

Registered Functional Encryption for Attribute-Weighted Sums with Access Control

Tapas Pal¹

Robert Schädlich²

December 5, 2024

¹ Karlsruhe Institute of Technology, KASTEL Security Research Labs

² DIENS, École normale supérieure, PSL University, CNRS, Inria

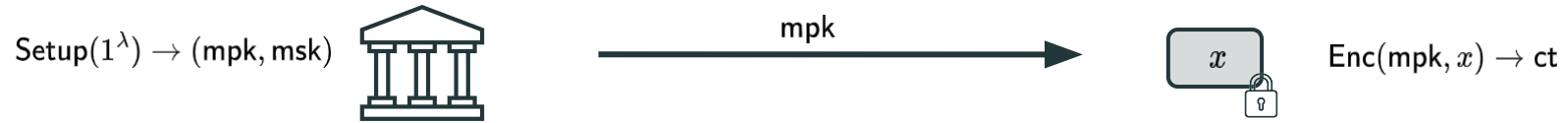


Functional Encryption (FE) [TCC:BSW11]

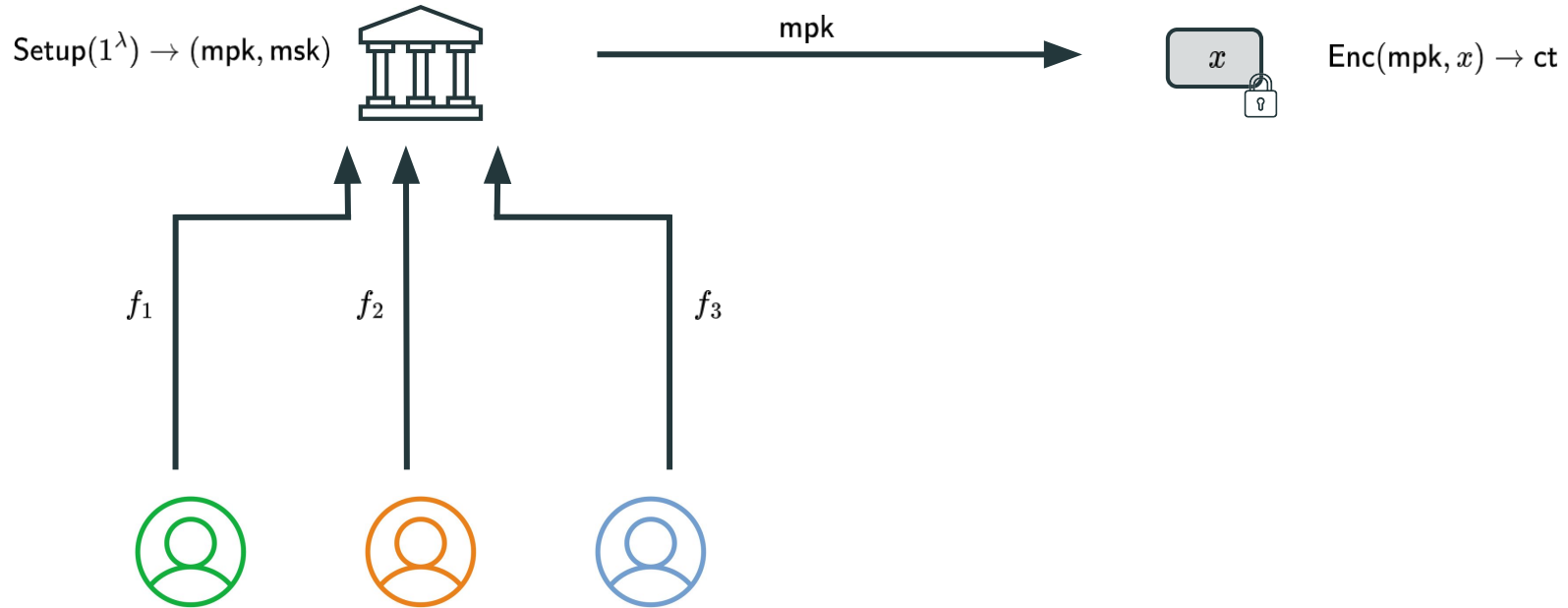
$\text{Setup}(1^\lambda) \rightarrow (\text{mpk}, \text{msk})$



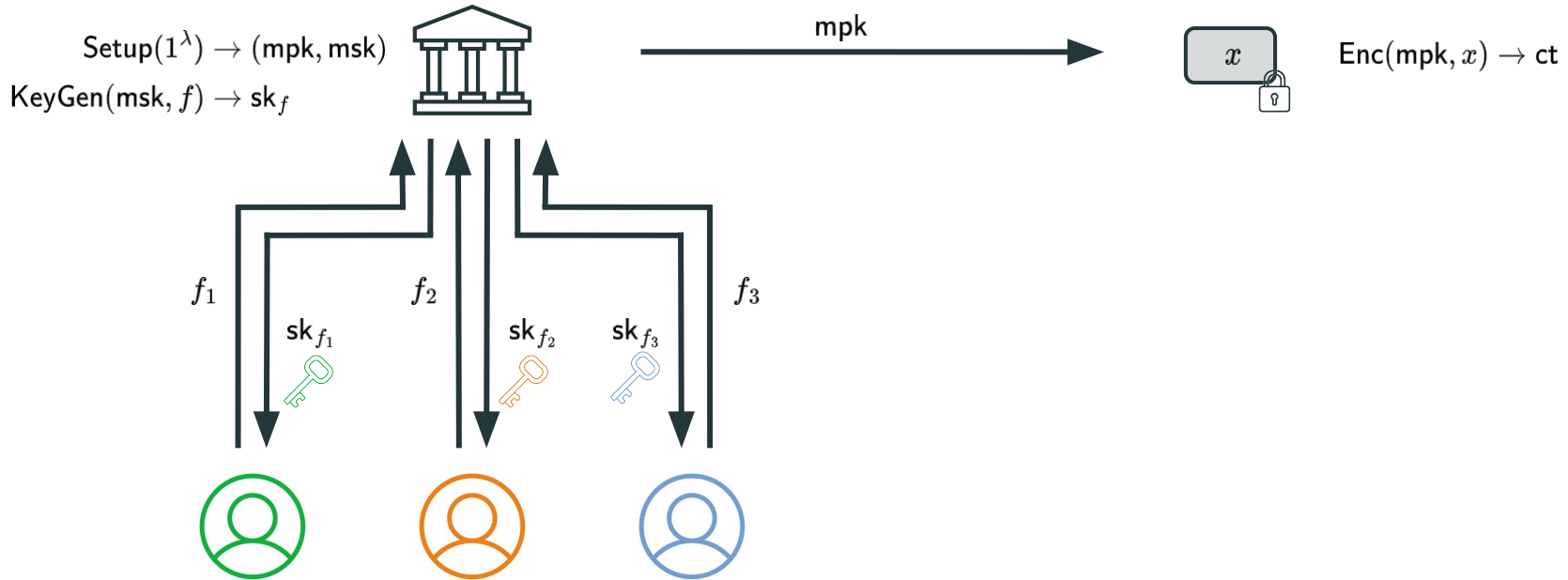
Functional Encryption (FE) [TCC:BSW11]



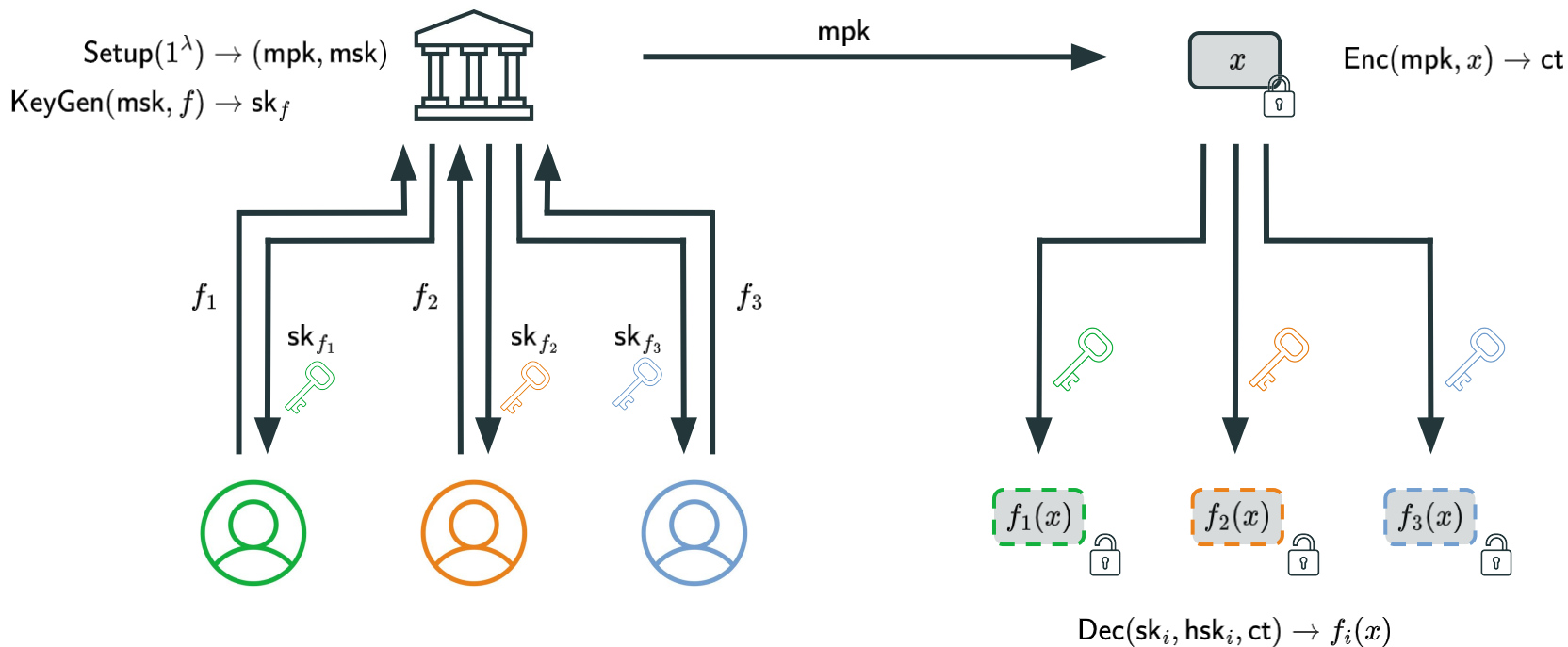
Functional Encryption (FE) [TCC:BSW11]



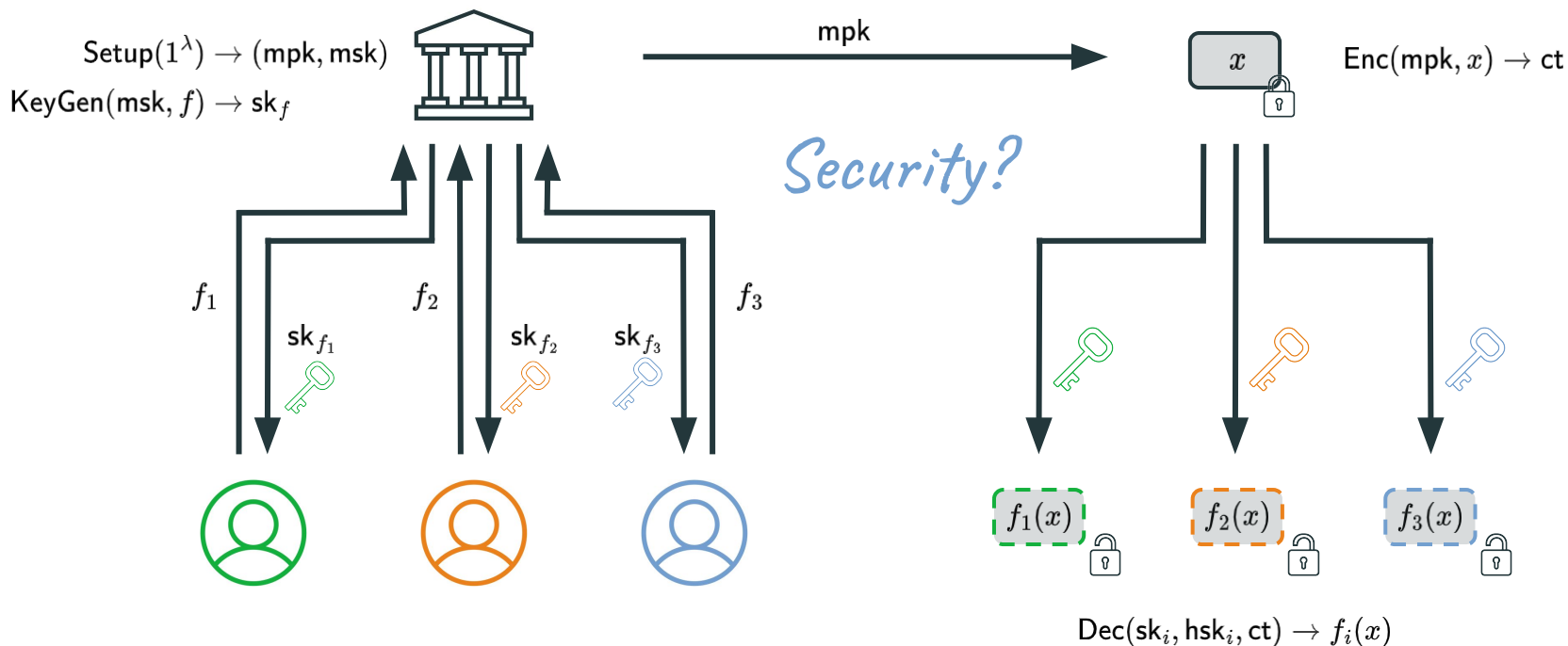
Functional Encryption (FE) [TCC:BSW11]



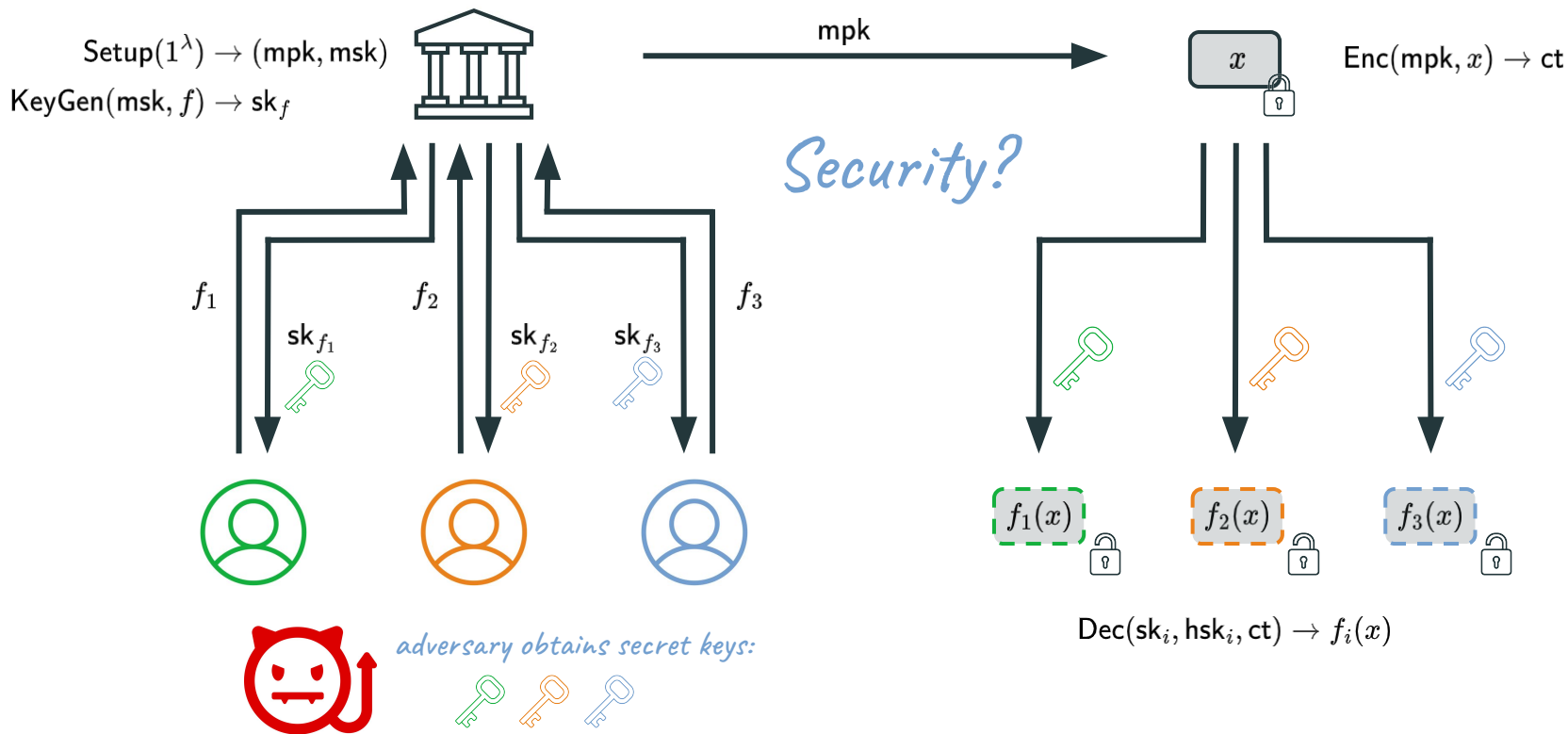
Functional Encryption (FE) [TCC:BSW11]



