

Lecture 4: PKE with CCA1 Security

Based on “Advanced Topics in Cryptography
[J.Katz]

Shengli Liu

slliu@sjtu.edu.cn

Lab of Cryptography and Information Security
Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai 200240, China

Definition (IND-CCA1/CCA2 Security)

\forall stateful PPT $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$,

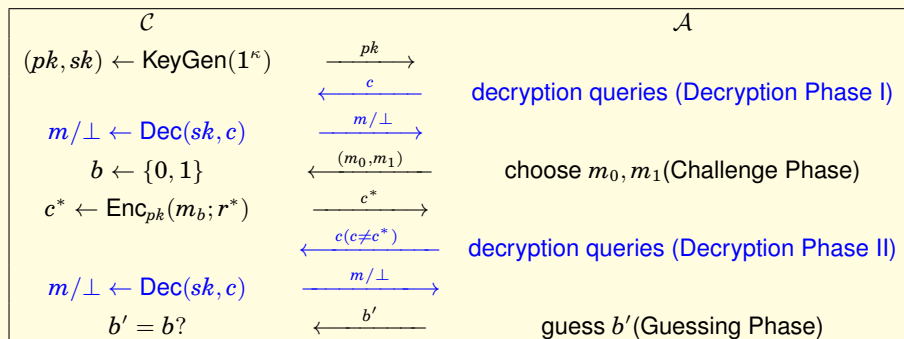
$$\Pr \left[\begin{array}{l} (pk, sk) \leftarrow \text{Gen}(1^\kappa); (s, m_0, m_1) \leftarrow \mathcal{A}_1^{\text{Dec}(sk, \cdot)}(pk); \\ b \leftarrow \{0, 1\}, c^* \leftarrow \text{Enc}_{pk}(m_b; r^*); b' \leftarrow \mathcal{A}_2^{\text{Dec}_{\neq c^*}(sk, \cdot)}(s, c) : b' = b \end{array} \right] = 1/2 \pm \text{negl}(\kappa).$$

Definition (IND-CCA1/CCA2 Security)

\forall stateful PPT $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$,

$$\Pr \left[\begin{array}{l} (pk, sk) \leftarrow \text{Gen}(1^\kappa); (s, m_0, m_1) \leftarrow \mathcal{A}_1^{\text{Dec}(sk, \cdot)}(pk); \\ b \leftarrow \{0, 1\}, c^* \leftarrow \text{Enc}_{pk}(m_b; r^*); b' \leftarrow \mathcal{A}_2^{\text{Dec}_{\neq c^*}(sk, \cdot)}(s, c) : b' = b \end{array} \right] = 1/2 \pm \text{negl}(\kappa).$$

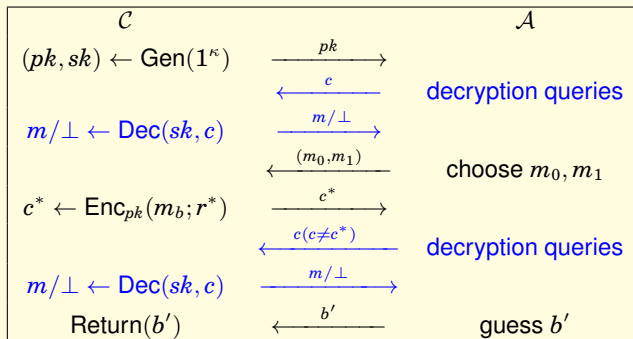
i.e., $|\Pr[\mathcal{A} \text{ wins in the CCA1/CCA2 game}] - 1/2| = |\Pr[b = b'] - 1/2| = \text{negl}(\kappa)$,



Definition (IND-CCA1/CCA2 Security)

\forall PPT $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$,
 $2|\Pr[b = b'] - 1/2| = |\Pr[b' = 1|b = 1] - \Pr[b' = 1|b = 0]| = 2\text{negl}(\kappa) = \text{negl}(\kappa)$,
 i.e., $\left| \Pr \left[\text{Exp}_{PKE, \mathcal{A}}^{CCA}(1) = 1 \right] - \Pr \left[\text{Exp}_{PKE, \mathcal{A}}^{CCA}(0) = 1 \right] \right| = \text{negl}(\kappa)$.

