Key Agreement Schemes

CSG 252 Lecture 9

November 25, 2008

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

Last time: Secret Key Distribution Scheme (SKDS):

- Assume a Trusted Authority (TA) - a server
- 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
- Uses public key cryptography

Main Goal of Key Agreement

At the end of an exchange:

- Two parties share a key K
- The value of K is not known to any other party
  - Secrecy

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
- Also called mutual authentication

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 private keys) to remain secret

Known private key attack:

- Attacker learns private key 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!

Diffie-Hellman Scheme

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

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

a b

Diffie-Hellman Scheme

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

$$\begin{tikzcd}
\text{Alice} \arrow[rr, "\alpha^a"] & & \text{Bob} \\
a & & b \arrow[ll, "\alpha^b"'] \\
 & & \alpha^a
\end{tikzcd}$$

Diffie-Hellman Scheme

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

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

$$\xleftarrow{\alpha^b}$$

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

Alice Oscar Bob

Man-in-the-Middle Attack on DHS

Oscar sits between Alice and Bob and substitutes his own messages

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

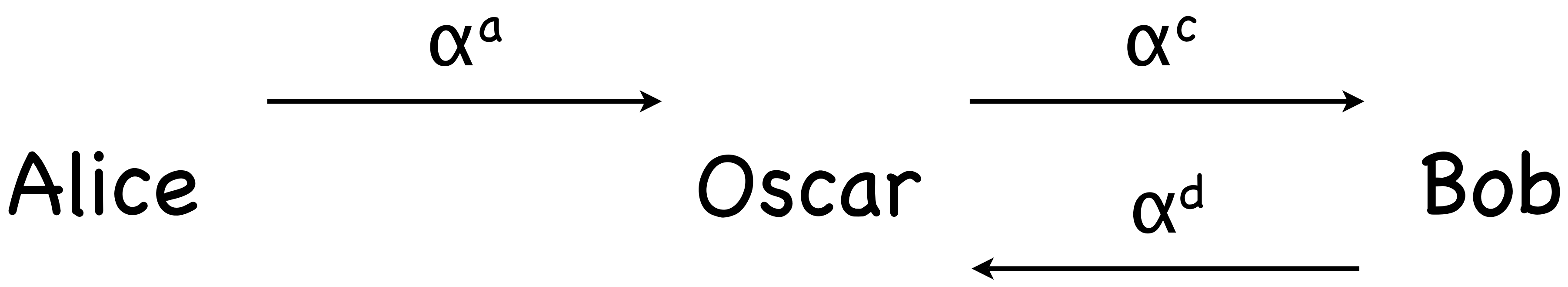
Man-in-the-Middle Attack on DHS

Oscar sits between Alice and Bob and substitutes his own messages

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

Man-in-the-Middle Attack on DHS

Oscar sits between Alice and Bob and substitutes his own messages



Man-in-the-Middle Attack on DHS

Oscar sits between Alice and Bob and substitutes his own messages

$$\begin{tikzcd}
\text{Alice} \arrow[r, shift left, "\alpha^a"] & \text{Oscar} \arrow[l, shift left, "\alpha^b"] \arrow[r, shift left, "\alpha^c"] & \text{Bob} \arrow[l, shift left, "\alpha^d"]
\end{tikzcd}$$

Man-in-the-Middle Attack on DHS

Oscar sits between Alice and Bob and substitutes his own messages

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

$$\text{Alice} \xleftarrow{\alpha^b} \text{Oscar} \xleftarrow{\alpha^d} \text{Bob}$$

$K_1 = (\alpha^a)^b$ $\qquad$ $K_2 = (\alpha^c)^d$

Authenticated DHS

Various ways to authenticate DH Scheme:

- Use signatures
- Use ElGamal-style public keys
- Use passwords

None of these approaches are perfect

Using signatures:

Station-to-Station Scheme (1)

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

Alice

a

Bob

b

Using signatures:

Station-to-Station Scheme (1)

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

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

a

b

α^a

Using signatures:

Station-to-Station Scheme (1)

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

Alice		Bob
	A, α^a $\longrightarrow$	
	$\longleftarrow$ B, α^b, $sig_B(A, \alpha^b, \alpha^a)$	
a		b
α^b		α^a

Using signatures:

Station-to-Station Scheme (1)

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

Alice		Bob
	A, α^a $\longrightarrow$	
a	$B, \alpha^b, sig_B(A, \alpha^b, \alpha^a)$ $\longleftarrow$	b
α^b	$sig_A(B, \alpha^a, \alpha^b)$ $\longrightarrow$	α^a

Using signatures:

Station-to-Station Scheme (1)

