

Protocols: Key Exchange Security - Pen & Paper Model and Proof

Doreen Riepel, Paul Rösler

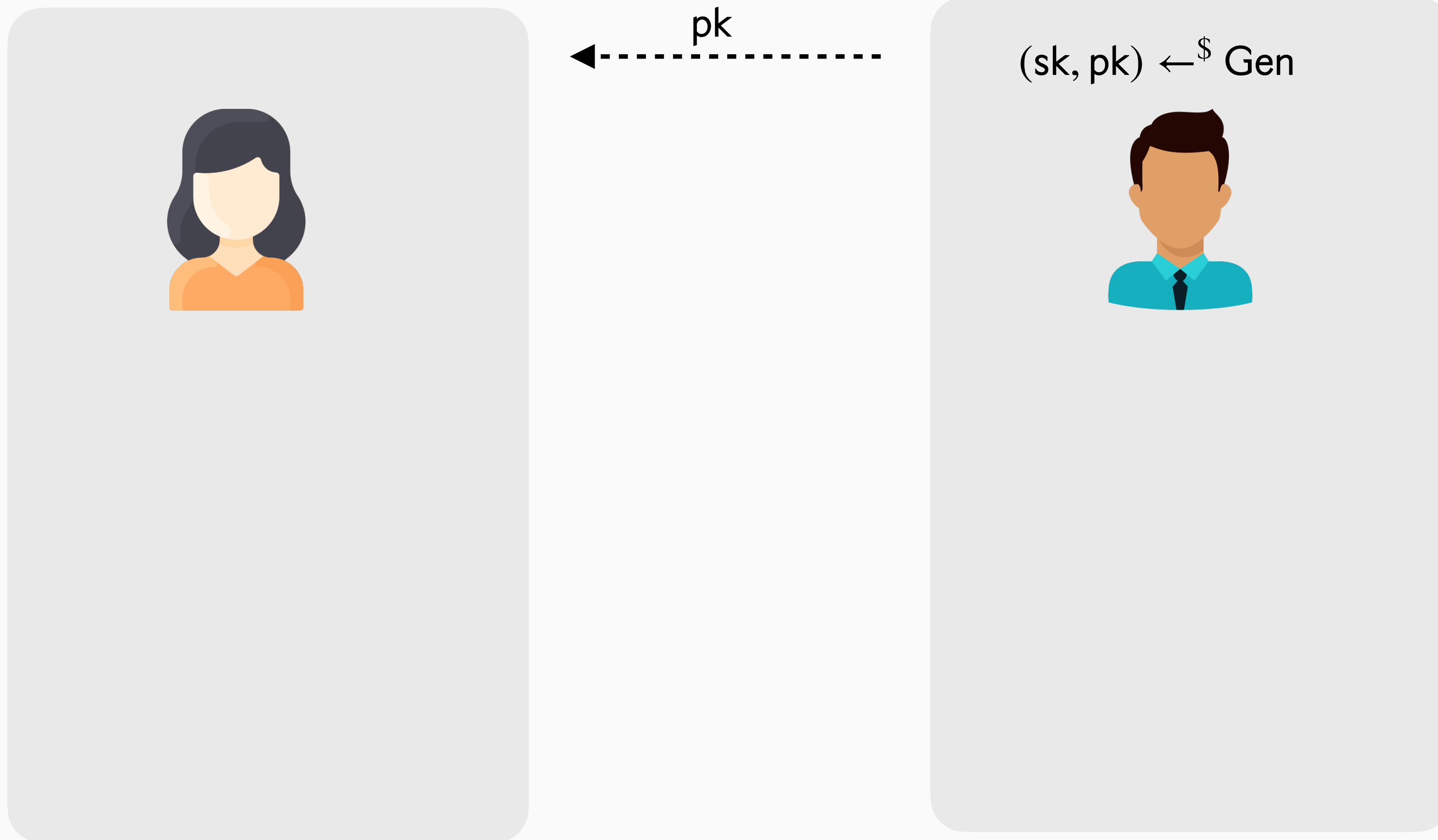
May 2025

Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:

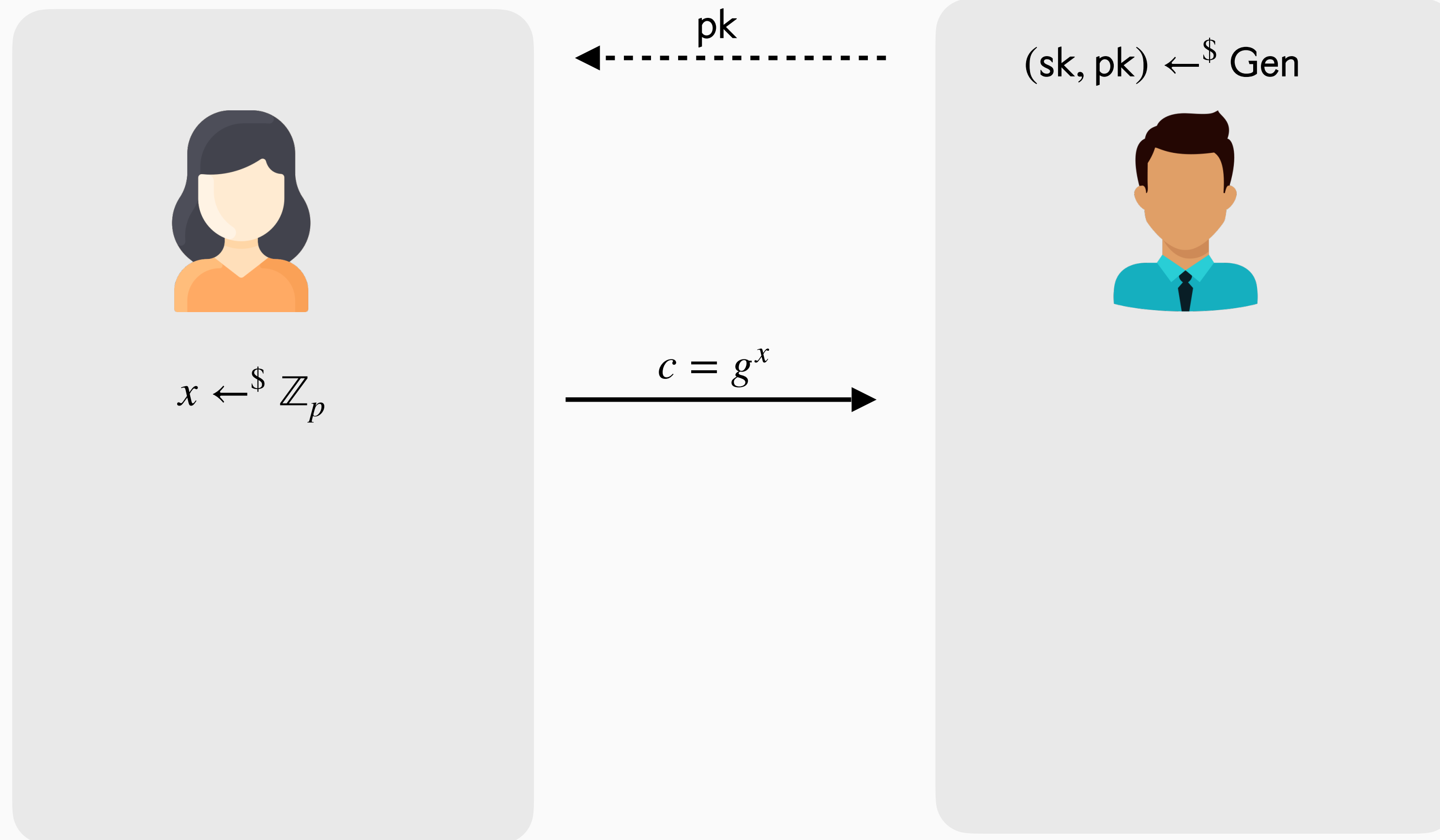
Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:



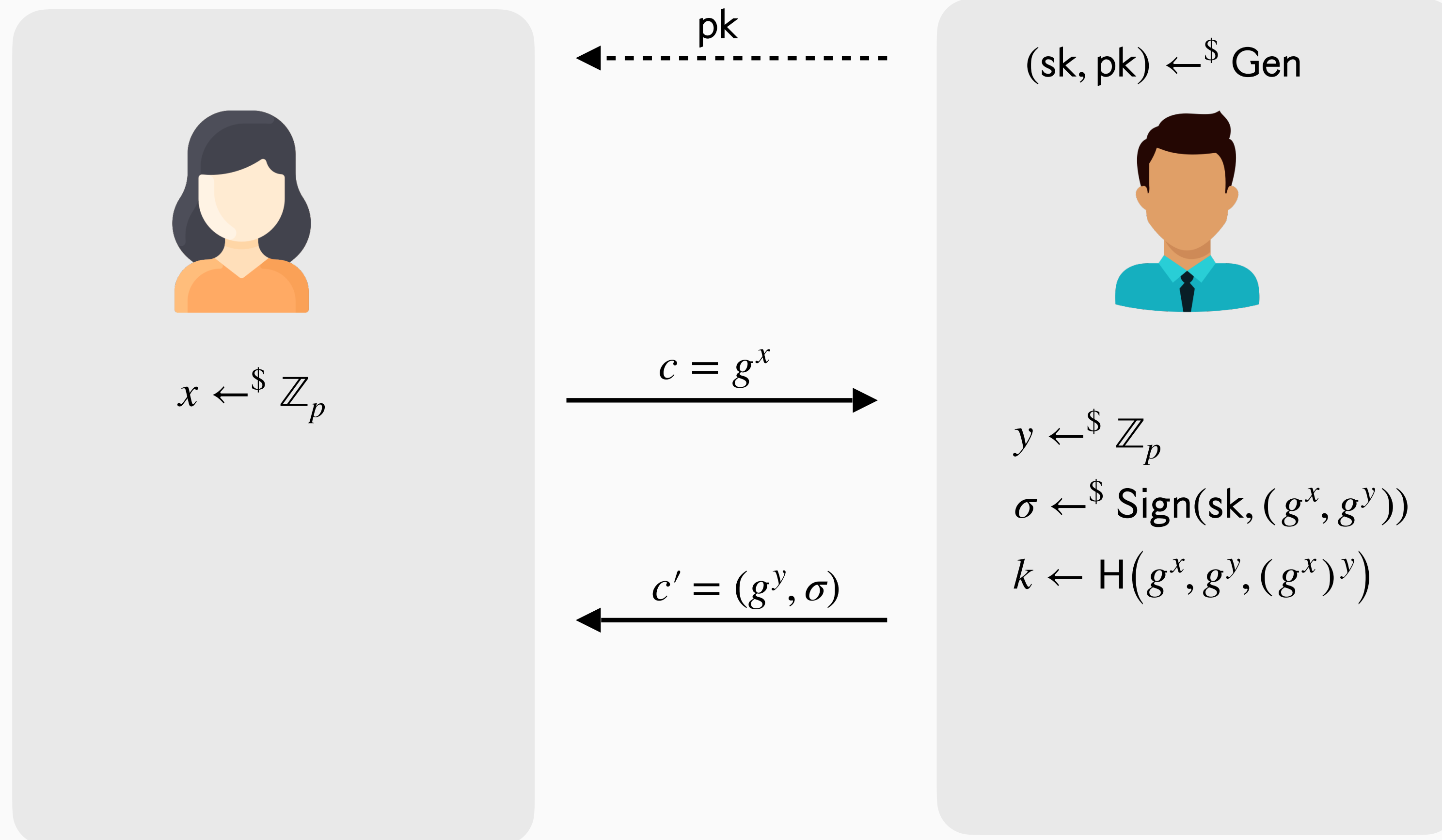
Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:



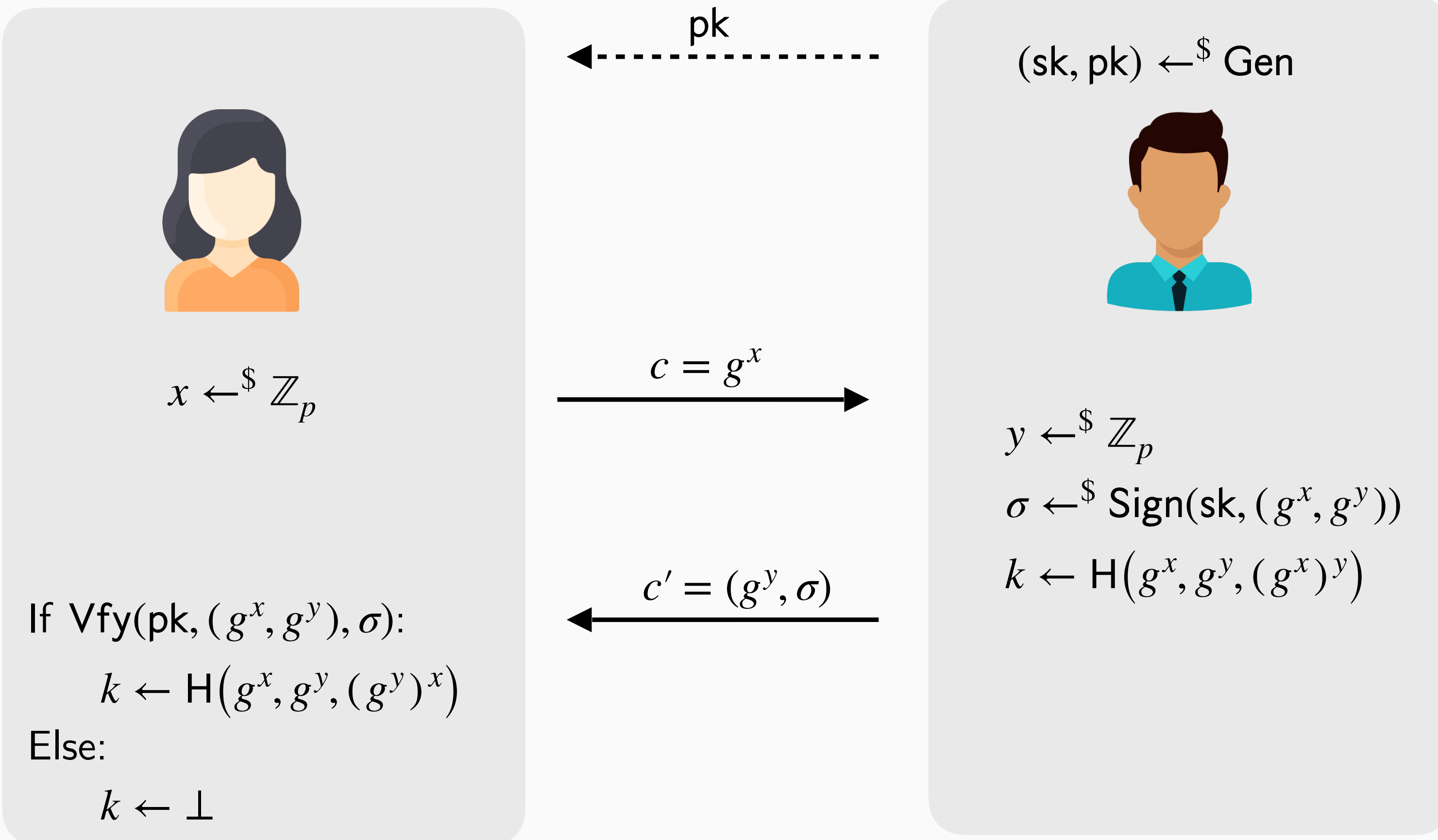
Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:




Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:



Signed Diffie-Hellman


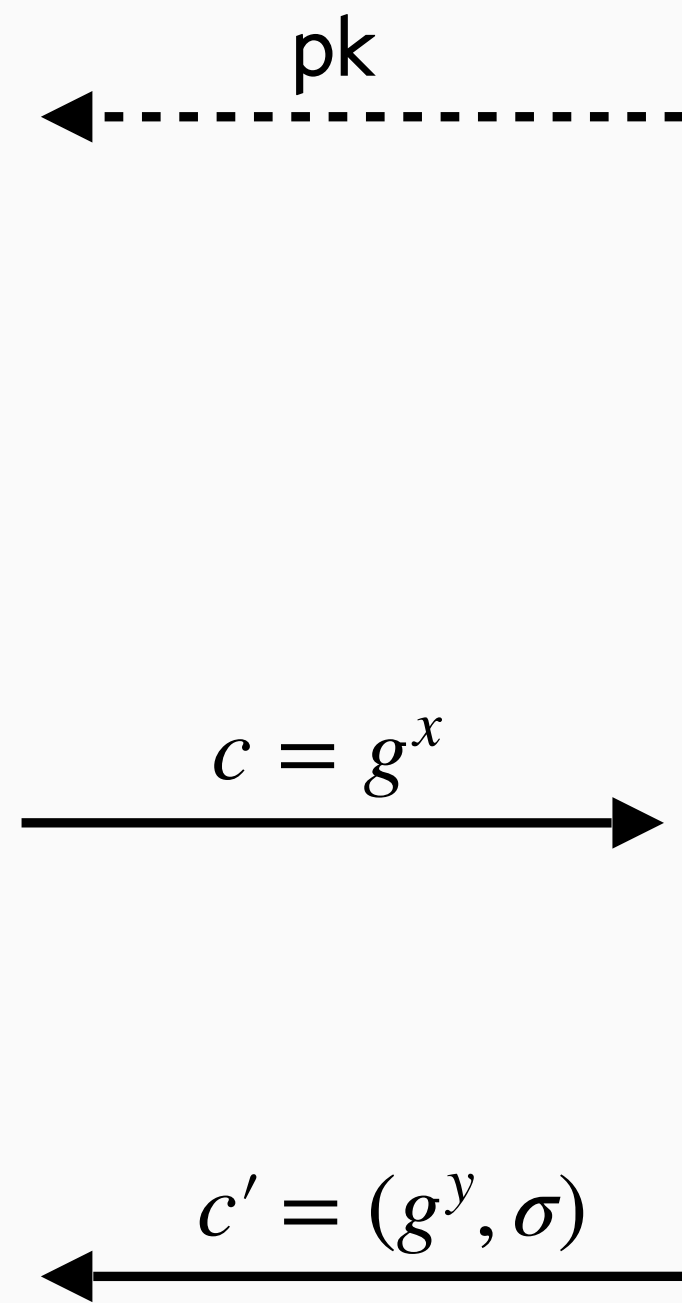
Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:


$$x \leftarrow^{\$} \mathbb{Z}_p$$

If $\text{Vfy}(\text{pk}, (g^x, g^y), \sigma)$:

$$k \leftarrow H(g^x, g^y, (g^y)^x)$$

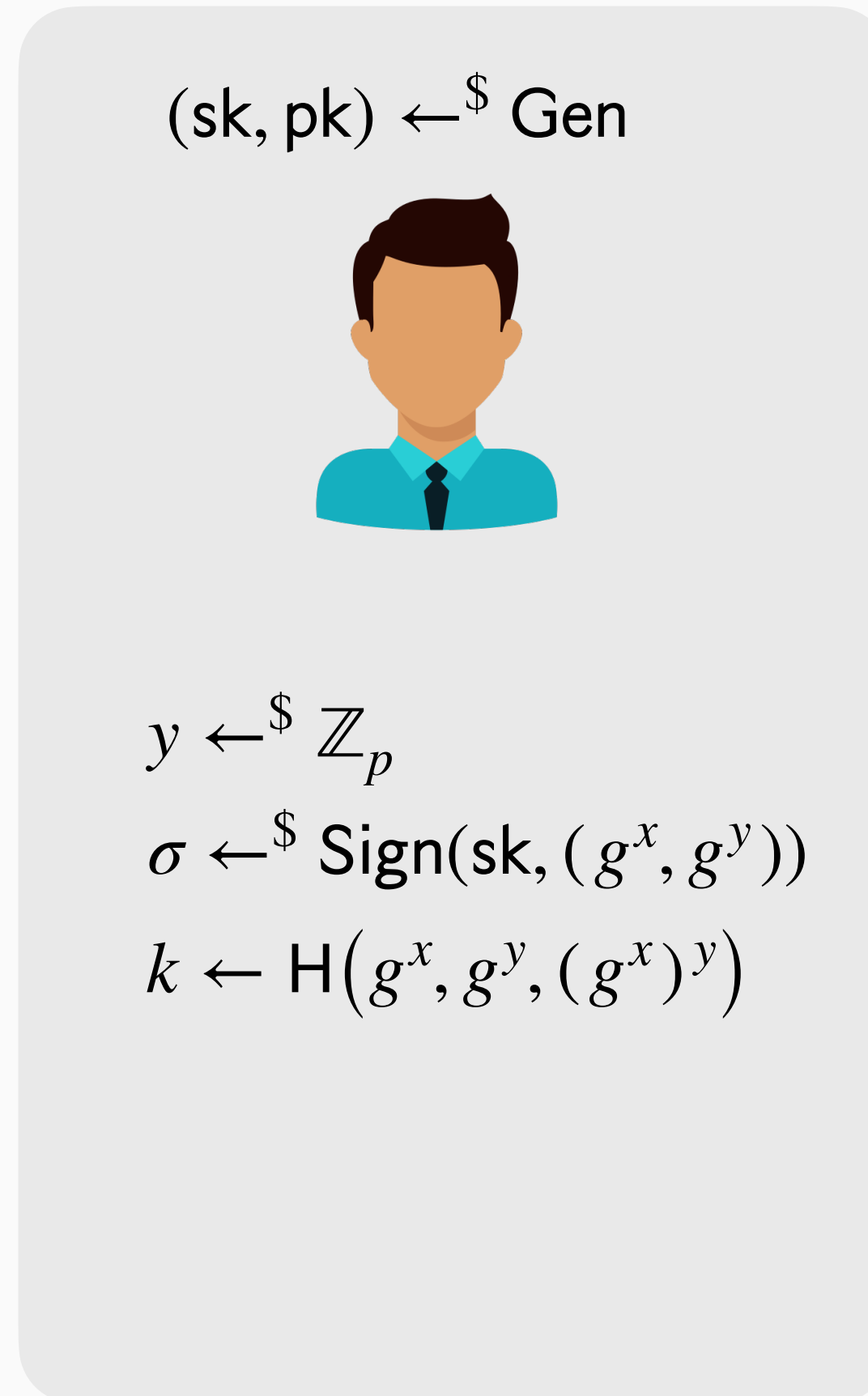
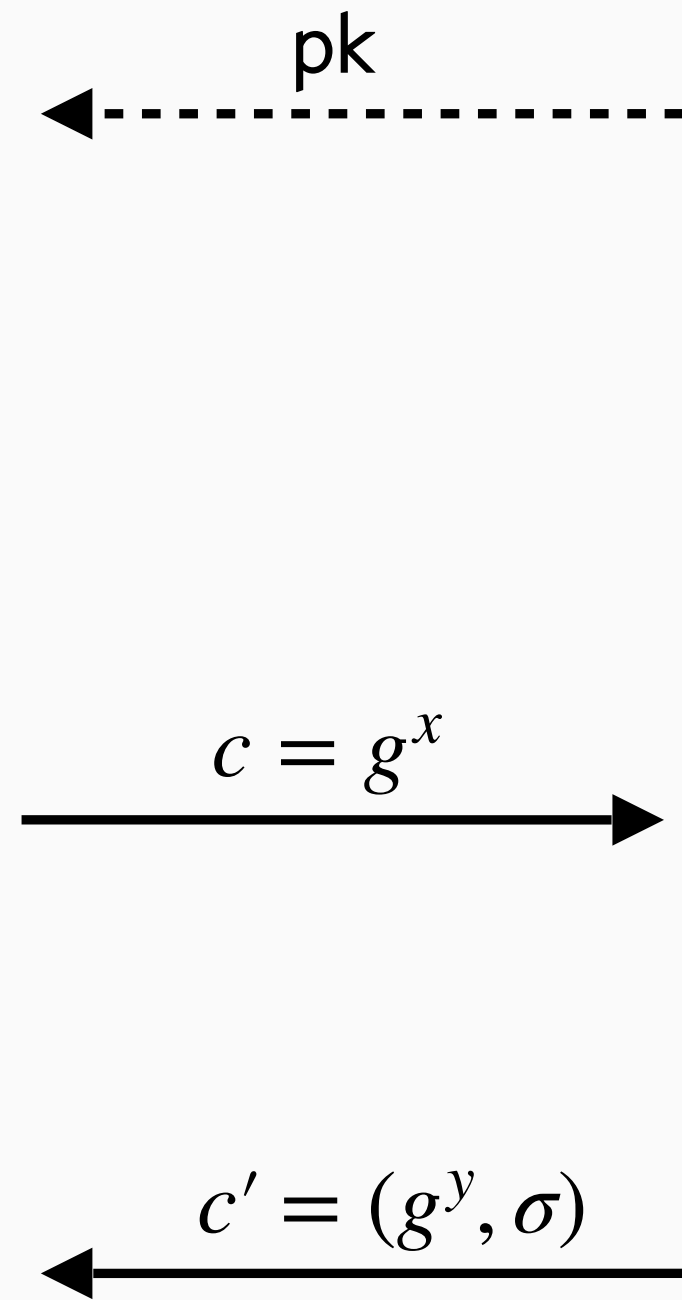
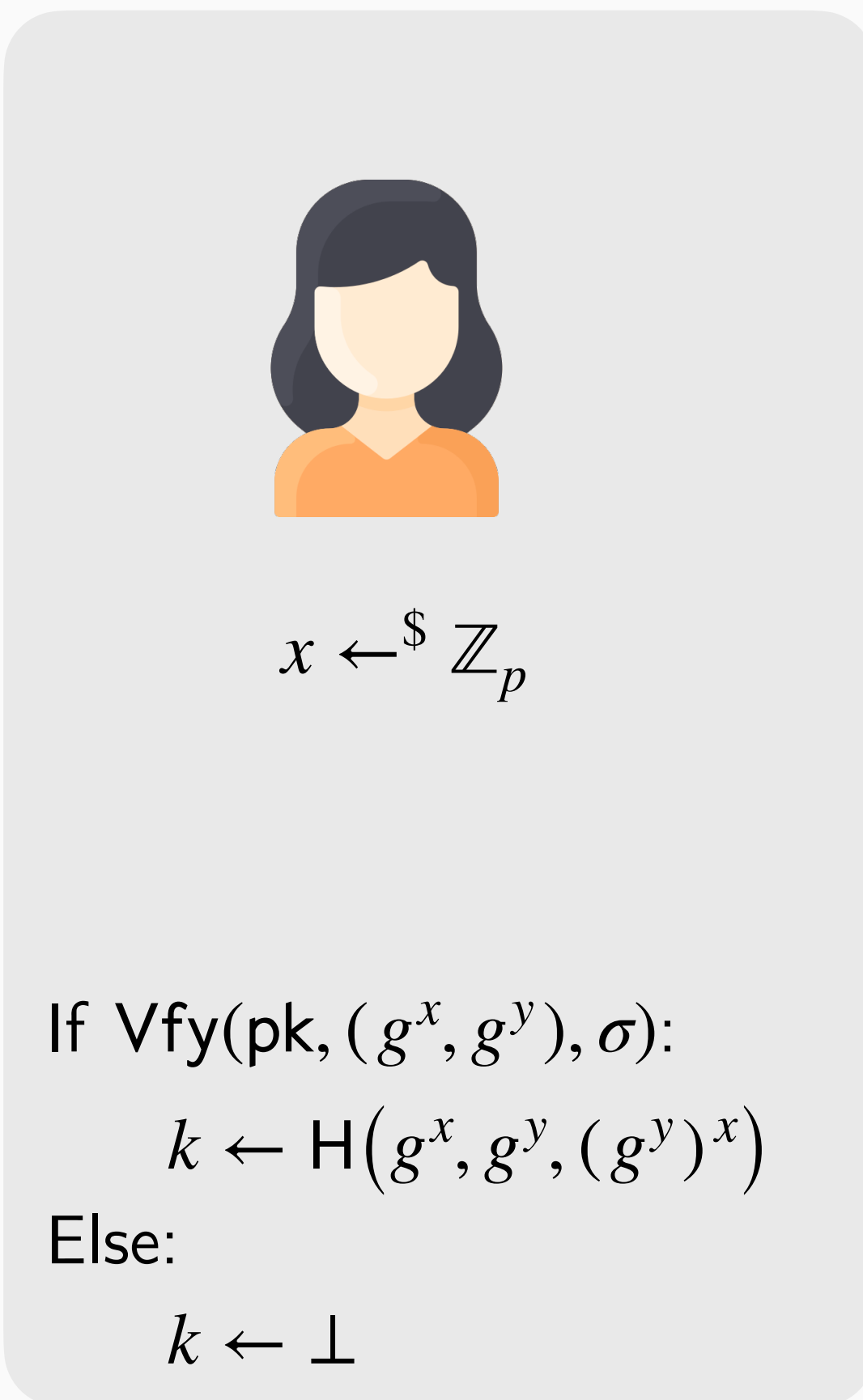
Else:

$$k \leftarrow \perp$$

$$(\text{sk}, \text{pk}) \leftarrow^{\$} \text{Gen}$$
$$y \leftarrow^{\$} \mathbb{Z}_p$$
$$\sigma \leftarrow^{\$} \text{Sign}(\text{sk}, (g^x, g^y))$$
$$k \leftarrow H(g^x, g^y, (g^x)^y)$$

Unilateral Authenticated Key Exchange
 $\text{KE} = (\text{Gen}, \text{Init}, \text{Resp}, \text{Recv})$

Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:



Unilateral Authenticated Key Exchange

$\text{KE} = (\text{Gen}, \text{Init}, \text{Resp}, \text{Recv})$

$(\text{sk}, \text{pk}) \leftarrow^{\$} \text{Gen}$

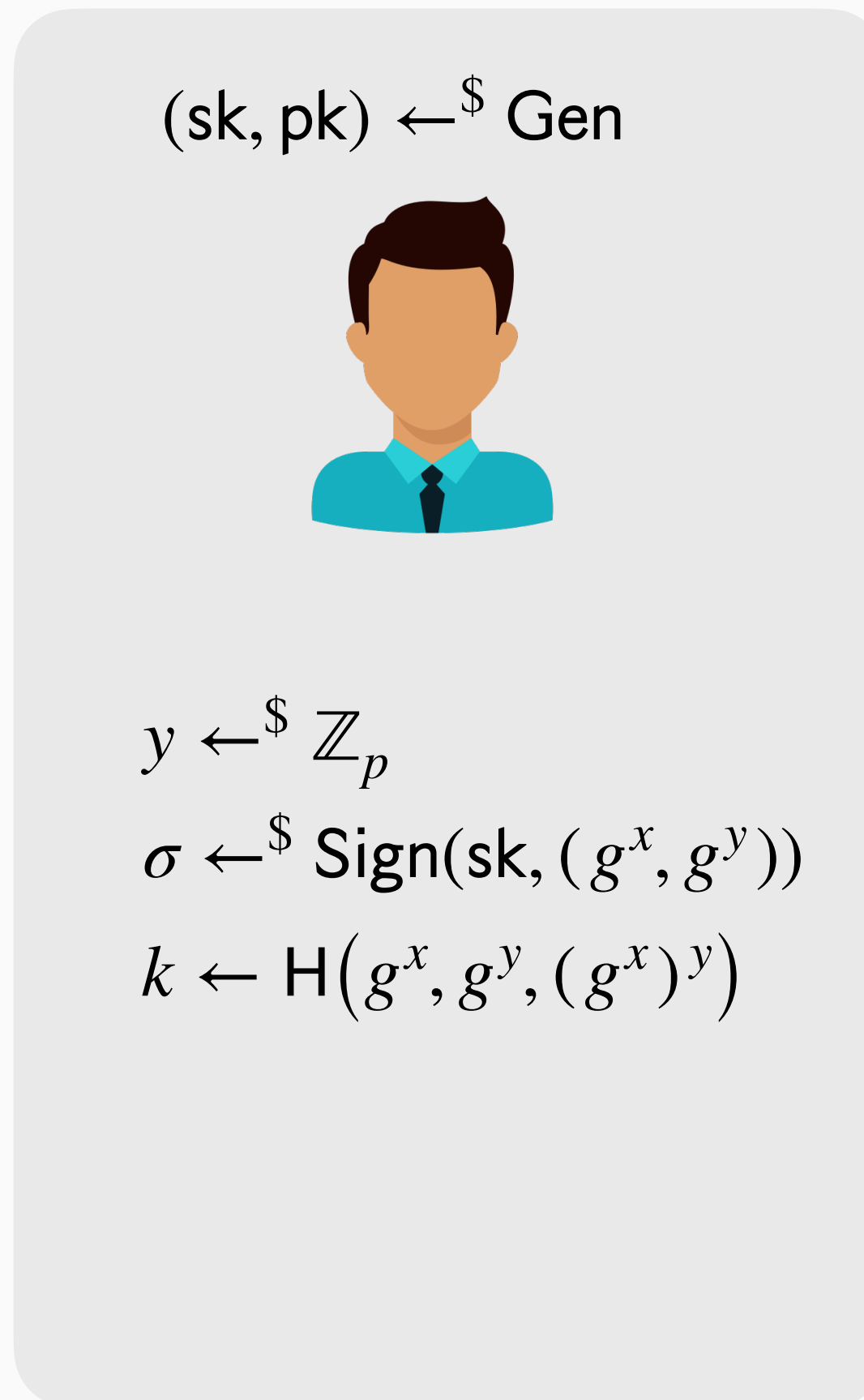
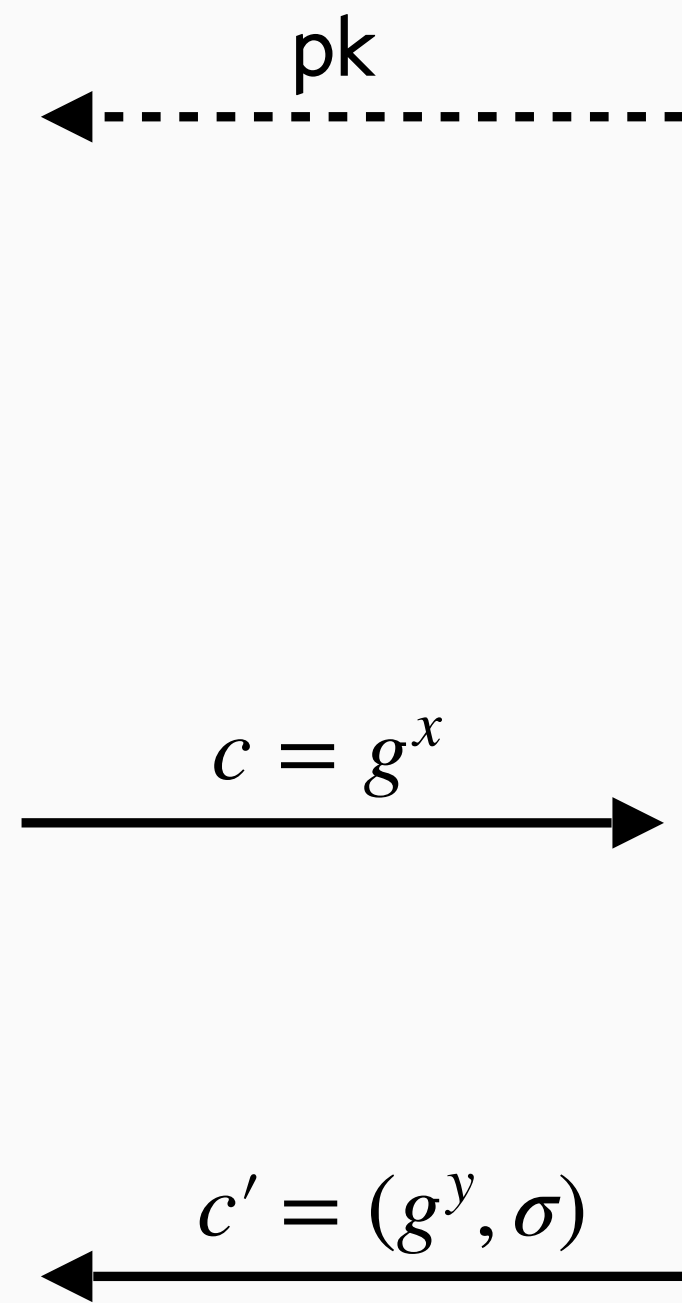
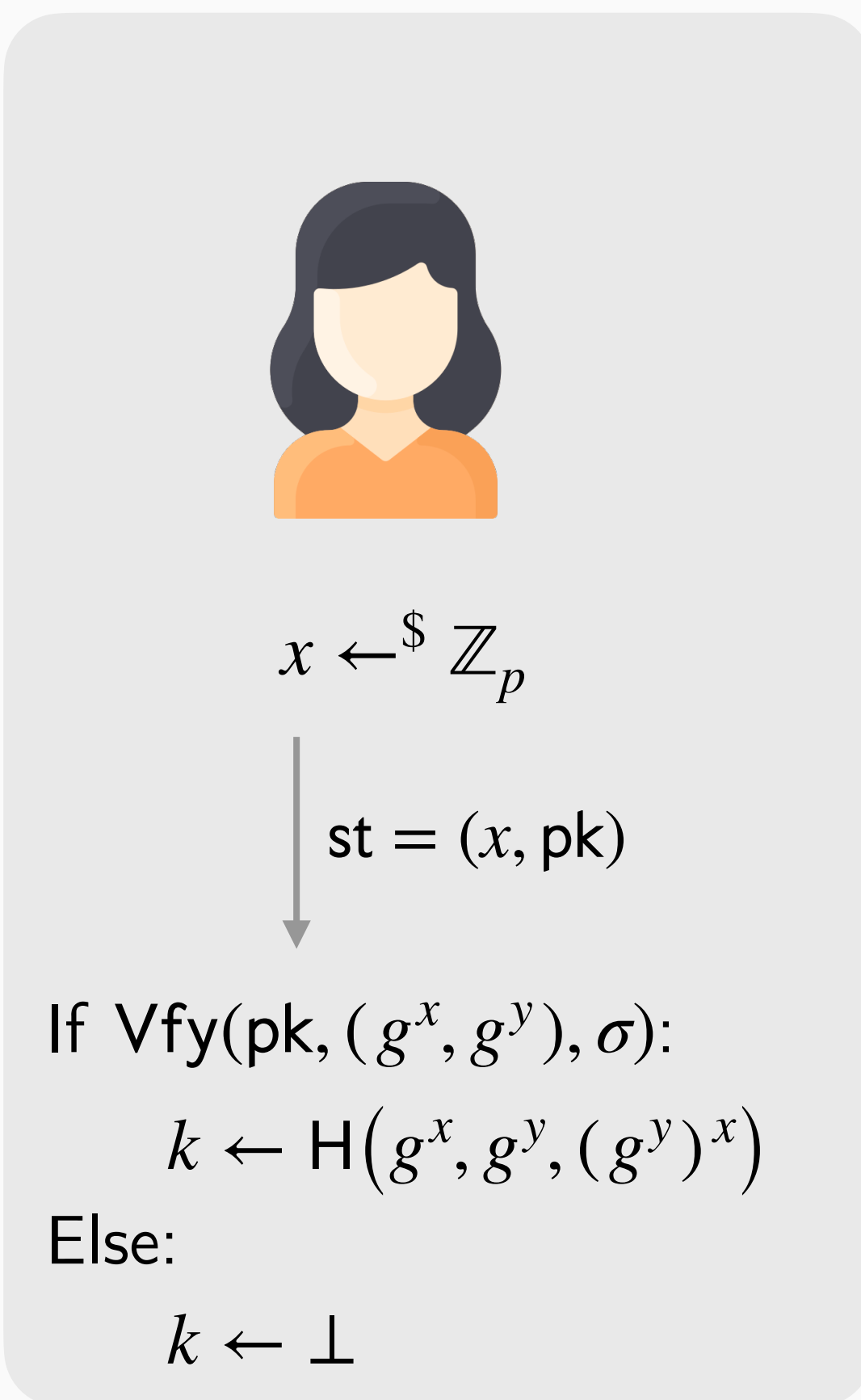
$(\text{st}, c) \leftarrow^{\$} \text{Init}(\text{pk})$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}, c)$

$k / \perp \leftarrow \text{Recv}(\text{st}, c')$

Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:



Unilateral Authenticated Key Exchange

$\text{KE} = (\text{Gen}, \text{Init}, \text{Resp}, \text{Recv})$

$(sk, pk) \leftarrow^{\$} \text{Gen}$

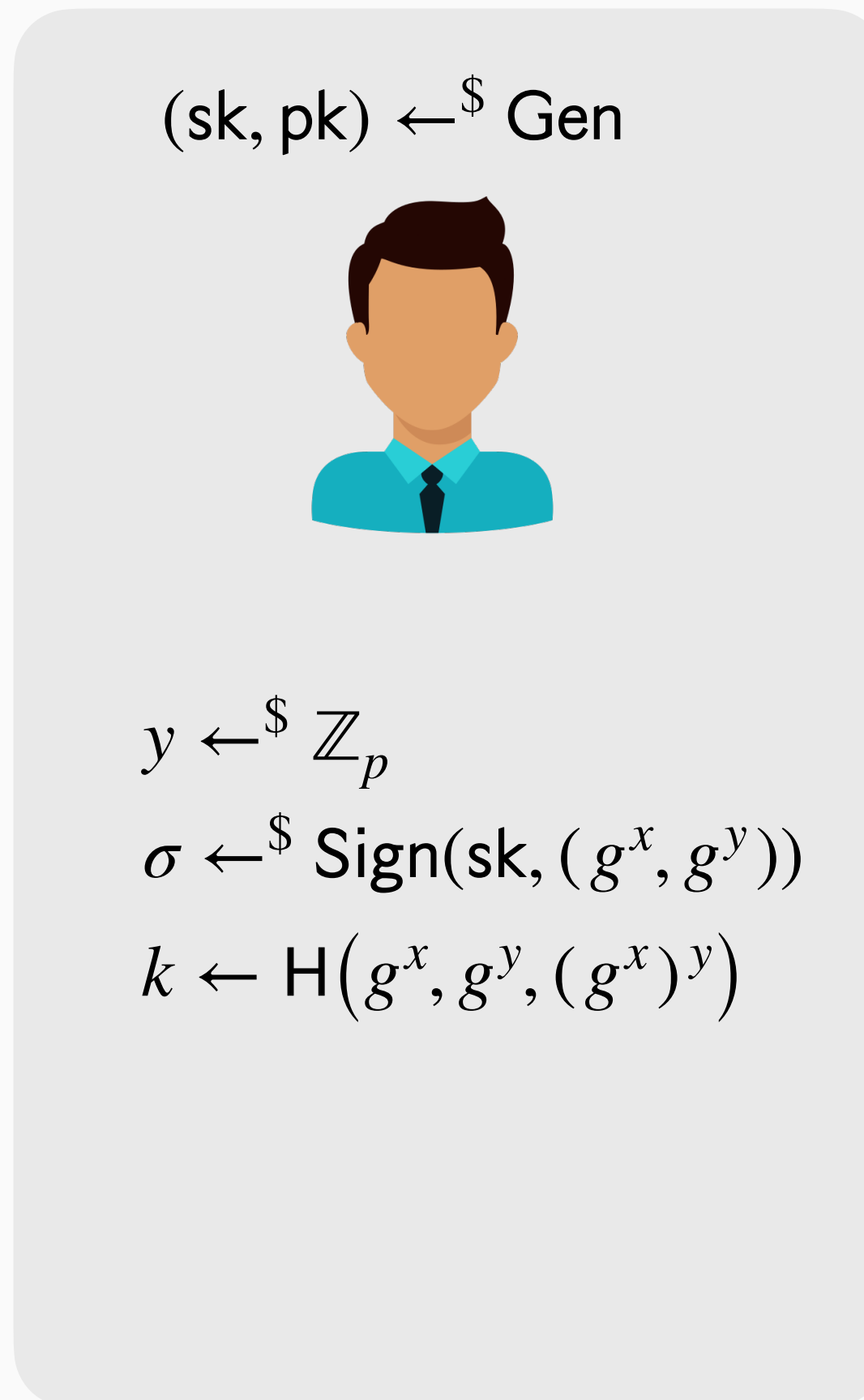
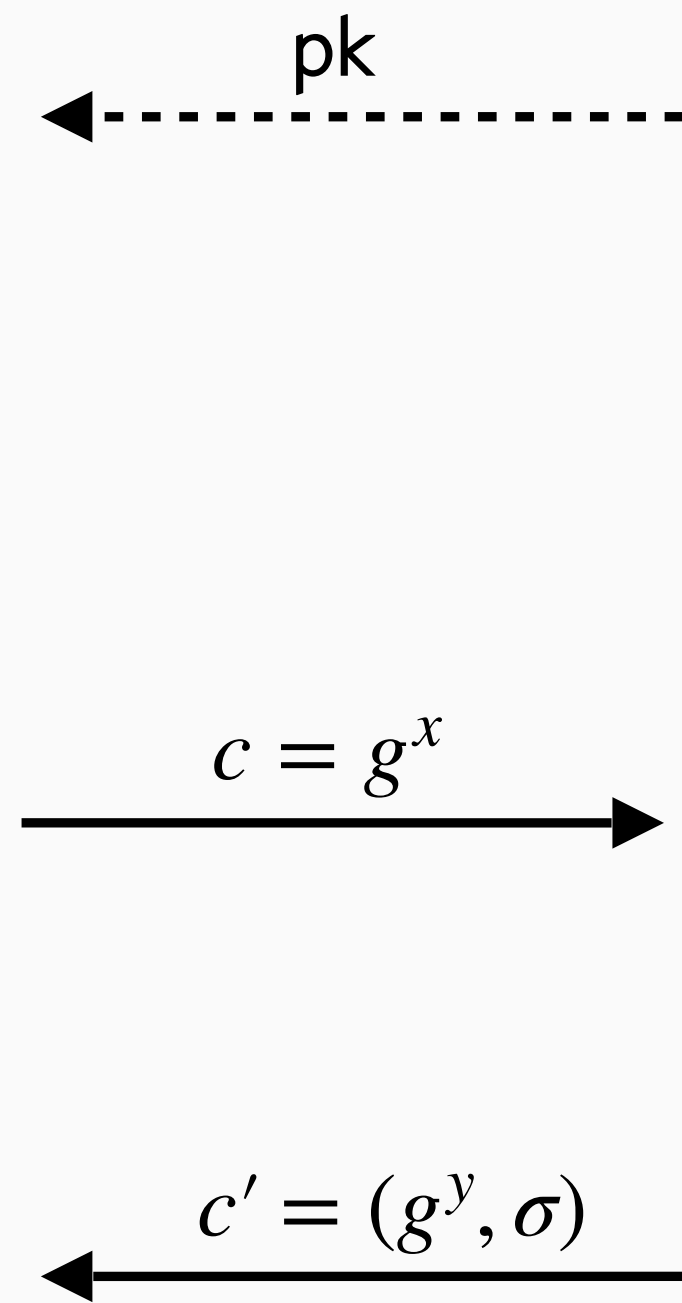
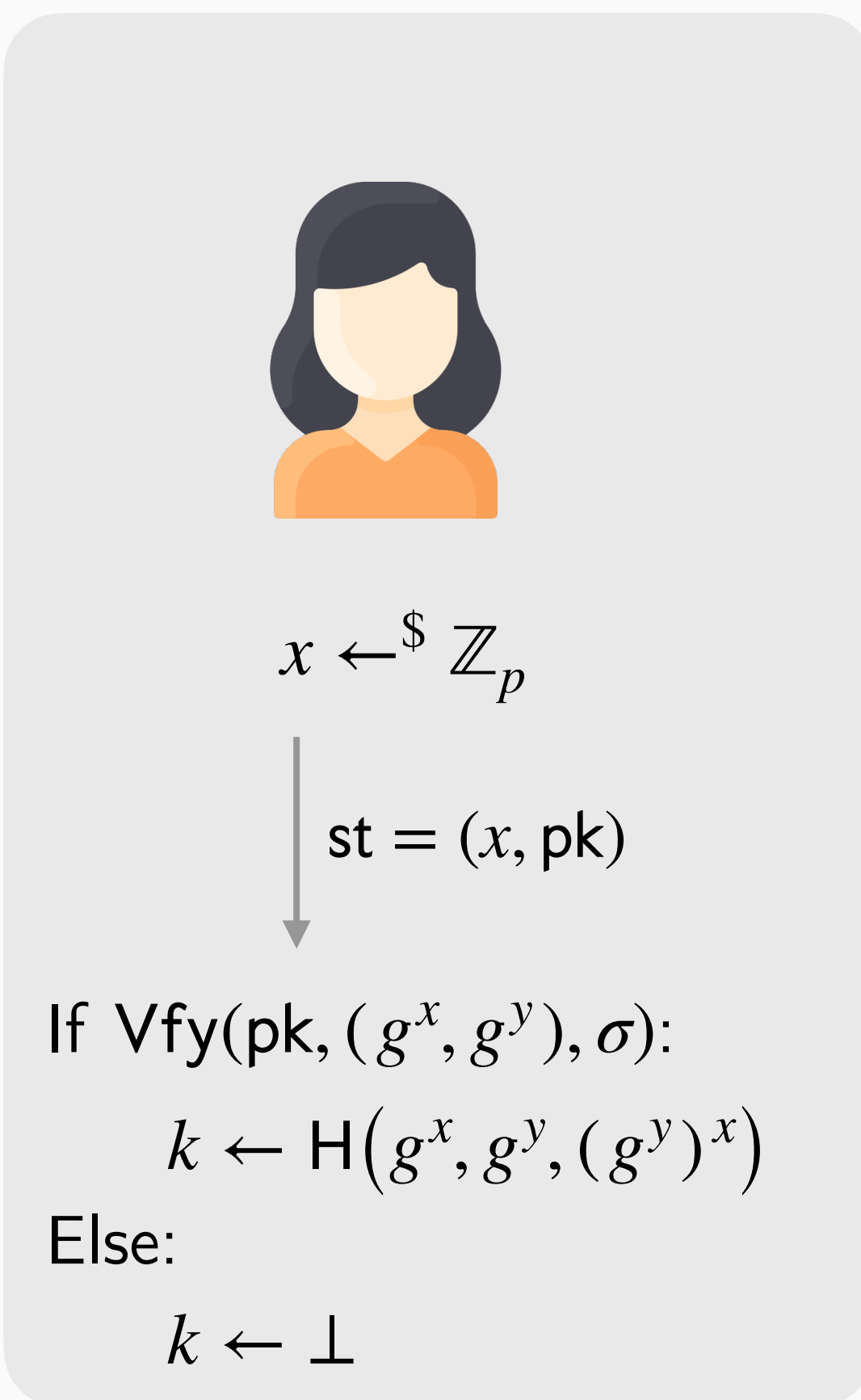
$(st, c) \leftarrow^{\$} \text{Init}(pk)$

