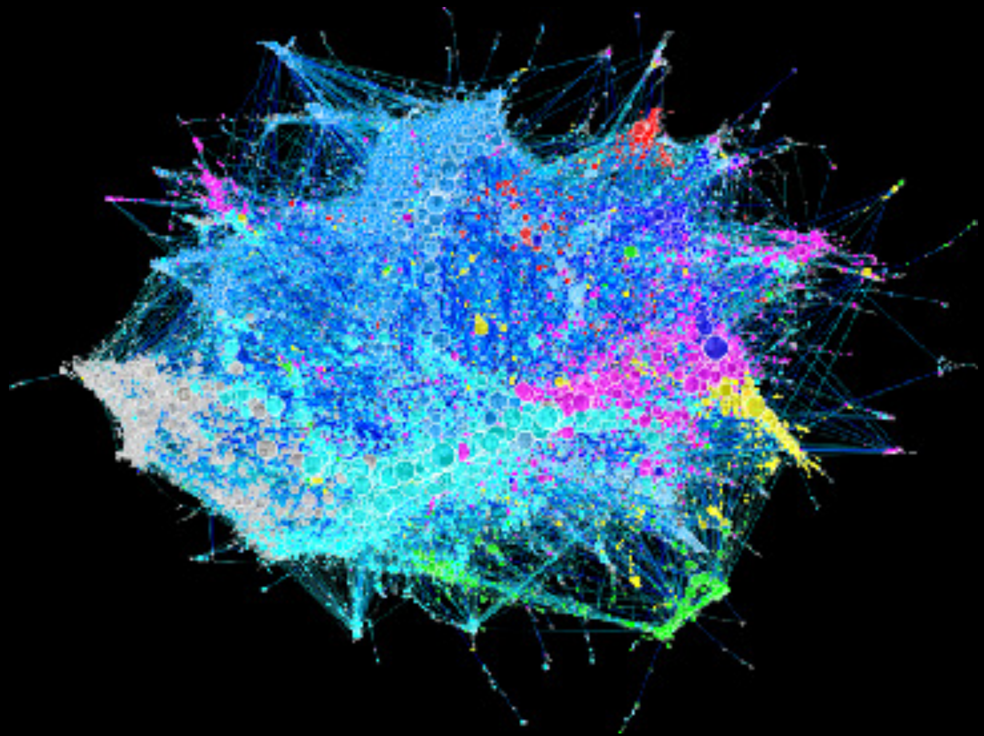
CISPA
HELMHOLTZ CENTER FOR INFORMATION SECURITY

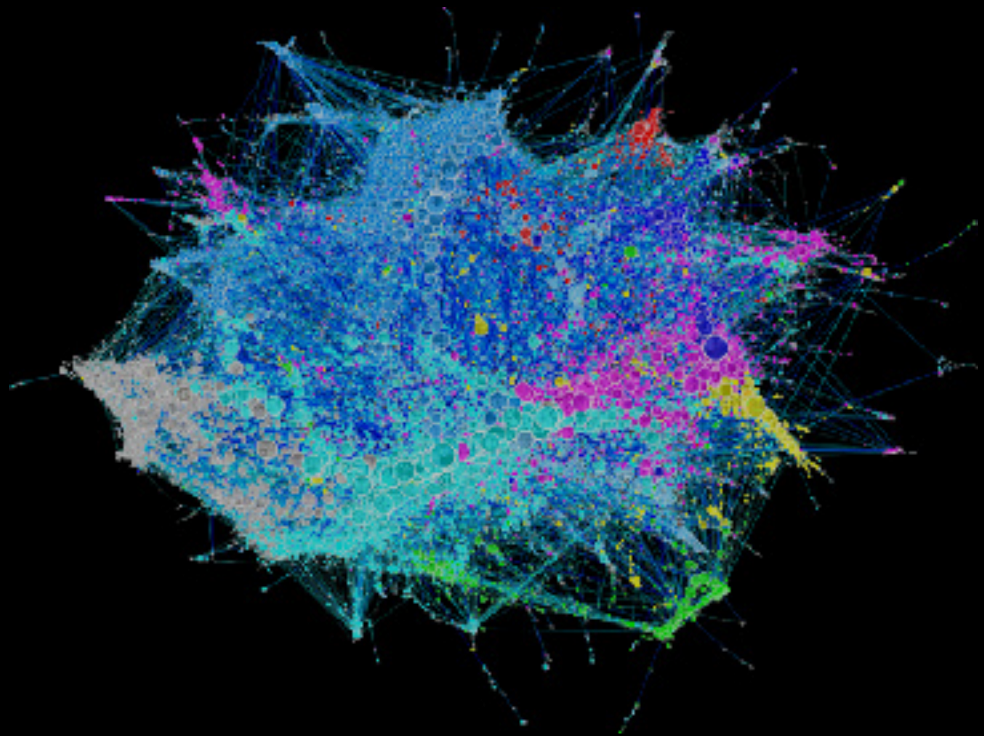
European Research Council
Established by the European Commission

Laconic Cryptography: New Paradigms, Constructions and Directions

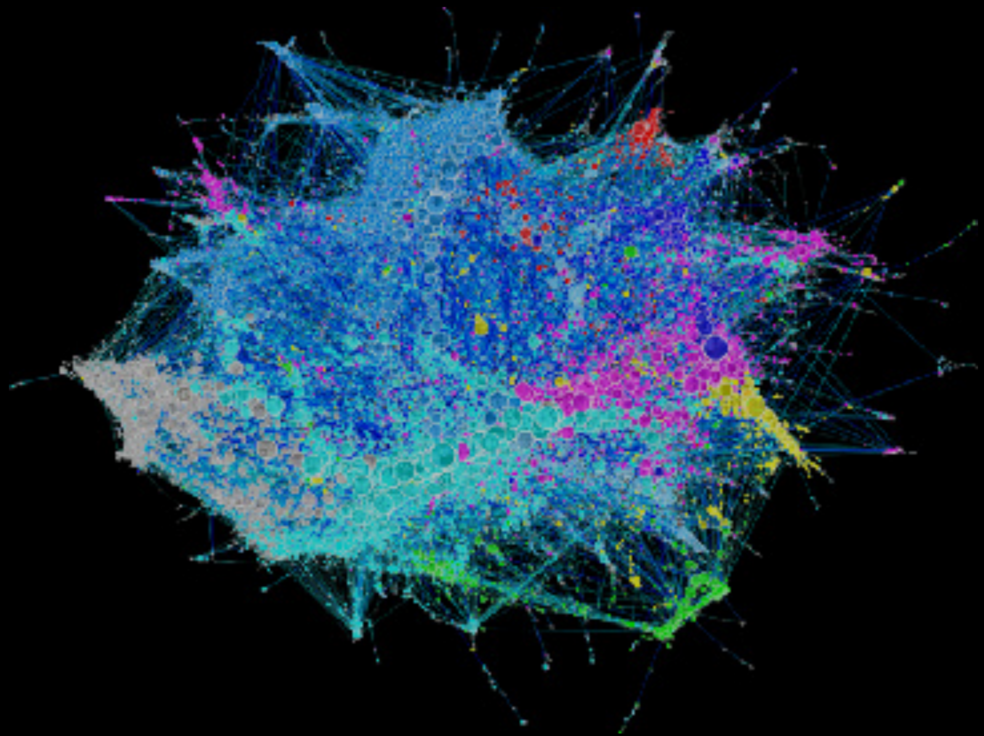
Nico Döttling | 06.03.2024

la·con·ic: using or involving the use of a minimum of words [Merriam Webster]



