CSE 599b: Cryptography **(Winter 2006)**

Lecture 11: Semantic Security vs Indistinguishability Security

8 February 2006

Lecturer: Paul Beame *Scribe: Paul Beame*

1 Semantic Security

From now on we will at least aim for the ability to handle chosen plaintext attacks (CPA). Also, of the two versions of chosen ciphertext attack, CCA1 and CCA2, we will only consider CCA2 attacks which allow the chosen ciphertexts to depend on (but be different from) the challenge ciphertext. We also have 3 security notions: Semantic Security (SS), Indistinguishability Security (IND) which is also sometimes called 'left-or-right' security (which is natural given the way the pair oracle works), and Nonmalleability (NM).

So far, this leaves 6 potential levels of security of interest: IND-CPA, SS-CPA, NM-CPA, IND-CCA2, SS-CCA2, NM-CCA2. However, we will see that the IND and SS versions are equivalent.

In order to show this we need to define semantic security formally. The basic idea we want to capture is that an adversary that gets to query the encryption algorithm and choose any distribution on plaintexts, can't predict the value of any polynomial-time computable function of a plaintext from that distribution given its ciphertext than it would if it received an encryption of an independently chosen (unrelated) plaintext from that distribution. To make it convenient to talk about what that adversary does, we split its operation into pieces that we describe separately.

Definition 1.1. *A symmetric encryption scheme $(\mathcal{K}, \mathcal{E}, D)$ is SS-CPA secure if and only if for every polynomial-time computable function f and for every PPTs A, P, $\mathcal{M}$, the function*

$$\begin{aligned}\epsilon(k) \;=\; & \Pr[P(S, \mathcal{E}_K(M), 1^k) = f(S, M, 1^k) \mid K \leftarrow \mathcal{K};\; S \leftarrow A^{\mathcal{E}_K}(1^k);\; M \leftarrow \mathcal{M}(S, 1^k)] \\ & - \Pr[P(S, \mathcal{E}_K(M'), 1^k) = f(S, M, 1^k) \mid K \leftarrow \mathcal{K};\; S \leftarrow A^{\mathcal{E}_K}(1^k); \\ & \qquad\qquad M \leftarrow \mathcal{M}(S, 1^k);\; M' \leftarrow \mathcal{M}(S, 1^k)]\end{aligned}$$

is negligible.

We think of A as the querier, $\mathcal{M}$ as the message distribution and P is the part of the adversary's computation that predicts the function f and S is the state information that gets passed from the querier to the distribution and prediction algorithms. Note that this definition even allows the output of the function f to depend on the information that the adversary learns from its chosen plaintext queries.

Also note that like in IND-CPA security, it is essential that $\mathcal{E}_K$ be randomized or stateful; otherwise if S records all $(plaintext, ciphertext)$ pairs then the distribution $\mathcal{M}$ could simply be on all plaintexts queried and the predicter P could simply look in its ciphertext list for its second argument C and return the corresponding plaintext which would be M in the first place and M' in the second case. However, this attack would fail if $\mathcal{E}_K$ returned different values each time it is used.

2 Equivalence of Semantic and Indistinguishability Security

Before we show this equivalence we make an additional observation about indistinguishability security. The original definition for IND-CPA security allowed unrestricted calls to the pair oracle. For simplicity we will consider the case in which the last call to the pair oracle is the only one in which the inputs to the oracle may be distinct. We will give this a (non-standard) name.

Definition 2.1. *A symmetric encryption scheme $(\mathcal{K}, \mathcal{E}, D)$ is* last-call-IND-CPA secure *if and only if for every (pair oracle) PPT A that makes a series of calls to its oracle in which the two queries are the same and ends with one call in which they may be distinct, the function*

$$\begin{aligned}\epsilon(k) &= \Pr[A^{\mathcal{E}_K(Select(\cdot,\cdot,0))}(1^k) = b \mid K \leftarrow \mathcal{K}(1^k)] \\ &\quad - \Pr[A^{\mathcal{E}_K(Select(\cdot,\cdot,1))}(1^k) = b \mid K \leftarrow \mathcal{K}(1^k)]\end{aligned}$$

is negligible where $Select(M^0, M^1, b) = M^b$ for $b \in \{0, 1\}$.

The following lemma follows by a hybrid argument.

Lemma 2.2. *IND-CPA security is equivalent to last-call-IND-CPA security.*

We now are ready to show that semantic security and indistinguishability security are equivalent.

Theorem 2.3. *A symmetric encryption scheme $(\mathcal{K}, \mathcal{E}, D)$ is SS-CPA secure if and only if it is last-call-IND-CPA secure.*

Proof. ($\Rightarrow$): Suppose that the system in SS-CPA secure. Let B be any pair oracle PPT and define

$$\epsilon(k) = \Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,0))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k)] - \Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,1))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k)].$$

We show that $\epsilon(k)$ is negligible. Let A on input 1^k simulate B on input 1^k until B produces a pair of queries that contains distinct elements. (Up until this point B has only made pair queries with the same argument so A can simulate them with a call to its oracle.) A then outputs the entire memory contents of B at this point including the pair (M^0, M^1) of distinct queries. Define $\mathcal{M}(S, 1^k)$ to be the distribution which chooses each of M^0 or M^1 with probability 1/2. Define $f(S, M, 1^k) = 1$ and let P simulate the remainder of the computation of B starting with memory S using the returned ciphertext C from the pair oracle.

