Proofs of Security

TUT n°1 — Public-key encryption

M2-SIF, M2-CRYPTO

Léo Ackermann

Answer all ▸ questions before looking at ⋆ questions.

Relations between properties

We recall the definition of an *indistinguishability-under-chosen-plaintext-attacks* secure public-key encryption (PKE) scheme, written IND-CPA PKE for short.

Definition 1 (IND-CPA PKE) *A PKE is said IND-CPA secure if for any polytime adversary $\mathcal{A}$ its advantage $\mathrm{Adv}_{\mathcal{A}}(\mathcal{G}^{\textsf{IND-CPA}}) := |\Pr(\mathcal{G}^{\textsf{Ind-CPA}}(\mathcal{A}, \lambda) \to \top) - 1/2|$ is negligible (in the security parameter). The security game is defined as follows.*

$\mathcal{G}^{\textsf{IND-CPA}}(\mathcal{A}, \lambda)$

1: $(pk, sk) \leftarrow \textsf{KeyGen}(1^\lambda)$
2: $(m_0, m_1) \leftarrow \mathcal{A}(pk)$
3: $b \leftarrow_{\$} \{0, 1\}$
4: $b' \leftarrow \mathcal{A}(pk, \textsf{Enc}(pk, m_b))$
5: *If* $b = b'$*, then return* $\top$*, else return* $\bot$

▸ **Question 1.** *Propose cryptographic games for the following security properties, defining the associated advantage:*

- *OW-CPA stands for one-wayness-under-chosen-plaintext-attacks, and captures the unability of an adversary to recover the message corresponding to a ciphertext.*
- *IND-CCA1 stands for indistinguishability-under-chosen-ciphertext-attack, and captures the unabilty of an adversary to distinguishing ciphertext even given access to a decryption oracle before* commiting its challenge messages.*
- *NM-CPA stands for non-malleability-under-chosen-plaintext-attacks, and capture the unability for an adversary to perturb a ciphertext with control. More precisely, she should not be able to come with a relation $\mathcal{R}$ that link the original plaintext and the plaintext corresponding to the modified ciphertext.†*

▸ **Question 2.** *Show that any IND-CCA1 PKE scheme is also IND-CPA. Recall that a scheme verifies a game-based property PROP if for all polytime adversary $\mathcal{A}$, its advantage against the game is negligible.*

▸ **Question 3.** *Show that any IND-CPA PKE scheme has a non-deterministic* Enc *function.*

▸ **Question 4.** *Show that any IND-CPA PKE scheme is also OW-CPA.*

⋆ **Question 5.** *Show that any NM-CPA scheme is also IND-CPA.*

▸ **Question 6.** *Show that no PKE scheme is perfectly secure. The latter is an information theoric notion capturing the fact that observing a ciphertext gives no clue on the underlying plaintext.*

Additionnal properties do not come for free

We say that a PKE scheme is additively homomorphic whenever for all messages (m_1, m_2), and evenly generated keypair $(pk, sk) \leftarrow \textsf{KeyGen}(1^\lambda)$ – where λ is the security parameter – it holds that

$$\textsf{Enc}(pk, m_1) \cdot \textsf{Enc}(pk, m_2) = \textsf{Enc}(pk, m_1 + m_2).$$

▸ **Question 7.** *Show that an additively homomorphic scheme cannot be IND-CCA2.*

*This "before challenge" restriction is represented by the "1" in the IND-CCA1 terminology. When IND-CCA is used instead, this restriction disappears. Naturally, the adversary is not allowed to decrypt the challenge ciphertext.

†This is trivial, by taking $\mathcal{R}$ being the non-equality for example. In order to be meaningful, we want a relation that does not hold between the original plaintext and a random one.

▸ **Question 8.** *Consider Π an IND-CCA2 PKE scheme. To detect decryption errors, one proposes to transmit $(\mathsf{Enc}(m, \mathsf{pk}), \mathcal{H}(m))$ as the ciphertext corresponding to m, where $\mathcal{H}$ is a hash function. Show that the resulting scheme is no more IND-CPA secure.*

Around ElGamal

Recall that the ElGamal PKE scheme consists in the following three algorithms.

- $\mathsf{KeyGen}(1^\lambda)$ produces (G, q, g) – a cyclic group, its order and a generator of it – then samples $x \hookleftarrow_\$ \mathbb{Z}_q$ and computes $h = g^x$. The secret key is x, the public key is (G, q, g, h).
- $\mathsf{Enc}(pk, m \in G)$ samples $y \hookleftarrow_\$ \mathbb{Z}_q$ and returns $c = (g^y, h^y \cdot m)$.
- $\mathsf{Dec}(sk, c = (c_1, c_2))$ returns c_2/c_1^x.

▸ **Question 9.** *Is El-Gamal NM-CPA? IND-CCA?*

▸ **Question 10.** *Can the salt y be reused for encrypting another message?*

Around RSA

In number theory, Euler's totient function – denoted φ here – counts the positive integers up to a given integer n that are relatively prime to n. An important fact concerning the cryptosystem we study here is that $\varphi(n)$ is the order of the multiplicative group of integers modulo n, denoted $\mathbb{Z}/n\mathbb{Z}$.

The RSA assumption, introduced by Rivest, Shamir and Adleman in 1977, posists that the following game cannot be won with non-negligible probability by any polytime adversary.

Definition 2 (RSA) *Let $N = pq$ be the product of two primes p and q. Let e be relatively prime to $\varphi(N)$, and y living in $\mathbb{Z}/n\mathbb{Z}$. Given (N, e, y), the RSA game consists in returning x such that $x^e = y \mod N$.*

▸ **Question 11.** *Show that RSA reduces to FACTO, the problem of recovering the pair of primes (p, q) given $N = pq$.*

We focus now on a PKE scheme made of the following three algorithms:

- KeyGen samples two primes p and q, and e relatively prime to $\varphi(N := pq)$. It then computes d such that $e \cdot d \equiv 1 \mod \varphi(N)$ using extended Euclidean algorithm, and output (e, N) as the public key, and d as the secret key.
- $\mathsf{Enc}(pk, m)$ samples a random salt $r \in \mathbb{Z}/N\mathbb{Z}$ and returns $(r^e, \mathcal{H}(r) * m) \in (\mathbb{Z}/N\mathbb{Z})^2$ where $\mathcal{H}$ is an idealized[‡] hash function.
- $\mathsf{Dec}(sk, (\alpha, \beta))$ returns $\beta/\mathcal{H}(\alpha^d) \in (\mathbb{Z}/N\mathbb{Z})$.

▸ **Question 11.** *Show that the scheme is correct.*

▸ **Question 12.** *Show how one can use an adversary against the RSA problem to build an adversary against the IND-CPA game.*

▸ **Question 13.** *Is it IND-CPA under the hypothesis that the RSA problem is hard?*

▸ **Question 14.** *Is it IND-CCA under the same hypothesis?*

[‡]Which means **(1)** for any input x, the output $\mathcal{H}(x)$ is uniformly distributed **(2)** the adversary must evaluate $\mathcal{H}(x)$ to know something about it. This is modeled by considering a random oracle $\mathcal{O}^{\mathsf{RO}}$ within the game-based proof: the adversary hands her x to get its hash, uniformly distributed.