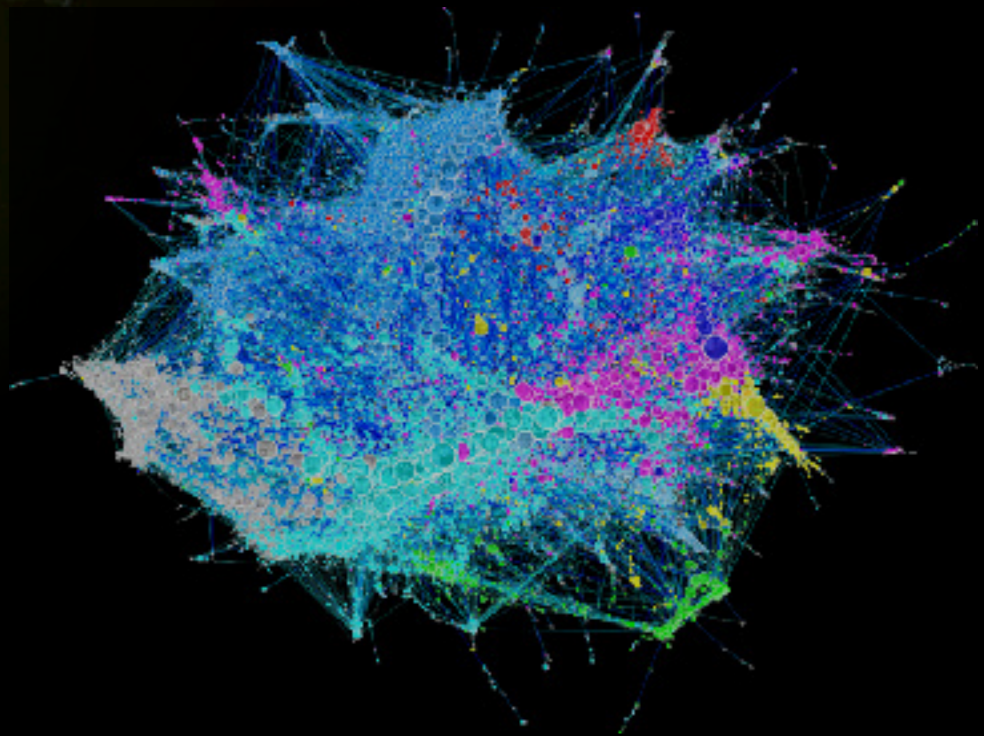


Data-Driven
Methods



Advanced
Cryptography

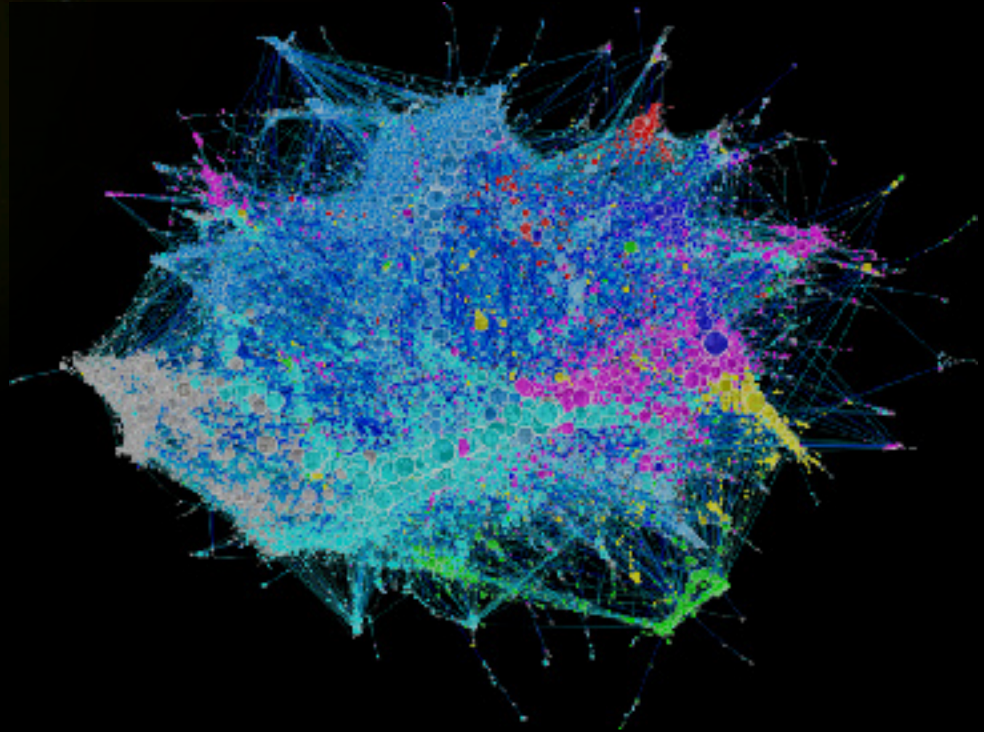
Data-Driven
Methods



Advanced Cryptography

Data-Driven Methods

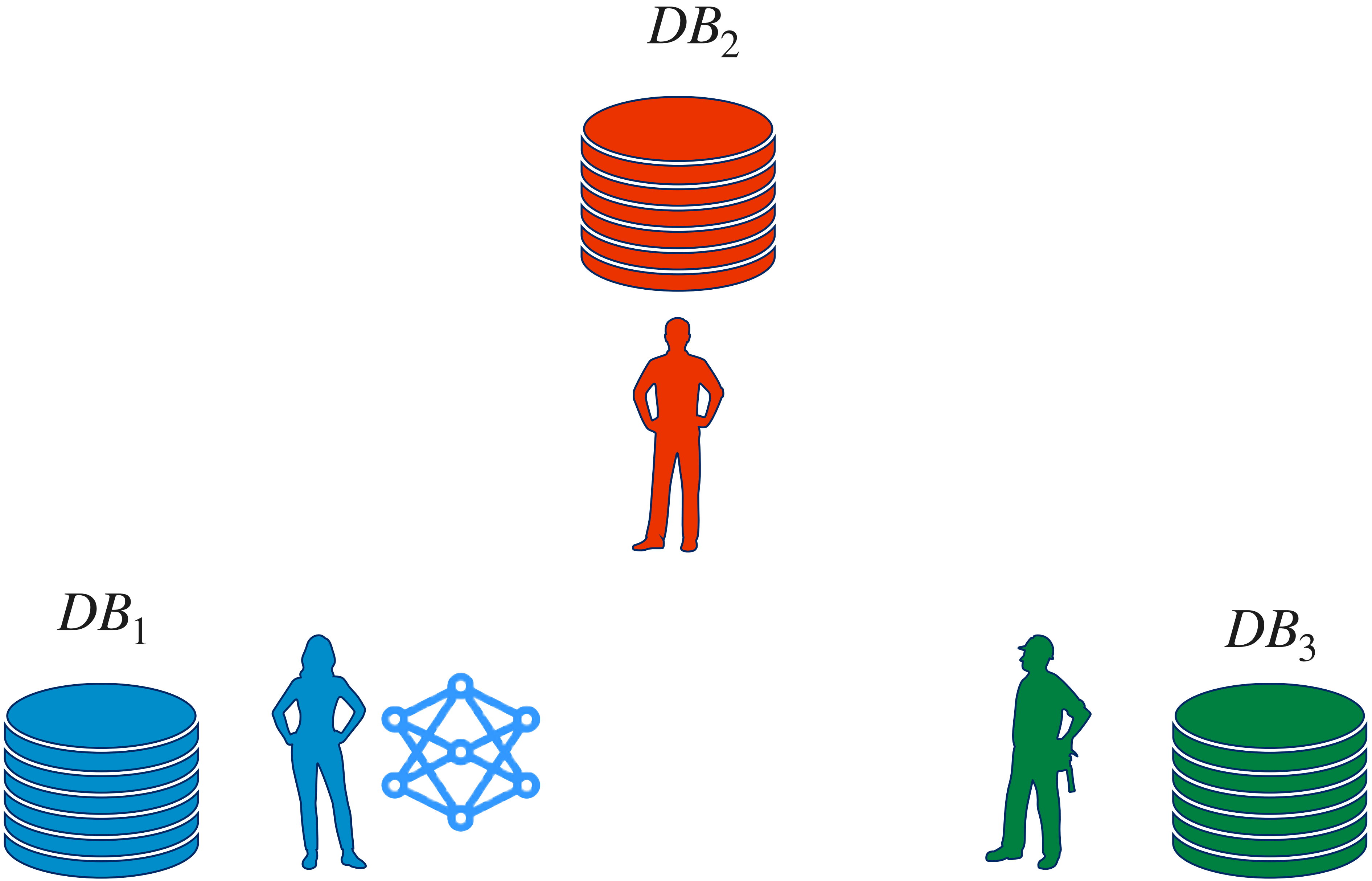
Advanced
Cryptography



Communication &
Computational Overhead

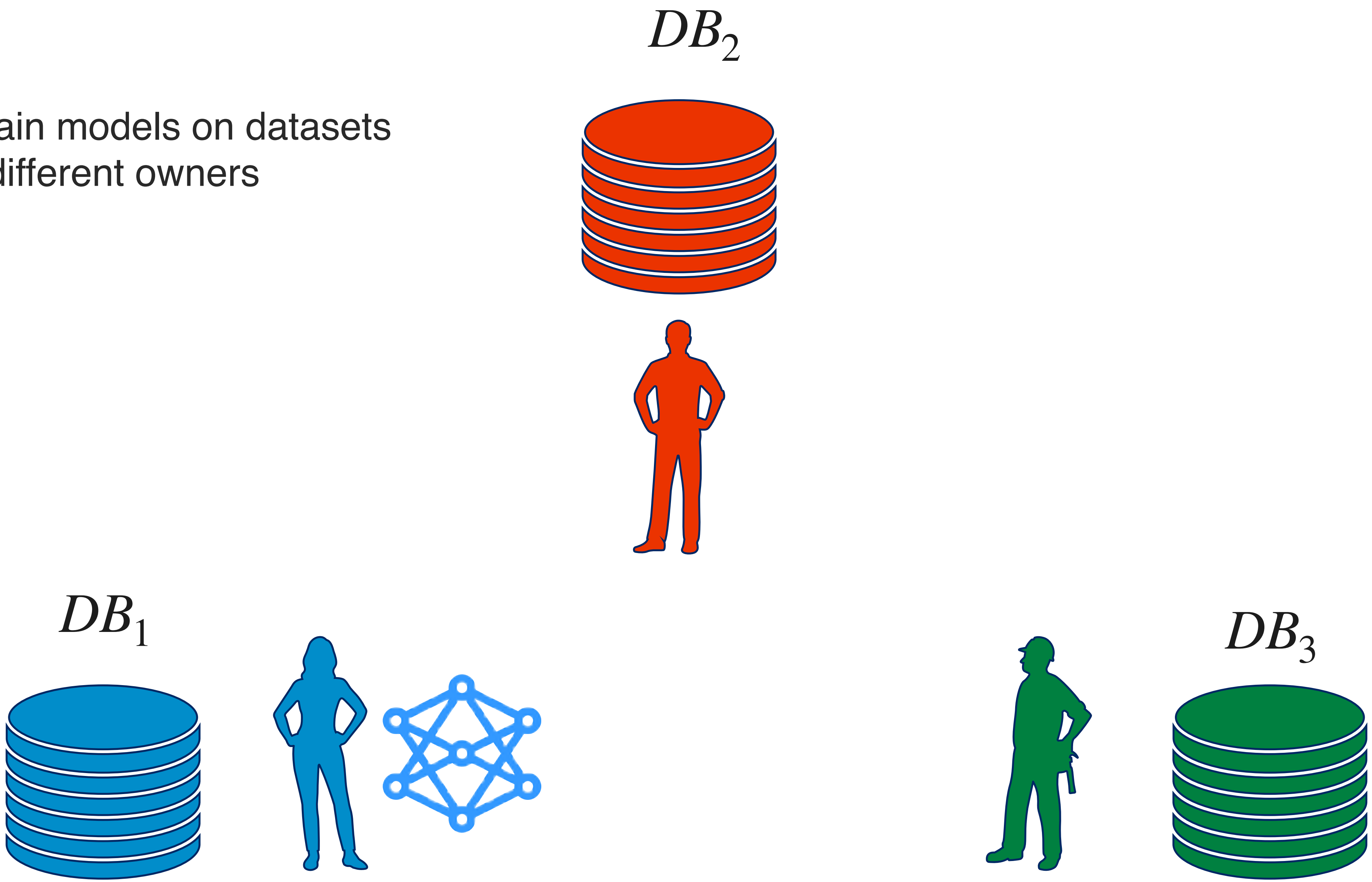
Data-Driven
Methods

Motivation: Federated Learning



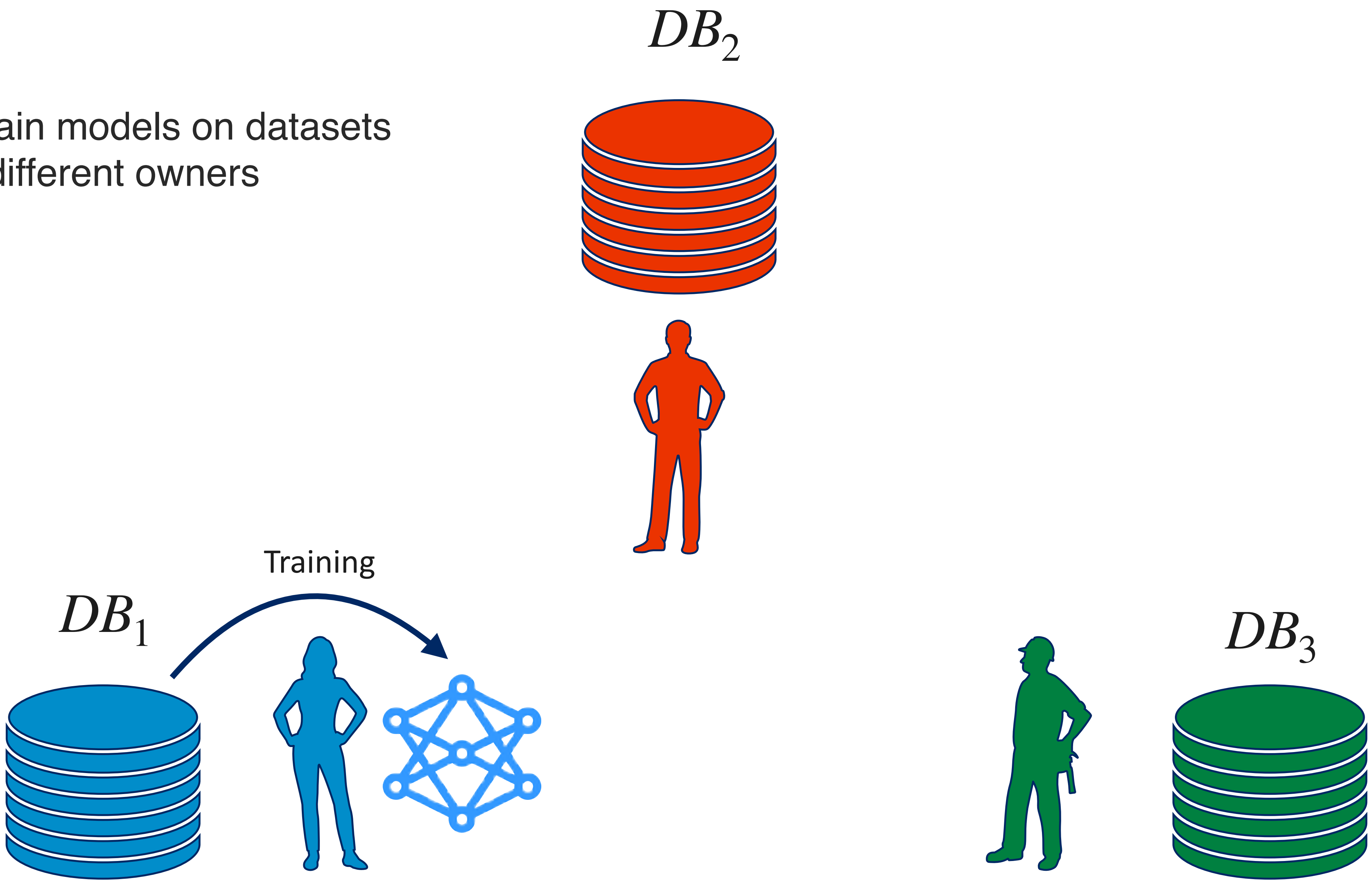
Motivation: Federated Learning

- Jointly train models on datasets held by different owners



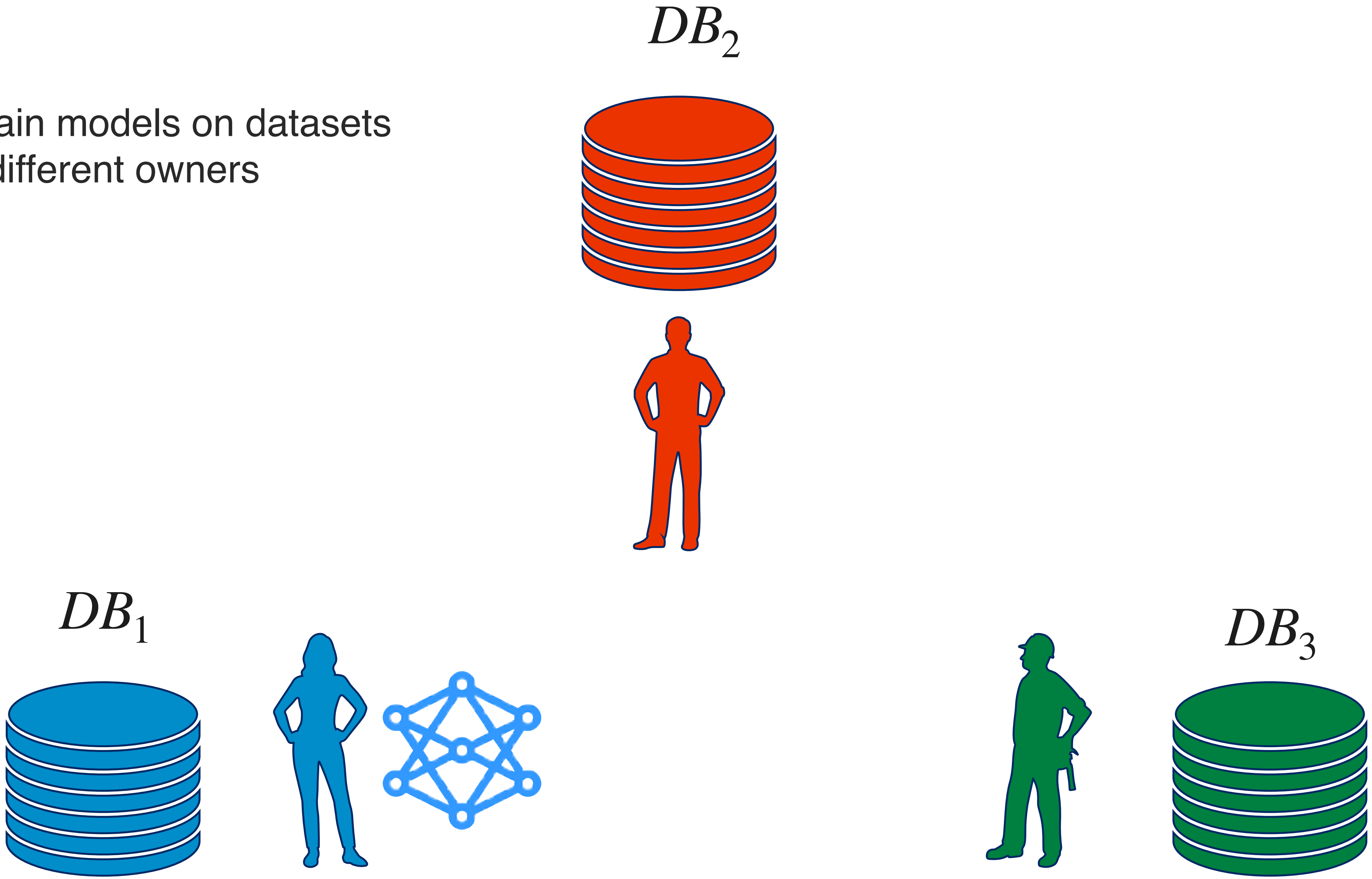
Motivation: Federated Learning

- Jointly train models on datasets held by different owners



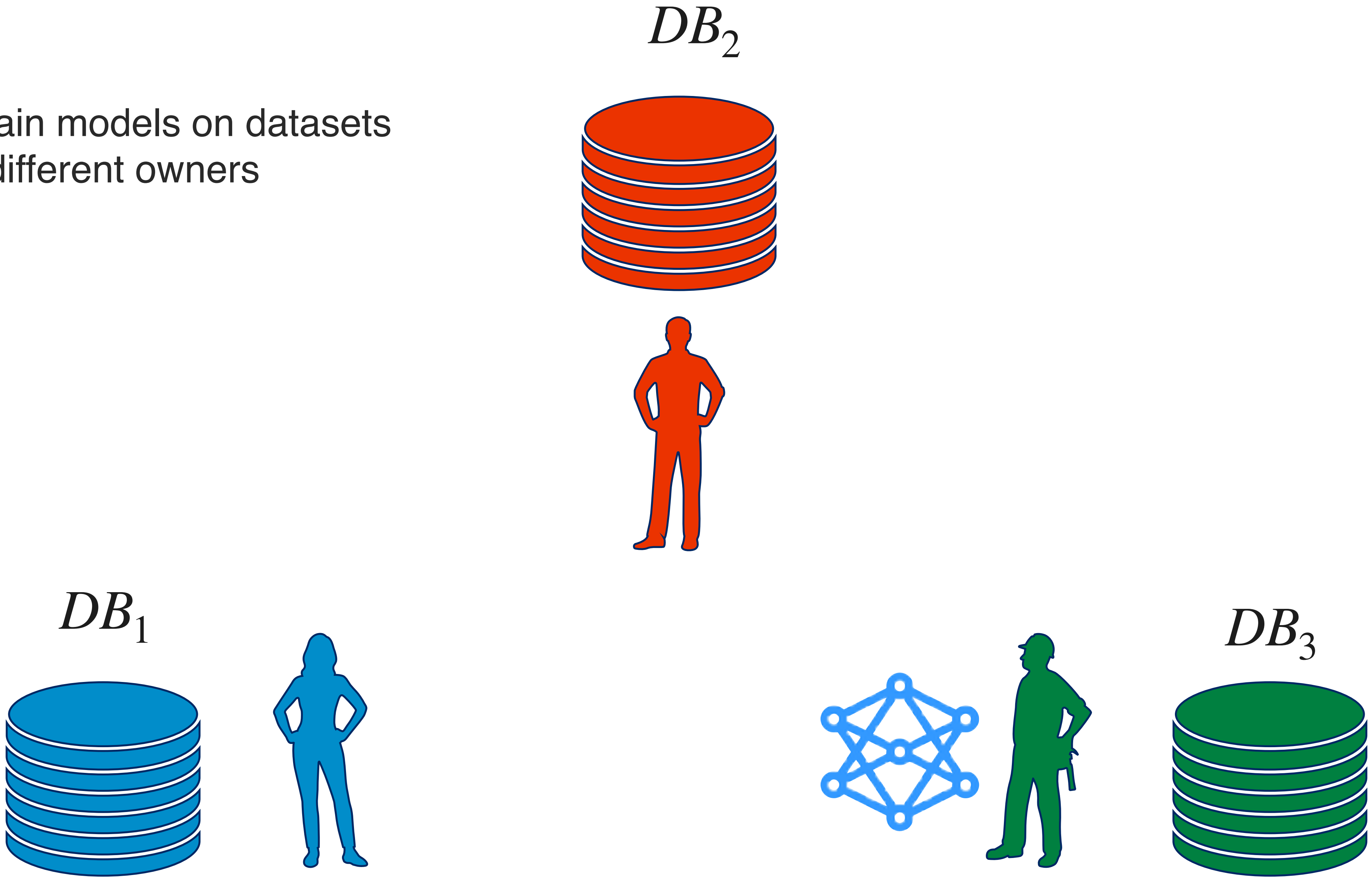
Motivation: Federated Learning

- Jointly train models on datasets held by different owners



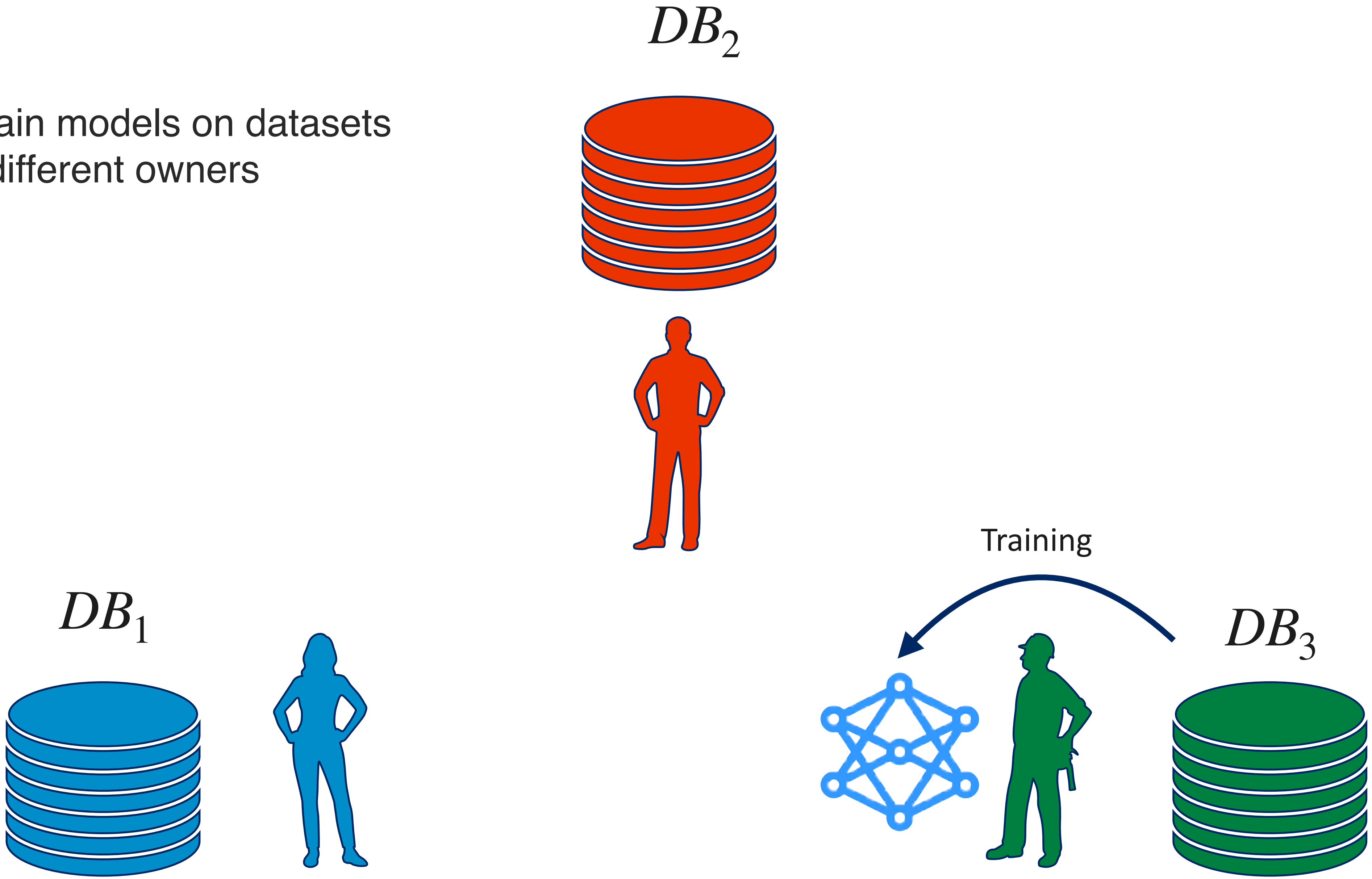
Motivation: Federated Learning

- Jointly train models on datasets held by different owners



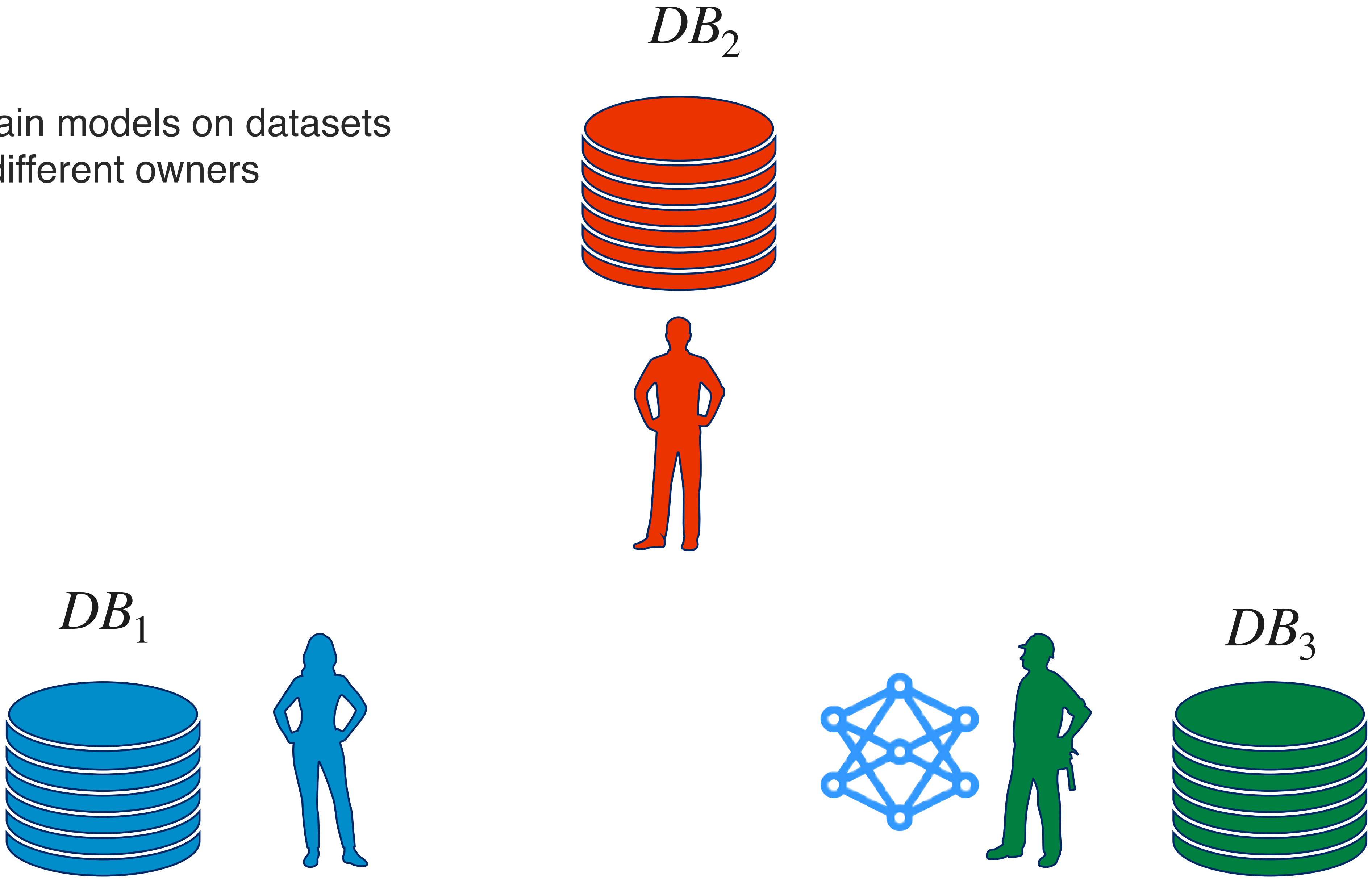
Motivation: Federated Learning

- Jointly train models on datasets held by different owners



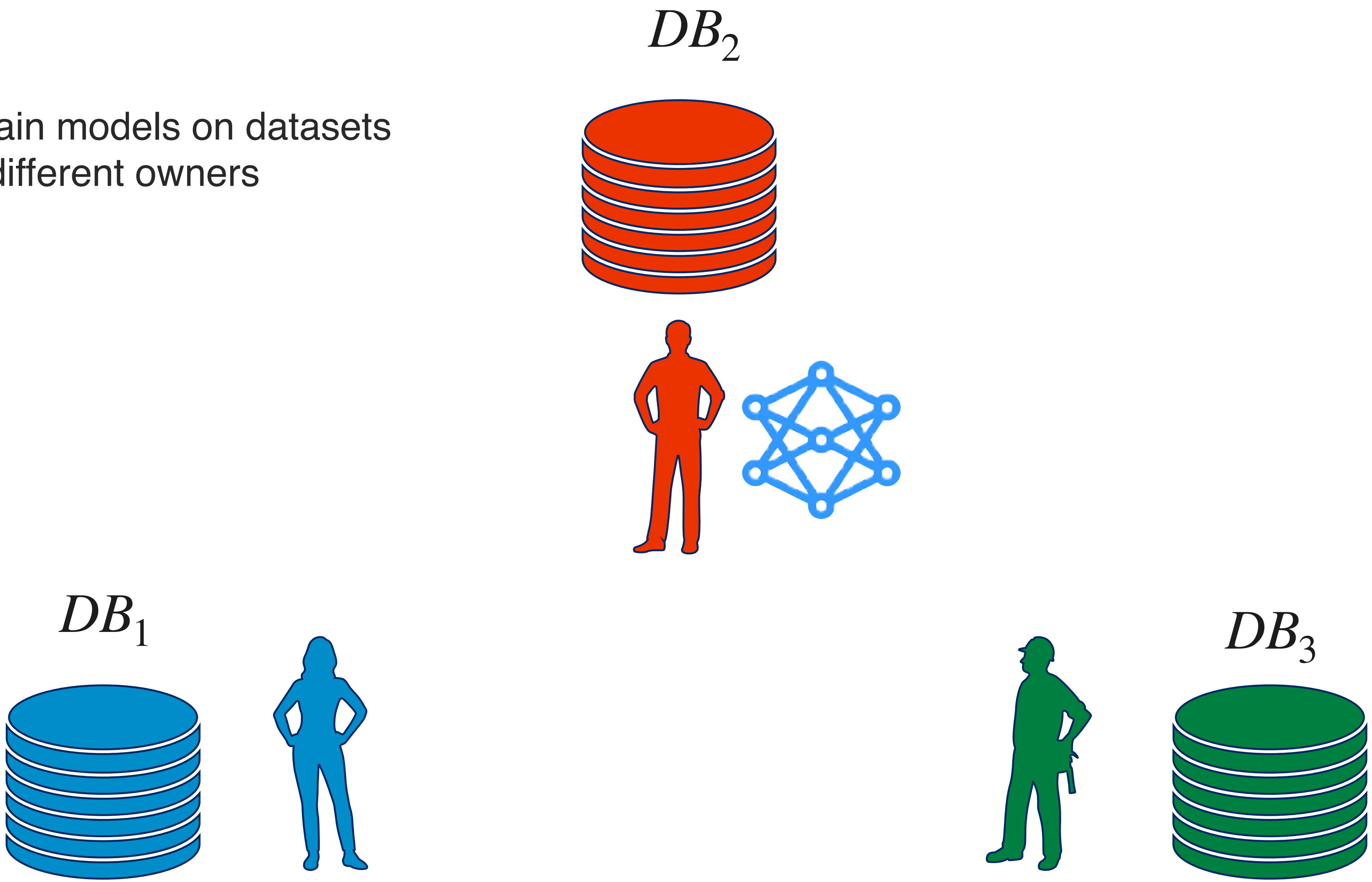
Motivation: Federated Learning

- Jointly train models on datasets held by different owners



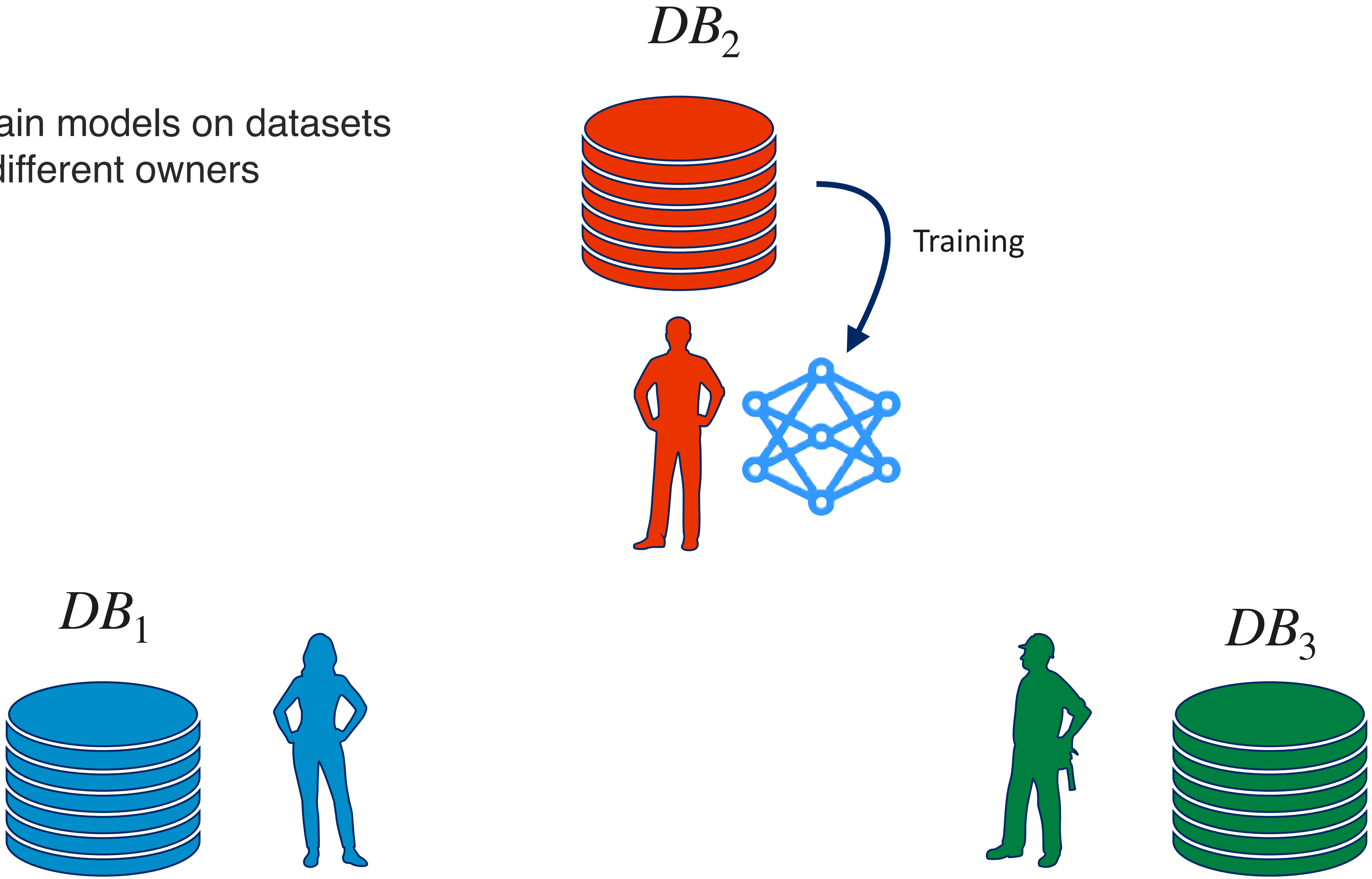
Motivation: Federated Learning

- Jointly train models on datasets held by different owners



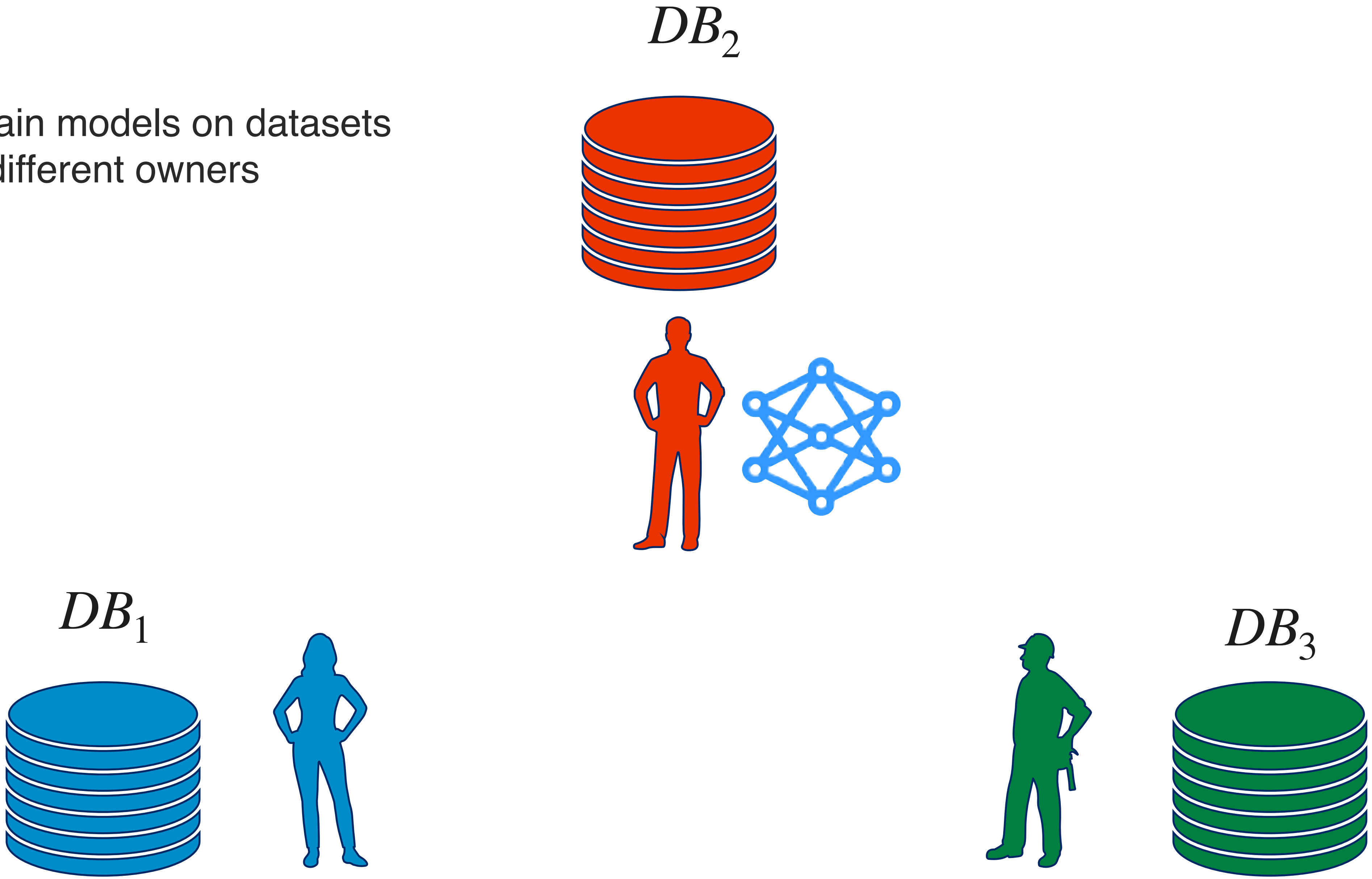
Motivation: Federated Learning

- Jointly train models on datasets held by different owners



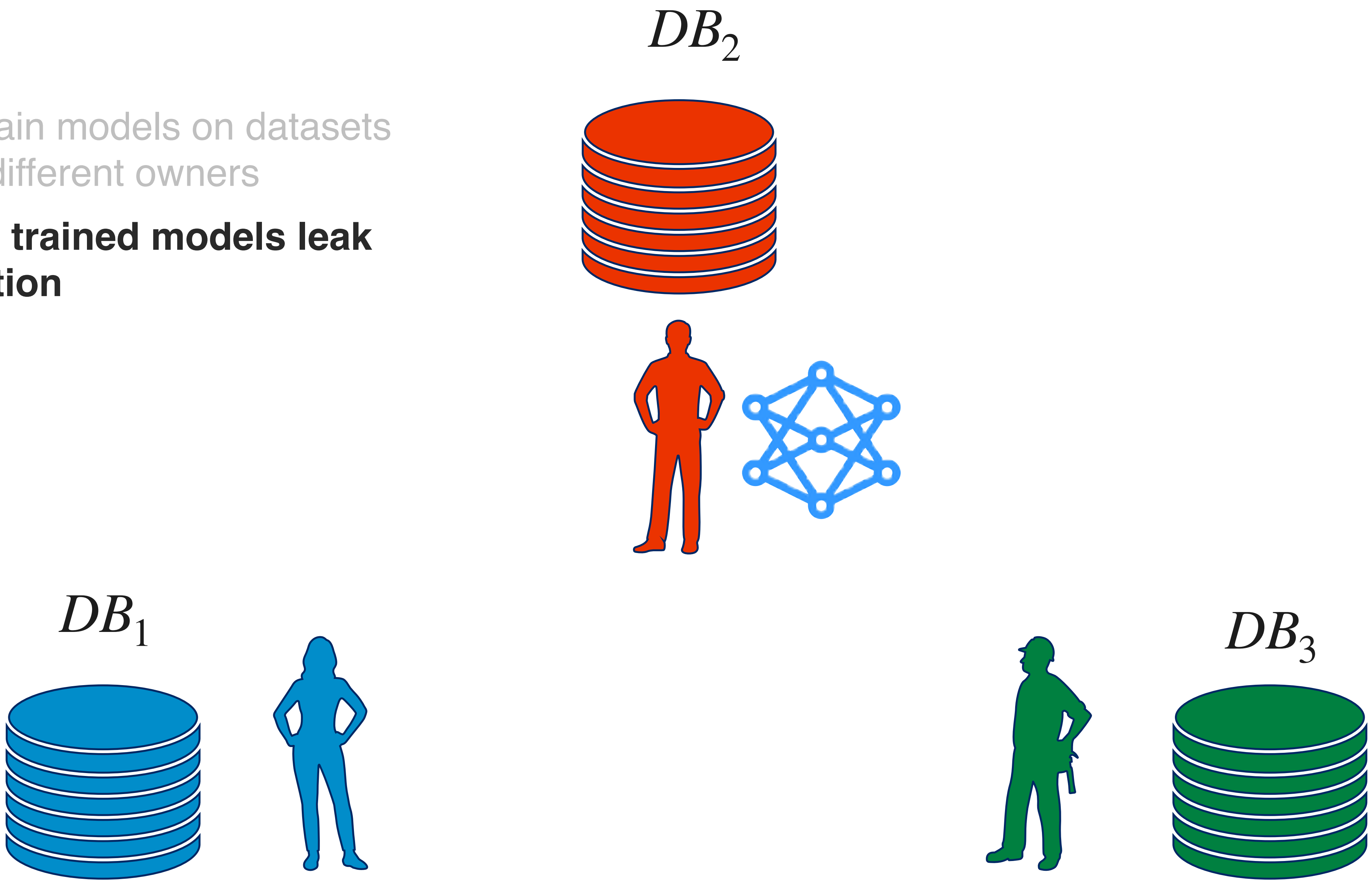
Motivation: Federated Learning

- Jointly train models on datasets held by different owners



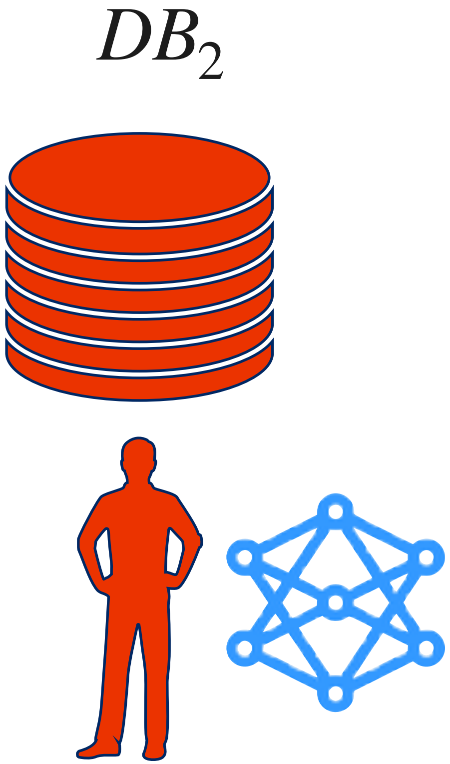
Motivation: Federated Learning

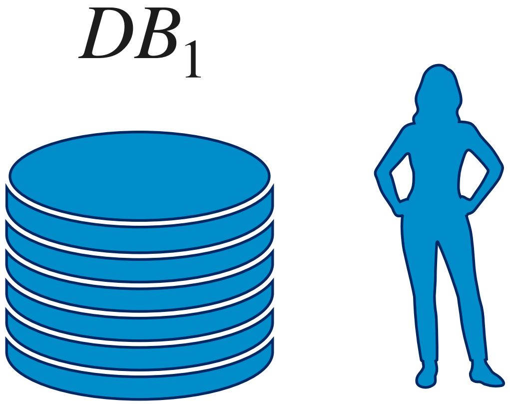
- Jointly train models on datasets held by different owners
- **Partially trained models leak information**

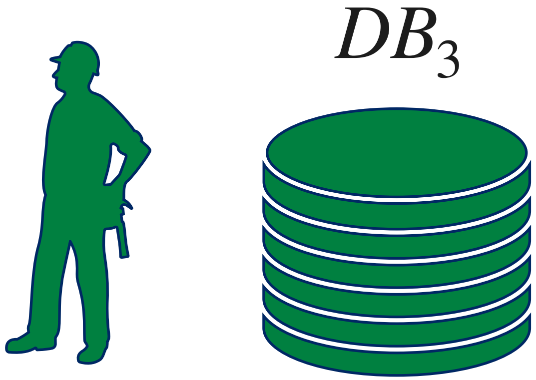


Motivation: Federated Learning

- Jointly train models on datasets held by different owners
- **Partially trained models leak information**
- **Conventional MPC leads to exorbitant communication and computation costs**







Homomorphic Encryption

Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices

Homomorphic Encryption

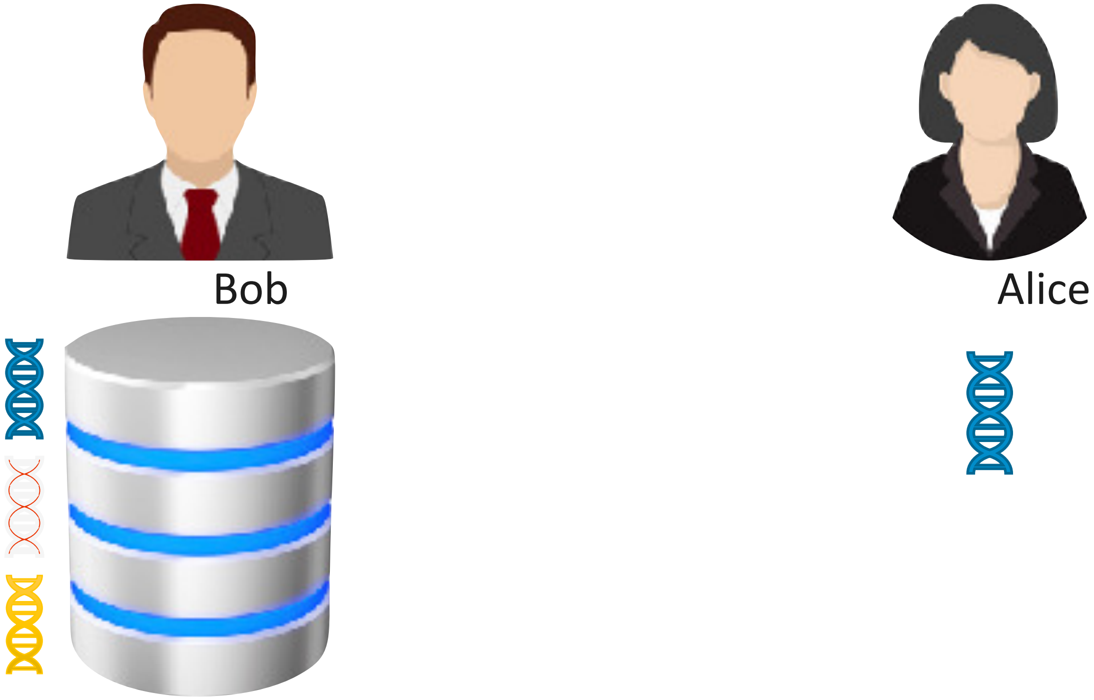
[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices

Bob

Alice

Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices



Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices

Bob

Alice

Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices

Bob

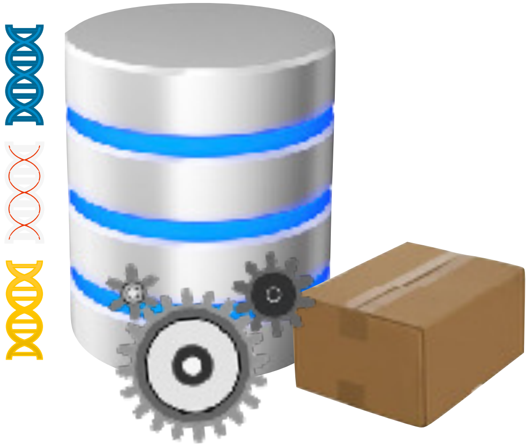
Alice

Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices

Bob

Alice

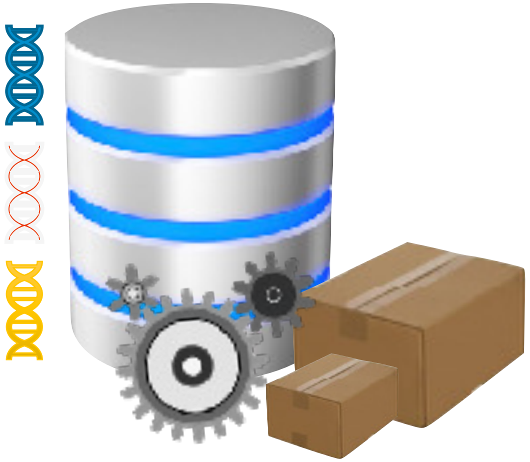


Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices

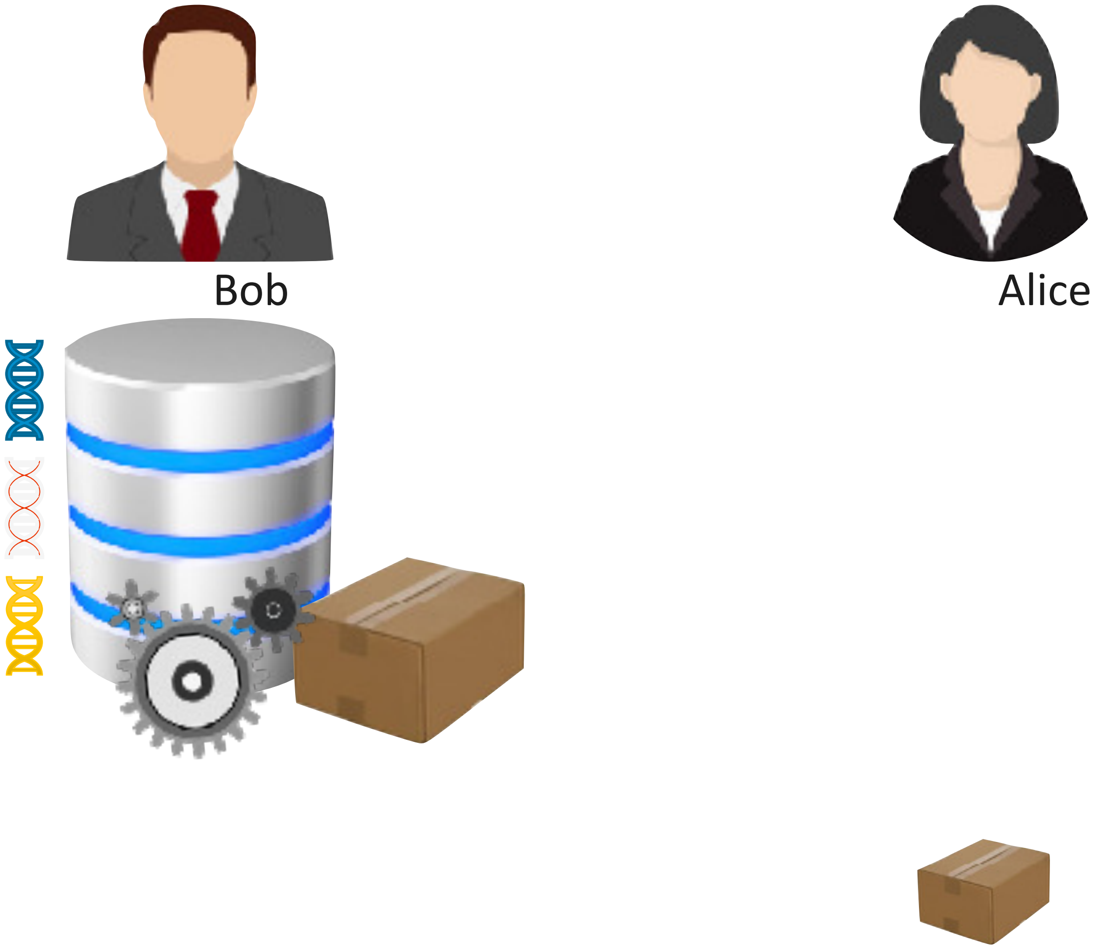
Bob

Alice



Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices

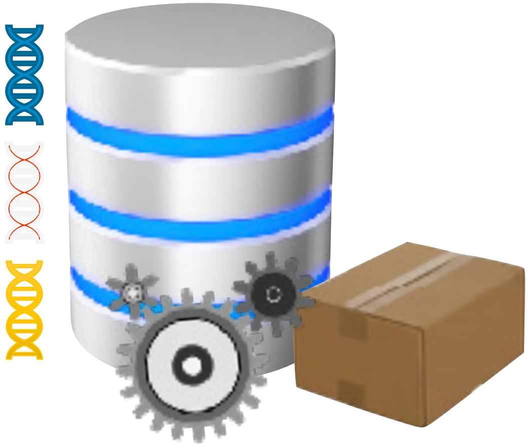


Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices

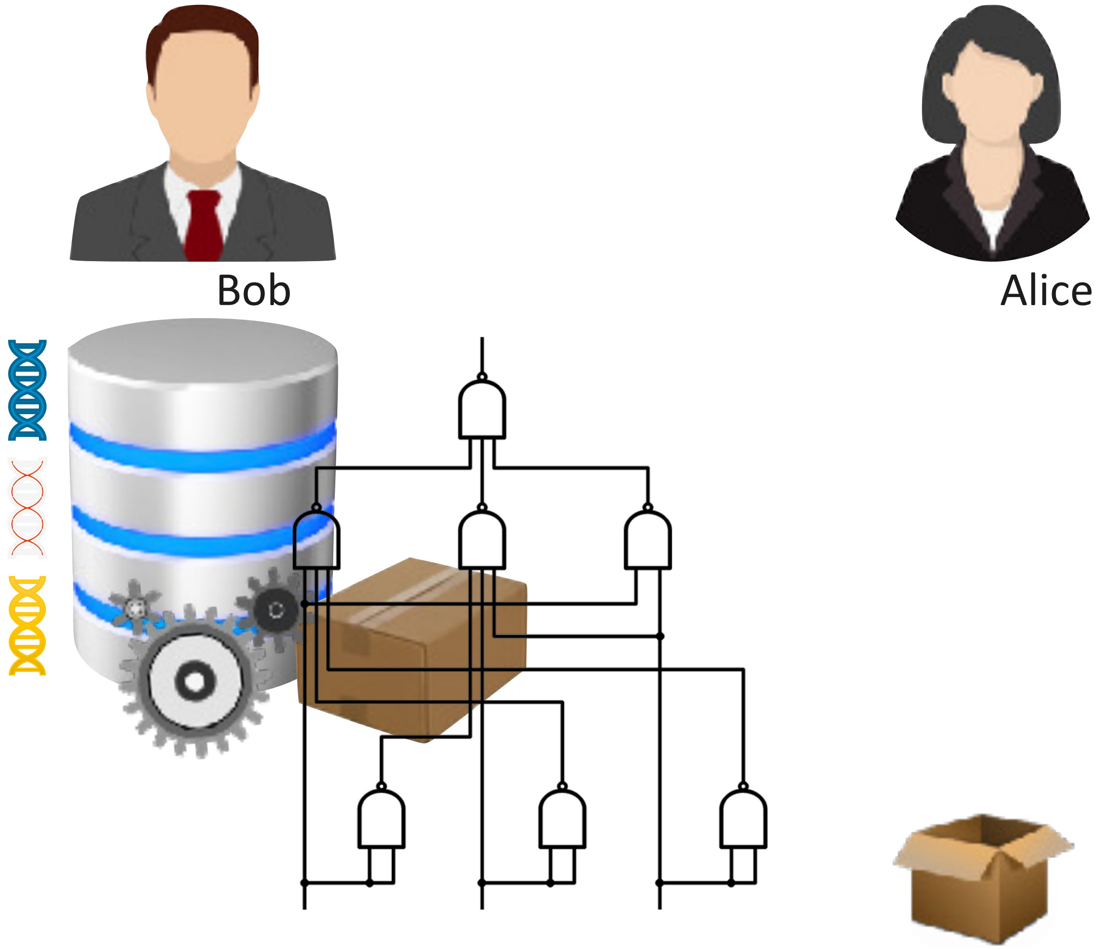
Bob

Alice



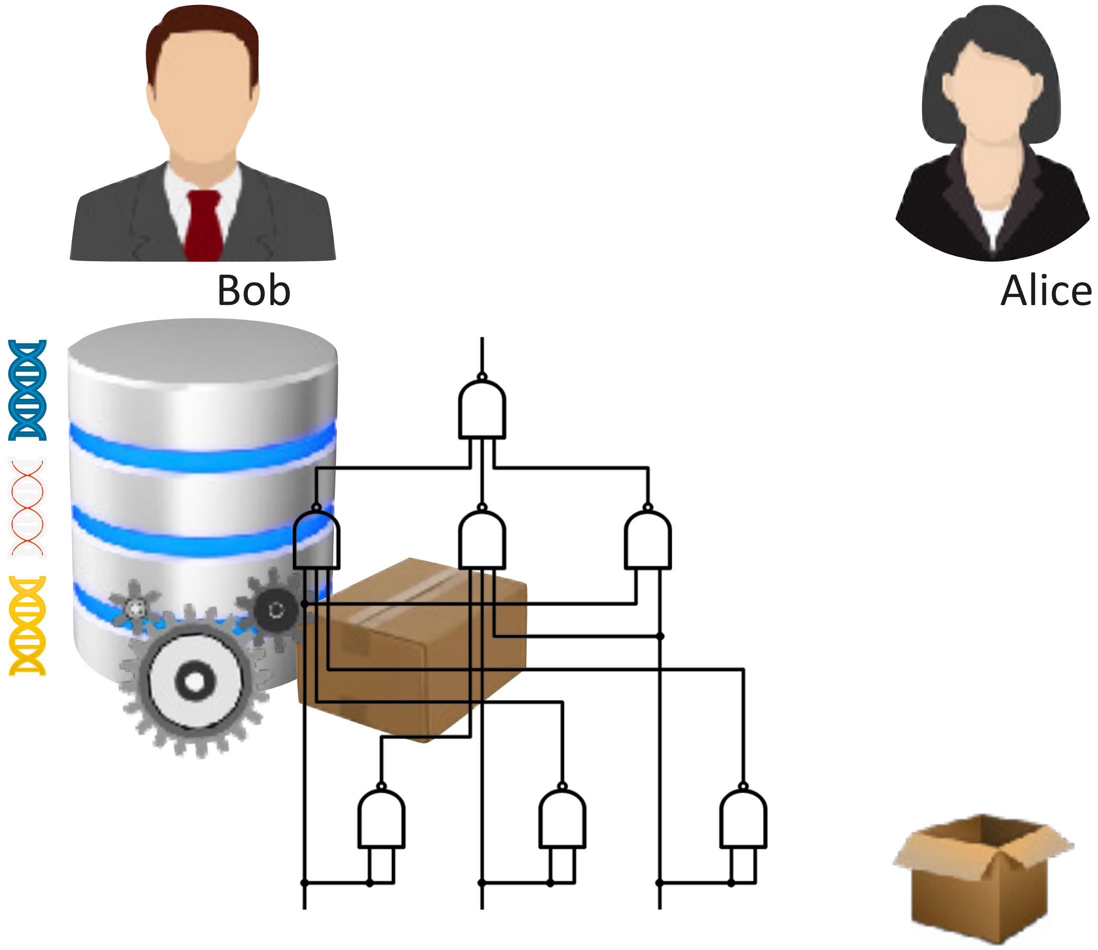
Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices



Homomorphic Encryption

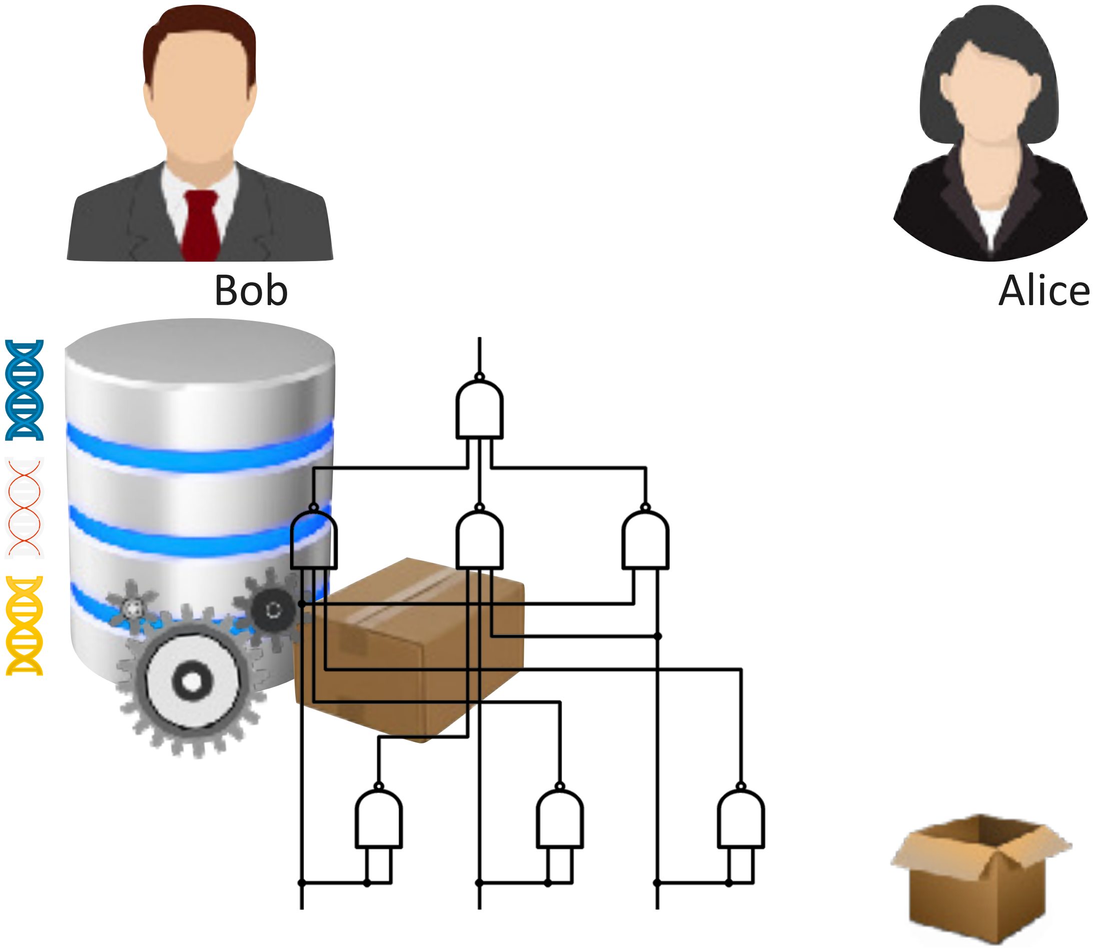
[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices



Laconic Cryptography

Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices

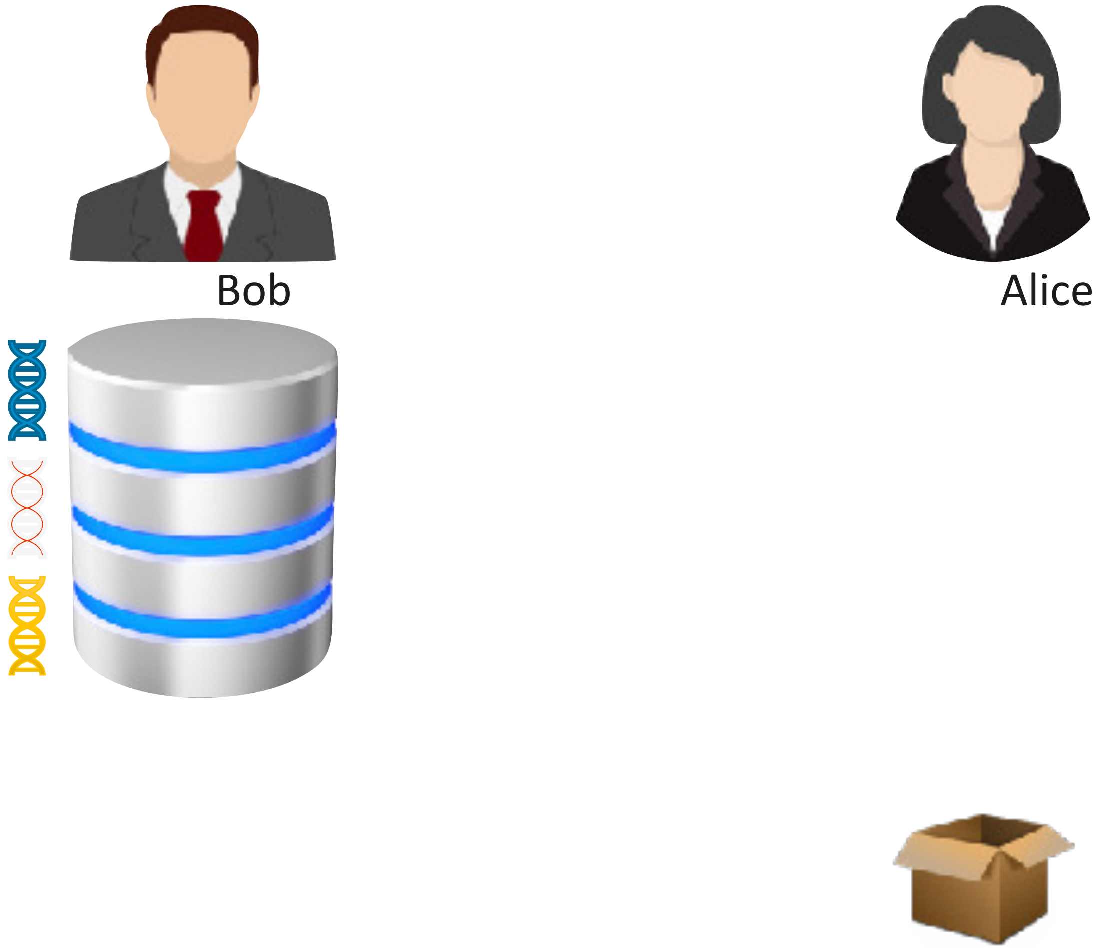


Laconic Cryptography

[CDGGMP CRYPTO'17]
Laconic Oblivious Transfer and Its Applications

Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices

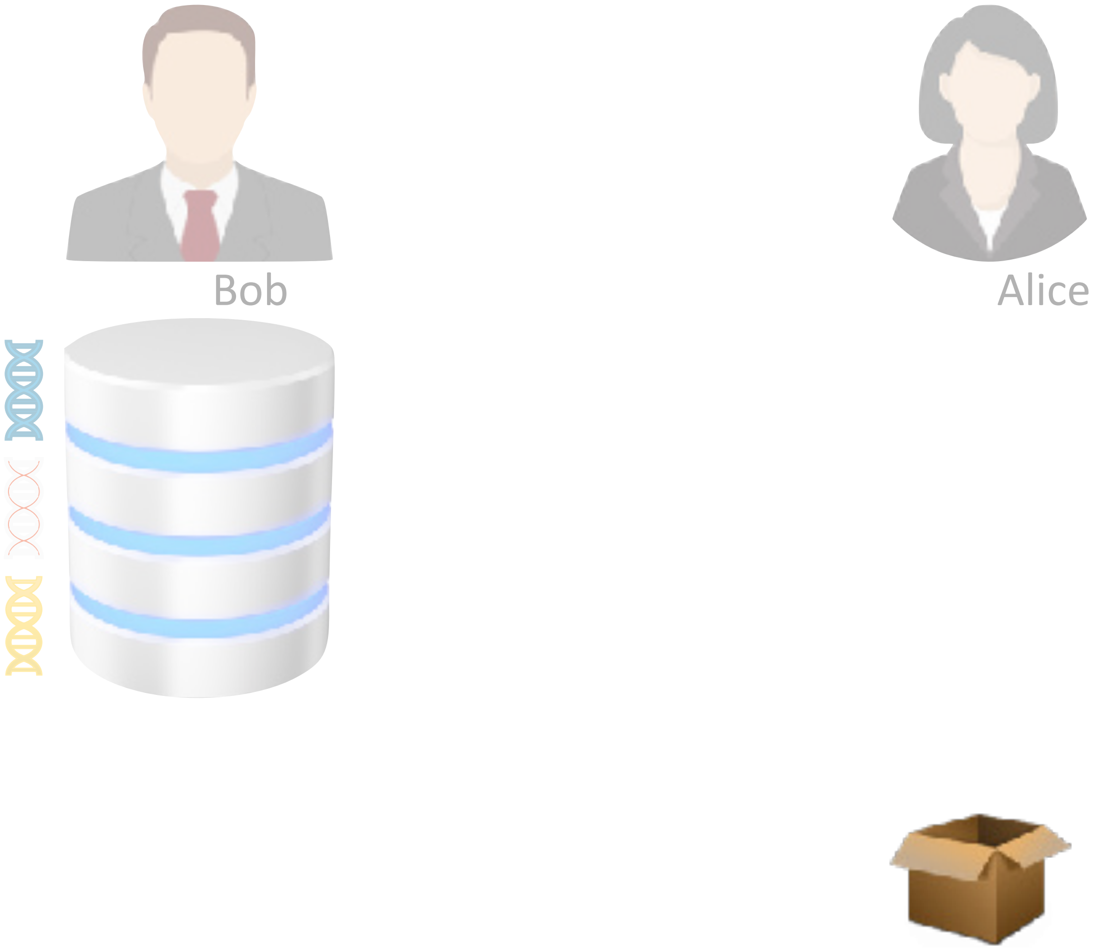


Laconic Cryptography

[CDGGMP CRYPTO’17]
Laconic Oblivious Transfer and Its Applications

Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices

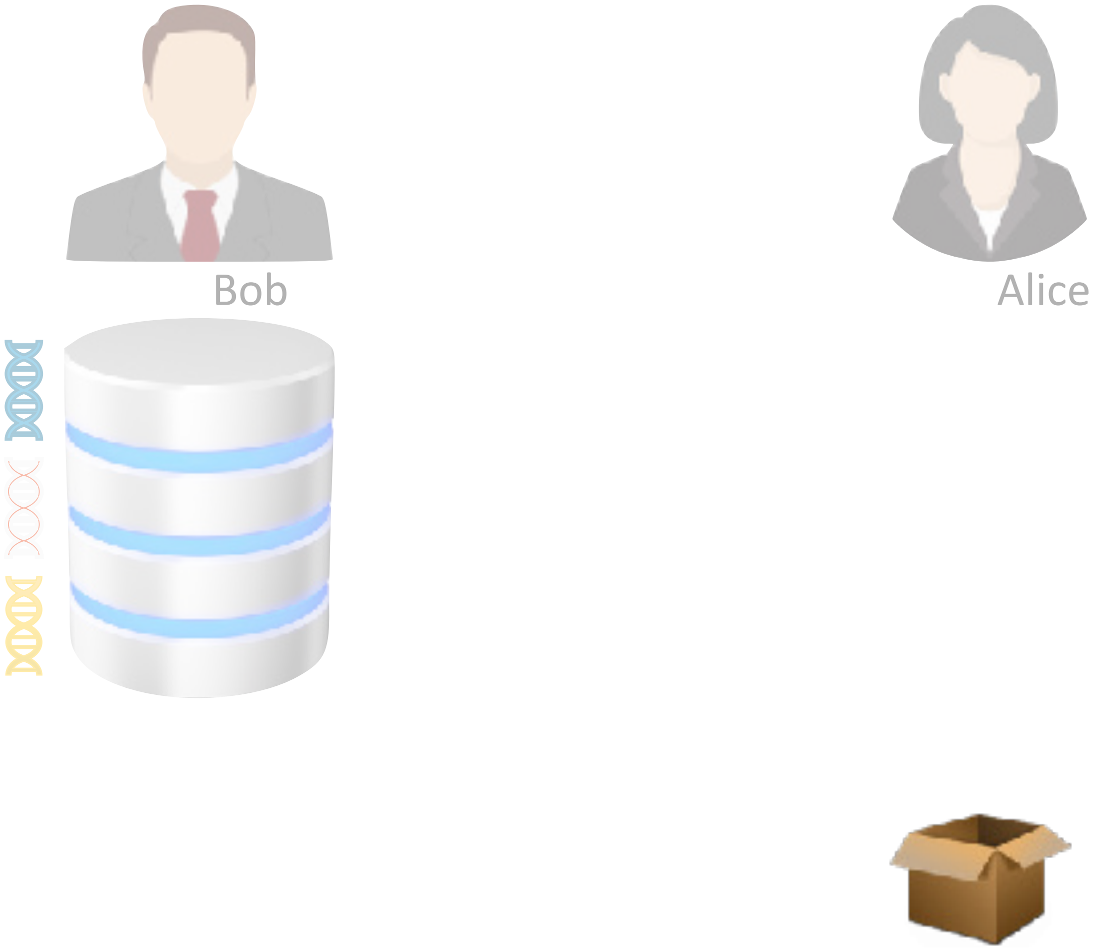


Laconic Cryptography

[C**D**GGMP CRYPTO’17]
Laconic Oblivious Transfer and Its Applications

Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices

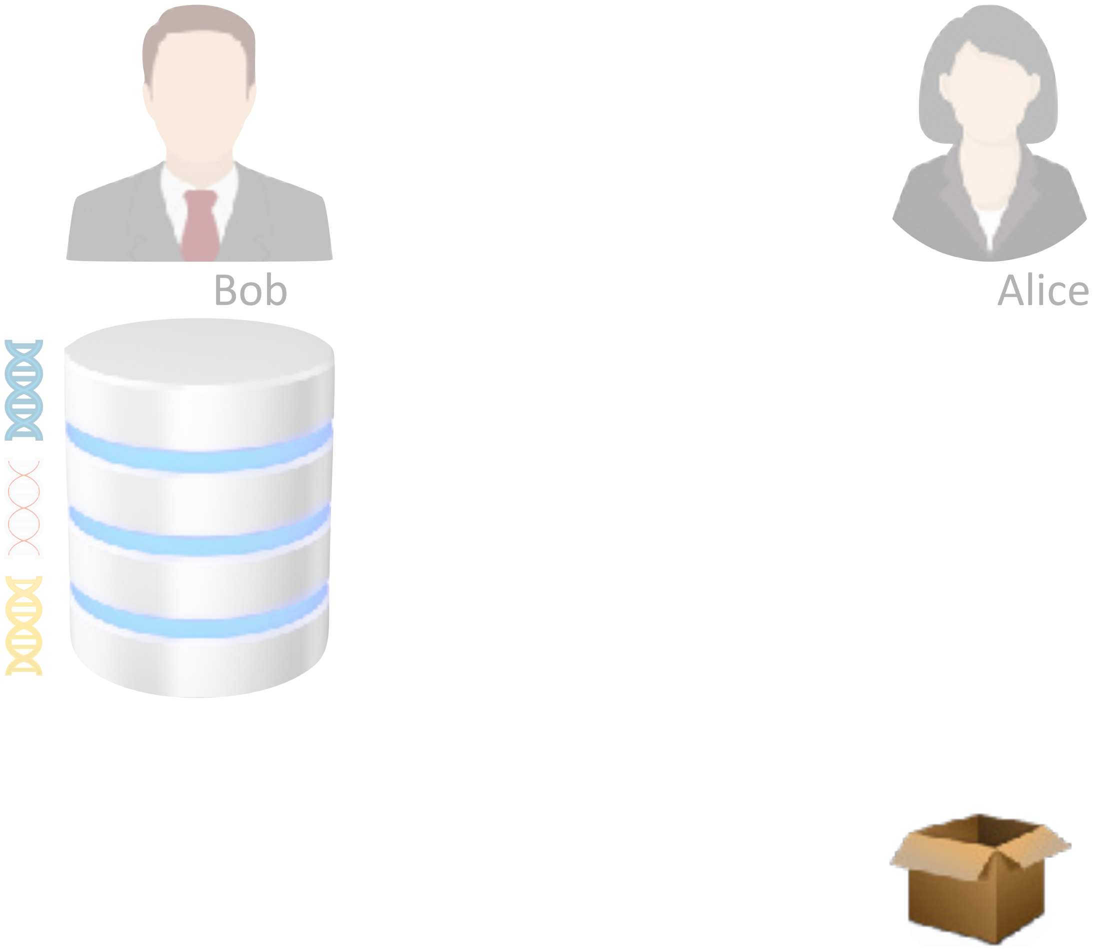


Laconic Cryptography

[CDGGMP CRYPTO'17]
Laconic Oblivious Transfer and Its Applications

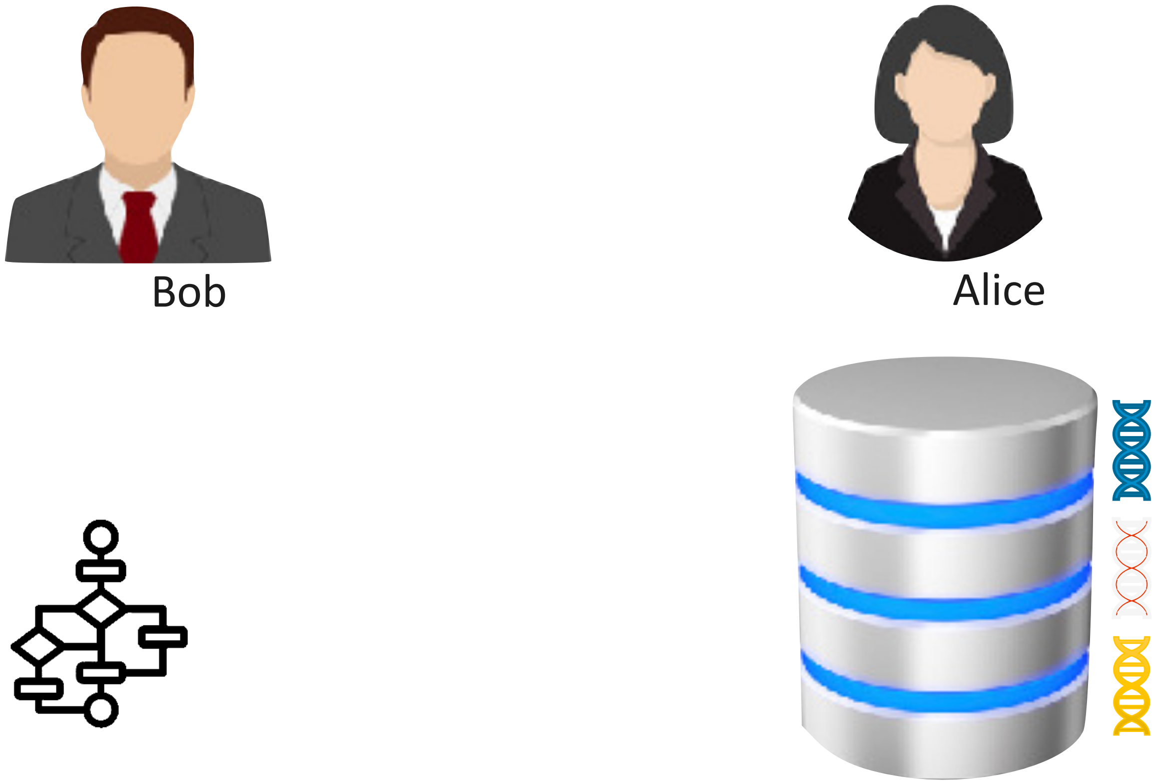
Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices



Laconic Cryptography

[CDGGMP CRYPTO'17]
Laconic Oblivious Transfer and Its Applications

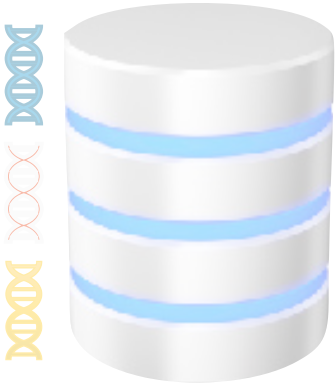


Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices

Bob

Alice

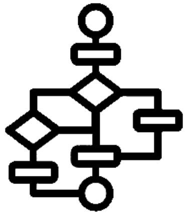


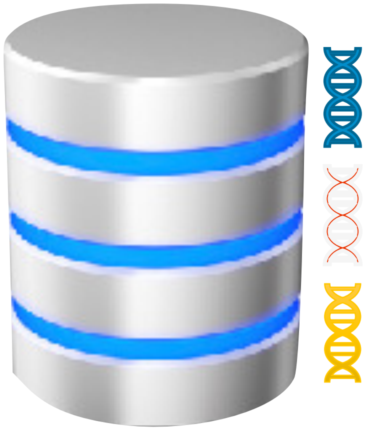
Laconic Cryptography

[CDGGMP CRYPTO’17]
Laconic Oblivious Transfer and Its Applications

Bob

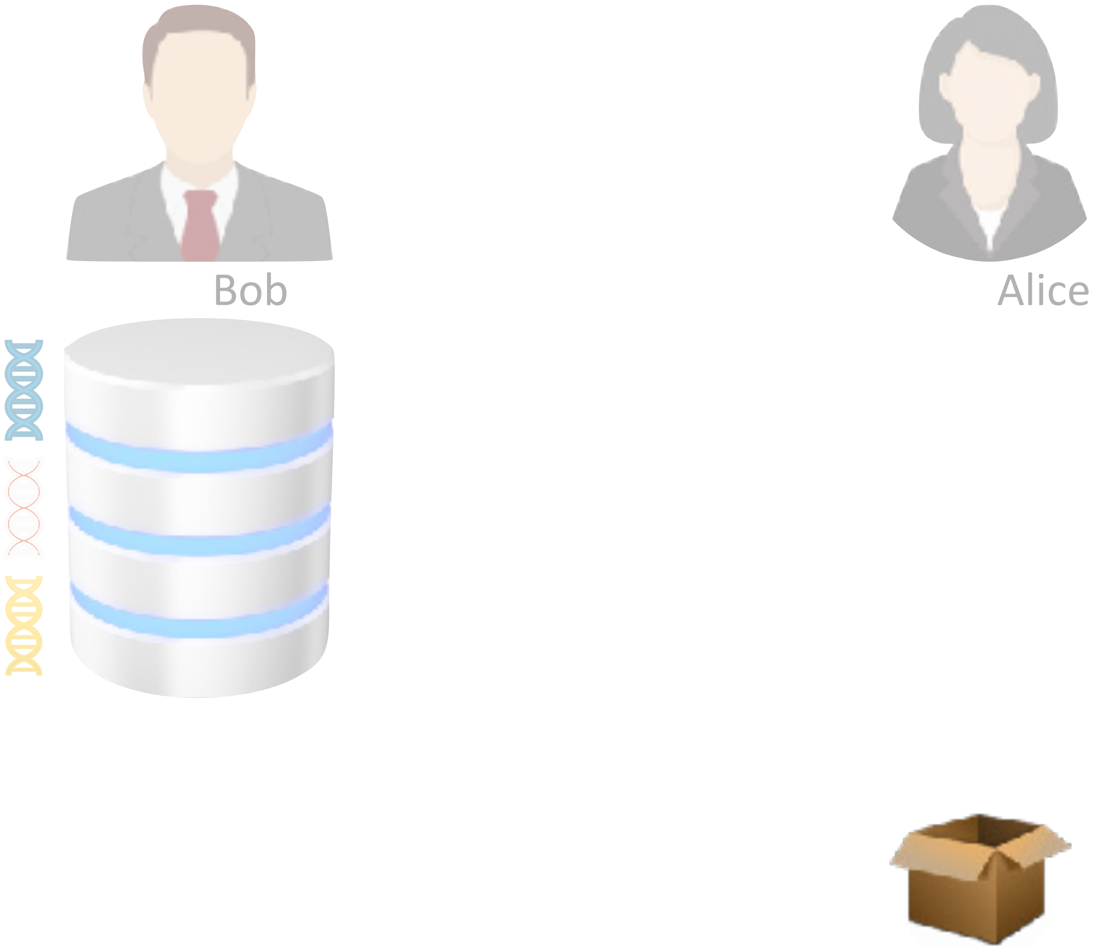
Alice



 $=H($  $)$

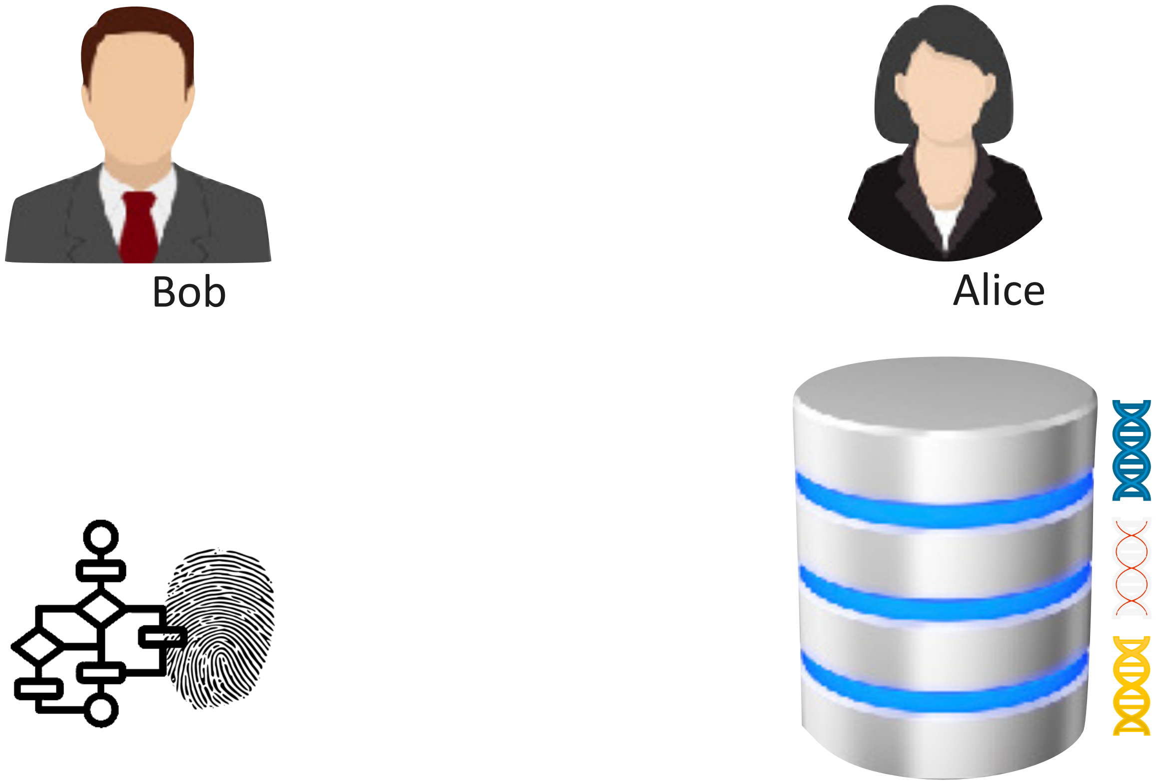
Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices



