names, in alternative formats for certificate subject and/or issuer
- certificate path constraints
 - allow constraints on use of certificates by other CA's

Public Key Infrastructure



Summary

- have considered:
 - Kerberos trusted key server system
 - X.509 authentication and certificates

