Almost Tight Multi-User Security under Adaptive Corruptions & Leakages in the Standard Model

Shuai Han, Shengli Liu, Dawu Gu

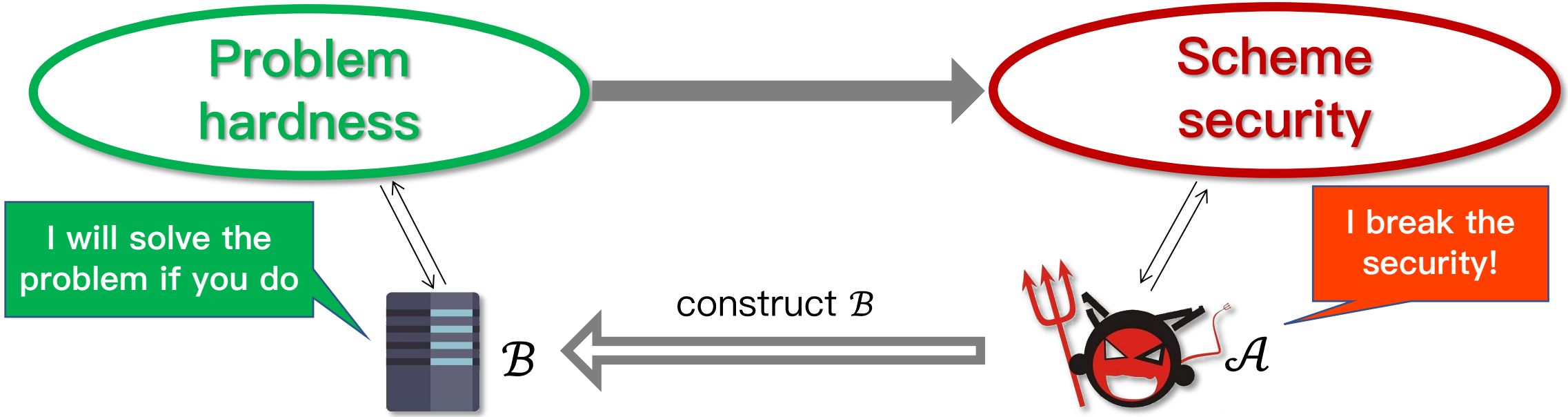
Shanghai Jiao Tong University

Eurocrypt 2023, Lyon, France

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

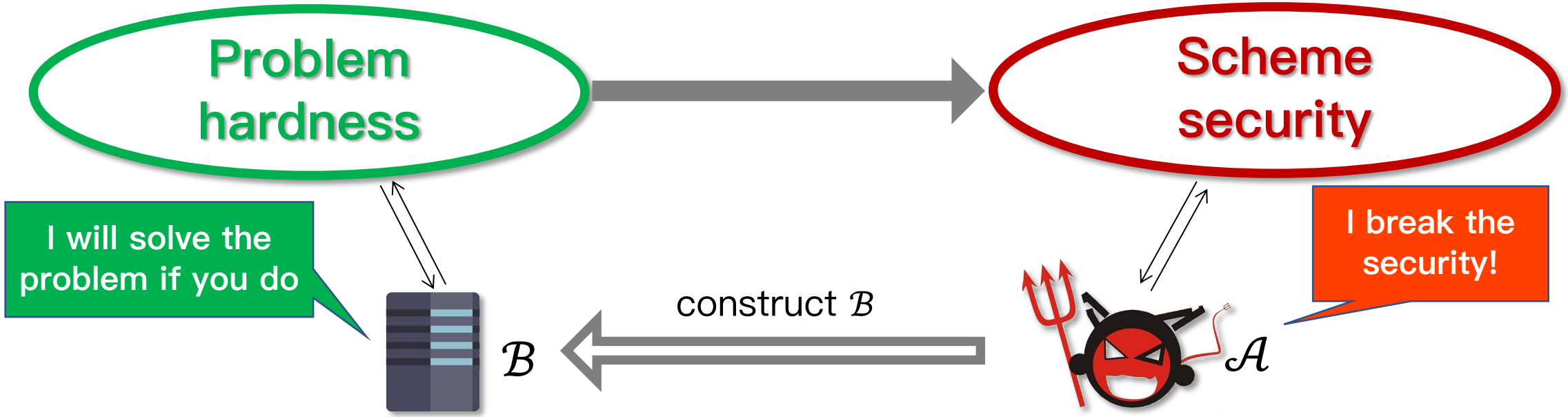
Almost Tight Security

Security of a cryptographic **Scheme** based on a hard **Problem**.



Almost Tight Security

Security of a cryptographic **Scheme** based on a hard **Problem**.



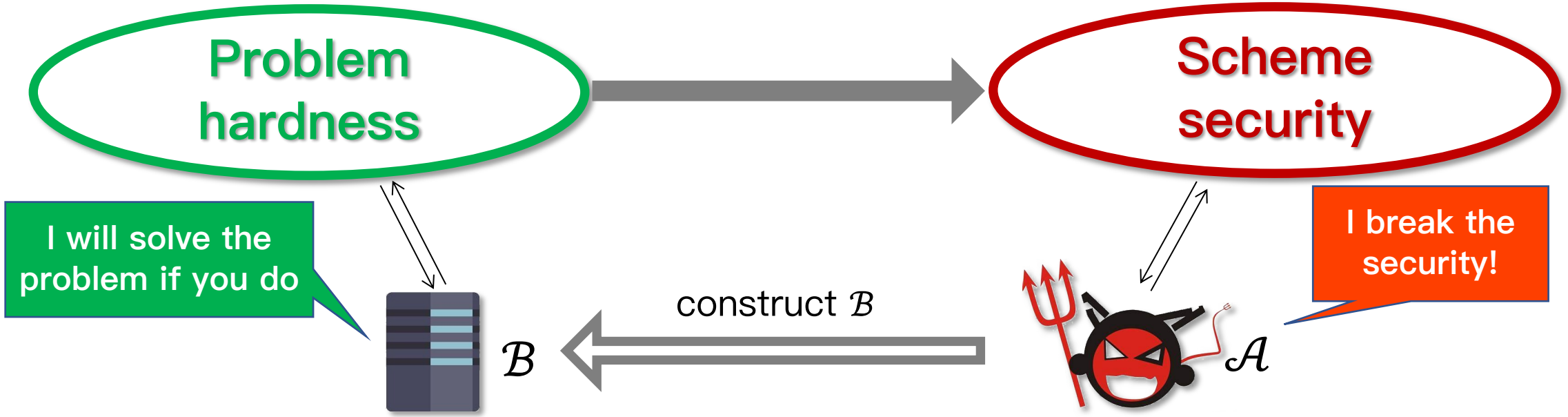
solving **Problem** in time $t_{\mathcal{B}}$ with advantage $\epsilon_{\mathcal{B}}$

attacking **Scheme** in time $t_{\mathcal{A}}$ with advantage $\epsilon_{\mathcal{A}}$

$$\frac{t_{\mathcal{B}}}{\epsilon_{\mathcal{B}}} \leq \frac{t_{\mathcal{A}}}{\epsilon_{\mathcal{A}}} \cdot \ell$$

Almost Tight Security

Security of a cryptographic **Scheme** based on a hard **Problem**.



solving **Problem** in time $t_{\mathcal{B}}$ with advantage $\epsilon_{\mathcal{B}}$

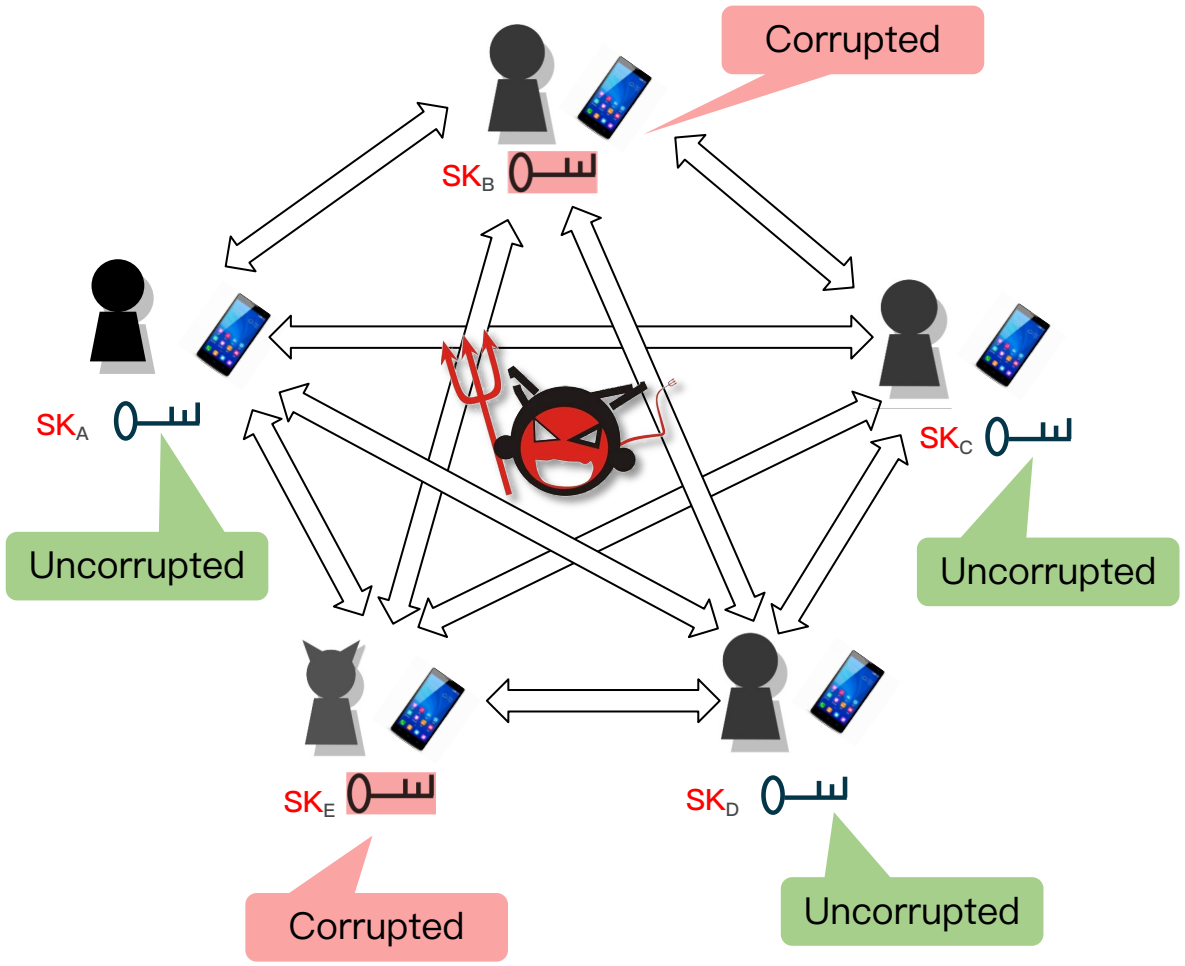
attacking **Scheme** in time $t_{\mathcal{A}}$ with advantage $\epsilon_{\mathcal{A}}$

$$\frac{t_{\mathcal{B}}}{\epsilon_{\mathcal{B}}} \leq \frac{t_{\mathcal{A}}}{\epsilon_{\mathcal{A}}} \cdot \ell$$

