Unit III. Asymmetric Ciphers

Key Management and Distribution

Er. Kobid Karkee
Himalaya College of Engineering

Key Management

- Public-key encryption helps address key distribution problems in symmetric encryption techniques.
- Key management have two aspects:
  - distribution of public keys
  - use of public-key encryption to distribute secret keys
- Key Distribution - Distributing the keys over communicating parties.
- Several techniques have been proposed for the distribution of public keys. Virtually all these proposals can be grouped into the following general schemes:
  - Public announcement
  - Publicly available directory
  - Public-key authority
  - Public-key certificates

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

Figure 14.9 Uncontrolled Public-Key Distribution

Publicly Available Directory

- A greater degree of security can be achieved by maintaining a publicly available dynamic directory of public keys.
- Maintenance and distribution of the public directory would have to be the responsibility of some trusted entity or organization.
- Directory must be trusted with following 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
- Directories are accessed electronically and still vulnerable to tampering or forgery.

Publicly Available Directory

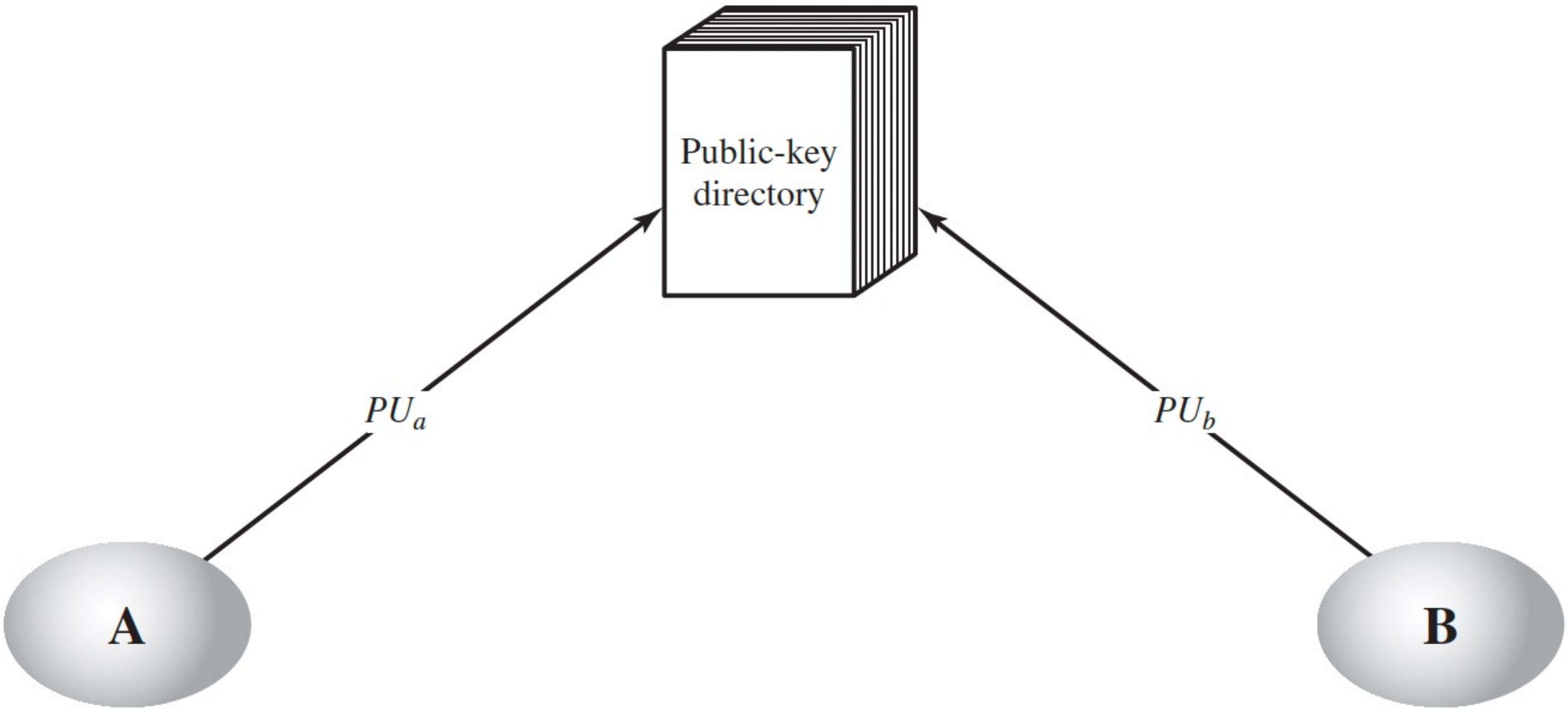

Figure 14.10 Public-Key Publication

Public-Key Authority

- Stronger security for public-key distribution can be achieved by providing tighter control over the distribution of public keys from the directory.
- As before, the scenario assumes that a central authority maintains a dynamic directory of public keys of all participants.
- In addition, each participant reliably knows a public key for the authority, with only the authority knowing the corresponding private key.
- However, this isn't perfect as the public-key authority could be somewhat of a bottleneck in the system.
- The reason for this is that a user must appeal to the authority for a public key for every other user that it wishes to contact.
- Also, the directory of names and public keys maintained by the authority is vulnerable to tampering.

