Katz, Lindell
Introduction to Modern Cryptrography
Slides Chapter 10

Markus Bläser, Saarland University

Key-exchange

What is a key-exchange protocol Π?

- Alice and Bob start by holding a security parameter 1^n.
- Then they run Π (using private random bits).
- Alice and Bob can communicate with each other using the protocol.
- The channel is authenticated, i.e., the adversary can listen to their communication but not manipulate it.
  (This is an issue in practical applications!)
- In the end, Alice and Bob output $k_A, k_B \in \{0,1\}^n$.
- Correctness requirement: $k_A = k_B (= k)$.
- Their communication is recorded in a transcript trans.

Key-exchange (2)

The key-exchange experiment $\mathsf{KE}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n)$:

1. Two parties holding 1^n execute Π resulting in a transcript trans and a key k.
2. $b \in \{0, 1\}$ is chosen uniformly at random.
   If $b = 0$, then set $\hat{k} = k$.
   If $b = 1$, then choose $\hat{k} \in \{0, 1\}^n$ uniformly at random
3. $\mathcal{A}$ is given trans and $\hat{k}$ and $\mathcal{A}$ outputs a bit b'.
4. The outcome of the experiment is 1 if $b = b'$ and 0 otherwise.

Definition (10.1)

Π is *secure in the presence of an eavesdropper* if for all ppt $\mathcal{A}$,

$$\Pr[\mathsf{KE}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n) = 1] \leq \frac{1}{2} + \mathsf{negl}(n).$$

Diffie–Hellman protocol

Construction 10.2:

Common input is 1^n

1. Alice runs Gen to obtain (G, q, g)
2. Alice chooses a uniform $x \in \mathbb{Z}_q$ and computes $h_A := g^x$.
3. Alice sends (G, q, g, h_A) to Bob.
4. Bob chooses a uniform $y \in \mathbb{Z}_q$ and computes $h_B := g^y$.
   Bob sends h_B to Alice and outputs $k_B := h_A^y$
5. Alice outputs $k_A := h_B^x$.

Protocol is correct, as

$$k_A = h_B^x = g^{xy} = h_A^y = k_B.$$

In practice, G and g are fixed in advance.

Diffie–Hellman protocol (2)

In the protocol, the keys are group elements. Modify the experiment accordingly $\longrightarrow \hat{\mathsf{KE}}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n)$.

Theorem (10.3)

If the decisional Diffie–Hellman problem is hard relative to Gen, *then the Diffie–Hellman key exchange protocol* Π *is EAV-secure (with respect to the experiment* $\hat{\mathsf{KE}}^{\mathsf{eav}}_{\mathcal{A},\Pi}(n)$*).*

Diffie-Hellman is insecure against man-in-the-middle attacks.