


6G7Z1009: Introduction to Computer Forensics and Security

Public Key Infrastructure


Outline

- Public-key Distribution
- X.509
- PKI


Public-key Distribution

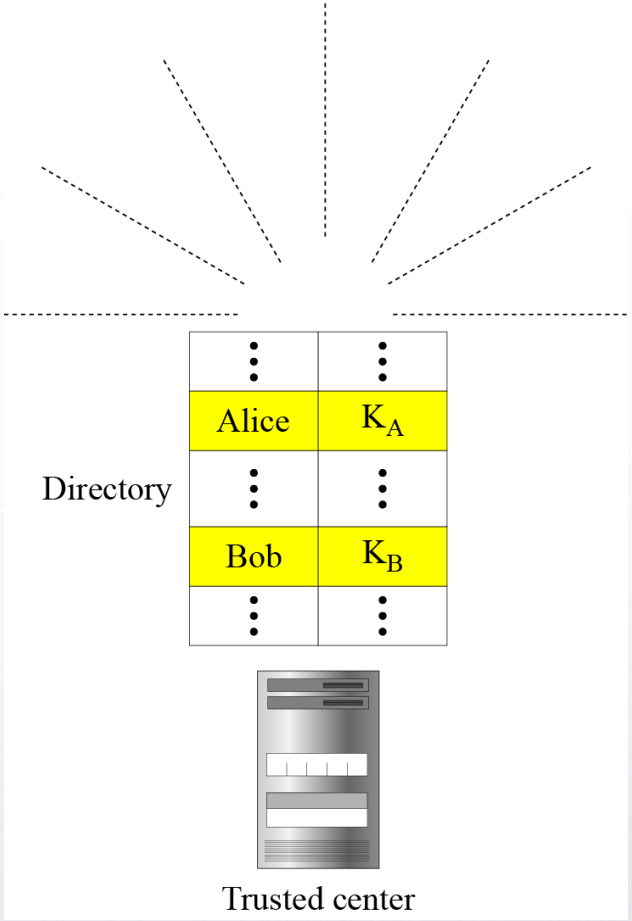
- In asymmetric-key cryptography, people do not need to know a symmetric shared key. The public key is advertised and the private key is kept privately


Trusted Center

- The trusted center retains and updates a directory of public keys. Each user must register with the trusted center and establish a user ID and password. The user can then deliver his/her public key for insertion into the directory
- The center can publicly advertise the director and respond to inquiries about public keys


Trusted Center

- Trusted center

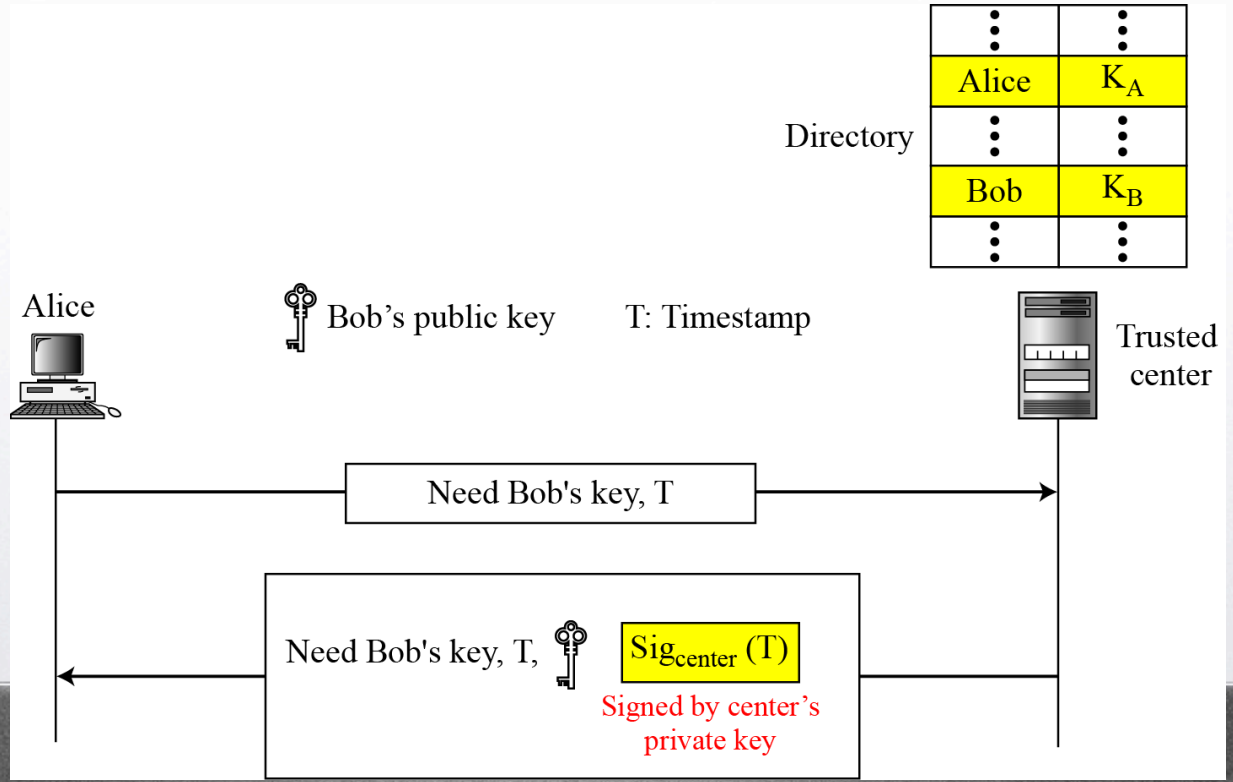



Controlled Trusted Center

- A controlled trusted center achieves higher level of security by adding control on the distribution of the public key. Requests for the public key must include a timestamp. The response of the center to the request includes the timestamp signed with the private key of the center. Alice decrypts the response using the center's public key to verify the timestamp before accepting Bob's public key.

