Key Management

Dept. of CSE, NITK

Placement of Encryption Function

Points of Vulnerability:

- Adversary can eavesdrop from a machine on the same LAN.
- Adversary can eavesdrop by dialing into communication server.
- Adversary can eavesdrop by gaining physical control of part of external links.
  - twisted pair, coaxial cable, or optical fiber
  - radio or satellite links

Consider Typical Scenarios

- Workstations on LANs access other workstations & servers on LAN.
- LANs interconnected using switches/ routers.
- With external lines or radio/satellite links.

Consider Attacks and Placement in this Scenario

- Snooping from another workstation.
- Use dial-in to LAN or server to snoop.
- Use external router link to enter & snoop.
- Monitor and/or modify traffic one external links.

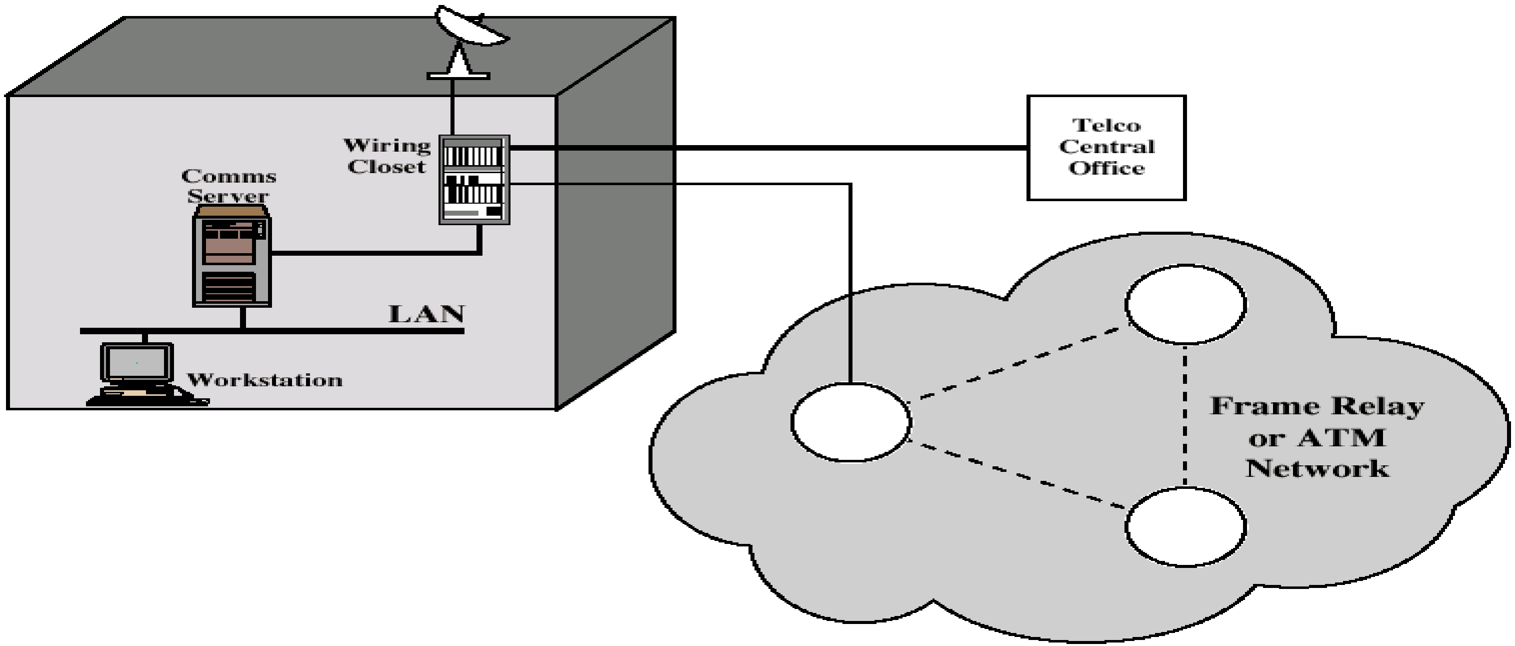


Figure 7.1 Points of Vulnerability

Confidentiality using Symmetric Encryption

- Have two major placement alternatives.

  Link Encryption

  End-to-End Encryption.

