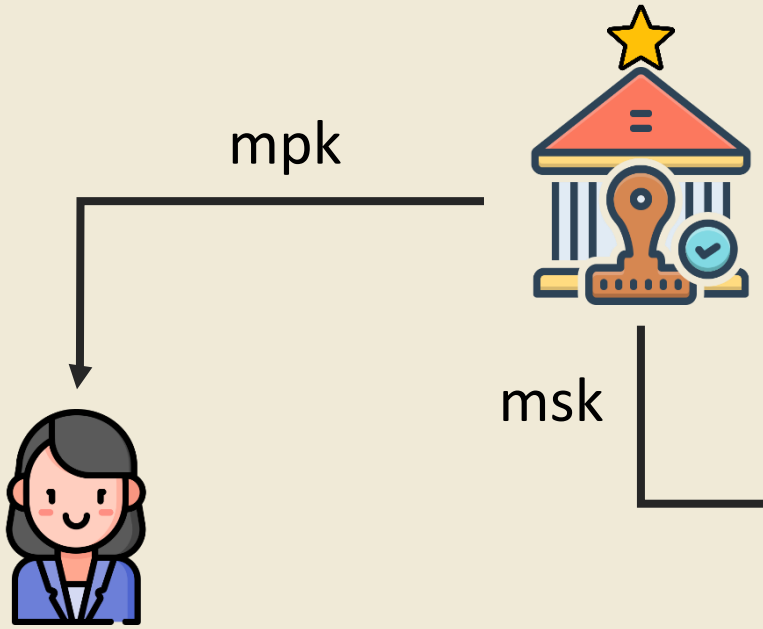
Predicate Encryption from Lattices: Enhanced Compactness and Refined Functionality

Yuejun Wang **Baocang Wang** **Qiqi Lai** **Huaxiong Wang**

Predicate Encryption / PE



mpk

Setup(1^λ) ⟶ (mpk, msk)

msk

KeyGen(msk, f) ⟶ sk

Enc(mpk, x, m) ⟶ ct

ct

Dec(sk, ct) ⟶ m or ⊥

Correctness

Predicate Encryption / PE

mpk

msk

$\text{Setup}(1^\lambda) \rightarrow (\text{mpk}, \text{msk})$

$\text{KeyGen}(\text{msk}, f) \rightarrow \text{sk}$

$\text{Enc}(\text{mpk}, x, m) \rightarrow \text{ct}$

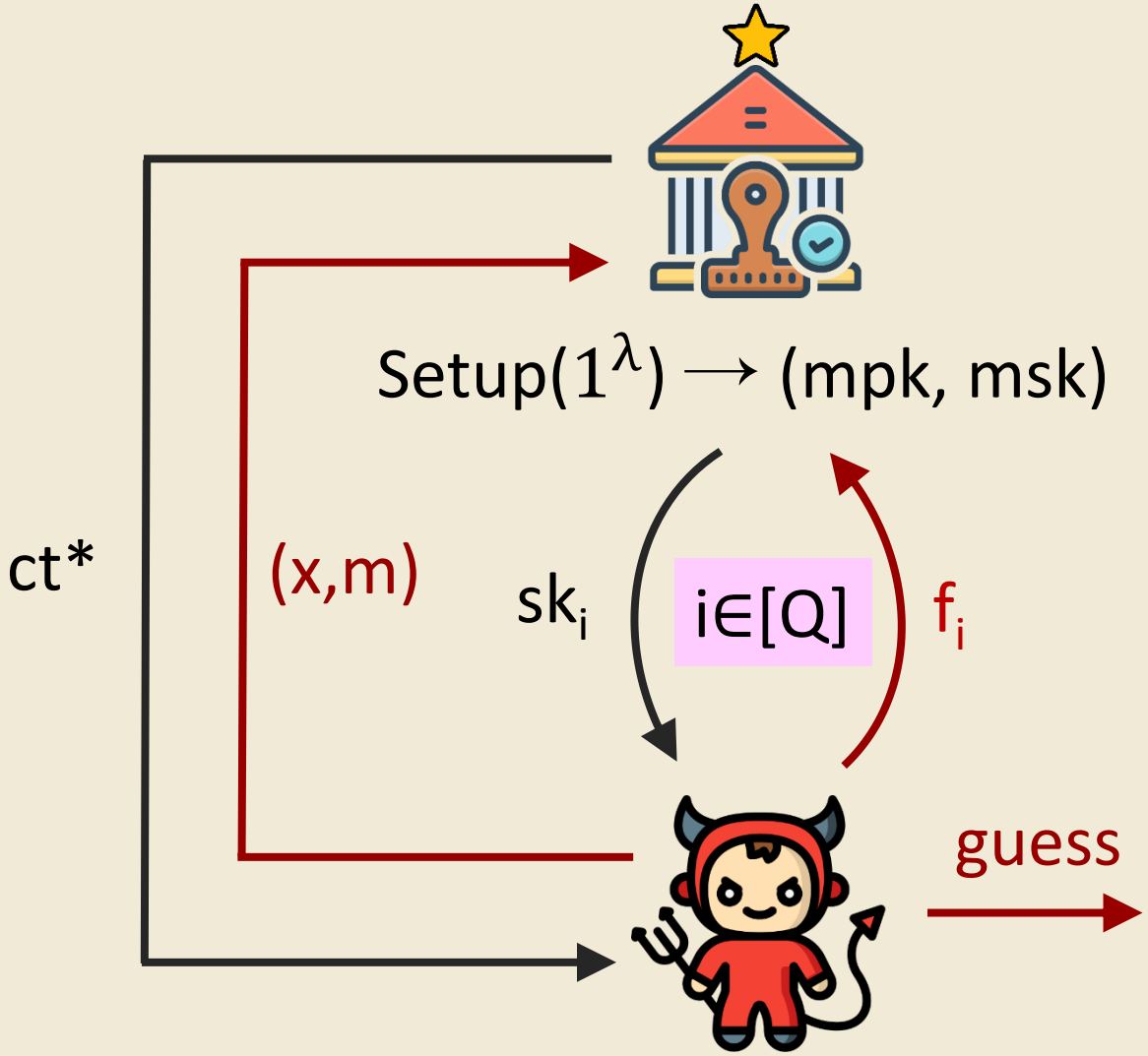
ct

Correctness:

$\text{Dec}(\text{sk}, \text{ct}) \rightarrow m$ if $f(x) = \text{true}$

1-key

Security
Predicate Encryption / PE



Security:

✓ Message m is hidden from Adversary

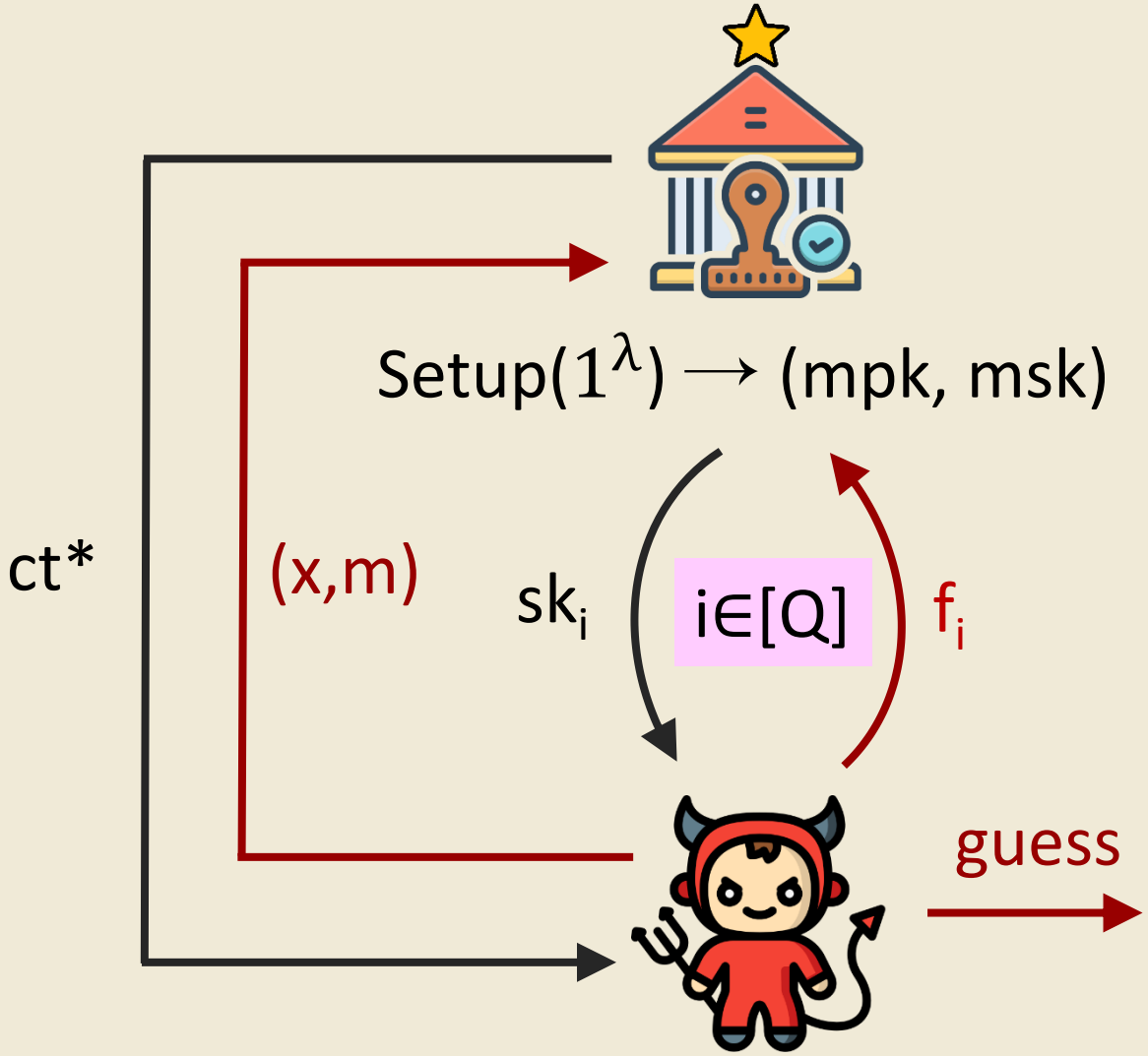
If all i∈[Q], $f_i(x)$=false (0-key)

✓ Attribute x is hidden from Adversary

If all i∈[Q], $f_i(x)$=false (0-key)

Weak Attribute hiding

Security
Predicate Encryption / PE



Security:

✓ Message m is hidden from Adversary

If all i∈[Q], $f_i(x)$=false (0-key)

✓ Attribute x is hidden from Adversary

Even ∃ i∈[Q], s.t. $f_i(x)$=true (0 & 1-key)

Fully Attribute hiding

Related Works

Predicate Encryption / PE

Lattice-based Fully Attribute-hiding Bounded Collusion PE

	(1-key, 0-key)	(Pre,Post)-Challenge	Ciphertext Query dependence
[Agr17]	(Q, poly)	(√, ×)	$+Q^2$
[Agr17]+[AV19]	(Q, poly)	(√, ×)	×Q
[LLW21]	(Q, poly)	(√, √)	+Q

Related Works

Predicate Encryption / PE

Lattice-based Fully Attribute-hiding Bounded Collusion PE

	(1-key, 0-key)	(Pre,Post)-Challenge	Ciphertext Query dependence
[Agr17]	(Q, poly)	(√, ×)	$+Q^2$
[Agr17]+[AV19]	(Q, poly)	(√, ×)	×Q
[LLW21]	(Q, poly)	(√, √)	+Q
Ours	(Q, poly)	(√, √)	+Q