$\text{Exp}_{PKE, \mathcal{A}}^{CCA1/CCA2}(b)$:



Language L : a set of strings, i.e., $L \subseteq \{0, 1\}^*$.

$L \in \mathcal{P}$: \exists Poly-Time(PT) Turing machine M s. t. $\forall x \in \{0, 1\}^*$,

$$x \in L \Leftrightarrow M(x) = 1.$$

$L \in \mathcal{NP}$: \exists Poly-Time(PT) Turing machine M s. t. $\forall x \in \{0, 1\}^*$,

$$x \in L \Leftrightarrow \exists w_x \in \{0, 1\}^{poly(|x|)}, \text{ s. t. } M(x, w_x) = 1.$$

$L_1 \leq_p L_2$: language L_1 is **poly-time reducible** to language L_2 if $\exists f$ s.t.

- (1) f is poly-time computable;
- (2) $x \in L_1 \Leftrightarrow f(x) \in L_2$.

Note 1: $L_1 \leq_p L_2, L_2 \in \mathcal{P} \Rightarrow L_1 \in \mathcal{P}$;

Note 2: $L_1 \leq_p L_2, L_2 \in \mathcal{NP} \Rightarrow L_1 \in \mathcal{NP}$;

\mathcal{NP} -Complete: L' is \mathcal{NP} -Complete if

- (1) $L' \in \mathcal{NP}$;
- (2) $L \leq_p L', \forall L \in \mathcal{NP}$.

Example: (1) **SAT**: the language of all satisfiable CNF formulae; (2) $\{ G : G \text{ is a graph which contains a Hamilton cycle} \}$. (3) $\{ G = (V, E) : G \text{ is a 3-colorable graph} \}$.

Interactive Proof (IP) system for language L .

- consisting of Prover P and ppt Verifier V .
- $P(x) \Rightarrow V(x)$: common input $x \in L$ and interactions between P and V . Finally V will output 0/1.
- **Completeness.** $\forall x \in L, (P(x) \Rightarrow V(x)) = 1$.
- **Soundness.** $\forall x \notin L, \forall \tilde{P}, (\tilde{P}(x) \Rightarrow V(x)) = 0$ with high probability.

Note. $P \subseteq NP \subseteq IP, IP = PSPACE$.

A pair of ppt algorithms (P, V) is a **non-interactive zero-knowledge (NIZK)** proof system for a language $L \in NP$ if:

Completeness. $\forall x \in L$ and its witness w_x ,

$$\Pr \left[r \leftarrow \{0, 1\}^{\text{poly}(\kappa)}; \pi \leftarrow P(r, x, w_x) : V(r, x, \pi) = 1 \right] = 1.$$

Soundness. $\forall x \notin L, \forall \tilde{P}$ (even all-powerful \tilde{P}), the following is negl (in κ):

$$\Pr \left[r \leftarrow \{0, 1\}^{\text{poly}(\kappa)}; \tilde{\pi} \leftarrow \tilde{P}(r, x) : V(r, x, \tilde{\pi}) = 1 \right] = \text{negl}(\kappa).$$

Zero-knowledge. There exists a ppt simulator **Sim** s.t. $\forall x \in L$, (with $|x| = \kappa$) and \forall witness w_x for x , the following distributions are computationally indistinguishable:

$$\{r \leftarrow \{0, 1\}^{\text{poly}(\kappa)}; \pi \leftarrow P(r, x, w_x) : (r, x, \pi)\} \approx_c \{(\tilde{r}, \tilde{\pi}) \leftarrow \text{Sim}(x) : (\tilde{r}, x, \tilde{\pi})\}$$

Note. The requirement that P is ppt is due to cryptographic applications.