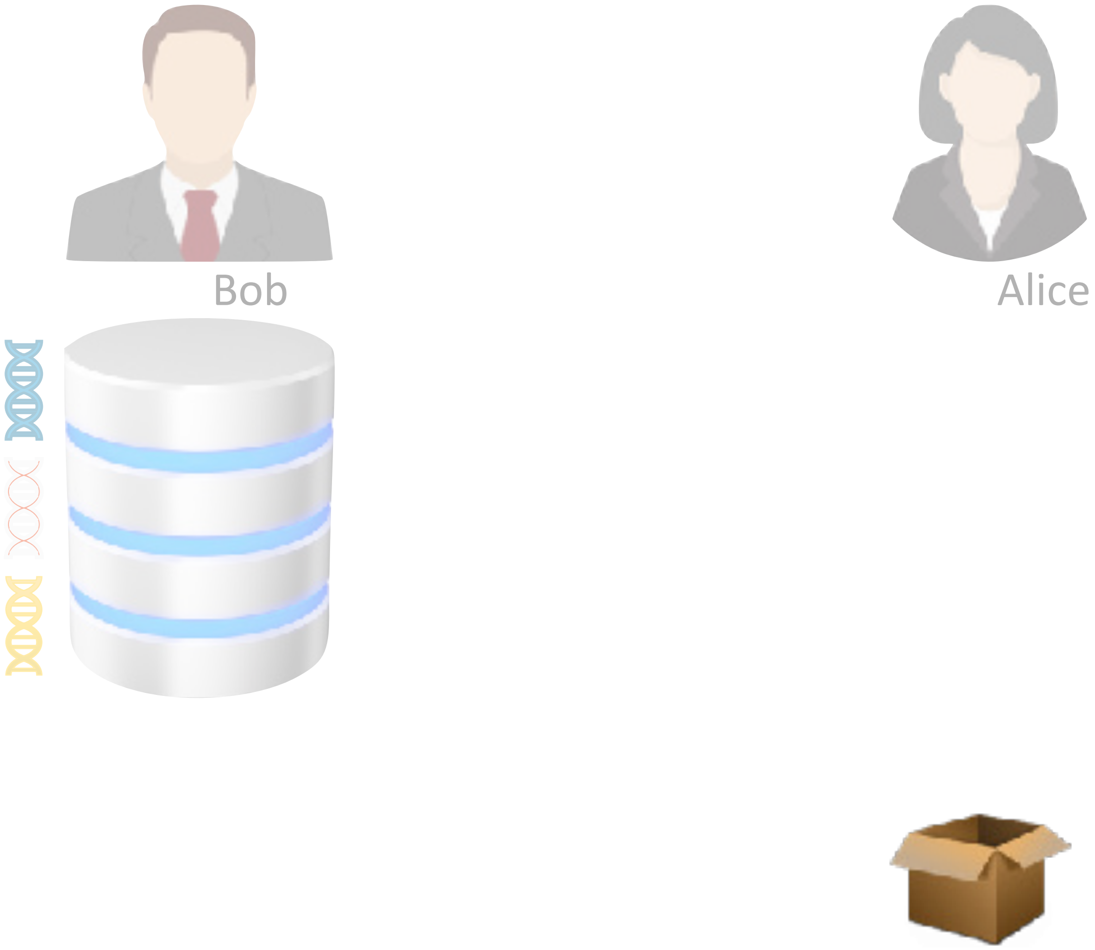
Laconic Cryptography

[CDGGMP CRYPTO'17]
Laconic Oblivious Transfer and Its Applications



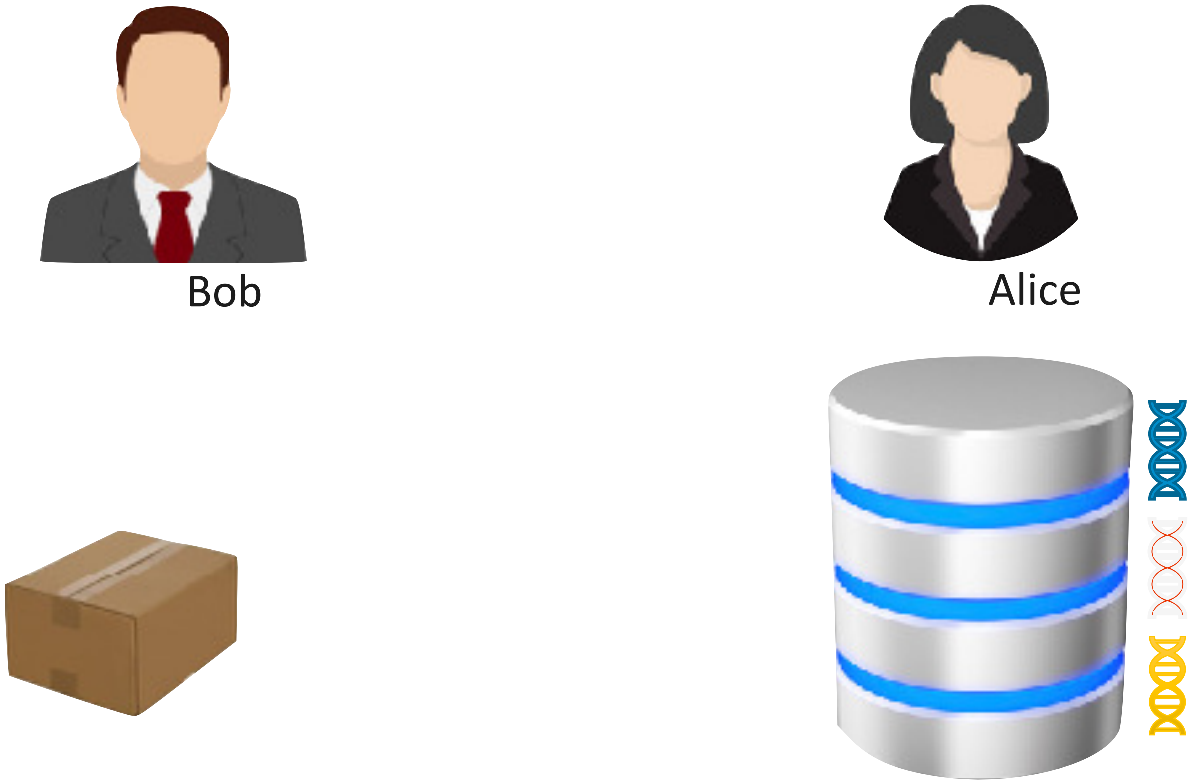
Homomorphic Encryption

[Gentry STOC’09]
Fully homomorphic encryption using ideal lattices



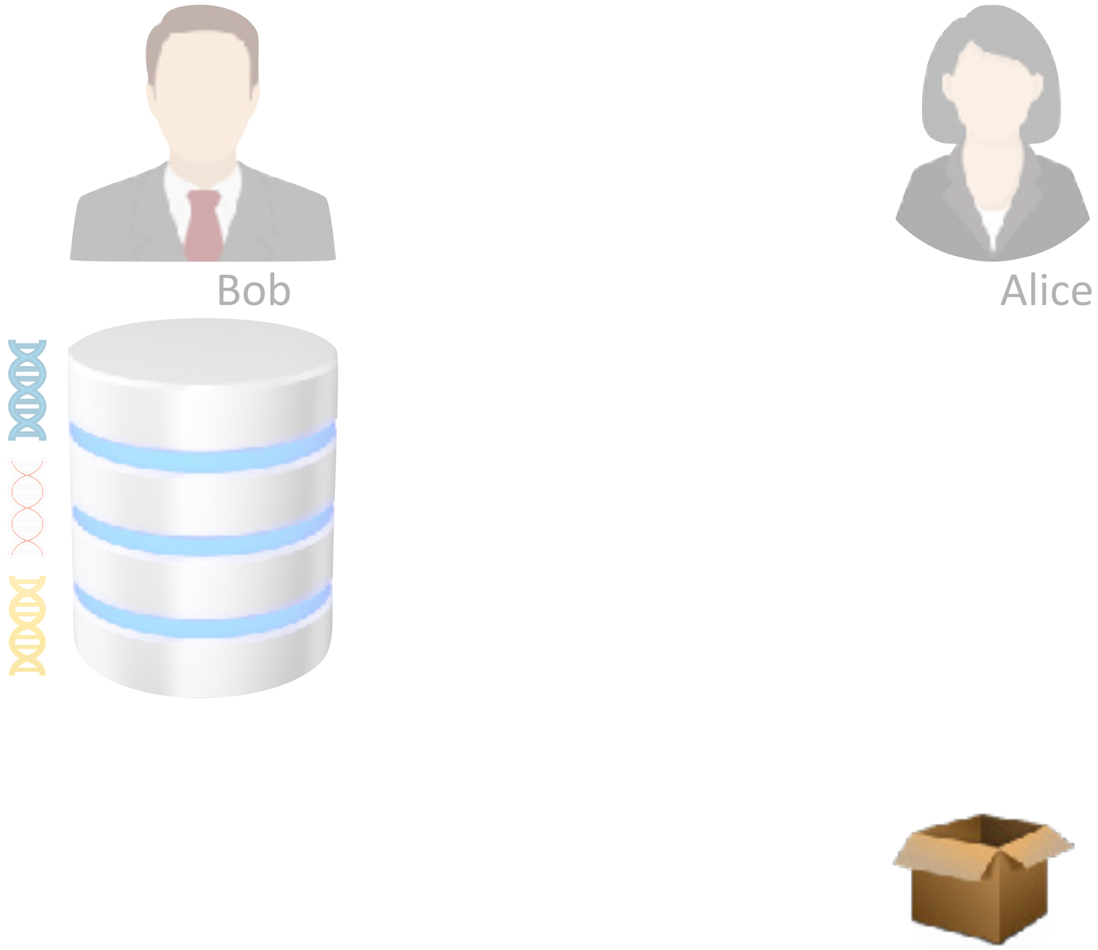
Laconic Cryptography

[CDGGMP CRYPTO’17]
Laconic Oblivious Transfer and Its Applications



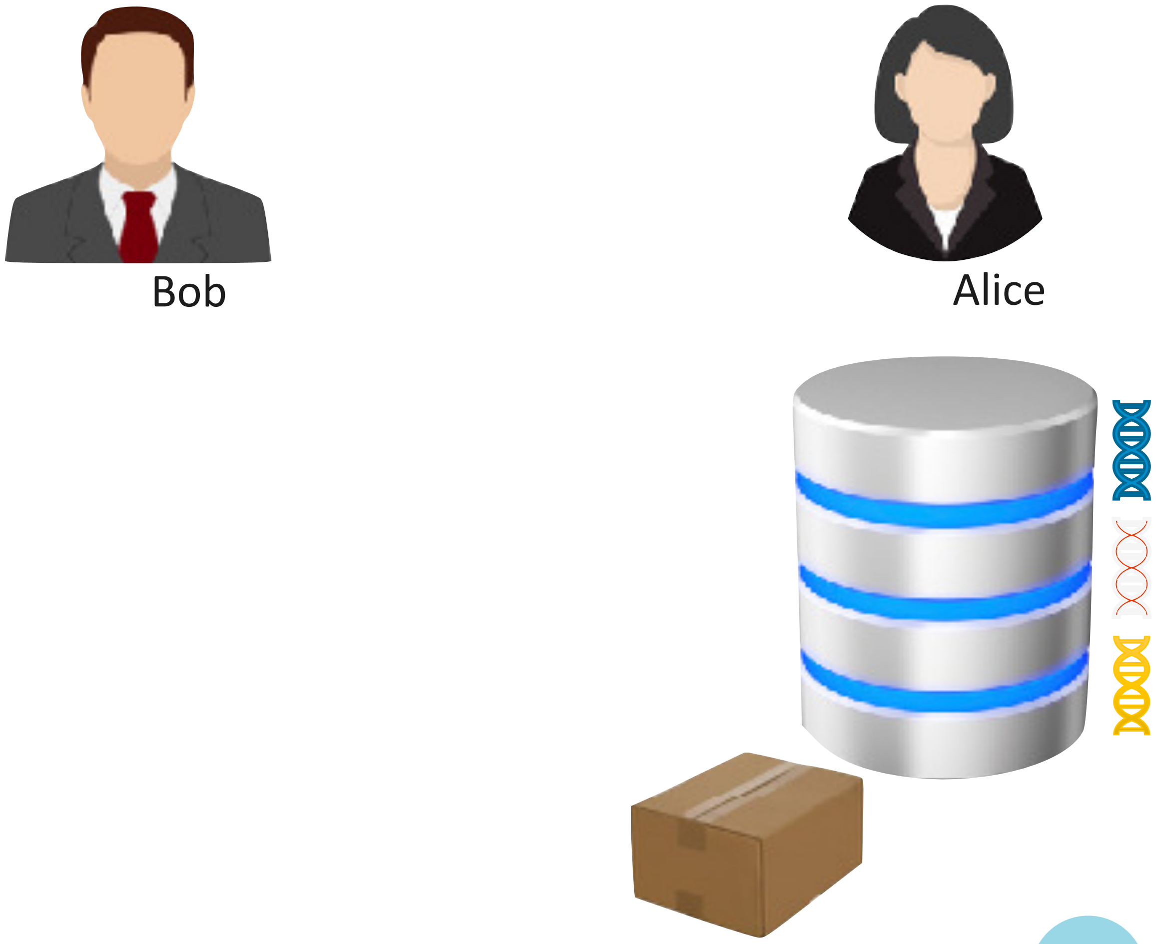
Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices



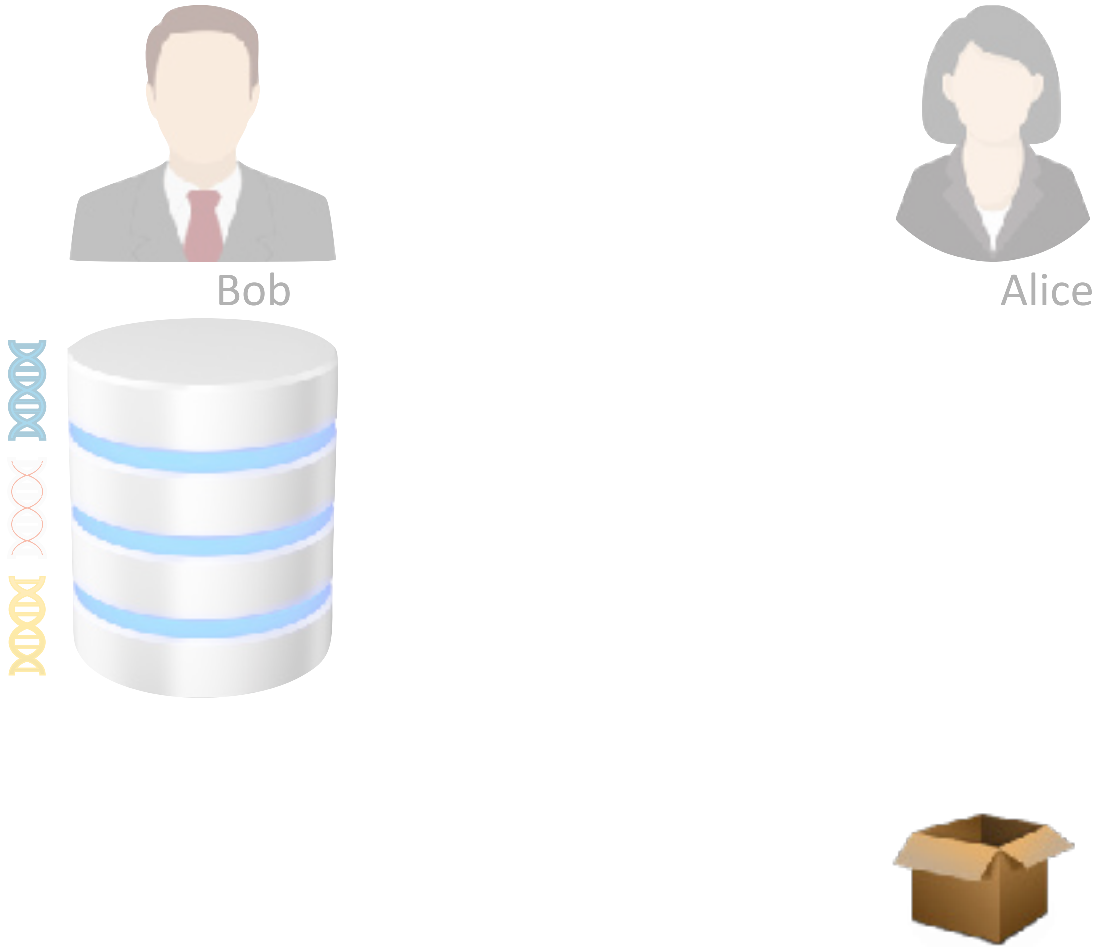
Laconic Cryptography

[CDGGMP CRYPTO'17]
Laconic Oblivious Transfer and Its Applications



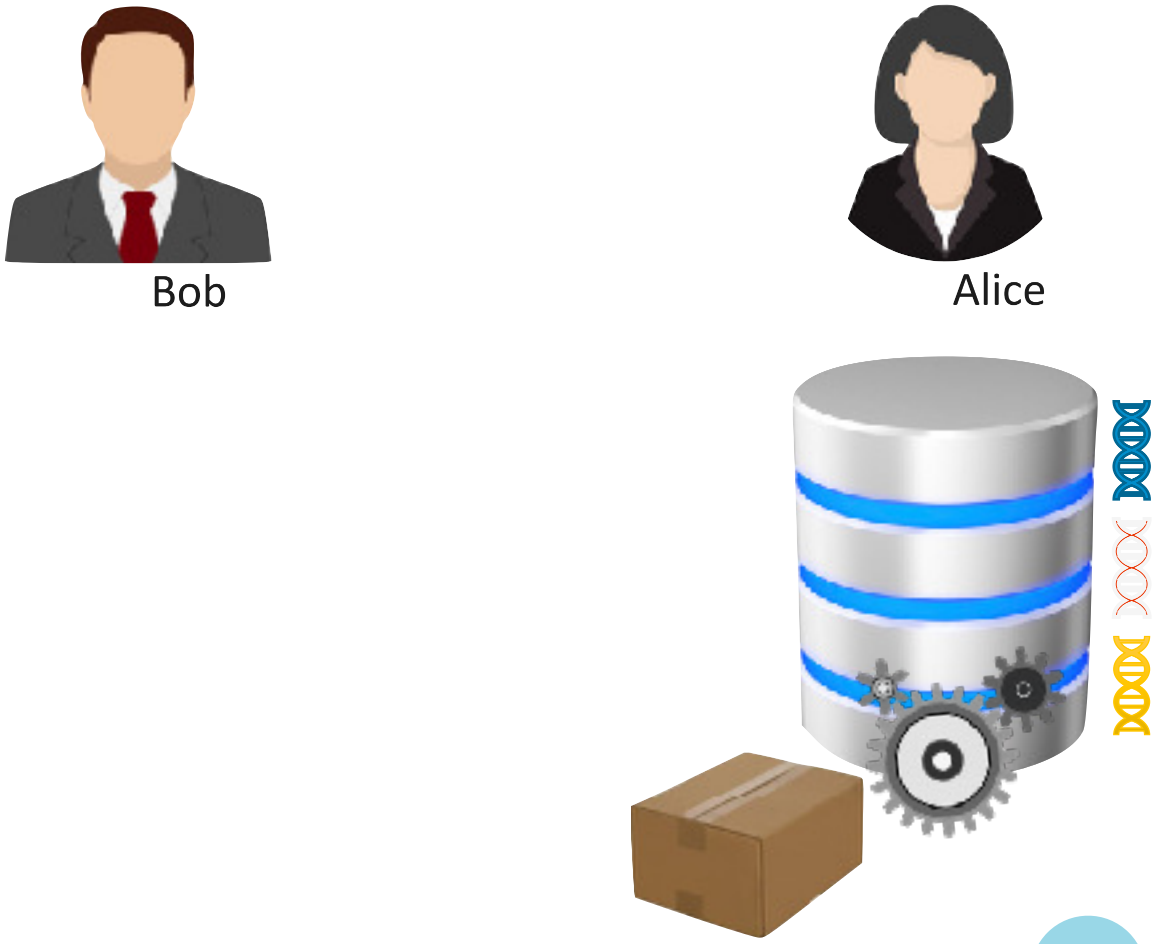
Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices



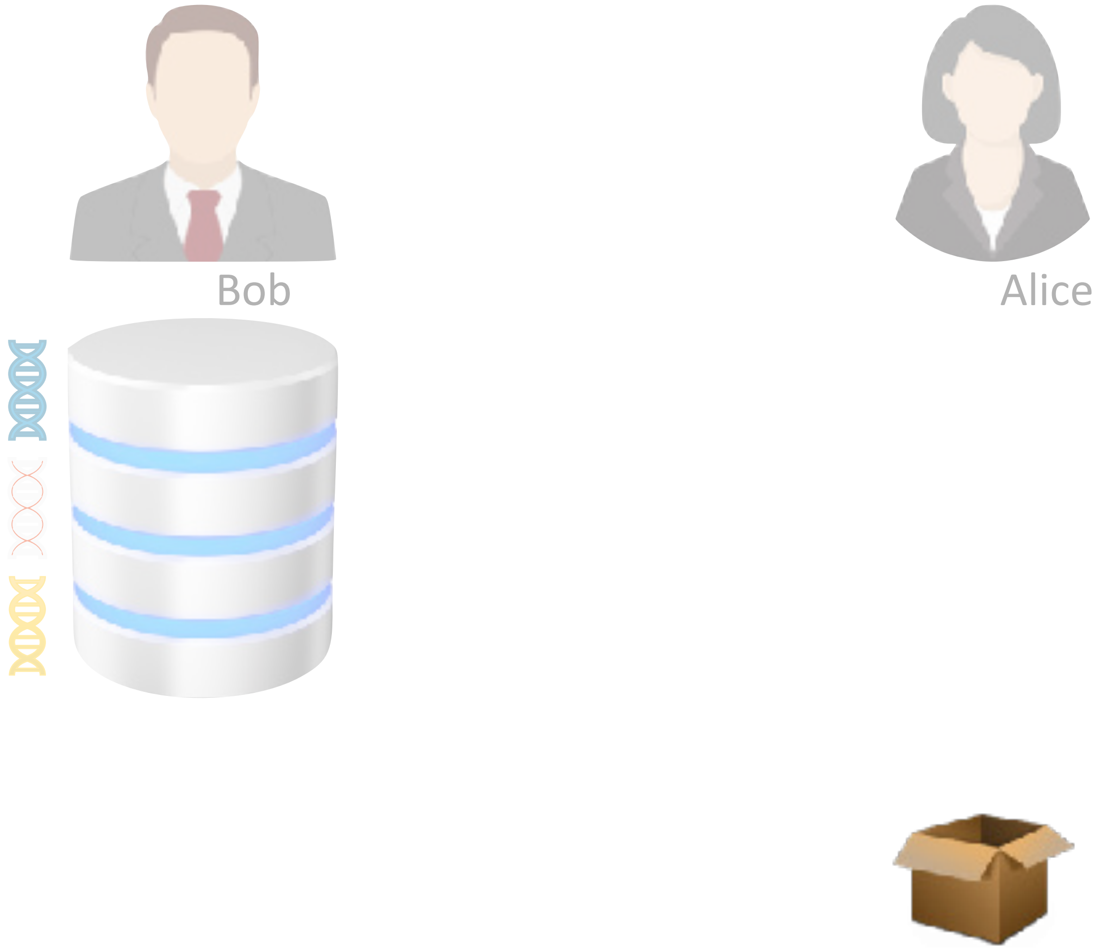
Laconic Cryptography

[CDGGMP CRYPTO'17]
Laconic Oblivious Transfer and Its Applications



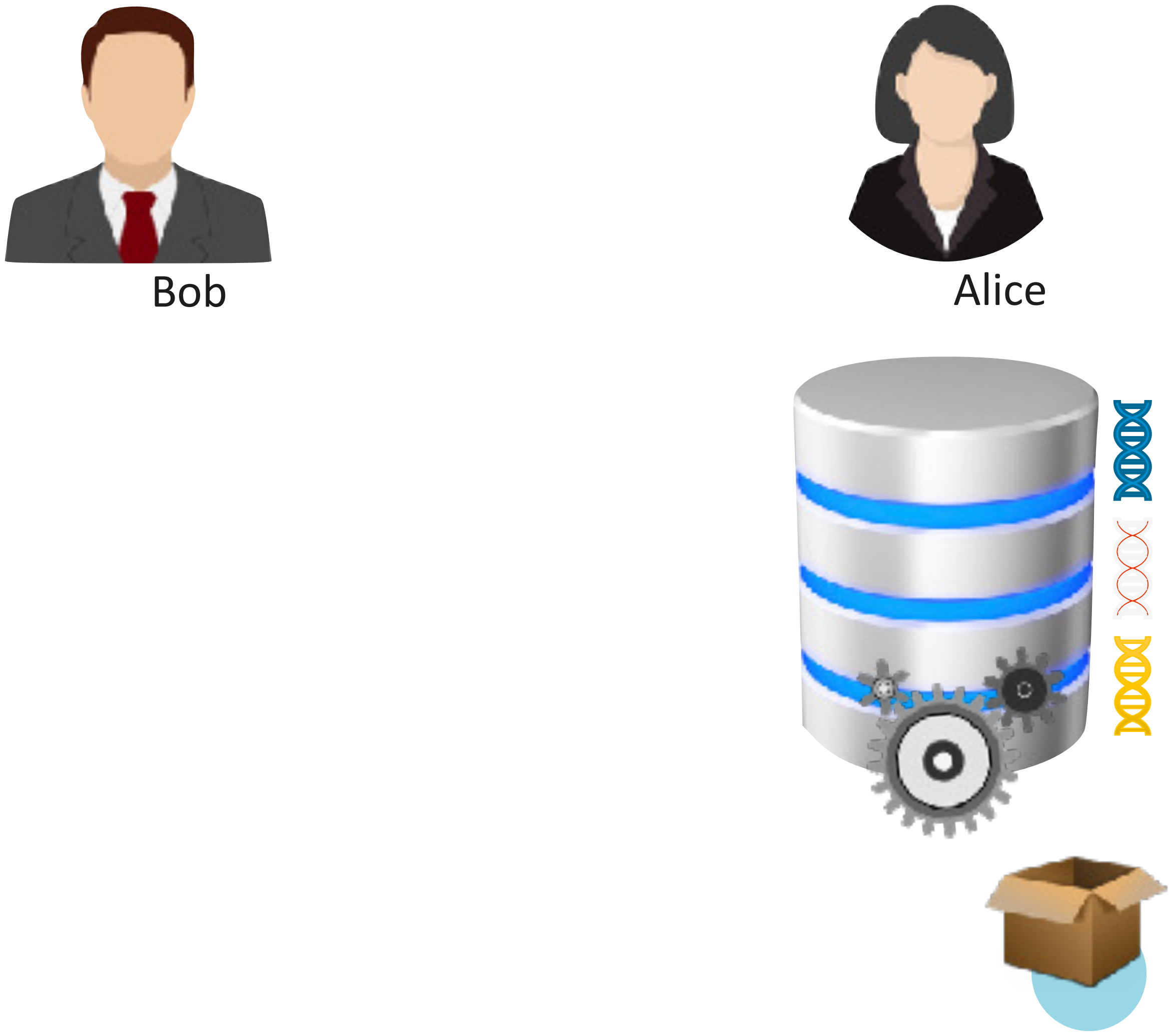
Homomorphic Encryption

[Gentry STOC'09]
Fully homomorphic encryption using ideal lattices



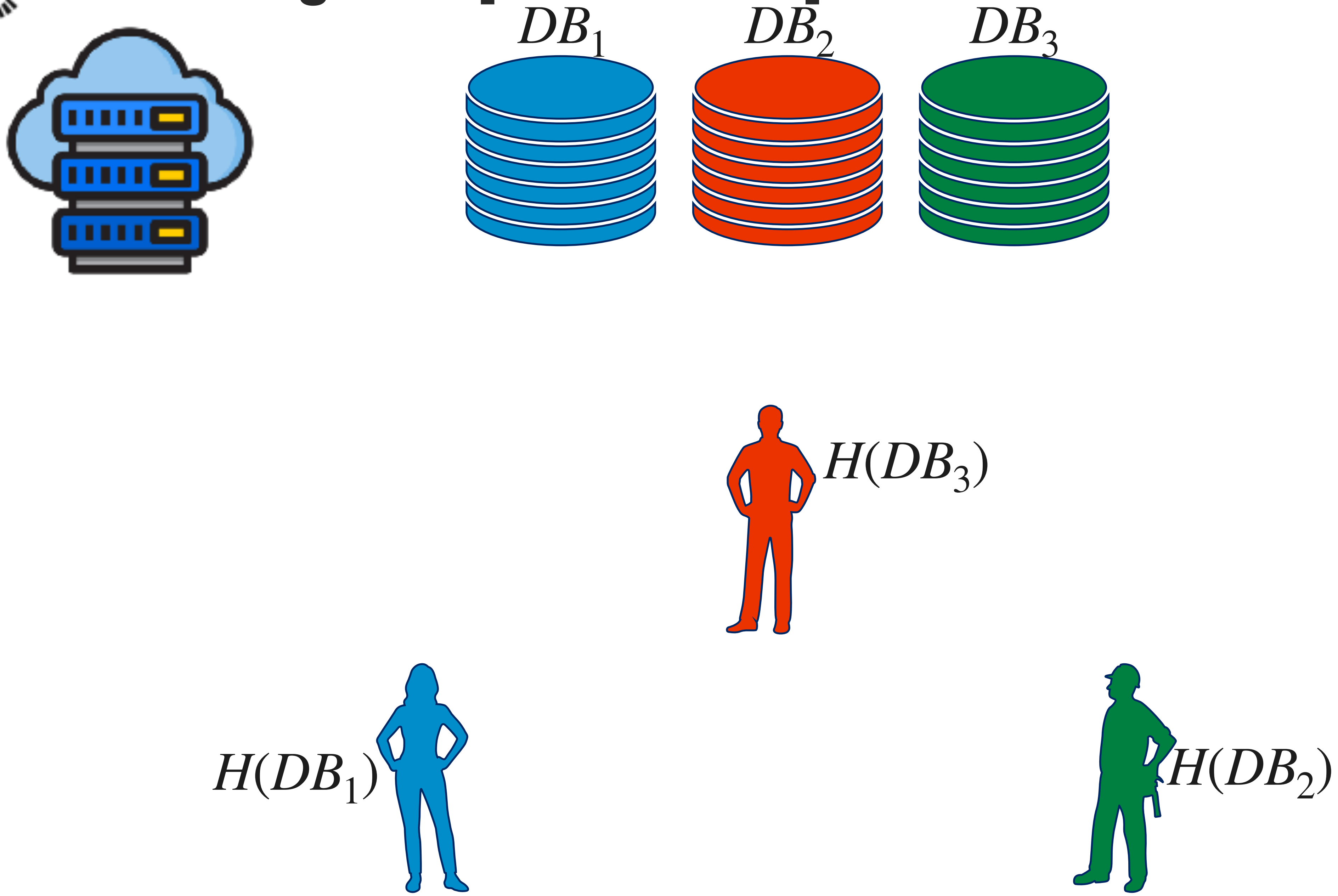
Laconic Cryptography

[CDGGMP CRYPTO'17]
Laconic Oblivious Transfer and Its Applications

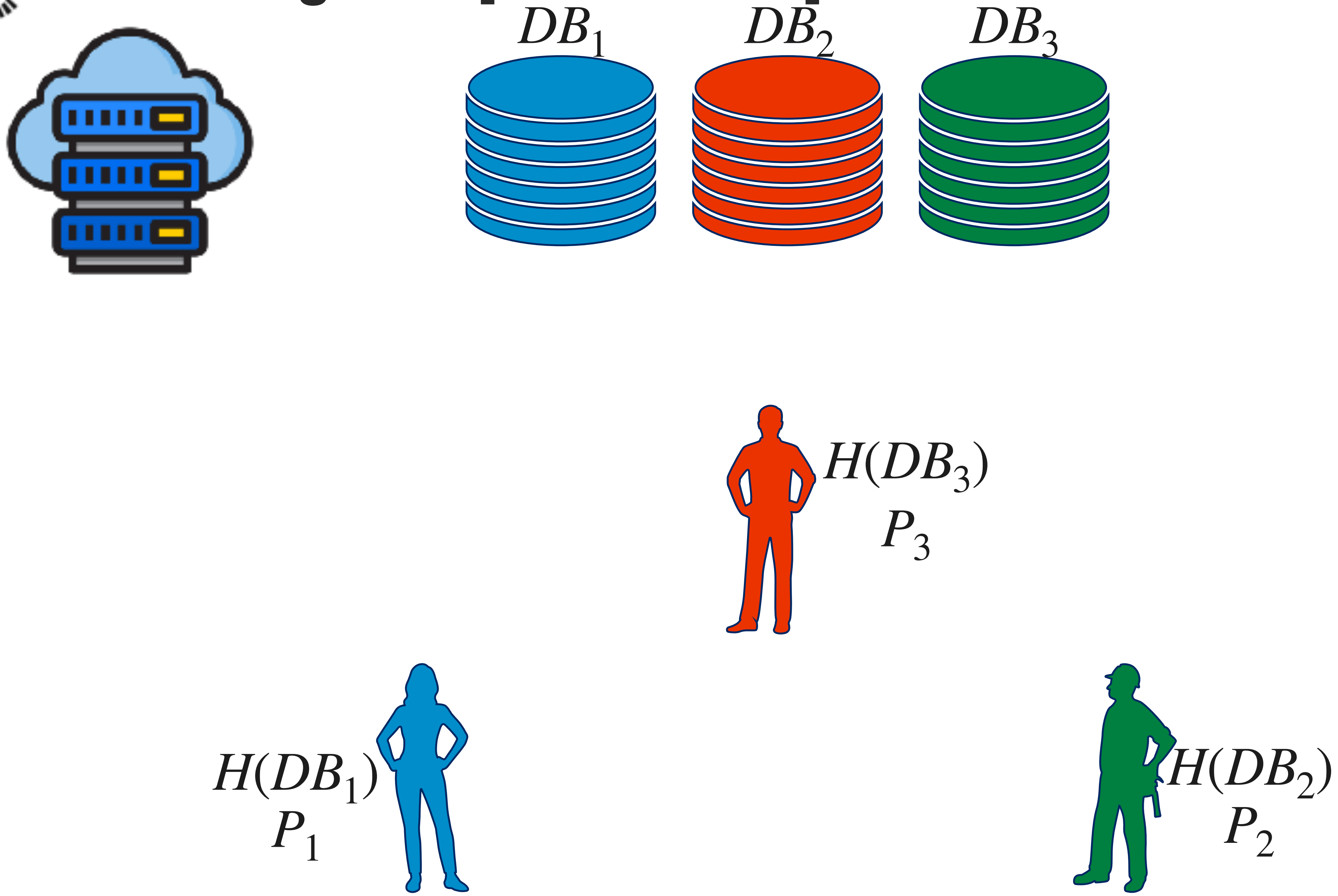


Paradigms

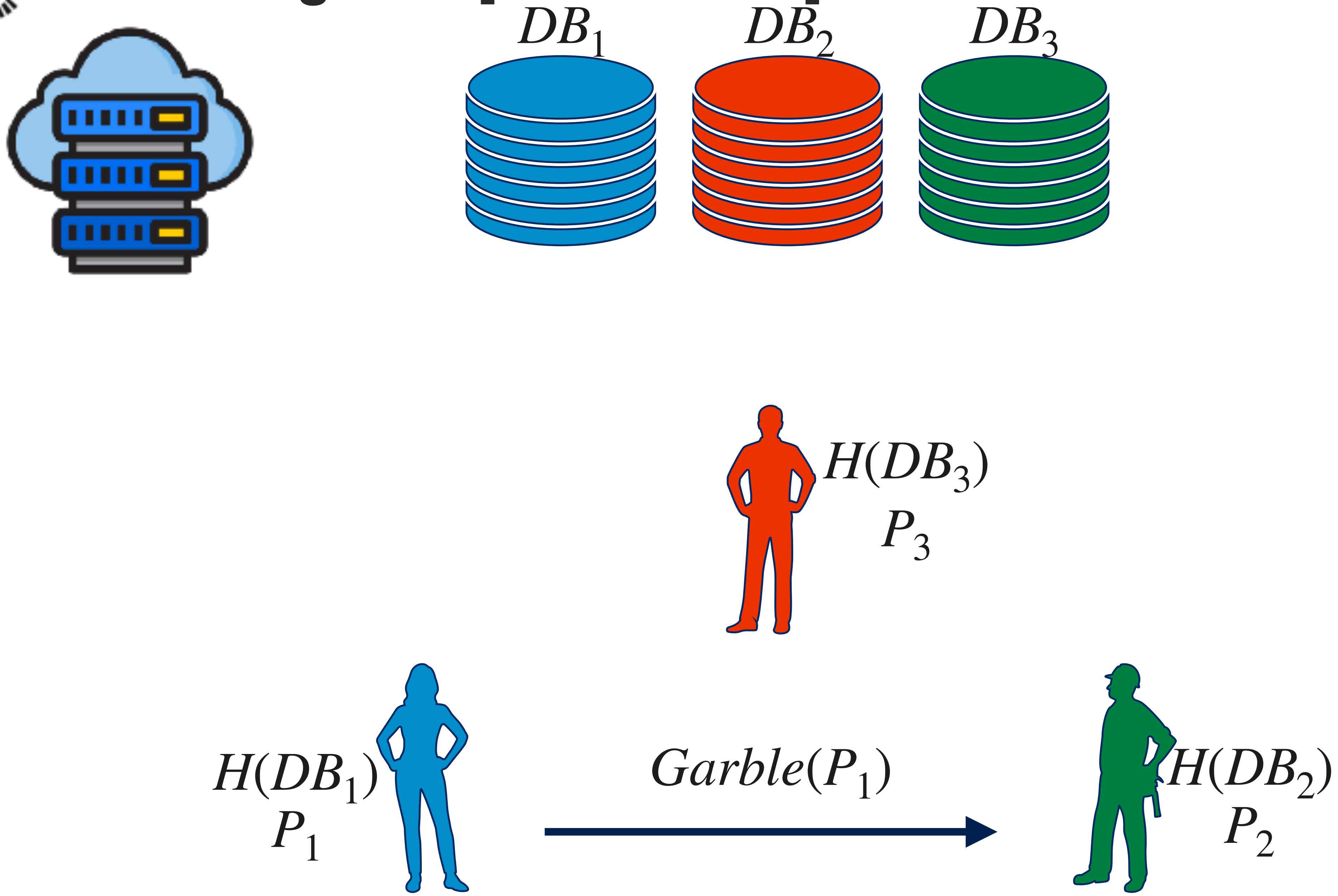
RAM Delegation [CDGGMP17]



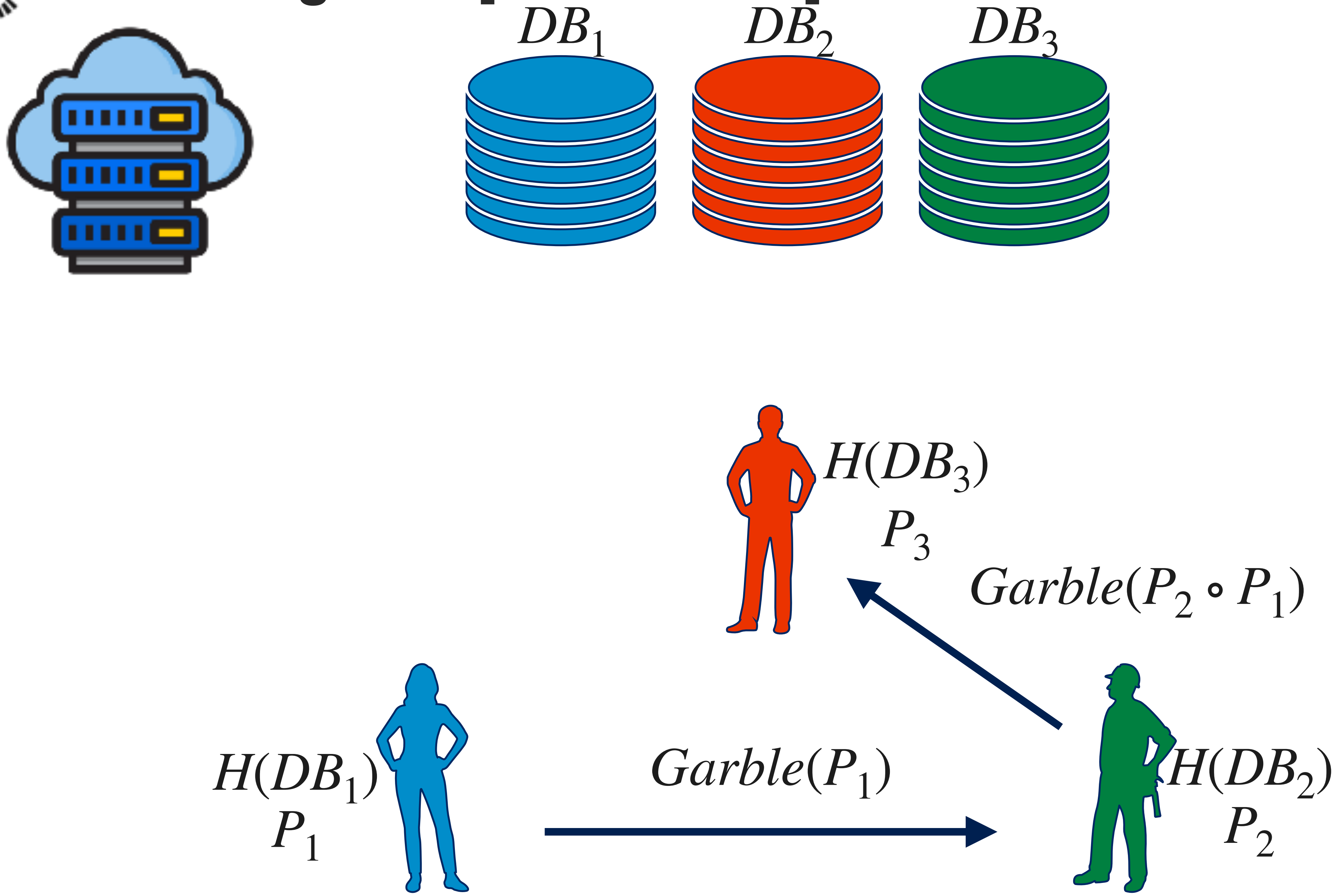
RAM Delegation [CDGGMP17]