Optimal additionally linear blow-up

Further compress both fixed overhead and per-unit expansion

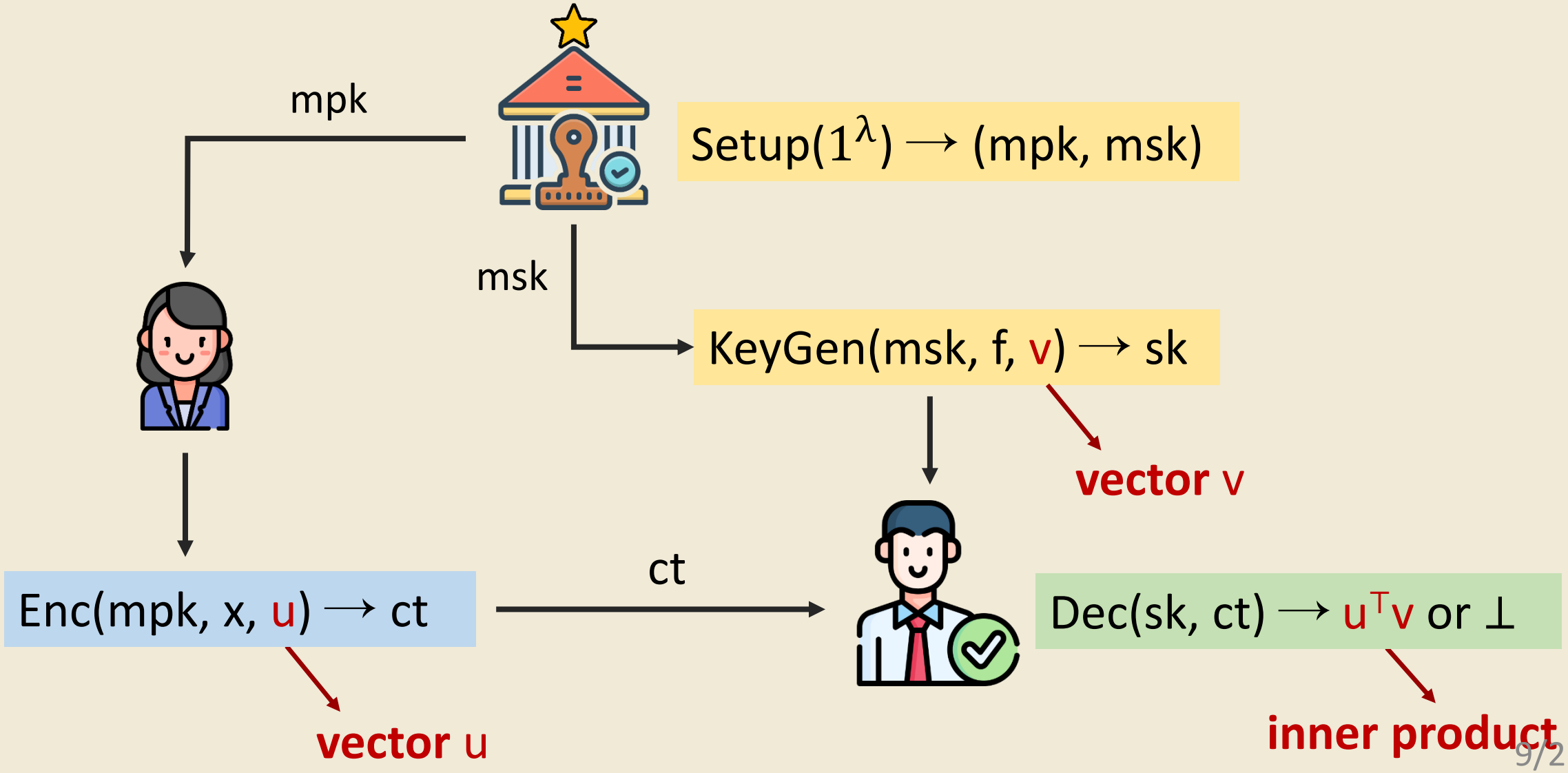
Predicate Encryption

"All or nothing"

Fine-grained Access Control over Encrypted Data

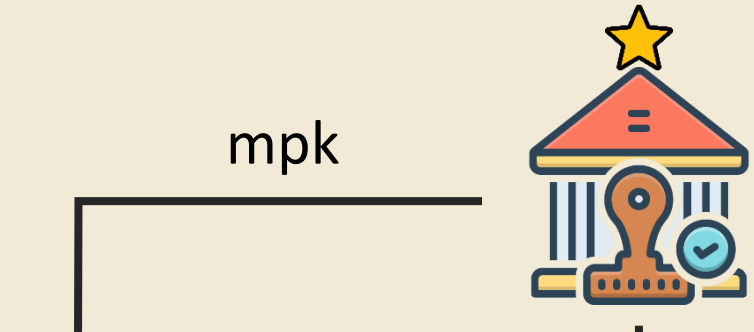
Predicate **Inner Product Functional Encryption**

Predicate **Inner Product Functional Encryption** / P-IPFE



Correctness

Predicate Inner Product Functional Encryption / P-IPFE



mpk

msk

$\mathsf{Setup}(1^\lambda) \longrightarrow (\mathsf{mpk}, \mathsf{msk})$

$\mathsf{KeyGen}(\mathsf{msk}, f, v) \longrightarrow \mathsf{sk}$

$\mathsf{Enc}(\mathsf{mpk}, x, u) \longrightarrow \mathsf{ct}$

ct

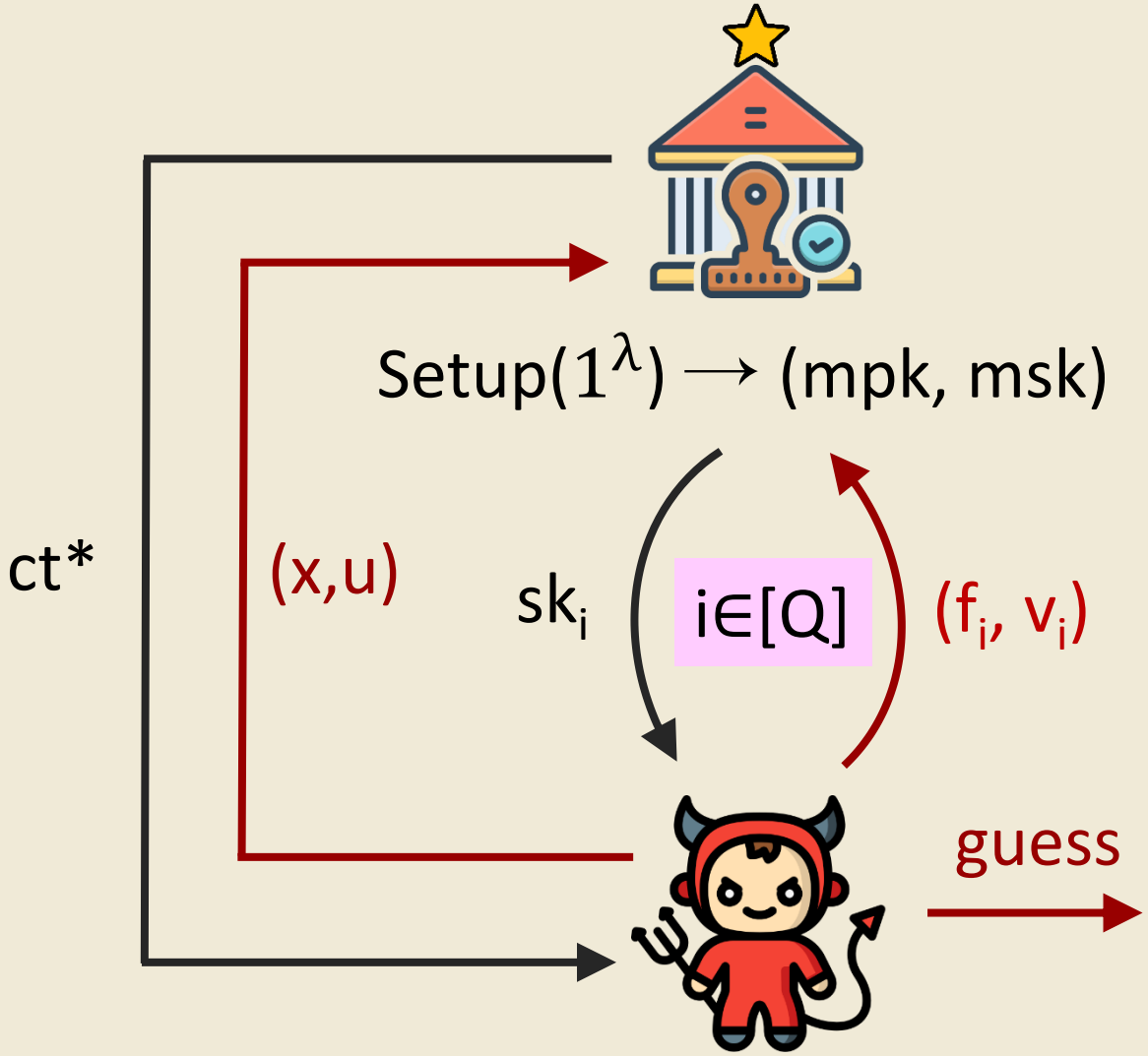
Correctness:

$\mathsf{Dec}(\mathsf{sk}, \mathsf{ct}) \longrightarrow u^\top v$ if $f(x) = \mathsf{true}$

1-key

Security

Predicate Inner Product Functional Encryption / P-IPFE



Security:

- ✓ **Message u** is hidden from Adversary

  If all i∈[Q], $f_i(x)$=false (0-key)

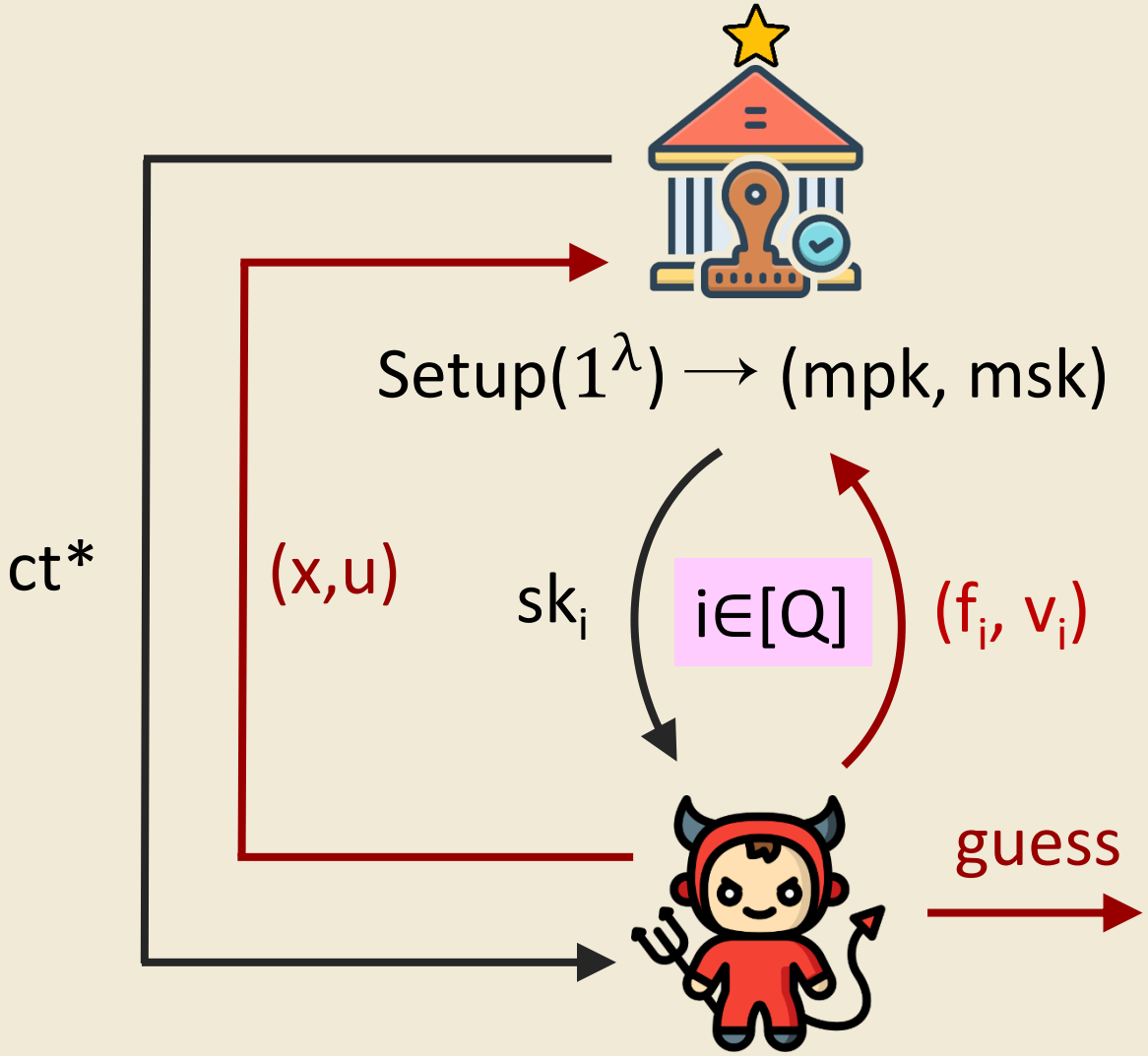
- ✓ **Attribute x** is hidden from Adversary

  If all i∈[Q], $f_i(x)$=false (0-key)

Weak Attribute hiding

Security

Predicate Inner Product Functional Encryption / P-IPFE



Security:

- ✓ **Message u** is hidden from Adversary

  If all i∈[Q], $f_i(x)$=false (0-key)

- ✓ **Attribute x** is hidden from Adversary

  Even ∃ i∈[Q], s.t. $f_i(x)$=true (0 & 1-key)

Fully Attribute hiding

Related Works

Predicate Inner Product Functional Encryption / P-IPFE

Predicate-IPFE (and Attribute-based IPFE)

	Attribute-hiding	Security Model	Assumption	Predicate Class
[LLW21]	×	IND-based	LWE	All poly-sized Boolean Circuit
[DDM+23]	Fully Attribute-hiding	IND-based	SXDH	(Zero) Inner Product Predicate
	Fully Attribute-hiding	SIM-based*	bilateral k-Lin	(Non-Zero) Inner Product Predicate

* The *secret-key* UNP-IPFE scheme in [DDM+23] achieves sim-based security.

