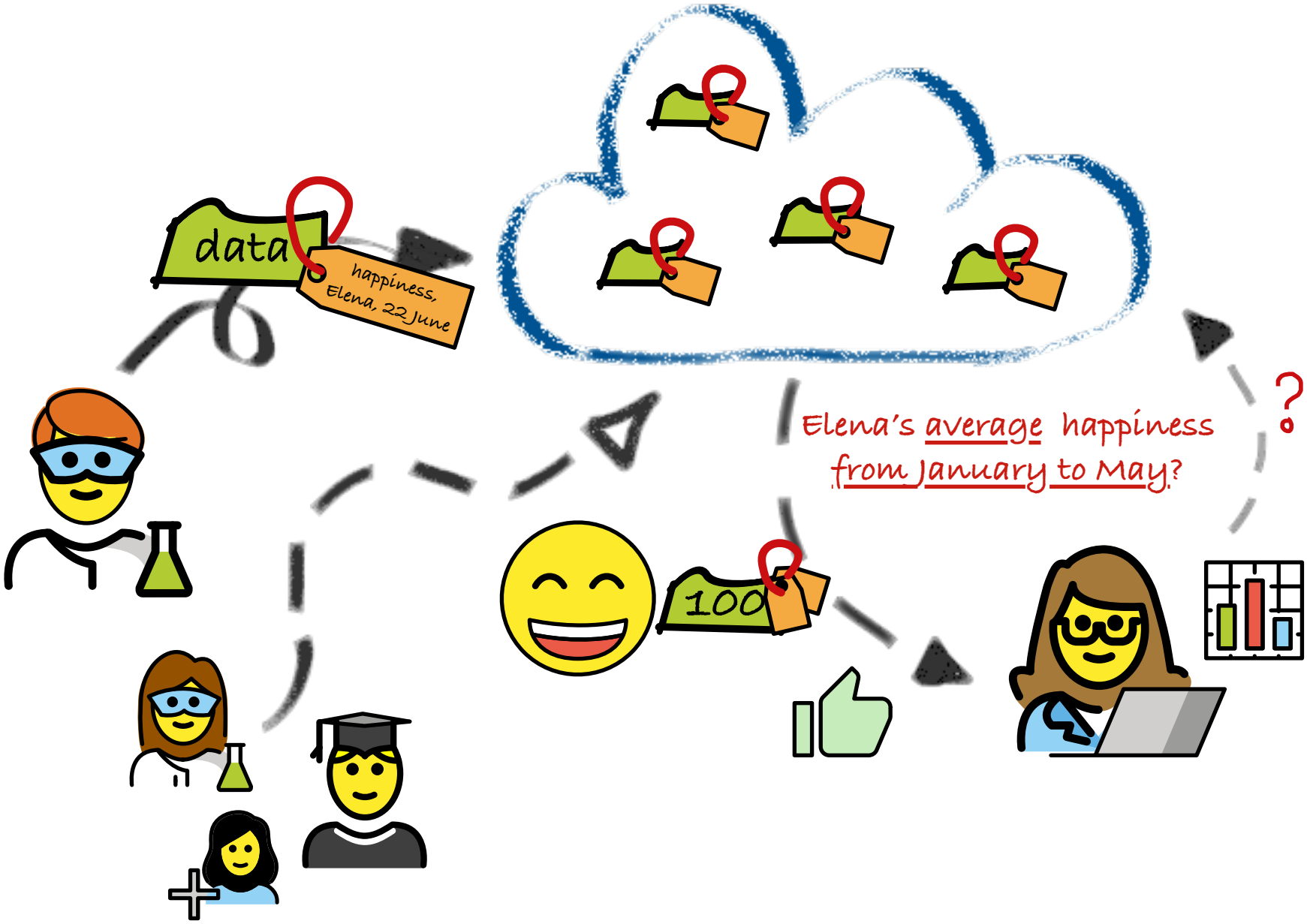
The Simplest Multi-key Linearly Homomorphic Signature Scheme