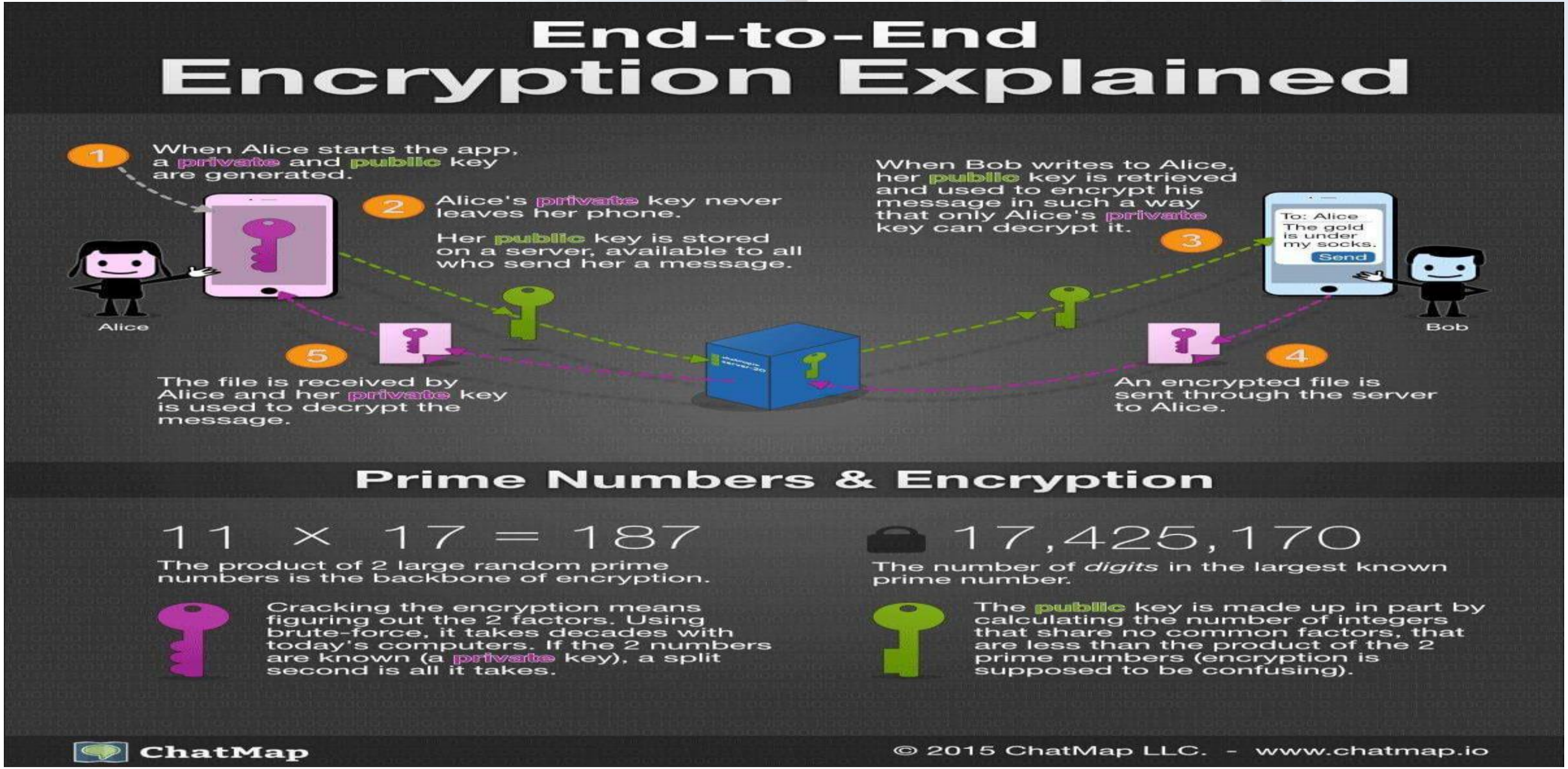
Link Encryption

- Encryption occurs independently on every link.
- All traffic over all communication links is secured.
- Implies must decrypt traffic between links because the switch must read the address in the packet header.
- Each pair of nodes that share a unique key, with a different key used on each link, many keys.
- If working with a public network, the user has not control over the security of the nodes.
- Message is vulnerable at each switch.

End-to-End Encryption

- Encryption occurs between original source and final destination.
- Need devices at each end with shared keys.
- Secure the transmission against attacks on the network links or switches.
- A degree of authentication, only alleged sender shares the relevant key.
- What part of each packet will the host encrypt? Header or user data?

ChatMap: An example



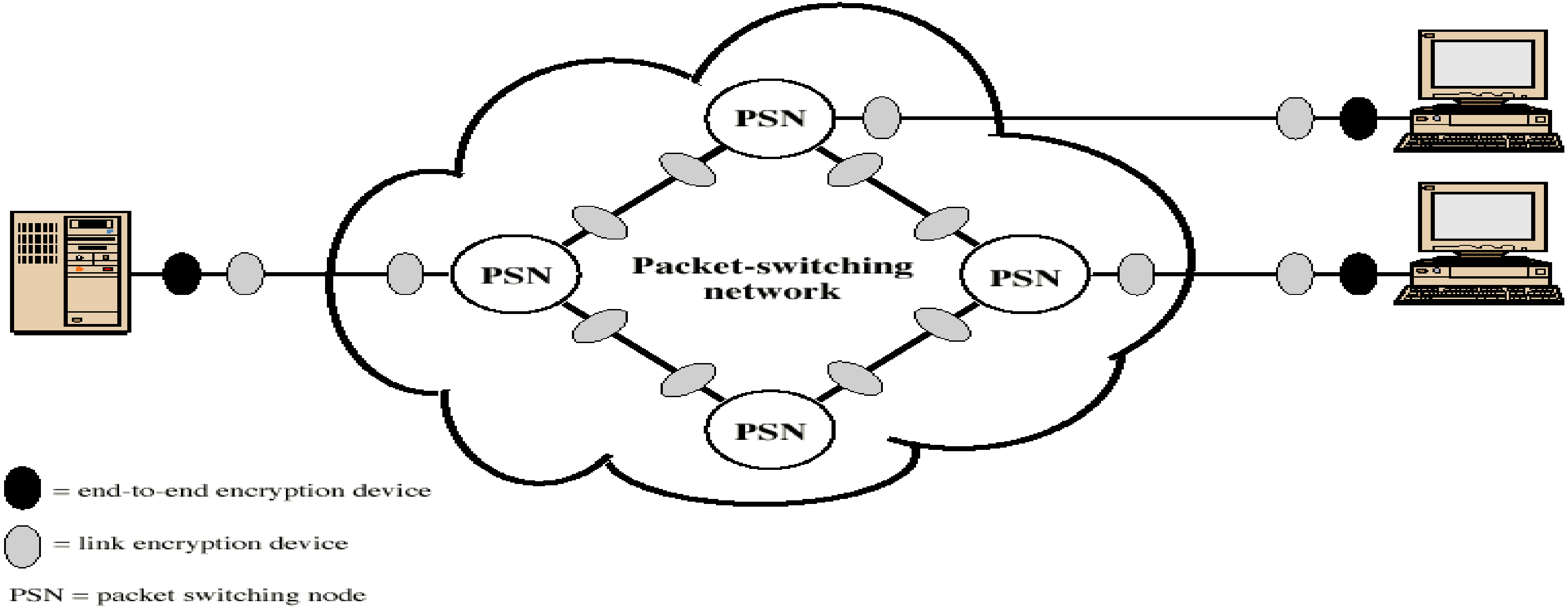
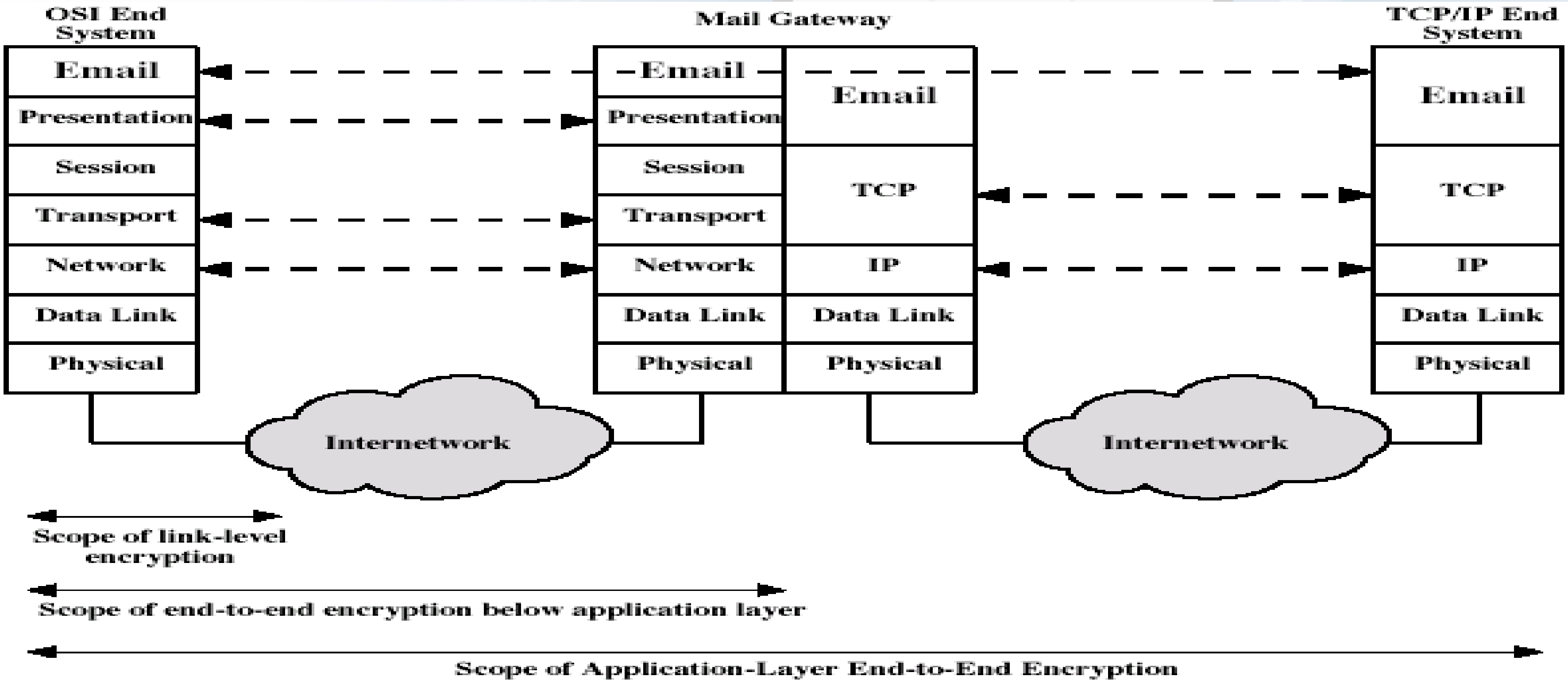