Related Works

Predicate Inner Product Functional Encryption / P-IPFE

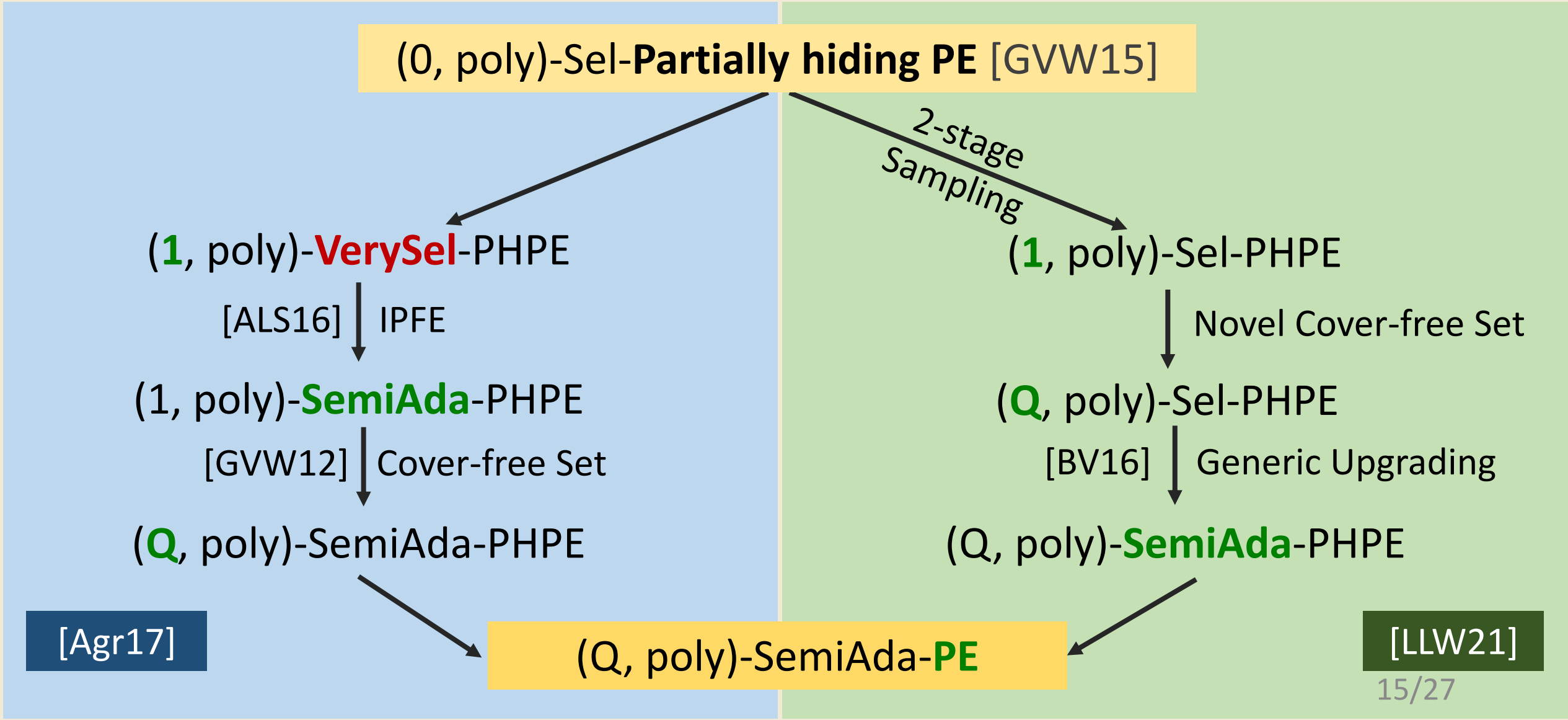
Predicate-IPFE (and Attribute-based IPFE)

	Attribute-hiding	Security Model	Assumption	Predicate Class
[LLW21]	×	IND-based	LWE	All poly-sized Boolean Circuit
[DDM+23]	Fully Attribute-hiding	IND-based	SXDH	(Zero) Inner Product Predicate
	Fully Attribute-hiding	SIM-based*	bilateral k-Lin	(Non-Zero) Inner Product Predicate
Ours	Fully Attribute-hiding	SIM-based	LWE	All poly-sized Boolean Circuit

Inherit the optimized compactness from (Q,poly) PE scheme

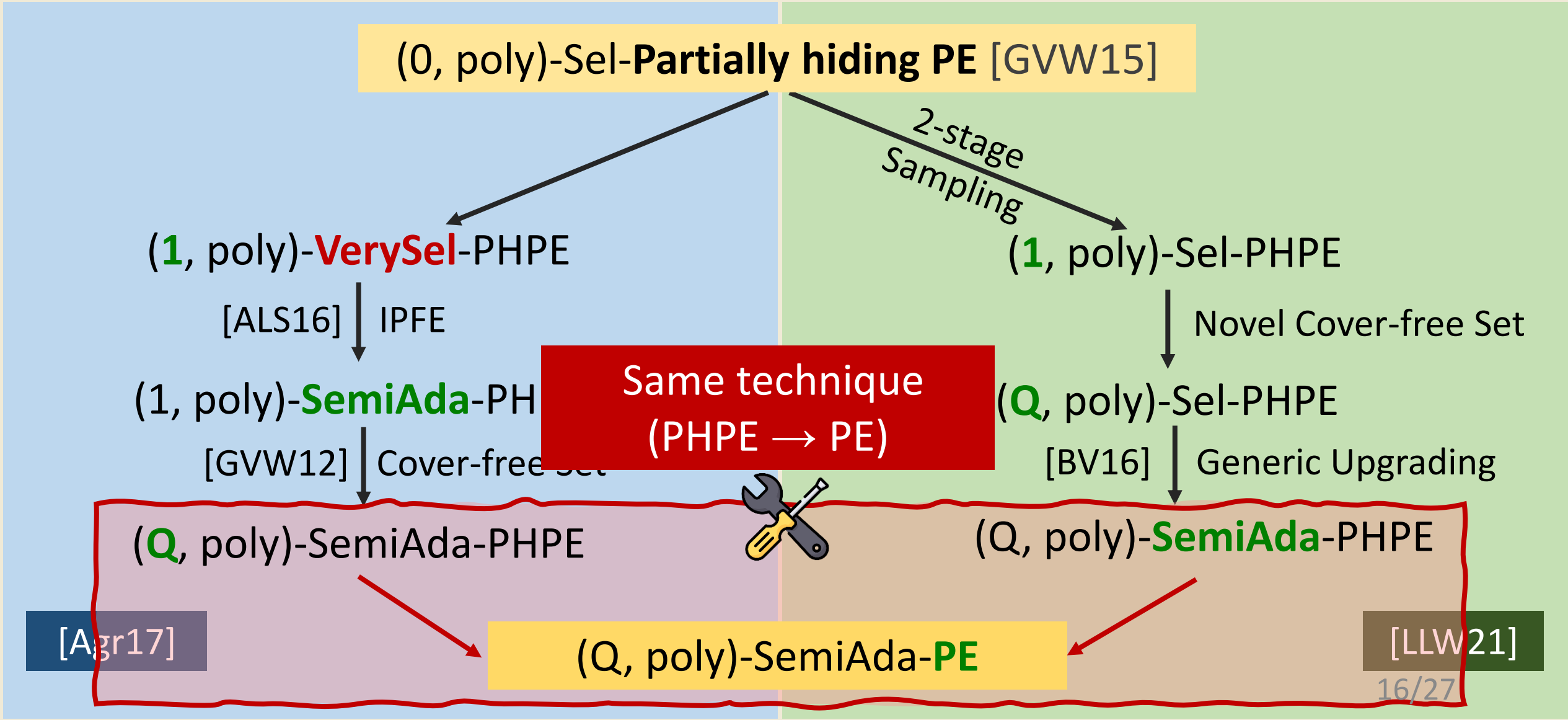
Technical Background

Predicate Encryption / PE



Technical Background

Predicate Encryption / PE



Technical Background

Predicate Encryption / PE

Attribute-based Encryption [BGG+14]