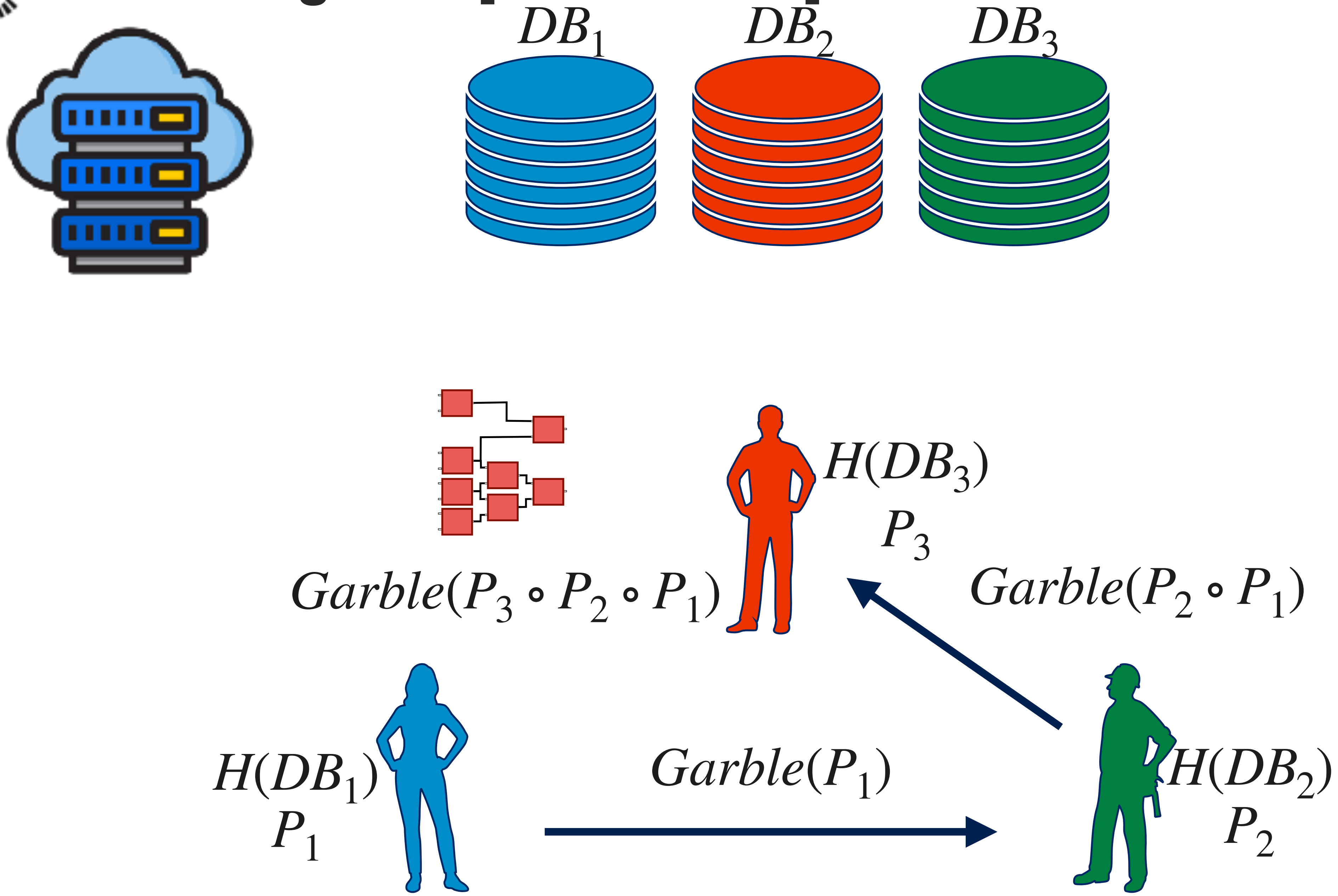
RAM Delegation [CDGGMP17]



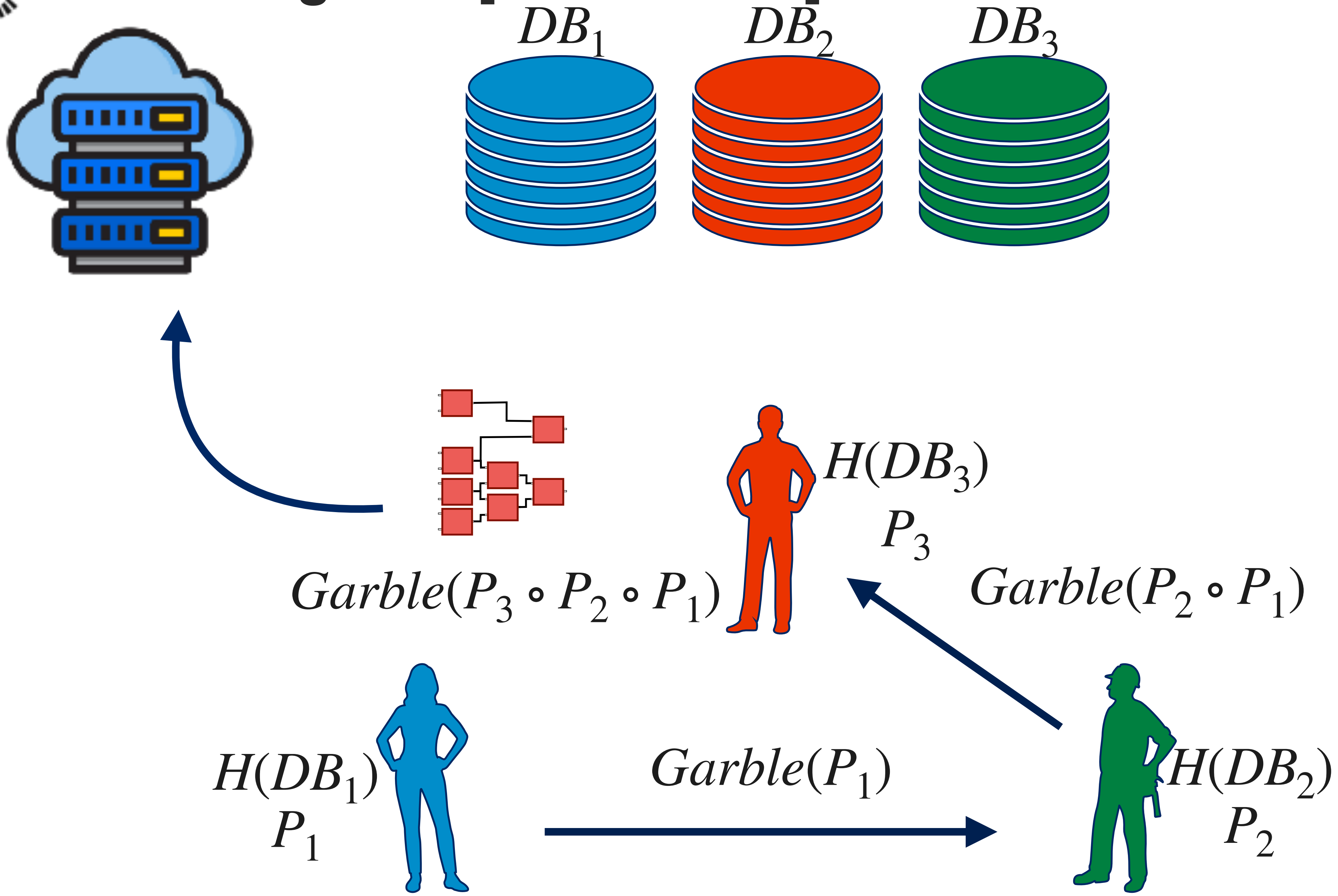
RAM Delegation [CDGGMP17]



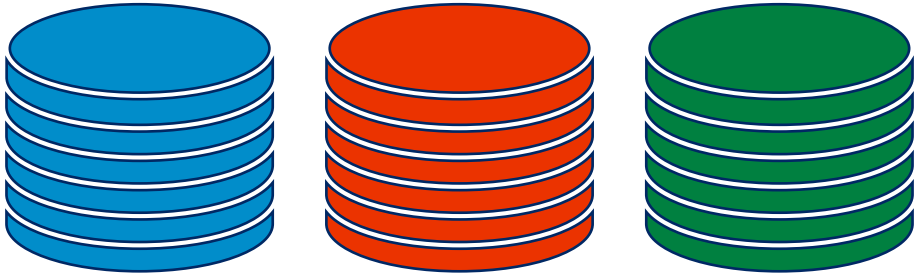
RAM Delegation [CDGGMP17]

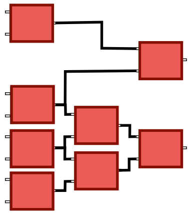


RAM Delegation [CDGGMP17]

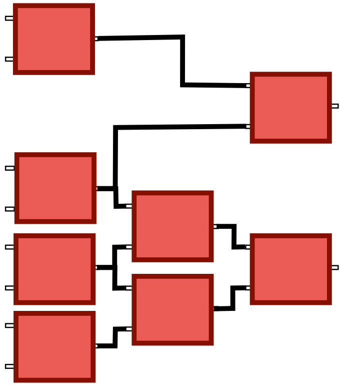


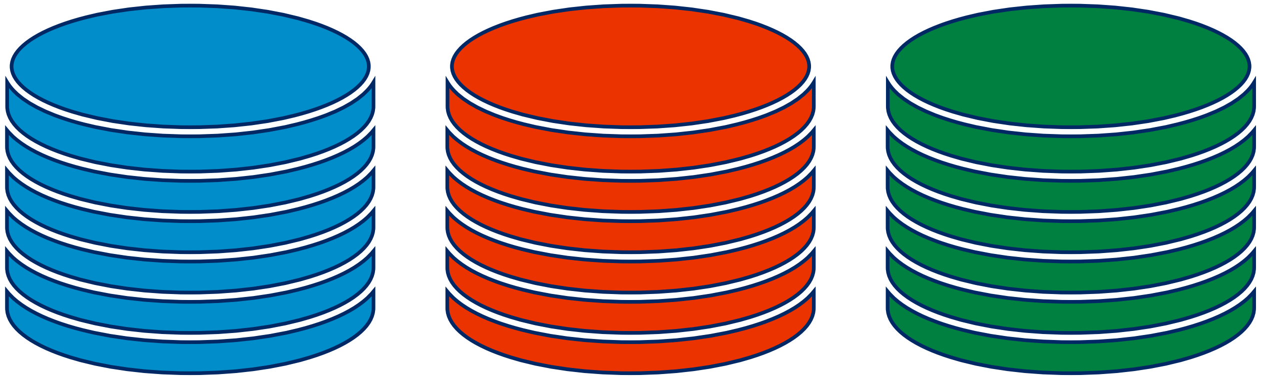
RAM Delegation [CDGGMP17]



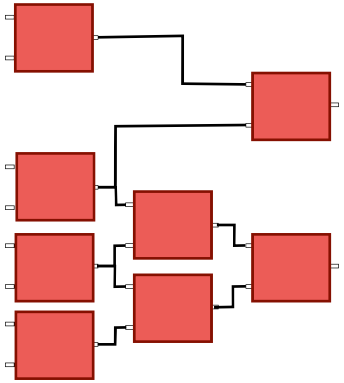


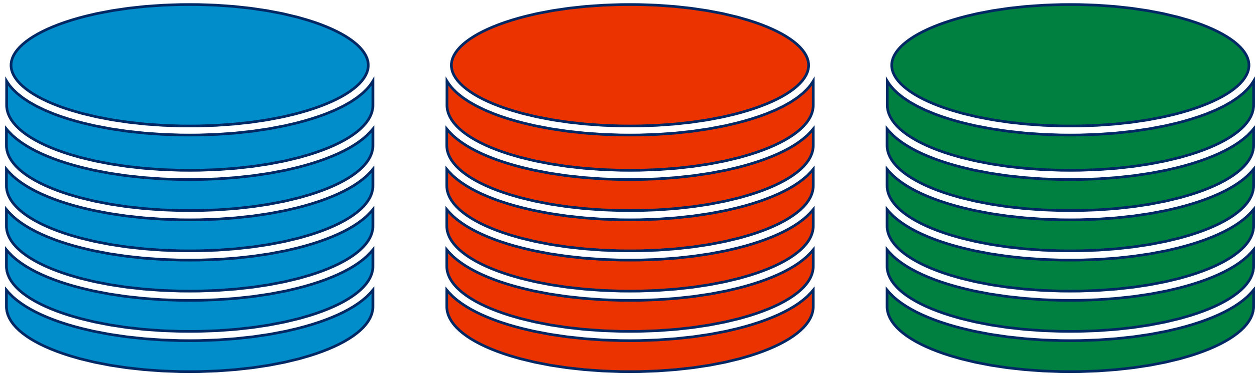
RAM Delegation [CDGGMP17]



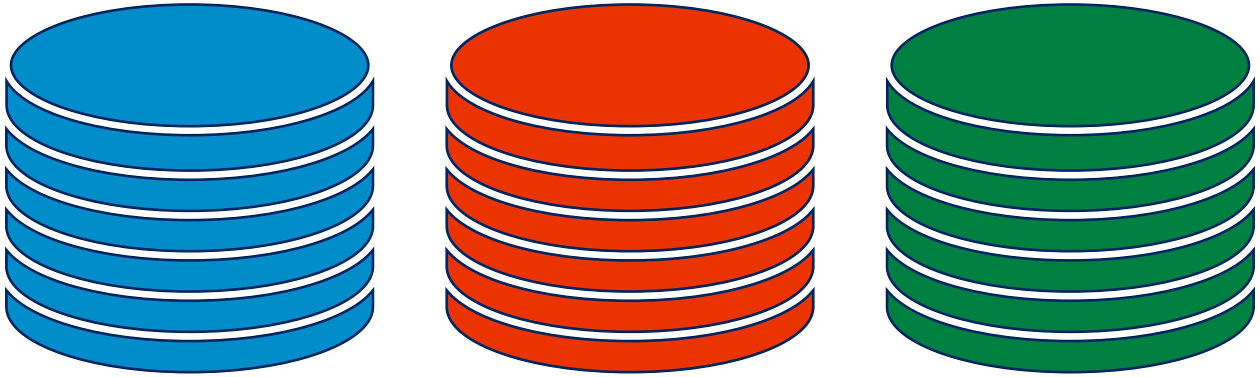


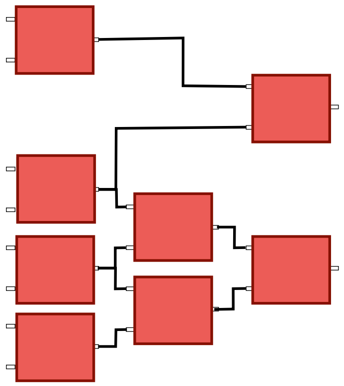
RAM Delegation [CDGGMP17]



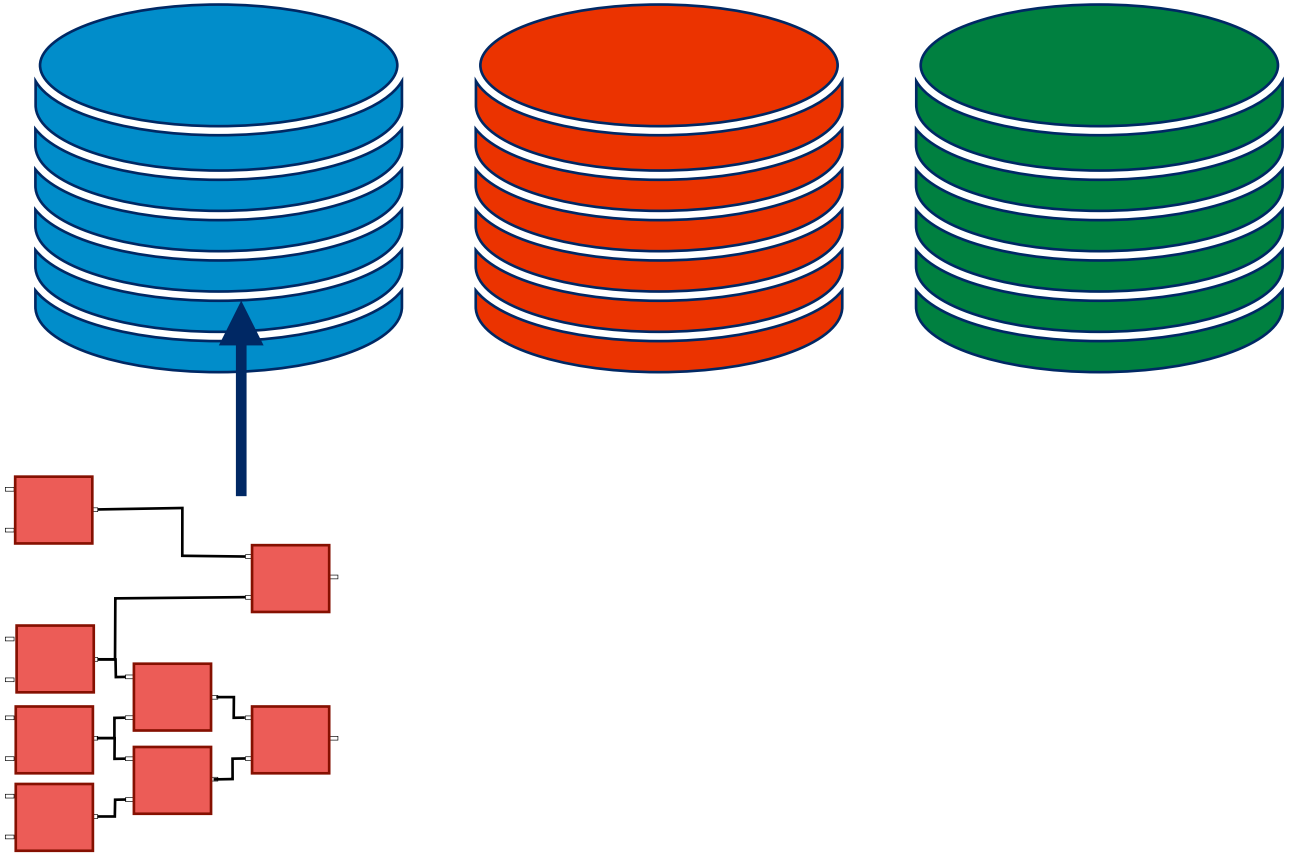


RAM Delegation [CDGGMP17]

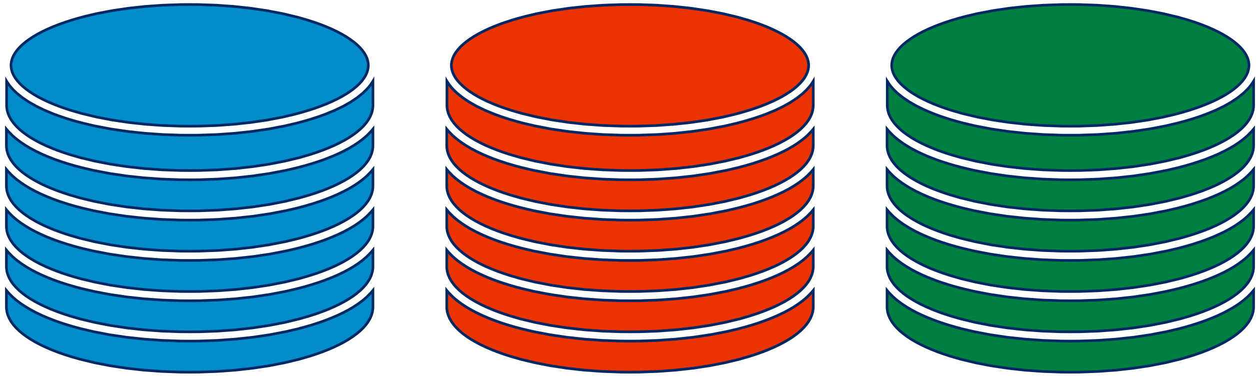


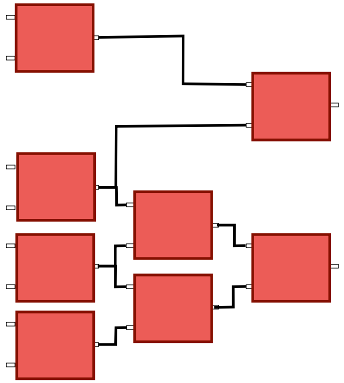


RAM Delegation [CDGGMP17]

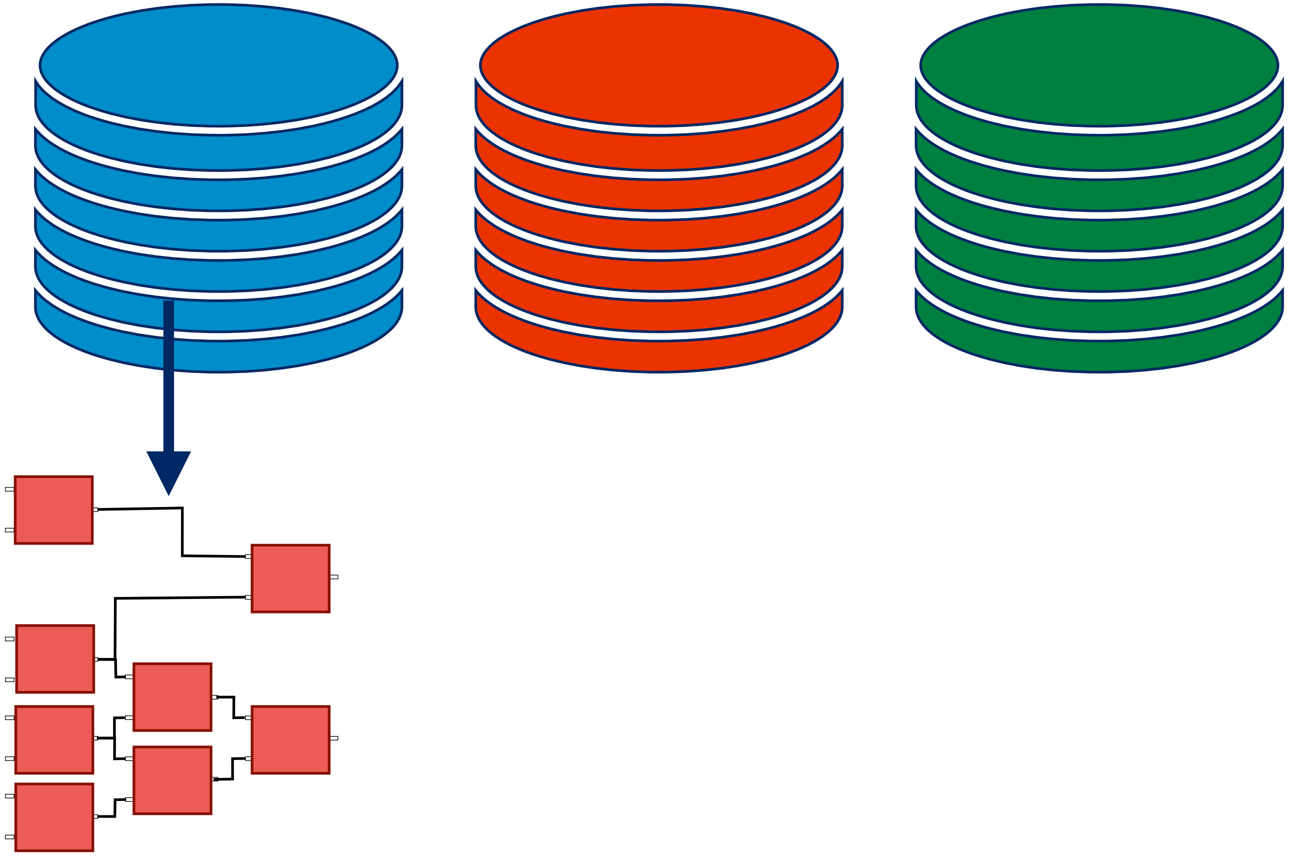


RAM Delegation [CDGGMP17]

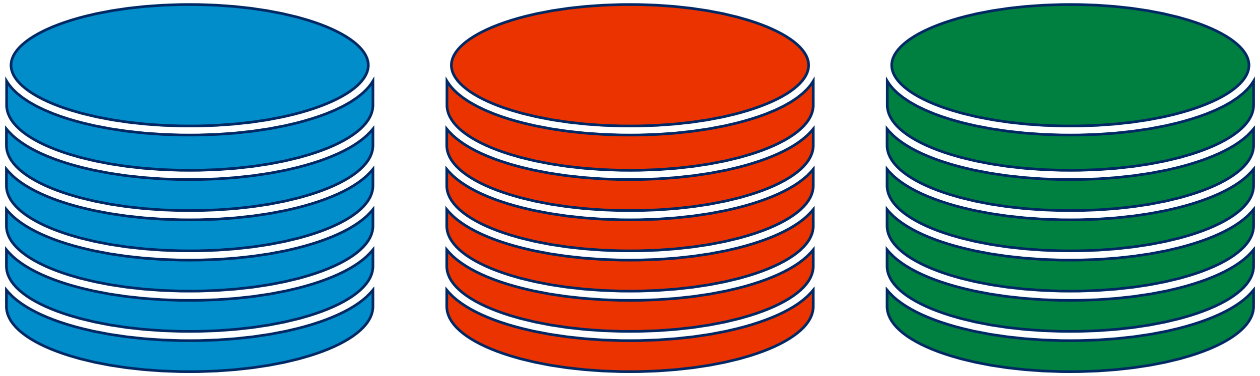


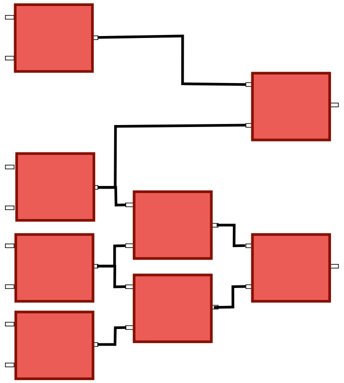


RAM Delegation [CDGGMP17]

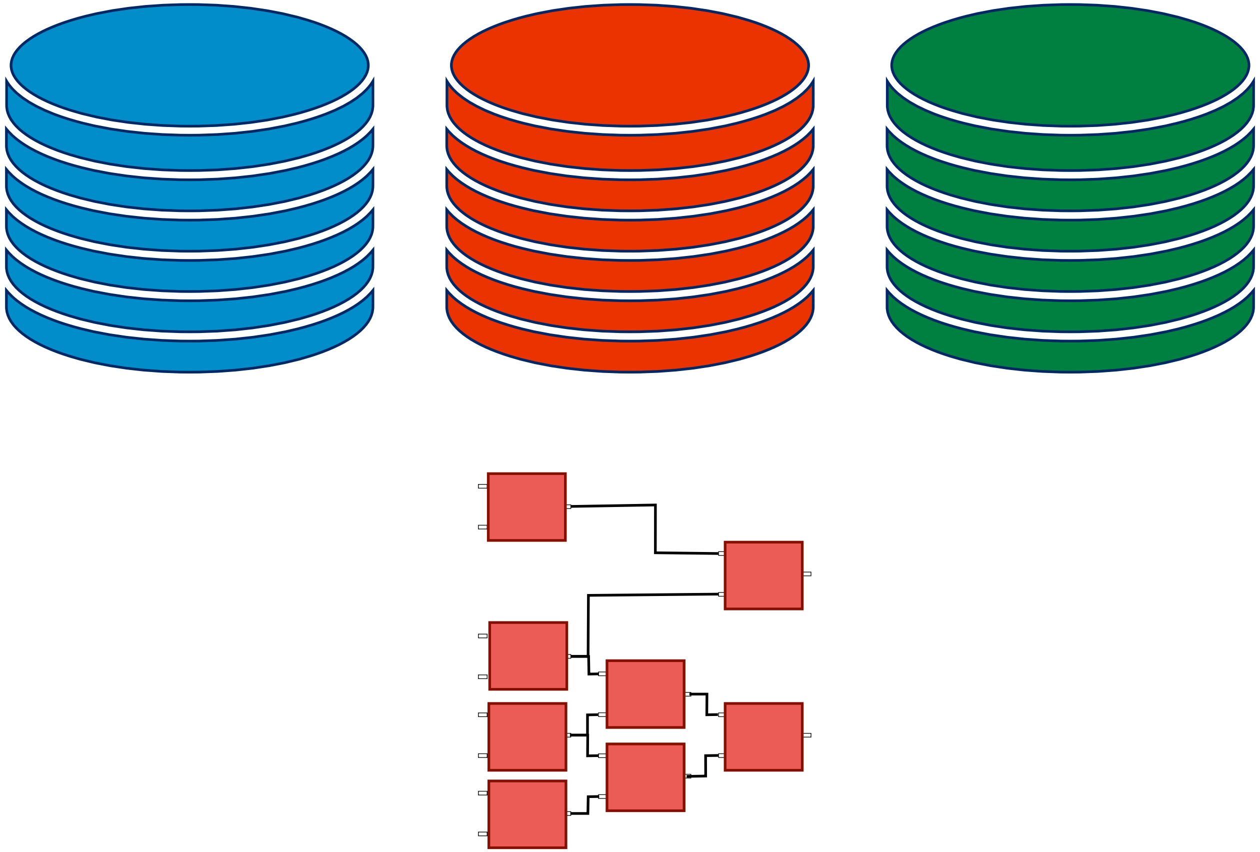


RAM Delegation [CDGGMP17]

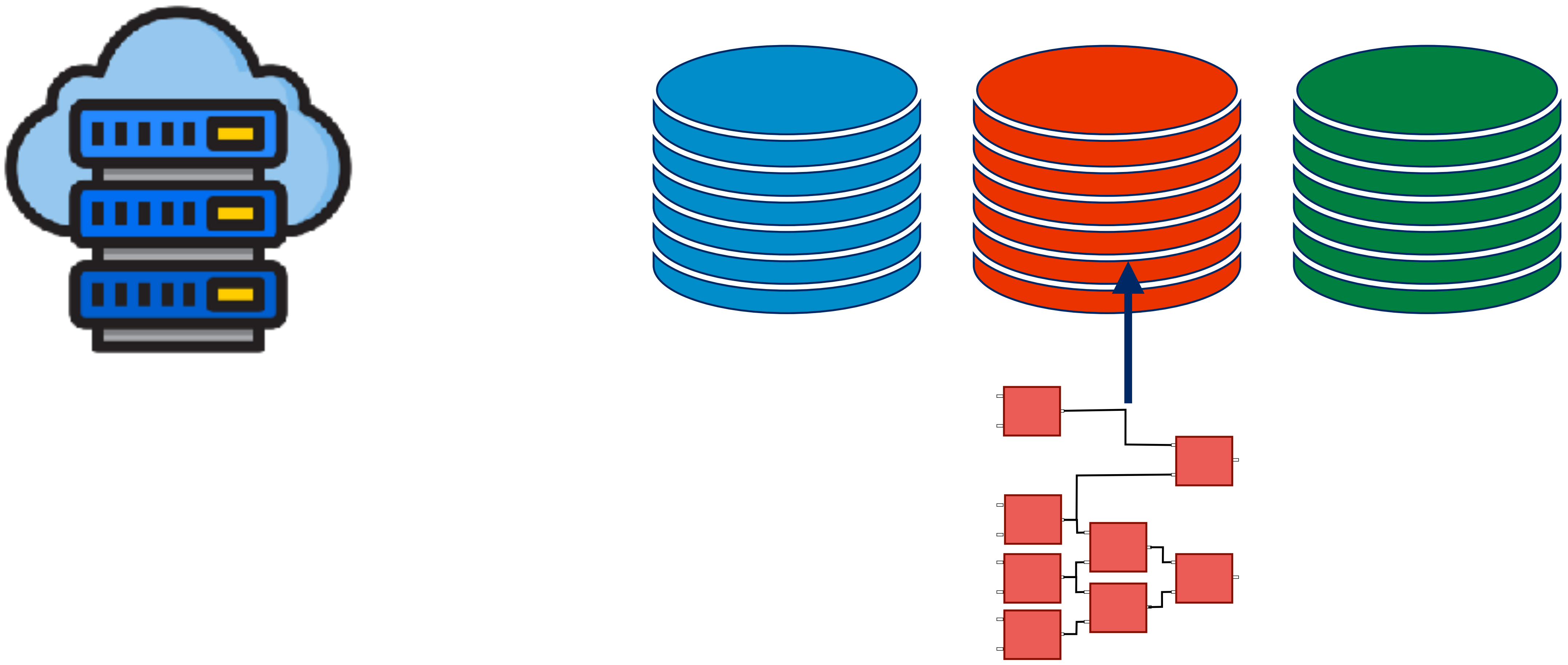




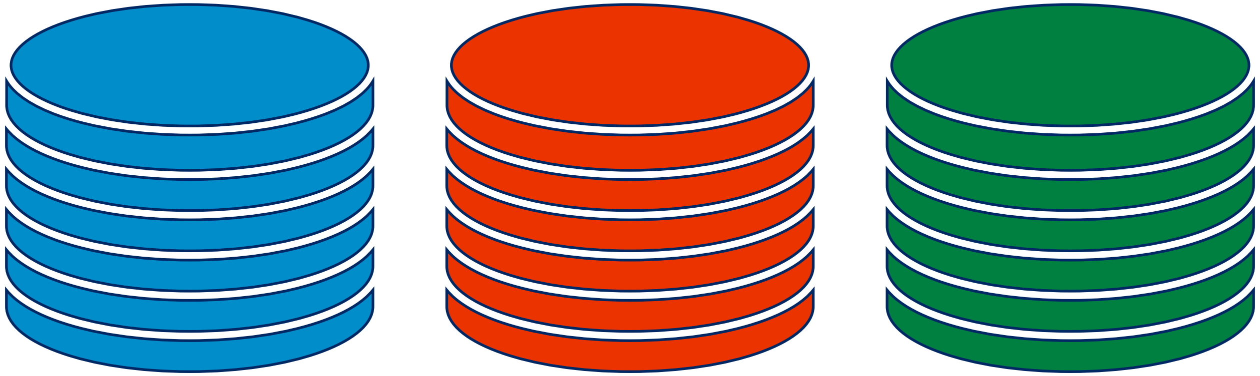
RAM Delegation [CDGGMP17]

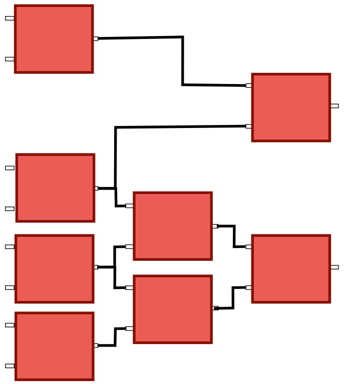


RAM Delegation [CDGGMP17]

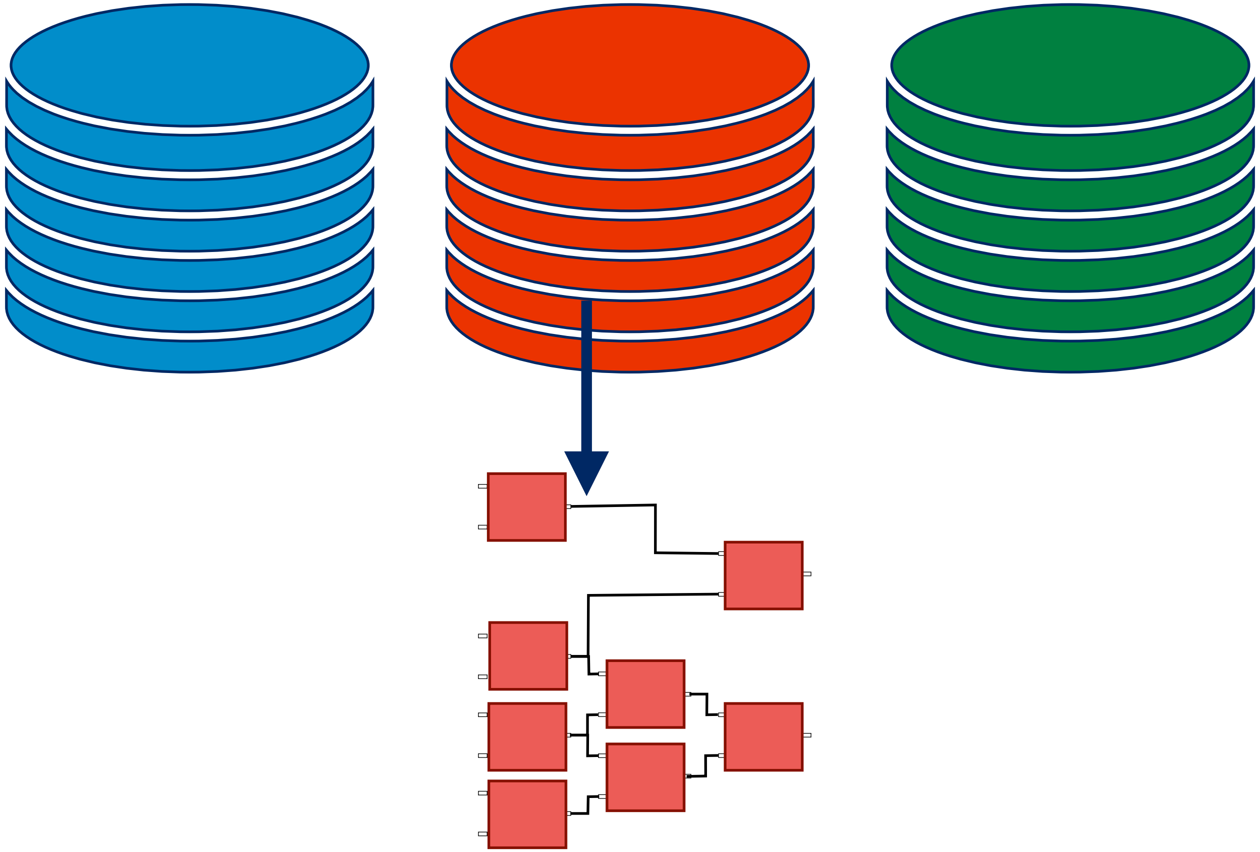


RAM Delegation [CDGGMP17]

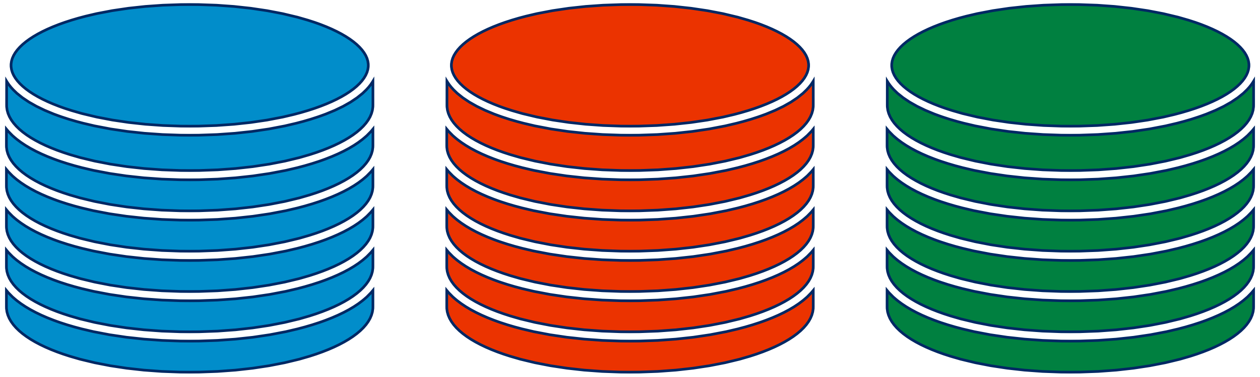


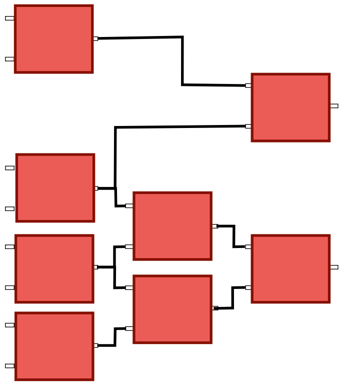


RAM Delegation [CDGGMP17]

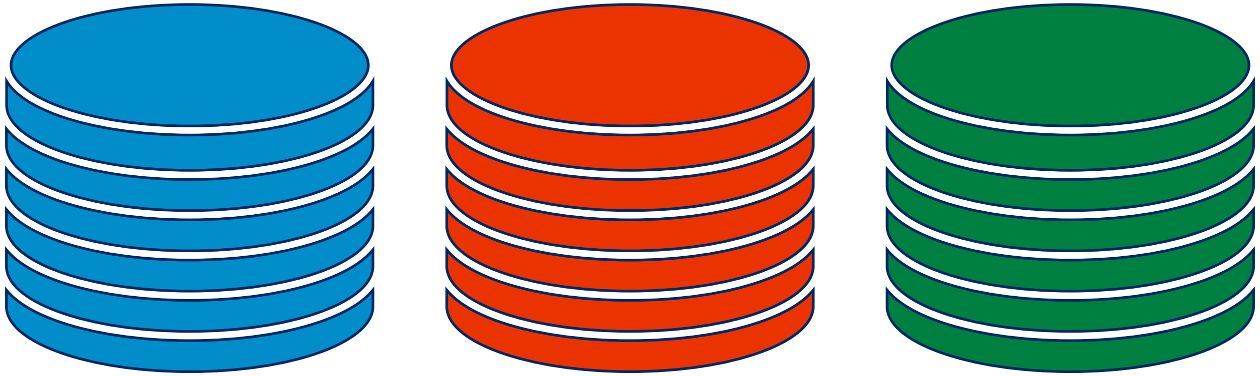


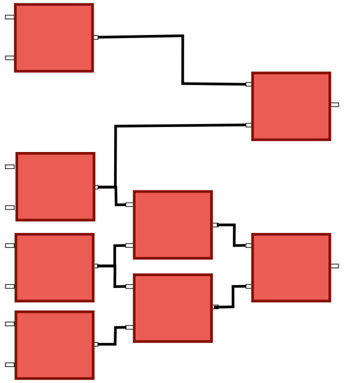
RAM Delegation [CDGGMP17]





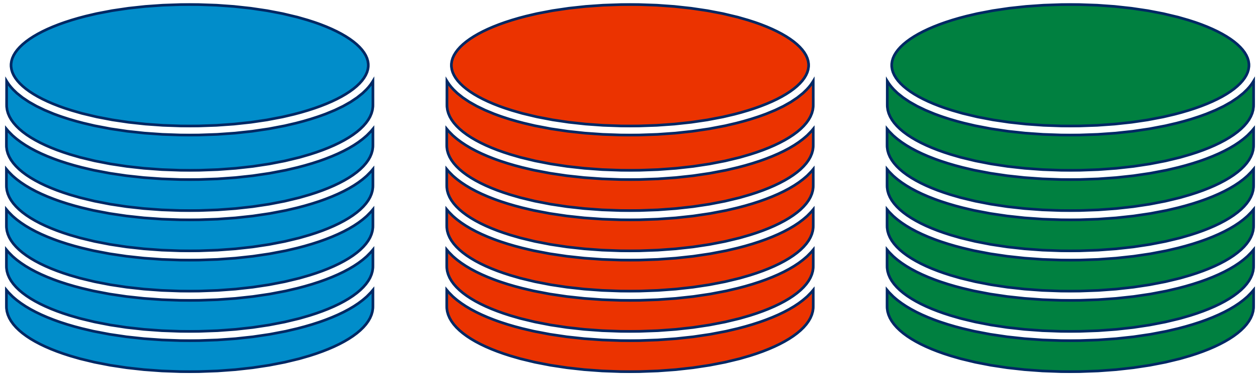
RAM Delegation [CDGGMP17]

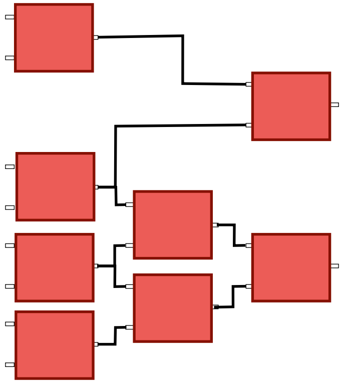




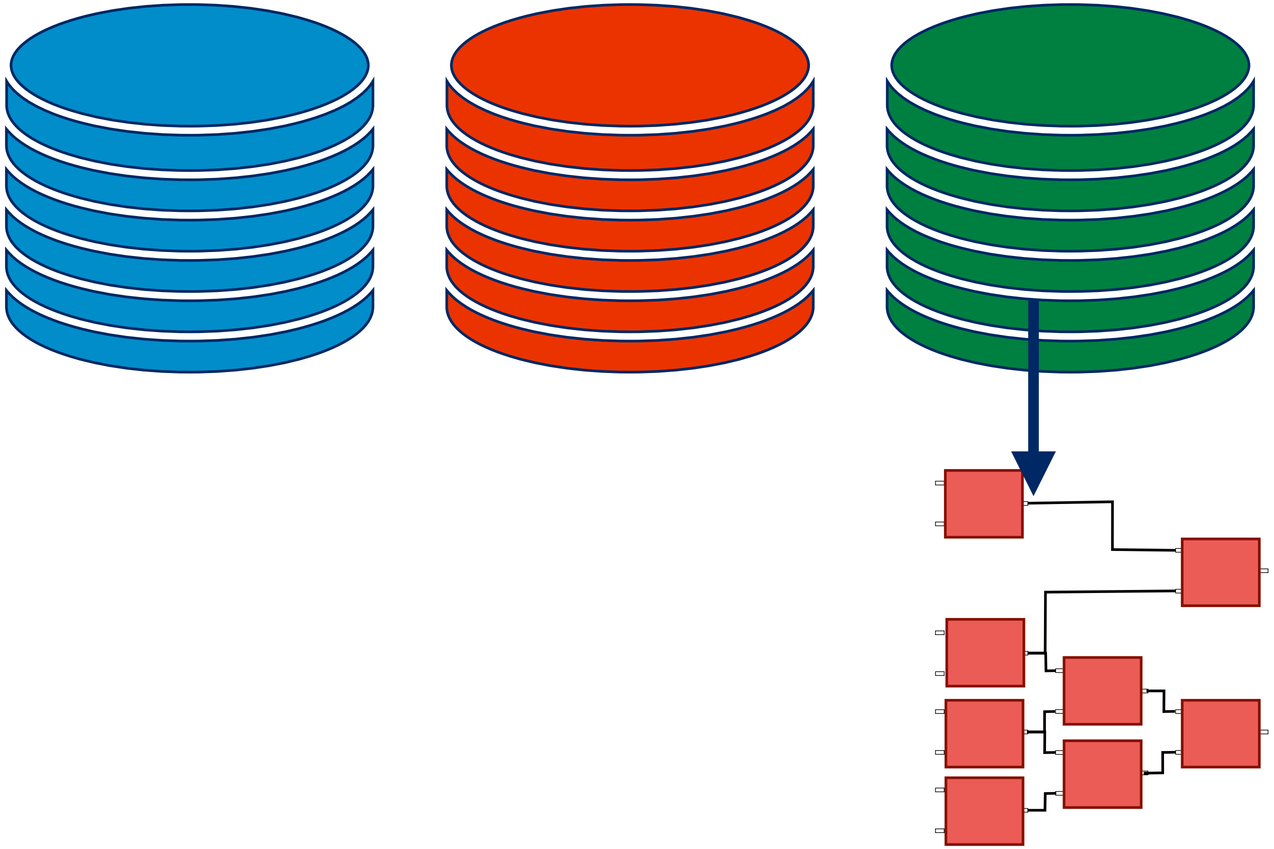
RAM Delegation [CDGGMP17]

RAM Delegation [CDGGMP17]

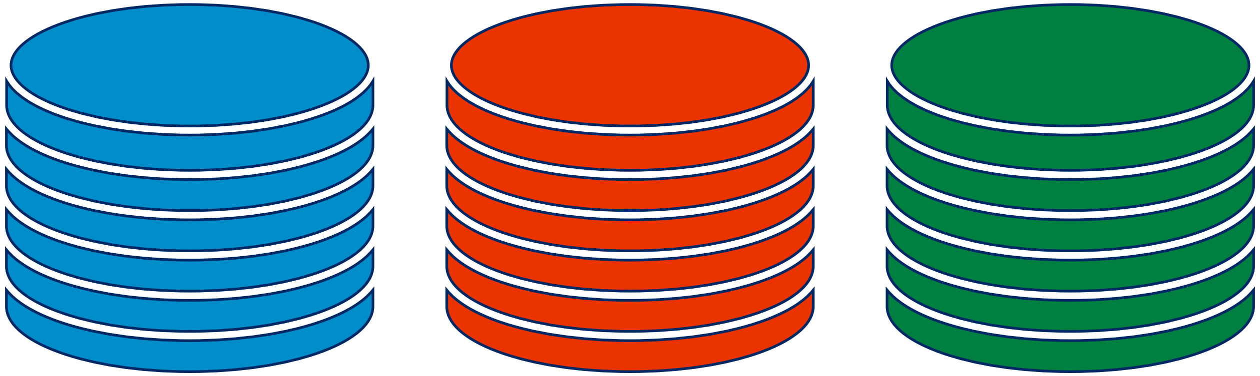


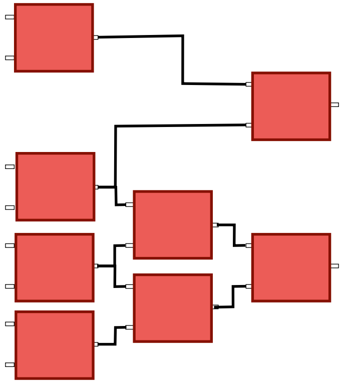


RAM Delegation [CDGGMP17]

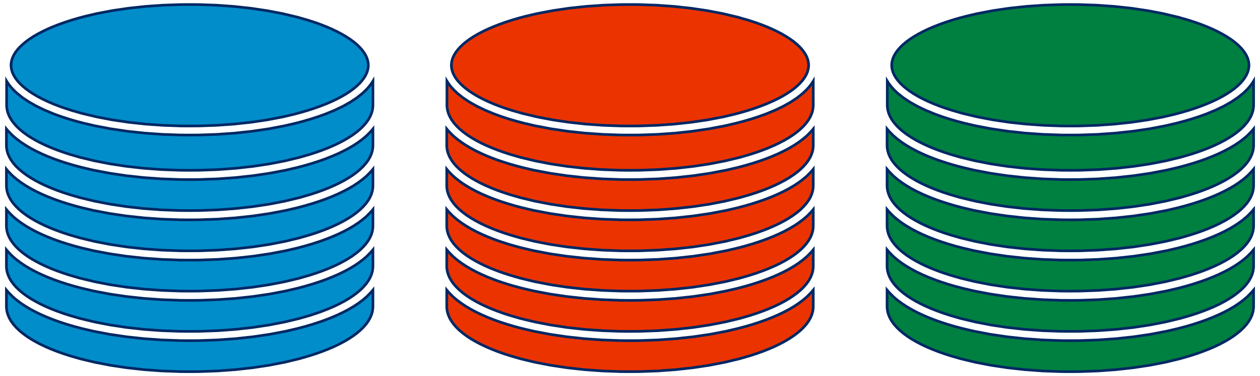


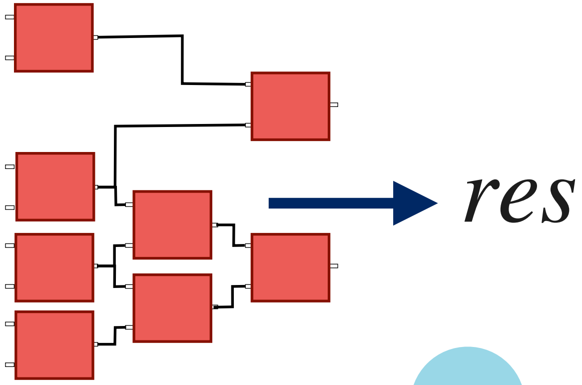
RAM Delegation [CDGGMP17]



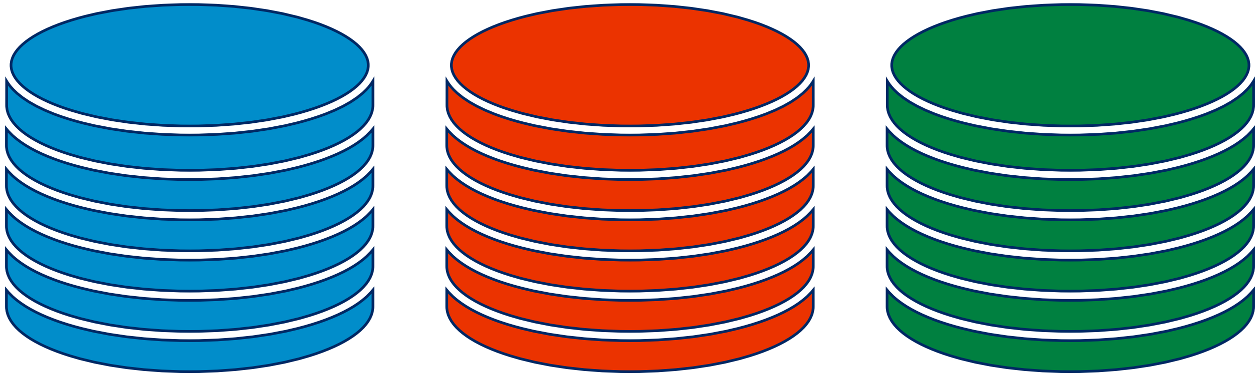


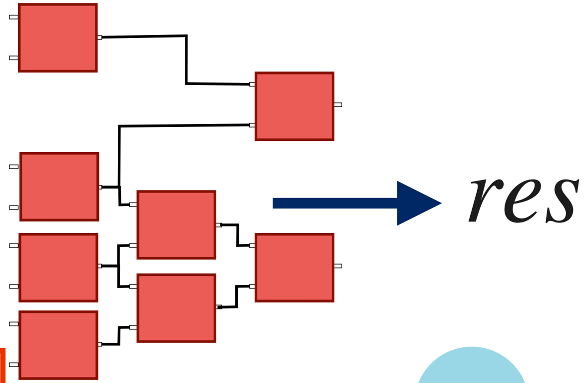
RAM Delegation [CDGGMP17]





RAM Delegation [CDGGMP17]





Nothing more than res is revealed, even if parties collude

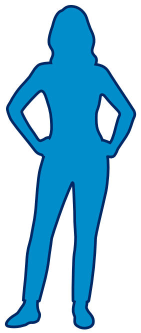
RAM Delegation [CDGGMP17]

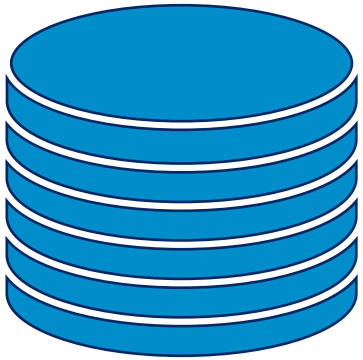
Need a primitive to let garbled program access large database!

