Key Management and Distribution

Department of Computer Engineering

CRYPTOGRAPHY

Faculty-
Vivek Kumar Anand

Key Management & Distribution

- Topics of cryptographic key management / key distribution are complex
  - Cryptographic, protocol, & management issues
- Symmetric schemes require both parties to share a common secret key
- Public key schemes require parties to acquire valid public keys
- Have concerns with doing both

Key Distribution

- Symmetric schemes require both parties to share a common secret key
- Issue is how to securely distribute this key
- Whilst protecting it from others
- Frequent key changes can be desirable
- Often secure system failure due to a break in the key distribution scheme

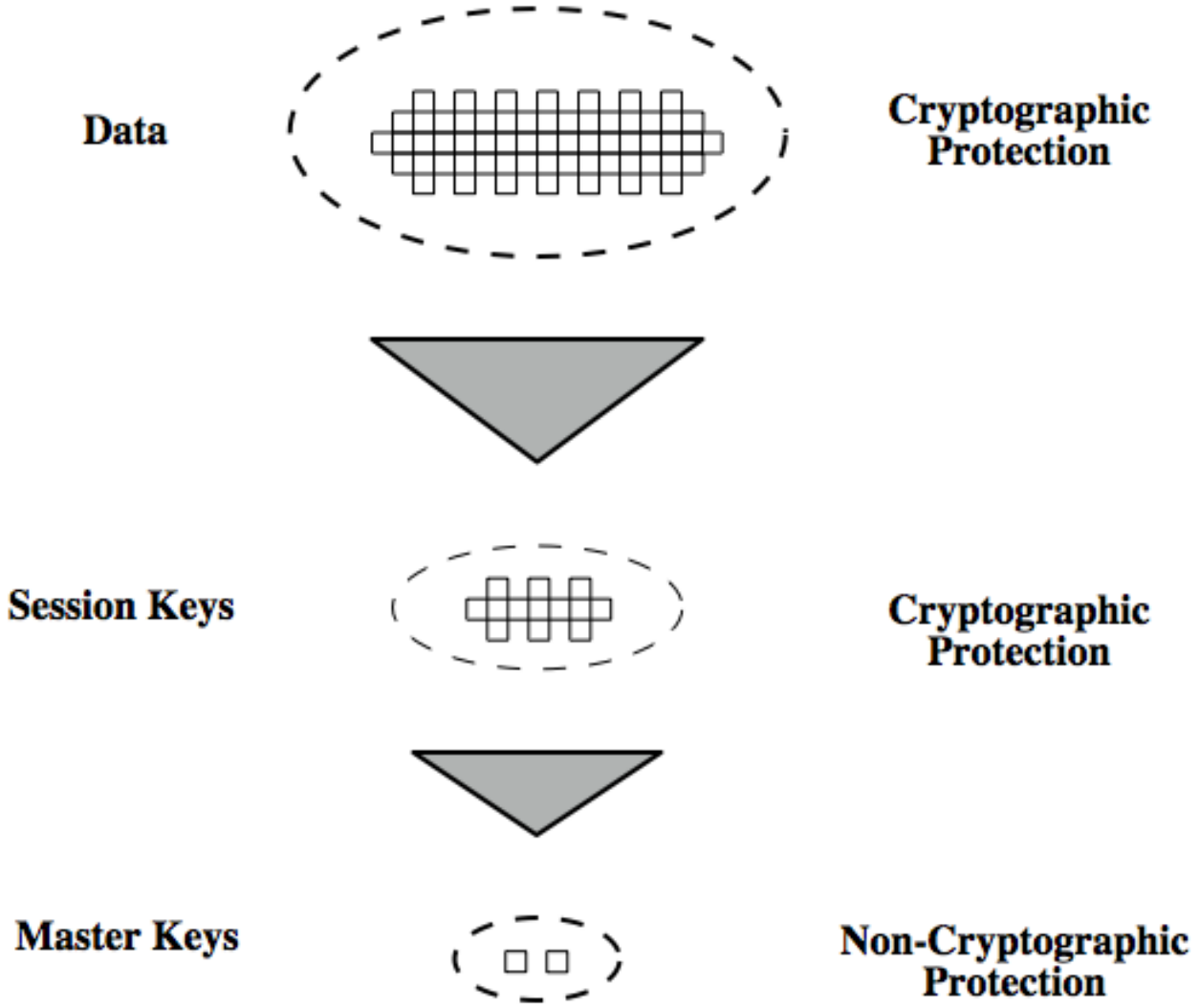
Symmetric Key Distribution Using Symmetric Encryption

- Given parties A and B have various **key distribution** alternatives:
  1. A can select key and physically deliver to B
  2. Third party can select & deliver key to A & B
  3. If A & B have communicated previously can use previous key to encrypt a new key
  4. If A & B have secure communications with a third party C, C can relay key between A & B

