Key Distribution and Agreement Schemes

CSG 252 Fall 2006

Riccardo Pucella

Key Establishment Problem

- PK cryptosystems have advantages over SK cryptosystems
  - PKCs do not need a secure channel to establish secret keys
  - However, PKCs generally less efficient than SKCs
  - So you often want SKCs anyways
- The problem: n agents on an insecure network
  - Want to establish keys between pairs of agents to communicate securely

Distribution vs Agreement

- Secret Key Distribution Scheme (SKDS):
  - Assume a special entity in the network, a Trusted Authority (TA)
  - TA chooses a secret key for communicating, and transmits it to parties that wants to communicate
- Key Agreement Scheme (KAS):
  - Two or more parties want to establish a secret key on their own

Main Goal of Schemes

- At the end of an exchange:
  - Two parties share a key K
  - The value of K is not known to any other party
    - Except maybe the TA
- Sometimes want more: mutual identification (chap. 9)
  - No honest participant in a session of the scheme will accept after any interaction in which an adversary is active

Long-Lived vs Session Keys

- LL keys:
  - Long-lived keys, usually shared between TA and users
- Session keys:
  - Used for a session-based communication
- Why the distinction?
  - Limit amount of ciphertext available to an attacker
  - Limit exposure in event of key compromise
    - Assuming session keys do not reveal info about LL keys or other session keys

Attacker Models

- May or may not be a user in the system
  - insider vs outsider attacker
- May be passive or active
  - Alter messages in transit (including intercepting)
  - Save messages for later reuse
  - Attempt to masquerade as other users

Possible Attacker Objectives

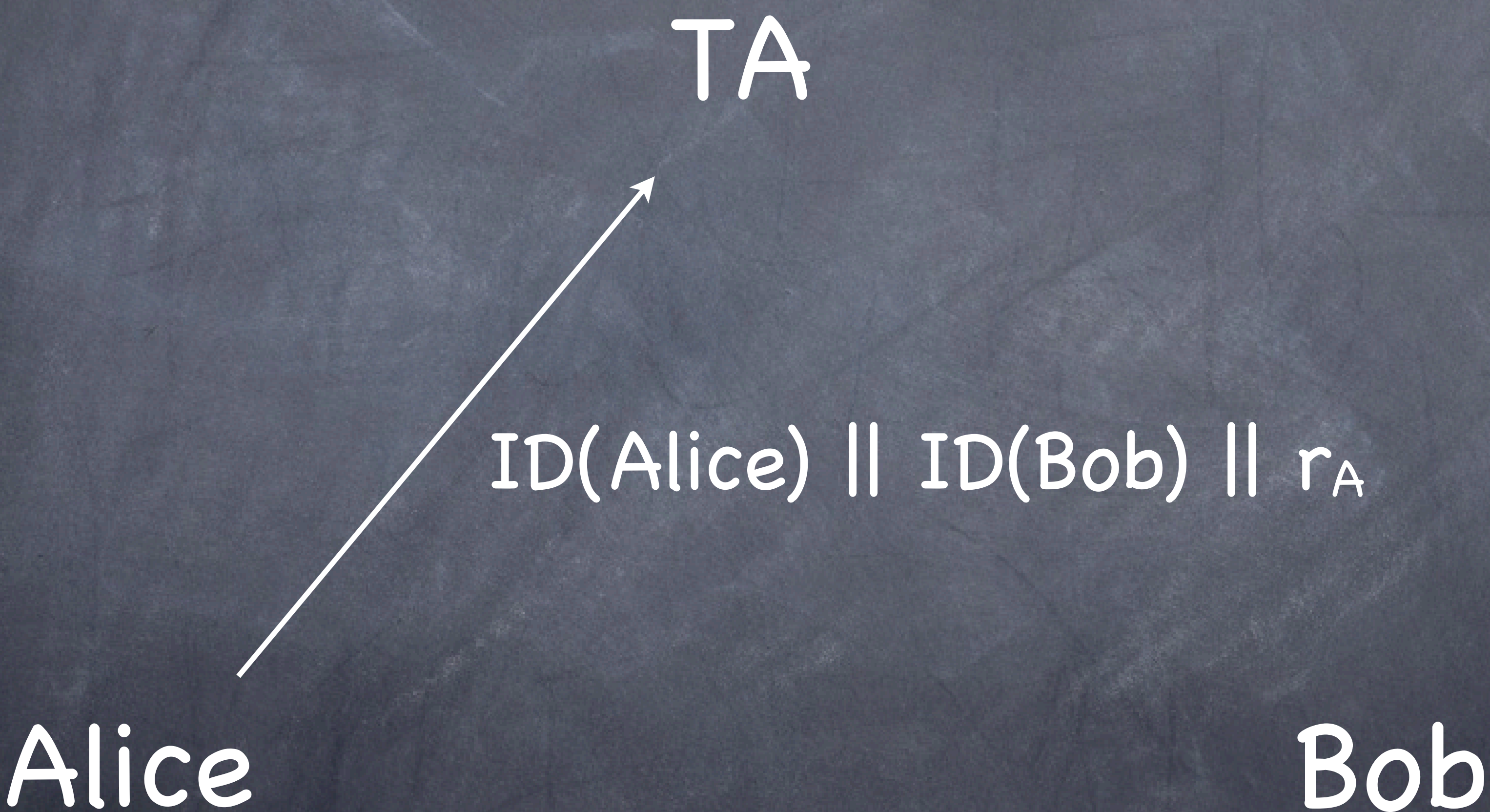
- Passive objectives:
  - Determine some (partial) information about key exchanged by users
- Active objectives:
  - Fool U and V into accepting an "invalid" key
    - E.g. an old expired key, or a key known to adv
  - Make U and V believe they have exchanged a key with each other when that is not the case