Public-Key Authority

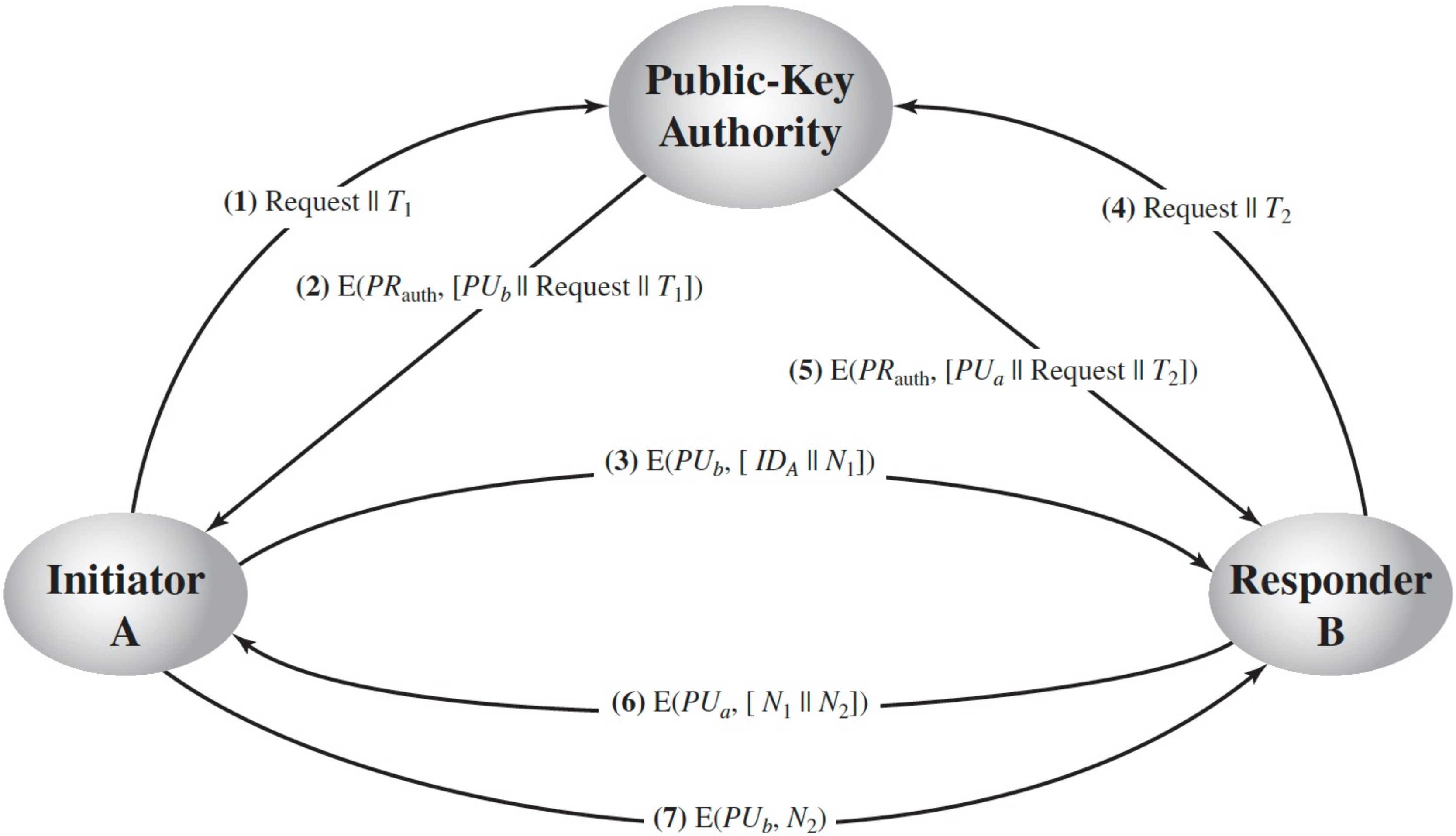

Figure 14.11 Public-Key Distribution Scenario

Public-Key Certificates

- In this scheme, authority provides a certificate (which binds an identity to the public key) to allow key exchange without real-time access to the public authority each time.
- The certificate is accompanied by some other info such as period of validity, rights of use, etc.
- All of this content is signed by the private key of the certificate authority, and it can be verified by anyone possessing the authority's public key.
- First sender and receiver both request CA for a certificate which contains a public key and other information and then they can exchange these certificates and can start communication.