Observe that

$$\begin{aligned}&\Pr[P(S, \mathcal{E}_K(M), 1^k) = f(S, M, 1^k) \mid K \leftarrow \mathcal{K};\ S \leftarrow A^{\mathcal{E}_K}(1^k);\ M \leftarrow \mathcal{M}(S, 1^k)] \\ &= \Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,0))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k)]\end{aligned}$$

and

$$\begin{aligned}&\Pr[P(S, \mathcal{E}_K(M'), 1^k) = f(S, M, 1^k) \mid K \leftarrow \mathcal{K};\ S \leftarrow A^{\mathcal{E}_K}(1^k);\ M \leftarrow \mathcal{M}(S, 1^k);\ M' \leftarrow \mathcal{M}(S, 1^k)] \\ &= \Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,b))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k);\ b \leftarrow \mathcal{U}_1].\end{aligned}$$

By SS-CPA security these two quantities are within some negligible $\epsilon'(k)$ of each other. The latter quantity is simply

$$(\Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,0))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k)] + \Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,1))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k)])/2$$

and thus

$$\Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,0))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k)] - \Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,1))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k)] \leq \epsilon'(k)/2.$$

($\Leftarrow$): Given f, A, P, and $\mathcal{M}$, define a pair oracle PPT B that on input 1^k simulates A on input 1^k and calls its oracle with (M_i, M_i) whenever A calls its oracle with M_i until A produces S. Then B calls $\mathcal{M}$ on input S and 1^k twice to yield M and M'. It calls its pair oracle on the pair (M, M') yielding some C, checks if $P(S, C, 1^k) = f(S, M, 1^k)$, outputting 1 if they are equal and 0 if they are not equal.

Now

$$\begin{aligned}&\Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,0))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k)]\\ &= \Pr[P(S, \mathcal{E}_K(M), 1^k) = f(S, M, 1^k) \mid K \leftarrow \mathcal{K};\ S \leftarrow A^{\mathcal{E}_K}(1^k);\ M \leftarrow \mathcal{M}(S, 1^k)]\end{aligned}$$

and

$$\begin{aligned}&\Pr[B^{\mathcal{E}_K(Select(\cdot,\cdot,1))}(1^k) = 1 \mid K \leftarrow \mathcal{K}(1^k)]\\ &= \Pr[P(S, \mathcal{E}_K(M'), 1^k) = f(S, M, 1^k) \mid K \leftarrow \mathcal{K};\ S \leftarrow A^{\mathcal{E}_K}(1^k);\ M \leftarrow \mathcal{M}(S, 1^k);\ M' \leftarrow \mathcal{M}(S, 1^k)]\end{aligned}$$

and thus the IND-CPA security implies that these two quantities are within some negligible $\epsilon(k)$ of each other which implies SS-CPA security. □

3 Public Key Cryptography

Before we consider CCA2 attacks or NM security it will be convenient first to apply IND-CPA definitions to public key cryptosystems since there are many similarities with the symmetric encryption case.

3.1 Diffie-Hellman Secret Key Exchange

Public key cryptography was introduced by Diffie and Hellman in 1976 in the form of method for producing a shared secret key that could then be used in conventional symmetric encryption schemes. The security of their system depends on the intractability of computing discrete logarithms.

The protocol $DH_{(p,g)}$ is defined given a prime p and generator g of $\mathbb{Z}_p^*$.

1. Alice chooses x uniformly at random from $\mathbb{Z}_{p-1}$.

2. Bob chooses y uniformly at random from $\mathbb{Z}_{p-1}$.
3. Alice computes $a = g^x \mod p$ and sends a to Bob.
4. Bob computes $b = g^x \mod p$ and sends b to Alice.
5. Alice computes $K = b^x \mod p = (g^y)^x \mod p = g^{xy} \mod p$.
6. Bob computes $K = a^x \mod p = (g^x)^y \mod p = g^{xy} \mod p$.

This relies on $EXP_{(p,g)}$ being a one-way function but the assumption that the adversary cannot get information about K is quite a bit stronger. There are two forms of the assumption on which the security of this scheme is based.

Computational Diffie-Hellman (CDH) Assumption For x, y randomly chosen from $\mathbb{Z}_{p-1}$ it is computationally infeasible to predict $g^{xy} \mod p$ given g, p, $g^x \mod p$, and $g^y \mod p$.

This is not enough because it permits partial information about K. The stronger form is what one would like. However, we shall see that it isn't exactly correct.

Decision-Diffie-Hellman Assumption (DDH) The distribution of $(g^x \mod p, g^y \mod p, g^{xy} \mod p)$ for x, y randomly chosen from $\mathbb{Z}_{p-1}$ is computationally indistinguishable from the distribution of $(g^x \mod p, g^y \mod p, g^z \mod p)$ for x, y, z randomly chosen from $\mathbb{Z}_{p-1}$.

Although the original Diffie-Hellman assumption is defined over the cyclic group $\mathbb{Z}_p^*$, the same construction makes sense over any cyclic group.