A pair of ppt algorithms (P, V) is an **adaptive non-interactive zero-knowledge (aNIZK)** proof system for a language $L \in NP$ if:

Completeness. $\forall x \in L$ and its witness w_x ,

$$\Pr \left[r \leftarrow \{0, 1\}^{\text{poly}(\kappa)}; \pi \leftarrow P(r, x, w_x) : V(r, x, \pi) = 1 \right] = 1.$$

Adaptive Soundness. $\forall \tilde{P}$ (even all-powerful \tilde{P}), the following is negligible in κ :

$$\begin{aligned} \Pr \left[r \leftarrow \{0, 1\}^{\text{poly}(\kappa)}; (x, \tilde{\pi}) \leftarrow \tilde{P}(r) : V(r, x, \tilde{\pi}) = 1 \wedge x \in \{0, 1\}^\kappa \setminus L \right] \\ = \text{negl}(\kappa). \end{aligned}$$

Adaptive Zero-knowledge. There exists a ppt stateful simulator $\text{Sim} = (\text{Sim}_1, \text{Sim}_2)$ s.t. and \forall stateful ppt $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, the following distributions are computationally indistinguishable:

$$\begin{aligned} \{ r \leftarrow \{0, 1\}^{\text{poly}(\kappa)}; (x \in L, w_x) \leftarrow \mathcal{A}_1(r); \pi \leftarrow P(r, x, w_x); b \leftarrow \mathcal{A}_2(r, x, \pi) : b = 1 \} \\ \approx_c \{ (\tilde{r} \leftarrow \text{Sim}_1(1^\kappa); (x \in L, w_x) \leftarrow \mathcal{A}_1(\tilde{r}); \tilde{\pi} \leftarrow \text{Sim}_2(\tilde{r}, x); b \leftarrow \mathcal{A}_2(\tilde{r}, x, \tilde{\pi}) : b = 1 \} \end{aligned}$$

Building blocks: CPA secure PKE = (KeyGen, Enc, Dec) with perfect correctness and aNIZK (P, V).

Noar-Yung's PKE' Construction

KeyGen'(1^κ): $(pk_1, sk_1) \leftarrow \text{KeyGen}(1^\kappa)$;
 $(pk_2, sk_2) \leftarrow \text{KeyGen}(1^\kappa)$;
 $r \leftarrow \{0, 1\}^{\text{poly}(\kappa)}$;
 $pk = (pk_1, pk_2, r)$; $sk = (sk_1)$; Return (pk, sk) .

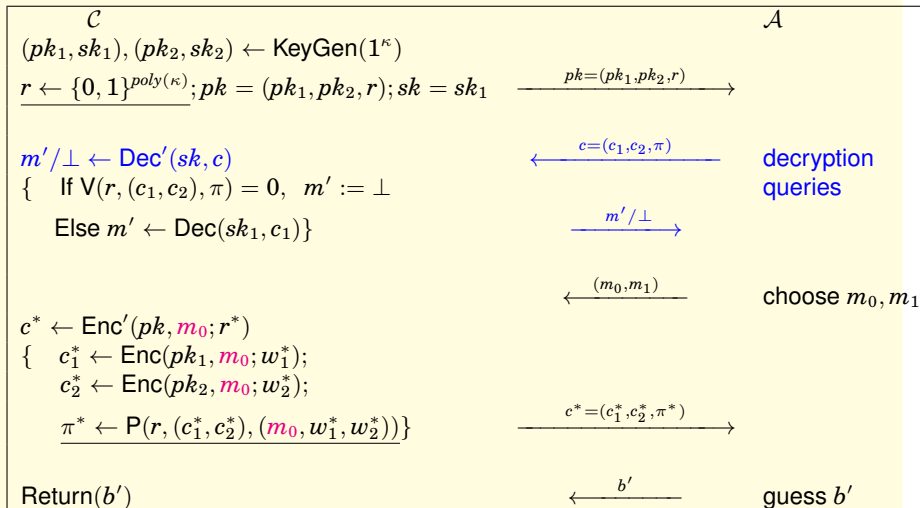
Enc'(pk, m): $c_1 \leftarrow \text{Enc}(pk_1, m; w_1)$;
 $c_2 \leftarrow \text{Enc}(pk_2, m; w_2)$;
 $\pi \leftarrow P(r, (c_1, c_2), (m, w_1, w_2))$.
 Return (c_1, c_2, π) .

Dec'(sk, c): $sk_1 := sk$; $(c_1, c_2, \pi) := c$
 If $V(r, (c_1, c_2), \pi) = 0$
 Return \perp ;
 Else $m' \leftarrow \text{Dec}(sk_1, c_1)$;
 Return m' .

