Simulation-Based Bi-Selective Opening Security for Public Key Encryption

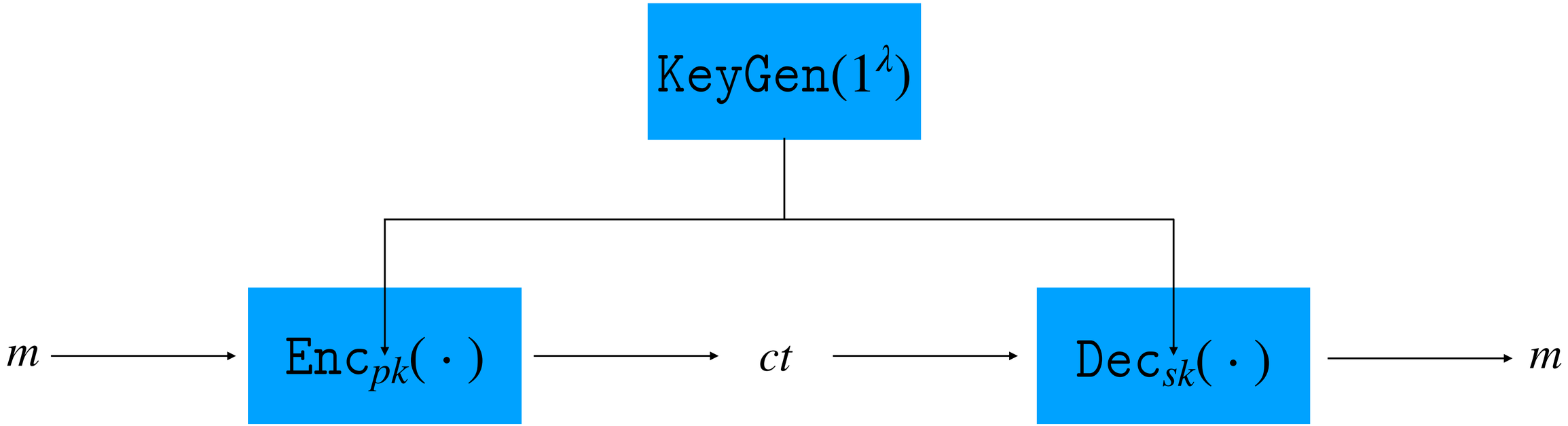
Junzuo Lai
Jinan University

Rupeng Yang
University of Hong Kong

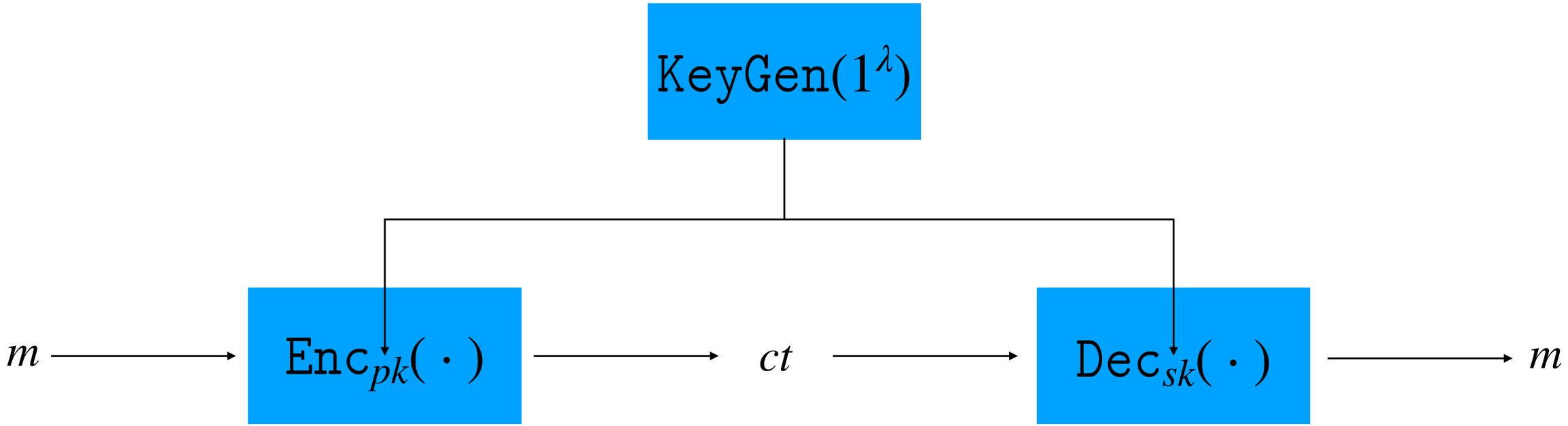
Zhengan Huang
Peng Cheng Laboratory

Jian Weng
Jinan University

PKE with Bi-Selective Opening Security



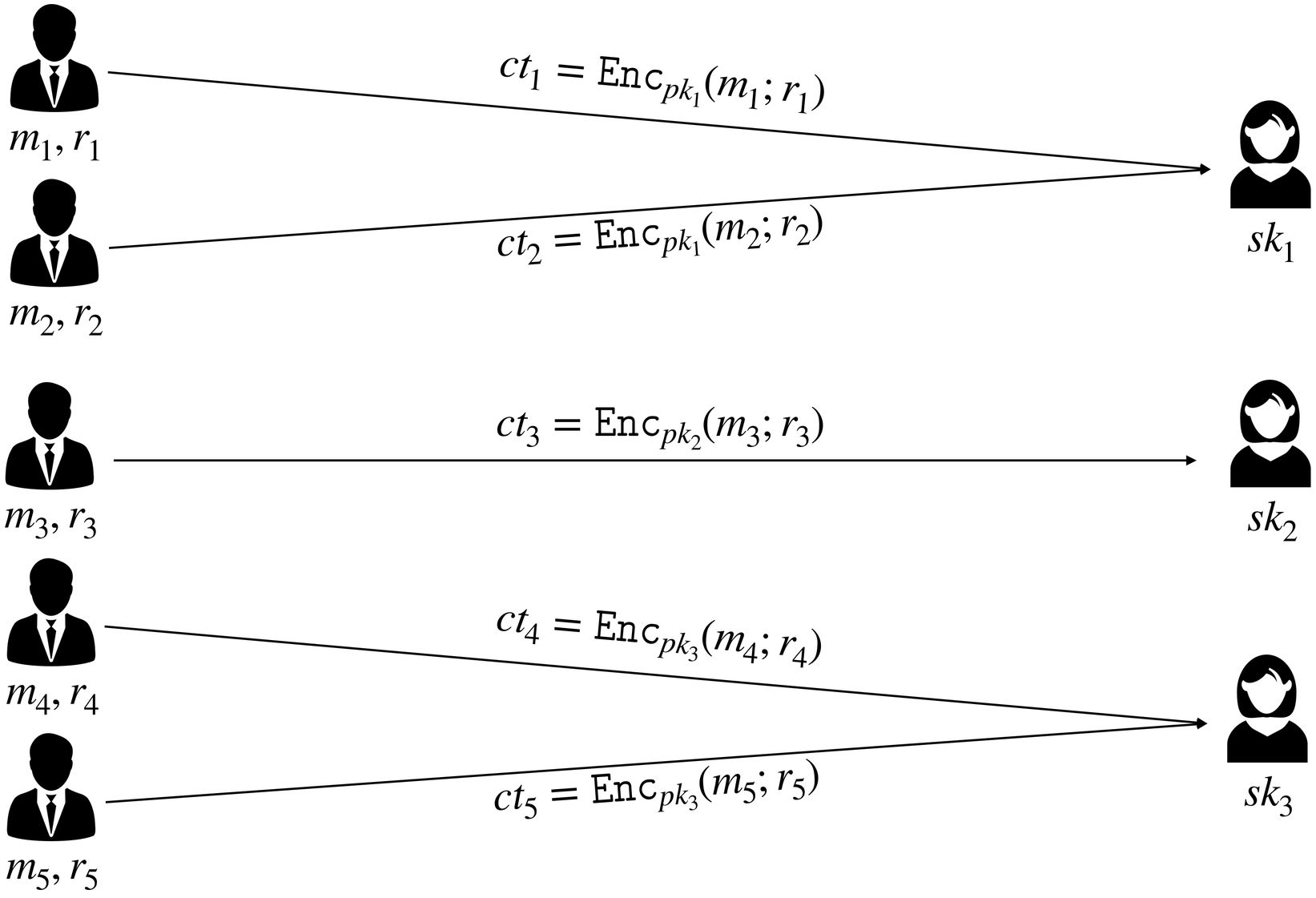
PKE with Bi-Selective Opening Security



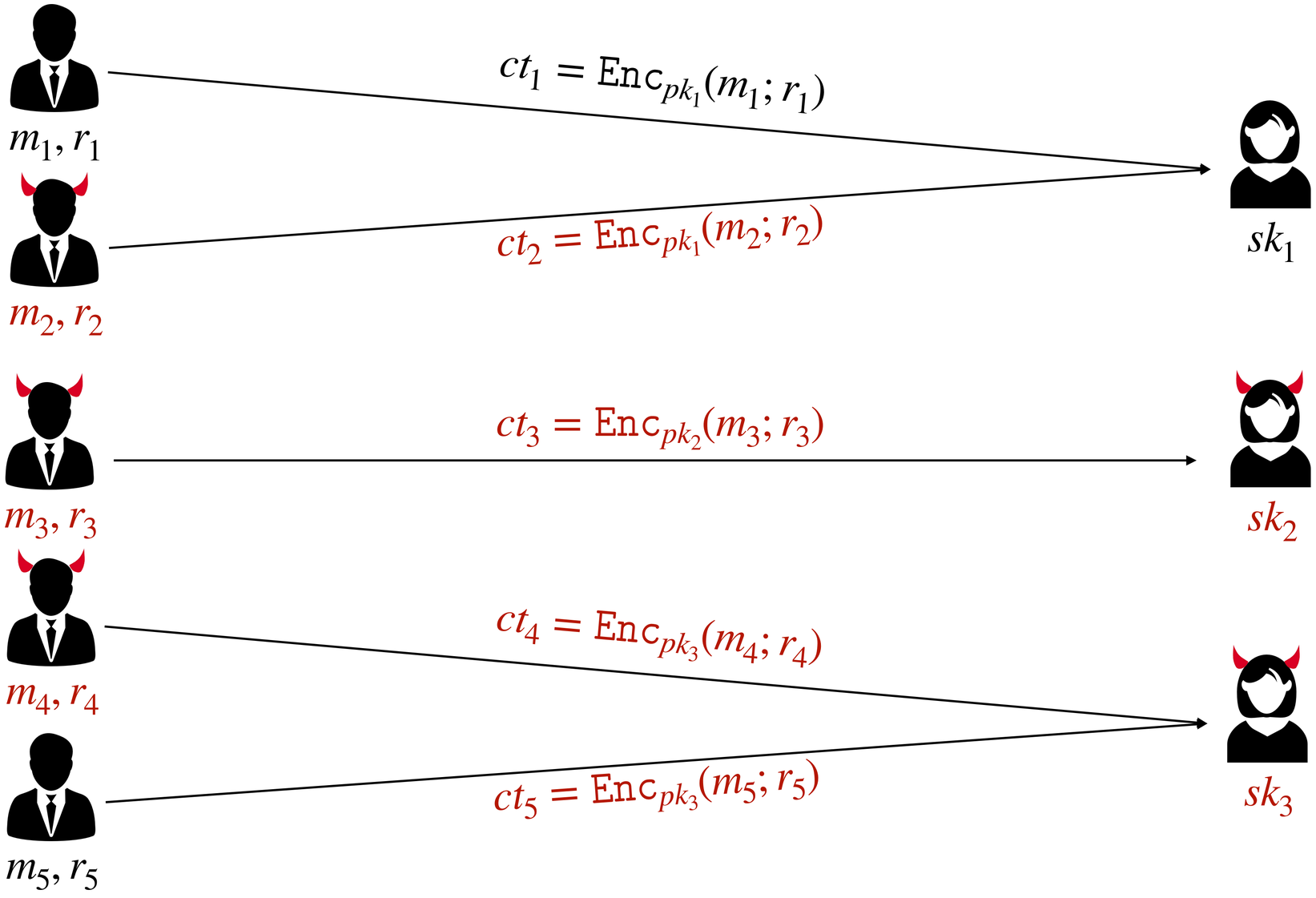
Correctness: $\mathtt{Dec}_{sk}(\mathtt{Enc}_{pk}(m)) = m$