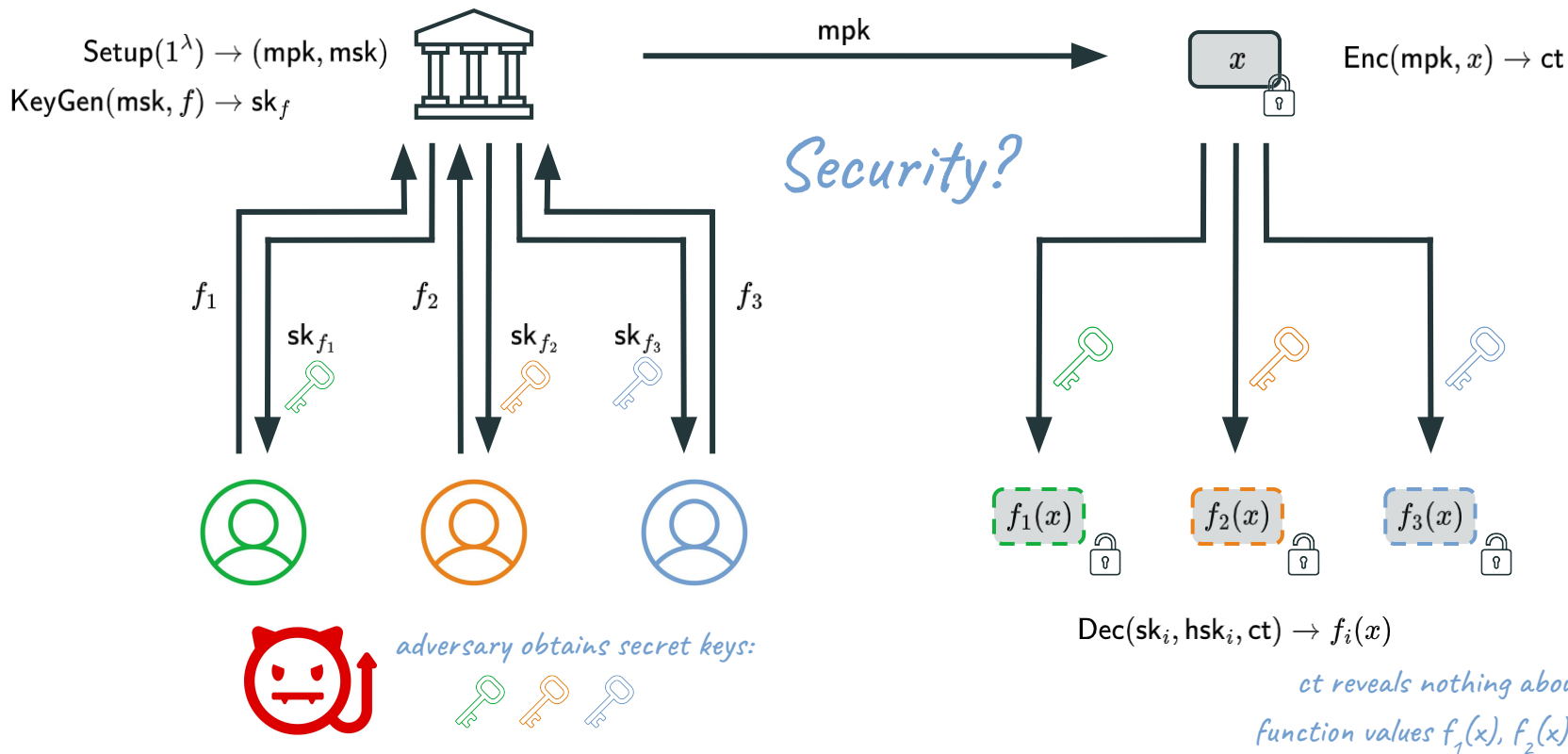
Functional Encryption (FE) [TCC:BSW11]



Functional Encryption (FE) [TCC:BSW11]

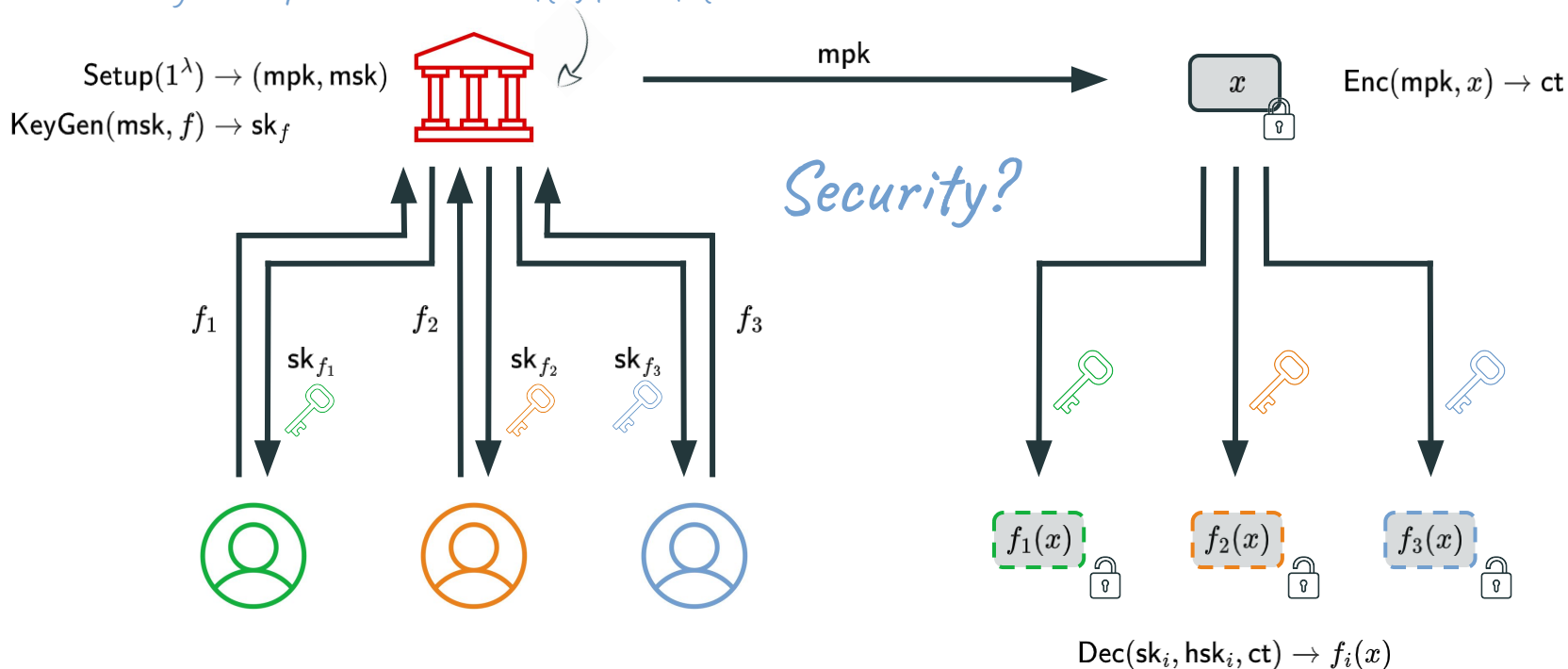


Functional Encryption (FE) [TCC:BSW11]



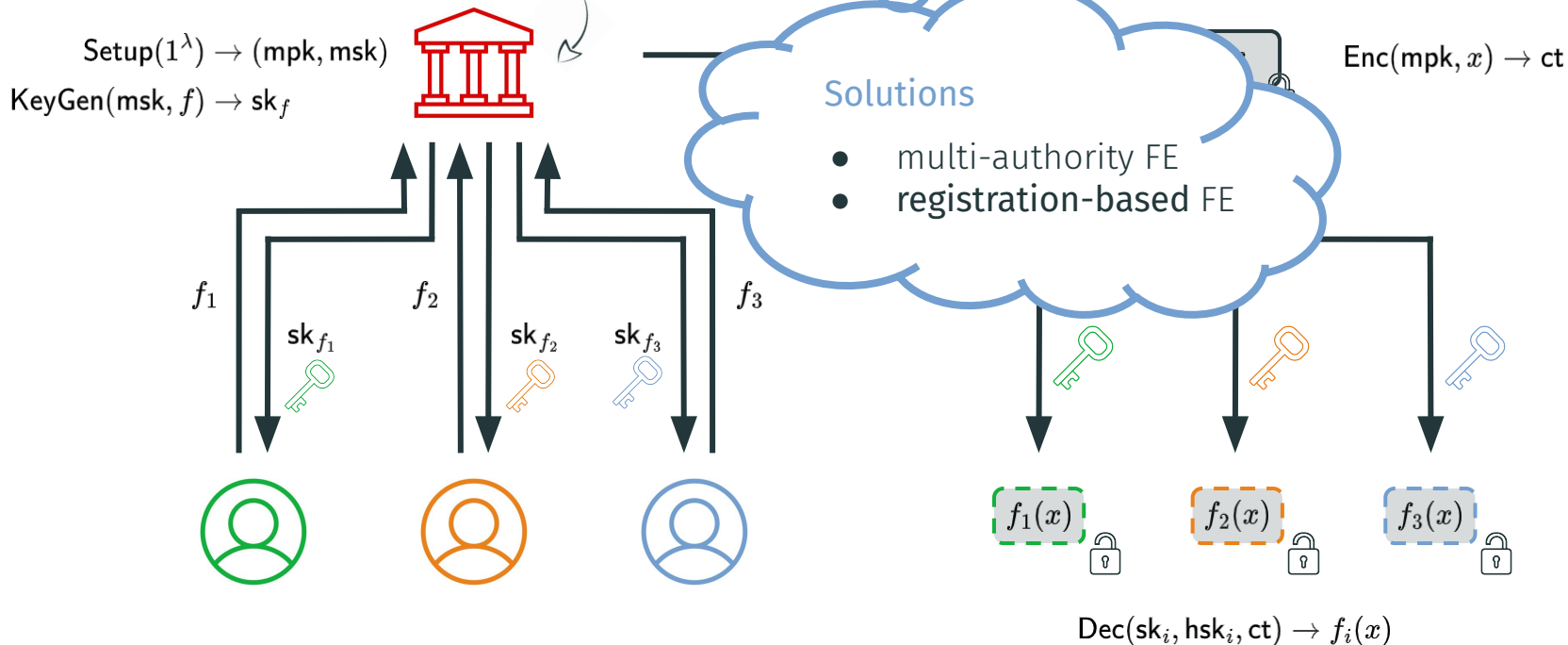
Functional Encryption (FE) [TCC:BSW11]

key-escrow problem: msk reveals $f(m)$ for all f :



Functional Encryption (FE) [TCC:BSW11]

key-escrow problem: msk reveals $f(m)$ for all f :



Registered Functional Encryption (RFE) [AC:FFM+23]

$\text{Setup}(1^\lambda) \rightarrow \text{crs}$



pk_1, sk_1



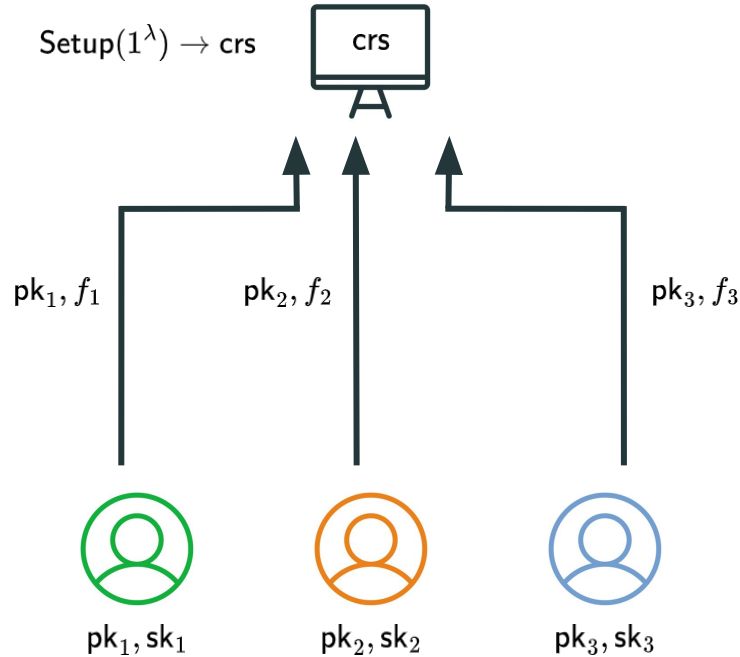
pk_2, sk_2



pk_3, sk_3

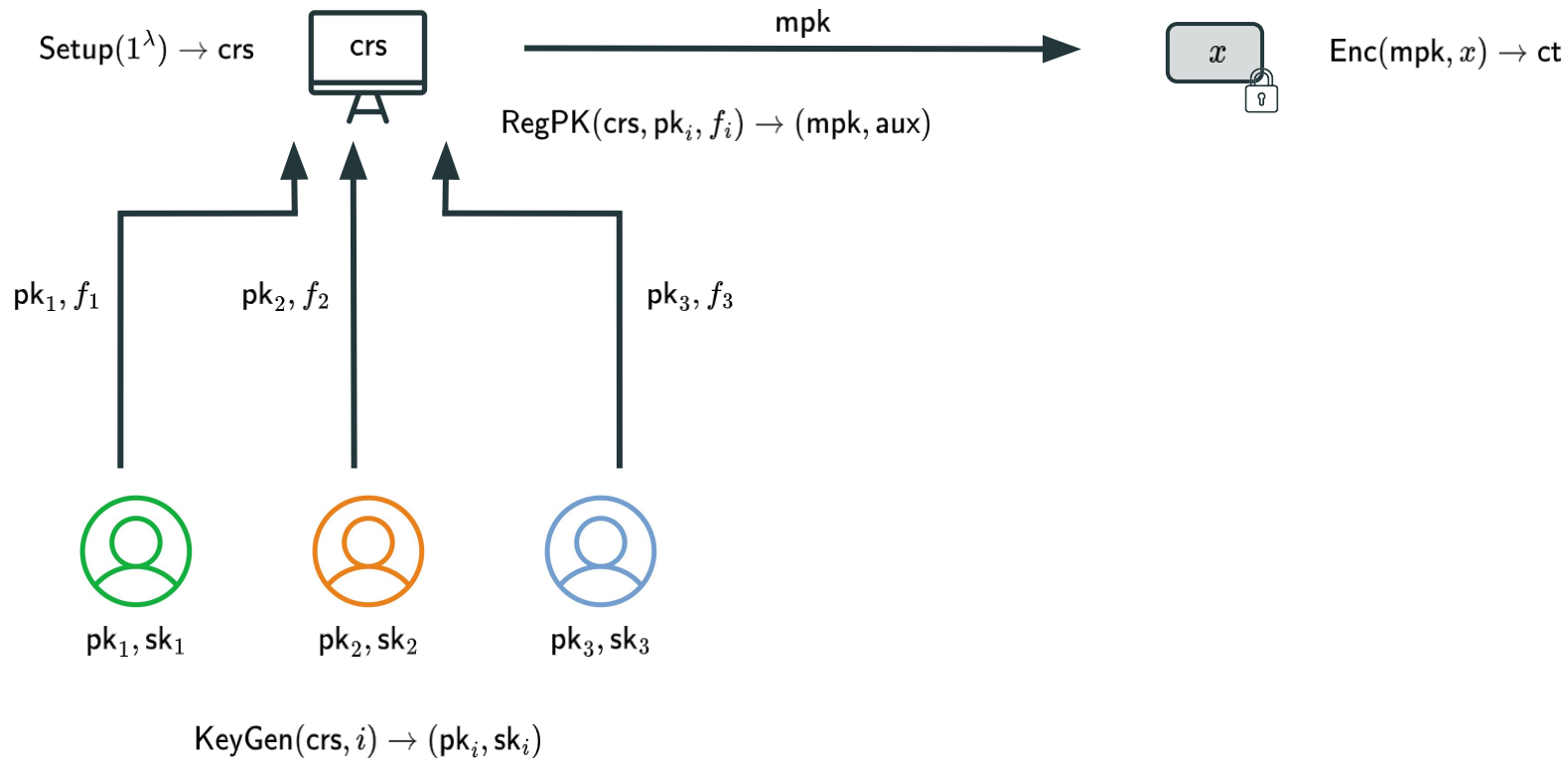
$\text{KeyGen}(\text{crs}, i) \rightarrow (\text{pk}_i, \text{sk}_i)$