Controlled Trusted Center

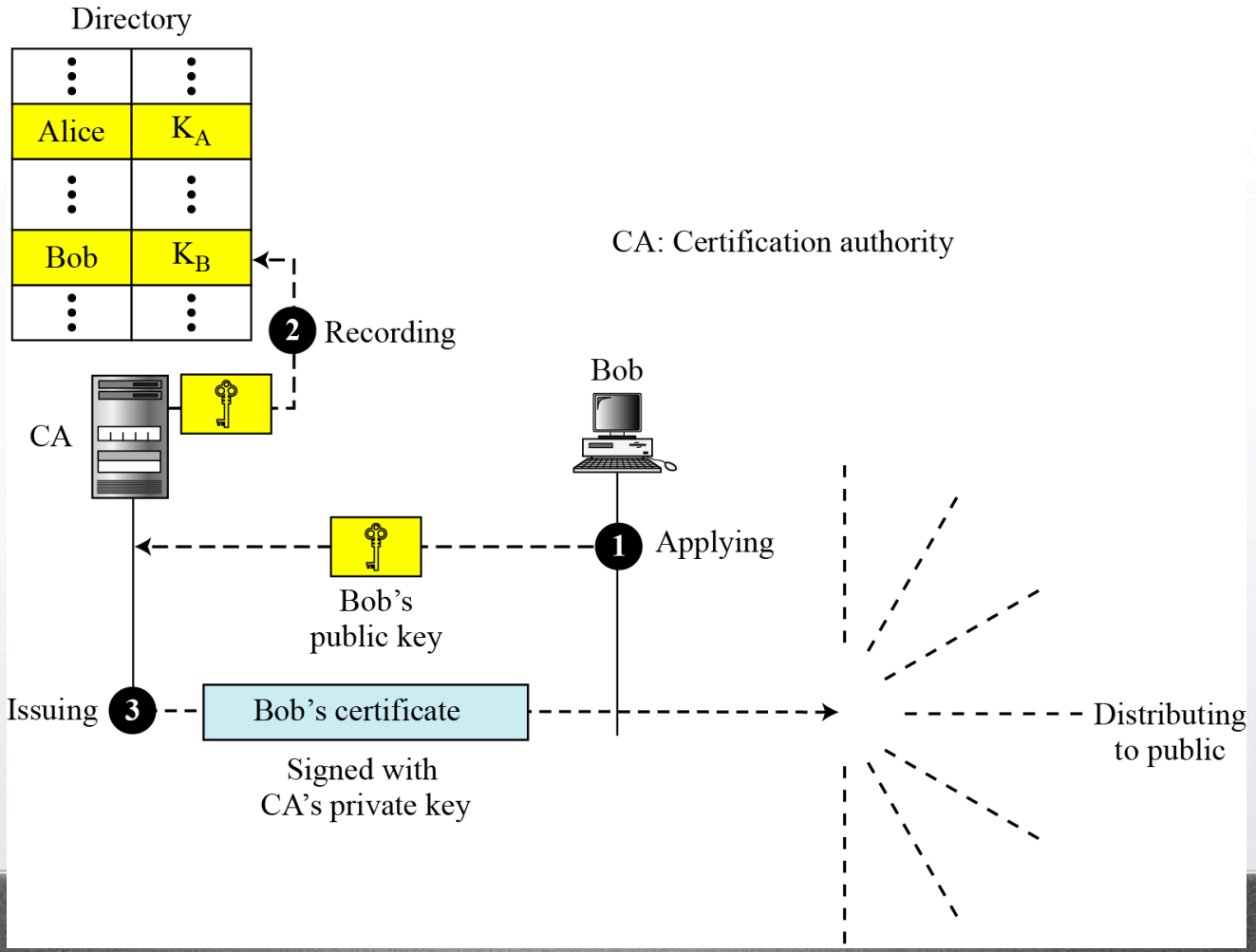
- A controlled trusted center




Certification Authority

- Security certificates are used to reduce the load on trusted centers.
- A server (Bob) can request a certificate from a certification
- The CA checks the identification of Bob. If verified, the CA writes Bob's public key on the certificate and signs it with its own private key
- Bob can now upload the signed certificate and store it on his site or Bob may send the certificate to users upon request.
- Any user who wants Bob's public key can download the certificate and decrypts it using the CA's public key to extract Bob's public key.