Semantic Security: $\mathcal{A}(pk, ct) \approx \mathcal{S}(1^\lambda)$

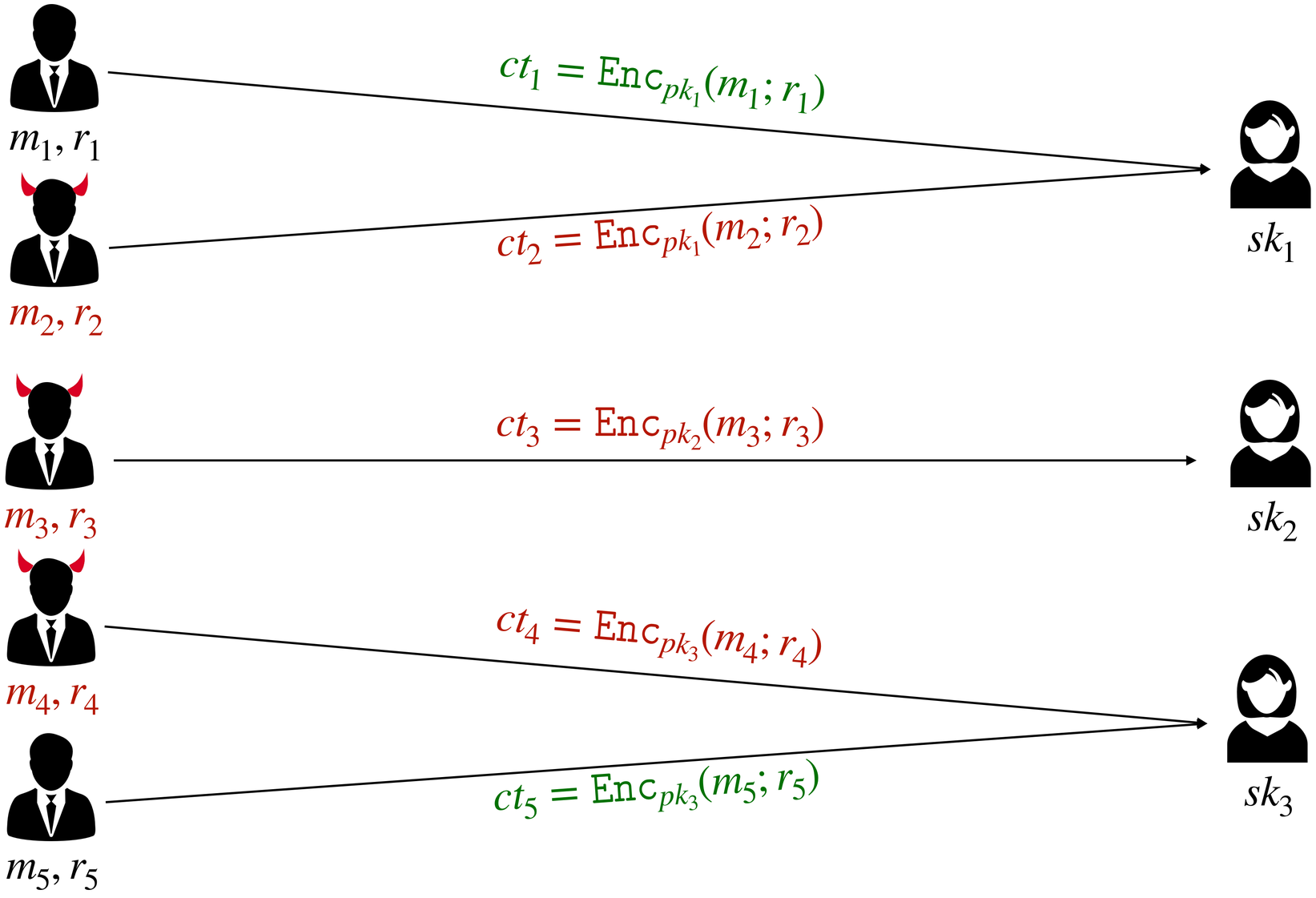
PKE with Bi-Selective Opening Security



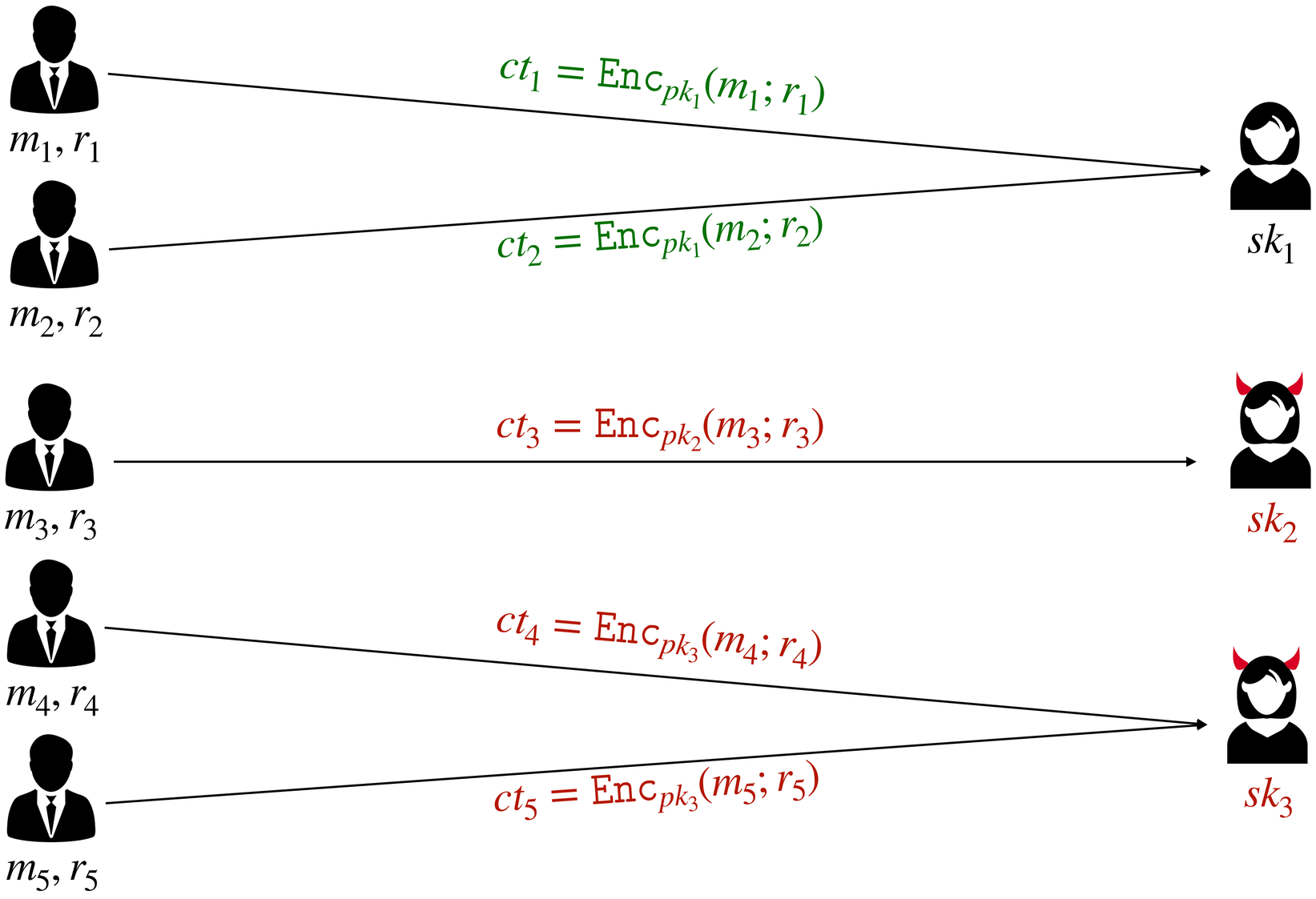
PKE with Bi-Selective Opening Security



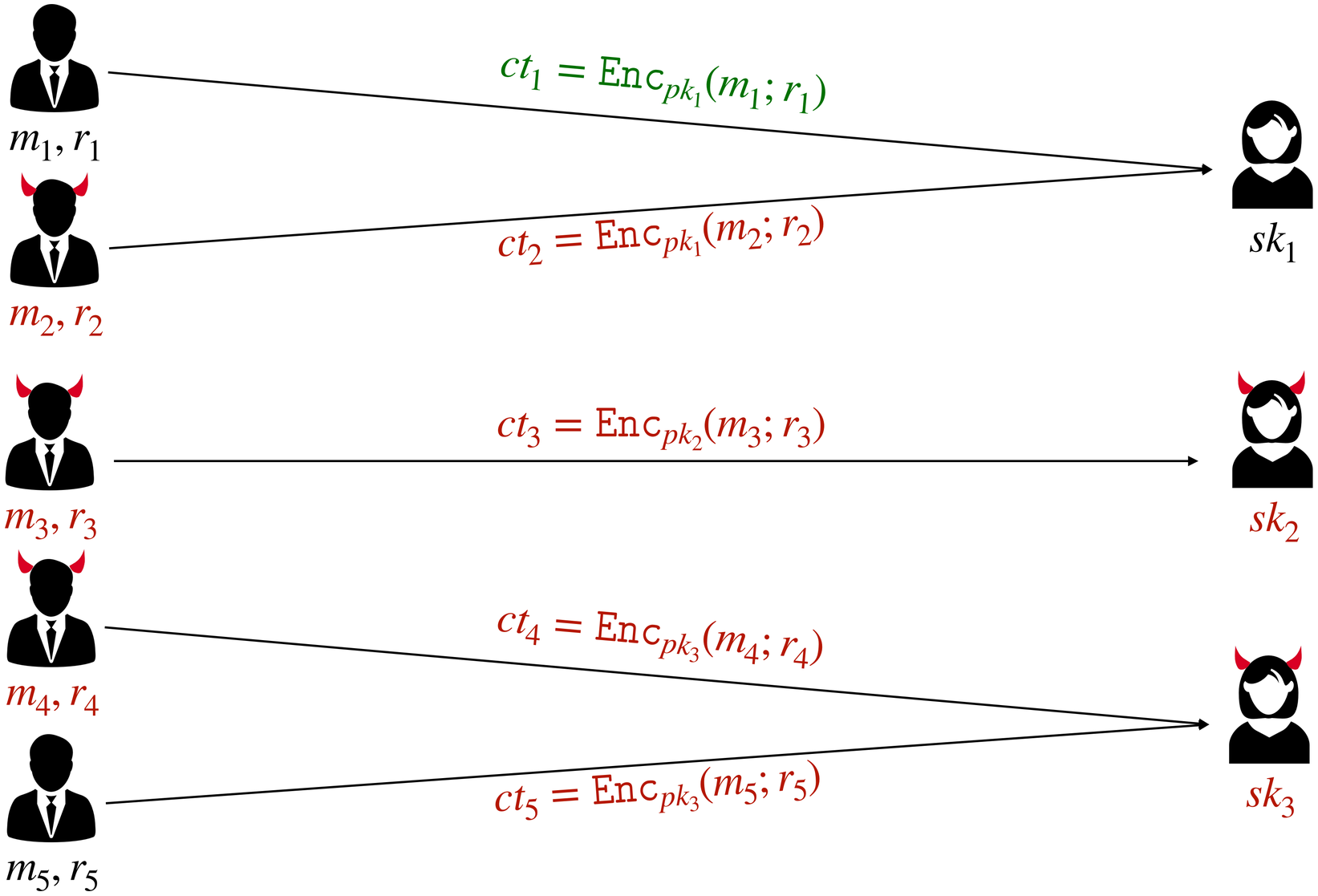
PKE with Sender-Selective Opening Security



PKE with Receiver-Selective Opening Security



PKE with Bi-Selective Opening Security



Our Results

- Define Bi-Selective Opening Security for PKE Schemes
- Define Weak Bi-Selective Opening Security for PKE Schemes
- Construct PKE with Bi-Selective Opening Security in the Random Oracle Model
- Construct PKE with Weak Bi-Selective Opening Security in the Plain Model

Our Results

- Define Bi-Selective Opening Security for PKE Schemes

- Construct PKE with Bi-Selective Opening Security in the Random Oracle Model

- Define Weak Bi-Selective Opening Security for PKE Schemes

- Construct PKE with Weak Bi-Selective Opening Security in the Plain Model

Definition: Bi-Selective Opening Security

$\mathscr{M} \leftarrow \mathscr{A}(pk_1, \ldots, pk_n)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathscr{M}$

For $i \in [n], j \in [k]$:

$ct_{i,j} \leftarrow Enc_{pk_i}(m_{i,j}; r_{i,j})$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathscr{A}(ct_{1,1}, \ldots, ct_{n,k})$

$\mathsf{Open} = \emptyset$

For $(i, j) \in \mathtt{I}_S$:

$\mathsf{Open} = \mathsf{Open} \cup \{(m_{i,j}, r_{i,j})\}$

For $i \in \mathtt{I}_R$:

$\mathsf{Open} = \mathsf{Open} \cup \{((m_{i,j})_{j\in[k]}, sk_i)\}$

$out \leftarrow \mathscr{A}(\mathsf{Open})$

$(\mathscr{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$

Definition: Bi-Selective Opening Security

$\mathscr{M} \leftarrow \mathscr{A}(pk_1, \ldots, pk_n)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathscr{M}$

For $i \in [n], j \in [k]$:

$ct_{i,j} \leftarrow Enc_{pk_i}(m_{i,j}; r_{i,j})$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathscr{A}(ct_{1,1}, \ldots, ct_{n,k})$

$\mathsf{Open} = \emptyset$

For $(i, j) \in \mathtt{I}_S$:

$\mathsf{Open} = \mathsf{Open} \cup \{(m_{i,j}, r_{i,j})\}$

For $i \in \mathtt{I}_R$:

$\mathsf{Open} = \mathsf{Open} \cup \{((m_{i,j})_{j\in[k]}, sk_i)\}$

$out \leftarrow \mathscr{A}(\mathsf{Open})$

$(\mathscr{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$

$\approx$

$\mathscr{M} \leftarrow \mathcal{S}(1^\lambda)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathscr{M}$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathcal{S}(1^\lambda)$

$out \leftarrow \mathcal{S}((m_{i,j})_{i\in\mathtt{I}_R,j\in[k]}, (m_{i,j})_{(i,j)\in\mathtt{I}_S})$

$(\mathscr{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$

Definition: Bi-Selective Opening CCA Security

$\mathscr{M} \leftarrow \mathscr{A}^{\mathtt{Dec}}(pk_1, \ldots, pk_n)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathscr{M}$

For $i \in [n], j \in [k]$:

$ct_{i,j} \leftarrow Enc_{pk_i}(m_{i,j}; r_{i,j})$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathscr{A}^{\mathtt{Dec}}(ct_{1,1}, \ldots, ct_{n,k})$

$\mathsf{Open} = \emptyset$

For $(i, j) \in \mathtt{I}_S$:

$\mathsf{Open} = \mathsf{Open} \cup \{(m_{i,j}, r_{i,j})\}$

For $i \in \mathtt{I}_R$:

$\mathsf{Open} = \mathsf{Open} \cup \{((m_{i,j})_{j\in[k]}, sk_i)\}$

$out \leftarrow \mathscr{A}^{\mathtt{Dec}}(\mathsf{Open})$

$(\mathscr{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$

$\approx$

$\mathscr{M} \leftarrow \mathcal{S}(1^\lambda)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathscr{M}$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathcal{S}(1^\lambda)$

$out \leftarrow \mathcal{S}((m_{i,j})_{i\in\mathtt{I}_R,j\in[k]}, (m_{i,j})_{(i,j)\in\mathtt{I}_S})$

$(\mathscr{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$

$\mathtt{Dec}(i, ct)$:

If $(i > n) \vee \exists j \in [k] ct_{i,j} = ct$: **Return** $\perp$

Return $Dec(sk_i, ct)$

Our Results

- Define Bi-Selective Opening Security for PKE Schemes
- Define Weak Bi-Selective Opening Security for PKE Schemes
- Construct PKE with Bi-Selective Opening Security in the Random Oracle Model
- Construct PKE with Weak Bi-Selective Opening Security in the Plain Model

Construction: Bi-Selective Opening Secure PKE

$\mathsf{KEM} = (\mathsf{KEM}.\mathtt{KeyGen}, \mathsf{KEM}.\mathtt{Enc}, \mathsf{KEM}.\mathtt{Dec})$
is a secure key encapsulation mechanism.

H is a hash function modelled as a random oracle

Construction: Bi-Selective Opening Secure PKE

$\mathsf{KEM} = (\mathsf{KEM}.\mathtt{KeyGen}, \mathsf{KEM}.\mathtt{Enc}, \mathsf{KEM}.\mathtt{Dec})$
is a secure key encapsulation mechanism.

H is a hash function modelled as a random oracle

$\mathtt{KeyGen}(1^\lambda)$:
$(pk, sk) \leftarrow \mathsf{KEM}.\mathtt{KeyGen}(1^\lambda)$
$SK = sk$
$PK = pk$

Construction: Bi-Selective Opening Secure PKE

$\mathsf{KEM} = (\mathsf{KEM}.\mathtt{KeyGen}, \mathsf{KEM}.\mathtt{Enc}, \mathsf{KEM}.\mathtt{Dec})$
is a secure key encapsulation mechanism.