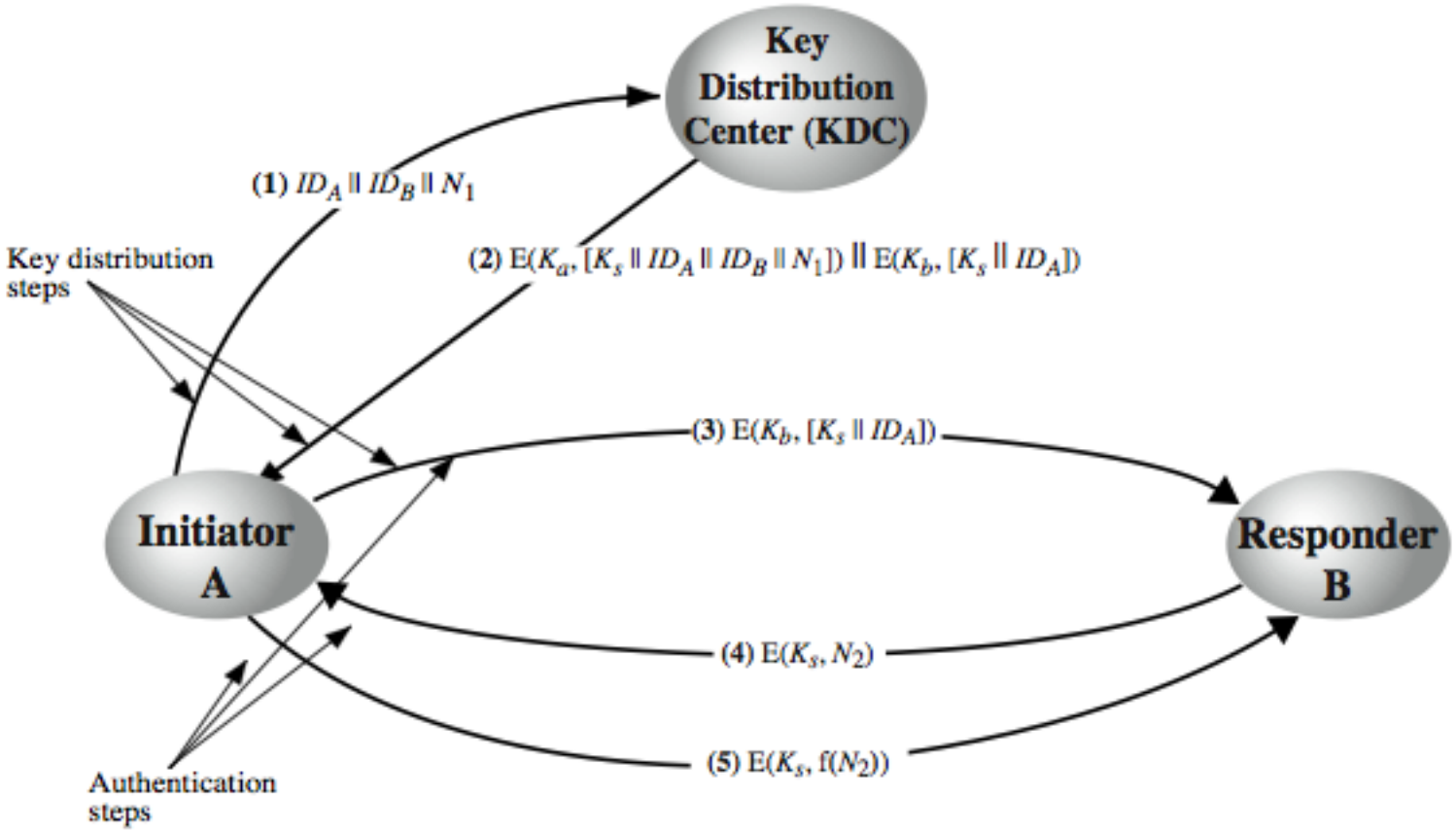
Key Hierarchy

- Typically have a hierarchy of keys
- Session key
  - Temporary key
  - Used for encryption of data between users
  - For one logical session then discarded
- Master key
  - Used to encrypt session keys
  - Shared by user & key distribution center

Key Hierarchy



Key Distribution Scenario

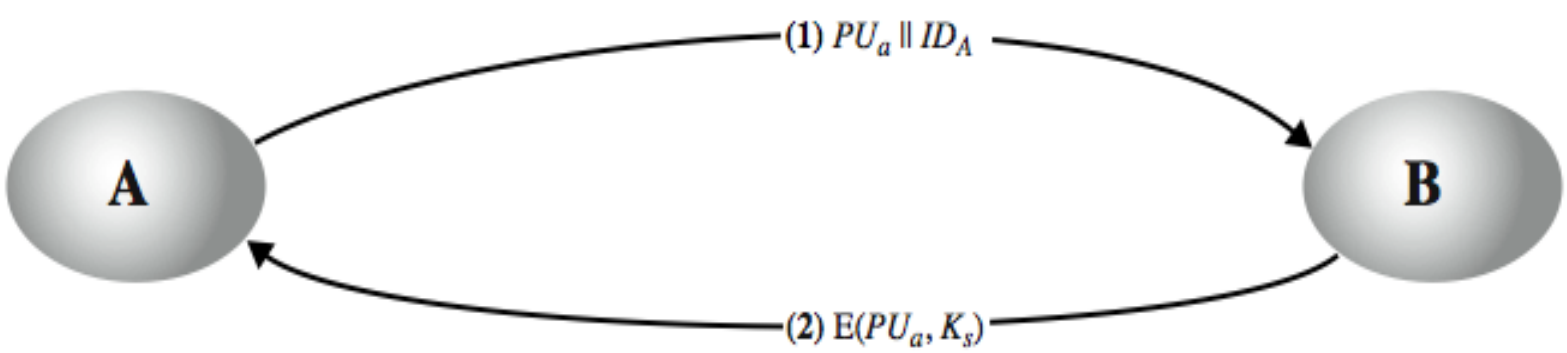


Key Distribution Issues

- Hierarchies of kdc’s required for large networks, but must trust each other
- Session key lifetimes should be limited for greater security
- Use of automatic key distribution on behalf of users, but must trust system
- Use of decentralized key distribution
- Controlling key usage