Certification Authority




X.509

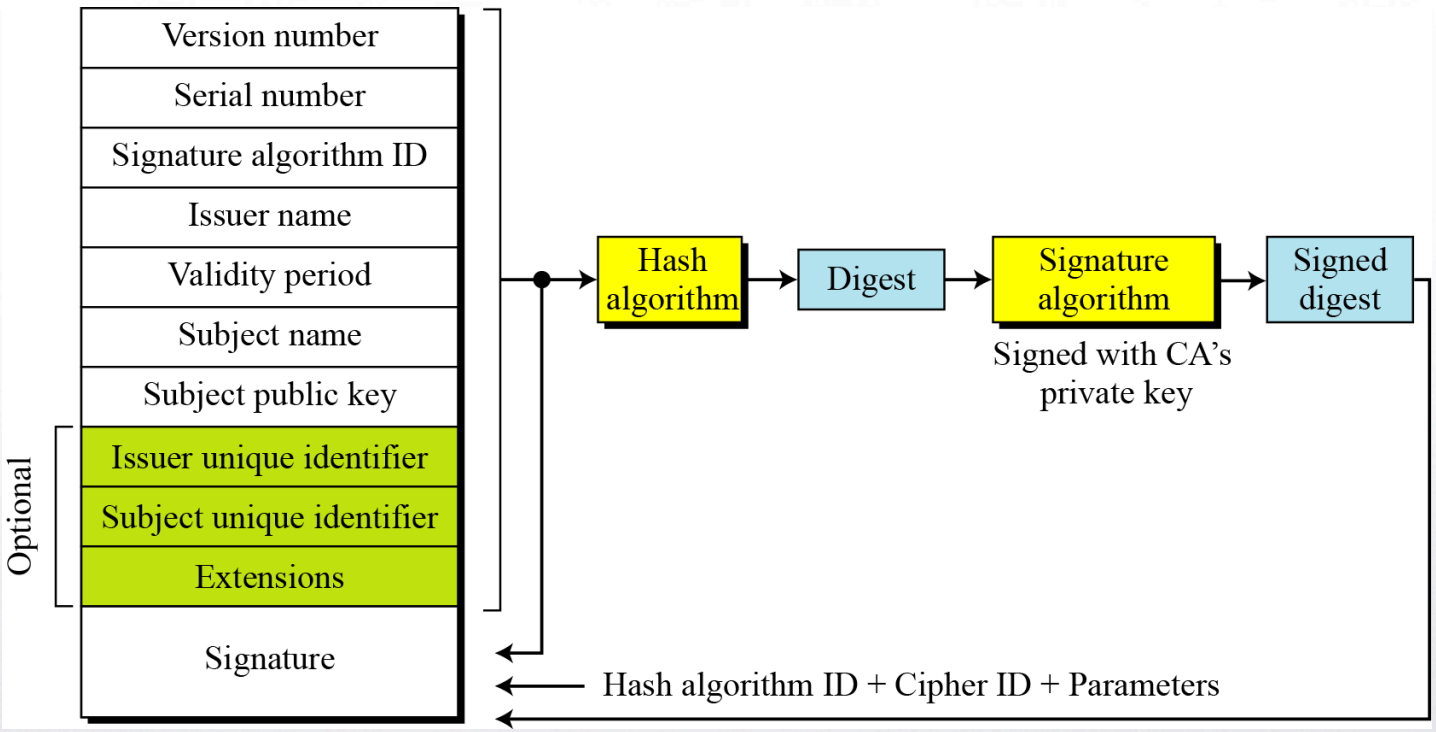
- The Internet community has accepted the ITU-T* recommendation X.509 as a way to unify certificate formats. In X.509, the certificate has the following important fields:
- Version number: this field is the version of X.509 (current version is 3).
- Serial number: this field is the serial number assigned to each certificate and is unique for each certificate issuer.
- Signature algorithm ID: this field identifies the signature algorithm used in the certificate. This field is repeated in the signature field.


X.509

- Issuer name: this field identifies the CA that issued the certificate.
- Validity Period: this field defines the earliest (not before) time and the latest
- Subject name: this field defines the entity that owns the public key stored in this certificate.
- Subject public key: this field gives the value of the public key of the certificate and defines the public key algorithm
- Signature: this field contains the digets of all other fields in the encrypted by CA's private key and also contains the ID of signature algorithm