H is a hash function modelled as a random oracle

$\mathtt{KeyGen}(1^\lambda)$:
$(pk, sk) \leftarrow \mathsf{KEM}.\mathtt{KeyGen}(1^\lambda)$
$SK = sk$
$PK = pk$

$\mathtt{Enc}(PK = pk, m)$:
$r \leftarrow \mathscr{R}$
$(k, c_1) \leftarrow \mathsf{KEM}.\mathtt{Enc}(pk; r)$
$K = \mathsf{H}(k)$
$c_2 = K \oplus m$
$CT = (c_1, c_2)$

Construction: Bi-Selective Opening Secure PKE

$\mathsf{KEM} = (\mathsf{KEM}.\mathtt{KeyGen}, \mathsf{KEM}.\mathtt{Enc}, \mathsf{KEM}.\mathtt{Dec})$
is a secure key encapsulation mechanism.

H is a hash function modelled as a random oracle

$\mathtt{KeyGen}(1^\lambda)$:
$(pk, sk) \leftarrow \mathsf{KEM}.\mathtt{KeyGen}(1^\lambda)$
$SK = sk$
$PK = pk$

$\mathtt{Enc}(PK = pk, m)$:
$r \leftarrow \mathscr{R}$
$(k, c_1) \leftarrow \mathsf{KEM}.\mathtt{Enc}(pk; r)$
$K = \mathsf{H}(k)$
$c_2 = K \oplus m$
$CT = (c_1, c_2)$

$\mathtt{Dec}(SK = sk, CT = (c_1, c_2))$:
$\bar{k} \leftarrow \mathsf{KEM}.\mathtt{Dec}(sk, c_1)$
$\bar{K} = \mathsf{H}(\bar{k})$
$m = \bar{K} \oplus c_2$

Security of the Construction

$PK_1, \ldots, PK_n$

$\mathscr{M}$

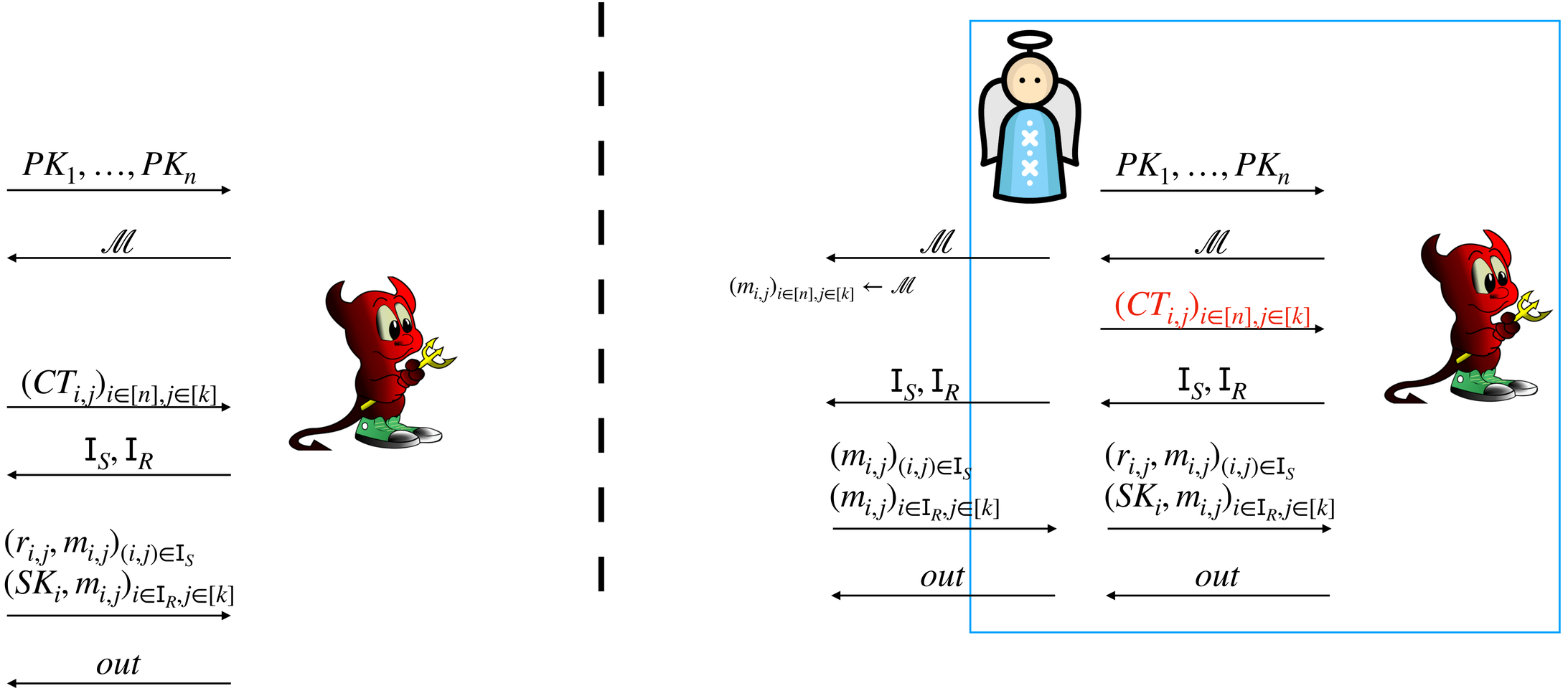
$(CT_{i,j})_{i\in[n],j\in[k]}$

$\mathtt{I}_S, \mathtt{I}_R$

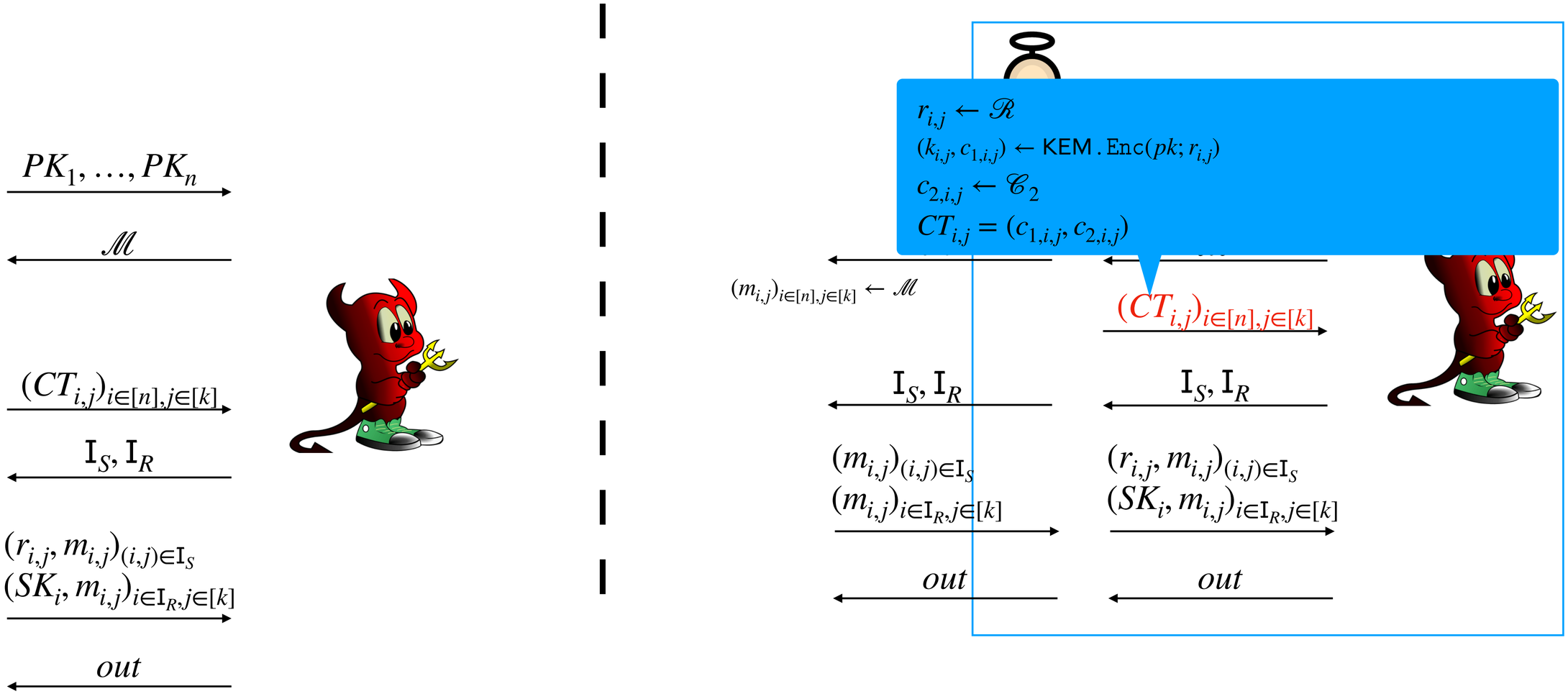
$(r_{i,j}, m_{i,j})_{(i,j)\in\mathtt{I}_S}$
$(SK_i, m_{i,j})_{i\in\mathtt{I}_R,j\in[k]}$

out

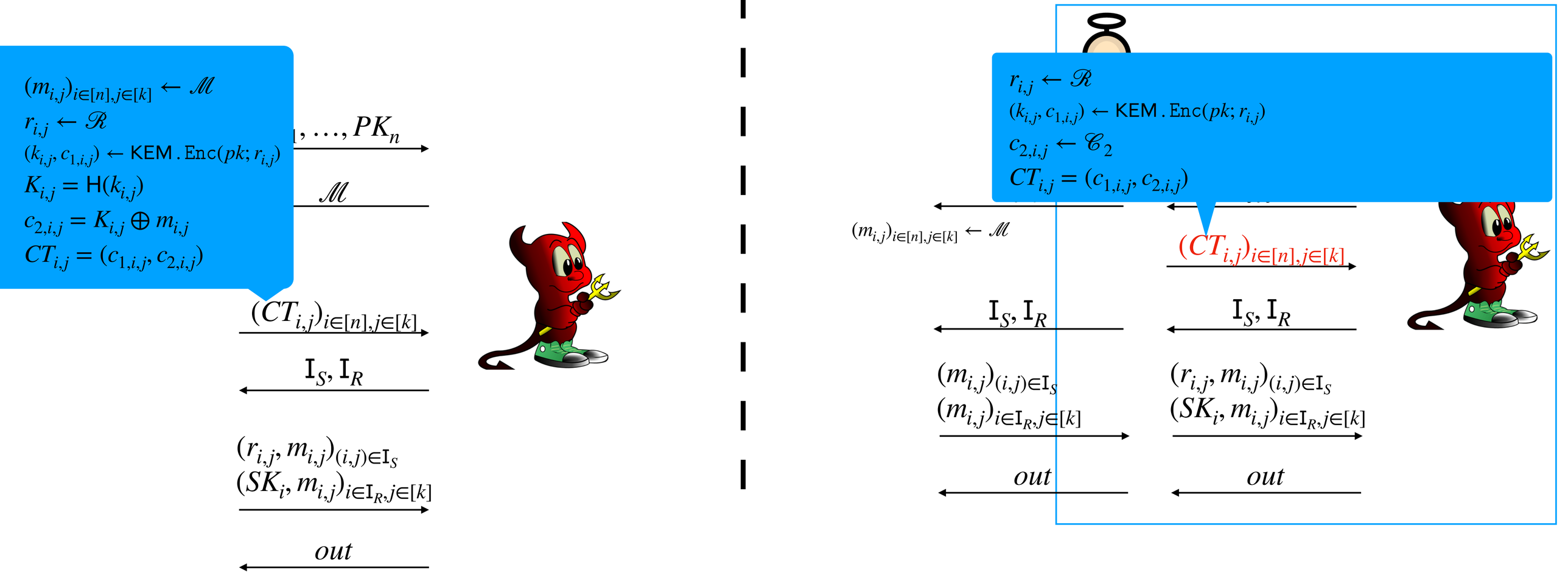
Security of the Construction



Security of the Construction



Security of the Construction



Security of the Construction

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathscr{M}$
$r_{i,j} \leftarrow \mathscr{R}$
$(k_{i,j}, c_{1,i,j}) \leftarrow \mathsf{KEM}.\mathtt{Enc}(pk; r_{i,j})$
$K_{i,j} = \mathsf{H}(k_{i,j})$
$c_{2,i,j} = K_{i,j} \oplus m_{i,j}$
$CT_{i,j} = (c_{1,i,j}, c_{2,i,j})$

$], \ldots, PK_n$

$\mathscr{M}$

$(CT_{i,j})_{i\in[n],j\in[k]}$

$\mathtt{I}_S, \mathtt{I}_R$

$(r_{i,j}, m_{i,j})_{(i,j)\in\mathtt{I}_S}$
$(SK_i, m_{i,j})_{i\in\mathtt{I}_R,j\in[k]}$

out

$r_{i,j} \leftarrow \mathscr{R}$
$(k_{i,j}, c_{1,i,j}) \leftarrow \mathsf{KEM}.\mathtt{Enc}(pk; r_{i,j})$
$c_{2,i,j} \leftarrow \mathscr{C}_2$
$CT_{i,j} = (c_{1,i,j}, c_{2,i,j})$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathscr{M}$

$(CT_{i,j})_{i\in[n],j\in[k]}$

$\mathtt{I}_S, \mathtt{I}_R$

$(m_{i,j})_{(i,j)\in\mathtt{I}_S}$
$(m_{i,j})_{i\in\mathtt{I}_R,j\in[k]}$

out

$\mathtt{I}_S, \mathtt{I}_R$

$(r_{i,j}, m_{i,j})_{(i,j)\in\mathtt{I}_S}$
$(SK_i, m_{i,j})_{i\in\mathtt{I}_R,j\in[k]}$

out

Program the random oracle H **:**
$\mathsf{H}(k_{i,j}) = m_{i,j} \oplus c_{2,i,j}$

Construction: Bi-Selective Opening Secure PKE

$\mathsf{KEM} = (\mathsf{KEM}.\mathtt{KeyGen}, \mathsf{KEM}.\mathtt{Enc}, \mathsf{KEM}.\mathtt{Dec})$
is a secure key encapsulation mechanism.

