



IEEE Security & Privacy 2025

Ringtail: Practical Two-Round Threshold Signatures from Learning with Errors

Cecilia
Boschini

*ETH
Zürich*

cecilia.boschini
@inf.ethz.ch

**Darya
Kaviani**

*UC
Berkeley*

daryakaviani
@berkeley.edu

Russell
W. F. Lai

*Aalto
University*

russell.lai
@aalto.fi

Giulio
Malavolta

*Bocconi
University*

giulio.malavolta
@hotmail.it

Akira
Takahashi

*J.P. Morgan AI
Research &
AlgoCRYPT CoE*

takahashi.akira.58s
@gmail.com

Mehdi
Tibouchi

*NTT
(Japan)*

mehdi.tibouchi
@normalesup.org

(t, n) -Threshold Signatures

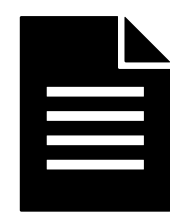


Signing key is split into several **shares** such that no server holds the original secret key.

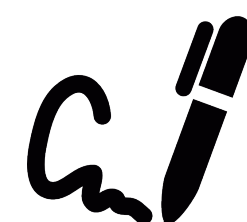


Any t out of n parties can jointly produce a valid signature for the same public key.

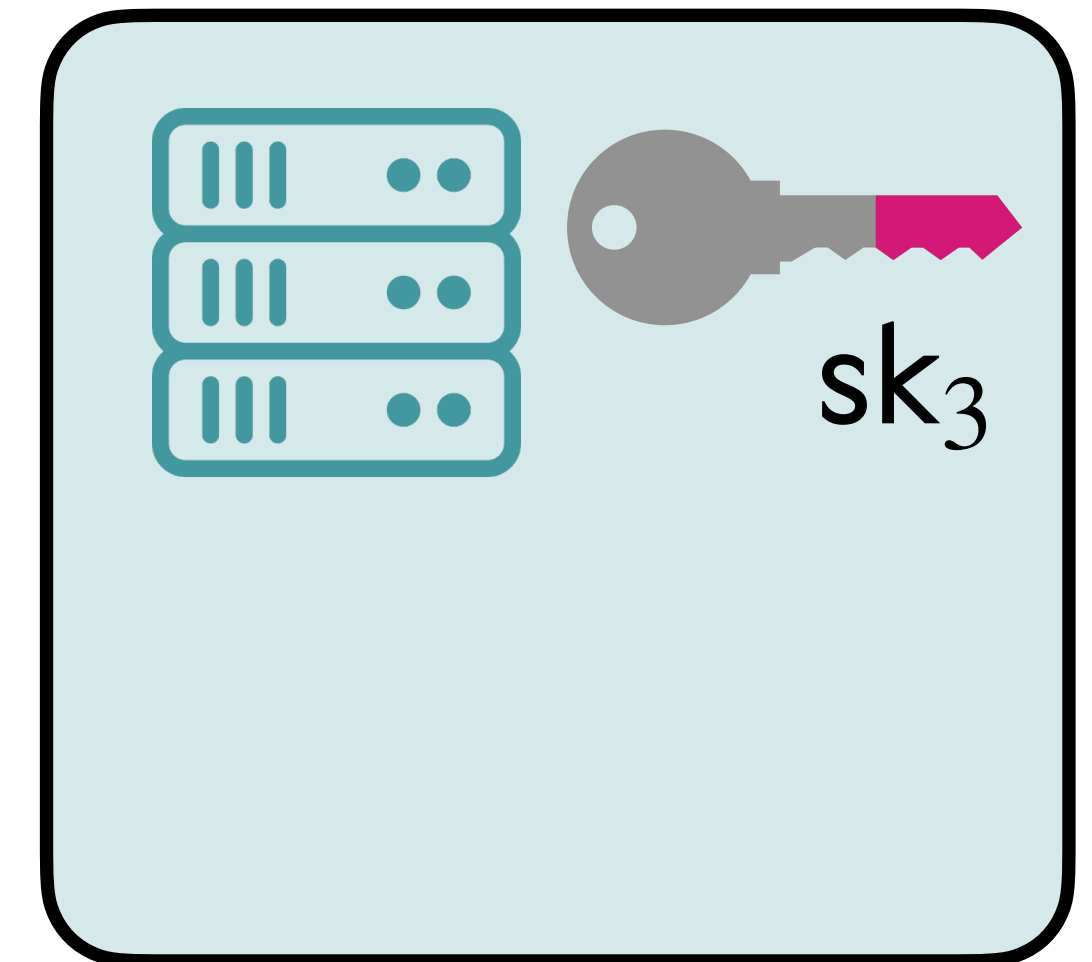
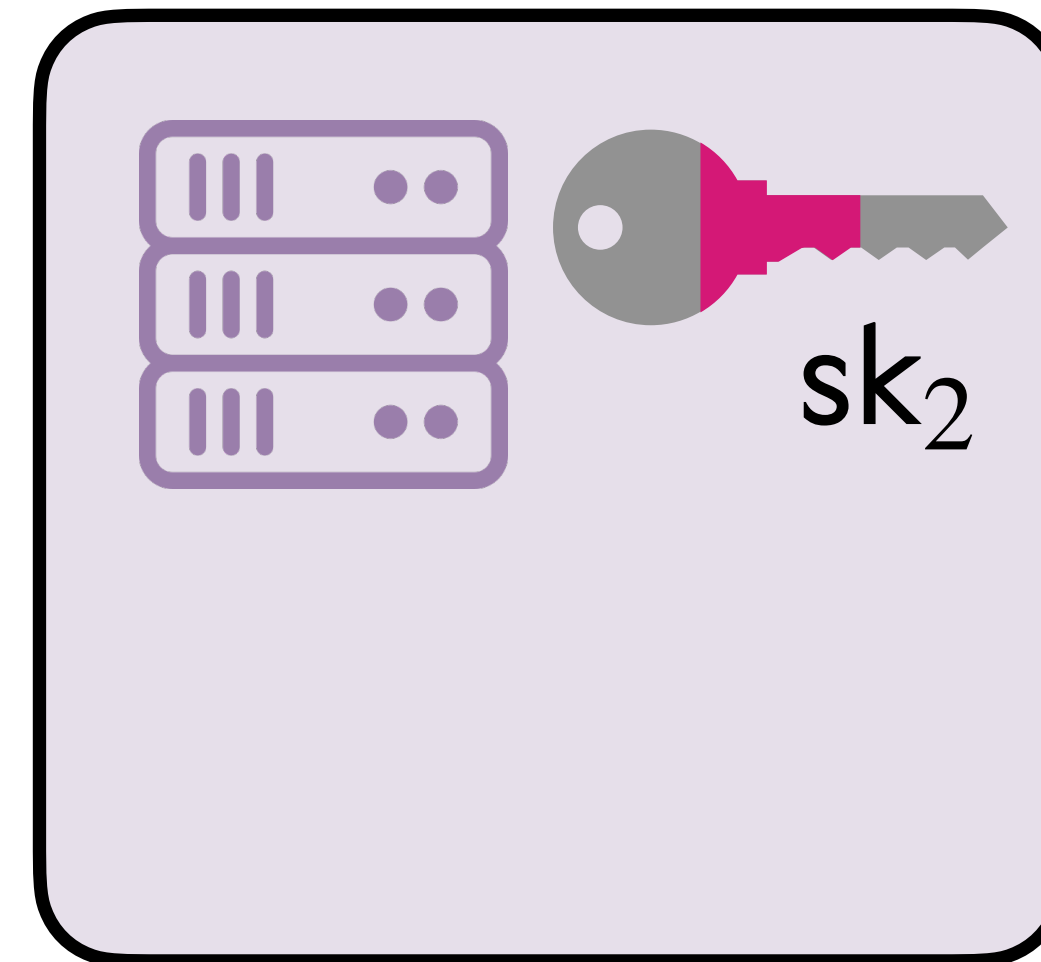
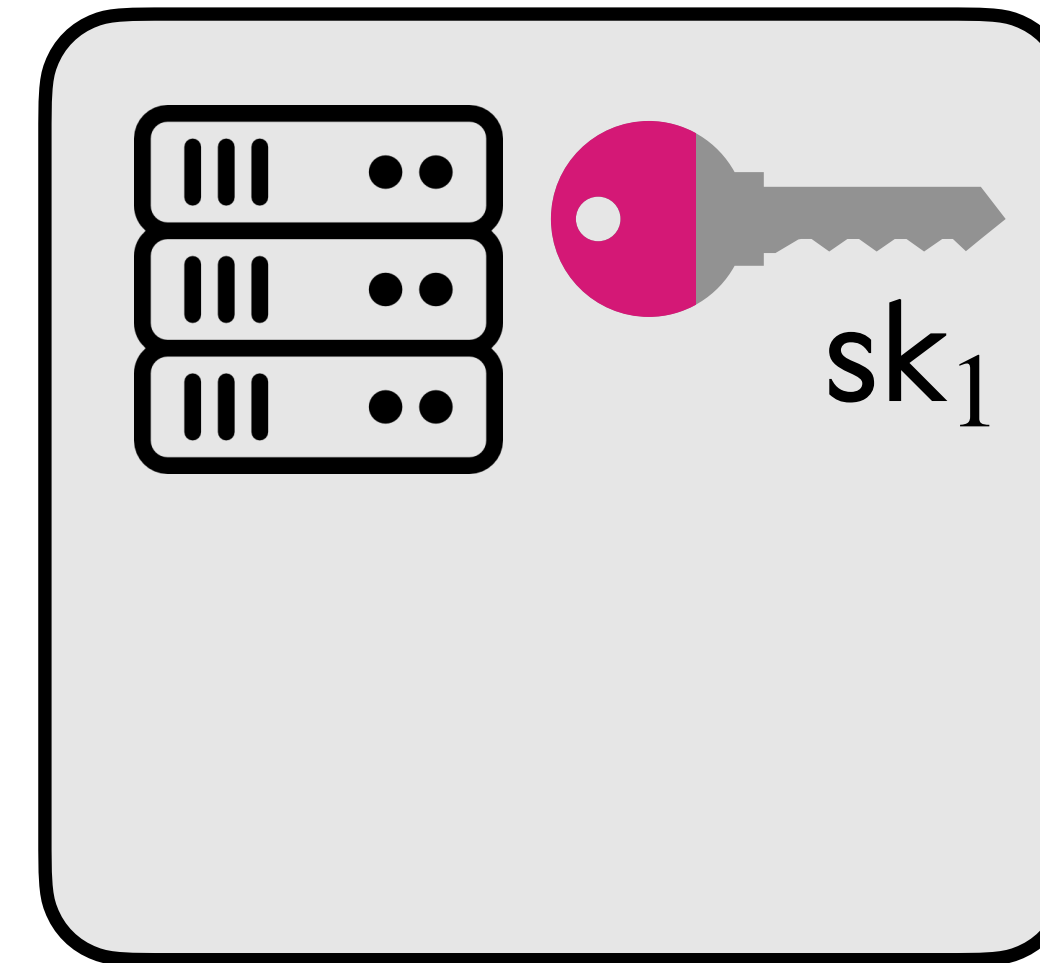
Not $t - 1$!