Diego Aranha

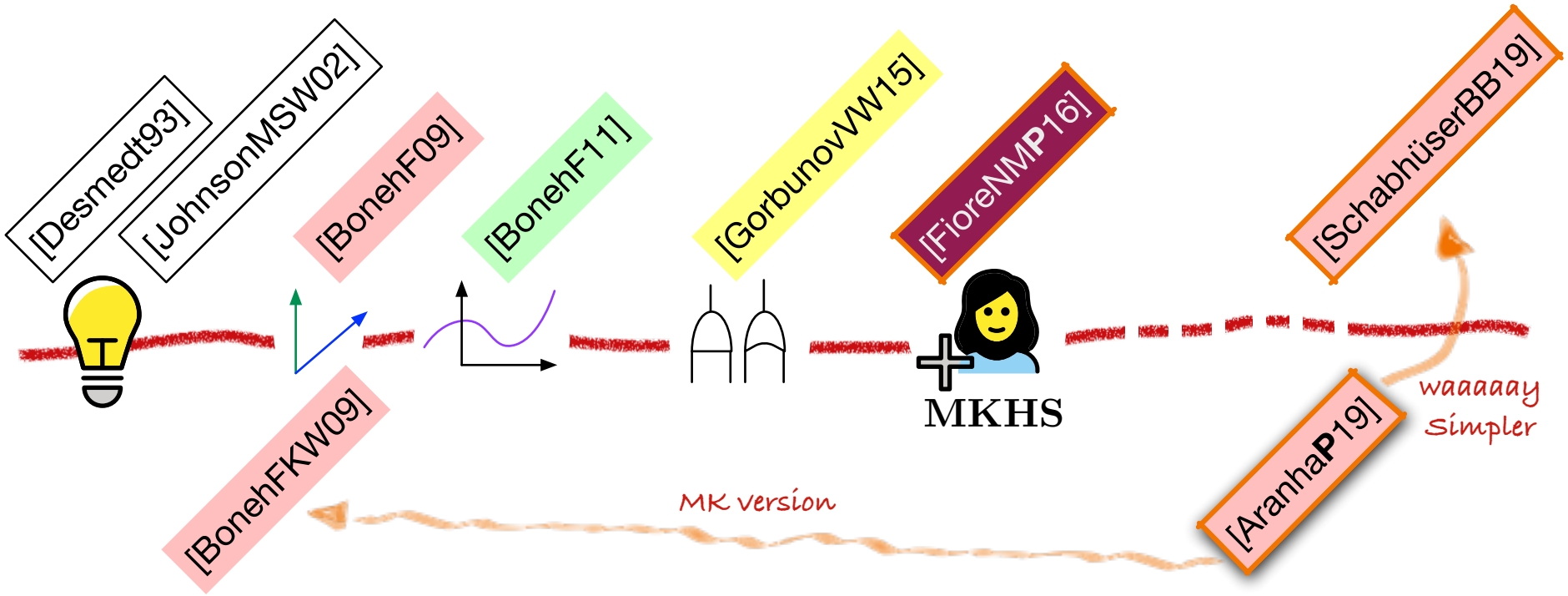
AARHUS UNIVERSITY

Elena Pagnin

Homomorphic Signatures: A Use Case

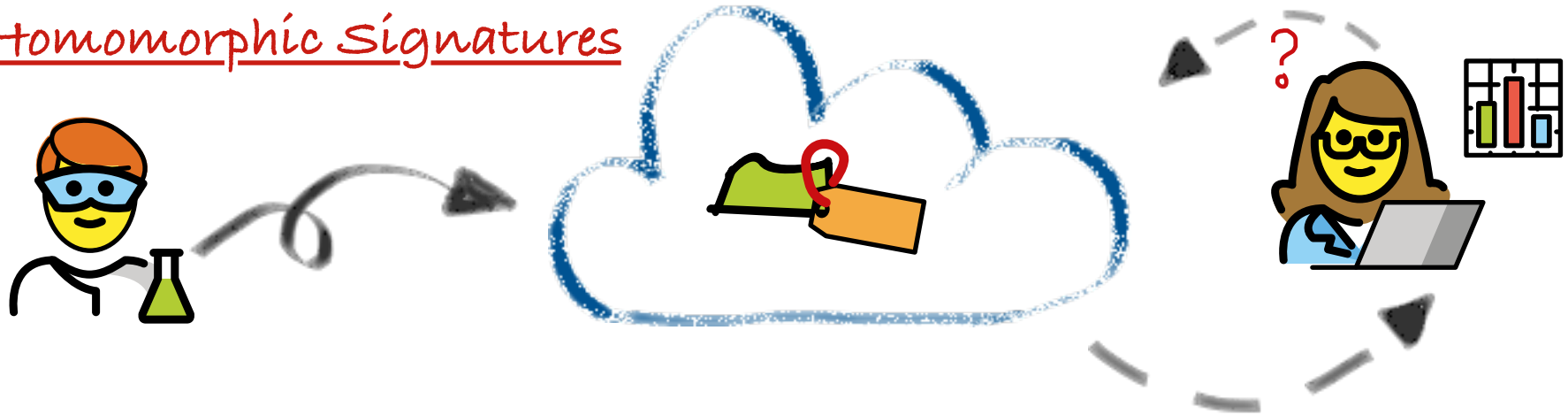


History of Homomorphic Signatures



The Simplest MKLHS scheme

Homomorphic Signatures



High Level Comparison with Relevant Work

[BonehFKW09]

Boneh, Freeman, Katz, Waters. (PKC2009) *Signing a linear subspace: Signature schemes for network coding*

Bilinear groups (type-2 pairings)

co-DDH in the random oracle model

MK version

The Simplest MKLHS scheme

[AranhaP19]

Bilinear groups (type-2 pairings)

co-DDH in the random oracle model

First direct construction (not a compiler)

Shortest signatures & fastest algorithms

$$\sigma = (H(\ell) \cdot g^m)^{\mathsf{sk}}$$

waaaaay simpler

[SchabhüserBB19]

