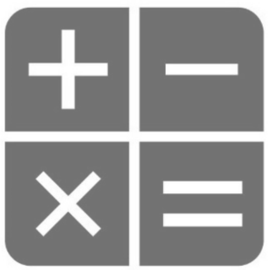
Decentralized Evaluation of Quadratic Polynomials on Encrypted Data

Chloé Hébant, Duong Hieu Phan and David Pointcheval

Cloud Computing

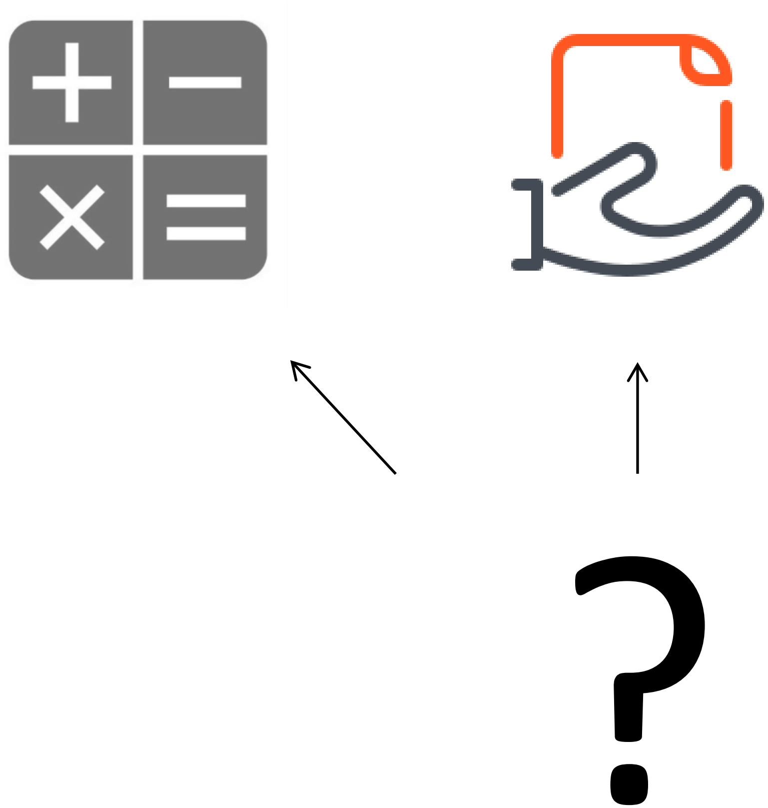
?

Cloud Computing

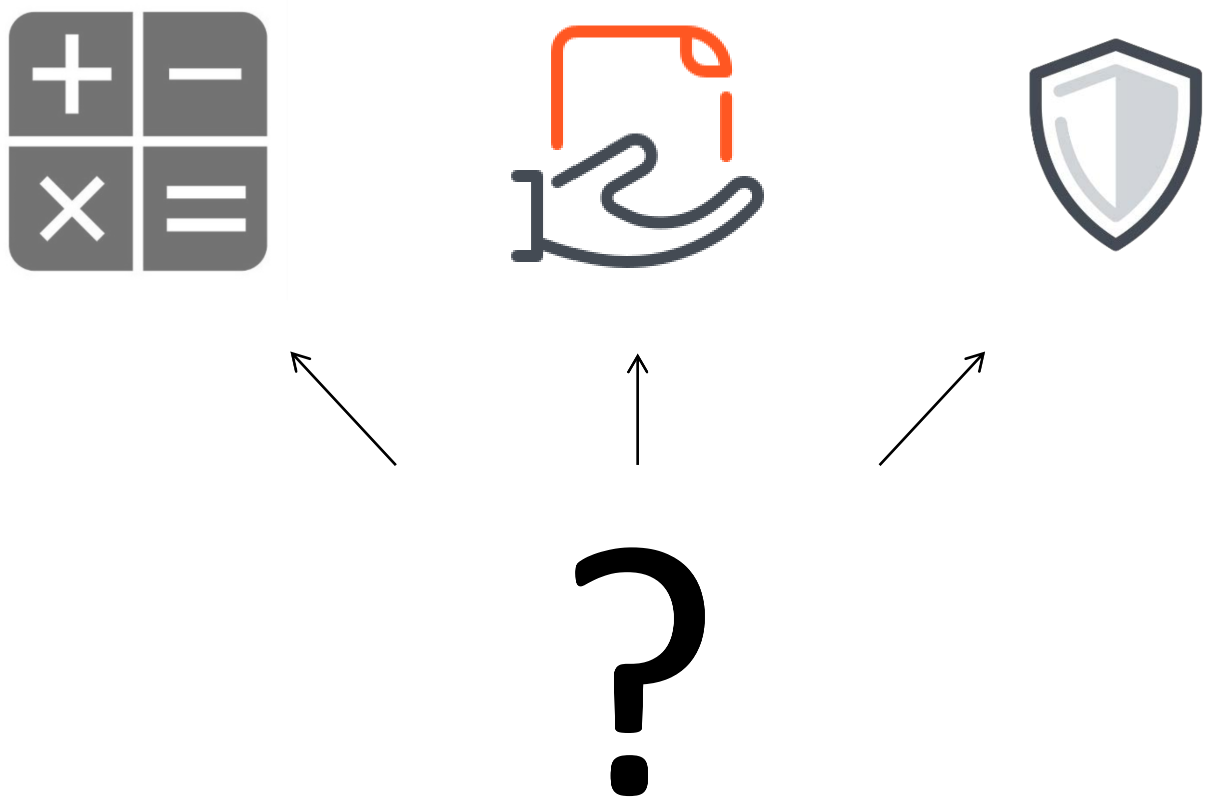


?

Cloud Computing



Cloud Computing



Fully Homomorphic Encryption Gentry 2009

$$E_{hom}(x_1) \star E_{hom}(x_2) = E_{hom}(x_1 + x_2)$$

$$E_{hom}(x_1) \diamond E_{hom}(x_2) = E_{hom}(x_1 \times x_2)$$

Fully Homomorphic Encryption Gentry 2009

$$E_{hom}(x_1) \star E_{hom}(x_2) = E_{hom}(x_1 + x_2)$$

$$E_{hom}(x_1) \diamond E_{hom}(x_2) = E_{hom}(x_1 \times x_2)$$

$x_1, \ldots, x_n$

Fully Homomorphic Encryption Gentry 2009

$$E_{hom}(x_1) \star E_{hom}(x_2) = E_{hom}(x_1 + x_2)$$
$$E_{hom}(x_1) \diamond E_{hom}(x_2) = E_{hom}(x_1 \times x_2)$$

$x_1, \ldots, x_n$

$E_{hom}(x_1), \ldots, E_{hom}(x_n)$

Fully Homomorphic Encryption Gentry 2009

$$E_{hom}(x_1) \star E_{hom}(x_2) = E_{hom}(x_1 + x_2)$$

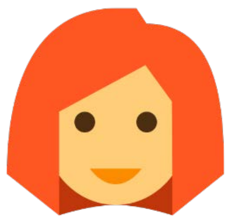
$$E_{hom}(x_1) \diamond E_{hom}(x_2) = E_{hom}(x_1 \times x_2)$$

$x_1, \ldots, x_n$

$E_{hom}(x_1), \ldots, E_{hom}(x_n)$

Fully Homomorphic Encryption Gentry 2009

$$E_{hom}(x_1) \star E_{hom}(x_2) = E_{hom}(x_1 + x_2)$$
$$E_{hom}(x_1) \diamond E_{hom}(x_2) = E_{hom}(x_1 \times x_2)$$



$x_1, \ldots, x_n$

$E_{hom}(x_1), \ldots, E_{hom}(x_n)$

$E_{hom}(f(x_1, \ldots, x_n))$

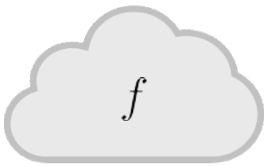
Fully Homomorphic Encryption Gentry 2009