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

Alice		Bob
	$\xrightarrow{A,\ \alpha^a}$	
a	$\xleftarrow{B,\ \alpha^b,\ sig_B(A,\alpha^b,\alpha^a)}$	b
α^b	$\xrightarrow{sig_A(B,\alpha^a,\alpha^b)}$	α^a
$K = (\alpha^b)^a$	$=$	$K = (\alpha^a)^b$

Using signatures:

Station-to-Station Scheme (2)

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

Alice

a

Bob

b

Using signatures:

Station-to-Station Scheme (2)

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

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

a

b

α^a

Using signatures:

Station-to-Station Scheme (2)

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

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

a

b

α^a

$K = (\alpha^a)^b$

Using signatures:

Station-to-Station Scheme (2)

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

Alice		Bob
	A, α^a $\longrightarrow$	
	$\longleftarrow$ $B, \alpha^b, \{sig_B(\alpha^b, \alpha^a)\}_K$	
a		b
α^b		α^a

$K = (\alpha^a)^b$

Using signatures:

Station-to-Station Scheme (2)

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

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

$$\text{Alice} \xleftarrow{B,\ \alpha^b,\ \{sig_B(\alpha^b, \alpha^a)\}_K} \text{Bob}$$

Alice	Bob
a	b
α^b	α^a

$K = (\alpha^b)^a \quad = \quad K = (\alpha^a)^b$

Using signatures:

Station-to-Station Scheme (2)

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

Alice		Bob
	A, α^a $\longrightarrow$	
a	$B, \alpha^b, \{sig_B(\alpha^b, \alpha^a)\}_K$ $\longleftarrow$	b
α^b	$\{sig_A(\alpha^a, \alpha^b)\}_K$ $\longrightarrow$	α^a
$K = (\alpha^b)^a$	$=$	$K = (\alpha^a)^b$

Using ElGamal public keys:

One-Sided Authentication

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

Let M be Alice's private key, α^M her public key

Alice	Bob
M, a	b

Using ElGamal public keys:

One-Sided Authentication

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

Let M be Alice's private key, α^M her public key

Alice		Bob
	A, α^a $\longrightarrow$	
M, a		b
		α^a
		$K = (\alpha^a)^b$

Using ElGamal public keys:

One-Sided Authentication

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

Let M be Alice's private key, α^M her public key

Alice		Bob
	$\xrightarrow{A,\ \alpha^a}$	
M, a		b
	$\xleftarrow{B,\ \alpha^{Mb}}$	
α^{Mb}		α^a
		$K = (\alpha^a)^b$

Using ElGamal public keys:

One-Sided Authentication

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

Let M be Alice's private key, α^M her public key

Alice		Bob
	α^a →	
M, a		b
α^{Mb}	←	α^a
$K = (\alpha^{Mb})^{M'a}$		$K = (\alpha^a)^b$

M' =
inverse of M
in G

Using ElGamal public keys:

One-Sided Authentication

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

Let M be Alice's private key, α^M her public key

Alice $\xrightarrow{A,\ \alpha^a}$ **Bob**

Alice		Bob
M, a	$\xleftarrow{B,\ \alpha^{Mb}}$	b
α^{Mb}		α^a
$K = (\alpha^{Mb})^{M'a}$	$=$	$K = (\alpha^a)^b$

Using ElGamal public keys:

MTI Scheme

Matsumoto, Takashima, Imai (1986)

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

Let SA, SB be Alice and Bob's private keys

Alice | **Bob**

SA, a | SB, b

Using ElGamal public keys:

MTI Scheme

Matsumoto, Takashima, Imai (1986)

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

Let SA, SB be Alice and Bob's private keys

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

SA, a

SB, b

α^a

Using ElGamal public keys:

MTI Scheme

Matsumoto, Takashima, Imai (1986)

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

Let SA, SB be Alice and Bob's private keys

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

SA, a

SB, b

α^a

$$K = (\alpha^{SA})^b\,(\alpha^a)^{SB} = \alpha^{SAb+aSB}$$

Using ElGamal public keys:

MTI Scheme

Matsumoto, Takashima, Imai (1986)

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

Let SA, SB be Alice and Bob's private keys

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

$$\text{Alice} \xleftarrow{B,\ \alpha^b} \text{Bob}$$

Alice	Bob
SA, a	SB, b
α^b	α^a

$$K = (\alpha^{SA})^b (\alpha^a)^{SB} = \alpha^{SAb+aSB}$$

Using ElGamal public keys:

MTI Scheme

Matsumoto, Takashima, Imai (1986)

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

Let SA, SB be Alice and Bob's private keys

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

SA, a | SB, b

$$\xleftarrow{B,\ \alpha^b}$$

α^b | α^a

$$K = (\alpha^{SB})^a (\alpha^b)^{SA} = \alpha^{SBa+bSA} \quad = \quad K = (\alpha^{SA})^b (\alpha^a)^{SB} = \alpha^{SAb+aSB}$$

