FO - Transform

KeyGen $\leftarrow$ PKE's KeyGen

$Enc(pk, m \in \mathcal{M}^{sym})$:

$\sigma \xleftarrow{\$} \mathcal{M}^{asym}$, $a \leftarrow G(\sigma)$, $c \xleftarrow{\$} E^{sym}(a, m)$

$h \leftarrow H(\sigma, c)$

$e \leftarrow E^{asym}(pk^{hy}, \sigma, h)$

output (e, c)

$Dec(sk, (e, c))$:

$\hat{\sigma} \leftarrow D^{asym}(sk^{hy}, e)$, $\hat{h} \leftarrow H(\hat{\sigma}, c)$

$\hat{e} \leftarrow E^{asym}(pk^{hy}, \hat{\sigma}, \hat{h})$

if $\hat{e} \neq e$:

output $\perp$

$\hat{a} \leftarrow G(\hat{\sigma})$

$\hat{m} \leftarrow D^{sym}(\hat{a}, c)$, output $\hat{m}$

G_0: IND-CCA2

G_1: G_0 but $\mathcal{O}_1^D$ checks tape of H:

$\exists (\sigma, c, h) \in Tape(H) \mid c = \tilde{c}, E(pk, \sigma, h) = \tilde{e}$

$$\boxed{P[G_0] - P[G_1] \leq P[\text{at least one "almost"}] \leq q_D \cdot 2^{-\gamma}}$$

G_2: same as G_1, but $a^* \xleftarrow{\$} \mathcal{K}^{sym}$, $h^* \xleftarrow{\$} Coin^{asym}$

G_1 and G_2 behaves differently iff $\mathcal{A}^{hy}_{cca}$ queries σ^* from H or G: $\boxed{P[G_0] - P[G_2] \leq P[\text{queries } \sigma^*]}$

$\mathcal{A}^{sym}_{IND\text{-}OTE}$: $\mathcal{A}^{sym}$ generates (pk^{hy}, sk^{hy}) and simulates both the hash oracle and $\mathcal{O}_2^D$

when $(m_0, m_1) \leftarrow \mathcal{A}^{hy}_{cca}$, $\mathcal{A}^{sym}$ submits (m_0, m_1) to the symmetric ~~Oracle~~ challenger and receives C^*;

$\mathcal{A}^{sym}$ samples σ^*, h^*, computes $e^* \leftarrow E^{asym}(pk, \sigma^*, h^*)$

given (e^*, C^*) to $\mathcal{A}^{hy}_{cca}$, then passes $b \leftarrow \mathcal{A}^{hy}_{cca}$

$$\boxed{P[G_2] = P[\mathcal{A}^{sym} \text{ wins IND-OTE}]}$$

$\mathcal{A}_{ow}$: $\mathcal{A}_{ow}$ receives pk and passes to $\mathcal{A}^{hy}_{cca}$

simulates $\mathcal{O}^G$, $\mathcal{O}^H$, $\mathcal{O}_1^D$

receives e^*, ~~samples~~ ~~σ^*~~, ~~h^*~~ ~~randomly~~ ~~σ^*~~

randomly samples a^*, $c^* \leftarrow E(a^*, m_b)$

picks a random value to return as $\hat{\sigma}$

$\mathcal{A}_{OW}$ wins if "$\mathcal{A}_{CCA}$ made query $(c^*, \cdot)$" and "the chosen value on tape is σ^*"

then $$\boxed{P[\mathcal{A}_{OW\text{-}CPA} \text{ wins}] = P[\mathcal{A}_2^{hy} \text{ queries } \sigma^*] \cdot \frac{1}{q_H}}$$

putting everything together:

$$\boxed{\varepsilon^{hy}_{CCA2} \leq q_D \cdot 2^{-\gamma} + \varepsilon^{sym}_{IND\text{-}CTE} + q_H \cdot \varepsilon^{asym}_{OW\text{-}CPA}}$$

Problems:

- $q_H \cdot \varepsilon^{asym}_{OW\text{-}CPA}$ is not tight
- we want a KEM, which carries different security requirement
- assumes there is never decryption error

PKE^T:

$E^T(pk, m)$
- $r \leftarrow G(m)$
- $c \leftarrow E(pk, m, r)$

output c

$D^T(sk, c)$
- $\hat{m} \leftarrow D(sk, c)$
- $\hat{r} \leftarrow G(\hat{m})$
- $\hat{c} \leftarrow E(pk, \hat{m}, \hat{r})$

if $\hat{c} == c$:
  output $\hat{m}$
else:
  output $\perp$

$CVO(\tilde{c})$:
- $\hat{m} \leftarrow D(sk, \tilde{c})$
- assert $E(pk, \hat{m}, G(\hat{m})) == \tilde{c}$

$PCO(\tilde{m}, \tilde{c})$
- $\hat{m} \leftarrow D(sk, \tilde{c})$

assert $\hat{m} == \tilde{m}$
assert $E(\tilde{m}, G(\tilde{m})) == \tilde{c}$

$CVO_1(\tilde{c})$:
check $\exists? (\tilde{m}, \tilde{r}) \in \mathcal{Q}^G$
s.t. $E(pk, \tilde{m}, \tilde{r}) == \tilde{c}$
assert $\tilde{m} == D(sk, \tilde{c})$

$CVO_2(\tilde{c})$
check $\exists? (\tilde{m}, \tilde{r}) \in \mathcal{Q}^G$
s.t. $E(pk, \tilde{m}, \tilde{r}) = \tilde{c}$

$PCO_2(\tilde{m}, \tilde{c})$
check $E(\tilde{m}, G(\tilde{m})) = \tilde{c}$

G_0: OW-PCVA

Game 1: replace CVO with CVO_1, differ if A_{PCVA} submits at least 1 CV query $\tilde{c}$ such that $E(\hat{m}, G(\hat{m})) = \tilde{c}$ but $\hat{m}$ is not typed.

$$P[S_0] - P[S_1] \leq P[1 \text{ "almost"}] \leq q_{CV} \cdot 2^{-\gamma}$$

Game 2: use CVO_2 and PCO_2. Differ from game 1 when $(\tilde{m}, \tilde{c})$ typed causes decryption error:

$$P[S_1] - P[S_2] \leq q_G \cdot \delta$$

Game 3: when encrypting challenge c^*, replace $r^* \leftarrow G(m^*)$ with a random $r^* \leftarrow$ coin. Differs from game 2 when A_{PCVA} makes m^* to $\mathcal{Q}^G$

$$P[S_2] - P[S_3] \leq P[\underset{m^*}{\text{queries}}]$$

$P[S_3]$ can be bounded by a OW-CPA attack $\mathcal{A}_{OW\text{-}CPA}$:

$$P[S_3] = P[\mathcal{A}_{OW\text{-}CPA} \text{ wins}]$$

$P[\text{queries } m^*]$ is bounded by another OW-CPA attack (which returns a random value from $\mathcal{OG}'$s tape)

$$P[\text{query } m^*] \cdot \frac{1}{q_G} = P[\mathcal{A}'_{OW\text{-}CPA} \text{ wins}]$$

putting all together:

$$\boxed{P[S_0] \leq q_V \cdot 2^{-\gamma} + q_G \cdot \delta + (q_G + 1) \cdot P[\mathcal{A}_{OW\text{-}CPA} \text{ wins}]}$$

But then again this is not tight because of the last term.

To tighten security, recall that:

$$P[S_0] \leq q_V 2^{-\delta} + q_G \delta + P[\mathcal{A}_{OW\text{-}PCVA} \text{ query } m^*] + P[\mathcal{A}_{OW\text{-}CPA}]$$

there is a lemma (IND-CPA implies IND-OW): let $\mathcal{A}_{OW}$ uses $\mathcal{A}_{IND}$ as sub-routine. When $\mathcal{A}_{OW}$ receives C^*, it randomly samples (m_0, m_1)

$$\boxed{\varepsilon_{OW\text{-}CPA} \leq \varepsilon_{IND\text{-}CPA} + \frac{1}{|\mathcal{M}|}} \quad \text{WHY?}$$

we can also bound $P[\mathcal{A}_{OW\text{-}PCVA} \text{ queries } m^*]$ by building an IND-CPA $\mathcal{A}$ that uses $\mathcal{A}_{OW\text{-}PCVA}$ as a sub-routines. $\mathcal{A}_{IND\text{-}CPA}$ simulates $\mathcal{OG}$, PCO_2, CVO_2, and passes pk and C^* (C^* is from (m_0, m_1) where m_0, m_1 both truly random) to $\mathcal{A}_{OW\text{-}PCVA}$. $\mathcal{A}_{IND\text{-}CPA}$ determines $\hat{b}$ by checking if m_0, m_1 is on the tape of $\mathcal{OG}$:

(this is not quite right):

$$P[\mathcal{A}_{IND\text{-}CPA} \text{ wins}] \geq P[\mathcal{A}_{PCVA} \text{ queries } m^*] + \tfrac{1}{2} P[\text{no query}]$$

$$= \tfrac{1}{2} + \tfrac{1}{2} P[\mathcal{A}_{OW\text{-}PVCA} \text{ queries } m^*]$$

so $\boxed{P[\text{query}] \leq 2\varepsilon_{IND\text{-}CPA}}$

KEM-02

$U^{\perp}$: with explicit rejection
with probabilistic, IND-CPA PKE
↳ transform into
a OW-PCVA PKE.

Encap: (pk)
- $m \xleftarrow{\$} \mathcal{M}$
- $c \leftarrow PKE_1.encrypt(pk, m)$
- $ss \leftarrow H(m, c)$

Decap (sk, c)
- $\hat{m} \leftarrow PKE_1.decrypt(sk, c)$
- if $\hat{m} \neq \perp$:
  output $H(m, c)$
  else:
  output $\perp$

Game 0: IND-CCA-KEM
A_{KEM} has pk, $\mathcal{O}^D$, ct^*, ss^*
Game 1:

what if we want $A_{OW\text{-}PCVA}$?
$\Rightarrow pk$, PCO, CVO, C^*

① $A_{OW\text{-}PCVA}$ can be used to simulate H and $\mathcal{O}^D$ under the RO model:
For query $(\tilde{m}, \tilde{c})$ to H:
- sample random $\tilde{K}$ or from tape H
  if $(\tilde{m}, \tilde{c})$ in fresh, then add $(\tilde{c}, \tilde{K})$ to tape $\mathcal{O}^D$
For query $\tilde{c}$ to $\mathcal{O}^D$
- Use CVO to check $\tilde{c}$ ⇒ if invalid, output $\perp$
- check the tape $\mathcal{O}^D$:
  if $\tilde{c} \in Tape(\mathcal{O}^D)$ then return the corresponding $\tilde{K}$
  else sample random $\tilde{K}$

② $A_{OW\text{-}PCVA}$ samples a random value for ss^* and given to A_{KEM}

Game 0:
- receives pk
- access $\mathcal{O}^H$ and $\mathcal{O}^D$
- receives C^* and K^*
- outputs guess " in K^* truly random" ?

Game 1:
- receives pk
- access $\mathcal{O}_1^H$, $\mathcal{O}_1^D$
  $\mathcal{O}_1^H$ checks PCO and CVO and updates $Tape(\mathcal{O}_1^D)$
  $\mathcal{O}_1^D$ checks tape and CVO otherwise outputs $\tilde{K} \xleftarrow{\$} \{0,1\}^n$

Game 0 and game 1 are identical

~~Game 2:~~
- ~~receiven pk~~
- ~~accesn $\mathcal{O}_1^H$, $\mathcal{O}_1^D$~~
- ~~receiven c^*, K^* but K^* is always random~~

~~$P[\text{win Game 2}] = \frac{1}{2}$~~

Game 2:
- receiven pk
- accesn $\mathcal{O}_2^H$, $\mathcal{O}_1^D$

$\mathcal{O}_2^H(\tilde{m}, \tilde{c})$: return $PCO(\tilde{c})$, ~~[illegible]~~ ($\tilde{m}$ → PCO)
if $\tilde{c} = c^*$: return $\tilde{K} \xleftarrow{\$} \{0,1\}$

⇒ the information about K^* becomes truly random (K^* originally depends on (m^*, c^*) but in G_2 when A_{KEM} queries (m^*, c^*) a truly random value is given)

if A_{KEM} made query (m^*, c^*) to $\mathcal{O}_2^H$, then $A_{\text{OW-PCVA}}$ can check if $\exists (\tilde{m}, \tilde{c}, \tilde{K}) \in \mathcal{O}_2^H$ s.t. $\tilde{c} = c^*$, then return $\tilde{m}$ as the guess. Thanks to $\mathcal{O}_2^H$ using PCO, we know $\tilde{m}$ will be the correct guess.