Registered Functional Encryption (RFE) [AC:FFM+23]

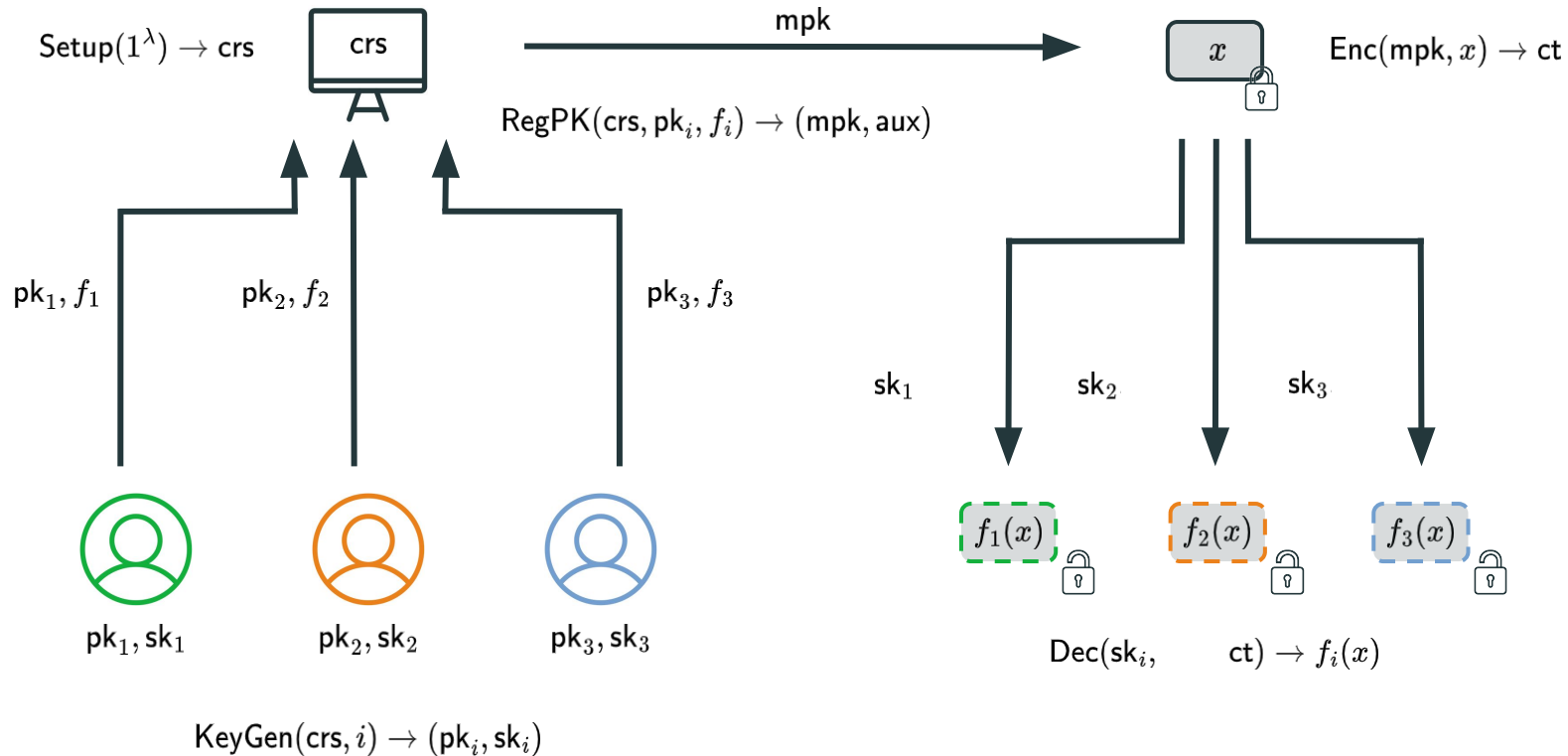


$\text{KeyGen}(\text{crs}, i) \rightarrow (pk_i, sk_i)$

Registered Functional Encryption (RFE) [AC:FFM+23]

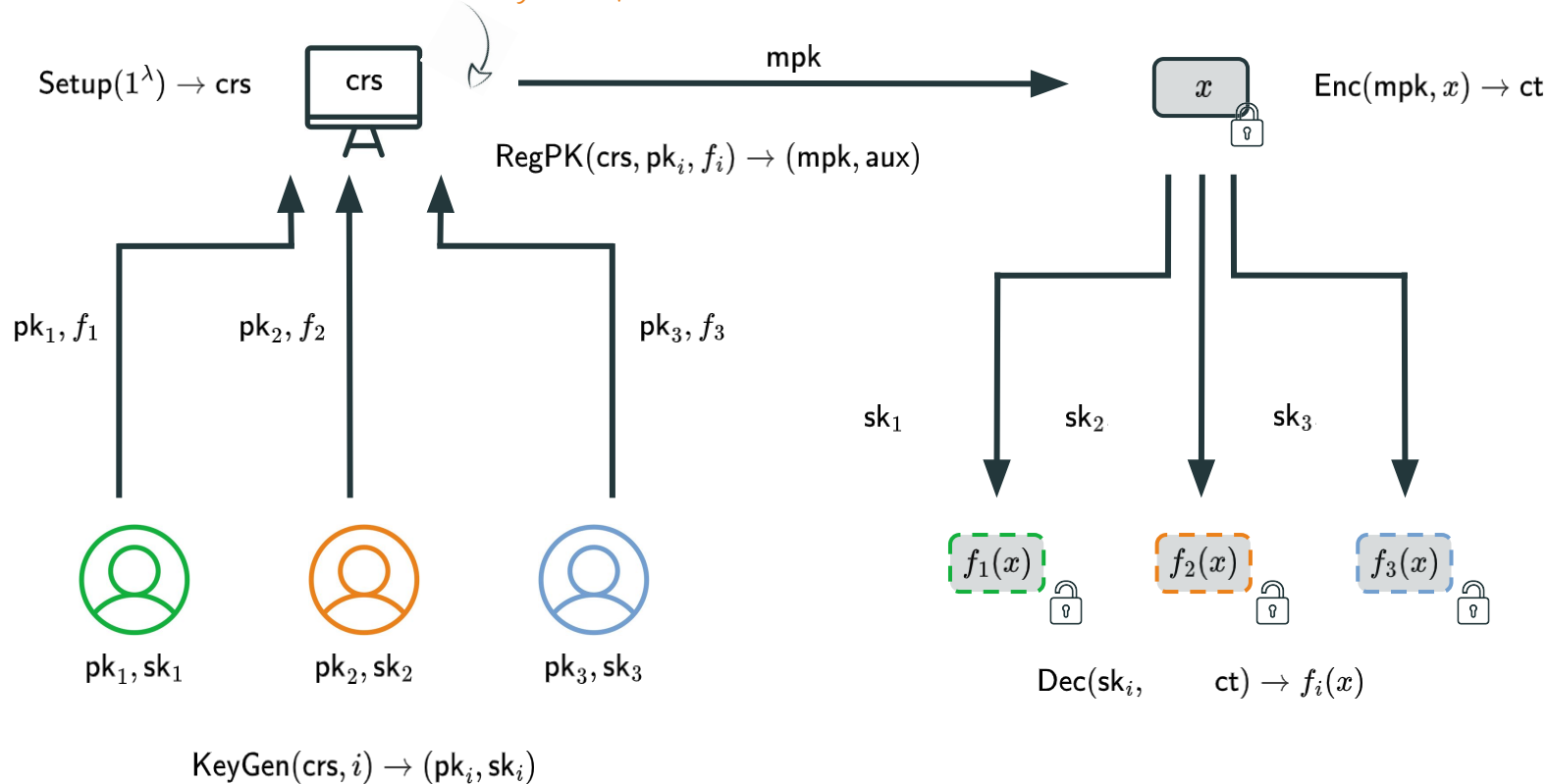


Registered Functional Encryption (RFE) [AC:FFM+23]



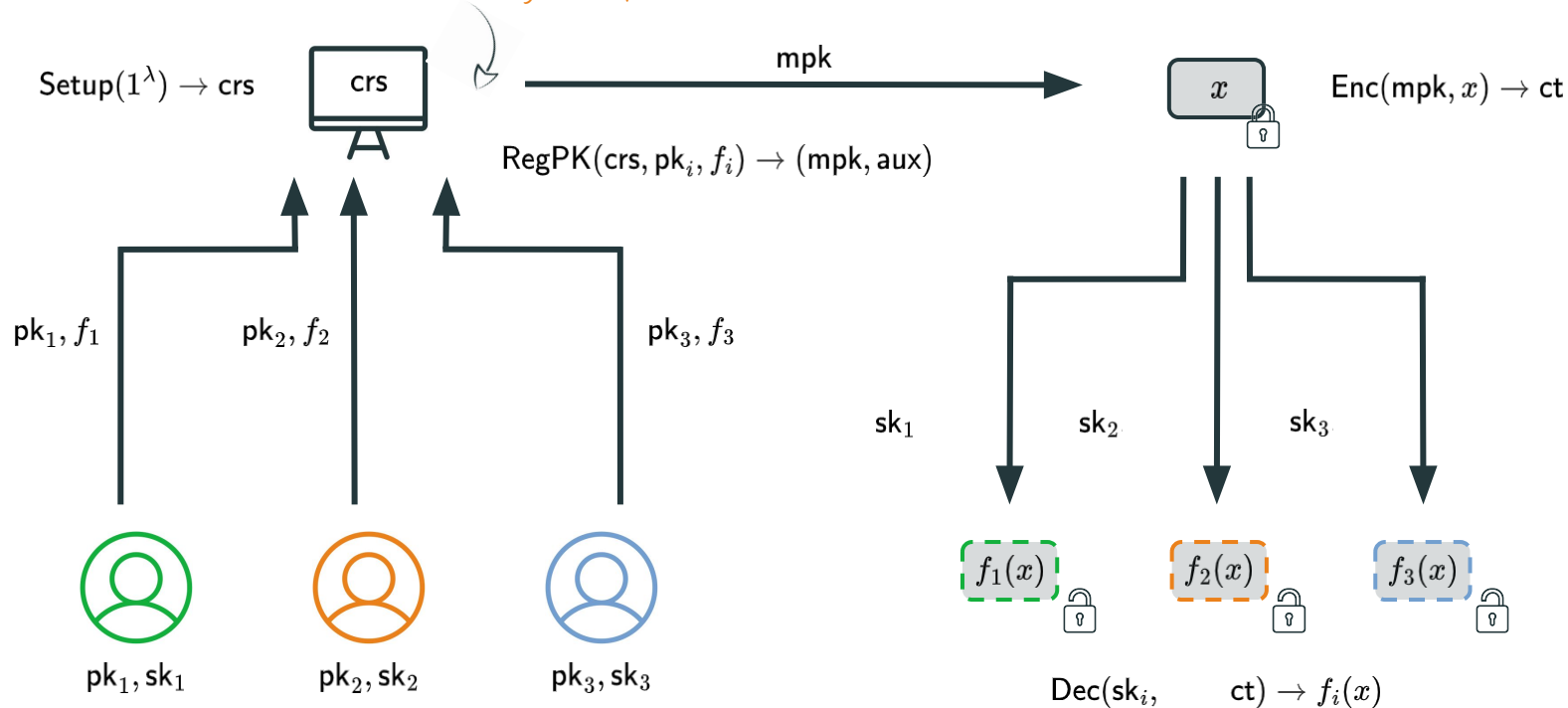
Registered Functional Encryption (RFE) [AC:FFM+23]

key curator is deterministic & holds no secret => key-escrow problem resolved!



Registered Functional Encryption (RFE) [AC:FFM+23]

key curator is deterministic & holds no secret => key-escrow problem resolved!

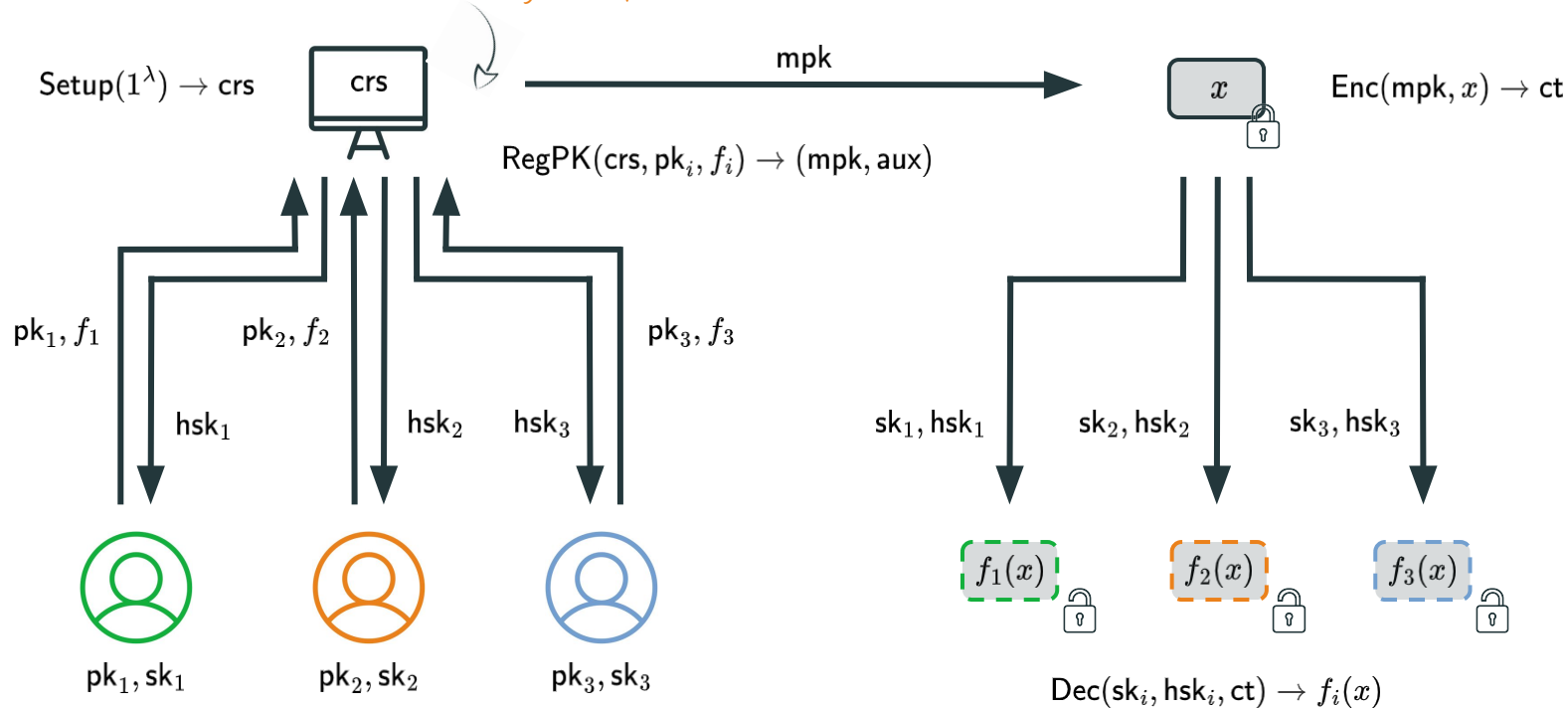


$\text{KeyGen}(\text{crs}, i) \rightarrow (pk_i, sk_i)$

compactness: $|mpk|, |ct| = \text{poly}(\log L)$ where $L = \#users$

Registered Functional Encryption (RFE) [AC:FFM+23]

key curator is deterministic & holds no secret => key-escrow problem resolved!