Nothing more than res is revealed, even if parties collude

Laconic OT

i, m_0, m_1

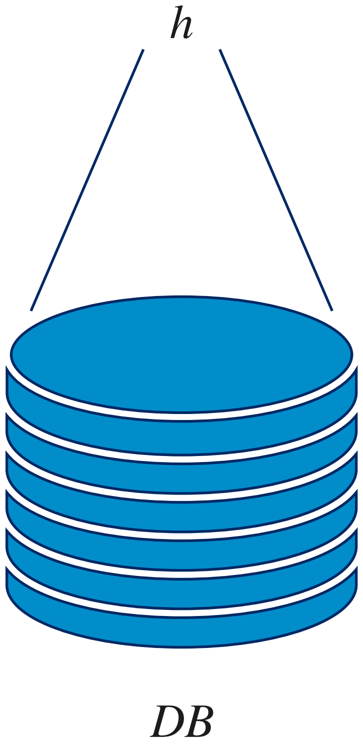


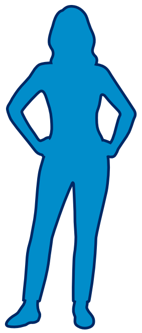


DB

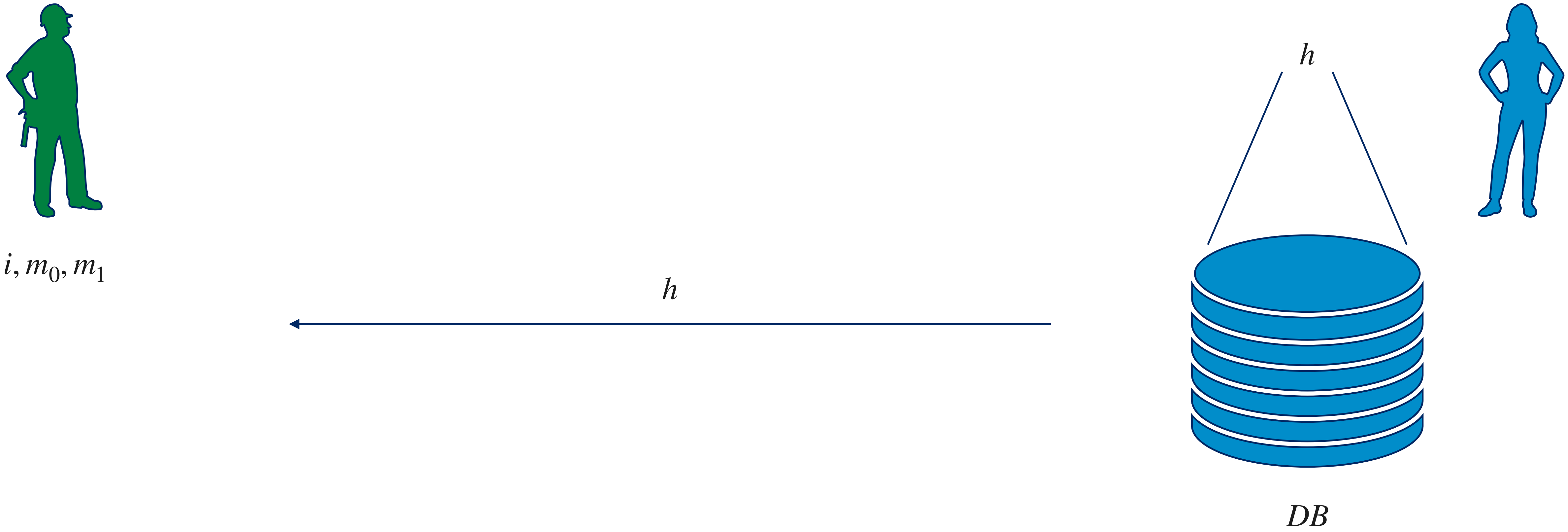
Laconic OT

i, m_0, m_1

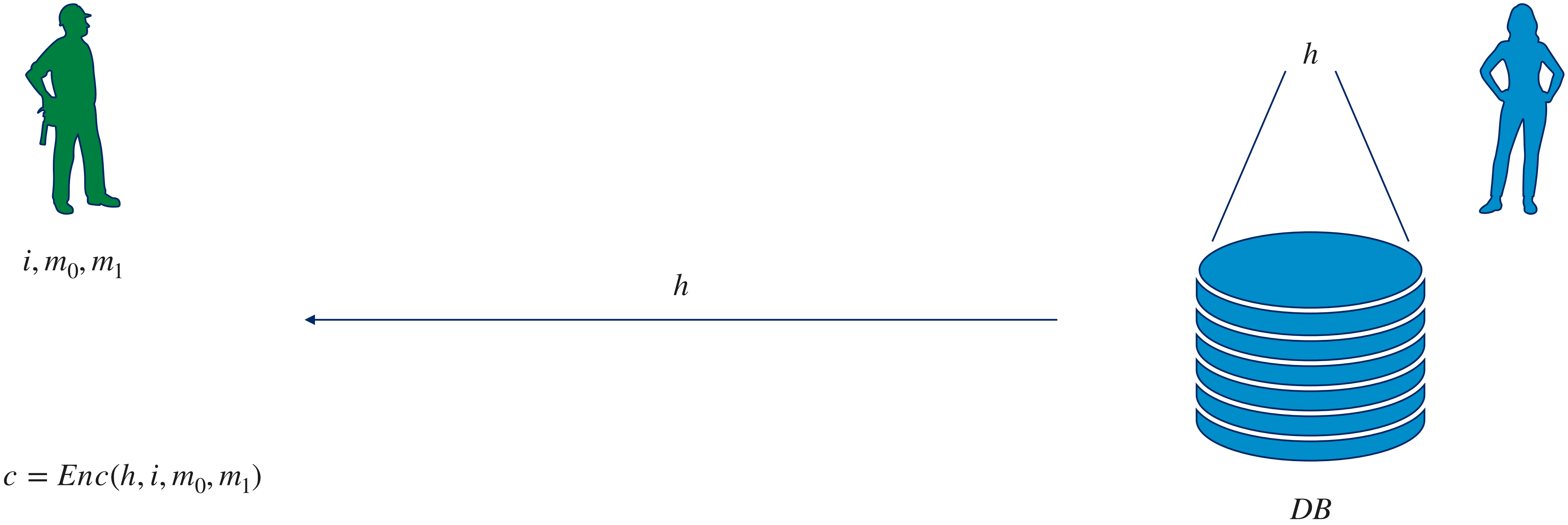




Laconic OT

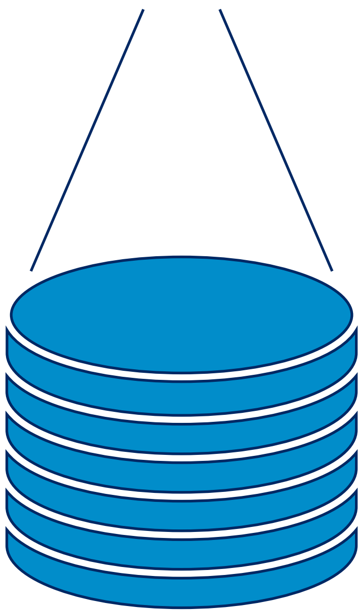


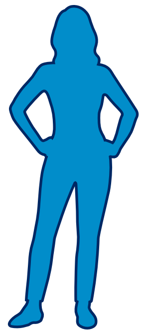
Laconic OT



Laconic OT

i, m_0, m_1





h

$\longleftarrow$

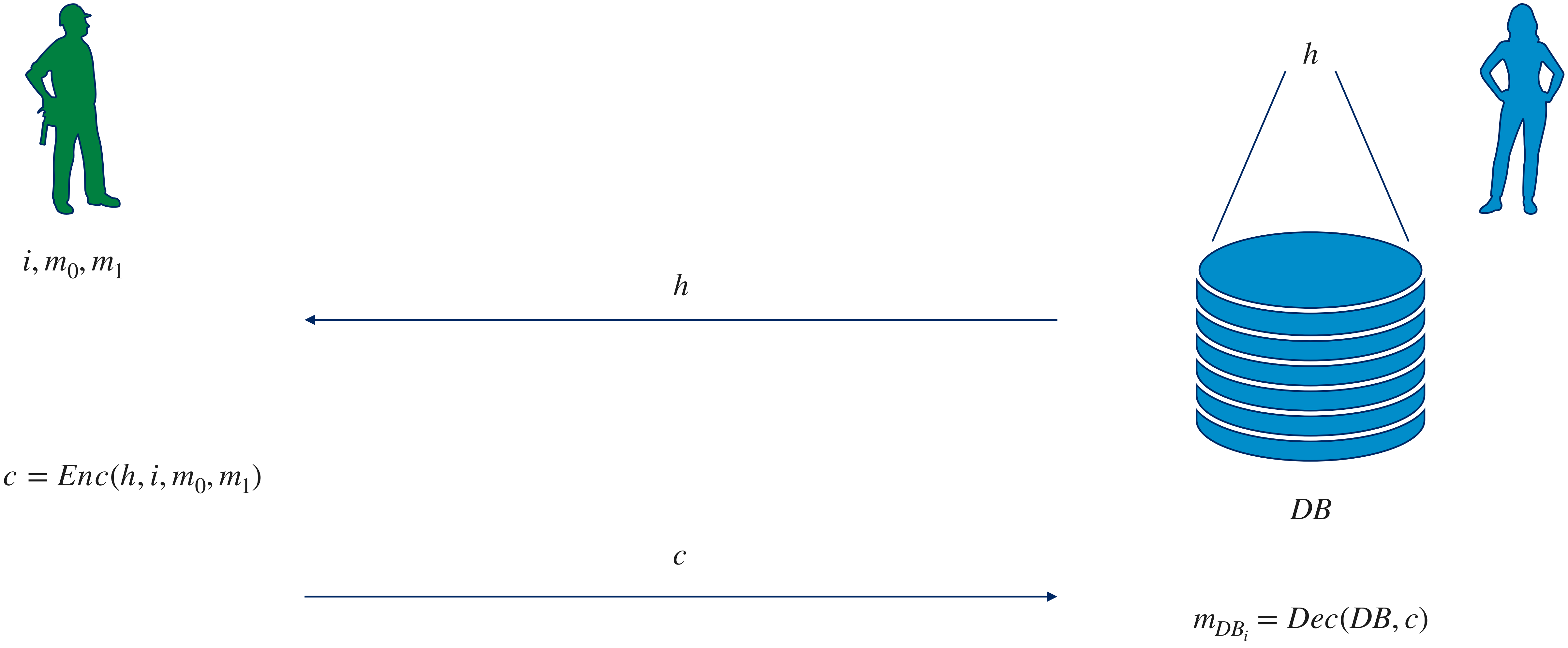
$c = Enc(h, i, m_0, m_1)$

DB

c

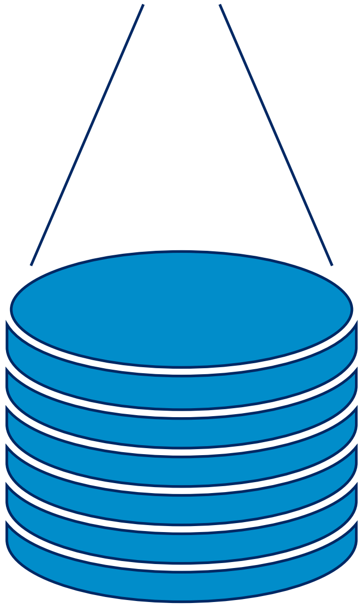
$\longrightarrow$

Laconic OT

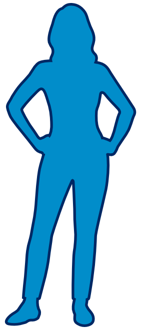


Laconic OT

i, m_0, m_1



h



h

$c = Enc(h, i, m_0, m_1)$

DB

c

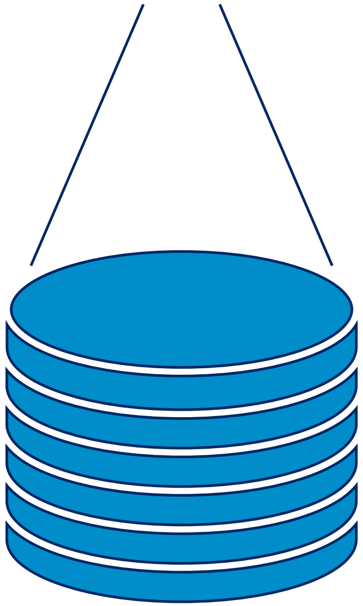
$m_{DB_i} = Dec(DB, c)$

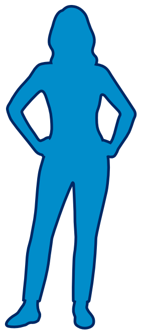
Learns nothing about m_{1-DB_i}

Laconic OT

i, m_0, m_1

h





h

succinct

$c = Enc(h, i, m_0, m_1)$

DB

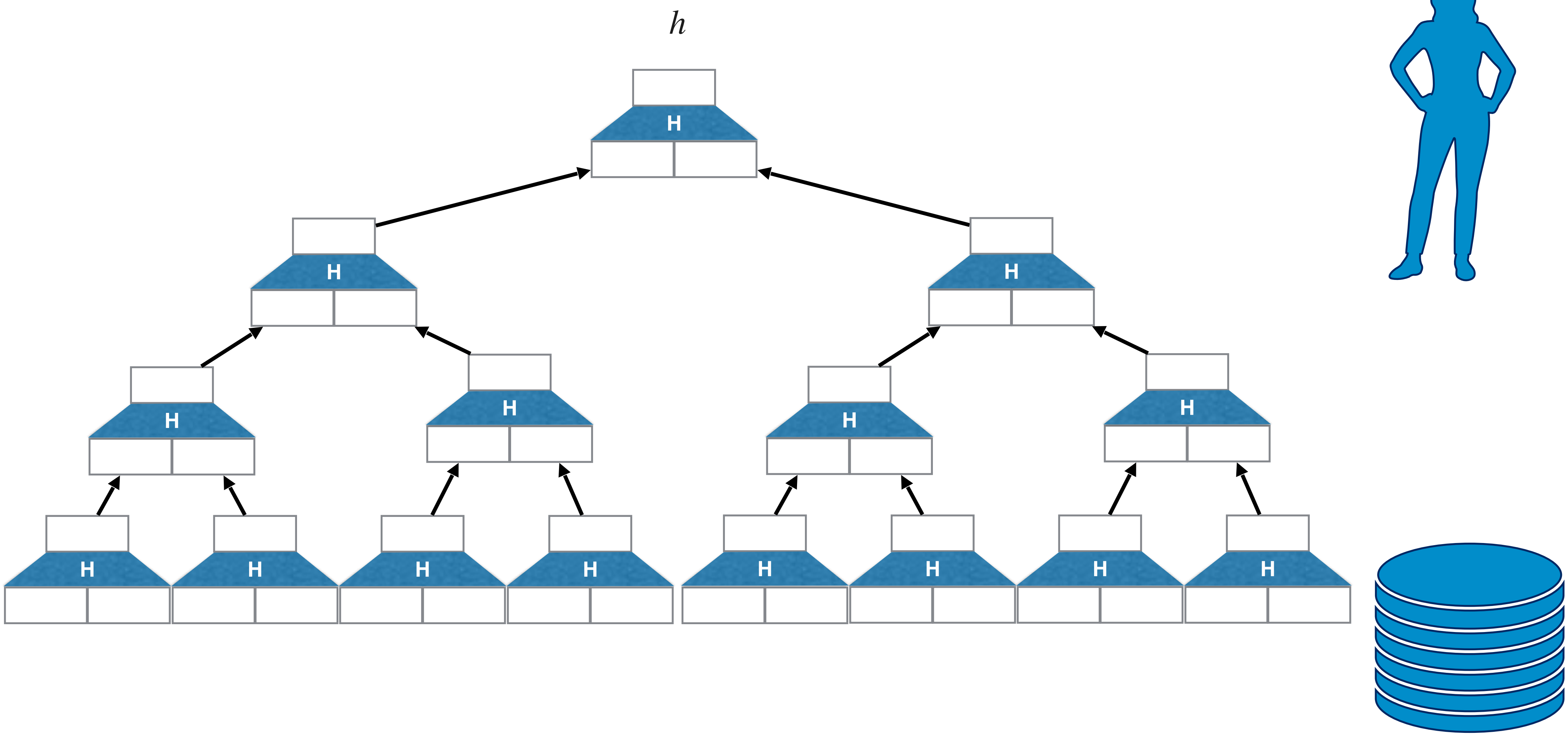
c

$m_{DB_i} = Dec(DB, c)$

Learns nothing about m_{1-DB_i}

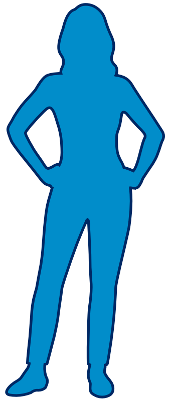
Constructing Laconic OT [CDGGMP17]

Hashing: Merkle Trees



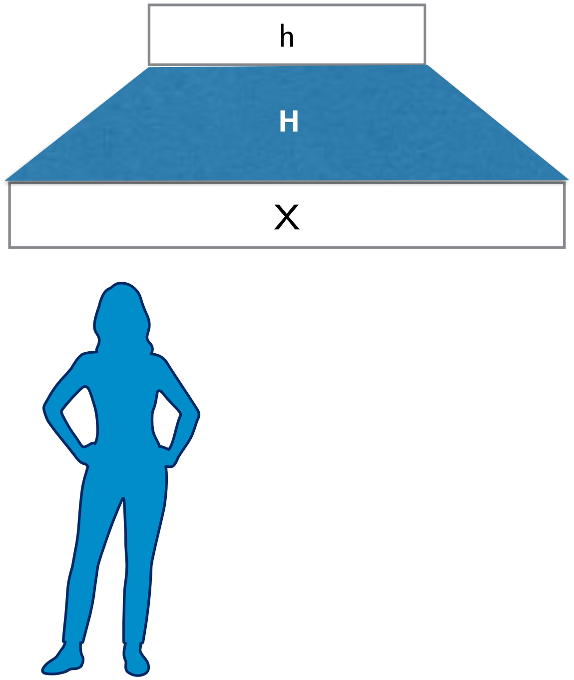
Building Block: Hash Encryption

$h = H(x)$



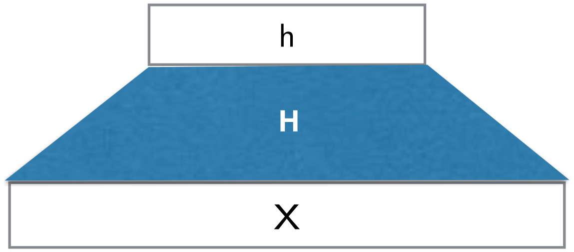
Building Block: Hash Encryption

$$h = H(x)$$

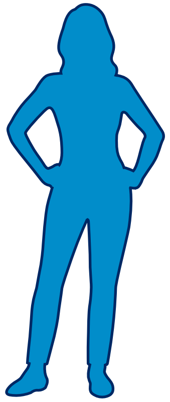


Building Block: Hash Encryption

$h = H(x)$

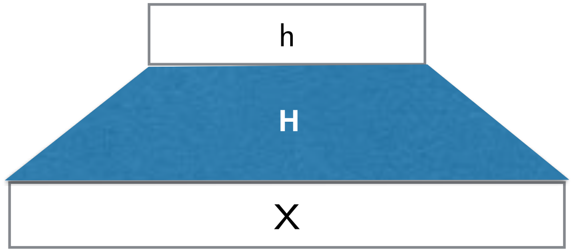


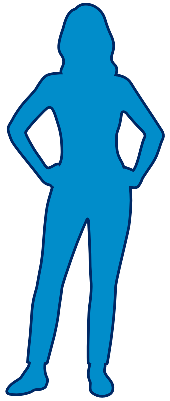
i, b, m



Building Block: Hash Encryption

$h = H(x)$

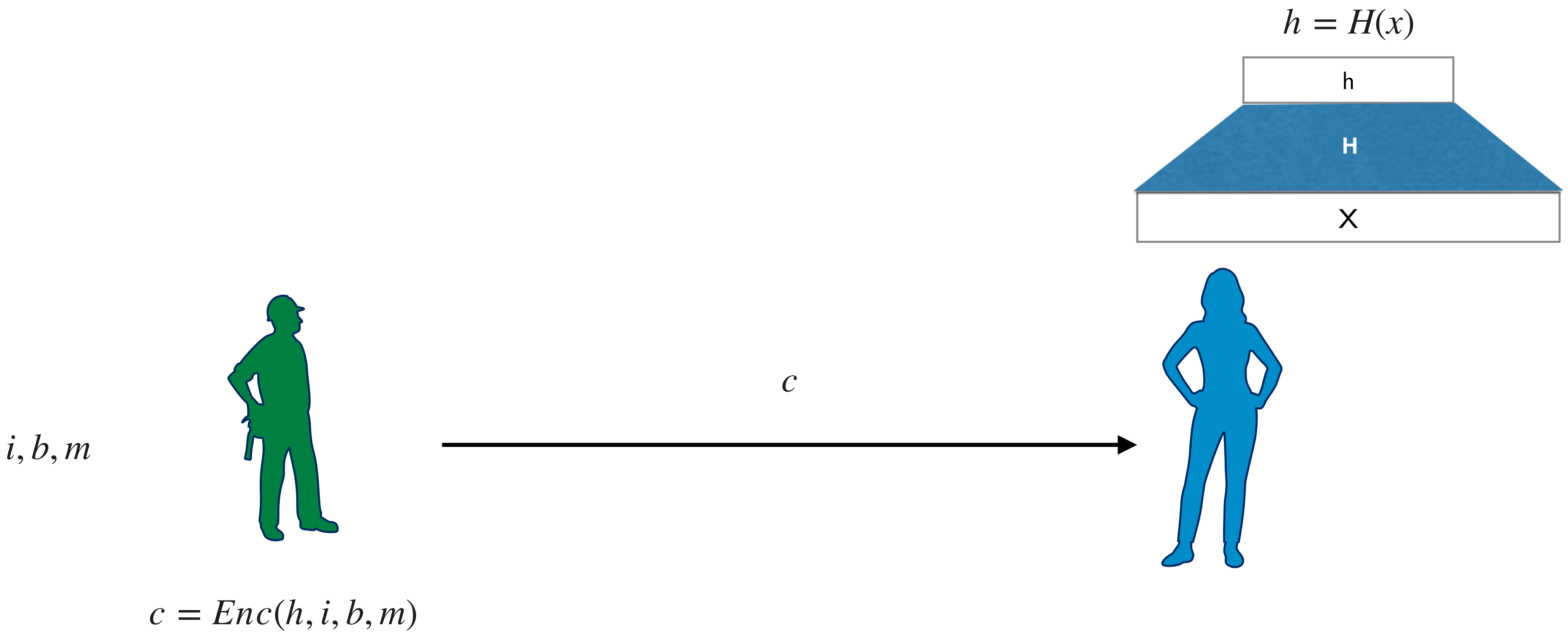




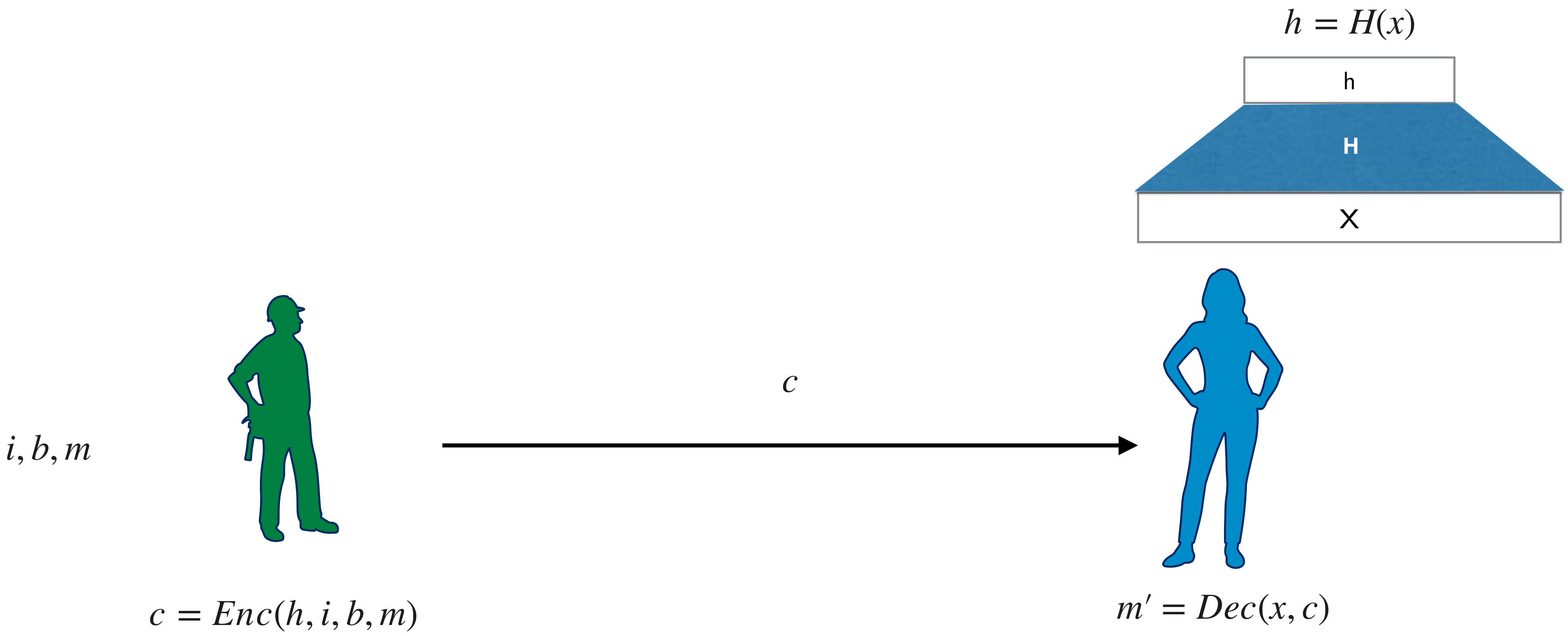
i, b, m

$c = Enc(h, i, b, m)$

Building Block: Hash Encryption

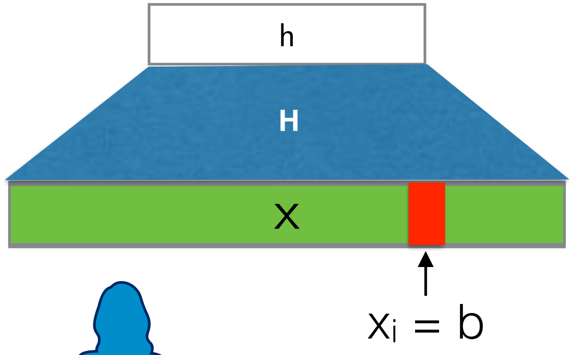


Building Block: Hash Encryption



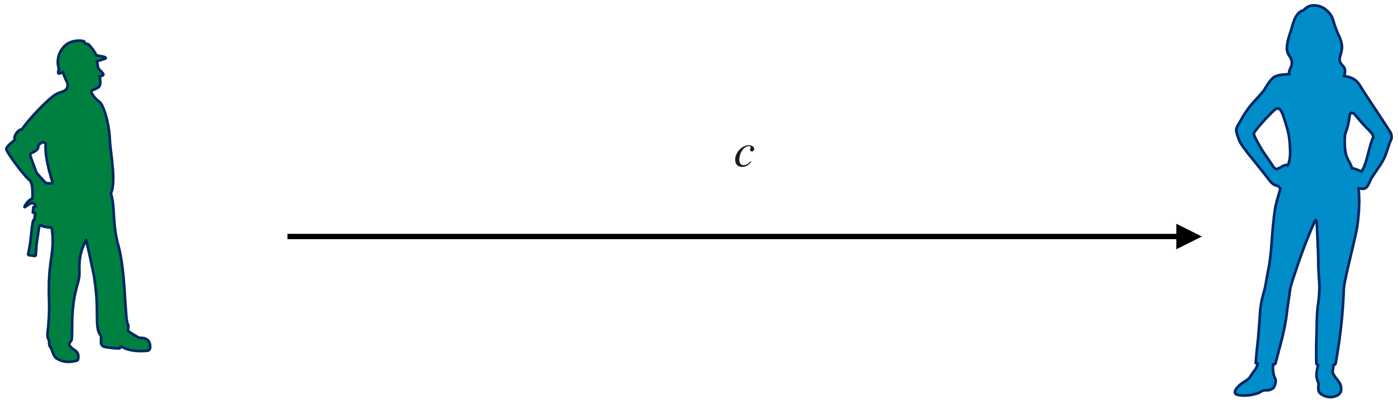
Building Block: Hash Encryption

$h = H(x)$



i, b, m

c



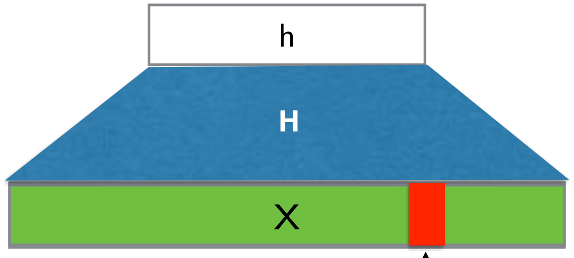
$c = Enc(h, i, b, m)$

$m' = Dec(x, c)$

$m' = m$ if $H(x) = h$ and $x_i = b$

Building Block: Hash Encryption

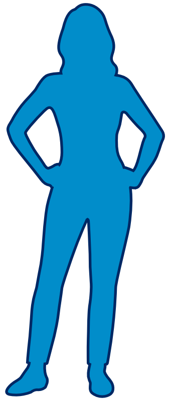
$h = H(x)$



$x_i = b$

c

i, b, m



m'

$c = Enc(h, i, b, m)$

$m' = Dec(x, c)$

$m' = m$ if $H(x) = h$ and $x_i = b$

Hash Encryption from DDH (CDH)

$$\mathsf{k} = \begin{pmatrix} g, & g_{1,0} = g^{\alpha_{1,0}}, \ldots, & g_{i,0} = g^{\alpha_{i,0}}, \ldots, & g_{n,0} = g^{\alpha_{n,0}} \\ & g_{1,1} = g^{\alpha_{1,1}}, \ldots, & g_{i,1} = g^{\alpha_{i,1}}, \ldots, & g_{n,1} = g^{\alpha_{n,}} \end{pmatrix}$$

$$H(k, x; r) \leftarrow g^r \cdot \prod_j g_{j,x_j} = h$$

$Enc(k, (h, i, b), m):$

$$c_1 \leftarrow h^s$$

$$e \leftarrow g_{i,b}^s \cdot m$$

$$c_0 \leftarrow g^s$$

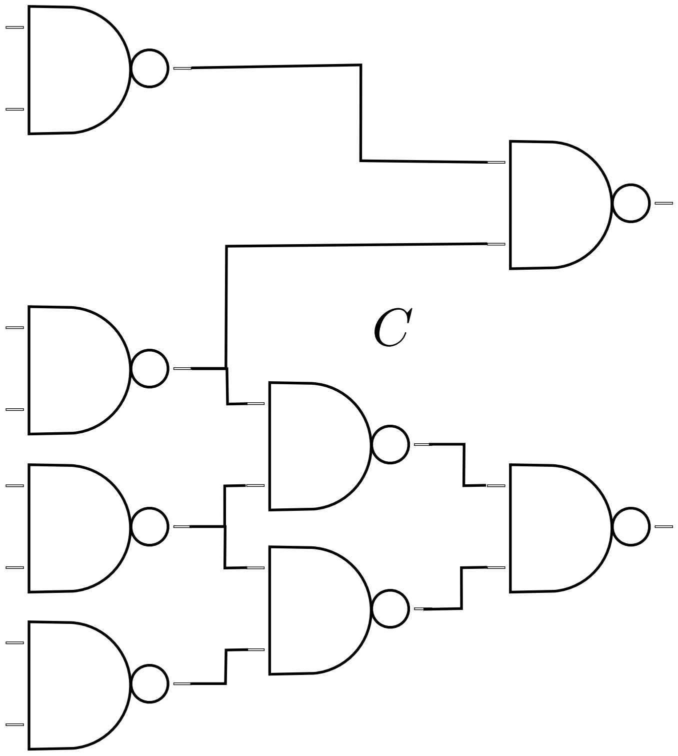
$$\forall j \neq i : c_{j,0} \leftarrow g_{j,0}^s$$

$$c_{j,1} \leftarrow g_{j,1}^s$$

$Dec(k, (x, r), c):$

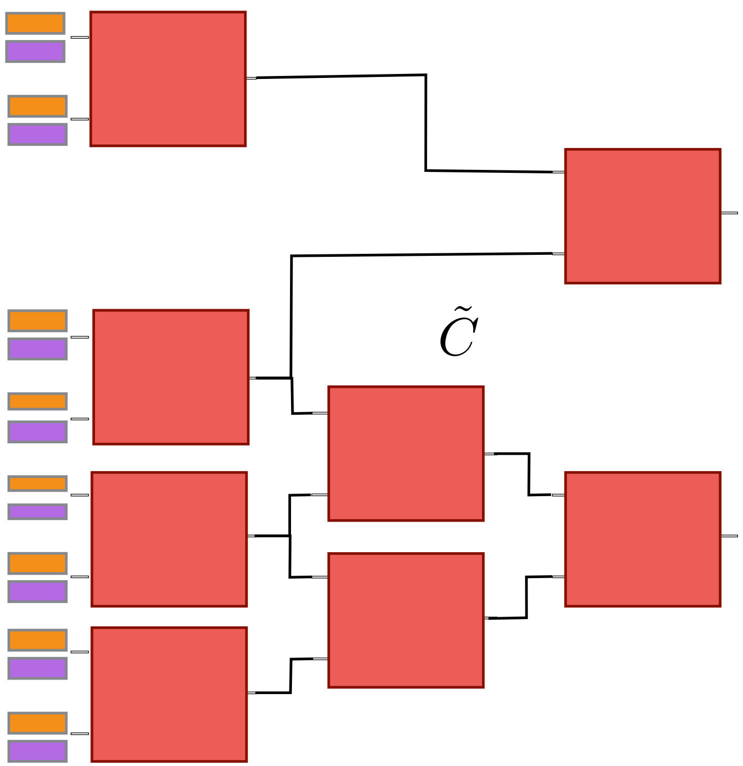
$$\begin{aligned} m &\leftarrow e \cdot \frac{c_0^r \cdot \prod_{j \neq i} c_{j,x_j}}{c_1} \\ &= e \cdot \frac{c_0^r \cdot \prod_{j \neq i} c_{j,x_j}}{h^s} \\ &= e \cdot \frac{c_0^r \cdot \prod_{j \neq i} c_{j,x_j}}{g_{i,x_i}^s \cdot c_0^r \cdot \prod_{j \neq i} c_{j,x_j}} \\ &= m \cdot \frac{g_{i,b}^s}{g_{i,x_i}^s} \quad = m \end{aligned}$$

Building Block: Garbled Circuits



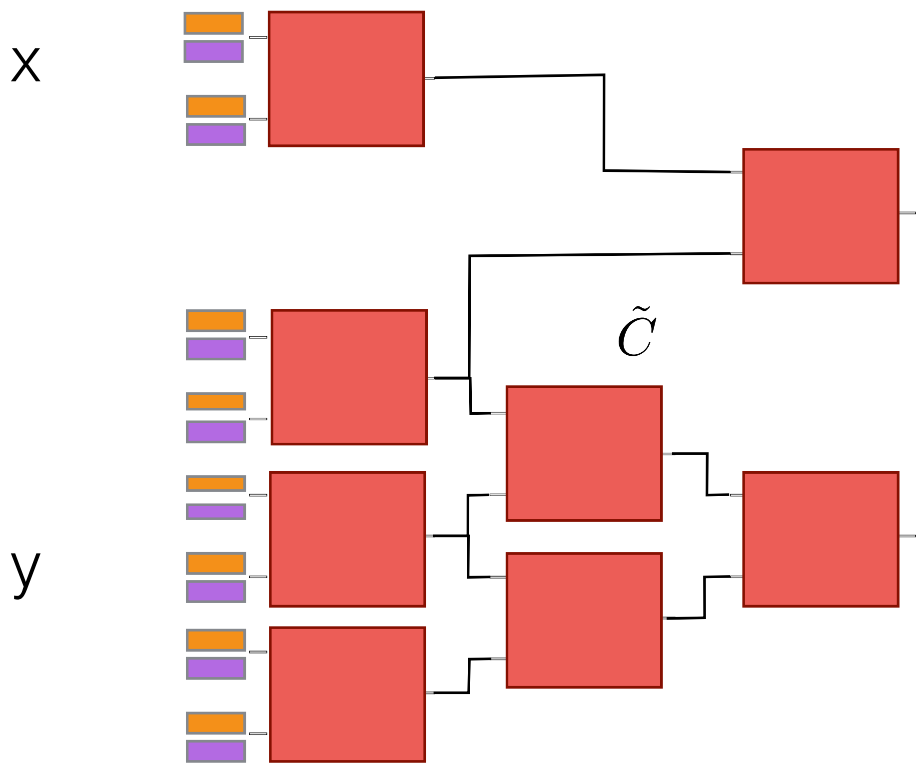
Building Block: Garbled Circuits

$$(\tilde{C}, \mathsf{lab}) \leftarrow Garble(C)$$



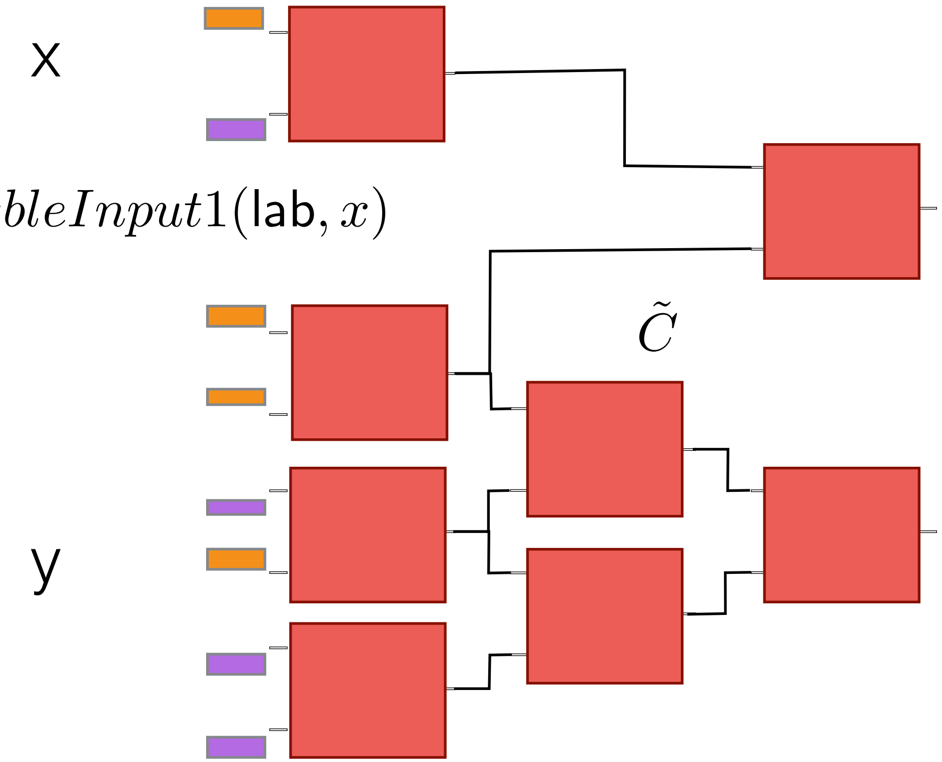
Building Block: Garbled Circuits

$(\tilde{C}, \mathsf{lab}) \leftarrow Garble(C)$



Building Block: Garbled Circuits

$$(\tilde{C}, \mathsf{lab}) \leftarrow Garble(C)$$



$$\tilde{x} \leftarrow GarbleInput1(\mathsf{lab}, x)$$

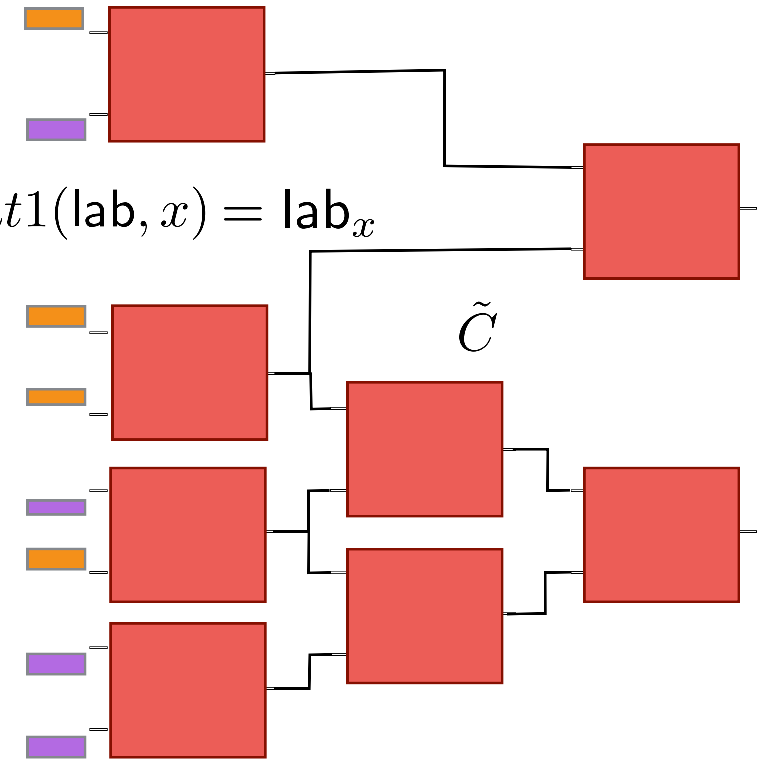
$$\tilde{y} \leftarrow GarbleInput1(\mathsf{lab}, y)$$

