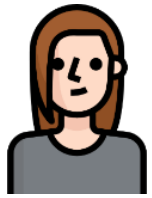
Statement-Oblivious Threshold Witness Encryption

Sebastian Faust[1], Carmit Hazay[2], **David Kretzler**[1], Benjamin Schlosser[1]

1. Technische Universität Darmstadt, Germany
2. Bar-Ilan University, Israel

Witness Encryption



NP-language $\mathcal{L}$
with relation $\mathcal{R}$

$m \quad x \in \mathcal{L}$

W-Enc

c

c

w

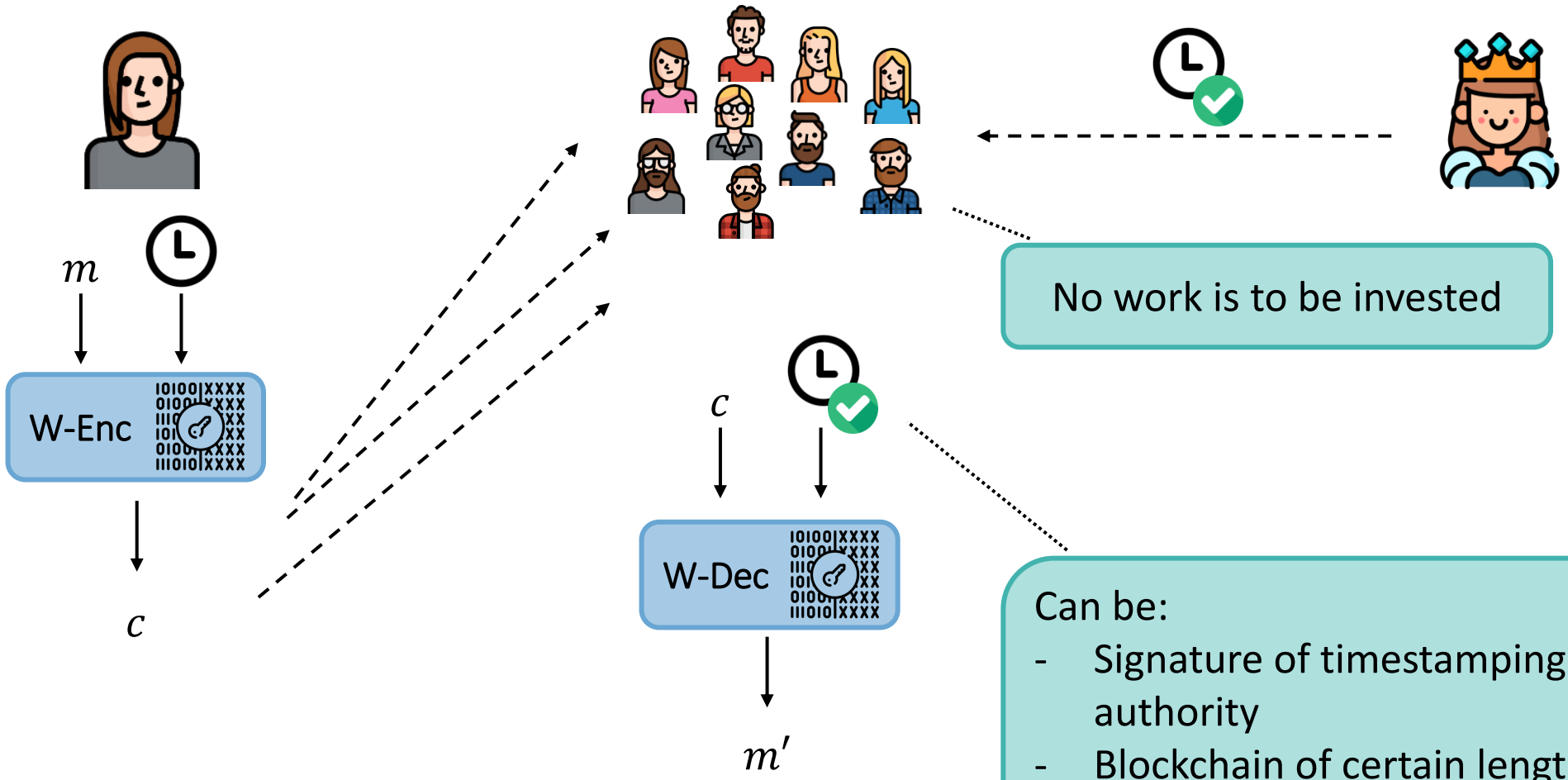
$c \quad w$

W-Dec

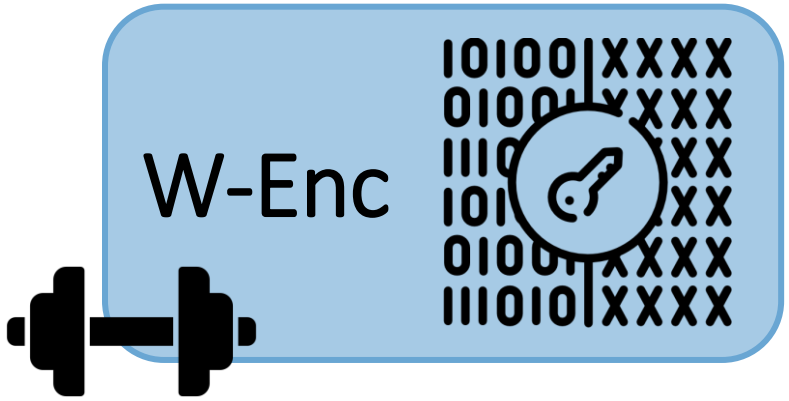
m'