$$E_{hom}(x_1) \star E_{hom}(x_2) = E_{hom}(x_1 + x_2)$$

$$E_{hom}(x_1) \diamond E_{hom}(x_2) = E_{hom}(x_1 \times x_2)$$

$x_1, \ldots, x_n$

$E_{hom}(x_1), \ldots, E_{hom}(x_n)$



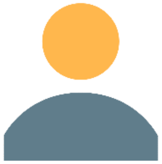
$E_{hom}(f(x_1, \ldots, x_n))$

$f(x_1, \ldots, x_n)$

Fully Homomorphic Encryption

$x_1, \ldots, x_n$

$f(x_1, \ldots, x_n)$

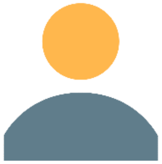


Fully Homomorphic Encryption

$x_1, \ldots, x_n$

$E^{\mathsf{pk}}_{hom}(x_1), \ldots, E^{\mathsf{pk}}_{hom}(x_n)$

$f(x_1, \ldots, x_n)$



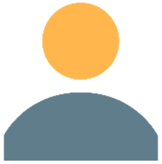
Fully Homomorphic Encryption

$x_1, \ldots, x_n$

$E^{\mathsf{pk}}_{hom}(x_1), \ldots, E^{\mathsf{pk}}_{hom}(x_n)$

$E^{\mathsf{pk}}_{hom}(f(x_1, \ldots, x_n))$

$f(x_1, \ldots, x_n)$



Fully Homomorphic Encryption

$x_1, \ldots, x_n$

$E^{\mathsf{pk}}_{hom}(x_1), \ldots, E^{\mathsf{pk}}_{hom}(x_n)$

$E^{\mathsf{pk}}_{hom}(f(x_1, \ldots, x_n))$

Re-encryption

$E^{\mathsf{pk}_U}_{hom}(f(x_1, \ldots, x_n))$

$f(x_1, \ldots, x_n)$

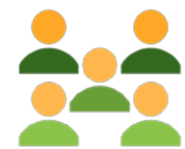
Fully Homomorphic Encryption

$x_1, \ldots, x_n$

$E^{\mathsf{pk}}_{hom}(x_1), \ldots, E^{\mathsf{pk}}_{hom}(x_n)$

$E^{\mathsf{pk}}_{hom}(f(x_1, \ldots, x_n))$

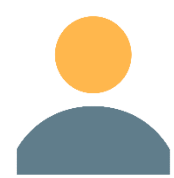
Re-encryption



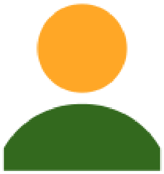
Distributed Controller

$E^{\mathsf{pk}_U}_{hom}(f(x_1, \ldots, x_n))$

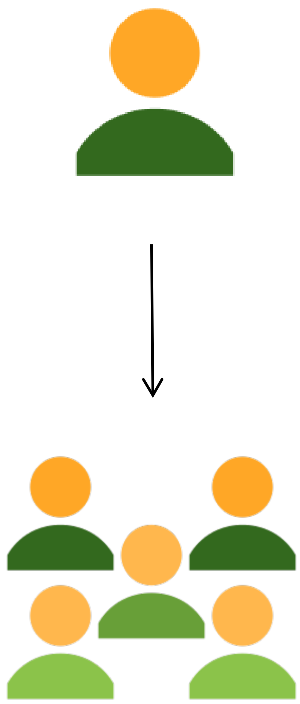
$f(x_1, \ldots, x_n)$



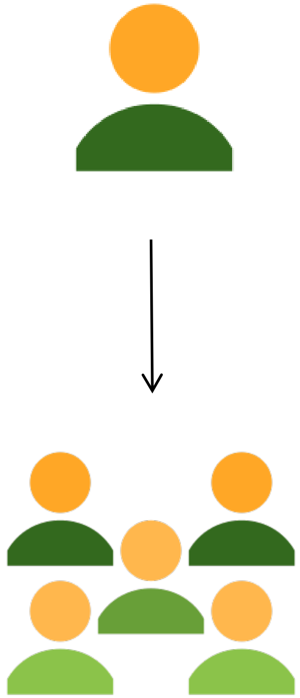
Decentralization



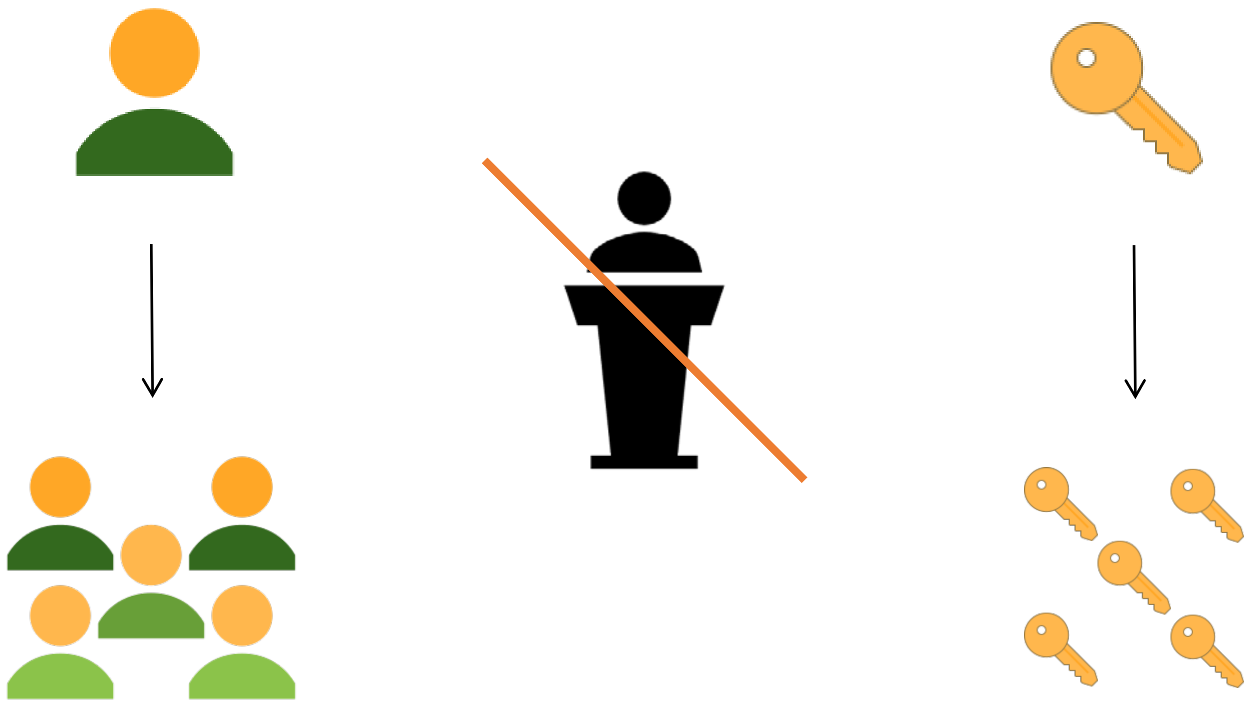
Decentralization



Decentralization



Decentralization



Group Testing

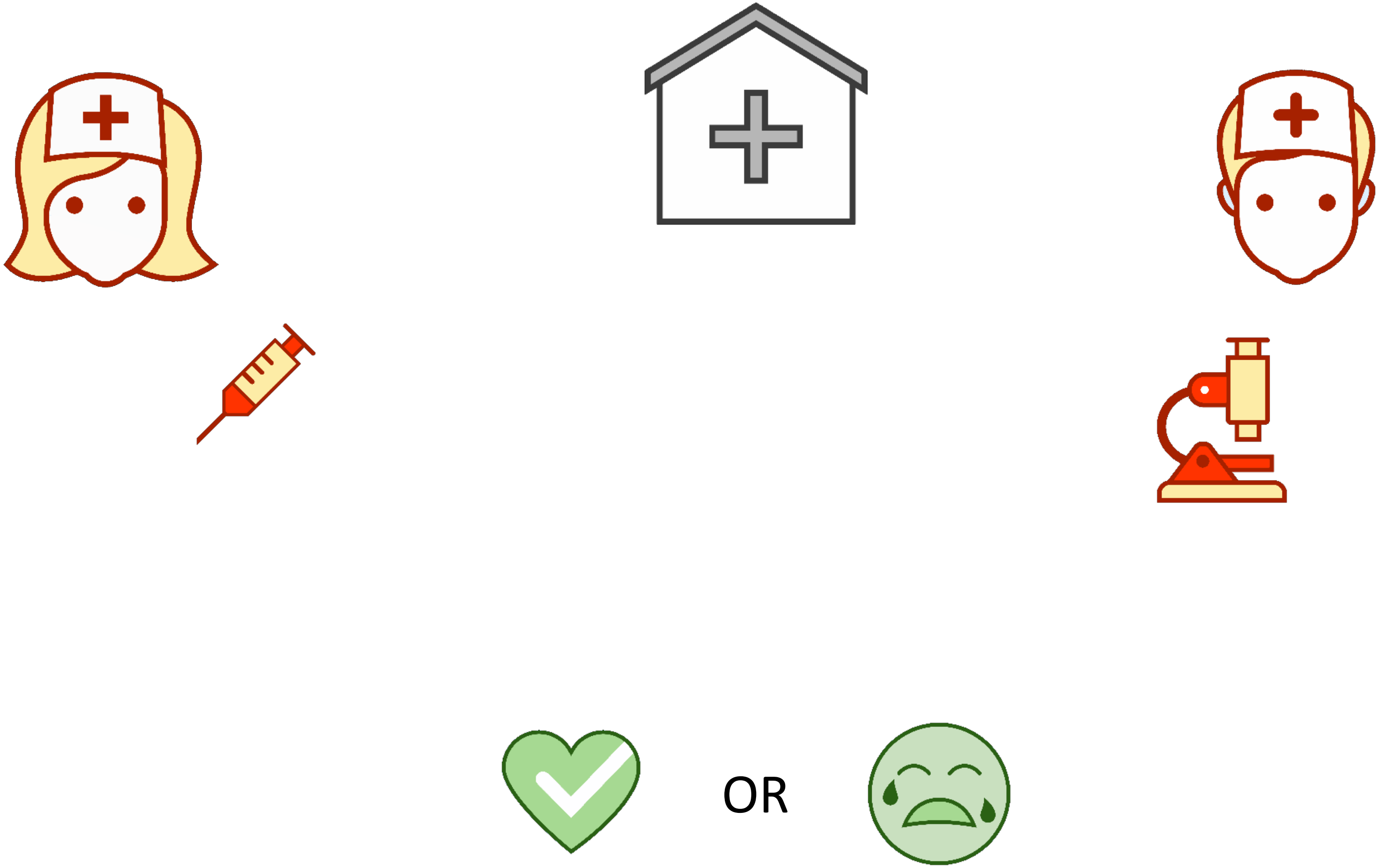
Motivation: Group Testing

OR

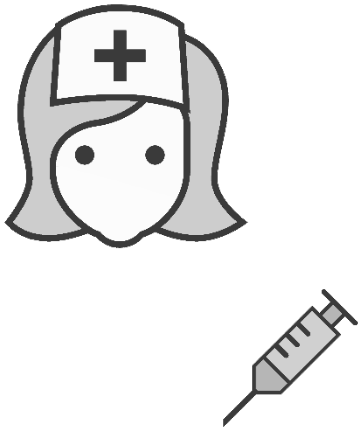
Motivation: Group Testing

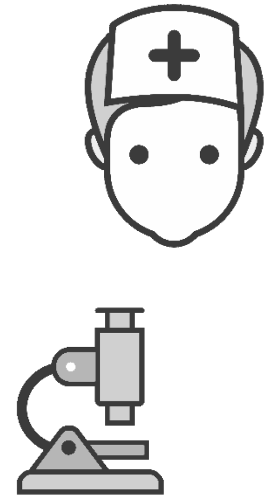
OR

Motivation: Group Testing



Motivation: Group Testing

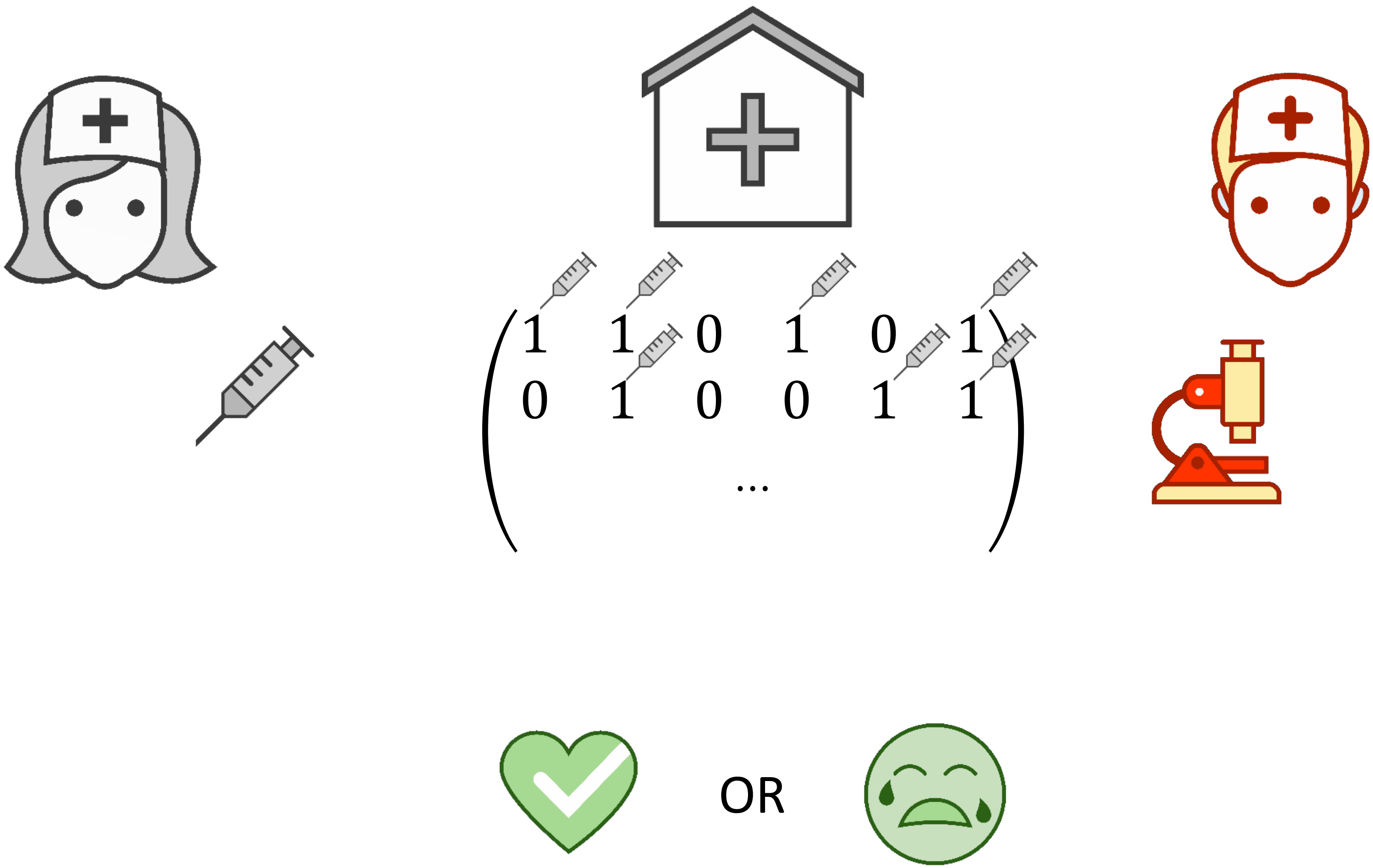




$$\begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ & & & \dots & & \\ & & & & & \end{pmatrix}$$