X.509

- Format

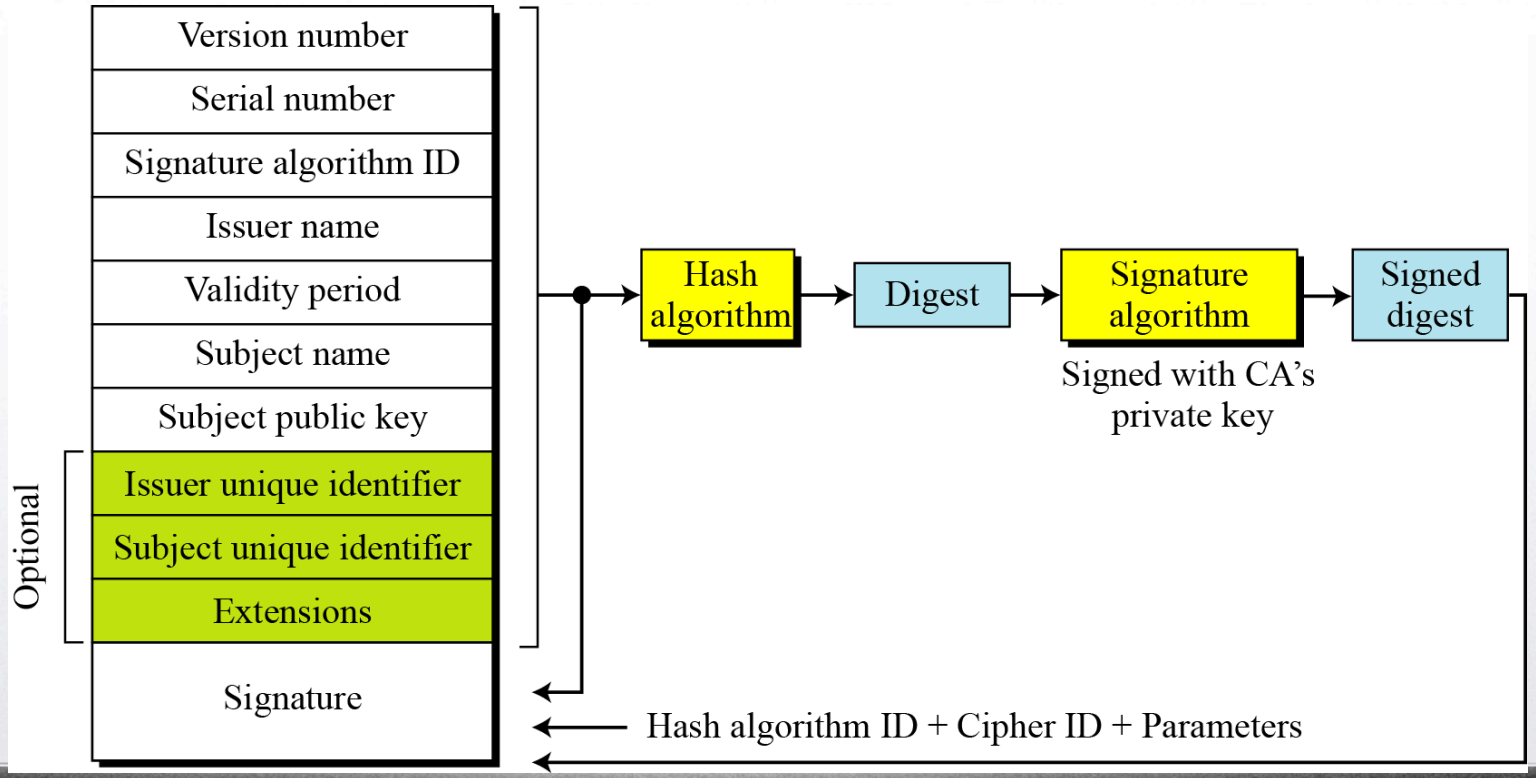

Version number
Serial number
Signature algorithm ID
Issuer name
Validity period
Subject name
Subject public key
Issuer unique identifier
Subject unique identifier
Extensions
Optional
Signature
Hash algorithm
Digest
Signature algorithm
Signed digest
Signed with CA's private key
Hash algorithm ID + Cipher ID + Parameters


X.509

- Certificate Renewal: Each certificate has a period of validity. If there is no problem with the certificate, the CA issues a new certificate before the old one expires.
- Certificate Renewal: In some cases a certificate must be revoked before its expiration.
- Delta Revocation: To make revocation more efficient, the delta certificate revocation list (delta CRL) has been introduced.