$(k, c') \leftarrow^{\$} \text{Resp}(sk, c)$

$k / \perp \leftarrow \text{Recv}(st, c')$

Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:



Unilateral Authenticated Key Exchange

$\text{KE} = (\text{Gen}, \text{Init}, \text{Resp}, \text{Recv})$

$(sk, pk) \leftarrow^{\$} \text{Gen}$

$(st, c) \leftarrow^{\$} \text{Init}(pk)$

$(k, c') \leftarrow^{\$} \text{Resp}(sk, c)$

$k / \perp \leftarrow \text{Recv}(st, c')$

Correctness: honest execution results in the same session keys

Game-based Security Model for KE (informal)

Overview

- Focus on 2-message protocols
- Multi-user multi-session setting
 - Initiator sessions are identified by an index i and a state st_i
 - Responders are identified by an index j and long-term key pair (sk_j, pk_j)
- Adversary can
 - Initiate sessions and send arbitrary messages
 - Use their own (potentially malicious) long-term keys
 - Corrupt secret keys
 - Expose session states
 - Reveal session keys

Game-based Security Model for KE (informal)

Overview

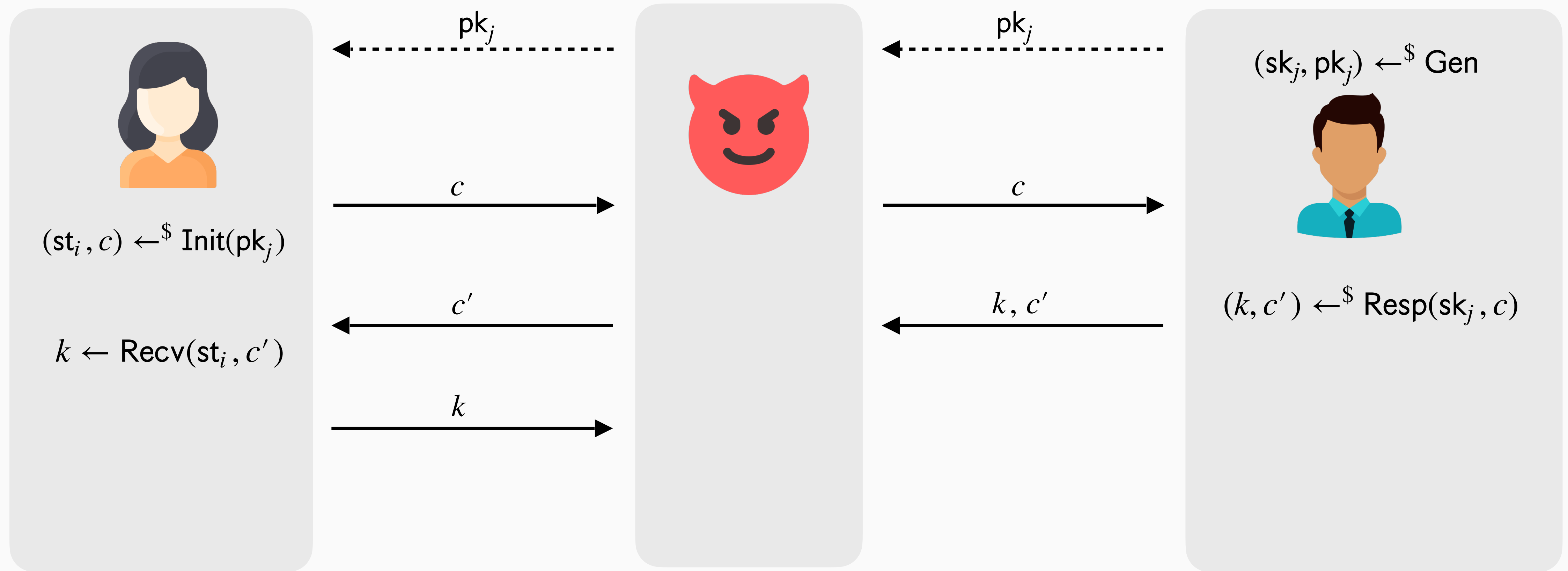
- Focus on 2-message protocols
- Multi-user multi-session setting
 - Initiator sessions are identified by an index i and a state st_i
 - Responders are identified by an index j and long-term key pair (sk_j, pk_j)
- Adversary can
 - Initiate sessions and send arbitrary messages
 - Use their own (potentially malicious) long-term keys
 - Corrupt secret keys
 - Expose session states
 - Reveal session keys

Goal: distinguish real from random session keys of *fresh* sessions

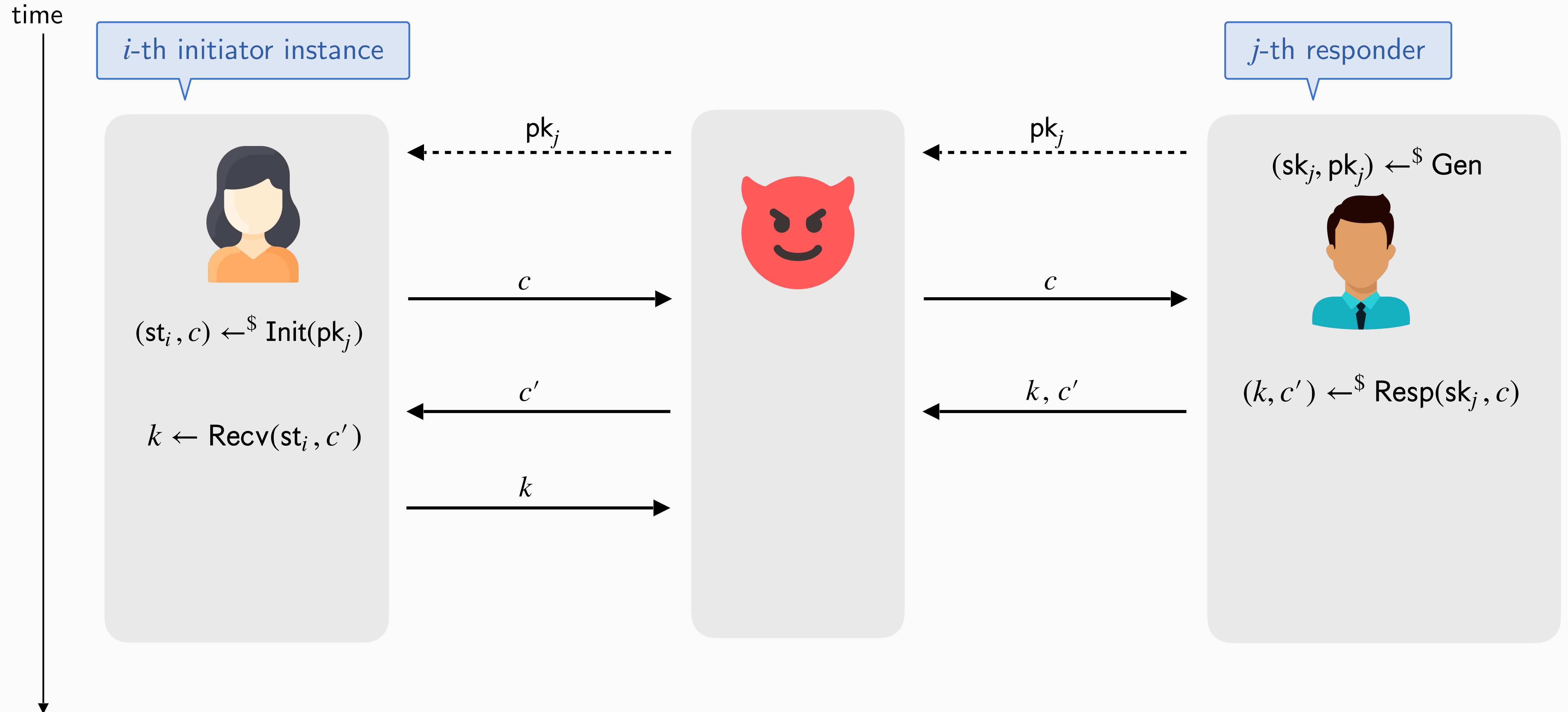
- Forward secrecy
- (Explicit) unilateral authentication

Game-based Security Model for KE (informal)

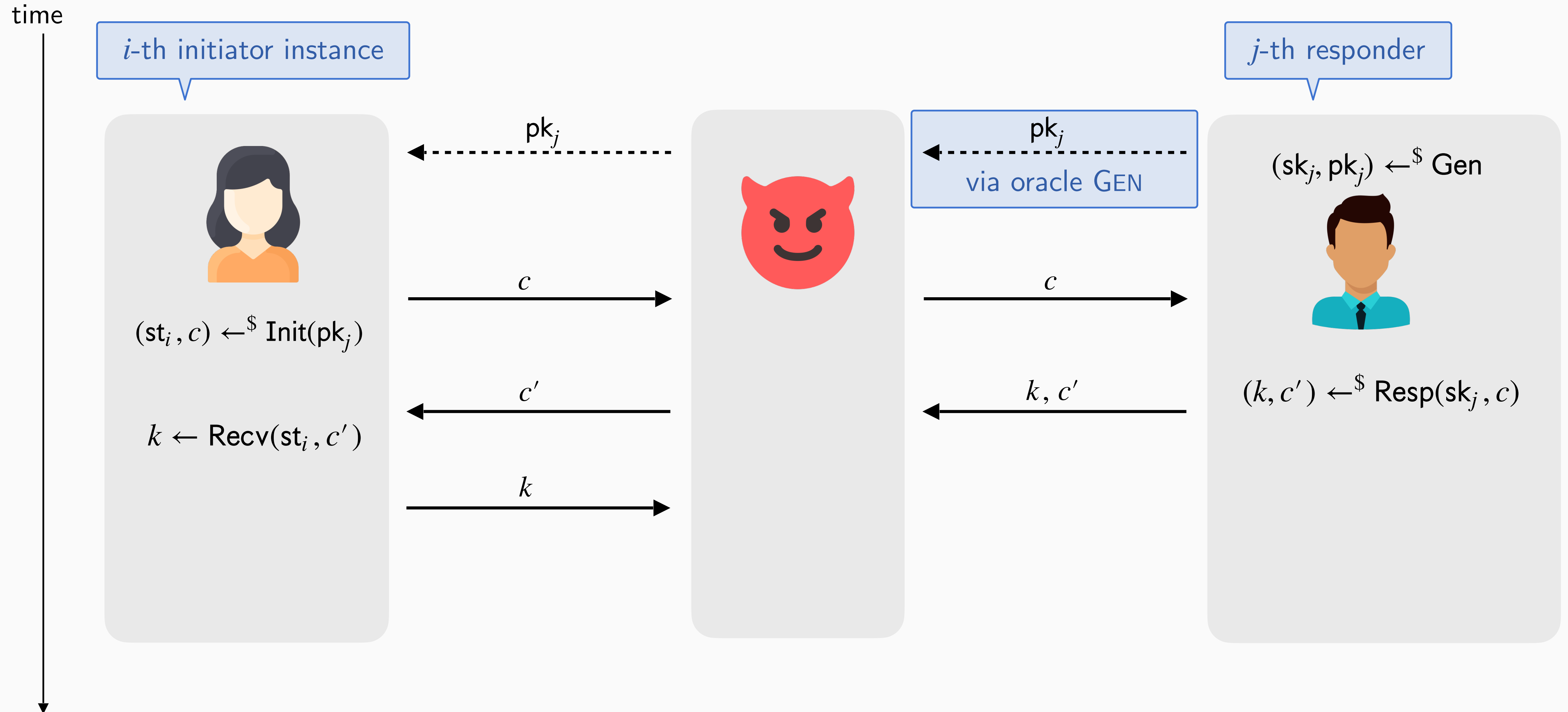
time



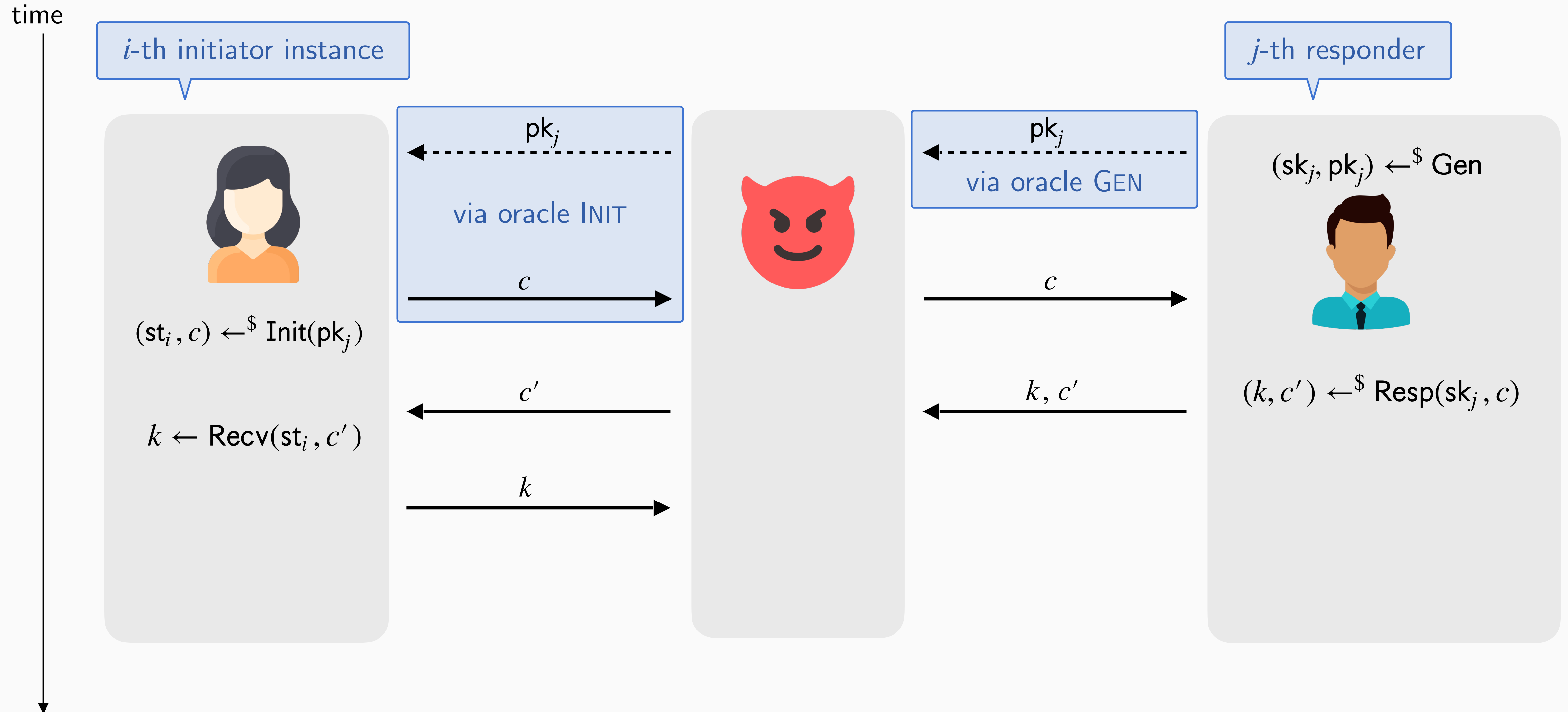
Game-based Security Model for KE (informal)