Building Block: Garbled Circuits

$(\tilde{C}, \mathsf{lab}) \leftarrow Garble(C)$

x

$\tilde{x} \leftarrow GarbleInput1(\mathsf{lab}, x) = \mathsf{lab}_x$



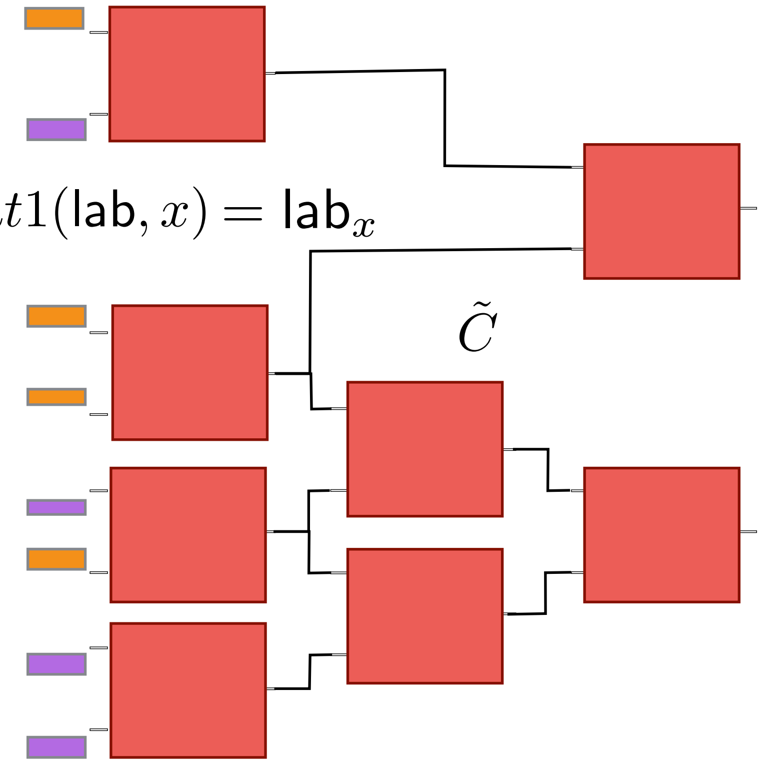
y

$\tilde{y} \leftarrow GarbleInput1(\mathsf{lab}, y) = \mathsf{lab}_y$

Building Block: Garbled Circuits

$(\tilde{C}, \mathsf{lab}) \leftarrow Garble(C)$

x



$\tilde{x} \leftarrow GarbleInput1(\mathsf{lab}, x) = \mathsf{lab}_x$

$Eval(\tilde{C}, \tilde{x}, \tilde{y})$

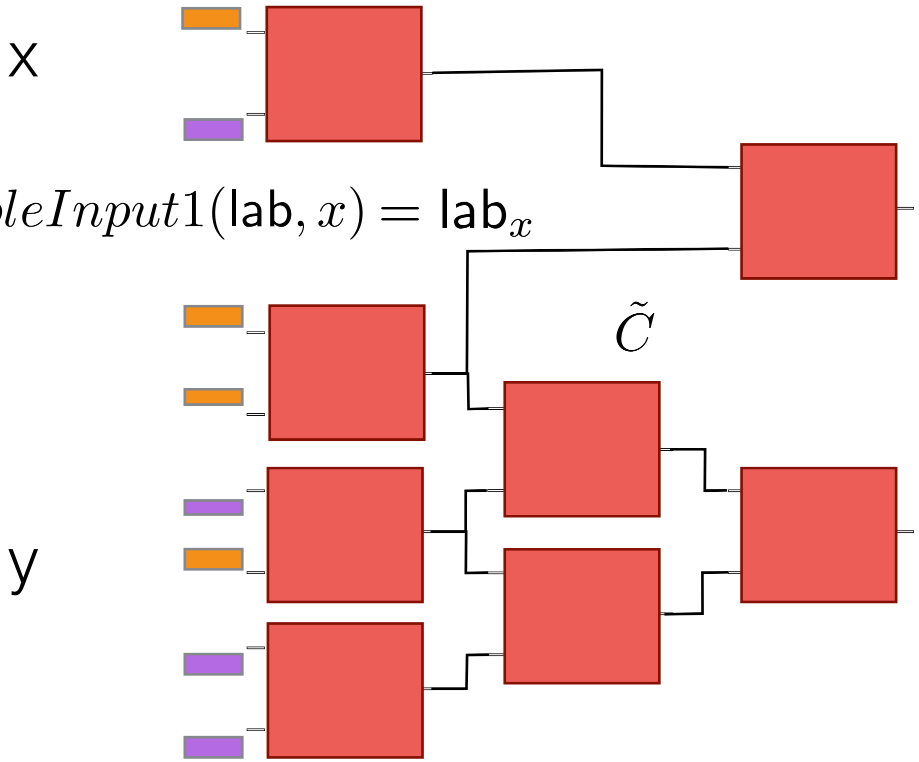
y

$\tilde{y} \leftarrow GarbleInput1(\mathsf{lab}, y) = \mathsf{lab}_y$

Building Block: Garbled Circuits

$$(\tilde{C}, \mathsf{lab}) \leftarrow Garble(C)$$

$$\tilde{x} \leftarrow GarbleInput1(\mathsf{lab}, x) = \mathsf{lab}_x$$



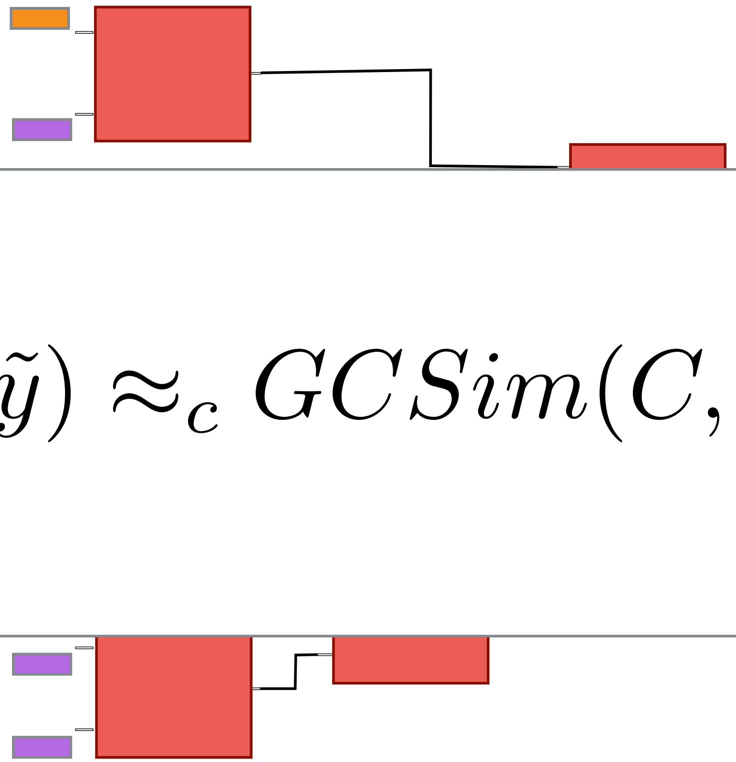
$$Eval(\tilde{C}, \tilde{x}, \tilde{y}) = C(x, y)$$