(Almost) Tight Security: $\ell = O(1)$ or $O(\lambda)$,

where λ = security parameter

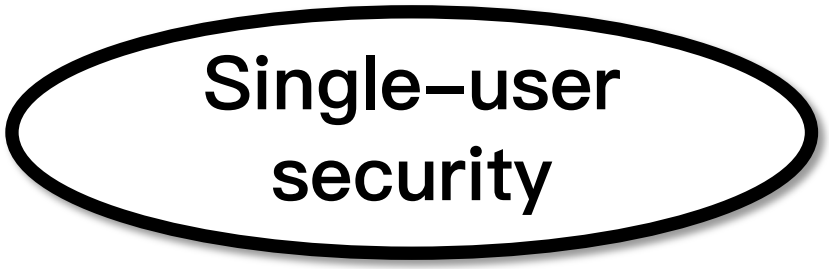
Multi-User Security under Adaptive Corruptions (MU^c Security)



MU^c security: protect the uncorrupted users

Users	Corrupted	Uncorrupted
$\mathcal{A}$'s knowledge about SK	All	Nothing
MU^c security	–	✓ protected

On Achieving Tight MU^c Security



PKE (Public-Key Encryption)	**IND-CPA/CCA security** (Indistinguishability under Chosen-Plaintexts/Ciphertexts Attacks)
SIG (Digital Signature)	**(Strong) EUF-CMA security** ((Strong) Existential Unforgeability under Chosen-Message Attacks)

On Achieving Tight MU^c Security

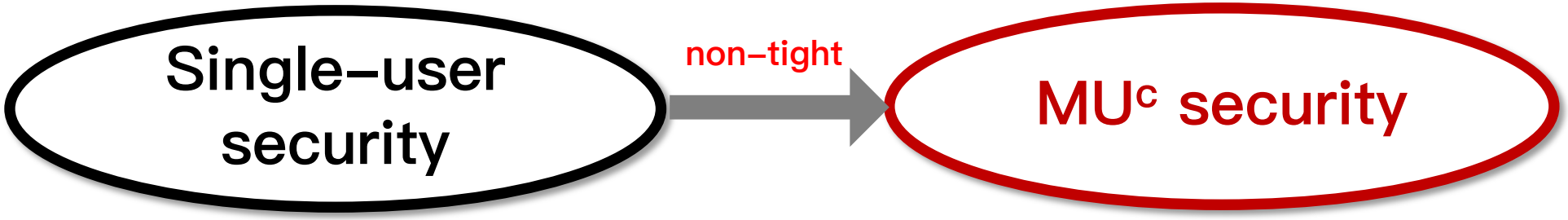
Single-user security

PKE (Public-Key Encryption)	IND-CPA/CCA security (Indistinguishability under Chosen-Plaintexts/Ciphertexts Attacks)
SIG (Digital Signature)	(Strong) EUF-CMA security ((Strong) Existential Unforgeability under Chosen-Message Attacks)

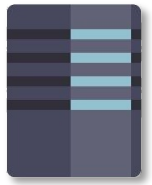
MU^c security

$MUMC^c$-CPA/CCA security (Multi-User and Multi-Challenge IND-CPA/CCA security under adaptive corruptions)
(Strong) MU^c-CMA security (Multi-User (Strong) EUF-CMA security under adaptive corruptions)

On Achieving Tight MU^c Security



PKE (Public-Key Encryption)	IND-CPA/CCA security (Indistinguishability under Chosen-Plaintexts/Ciphertexts Attacks)	non-tight	$MUMC^c$-CPA/CCA security (Multi-User and Multi-Challenge IND-CPA/CCA security under adaptive corruptions)
SIG (Digital Signature)	(Strong) EUF-CMA security ((Strong) Existential Unforgeability under Chosen-Message Attacks)	non-tight	(Strong) MU^c-CMA security (Multi-User (Strong) EUF-CMA security under adaptive corruptions)



Non-tight reduction!

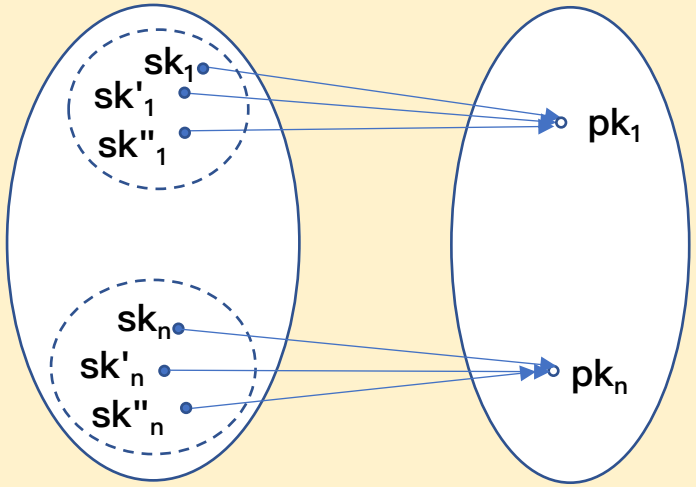
$\ell \geqslant$ #users, #ciphertexts, or #signatures

On Achieving Tight MU^c Security: Impossibility Results

Tight MU^c security

PKE (Public-Key Encryption)	Tight $MUMC^c$-CPA/CCA security
SIG (Digital Signature)	Tight (Strong) MU^c-CMA security

- [Bader–Jager–Li–Schäge, EC16]
  Impossible if the relation (pk, sk) is "unique" or "re-randomizable"



- [Morgan–Pass–Shi, AC20]
  Impossible if the signing algorithm is deterministic

- [Bader–Jager–Li–Schäge, EC16]
  Impossible if the relation (vk, sk) is "unique" or "re-randomizable"

On Achieving Tight MU^c Security: Possibility Results

PKE	Std/RO model?	MU^c Security?	Security Loss	Assumption
[LLP20, DCC]	RO	✓	O(1)	CDH

Only one PKE scheme is proved to be **tightly MUMC^c–CCA** secure, but in the **RO** model.

SIG	Std/RO model?	Strong Security?	MU^c Security?	Security Loss	Assumption
[BHJKL15, TCC]	Std	–	✓	O(1)	MDDH
[GJ18, C]	RO	–	✓	O(1)	DDH
[DGJL21, PKC]	RO	✓	✓	O(1)	DDH/Φ–hiding
[HJKLPRS21, C]	Std	×	✓	O(λ)	MDDH
[PW22, PKC]	RO	–	✓	O(1)	LWE

Only 5 SIG schemes are proved to be **tightly MU^c–CMA** secure.

Only one of them is proved to be **tightly strong MU^c–CMA** secure, but in the **RO** model.

On Achieving Tight MU^c Security: Possibility Results

PKE	Std/RO model?	MU^c Security?	Security Loss	Assumption
[LLP20, DCC]	RO	✓	O(1)	CDH

Only one PKE scheme is proved to be tightly MUMC^c–CCA secure, but in the RO model.

SIG	Std/RO model?	Strong Security?	MU^c Security?	Security Loss	Assumption
[BHJKL15, TCC]	Std	–	✓	O(1)	MDDH
[GJ18, C]	RO	–	✓	O(1)	DDH
[DGJL21, PKC]	RO	✓	✓	O(1)	DDH/Φ–hiding
[HJKLPRS21, C]	Std	×	✓	O(λ)	MDDH
[PW22, PKC]	RO	–	✓	O(1)	LWE

Only 5 SIG schemes are proved to be tightly MU^c–CMA secure.

Only one of them is proved to be tightly strong MU^c–CMA secure, but in the RO model.

Can we achieve (almost) tight MU^c security in the standard model?