$m' = m$ iff $(x, w) \in \mathcal{R}$

Example: Time-lock Encryption



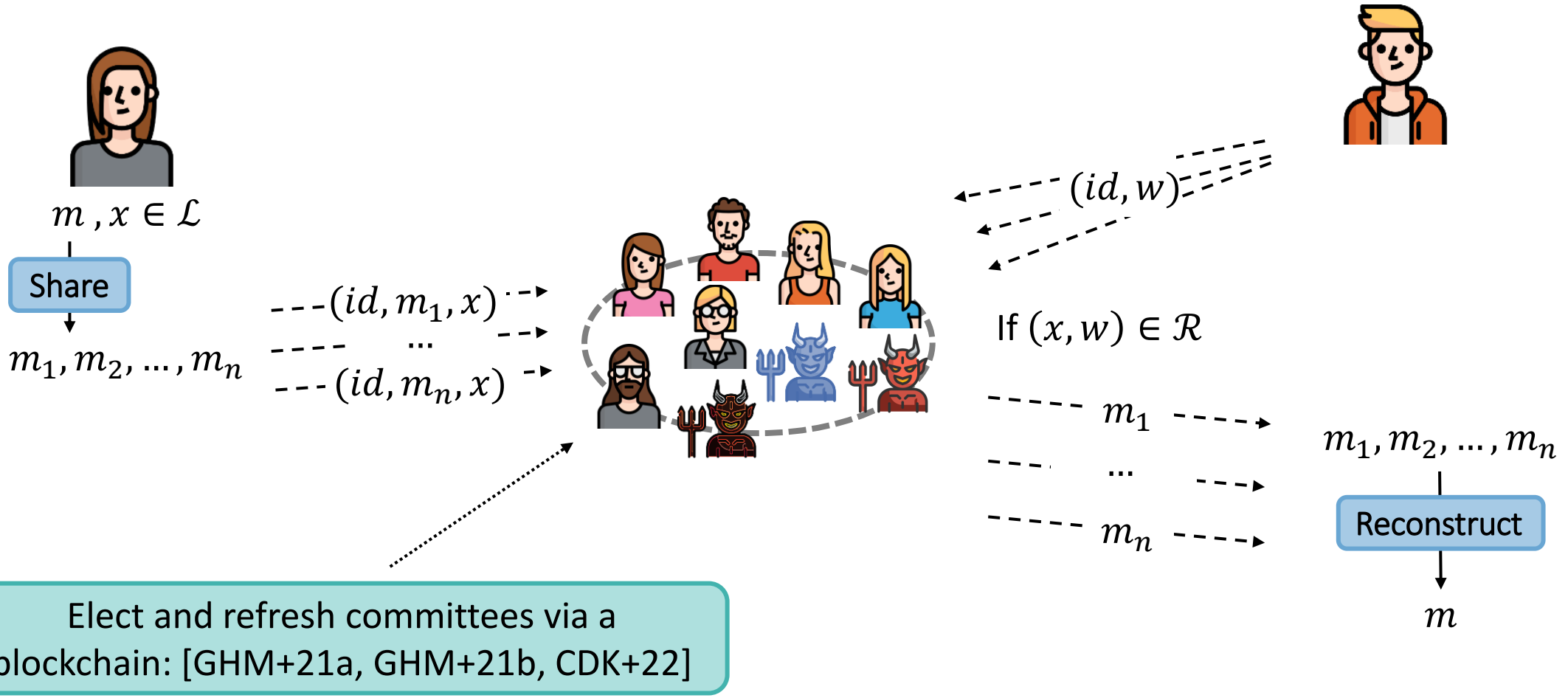
Problem 1: Strong Assumptions



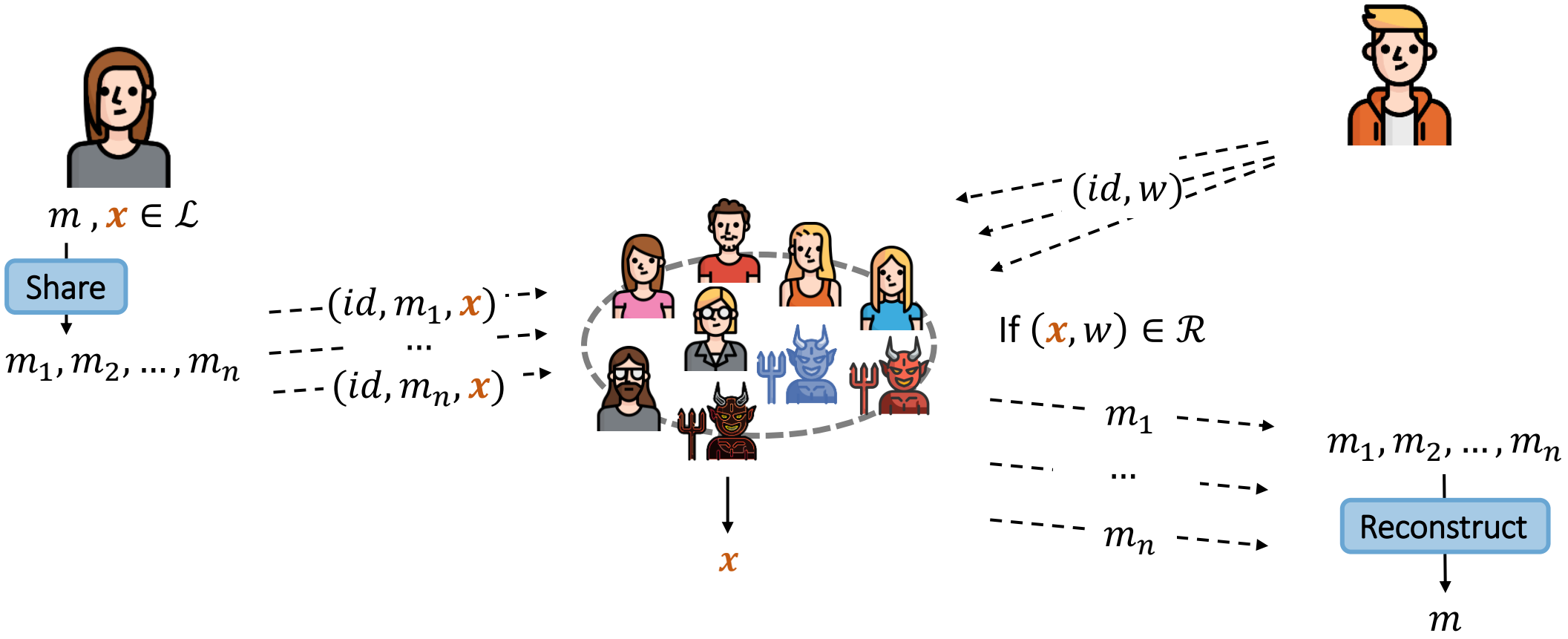
- Multilinear maps
- Indistinguishability obfuscation
- Cryptographic invariant maps