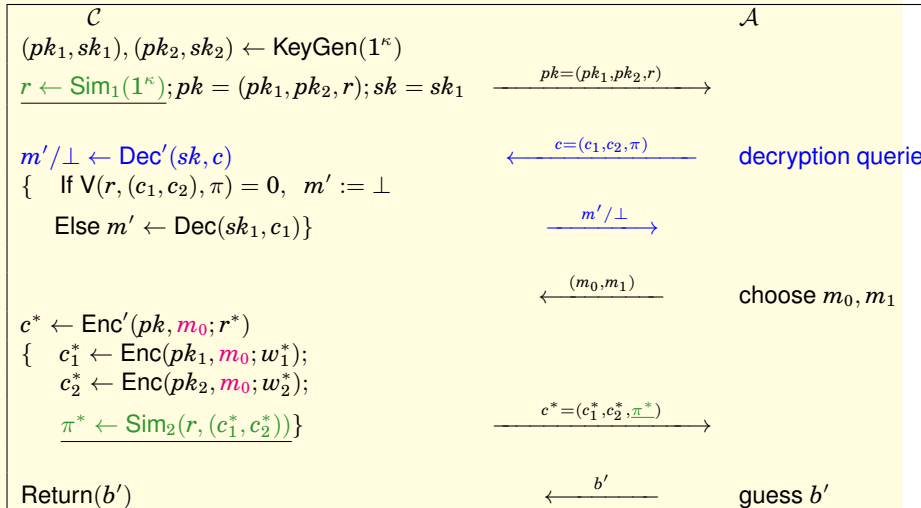
aNIZK for $L \in \mathcal{NP}$

$$L := \{(c_1, c_2) \mid \exists (m, w_1, w_2) \text{ s.t. } c_1 = \text{Enc}(pk_1, m; w_1) \wedge c_2 = \text{Enc}(pk_2, m; w_2)\}.$$

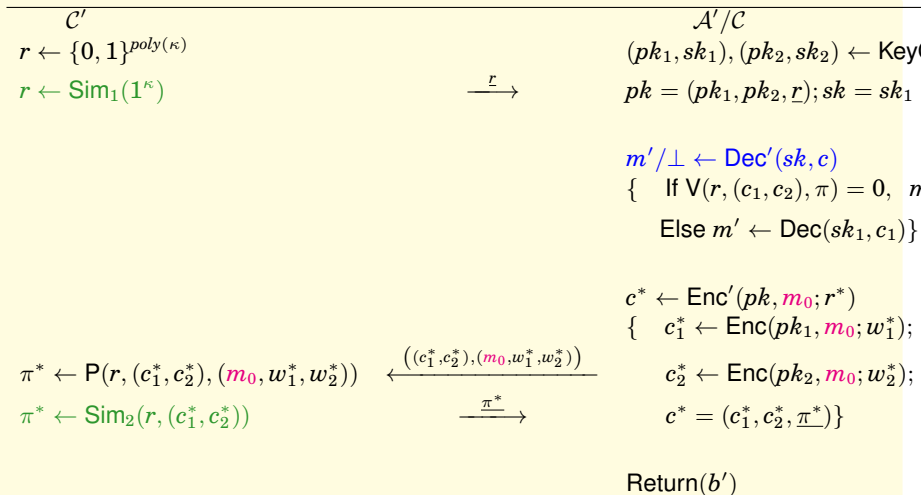
Game 0: $= \text{Exp}_{PKE', \mathcal{A}}^{\text{CCA1-0}}(\kappa)$



