Cryptography

Lecture 23
Announcements

• HW8 due 5/4
• HW9 due 5/11
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 $KE^{eav}_{A,\Pi}(n)$:

1. Two parties holding $1^n$ execute protocol $\Pi$. 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 $\Pi$ is secure in the presence of an eavesdropper if for all ppt adversaries $A$ there is a negligible function $neg$ such that

$$\Pr[KE^{eav}_{A,\Pi}(n) = 1] \leq \frac{1}{2} + neg(n).$$
Discussion of Definition

• Why is this the “right” definition?
• Why does the adversary get to see $\hat{\kappa}$?
Diffie-Hellman Key Exchange

\[ x \leftarrow \mathbb{Z}_q \]
\[ h_1 := g^x \]
\[ y \leftarrow \mathbb{Z}_q \]
\[ h_2 := g^y \]
\[ k_A := h_2^x \]
\[ k_B := h_1^y \]

**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

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

Theorem: If the DDH problem is hard relative to $G$, then the Diffie-Hellman key-exchange protocol $\Pi$ 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 $\bot$ 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 $PubK_{A,\Pi}^{cpa}(n)$:

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

$$\Pr \left[ PubK_{A,\Pi}^{cpa}(n) = 1 \right] \leq \frac{1}{2} + neg(n).$$
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 $(\mathcal{G}, q, g)$. Then choose a uniform $x \leftarrow \mathbb{Z}_q$ and compute $h := g^x$. The public key is $(\mathcal{G}, q, g, h)$ and the private key is $(\mathcal{G}, q, g, x)$. The message space is $\mathcal{G}$.

- **Enc:** on input a public key $pk = (\mathcal{G}, q, g, h)$ and a message $m \in \mathcal{G}$, 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 $sk = (\mathcal{G}, q, g, x)$ 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.
**CONSTRUCTION 11.25**

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

- **Gen**: on input $1^n$ run GenRSA($1^n$) to obtain $N, e, \text{ and } d$. 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
  \[ c := [m^e \mod N]. \]

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

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 discussion sessions.