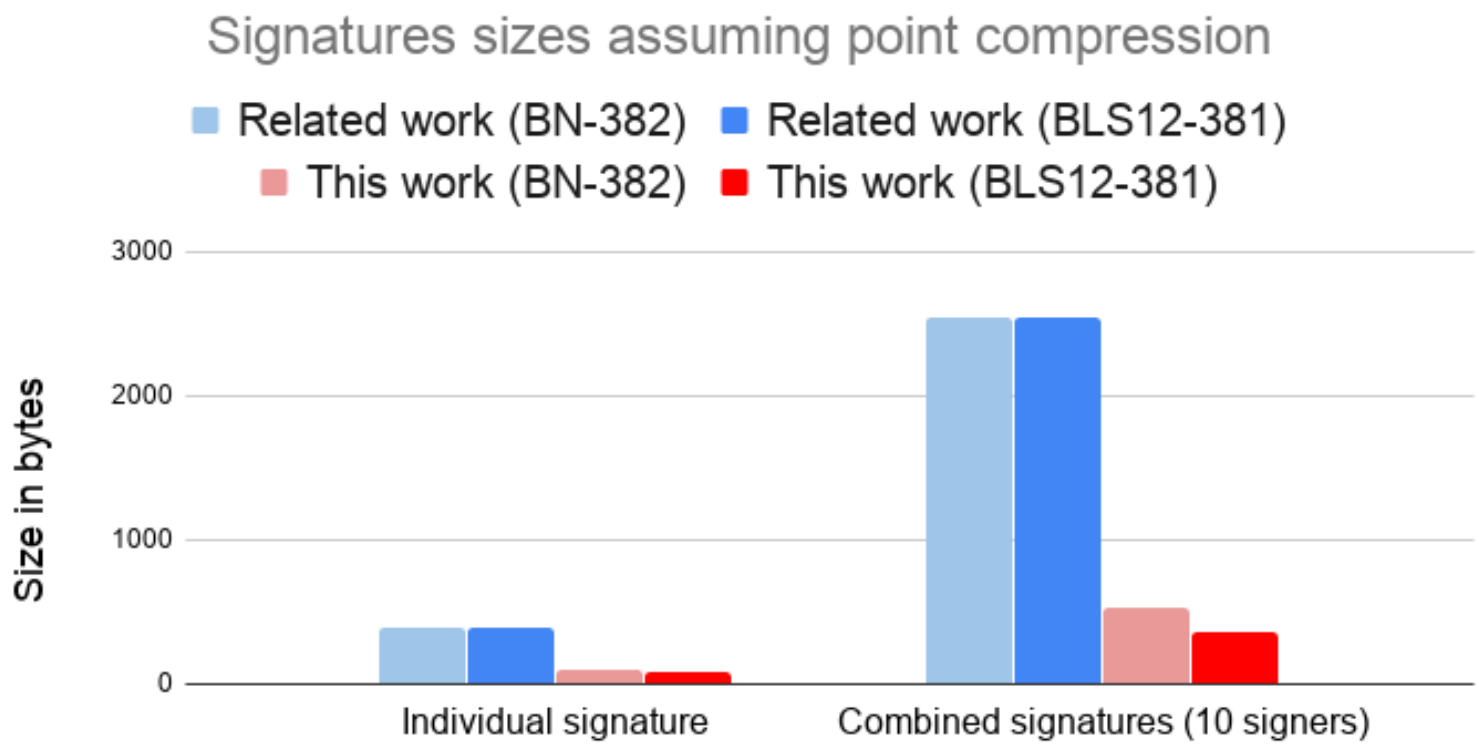
Achieves context-hiding in the multi-key settings

Compiler: any signature -> MKLHS

Bilinear groups. DDH + Flexible DH inversion hardness*