m



σ



(t, n) -Threshold Signatures



Signing key is split into several **shares** such that no server holds the original secret key.

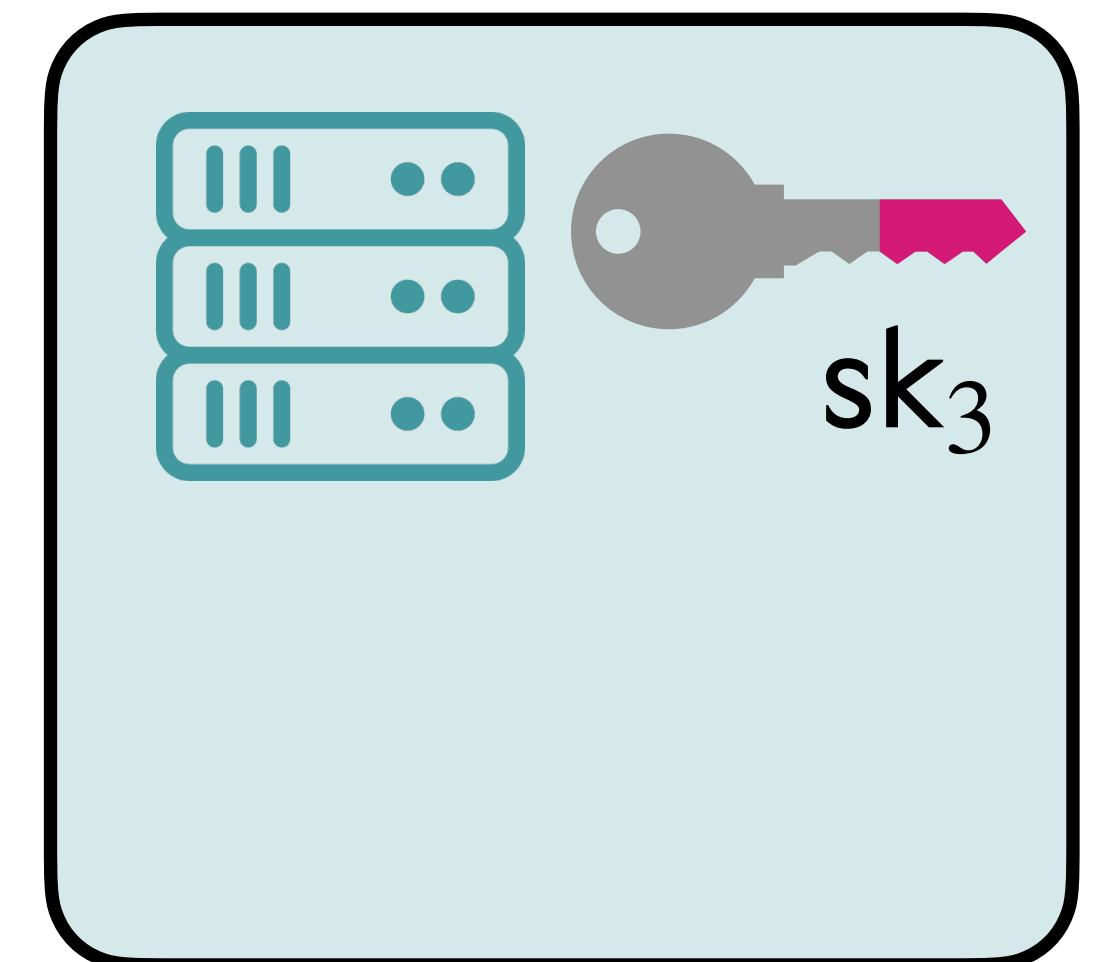
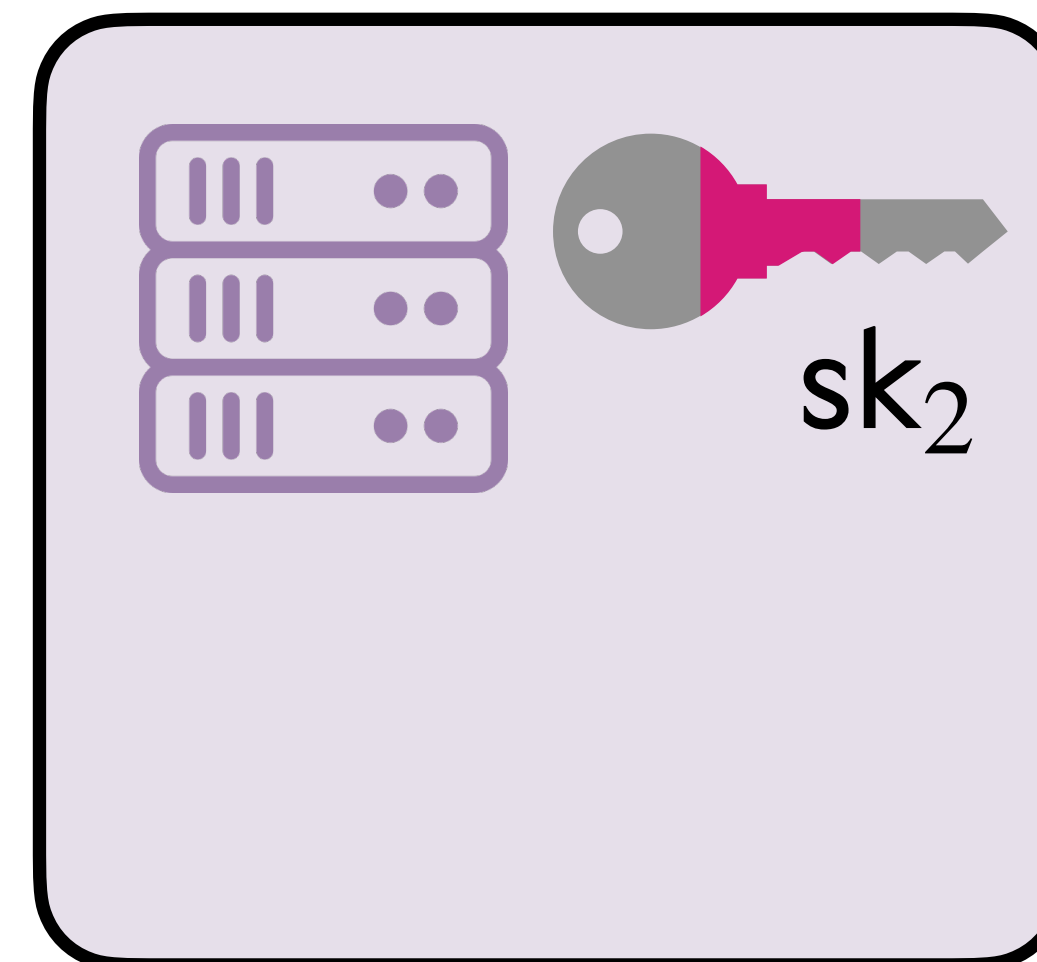
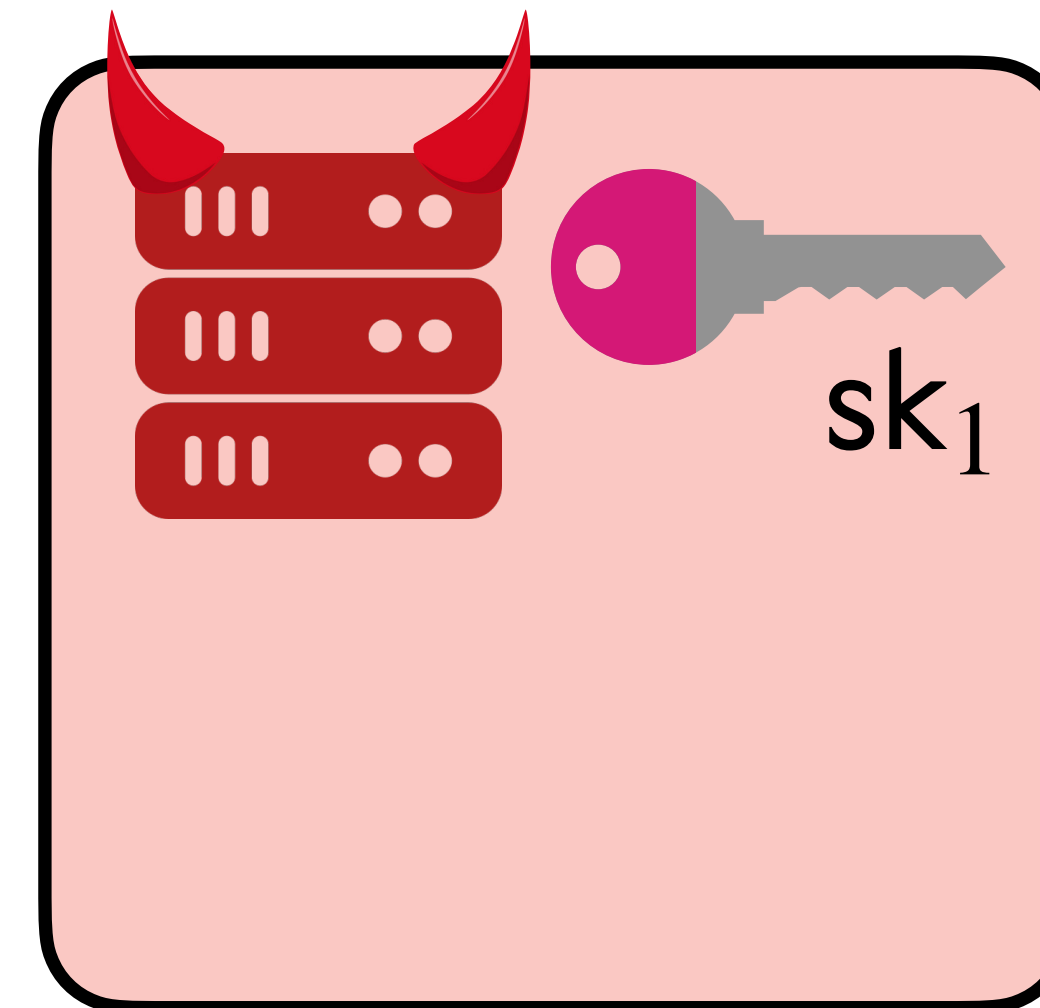


Any t out of n parties can jointly produce a valid signature for the same public key.

Not $t - 1$!



Even if a subset of servers are corrupted, the secret key does not leak.



NIST Call: Multi-Party Threshold Cryptography



PUBLICATIONS

NIST IR 8214C (2nd Public Draft) 