Public-Key Certificates

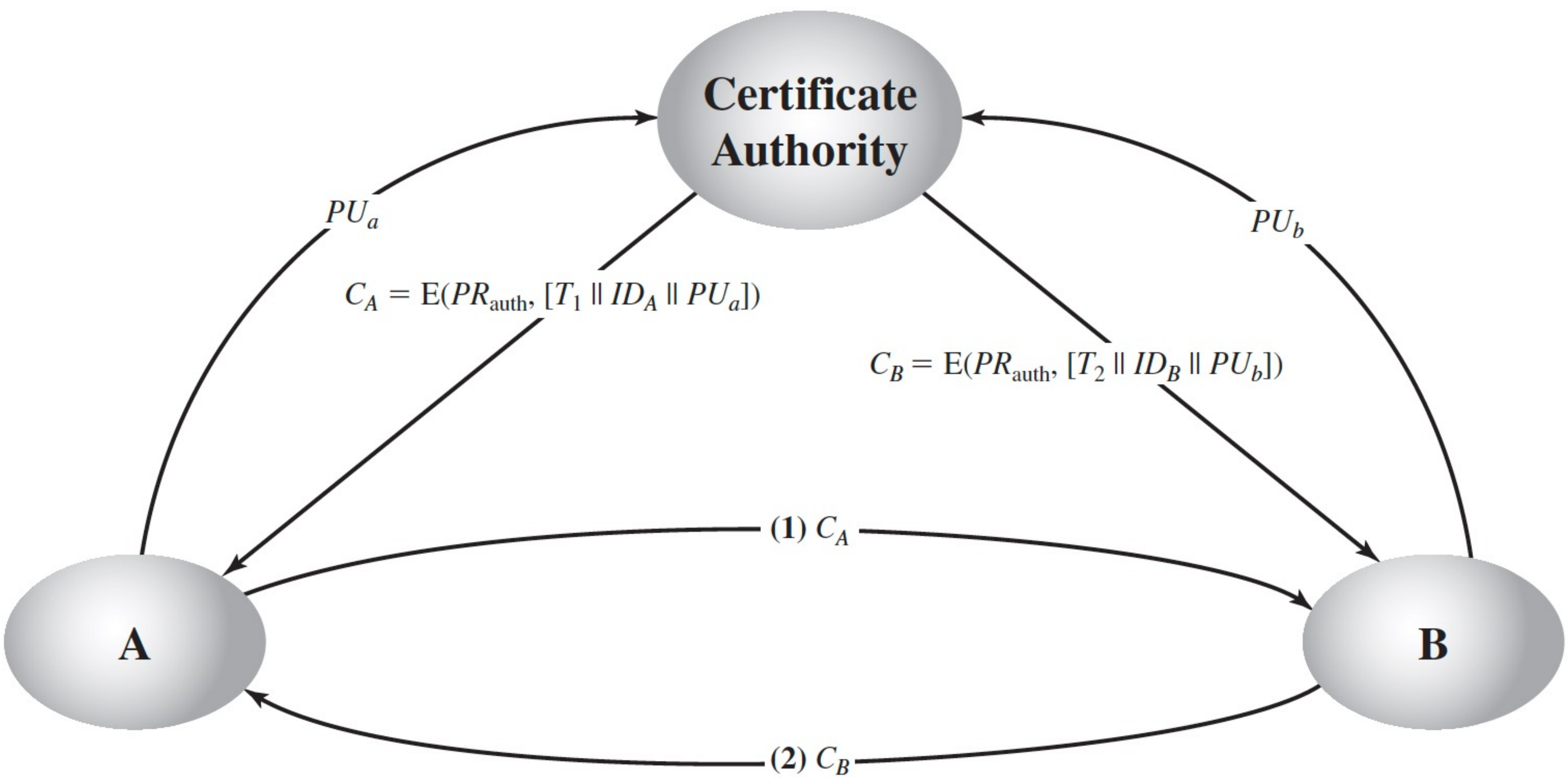

Figure 14.12 Exchange of Public-Key Certificates

Public-Key Distribution of Secret Keys

- Once public keys have been distributed or have become accessible, secure communication that thwarts eavesdropping, tampering , or both is possible.
- However, few users will wish to make exclusive use of public-key encryption for communications because of the relatively slow data rates that can be achieved.
- Accordingly, public-key encryption is more reasonably viewed as a vehicle for the distribution of secret keys to be used for conventional encryption.
- Two protocols for public-key distribution of secret keys are discussed here:

1. Simple Secret-Key Distribution

- An extremely simple scheme put forward by Ralph Merkle in 1979.
- If A wishes to communicate with B, the following procedure is employed:
  1. A generates a public/private key pair $\{KU_A, KR_A\}$ and transmits a message to B consisting of KU_A, and an identifier of A, ID_A.
  2. B generates a secret key, K_s, and transmits it to A, encrypted with A's public key.