Variants

There are **many** variants of these schemes

- Using a keyed hash (a MAC) instead of encryption in STS(2)
- MQV protocol - MTI with more complex key computation
- IKE - a component of IPSec

Each solves or addresses different issues, or make different tradeoff choices

Using passwords:

Simple Password Encrypted Key Exchange

Jablon (1996)

Suppose Alice and Bob share a password **p** (not a key)

Alice	Bob
a	b

Using passwords:

Simple Password Encrypted Key Exchange

Jablon (1996)

Suppose Alice and Bob share a password **p** (not a key)

$$\text{Alice} \xrightarrow{\text{hash}(\mathbf{p})^{2a}} \text{Bob}$$

a

b

$\text{hash}(\mathbf{p})^{2a}$

$K = (\text{hash}(\mathbf{p})^{2a})^b$

Using passwords:

Simple Password Encrypted Key Exchange

Jablon (1996)

Suppose Alice and Bob share a password $\mathbf{p}$ (not a key)

$$\text{Alice} \xrightarrow{\text{hash}(\mathbf{p})^{2a}} \text{Bob}$$

$$a \xleftarrow{\text{hash}(\mathbf{p})^{2b}} b$$

$\text{hash}(\mathbf{p})^{2b}$ $\qquad$ $\text{hash}(\mathbf{p})^{2a}$

$$K = (\text{hash}(\mathbf{p})^{2b})^{a} \quad = \quad K = (\text{hash}(\mathbf{p})^{2a})^{b}$$

Using passwords:

Simple Password Encrypted Key Exchange

Jablon (1996)

Su

Main worry: dictionary attacks
(brute force attacks against the password)

Need to make sure that the opponent cannot mount offline dictionary attacks, and cannot use parties as guess validators

hash(**p**)

hash(**p**)

$K = (\mathrm{hash}(\mathbf{p})^{2b})^a$ $\equiv$ $K = (\mathrm{hash}(\mathbf{p})^{2a})^b$

PKI and Certificates

The main problem with the previous protocols:

- requiring a priori knowledge (public keys, password)

A solution:

- send the public key as part of the protocol
- how do you know the key is the "right" key?

PKI and Certificates

The main problem with the previous protocols:

- requiring a priori knowledge (public keys, password)

A solution:

- send the public key as part of
- how do you know the key

X.509 standard

Certificate, signed by a **Certifi**

$$Cert_{CA}(U, pk_U) = (U,\ pk_U,\ sig_{CA}(U, pk_U))$$

Using signatures and certificates:

Full Station-to-Station Scheme (1)

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

Alice	Bob
a	b

Using signatures and certificates:

Full Station-to-Station Scheme (1)

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

$$\text{Alice} \xrightarrow{Cert_{CA}(A, ver_A),\ \alpha^a} \text{Bob}$$

a

b

α^a

$K = (\alpha^a)^b$

Using signatures and certificates:

Full Station-to-Station Scheme (1)

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

Alice		Bob
	$Cert_{CA}(A, ver_A),\ \alpha^a$ $\longrightarrow$	
a	$Cert_{CA}(B, ver_B),\ \alpha^b,\ sig_B(A, \alpha^b, \alpha^a)$	b
	$\longleftarrow$	
α^b		α^a

$K = (\alpha^b)^a$ $\quad = \quad$ $K = (\alpha^a)^b$

Using signatures and certificates:

Full Station-to-Station Scheme (1)

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

Alice		Bob
	$Cert_{CA}(A, ver_A),\ \alpha^a$ $\longrightarrow$	
a	$Cert_{CA}(B, ver_B),\ \alpha^b,\ sig_B(A, \alpha^b, \alpha^a)$ $\longleftarrow$	b
α^b	$sig_A(B, \alpha^a, \alpha^b)$ $\longrightarrow$	α^a

$K = (\alpha^b)^a$ $\qquad = \qquad$ $K = (\alpha^a)^b$

Certificates: Problem 1

Certificate Revocation

- When a secret key is compromised, we should revoke the certificate
- Expiration/TTL is not enough - all expired certificates are invalid, but not all invalid certificates are expired

Use Certificate Revocation Lists (CRL)

- Need to be checked at every certificate validation
- Potential for denial-of-service attacks
- Somewhat defeats the purpose of PK