Contribution I: Almost Tight MU^c Security in the Standard Model

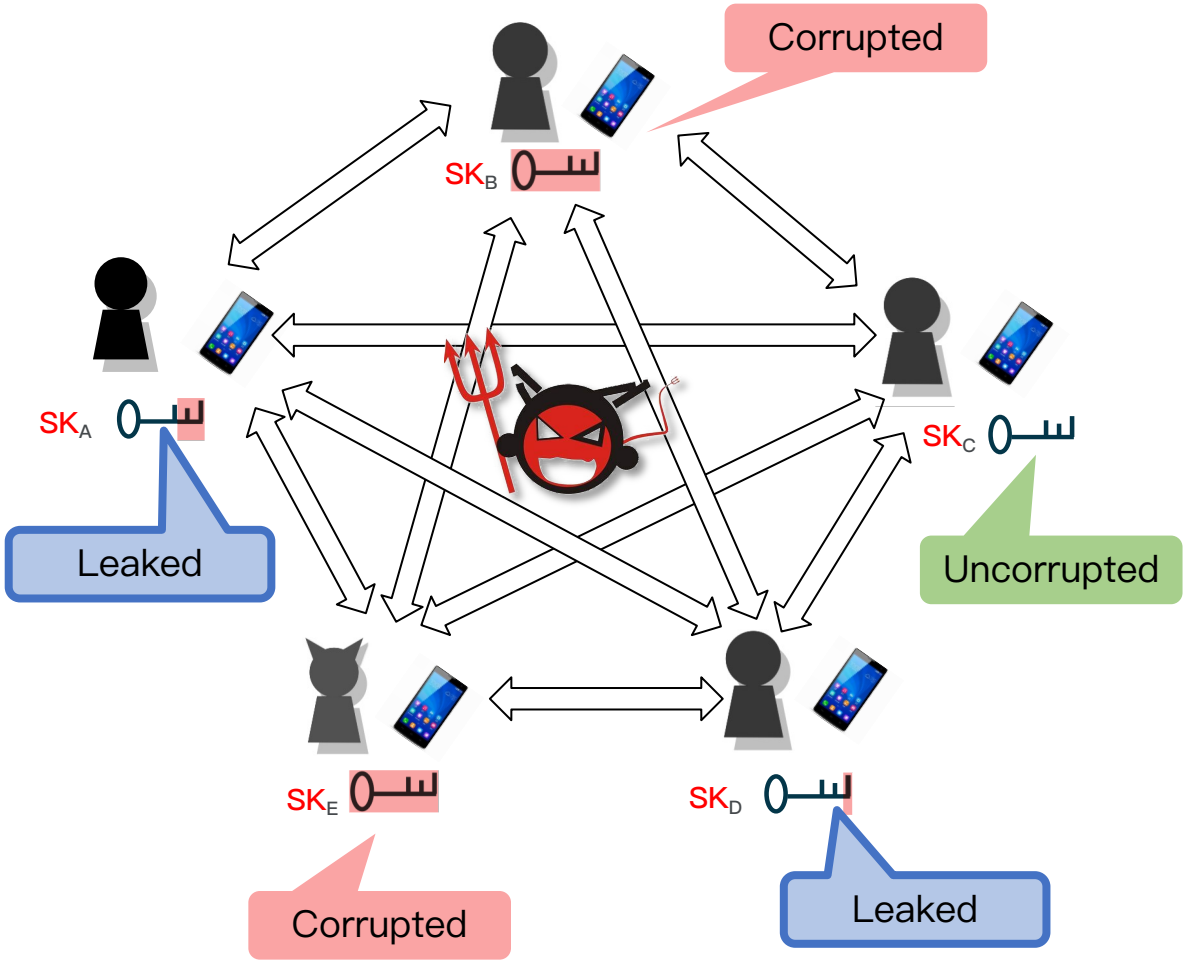
PKE	Std/RO model?	MU^c Security?	Security Loss	Assumption
[LLP20, DCC]	RO	✓	O(1)	CDH
Ours	**Std**	✓	**O(log λ)**	**MDDH** (SXDH, k–LIN)

- The *first* PKE scheme with **almost tight MUMC^c–CCA** security in the standard model

SIG	Std/RO model?	Strong Security?	MU^c Security?	Security Loss	Assumption
[BHJKL15, TCC]	Std	–	✓	O(1)	MDDH
[GJ18, C]	RO	–	✓	O(1)	DDH
[DGJL21, PKC]	RO	✓	✓	O(1)	DDH/Φ–hiding
[HJKLPRS21, C]	Std	×	✓	O(λ)	MDDH
[PW22, PKC]	RO	–	✓	O(1)	LWE
Ours	**Std**	✓	✓	**O(log λ)**	**MDDH** (SXDH, k–LIN)

- The *first* SIG scheme with **almost tight strong MU^c–CMA** security in the standard model

Multi-User Security under Adaptive Corruptions & Leakages ($MU^{c\&l}$ Security)



MU^{c} security: protect the uncorrupted users

Users	Corrupted	Leaked	Uncorrupted
$\mathcal{A}$'s knowledge about SK	All	Part	Nothing
MU^{c} security	–	unprotected ✗	protected ✓
$MU^{c\&l}$ security	–	protected ✓	protected ✓

$MU^{c\&l}$ security: protect the uncorrupted users & the users whose SKs are partially leaked.

Contribution II: Almost Tight $MU^{c\&I}$ Security

PKE	Std/RO model?	MU^{c} Security?	$MU^{c\&I}$ Security?	Security Loss	Assumption
[LLP20, DCC]	RO	✓	–	O(1)	CDH
Ours	**Std**	✓	✓ (1/3 – o(1))	**O(log λ)**	**MDDH** (SXDH, k–LIN)

- The *first* PKE scheme with **almost tight** $MUMC^{c\&I}$–CCA security (no matter in the standard model or in the RO model)

SIG	Std/RO model?	Strong Security?	MU^{c} Security?	$MU^{c\&I}$ Security?	Security Loss	Assumption
[BHJKL15, TCC]	Std	–	✓	–	O(1)	MDDH
[GJ18, C]	RO	–	✓	–	O(1)	DDH
[DGJL21, PKC]	RO	✓	✓	–	O(1)	DDH/Φ–hiding
[HJKLPRS21, C]	Std	×	✓	–	O(λ)	MDDH
[PW22, PKC]	RO	–	✓	–	O(1)	LWE
Ours	**Std**	✓	✓	✓ (1/6 – o(1))	**O(log λ)**	**MDDH** (SXDH, k–LIN)

- The *first* SIG scheme with **almost tight strong** $MU^{c\&I}$–CMA security (no matter in the standard or RO model)

Contribution II: Almost Tight $MU^{c\&I}$ Security & Full Compactness

PKE	Std/RO model?	MU^{c} Security?	$MU^{c\&I}$ Security?	Security Loss	Assumption	Fully compact?
[LLP20, DCC]	RO	✓	–	O(1)	CDH	✓
Ours	**Std**	✓	✓ (1/3 – o(1))	**O(log λ)**	**MDDH** (SXDH, k–LIN)	✓

SIG	Std/RO model?	Strong Security?	MU^{c} Security?	$MU^{c\&I}$ Security?	Security Loss	Assumption	Fully compact?
[BHJKL15, TCC]	Std	–	✓	–	O(1)	MDDH	×
[GJ18, C]	RO	–	✓	–	O(1)	DDH	✓
[DGJL21, PKC]	RO	✓	✓	–	O(1)	DDH/Φ–hiding	✓
[HJKLPRS21, C]	Std	×	✓	–	O(λ)	MDDH	×
[PW22, PKC]	RO	–	✓	–	O(1)	LWE	×
Ours	**Std**	✓	✓	✓ (1/6 – o(1))	**O(log λ)**	**MDDH** (SXDH, k–LIN)	✓

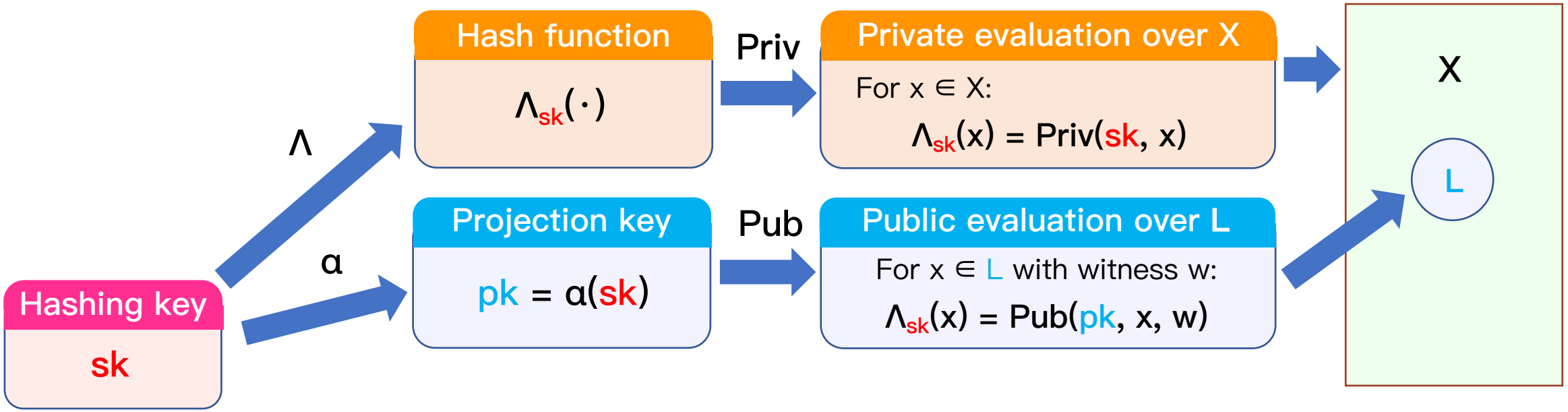
All our schemes are **fully compact**!

(Namely, all the parameters, keys, signatures, ciphertexts consist of only a constant number of group elements.)