Figure 7.2 Encryption Across a Packet-Switching Network

Placement of Encryption

- Can place encryption function at various layers in OSI Reference Model.
  - Link encryption occurs at layers 1 or 2.
  - End-to-end can occur at layers 3, 4, 6, 7.
- If move encryption toward higher layer.
  - Less information is encrypted but is more secure.
  - Application layer encryption is more complex, with more entities and need more keys.

Scope of Encryption



Traffic Analysis

- Is monitoring of communications flows between parties?
  - Useful both in military & commercial spheres
  - Can also be used to create a covert channel
- Link encryption obscures header details
  - But overall traffic volumes in networks and at end-points is still visible.
- Traffic padding can further obscure flows.
  - but at cost of continuous traffic.

Traffic Analysis Cont..

- When using end-to-end encryption must leave headers in clear
  - So network can correctly route information.
- Hence although contents protected, traffic pattern flows are not.
- Ideally want both at once
  - End-to-end protects data contents over entire path and provides authentication.
  - Link protects traffic flows from monitoring.

Key Distribution and Management

- **Symmetric key cryptography:**

  Fast implementations, good for encrypting large amounts of data; requires shared secret key.

- **Asymmetric (public) key cryptography:**

  Inefficient for large data, good for authentication; no need to share a secret.

- How to share symmetric keys?
- How to distribute public keys?

Symmetric Key Distribution using Symmetric Encryption

Objective: Two entities share same secret key.

Principle: Change keys frequently.

- How to exchange a secret key?
  - A physically delivers key to B.
  - Third party, C, can physically deliver key to A and B.
  - If A and B already have a key, can securely transmit new key to each other, encrypted with old key.
  - If A and B have secure connection with third party C, C can securely send keys to A and B.

Options

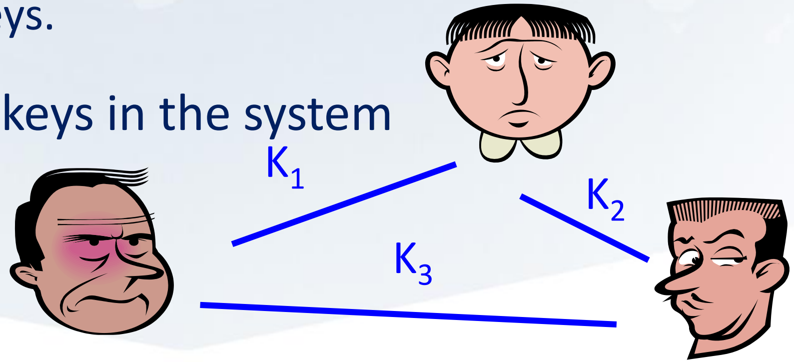
Option 1 and 2: manual delivery; feasible if number of entities is small (link encryption)

Option 3: requires initial distribution of key; discovery of initial key releases all subsequent keys.

Option 4: requires initial distribution of key with C; practical for large-scale systems (end-to-end encryption)

Symmetric Key Management

- Each pair of communicating entities needs a shared key
  - Why?
  - For a n-party system, there are $n(n-1)/2$ distinct keys in the system and each party needs to maintain n-1 distinct keys.
- How to reduce the number of shared keys in the system
  - Centralized key management
  - Public keys