$$\tilde{y} \leftarrow GarbleInput1(\mathsf{lab}, y) = \mathsf{lab}_y$$

Building Block: Garbled Circuits

$(\tilde{C}, \mathsf{lab}) \leftarrow Garble(C)$

X



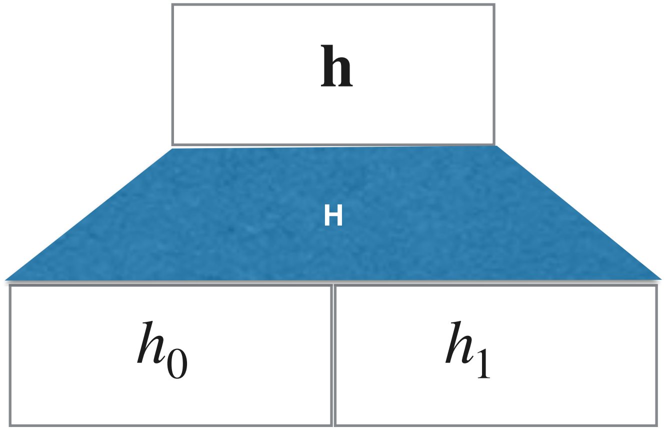
$\tilde{x}$

$$(\tilde{C}, \tilde{x}, \tilde{y}) \approx_c GCSim(C, C(x, y))$$

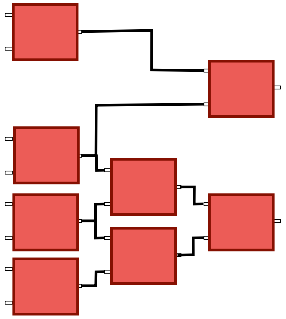
$= C(x, y)$

$\tilde{y} \leftarrow GarbleInput1(\mathsf{lab}, y) = \mathsf{lab}_y$

Bootstrapping/Recryption

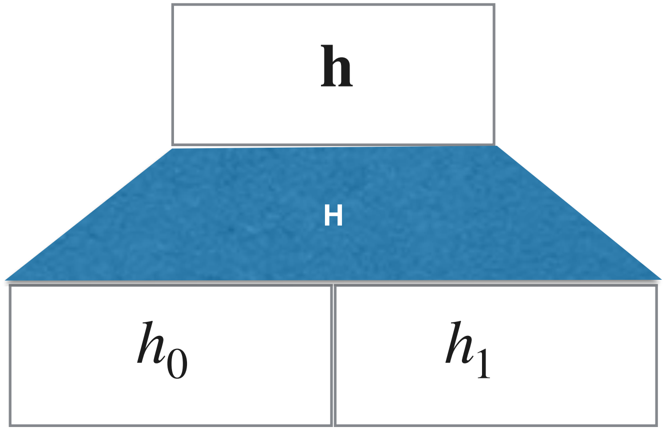


GC

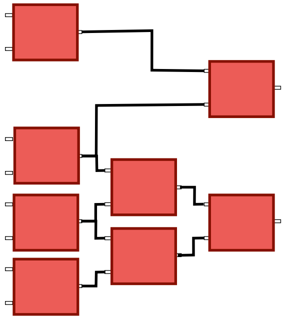


Bootstrapping/Recryption

$$\mathbf{c} = \mathsf{Enc}(h, lab)$$

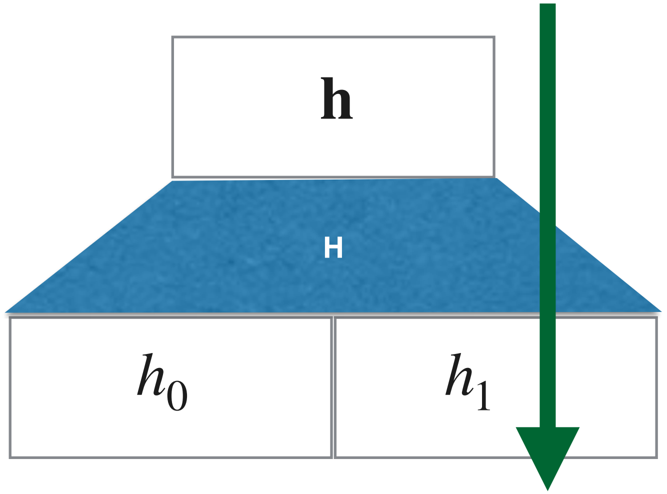


GC



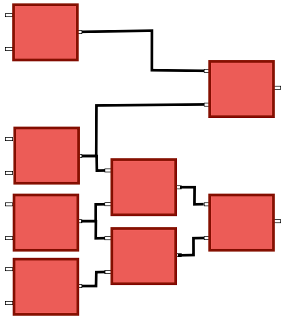
Bootstrapping/Recryption

$\mathbf{c} = \mathsf{Enc}(h, lab)$



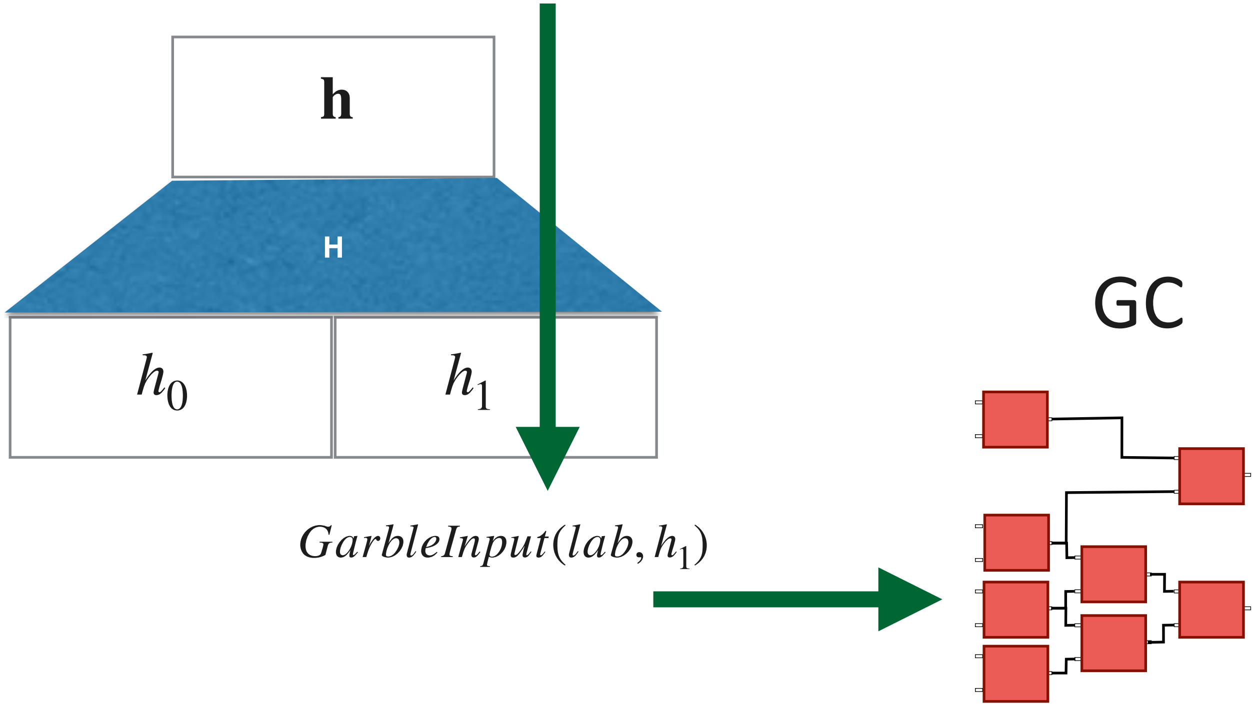
$GarbleInput(lab, h_1)$

GC



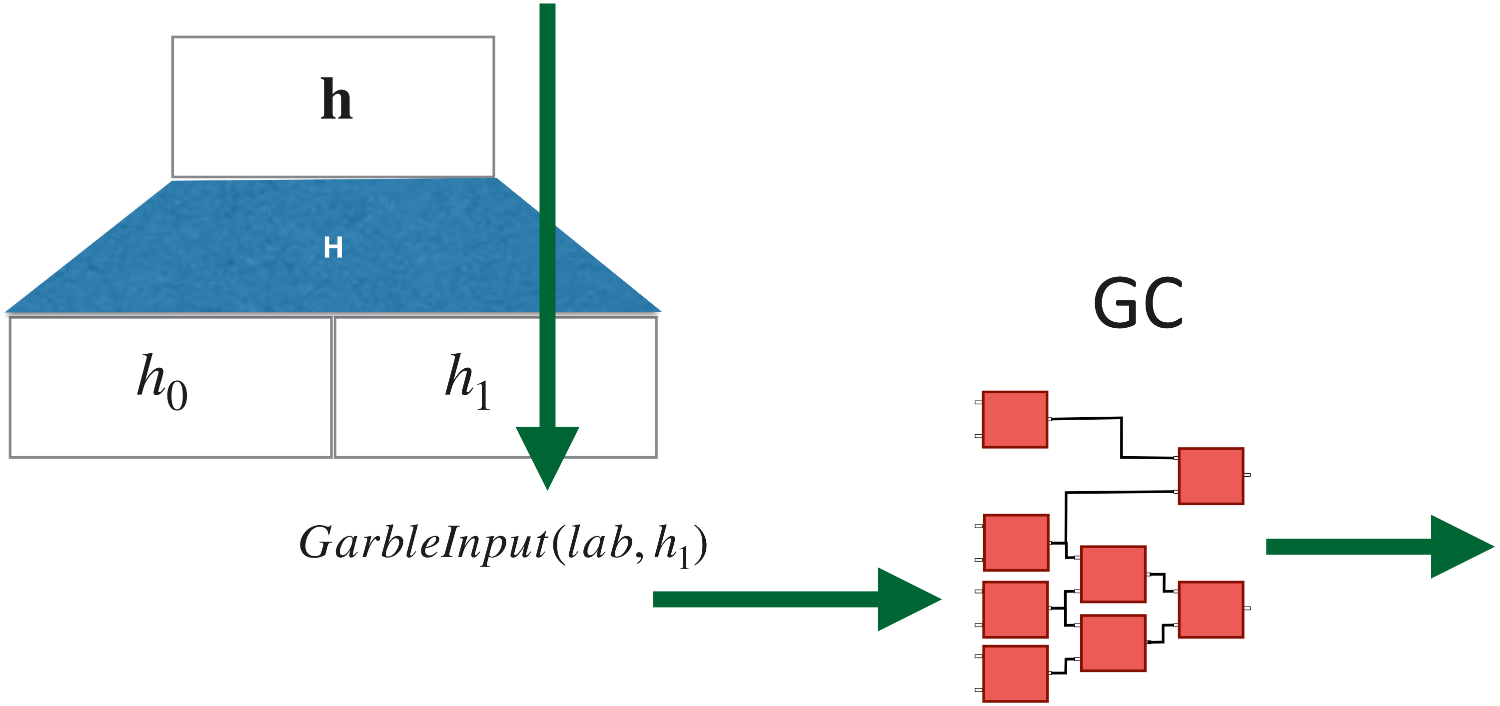
Bootstrapping/Recryption

$\mathbf{c} = \mathsf{Enc}(h, lab)$



Bootstrapping/Recryption

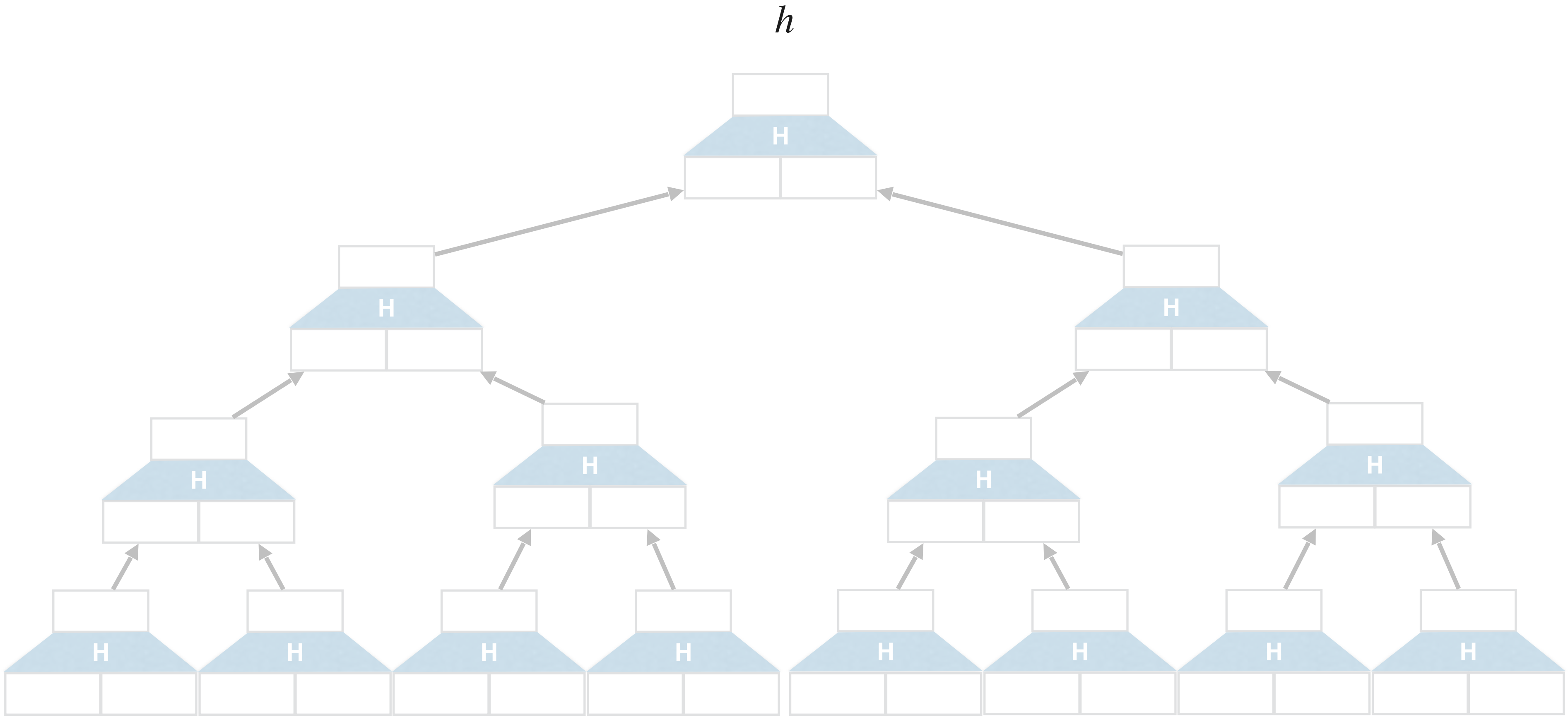
$\mathbf{c} = \mathsf{Enc}(h, lab)$



$\mathbf{c}' = \mathsf{Enc}(h_1, m)$

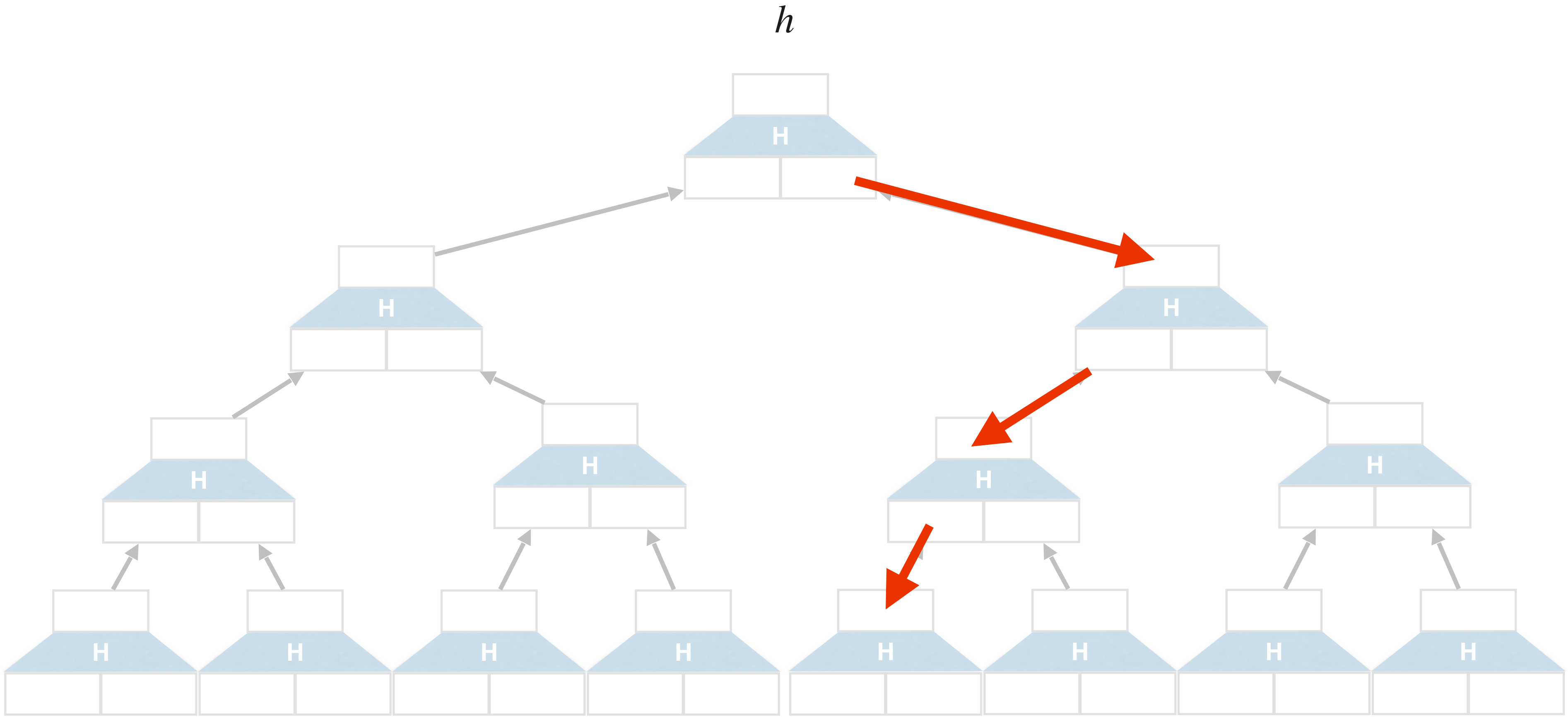
Laconic OT: Encryption

i m_0, m_1

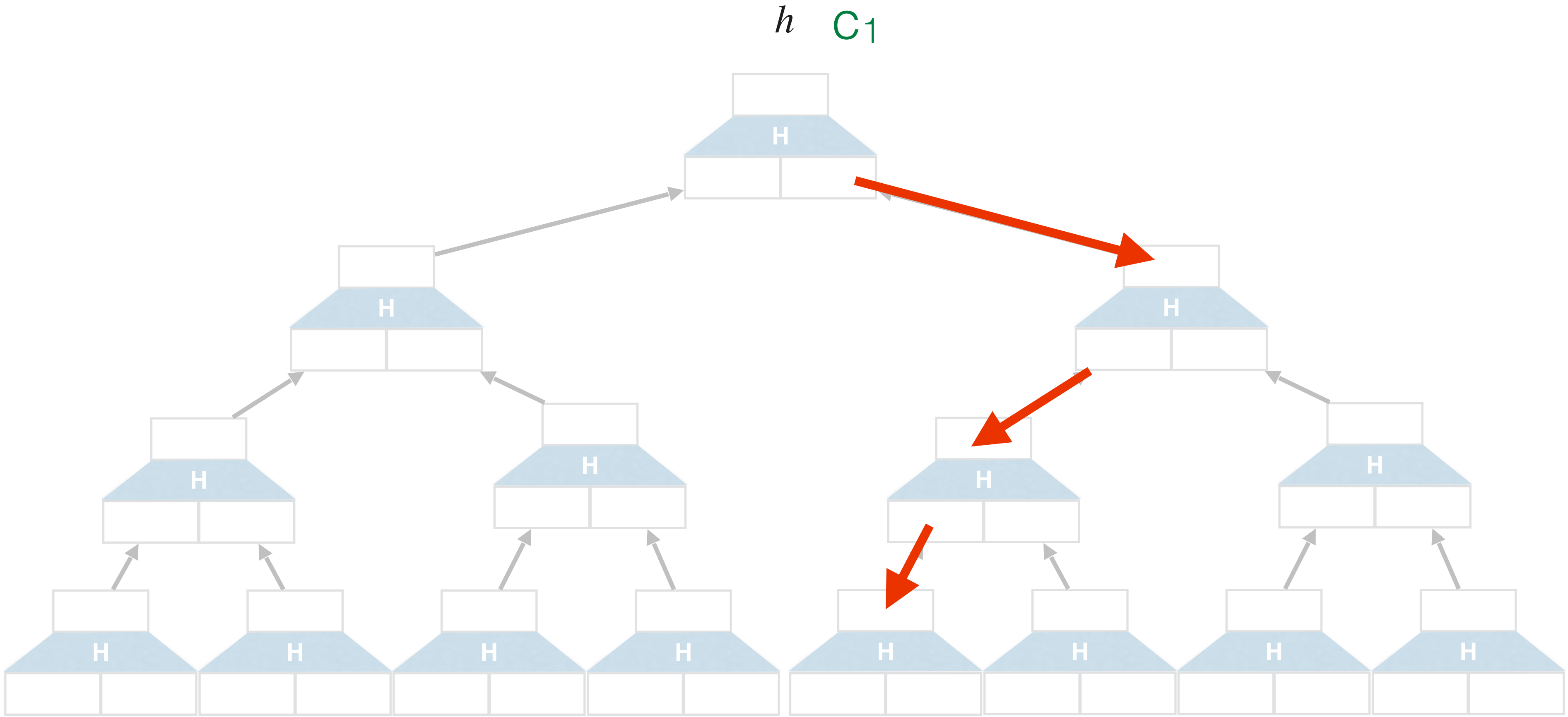


Laconic OT: Encryption

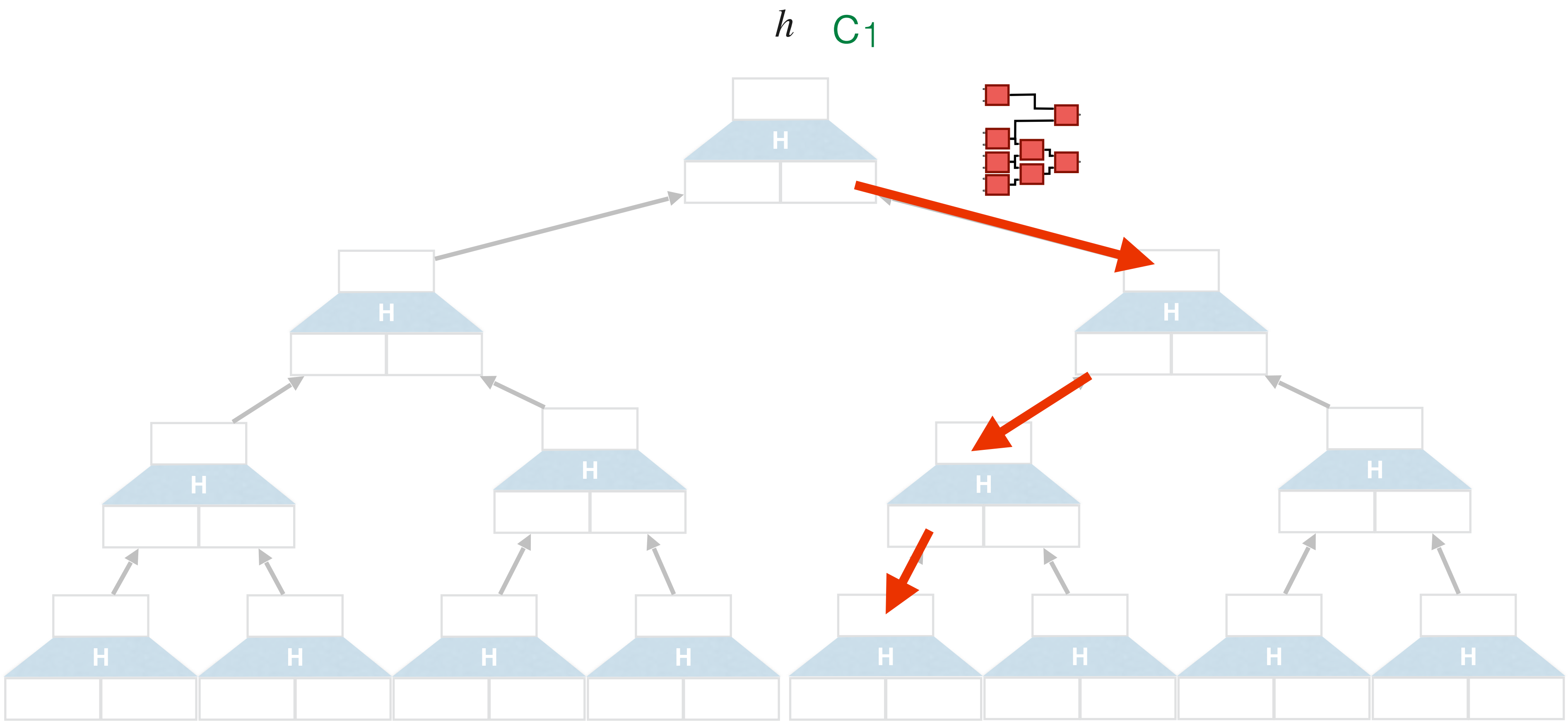
$i \quad m_0, m_1$



Laconic OT: Encryption

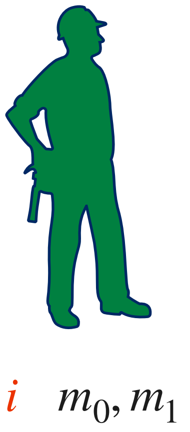


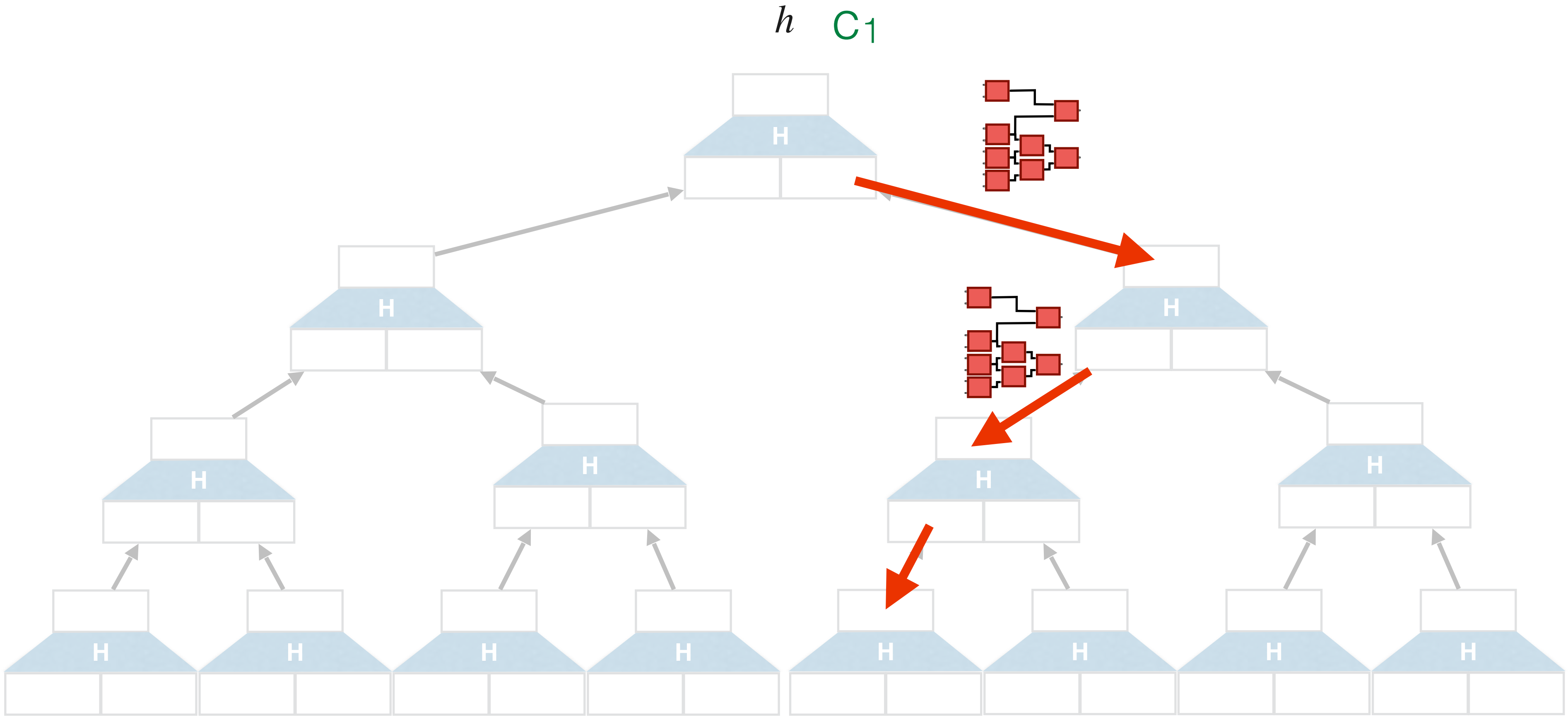
Laconic OT: Encryption



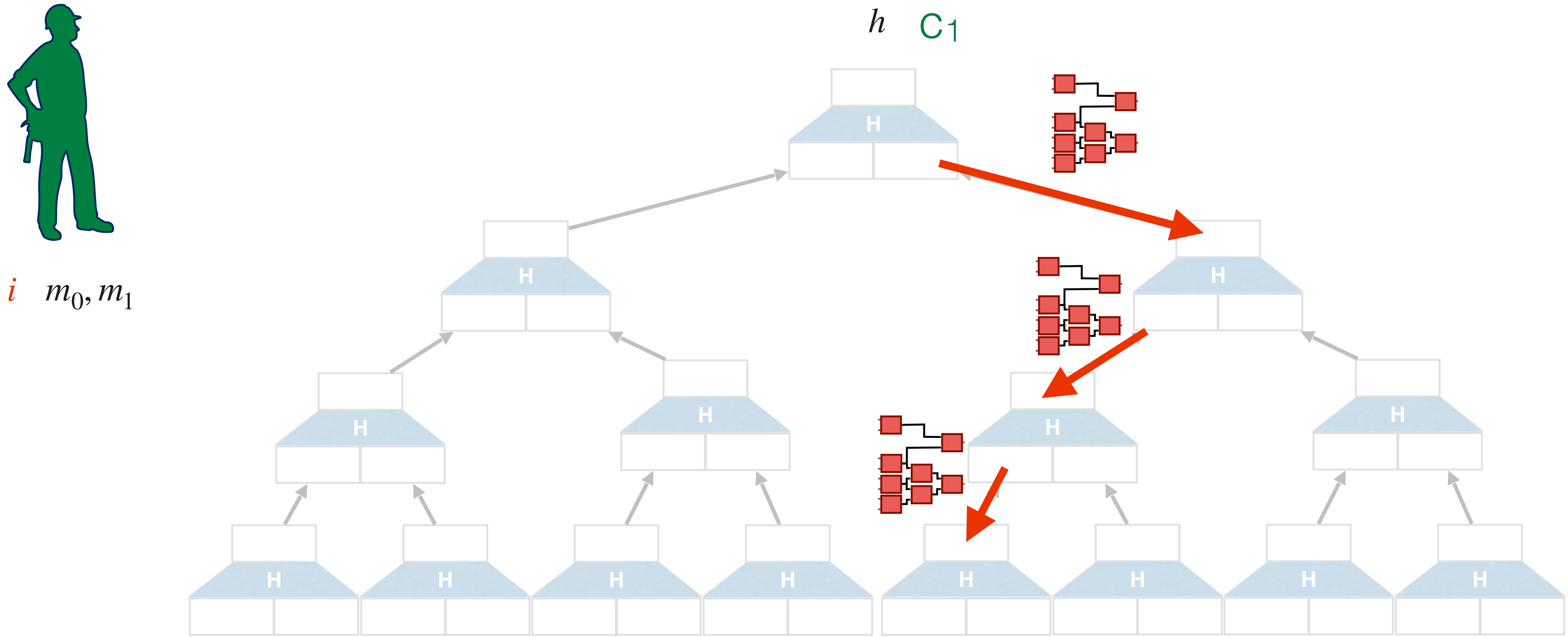
i m_0, m_1

Laconic OT: Encryption

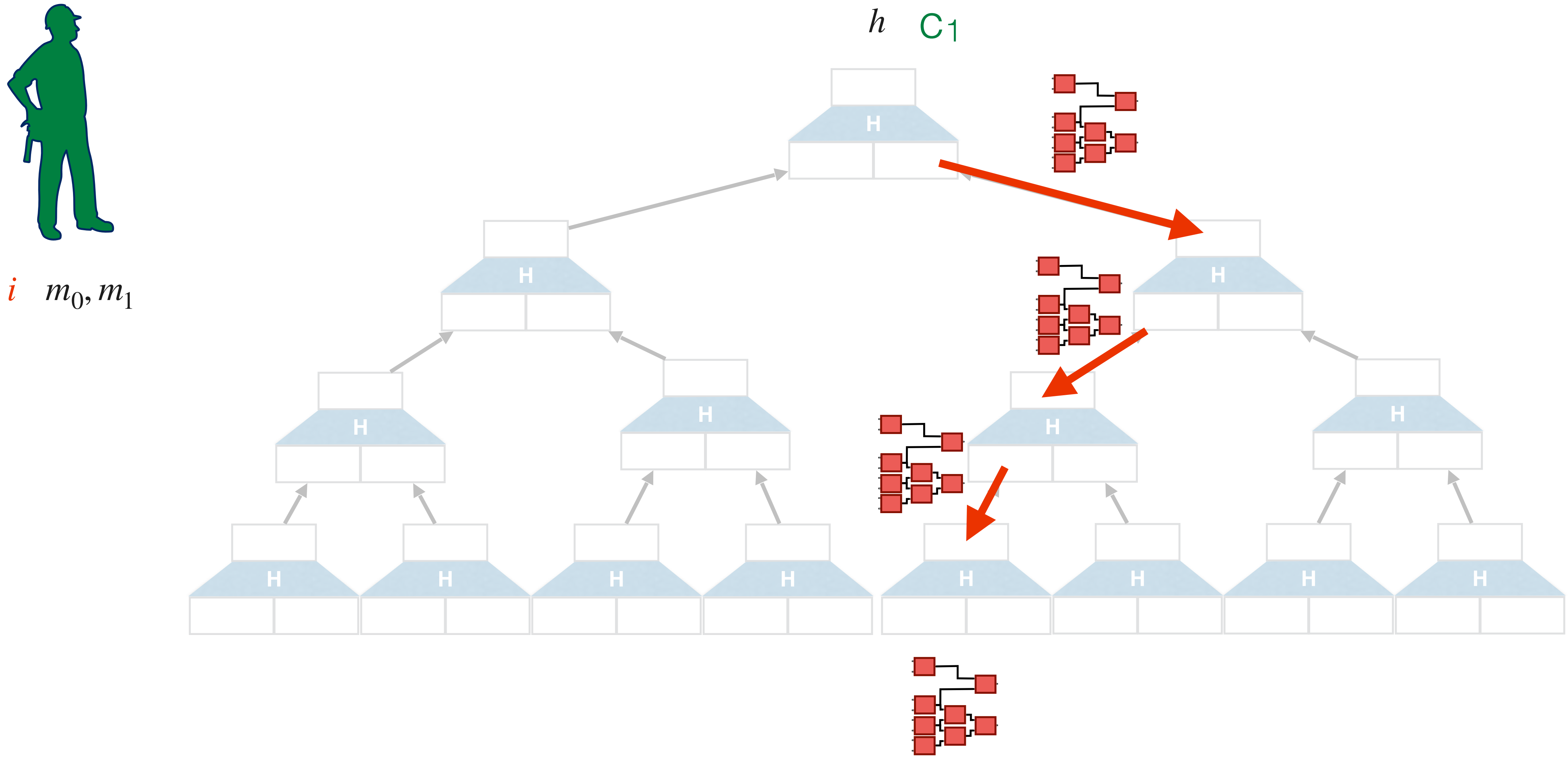




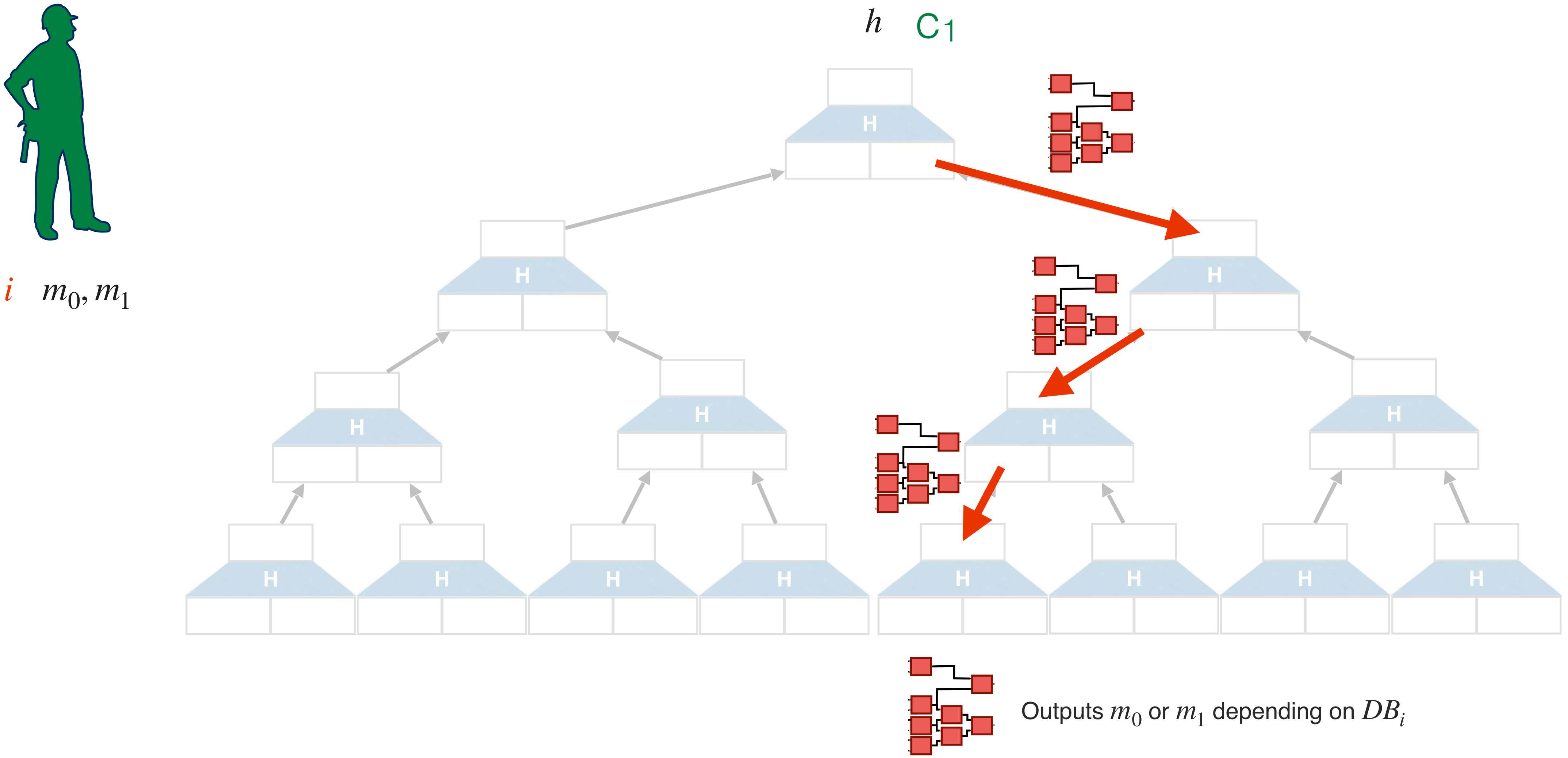
Laconic OT: Encryption



Laconic OT: Encryption

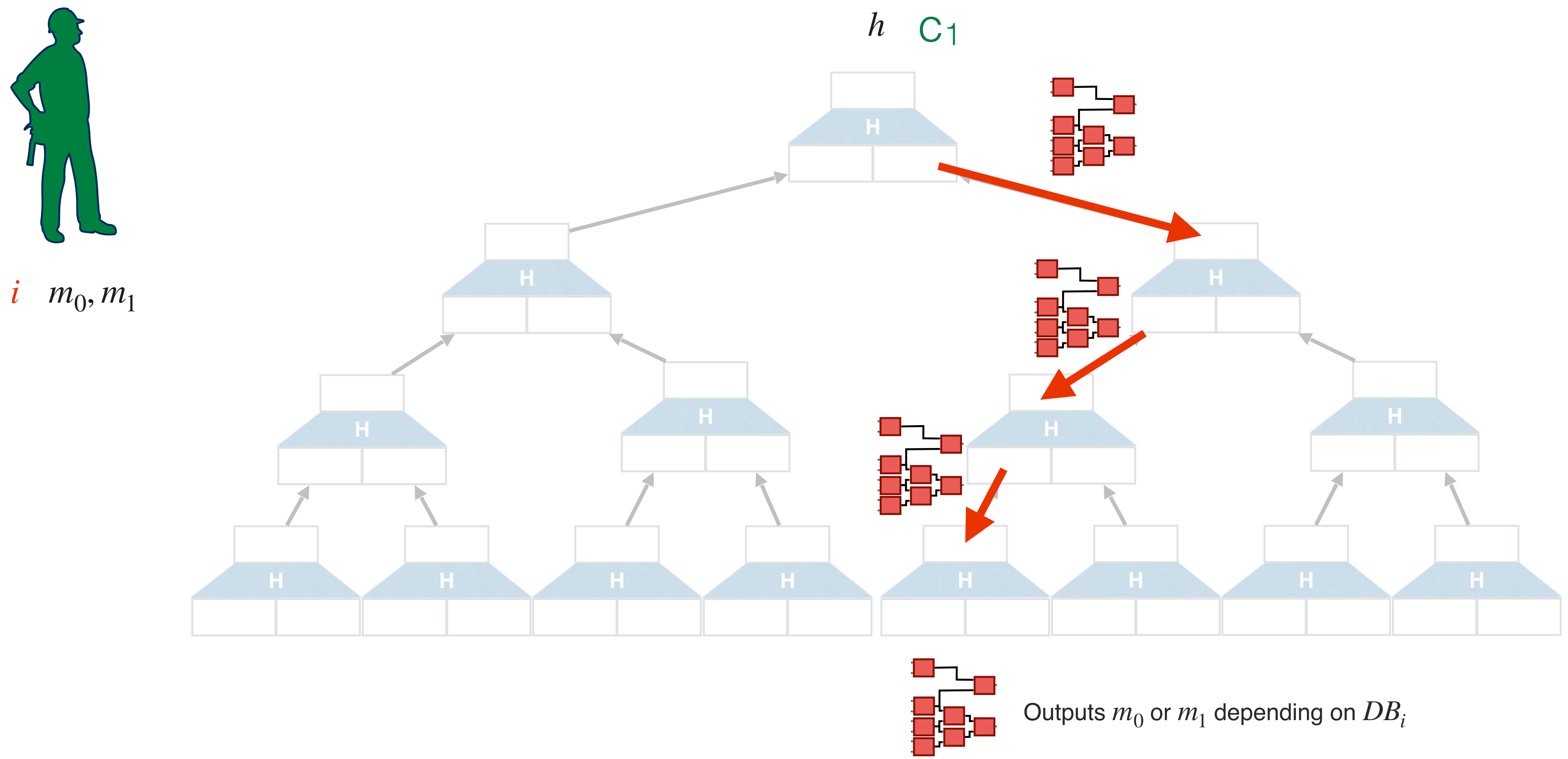


Laconic OT: Encryption

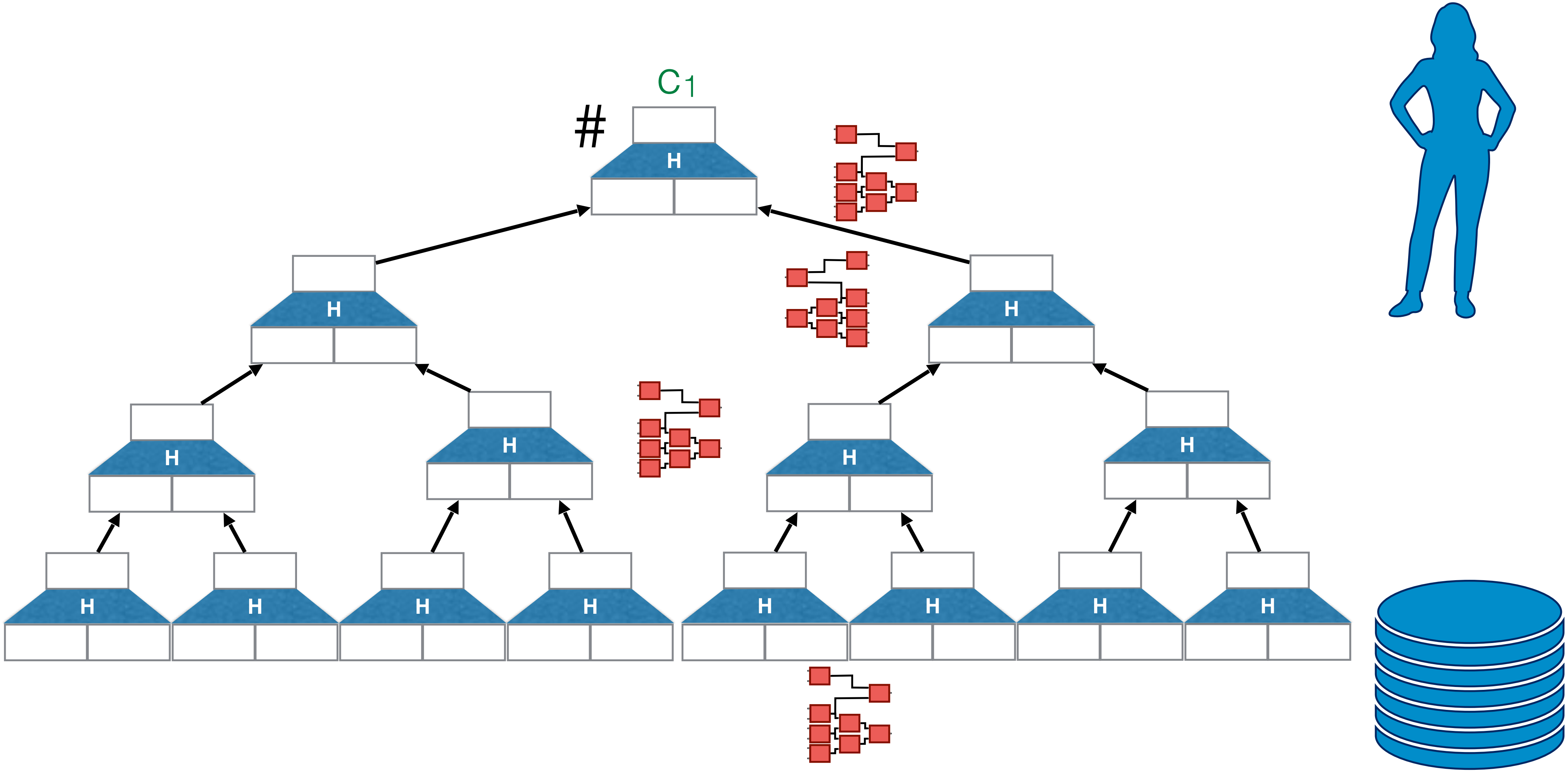


Laconic OT: Encryption

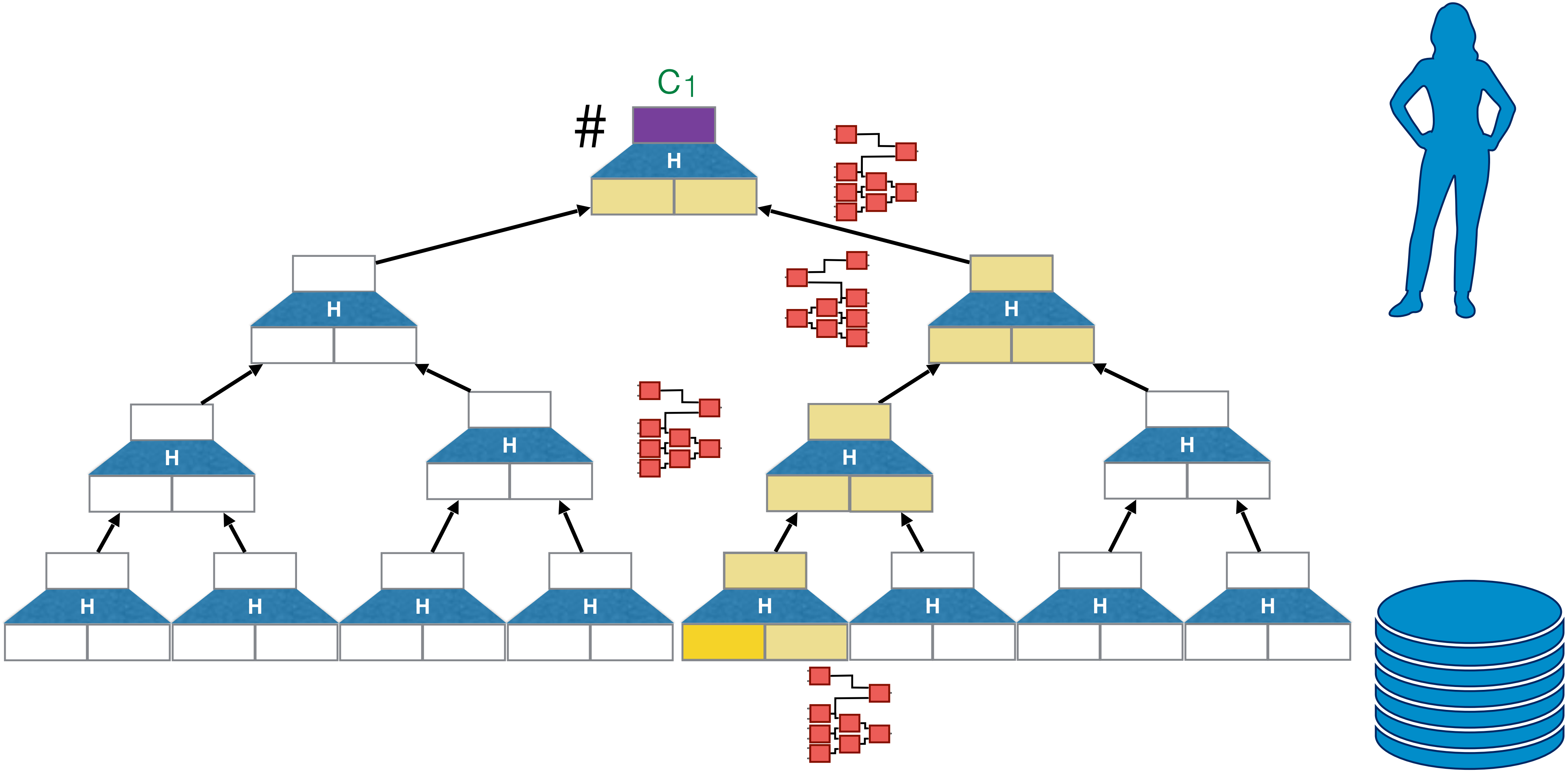
Core Paradigm: Delegate Work “into the Future”