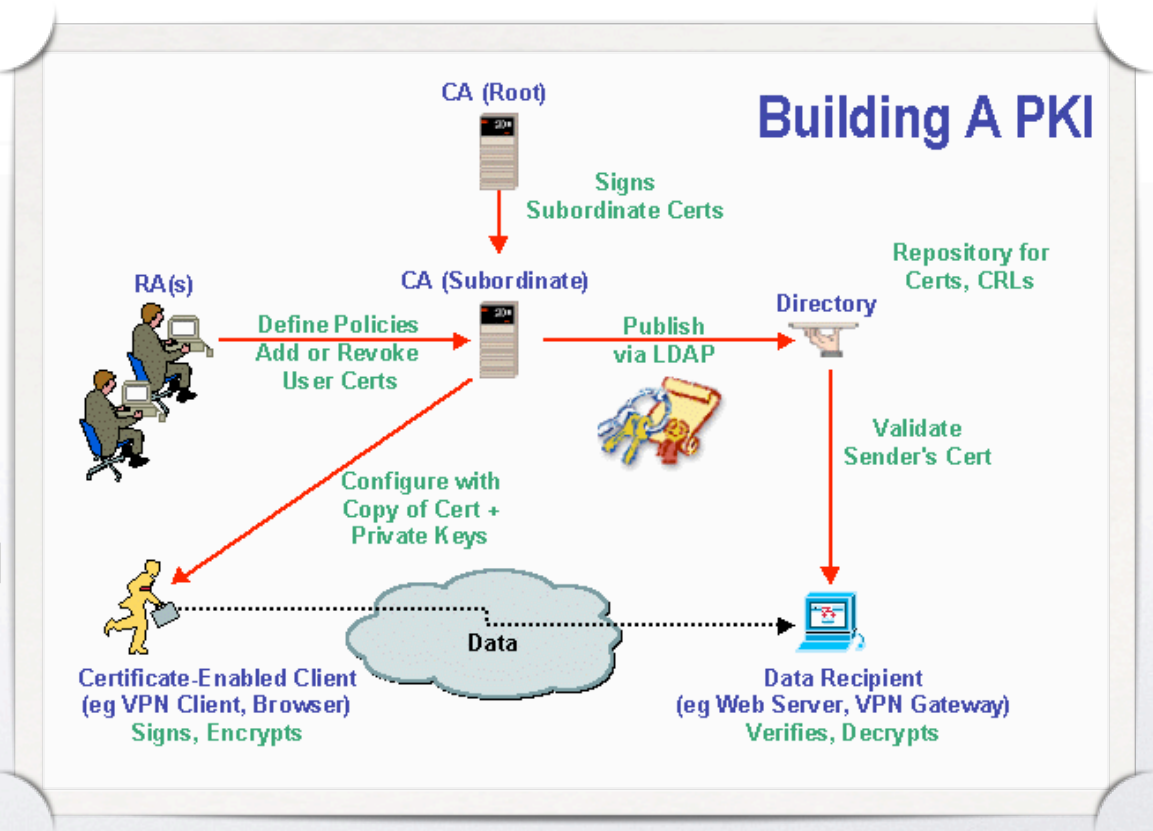
X.509

- Revocation Format



Public-Key Infrastructure

- PKI is a model for creating, distributing and revoking certificates based on the X. 509. IETF (Internet Engineering Task Force) has created the public-key infrastructure X.59 (PKIX)
- PKI (Public Key Infrastructure): is an infrastructure to support and manage public key based digital certificates, i.e., a set of hardware, software, people, process and policies to create, manage, distribute, use, store and revoke digital certificates.




Public-Key Infrastructure

- Some duties of PKI
  - Issue, renew and revoke certificates
  - Store and update private keys for members who wish to hold their private keys at a safe place
  - Provide services to other internet security protocols that need public key info such as IPSec and TLS
  - Provide access control, i.e., provide different levels of access to the information stored in its database


X.509 PKI

- There are different types of certificates: X.509, PGP (Pretty Good Privacy), SPKI (Simple Public key Infrastructure), etc
- We focus on X.509 because it is "the standard standard"


X.509 PKI

- Basic components: provider and consumer sides
- Provider side
  - Certificate Authority (CA)
  - Registration Authority (RA)
  - Certification Distribution System: certs, CRL (certificate revocation list)
- Consumer side: PKI enabled applications


X.509 PKI

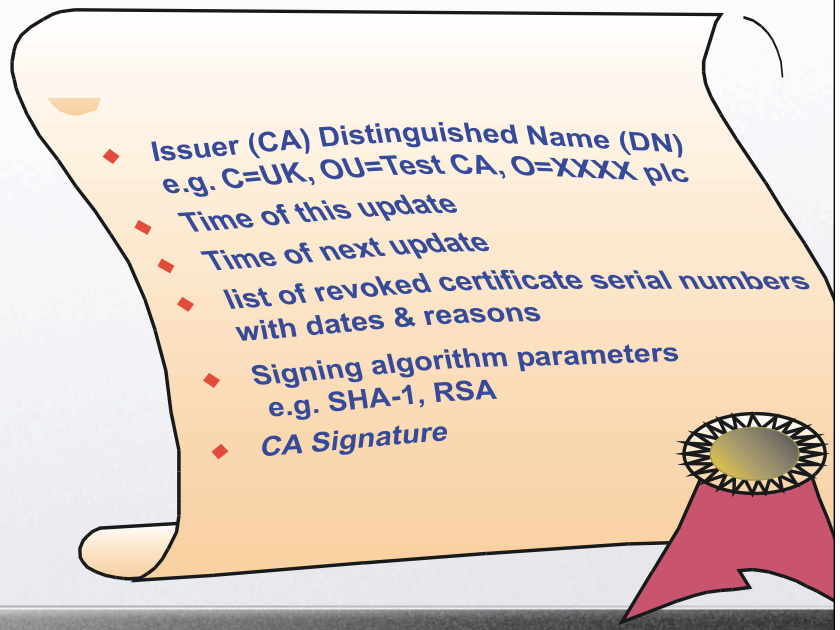
- Certificate Authority (CA): is trusted authority for certifying individuals and creating an electronic document. The basic tasks:
  - Key generation
  - Digital certificate generation
  - Certificate issuance and distribution
  - Revocation
  - Key backup and recovery system
  - Cross certification


X.509 PKI

- Registration Authority (RA): accepts a request for a digital certificate and performs the necessary authority steps of registering and authenticating the person. The basic tasks:
  - Registration of certificate information
  - Face-to-face registration
  - Remote registration
  - Automatic registration
  - Revocation
  - Cross certification

