CS 7880 Graduate Cryptography November 15, 2017

Lecture 20: CCA Security, Construction with RO, and Cramer-Shoup

Lecturer: Daniel Wichs *Scribe: Xuangui Huang*

1 Topics Covered

- CCA Security
- CCA2-Secure Scheme in the RO Model
- CCA1-Secure Scheme: Cramer-Shoup

In this lecture we introduce the concept of Chosen Ciphertext Attack (CCA) Security. We give a construction in the ideal world with random oracles. Then we start to construt a CCA1-secure scheme, called "Cramer-Shoup", from DDH via Hash Proof Systems.

2 CCA Security

In the CCA Game we will allow the adversary to choose ciphertext and get its plaintext. For a public key encryption scheme $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$, we define its CCA game $\mathsf{CCAGame}^b(n)$ to be the following game for $b \in \{0,1\}$:

1. Challenger samples key pairs $(pk, sk) \leftarrow \mathsf{KeyGen}$, and sends $\mathsf{PK} - only$ to Adversary;
2. Adversary sends query ciphertext ct_i to Challenger and receives from Challenger $m_i = \mathsf{Dec}_{sk}(ct_i)$ for any number of rounds as Adversary wants;
3. Adversary sends challenge massages m_0^*, m_1^* to Challenger and receives the challenge ciphertext $ct^* = \mathsf{Enc}_{pk}(m_b^*)$;
4. Adversary then continue to send query ciphertext $ct_i \neq ct^*$ and gets its m_i for any number of rounds;
5. Adversary output a number $b' \in \{0,1\}$.

The value of this game is defined as the b' the Adversary outputs at last.

We call an encryption scheme is $\mathsf{CCA2}$ secure if $\mathsf{CCAGame}^0 \approx \mathsf{CCAGame}^1$ for any PPT adversary. If we remove the 4-th step in this game, we get the corresponding definition of $\mathsf{CCA1}$ security.

Obviously Rabin scheme is not $\mathsf{CCA1}$ secure. For ElGamal scheme, notice that we can "rerandomize" the ciphertext so that we can get a queried ciphertext $ct_i \neq ct^*$ but still encrypting the same message. It is not clear if it is $\mathsf{CCA1}$ secure.

Conceptually speaking, CCA security means that after seeing that Bob sends the ciphertext $\mathsf{Enc}_{pk}(m)$ to Alice, Eve cannot come up with a ciphertext of a message that is related

to m, say $\mathsf{Enc}_{pk}(m+1)$, to send to Alice. In symmetric key settings, this can be achieved by authentication combining with CPA security; Bob authenticate his own message then encrypt so Eve cannot counterfeit the ciphertext of a message related to Bob's. But this idea doesn't work in public key encryption.

3 Construction in the RO Model

Generally, we don't know how to convert a CPA secure scheme into a CCA secure scheme. But we can do this in the random oracle model. Intuitively we are using the random oracle to authenticate the messages, similar to the aforementioned case in symmetric key settings.

Given a CPA secure scheme $(\mathsf{KeyGen}, \mathsf{Enc}, \mathsf{Dec})$, we get the following scheme $(\mathsf{KeyGen}', \mathsf{Enc}', \mathsf{Dec}')$:

- $\mathsf{KeyGen}' = \mathsf{KeyGen}$;
- $\mathsf{Enc}'_{pk}(m)$: samples $x \leftarrow \{0,1\}^n$, returns $ct = \mathsf{Enc}_{pk}((m,x); RO(m,x))$;
- $\mathsf{Dec}'_{sk}(ct)$: gets $(m,x) \leftarrow \mathsf{Dec}_{sk}(ct)$, checks if $\mathsf{Enc}_{pk}((m,x); RO(m,x)) = ct$, if so outputs m, otherwise outputs $\perp$.

We define the following hybrids to prove its CCA security:

- H_0^b: $\mathsf{CCAGame}^b$
- H_1^b: In $\mathsf{Dec}'_{sk}(ct)$, instead of $\mathsf{Dec}_{sk}(ct)$, try all previous RO queries (m_j, x_j), check if $\mathsf{Enc}_{pk}((m,x); RO(m,x)) = ct$ holds, if so output m_j; Note that after this hybrid we get rid of sk;
- H_2^b: If adversary calls (m_b^*, x) to RO, then independently sample a result at random.