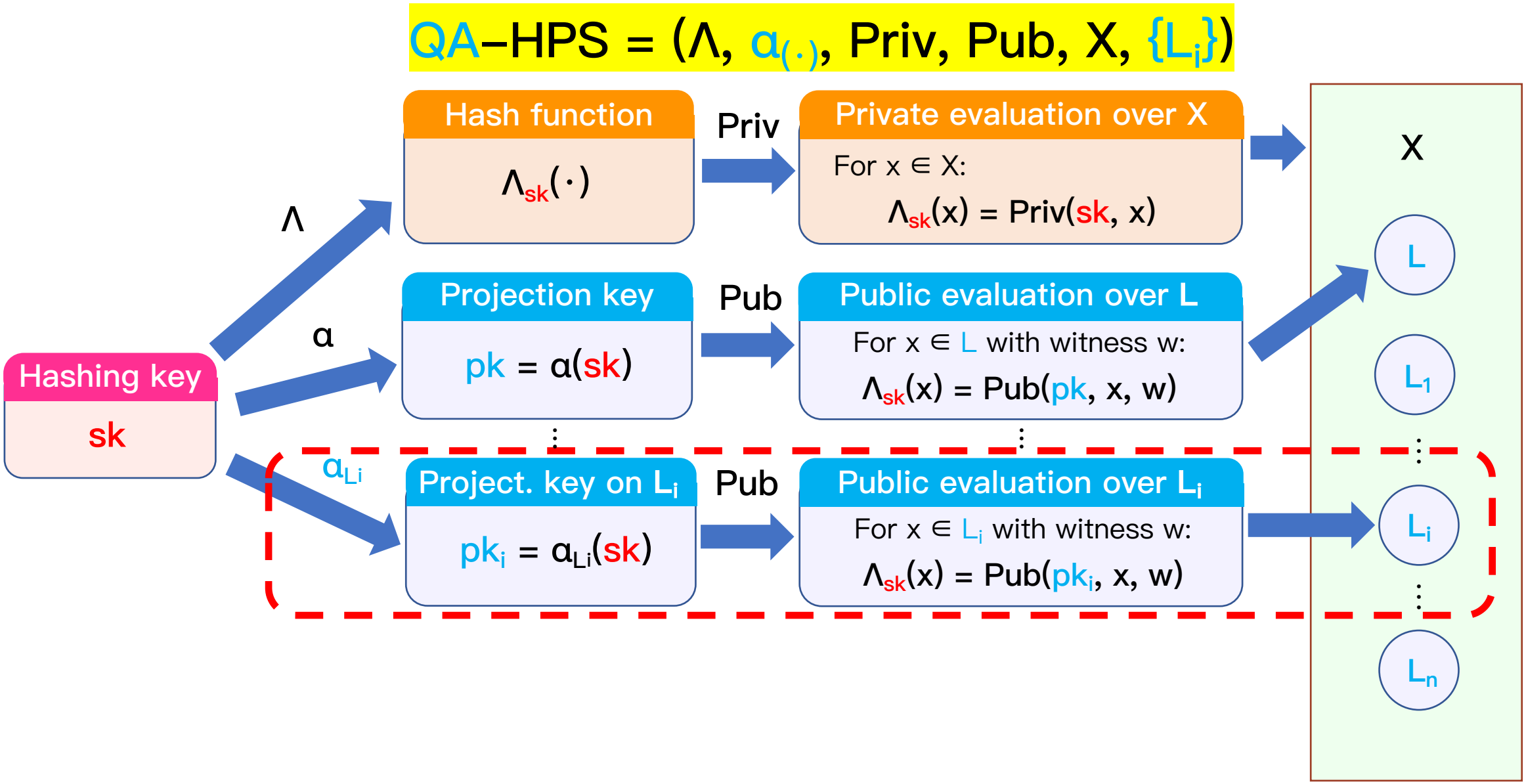
Contents

Recap: Hash Proof System [Cramer–Shoup, EC02]

HPS = (Λ, α, Priv, Pub, X, L)

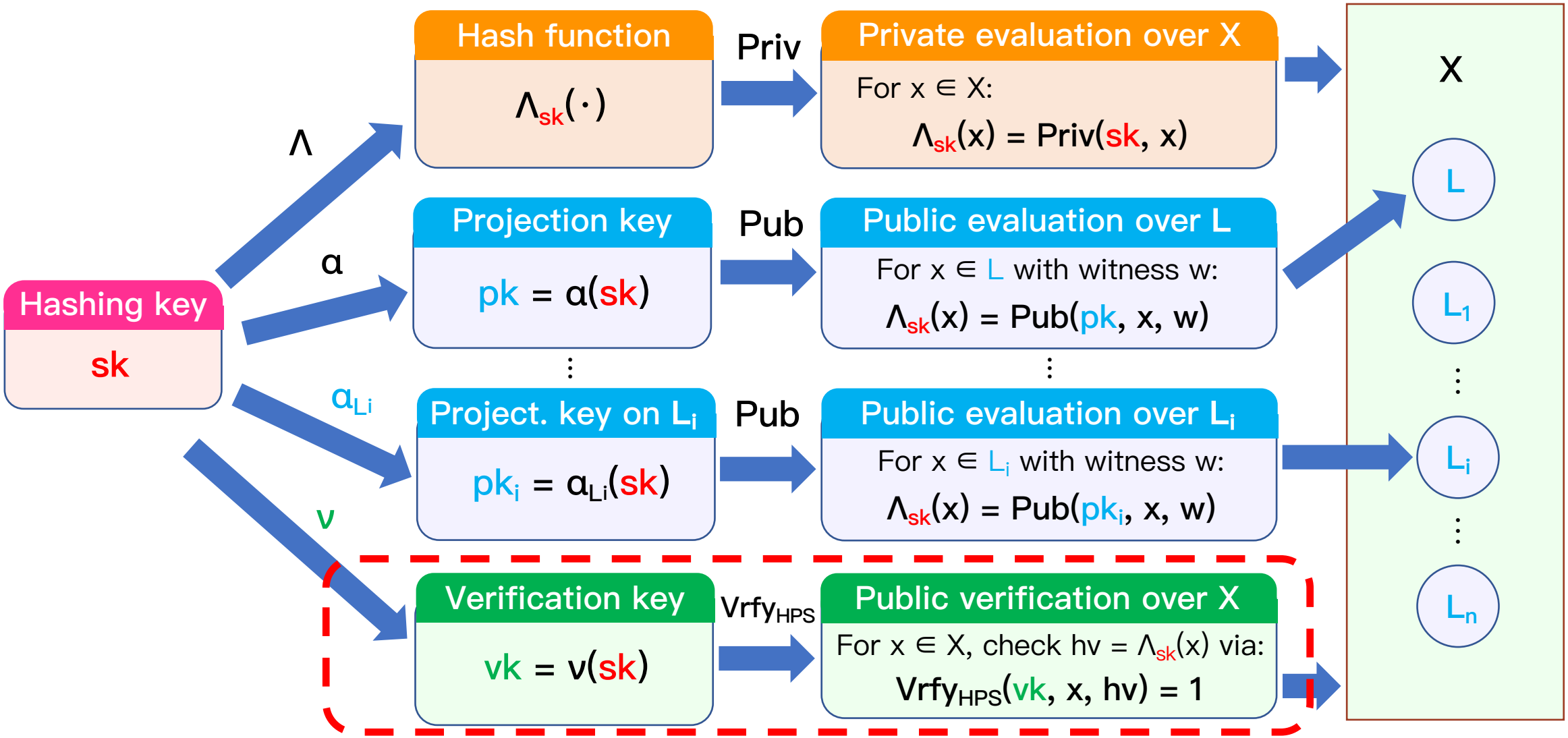


Recap: Quasi–Adaptive HPS [Han–Liu–Lyu–Gu, C19]



Our New Tool: Publicly-Verifiable QA-HPS

$$\text{PV-QA-HPS} = (\Lambda, \alpha_{(\cdot)}, \nu, \text{Priv}, \text{Pub}, \text{Vrfy}_{\text{HPS}}, X, \{L_i\})$$



Properties of PV–QA–HPS (I)

- **Verification Completeness:**

  For $\mathbf{x} \in X$ and honestly generated $\mathbf{hv} = \Lambda_{sk}(\mathbf{x})$, it holds $\mathbf{Vrfy}_{HPS}(vk, \mathbf{x}, \mathbf{hv}) = 1$.

  (Honestly computed hash values always pass the verification.)

- **Verification Soundness:**

  Given sk and $vk = \nu(sk)$, it is computationally hard to find $\mathbf{x} \in X$ and $\mathbf{hv}$, such that $\mathbf{hv} \neq \Lambda_{sk}(\mathbf{x})$ but $\mathbf{Vrfy}_{HPS}(vk, \mathbf{x}, \mathbf{hv}) = 1$.

  (Hard to find an incorrect hash value to pass the verification.)

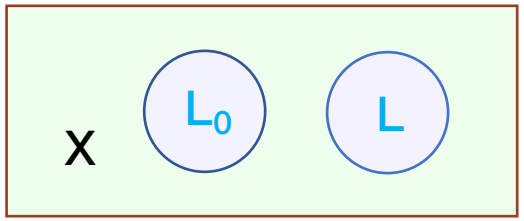
Properties of PV–QA–HPS (II)

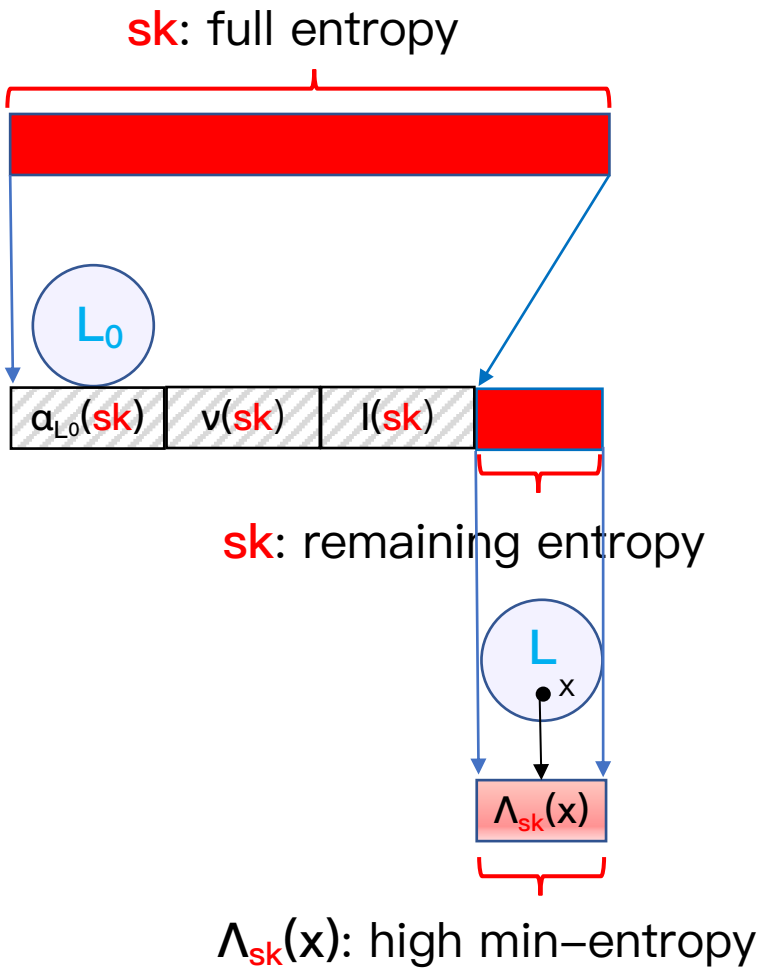
- **Leakage–Resilient (LR)**
  <L_0, L>–One–Time (OT)–Extracting:

  Conditioned on

  - $pk_0 = \alpha_{L_0}(sk)$,
  - $vk = \nu(sk)$,
  - bounded leakage information $l(sk)$,