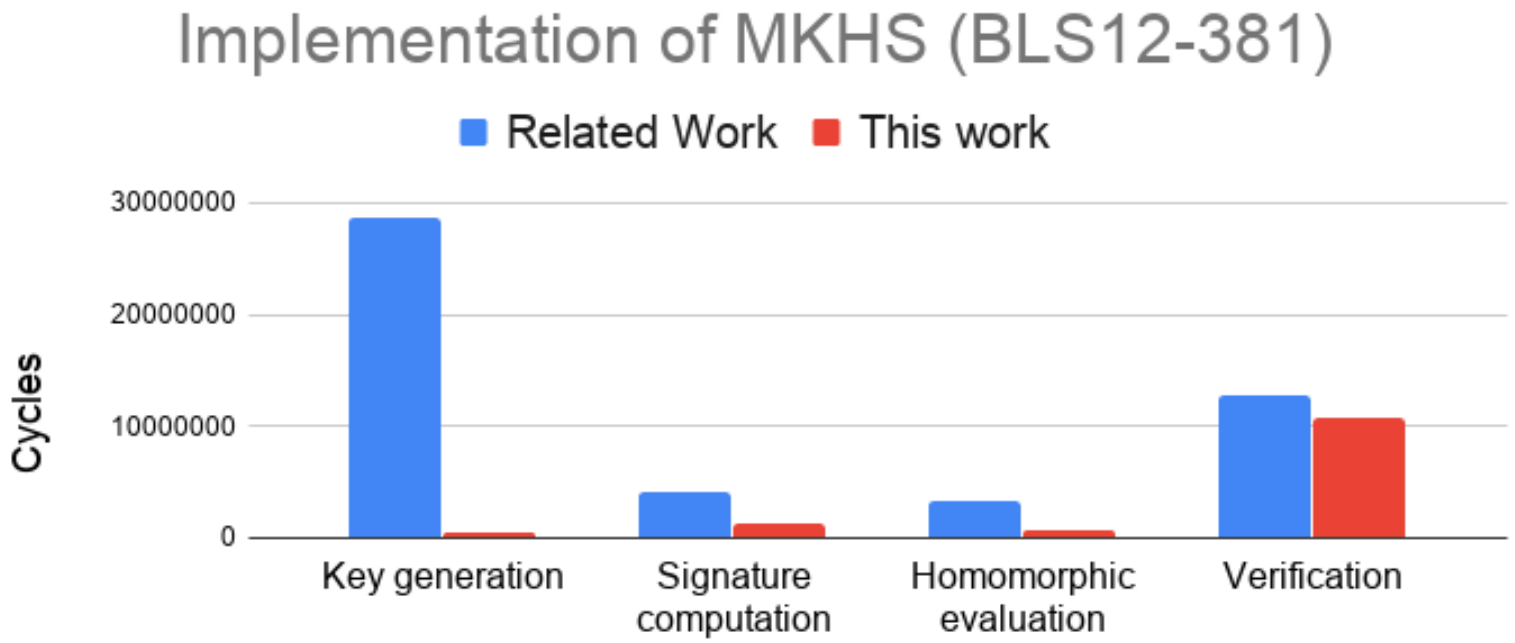
Schabhüser, Butin, Buchmann (CT-RSA2019) *Context Hiding Multi-Key Linearly Homomorphic Authenticators*

