Introducción a la Criptografía y a la Seguridad de la Información

Part 7
Key Establishment Protocols

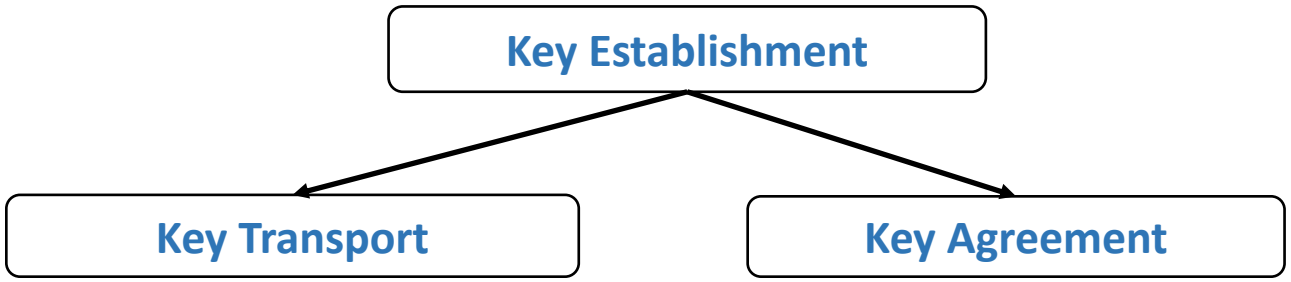
Jorge E. Camargo, PhD

Agenda

- Key Establishment Protocols
  - Key Transport Vs Key Agreement
  - Deffie-Hellman Key Exchange
  - Man-in-the-middle Attack
  - Shamir's Three-Pass Protocol

Key Transport vs. Key Agreement

Key establishment: process to establish a shared secret key available to two or more parties.



Key Transport (or key distribution): one party has decided a key and transmits it to the other party.

Key Agreement: key establishment technique in which a shared secret is derived by two (or more) parties.

In key agreement protocols neither party knows the key in advance; it is determined as a result of their interaction.

Deffie-Hellman Key Exchange

The Diffie-Hellman (DH) key agreement protocol (also called exponential key agreement) was developed by Whitfield Diffie and Martin Hellman in 1976 and published in the ground-breaking paper *"New Directions in Cryptography."*

This protocol allows two users to exchange a secret key over an insecure medium without any prior secrets.

It relies on the fact that the two parties who are exchanging messages only have to compute powers in modular arithmetic in order to know what the key they are agreeing on is, but the eavesdropper will have to compute discrete logarithms to obtain the same key.

in 1997, it was revealed that this idea was actually first discovered in 1974, by Malcom Williamson of GCHQ (British Intelligence)

The DH patent expired on April 29, 1997

Steps in Negotiating Diffie-Hellman Key Agreement

Alice (A) and Bob (B) want to establish a key for communicating.

1) A and B agree on two numbers, a generator g of $\mathbb{Z}_p^*$ and a prime p.

2) A chooses a secret random integer x, $1 \leq x \leq p-2$.

3) B chooses a secret random integer y, $1 \leq y \leq p-2$.

4) A computes X = g^x mod p.

5) B computes Y = g^y mod p.

6) A sends X to B , and B sends Y to A .

7) A computes $k = Y^x \bmod p$

8) B computes $k = X^y \bmod p$

9) k is the encryption key for this communication session.

When x and y are used only once we call this an ephemeral DH *secret key agreement*.

Example

1) **A** and **B** agree g = 7, while p = 23.

2) **A** chooses x = 5.

3) **B** chooses y = 8.

4) **A** computes X = 7^5 mod 23 = 17.

5) **B** computes Y = 7^8 mod 23 = 12.

6) **A** sends 17 to **B** , and **B** sends 12 to **A** .