NIST First Call for Multi-Party Threshold Schemes

[Documentation](#) [Topics](#)

Date Published: March 27, 2025
Comments Due: May 30, 2025

Includes a call for additional PQ signatures!

Ringtail in a Nutshell



The signing protocol consists of only **2 rounds**.
(1 offline + 1 online)



13.4 KB signature size & 10.5 KB of online communication for 128-bit security & $t = 1024$ parties.




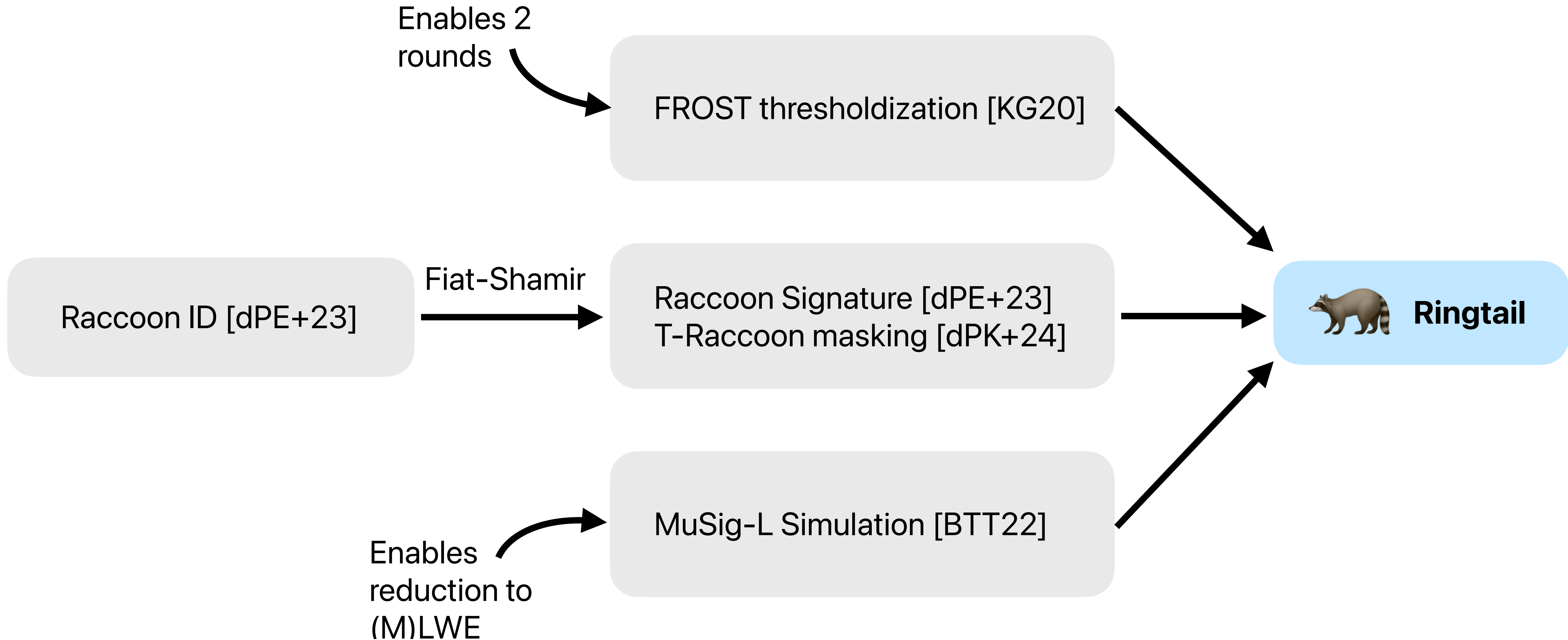
Provably secure from standard (module) **LWE** & **SelfTarget SIS** assumptions in the classical **ROM**.