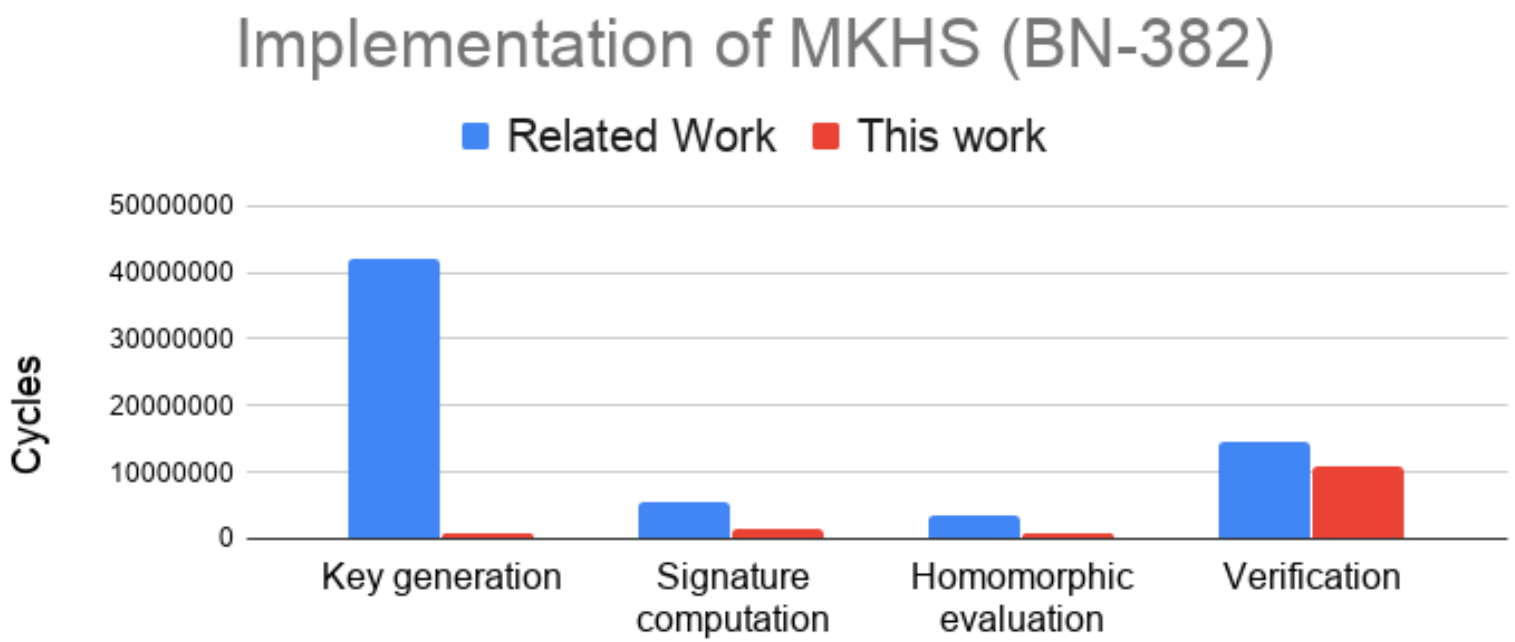
Quick Numeric Comparison with [SchabhüserBB19]