Game-based Security Model for KE (informal)




Game-based Security Model for KE (informal)



Game-based Security Model for KE (informal)

time

i-th initiator instance




$(st_i, c) \leftarrow^{\$} \text{Init}(pk_j)$

$k \leftarrow \text{Recv}(st_i, c')$

pk_j

via oracle INIT

c



pk_j

via oracle GEN


c

via oracle RESPOND

k, c'

j-th responder

$(sk_j, pk_j) \leftarrow^{\$} \text{Gen}$




$(k, c') \leftarrow^{\$} \text{Resp}(sk_j, c)$

Game-based Security Model for KE (informal)

time


i-th initiator instance



$(st_i, c) \leftarrow^{\$} \text{Init}(pk_j)$
 $k \leftarrow \text{Recv}(st_i, c')$

pk_j
via oracle INIT
 c


c'
via oracle RECEIVE
 k



pk_j
via oracle GEN

c
via oracle RESPOND
 k, c'

j-th responder

$(sk_j, pk_j) \leftarrow^{\$} \text{Gen}$

 $(k, c') \leftarrow^{\$} \text{Resp}(sk_j, c)$

Game-based Security Model for KE (formal)

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$b' \leftarrow \mathcal{A}$
Return b'

INIT(pk)

$(st_n, c) \leftarrow^{\$} \text{Init}(pk)$

Return c

RECEIVE(i, c')

$k \leftarrow \text{Recv}(st_i, c')$

Return k

GEN

$(sk_m, pk_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c)

$(k, c') \leftarrow^{\$} \text{Resp}(sk_j, c)$

Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (formal)

Book-keeping and consistency checks

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$

$b' \leftarrow \mathcal{A}$
Return b'

INIT(pk)

$n++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$

Return c

RECEIVE(i, c') $\forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$

$k \leftarrow \text{Recv}(\text{st}_i, c')$

Return k

GEN

$m++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c) $\forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$

Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (formal)

Partnering (matching conversations)

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$

$b' \leftarrow \mathcal{A}$
Return b'

INIT(pk)

$n++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c') $\forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$

 $k \leftarrow \text{Recv}(\text{st}_i, c')$

Return k

GEN

$m++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c) $\forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$

Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (formal)

Partnering (matching conversations)

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$

$b' \leftarrow \mathcal{A}$
Return b'

INIT(pk)

$n++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c') $\forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$

Return k

GEN

$m++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c) $\forall j \in [m]$

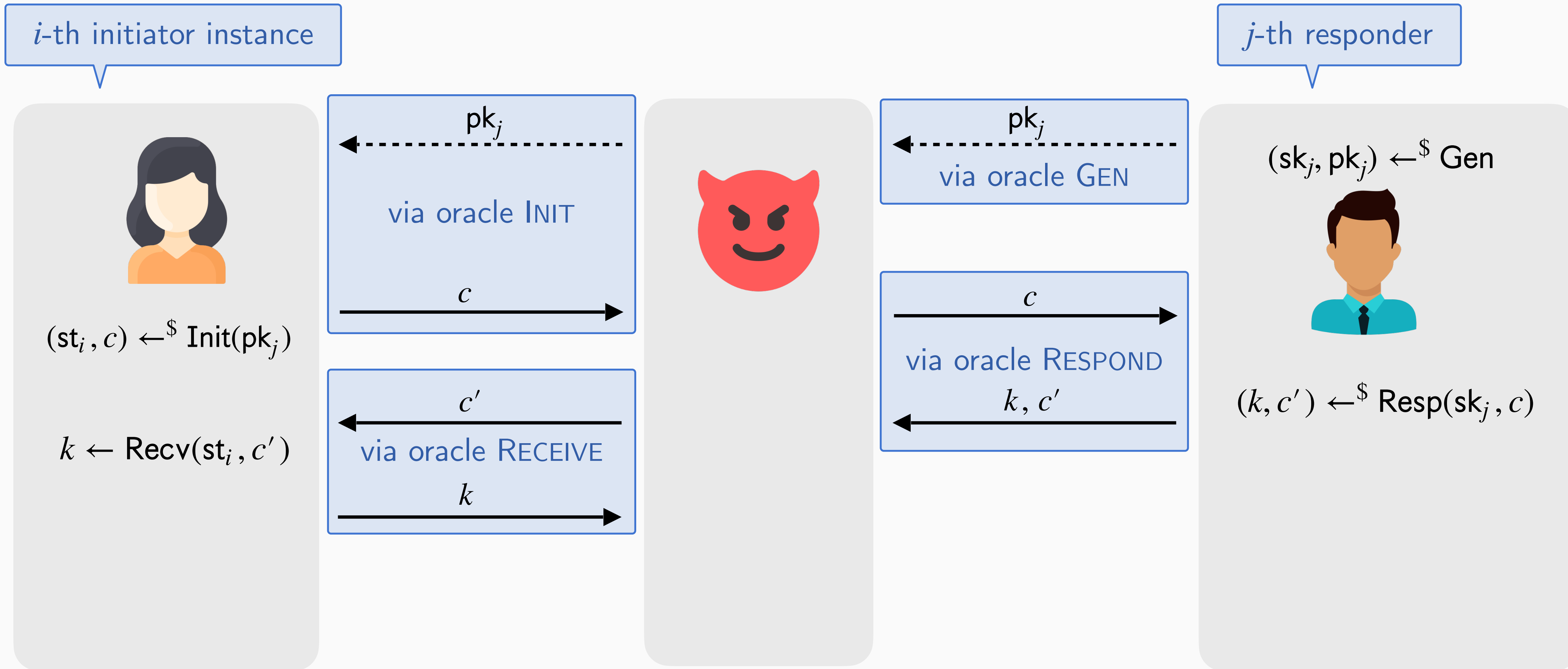
$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$

Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

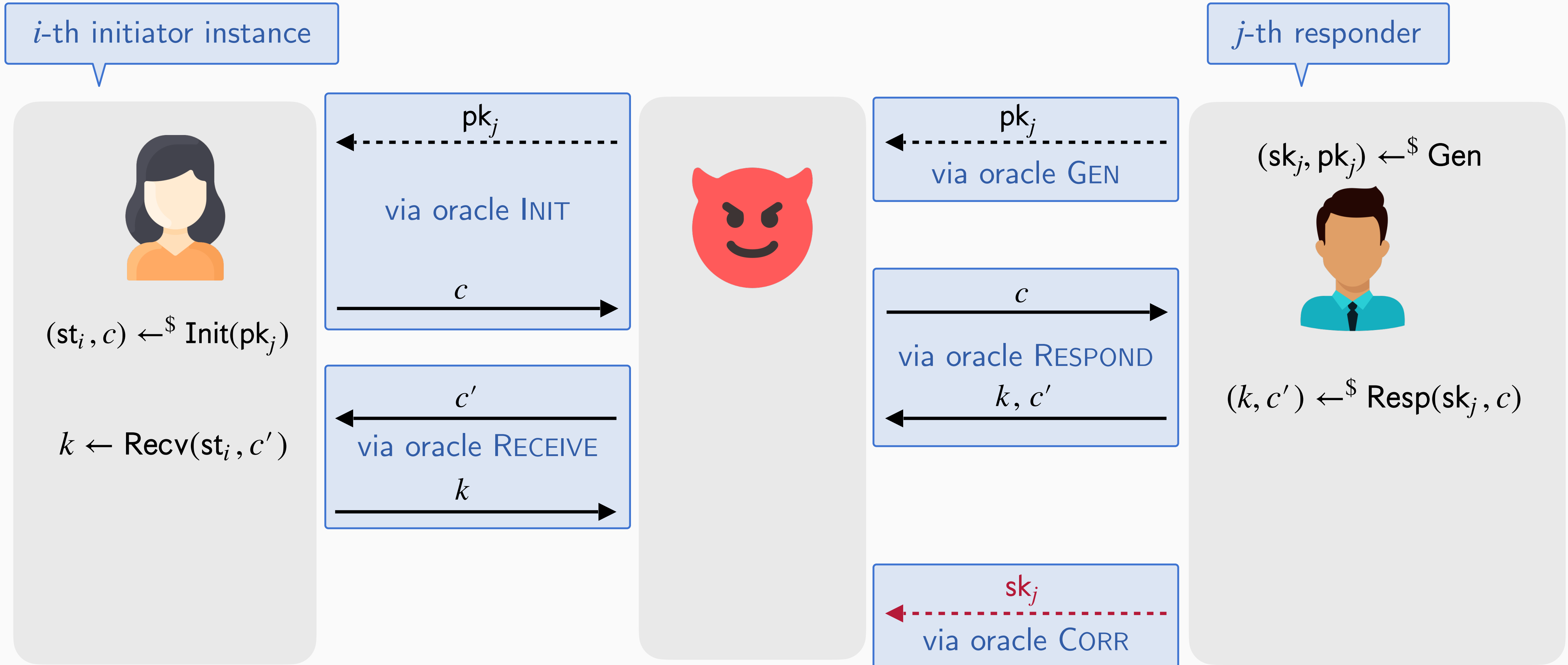
Game-based Security Model for KE (informal)

time



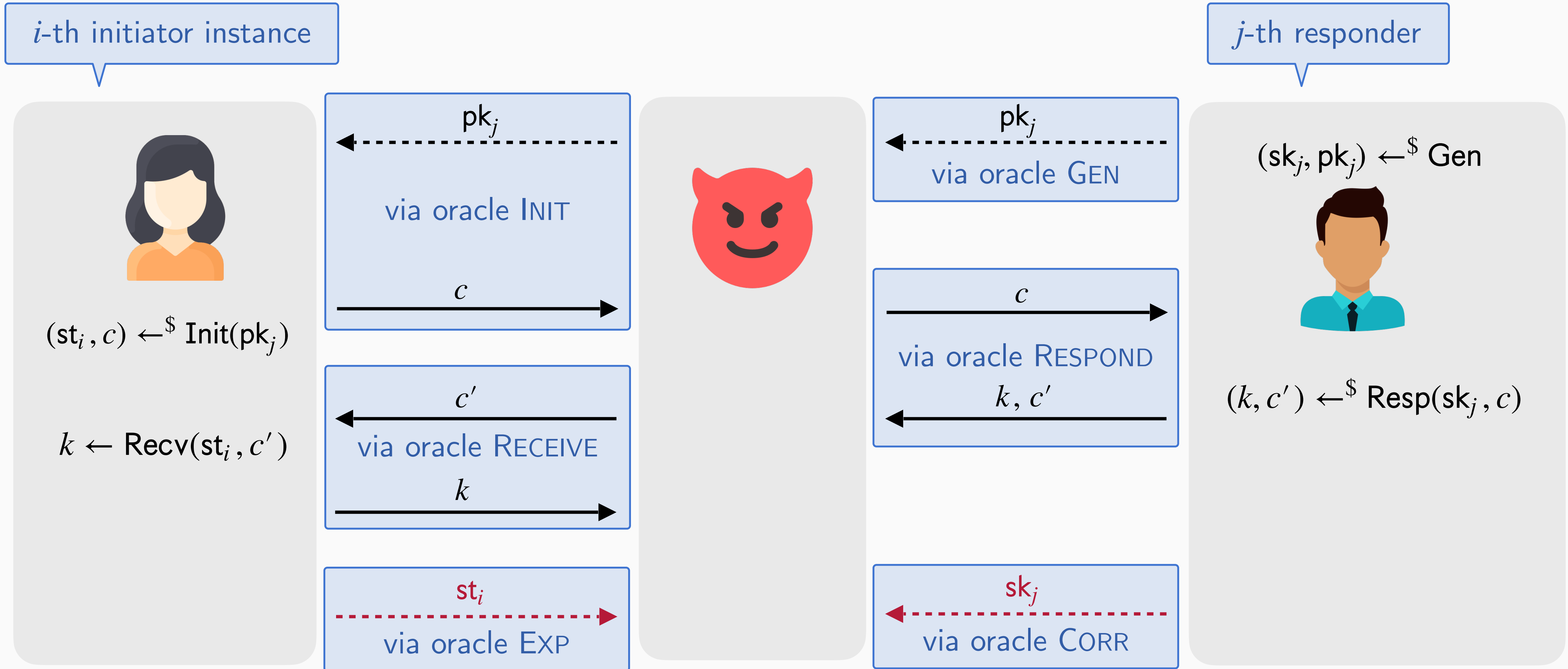
Game-based Security Model for KE (informal)

time



Game-based Security Model for KE (informal)

time



Game-based Security Model for KE (formal)

Corruptions and
state exposures

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$

$b' \leftarrow \mathcal{A}$
Return b'

INIT(pk)

$n++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c') $\forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$

Return k

GEN

$m++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c) $\forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$

Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (formal)

Corruptions and
state exposures

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$

$b' \leftarrow \mathcal{A}$
Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
Return sk_j

EXP(i) $\quad \forall i \in [n]$

$XP \leftarrow XP \cup \{i\}$
Return st_i

INIT(pk)

$n++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c') $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$

Return k

GEN

$m++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$


Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (informal)

time


i-th initiator instance



$(st_i, c) \leftarrow^{\$} \text{Init}(pk_j)$
 $k \leftarrow \text{Recv}(st_i, c')$

pk_j
via oracle INIT
 c


c'
via oracle RECEIVE
 k



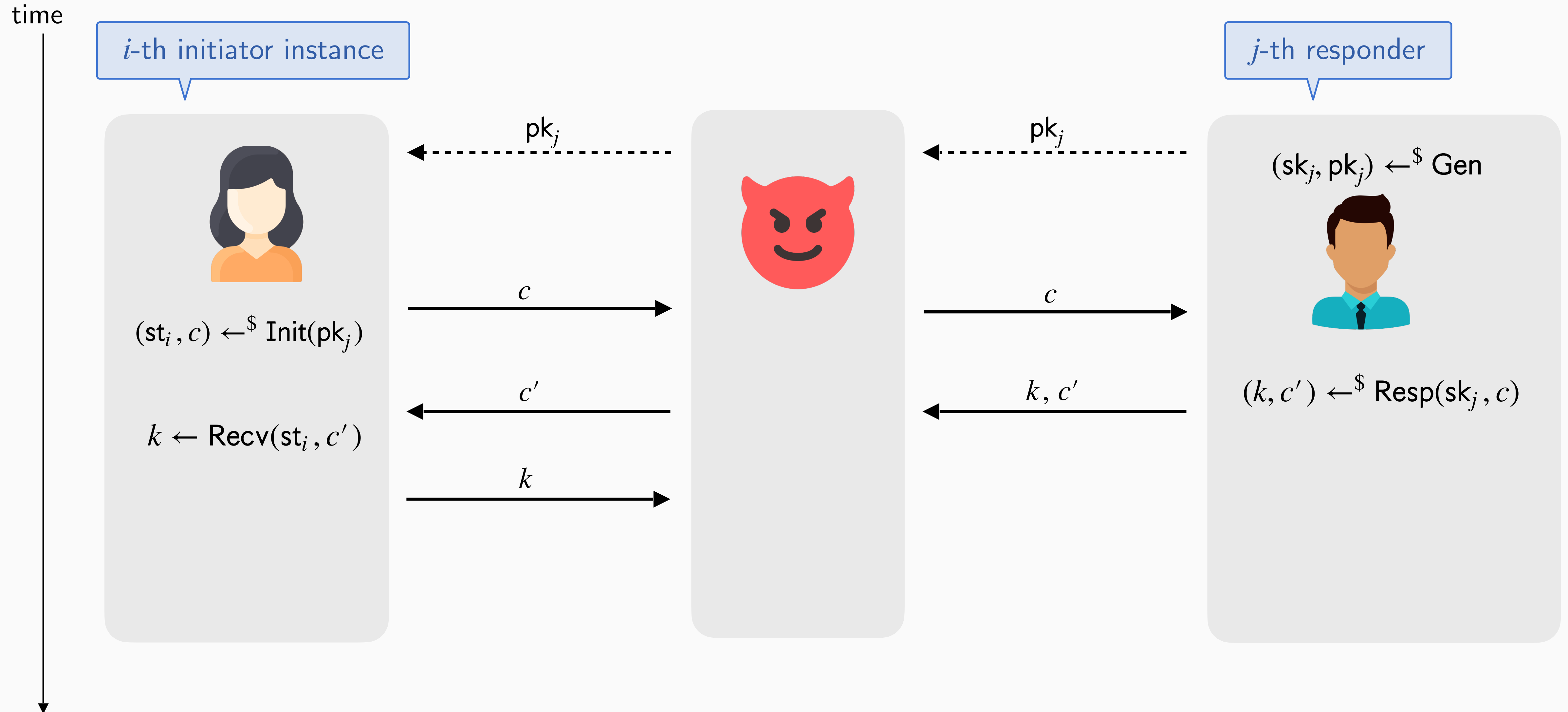
pk_j
via oracle GEN

c
via oracle RESPOND
 k, c'

