

Cryptography

Lecture 21

Announcements

- HW8 due 5/3

Agenda

- Last time:
 - Elliptic Curve Groups
 - Key Exchange Definitions (10.3)
- This time:
 - More on Key Exchange Definitions
 - Diffie-Hellman Key Exchange (10.3)
 - El Gamal Encryption (11.4)
 - RSA Encryption (11.5)

Key Agreement

The key-exchange experiment $KE_{A,\Pi}^{eav}(n)$:

1. Two parties holding 1^n execute protocol Π . This results in a transcript $trans$ containing all the messages sent by the parties, and a key k output by each of the parties.
2. A uniform bit $b \in \{0,1\}$ is chosen. If $b = 0$ set $\hat{k} := k$, and if $b = 1$ then choose $\hat{k} \in \{0,1\}^n$ uniformly at random.
3. A is given $trans$ and \hat{k} , and outputs a bit b' .
4. The output of the experiment is defined to be 1 if $b' = b$ and 0 otherwise.

Definition: A key-exchange protocol Π is secure in the presence of an eavesdropper if for all ppt adversaries A there is a negligible function neg such that

$$\Pr \left[KE_{A,\Pi}^{eav}(n) = 1 \right] \leq \frac{1}{2} + neg(n).$$

Discussion of Definition

- Why is this the “right” definition?
- Why does the adversary get to see \hat{k} ?

Diffie-Hellman Key Exchange

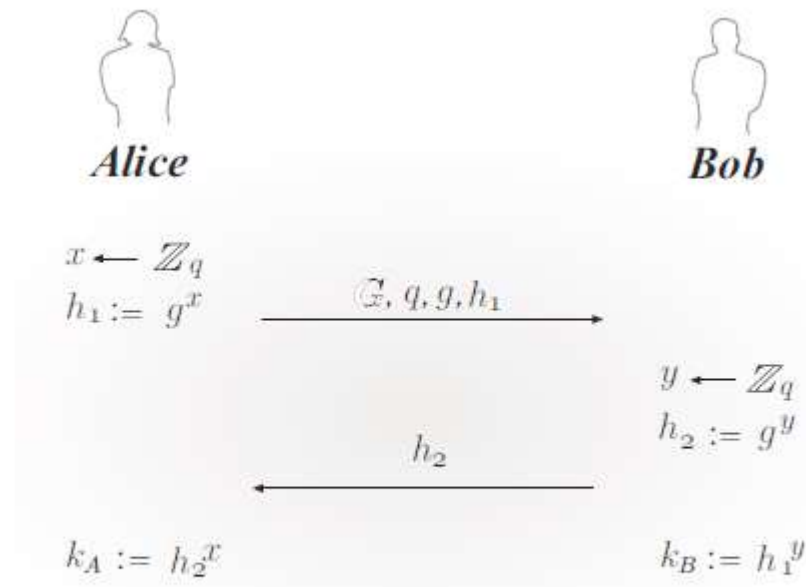


FIGURE 10.2: The Diffie-Hellman key-exchange protocol.

Recall DDH problem

We say that the DDH problem is hard relative to G if for all ppt algorithms A , there exists a negligible function neg such that

$$\begin{aligned} & |\Pr[A(G, q, g, g^x, g^y, g^z) = 1] \\ & - \Pr[A(G, q, g, g^x, g^y, g^{xy}) = 1]| \leq neg(n). \end{aligned}$$

Security Analysis

Theorem: If the DDH problem is hard relative to \mathcal{G} , then the Diffie-Hellman key-exchange protocol Π is secure in the presence of an eavesdropper.

Public Key Encryption

Definition: A public key encryption scheme is a triple of ppt algorithms (Gen, Enc, Dec) such that:

1. The key generation algorithm Gen takes as input the security parameter 1^n and outputs a pair of keys (pk, sk) . We refer to the first of these as the public key and the second as the private key. We assume for convenience that pk and sk each has length at least n , and that n can be determined from pk, sk .
2. The encryption algorithm Enc takes as input a public key pk and a message m from some message space. It outputs a ciphertext c , and we write this as $c \leftarrow Enc_{pk}(m)$.
3. The deterministic decryption algorithm Dec takes as input a private key sk and a ciphertext c , and outputs a message m or a special symbol \perp denoting failure. We write this as $m := Dec_{sk}(c)$.