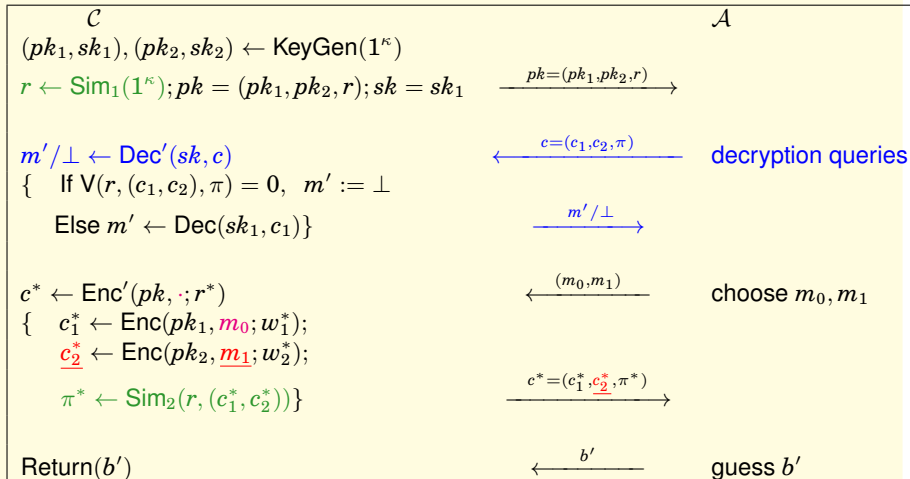
Game 1: $|\Pr[\text{Game 1} = 1] - \Pr[\text{Game 0} = 1]| = \text{Adv}_{a\text{NIZK}}^{\text{ZK}}(\kappa)$ (adaptive ZK).



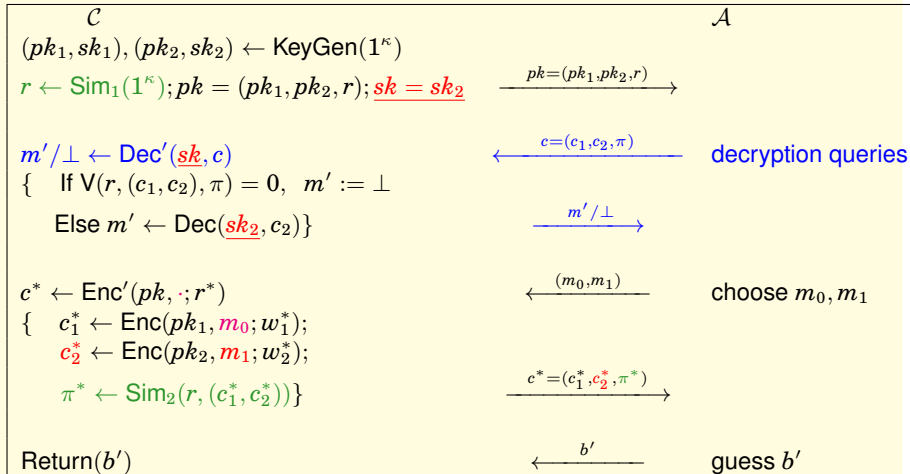
$$|\Pr[\text{Game 1} = 1] - \Pr[\text{Game 0} = 1]| = \text{Adv}_{a\text{NIZK}}^{\text{ZK}}(\kappa) \text{ (adaptive ZK).}$$



Game 2: $|\Pr[\text{Game 2} = 1] - \Pr[\text{Game 1} = 1]| \leq \text{Adv}_{\text{PKE}}^{\text{cpa}}(\kappa).$



Game 3: $|\Pr[\text{Game 3} = 1] - \Pr[\text{Game 2} = 1]| \leq \text{Adv}_{a\text{NIZK}}^{\text{ZK}}(\kappa) + \text{Adv}_{a\text{NIZK}}^{\text{sound}}(\kappa).$



Fake_i: the event that \mathcal{A} submits (c_1, c_2, π) in Game i to the decryption oracle with

$$(\text{Dec}(sk_1, c_1) \neq \text{Dec}(sk_2, c_2)) \wedge (V(r, (c_1; c_2), \pi) = 1).$$

- $\Pr[\text{Fake}_3] = \Pr[\text{Fake}_2]$;
- $\text{Game } 3 | \neg \text{Fake}_3 = \text{Game } 2 | \neg \text{Fake}_2$.
- So $|\Pr[\text{Game } 3 = 1] - \Pr[\text{Game } 2 = 1]| \leq \Pr[\text{Fake}_2]$.

Lemma (Shoup, Difference Lemma)

Let A, B, C be events. If $\Pr[A | \neg C] = \Pr[B | \neg C]$, then $|\Pr[A] - \Pr[B]| \leq \Pr[C]$.