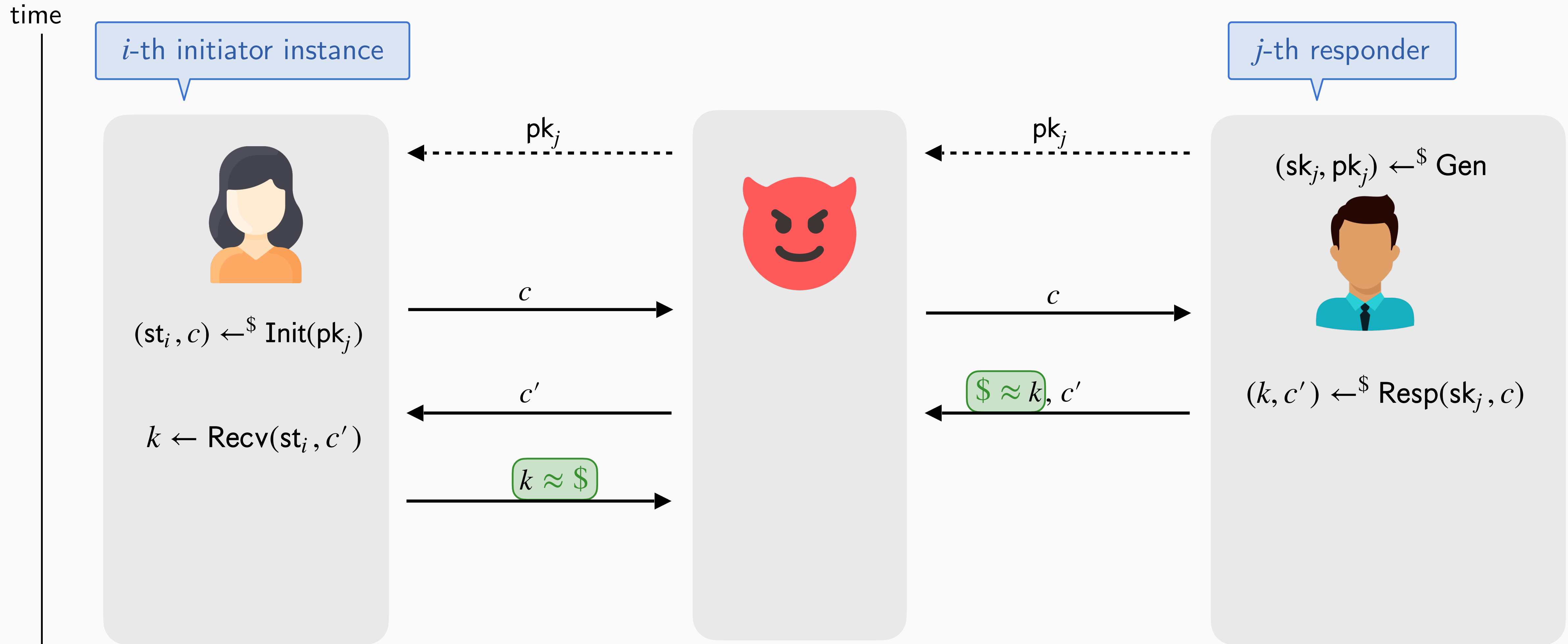
j-th responder

$(sk_j, pk_j) \leftarrow^{\$} \text{Gen}$

 $(k, c') \leftarrow^{\$} \text{Resp}(sk_j, c)$

Game-based Security Model for KE (informal)



Game-based Security Model for KE (informal)



Key indistinguishability and trivial attacks: Under which conditions should k look like random?

Game-based Security Model for KE (formal)

CHALLENGE vs. REVEAL

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$

$b' \leftarrow \mathcal{A}$
Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
Return sk_j

EXP(i) $\quad \forall i \in [n]$

$XP \leftarrow XP \cup \{i\}$
Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$

If ch
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$

Return k

GEN

$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

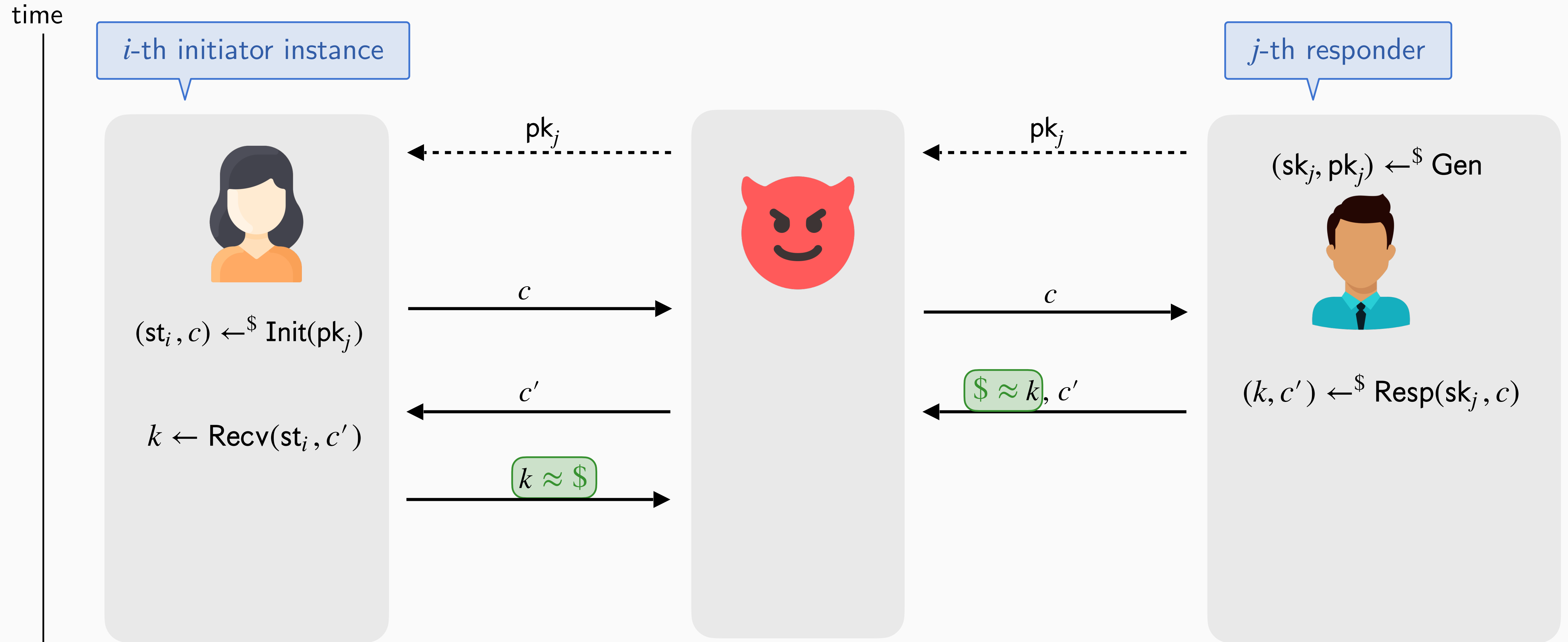
RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If ch
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$

Return (k, c')

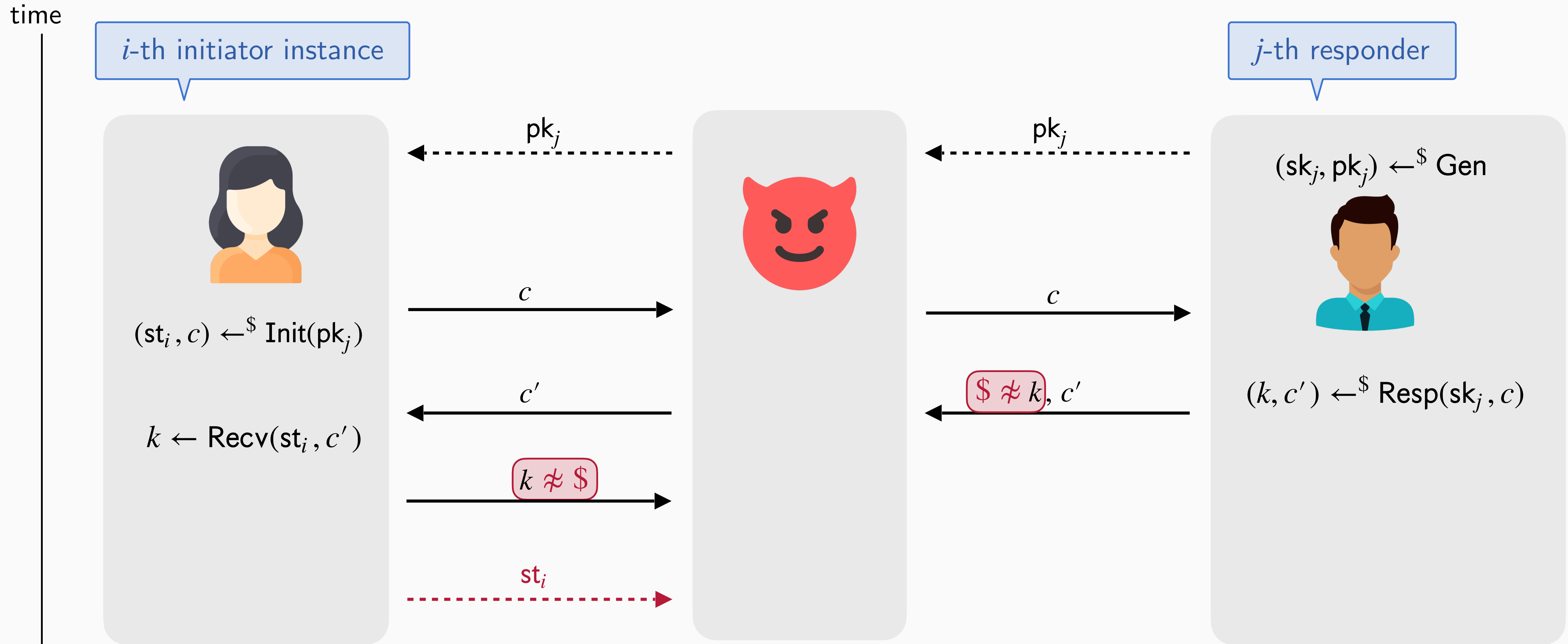
$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (informal)



Key Indistinguishability and trivial attacks: Under which conditions should k look like random?

Game-based Security Model for KE (informal)



Key Indistinguishability and trivial attacks: Under which conditions should k look like random?

- State exposure always allows to trivially distinguish.

Game-based Security Model for KE (formal)

CHALLENGE vs. REVEAL

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$

$b' \leftarrow \mathcal{A}$
Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
Return sk_j

EXP(i) $\quad \forall i \in [n]$

$XP \leftarrow XP \cup \{i\}$
Return st_i

INIT(pk)

$n++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$

If ch
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$

Return k

GEN

$m++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If ch
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$

Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (formal)

CHALLENGE vs. REVEAL

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$

$b' \leftarrow \mathcal{A}$
Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
Return sk_j

EXP(i) $\quad \forall i \in [n]$

$XP \leftarrow XP \cup \{i\}$
Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$

If $ch \wedge i \notin XP$
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$

Return k

GEN

$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If $ch \wedge i \notin XP$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$

Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (formal)

CHALLENGE vs. REVEAL

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$
 $ICH \leftarrow \emptyset$
 $b' \leftarrow \mathcal{A}$
Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
Return sk_j

EXP(i) $\quad \forall i \in [n] \setminus ICH$

$XP \leftarrow XP \cup \{i\}$
Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$

If $ch \wedge i \notin XP$
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
Return k

GEN

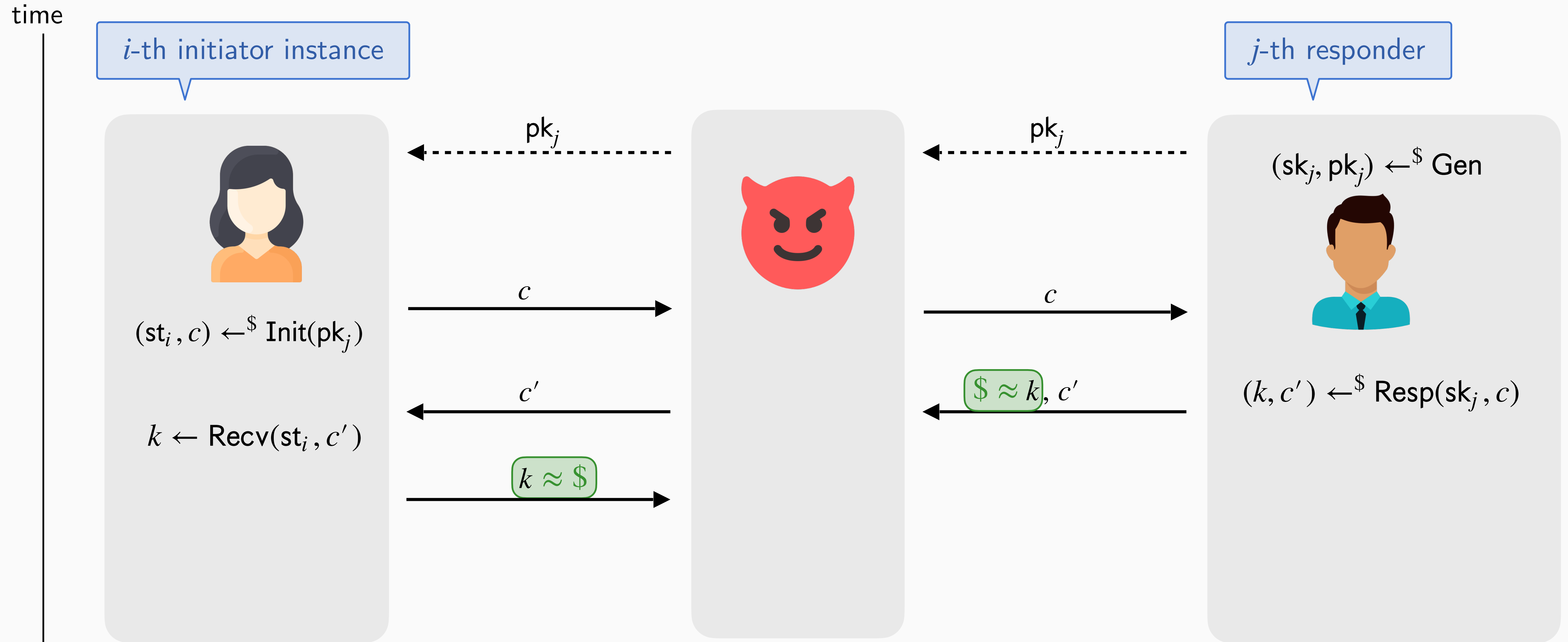
$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If $ch \wedge i \notin XP$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
Return (k, c')

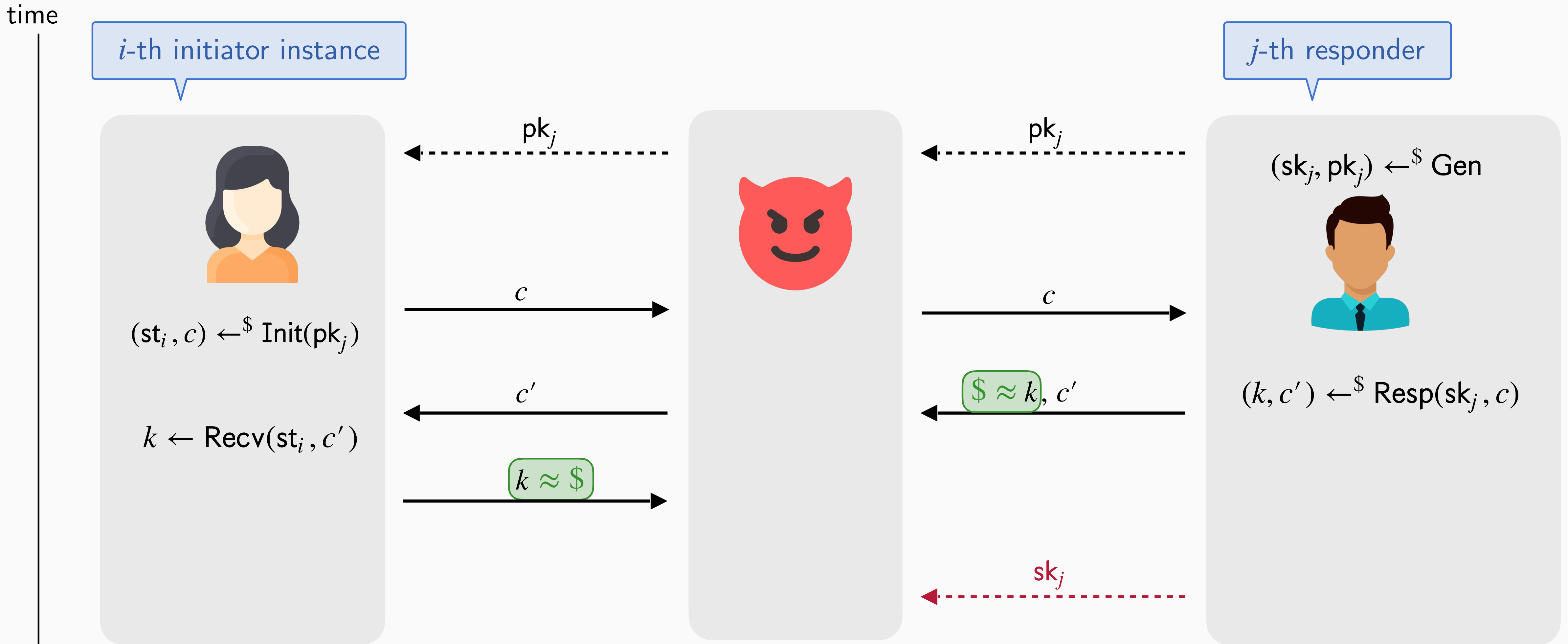
$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (informal)



Key Indistinguishability and trivial attacks: Under which conditions should k look like random?