Heavy offline communication (~**600 KB**), though this seems to be an artifact of the security proof.

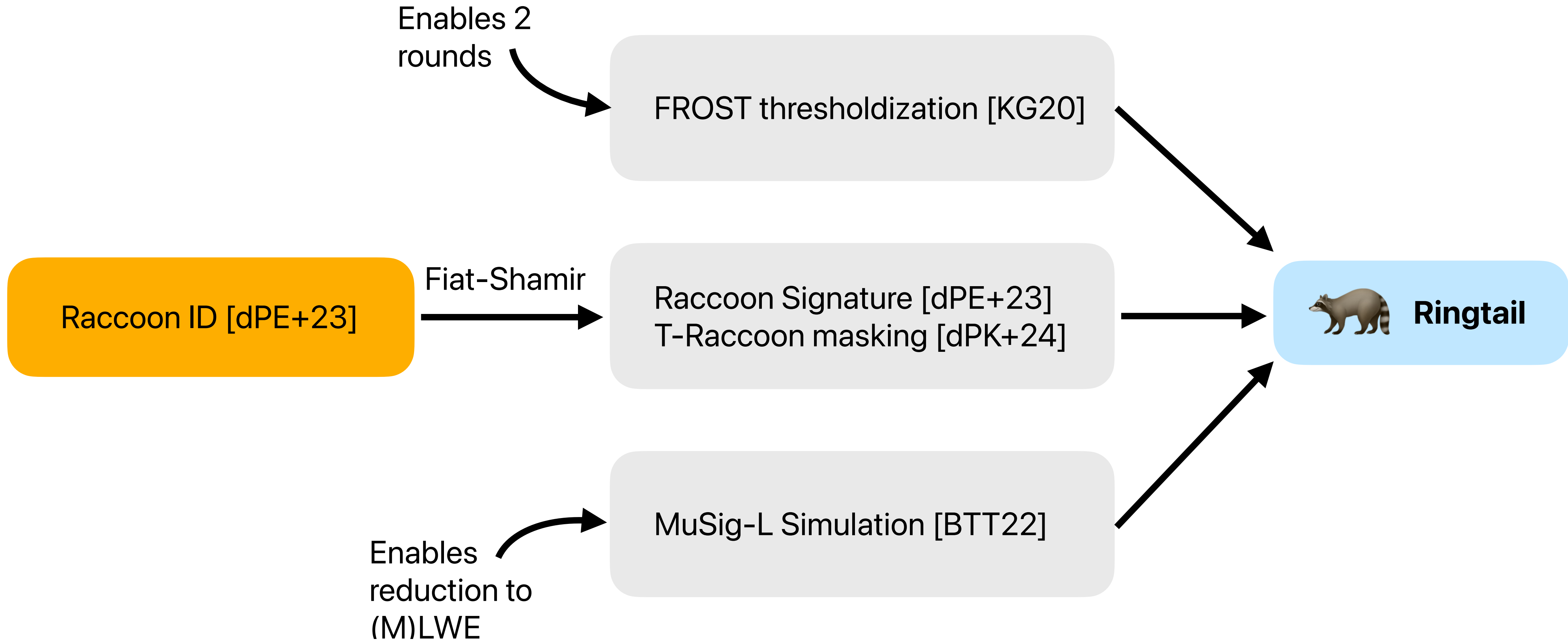
The Road to Ringtail

 Generic approaches (MPC or FHE) are not concretely efficient.

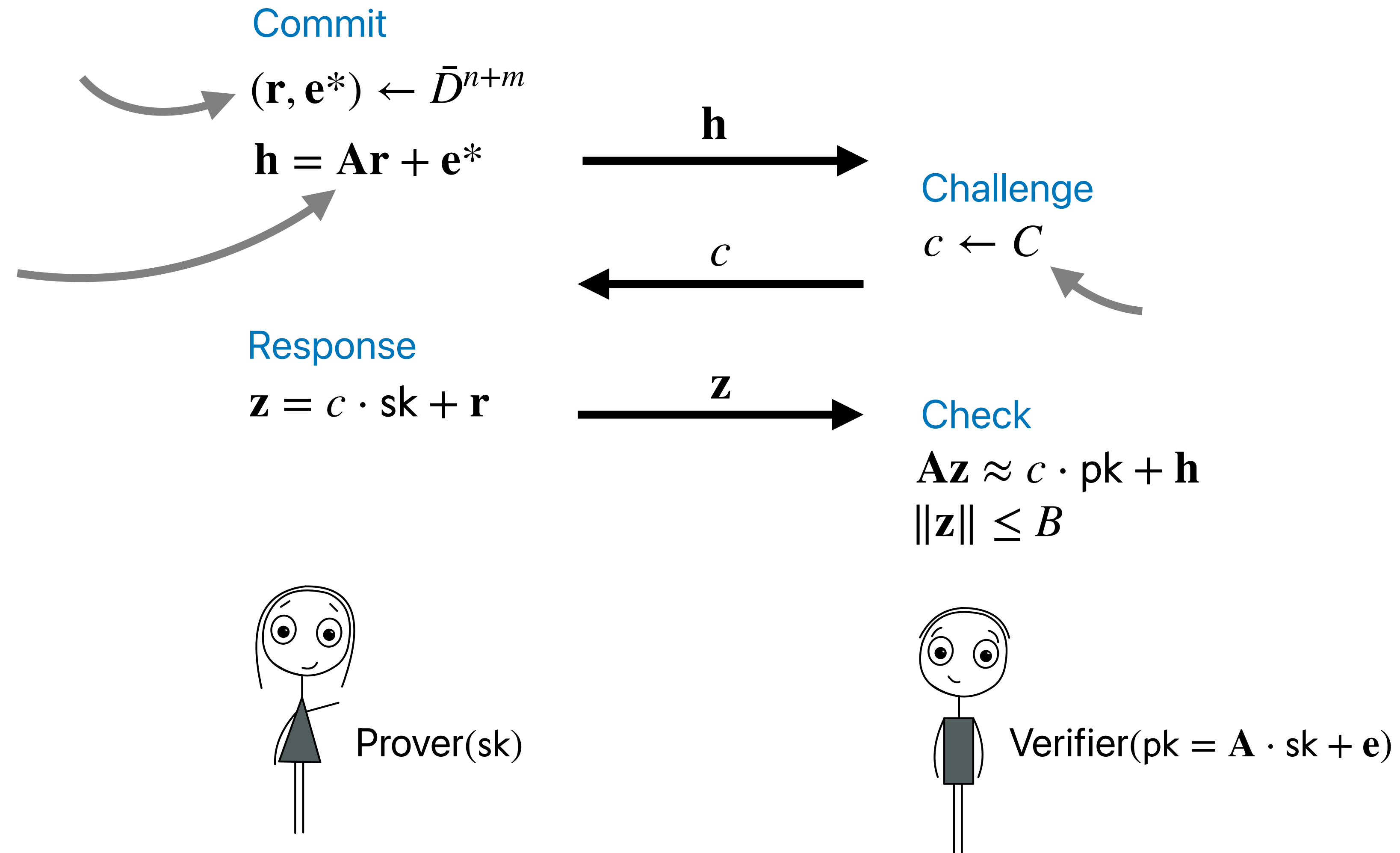


The Road to Ringtail

 Generic approaches (MPC or FHE) are not concretely efficient.