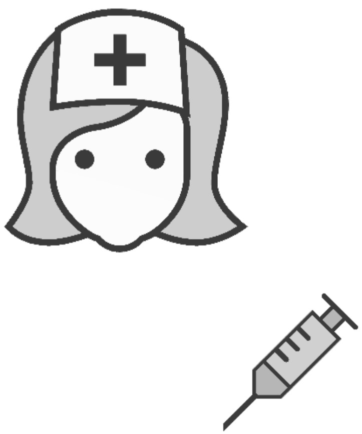
OR

Motivation: Group Testing

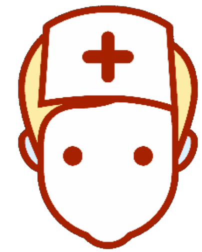


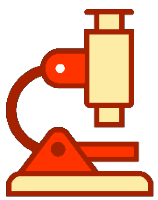
$$\begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ & & & \cdots & & \end{pmatrix}$$

Motivation: Group Testing



$$\begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ & & & \cdots & & \end{pmatrix}$$

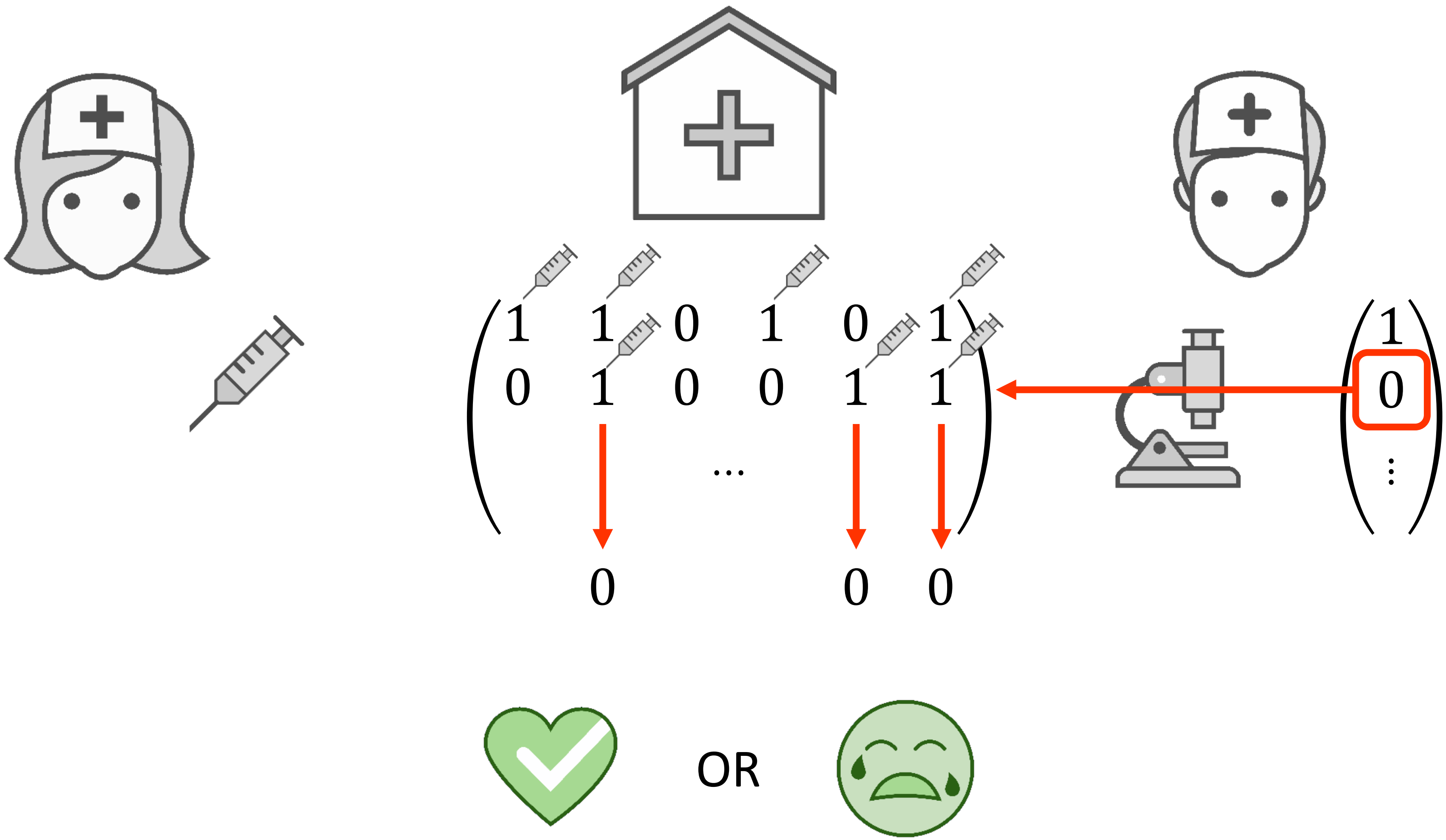




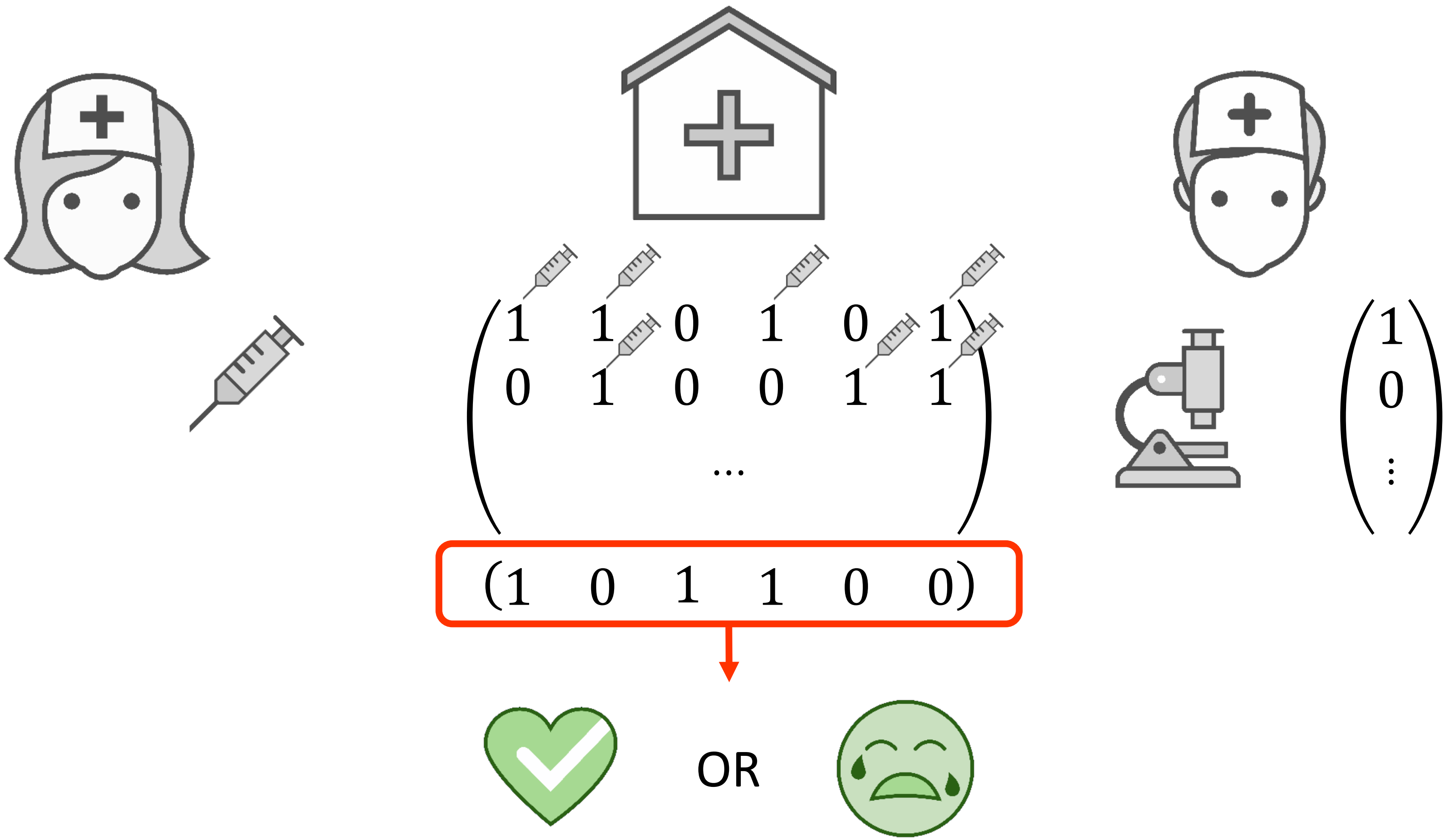
$$\begin{pmatrix} 1 \\ 0 \\ \vdots \end{pmatrix}$$

 OR 

Motivation: Group Testing



Motivation: Group Testing



Motivation: Group Testing

$$\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ & & \cdots & \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{pmatrix} \qquad \begin{pmatrix} y_1 \\ y_2 \\ \cdots \\ y_m \end{pmatrix}$$

$$\boxed{\bar{F}_j = \bigvee_i (x_{ij} \wedge \bar{y}_i)}$$

OR

Motivation: Group Testing

$$\begin{pmatrix} x_{11} & x_{12} & & \cdots & x_{1n} \\ & & \cdots & & \\ x_{m1} & x_{m2} & & \cdots & x_{mn} \end{pmatrix} \qquad \begin{pmatrix} y_1 \\ y_2 \\ \cdots \\ y_m \end{pmatrix}$$

$$\boxed{\bar{F}_j = \sum_i (x_{ij} \cdot (1 - y_i))}$$

OR

2-DNF on Encrypted Data

$x_1, \dots, x_n \in \{0,1\}$

2-DNF:

$$\bigvee_{i=1}^{m} (\ell_{i,1} \wedge \ell_{i,2}) \qquad \ell_{i,1} \wedge \ell_{i,2} \in \{x_1, \dots, x_n\} \cup \{\overline{x_1}, \dots, \overline{x_n}\}$$

2-DNF on Encrypted Data

$x_1, \dots, x_n \in \{0,1\}$

2-DNF:

$$\bigvee_{i=1}^{m} (\ell_{i,1} \wedge \ell_{i,2}) \qquad \ell_{i,1} \wedge \ell_{i,2} \in \{x_1, \dots, x_n\} \cup \{\overline{x_1}, \dots, \overline{x_n}\}$$