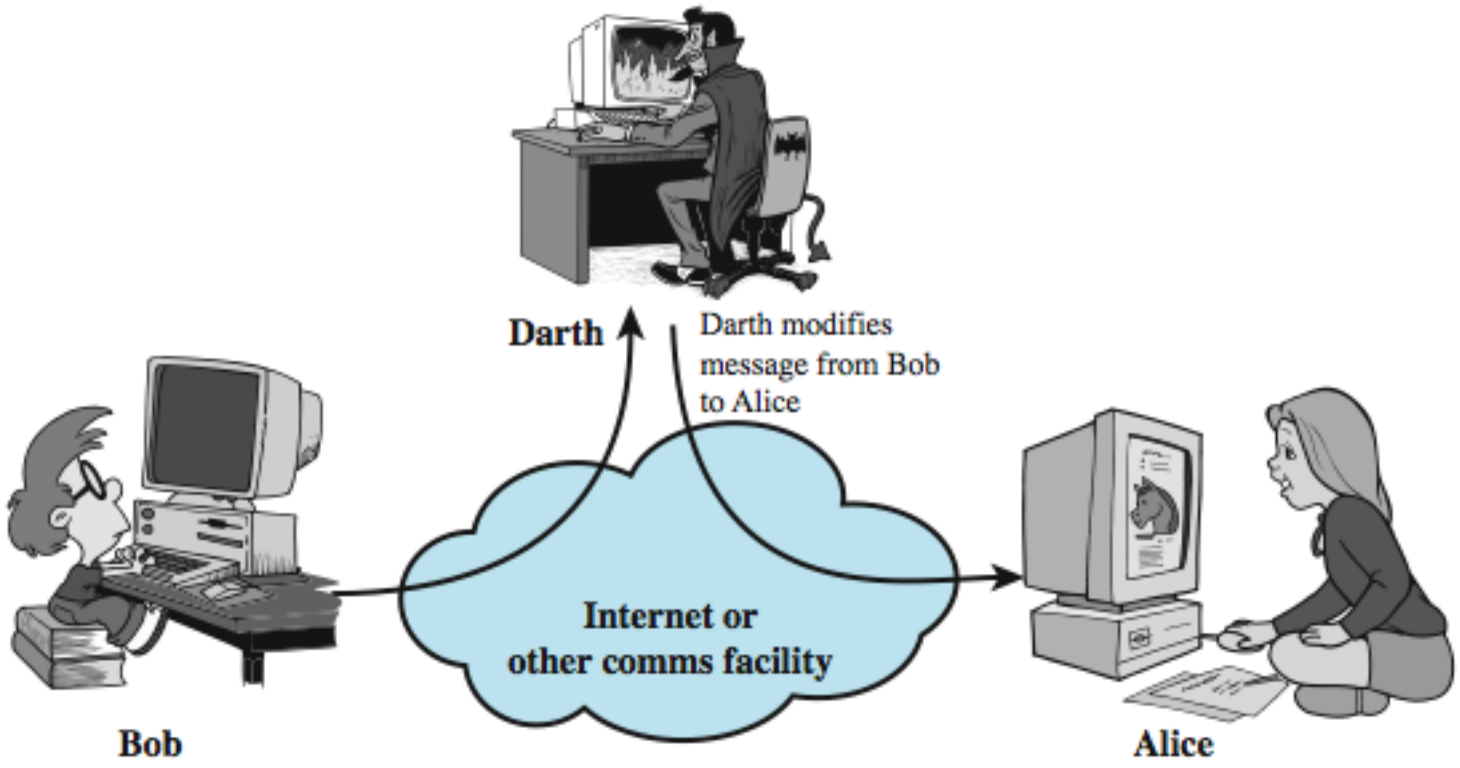
Simple Secret Key Distribution

- Merkle proposed this very simple scheme
  - Allows secure communications
  - No keys before/after exist