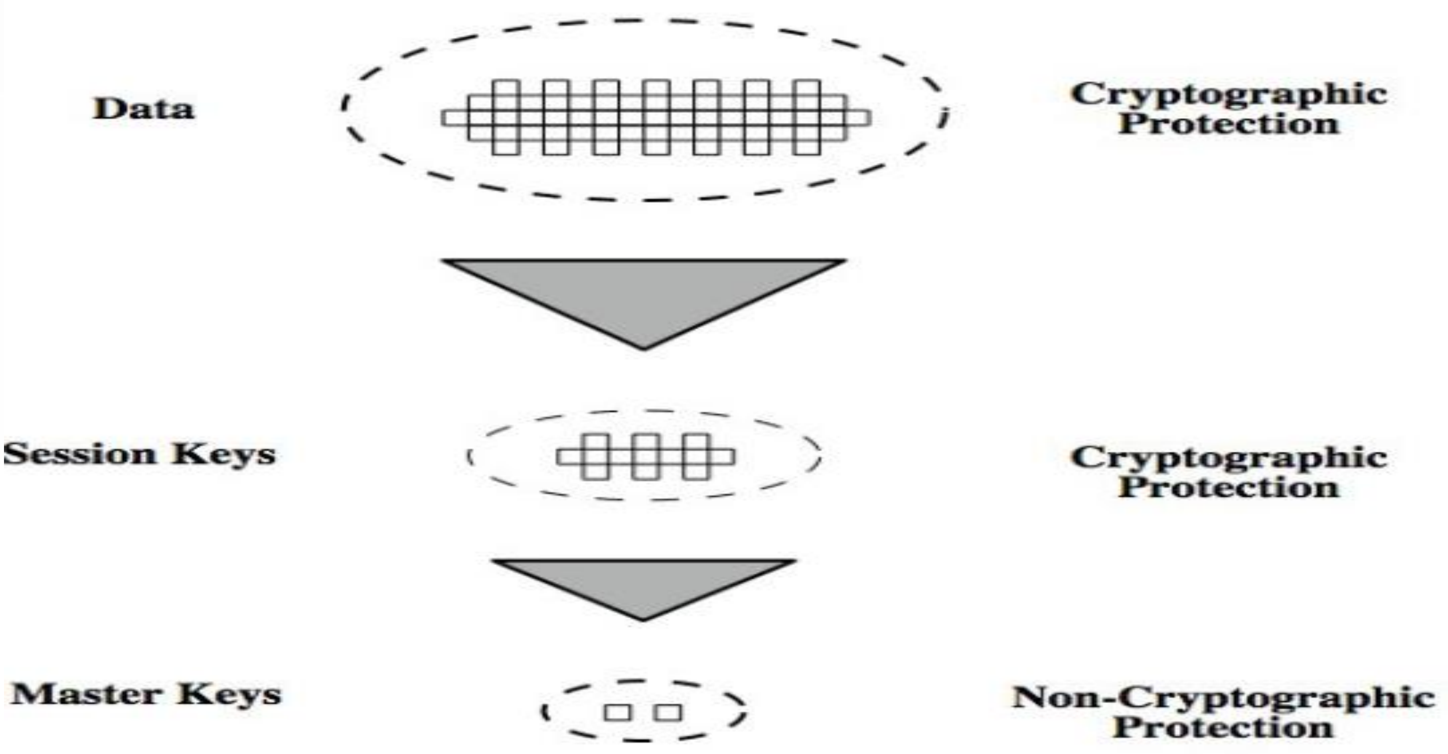
Using a Key Distribution Centre

- ❑Key Distribution Centre (KDC) is trusted third party
- ❑Hierarchy of keys used: Data sent between end-systems encrypted with temporary session key.
  - ✓It is used for the duration of a logical connection, such as a frame relay connection or transport connection, and then discarded.
  - ✓Session keys obtained from KDC over network; encrypted with master key.

Cont..

- For each end system or user, there is a unique master key that it shares with the key distribution center.
- If there are N entities that wish to communicate in pairs, then, as was mentioned, as many as $[N(N-1)]/2$ session keys are needed at any one time.
- However, only N master keys are required, one for each entity. Thus, master keys can be distributed using manual delivery.

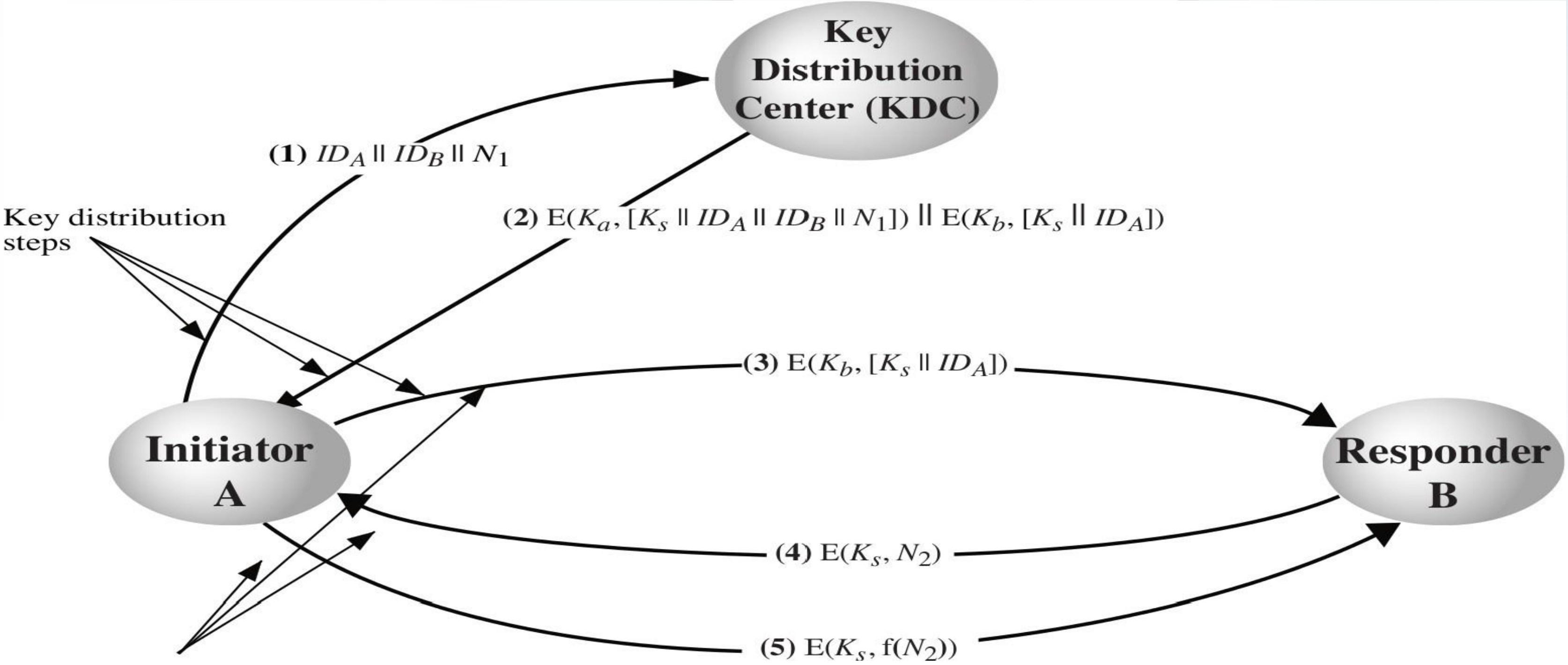
Use of a Key Hierarchy



KDC Scenario Notation

- End-systems: A and B, identified by ID_A and ID_B
- Master keys: K_a, K_b
- Session key (between A and B): K_s
- Nonce values: N_1, N_2
  - ✓E.g. timestamp, counter, random value
  - ✓Must be different for each request
  - ✓Must be difficult for attacker to guess

Key Distribution Scenario



Practical Considerations

Hierarchical Key Control

- Use multiple KDCs in a hierarchy.

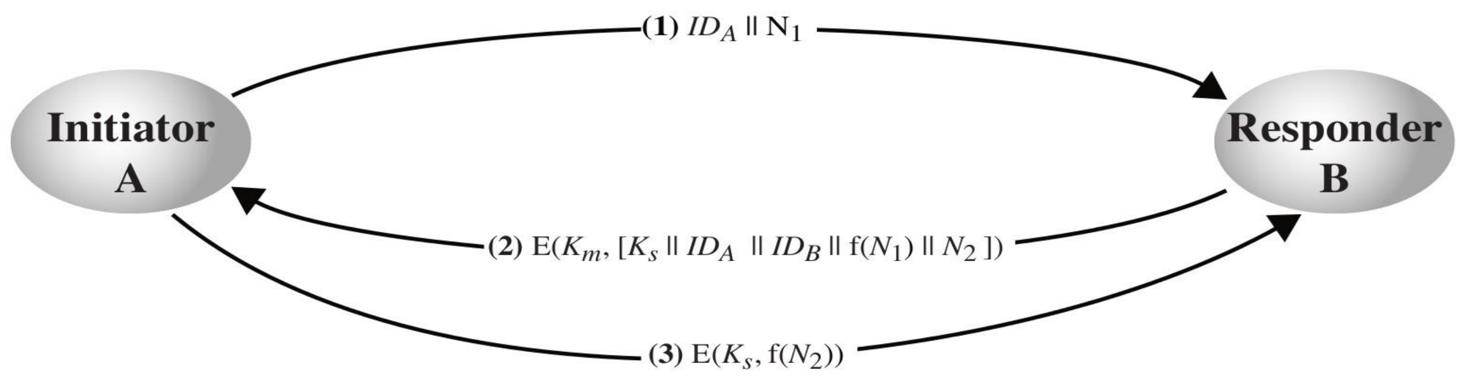
  E.g. KDC for each LAN (or building); central KDC to exchange keys between hosts in different LANs.
- Reduces effort in key distribution; limits damage if local KDC is compromised.