Trust in strong assumptions → Trust in honesty of **some** parties within a committee

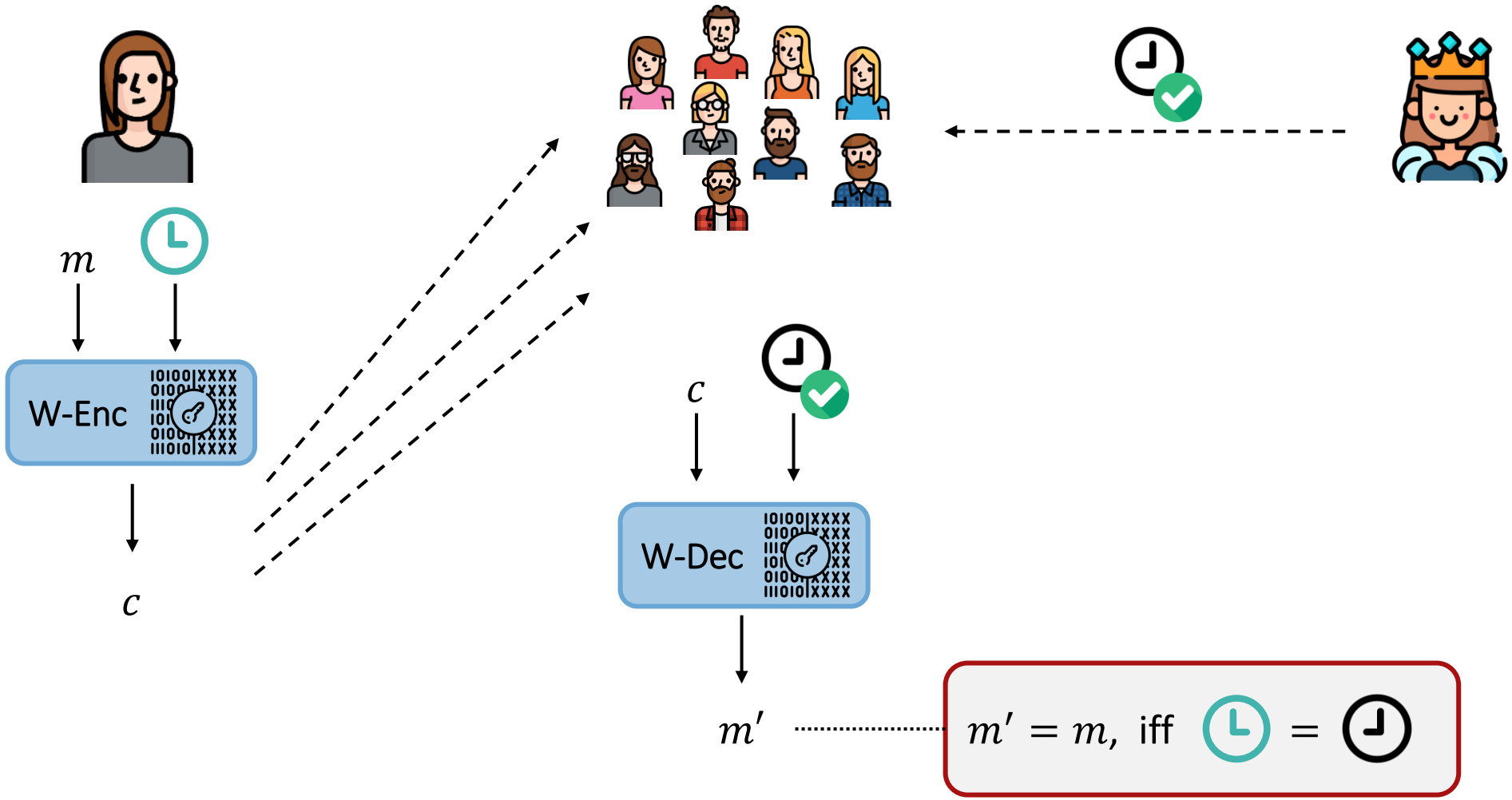
Solution: Committees



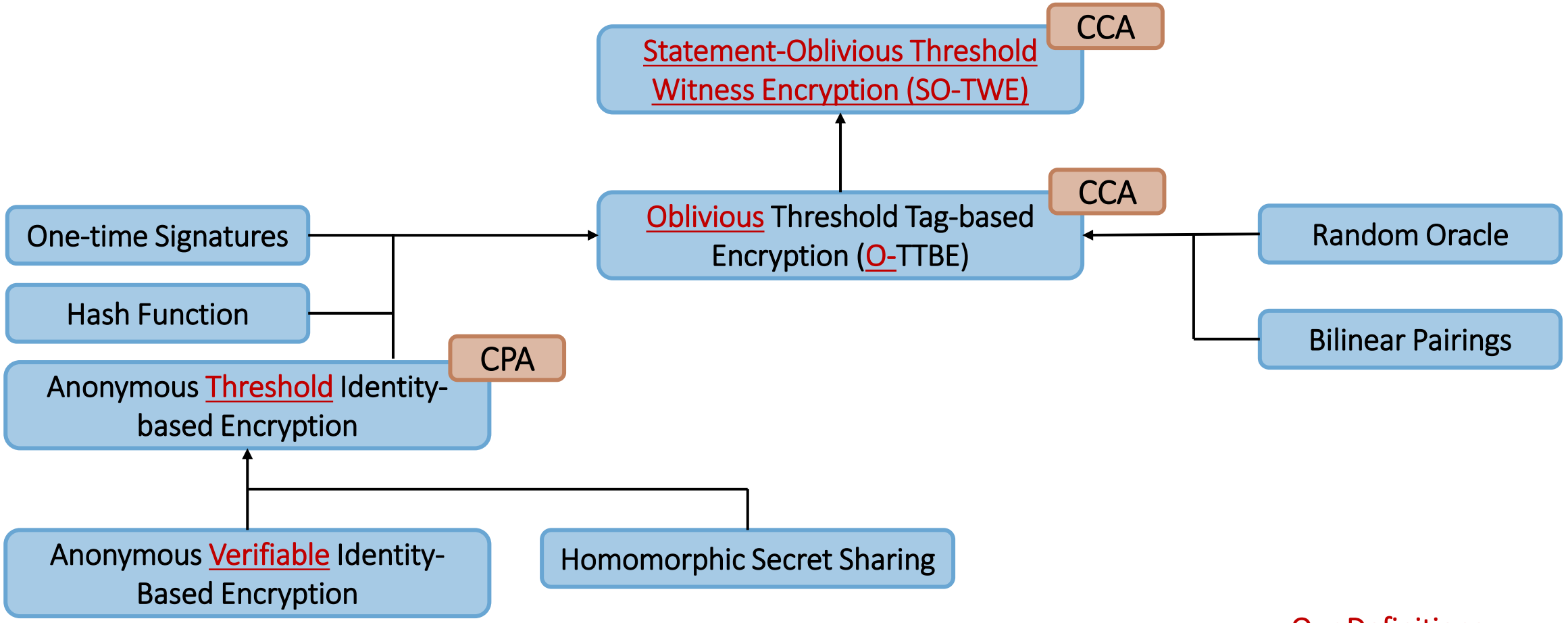
Problem 2: Statement is Public



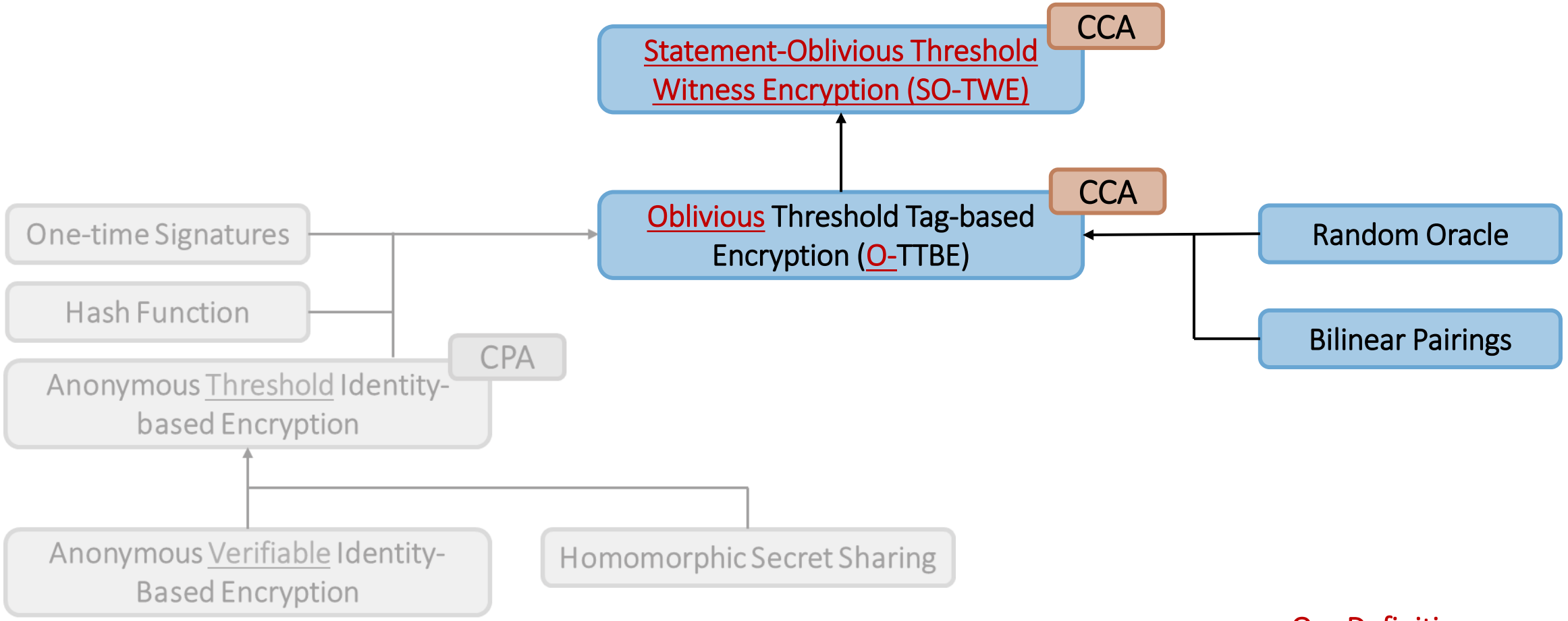
Example: Hidden Release Time



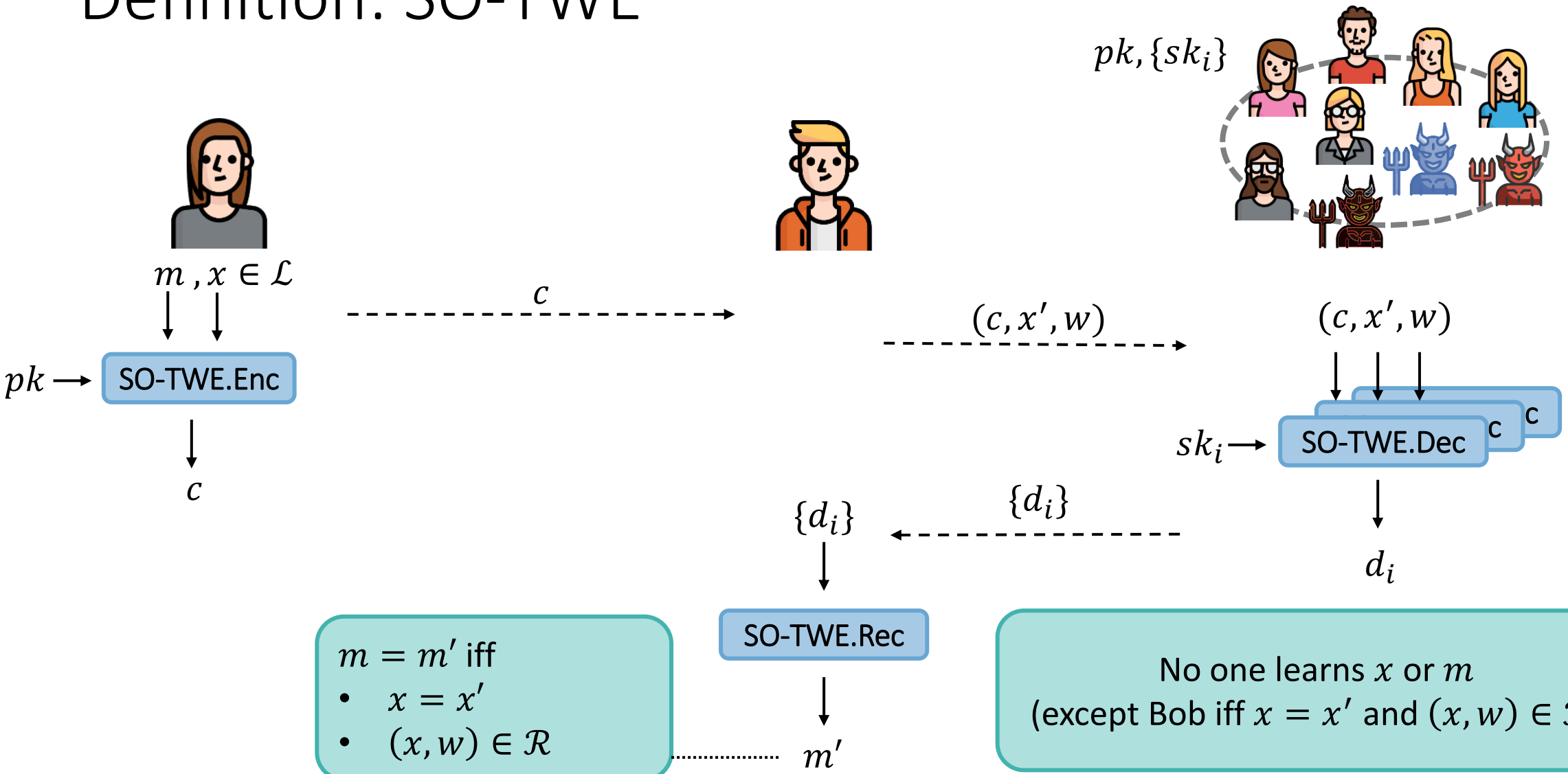
Contribution



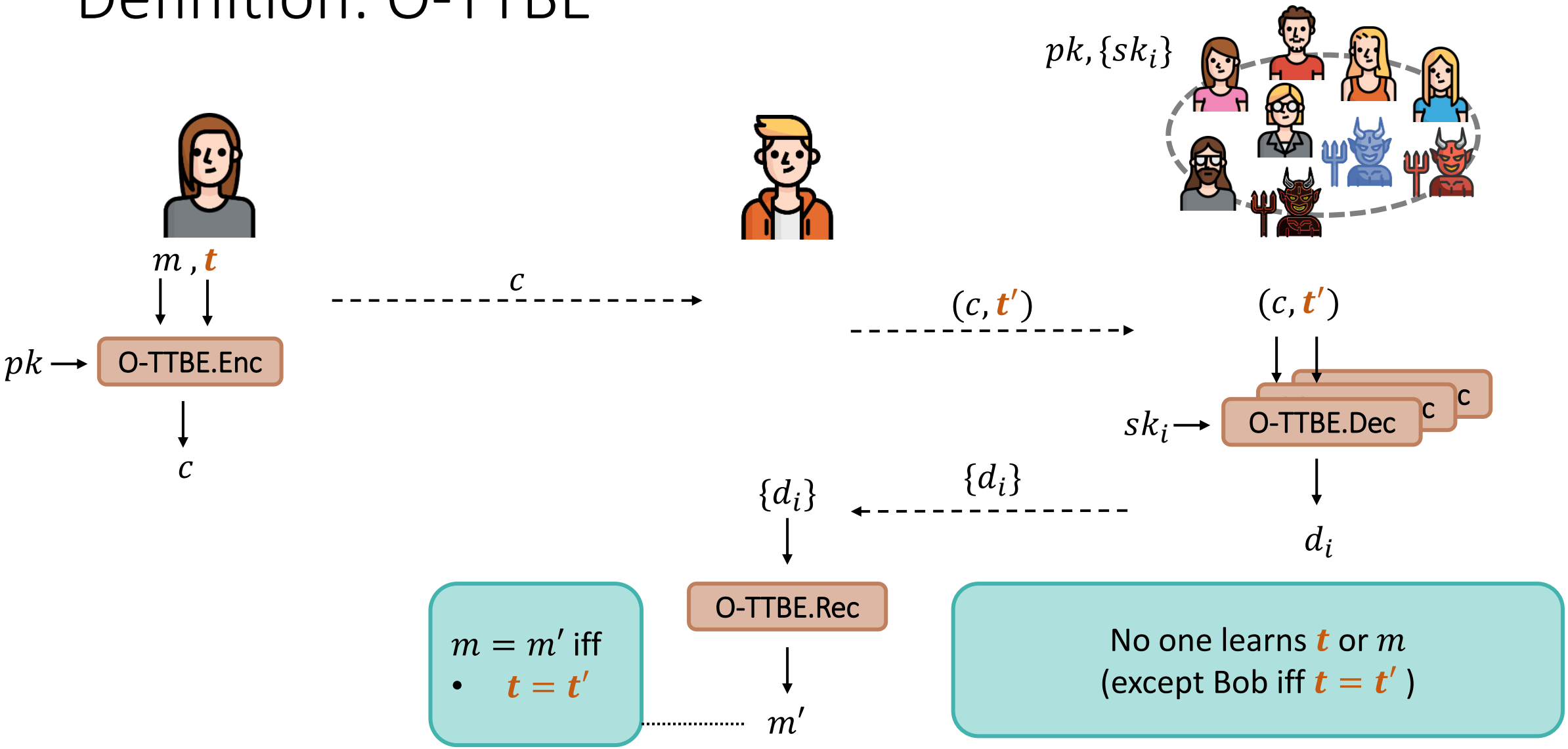
Contribution