	Protocol	
Size of	Schabhüser *et al.*	mklhs
Secret Key (sk)	$\lvert\mathsf{hash\text{-}key}\rvert + \lvert\mathsf{sk}_{sig}\rvert + (\lvert\mathcal{T}\rvert + 1)\lvert\mathbb{Z}_q\rvert$	$\lvert\mathbb{Z}_q\rvert$
Public Key (pk)	$\lvert\mathsf{pk}_{sig}\rvert + \lvert\mathcal{T}\rvert \cdot \lvert\mathbb{G}_T\rvert + \lvert\mathbb{G}_2\rvert$	$\lvert\mathbb{G}_2\rvert$
Fresh Signatures	$\lvert\mathsf{ID}\rvert + \lvert\sigma_{sig}\rvert + 2\lvert\mathbb{G}_2\rvert + 3\lvert\mathbb{G}_1\rvert$	$\lvert\mathsf{ID}\rvert + \lvert\mathbb{G}_1\rvert + \lvert\mathbb{Z}_q\rvert$
Evaluated Signatures	$t\lvert\mathsf{ID}\rvert + t\lvert\sigma_{sig}\rvert + (t+1)\lvert\mathbb{G}_2\rvert + (2t+1)\lvert\mathbb{G}_1\rvert$	$t\lvert\mathsf{ID}\rvert + \lvert\mathbb{G}_1\rvert + t\lvert\mathbb{Z}_q\rvert$

Quick Numeric Comparison with [SchabhüserBB19]





MKHS: Our Construction

TYPE-2 PAIRINGS

+ hash into $\mathbb{G}_1$

+ homomorphism $\mathbb{G}_2 \rightarrow \mathbb{G}_1$

$\mathrm{Setup}(1^\lambda) \rightarrow \mathrm{pp}$ BilinGroup

$\mathrm{KeyGen}(\mathrm{pp}) \rightarrow (\mathrm{sk}, \mathrm{pk}, \mathrm{id})$

$\mathrm{sk} \xleftarrow{\$} \mathbb{Z}_q^*$

$\mathrm{pk} \leftarrow g_2^{\mathrm{sk}}$

$\mathrm{Sign}(\mathrm{sk}, \ell, m) \rightarrow \sigma$

$\sigma = (\mathrm{id}, \gamma, \mu)$

$\gamma \leftarrow (H(\ell) \cdot g_1^m)^{\mathrm{sk}}$

$\mu \leftarrow m$

$\mathrm{Eval}(f, \sigma_1, .., \sigma_n) \rightarrow \sigma = (\gamma, \mu_1, \ldots, \mu_t)$

$$\gamma \leftarrow \prod_{i=1}^{n} \gamma_i^{f_i} \qquad \mu_j \leftarrow \sum_{j \in I_j} f_i \cdot \mu_i$$