Session Key Lifetime

- Shorter lifetime is more secure; but increases overhead of exchanges.
- Connection-oriented protocols (e.g. TCP): new session key for each connection.
- Connection-less protocols (e.g. UDP/IP): change after fixed period or certain number of packets sent.

Decentralised Key Distribution

- ✓ Alternative that doesn't rely on KDC.
- ✓Each end-system must manually exchange $n - 1$ master keys (K_m) with others.



Automatic Key Distribution

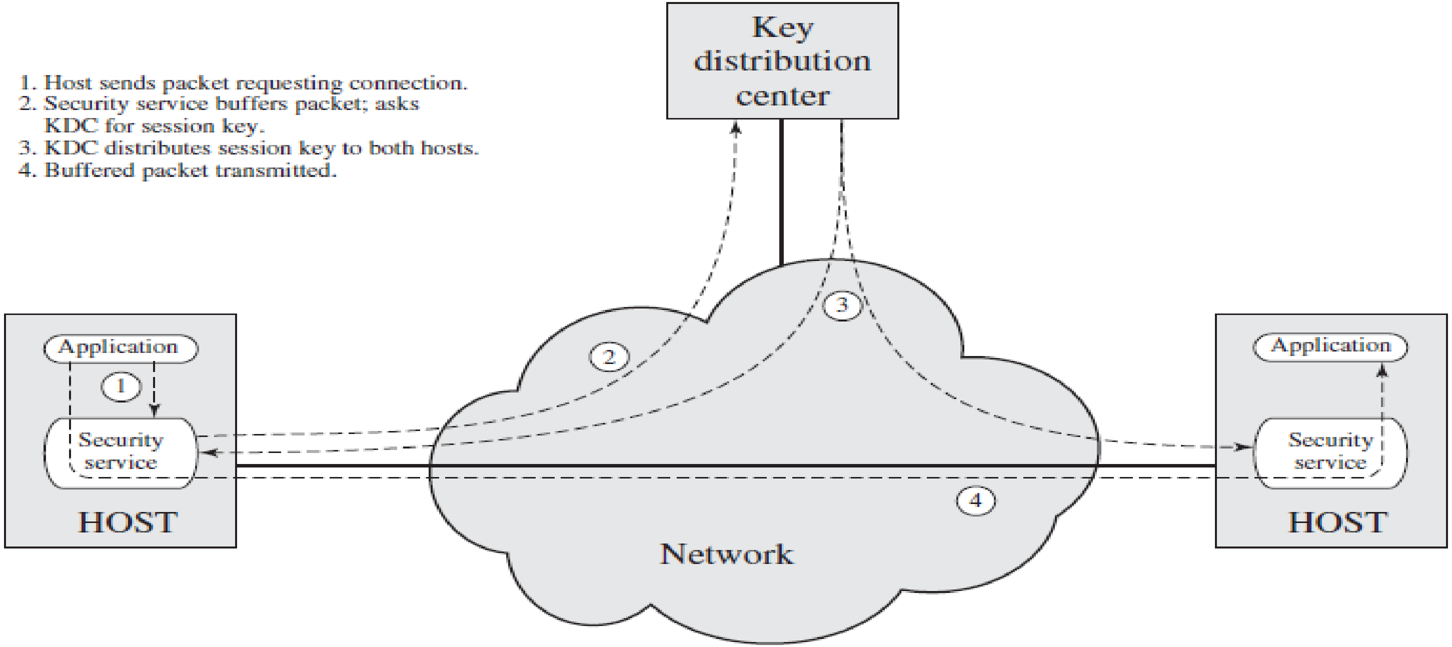


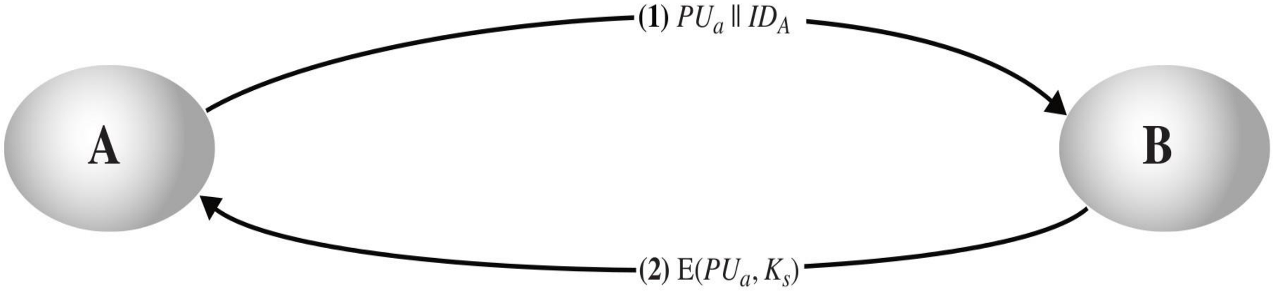
Figure 14.4 Automatic Key Distribution for Connection-Oriented Protocol