Multivariate polynomial degree 2:

$$\sum_{i=1}^{m} (y_{i,1} \cdot y_{i,2}) \qquad \begin{cases} y_{i,j} = \ell_{i,j} & \text{if } \ell_{i,j} \in \{x_1, \dots, x_n\} \\ y_{i,j} = 1 - \ell_{i,j} & \text{if } \ell_{i,j} \in \{\overline{x_1}, \dots, \overline{x_n}\} \end{cases}$$

Encryption Scheme

Related Work

- BGN 2005

Related Work

- BGN 2005
- Freeman 2010

Related Work

- BGN 2005
- Freeman 2010

- Our Scheme

Related Work

- BGN 2005
- Freeman 2010
- Attrapadung *et al* ASIACCS 18
  - Concurrent Work

- Our Scheme

Related Work

- Our Scheme

- BGN 2005
- Freeman 2010
- Attrapadung *et al* ASIACCS 18
  - Concurrent Work

link

Related Work

- BGN 2005
- Freeman 2010
- Attrapadung *et al* ASIACCS 18
  - Concurrent Work

- Our Scheme

link

+

Related Work

- BGN 2005
- Freeman 2010
- Attrapadung *et al* ASIACCS 18
  - Concurrent Work

- Our Scheme

link

+

- Multi-user setting

Related Work

- BGN 2005
- Freeman 2010
- Attrapadung *et al* ASIACCS 18
  - Concurrent Work

- Our Scheme

link

+

- Multi-user setting
- Efficient distributed decryption

Related Work

- BGN 2005
- Freeman 2010
- Attrapadung *et al* ASIACCS 18
  - Concurrent Work

- Our Scheme

link

+

- Multi-user setting
- Efficient distributed decryption
- Efficient distributed re-encryption

Related Work

- BGN 2005
- Freeman 2010
- Attrapadung *et al* ASIACCS 18
  - Concurrent Work

- Our Scheme

link

+

- Multi-user setting
- Efficient distributed decryption
- Efficient distributed re-encryption
- Decentralized key generation

Notations

$$\mathbb{G}_s =< g_s >$$

$$a \in \mathbb{Z}_p; \qquad [a]_s = g_s^a$$

Notations

$$\mathbb{G}_s = < g_s >$$

$$a \in \mathbb{Z}_p; \qquad [a]_s = g_s^a$$

$$\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_p^n; \qquad [\mathbf{x}]_s = (g_s^{x_1}, \ldots, g_s^{x_n})$$

Notations

$$\mathbb{G}_s =< g_s >$$

$$a \in \mathbb{Z}_p; \qquad [a]_s = g_s^a$$

$$\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_p^n; \qquad [\mathbf{x}]_s = (g_s^{x_1}, \ldots, g_s^{x_n})$$

$$e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$$

Notations

$$\mathbb{G}_s =< g_s >$$

$$a \in \mathbb{Z}_p; \qquad [a]_s = g_s^a$$

$$\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_p^n; \qquad [\mathbf{x}]_s = (g_s^{x_1}, \ldots, g_s^{x_n})$$

$$e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$$

$$[\mathbf{A}]_1 \bullet [\mathbf{B}]_2 = [\mathbf{A} \otimes \mathbf{B}]_T$$

Notations

$$\mathbb{G}_s =< g_s >$$

$$a \in \mathbb{Z}_p; \qquad [a]_s = g_s^a$$

$$\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{Z}_p^n; \qquad [\mathbf{x}]_s = (g_s^{x_1}, \ldots, g_s^{x_n})$$

$$e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$$

$$[\mathbf{A}]_1 \bullet [\mathbf{B}]_2 = [\mathbf{A} \otimes \mathbf{B}]_T$$

$$\begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \otimes \mathbf{B} = \begin{pmatrix} a_{11} \cdot \mathbf{B} & a_{12} \cdot \mathbf{B} \\ a_{21} \cdot \mathbf{B} & a_{22} \cdot \mathbf{B} \end{pmatrix}$$

The Encryption Scheme

- Keygen

$$\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

The Encryption Scheme

- Keygen

Projection

$$\in \mathrm{GL}_2(\mathbb{Z}_p)$$

$$\mathbf{P}_s = \left(\mathbf{B}_s^{-1}\right)\left(\mathbf{U}_2\right)\left(\mathbf{B}_s\right)$$

$$/\!/$$

sk_s

The Encryption Scheme

- Keygen

Projection

$\in \mathrm{GL}_2(\mathbb{Z}_p)$

$$\mathbf{P}_s = \left(\mathbf{B}_s^{-1}\right)\left(\mathbf{U}_2\right)\left(\mathbf{B}_s\right)$$

sk_s

$$\mathbf{p}_s \in \ker(\mathbf{P}_s) = \{\mathbf{x} : \mathbf{x} \cdot \mathbf{P}_s = \begin{pmatrix} 0 & 0 \end{pmatrix}\}$$

$$\mathsf{pk}_s = [\mathbf{p}_s]_s \Rightarrow [\mathbf{p}_s]_s \cdot \mathbf{P}_s = [\begin{pmatrix} 0 & 0 \end{pmatrix}]_s$$

The Encryption Scheme

- Keygen

$$\mathsf{sk}_s = \mathbf{P}_s = \left(\mathbf{B}_s^{-1}\right)\left(\mathbf{U}_2\right)\left(\mathbf{B}_s\right) \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$

$$\mathsf{pk}_s = [\mathbf{p}_s]_s \Rightarrow [\mathbf{p}_s]_s \cdot \mathbf{P}_s = [\mathbf{0}]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

The Encryption Scheme

- Keygen

$$\mathsf{sk}_s = \mathbf{P}_s = \left(\mathbf{B}_s^{-1}\right)\left(\mathbf{U}_2\right)\left(\mathbf{B}_s\right) \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$

$$\mathsf{pk}_s = [\mathbf{p}_s]_s \Rightarrow [\mathbf{p}_s]_s \cdot \mathbf{P}_s = [\mathbf{0}]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$$C_s = (m \cdot [a_s]_s + r \cdot [\mathbf{p}_s]_s, [a_s]_s) \qquad r \xleftarrow{\$} \mathbb{Z}_p$$

The Encryption Scheme

- Keygen

$$\mathsf{sk}_s = \mathbf{P}_s = \begin{pmatrix} \mathbf{B}_s^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{U}_2 \end{pmatrix} \begin{pmatrix} \mathbf{B}_s \end{pmatrix} \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$

$$\mathsf{pk}_s = [\mathbf{p}_s]_s \Rightarrow [\mathbf{p}_s]_s \cdot \mathbf{P}_s = [\mathbf{0}]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$$C_s = (m \cdot [a_s]_s + r \cdot [\mathbf{p}_s]_s, [a_s]_s) \qquad r \xleftarrow{\$} \mathbb{Z}_p$$

$$C_T = (m \cdot [a_1]_1 \bullet [a_2]_2 + [\mathbf{p}_1]_1 \bullet [\mathbf{r}_2]_2 + [\mathbf{r}_1]_1 \bullet [\mathbf{p}_2]_2, [a_1]_1 \bullet [a_2]_2)$$

$$[\mathbf{r}_1]_1 \xleftarrow{\$} \mathbb{G}_1^2, [\mathbf{r}_2]_2 \xleftarrow{\$} \mathbb{G}_2^2$$

The Encryption Scheme

- Keygen

$$\mathsf{sk}_s = \mathbf{P}_s = \left(\mathbf{B}_s^{-1}\right)\left(\ \mathbf{U}_2\ \right)\left(\ \mathbf{B}_s\ \right) \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$

$$\mathsf{pk}_s = [\mathbf{p}_s]_s \Rightarrow [\mathbf{p}_s]_s \cdot \mathbf{P}_s = [\mathbf{0}]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$$C_s = (m \cdot [a_s]_s + r \cdot [\mathbf{p}_s]_s, [a_s]_s) \qquad r \xleftarrow{\$} \mathbb{Z}_p$$

$$C_T = (m \cdot [a_1]_1 \bullet [a_2]_2 + [\mathbf{p}_1]_1 \bullet [\mathbf{r}_2]_2 + [\mathbf{r}_1]_1 \bullet [\mathbf{p}_2]_2, [a_1]_1 \bullet [a_2]_2)$$

$$[\mathbf{r}_1]_1 \xleftarrow{\$} \mathbb{G}_1^2, [\mathbf{r}_2]_2 \xleftarrow{\$} \mathbb{G}_2^2$$

- Decrypt

The Encryption Scheme

- Keygen

$$\mathsf{sk}_s = \mathbf{P}_s = \left(\mathbf{B}_s^{-1}\right)\left(\mathbf{U}_2\right)\left(\mathbf{B}_s\right) \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$

$$\mathsf{pk}_s = [\mathbf{p}_s]_s \Rightarrow [\mathbf{p}_s]_s \cdot \mathbf{P}_s = [\mathbf{0}]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$\in \ker(\boldsymbol{P_s})$

$$C_s = (m \cdot [a_s]_s + \boxed{r \cdot [\mathbf{p}_s]_s}, [a_s]_s) \qquad r \xleftarrow{\$} \mathbb{Z}_p$$

$$C_T = (m \cdot [a_1]_1 \bullet [a_2]_2 + [\mathbf{p}_1]_1 \bullet [\mathbf{r}_2]_2 + [\mathbf{r}_1]_1 \bullet [\mathbf{p}_2]_2, [a_1]_1 \bullet [a_2]_2)$$

$$[\mathbf{r}_1]_1 \xleftarrow{\$} \mathbb{G}_1^2, [\mathbf{r}_2]_2 \xleftarrow{\$} \mathbb{G}_2^2$$

- Decrypt

The Encryption Scheme

- Keygen

$$\mathsf{sk}_s = \mathbf{P}_s = \begin{pmatrix} \mathbf{B}_s^{-1} \end{pmatrix} \begin{pmatrix} \mathbf{U}_2 \end{pmatrix} \begin{pmatrix} \mathbf{B}_s \end{pmatrix} \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$

$$\mathsf{pk}_s = [\mathbf{p}_s]_s \Rightarrow [\mathbf{p}_s]_s \cdot \mathbf{P}_s = [\mathbf{0}]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$$C_s = (m \cdot [a_s]_s + \underbrace{\boxed{r \cdot [\mathbf{p}_s]_s}}_{\in \ker(\boldsymbol{P}_s)}, [a_s]_s) \qquad r \xleftarrow{\$} \mathbb{Z}_p$$

$$C_T = (m \cdot [a_1]_1 \bullet [a_2]_2 + \underbrace{\boxed{[\mathbf{p}_1]_1 \bullet [\mathbf{r}_2]_2 + [\mathbf{r}_1]_1 \bullet [\mathbf{p}_2]_2}}_{\in \ker(\boldsymbol{P}_1 \otimes \boldsymbol{P}_2)}, [a_1]_1 \bullet [a_2]_2)$$

$$[\mathbf{r}_1]_1 \xleftarrow{\$} \mathbb{G}_1^2, [\mathbf{r}_2]_2 \xleftarrow{\$} \mathbb{G}_2^2$$