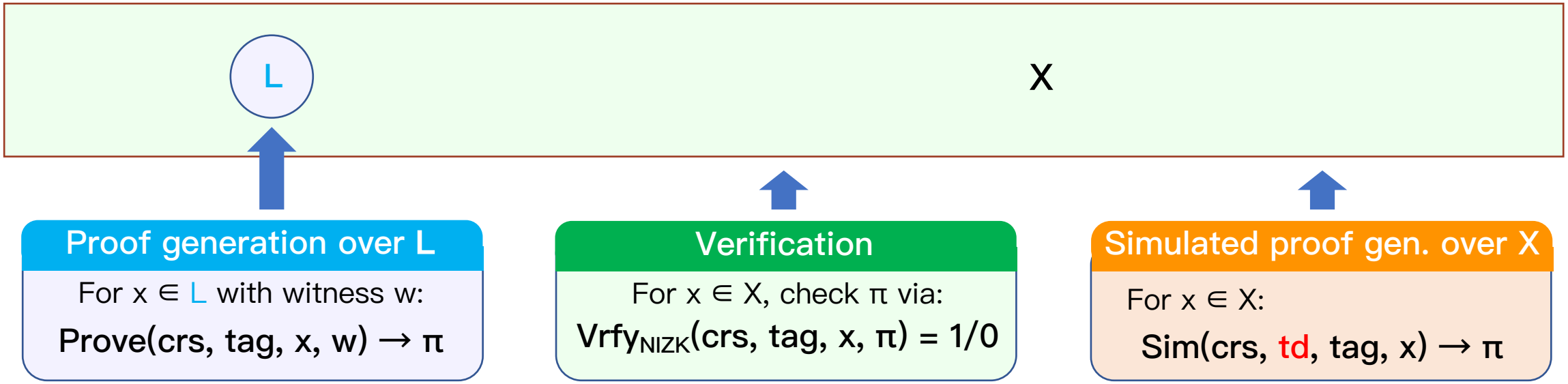
  for any $x \in L$, $\Lambda_{sk}(x)$ has *high min-entropy*.





Recap: Quasi-Adaptive NIZK [Jutla-Roy, AC13]

QA-NIZK = (Prove, $\text{Vrfy}_{\text{NIZK}}$, Sim): tag-based



- **Perfect Zero-Knowledge (ZK):**

  Prove ≡ Sim over L.

- **Unbounded Simulation-Soundness (USS):**

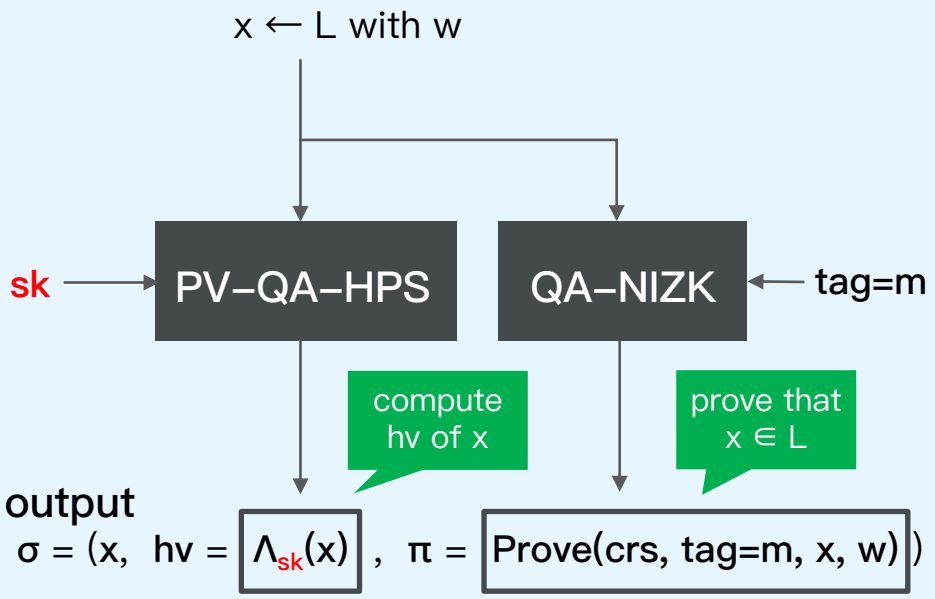
  Hard to prove a false x ∉ L, even given many simulated proofs.

Contents

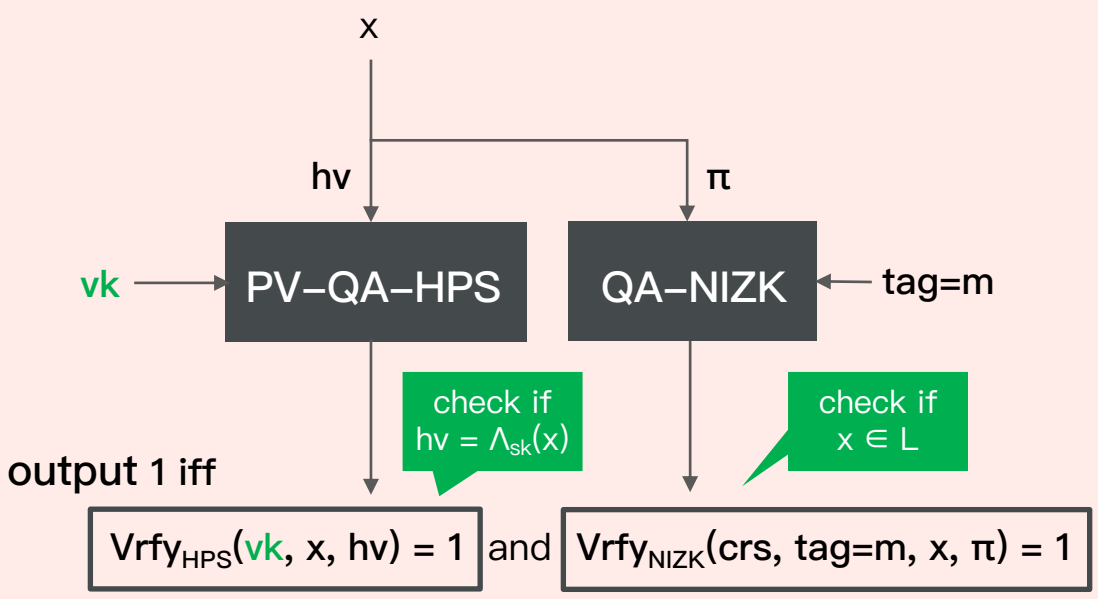
Our SIG from PV–QA–HPS and QA–NIZK

Gen → (vk = v(sk), sk) : Verification key and Hashing key of PV–QA–HPS

Sign(sk, m):



Vrfy(vk, m, σ = (x, hv, π)):



Tight Strong MU$^{c\&I}$–CMA Security Proof of Our SIG

Verification Keys:

$\{ vk_i = v(sk_i) \}_{i \in [n]}$

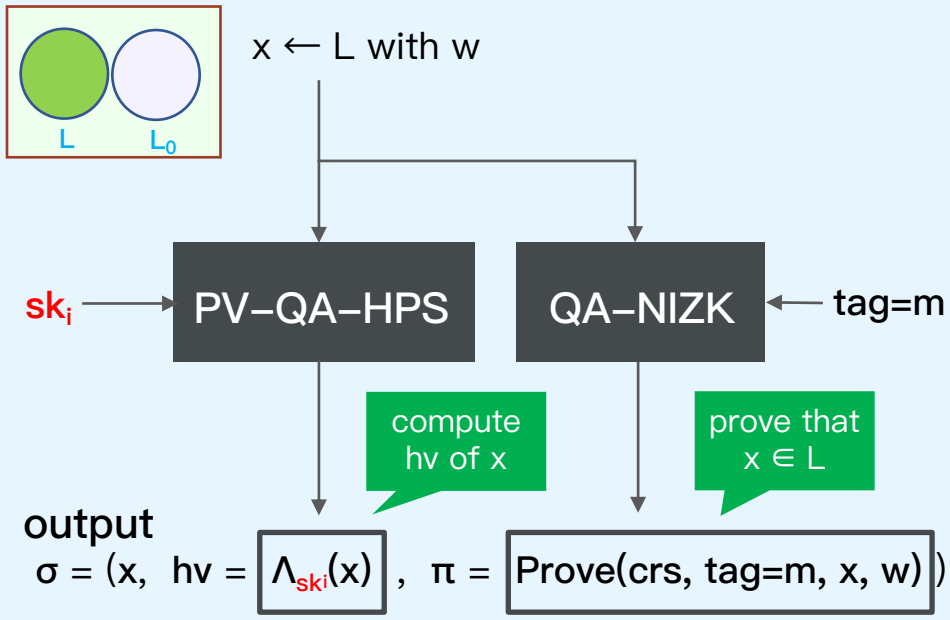
Corruption Queries (i):

$\{ sk_i \}_{i \in Q_{cor}}$

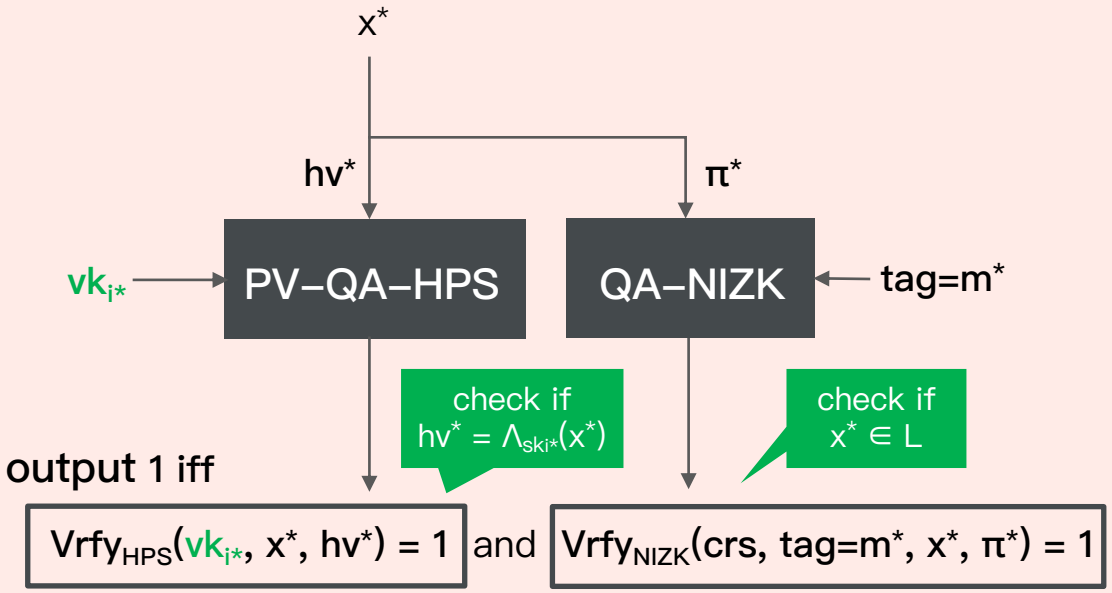
Leakage Queries (i, l):