No comprehensive solution

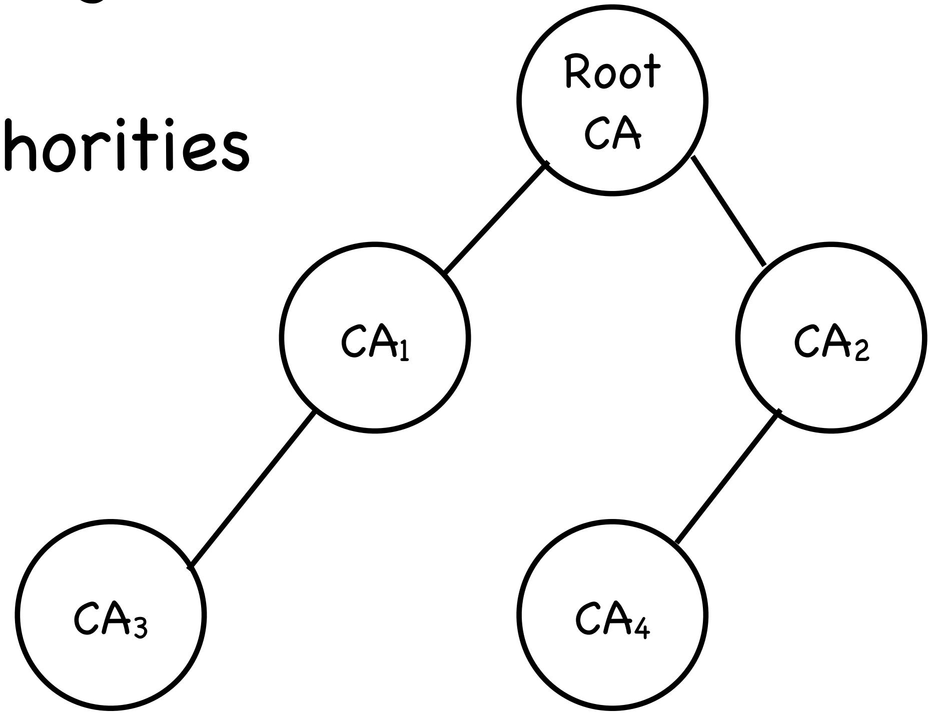
Certificates: Problem 2

Compromised Certification Authorities

- If a certificate authority is compromised, we cannot trust certificates it has signed

Hierarchical certification authorities

- The chain of trust
- CA_3's verification key is signed by CA_1
- CA_1's verification key is signed by RootCA



- BUT: Need to send all certificates in a chain

Ce blem 2

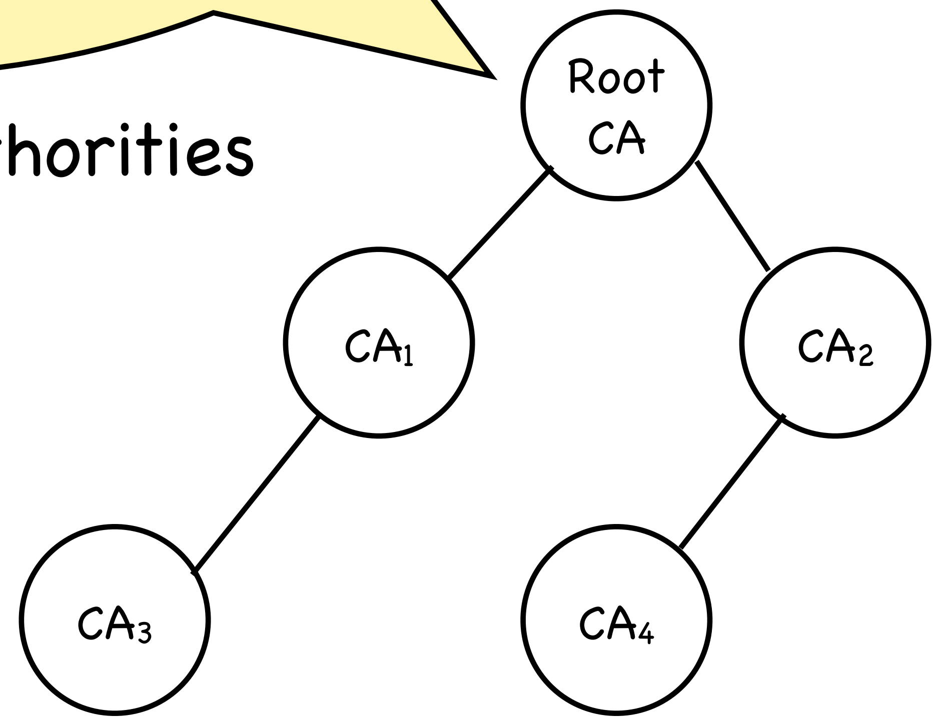
Who vouches
for the RootCA's
verification key?
No one – has to be trusted
by some other means

Compro

- If a nised, we cannot
trust c

Hierarchical certification authorities

- The chain of trust
- CA_3's verification key is signed by CA_1
- CA_1's verification key is signed by RootCA



- BUT: Need to send all certificates in a chain