H is a hash function modelled as a random oracle

$\mathtt{KeyGen}(1^\lambda)$:
$(pk, sk) \leftarrow \mathsf{KEM}.\mathtt{KeyGen}(1^\lambda)$
$SK = sk$
$PK = pk$

$\mathtt{Enc}(PK = pk, m)$:
$r \leftarrow \mathscr{R}$
$(k, c_1) \leftarrow \mathsf{KEM}.\mathtt{Enc}(pk; r)$
$K = \mathsf{H}(k)$
$c_2 = K \oplus m$
$CT = (c_1, c_2)$

$\mathtt{Dec}(SK = sk, CT = (c_1, c_2))$:
$\bar{k} \leftarrow \mathsf{KEM}.\mathtt{Dec}(sk, c_1)$
$\bar{K} = \mathsf{H}(\bar{k})$
$m = \bar{K} \oplus c_2$

Achieving CCA-Security

https://eprint.iacr.org/2021/1268

Our Results

- Define Bi-Selective Opening Security for PKE Schemes
- Define Weak Bi-Selective Opening Security for PKE Schemes

- Construct PKE with Bi-Selective Opening Security in the Random Oracle Model
- Construct PKE with Weak Bi-Selective Opening Security in the Plain Model

Definition: Weak Bi-Selective Opening Security

$(\mathcal{M}, \beta) \leftarrow \mathcal{A}(pk_1, \ldots, pk_n)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathcal{M}$

For $i \in [n], j \in [k]$:

$ct_{i,j} \leftarrow Enc_{pk_i}(m_{i,j}; r_{i,j})$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathcal{A}(ct_{1,1}, \ldots, ct_{n,k})$

If $\beta =$ **"Sender"** : $\mathtt{I}_R = \emptyset$

If $\beta =$ **"Receiver"** : $\mathtt{I}_S = \emptyset$

$\mathsf{Open} = \emptyset$

For $(i,j) \in \mathtt{I}_S$:

$\mathsf{Open} = \mathsf{Open} \cup \{(m_{i,j}, r_{i,j})\}$

For $i \in \mathtt{I}_R$:

$\mathsf{Open} = \mathsf{Open} \cup \{((m_{i,j})_{j\in[k]}, sk_i)\}$

$out \leftarrow \mathcal{A}(\mathsf{Open})$

$(\mathcal{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$

Definition: Weak Bi-Selective Opening Security

$(\mathcal{M}, \beta) \leftarrow \mathcal{A}(pk_1, \ldots, pk_n)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathcal{M}$

For $i \in [n], j \in [k]$:

$ct_{i,j} \leftarrow Enc_{pk_i}(m_{i,j}; r_{i,j})$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathcal{A}(ct_{1,1}, \ldots, ct_{n,k})$

If $\beta =$ **"Sender"** : $\mathtt{I}_R = \emptyset$

If $\beta =$ **"Receiver"** : $\mathtt{I}_S = \emptyset$

$\mathsf{Open} = \emptyset$

For $(i,j) \in \mathtt{I}_S$:

$\mathsf{Open} = \mathsf{Open} \cup \{(m_{i,j}, r_{i,j})\}$

For $i \in \mathtt{I}_R$:

$\mathsf{Open} = \mathsf{Open} \cup \{((m_{i,j})_{j\in[k]}, sk_i)\}$

$out \leftarrow \mathcal{A}(\mathsf{Open})$

$(\mathcal{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$

$\approx$

If $\beta =$ **"Sender"** : $\mathtt{I}_R = \emptyset$

If $\beta =$ **"Receiver"** : $\mathtt{I}_S = \emptyset$

$(\mathcal{M}, \beta) \leftarrow \mathcal{S}(1^\lambda)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathcal{M}$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathcal{S}(\beta)$

$out \leftarrow \mathcal{S}((m_{i,j})_{i\in\mathtt{I}_R,j\in[k]}, (m_{i,j})_{(i,j)\in\mathtt{I}_S})$

$(\mathcal{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$

Definition: Weak Bi-Selective Opening CCA Security

$(\mathcal{M}, \beta) \leftarrow \mathcal{A}^{\mathtt{Dec}}(pk_1, \ldots, pk_n)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathcal{M}$

For $i \in [n], j \in [k]$:

$ct_{i,j} \leftarrow Enc_{pk_i}(m_{i,j}; r_{i,j})$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathcal{A}^{\mathtt{Dec}}(ct_{1,1}, \ldots, ct_{n,k})$

Open $= \emptyset$

For $(i,j) \in \mathtt{I}_S$:

Open $=$ Open $\cup \{(m_{i,j}, r_{i,j})\}$

For $i \in \mathtt{I}_R$:

Open $=$ Open $\cup \{((m_{i,j})_{j\in[k]}, sk_i)\}$

$out \leftarrow \mathcal{A}^{\mathtt{Dec}}(\text{Open})$

$(\mathcal{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$

$\mathtt{Dec}(i, ct)$:

If $(i > n) \vee \exists j \in [k] ct_{i,j} = ct$: **Return** $\perp$

Return $Dec(sk_i, ct)$

If $\beta =$ **"Sender"** : $\mathtt{I}_R = \emptyset$

If $\beta =$ **"Receiver"** : $\mathtt{I}_S = \emptyset$

$\approx$

If $\beta =$ **"Sender"** : $\mathtt{I}_R = \emptyset$

If $\beta =$ **"Receiver"** : $\mathtt{I}_S = \emptyset$

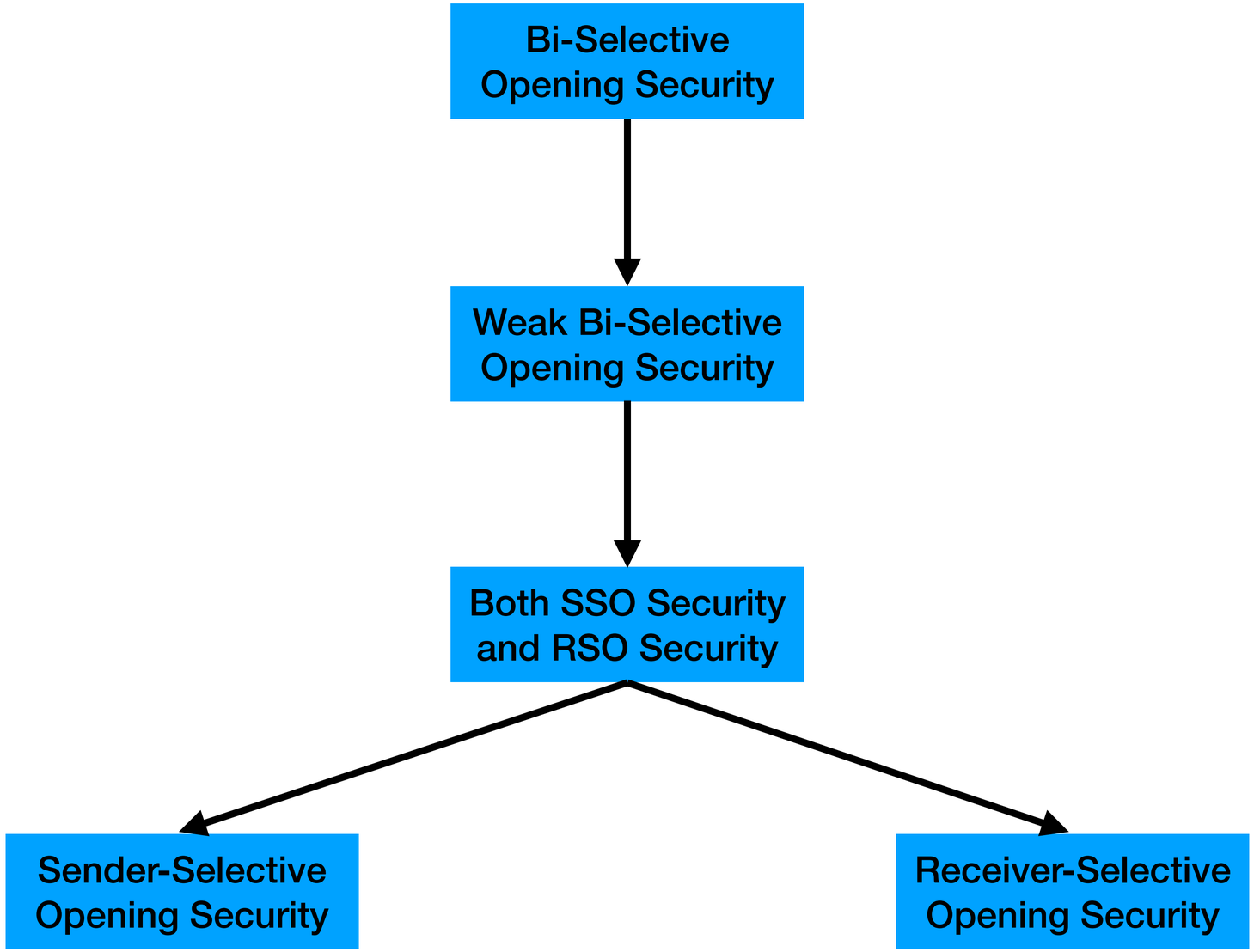
$(\mathcal{M}, \beta) \leftarrow \mathcal{S}(1^\lambda)$

$(m_{i,j})_{i\in[n],j\in[k]} \leftarrow \mathcal{M}$

$(\mathtt{I}_S, \mathtt{I}_R) \leftarrow \mathcal{S}(\beta)$

$out \leftarrow \mathcal{S}((m_{i,j})_{i\in\mathtt{I}_R,j\in[k]}, (m_{i,j})_{(i,j)\in\mathtt{I}_S})$

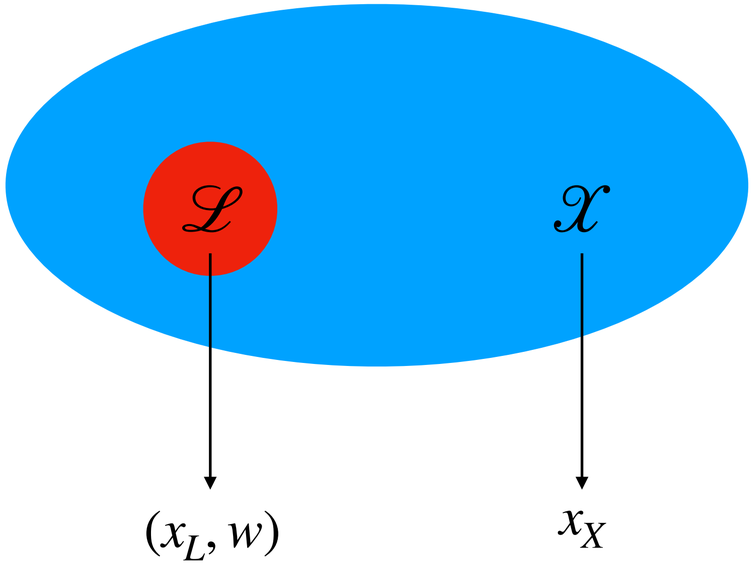
$(\mathcal{M}, \mathtt{I}_S, \mathtt{I}_R, out, (m_{i,j})_{i\in[n],j\in[k]})$


Bi-Selective Opening Security
Weak Bi-Selective Opening Security
Both SSO Security and RSO Security
Sender-Selective Opening Security
Receiver-Selective Opening Security

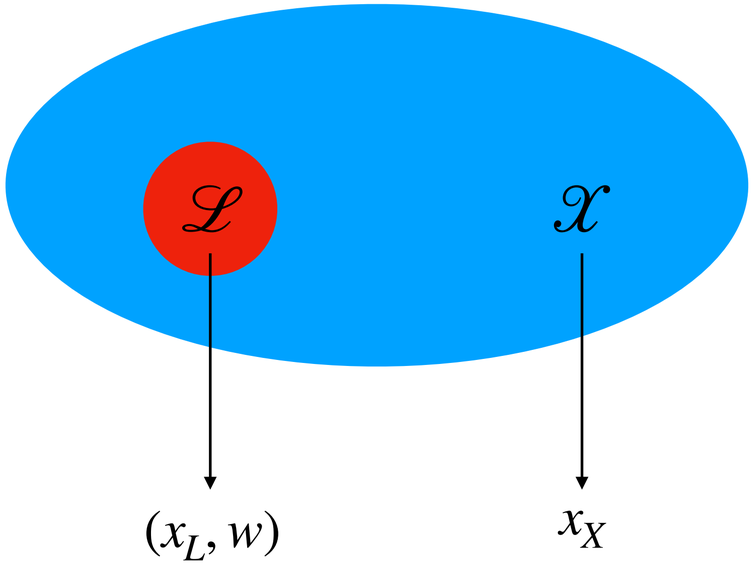
Our Results

- Define Bi-Selective Opening Security for PKE Schemes
- Construct PKE with Bi-Selective Opening Security in the Random Oracle Model
- Define Weak Bi-Selective Opening Security for PKE Schemes
- Construct PKE with Weak Bi-Selective Opening Security in the Plain Model ☜

Key-Equivocable Hash Proof System

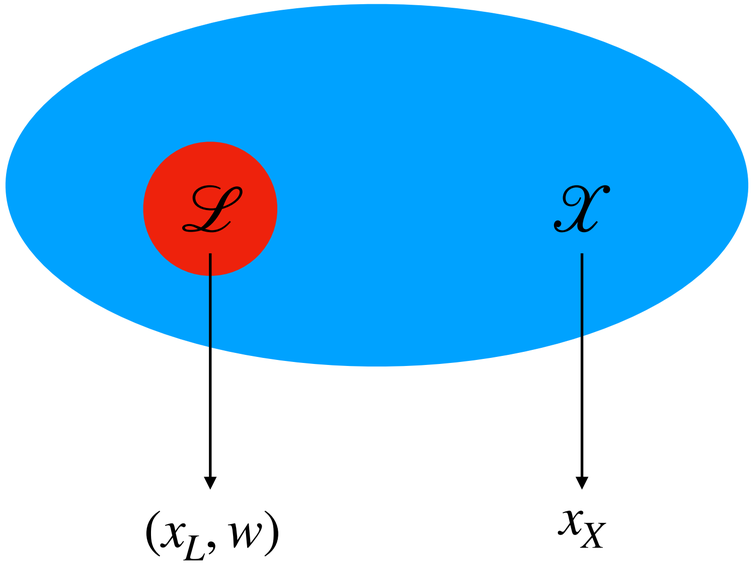


Key-Equivocable Hash Proof System



Hardness: $x_L \approx_c x_X$

Key-Equivocable Hash Proof System



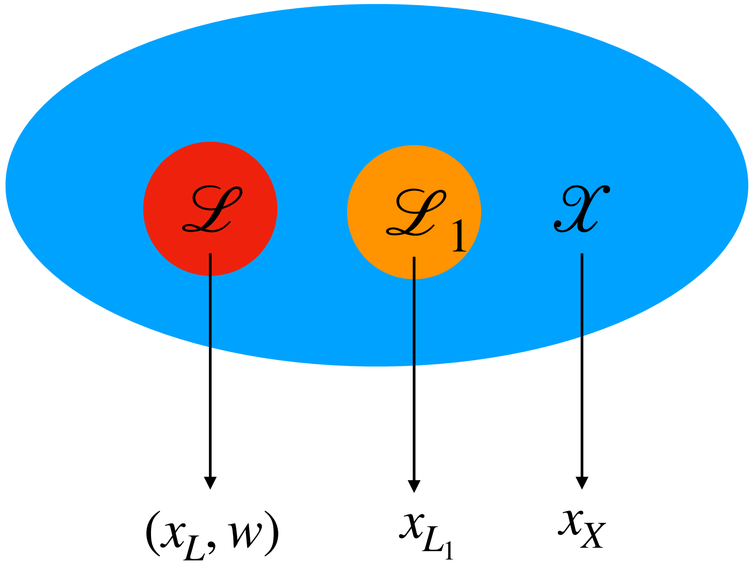
Hardness: $x_L \approx_c x_X$

$\texttt{KeyGen}(1^\lambda) \to (hsk, hpk)$
$\texttt{SecEv}(hsk, x) \to k$
$\texttt{PubEv}(hpk, x, w) \to k$