Starting Point: Raccoon ID [dPE+23]



Starting Point: Raccoon ID [dPE+23]

Commit

$$(\mathbf{r}, \mathbf{e}^*) \leftarrow \bar{D}^{n+m}$$

$$\mathbf{h} = \mathbf{A}\mathbf{r} + \mathbf{e}^*$$



Challenge

$$c \leftarrow C$$



Response

$$\mathbf{z} = c \cdot \mathbf{s}k + \mathbf{r}$$



Check

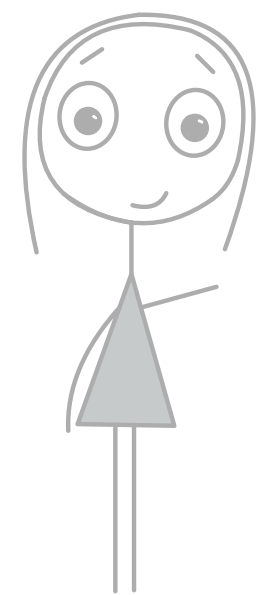
$$\mathbf{A}\mathbf{z} \approx c \cdot \mathbf{p}k + \mathbf{h}$$

$$\|\mathbf{z}\| \leq B$$

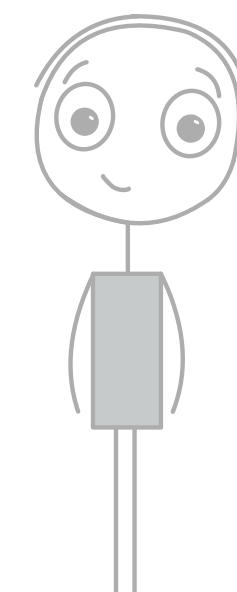
Verification

$$\begin{aligned} c \cdot \mathbf{p}k + \mathbf{h} &= c(\mathbf{A}\mathbf{s}k + \mathbf{e}) + \mathbf{A}\mathbf{r} + \mathbf{e}^* \\ &= \mathbf{A}(c \cdot \mathbf{s}k + \mathbf{r}) + \underbrace{c \cdot \mathbf{e} + \mathbf{e}^*}_{\text{Small!}} \approx \mathbf{A}\mathbf{z} \end{aligned}$$

Small!

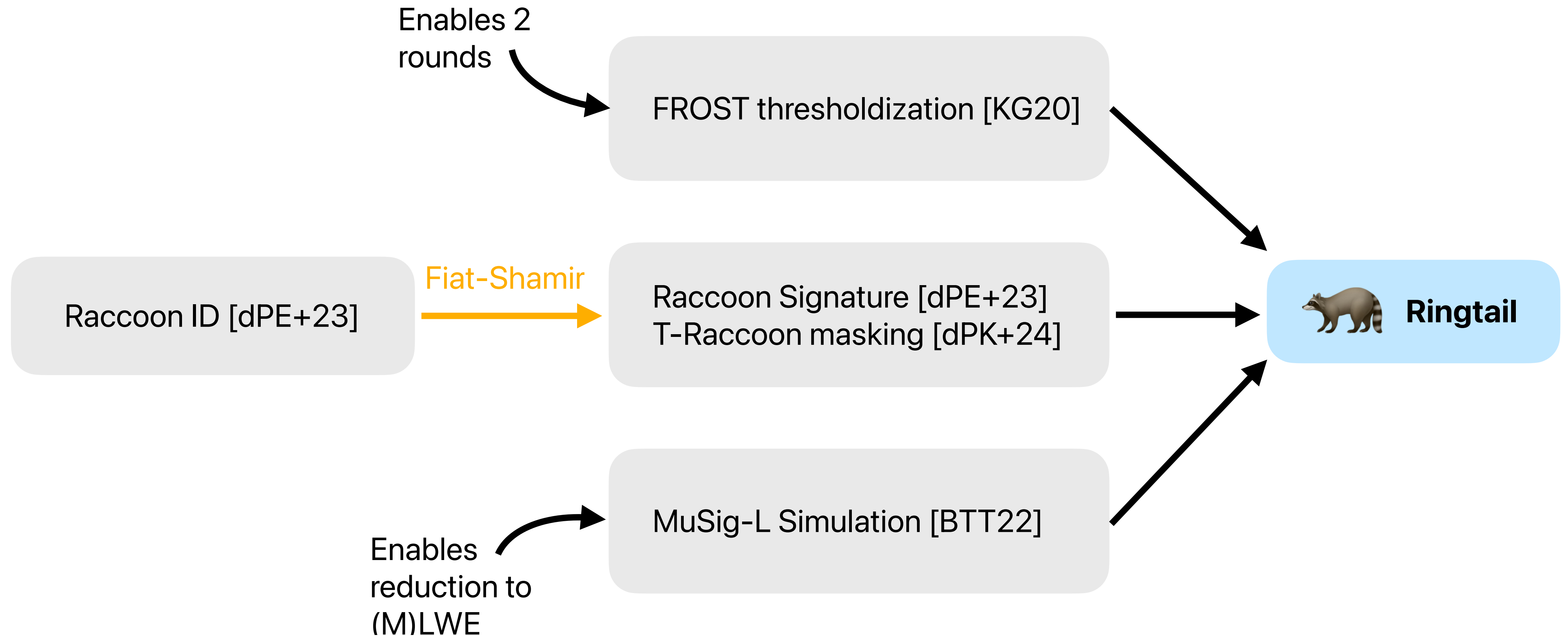


Prover($\mathbf{s}k$)



Verifier($\mathbf{p}k = \mathbf{A} \cdot \mathbf{s}k + \mathbf{e}$)

The Road to Ringtail



Raccoon Signature [dPE+23]

Commit

$$(\mathbf{r}, \mathbf{e}^*) \leftarrow \bar{D}^{n+m}$$

$$\mathbf{h} = \mathbf{A}\mathbf{r} + \mathbf{e}^*$$

Fiat-Shamir

$$c = H(\text{pk}, m, \mathbf{h})$$

Response

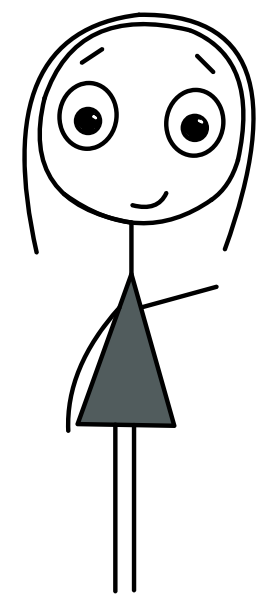
$$\mathbf{z} = c \cdot \text{sk} + \mathbf{r}$$