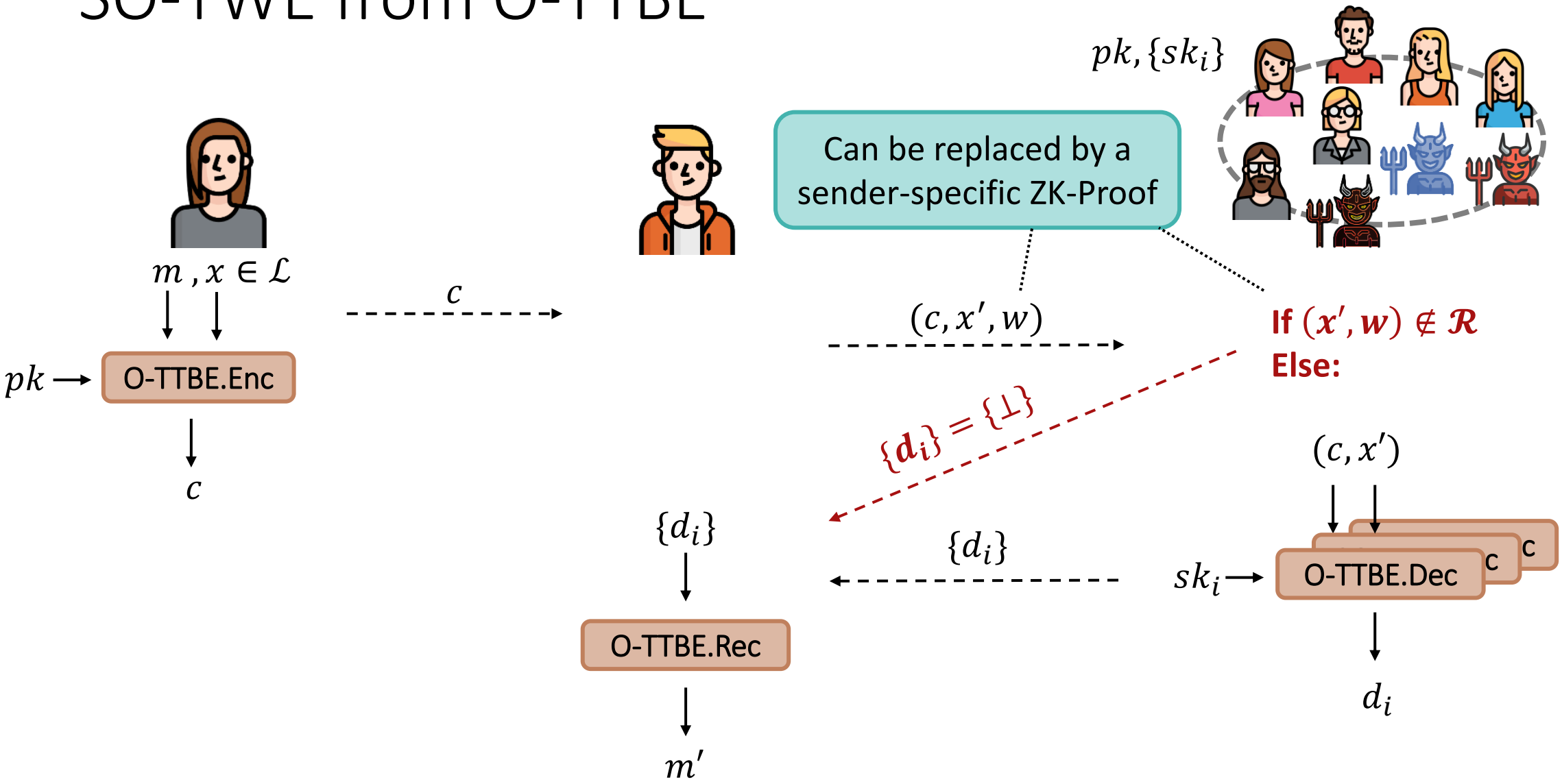
Definition: SO-TWE



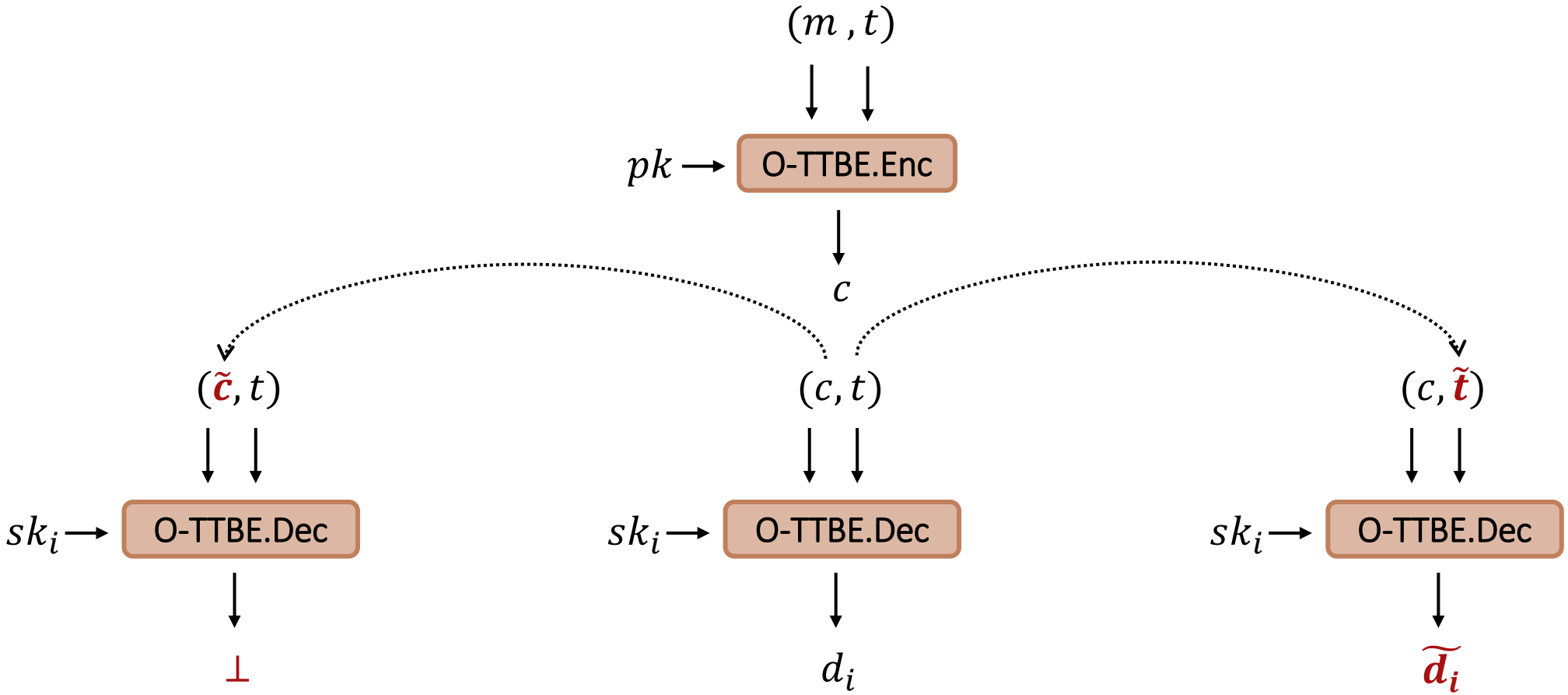
Definition: O-TTBE



SO-TWE from O-TTBE



Instantiating O-TTBE: The Challenge



No info about d_i (CCA)

Valid decryption share

Looks correct (obliviousness)
Combines to „garbage“

Bilinear Pairings

- Three groups $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ of prime order q