$\mathrm{Verify}(\mathscr{P}, \{\mathrm{pk}_{\mathrm{id}}\}, m, \sigma) \rightarrow$

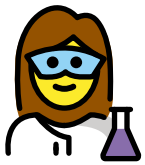
$$m == \sum_{j=1}^{t} \mu_j$$

$$e(\gamma, g_2) == \prod_{j=1}^{t} e(g_1^{\mu_j} \cdot \prod_{i \in I_j} H_i^{f_i}, \mathrm{pk}_j)$$

$$H_i^{f_i \mathrm{sk}_1} \cdot g_1^{(\sum_{I \in I_1} f_i m_i)\mathrm{sk}_1} \cdot H_i^{f_i \mathrm{sk}_2} \cdot g_1^{(\sum_{I \in I_2} f_i m_i)\mathrm{sk}_2}$$

$$\sum_{j \in I_1} f_i \cdot \mu_i$$

$$\sum_{j \in I_2} f_i \cdot \mu_i$$



The Simplest MKLHS scheme

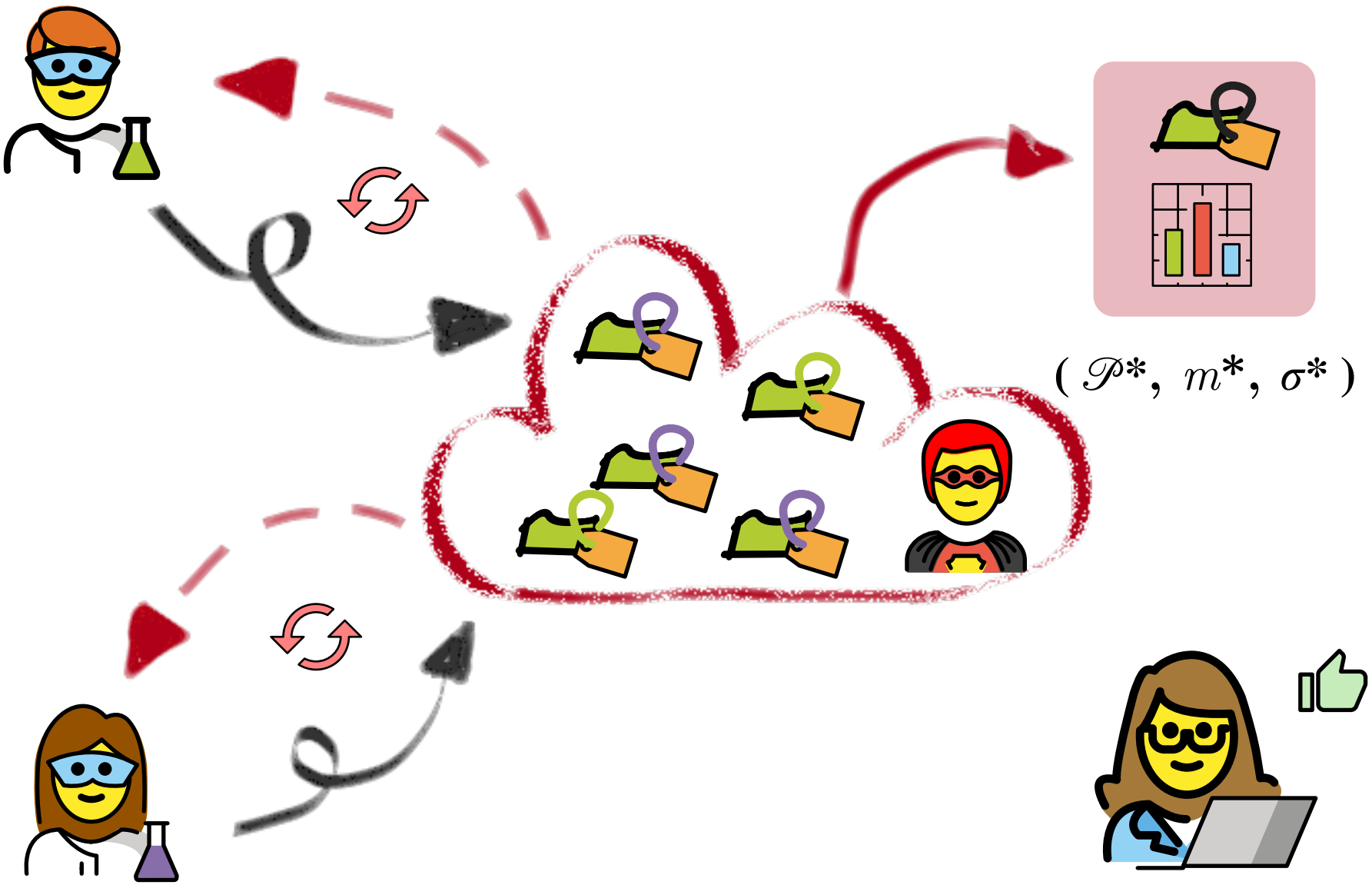
Security of our MKLHS

Theorem 1. *The* mklhs *scheme is secure in the random oracle model assuming that the co-CDH problem is computationally infeasible. In detail, let $\mathscr{A}$ be a probabilistic polynomial-time adversary in the security experiment (HUFCMA) described in Section 2.2, then its advantage is bounded by*

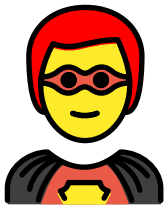
$$\mathsf{Adv}_{\mathsf{mklhs},\mathscr{A}}[\lambda] \leq \frac{1}{2} \cdot \left[R_H + Q_{\mathsf{id}} \cdot \mathsf{Adv}_{\mathscr{B}}^{\mathsf{coCDH}[\lambda]} \right].$$