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

Signing Queries (i, m):



Forgery ($i^*, m^*, \sigma^* = (x^*, hv^*, \pi^*)$): $i^* \notin Q_{cor}$



Step 1: Switch Language from L to L_0 for Signing Queries

Verification Keys:

$\{ vk_i = v(sk_i) \}_{i \in [n]}$

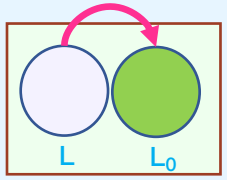
Corruption Queries (i):

$\{ sk_i \}_{i \in Q_{cor}}$

Leakage Queries (i, l):

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

Signing Queries (i, m):



$x \leftarrow L_0$

Subset Membership Problem (SMP)

sk_i → PV–QA–HPS

QA–NIZK ← td, tag=m

Perfect ZK of QA–NIZK

output $\sigma = (x, hv = \Lambda_{sk_i}(x), \pi = Sim(crs, td, tag=m, x))$

Forgery ($i^*, m^*, \sigma^* = (x^*, hv^*, \pi^*)$): $i^* \notin Q_{cor}$

x^*

hv^* → PV–QA–HPS ← vk_{i^*}

π^* → QA–NIZK ← tag=m^*

output 1 iff

$Vrfy_{HPS}(vk_{i^*}, x^*, hv^*) = 1$ and $Vrfy_{NIZK}(crs, tag=m^*, x^*, \pi^*) = 1$

Step 1: Switch Language from L to L_0 for Signing Queries

Verification Keys:

$\{ vk_i = v(sk_i) \}_{i \in [n]}$

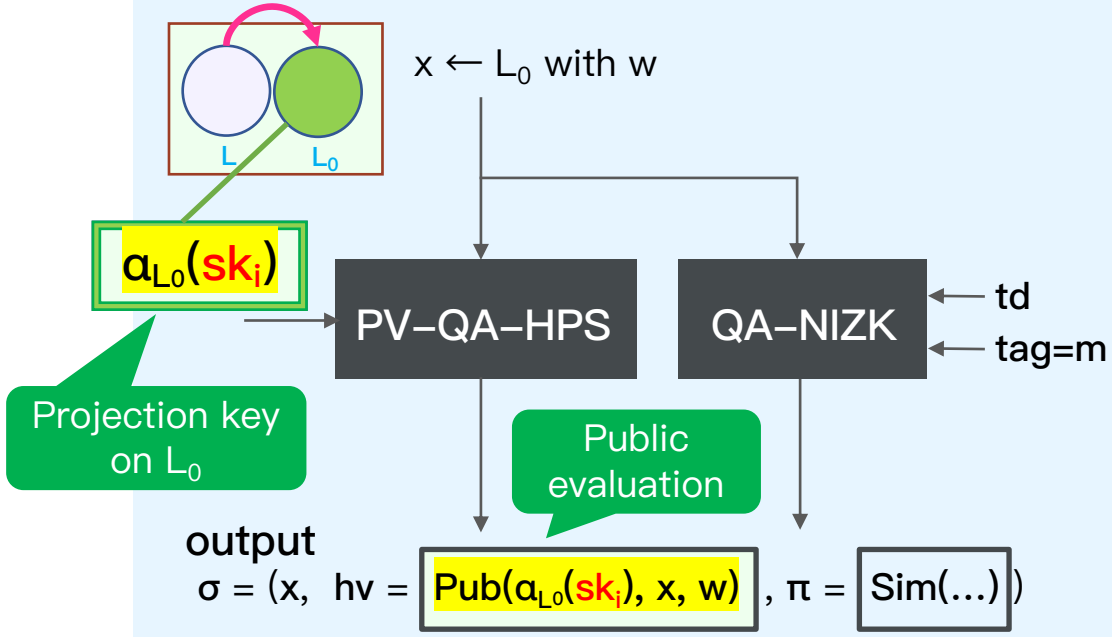
Corruption Queries (i):

$\{ sk_i \}_{i \in Q_{cor}}$

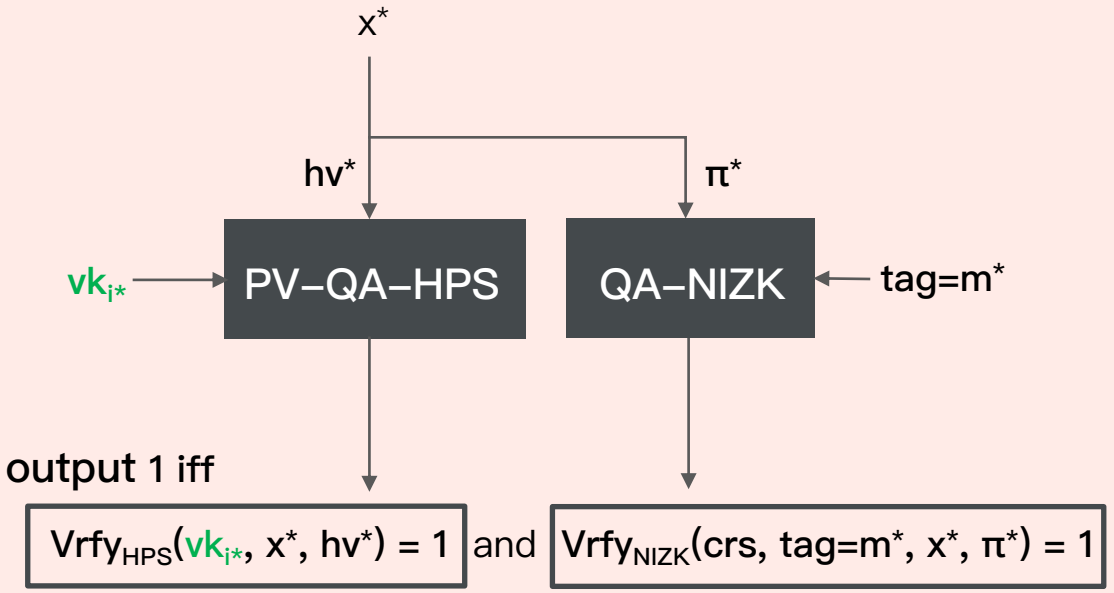
Leakage Queries (i, l):

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

Signing Queries (i, m):



Forgery ($i^*, m^*, \sigma^* = (x^*, hv^*, \pi^*)$): $i^* \notin Q_{cor}$



Step 2: Restrict Language from X to L for Forgery

Verification Keys:

$\{ vk_i = v(sk_i) \}_{i \in [n]}$

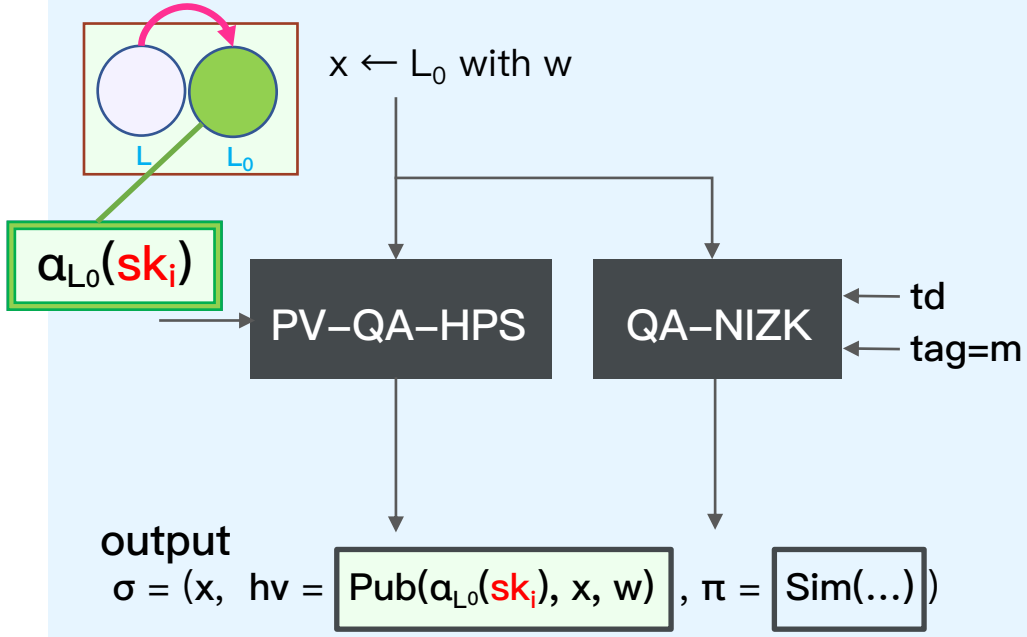
Corruption Queries (i):