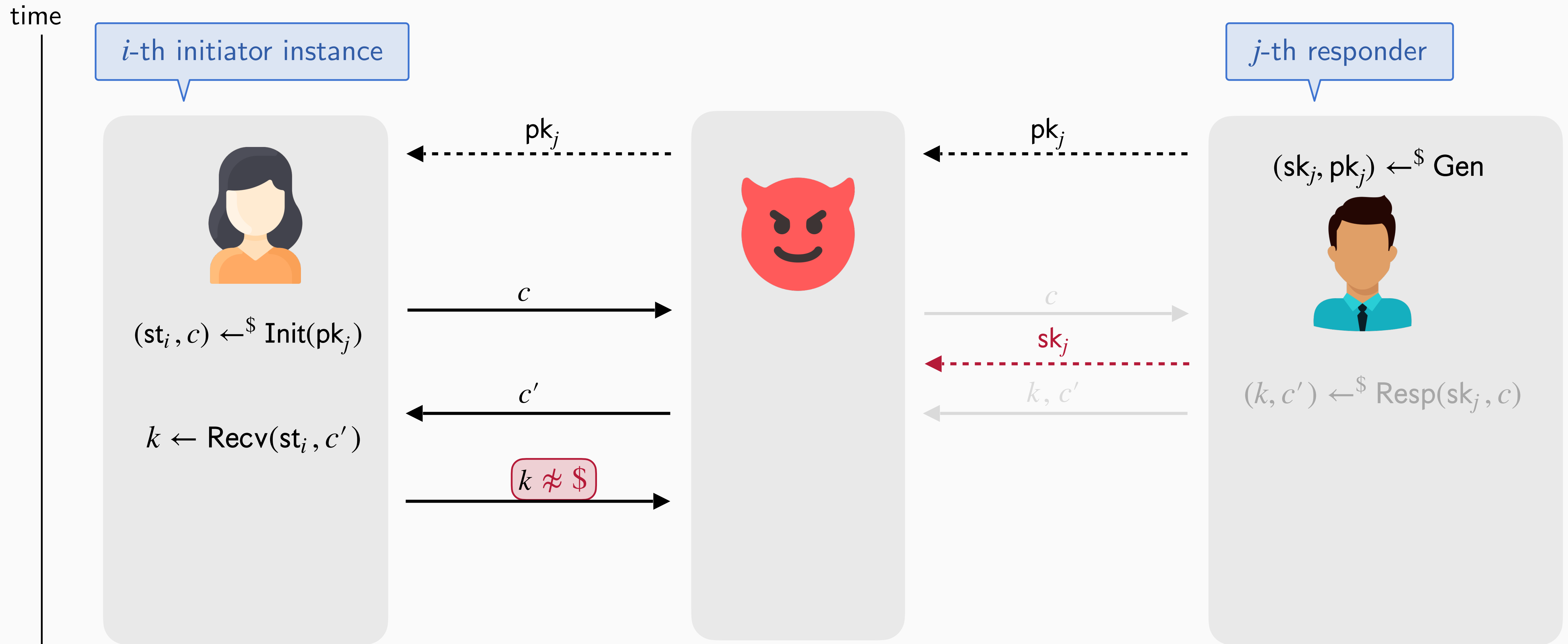
Game-based Security Model for KE (informal)



Key Indistinguishability and trivial attacks: Under which conditions should k look like random?

- Corruption may happen after session is completed (weak forward secrecy).

Game-based Security Model for KE (informal)



Key Indistinguishability and trivial attacks: Under which conditions should k look like random?

- When there is no partnered session, corruption must not happen before session is completed.

Game-based Security Model for KE (formal)

CHALLENGE vs. REVEAL

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$

$b' \leftarrow \mathcal{A}$
 Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
 Return sk_j

EXP(i) $\quad \forall i \in [n] \setminus ICH$

$XP \leftarrow XP \cup \{i\}$
 Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
 If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
 Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
 If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$

If $ch \wedge i \notin XP$
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$

$ICH \leftarrow ICH \cup \{i\}$
 Return k

GEN

$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
 Return pk_m

RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
 If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If $ch \wedge i \notin XP$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
 Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (formal)

CHALLENGE vs. REVEAL

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$

$b' \leftarrow \mathcal{A}$
 Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
 Return sk_j

EXP(i) $\quad \forall i \in [n] \setminus ICH$

$XP \leftarrow XP \cup \{i\}$
 Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
 If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
 Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
 If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$
 If $ch \wedge i \notin XP \wedge P[i] \in [m] \setminus CR$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
 Return k

GEN

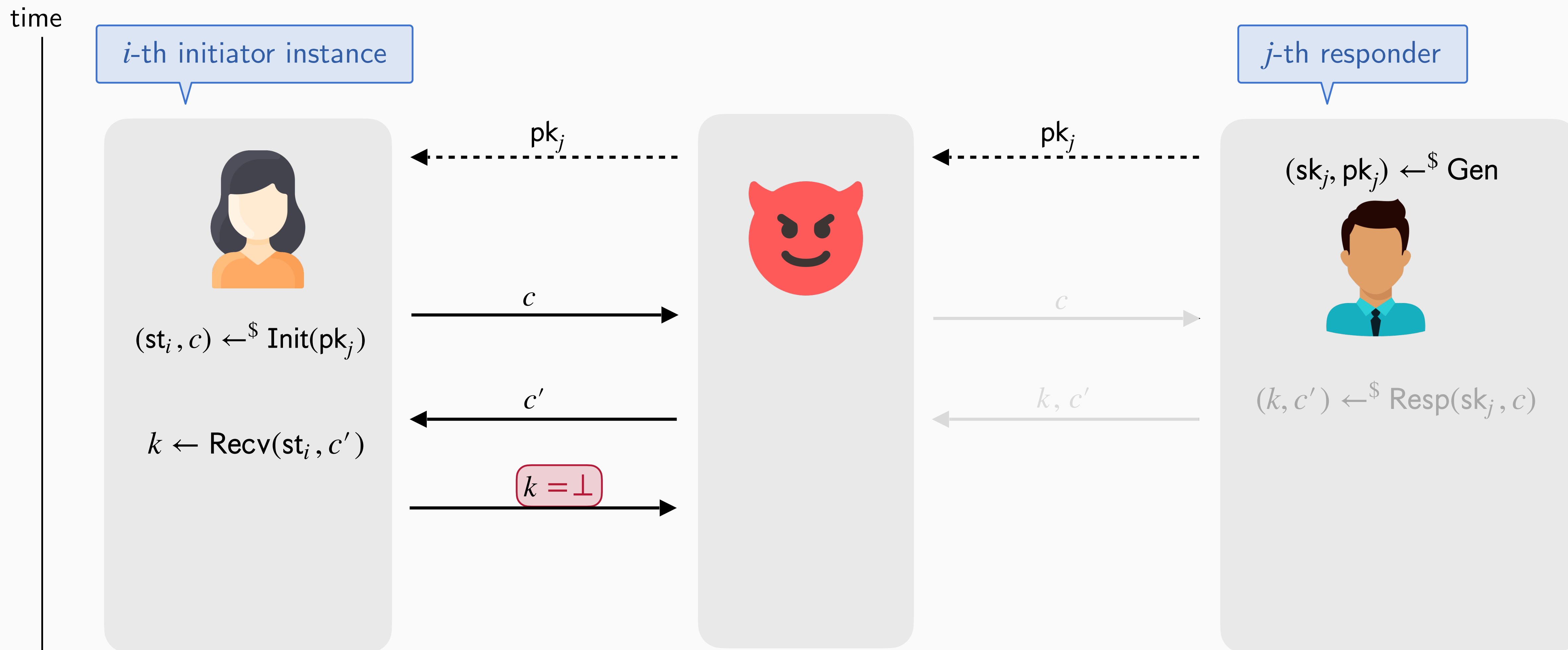
$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
 Return pk_m

RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
 If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If $ch \wedge i \notin XP$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
 Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (informal)



Key Indistinguishability and trivial attacks: Under which conditions should k look like random?

- When there is no partnered session, \mathcal{A} should not authenticate successfully (here: explicit authentication).

Game-based Security Model for KE (formal)

CHALLENGE vs. REVEAL

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$
 $ICH \leftarrow \emptyset$
 $b' \leftarrow \mathcal{A}$
Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
Return sk_j

EXP(i) $\quad \forall i \in [n] \setminus ICH$

$XP \leftarrow XP \cup \{i\}$
Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$

If $ch \wedge i \notin XP \wedge P[i] \in [m] \setminus CR$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
Return k

GEN

$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If $ch \wedge i \notin XP$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (formal)

CHALLENGE vs. REVEAL

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$
 $ICH \leftarrow \emptyset$
 $b' \leftarrow \mathcal{A}$
Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
Return sk_j

EXP(i) $\quad \forall i \in [n] \setminus ICH$

$XP \leftarrow XP \cup \{i\}$
Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$
If $k = \perp$: Return \perp
If $ch \wedge i \notin XP \wedge P[i] \in [m] \setminus CR$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
Return k

GEN

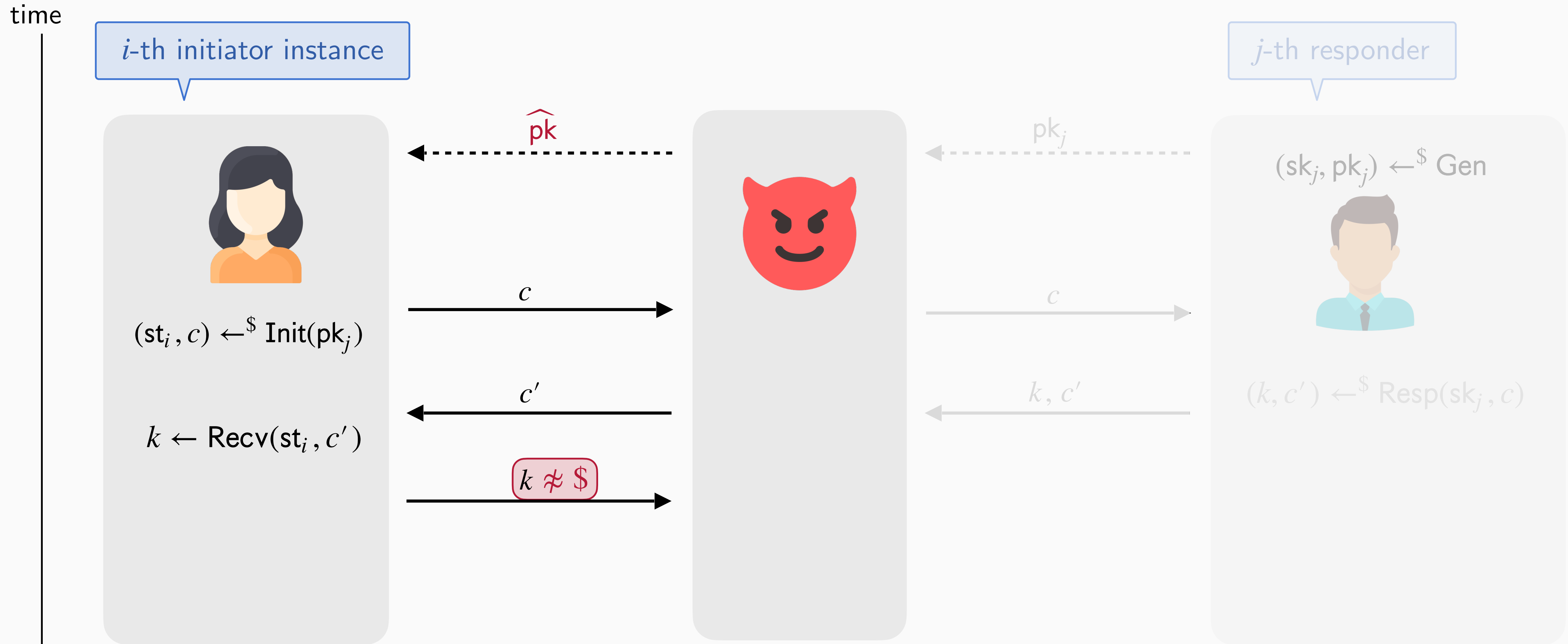
$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
Return pk_m

RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If $ch \wedge i \notin XP$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (informal)



Key Indistinguishability and trivial attacks: Under which conditions should k look like random?

- \mathcal{A} can create (dishonest) responders and reveal the initiator's session key.

Game-based Security Model for KE (formal)

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$
 $ICH \leftarrow \emptyset$
 $b' \leftarrow \mathcal{A}$
 Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
 Return sk_j

EXP(i) $\quad \forall i \in [n] \setminus ICH$

$XP \leftarrow XP \cup \{i\}$
 Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
 If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
 Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
 If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$
 If $k = \perp$: Return \perp
 If $ch \wedge i \notin XP \wedge P[i] \in [m] \setminus CR$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
 Return k

GEN

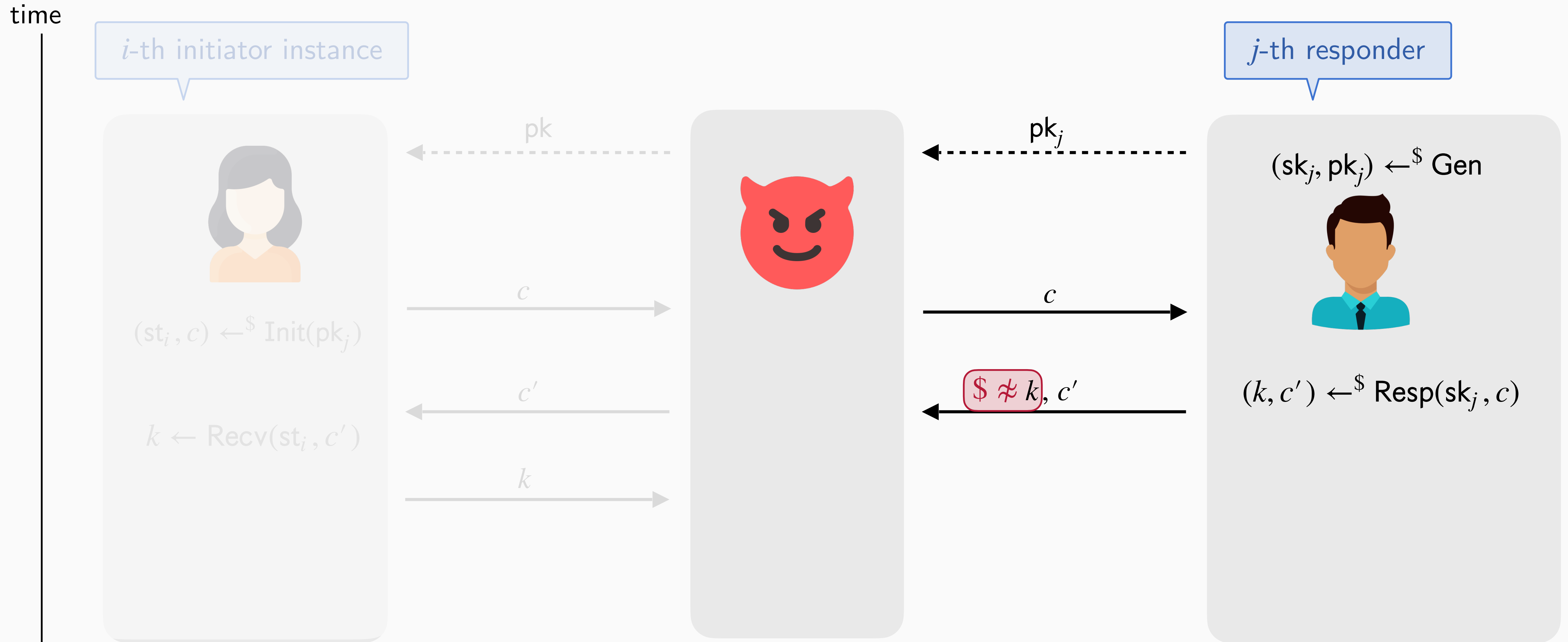
$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
 Return pk_m

RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
 If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If $ch \wedge i \notin XP$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
 Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (informal)



Key Indistinguishability and trivial attacks: Under which conditions should k look like random?