- Decrypt

The Encryption Scheme

- Keygen

$$\mathsf{sk}_s = \mathbf{P}_s = \left(\mathbf{B}_s^{-1}\right)\left(\mathbf{U}_2\right)\left(\mathbf{B}_s\right) \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$

$$\mathsf{pk}_s = [\mathbf{p}_s]_s \Rightarrow [\mathbf{p}_s]_s \cdot \mathbf{P}_s = [\mathbf{0}]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$$C_s = (m \cdot [a_s]_s + \underbrace{\boxed{r \cdot [\mathbf{p}_s]_s}}_{\in \ker(\boldsymbol{P}_s)}, [a_s]_s) \qquad r \xleftarrow{\$} \mathbb{Z}_p$$

$$C_T = (m \cdot [a_1]_1 \bullet [a_2]_2 + \underbrace{\boxed{[\mathbf{p}_1]_1 \bullet [\mathbf{r}_2]_2 + [\mathbf{r}_1]_1 \bullet [\mathbf{p}_2]_2}}_{\in \ker(\boldsymbol{P}_1 \otimes \boldsymbol{P}_2)}, [a_1]_1 \bullet [a_2]_2)$$

$$[\mathbf{r}_1]_1 \xleftarrow{\$} \mathbb{G}_1^2, [\mathbf{r}_2]_2 \xleftarrow{\$} \mathbb{G}_2^2$$

- Decrypt

$$C_s \cdot \mathbf{P}_s = (m \cdot [a_s]_s \cdot \mathbf{P}_s + [\mathbf{0}]_s, [a_s]_s \cdot \mathbf{P}_s)$$

The Encryption Scheme

- Keygen

$$\mathsf{sk}_s = \mathbf{P}_s = \left(\mathbf{B}_s^{-1}\right)\left(\mathbf{U}_2\right)\left(\mathbf{B}_s\right) \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$

$$\mathsf{pk}_s = [\mathbf{p}_s]_s \Rightarrow [\mathbf{p}_s]_s \cdot \mathbf{P}_s = [\mathbf{0}]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$$C_s = (m \cdot [a_s]_s + \overbrace{\boxed{r \cdot [\mathbf{p}_s]_s}}^{\in \ker(\boldsymbol{P}_s)}, [a_s]_s) \qquad r \xleftarrow{\$} \mathbb{Z}_p$$

$$C_T = (m \cdot [a_1]_1 \bullet [a_2]_2 + \overbrace{\boxed{[\mathbf{p}_1]_1 \bullet [\mathbf{r}_2]_2 + [\mathbf{r}_1]_1 \bullet [\mathbf{p}_2]_2}}^{\in \ker(\boldsymbol{P}_1 \otimes \boldsymbol{P}_2)}, [a_1]_1 \bullet [a_2]_2)$$

$$[\mathbf{r}_1]_1 \xleftarrow{\$} \mathbb{G}_1^2, [\mathbf{r}_2]_2 \xleftarrow{\$} \mathbb{G}_2^2$$

- Decrypt

$$C_s \cdot \mathbf{P}_s = (m \cdot [a_s]_s \cdot \mathbf{P}_s + [\mathbf{0}]_s, [a_s]_s \cdot \mathbf{P}_s)$$

$$C_T \cdot (\mathbf{P}_1 \otimes \mathbf{P}_2) = (m \cdot [a_1]_1 \bullet [a_2]_2 \cdot (\mathbf{P}_1 \otimes \mathbf{P}_2) + [\mathbf{0}]_T, [a_1]_1 \bullet [a_2]_2 \cdot (\mathbf{P}_1 \otimes \mathbf{P}_2))$$

The Homomorphic Properties

- Add: *Many times*

$$[c_s]_s + [c'_s]_s = (m + m') \cdot [a_s]_s + (r + r') \cdot [\mathbf{p}_s]_s$$

The Homomorphic Properties

- Add: *Many times*

$$[c_s]_s + [c'_s]_s = (m + m') \cdot [a_s]_s + (r + r') \cdot [\mathbf{p}_s]_s$$

$$[c_T]_T + [c_T]_T = (m + m') \cdot [\mathbf{a}_1]_1 \bullet [\mathbf{a}_2]_2 + [\mathbf{p}_1]_1 \bullet [\mathbf{r}_2 + \mathbf{r}'_2]_2 + [\mathbf{r}_1 + \mathbf{r}'_1]_1 \bullet [\mathbf{p}_2]_2$$

The Homomorphic Properties

- Add: *Many times*

$$[c_s]_s + [c'_s]_s = (m + m') \cdot [a_s]_s + (r + r') \cdot [\mathbf{p}_s]_s$$

$$[c_T]_T + [c_T]_T = (m + m') \cdot [\mathbf{a}_1]_1 \bullet [\mathbf{a}_2]_2 + [\mathbf{p}_1]_1 \bullet [\mathbf{r}_2 + \mathbf{r}'_2]_2 + [\mathbf{r}_1 + \mathbf{r}'_1]_1 \bullet [\mathbf{p}_2]_2$$

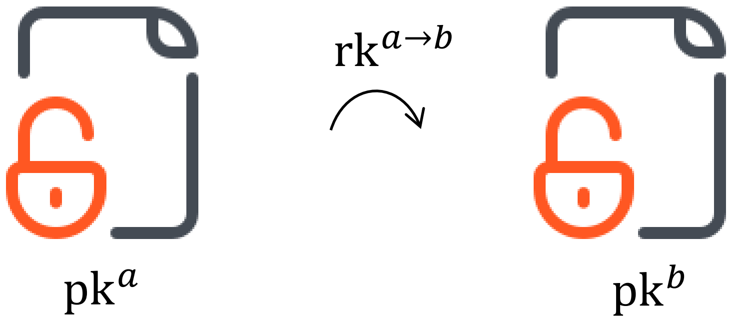
- Multiply: *Once*

$$[\mathbf{c}_1]_1 \bullet [\mathbf{c}_2]_2 = (m_1 \cdot m_2) \cdot [\mathbf{a}_1]_1 \bullet [\mathbf{a}_2]_2 + [\mathbf{p}_1]_1 \bullet [\mathbf{r}']_2 + [\mathbf{r}]_1 \bullet [\mathbf{p}_2]_2$$

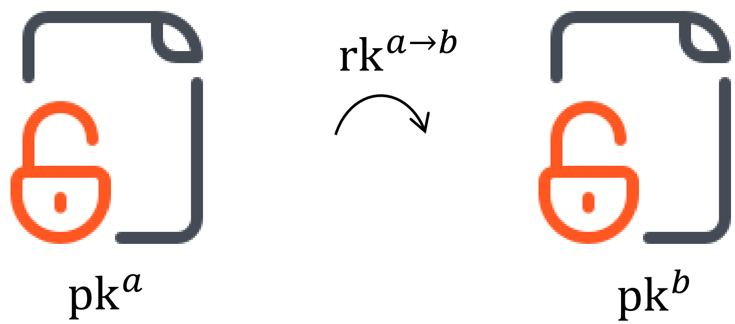
with $[\mathbf{r}]_1 = m_1 r_2 \mathbf{a}_1$

$[\mathbf{r}']_2 = m_2 r_1 \mathbf{a}_2 + r_1 r_2 \mathbf{p}_2$

Re-Encryption

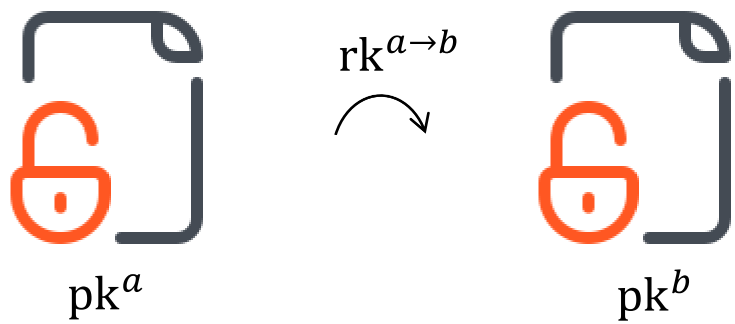


Re-Encryption



$$C_s \cdot \mathsf{rk}^{a\to b} = (m \cdot [a_s]_s \cdot \mathbf{R} + r \cdot [\mathbf{p}_s]_s \cdot \mathbf{R}, [a_s]_s \cdot \mathbf{R}) = C'_s$$

Re-Encryption



$$C_s \cdot \mathsf{rk}^{a\to b} = (m\cdot[a_s]_s\cdot\mathbf{R} + \boxed{r\cdot[\mathbf{p}_s]_s\cdot\mathbf{R}}, [a_s]_s\cdot\mathbf{R}) = C'_s$$

$$[\mathbf{p}_s]_s\cdot\mathbf{R} = r'\cdot[\mathbf{p}'_s]_s$$

$$\boxed{r''\cdot[\mathbf{p}'_s]_s}$$

Problem

Problem

- Distributed decryption and re-encryption?

Problem

- Distributed decryption and re-encryption?
  - Yes, with distributed keys

Problem

- Distributed decryption and re-encryption?
  - Yes, with distributed keys
- Decentralized key generation?

Problem

- Distributed decryption and re-encryption?
  - Yes, with distributed keys
- Decentralized key generation?

$$\mathbf{P}_s = \left(\mathbf{B}_s^{-1} \right) \left(\mathbf{U}_2 \right) \left(\mathbf{B}_s \right)$$

Problem

- Distributed decryption and re-encryption?
  - Yes, with distributed keys
- Decentralized key generation?

$$\mathbf{P}_s = \left(\mathbf{B}_s^{-1} \right) \left(\mathbf{U}_2 \right) \left(\mathbf{B}_s \right)$$

  - No …

Simplification (still secure)

- Size of the keys:

$$\mathbf{P}_s = \begin{pmatrix} 1 & 0 \\ x & 0 \end{pmatrix}$$

$$[\mathbf{p}_s]_s = [-x \quad 1]_s$$

Simplification (still secure)

- Size of the keys:

$$\mathbf{P}_s = \begin{pmatrix} 1 & 0 \\ x & 0 \end{pmatrix} \qquad \mathsf{sk}_s = x$$

$$[\mathbf{p}_s]_s = [-x \quad 1]_s \qquad \mathsf{pk}_s = [-x]_s$$

Simplification (still secure)

- Size of the keys:

$$\mathbf{P}_s = \begin{pmatrix} 1 & 0 \\ x & 0 \end{pmatrix} \qquad \mathsf{sk}_s = x$$

$$[\mathbf{p}_s]_s = [-x \quad 1]_s \qquad \mathsf{pk}_s = [-x]_s$$

- Size of the ciphertexts:

$$[\mathbf{a}_s]_s = [1 \quad 0]_s$$

Simplification (still secure)

- Size of the keys:

$$\mathbf{P}_s = \begin{pmatrix} 1 & 0 \\ x & 0 \end{pmatrix}$$

$$\mathsf{sk}_s = x$$

$$[\mathbf{p}_s]_s = [-x \quad 1]_s$$

$$\mathsf{pk}_s = [-x]_s$$

- Size of the ciphertexts:

$$[\mathbf{a}_s]_s = [1 \quad 0]_s$$

$$C_s \in \mathbb{G}_s^2 \times \mathbb{G}_s^2 \Rightarrow \boxed{C_s \in \mathbb{G}_s^2}$$

$$C_T \in \mathbb{G}_T^4 \times \mathbb{G}_T^4 \Rightarrow \boxed{C_T \in \mathbb{G}_T^4}$$

The Optimized Encryption Scheme

- Keygen

$$\mathsf{sk}_s = x \qquad\qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$
$$\mathsf{pk}_s = [-x]_s \qquad\qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