  3. A computes $D_{KR_A}[E_{KU_A}[K_s]]$ to recover the secret key. Since only A can decrypt the message, only A and B will know the identity of K_s.
  4. A discards KU_A and KR_A and B discards KU_A.

1. Simple Secret-Key Distribution

- Despite its simplicity, this is an attractive protocol.
- No keys exist before the start of the communication, and none exist after the completion of communication. Thus, the risk of compromise of the keys is minimal.
- At the same time, the communication is secure from eavesdropping.

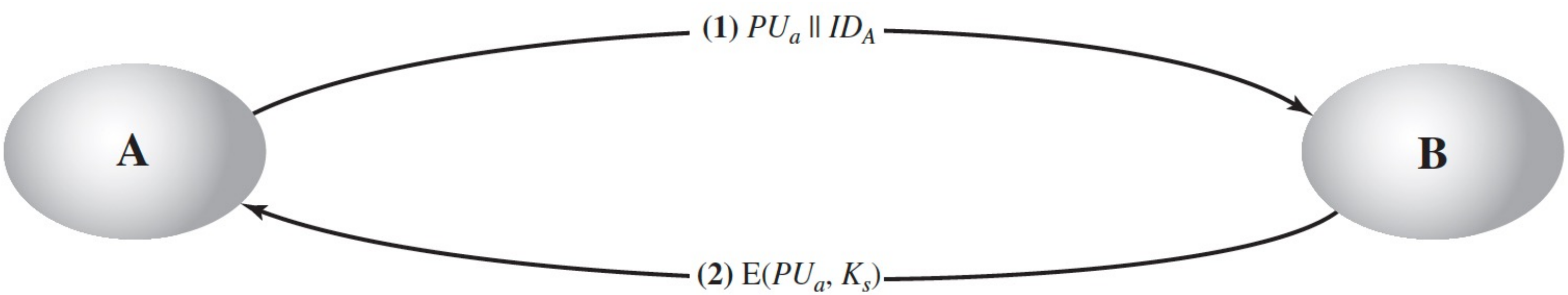

Figure 14.7 Simple Use of Public-Key Encryption to Establish a Session Key

1. Simple Secret-Key Distribution

- **Man-in-the Middle Attack on Simple Secret-Key Distribution:**

1. A generates a public/private key pair $\{KU_A, KR_A\}$ and transmits a message intended for B consisting of KUA and identifier of A, ID_A.
2. E intercepts the message, creates its own public/private key pair $\{KU_E, KR_E\}$ and transmits $KU_E || ID_A$ to B.
3. B generates a secret key, Ks, and transmits $E_{KU_E}[K_s]$.
4. E intercepts the message and learns K_s by computing $D_{KR_E}[E_{KU_E}[K_s]]$.
5. E transmits $E_{KU_A}[K_s]$ to A.