$\{ sk_i \}_{i \in Q_{cor}}$

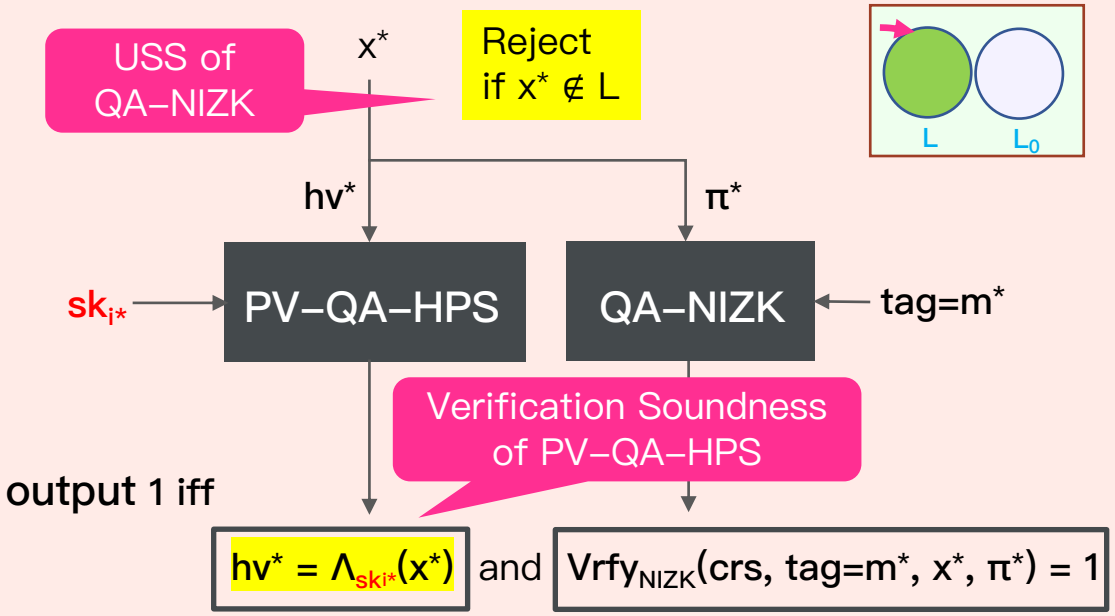
Leakage Queries (i, l):

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

Signing Queries (i, m):



Forgery ($i^*, m^*, \sigma^* = (x^*, hv^*, \pi^*)$): $i^* \notin Q_{cor}$



Step 2: Restrict Language from X to L for Forgery

Verification Keys:

$\{ vk_i = v(sk_i) \}_{i \in [n]}$

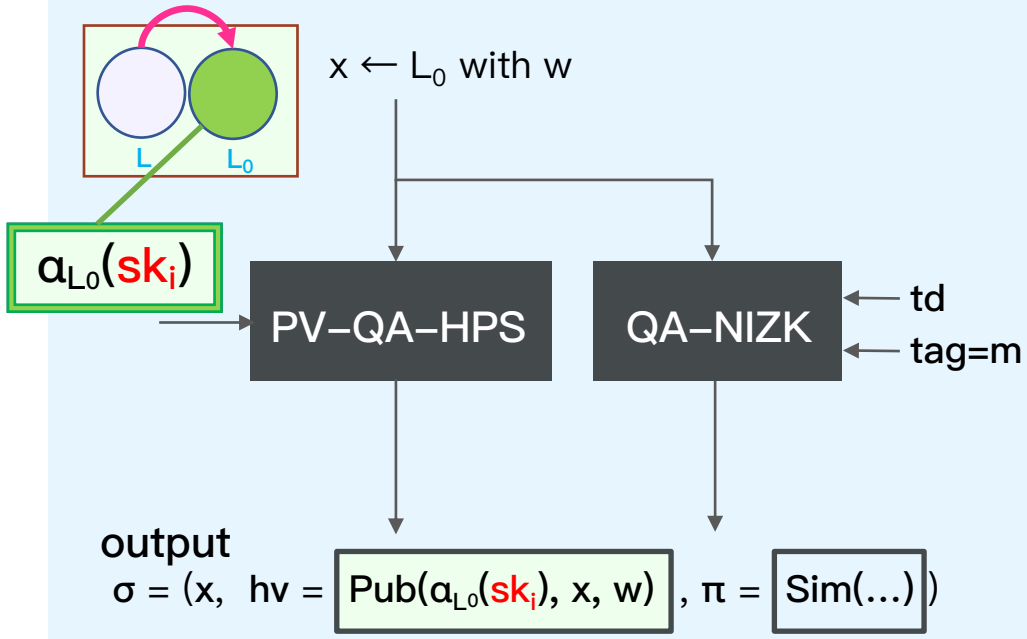
Corruption Queries (i):

$\{ sk_i \}_{i \in Q_{cor}}$

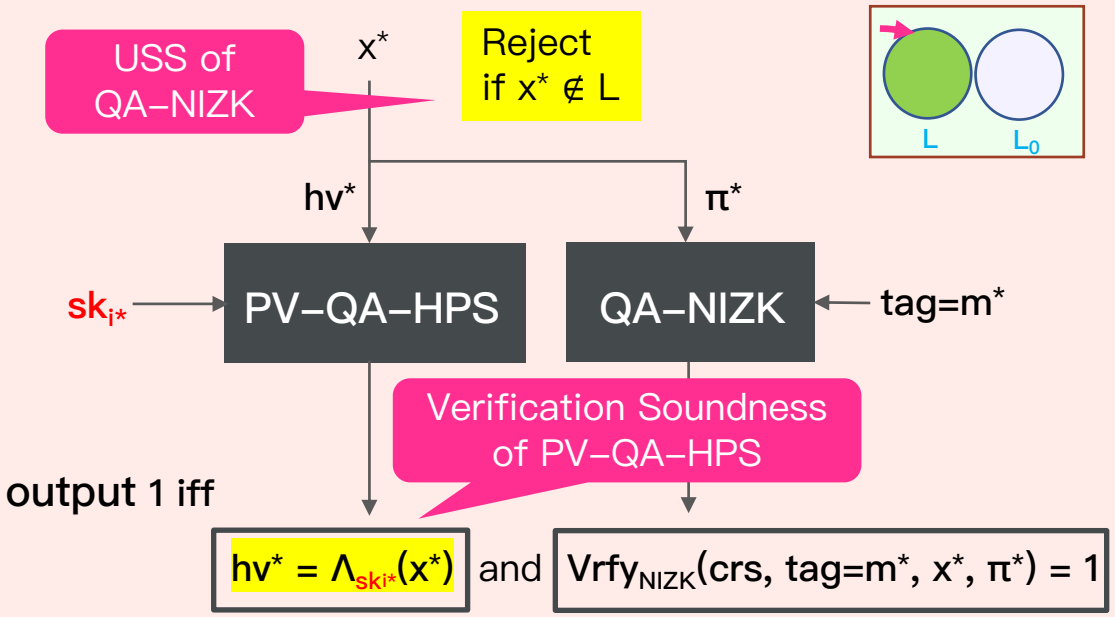
Leakage Queries (i, l):

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

Signing Queries (i, m):



Forgery (i*, m*, σ* = (x*, hv*, π*)): $i^* \notin Q_{cor}$



All reductions have all signing keys to handle adaptive Corruption & Leakage queries.

Now $\mathcal{A}$'s forgery hardly succeeds

Verification Keys:

$\{ vk_i = v(sk_i) \}_{i \in [n]}$

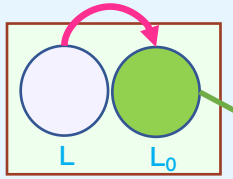
Corruption Queries (i):

$\{ sk_i \}_{i \in Q_{cor}}$

Leakage Queries (i, l):

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

Signing Queries (i, m):



$x \leftarrow L_0$ with w

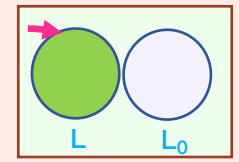
$\alpha_{L_0}(sk_i)$

output

$\sigma = (x,\ hv = \boxed{Pub(\alpha_{L_0}(sk_i), x, w)},\ \pi = \boxed{Sim(\ldots)})$

Forgery ($i^*, m^*, \sigma^* = (x^*, hv^*, \pi^*)$): $i^* \notin Q_{cor}$

Reject if $x^* \notin L$



output 1 iff

$\boxed{hv^* = \Lambda_{sk_{i^*}}(x^*)}$ and $\boxed{Vrfy_{NIZK}(crs, tag=m^*, x^*, \pi^*) = 1}$

$\mathcal{A}$'s knowledge about sk_{i^*} in its forgery

- $vk_{i^*} = v(sk_{i^*})$ in the verification key,
- $l(sk_{i^*})$ from leakage queries,
- $pk_{0,i^*} = \boxed{\alpha_{L_0}(sk_{i^*})}$ from signing queries.

LR–<L_0, L>–OT–Extracting of PV–QA–HPS

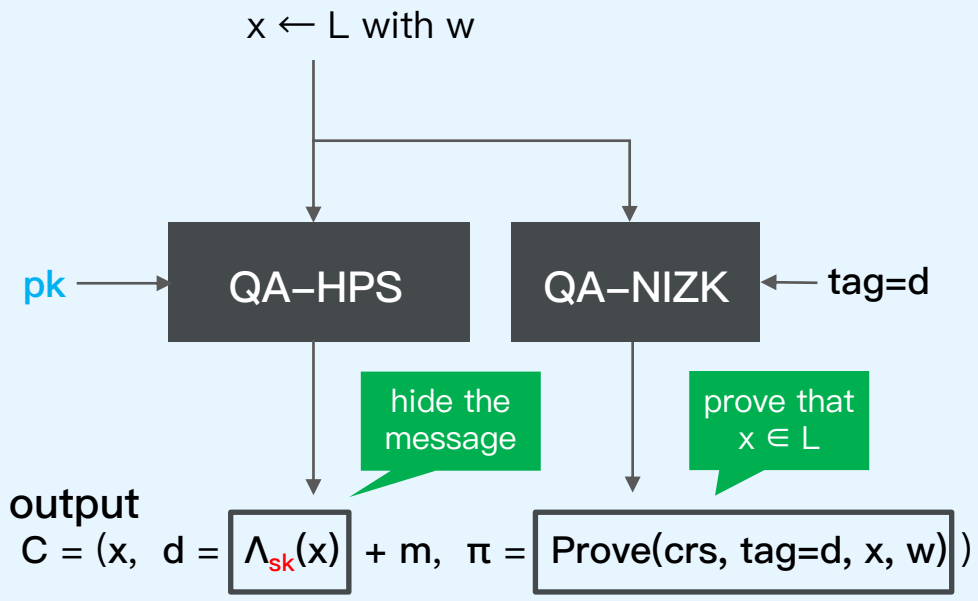
$\mathcal{A}$'s forgery fails since

- For any $x^* \in L$, $\Lambda_{sk_{i^*}}(x^*)$ has *high min-entropy*.