Then intuition is that the adversary must use Enc' to get the query ciphertexts so it learns nothing from these queries. We can prove:

- $H_1^b \approx H_0^b$: if there exists RO query (m,x) and ciphertext ct such that $\Pr[\mathsf{Enc}_{pk}(m,x) = ct] \neq negl(n)$, then the original scheme is not CPA secure;
- $H_2^b \approx H_1^b$: by CPA security, there is no way for the adversary to figure out (m_b^*, x^*);
- $H_2^0 \approx H_2^1$: by CPA security, we can safely switch from ciphertext of m_0^* to that of m_1^*;

4 Cramer-Shoup Scheme

We can construct a CCA1 secure scheme under DDH assumption via "Hash Proof Systems" (HPS). The idea is that in the query rounds the adversary need to prove to the challenger that it knows the plaintexts of those query ciphertexts. This idea looks like Non-Interactive Zero-Knowledge Proofs (NIZK), but we don't know how to built NIZK from DDH.

As usual, we can sample $(\mathbb{G}, g, q) \leftarrow \mathsf{GroupGen}(1^n)$ and $h \leftarrow \mathbb{G}$, and these parameters $(\mathbb{G}, g, q, h)$ is publicly shared. Now we define a language $L = \{(g^r, h^r) \colon r \in \mathbb{Z}_q\} \in \mathbf{NP}$, and another language $\overline{L} = \{(g^{r_1}, h^{r_2}) \colon r_1 \neq r_2 \in \mathbb{Z}_q\} \in \mathbf{NP}$. Let U_L, $U_{\overline{L}}$ be the uniform distribution over L and $\overline{L}$ respectively, then we have the following property:

Claim 1 (Property I) *Under* DDH, $U_L \approx U_{\overline{L}}$ *even given* g, h.

Proof: $(g, h, g^y, h^y) = (g, g^x, g^y, g^{xy}) \approx (g, g^x, g^y, g^z) \equiv (g, h, g^y, h^z) \approx (g, h, g^x, h^z | x \neq z)$. □

We define our HPS as follows:

- $\mathsf{HPSGen}(1^n)$: $(x, y) \leftarrow \mathbb{Z}_q^2$, set $sk = (x, y)$, $pk = g^x h^y = f$;
- $H_{pk}((c_1, c_2), r) = f^r$;
- $H_{sk}(c_1, c_2) = c_1^x c_2^y$.

Then we have the following two properties. Property II, the completeness property, means that the proof reveals nothing about r; Property III, the soundedness property, means that the adversary has no idea of the hash even though knowing pk.

Claim 2 (Property II, Completeness) $\forall (c_1, c_2) \in L$ *with unique witness* r, *we have*

$$H_{pk}((c_1, c_2), r) = f^r = H_{sk}(c_1, c_2).$$

Claim 3 (Property III, Soundedness) $\forall pk$, $\forall (c_1, c_2) \in \overline{L}$, $H_{sk}(c_1, c_2) \equiv U_{\mathbb{G}}$ *conditioned on* pk.

Proof: Property II is obvious; we prove property III. Let $a \in \mathbb{Z}_q$ such that $h = g^a$. Then $(c_1, c_2) = (g^{r_1}, h^{r_2}) = (g^{r_1}, h^{ar_2})$. Thus $pk = g^{a+ay}$, $H_{sk}(c_1, c_2) = g^{r_1 x + ar_2 y}$. As $r_1 \neq r_2$, $x + ay$ and $r_1 x + ar_2 y$ are linearly independent. Thus conditioned on pk, $H_{sk}(c_1, c_2)$ is still uniform. □

Finally we can describe the Cramer-Shoup Scheme:

- $\mathsf{KeyGen}(1^n)$: $(pk_1, sk_1) \leftarrow \mathsf{HPSGen}(1^n)$, $(pk_2, sk_2) \leftarrow \mathsf{HPSGen}(1^n)$, set $pk = (pk_1, pk_2)$, $sk = (sk_1, sk_2)$;
- $\mathsf{Enc}_{pk}(m)$: $(g^r, h^r) \leftarrow L$, return $(g^r, h^r, H_{pk_1}((g^r, h^r), r) \cdot m, H_{pk_2}((g^r, h^r), r))$;
- $\mathsf{Dec}_{sk}(c_1, c_2, h_1, h_2)$: check that $h_2 = H_{sk_2}(c_1, c_2)$, if so output $h_1 / H_{sk_1}(c_1, c_2)$, otherwise output $\perp$.

We will see the proof of its CCA1 security in the next lecture.