Projectivity: For any $(x, w) \leftarrow \mathscr{L}$,
$\texttt{SecEv}(hsk, x) = \texttt{PubEv}(hpk, x, w)$

Universality: $(hsk, hpk) \leftarrow \texttt{KeyGen}(1^\lambda), k \leftarrow \mathscr{K}$
for any $x \in \mathscr{X} \backslash \mathscr{L}$, $(hpk, \texttt{SecEv}(hsk, x)) \approx_p (hpk, k)$

Key-Equivocable Hash Proof System



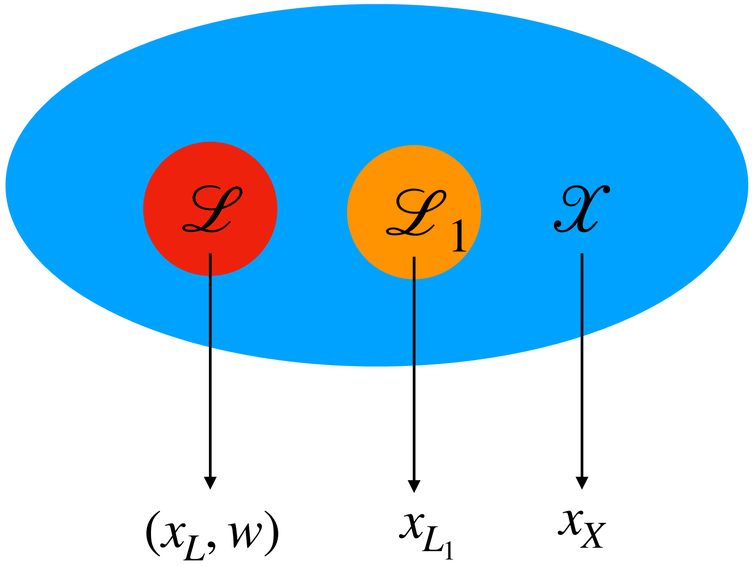
Hardness: $x_L \approx_c x_X$

$\texttt{KeyGen}(1^\lambda) \to (hsk, hpk)$
$\texttt{SecEv}(hsk, x) \to k$
$\texttt{PubEv}(hpk, x, w) \to k$

Projectivity: For any $(x, w) \leftarrow \mathscr{L}$,
$\texttt{SecEv}(hsk, x) = \texttt{PubEv}(hpk, x, w)$

Universality: $(hsk, hpk) \leftarrow \texttt{KeyGen}(1^\lambda)$, $k \leftarrow \mathscr{K}$
for any $x \in \mathscr{X} \backslash \mathscr{L}$, $(hpk, \texttt{SecEv}(hsk, x)) \approx_p (hpk, k)$

Key-Equivocable Hash Proof System



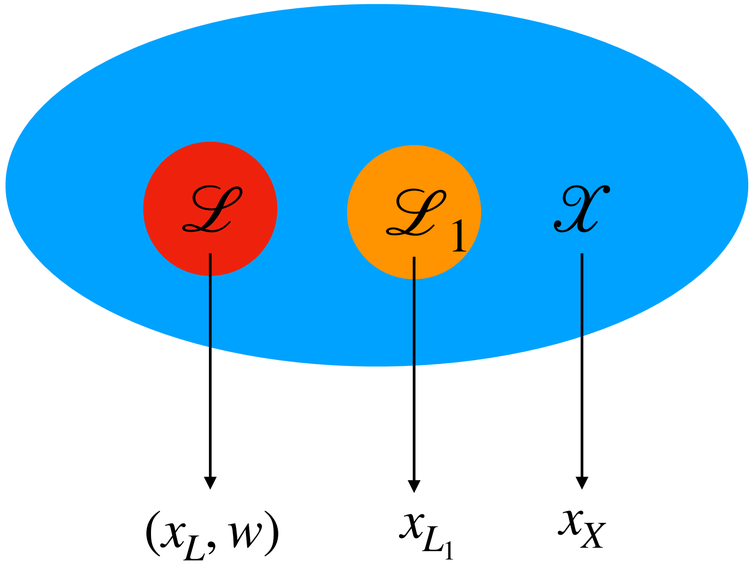
Hardness: $x_L \approx_c x_X$, $x_{L_1} \approx_c x_X, x_{L_1} \approx_c x_L$

$\texttt{KeyGen}(1^\lambda) \to (hsk, hpk)$
$\texttt{SecEv}(hsk, x) \to k$
$\texttt{PubEv}(hpk, x, w) \to k$

Projectivity: For any $(x, w) \leftarrow \mathscr{L}$,
$\texttt{SecEv}(hsk, x) = \texttt{PubEv}(hpk, x, w)$

Universality: $(hsk, hpk) \leftarrow \texttt{KeyGen}(1^\lambda), k \leftarrow \mathscr{K}$
for any $x \in \mathscr{X} \backslash \mathscr{L}$, $(hpk, \texttt{SecEv}(hsk, x)) \approx_p (hpk, k)$

Key-Equivocable Hash Proof System



Hardness: $x_L \approx_c x_X$, $x_{L_1} \approx_c x_X, x_{L_1} \approx_c x_L$

$\texttt{KeyGen}(1^\lambda) \to (hsk, hpk)$
$\texttt{SecEv}(hsk, x) \to k$
$\texttt{PubEv}(hpk, x, w) \to k$
$\texttt{SampHsk}(hsk, x) \to hsk'$

Projectivity: For any $(x, w) \leftarrow \mathscr{L}$,
$\texttt{SecEv}(hsk, x) = \texttt{PubEv}(hpk, x, w)$

Universality: $(hsk, hpk) \leftarrow \texttt{KeyGen}(1^\lambda), k \leftarrow \mathscr{K}$
for any $x \in \mathscr{X} \backslash \mathscr{L}$, $(hpk, \texttt{SecEv}(hsk, x)) \approx_p (hpk, k)$

Key Equivocability: For any $x \in \mathscr{L}_1$

$(hsk, hpk) \leftarrow \texttt{KeyGen}(1^\lambda)$
$k = \texttt{SecEv}(hsk, x)$
$hsk' \leftarrow \texttt{SampHsk}(hsk, x)$
(hsk', hpk, k)

$\approx$

$(hsk, hpk) \leftarrow \texttt{KeyGen}(1^\lambda)$
$k \leftarrow \mathscr{K}$
(hsk, hpk, k)

Construction: Weak Bi-Selective Opening Secure PKE

$\mathsf{HPS} = (\mathsf{HPS}.\mathtt{KeyGen}, \mathsf{HPS}.\mathtt{SecEv}, \mathsf{HPS}.\mathtt{PubEv})$
is a key-equivocable hash proof system.

Construction: Weak Bi-Selective Opening Secure PKE

$\mathsf{HPS} = (\mathsf{HPS}.\mathtt{KeyGen}, \mathsf{HPS}.\mathtt{SecEv}, \mathsf{HPS}.\mathtt{PubEv})$
is a key-equivocable hash proof system.

$\mathtt{KeyGen}(1^\lambda)$:
$(hpk, hsk) \leftarrow \mathsf{HPS}.\mathtt{KeyGen}(1^\lambda)$
$SK = hsk$
$PK = hpk$

Construction: Weak Bi-Selective Opening Secure PKE

$\mathsf{HPS} = (\mathsf{HPS}.\mathtt{KeyGen}, \mathsf{HPS}.\mathtt{SecEv}, \mathsf{HPS}.\mathtt{PubEv})$
is a key-equivocable hash proof system.

$\mathtt{KeyGen}(1^\lambda)$:
$(hpk, hsk) \leftarrow \mathsf{HPS}.\mathtt{KeyGen}(1^\lambda)$
$SK = hsk$
$PK = hpk$

$\mathtt{Enc}(PK = hpk, m \in \{0,1\})$:
If $m = 0$:
$x \leftarrow \mathscr{X}$
$k \leftarrow \mathscr{K}$ ($\mathscr{K}$ is the output space of $\mathtt{PubEv}$ and $\mathtt{SecEv}$)
If $m = 1$:
$(x, w) \leftarrow \mathscr{L}$
$k = \mathtt{PubEv}(hpk, x, w)$
$CT = (x, k)$

Construction: Weak Bi-Selective Opening Secure PKE

$\mathsf{HPS} = (\mathsf{HPS}.\mathtt{KeyGen}, \mathsf{HPS}.\mathtt{SecEv}, \mathsf{HPS}.\mathtt{PubEv})$
is a key-equivocable hash proof system.

$\mathtt{KeyGen}(1^\lambda)$:
$(hpk, hsk) \leftarrow \mathsf{HPS}.\mathtt{KeyGen}(1^\lambda)$
$SK = hsk$
$PK = hpk$

$\mathtt{Enc}(PK = hpk, m \in \{0,1\})$:
If $m = 0$:
$x \leftarrow \mathscr{X}$
$k \leftarrow \mathscr{K}$
If $m = 1$:
$(x, w) \leftarrow \mathscr{L}$
$k = \mathtt{PubEv}(hpk, x, w)$
$CT = (x, k)$

$\mathtt{Dec}(SK = hsk, CT = (x, k))$:
$\bar{k} \leftarrow \mathsf{HPS}.\mathtt{SecEv}(hsk, x)$
If $k = \bar{k} : m = 1$
Otherwise : $m = 0$

Security of the Construction

$PK_1, \ldots, PK_n$ →

← $(\mathscr{M}, \beta)$

$(CT_i)_{i \in [n]}$ →

← $\mathrm{I}_S, \mathrm{I}_R$

$(r_i, m_i)_{i \in \mathrm{I}_S}$
$(SK_i, m_i)_{i \in \mathrm{I}_R}$ →

← out

We consider the case that each public key is used for only once.

Security of the Construction

$PK_1, \ldots, PK_n$

$(\mathscr{M}, \beta)$

$(CT_i)_{i\in[n]}$

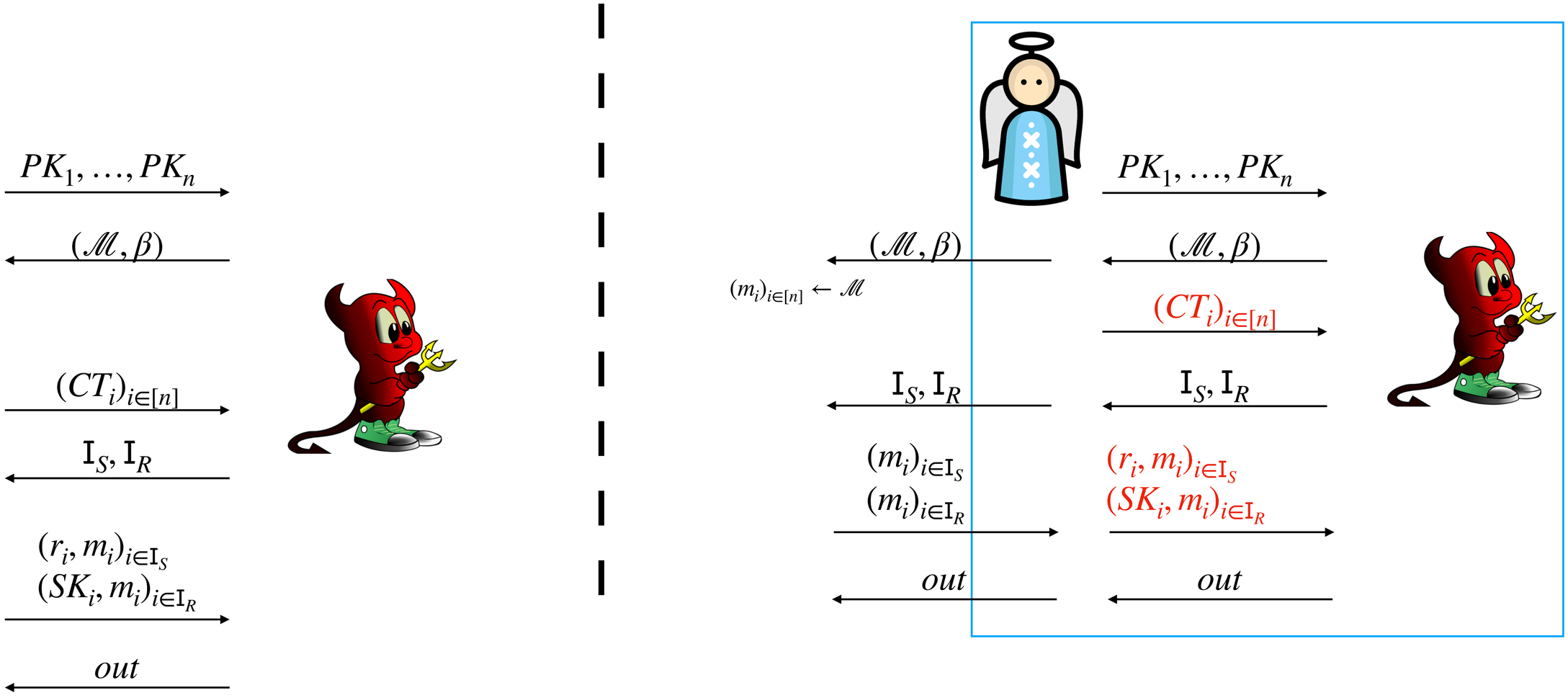
$\mathtt{I}_S, \mathtt{I}_R$

If $\beta =$ "Sender" : $\mathtt{I}_R = \varnothing$
If $\beta =$ "Receiver" : $\mathtt{I}_S = \varnothing$