1. Simple Secret-Key Distribution

- The result is that both A and B know K_s and are unaware that K_s has also been revealed to E.
- A and B can now exchange messages using K_s.
- E no longer actively interferes with the communications channel but simply eavesdrops.
- Knowing, K_s, E can decrypt all messages, and both A and B are unaware of the problem.
- Thus, this simple protocol is only useful in an environment where the only threat is eavesdropping.
- The protocol is insecure against an adversary who can intercept messages and then either relay the intercepted message or substitute another message.

Secret Key Distribution with Confidentiality and Authentication

- An approach as in following figure, provides protection against both active and passive attacks.
- We begin at a point when it is assumed that A and B have exchanged public keys by one of the schemes described earlier.

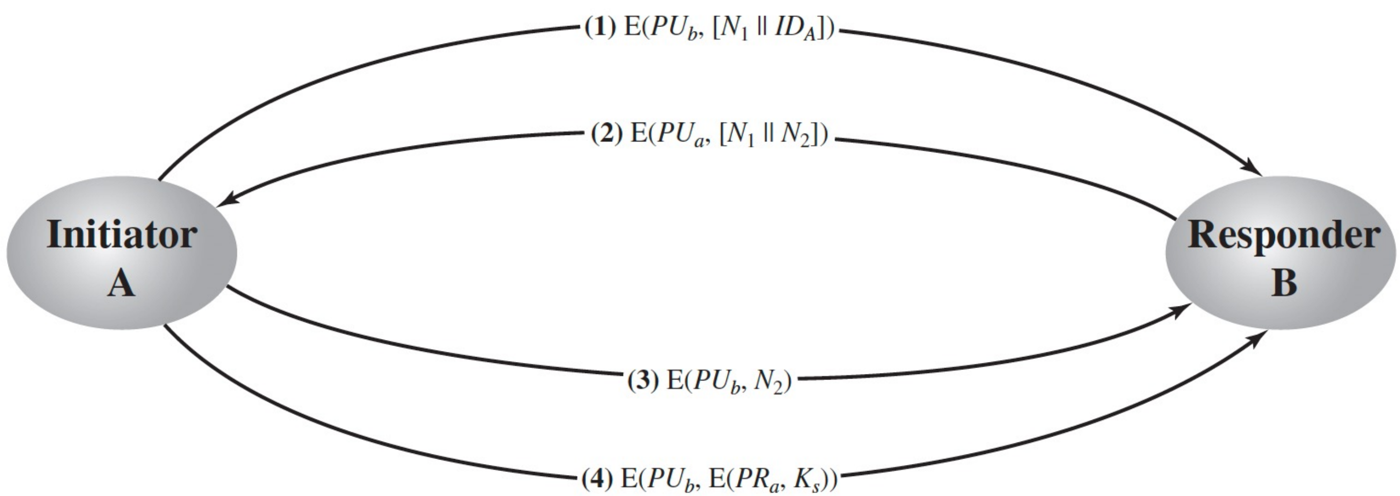

Figure 14.8 Public-Key Distribution of Secret Keys

- Then the following steps occur:

Secret Key Distribution with Confidentiality and Authentication

1. A uses B's public key to encrypt a message to B containing an identifier of A (ID_A) and a nonce (N_1), which is used to uniquely identify this transaction.
2. B sends a message to A encrypted with KU_A and containing A's nonce (N_1) as well as a new nonce generated by B (N_2).

   Since only B could have decrypted message (1), the presence of N_1 in message (2) assures A that the correspondent is B.
3. A returns N_2, encrypted using B's public key, to assure B that its correspondent is A.

Note: Nonce means number (usually random) used only once.

Secret Key Distribution with Confidentiality and Authentication

4. A selects a secret key Ks and sends $M = E_{KU_B}[E_{KR_A}[K_s]]$ to B.

   Encryption of this message with B's public key ensures that only B can read it, encryption with A's private key ensures that only A could have sent it.

5. B computes $D_{KU_A}[E_{KR_b}[M]]$ to recover the secret key.

This scheme ensures both confidentiality and authentication in the exchange of a secret key