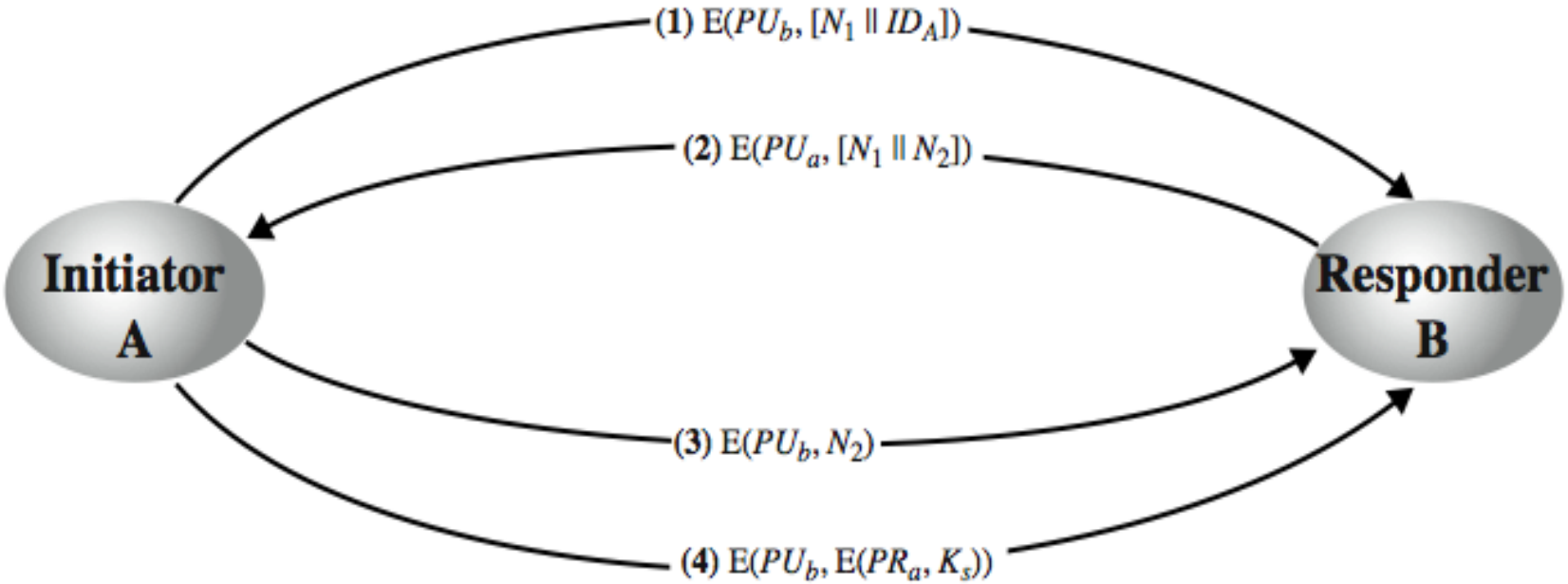


Man-in-the-Middle Attack

- This very simple scheme is vulnerable to an active man-in-the-middle attack



Secret Key Distribution with Confidentiality and Authentication



Hybrid Key Distribution

- retain use of private-key KDC
- shares secret master key with each user
- distributes session key using master key
- public-key used to distribute master keys
  - especially useful with widely distributed users

Distribution of Public Keys

- can be considered as using one of:
  - 1. public announcement
  - 2. publicly available directory
  - 3. public-key authority
  - 4. public-key certificates

1. Public Announcement

- users distribute public keys to recipients or broadcast to community at large
  - eg. append PGP keys to email messages or post to news groups or email list
- major weakness is forgery
  - anyone can create a key claiming to be someone else and broadcast it
  - until forgery is discovered can masquerade as claimed user