- Map e: $\mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T : e(g^a, h^b) = e(g, h)^{ab}$ for all g, h, a, b

→ Here: $\mathbb{G}_1 = \mathbb{G}_2 = \mathbb{G}$ (Type-1 Pairing)

Type-2 would work as well

The O-TTBE Construction (Simplified)

- $Setup(1^{\kappa})$
  - Public parameters:
    - Groups $\mathbb{G}, \mathbb{G}_T$ of order q
    - Mapping: $e: \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$
    - Generator: $g \leftarrow_R \mathbb{G}$
    - Random oracles: $\mathcal{H}_1, \mathcal{H}_2$
  - Secret key: $x \leftarrow_R \mathbb{Z}_q$
  - Secret key shares: $\{x_i \in_R \mathbb{Z}_q\}$ such that $\sum x_i = x$
  - Public keys: $(X = g^x)$

Can also be Shamir-shared

The O-TTBE Construction (Simplified)

Key shares: $\{x_i\}$
Public key: $X = g^x$

- $Encrypt(pk, t, m)$:

$y \in_R \mathbb{Z}_q$ — Encryption randomness

$Z = X^y$ — Encryption mask

$\mathrm{Y} = g^y$ — Decryption hint

$=$ Standard ElGamal

$T = \mathcal{H}_1(t, Y)$ — (Tag+cipher)-unique mask

$M = e(T, Z)$ — Combined mask

$c = \mathcal{H}_2(M) \oplus m$ — Masked message

π: Zero-knowledge proof of knowledge of y

Prevent mallability attacks

Standard + leightweight

➔ Output (Y, c, π)

The O-TTBE Construction (Simplified)

Key shares: $\{x_i\}$
Public key: $X = g^x$

- $Decrypt(x_i, C = (Y, c, \pi), t')$:

Check π

Prevent mallability attacks

Correct share if $t' = t$
Random offset if $t' \neq t$

$T' = \mathcal{H}_1(t', Y)$

(Tag+cipher)-unique mask

$D_i = e(T', Y^{x_i})$

Share of combined mask

π_i: Proof of correct decryption

For selection of correct shares

Standard + leightweight

➔ Output (D_i, π_i)

The O-TTBE Construction (Simplified)

Key shares: $\{x_i\}$
Public key: $X = g^x$

- $Combine(C = (Y, c, \pi), \{D_i\})$:

$D = \prod D_i$

Combine mask

Check $m = c \oplus \mathcal{H}_2(D)$

Unmask ciphertext

Interpolation for Shamir-shared keys: $D = \prod D_i^{\lambda_i^S}$

The O-TTBE Construction (Simplified)

Key shares: $\{x_i\}$
Public key: $X = g^x$

- $Combine(C = (Y, c, \pi), \{D_i\})$:

$D = \prod D_i$

Combine mask

Check $m = c \oplus \mathcal{H}_2(D)$

Unmask ciphertext

➔ Correctness:

Encryption: $m = c \oplus \mathcal{H}_2\big(e(\mathcal{H}_1(t, g^y), g^{xy})\big)$

Equal if $\mathbf{t} = \mathbf{t}'$

Decryption: $m = c \oplus \mathcal{H}_2(\prod D_i) = c \oplus \mathcal{H}_2(\prod e(T', Y^{x_i})) = c \oplus \mathcal{H}_2\big(e(\mathcal{H}_1(t', g^y), g^{xy})\big)$

CCA-Challenge: Answering Decryption Queries

$(g, h, g^x, g^y, e(h, g^z))$

? $z = xy$ **?**

$(pk = g^x)$

Decryption query: (Y, c, π)

Decrypt without knowing $\mathrm{sk} = x$?

CCA-Challenge: Answering Decryption Queries

Instantiates $T' = \mathcal{H}_1(\cdot)$

- Don't sample $T' \leftarrow_r \mathbb{G}$
- Sample $k \leftarrow_r \mathbb{Z}_q$
  Return $T' = g^k$

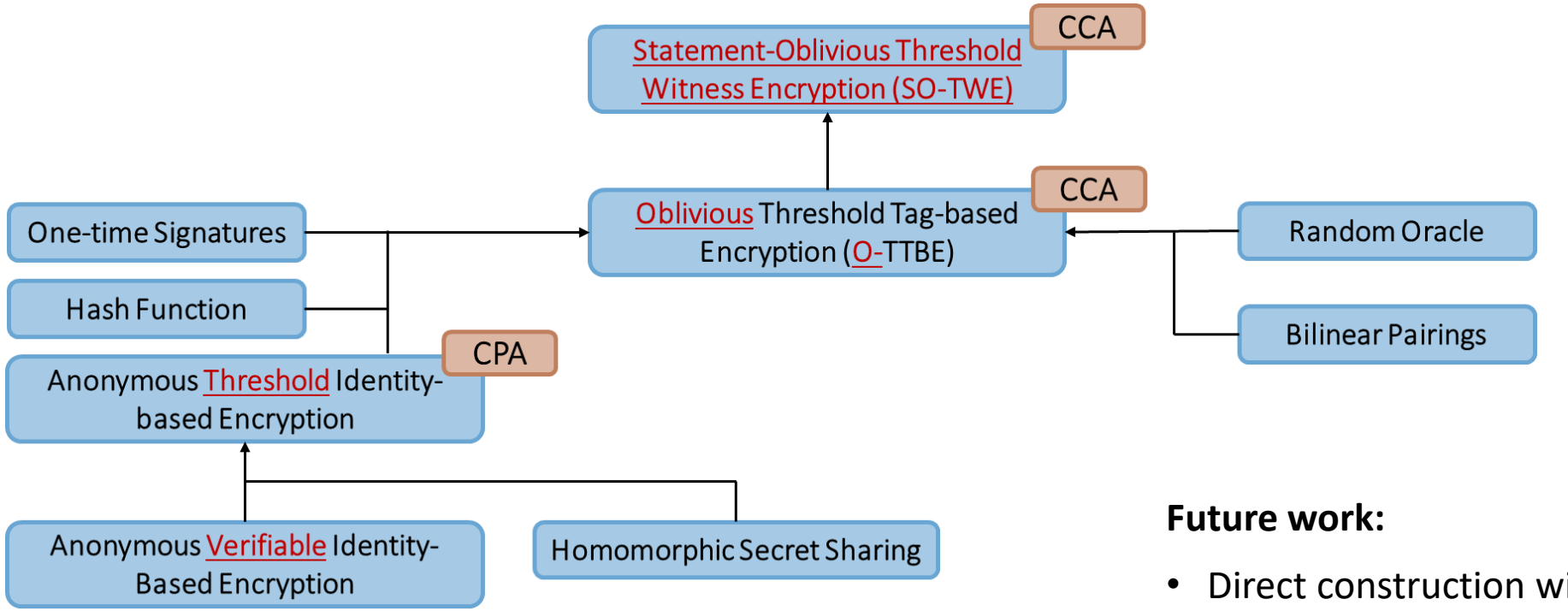
$Y = g^y \overset{?}{\rightarrow} D = e(T', g^{xy}) = e(g^k, g^{xy})$

Let $k = \log_g T'$

$D = e(X, Y^k) = e(g^x, g^{yk})$

pk

Conclusion



Our Definitions

Future work:

- Direct construction without random oracles
- Anonymous Threshold Identity-based Encryption
- Pro-active security

Any questions?

https://eprint.iacr.org/2023/668.pdf

David Kretzler: david.kretzler@tu-darmstadt.de
Benjamin Schlosser: benjamin.schlosser@tu-darmstadt.de