PHPE for $C \circ IP(x_{pub}, x_{pri}) = \langle C(x_{pub}), x_{pri} \rangle$

- Use [GSW13] FHE to hide **public attribute** in PHPE & Set FHE.sk as **private attribute**
- Require "lazy-OR" + Smudging noise

[GVW15, Agr17, LLW21]

Fully attribute-hiding PE

- Encrypt x using **FHE**
- Automatic Decryption by "Dual-used"

[BTVW17]

Technical Background

Predicate Encryption / PE

Attribute-based Encryption [BGG+14]

- Encrypt x using **FHE**
- **Automatic Decryption** by "**Dual-used**" (encoding secret = FHE secret key)

Weak attribute-hiding PE

[BTVW17]

PHPE for $C \circ IP(x_{pub}, x_{pri}) = \langle C(x_{pub}), x_{pri} \rangle$

- Require "lazy-OR" + Smudging noise

[GVW15, Agr17, LLW21]

Technical Approach

Predicate Encryption / PE

(0, poly)-Sel-**Partially hiding PE** [GVW15]

2-stage Sampling

(**1**, poly)-**VerySel**-PHPE

[ALS16] IPFE

(1, poly)-**SemiAda**-PHPE

[GVW12] Cover-free Set

(**Q**, poly)-SemiAda-PHPE

(**1**, poly)-Sel-PHPE

[LLW21] Cover-free Set

(**Q**, poly)-Sel-PHPE

[BV16] Generic Upgrading

(Q, poly)-**SemiAda**-PHPE

(0, poly)-Sel-**PE** [BTVW17]

[LLW21] 2-stage Sampling

(**1**, poly)-Sel-PE

[LLW21] Cover-free Set

(**Q**, poly)-Sel-PE

[BV16] Generic Upgrading

(Q, poly)-SemiAda-PE

[Agr17]

[LLW21]

This Work

Technical Approach

Predicate Encryption / PE

Our Results

- **Avoid** both “lazy-OR” and Smudging noise
- Secret key is **independent** of **FHE dec. noise**
- Reduced Fixed Overhead:

  Encodings for private attribute (FHE.sk)
- Reduced Per-unit expansion:

  Dummy FHE.ct **&**

  Encodings for dummy FHE.ct

(0, poly)-Sel-**PE** [BTVW17]