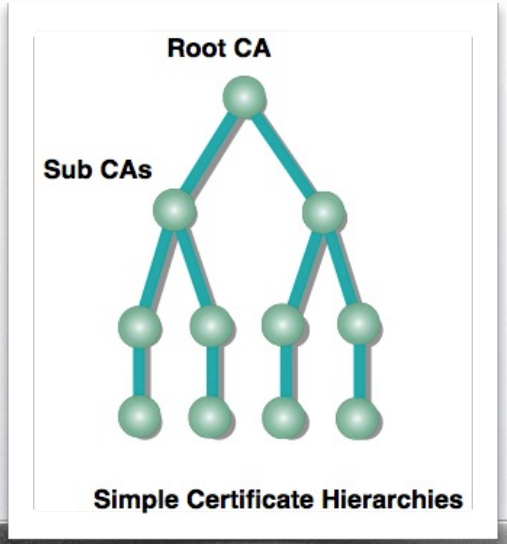

X.509 PKI

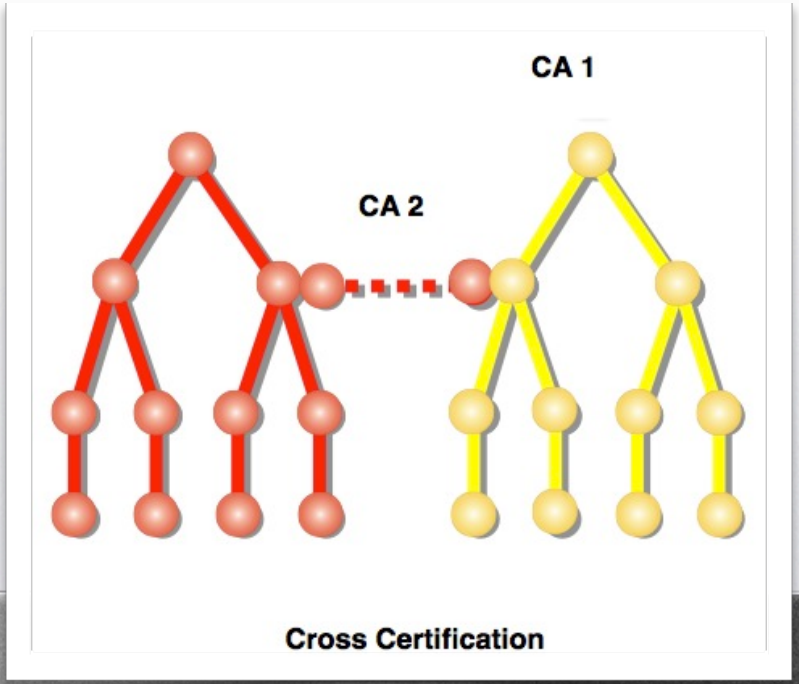
- Certificate distribution system
  - Digital certificates
  - Certification revocation lists (CRLs)
  - Special purpose databases
  - LDAP directories



X.509 PKI

- X.509 PKI trust and legal aspects
  - Trust approaches: certificate hierarchies, cross-certification, etc.






X.509 PKI

- X.509 PKI trust and legal aspects
  - Legal aspect: Certificate policies and certificate policy statement (CPS)
    - Certificate policy: a document that sets out the rights, duties and obligations of each party in a public key infrastructure
    - Certificate policy statement: is a document which may have legal effect in limited circumstances


X.509 PKI

- PKI trust model: for scalability, there should be many certification authorities in the world. Each CA handles a specified number of certificates. The PKI trust model defines rules that specify how a user can verify a certificate received from a CA
- As an example, the PKI hierarchical trust model defines hierarchical rules that specify how a user can verify a certificate received from a CA
- PKI uses the following notation to denote the certificate issued and signed by certification authority X for entity Y X<<Y>>

X.509 PKI

- User1 knows only the public key of the root CA. Show how can User1 obtain a verified copy of User3's public key.
- Solutions

User3 sends a chain of certificates, CA<<CA1>> and CA1<<User3>>, to User1.

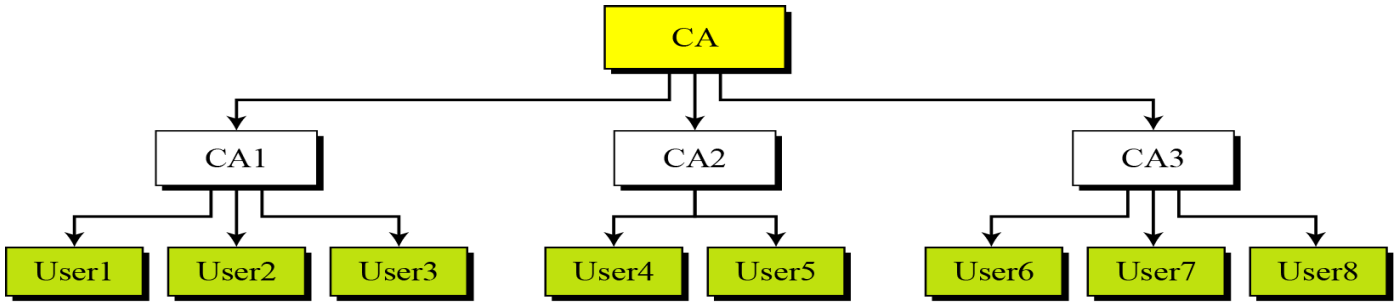
a. User1 validates CA<<CA1>> using the public key of CA.

b. User1 extracts the public key of CA1 from CA<<CA1>>.

c. User1 validates CA1<<User3>> using the public key of CA1.

d. User1 extracts the public key of User 3 from CA1<<User3>>.