7) **A** computes k = 12^5mod 23 = 18

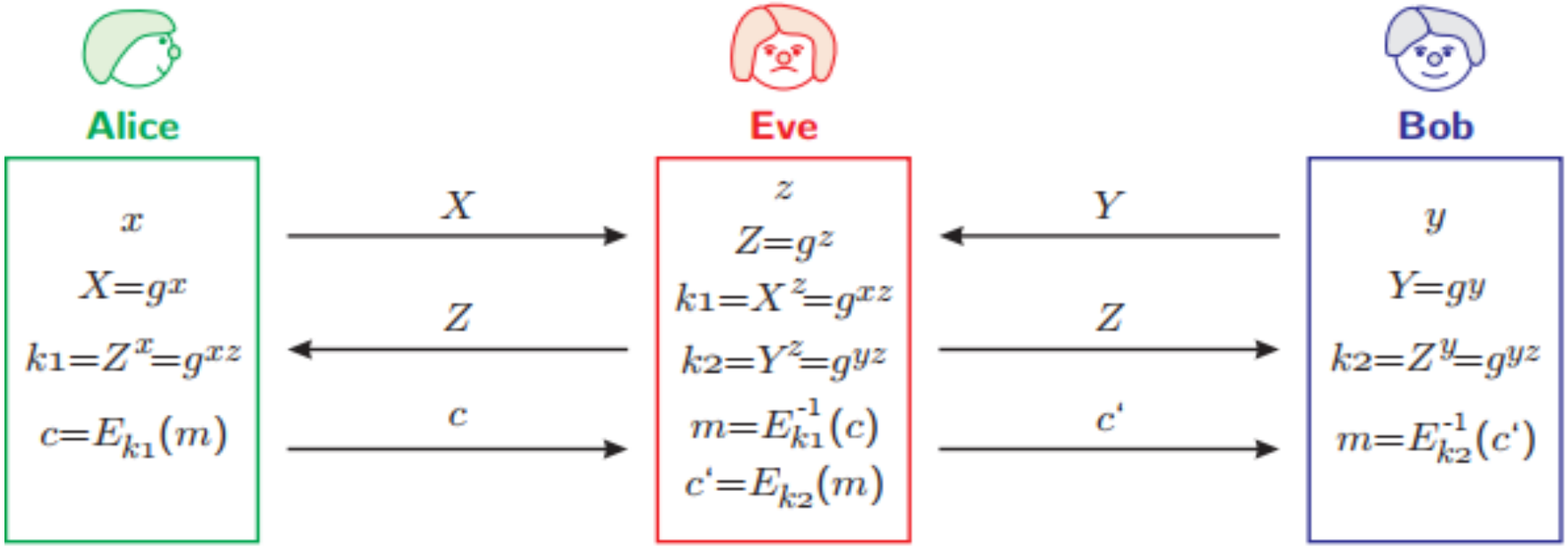
8) **B** computes k = 17^8 mod 23 = 18

9) k = 18 is the encryption key for this communication session.

The length of the Diffie-Hellman key k itself is typically about 128 bytes (1024 bits). This is often further processed down to 8 bytes (64 bits) to obtain a DES encryption key

Man-in-the-middle Attack

An intruder Eve (E) can interpose itself between A and B .



This vulnerability is due to the fact that DH key exchange does not authenticate the participants. Possible solutions include the use of digital signatures.

Steps of Man-in-the-middle Attack

1) E chooses an exponent z, $1 \leq z \leq p - 2$.
2) E computes $Z = g^z \bmod p$.
3) E intercepts X and Y .
4) E sends Z to A and B .
5) E computes $k_1 = X^z \bmod p$ and $k_2 = Y^z \bmod p$.
6) A computes $k_1 = Z^x \bmod p$.
7) B computes $k_2 = Z^y \bmod p$.
8) A sends $c = E_{k_1}(m)$ to B (E).
9) E computes $m = E_{k_1}^{-1}$ (c) and reads *m*!!!
10) E sends $c' = E_{k_2}(m)$ to B
11) B computes $m = E_{k_2}^{-1}(c')$ and reads *m*

Bob has no reason to believe the communication was insecure. Meanwhile, Eve is reading the juicy gossip that she has obtained.

Shamir's Three-Pass Protocol

A and **B** exchange 3 messages over a public channel. As a result, the secret *k* is transferred with privacy (but no authentication).

(a) One-time setup:

1) Select and publish a prime p.

2) **A** and **B** choose respectively secret random numbers *a*, *b*, $1 \le a, b \le p - 2$, $gcd(a, p - 1) = gcd(b, p - 1) = 1$ and compute $a^{-1} \bmod p - 1$ and $b^{-1} \bmod p - 1$.

(b) Protocol:

1) **A** chooses a random key *k* (the secret), $1 \le k \le p - 1$.

2) **A** sends $k_1 = k^a \bmod p$ to **B**.

3) **B** sends $k_2 = k_1^b \bmod p$ to **A**.

4) **A** sends $k_3 = k_2^{a^{-1}} \bmod p$ to **B**.

5) **B** computes $k = k_3^{b^{-1}} \bmod p$

Why does it works?

(b) Protocol:

1) **A** chooses a random key k (the secret), $1 \leq k \leq p - 1$.

2) **A** sends $k_1 = k^a \bmod p$ to **B**.

3) **B** computes $k_2 = k_1^b \bmod p = k^{ab} \bmod p$ and sends k_2 to **A**.

4) **A** computes $k_3 = k_2^{a^{-1}} \bmod p = \left(k^{ab}\right)^{a^{-1}} \bmod p = k^b \bmod p$ and sends k_3 to B.

5) **B** computes $k_3^{b^{-1}} \bmod p = \left(k^b\right)^{b^{-1}} \bmod p = k \bmod p$.

The security of this protocol lies in the difficulty of the discrete logarithm problem. This protocol is not secure against a man-in-the-middle attack.

Correctness

Notice that if $x \equiv y \pmod{p-1}$ then for FLT, $k^x \equiv k^y \pmod{p}$.

Thus, for some $\ell \in \mathbb{Z}$, $x = \ell(p-1) + y$, and therefore

$k^x \equiv k^{\ell}(p-1)+ y \equiv (k^{p-1})^{\ell} k^y \equiv k^y \pmod{p}$

Since $k^{p-1} \equiv 1 \pmod{p}$ by FLT.

Now, in Shamir's Three-Pass Protocol we have that $aa^{-1} \equiv 1 \pmod{p-1}$ then $k^{aa^{-1}} \equiv k \pmod{p}$ and the proof follows acordingly.

References

- Pinzon, Yoan. “Introducción a la criptografía y a la seguridad de la información”, 2013.
- Menezes A., Handbook of applied Cryptografy, 5th Edition. CRC Press, 2001.
- A Graduate Course in Applied Cryptography by D. Boneh and V. Shoup
- H. Delfs and H. Knebl, Introduction to Cryptography, 3rd ed. 2015.
- J. A. Buchmann, Introduction to Cryptography. 2004.
- T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, Ch. 31 in Introduction to Algorithms, 3rd ed. 2009.
- B. Lomas de Zamora: Gradi. Hacking desde cero, Fox Andina, 2011.
- Secure Coding Working Group, Japan Smartphone Security Association (JSSEC), Android Application Secure Design/Secure Coding Guidebook, 2017.