[LLW21] 2-stage Sampling

(**1**, poly)-Sel-PE

[LLW21] Cover-free Set

(**Q**, poly)-Sel-PE

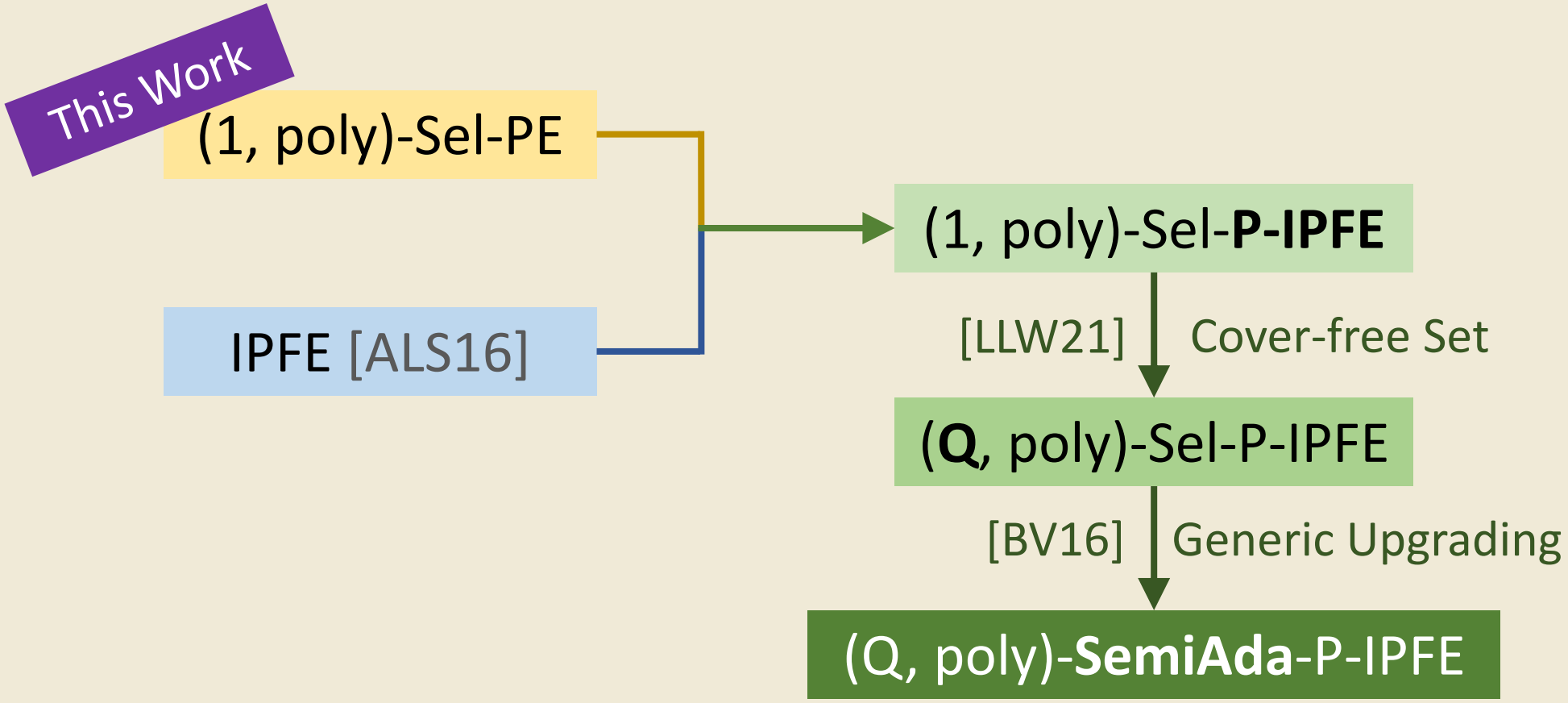
[BV16] Generic Upgrading

(Q, poly)-SemiAda-PE

Technical Approach

Predicate Inner Product Functional Encryption / P-IPFE

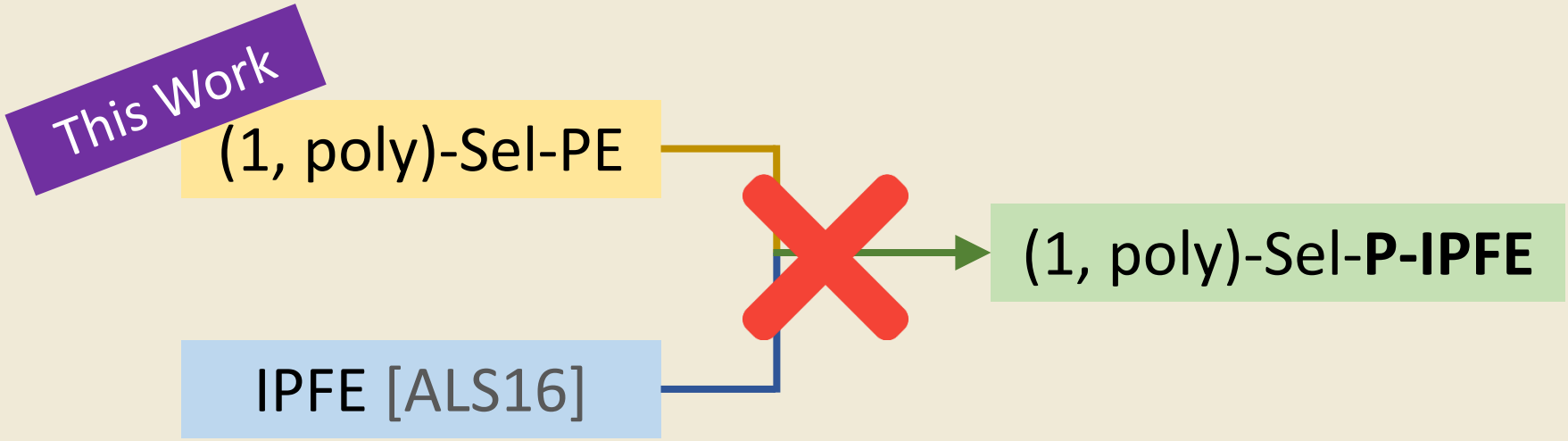
Initial Idea



Technical Approach

Predicate Inner Product Functional Encryption / P-IPFE

Initial Idea



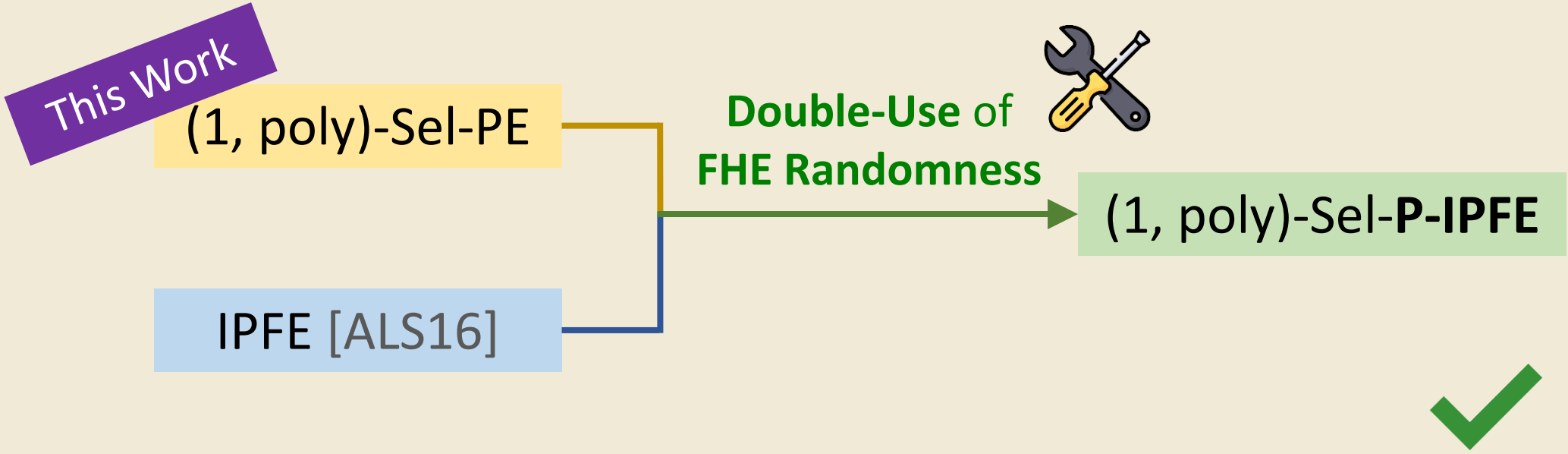
During Security proof,

- **FHE.pk** is completely known until the **challenge phase**
- **FHE.pk** is required to **compute hct(x)** and then **program A_{attr}** in the **Setup phase**!

Technical Approach

Predicate Inner Product Functional Encryption / P-IPFE

Our Approach

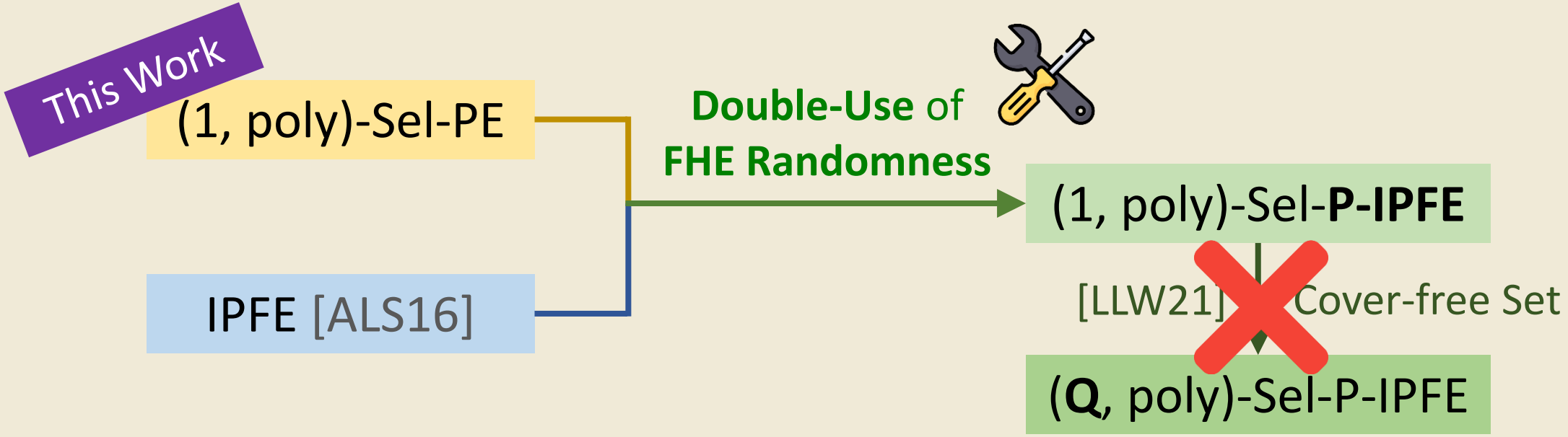


- **Encrypt** attribute x **twice** with the **different FHE.sk** but **the same FHE randomness**
- Useful fact: **Upper part** (except for last row) of [GSW13] **FHE.ct** doesn't include **FHE.sk**