Extended Attacker Models

- Known session key attack:
  - Attacker learns session keys, want other session keys (as well as LL keys) to remain secret
- Known LL key attack:
  - Attacker learns LL keys of a participant, want previous session keys to remain secret
  - Perfect forward secrecy
    - This is not a property of a cryptosystem, but of how a cryptosystem is used!

Key Distribution Scheme:

Needham-Schroeder Scheme



Key Distribution Scheme:

Needham-Schroeder Scheme

TA

$t_{Bob} = e_{KBob}(K \parallel ID(Alice))$

$e_{KAlice}(r_A \parallel ID(Bob) \parallel K \parallel t_{Bob})$

Alice

Bob

Key Distribution Scheme:

Needham-Schroeder Scheme

TA

t_{Bob}

Alice → Bob

Key Distribution Scheme:

Needham-Schroeder Scheme

TA

$e_K\ (r_B)$

Alice ← Bob

Key Distribution Scheme:

Needham-Schroeder Scheme

TA

$e_K\ (r_B - 1)$

Alice → Bob

Denning-Sacco Attack on NSS

- Known session key attack
  - Suppose Oscar eavesdropped on the messages exchanges in an old session between Alice and Bob (which used key K)

- Oscar sends intercepted ticket t_{Bob} to Bob
- Bob replies with $e_K(r_B)$ for some random r_B
- Oscar can decrypt and send back $e_K(r_B-1)$

- Key K is not (necessarily) known to Bob's intended recipient Alice
- Key K is know to Oscar

Key Distribution Scheme:

Bellare-Rogaway Scheme

TA

ID(Alice) || ID(Bob) || r_A

Alice ⟶ Bob

Key Distribution Scheme:

Bellare-Rogaway Scheme

TA

$ID(Alice) \,||\, ID(Bob) \,||\, r_A \,||\, r_B$

Alice

Bob

Key Distribution Scheme:

Bellare-Rogaway Scheme

TA

$e_{K_{Alice}}(K) \,||$
$MAC_{Alice}(ID(Bob) \,||\, ID(Alice) \,||$
$r_A \,||\, e_{K_{Alice}}(K))$

Alice

Bob

Key Distribution Scheme:

Bellare-Rogaway Scheme

TA

e_{KBob} (K) ||
MAC_{Bob} (ID(Alice) || ID(Bob) ||
r_B || e_{KBob} (K))

TA → Bob

Alice

Bob

Key Agreement Scheme:

Diffie-Hellman Scheme

G a group and $\alpha \in G$ of order n

$$\text{Alice} \xrightarrow{\alpha^a} \text{Bob}$$

a — b

Key Agreement Scheme:

Diffie-Hellman Scheme

G a group and $\alpha \in G$ of order n

α^b

Alice ← Bob

a

b

α^a

Key Agreement Scheme:

Diffie-Hellman Scheme

G a group and $\alpha \in G$ of order n

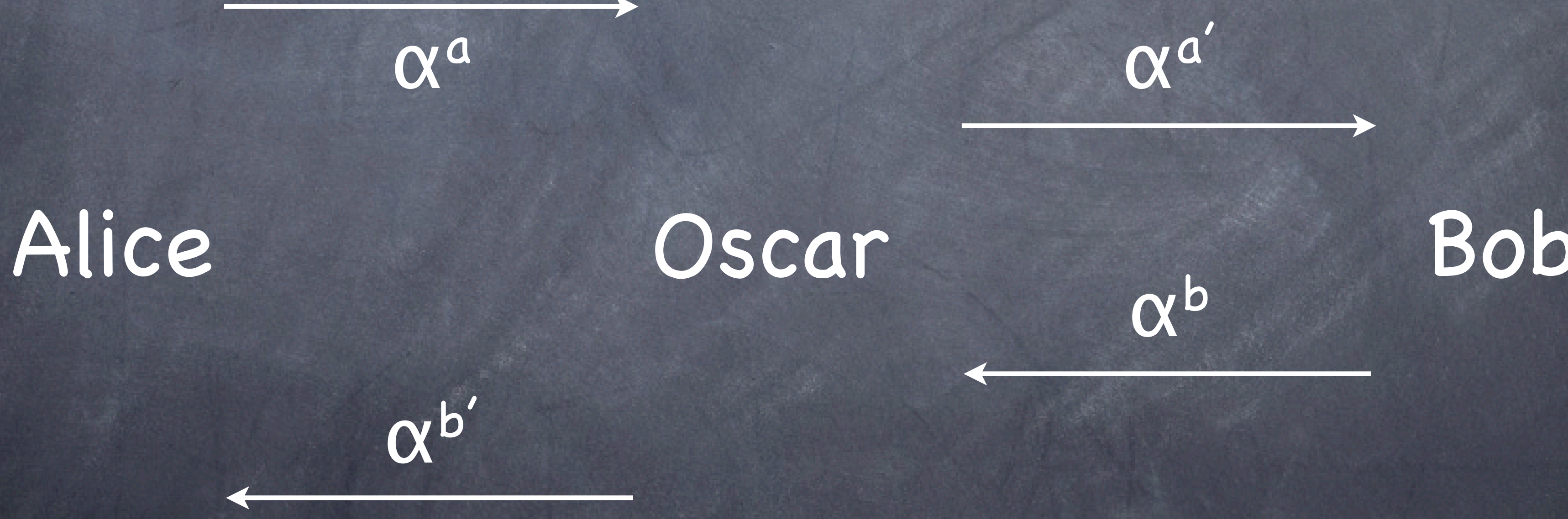
Alice		Bob
a		b
α^b		α^a
$K = (\alpha^b)^a$	$=$	$K = (\alpha^a)^b$

Computational Diffie-Hellman Problem

- For the previous scheme to be secure, need for the group G and α to be such that:
  - Given α^a and α^b, it is hard to find α^{ab}
- Can show (6.7.3) that if you can solve the CDH problem, then you can solve the discrete log problem in G

Man-in-the-Middle Attack on DHS

- Oscar sits between Alice and Bob and substitutes his own messages



Key Agreement Scheme:

Station-to-Station Scheme

G a group and $\alpha \in G$ of order n

$Cert(U) = (ID(U), ver_U, sig_{TA}(ID(U), ver_U))$

α^a, Cert(Alice)

Alice ⟶ Bob

a b

Key Agreement Scheme:

Station-to-Station Scheme

G a group and $\alpha \in G$ of order n

$Cert(U) = (ID(U), ver_U, sig_{TA}(ID(U), ver_U))$

α^b, $sig_{Bob}(ID(Alice)||\alpha^a||\alpha^b)$, Cert(Bob)

Alice ⟵ Bob

a   b

α^a

Key Agreement Scheme:

Station-to-Station Scheme

G a group and $\alpha \in G$ of order n

$Cert(U) = (ID(U), ver_U, sig_{TA}(ID(U), ver_U))$

$sig_{Alice}(ID(Bob)||\alpha^a||\alpha^b)$

Alice ⟶ Bob

a b

α^b α^a

Key Agreement Scheme:

Station-to-Station Scheme

G a group and $\alpha \in G$ of order n

$Cert(U) = (ID(U), ver_U, sig_{TA}(ID(U), ver_U))$

Alice		Bob
a		b
α^b		α^a
$K = (\alpha^b)^a$	$=$	$K = (\alpha^a)^b$

Other Schemes

- Other schemes are modifications of DH-style schemes to reduce computation, or the amount of data the needs to be exchanged

- MTI Schemes
  - Does not require users to sign messages
  - Put α^a in certificates
- Girault Scheme
  - Does not require certificates
  - Need to go through a TA
- Encrypted Key Exchange
  - Encrypt DHS exponents using a shared key