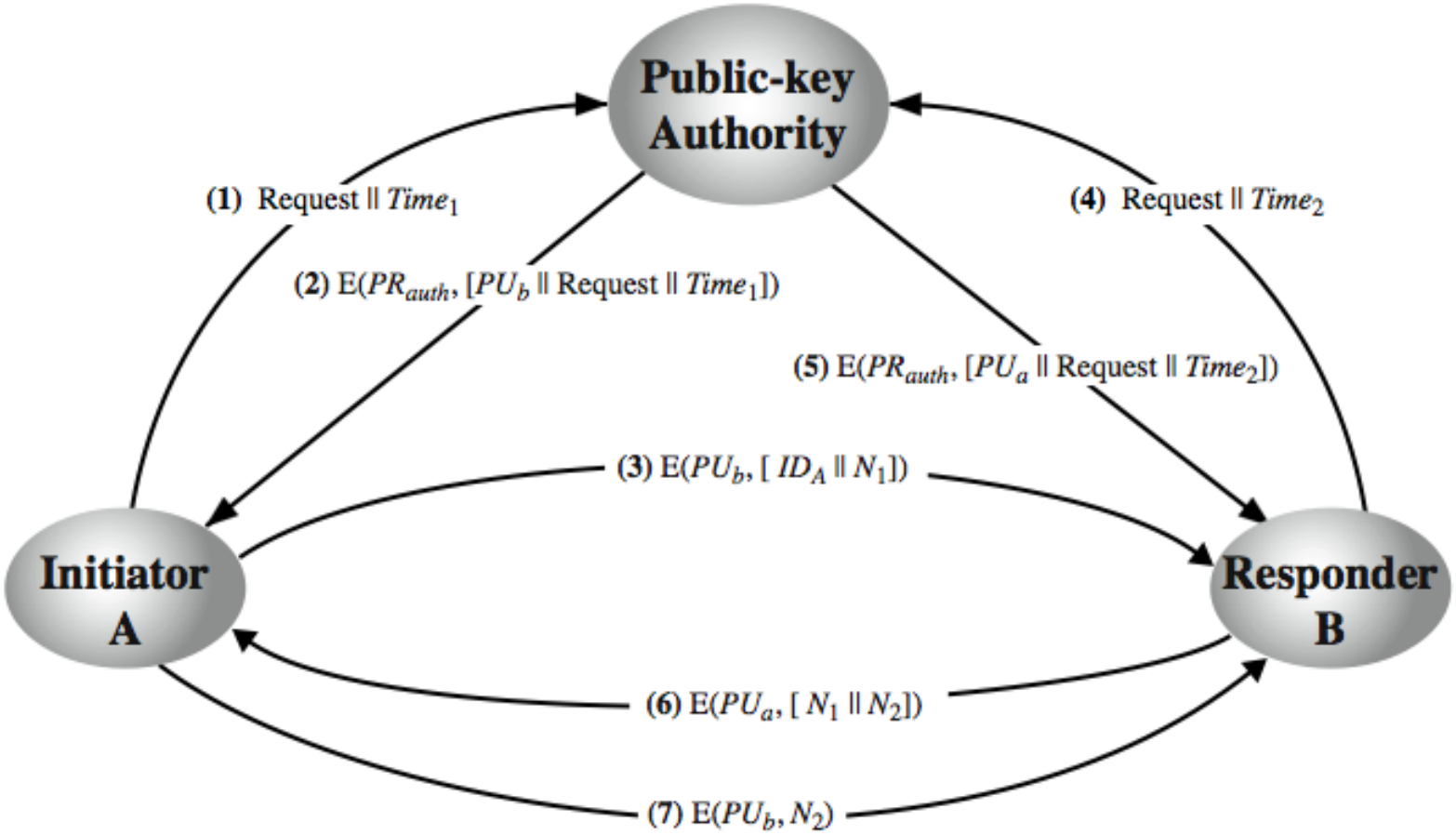
2. Publicly Available Directory

- can obtain greater security by registering keys with a public directory
- directory must be trusted with properties:
  - contains {name,public-key} entries
  - participants register securely with directory
  - participants can replace key at any time
  - directory is periodically published
  - directory can be accessed electronically
- still vulnerable to tampering or forgery

3. Public-Key Authority

- improve security by tightening control over distribution of keys from directory
- has properties of directory
- and requires users to know public key for the directory
- then users interact with directory to obtain any desired public key securely
  - does require real-time access to directory when keys are needed
  - may be vulnerable to tampering

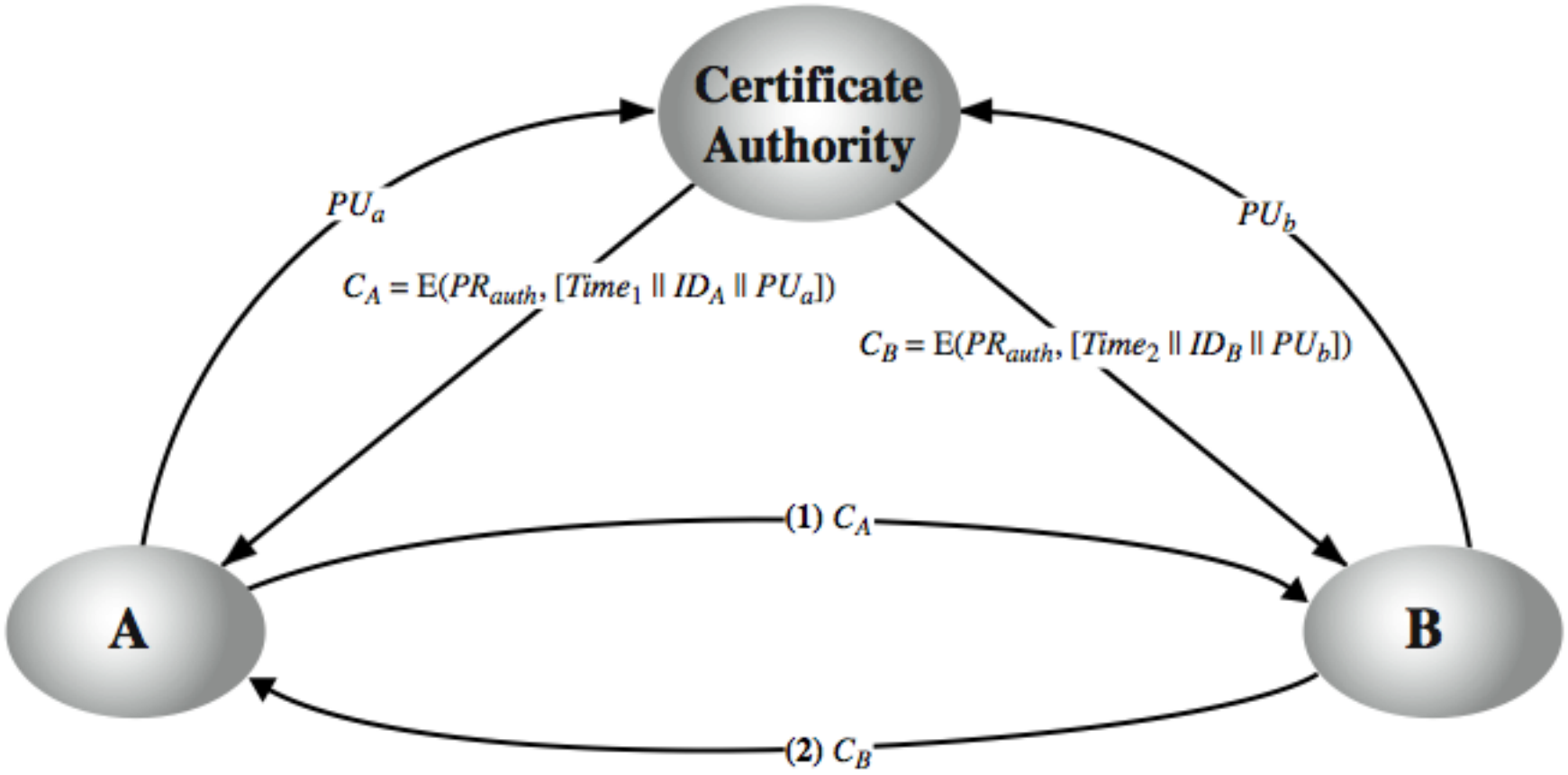
3. Public-Key Authority



4. Public-Key Certificates

- certificates allow key exchange without real-time access to public-key authority
- a certificate binds **identity** to **public key**
  - usually with other info such as period of validity, rights of use etc
- with all contents **signed** by a trusted Public-Key or Certificate Authority (CA)
- can be verified by anyone who knows the public-key authorities public-key