Symmetric Key Distribution using Asymmetric Encryption

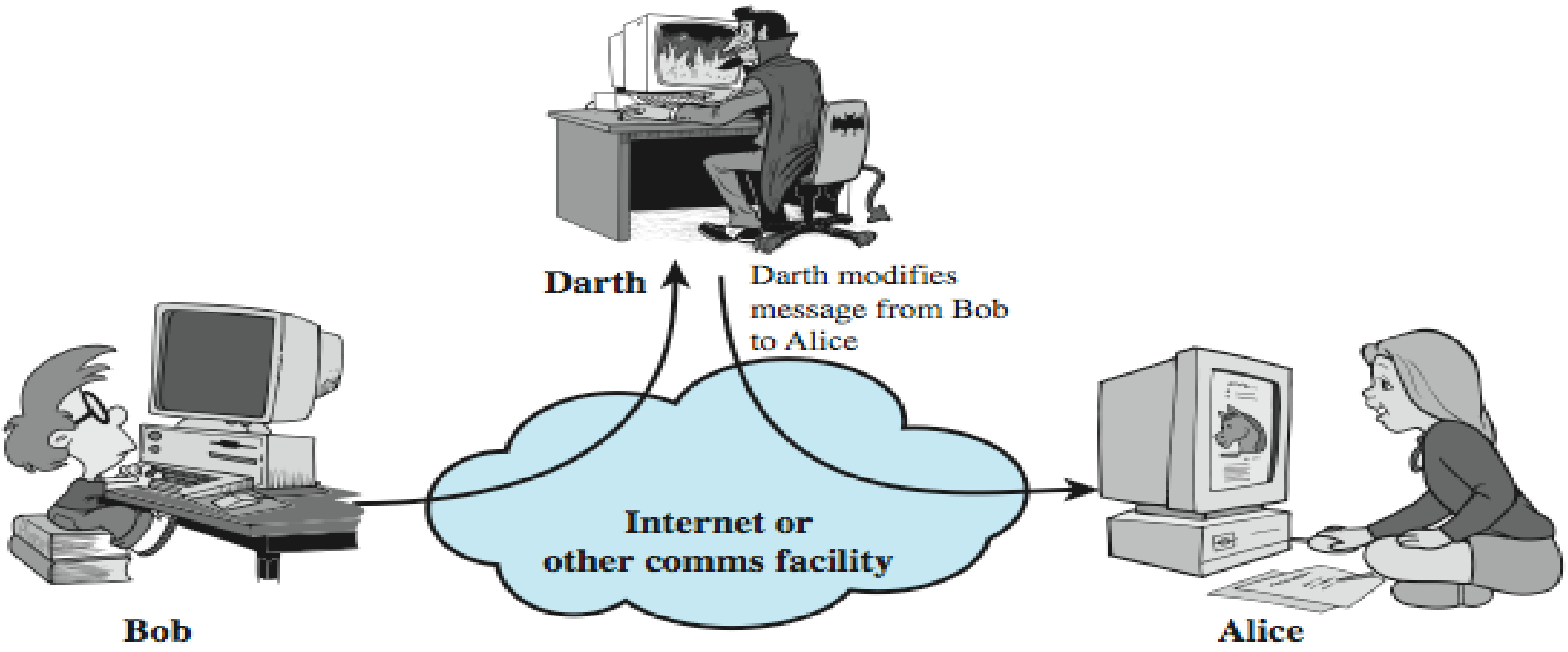
- Asymmetric encryption generally too slow for encrypting large amount of data.
- Common application of asymmetric encryption is exchanging secret keys.
- Three ways to exchange key:
  - Simple Secret Key Distribution
  - Secret Key Distribution with Confidentiality and Authentication
  - Hybrid Scheme: Public-Key Distribution of KDC Master Keys.

Simple Secret Key Distribution

- Simple: no keys prior to or after communication.
- Provides confidentiality for session key.
- Subject to man-in-the-middle attack.
- Only useful if attacker cannot modify/insert messages.



Man-in-the-Middle Attack

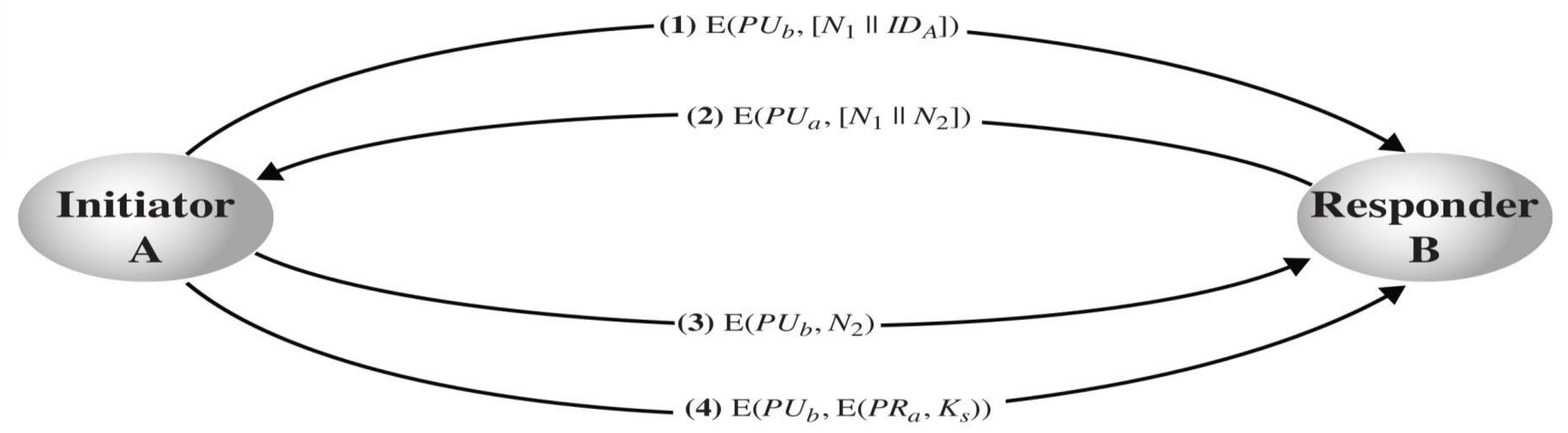


Cont..

- A generates a public/private key pair {*PUa*, *PRa*} and transmits a message intended for B consisting of *PUa* and an identifier of A, ID_A .
- E intercepts the message, creates its own public/private key pair {*PUe*, *PRe*} and transmits $PUe \| ID_A$ to B.
- B generates a secret key, *Ks*, and transmits E(*PUe*, *Ks*) .
- E intercepts the message and learns *Ks* by computing D(*PRe*, E(*PUe*, *Ks*)).
- E transmits E(*PUa*, *Ks*) to A.

Secret Key Distribution with Confidentiality and Authentication

- Provides both confidentiality and authentication in exchange of secret key.



Hybrid Scheme: Public-Key Distribution of KDC Master Keys

- Use public-key distribution to distribute master keys between end-systems and KDC.
- Efficient method of delivering master keys (rather than manual delivery).
- Useful for large networks, widely distributed set of users with single KDC.

Distribution of Public Keys

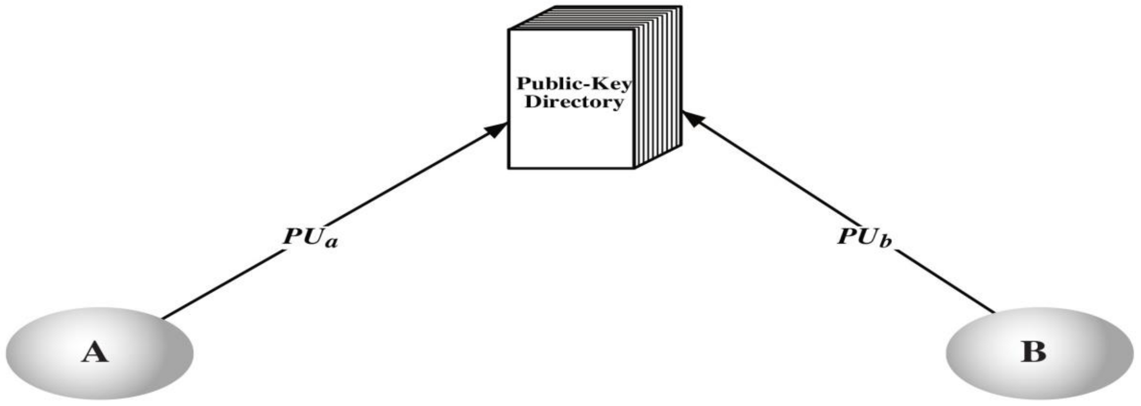
- By design, public keys are made public.
- Issue: how to ensure public key of *A* actually belongs to *A* (and not someone pretending to be *A*).
- Four approaches for distributing public keys
  - ✓ Public announcement
  - ✓ Publicly available directory
  - ✓ Public-key authority
  - ✓ Public-key certificates

Public Announcements

- Make public key available in open forum: newspaper, email signature, website, conference, . . .
- Problem: anyone can announce a key pretending to be another user.