Correctness: It is required that, except possibly with negligible probability over (pk, sk) output by $Gen(1^n)$, we have $Dec_{sk}(Enc_{pk}(m)) = m$ for any legal message m .

CPA-Security

The CPA experiment $PubK^{cpa}_{A,\Pi}(n)$:

1. $Gen(1^n)$ is run to obtain keys (pk, sk) .
2. Adversary A is given pk , and outputs a pair of equal-length messages m_0, m_1 in the message space.
3. A uniform bit $b \in \{0,1\}$ is chosen, and then a challenge ciphertext $c \leftarrow Enc_{pk}(m_b)$ is computed and given to A .
4. A outputs a bit b' . The output of the experiment is 1 if $b' = b$, and 0 otherwise.

Definition: A public-key encryption scheme $\Pi = (Gen, Enc, Dec)$ is CPA-secure if for all ppt adversaries A there is a negligible function neg such that

$$\Pr \left[PubK^{cpa}_{A,\Pi}(n) = 1 \right] \leq \frac{1}{2} + neg(n).$$

Discussion

- Discuss how in the public key setting security in the presence of an eavesdropper and CPA security are equivalent (since anyone can encrypt using the public key).
- Discuss how CPA-secure encryption cannot be deterministic!!
 - Why not?

El Gamal Encryption

--Show how we can derive El Gamal PKE from
Diffie-Hellman Key Exchange

Important Property

Lemma: Let G be a finite group, and let $m \in G$ be arbitrary. Then choosing uniform $k \in G$ and setting $k' := k \cdot m$ gives the same distribution for k' as choosing uniform $k' \in G$. Put differently, for any $\hat{g} \in G$ we have

$$\Pr[k \cdot m = \hat{g}] = 1/|G|.$$

El Gamal Encryption Scheme

CONSTRUCTION 11.16

Let \mathcal{G} be as in the text. Define a public-key encryption scheme as follows:

- Gen: on input 1^n run $\mathcal{G}(1^n)$ to obtain (\mathbb{G}, q, g) . Then choose a uniform $x \leftarrow \mathbb{Z}_q$ and compute $h := g^x$. The public key is $\langle \mathbb{G}, q, g, h \rangle$ and the private key is $\langle \mathbb{G}, q, g, x \rangle$. The message space is \mathbb{G} .
- Enc: on input a public key $pk = \langle \mathbb{G}, q, g, h \rangle$ and a message $m \in \mathbb{G}$, choose a uniform $y \leftarrow \mathbb{Z}_q$ and output the ciphertext

$$\langle g^y, h^y \cdot m \rangle.$$

- Dec: on input a private key $sk = \langle \mathbb{G}, q, g, x \rangle$ and a ciphertext $\langle c_1, c_2 \rangle$, output

$$\hat{m} := c_2 / c_1^x.$$

The El Gamal encryption scheme.

Security Analysis

Theorem: If the DDH problem is hard relative to G , then the El Gamal encryption scheme is CPA-secure.

Textbook RSA Encryption

CONSTRUCTION 11.25

Let GenRSA be as in the text. Define a public-key encryption scheme as follows:

- **Gen:** on input 1^n run $\text{GenRSA}(1^n)$ to obtain N, e , and d . The public key is $\langle N, e \rangle$ and the private key is $\langle N, d \rangle$.
- **Enc:** on input a public key $pk = \langle N, e \rangle$ and a message $m \in \mathbb{Z}_N^*$, compute the ciphertext

$$c := [m^e \bmod N].$$

- **Dec:** on input a private key $sk = \langle N, d \rangle$ and a ciphertext $c \in \mathbb{Z}_N^*$, compute the message

$$m := [c^d \bmod N].$$

The plain RSA encryption scheme.

Is Plain-RSA Secure?

- It is deterministic so cannot be secure!

Additional Attacks

We will look at additional attacks in one of the upcoming class exercises.