The Optimized Encryption Scheme

- Keygen

$$\mathsf{sk}_s = x \qquad\qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$
$$\mathsf{pk}_s = [-x]_s \qquad\qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$$C_s = (g_s^m \cdot {\mathsf{pk}_s}^r, g_s^r) \qquad\qquad r \xleftarrow{\$} \mathbb{Z}_p$$

The Optimized Encryption Scheme

- Keygen

$$\mathsf{sk}_s = x \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$
$$\mathsf{pk}_s = [-x]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$$C_s = (g_s^m \cdot \mathsf{pk}_s{}^r, g_s^r) \qquad r \xleftarrow{\$} \mathbb{Z}_p$$

$$C_T = \begin{cases} e(g_1, g_2)^m \cdot e(g_1, \mathsf{pk}_2)^{r_{11}} \cdot e(\mathsf{pk}_1, g_2)^{r_{21}} \\ e(g_1, g_2)^{r_{11}} \cdot e(\mathsf{pk}_1, g_2)^{r_{22}} \\ e(g_1, \mathsf{pk}_2)^{r_{12}} \cdot e(g_1, g_2)^{r_{21}} \\ e(g_1, g_2)^{r_{12}+r_{22}} \end{cases} \qquad r_{11}, r_{12}, r_{21}, r_{22} \xleftarrow{\$} \mathbb{Z}_p^4$$

The Optimized Encryption Scheme

- Keygen

$$\mathsf{sk}_s = x \qquad \mathsf{sk}_T = (\mathsf{sk}_1, \mathsf{sk}_2)$$
$$\mathsf{pk}_s = [-x]_s \qquad \mathsf{pk}_T = (\mathsf{pk}_1, \mathsf{pk}_2)$$

- Encrypt

$$C_s = (g_s^m \cdot \mathsf{pk}_s{}^r, g_s^r) \qquad r \xleftarrow{\$} \mathbb{Z}_p$$

$$C_T = \begin{cases} e(g_1, g_2)^m \cdot e(g_1, \mathsf{pk}_2)^{r_{11}} \cdot e(\mathsf{pk}_1, g_2)^{r_{21}} \\ e(g_1, g_2)^{r_{11}} \cdot e(\mathsf{pk}_1, g_2)^{r_{22}} \\ e(g_1, \mathsf{pk}_2)^{r_{12}} \cdot e(g_1, g_2)^{r_{21}} \\ e(g_1, g_2)^{r_{12}+r_{22}} \end{cases} \qquad r_{11}, r_{12}, r_{21}, r_{22} \xleftarrow{\$} \mathbb{Z}_p^4$$

- Decrypt

$$c_{s,1} \cdot c_{s,2}^{\mathsf{sk}_s} \rightarrow m$$

$$c_{T,1} \cdot c_{T,2}^{\mathsf{sk}_2} \cdot c_{T,3}^{\mathsf{sk}_1} \cdot c_{T,4}^{\mathsf{sk}_1 \cdot \mathsf{sk}_2} \rightarrow m$$

Decentralization

- 1) Decentralized Key Generation
- 2) Distributed Decryption
- 3) Distributed Re-Encryption

Distributive Decryption

- Additive Secret Sharing

Distributive Decryption

- Additive Secret Sharing
  - Each player i, $\mathsf{sk}_{s,i}$

$$\mathsf{sk}_s = \sum_i \lambda_i \mathsf{sk}_{s,i} \qquad \lambda_i \text{ public}$$

Distributive Decryption

- Additive Secret Sharing
  - Each player i, $\mathsf{sk}_{s,i}$

$$\mathsf{sk}_s = \sum_i \lambda_i \mathsf{sk}_{s,i} \qquad \lambda_i \text{ public}$$

  - $c_s = (c_{s,1}, c_{s,2})$

Distributive Decryption

- Additive Secret Sharing
  - Each player i, $\mathsf{sk}_{s,i}$

$$\mathsf{sk}_s = \sum_i \lambda_i \mathsf{sk}_{s,i} \qquad \lambda_i \text{ public}$$

  - $c_s = (c_{s,1}, c_{s,2})$
  - Distributed decryption

$$c_{s,1} \cdot c_{s,2}^{\mathsf{sk}_s} = c_{s,1} \cdot c_{s,2}^{\sum_i \lambda_i \mathsf{sk}_{s,i}} = c_{s,1} \cdot \prod_i (c_{s,2}^{\mathsf{sk}_{s,i}})^{\lambda_i}$$

Distributed Re-encryption

- $c_s = (c_{s,1}, c_{s,2})$ under $\mathsf{pk}_s \rightarrow c'_s = (c'_{s,1}, c'_{s,2})$ under pk'_s

Distributed Re-encryption

- $c_s = (c_{s,1}, c_{s,2})$ under $\mathsf{pk}_s \to c'_s = (c'_{s,1}, c'_{s,2})$ under ${\mathsf{pk}'}_s$
- $\mathsf{sk}_s = \sum_i \lambda_i \mathsf{sk}_{s,i}$

Distributed Re-encryption

- $c_s = (c_{s,1}, c_{s,2})$ under $\mathsf{pk}_s \to c'_s = (c'_{s,1}, c'_{s,2})$ under $\mathsf{pk'}_s$
- $\mathsf{sk}_s = \sum_i \lambda_i \mathsf{sk}_{s,i}$
- Player i computes:

$$r'_i \xleftarrow{\$} \mathbb{Z}_p, \alpha_i = c_{s,2}^{\mathsf{sk}_{s,i}} \cdot \mathsf{pk'}_s{}^{r'_i}, \beta_i = g_s^{r'_i}$$

Distributed Re-encryption

- $c_s = (c_{s,1}, c_{s,2})$ under $\mathsf{pk}_s \to c'_s = (c'_{s,1}, c'_{s,2})$ under pk'_s
- $\mathsf{sk}_s = \sum_i \lambda_i \mathsf{sk}_{s,i}$
- Player i computes:

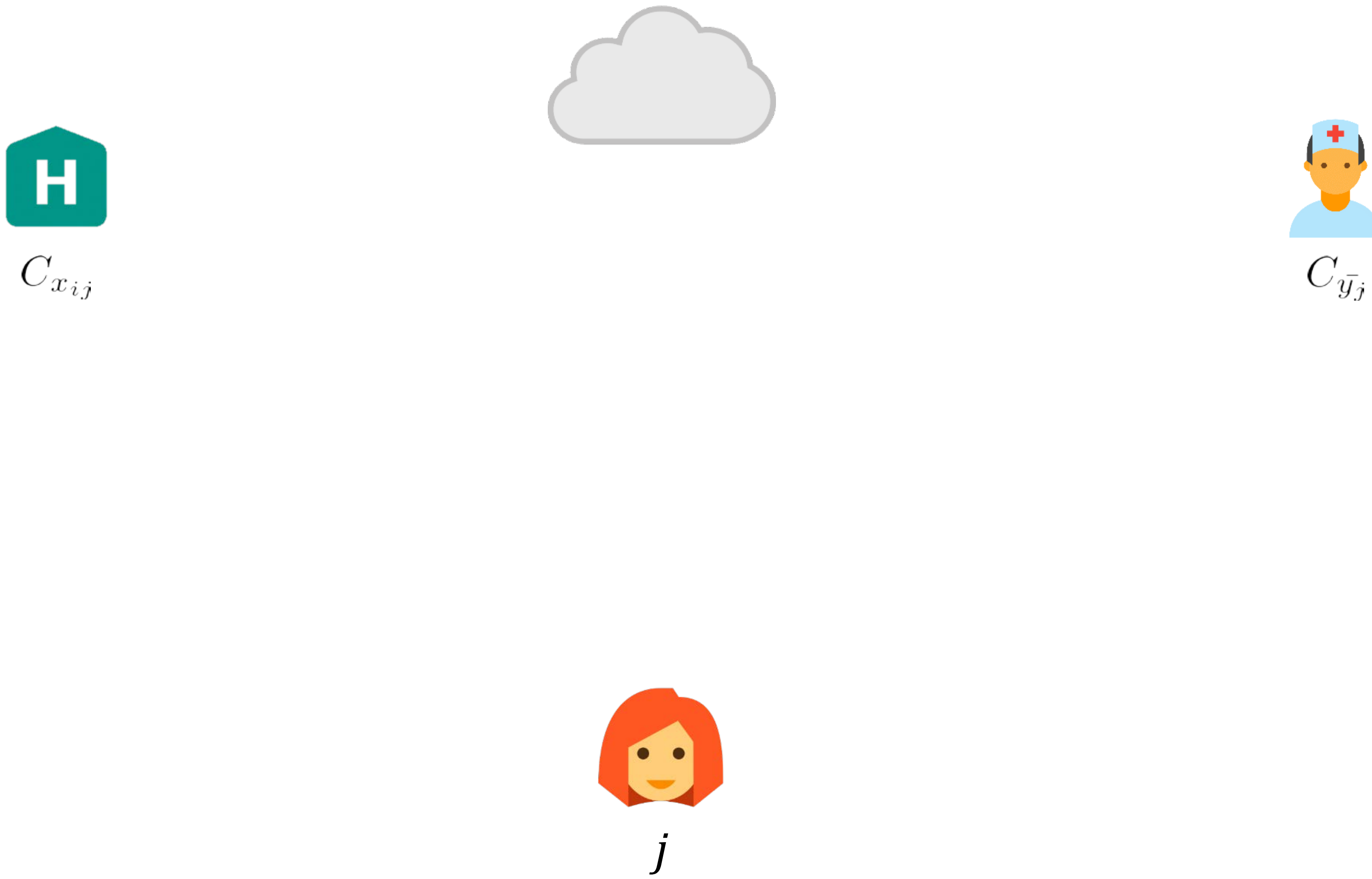
$$r'_i \xleftarrow{\$} \mathbb{Z}_p, \alpha_i = c_{s,2}^{\mathsf{sk}_{s,i}} \cdot {\mathsf{pk}'_s}^{r'_i}, \beta_i = g_s^{r'_i}$$

- Anybody can compute:

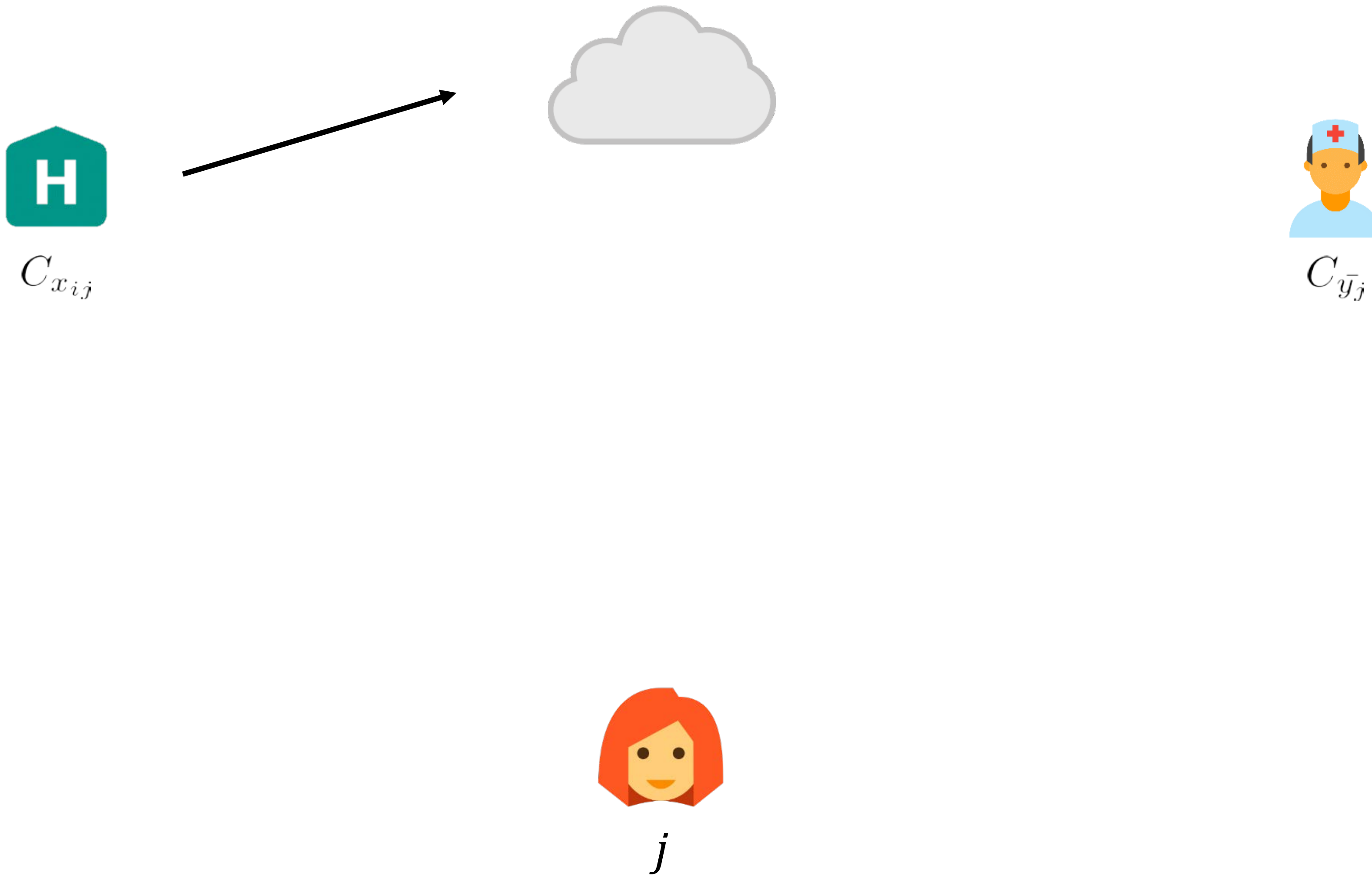
$$c'_s = (c_{s,1} \times \prod_i \alpha_i^{\lambda_i}, \prod_i \beta_i^{\lambda_i}) = (g_s^m \cdot {\mathsf{pk}'_s}^{r'}, g_s^{r'})$$