Users1 has used the following chain **CA<<CA1>> CA1<<User3>>**




CA hierarchy-Certificate validation path

- A only knows the public key of X and B only knows the public key of Z.
- A acquires B's certificate using the chain:

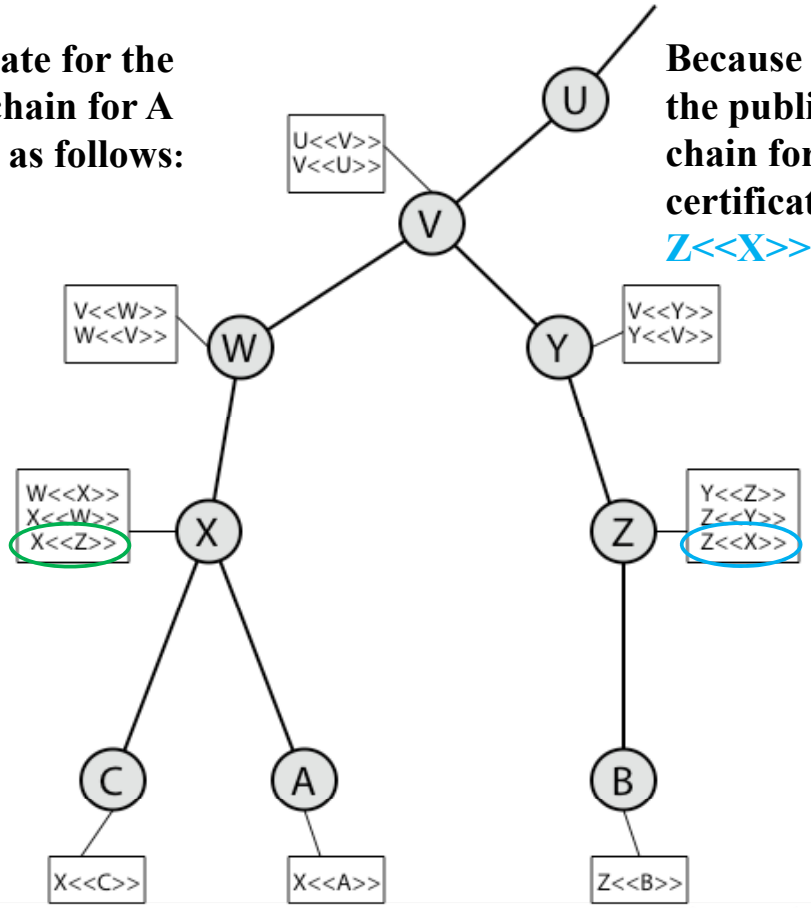
X<<W>> W<<V>> V<<Y>> Y<<Z>> Z<<B>>

- B acquires A's certificate using the chain:

Z<<Y>> Y<<V>> V<<W>> W<<X>> X<<A>>

CA hierarchy-Certificate validation path

Because X signed a certificate for the public key of Z, a shorter chain for A to acquire B's certificate is as follows:
X<<Z>> Z<<B>>



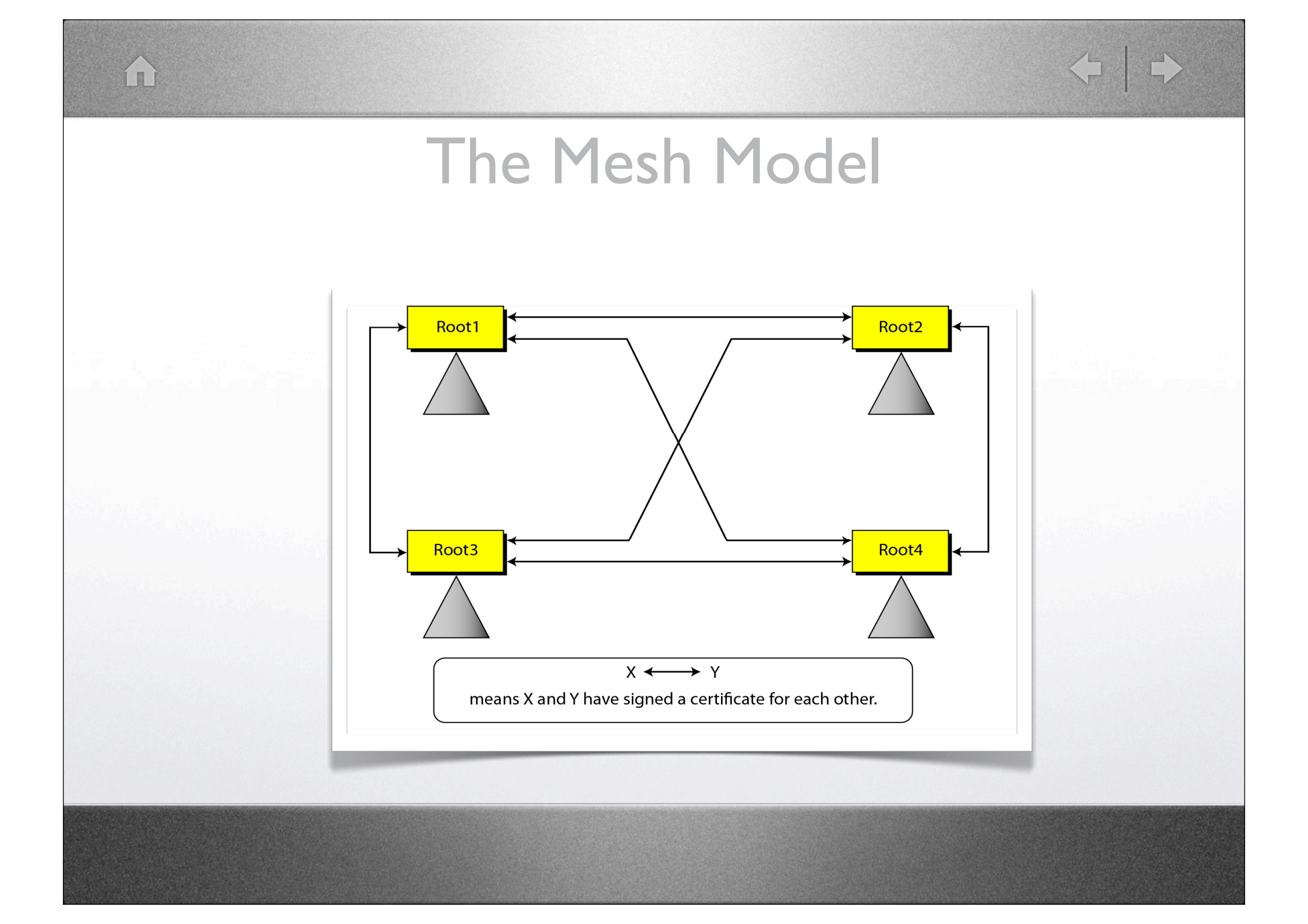
Because Z signed a certificate the public key of X, a shorter chain for B to acquire A's certificate is as follows:
Z<<X>> X<<A>>