where $\mathscr{B}$ is a polynomial-time algorithm that solves a co-CDH instance with probability $\mathsf{Adv}_{\mathscr{B}}^{\mathsf{coCDH}}[\lambda]$, $R_H = \frac{1}{\mathsf{poly}(\lambda)}$ is determined by the prime number q corresponding to the order of the group $\mathbb{G}_1$ (the range of the hash function) and $Q_{\mathsf{id}} = \mathsf{poly}(\lambda)$ is the total number of identities generated during the game.

Security Model: Malicious Evaluatos



Security Model: Forgeries



$(\mathscr{P}^*,\ m^*,\ \sigma^*)$

no corrupted users in $\mathcal{P}^*$

<u>Forgery:</u> $\mathsf{Ver}\big(\mathcal{P}^*, \{\mathsf{pk}_{\mathsf{id}}\}, m^*, \sigma^*\big) = 1$

1) the adversary successfully tampers with the **output** of the function

$$m^* \neq f^*(m_1, \ldots, m_t)$$

2) the adversary never did an authentication query for a **label** in $\mathscr{P}^*$

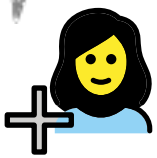
$$\exists\ k \in [t] \quad : \quad \ell_k^* \notin Q_{\mathsf{labels}}$$

Why not ... security against corrupted users?

some sort of input secrecy?

[FioreNM**P**16]

+ first construction



MKHS

[Fiore**P**18]

HS → MKHS

[Aranha**P**19]

+ direct construction

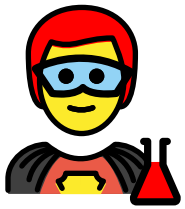
?

DS+SNARK → MKHS

+ security under insider corruption

+ full signature succinctness

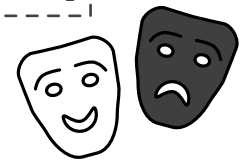
[LaiTWC18]



DS → MKLHS

+ context-hiding

[SchabhüserBB19]



The Simplest MKLHS scheme