4. Public-Key Certificates



X.509 Authentication Service

- 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
  - have 3 versions

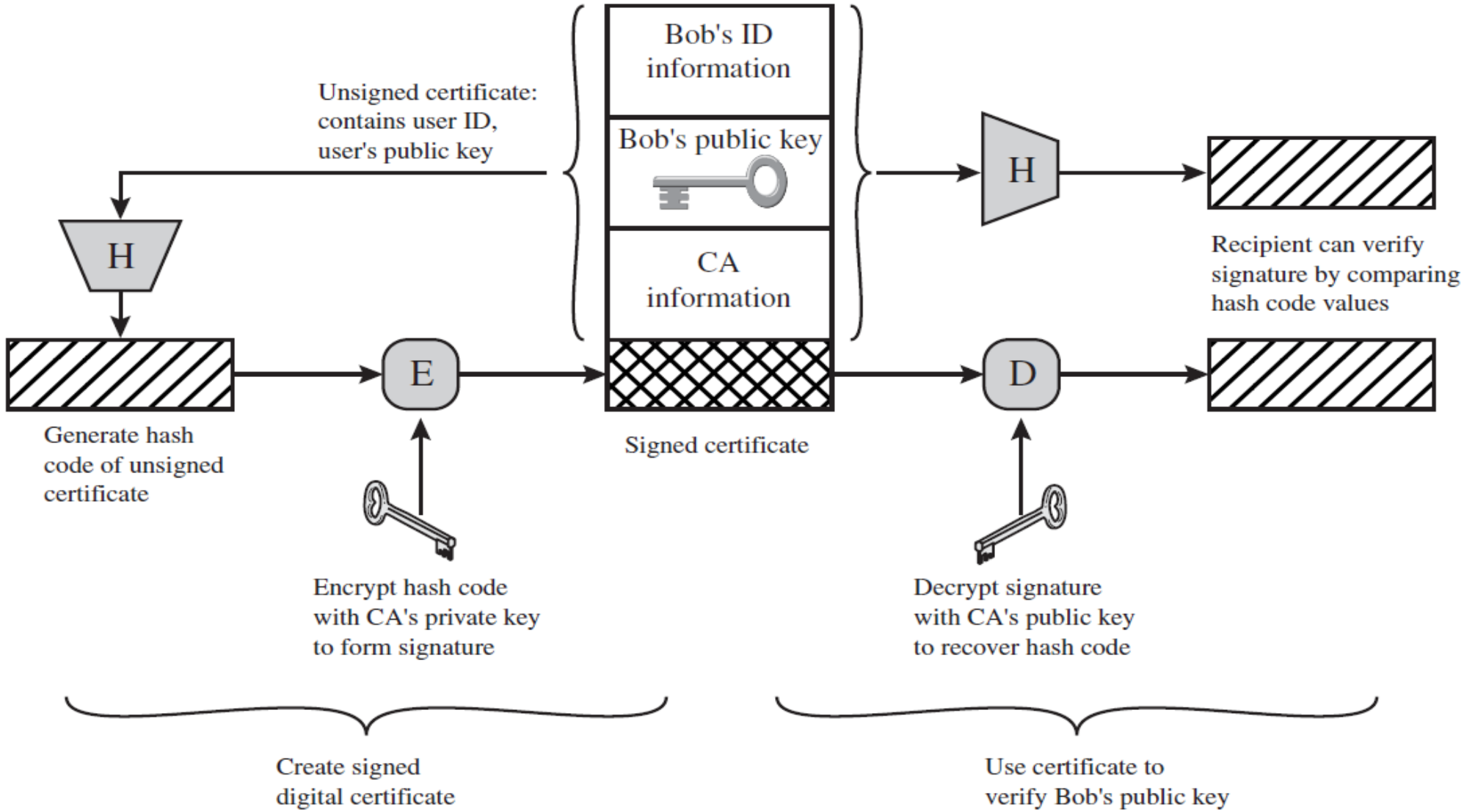
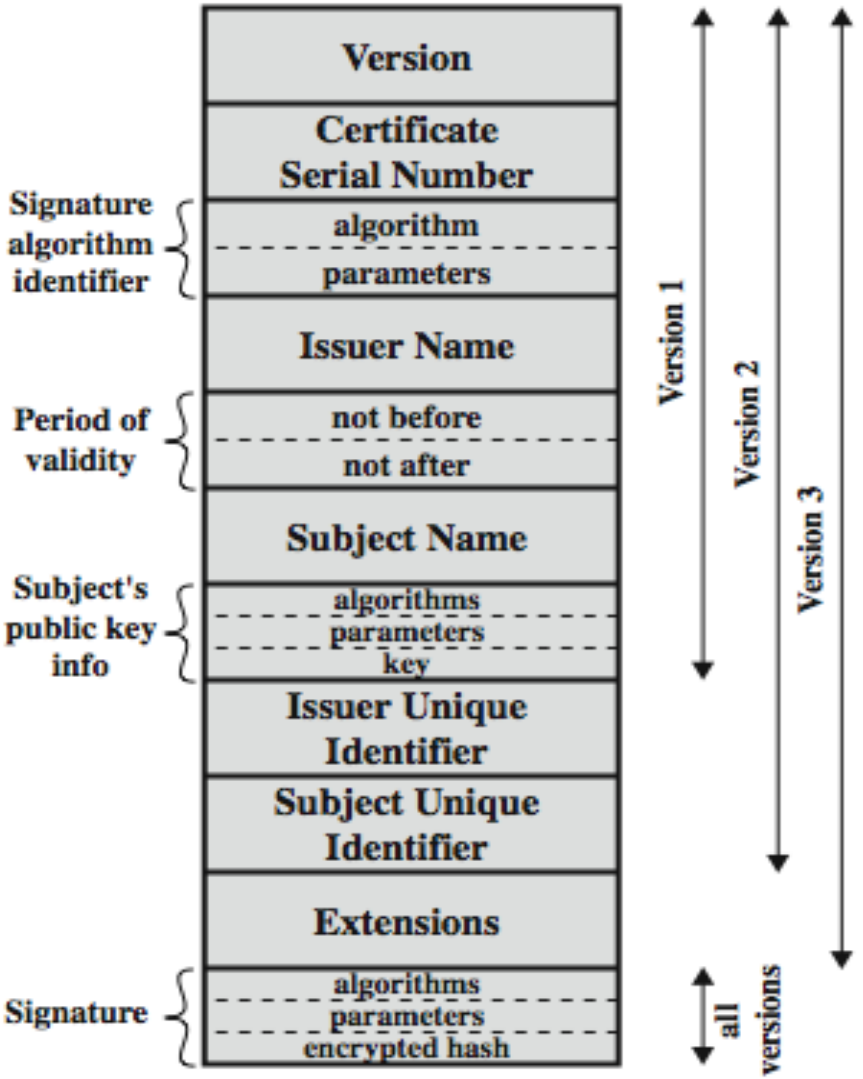