- \mathcal{A} can create (dishonest) initiators and reveal the responder's session key.

Game-based Security Model for KE (formal)

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$
 $ICH \leftarrow \emptyset$
 $b' \leftarrow \mathcal{A}$
 Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
 Return sk_j

EXP(i) $\quad \forall i \in [n] \setminus ICH$

$XP \leftarrow XP \cup \{i\}$
 Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
 If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
 Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
 If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$
 If $k = \perp$: Return \perp
 If $ch \wedge i \notin XP \wedge P[i] \in [m] \setminus CR$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
 Return k

GEN

$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
 Return pk_m

RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
 If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If $ch \wedge i \notin XP$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
 Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

Game-based Security Model for KE (formal)

Game $\text{IND}_{\text{KE}}^b(\mathcal{A})$

$n, m \leftarrow 0$
 $Q \leftarrow \emptyset$
 $P[\cdot], I[\cdot] \leftarrow \perp$
 $R[\cdot, \cdot] \leftarrow \emptyset$
 $CR, XP \leftarrow \emptyset$
 $ICH \leftarrow \emptyset$
 $b' \leftarrow \mathcal{A}$
 Return b'

CORR(j) $\quad \forall j \in [m]$

$CR \leftarrow CR \cup \{j\}$
 Return sk_j

EXP(i) $\quad \forall i \in [n] \setminus ICH$

$XP \leftarrow XP \cup \{i\}$
 Return st_i

INIT(pk)

$n ++$
 $(\text{st}_n, c) \leftarrow^{\$} \text{Init}(\text{pk})$
 If $\exists j \in [m] : \text{pk} = \text{pk}_j$:
 $P[n] \leftarrow j; I[n] \leftarrow c$
 Return c

RECEIVE(i, c', ch) $\quad \forall i \in [n] \setminus Q$

$Q \leftarrow Q \cup \{i\}$
 If $c' \in R[P[i], i]$: Return
 $k \leftarrow \text{Recv}(\text{st}_i, c')$
 If $k = \perp$: Return \perp
 If $ch \wedge i \notin XP \wedge P[i] \in [m] \setminus CR$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
 Return k

GEN

$m ++$
 $(\text{sk}_m, \text{pk}_m) \leftarrow^{\$} \text{Gen}$
 Return pk_m

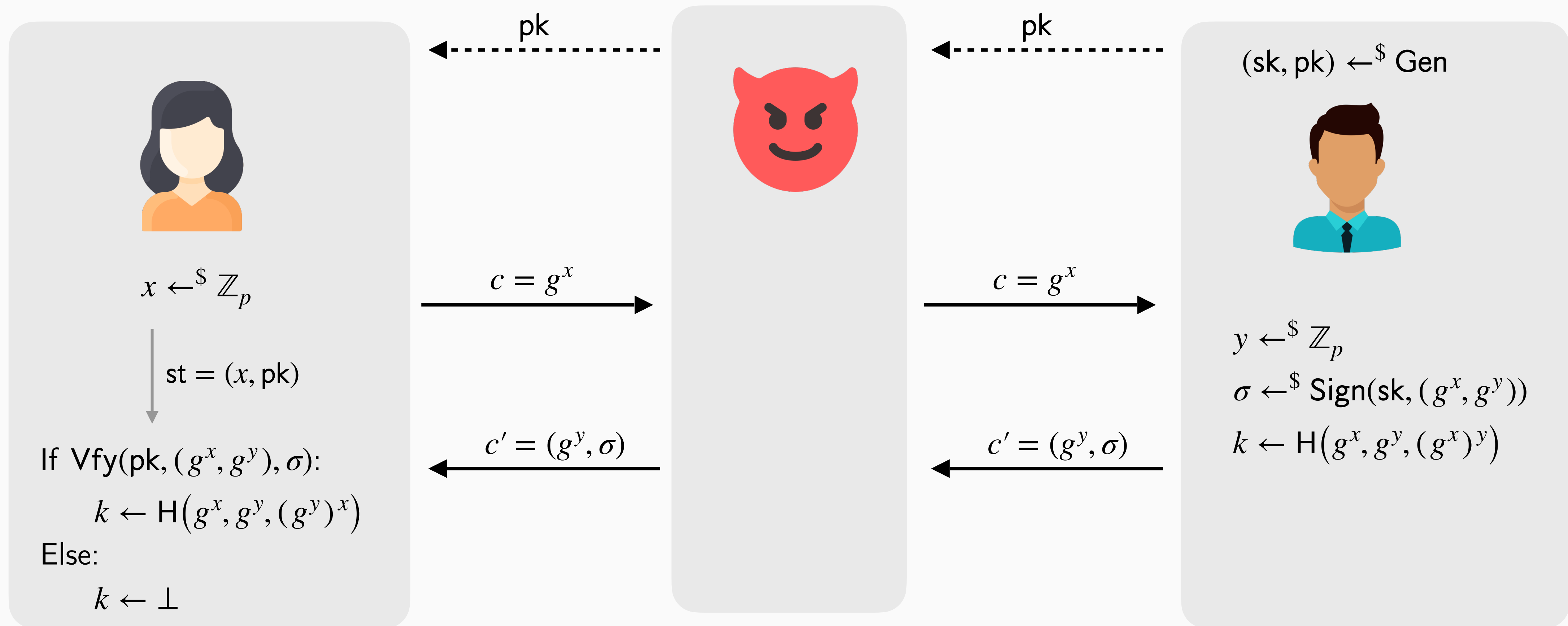
RESPOND(j, c, ch) $\quad \forall j \in [m]$

$(k, c') \leftarrow^{\$} \text{Resp}(\text{sk}_j, c)$
 If $\exists i \in [n] : P[i] = j \wedge I[i] = c$:
 $R[j, i] \leftarrow R[j, i] \cup \{c'\}$
 If $ch \wedge i \notin XP$:
 If $b = 1$: $k \leftarrow^{\$} \mathcal{K}$
 $ICH \leftarrow ICH \cup \{i\}$
 Return (k, c')

$$\text{Adv}_{\text{KE}}^{\text{ind}}(\mathcal{A}) := \left| \Pr[\text{IND}_{\text{KE}}^0(\mathcal{A}) = 1] - \Pr[\text{IND}_{\text{KE}}^1(\mathcal{A}) = 1] \right|$$

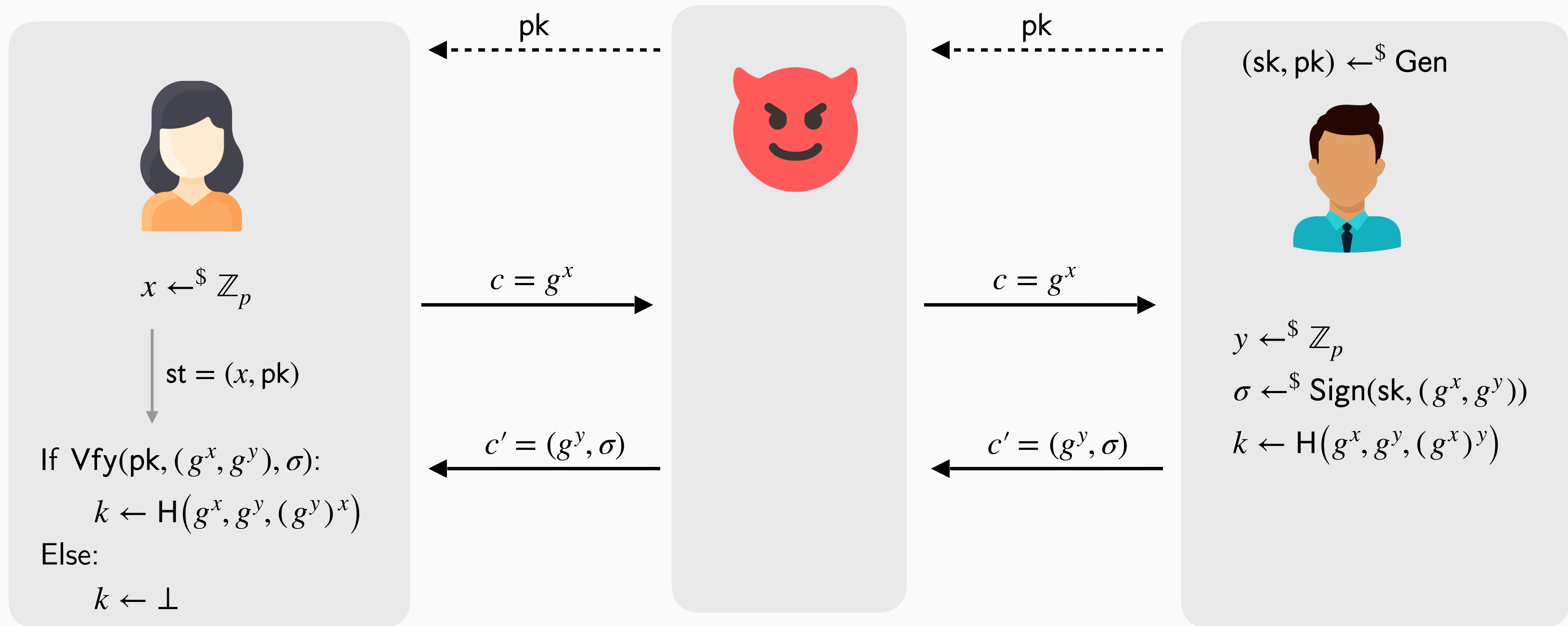
Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:



Signed Diffie-Hellman

Given a signature scheme $SIG = (\text{Gen}, \text{Sign}, \text{Vfy})$, a prime-order group (\mathbb{G}, p, g) and a hash function $H : \mathbb{G}^3 \rightarrow \mathcal{K}$, we define the following protocol:



Security: relies on strong unforgeability (suf-cma) of SIG and the strong Diffie-Hellman (st-cdh) problem for (\mathbb{G}, p, g)

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$

Proof (Sketch):

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$

Proof (Sketch):

- G_0^b is the original game with bit b

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p} \left| \Pr[G_0^b] - \Pr[G_1^b] \right|$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p} \left| \Pr[G_0^b] - \Pr[G_1^b] \right|$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair
- G_2^b aborts if **RECEIVE** gets as input a valid signature, but intended peer was not corrupted

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq \frac{2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1)}{\left| \Pr[G_1^b] - \Pr[G_2^b] \right|} + q_{\text{init}} \cdot \frac{\text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2)}{\left| \Pr[G_0^b] - \Pr[G_1^b] \right|} + 2 \cdot q_{\text{init}} q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair
- G_2^b aborts if **RECEIVE** gets as input a valid signature, but intended peer was not corrupted

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$
$$\left| \Pr[G_1^b] - \Pr[G_2^b] \right| \qquad \left| \Pr[G_0^b] - \Pr[G_1^b] \right|$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair
- G_2^b aborts if **RECEIVE** gets as input a valid signature, but intended peer was not corrupted
- G_3^b aborts if \mathcal{A} “predicts” CDH challenge (proof artifact)

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$
$$\left| \Pr[G_1^b] - \Pr[G_2^b] \right| \qquad \left| \Pr[G_0^b] - \Pr[G_1^b] \right| \qquad \left| \Pr[G_2^b] - \Pr[G_3^b] \right|$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair
- G_2^b aborts if **RECEIVE** gets as input a valid signature, but intended peer was not corrupted
- G_3^b aborts if \mathcal{A} “predicts” CDH challenge (proof artifact)

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{Sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$
$$\left| \Pr[G_1^b] - \Pr[G_2^b] \right| \quad \left| \Pr[G_3^0] - \Pr[G_3^1] \right| \quad \left| \Pr[G_0^b] - \Pr[G_1^b] \right| \quad \left| \Pr[G_2^b] - \Pr[G_3^b] \right|$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair
- G_2^b aborts if **RECEIVE** gets as input a valid signature, but intended peer was not corrupted
- G_3^b aborts if \mathcal{A} “predicts” CDH challenge (proof artifact)

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$
$$\left| \Pr[G_1^b] - \Pr[G_2^b] \right| \quad \left| \Pr[G_3^0] - \Pr[G_3^1] \right| \quad \left| \Pr[G_0^b] - \Pr[G_1^b] \right| \quad \left| \Pr[G_2^b] - \Pr[G_3^b] \right|$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair
- G_2^b aborts if **RECEIVE** gets as input a valid signature, but intended peer was not corrupted
- G_3^b aborts if \mathcal{A} “predicts” CDH challenge (proof artifact)

More details: <https://github.com/proof-ladders/protocol-ladder/blob/main/Notes/computational/main.pdf>

- Full model and tight proof from multi-user assumptions

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}}q_{\text{gen}} \cdot 2^{-\gamma_{\text{sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$
$$\left| \Pr[G_1^b] - \Pr[G_2^b] \right| \quad \left| \Pr[G_3^0] - \Pr[G_3^1] \right| \quad \left| \Pr[G_0^b] - \Pr[G_1^b] \right| \quad \left| \Pr[G_2^b] - \Pr[G_3^b] \right|$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair
- G_2^b aborts if **RECEIVE** gets as input a valid signature, but intended peer was not corrupted
- G_3^b aborts if \mathcal{A} “predicts” CDH challenge (proof artifact)

More details: <https://github.com/proof-ladders/protocol-ladder/blob/main/Notes/computational/main.pdf>

- Full model and tight proof from multi-user assumptions

Next up: Vincent (ProVerif) and Cas (Tamarin)

Security of Signed Diffie-Hellman

Theorem:

Let \mathcal{A} be an adversary against the Signed-DH protocol, where H is modeled as a random oracle.

Then we construct adversaries \mathcal{B}_1 against SIG and \mathcal{B}_2 against (\mathbb{G}, p, g) such that:

$$\text{Adv}_{\text{Signed-DH}}^{\text{ind}}(\mathcal{A}) \leq 2q_{\text{gen}} \cdot \text{Adv}_{\text{SIG}}^{\text{suf-cma}}(\mathcal{B}_1) + q_{\text{init}} \cdot \text{Adv}_{\mathbb{G}, p, g}^{\text{st-cdh}}(\mathcal{B}_2) + 2 \cdot q_{\text{init}} q_{\text{gen}} \cdot 2^{-\gamma_{\text{sig}}} + \frac{2q_H(q_{\text{init}} + q_{\text{resp}})}{p}$$
$$\left| \Pr[G_1^b] - \Pr[G_2^b] \right| \quad \left| \Pr[G_3^0] - \Pr[G_3^1] \right| \quad \left| \Pr[G_0^b] - \Pr[G_1^b] \right| \quad \left| \Pr[G_2^b] - \Pr[G_3^b] \right|$$

Proof (Sketch):

- G_0^b is the original game with bit b
- G_1^b aborts if \mathcal{A} “predicts” signing key pair
- G_2^b aborts if **RECEIVE** gets as input a valid signature, but intended peer was not corrupted
- G_3^b aborts if \mathcal{A} “predicts” CDH challenge (proof artifact)

More details: <https://github.com/proof-ladders/protocol-ladder/blob/main/Notes/computational/main.pdf>

- Full model and tight proof from multi-user assumptions

Next up: Vincent (ProVerif) and Cas (Tamarin)

Thank you!