# TD 9: IND-CCA Security

**Exercise 1.**
Recall the (Lyubashevsky-Palacio-Segev) LWE-based encryption scheme from the lecture.

- $\mathsf{KeyGen}(1^\lambda)$: Let $m, n, q, B$ be integers such that $m \geq n$ and $q > 12mB^2$. Sample $\mathbf{A} \hookleftarrow U(\mathbb{Z}_q^{m \times n})$, $\mathbf{s} \hookleftarrow U((-B, B]^n)$ and $\mathbf{e} \hookleftarrow U((-B, B]^m)$. Return
$$\mathsf{pk} := (\mathbf{A}, \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}) \text{ and } \mathsf{sk} := \mathbf{s}.$$
- $\mathsf{Enc}(\mathsf{pk}, \mu \in \{0,1\})$: Sample $(\mathbf{t}, \mathbf{f}, g) \hookleftarrow U((-B, B]^m \times (-B, B]^n \times (-B, B])$ and output
$$(c_1, c_2) = (\mathbf{t}^\top \mathbf{A} + \mathbf{f}^\top, \mathbf{t}^\top \mathbf{b} + g + \lfloor \frac{q}{2} \rfloor \mu).$$
- $\mathsf{Dec}(\mathsf{sk}, c_1, c_2)$: take the representative of $\mu' = c_2 - c_1 \cdot \mathsf{sk}$ in $(-q/2, q/2]$ and return 0 if it has norm $< q/4$, 1 otherwise.

1. Prove correctness and IND-CPA security of this scheme.

2. Show that this scheme is not IND-CCA2 secure.

**Exercise 2.**
Let $\Pi_0 = (\mathsf{Keygen}_0, \mathsf{Encrypt}_0, \mathsf{Decrypt}_0)$ be an IND-CCA2-secure public-key encryption scheme which only encrypts single bits (i.e., the message space is $\{0,1\}$). We consider the following multi-bit encryption scheme $\Pi_1 = (\mathsf{Keygen}_1, \mathsf{Encrypt}_1, \mathsf{Decrypt}_1)$ , where the message space is $\{0,1\}^L$ for some $L$ polynomial in the security parameter $\lambda$.

**$\mathsf{Keygen}_1(1^\lambda)$:** Generate a key pair $(PK, SK) \leftarrow \Pi_0.\mathsf{Keygen}_0(1^\lambda)$. Output $(PK, SK)$.

**$\mathsf{Encrypt}_1(PK, M)$:** In order to encrypt $M = M[1] \dots M[L] \in \{0,1\}^L$, do the following.

1. For $i = 1$ to $L$, compute $C[i] \leftarrow \Pi_0.\mathsf{Encrypt}_0(PK, M[i])$.
2. Output $C = (C[1], \dots, C[L])$.

**$\mathsf{Decrypt}_1(SK, C)$** Parse the ciphertext $C$ as $C = (C[1], \dots, C[L])$. Then, for each $i \in \{1, \dots, L\}$, compute $M[i] = \Pi_0.\mathsf{Decrypt}_0(SK, C[i])$. If there exists $i \in \{1, \dots, L\}$ such that $M[i] = \perp$, output $\perp$. Otherwise, output $M = M[1] \dots M[L] \in \{0,1\}^L$.

1. Show that $\Pi_1$ does not provide IND-CCA2 security, even if $\Pi_0$ is secure in the IND-CCA2 sense.

Let $\Pi = (\mathsf{Keygen}, \mathsf{Encrypt}, \mathsf{Decrypt})$ be an IND-CCA2-secure public-key encryption scheme with message space $\{0,1\}^L$ for some $L \in \mathbb{N}$. We consider the modified public-key encryption scheme $\Pi' = (\mathsf{Keygen}', \mathsf{Encrypt}', \mathsf{Decrypt}')$ where the message space is $\{0,1\}^{L-1}$ and which works as follows.

**$\mathsf{Keygen}'(1^\lambda)$:** Generate two key pairs $(PK_0, SK_0) \leftarrow \mathsf{Keygen}(1^\lambda)$, $(PK_1, SK_1) \leftarrow \mathsf{Keygen}(1^\lambda)$.
Define $PK := (PK_0, PK_1)$, $SK := (SK_0, SK_1)$.

**$\mathsf{Encrypt}'(PK, M)$:** In order to encrypt $M \in \{0,1\}^{L-1}$, do the following.

1. Choose a random string $R \leftarrow U(\{0,1\}^{L-1})$ and define $M_L = M \oplus R \in \{0,1\}^{L-1}$ and $M_R = R$.

2. Compute $C_L \leftarrow \Pi.\mathsf{Encrypt}(PK_0, 0||M_L)$ and $C_1 \leftarrow \Pi.\mathsf{Encrypt}(PK_1, 1||M_R)$.

Output $C = (C_L, C_R)$.

**Decrypt'**$(SK, C)$ Parse $C$ as $(C_L, C_R)$. Then, compute $\tilde{M}_L = \Pi.\mathsf{Decrypt}(SK_0, C_L)$ and $\tilde{M}_R = \Pi.\mathsf{Decrypt}(SK_1, C_R)$. If $\tilde{M}_L = \perp$ or $\tilde{M}_R = \perp$, output $\perp$. If the first bit of $M_L$ (resp. $M_R$) is not 0 (resp. 1), return $\perp$. Otherwise, parse $\tilde{M}_L$ as $0||M_L$ and $\tilde{M}_R$ as $1||M_R$, respectively, where $M_L, M_R \in \{0,1\}^{L-1}$, and output $M = M_L \oplus M_R \in \{0,1\}^{L-1}$.

2. Show that the modified scheme $\Pi'$ does not provide IND-CCA2 security, even if the underlying scheme $\Pi$ does.

3. Show that, if $\Pi$ provides IND-CCA1 security, so does the modified scheme $\Pi'$. Namely, show that an IND-CCA1 adversary against $\Pi'$ implies an IND-CCA1 adversary againt $\Pi$.

**Exercise 3.**
We are looking here at different modifications of the Fujisaki-Okamoto (FO) transform that fail at providing CCA2 security. Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key encryption scheme assumed to be IND-CPA secure with message space $\{0,1\}^{k+\ell}$. We recall the FO transform, where $H$ is a hash function that is modeled as a RO.

$\mathsf{KeyGen}(1^\lambda)$**:** Sample and return $(pk, sk) \leftarrow \mathsf{Gen}(1^\lambda)$.

$\mathsf{Enc}'(pk, m \in \{0,1\}^k)$**:** Sample $r \hookleftarrow U(\{0,1\}^\ell)$ and return $c = \mathsf{Enc}(pk, m||r; H(m||r))$, where $H(m||r)$ is the randomness used by the algorithm.

$\mathsf{Dec}'(sk, c)$**:** Compute $m||r \leftarrow \mathsf{Dec}(sk, c)$ and return $m$ if $c = \mathsf{Enc}(pk, m||r; H(m||r))$. Otherwise, return $\perp$.

1. What happens if $\ell = O(\log(\lambda))$?

For the next questions, do not forget to look at the previous exercises.

2. Show that there exists an IND-CPA secure encryption scheme such that if we replace every instance of $H(m||r)$ with $H(r)$, then its FO transform is not IND-CCA2 secure.

3. Show that there exists an IND-CPA secure encryption scheme such that if we always return $m$ in the decryption algorithm, without checking the consistency of the randomness used in the ecnryption, then its FO transform is not IND-CCA2 secure.