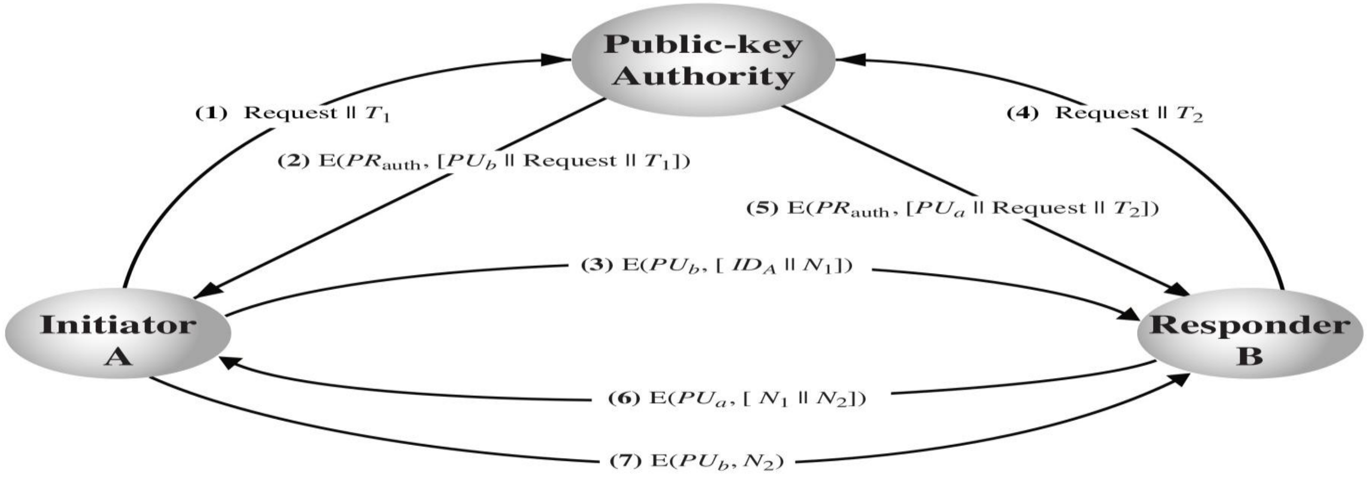
Publicly Available Directory

- All users publish keys in central directory.
- Users must provide identification when publishing key.
- Users can access directory electronically.
- Weakness: directory must be secure.



Public-Key Authority

- Specific instance of using publicly available directory.
- Assume each user has already securely published public-key at authority; each user knows authorities public key.

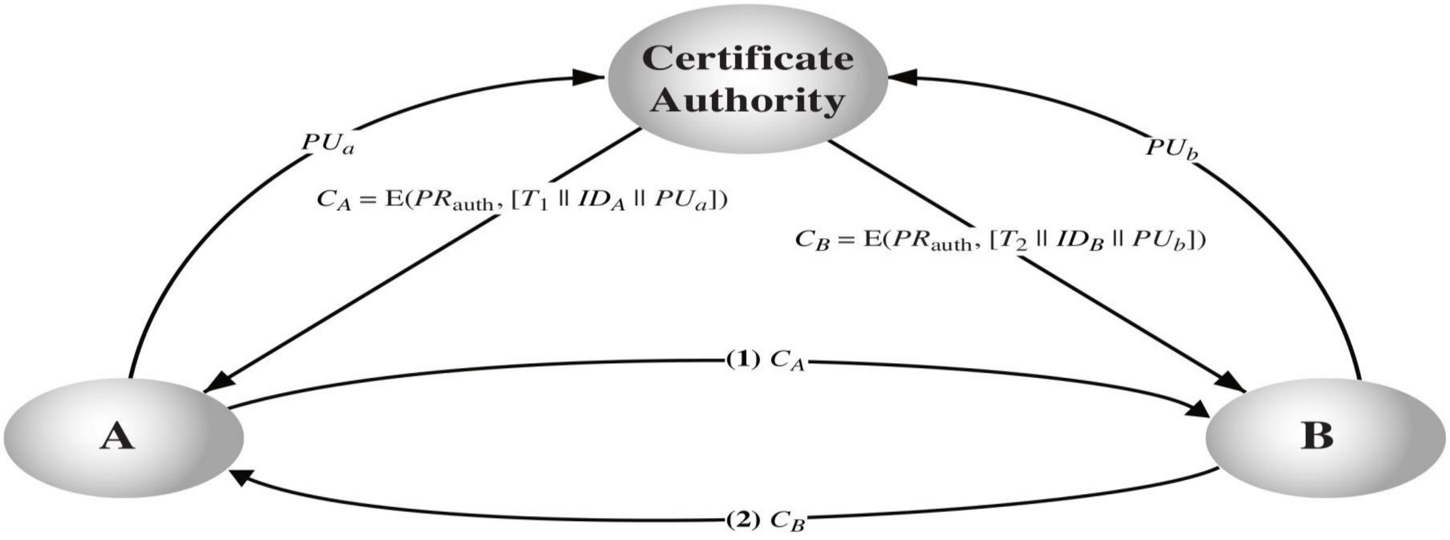


Cont..

- First 5 messages are for key exchange; last 2 are authentication of users.
- Although 7 messages, public keys obtained from authority can be cached.
- Problem: authority can be bottleneck.
- Alternative: public-key certificates.

Public-Key Certificates

- Assume public keys sent to CA can be authenticated by C_A; each user has certificate of C_A.



Cont..