Technical Approach

Predicate Inner Product Functional Encryption / P-IPFE

Initial Idea

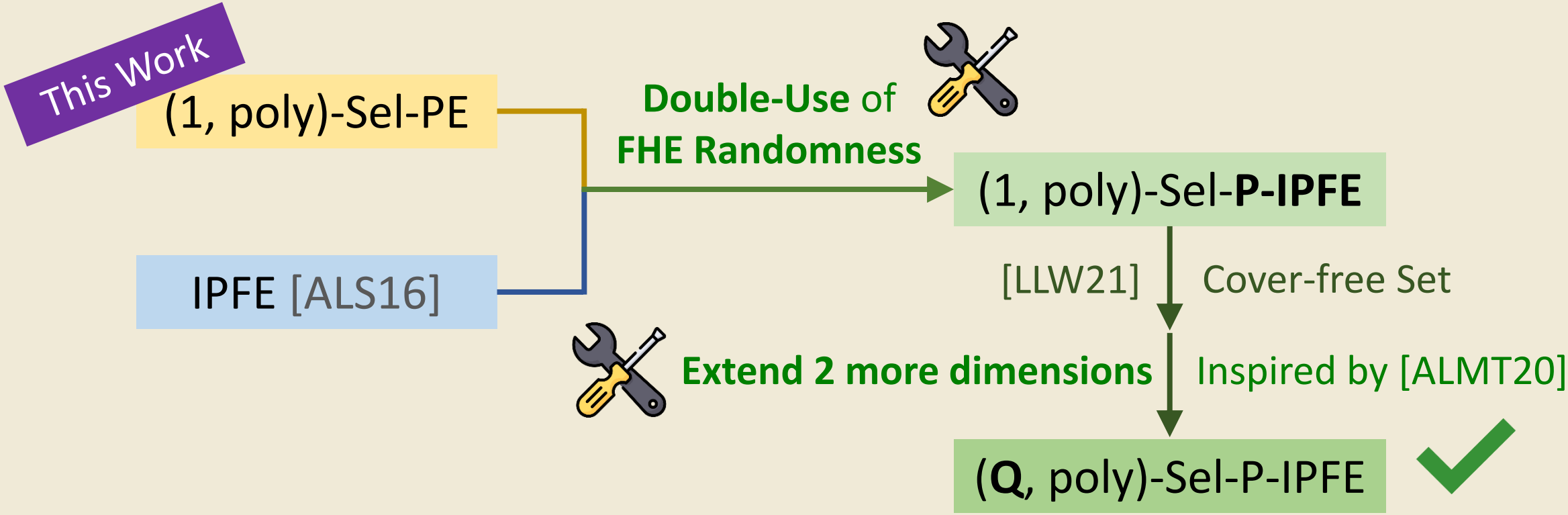


- Sim-based security requires additional programming space

Technical Approach

Predicate Inner Product Functional Encryption / P-IPFE

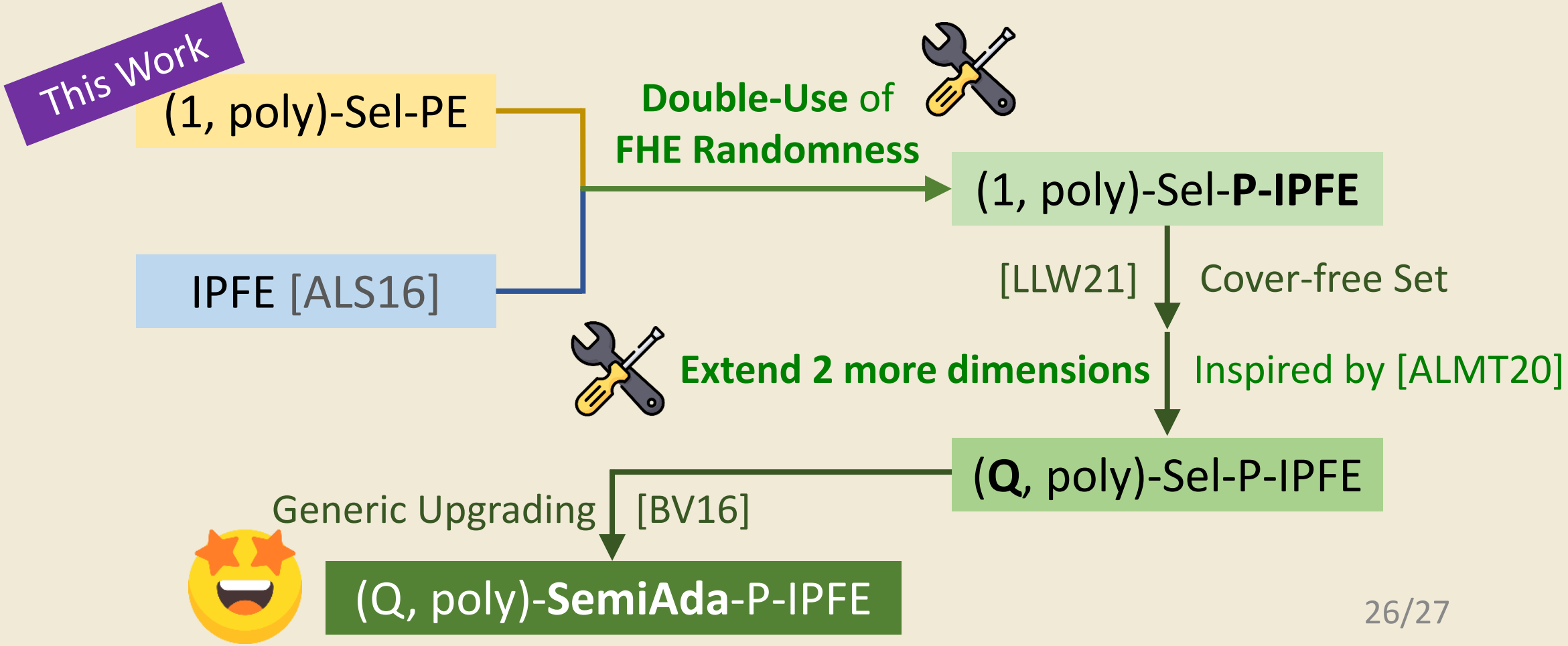
Our Approach



Technical Approach

Predicate Inner Product Functional Encryption / P-IPFE

Our Approach



Thank you!! Questions?

https://eprint.iacr.org/2025/361

All icons are from *flaticon.com*