\mathbf{z}

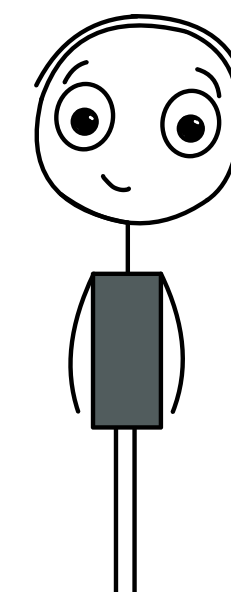
Check

$$\mathbf{A}\mathbf{z} \approx c \cdot \text{pk} + \mathbf{h}$$

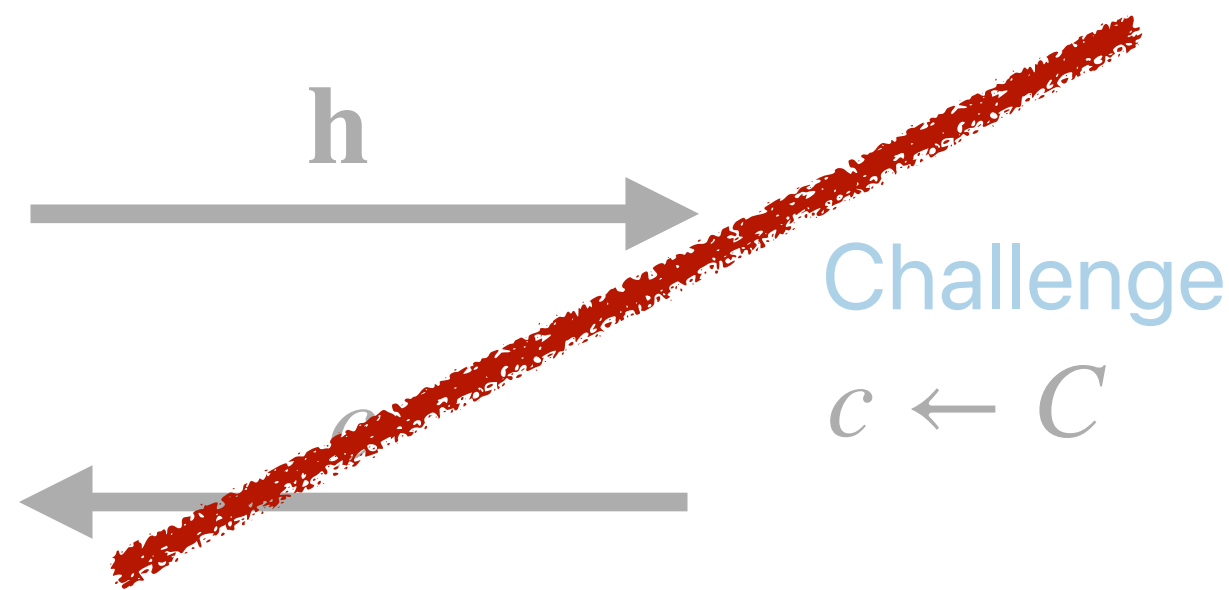
$$\|\mathbf{z}\| \leq B$$



Prover(sk)



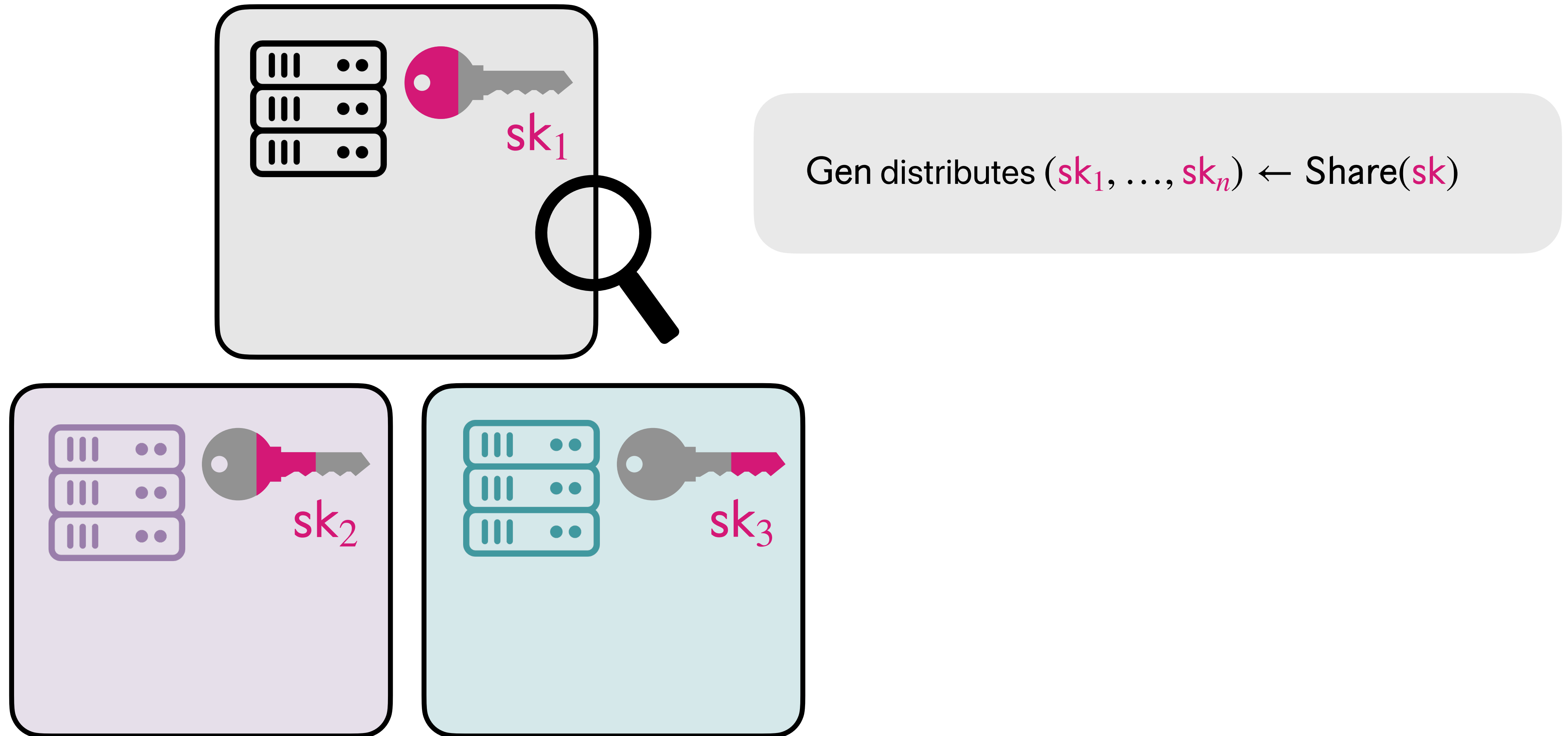
Verifier(pk = $\mathbf{A} \cdot \text{sk} + \mathbf{e}$)



