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 construct 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 \((\text{KeyGen}, \text{Enc}, \text{Dec})\), we define its CCA game \(\text{CCAGame}^b(n)\) to be the following game for \(b \in \{0, 1\}\):

1. Challenger samples key pairs \((pk, sk) \leftarrow \text{KeyGen}\), and sends \(PK - \text{only}\) to Adversary;
2. Adversary sends query ciphertext \(ct_i\) to Challenger and receives from Challenger \(m_i = \text{Dec}_{sk}(ct_i)\) for any number of rounds as Adversary wants;
3. Adversary sends challenge messages \(m_0^*, m_1^*\) to Challenger and receives the challenge ciphertext \(ct^* = \text{Enc}_{pk}(m_0^*)\);
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 CCA2 secure if \(\text{CCAGame}^0 \approx \text{CCAGame}^1\) for any PPT adversary. If we remove the 4-th step in this game, we get the corresponding definition of CCA1 security.

Obviously Rabin scheme is not 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 CCA1 secure.

Conceptually speaking, CCA security means that after seeing that Bob sends the ciphertext \(\text{Enc}_{pk}(m)\) to Alice, Eve cannot come up with a ciphertext of a message that is related
to \( m \), say \( \text{Enc}_{pk}(m + 1) \), 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 \((\text{KeyGen}, \text{Enc}, \text{Dec})\), we get the following scheme \((\text{KeyGen}', \text{Enc}', \text{Dec}')\):

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

We define the following hybrids to prove its CCA security:

- \( H^0_b \): CCAGame
- \( H^1_b \): In \( \text{Dec}'_{sk}(ct) \), instead of \( \text{Dec}_{sk}(ct) \), try all previous \( RO \) queries \((m_j, x_j)\), check if \( \text{Enc}_{pk}((m, x); RO(m, x)) = ct \) 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 \( \text{Enc}' \) to get the query ciphertexts so it learns nothing from these queries. We can prove:

- \( H^0_b \approx H^1_0 \): if there exists \( RO \) query \((m, x)\) and ciphertext \( ct \) such that \( \Pr[\text{Enc}_{pk}(m, x) = ct] \neq \text{negl}(n) \), then the original scheme is not CPA secure;
- \( H^1_b \approx H^1_b \): by CPA security, there is no way for the adversary to figure out \((m^*_b, x^*)\);
- \( H^0_2 \approx H^1_2 \): 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 build NIZK from DDH.

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

Lecture 20, Page 2
Claim 1 (Property I) Under DDH, $U_L \approx U_T$ even given $g, h$.

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

We define our HPS as follows:

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

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) $\forall pk, \forall (c_1, c_2) \in T, H_{sk}(c_1, c_2) \equiv U_G$ conditioned on $pk$.

Proof: Property II is obvious; we prove property III. Let $a \in \mathbb{Z}_q$ such that $h = g^a$. Then $(c_1, c_2) = (g^r, h^z) = (g^{r_1}, h^{ar_2})$. Thus $pk = g^{a+ay}, H_{sk}(c_1, c_2) = g^{r_1x + ar_2y}$. As $r_1 \neq r_2, x + ay$ and $r_1x + ar_2y$ are linearly independent. Thus conditioned on $pk, H_{sk}(c_1, c_2)$ is still uniform.

Finally we can describe the Cramer-Shoup Scheme:

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