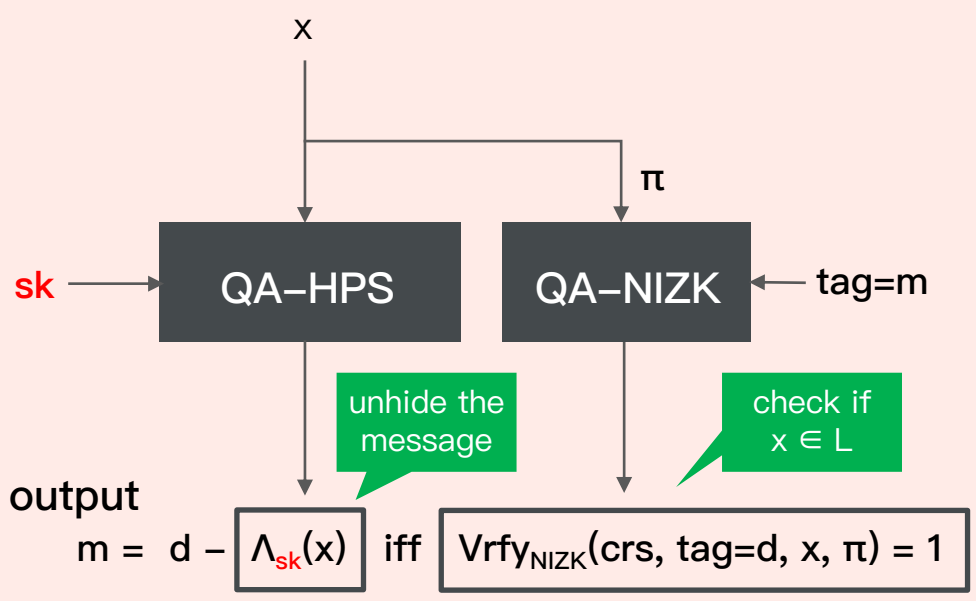
Our PKE from QA–HPS with New Properties and QA–NIZK

Gen → (pk = α_L(sk), sk): Projection key on L and Hashing key of QA–HPS

Enc(pk, m):



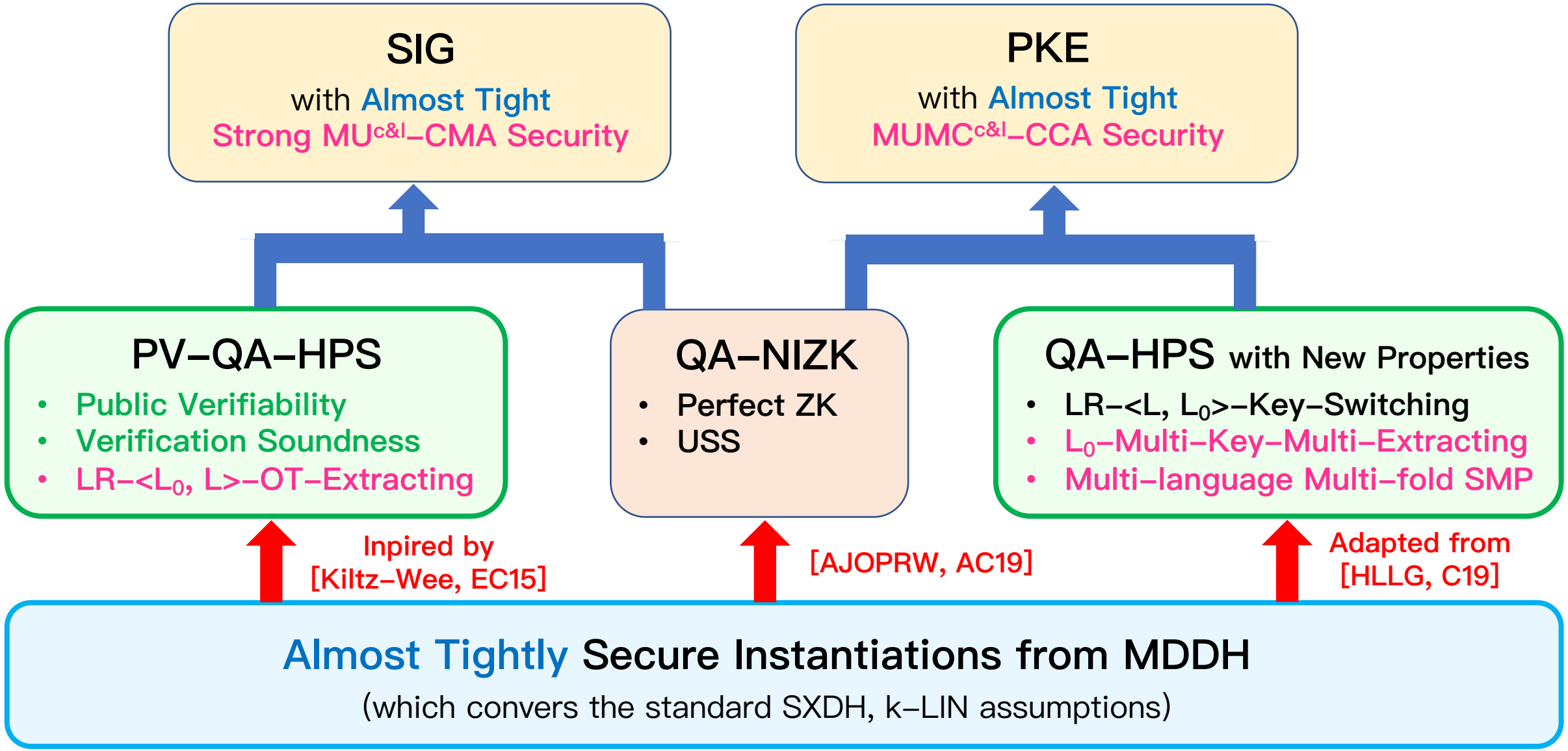
Dec(sk, m, C = (x, d, π)):



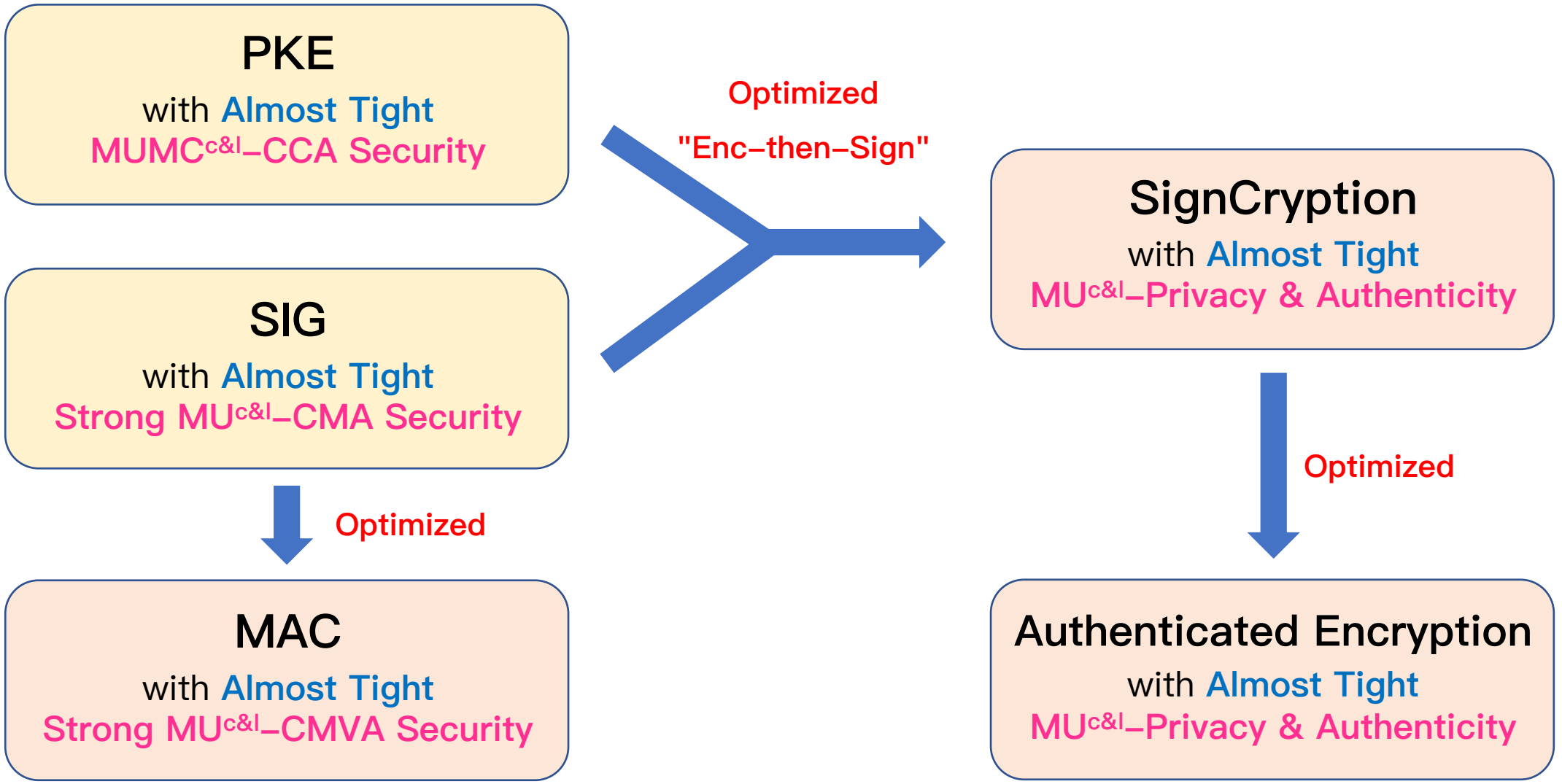
A similar design with our SIG, but quite different tight proofs (see ePrint: ia.cr/2023/153/).

Contents

Overview and Instantiations



Contribution III: More Primitives with Almost Tight MU$^{c\&l}$ Security



Conclusion

- The first SIG, PKE, SC, MAC and AE schemes
  - ✓ with almost tight MU^{c} security in the standard model,
  - ✓ with almost tight $\mathrm{MU}^{c\&l}$ security.
- Generic constructions of SIG and PKE by using
  - New technical tool: Publicly-Verifiable QA-HPS.
- Fully compact instantiations from MDDH over pairing groups.

Thanks! Questions?

ePrint: ia.cr/2023/153