CA hierarchy-Certificate validation path

- Some web browsers, such as internet explorer, include a set of certificates from independent roots without a single, high-level authority to certify each root. One can find the list of these roots in the Internet Explorer at Tools/Internet Options/Contents/Certificate/Trusted roots (using pull-down menu). The user then can choose any of these roots and view the certificate.


The Mesh Model

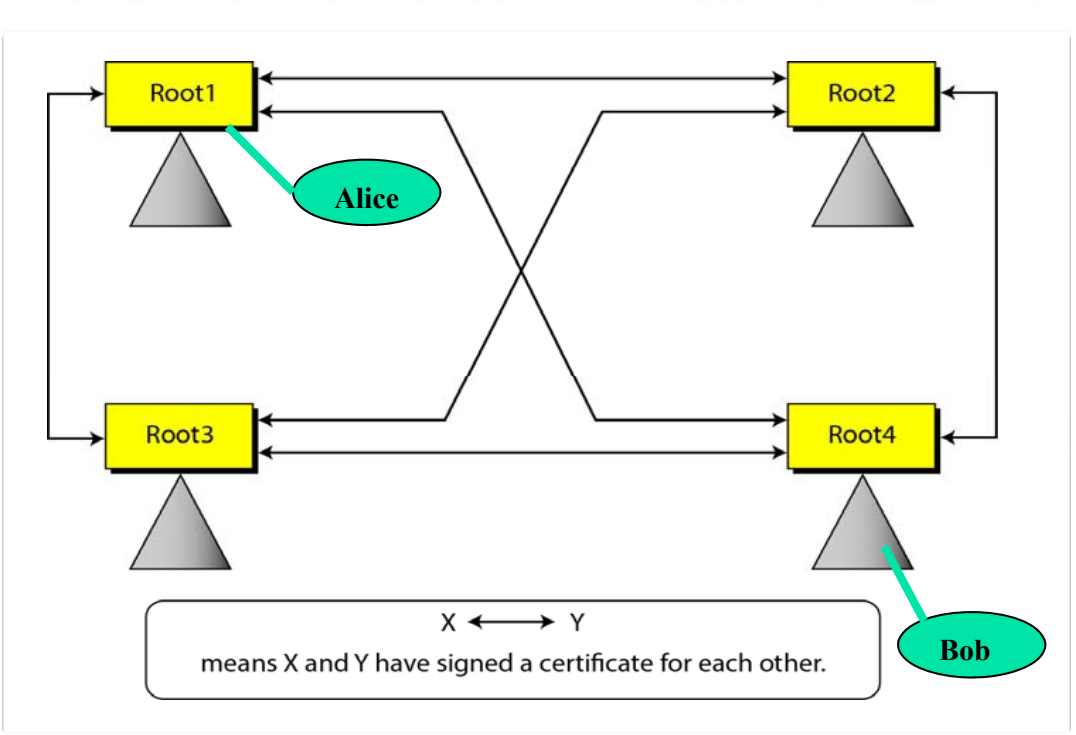
- The hierarchical model with a single root is not suitable for a very large community. The mesh model that connects several roots together, is more useful. Roots are connected by the mesh but each root has its own hierarchical structure. Certificates in the mesh are cross-certificates, i.e., each root certifies each other root. For a fully connected mesh with four root certifies each other root. For a fully connected mesh with four roots, we need 4*3=12 certificates. Each double-arrow represents 2 certificates


The Mesh Model
Root1
Root2
Root3
Root4
X ⟷ Y
means X and Y have signed a certificate for each other.


The Mesh Model

- Alice is under the authority Root1; Bob is under the authority Root4. Show how Alice can obtain Bob's verified public key.
- Solution: Alice looks at the directory of Root 1 to find Root1<<Root4>>. Using the public key of Root4. Alice can follow the chain of certificates from Root4 to Bob. Alice can then extract and verify Bob's public key.

The Mesh Model




PKI in Use

- Some examples:
  - The federal public key infrastructure FPKI authority
  - NASA PKI
  - State of Illinois PKI
  - University of Wisconsin PKI


PKI in Use

- Secure e-mail
- Virtual private network
- Wireless
- Web servers (SSL/TLS)
- Network authentication