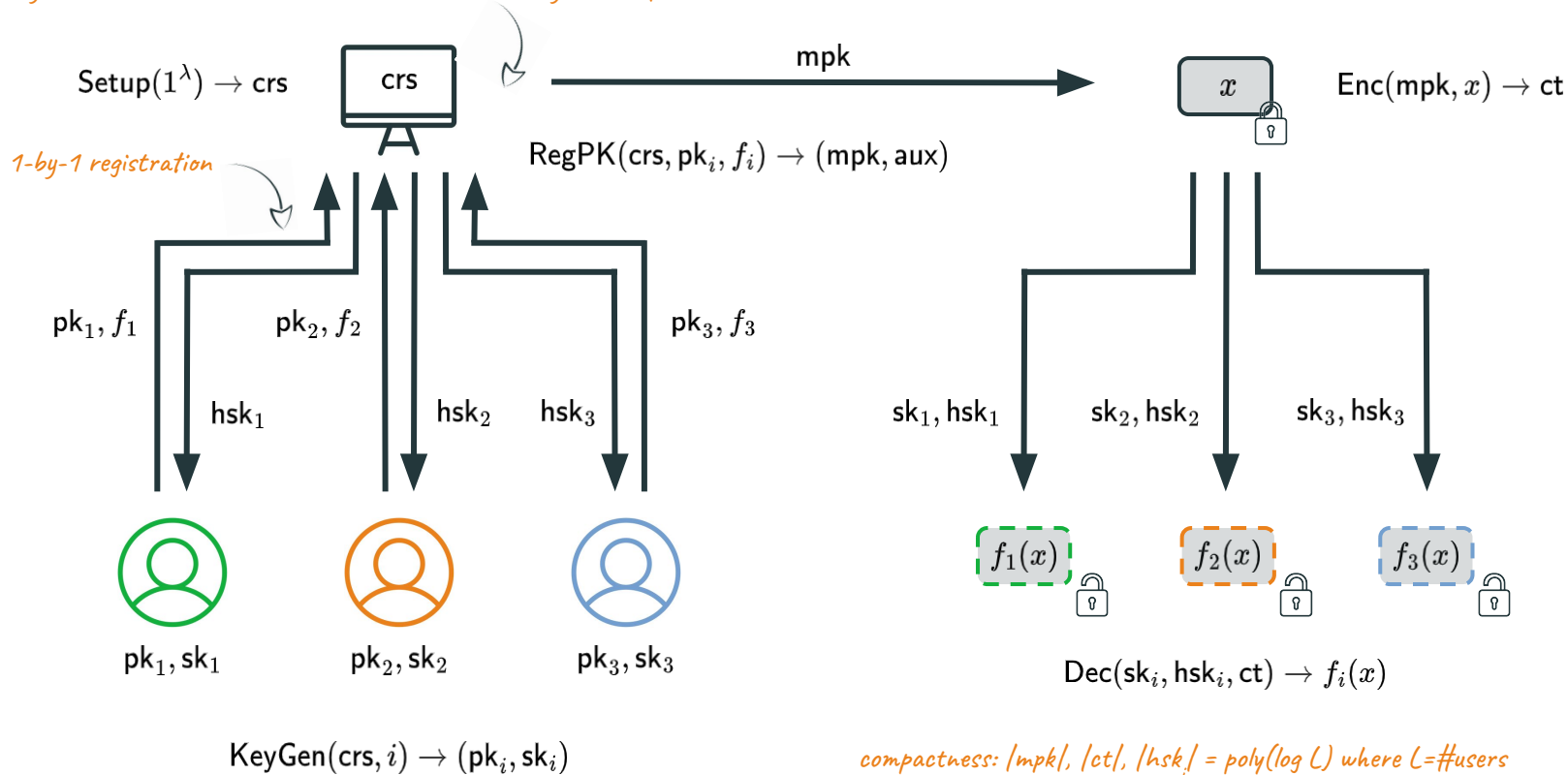


$\text{KeyGen}(\text{crs}, i) \rightarrow (\text{pk}_i, \text{sk}_i)$

compactness: $|\text{mpk}|, |\text{ct}|, |\text{hsk}_i| = \text{poly}(\log L)$ where $L = \# \text{users}$

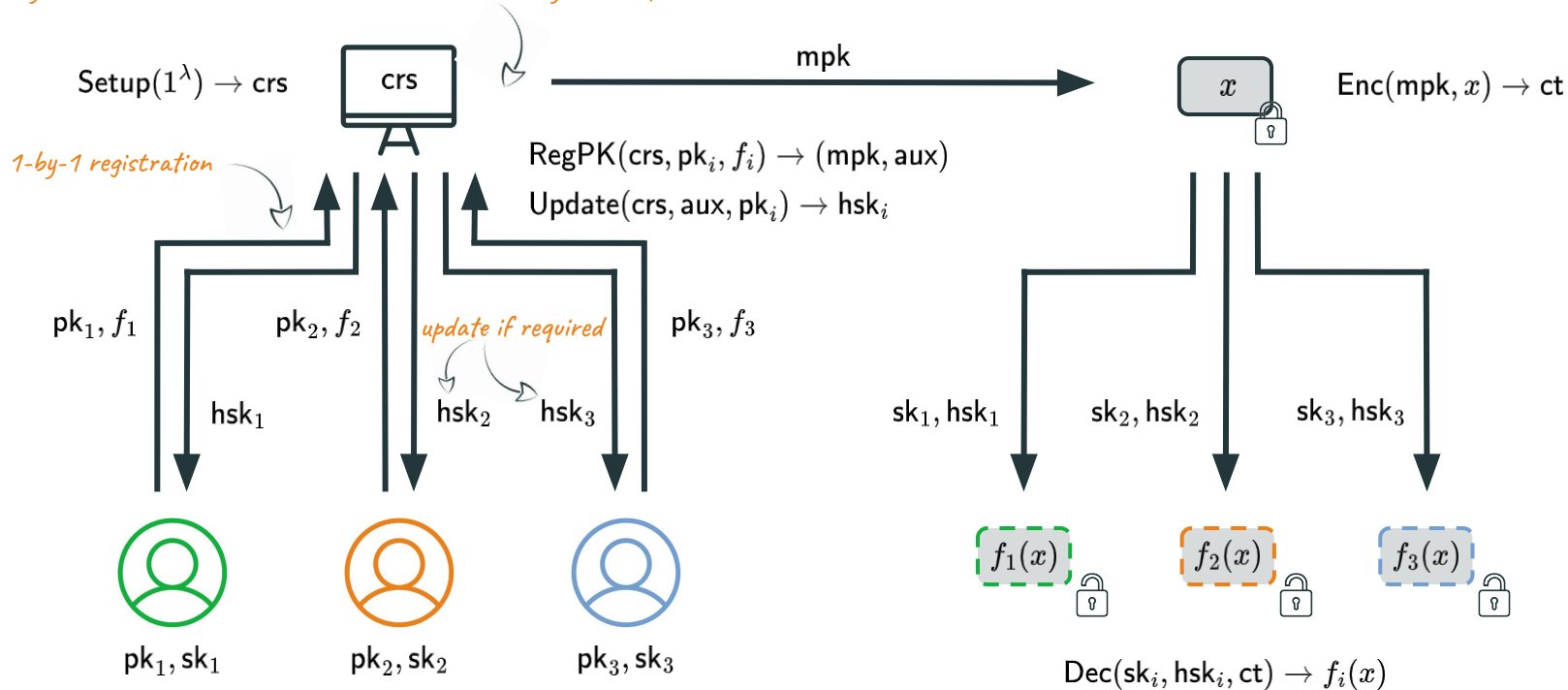
Registered Functional Encryption (RFE) [AC:FFM+23]

key curator is deterministic & holds no secret => key-escrow problem resolved!



Registered Functional Encryption (RFE) [AC:FFM+23]

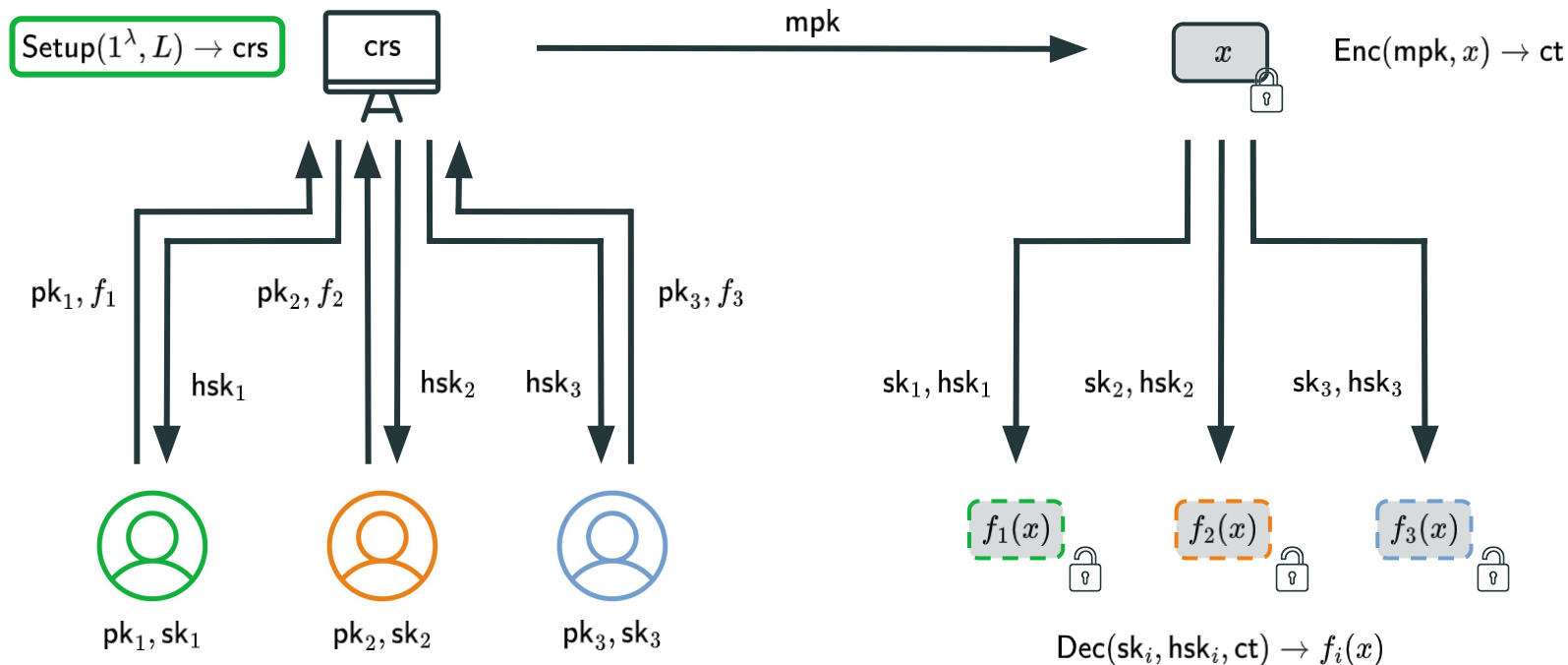
key curator is deterministic & holds no secret => key-escrow problem resolved!



$\text{KeyGen}(\text{crs}, i) \rightarrow (\text{pk}_i, \text{sk}_i)$

compactness: $|\text{mpk}|, |\text{ct}|, |\text{hsk}_i|, \# \text{updates} = \text{poly}(\log L)$ where $L = \# \text{users}$

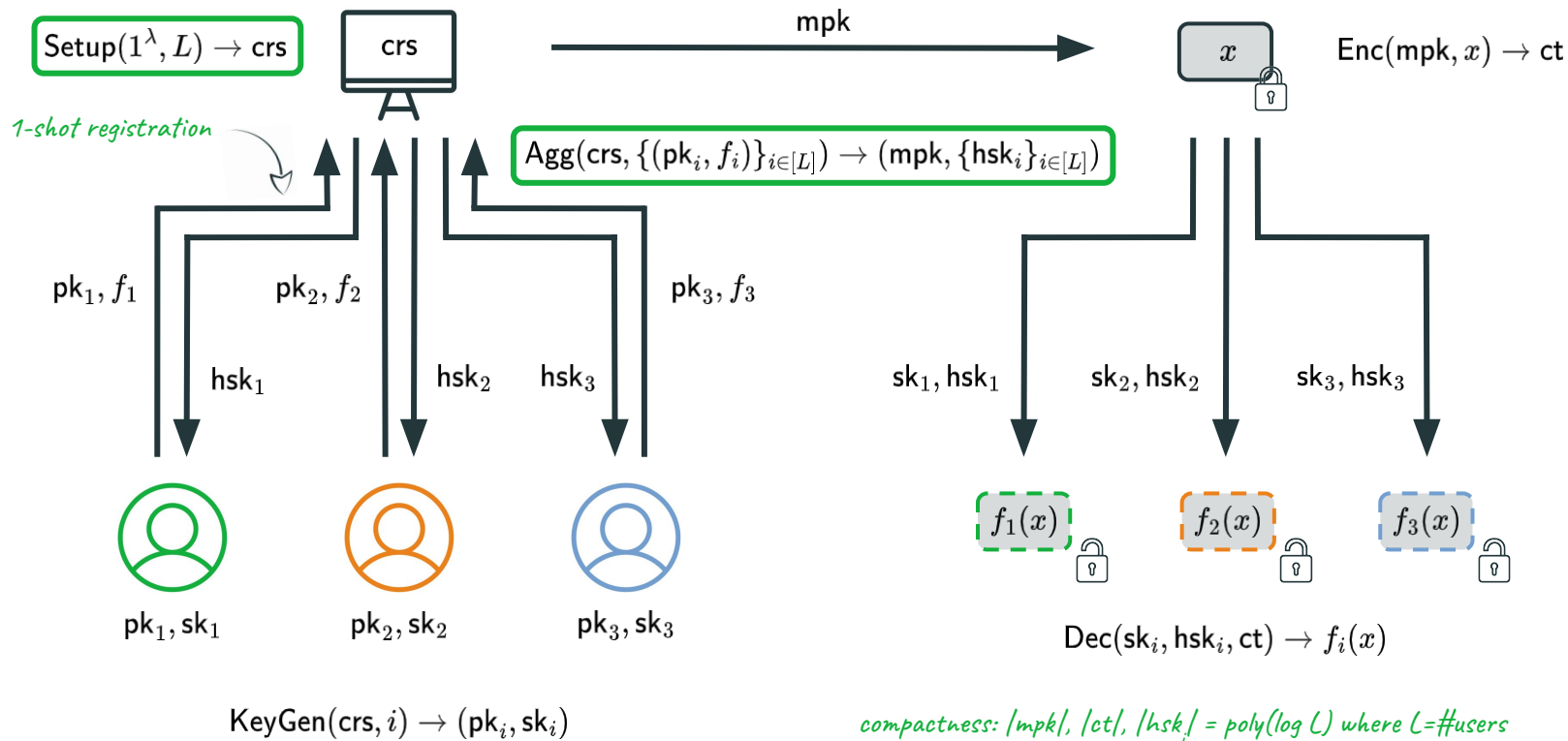
Slotted Registered Functional Encryption (sRFE)



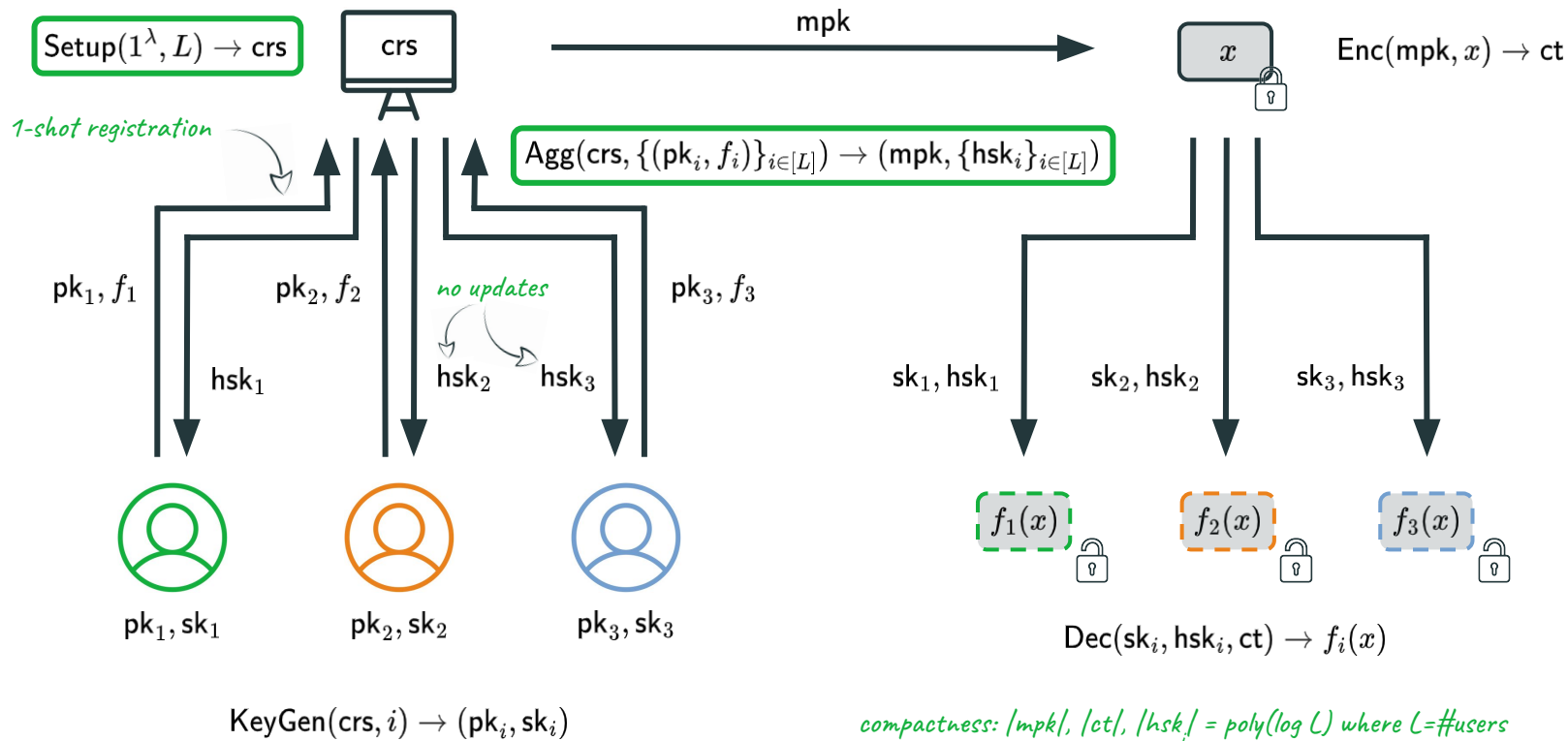
$$\text{KeyGen}(\text{crs}, i) \rightarrow (pk_i, sk_i)$$

compactness: $|mpk|, |ct|, |hsk_i| = \text{poly}(\log L)$ where $L = \#users$

Slotted Registered Functional Encryption (sRFE)