Figure 14.13 Public-Key Certificate Use

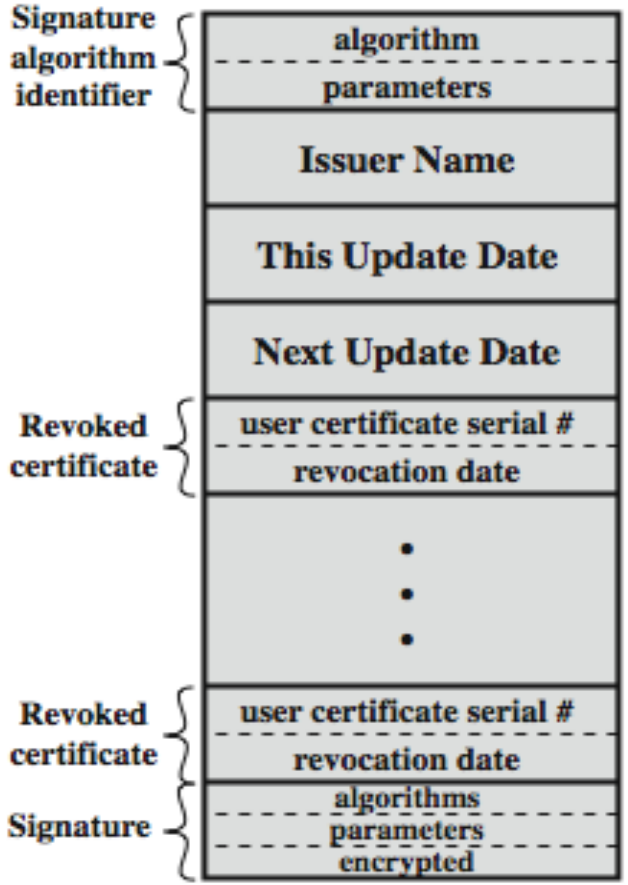
X.509 Certificates

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

X.509 Certificates



(a) X.509 Certificate



(b) Certificate Revocation List

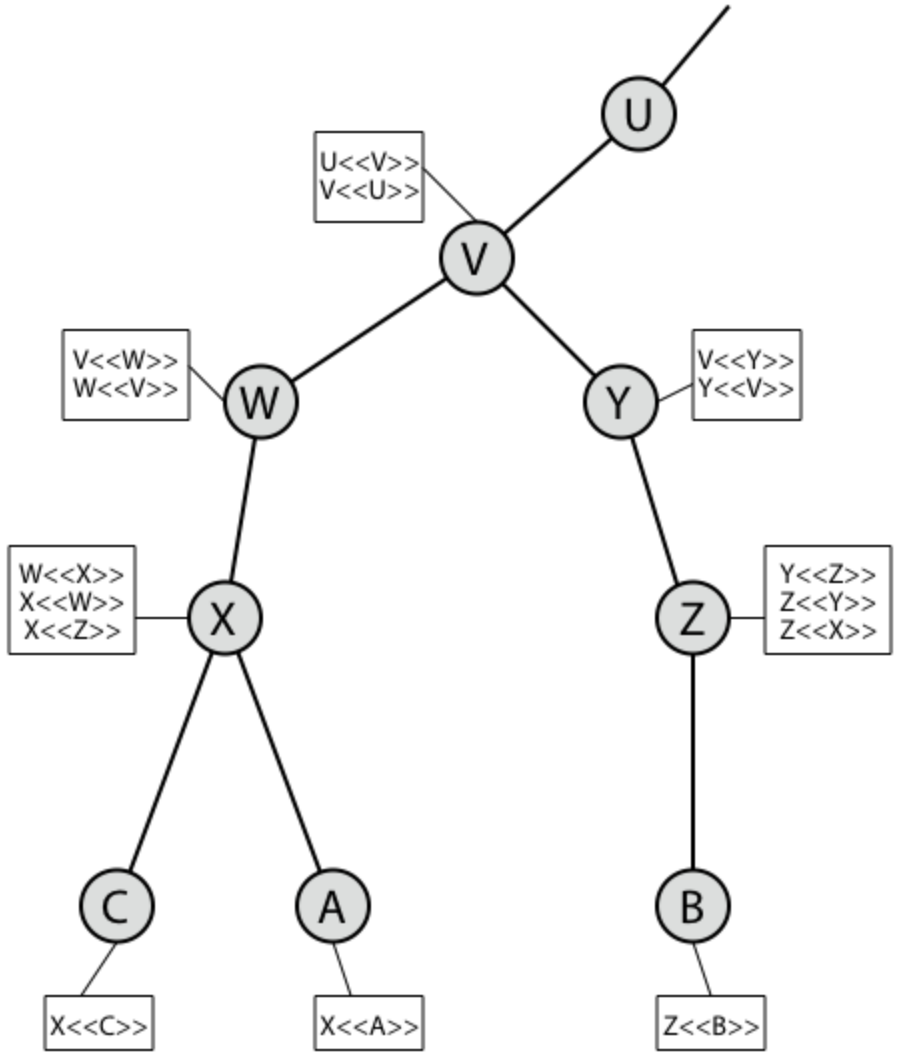
Obtaining a Certificate

- 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

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

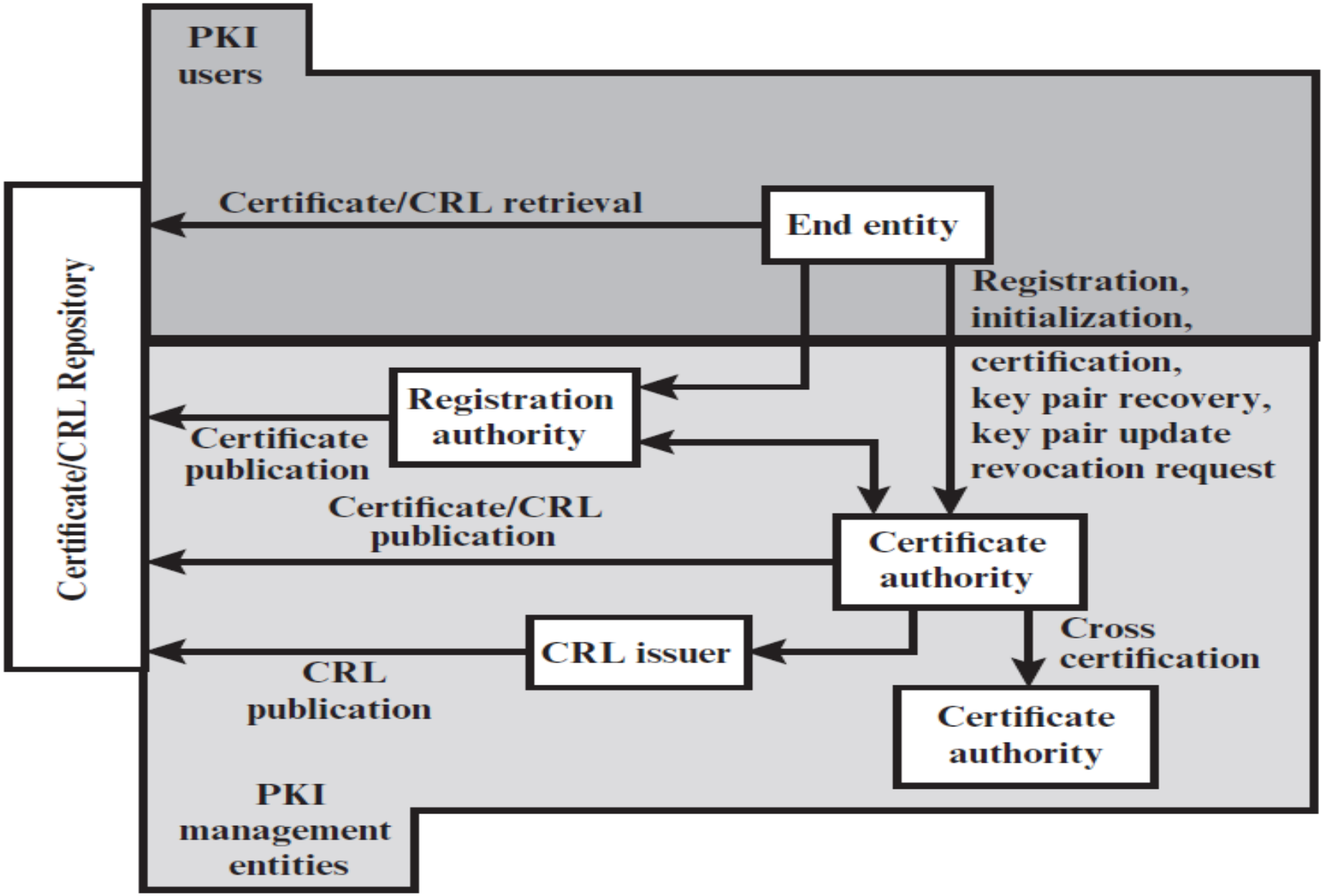


Figure 14.17 PKIX Architectural Model

PKIX Management

- functions:
  - registration
  - initialization
  - certification
  - key pair recovery
  - key pair update
  - revocation request
  - cross certification

Summary

- have considered:
  - symmetric key distribution using symmetric encryption
  - symmetric key distribution using public-key encryption
  - distribution of public keys
    - announcement, directory, authrority, CA
  - X.509 authentication and certificates
  - public key infrastructure (PKIX)