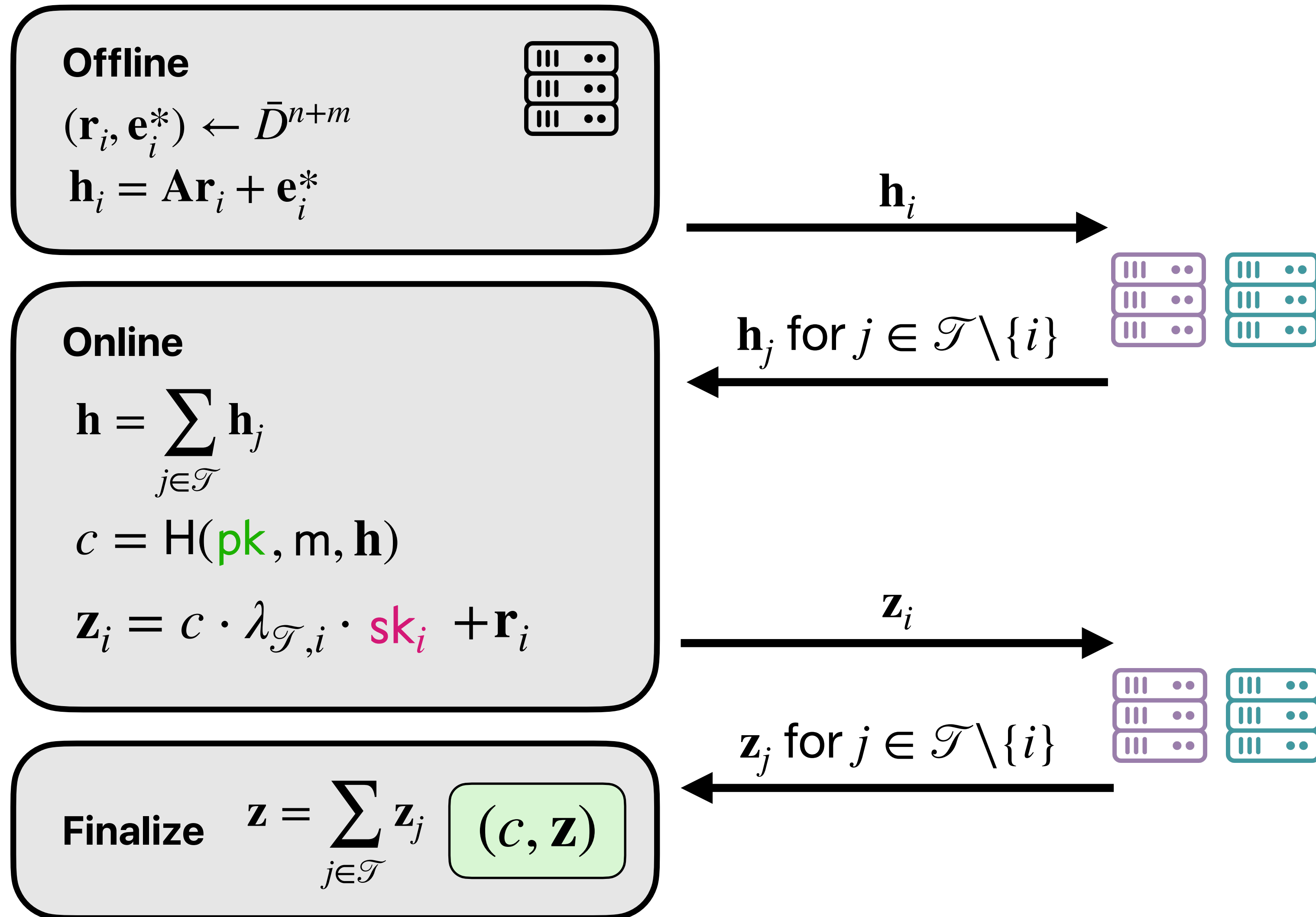
Challenge

$$c \leftarrow C$$

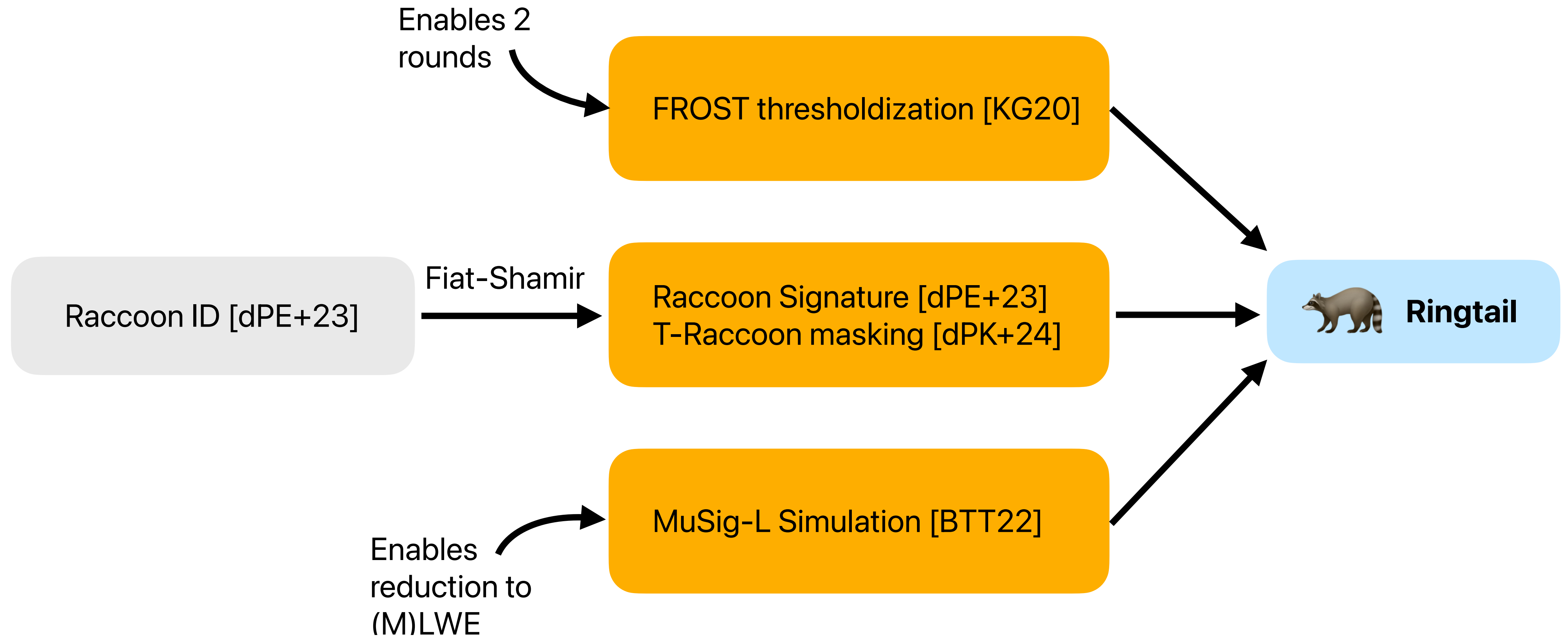
Thresholdizing Lattice-Based Signatures



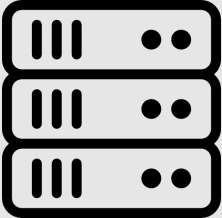
Thresholdizing Lattice-Based Signatures



The Road to Ringtail



Protecting Against Active Adversaries

Offline $(\mathbf{r}_i, \mathbf{e}_i^*) \leftarrow \bar{D}^{n+m}$ 
 $\mathbf{R}_i \leftarrow D^{n \times d}; \mathbf{E}_i \leftarrow D^{m \times d}$
 $\mathbf{D}_i = \mathbf{A}[\mathbf{r}_i | \mathbf{R}_i] + [\mathbf{e}_i^* | \mathbf{E}_i]$

One-time PRF mask with 0-share [dPK+24] [KRT24]

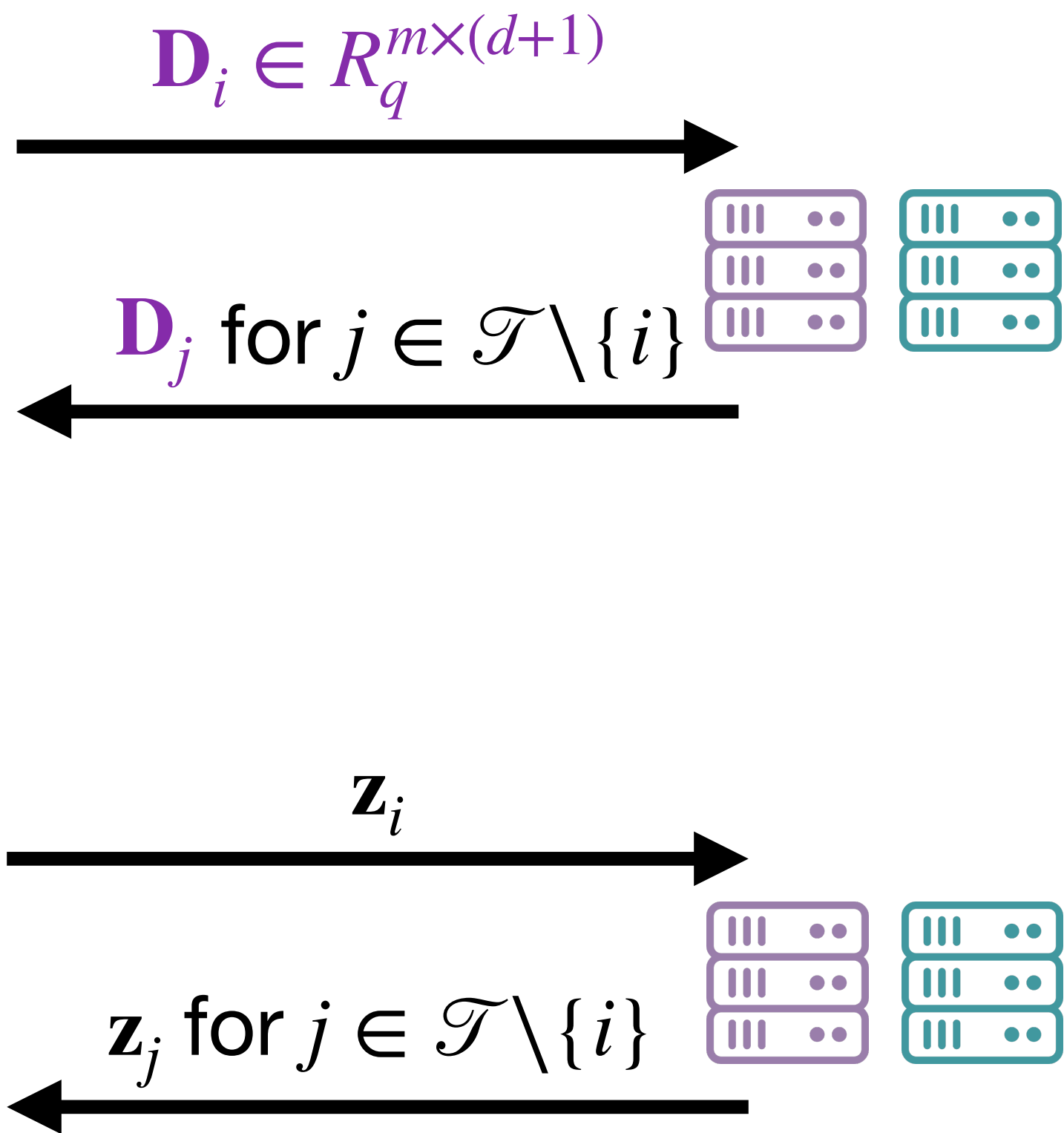
Online $\mathbf{D} = \sum_{j \in \mathcal{T}} \mathbf{D}_j$
 $\mathbf{u} = \mathbf{H}(\mathbf{pk}, m, (\mathbf{D}_j)_{j \in \mathcal{T}})$
 $\mathbf{h} = \mathbf{D}\mathbf{u}$
 $c = \mathbf{H}(\mathbf{pk}, m, \mathbf{h})$
 $\mathbf{z}_i = c \cdot \lambda_{\mathcal{T}, i} \cdot \mathbf{sk}_i + \mathbf{r}_i + \mathbf{R}_i \mathbf{u} + \mathbf{m}_i$