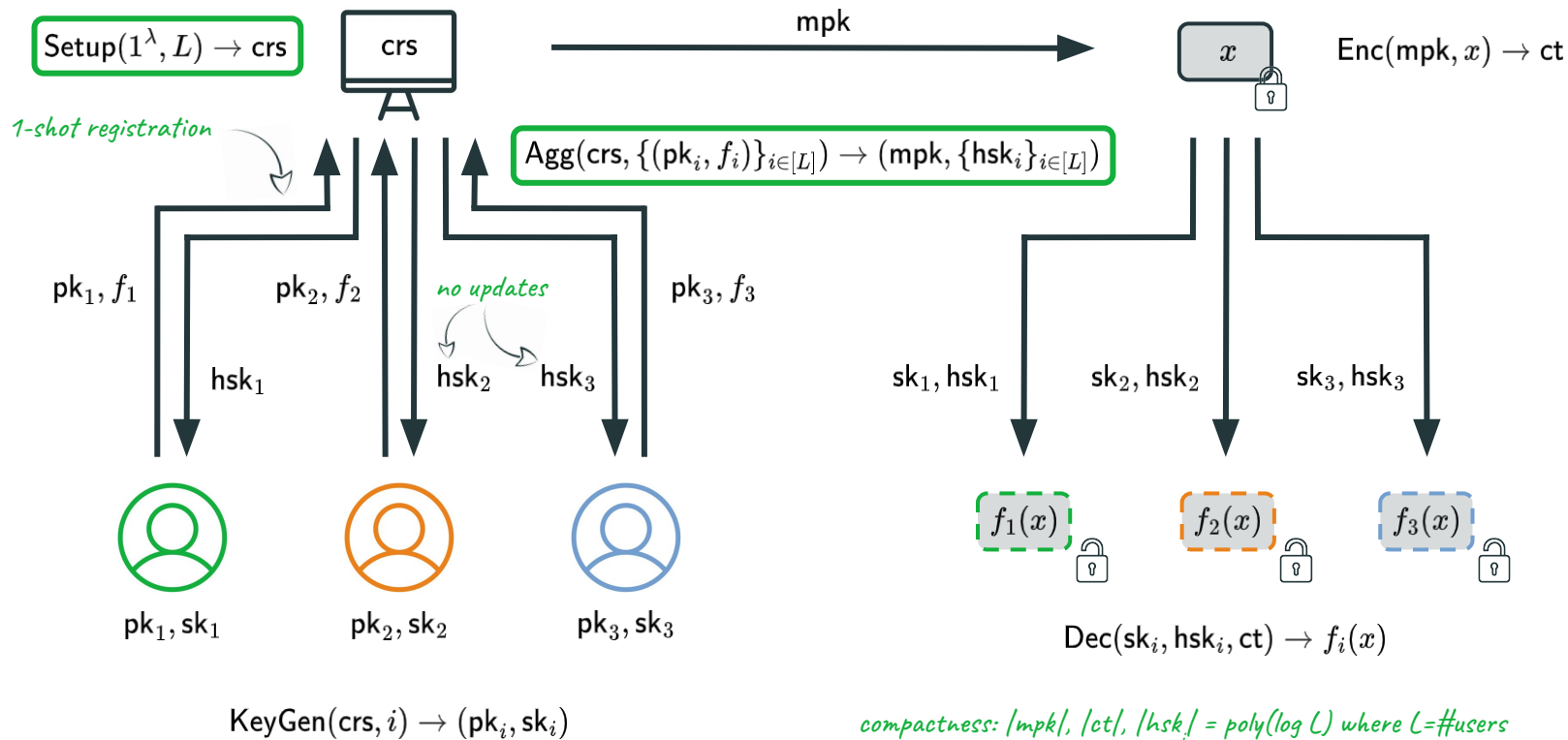


Slotted Registered Functional Encryption (sRFE)

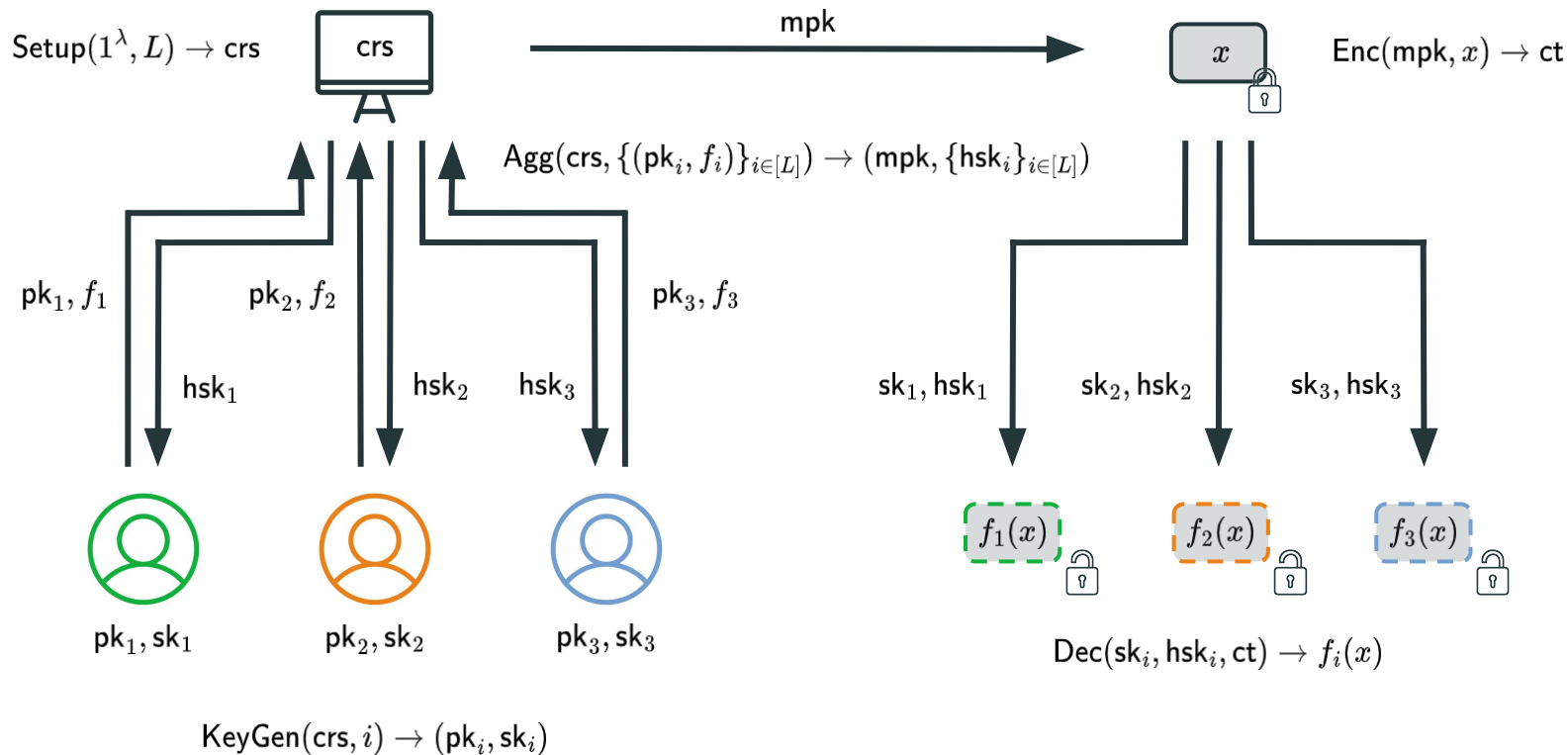


Slotted Registered Functional Encryption (sRFE)

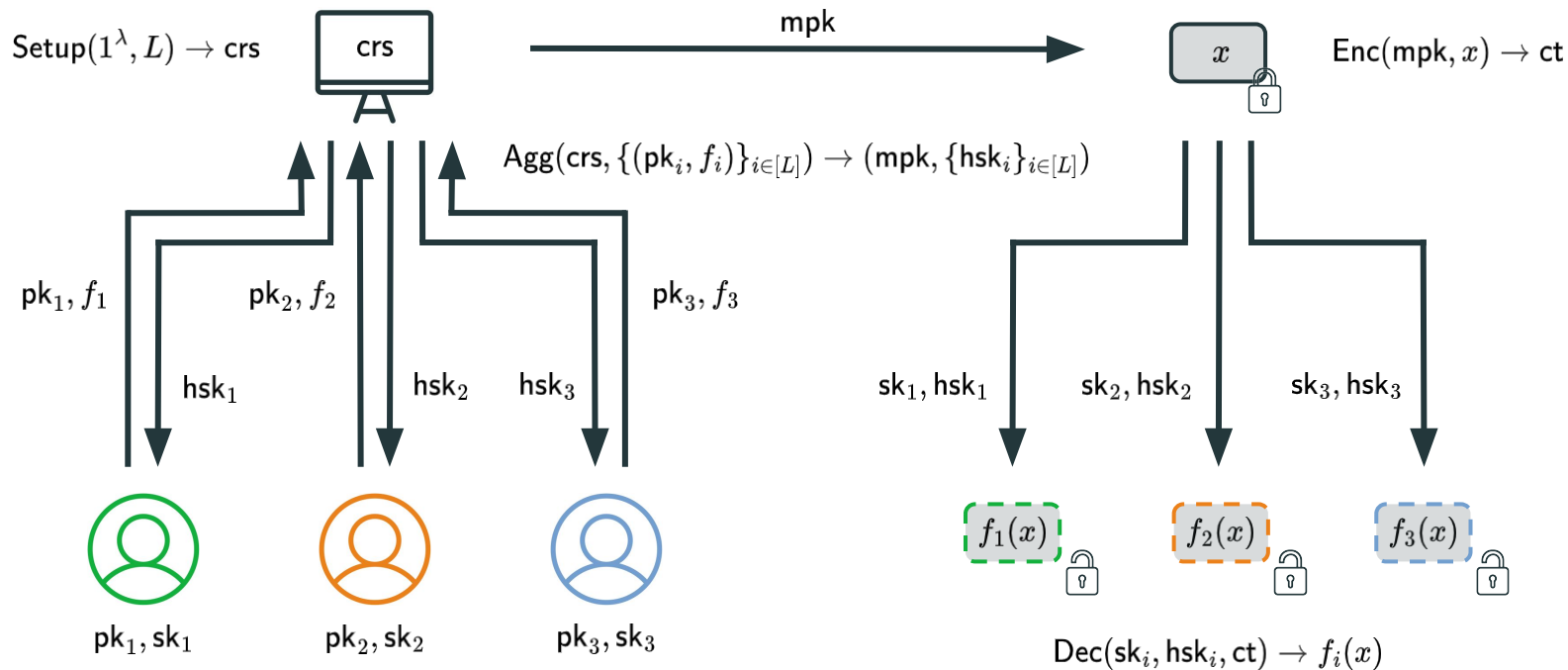
[HLWW23]: sRFE \Rightarrow RFE ("powers-of-two compiler")



Slotted Registered Functional Encryption (sRFE)



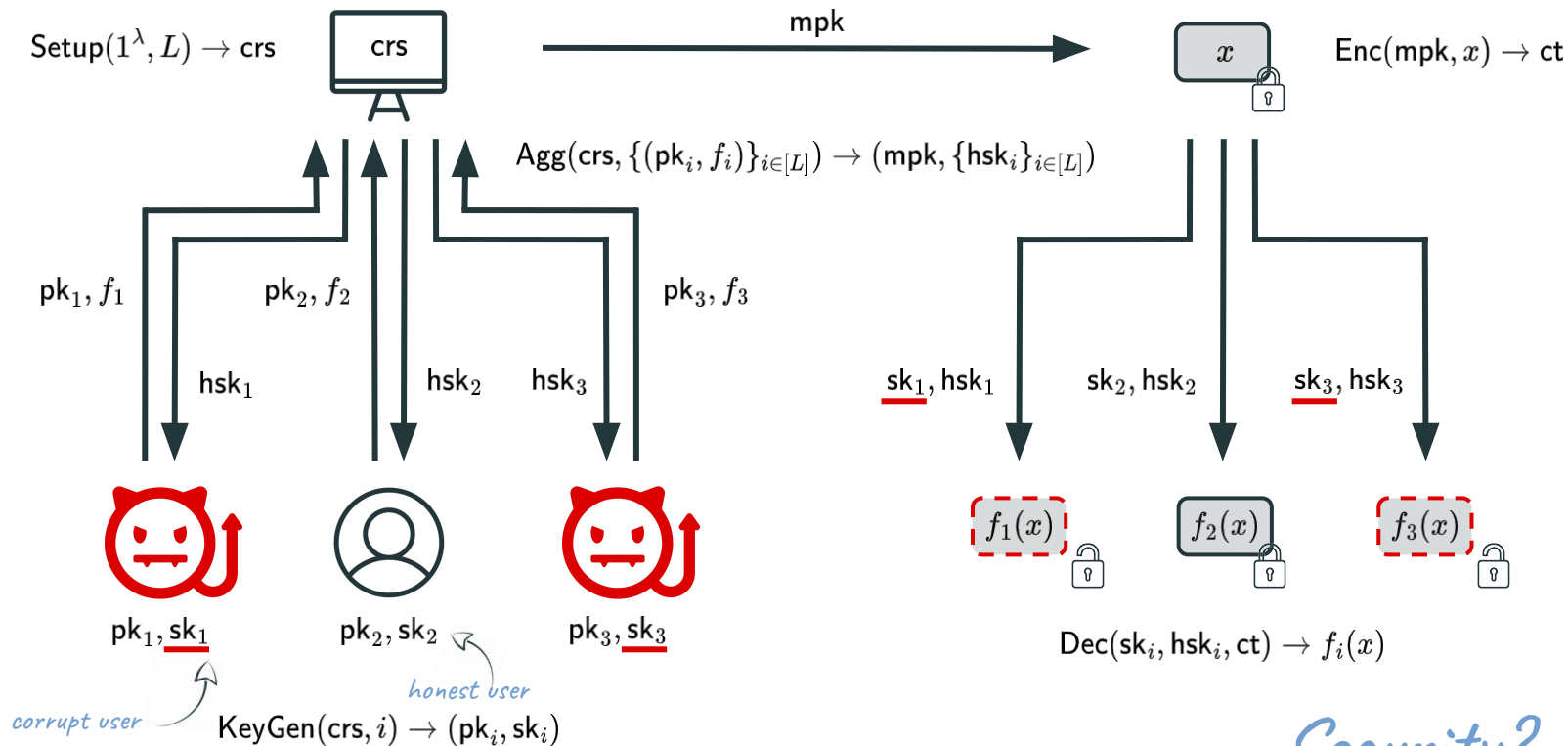
Slotted Registered Functional Encryption (sRFE)