Proof.

$$\Pr[A] = \Pr[A \wedge C] + \Pr[A \wedge \neg C].$$

$$\Pr[B] = \Pr[B \wedge C] + \Pr[B \wedge \neg C].$$

$$\Pr[A] - \Pr[B] = \Pr[A \wedge C] - \Pr[B \wedge C].$$

$$|\Pr[A] - \Pr[B]| = |\Pr[A \wedge C] - \Pr[B \wedge C]| = |(\Pr[A|C] - \Pr[B|C]) \Pr[C]| \leq \Pr[C].$$

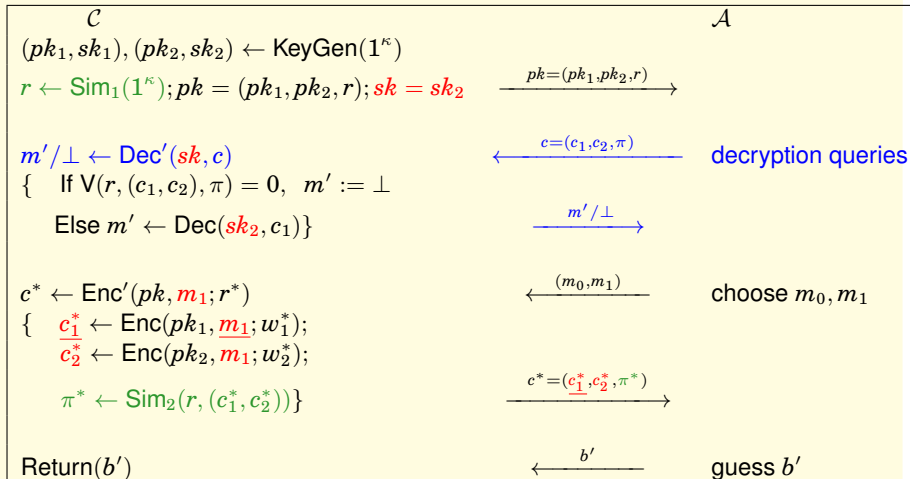
□

- $|\Pr[\text{Game 3} = 1] - \Pr[\text{Game 2} = 1]| \leq \Pr[\text{Fake}_2]$.
- $\Pr[\text{Fake}_2] = \Pr[\text{Fake}_1]$. \mathcal{A} has the same view before the challenge phase in both Game 1 and Game 2.
- $|\Pr[\text{Fake}_1] - \Pr[\text{Fake}_0]| \leq \text{Adv}_{a\text{NIZK}}^{\text{ZK}}(\kappa)$.
- $\Pr[\text{Fake}_0] = \text{Adv}_{a\text{NIZK}}^{\text{sound}}(\kappa)$.

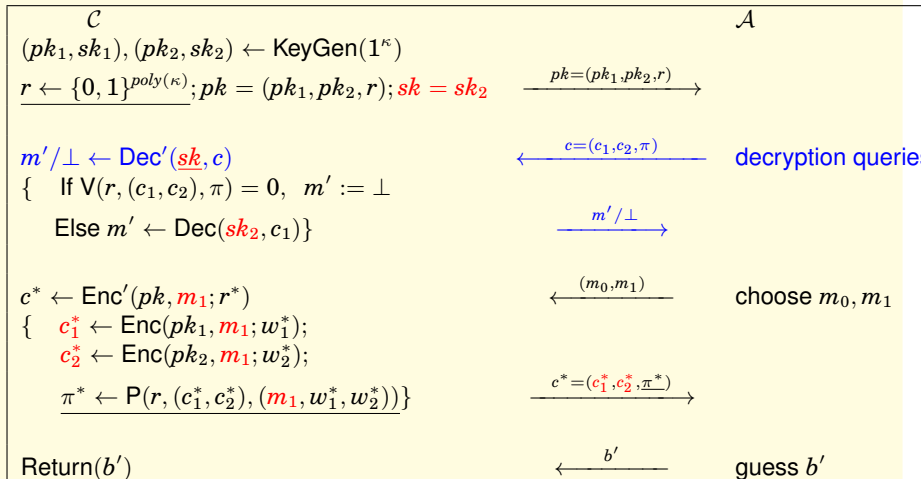
Hence

$$|\Pr[\text{Game 3} = 1] - \Pr[\text{Game 2} = 1]| \leq \Pr[\text{Fake}_2] \leq \text{Adv}_{a\text{NIZK}}^{\text{ZK}}(\kappa) + \text{Adv}_{a\text{NIZK}}^{\text{sound}}(\kappa).$$