$$r' = \sum_i \lambda_i r'_i$$

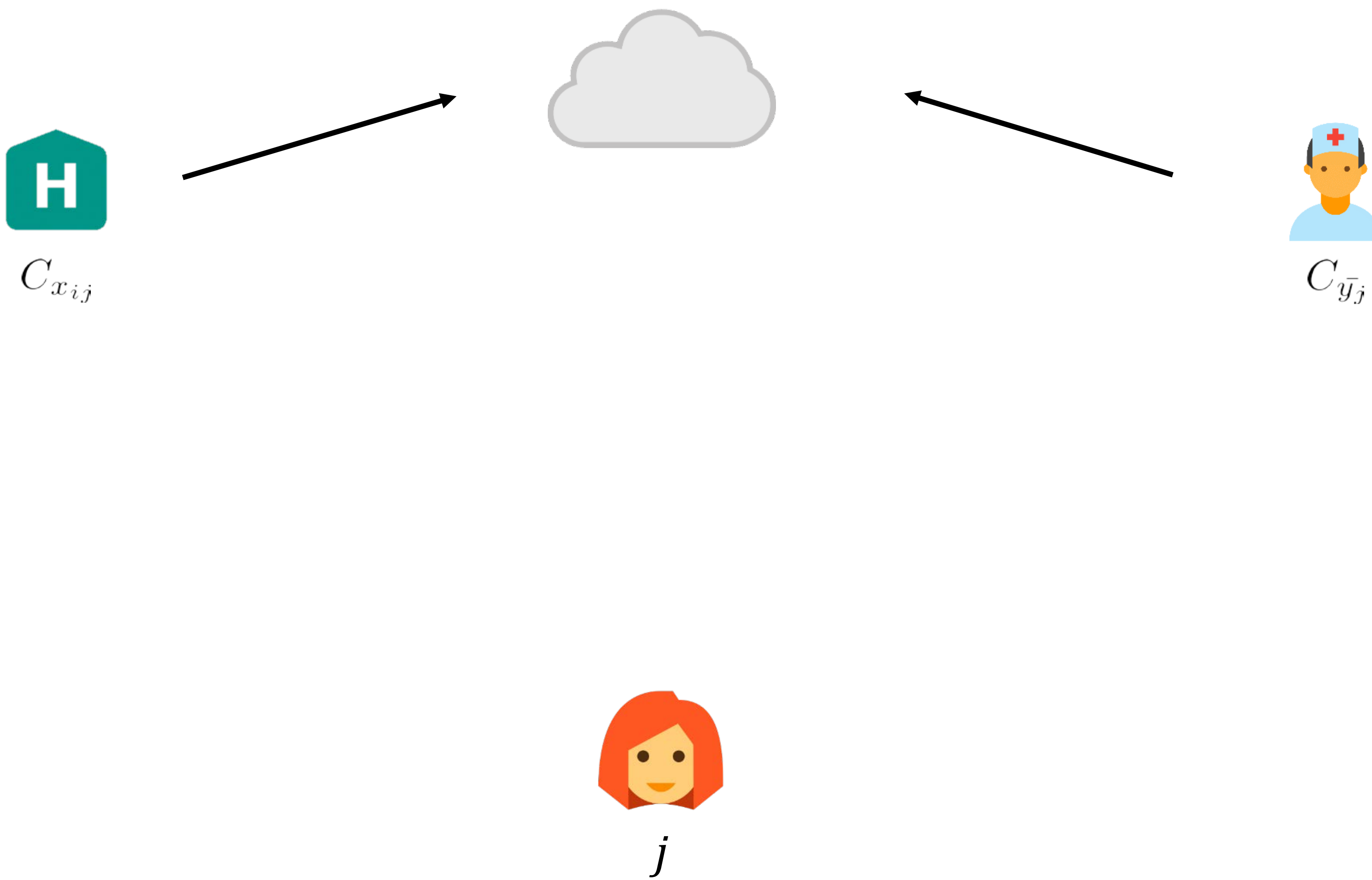
Solution: Group Testing



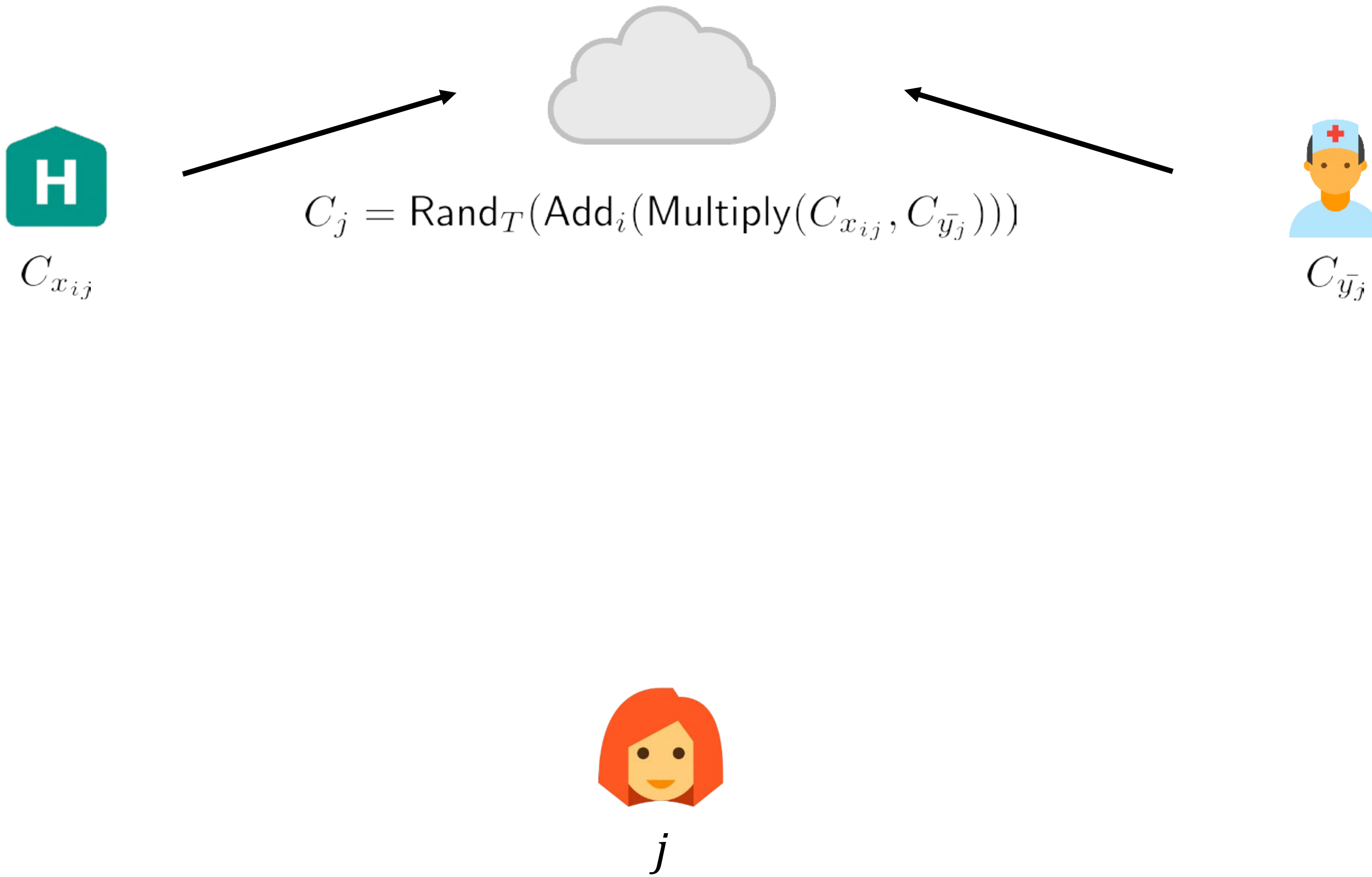
Solution: Group Testing



Solution: Group Testing

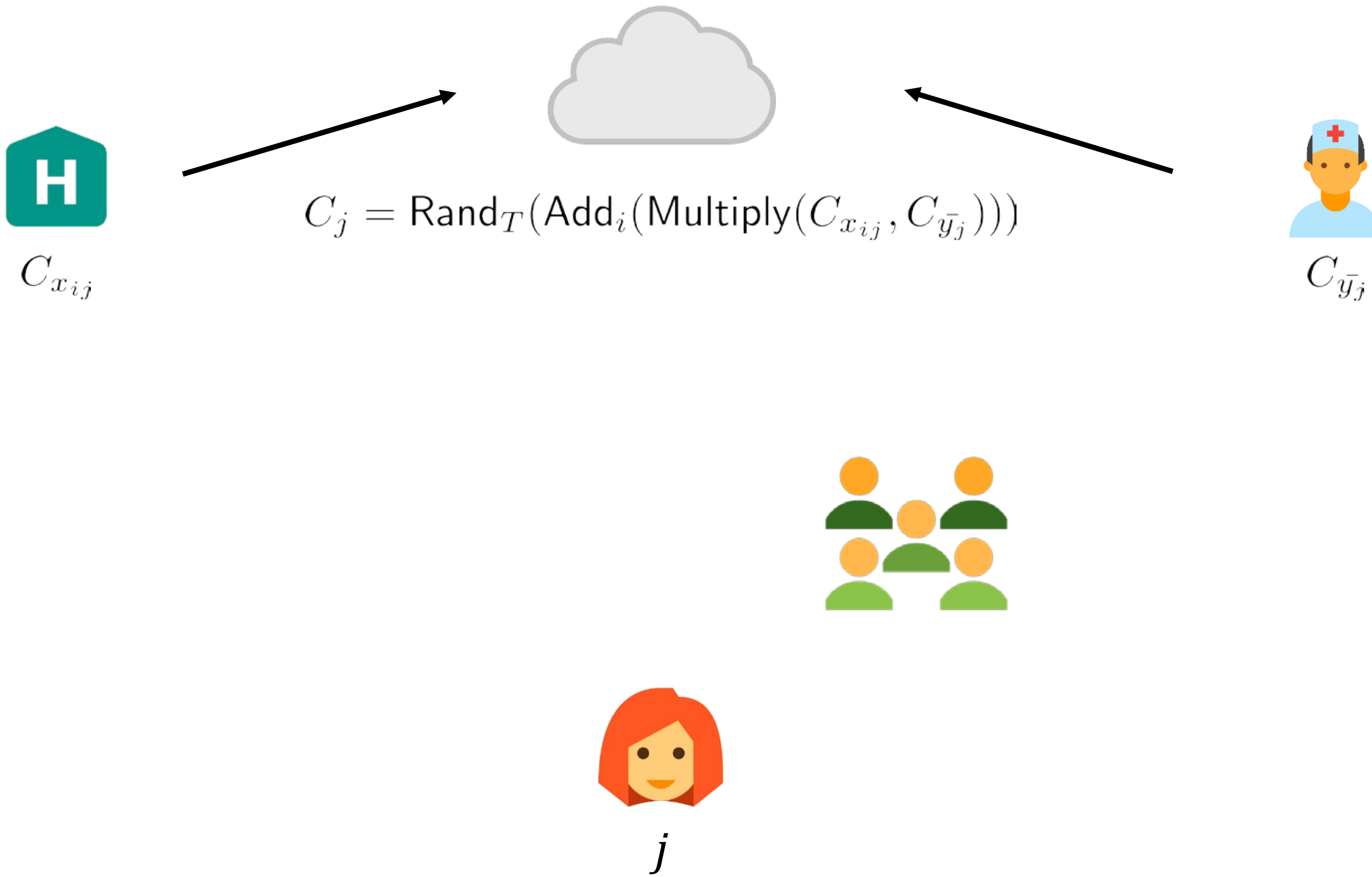


Solution: Group Testing

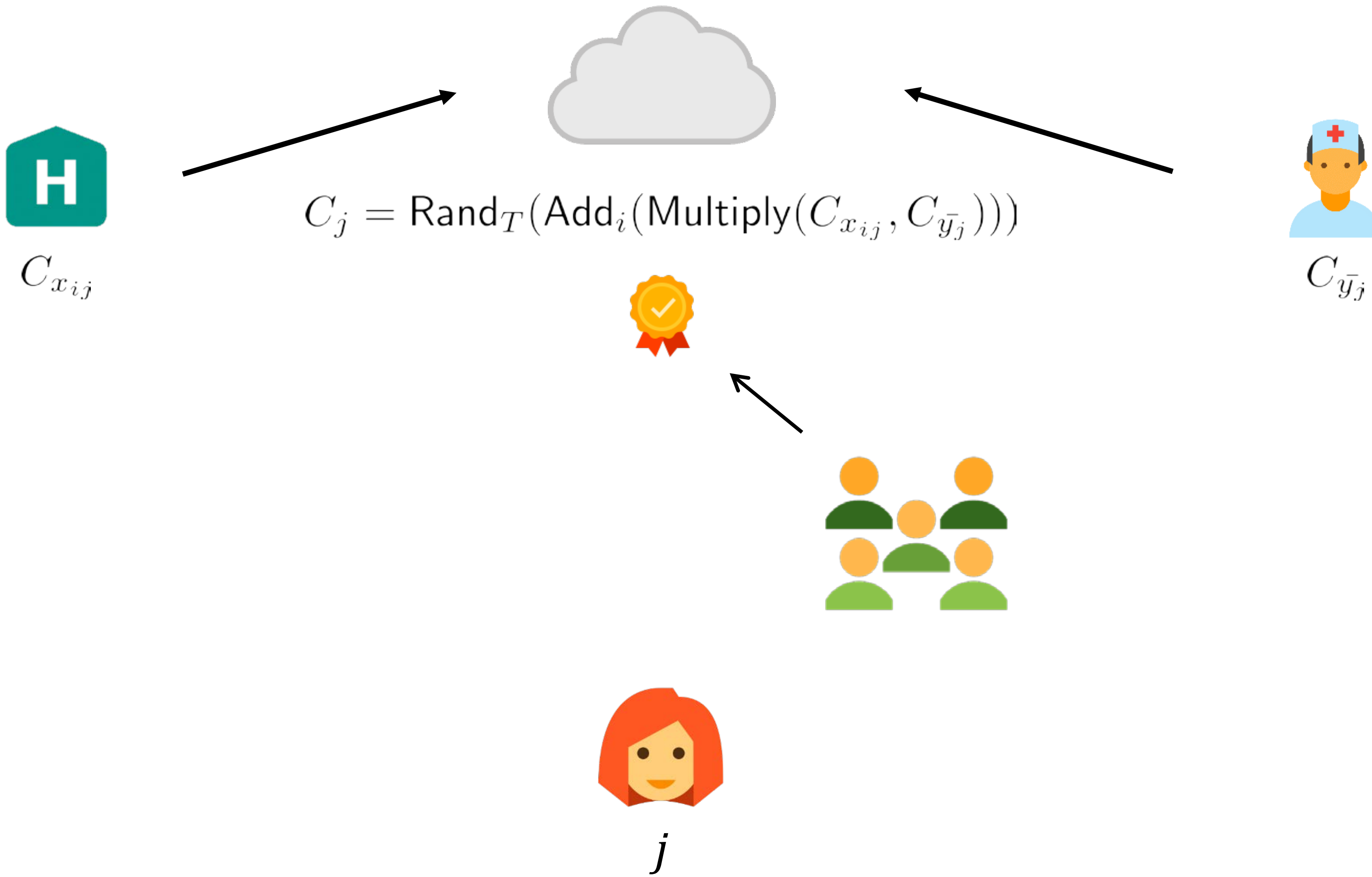


$$C_j = \mathsf{Rand}_T(\mathsf{Add}_i(\mathsf{Multiply}(C_{x_{ij}}, C_{\bar{y}_j})))$$

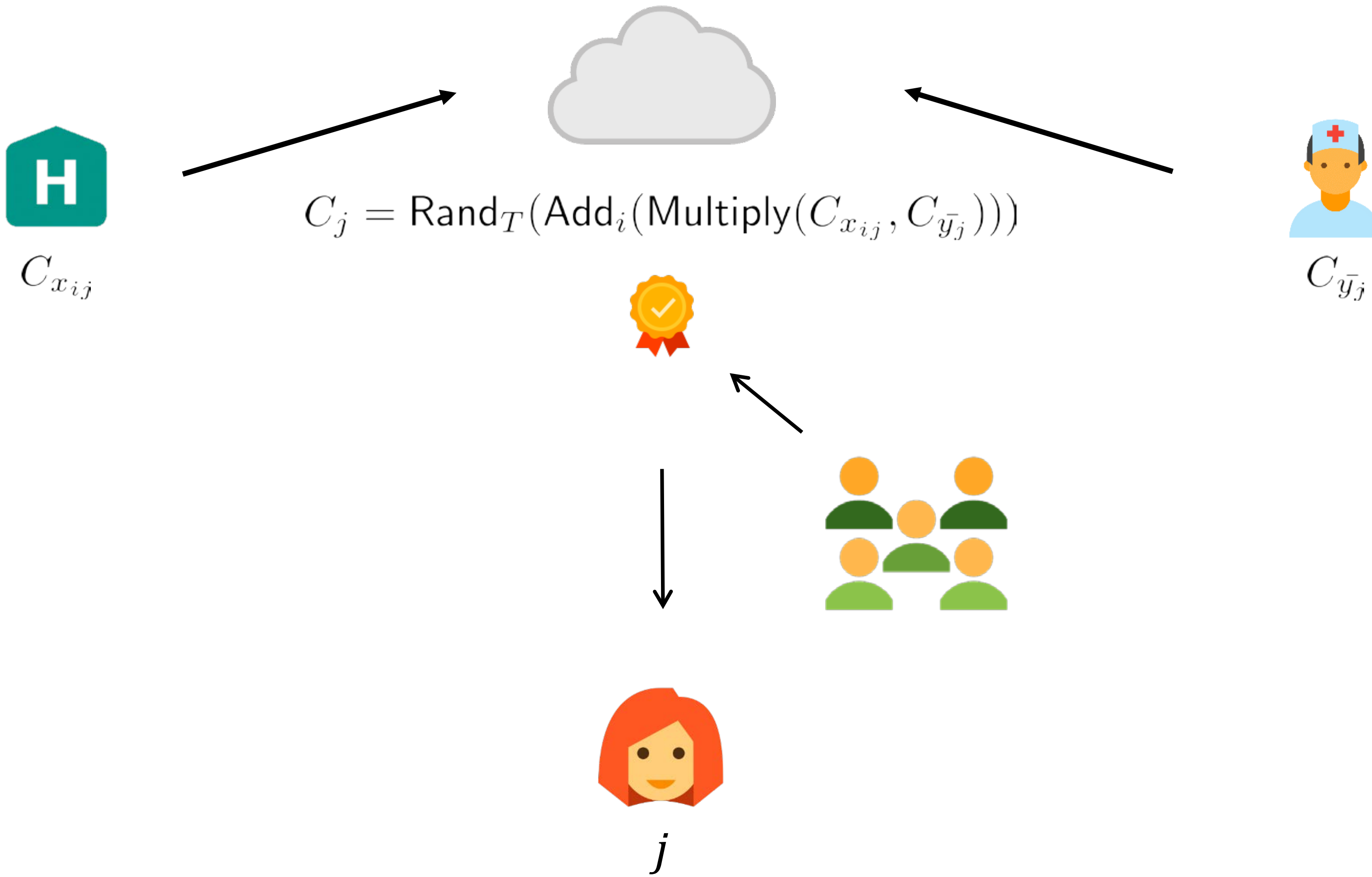
Solution: Group Testing



Solution: Group Testing



Solution: Group Testing



Conclusion

Conclusion

- Efficient scheme to evaluate quadratic multivariate polynomials
  - Distributed decryption
  - Distributed re-encryption
  - Decentralized key generation

Conclusion

- Efficient scheme to evaluate quadratic multivariate polynomials
  - Distributed decryption
  - Distributed re-encryption
  - Decentralized key generation

- Open problem:

Efficient Decentralized FHE

Thank you

ia.cr/2018/1019