$\text{KeyGen}(\text{crs}, i) \rightarrow (\text{pk}_i, \text{sk}_i)$

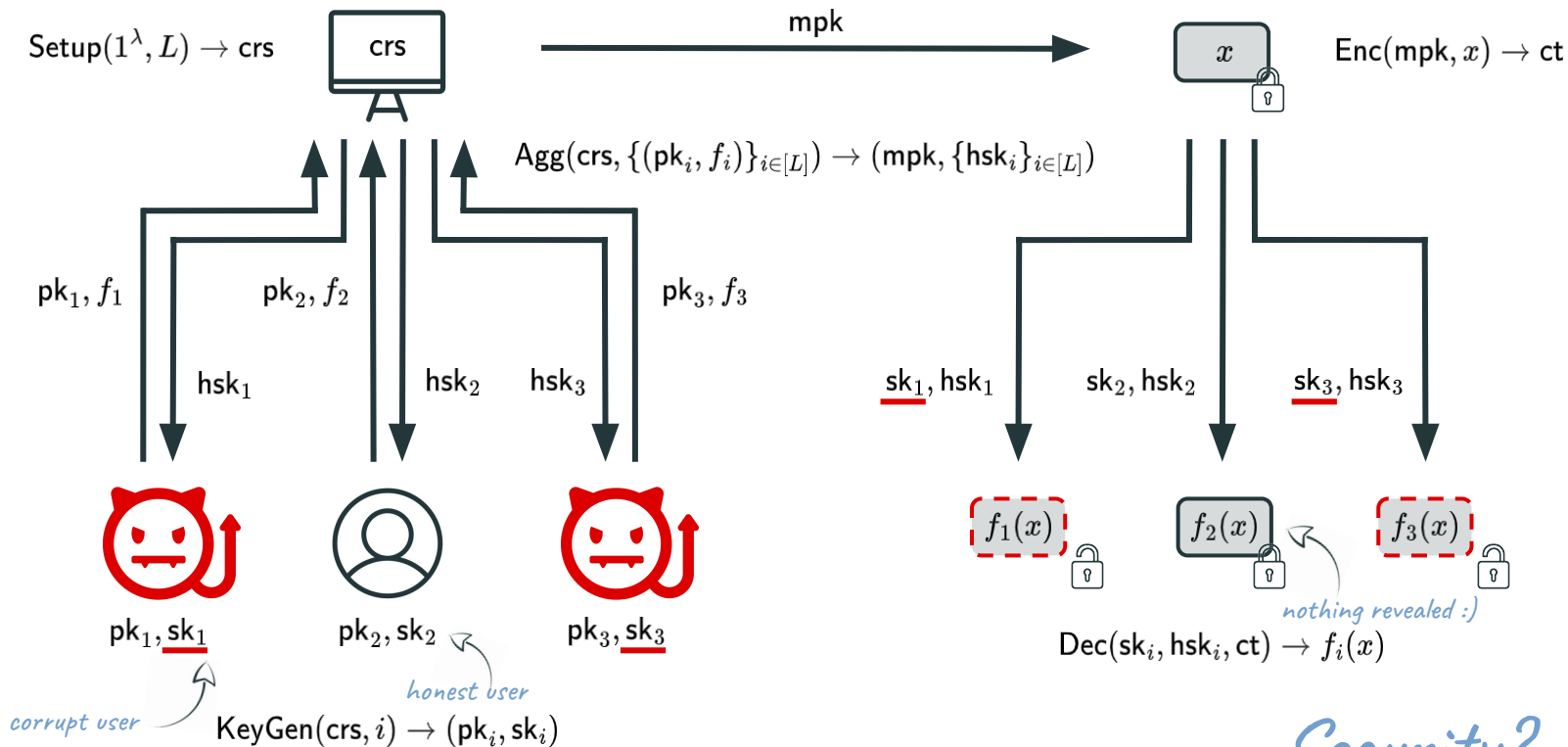
Security?

Slotted Registered Functional Encryption (sRFE)



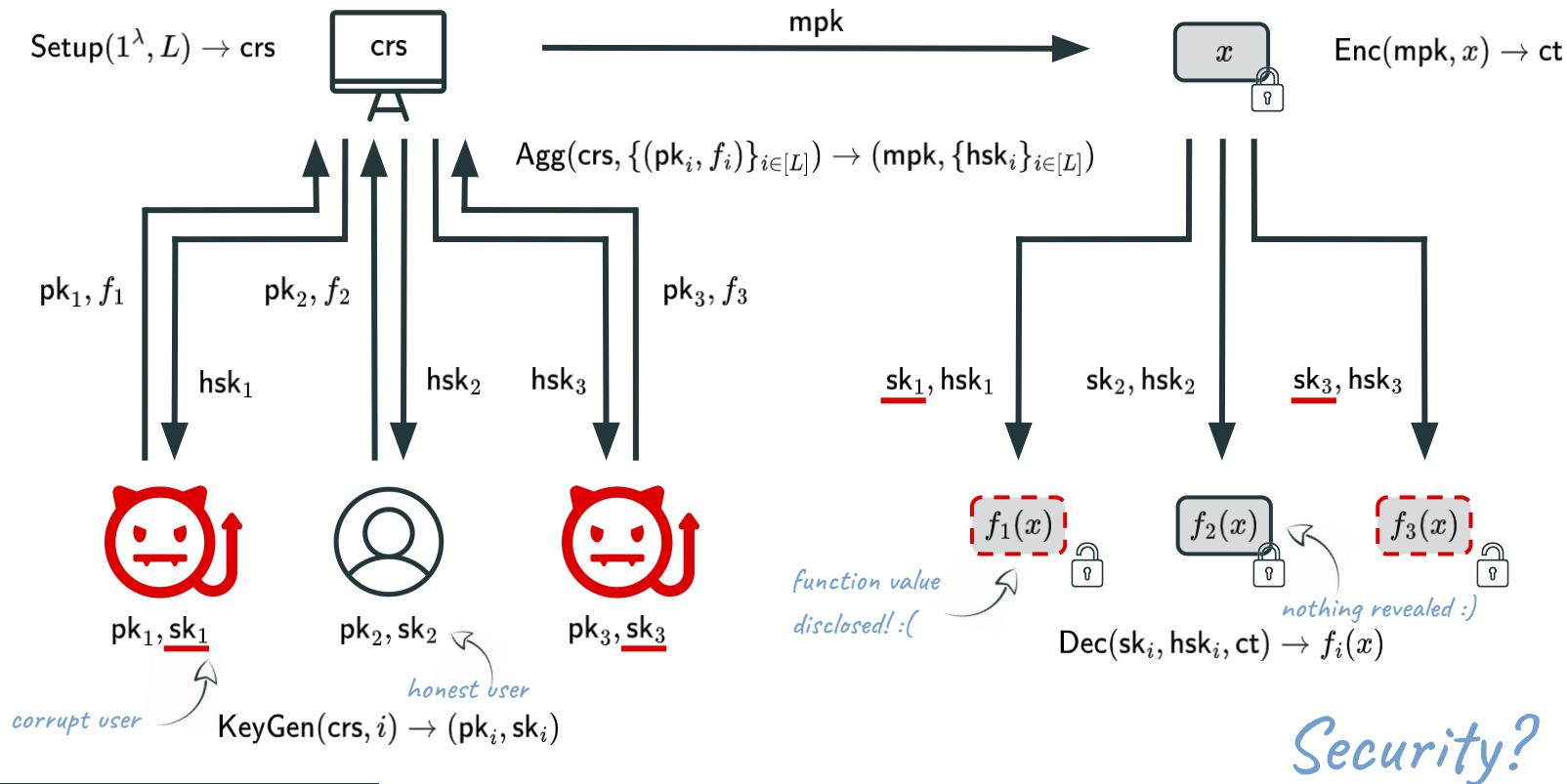
Security?

Slotted Registered Functional Encryption (sRFE)

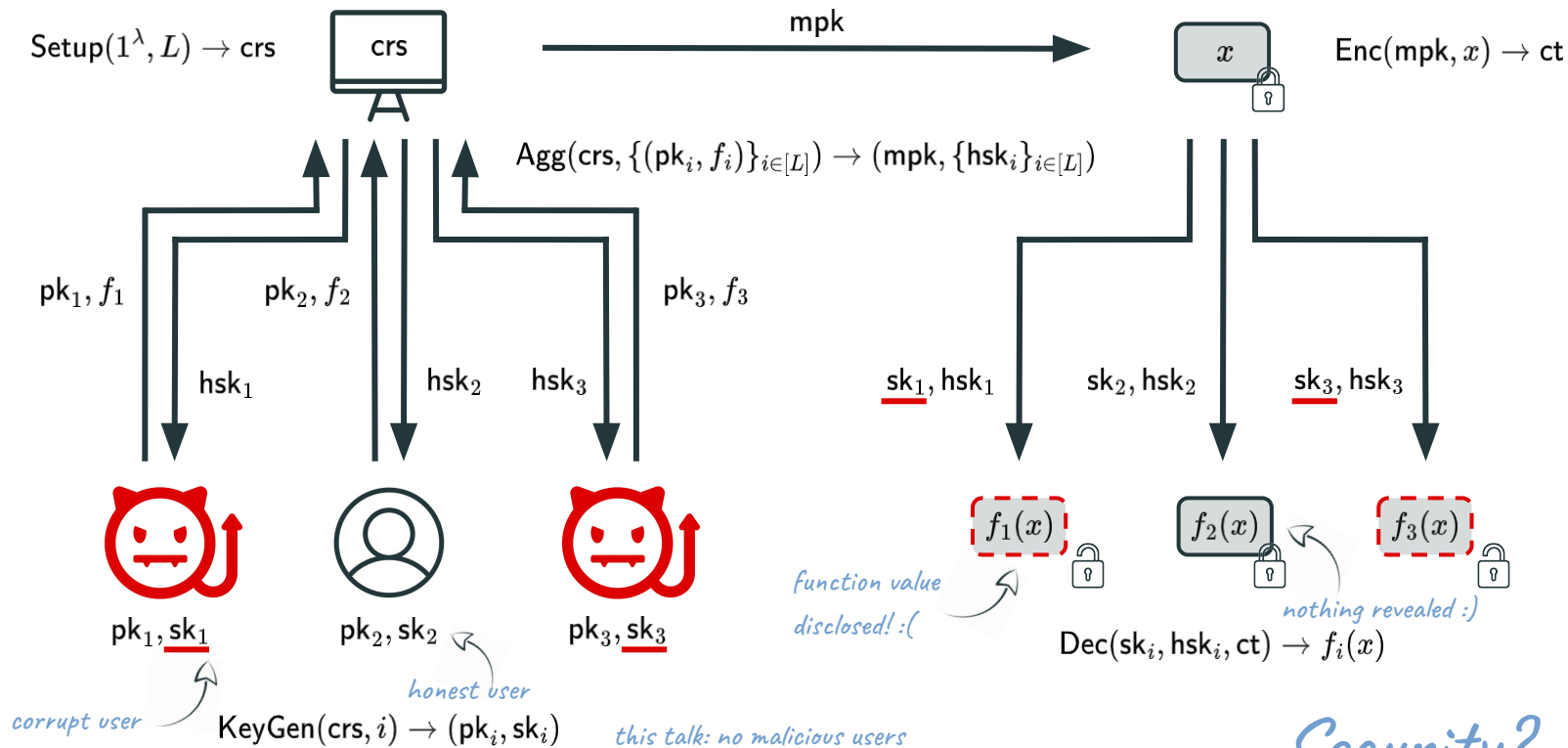


Security?

Slotted Registered Functional Encryption (sRFE)



Slotted Registered Functional Encryption (sRFE)



Security?

Existing RFE beyond Predicates

Work	Function Class	Assumption	Remarks
[AC:FFM ⁺ 23, AC:DPY24]	general	iO, SSB	
[AC:DPY24]	AB-IP	GGM	LSSS access policies
[AC:BLM ⁺ 24]	IP, weak QF	q-type	

Existing RFE beyond Predicates

Work	Function Class	Assumption	Remarks
[AC:FFM ⁺ 23, AC:DPY24]	general	iO, SSB	
[AC:DPY24]	AB-IP	GGM	LSSS access policies
[AC:BLM ⁺ 24]	IP, weak QF	q-type	
[EC:ZLZ ⁺ 24]	IP, QF	bilateral MDDH	

Existing RFE beyond Predicates

Work	Function Class	Assumption	Remarks
[AC:FFM ⁺ 23, AC:DPY24]	general	iO, SSB	
[AC:DPY24]	AB-IP	GGM	LSSS access policies
[AC:BLM ⁺ 24]	IP, weak QF	q-type	
[EC:ZLZ ⁺ 24]	IP, QF	bilateral MDDH	
[this work]	AB-AWS	bilateral MDDH	ABP access policies



attribute-based attribute-weighted sums (see next slide)

Attribute-Weighted Sums [C:AGW20]

- inner product (IP) *[EC:ZLZ+24]*

$$f(\mathbf{z}) = \mathbf{z} \cdot \mathbf{c}^\top$$

Attribute-Weighted Sums [C:AGW20]

- inner product (IP) *[EC:ZLZ+24]*

$$f(\mathbf{z}) = \mathbf{z} \cdot \mathbf{c}^\top$$

*variable coefficient vectors
(computable by ABP)*

- 1-input attribute-weighted sum (1AWS)

$$f(\mathbf{x}, \mathbf{z}) = \mathbf{z} \cdot h(\mathbf{x})^\top$$



Attribute-Weighted Sums [C:AGW20]

- inner product (IP) *[EC:ZLZ+24]*

$$f(\mathbf{z}) = \mathbf{z} \cdot \mathbf{c}^\top$$

*variable coefficient vectors
(computable by ABP)*

- 1-input attribute-weighted sum (1AWS)

$$f(\mathbf{x}, \mathbf{z}) = \mathbf{z} \cdot h(\mathbf{x})^\top$$

unbounded-size data sets

- (unbounded-input) attribute-weighted sum (AWS)

$$f(\{(\mathbf{x}_j, \mathbf{z}_j)\}_{j \in [N]}) = \sum_{j \in [N]} \mathbf{z}_j \cdot h(\mathbf{x}_j)^\top$$

Attribute-Weighted Sums [C:AGW20]

- inner product (IP) *[EC:ZLZ+24]*

$$f(\mathbf{z}) = \mathbf{z} \cdot \mathbf{c}^\top$$

*variable coefficient vectors
(computable by ABP)*

- 1-input attribute-weighted sum (1AWS)

$$f(\mathbf{x}, \mathbf{z}) = \mathbf{z} \cdot h(\mathbf{x})^\top$$

unbounded-size data sets

- (unbounded-input) attribute-weighted sum (AWS)

$$f(\{(\mathbf{x}_j, \mathbf{z}_j)\}_{j \in [N]}) = \sum_{j \in [N]} \mathbf{z}_j \cdot h(\mathbf{x}_j)^\top$$

- attribute-based attribute-weighted sum (AB-AWS)

$$f(\mathbf{y}, \{(\mathbf{x}_j, \mathbf{z}_j)\}_{j \in [N]}) = \begin{cases} \sum_{j \in [N]} \mathbf{z}_j \cdot h(\mathbf{x}_j)^\top & \text{if } g(\mathbf{y}) = 0 \\ \perp & \text{if } g(\mathbf{y}) \neq 0 \end{cases}$$

fine-grained access control

Inner Product Functional Encryption (IPFE) [PKC:ABDP15, C:ALS16]

- **setup:** sample random matrices \mathbf{A} , \mathbf{W} and define $\mathbf{mpk} = ([\mathbf{A}], [\mathbf{A}\mathbf{W}])$, $\mathbf{msk} = \mathbf{W}$

Inner Product Functional Encryption (IPFE) [PKC:ABDP15, C:ALS16]

- **setup:** sample random matrices \mathbf{A} , \mathbf{W} and define $\text{mpk} = ([\mathbf{A}], [\mathbf{A}\mathbf{W}])$, $\text{msk} = \mathbf{W}$
- **encryption:** to encrypt \mathbf{z} , sample random vector \mathbf{s} and output $\text{ct} = ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\mathbf{W}])$

Inner Product Functional Encryption (IPFE) [PKC:ABDP15, C:ALS16]

- **setup:** sample random matrices \mathbf{A} , \mathbf{W} and define $\text{mpk} = ([\mathbf{A}], [\mathbf{A}\mathbf{W}])$, $\text{msk} = \mathbf{W}$
- **encryption:** to encrypt \mathbf{z} , sample random vector \mathbf{s} and output $\text{ct} = ([\mathbf{s}\mathbf{A}], [\mathbf{z} - \mathbf{s}\mathbf{A}\mathbf{W}])$
- **key generation:** to generate a key for \mathbf{y} , output $\text{sk}_{\mathbf{y}} = \mathbf{d}^{\top} := \mathbf{W}\mathbf{y}^{\top}$
- **decryption:** output $[\mathbf{c}_1]\mathbf{d}^{\top} + [\mathbf{c}_2]\mathbf{y}^{\top} = [\mathbf{z}\mathbf{y}^{\top}]$

Inner Product Functional Encryption (IPFE) [PKC:ABDP15, C:ALS16]

- **setup:** sample random matrices \mathbf{A} , \mathbf{W} and define $\text{mpk} = ([\mathbf{A}], [\mathbf{A}\mathbf{W}])$, $\text{msk} = \mathbf{W}$
- **encryption:** to encrypt \mathbf{z} , sample random vector \mathbf{s} and output $\text{ct} = ([\mathbf{s}\mathbf{A}], [\mathbf{z} - \mathbf{s}\mathbf{A}\mathbf{W}])$