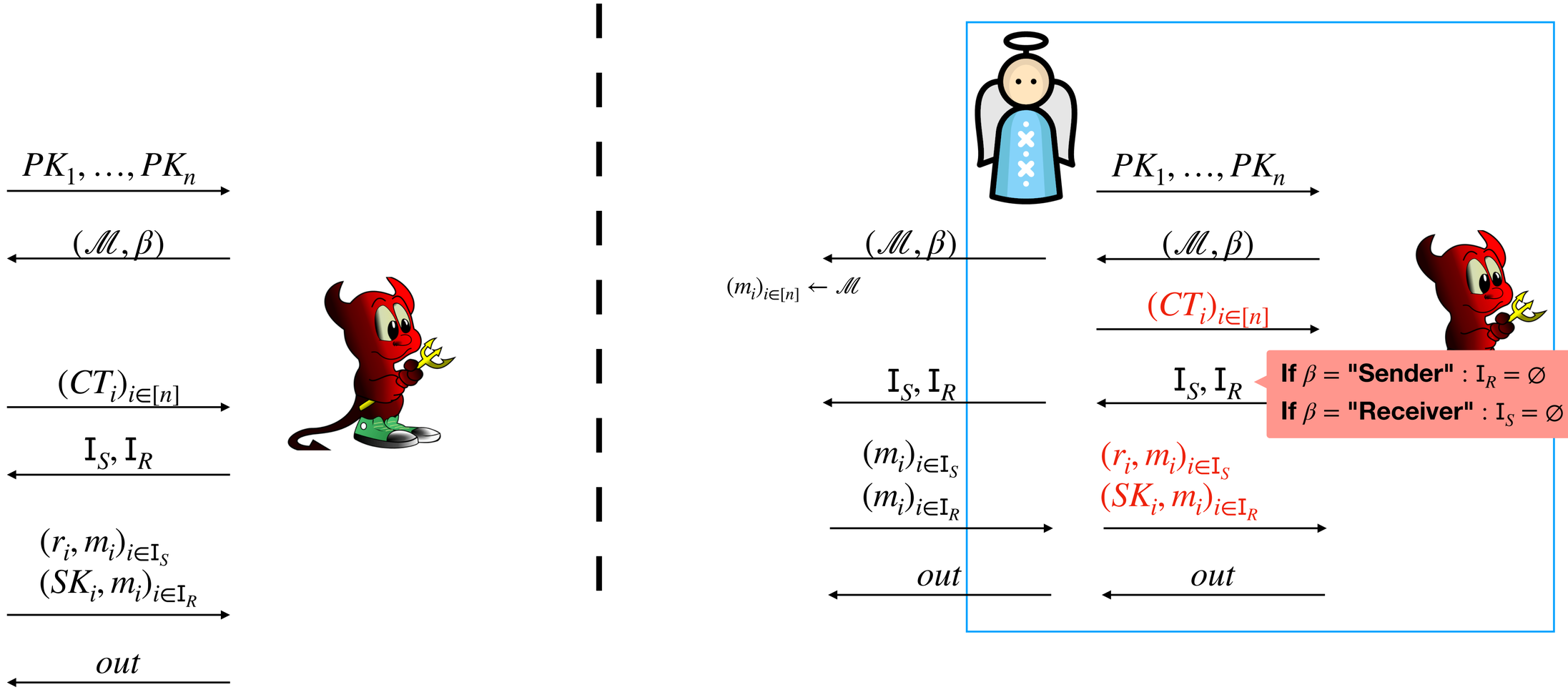
$(r_i, m_i)_{i\in\mathtt{I}_S}$
$(SK_i, m_i)_{i\in\mathtt{I}_R}$

out

Security of the Construction



Security of the Construction



Security of the Construction

$PK_1, \ldots, PK_n$

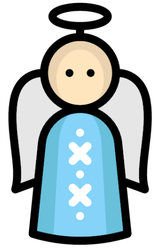
$(\mathcal{M}, \beta)$

$(CT_i)_{i\in[n]}$

$\mathtt{I}_S, \mathtt{I}_R$

$(r_i, m_i)_{i\in\mathtt{I}_S}$
$(SK_i, m_i)_{i\in\mathtt{I}_R}$

out



$PK_1, \ldots, PK_n$

$(\mathcal{M}, \beta)$

$(\mathcal{M}, \beta)$

$(m_i)_{i\in[n]} \leftarrow \mathcal{M}$

If $\beta =$ "sender":

$(CT_i)_{i\in[n]}$

$\mathtt{I}_S$

$\mathtt{I}_S$

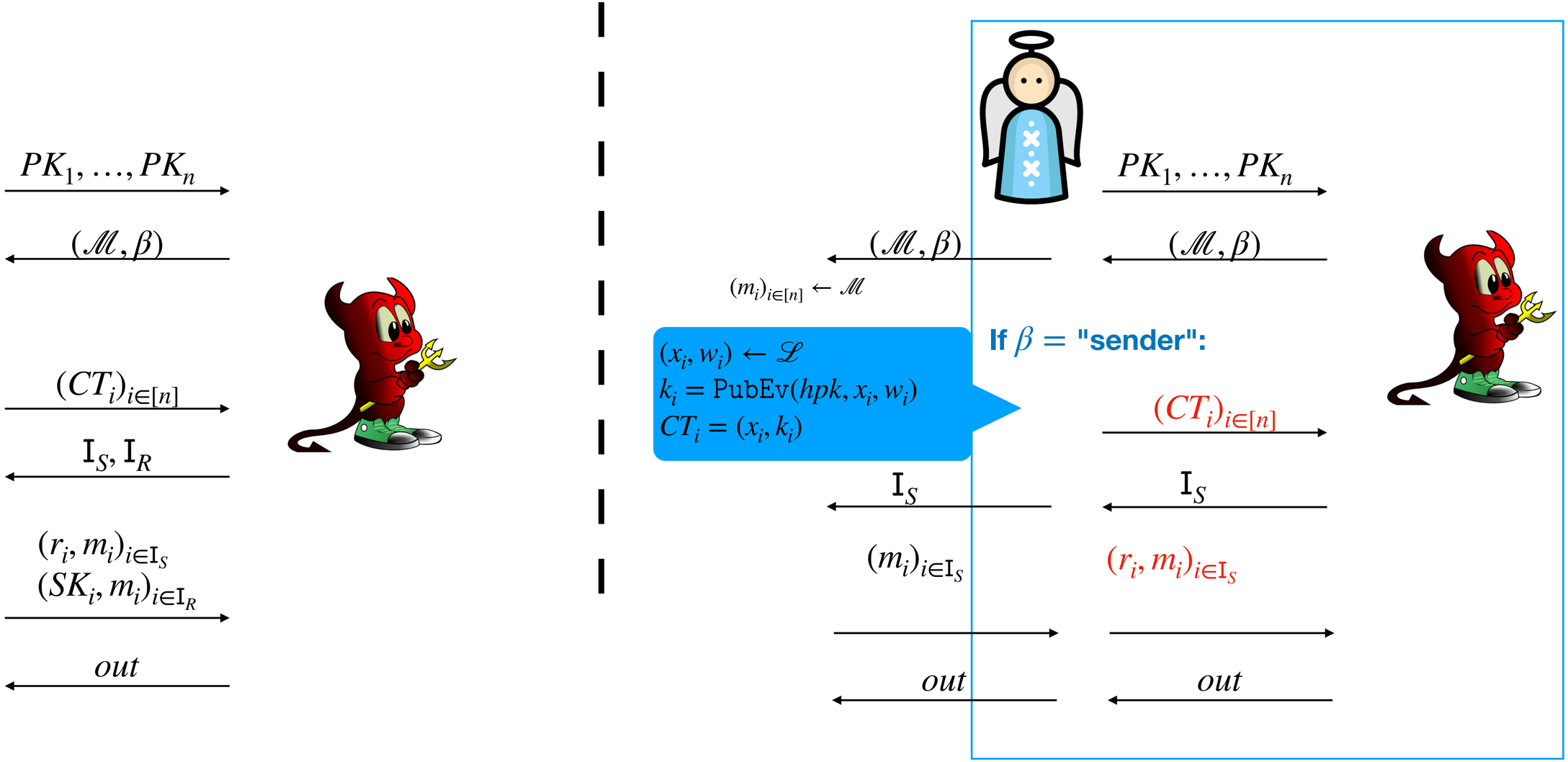
$(m_i)_{i\in\mathtt{I}_S}$

$(r_i, m_i)_{i\in\mathtt{I}_S}$

out

out

Security of the Construction



Security of the Construction

$(m_i)_{i\in[n]} \leftarrow \mathscr{M}$
If $m_i = 0$:
$x_i \leftarrow \mathscr{X}$
$k_i \leftarrow \mathscr{K}$
If $m_i = 1$:
$(x_i, w_i) \leftarrow \mathscr{L}$
$k_i = \mathtt{PubEv}(hpk, x_i, w_i)$
$CT_i = (x_i, k_i)$

$, \ldots, PK_n$

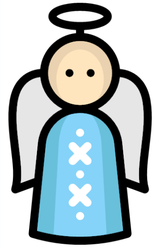
$(\mathscr{M}, \beta)$

$(CT_i)_{i\in[n]}$

$\mathtt{I}_S, \mathtt{I}_R$

$(r_i, m_i)_{i\in\mathtt{I}_S}$
$(SK_i, m_i)_{i\in\mathtt{I}_R}$

out



$PK_1, \ldots, PK_n$

$(\mathscr{M}, \beta)$

$(\mathscr{M}, \beta)$

$(m_i)_{i\in[n]} \leftarrow \mathscr{M}$

$(x_i, w_i) \leftarrow \mathscr{L}$
$k_i = \mathtt{PubEv}(hpk, x_i, w_i)$
$CT_i = (x_i, k_i)$

If β = "sender":

$(CT_i)_{i\in[n]}$

$\mathtt{I}_S$

$\mathtt{I}_S$

$(m_i)_{i\in\mathtt{I}_S}$

$(r_i, m_i)_{i\in\mathtt{I}_S}$

out

out

Security of the Construction

$(m_i)_{i\in[n]} \leftarrow \mathscr{M}$
If $m_i = 0$:
$x_i \leftarrow \mathscr{X}$
$k_i \leftarrow \mathscr{K}$
If $m_i = 1$:
$(x_i, w_i) \leftarrow \mathscr{L}$
$k_i = \mathtt{PubEv}(hpk, x_i, w_i)$
$CT_i = (x_i, k_i)$

$, \ldots, PK_n$

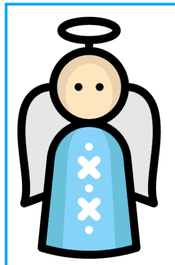
$(\mathscr{M}, \beta)$

$(CT_i)_{i\in[n]}$

$\mathtt{I}_S, \mathtt{I}_R$

$(r_i, m_i)_{i\in\mathtt{I}_S}$
$(SK_i, m_i)_{i\in\mathtt{I}_R}$

out



$PK_1, \ldots, PK_n$

$(\mathscr{M}, \beta)$

$(\mathscr{M}, \beta)$

$(m_i)_{i\in[n]} \leftarrow \mathscr{M}$

$(x_i, w_i) \leftarrow \mathscr{L}$
$k_i = \mathtt{PubEv}(hpk, x_i, w_i)$
$CT_i = (x_i, k_i)$

If β = "sender":