- A certificate is the ID and public-key of a user signed by CA

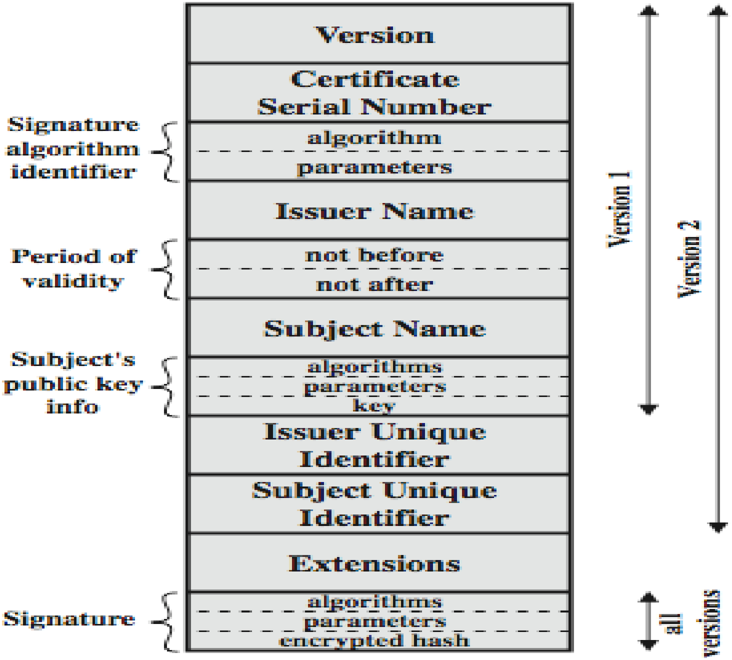
$$C_A = E(PR_{auth}, [T \,||\, ID_A \,||\, PU_a])$$

- Timestamp *T* validates the certificate (expiration date).
- Common format for certificates is X.509 standard (by ITU)
  - S/MIME (secure email)
  - IP security (network layer security)
  - SSL/TLS (transport layer security)
  - SET (e-commerce)

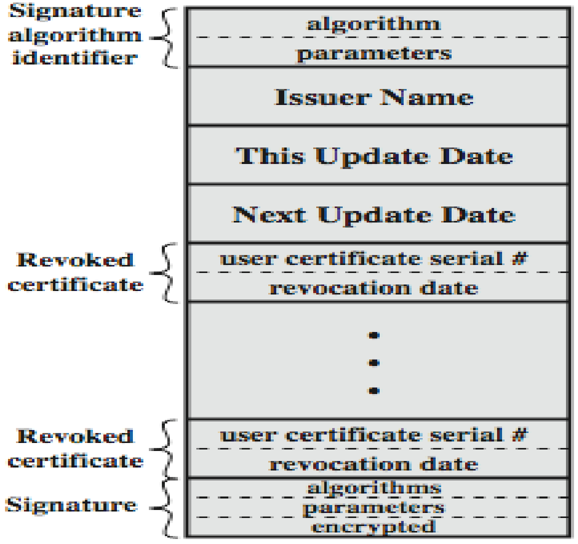
X.509 Certificate

- Each user has a certificate, although it is created by the Certificate Authority (CA).
- Certificates are stored in a public directory.
- Certificate format includes:

X.509 Formats

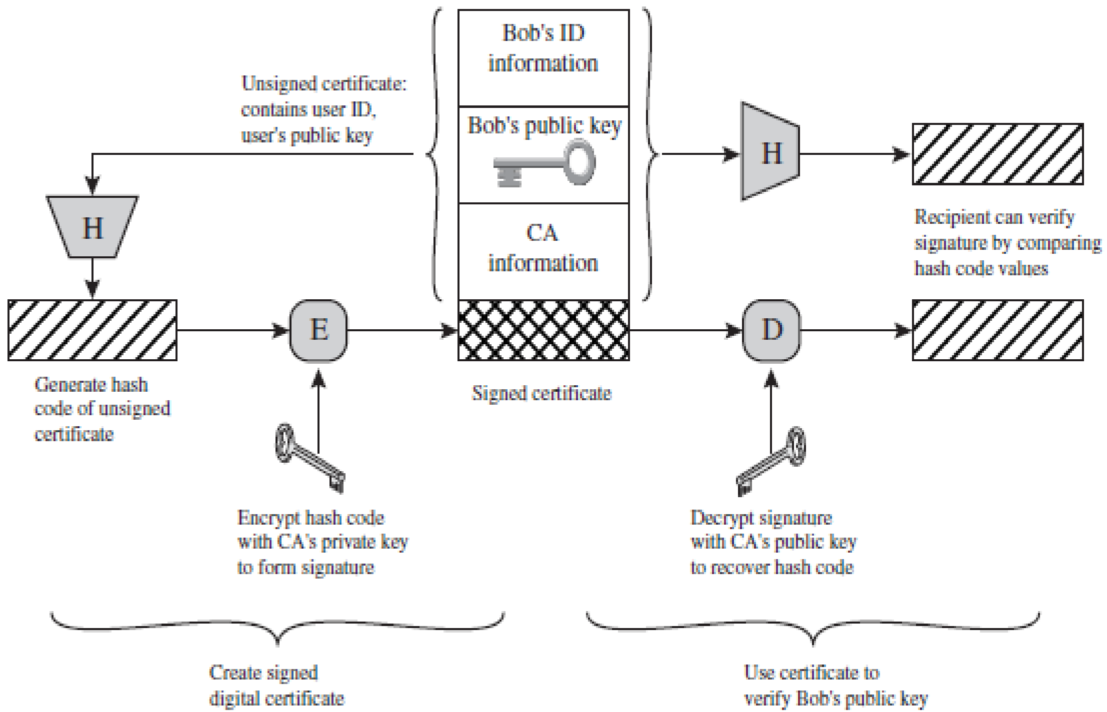


(a) X.509 Certificate



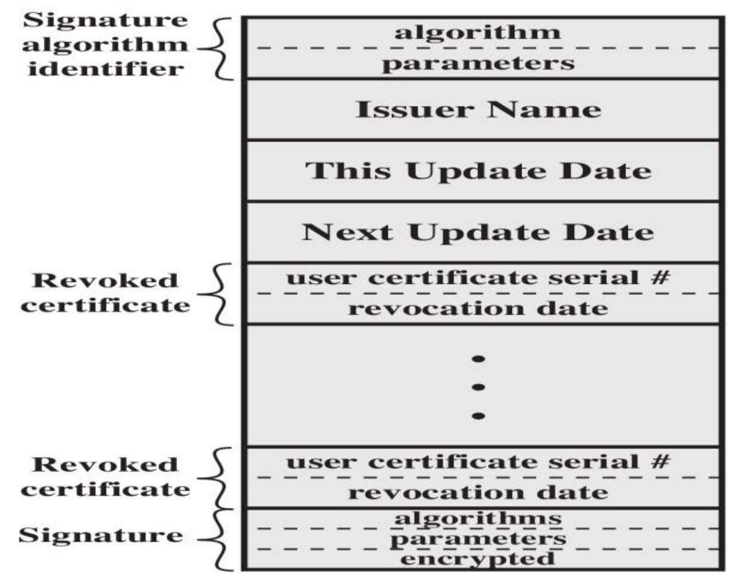
(b) Certificate Revocation List

Public-Key Certificate Use



Certificate Revocation List

- Certificates may be revoked before expiry.



Multiple Certificate Authorities

- Multiple CA's can be arranged in hierarchy.
- Notation: *Y* << *X* >> certificate of X issued by CA Y.
- A acquires B certificate using chain: X<<W>>W<<V>>V<<Y>>Y<<Z>>Z<<B>>

X.509 Hierarchy