Interactive adversary can forge with malicious $(\mathbf{h}_j, \mathbf{z}_j)$. [DEK+19] [DOTT21/22]

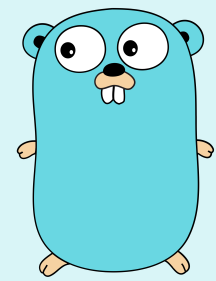
Finalize $\mathbf{z} = \sum_{j \in \mathcal{T}} \mathbf{z}_j$ (c, \mathbf{z})

FROST-Schnorr random linear combination [KG20]

Find additional protocol details in our paper!



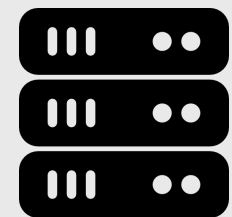
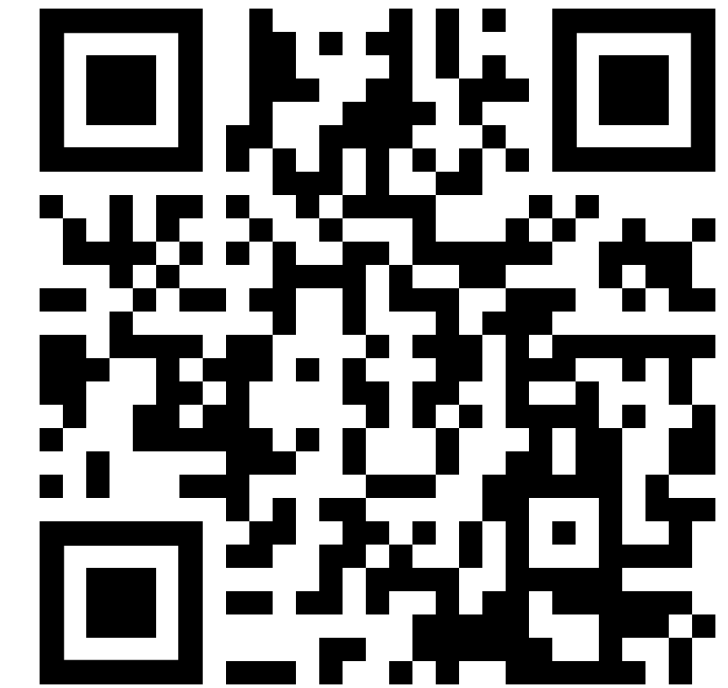
Implementation & Evaluation



1800 lines of Go



github.com/daryakaviani/ringtail



Servers

AWS 16 vCPU, 32 GB c5.4xlarge

tRaccoon
[dPK+24]

*3 rounds & standard
assumptions*

EKT
[EKT24]¹

*2 rounds & non-
standard assumptions*



¹[ZT25] proves EKT and Ringtail secure under standard assumptions via a new approach, at the cost of larger parameters.

Bandwidth (KB)

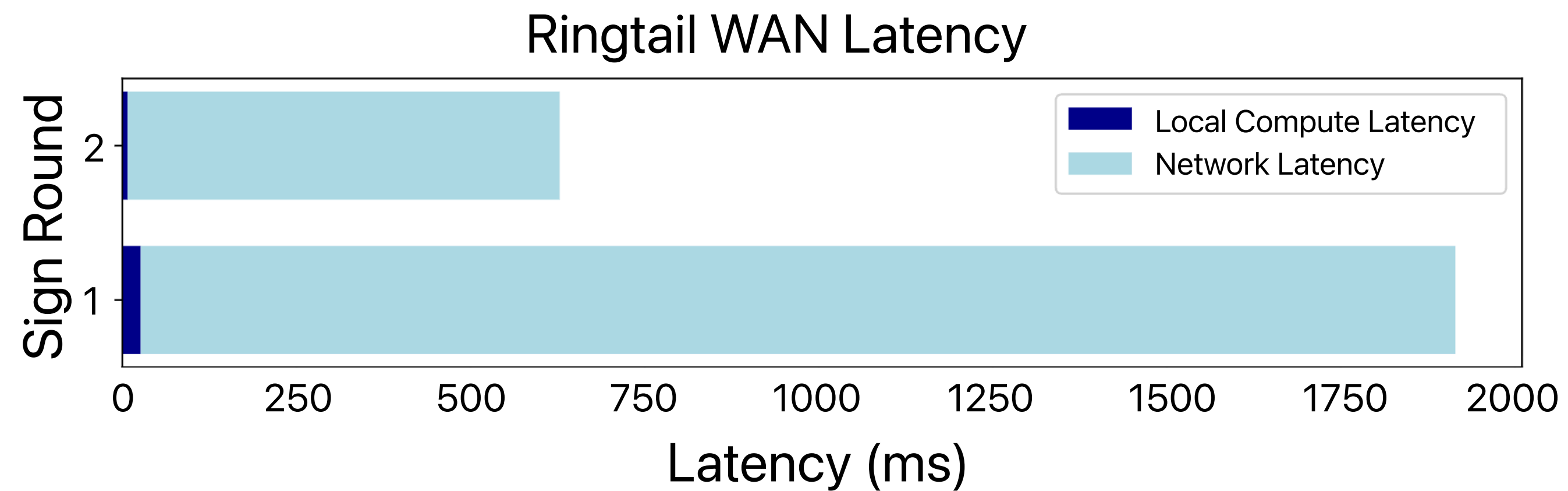
Scheme	Public Key Size	Signature Size	Sign (Total)	Sign (Online)
Ringtail	4.50	13.38	$598.50 + 0.02 t$	10.50
tRaccoon [dPK+24]	3.77	12.44	$39.84 + 0.02 t$	$39.84 + 0.02 t$
EKT [EKT24]	5.50	10.80	275.69	14.06



While Ringtail's local latency & total communication costs are higher, it exhibits the **best online costs**.

Global Wide Area Network Latency

$$t = n = 8$$



Network latency dominates the E2E latency, highlighting the importance of minimizing the number of rounds.

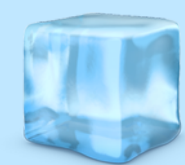
Ringtail



Two-round threshold signature scheme with 1 offline round + 1 online round.



Reasonable signature sizes & communication costs.



Reduction to standard lattice-based assumptions.



Underscores importance of message-specific roundtrips in the WAN setting.



Future work: Reducing the communication complexity.

Better security guarantees e.g., adaptive security or UC security.



eprint.iacr.org/2024/1113

github.com/daryakaviani/ringtail

 @daryakaviani

 cs.berkeley.edu/~daryakaviani

 daryakaviani@berkeley.edu