$(CT_i)_{i\in[n]}$

$\mathtt{I}_S$

$\mathtt{I}_S$

$(m_i)_{i\in\mathtt{I}_S}$

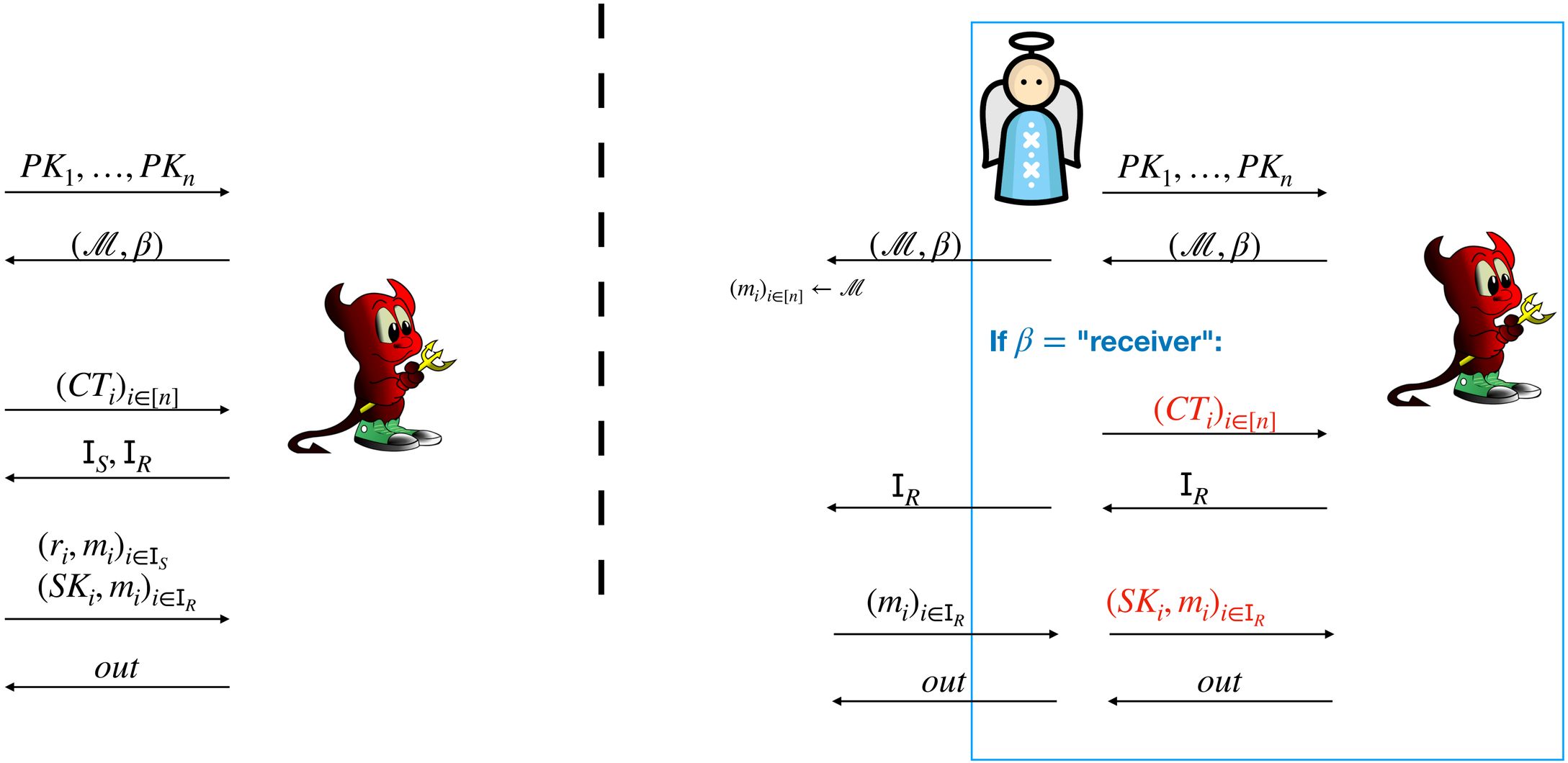
$(r_i, m_i)_{i\in\mathtt{I}_S}$

out

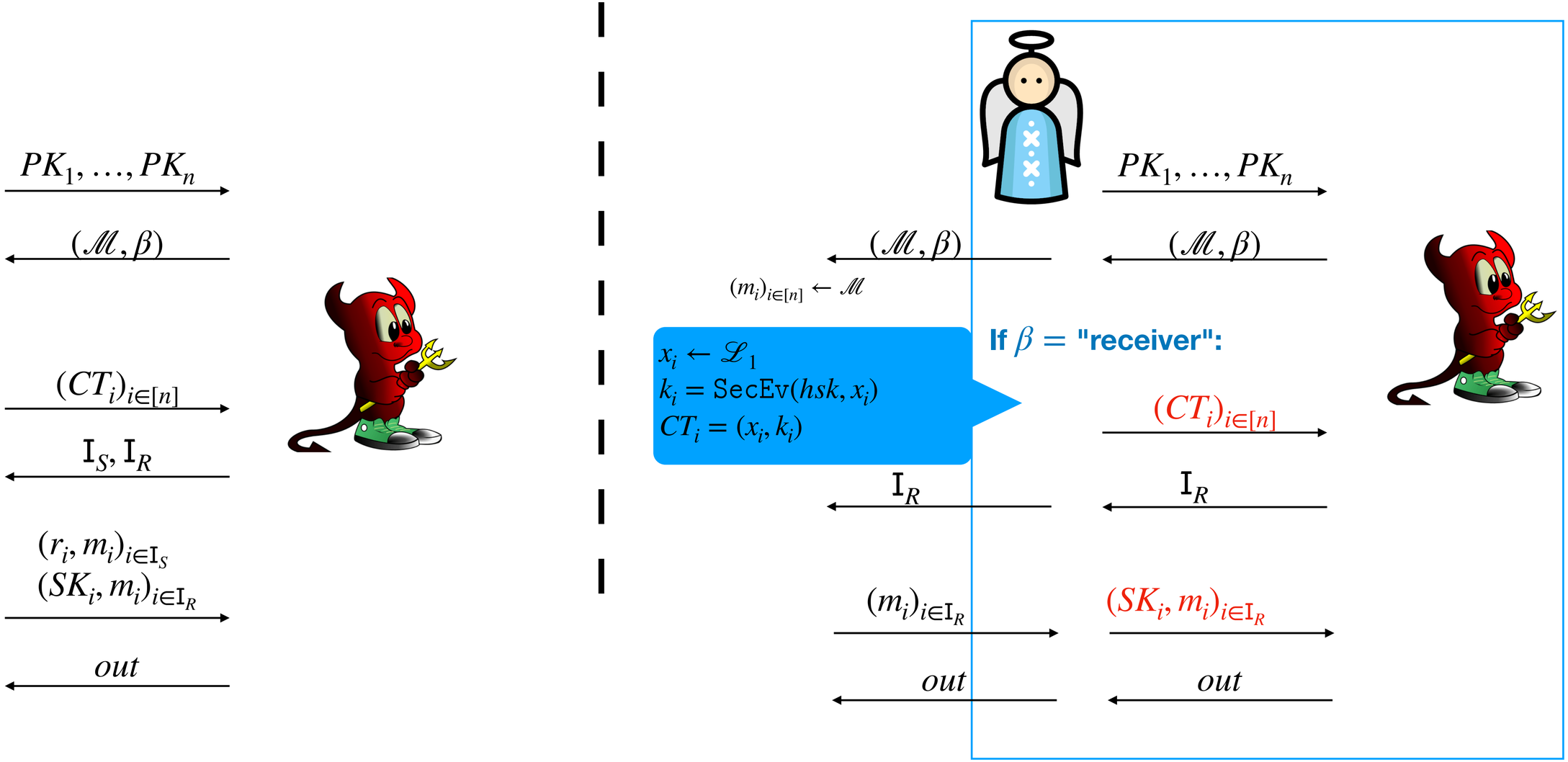
out

If $m_i = 0 : r_i = (x_i, k_i)$
If $m_i = 1$:use the correct randomness

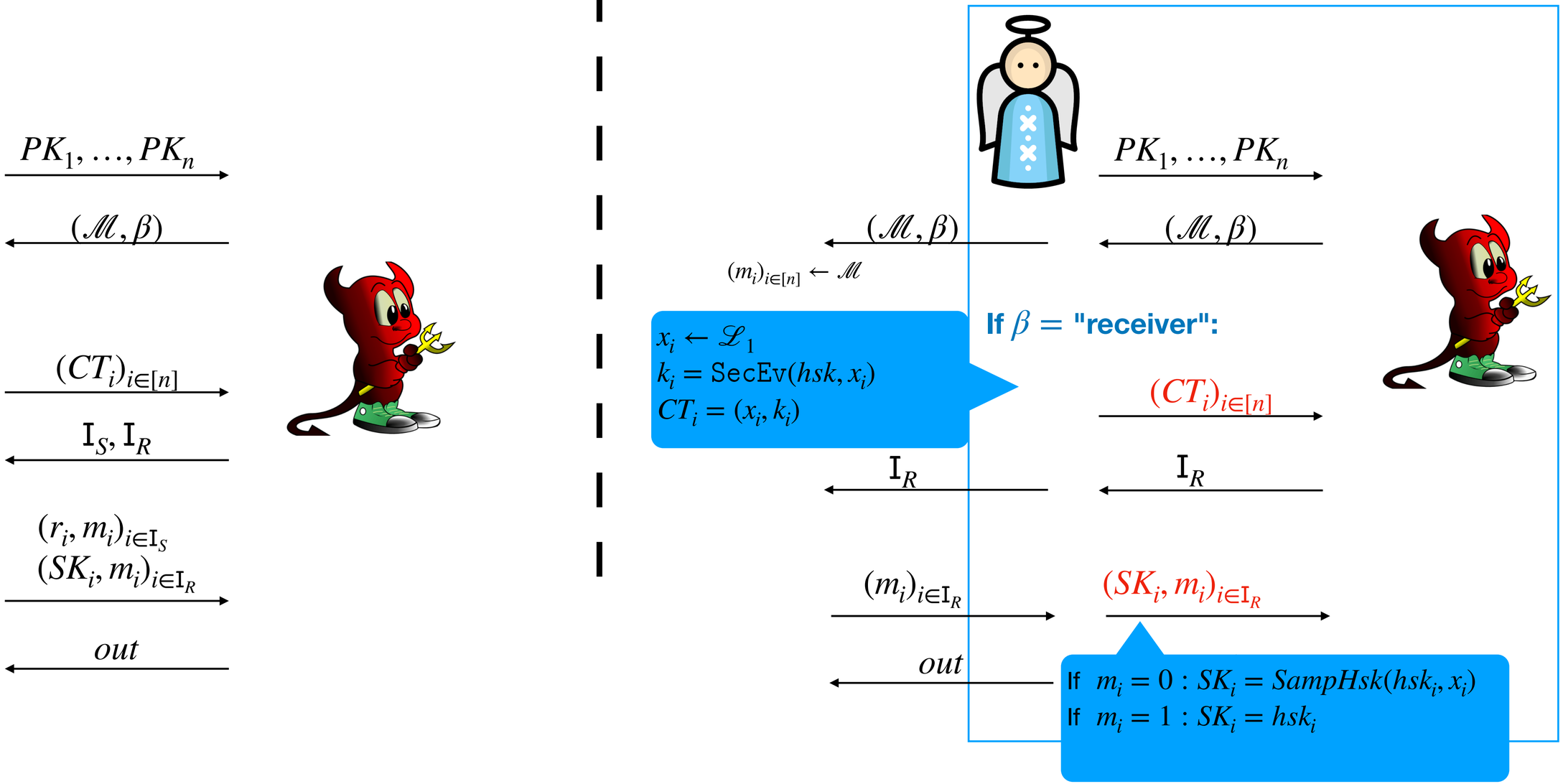
Security of the Construction



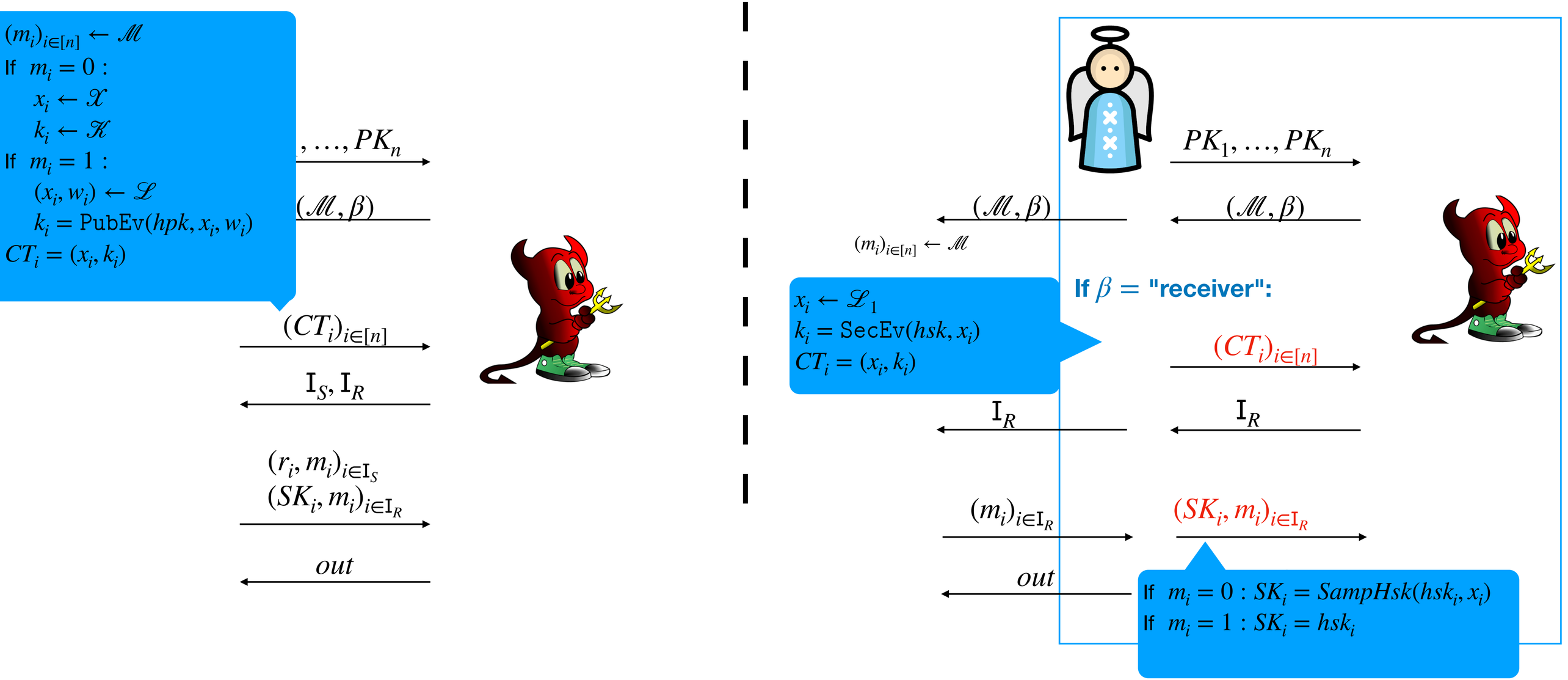
Security of the Construction



Security of the Construction



Security of the Construction



Construction: Weak Bi-Selective Opening Secure PKE

$\mathsf{HPS} = (\mathsf{HPS}.\mathtt{KeyGen}, \mathsf{HPS}.\mathtt{SecEv}, \mathsf{HPS}.\mathtt{PubEv})$
is a key-equivocable hash proof system.

$\mathtt{KeyGen}(1^\lambda)$:
$(hpk, hsk) \leftarrow \mathsf{HPS}.\mathtt{KeyGen}(1^\lambda)$
$SK = hsk$
$PK = hpk$

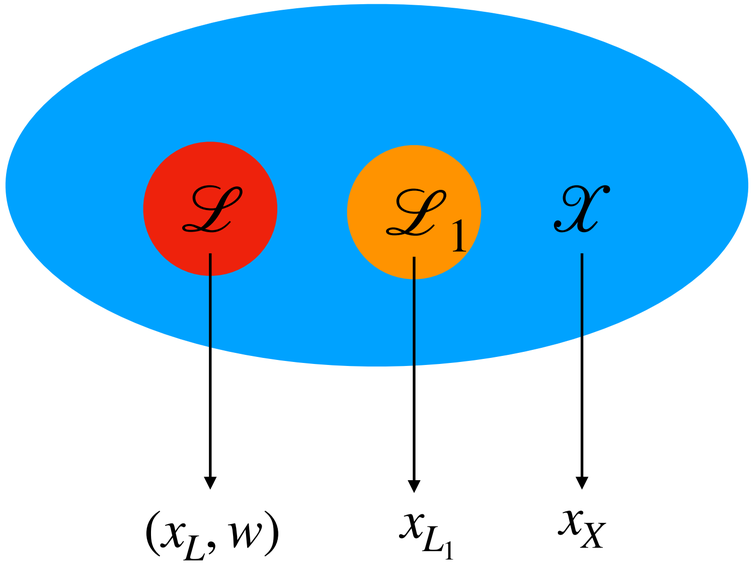
$\mathtt{Enc}(PK = hpk, m \in \{0,1\})$:
If $m = 0$:
$x \leftarrow \mathscr{X}$
$k \leftarrow \mathscr{K}$
If $m = 1$:
$(x, w) \leftarrow \mathscr{L}$
$k = \mathtt{PubEv}(hpk, x, w)$
$CT = (x, k)$

$\mathtt{Dec}(SK = hsk, CT = (x, k))$:
$\bar{k} \leftarrow \mathsf{HPS}.\mathtt{SecEv}(hsk, x)$
If $k = \bar{k} : m = 1$
Otherwise : $m = 0$