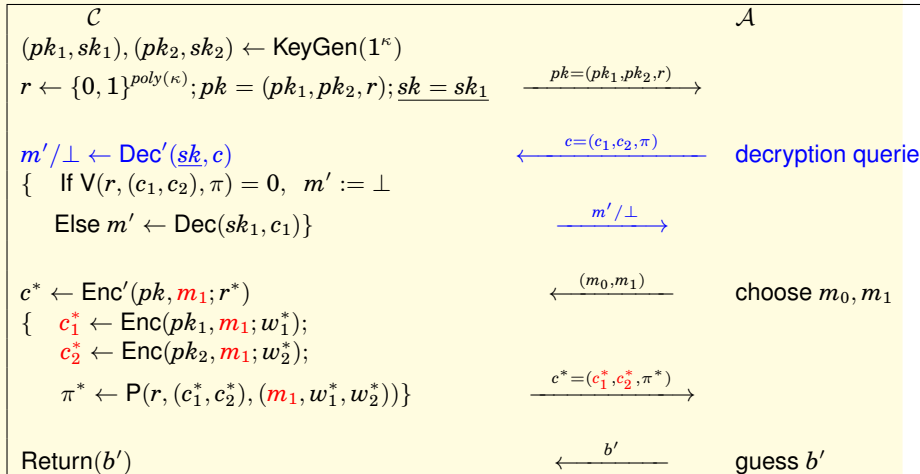
Game 4: $|\Pr[\text{Game 4} = 1] - \Pr[\text{Game 3} = 1]| \leq \text{Adv}_{\text{PKE}}^{\text{CPA}}(\kappa).$



Game 5: $|\Pr[\text{Game 5} = 1] - \Pr[\text{Game 4} = 1]| \leq \text{Adv}_{a\text{NIZK}}^{\text{ZK}}(\kappa).$



Game 6: $|\Pr[\text{Game 6} = 1] - \Pr[\text{Game 5} = 1]| \leq \text{Adv}_{a\text{NIZK}}^{\text{sound}}(\kappa).$



$$\begin{aligned}
& \left| \Pr \left[\text{Exp}_{PKE, \mathcal{A}}^{CCA}(1) = 1 \right] - \Pr \left[\text{Exp}_{PKE, \mathcal{A}}^{CCA}(0) = 1 \right] \right| \\
&= \left| \Pr [\text{Game 6} = 1] - \Pr [\text{Game 0} = 1] \right| \\
&\leq 3\text{Adv}_{aNIZK}^{ZK}(\kappa) + 2\text{Adv}_{PKE}^{CPA}(\kappa) + 2\text{Adv}_{aNIZK}^{sound}(\kappa) = \text{negl}(\kappa).
\end{aligned}$$

Theorem

The Noar-Yung scheme PKE' is NOT secure against adaptive chosen- ciphertext attacks (in general). More precisely, for any semantically-secure encryption scheme $PKE = (\text{KeyGen}, \text{Enc}, \text{Dec})$ there exists an adaptively-secure NIZK proof system (P', V') such that the resulting Noar-Yung construction is demonstrably insecure against adaptive chosen- ciphertext attacks.

Proof.

Let (P, V) be an aNIZK used in Noar-Yung scheme. Then (P', V') is also an aNIZK.

- $P'(r, (c_1, c_2), (m, w_1, w_2))$: Return $P(r, (c_1, c_2), (m, w_1, w_2)) || 0$.
- $V'(r, (c_1, c_2), \pi || 0)$: Return $V(r, (c_1, c_2), \pi)$.

If (P', V') is used in PKE', then \mathcal{A} can always submit $(c_1^*, c_2^*, \pi^* || 1)$ to the decryption oracle and succeed with probability 1.