 $[c_1] :=$ $[c_2]$
- **key generation:** to generate a key for \mathbf{y} , output $\text{sk}_{\mathbf{y}} = \mathbf{d}^{\top} := \mathbf{W}\mathbf{y}^{\top}$
or a matrix \mathbf{Y} *(in which case the secret key is* $\text{sk}_{\mathbf{Y}} = \mathbf{D} := \mathbf{W}\mathbf{Y}$ *)*
- **decryption:** output $[c_1]\mathbf{d}^{\top} + [c_2]\mathbf{y}^{\top} = [\mathbf{z}\mathbf{y}^{\top}]$ *(or $[c_1]\mathbf{D} + [c_2]\mathbf{Y} = [\mathbf{z}\mathbf{Y}]$)*

Partial Garbling for 1AWS [ICALP:IW14]

- **garbling**: given an ABP h and public input \mathbf{x} , compute matrix $\mathbf{L}_{\mathbf{x}}$, sample randomness \mathbf{w} , and output

$$\text{pgb}(h, \mathbf{x}, \mathbf{z}; \mathbf{w}) = (\mathbf{p}_1, \mathbf{p}_2) := (\mathbf{z} - \mathbf{w}, \mathbf{w}\mathbf{L}_{\mathbf{x}})$$

Partial Garbling for 1AWS [ICALP:IW14]

- **garbling**: given an ABP h and public input \mathbf{x} , compute matrix \mathbf{L}_x , sample randomness \mathbf{w} , and output

$$\text{pgb}(h, \mathbf{x}, \mathbf{z}; \mathbf{w}) = (\mathbf{p}_1, \mathbf{p}_2) := (\mathbf{z} - \mathbf{w}, \mathbf{w}\mathbf{L}_x)$$

some subvector

Partial Garbling for 1AWS [ICALP:IW14]

- **garbling:** given an ABP h and public input \mathbf{x} , compute matrix $\mathbf{L}_{\mathbf{x}}$, sample randomness \mathbf{w} , and output

$$\text{pgb}(h, \mathbf{x}, \mathbf{z}; \mathbf{w}) = (\mathbf{p}_1, \mathbf{p}_2) := (\mathbf{z} - \mathbf{w}, \mathbf{w}\mathbf{L}_{\mathbf{x}})$$

- **reconstruction:** given (h, \mathbf{x}) , find vector $\mathbf{d}_{h,\mathbf{x}}$ such that

$$(\mathbf{p}_1, \mathbf{p}_2) \cdot \mathbf{d}_{h,\mathbf{x}}^\top = \mathbf{z} \cdot h(\mathbf{x})^\top$$

 *some subvector*

Partial Garbling for 1AWS [ICALP:IW14]

- **garbling:** given an ABP h and public input \mathbf{x} , compute matrix \mathbf{L}_x , sample randomness \mathbf{w} , and output

$$\text{pgb}(h, \mathbf{x}, \mathbf{z}; \mathbf{w}) = (\mathbf{p}_1, \mathbf{p}_2) := (\mathbf{z} - \underline{\mathbf{w}}, \mathbf{w}\mathbf{L}_x)$$

- **reconstruction:** given (h, \mathbf{x}) , find vector $\mathbf{d}_{h,\mathbf{x}}$ such that

$$(\mathbf{p}_1, \mathbf{p}_2) \cdot \mathbf{d}_{h,\mathbf{x}}^\top = \mathbf{z} \cdot h(\mathbf{x})^\top$$

 *some subvector*

- **privacy:** for random \mathbf{w} , the following distributions are indistinguishable

$$\{(\mathbf{z} - \underline{\mathbf{w}}, \mathbf{w}\mathbf{L}_x)\} \approx_s \{(-\underline{\mathbf{w}}, \mathbf{w}\mathbf{L}_x + \mathbf{z}h(\mathbf{x})^\top \cdot \mathbf{e}_1)\}$$

Combining the Two — Classical FE for 1AWS

$$\text{FE.ct} \quad ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}])$$

$$\text{FE.sk}_{h,\mathbf{x}} \quad \mathbf{WL}_{\mathbf{x}}$$

Reminder.

- ALS IFPE:

$$\text{ct} = ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}]) , \quad \text{sk}_Y = \mathbf{D} := \mathbf{WY}$$

- partial garbling for 1AWS:

$$\text{pgb}(h, \mathbf{x}, \mathbf{z}; \mathbf{w}) = (\mathbf{p}_1, \mathbf{p}_2) := (\mathbf{z} - \underline{\mathbf{w}}, \mathbf{wL}_{\mathbf{x}})$$

Combining the Two — Classical FE for 1AWS

$$\begin{array}{ll}
 \text{FE. ct} & ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}]) \\
 \text{FE. sk}_{h,\mathbf{x}} & \mathbf{WL}_{\mathbf{x}}
 \end{array}
 \left. \vphantom{\begin{array}{l} \text{FE. ct} \\ \text{FE. sk}_{h,\mathbf{x}} \end{array}} \right\} \rightarrow ([\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}], [\mathbf{sA}\mathbf{WL}_{\mathbf{x}}])$$

“variable random pad” $w = \mathbf{sA}\underline{\mathbf{W}}$
 $[p_r] :=$ *$:= [p_z]$*

Reminder.

- ALS IFPE:

$$\text{ct} = ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}]), \quad \text{sk}_Y = \mathbf{D} := \mathbf{WY}$$

- partial garbling for 1AWS:

$$\text{pgb}(h, \mathbf{x}, \mathbf{z}; \mathbf{w}) = (\mathbf{p}_1, \mathbf{p}_2) := (\mathbf{z} - \underline{\mathbf{w}}, \mathbf{wL}_{\mathbf{x}})$$

Combining the Two — Classical FE for 1AWS

note: this is not the actual 1AWS functionality

$$\begin{array}{ll}
 \text{FE. ct} & ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}]) \\
 \text{FE. sk}_{h,\mathbf{x}} & \mathbf{WL}_{\mathbf{x}}
 \end{array}
 \left. \vphantom{\begin{array}{l} \text{FE. ct} \\ \text{FE. sk}_{h,\mathbf{x}} \end{array}} \right\} \rightarrow ([\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}], [\mathbf{sA}\mathbf{W}\mathbf{L}_{\mathbf{x}}])$$

“variable random pad” $w = \mathbf{sA}\underline{\mathbf{W}}$
 $[p_r] :=$ (points to $[\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}]$)
 $:= [p_z]$ (points to $[\mathbf{sA}\mathbf{W}\mathbf{L}_{\mathbf{x}}]$)

Reminder.

- ALS IFPE:

$$\text{ct} = ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}])$$

$$\text{sk}_Y = \mathbf{D} := \mathbf{WY}$$

- partial garbling for 1AWS:

$$\text{pgb}(h, \mathbf{x}, \mathbf{z}; \mathbf{w}) = (\mathbf{p}_1, \mathbf{p}_2) := (\mathbf{z} - \underline{\mathbf{w}}, \mathbf{w}\mathbf{L}_{\mathbf{x}})$$

RFE for a Single User

$$\begin{array}{lcl}
 \text{FE. ct} & ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}]) & \\
 \text{FE. sk}_{h,\mathbf{x}} & \mathbf{WL}_{\mathbf{x}} &
 \end{array}
 \left. \vphantom{\begin{array}{l} \text{FE. ct} \\ \text{FE. sk}_{h,\mathbf{x}} \end{array}} \right\} \rightarrow \left([\mathbf{z} - \mathbf{sA}\underline{\mathbf{W}}], [\mathbf{sA}\mathbf{WL}_{\mathbf{x}}] \right)$$

“variable random pad” $w = \mathbf{sA}\underline{\mathbf{W}}$
 $[\mathbf{p}_r] :=$ *$:= [\mathbf{p}_z]$*

RFE for a Single User

crs

$$\underbrace{([\mathbf{A}], [\mathbf{A}\mathbf{W}])}_{FE.mpk}$$

$$\left. \begin{array}{ll} FE.ct & ([s\mathbf{A}], [z - s\mathbf{A}\mathbf{W}]) \\ FE.sk_{h,x} & \mathbf{WL}_x \end{array} \right\} \rightarrow ([z - s\mathbf{A}\mathbf{W}], [s\mathbf{A}\mathbf{W}\mathbf{L}_x])$$

“variable random pad” $w = s\mathbf{A}\mathbf{W}$
 $[p_r] :=$ *$:= [p_z]$*

RFE for a Single User

$$\text{crs} \quad \underbrace{([\mathbf{A}], [\mathbf{A}\mathbf{W}])}_{FE.mpk}$$

$$(\text{pk}, \text{sk}) \quad ([\mathbf{A}\mathbf{U}], \mathbf{U}) \quad (\text{for a random matrix } \mathbf{U})$$

$$\left. \begin{array}{ll} \text{FE. ct} & ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\mathbf{W}]) \\ \text{FE. sk}_{h,\mathbf{x}} & \mathbf{WL}_{\mathbf{x}} \end{array} \right\} \rightarrow ([\mathbf{z} - \mathbf{sA}\mathbf{W}], [\mathbf{sA}\mathbf{W}\mathbf{L}_{\mathbf{x}}])$$

“variable random pad” $w = \mathbf{sA}\mathbf{W}$
 $[p_r] :=$ *$:= [p_z]$*

RFE for a Single User

$$\begin{array}{ll}
 \text{crs} & (\underbrace{[\mathbf{A}], [\mathbf{A}\mathbf{W}]}_{FE.mpk}) \\
 (\text{pk}, \text{sk}) & ([\mathbf{A}\mathbf{U}], \mathbf{U}) \quad (\text{for a random matrix } \mathbf{U}) \\
 \text{mpk} & ([\mathbf{A}], [\mathbf{A}\mathbf{W}], [\mathbf{A}\mathbf{U} + \mathbf{A}\mathbf{W}\mathbf{L}_x]) \\
 \text{ct} & (\underbrace{[\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\mathbf{W}]}_{FE.ct}, \underbrace{[\mathbf{sA}\mathbf{U} + \mathbf{sA}\mathbf{W}\mathbf{L}_x]}_{Enc(pk, FE.sk_{h,x})})
 \end{array}$$

$$\left. \begin{array}{ll}
 FE.ct & ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\mathbf{W}]) \\
 FE.sk_{h,x} & \mathbf{W}\mathbf{L}_x
 \end{array} \right\} \rightarrow ([\mathbf{z} - \mathbf{sA}\mathbf{W}], [\mathbf{sA}\mathbf{W}\mathbf{L}_x])$$

"variable random pad" $w = \mathbf{sA}\mathbf{W}$
 $[p_r] :=$ \swarrow \nwarrow $:= [p_z]$

RFE for a Single User

$$\begin{array}{ll}
 \text{crs} & (\underbrace{[\mathbf{A}], [\mathbf{A}\mathbf{W}]}_{FE.mpk}) \\
 (\text{pk}, \text{sk}) & ([\mathbf{A}\mathbf{U}], \mathbf{U}) \quad (\text{for a random matrix } \mathbf{U}) \\
 \text{mpk} & ([\mathbf{A}], [\mathbf{A}\mathbf{W}], [\mathbf{A}\mathbf{U} + \mathbf{A}\mathbf{W}\mathbf{L}_x]) \\
 \text{ct} & (\underbrace{[\mathbf{s}\mathbf{A}], [\mathbf{z} - \mathbf{s}\mathbf{A}\mathbf{W}]}_{FE.ct}, \underbrace{[\mathbf{s}\mathbf{A}\mathbf{U} + \mathbf{s}\mathbf{A}\mathbf{W}\mathbf{L}_x]}_{Enc(pk, FE.sk_{h,x})})
 \end{array}$$

Security.

- 1) $sk=U$ is secret (i.e. user honest):
 \rightarrow nothing revealed under $MDDH_k$

$$\left. \begin{array}{ll}
 FE.ct & ([\mathbf{s}\mathbf{A}], [\mathbf{z} - \mathbf{s}\mathbf{A}\mathbf{W}]) \\
 FE.sk_{h,x} & \mathbf{W}\mathbf{L}_x
 \end{array} \right\} \rightarrow ([\mathbf{z} - \mathbf{s}\mathbf{A}\mathbf{W}], [\mathbf{s}\mathbf{A}\mathbf{W}\mathbf{L}_x])$$

"variable random pad" $w = \mathbf{s}\mathbf{A}\mathbf{W}$
 $[p_r] :=$ \swarrow \nwarrow $:= [p_z]$

RFE for a Single User

$$\begin{array}{ll}
 \text{crs} & (\underbrace{[\mathbf{A}], [\mathbf{A}\mathbf{W}]}_{FE.mpk}) \\
 (\text{pk}, \text{sk}) & ([\mathbf{A}\mathbf{U}], \mathbf{U}) \quad (\text{for a random matrix } \mathbf{U}) \\
 \text{mpk} & ([\mathbf{A}], [\mathbf{A}\mathbf{W}], [\mathbf{A}\mathbf{U} + \mathbf{A}\mathbf{W}\mathbf{L}_x]) \\
 \text{ct} & (\underbrace{[\mathbf{s}\mathbf{A}], [\mathbf{z} - \mathbf{s}\mathbf{A}\mathbf{W}]}_{FE.ct}, \underbrace{[\mathbf{s}\mathbf{A}\mathbf{U} + \mathbf{s}\mathbf{A}\mathbf{W}\mathbf{L}_x]}_{Enc(pk, FE.sk_{h,x})})
 \end{array}$$

Security.

- 1) $sk = \mathbf{U}$ is secret (i.e. user honest):
 \rightarrow nothing revealed under $MDDH_k$
- 2) $sk = \mathbf{U}$ known to A (i.e. user corrupted):
 \rightarrow only $zh(\mathbf{x})^T$ revealed under security of pgb

$$\left. \begin{array}{ll}
 FE.ct & ([\mathbf{s}\mathbf{A}], [\mathbf{z} - \mathbf{s}\mathbf{A}\mathbf{W}]) \\
 FE.sk_{h,x} & \mathbf{W}\mathbf{L}_x
 \end{array} \right\} \rightarrow ([\mathbf{z} - \mathbf{s}\mathbf{A}\mathbf{W}], [\mathbf{s}\mathbf{A}\mathbf{W}\mathbf{L}_x])$$

“variable random pad” $w = \mathbf{s}\mathbf{A}\mathbf{W}$
 $[p_r] :=$ $:= [p_z]$

RFE for Multiple Users

crs

(pk_i, sk_i)

mpk

ct

$$\left. \begin{array}{ll} \text{FE. ct} & ([sA], [z - sA\underline{W}]) \\ \text{FE. sk}_{h,x} & \underline{WL}_x \end{array} \right\} \rightarrow ([z - sA\underline{W}], [sA\underline{W}L_x])$$

“variable random pad” $w = sA\underline{W}$

$[p_r] :=$

$:= [p_z]$

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{\text{FE.mpk}}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

mpk

ct

$$\left. \begin{array}{ll} \text{FE. ct} & ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\mathbf{W}]) \\ \text{FE. sk}_{h,\mathbf{x}} & \mathbf{WL}_{\mathbf{x}} \end{array} \right\} \rightarrow ([\mathbf{z} - \mathbf{sA}\mathbf{W}], [\mathbf{sA}\mathbf{W}\mathbf{L}_{\mathbf{x}}])$$

“variable random pad” $w = \mathbf{sA}\mathbf{W}$
 $[\mathbf{p}_r] :=$ $:= [\mathbf{p}_z]$

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i], \sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}])$$

$$\text{ct} \quad ([\mathbf{sA}], [\mathbf{z} - \sum_{i \in [L]} \mathbf{sA}\mathbf{W}_i], \sum_{i \in [L]} [\mathbf{sA}\mathbf{U}_i + \mathbf{sA}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}])$$

↑
sum of L independent 1-slot instances

$$\left. \begin{array}{ll} \text{FE. ct} & ([\mathbf{sA}], [\mathbf{z} - \mathbf{sA}\mathbf{W}]) \\ \text{FE. sk}_{h,\mathbf{x}} & \mathbf{W}\mathbf{L}_{\mathbf{x}} \end{array} \right\} \rightarrow \begin{array}{l} \text{"variable random pad" } w = \mathbf{sA}\mathbf{W} \\ ([\mathbf{z} - \mathbf{sA}\mathbf{W}], [\mathbf{sA}\mathbf{W}\mathbf{L}_{\mathbf{x}}]) \\ \uparrow \quad \quad \quad \uparrow \\ [p_r] := \quad \quad \quad := [p_z] \end{array}$$


RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i]}, \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$

$$\text{ct} \quad ([\mathbf{sA}], [\mathbf{z} - \underline{\sum_{i \in [L]} \mathbf{sA}\mathbf{W}_i}], \underline{\sum_{i \in [L]} [\mathbf{sA}\mathbf{U}_i + \mathbf{sA}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$



 sum of L independent 1-slot instances

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i]}, \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$

$$\text{ct} \quad ([\mathbf{sA}], [\mathbf{z} - \underline{\sum_{i \in [L]} \mathbf{sA}\mathbf{W}_i}], \underline{\sum_{i \in [L]} [\mathbf{sA}\mathbf{U}_i + \mathbf{sA}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$



sum of L independent 1-slot instances



... how to decrypt? -> helper secret keys

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i]}, \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$

$$\text{ct} \quad ([\mathbf{sA}], [\mathbf{z} - \underline{\sum_{i \in [L]} \mathbf{sA}\mathbf{W}_i}], \underline{\sum_{i \in [L]} [\mathbf{sA}\mathbf{U}_i + \mathbf{sA}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$



sum of L independent 1-slot instances

... how to decrypt? \rightarrow helper secret keys

Intuition.