Support Multi-Bit Messages
Achieving Security in the Multi-Challenge Setting
Achieving CCA-Security

https://eprint.iacr.org/2021/1268

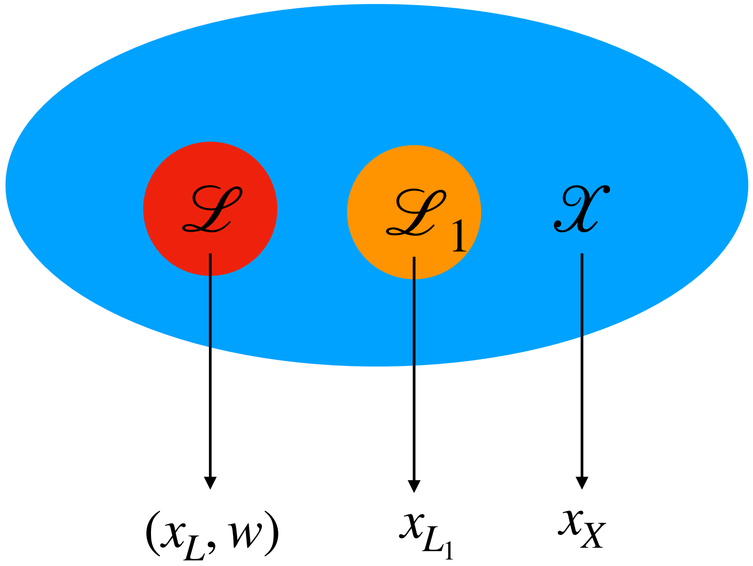
Key-Equivocable Hash Proof System from DDH



$$g \in \mathbb{G}, g_1 = g^{a_1}, g_2 = g^{a_2}, g_3 = g^{a_3}$$

$$\mathscr{X} = \mathbb{G} \times \mathbb{G} \times \mathbb{G}$$
$$\mathscr{L} = \{(g_1^w, g_2^w, g_3^w)\}$$
$$\mathscr{L}_1 = \{(g_1^w, g_2^{w'}, g_3^w) : w \neq w'\}$$

Key-Equivocable Hash Proof System from DDH

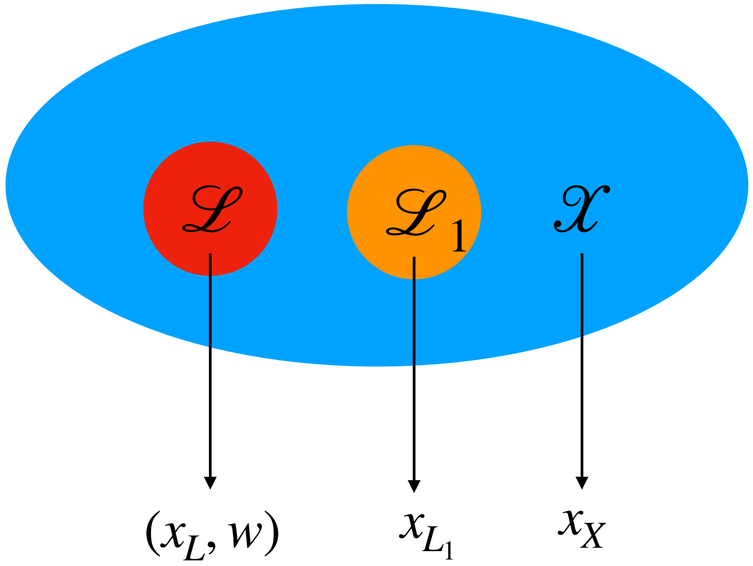


$$g \in \mathbb{G}, g_1 = g^{a_1}, g_2 = g^{a_2}, g_3 = g^{a_3}$$

$$\mathscr{X} = \mathbb{G} \times \mathbb{G} \times \mathbb{G}$$
$$\mathscr{L} = \{(g_1^w, g_2^w, g_3^w)\}$$
$$\mathscr{L}_1 = \{(g_1^w, g_2^{w'}, g_3^w) : w \neq w'\}$$

Hardness: $x_L \approx_c x_X$, $x_{L_1} \approx_c x_X, x_{L_1} \approx_c x_L$

Key-Equivocable Hash Proof System from DDH



$\texttt{KeyGen}(1^\lambda)$:
$hsk = (s_1, s_2, s_3) \leftarrow \mathbb{Z}_q^3$
$hpk = g_1^{s_1} g_2^{s_2} g_3^{s_3}$

$\texttt{SecEv}(hsk = (s_1, s_2, s_3), x = (x_1, x_2, x_3))$:
$k = x_1^{s_1} x_2^{s_2} x_3^{s_3}$

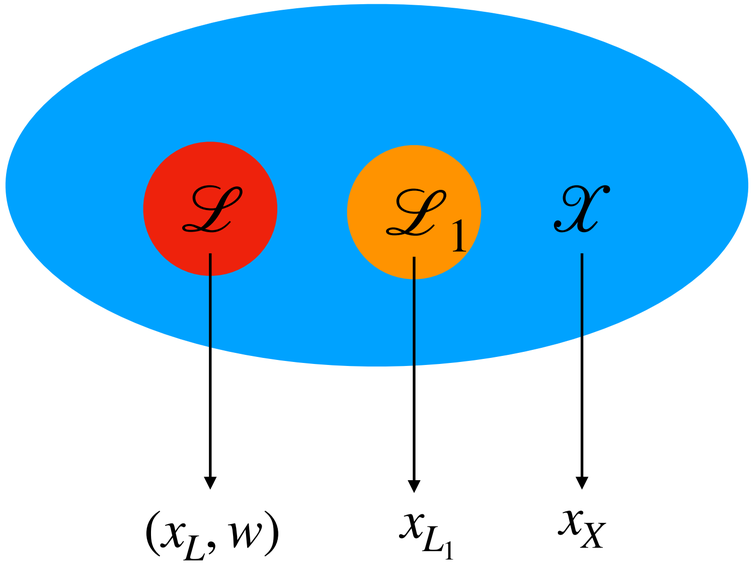
$\texttt{PubEv}(hpk, x, w)$:
$k = hpk^w$

$\texttt{SampHsk}(hsk = (s_1, s_2, s_3), x)$:
$s_1' = s_1, s_2' \leftarrow \mathbb{Z}_q$
$s_3' = a_3^{-1}(a_1 s_1 + a_2 s_2 + a_3 s_3 - a_1 s_1' - a_2 s_2')$

$g \in \mathbb{G}, g_1 = g^{a_1}, g_2 = g^{a_2}, g_3 = g^{a_3}$

$\mathscr{X} = \mathbb{G} \times \mathbb{G} \times \mathbb{G}$
$\mathscr{L} = \{(g_1^w, g_2^w, g_3^w)\}$
$\mathscr{L}_1 = \{(g_1^w, g_2^{w'}, g_3^w) : w \neq w'\}$

Key-Equivocable Hash Proof System from DDH



$\texttt{KeyGen}(1^\lambda)$:
$hsk = (s_1, s_2, s_3) \leftarrow \mathbb{Z}_q^3$
$hpk = g_1^{s_1} g_2^{s_2} g_3^{s_3}$

$\texttt{SecEv}(hsk = (s_1, s_2, s_3), x = (x_1, x_2, x_3))$:
$k = x_1^{s_1} x_2^{s_2} x_3^{s_3}$

$\texttt{PubEv}(hpk, x, w)$:
$k = hpk^w$

$\texttt{SampHsk}(hsk = (s_1, s_2, s_3), x)$:
$s_1' = s_1, s_2' \leftarrow \mathbb{Z}_q$
$s_3' = a_3^{-1}(a_1 s_1 + a_2 s_2 + a_3 s_3 - a_1 s_1' - a_2 s_2')$

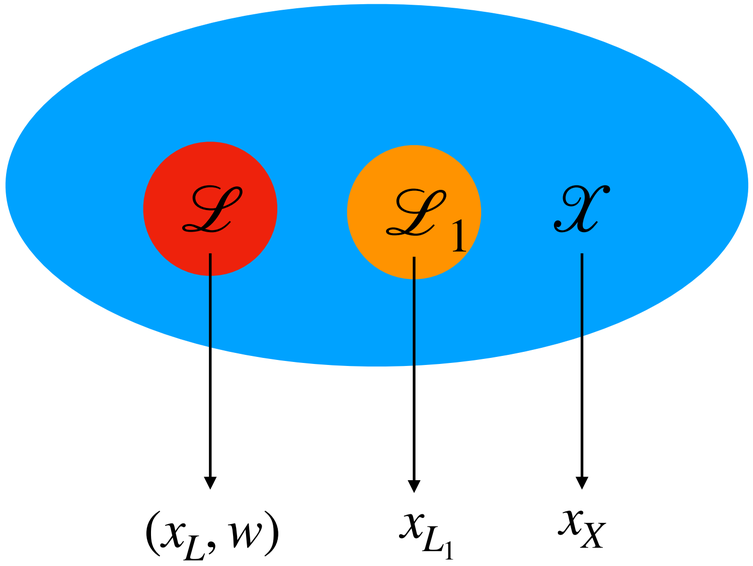
$g \in \mathbb{G}, g_1 = g^{a_1}, g_2 = g^{a_2}, g_3 = g^{a_3}$

$\mathscr{X} = \mathbb{G} \times \mathbb{G} \times \mathbb{G}$
$\mathscr{L} = \{(g_1^w, g_2^w, g_3^w)\}$
$\mathscr{L}_1 = \{(g_1^w, g_2^{w'}, g_3^w) : w \neq w'\}$

Projectivity: $x = (x_1, x_2, x_3) = (g_1^w, g_2^w, g_3^w)$,
$\texttt{SecEv}(hsk, x) = x_1^{s_1} x_2^{s_2} x_3^{s_3} = g_1^{s_1 w} g_2^{s_2 w} g_3^{s_3 w} = hpk^w = \texttt{PubEv}(hsk, x, w)$

Universality: $x = (x_1, x_2, x_3) = (g_1^{w_1}, g_2^{w_2}, g_3^{w_3})$
$\texttt{SecEv}(hsk, x) = x_1^{s_1} x_2^{s_2} x_3^{s_3} = g_1^{s_1 w_1} g_2^{s_2 w_2} g_3^{s_3 w_3} = hpk^{w_1} g_2^{s_2(w_2 - w_1)} g_3^{s_3(w_3 - w_1)}$

Key-Equivocable Hash Proof System from DDH



$\texttt{KeyGen}(1^\lambda)$:
$hsk = (s_1, s_2, s_3) \leftarrow \mathbb{Z}_q^3$
$hpk = g_1^{s_1} g_2^{s_2} g_3^{s_3}$

$\texttt{SecEv}(hsk = (s_1, s_2, s_3), x = (x_1, x_2, x_3))$:
$k = x_1^{s_1} x_2^{s_2} x_3^{s_3}$

$\texttt{PubEv}(hpk, x, w)$:
$k = hpk^w$

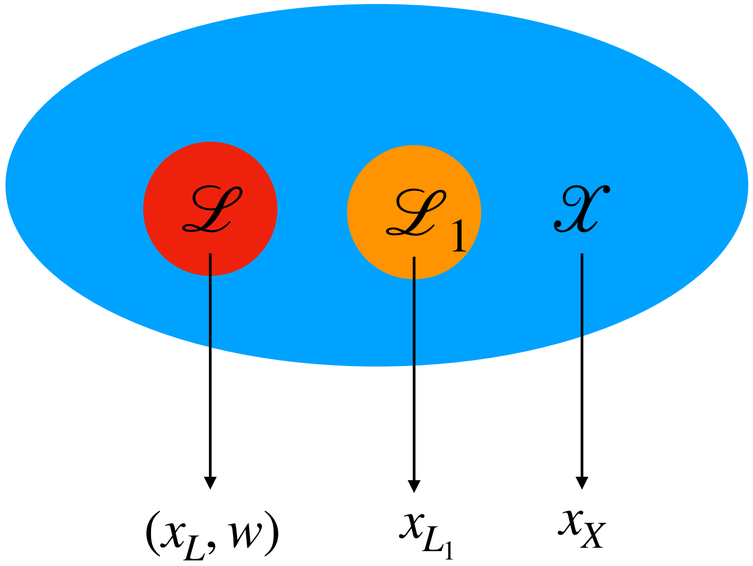
$\texttt{SampHsk}(hsk = (s_1, s_2, s_3), x)$:
$s_1' = s_1, s_2' \leftarrow \mathbb{Z}_q$
$s_3' = a_3^{-1}(a_1 s_1 + a_2 s_2 + a_3 s_3 - a_1 s_1' - a_2 s_2')$

$g \in \mathbb{G}, g_1 = g^{a_1}, g_2 = g^{a_2}, g_3 = g^{a_3}$

$\mathscr{X} = \mathbb{G} \times \mathbb{G} \times \mathbb{G}$
$\mathscr{L} = \{(g_1^w, g_2^w, g_3^w)\}$
$\mathscr{L}_1 = \{(g_1^w, g_2^{w'}, g_3^w) : w \neq w'\}$

Key Equivocability: $x = (x_1, x_2, x_3) = (g_1^w, g_2^{w'}, g_3^w)$
$hsk = (s_1, s_2, s_3)$
$hsk' = (s_1, s_2', s_3')$
$hpk = g_1^{s_1} g_2^{s_2} g_3^{s_3} = g_1^{s_1} g_2^{s_2'} g_3^{s_3'}$
$k = x_1^{s_1} x_2^{s_2} x_3^{s_3} = hpk^w g_2^{s_2(w'-w)}$

Key-Equivocable Hash Proof System from DDH



$\texttt{KeyGen}(1^\lambda)$:
$hsk = (s_1, s_2, s_3) \leftarrow \mathbb{Z}_q^3$
$hpk = g_1^{s_1} g_2^{s_2} g_3^{s_3}$

$\texttt{SecEv}(hsk = (s_1, s_2, s_3), x = (x_1, x_2, x_3))$:
$k = x_1^{s_1} x_2^{s_2} x_3^{s_3}$

$\texttt{PubEv}(hpk, x, w)$:
$k = hpk^w$

$\texttt{SampHsk}(hsk = (s_1, s_2, s_3), x)$:
$s_1' = s_1, s_2' \leftarrow \mathbb{Z}_q$
$s_3' = a_3^{-1}(a_1 s_1 + a_2 s_2 + a_3 s_3 - a_1 s_1' - a_2 s_2')$

$g \in \mathbb{G}, g_1 = g^{a_1}, g_2 = g^{a_2}, g_3 = g^{a_3}$

$\mathscr{X} = \mathbb{G} \times \mathbb{G} \times \mathbb{G}$
$\mathscr{L} = \{(g_1^w, g_2^w, g_3^w)\}$
$\mathscr{L}_1 = \{(g_1^w, g_2^{w'}, g_3^w) : w \neq w'\}$

DCR-Based Instantiation
https://eprint.iacr.org/2021/1268

Our Results

- Define Bi-Selective Opening Security for PKE Schemes
- Define Weak Bi-Selective Opening Security for PKE Schemes
- Construct PKE with Bi-Selective Opening Security in the Random Oracle Model
- Construct PKE with Weak Bi-Selective Opening Security in the Plain Model

Thanks for your Attention!

https://eprint.iacr.org/2021/1268