- user i could decrypt given $\text{hsk}_i = (\sum_{j \in [L] \setminus i} \mathbf{W}_j, \sum_{j \in [L] \setminus i} \mathbf{U}_j)$

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i]}, \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$

$$\text{ct} \quad ([\mathbf{s}\mathbf{A}], [\mathbf{z} - \underline{\sum_{i \in [L]} \mathbf{s}\mathbf{A}\mathbf{W}_i}], \underline{\sum_{i \in [L]} [\mathbf{s}\mathbf{A}\mathbf{U}_i + \mathbf{s}\mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$



sum of L independent 1-slot instances



... how to decrypt? -> helper secret keys

Intuition.

- user i could decrypt given $\text{hsk}_i = (\sum_{j \in [L] \setminus i} \mathbf{W}_j, \sum_{j \in [L] \setminus i} \mathbf{U}_j)$
- **problem 1:** helper secret key contains scalar values

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \underbrace{\sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i]}, \underbrace{\sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$

$$\text{ct} \quad ([\mathbf{sA}], [\mathbf{z} - \underbrace{\sum_{i \in [L]} \mathbf{sA}\mathbf{W}_i}, \underbrace{\sum_{i \in [L]} [\mathbf{sA}\mathbf{U}_i + \mathbf{sA}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$



sum of L independent 1-slot instances



... how to decrypt? -> helper secret keys

Intuition.

- user i could decrypt given $\text{hsk}_i = (\sum_{j \in [L] \setminus i} \mathbf{W}_j, \sum_{j \in [L] \setminus i} \mathbf{U}_j)$
- **problem 1:** helper secret key contains scalar values
- **solution 1:** switch to pairing group with ciphertexts in \mathbb{G}_1 and helper secret keys in \mathbb{G}_2

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i]}, \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$

$$\text{ct} \quad ([\mathbf{sA}], [\mathbf{z} - \underline{\sum_{i \in [L]} \mathbf{sA}\mathbf{W}_i}], \underline{\sum_{i \in [L]} [\mathbf{sA}\mathbf{U}_i + \mathbf{sA}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$



sum of L independent 1-slot instances

... how to decrypt? \rightarrow helper secret keys

Intuition.

- user i could decrypt given $\text{hsk}_i = (\sum_{j \in [L] \setminus i} [\mathbf{W}_j]_2, \sum_{j \in [L] \setminus i} [\mathbf{U}_j]_2)$

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i]}, \underline{\sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$

$$\text{ct} \quad ([\mathbf{sA}], [\mathbf{z} - \underline{\sum_{i \in [L]} \mathbf{sA}\mathbf{W}_i}], \underline{\sum_{i \in [L]} [\mathbf{sA}\mathbf{U}_i + \mathbf{sA}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$



sum of L independent 1-slot instances



... how to decrypt? -> helper secret keys

Intuition.

- user i could decrypt given $\text{hsk}_i = (\sum_{j \in [L] \setminus i} [\mathbf{W}_j]_2, \sum_{j \in [L] \setminus i} [\mathbf{U}_j]_2)$
- **problem 2:** masking terms for different users are correlated

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \underbrace{\sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i]}, \underbrace{\sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$

$$\text{ct} \quad ([\mathbf{sA}], [\mathbf{z} - \underbrace{\sum_{i \in [L]} \mathbf{sA}\mathbf{W}_i}, \underbrace{\sum_{i \in [L]} [\mathbf{sA}\mathbf{U}_i + \mathbf{sA}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$



sum of L independent 1-slot instances



... how to decrypt? -> helper secret keys

Intuition.

- user i could decrypt given $\text{hsk}_i = (\sum_{j \in [L] \setminus i} [\mathbf{W}_j]_2, \sum_{j \in [L] \setminus i} [\mathbf{U}_j]_2)$
- **problem 2:** masking terms for different users are correlated
- **(partial) solution 2:** user-specific re-randomization of helper secret keys

RFE for Multiple Users

$$\text{crs} \quad \underbrace{([\mathbf{A}], \{[\mathbf{A}\mathbf{W}_i]\}_{i \in [L]})}_{FE.mpk}$$

$$(\text{pk}_i, \text{sk}_i) \quad ([\mathbf{A}\mathbf{U}_i], \mathbf{U}_i) \quad (\text{for random matrices } \mathbf{U}_i)$$

$$\text{mpk} \quad ([\mathbf{A}], \underbrace{\sum_{i \in [L]} [\mathbf{A}\mathbf{W}_i]}, \underbrace{\sum_{i \in [L]} [\mathbf{A}\mathbf{U}_i + \mathbf{A}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$

$$\text{ct} \quad ([\mathbf{sA}], [\mathbf{z} - \underbrace{\sum_{i \in [L]} \mathbf{sA}\mathbf{W}_i}, \underbrace{\sum_{i \in [L]} [\mathbf{sA}\mathbf{U}_i + \mathbf{sA}\mathbf{W}_i \mathbf{L}_{i,\mathbf{x}}]})$$


sum of L independent 1-slot instances
... how to decrypt? → helper secret keys


Intuition.

- user i could decrypt given $\text{hsk}_i = (\sum_{j \in [L] \setminus i} [\mathbf{W}_j]_2, \sum_{j \in [L] \setminus i} [\mathbf{U}_j]_2)$
- **problem 2:** masking terms for different users are correlated
- **(partial) solution 2:** user-specific re-randomization of helper secret keys

$$\text{hsk}_i = ([\mathbf{B}\mathbf{r}_j^\top]_2, \sum_{j \in [L] \setminus i} [\mathbf{W}_j \mathbf{B}\mathbf{r}_j^\top]_2, \sum_{j \in [L] \setminus i} [\mathbf{U}_j \mathbf{B}\mathbf{r}_j^\top]_2)$$

Pad Re-Randomization

ciphertext *helper secret key*


$$\begin{aligned} [\mathbf{p}_1]_1 \cdot [\mathbf{R}]_2 &= [\mathbf{z} - \mathbf{sAW}]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} \cdot \mathbf{R} - \mathbf{sAW} \cdot \mathbf{R}]_t \\ [\mathbf{p}_2]_1 \cdot [\mathbf{R}]_2 &= [\mathbf{sAWL}_x]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x \cdot \mathbf{R}]_t \end{aligned}$$

Question: how to choose \mathbf{R} ?

- **naive approach:** a random (uniform) matrix

Pad Re-Randomization

ciphertext *helper secret key* *problem 1: input vector changes*

$$[\mathbf{p}_1]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} - \mathbf{sAW}]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} \cdot \mathbf{R} - \mathbf{sAW} \cdot \mathbf{R}]_t$$
$$[\mathbf{p}_2]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x \cdot \mathbf{R}]_t$$

Question: how to choose \mathbf{R} ?

- **naive approach:** a random (uniform) matrix

Pad Re-Randomization

ciphertext *helper secret key* *problem 1: input vector changes*

$$[\mathbf{p}_1]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} - \mathbf{sAW}]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} \cdot \mathbf{R} - \mathbf{sAW} \cdot \mathbf{R}]_t$$
$$[\mathbf{p}_2]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x \cdot \mathbf{R}]_t$$

*problem 2: correctly randomized encoding
should be $\mathbf{sAWR} \cdot \mathbf{L}_x$*

Question: how to choose \mathbf{R} ?

- **naive approach:** a random (uniform) matrix

Pad Re-Randomization

ciphertext *helper secret key* *problem 1: input vector changes*

$$[\mathbf{p}_1]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} - \mathbf{sAW}]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} \cdot \mathbf{R} - \mathbf{sAW} \cdot \mathbf{R}]_t$$
$$[\mathbf{p}_2]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x \cdot \mathbf{R}]_t$$

problem 3: what is \mathbf{R} ?

*problem 2: correctly randomized encoding
should be $\mathbf{sAWR} \cdot \mathbf{L}_x$*

Question: how to choose \mathbf{R} ?

- **naive approach:** a random (uniform) matrix

Pad Re-Randomization

ciphertext *helper secret key* *problem 1: input vector changes*

$$[\mathbf{p}_1]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} - \mathbf{sAW}]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} \cdot \mathbf{R} - \mathbf{sAW} \cdot \mathbf{R}]_t$$

$$[\mathbf{p}_2]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x \cdot \mathbf{R}]_t$$

problem 3: what is \mathbf{R} ? $\rightarrow \mathbf{R} = (\mathbf{I} \otimes \mathbf{r}^T)$

problem 2: correctly randomized encoding
should be $\mathbf{sAWR} \cdot \mathbf{L}_x$
 $\rightarrow \mathbf{sAWL}_x \cdot (\mathbf{I} \otimes \mathbf{r}^T) = \mathbf{sAW}(\mathbf{I} \otimes \mathbf{r}^T) \cdot \mathbf{L}_x$

Question: how to choose \mathbf{R} ?

- **naive approach:** a random (uniform) matrix
- **solution 1:** $\mathbf{R} = (\mathbf{I} \otimes \mathbf{r}^T)$ for $\mathbf{r} \leftarrow_{\$} \mathbf{Z}_p^k$ (tensored ALS encodings)

Pad Re-Randomization

ciphertext helper secret key

$$\begin{aligned}
 [\mathbf{p}_1]_1 \cdot [\mathbf{R}]_2 &= [\mathbf{z} - \mathbf{sAW}]_1 \cdot [\mathbf{R}]_2 = [\mathbf{z} \cdot \mathbf{R} - \mathbf{sAW} \cdot \mathbf{R}]_t \\
 [\mathbf{p}_2]_1 \cdot [\mathbf{R}]_2 &= [\mathbf{sAWL}_x]_1 \cdot [\mathbf{R}]_2 = [\mathbf{sAWL}_x \cdot \mathbf{R}]_t
 \end{aligned}$$

problem 1: input vector changes \rightarrow encode $\mathbf{z} \otimes \mathbf{sA}$ and decode in new basis \mathbf{sAR}

problem 3: what is \mathbf{R} ? $\rightarrow \mathbf{R} = (\mathbf{I} \otimes \mathbf{r}^T)$

problem 2: correctly randomized encoding should be $\mathbf{sAWR} \cdot \mathbf{L}_x$
 $\rightarrow \mathbf{sAWL}_x \cdot (\mathbf{I} \otimes \mathbf{r}^T) = \mathbf{sAW}(\mathbf{I} \otimes \mathbf{r}^T) \cdot \mathbf{L}_x$

Question: how to choose \mathbf{R} ?

- naive approach: a random (uniform) matrix
- solution 1: $\mathbf{R} = (\mathbf{I} \otimes \mathbf{r}^T)$ for $\mathbf{r} \leftarrow_{\$} \mathbf{Z}_p^k$ (tensored ALS encodings)

Pad Re-Randomization

ciphertext \rightarrow $[p_1]_1$ *helper secret key* \rightarrow $[R]_2$

$$[p_1]_1 \cdot [R]_2 = [z - sAW]_1 \cdot [R]_2 = [z \cdot R - sAW \cdot R]_t$$

$$[p_2]_1 \cdot [R]_2 = [sAWL_x]_1 \cdot [R]_2 = [sAWL_x \cdot R]_t$$

problem 1: input vector changes \rightarrow encode $z \otimes sA$ and decode in new basis sAR

problem 3: what is R ? $\rightarrow R = (I \otimes r^T)$

problem 2: correctly randomized encoding should be $sAWR \cdot L_x$
 $\rightarrow sAWL_x \cdot (I \otimes r^T) = sAW(I \otimes r^T) \cdot L_x$

Question: how to choose R ?

- **naive approach:** a random (uniform) matrix
- **solution 1:** $R = (I \otimes r^T)$ for $r \leftarrow_{\$} \mathbb{Z}_p^k$ (tensored ALS encodings)
- **solution 2:** use different ALS keys (nested ALS encodings)

Solution 2: Nested ALS Encodings

ciphertext *helper secret key*

$$\begin{aligned} [\mathbf{p}_{1,\text{in}}]_t &= [\mathbf{z} - \mathbf{sA}\mathbf{W}_{\text{in}}]_1 \cdot [\mathbf{I}]_2 = [\mathbf{z} - \mathbf{sA}\mathbf{W}_{\text{in}}]_t \\ [\mathbf{p}_{1,\text{out}}]_t &= [\mathbf{sA}]_1 \cdot [\mathbf{W}_{\text{in}} - \mathbf{W}_{\text{out}}]_2 = [\mathbf{sA}\mathbf{W}_{\text{in}} - \mathbf{sA}\mathbf{W}_{\text{out}}]_t \\ [\mathbf{p}_2]_t &= [\mathbf{sA}]_1 \cdot [\mathbf{W}_{\text{out}}\mathbf{L}_{\mathbf{x}}]_2 = [\mathbf{sA}\mathbf{W}_{\text{out}}\mathbf{L}_{\mathbf{x}}]_t \end{aligned} \quad \left. \vphantom{\begin{aligned} [\mathbf{p}_{1,\text{in}}]_t &= [\mathbf{z} - \mathbf{sA}\mathbf{W}_{\text{in}}]_1 \cdot [\mathbf{I}]_2 = [\mathbf{z} - \mathbf{sA}\mathbf{W}_{\text{in}}]_t \\ [\mathbf{p}_{1,\text{out}}]_t &= [\mathbf{sA}]_1 \cdot [\mathbf{W}_{\text{in}} - \mathbf{W}_{\text{out}}]_2 = [\mathbf{sA}\mathbf{W}_{\text{in}} - \mathbf{sA}\mathbf{W}_{\text{out}}]_t \end{aligned}} \right\} \rightarrow \begin{aligned} [\mathbf{p}_1]_t &= [\mathbf{p}_{1,\text{in}}]_t + [\mathbf{p}_{1,\text{out}}]_t = [\mathbf{z} - \mathbf{sA}\mathbf{W}_{\text{out}}]_t \\ [\mathbf{p}_2]_t &= [\mathbf{sA}\mathbf{W}_{\text{out}}\mathbf{L}_{\mathbf{x}}]_t \end{aligned}$$

Conclusion

- **classical FE** provides security against malicious user but needs to trust authority
- **registered FE** circumvents the need for trusted authority

Conclusion

- **classical FE** provides security against malicious user but needs to trust authority
- **registered FE** circumvents the need for trusted authority
- **this work:**
 - RFE for **1AWS** with adaptive security using **tensorized ALS encodings**
 - RFE for **AB-AWS** with selective security using **nested ALS encodings**

Conclusion

- **classical FE** provides security against malicious user but needs to trust authority
- **registered FE** circumvents the need for trusted authority
- **this work:**
 - RFE for **1AWS** with adaptive security using **tensorized ALS encodings**
 - RFE for **AB-AWS** with selective security using **nested ALS encodings**
- **follow-up work:**
 - **modular framework** (pre-constrained IP-RFE + garbling scheme)
 - **new functionalities** (AB-AWS and AB-QF for log-space TMs)

Conclusion

- **classical FE** provides security against malicious user but needs to trust authority
- **registered FE** circumvents the need for trusted authority
- **this work:**
 - RFE for **1AWS** with adaptive security using **tensorized ALS encodings**
 - RFE for **AB-AWS** with selective security using **nested ALS encodings**
- **follow-up work:**
 - **modular framework** (pre-constrained IP-RFE + garbling scheme)
 - **new functionalities** (AB-AWS and AB-QF for log-space TMs)
- **open problems:**
 - adaptive security
 - compression of CRS

Conclusion

- **classical FE** provides security against malicious user but needs to trust authority
- **registered FE** circumvents the need for trusted authority
- **this work:**
 - RFE for **1AWS** with adaptive security using **tensorized ALS encodings**
 - RFE for **AB-AWS** with selective security using **nested ALS encodings**
- **follow-up work:**
 - **modular framework** (pre-constrained IP-RFE + garbling scheme)
 - **new functionalities** (AB-AWS and AB-QF for log-space TMs)
- **open problems:**
 - adaptive security
 - compression of CRS

Thank you!!! :)