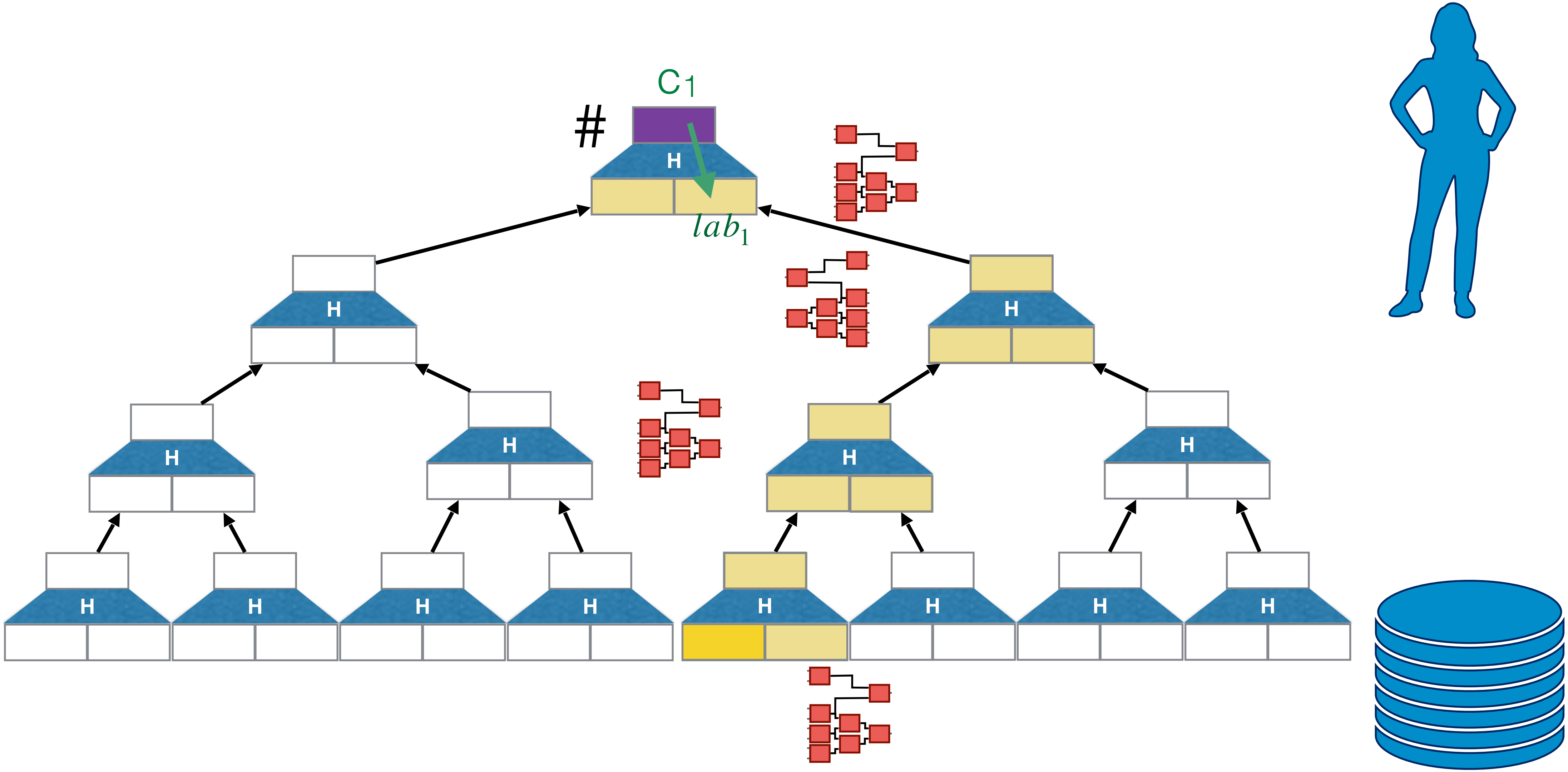
Laconic OT: Decryption



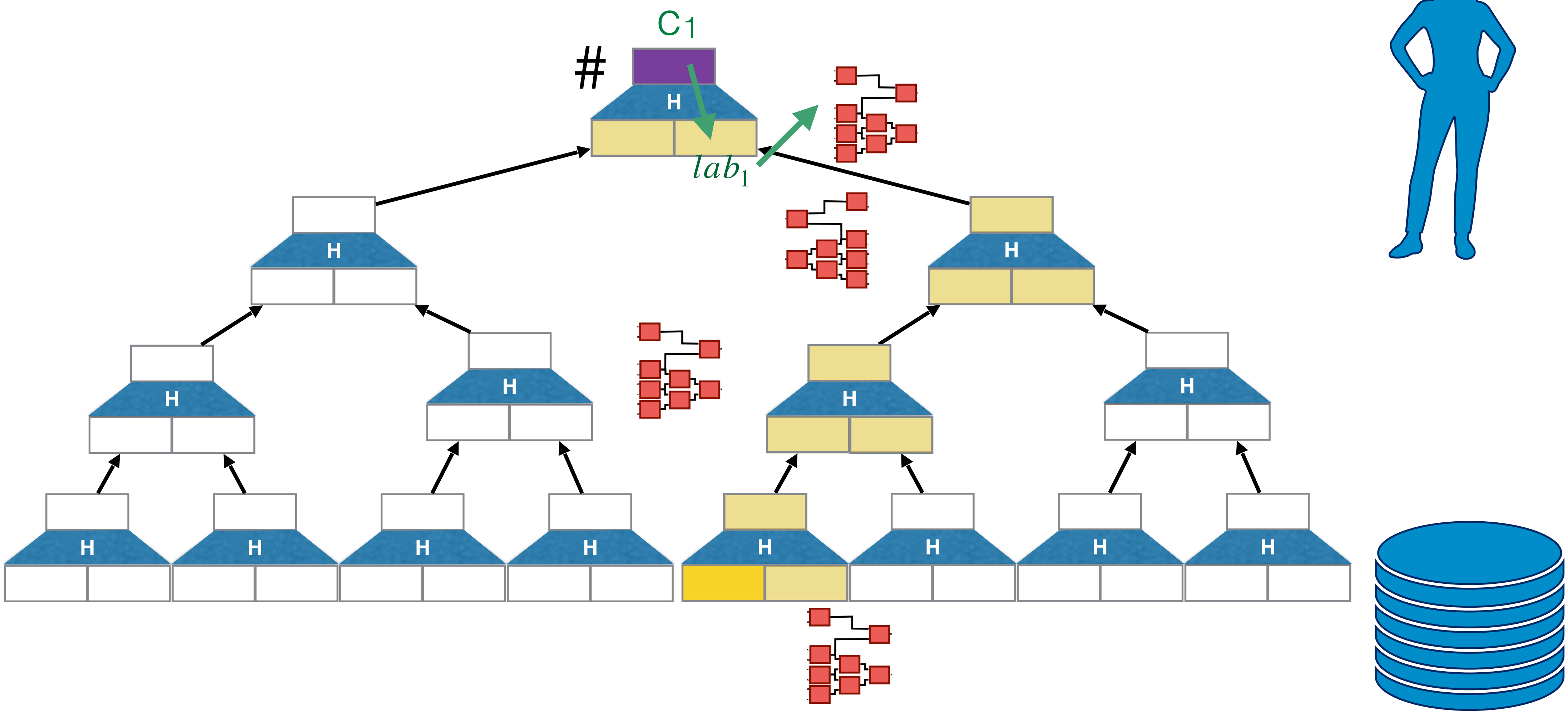
Laconic OT: Decryption



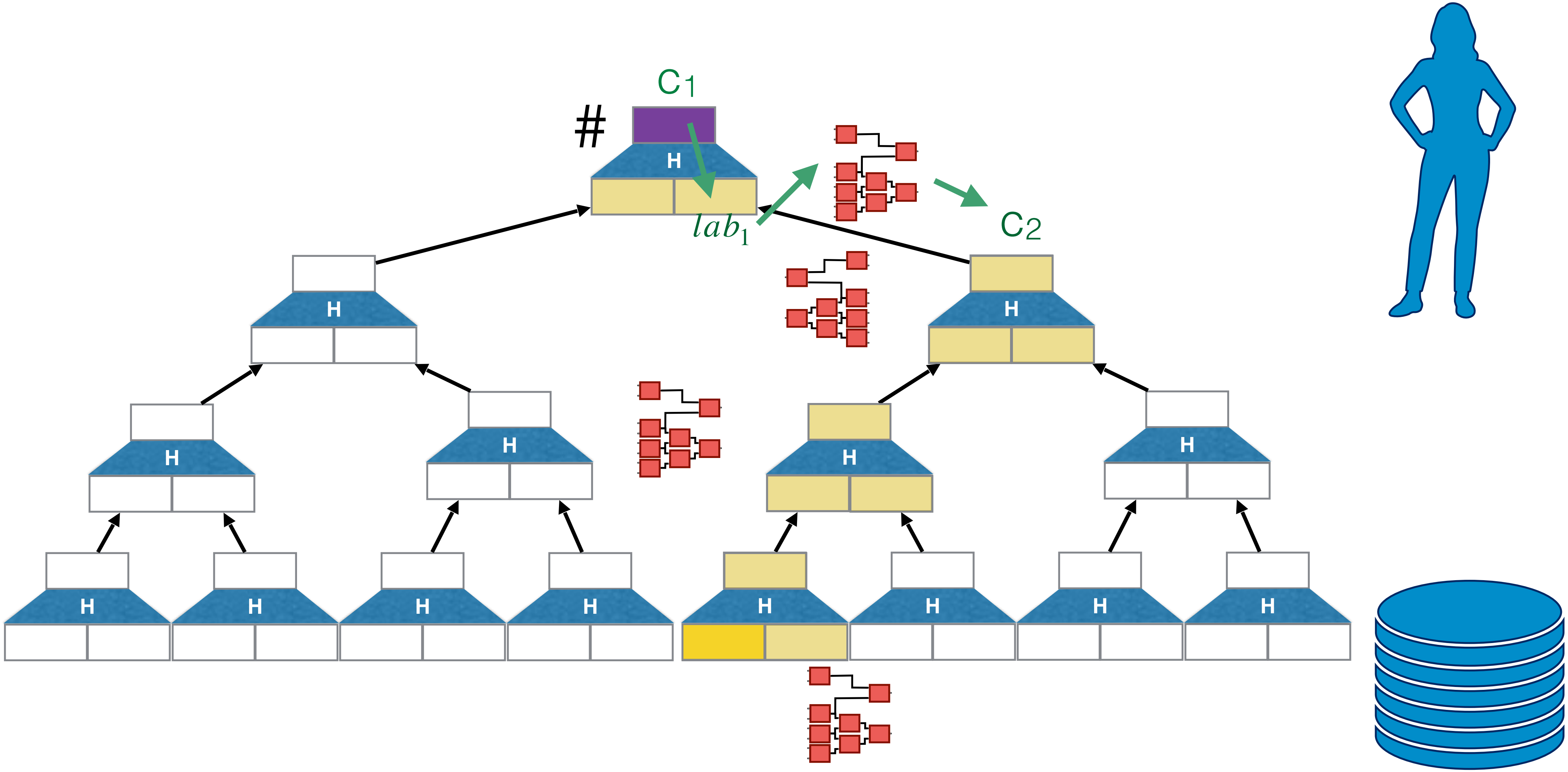
Laconic OT: Decryption



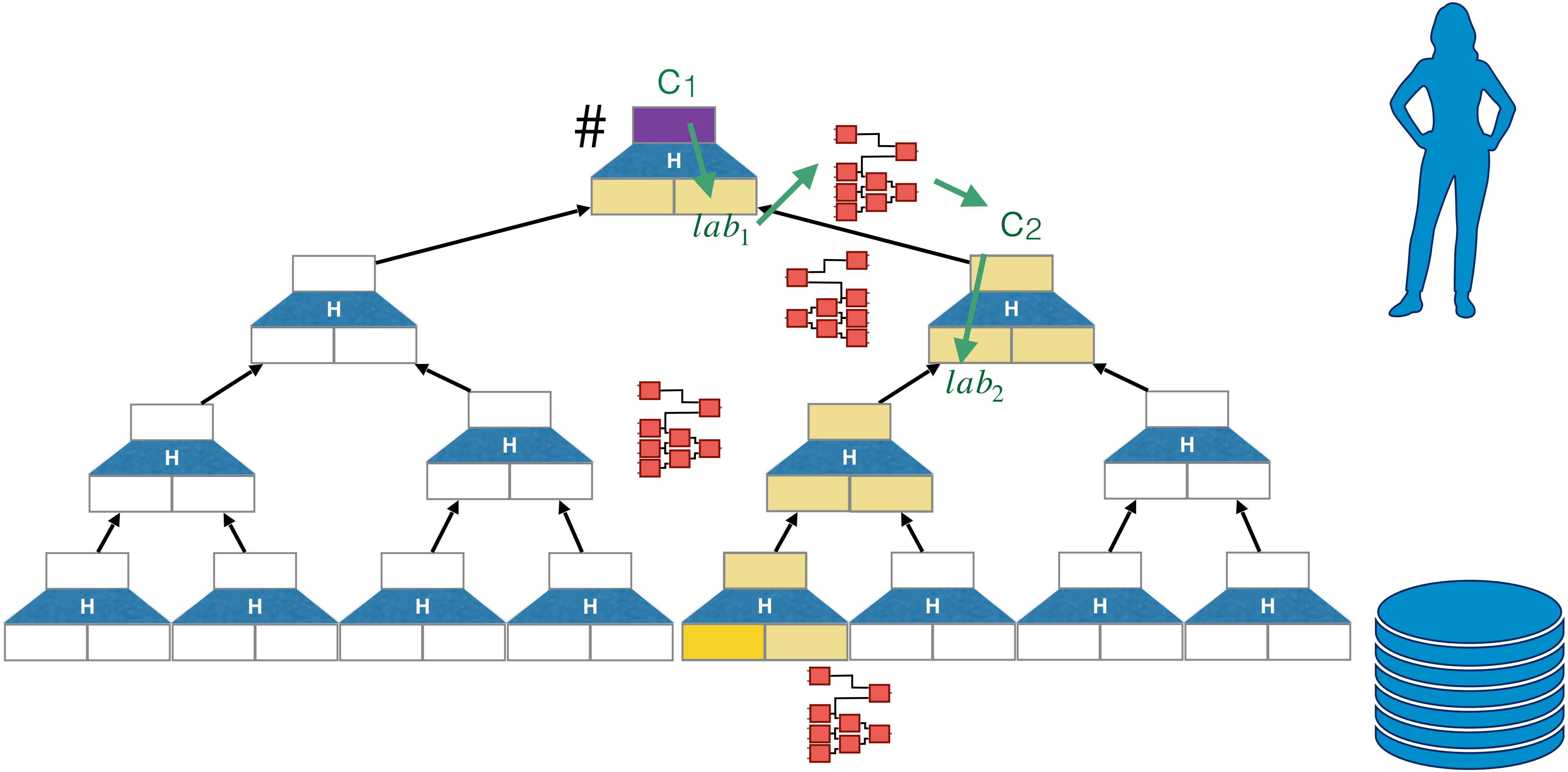
Laconic OT: Decryption



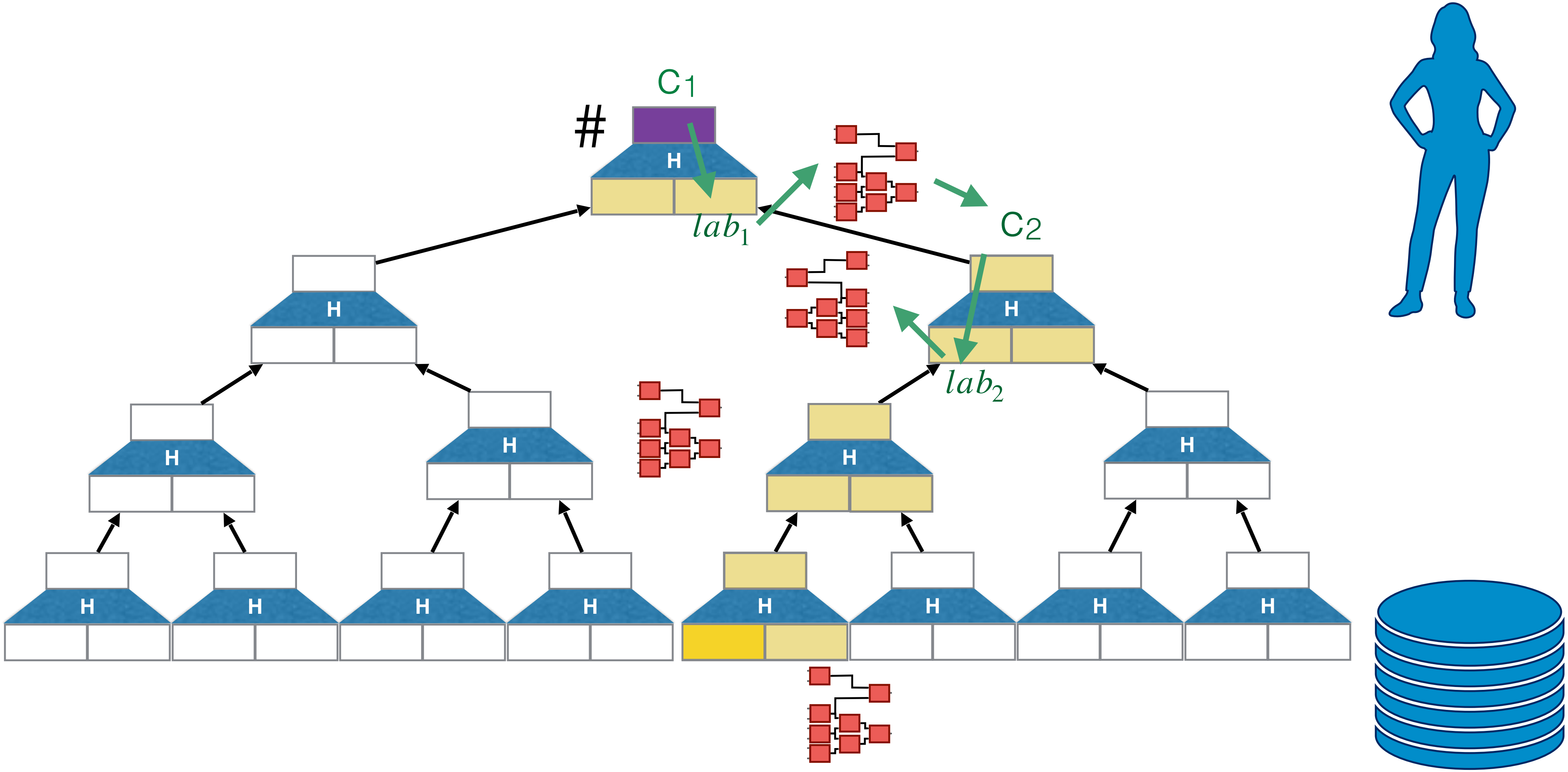
Laconic OT: Decryption



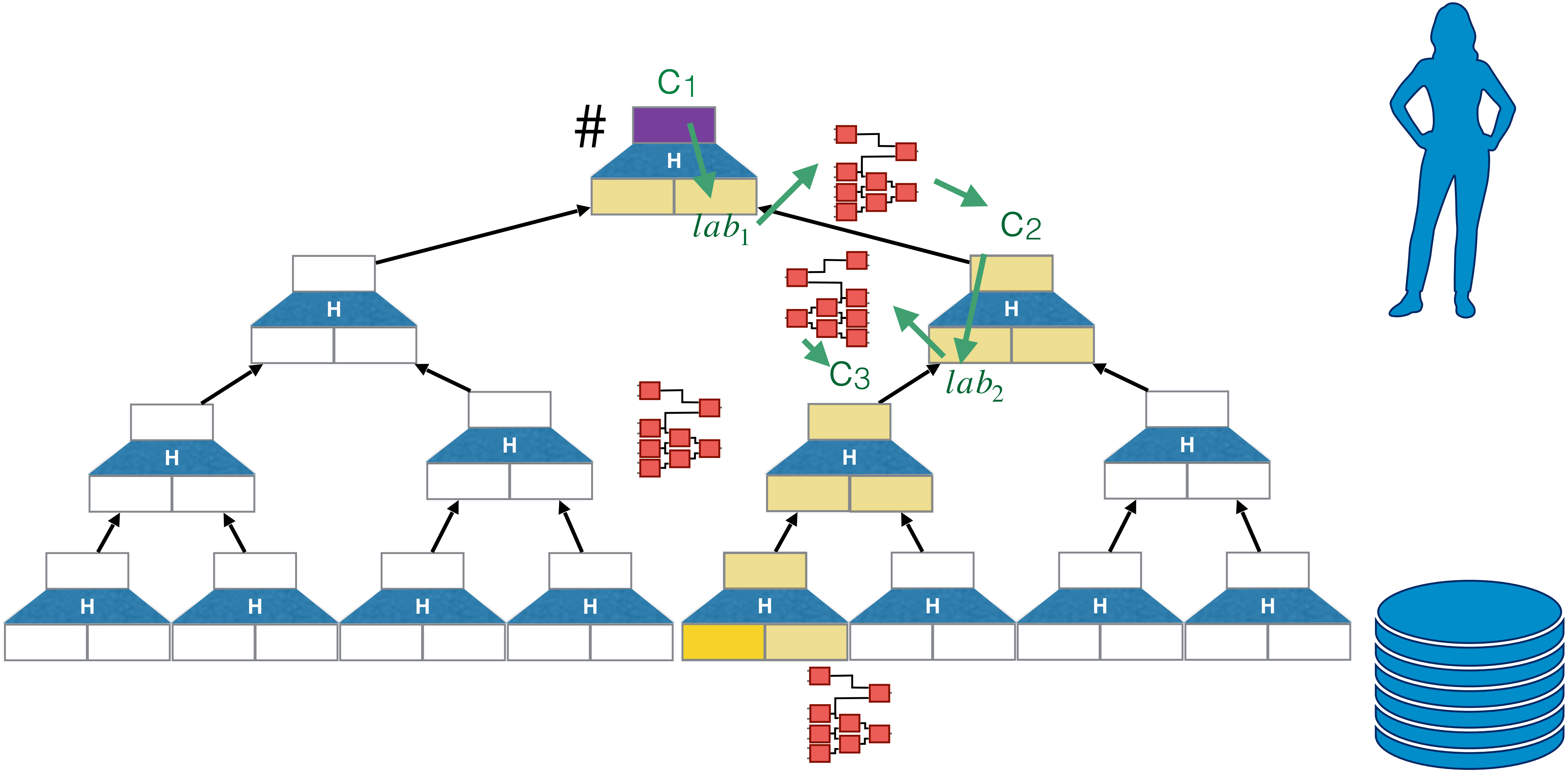
Laconic OT: Decryption



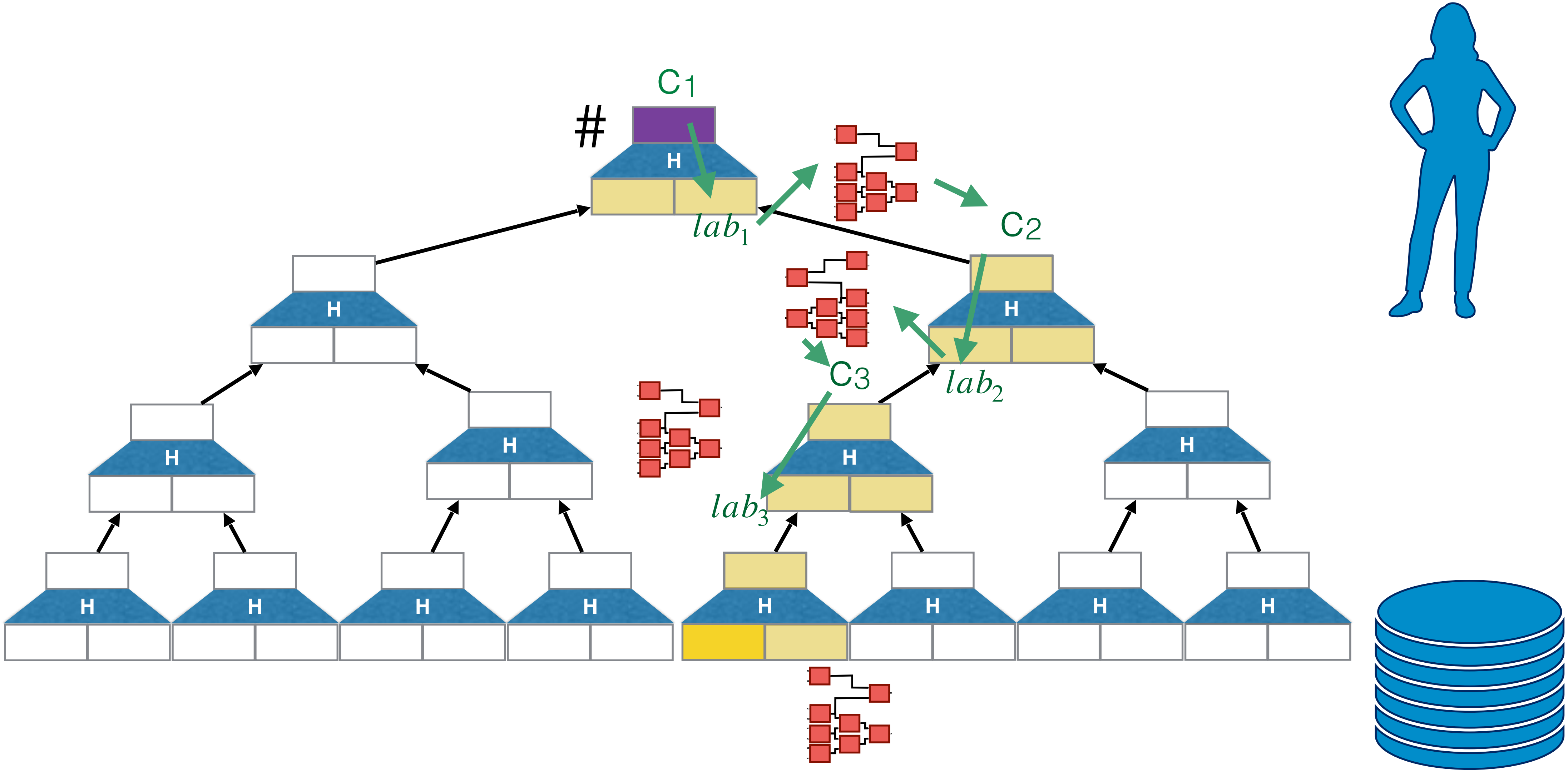
Laconic OT: Decryption



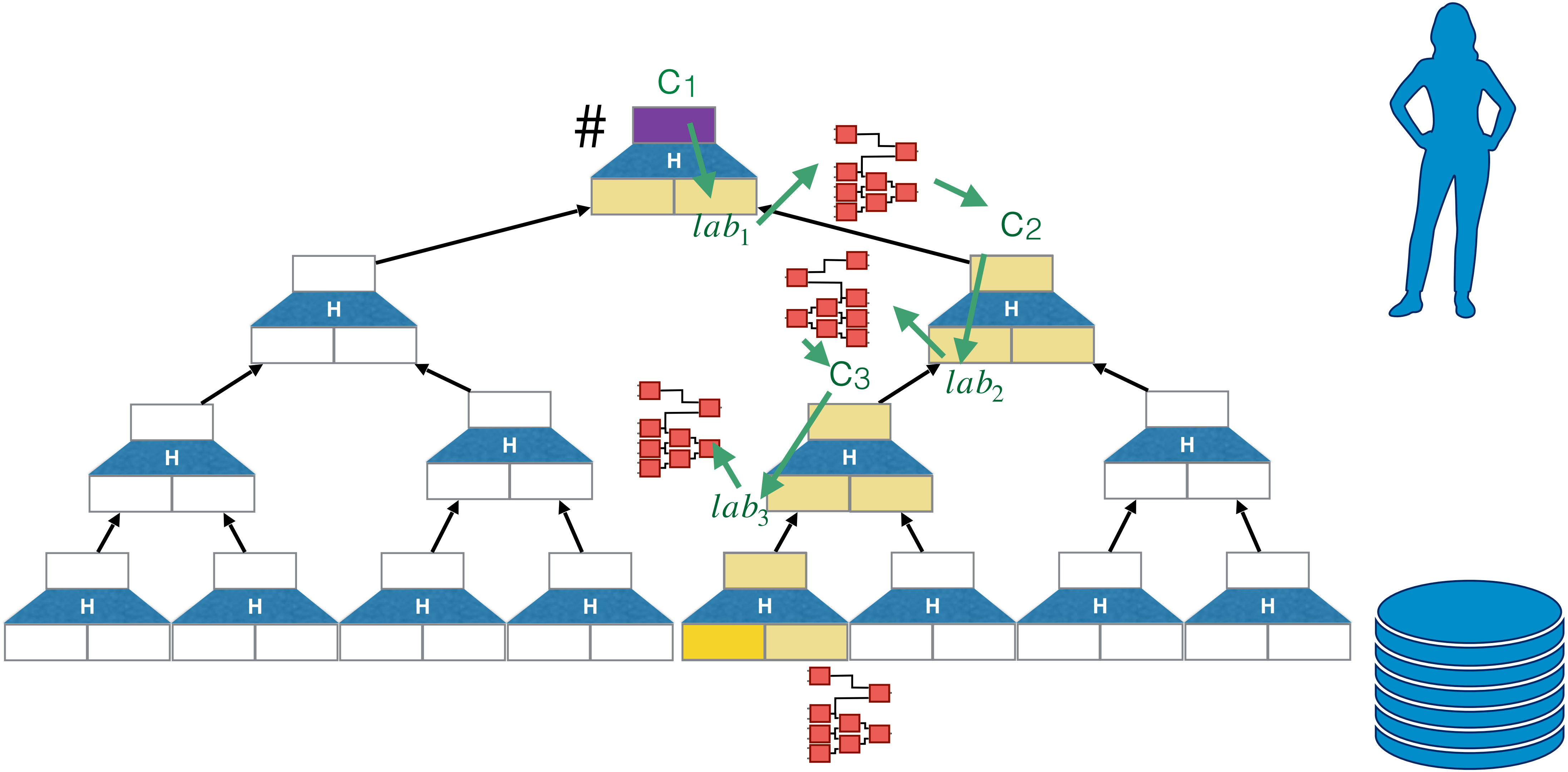
Laconic OT: Decryption



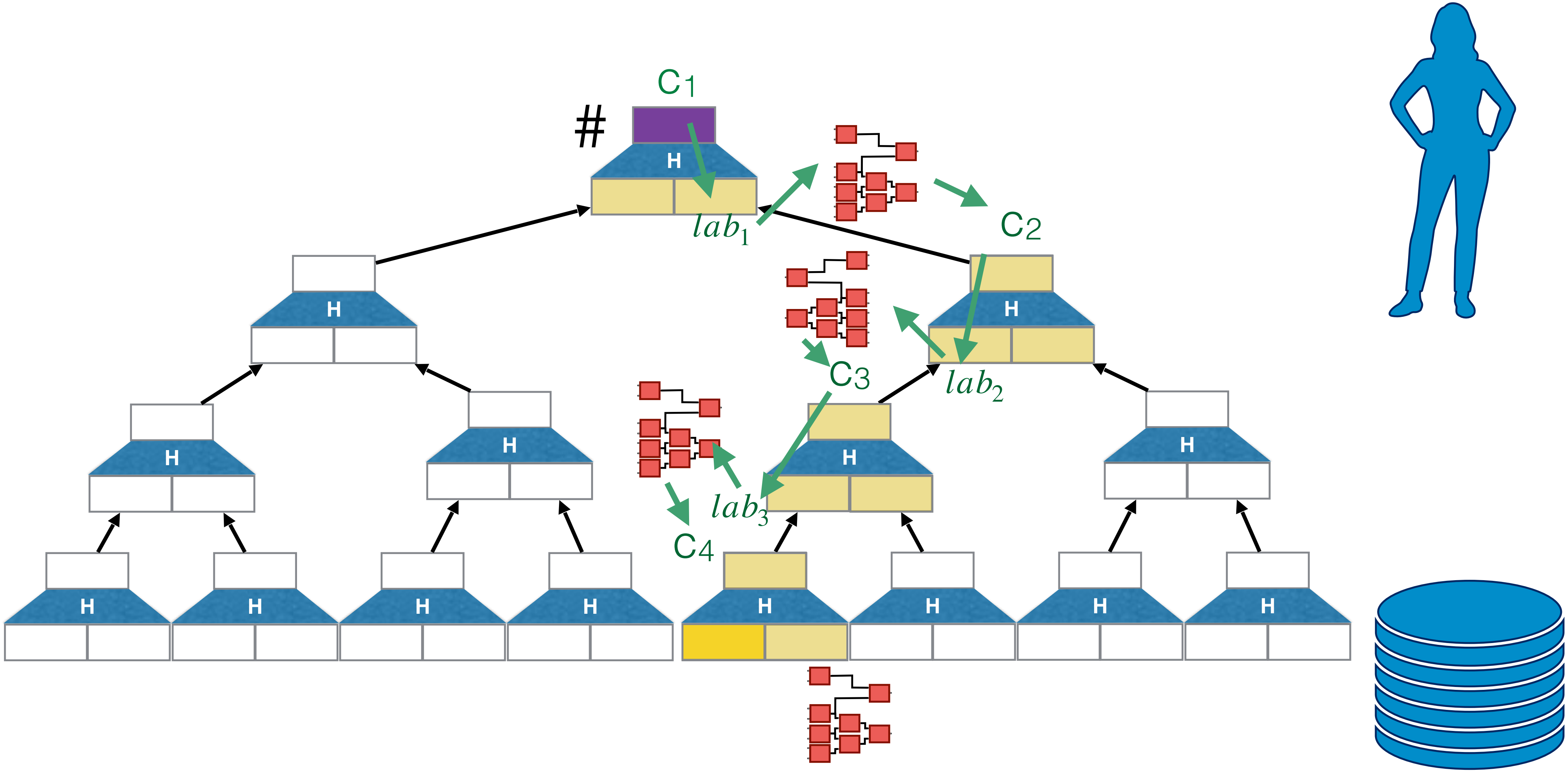
Laconic OT: Decryption



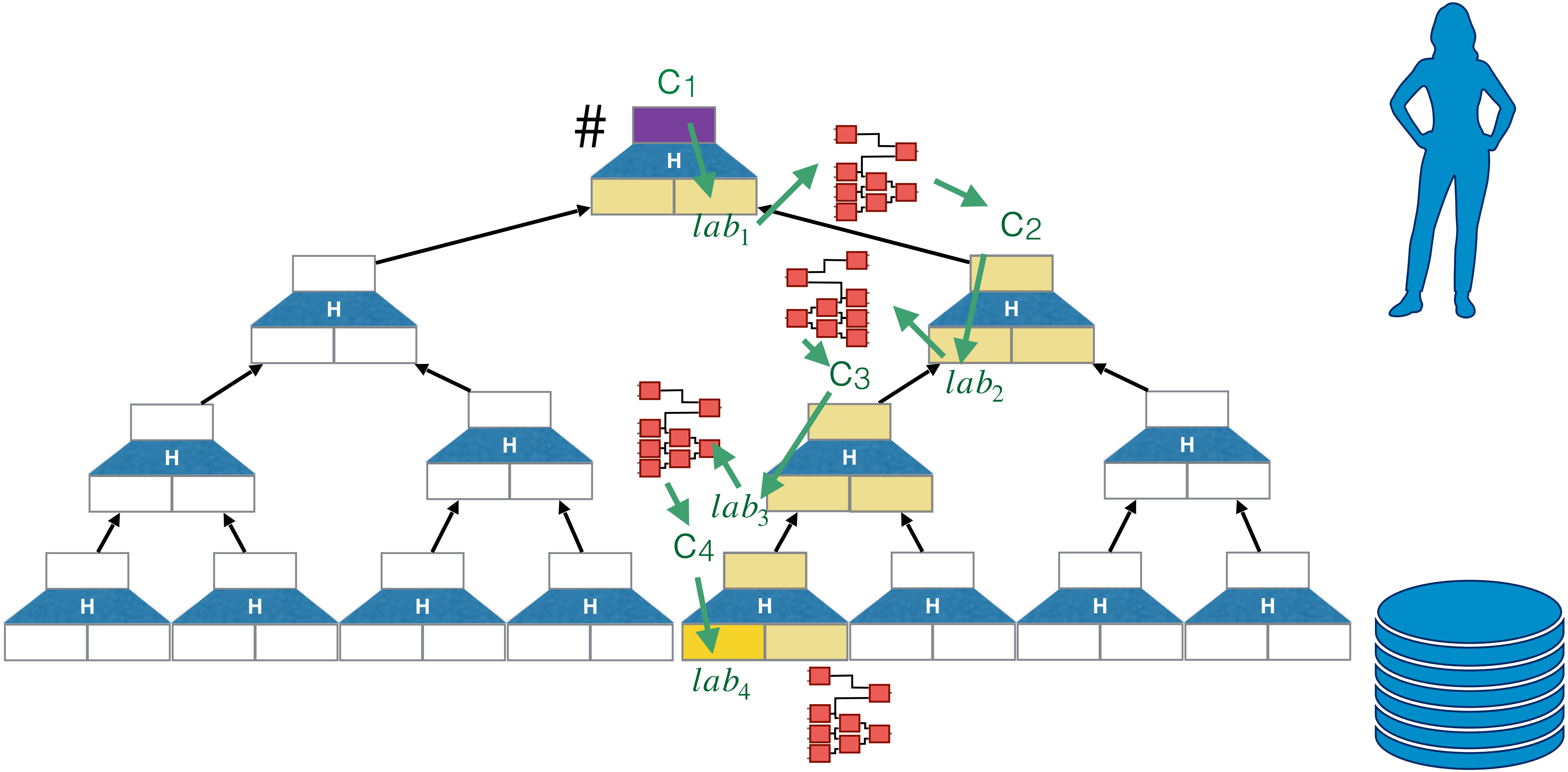
Laconic OT: Decryption



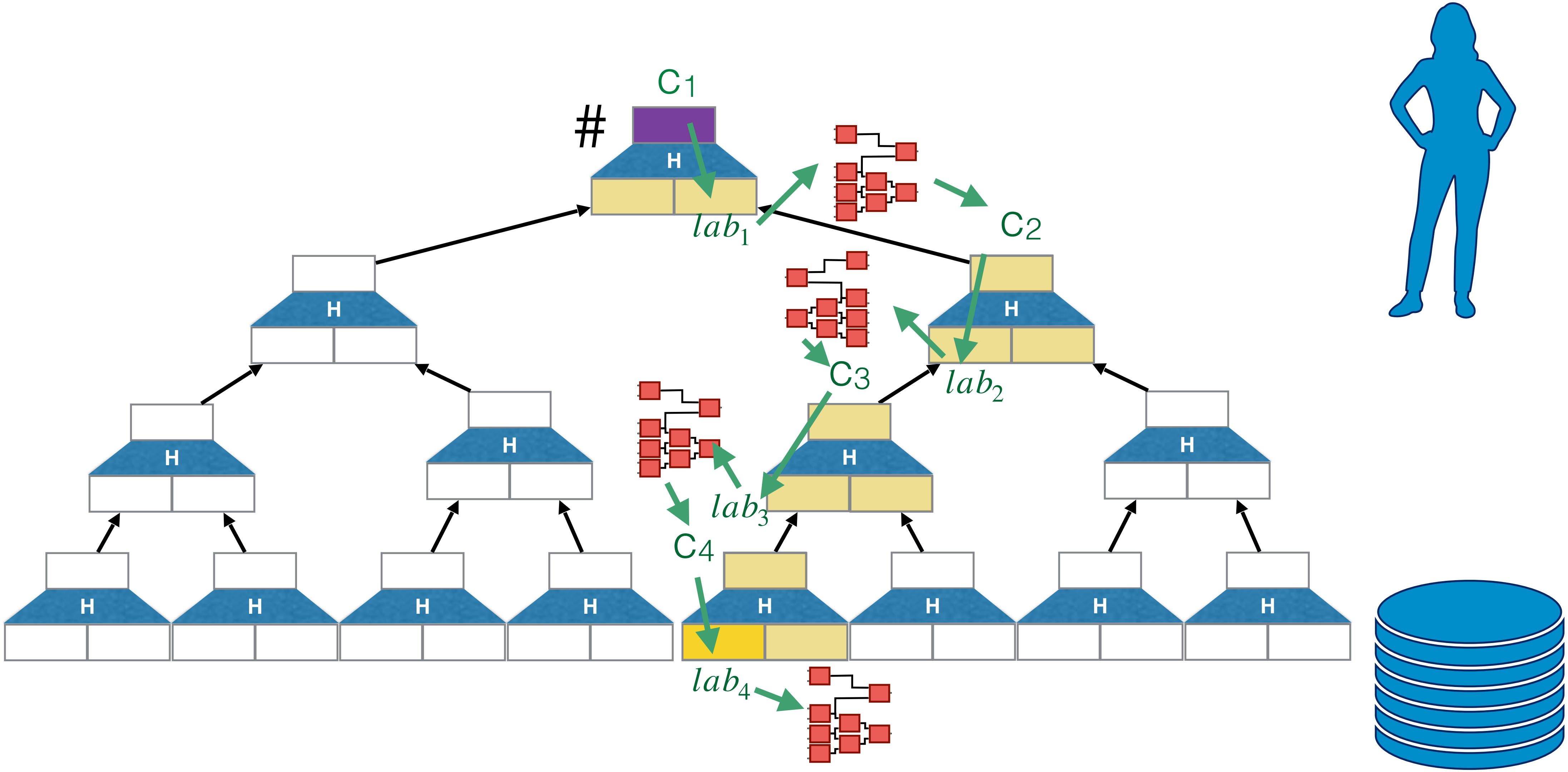
Laconic OT: Decryption



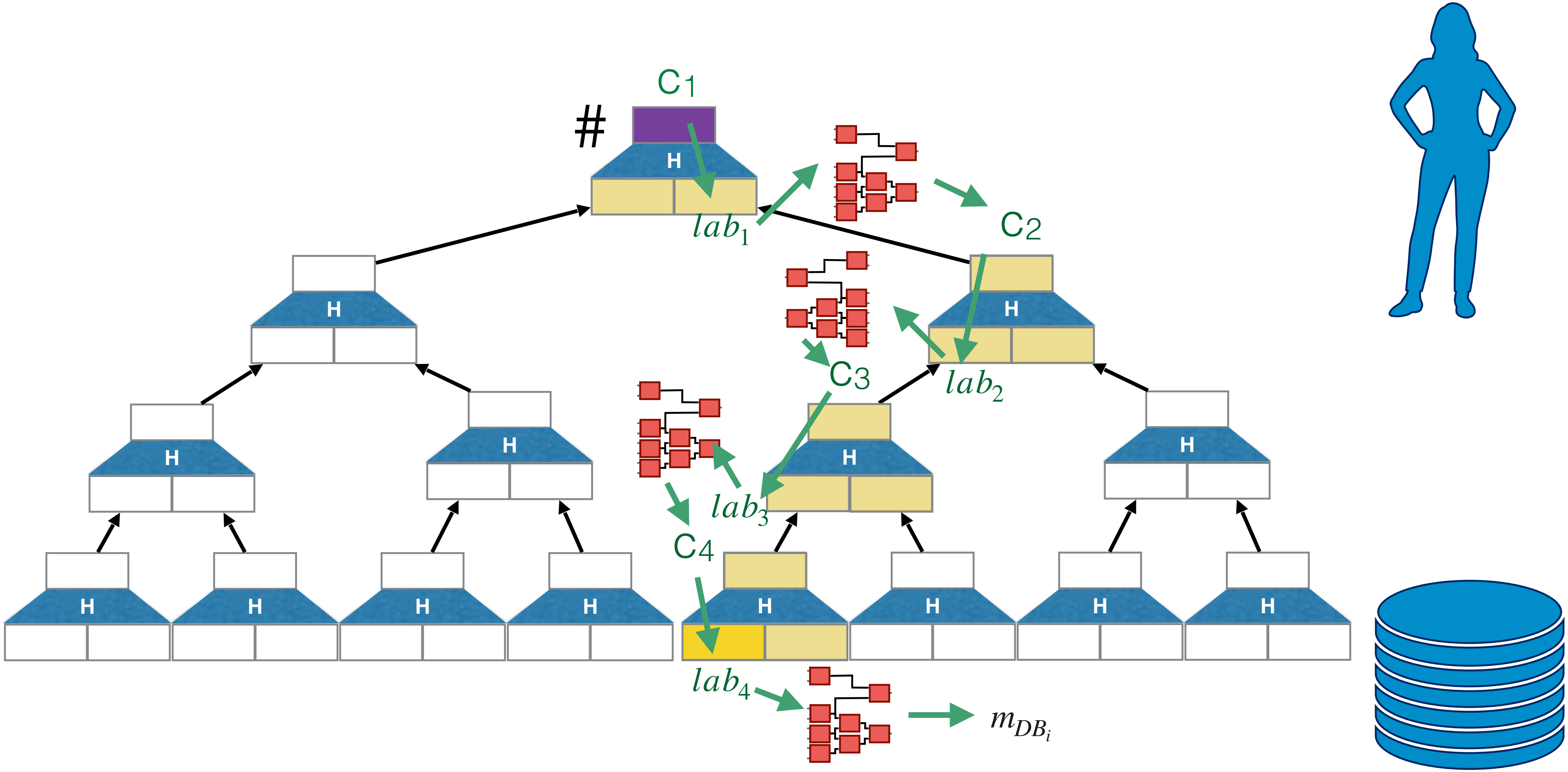
Laconic OT: Decryption



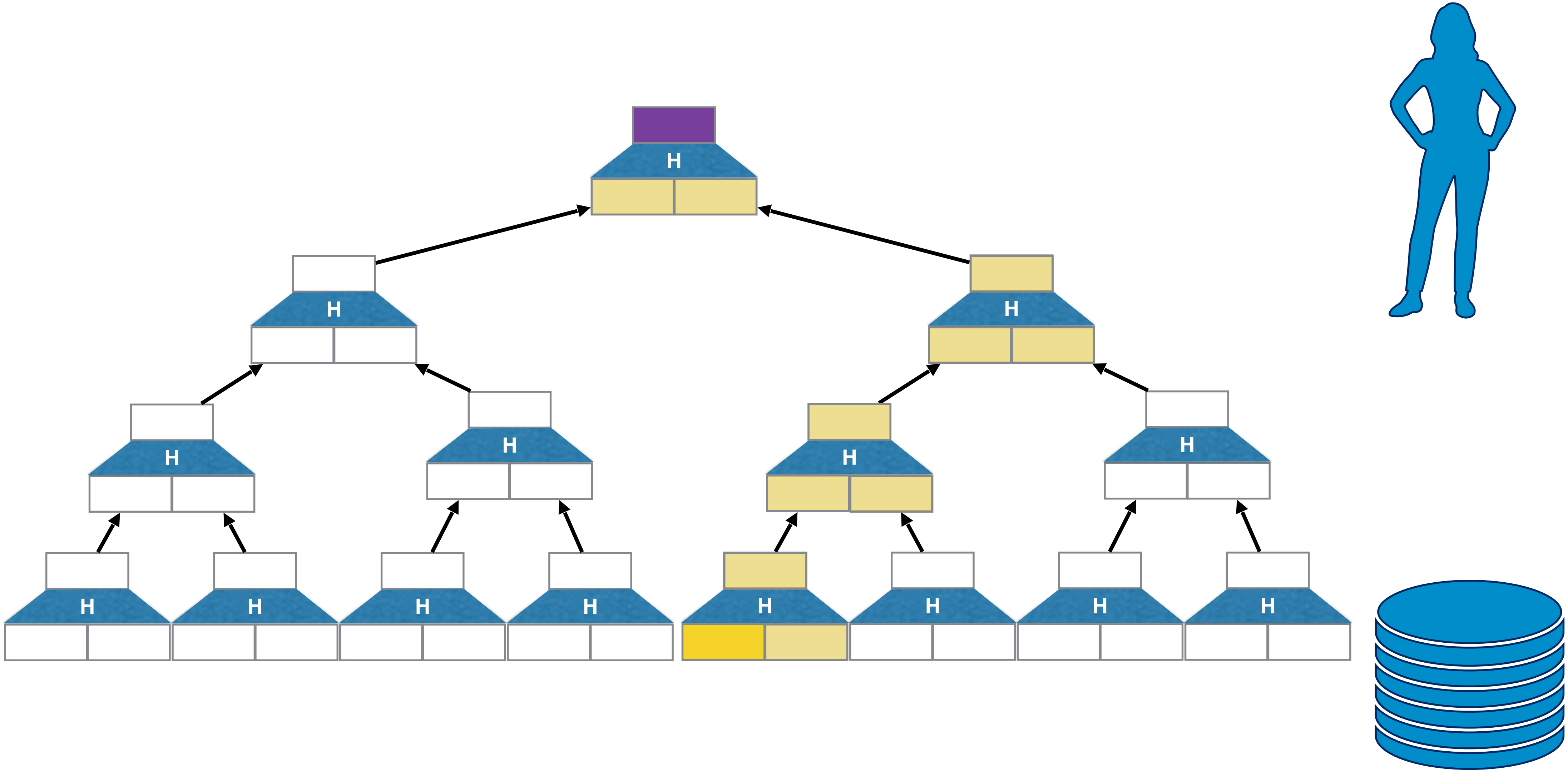
Laconic OT: Decryption



Laconic OT: Decryption

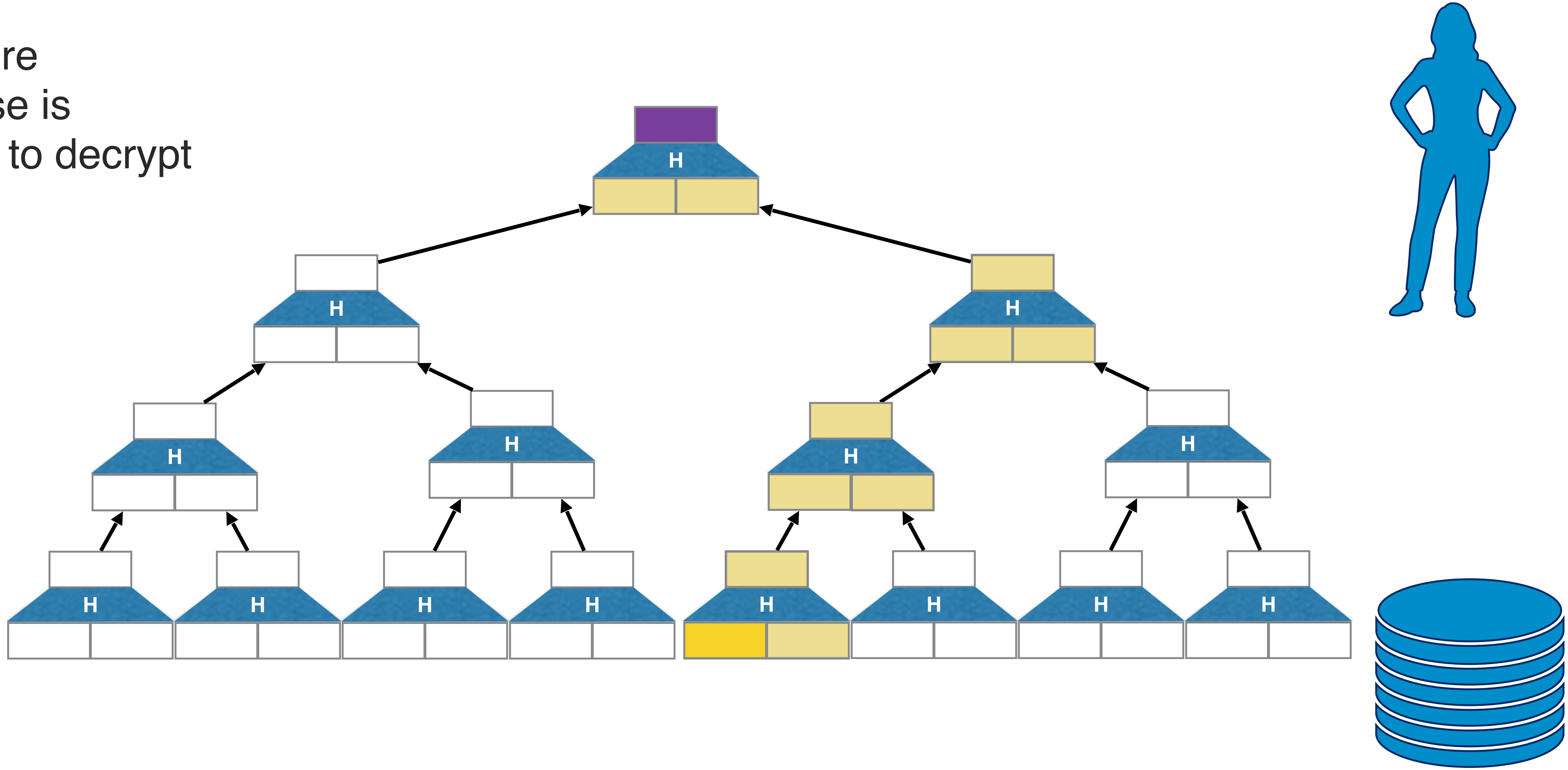


Laconic OT: Decryption



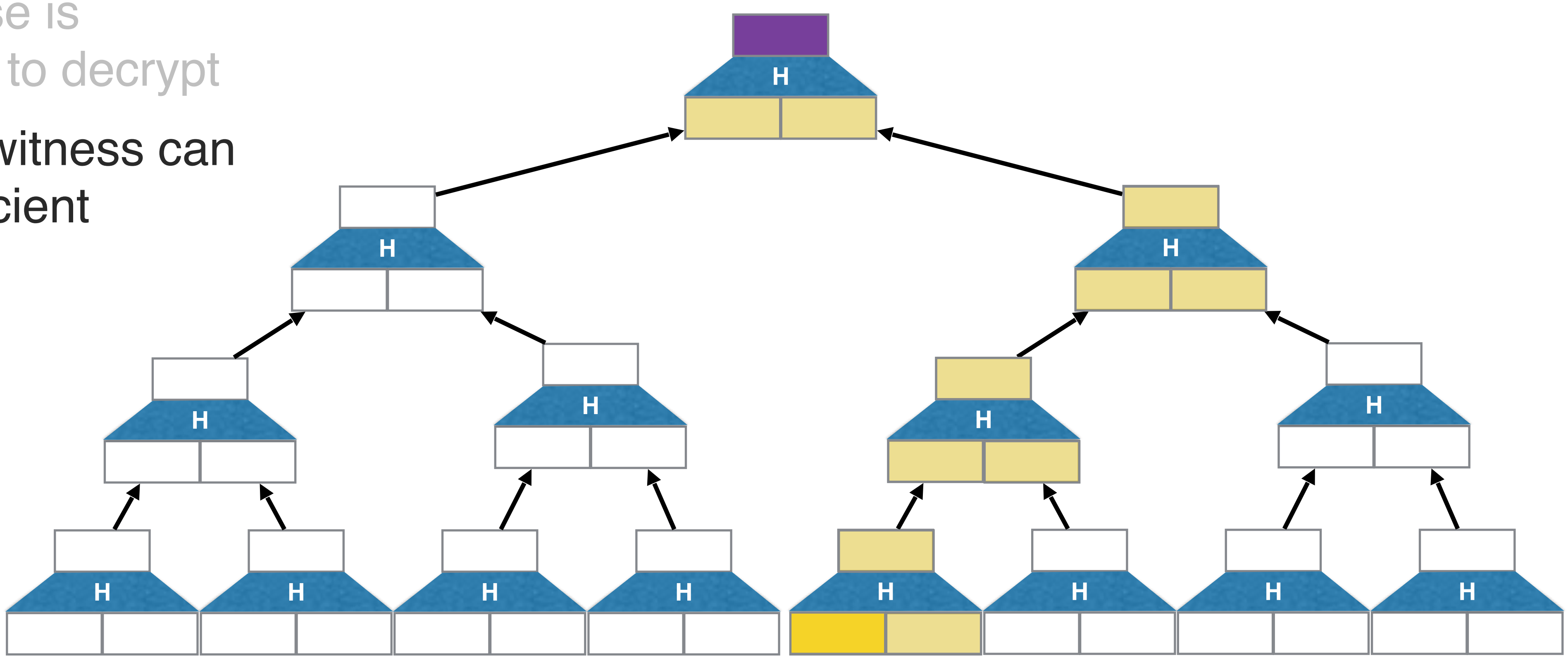
Laconic OT: Decryption

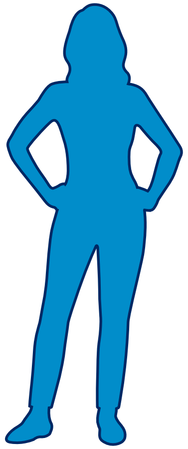
- Not entire database is needed to decrypt

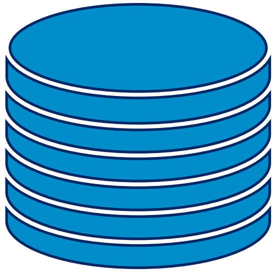


Laconic OT: Decryption

- Not entire database is needed to decrypt
- Partial witness can be sufficient





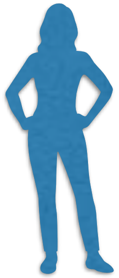


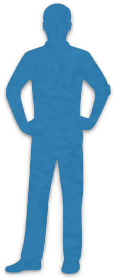
Advanced Constructions

Identity-Based Encryption [Sha84]

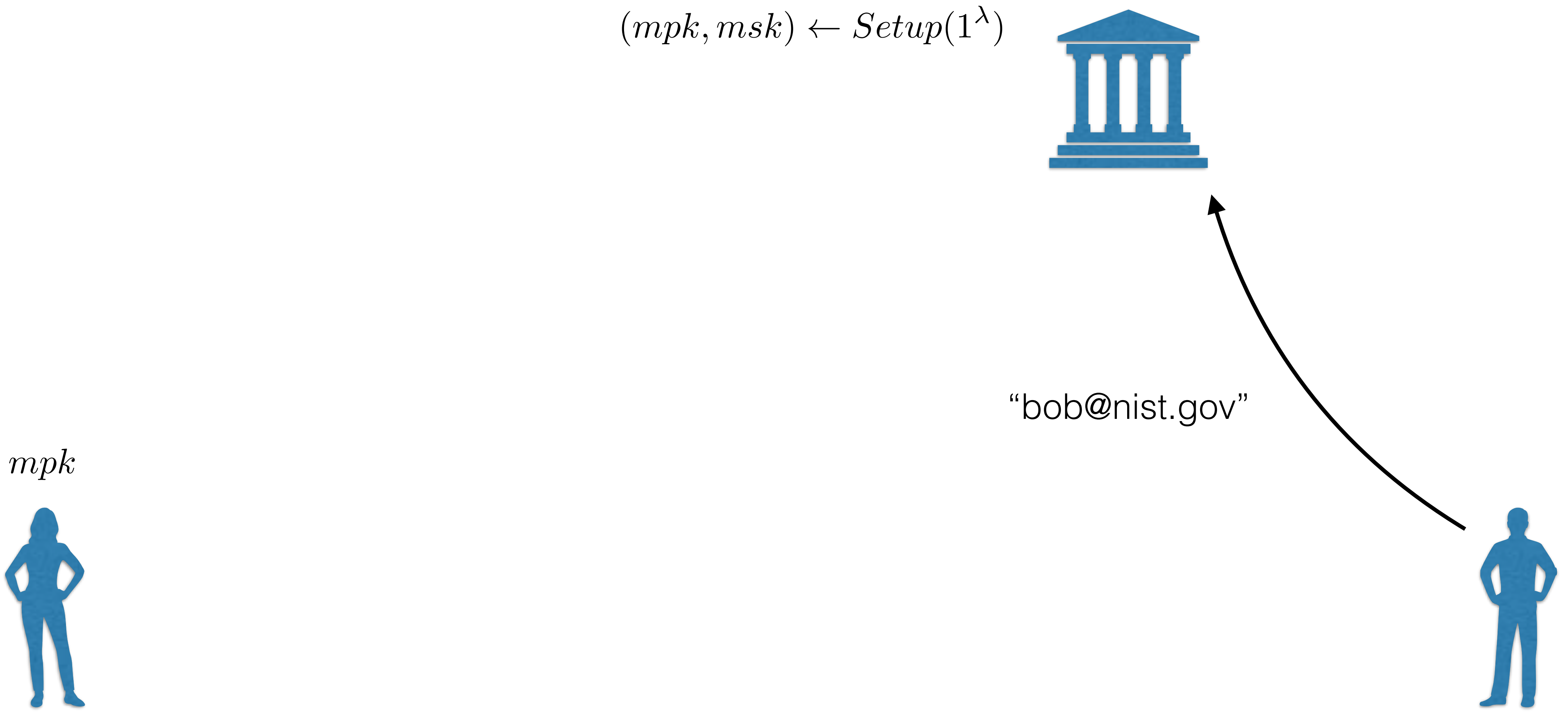
$(mpk, msk) \leftarrow Setup(1^\lambda)$

mpk

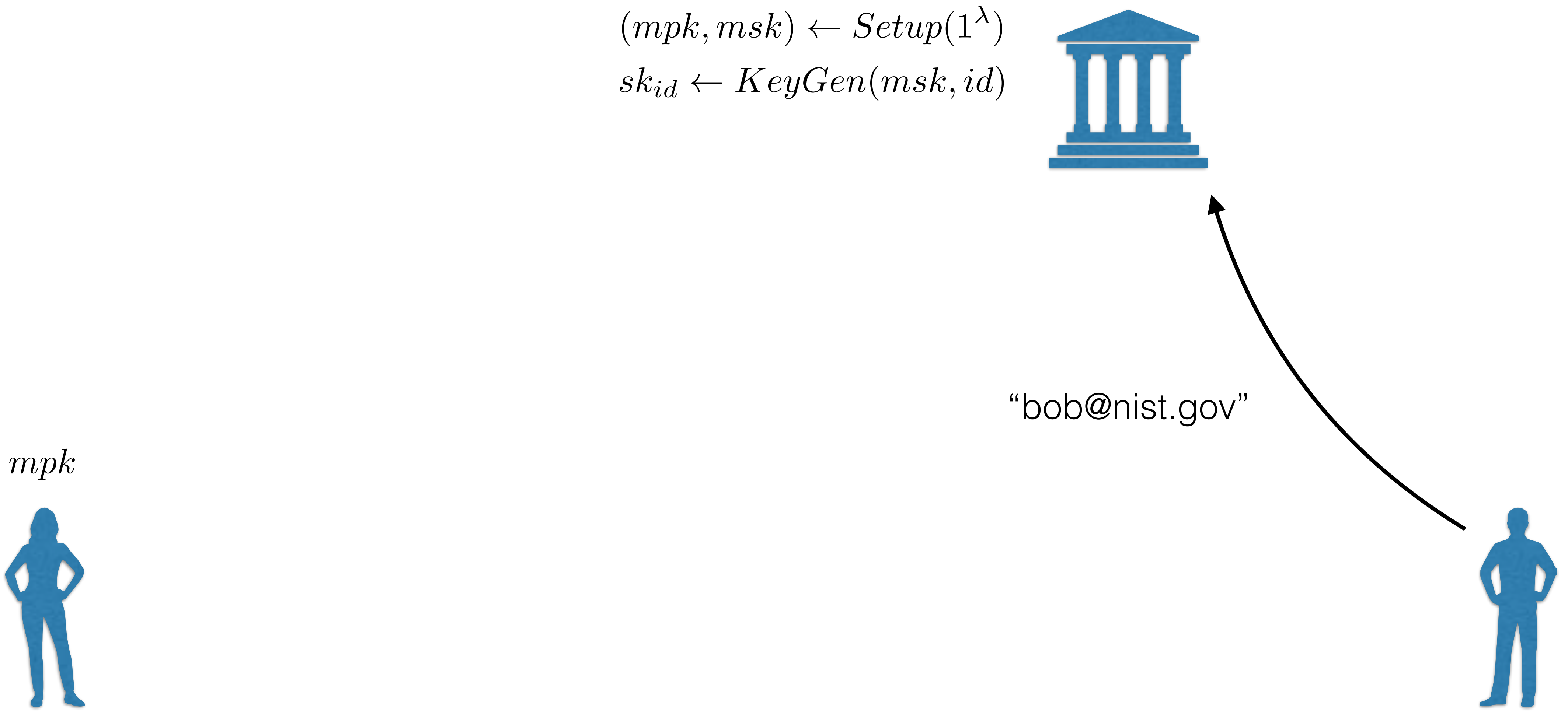




Identity-Based Encryption [Sha84]



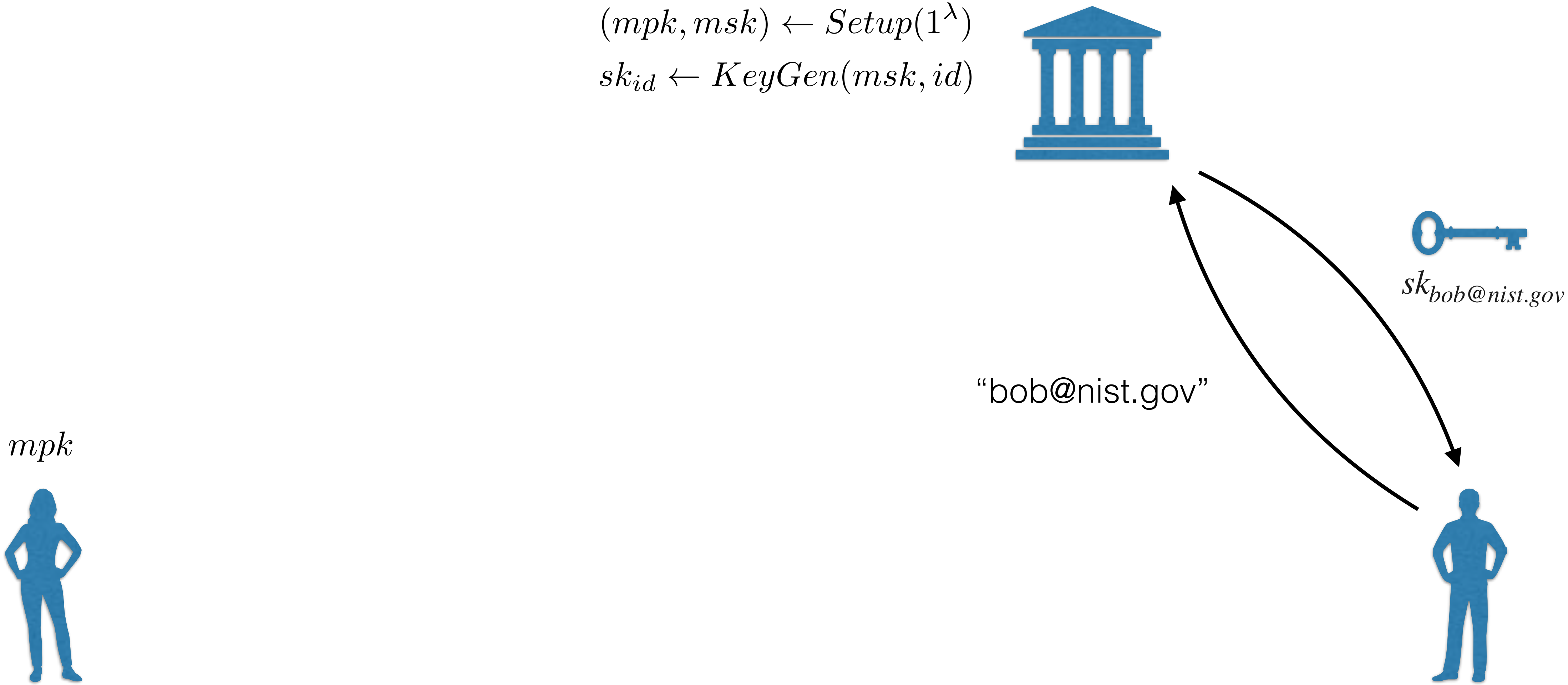
Identity-Based Encryption [Sha84]



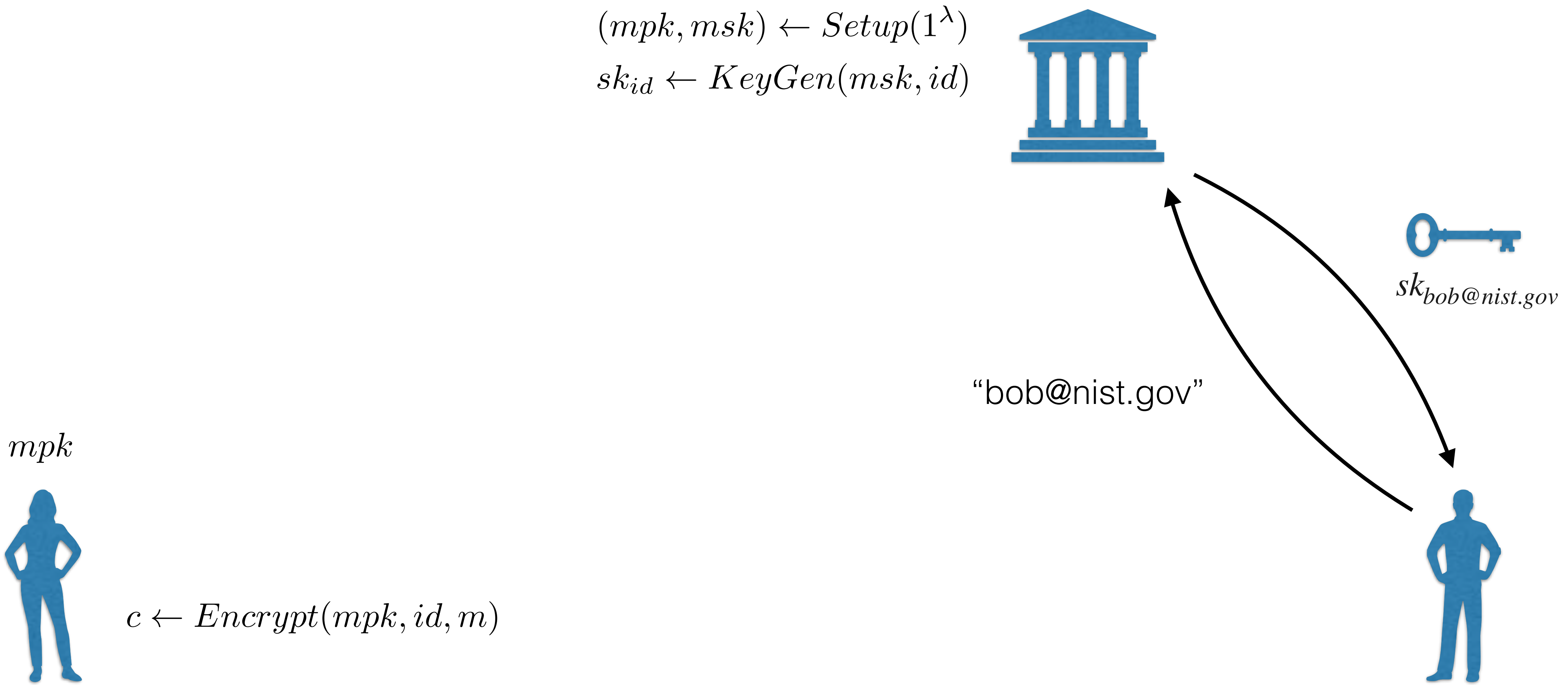
Identity-Based Encryption [Sha84]

$$(mpk, msk) \leftarrow Setup(1^\lambda)$$

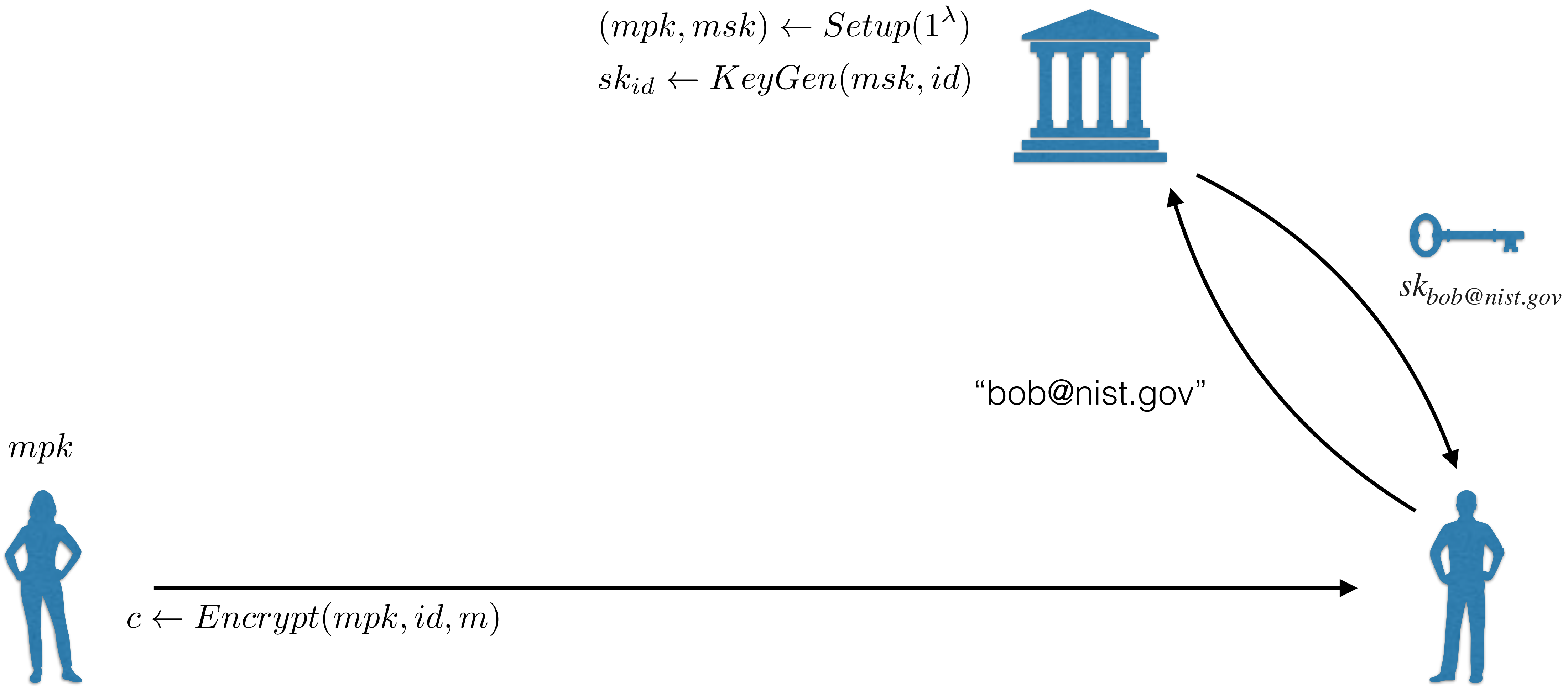
$$sk_{id} \leftarrow KeyGen(msk, id)$$



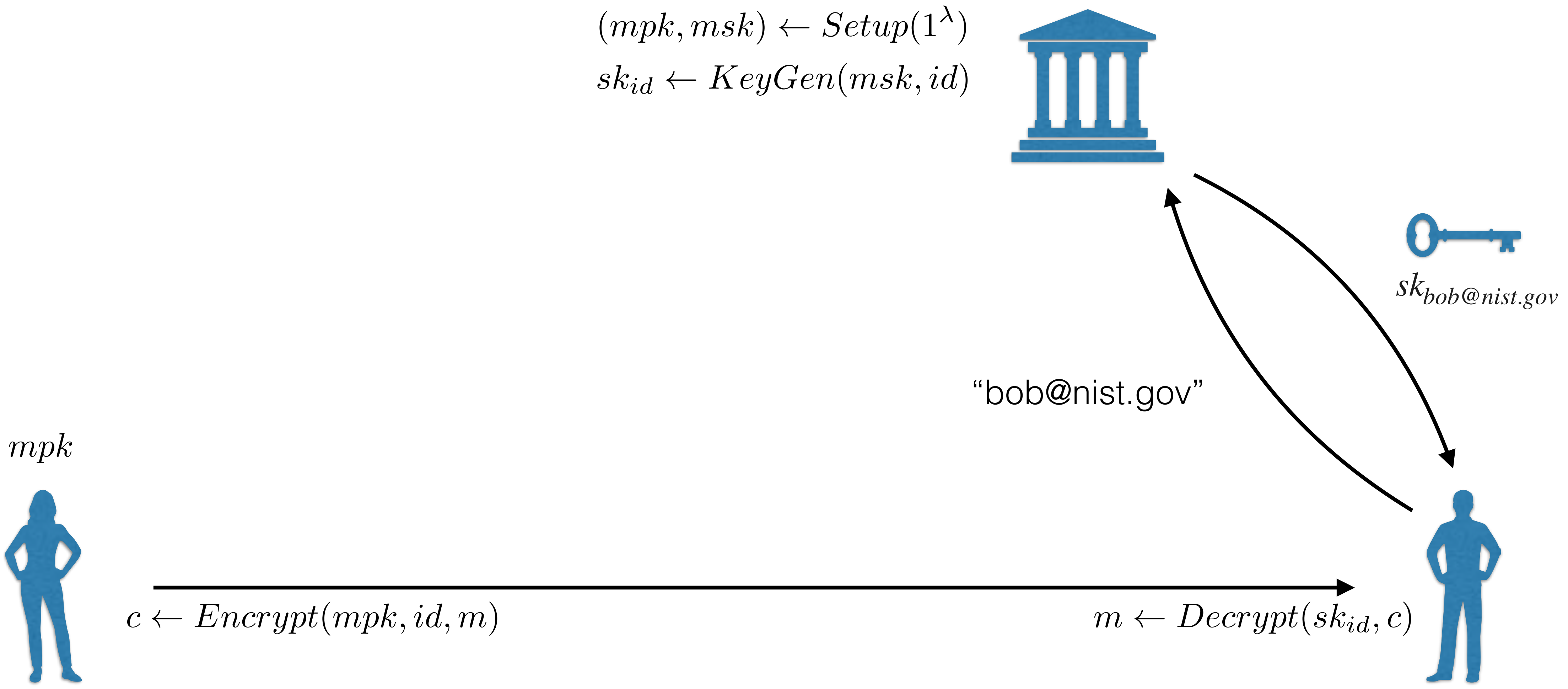
Identity-Based Encryption [Sha84]



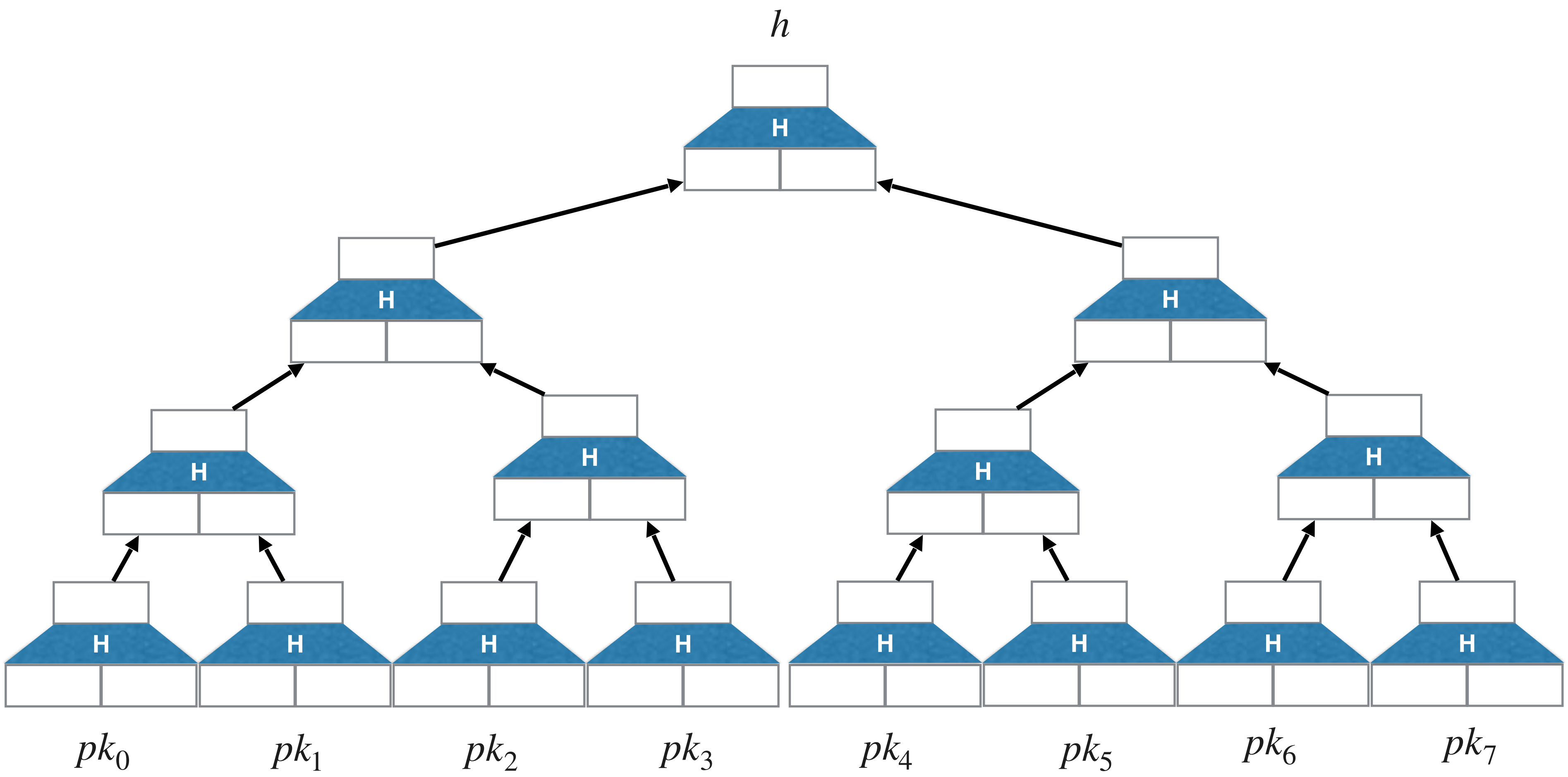
Identity-Based Encryption [Sha84]



Identity-Based Encryption [Sha84]

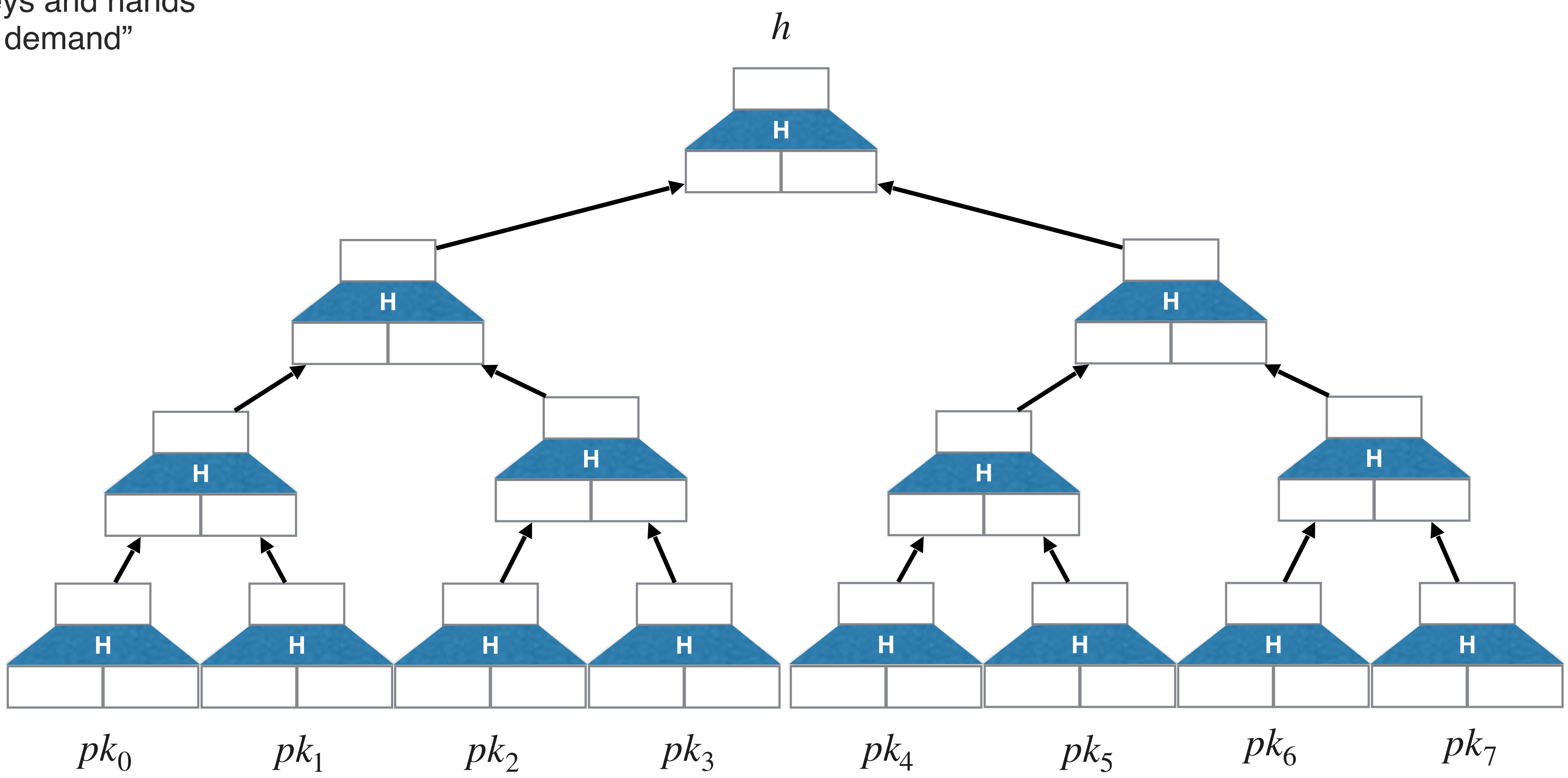


Identity-Based Encryption [DG17]: Setup



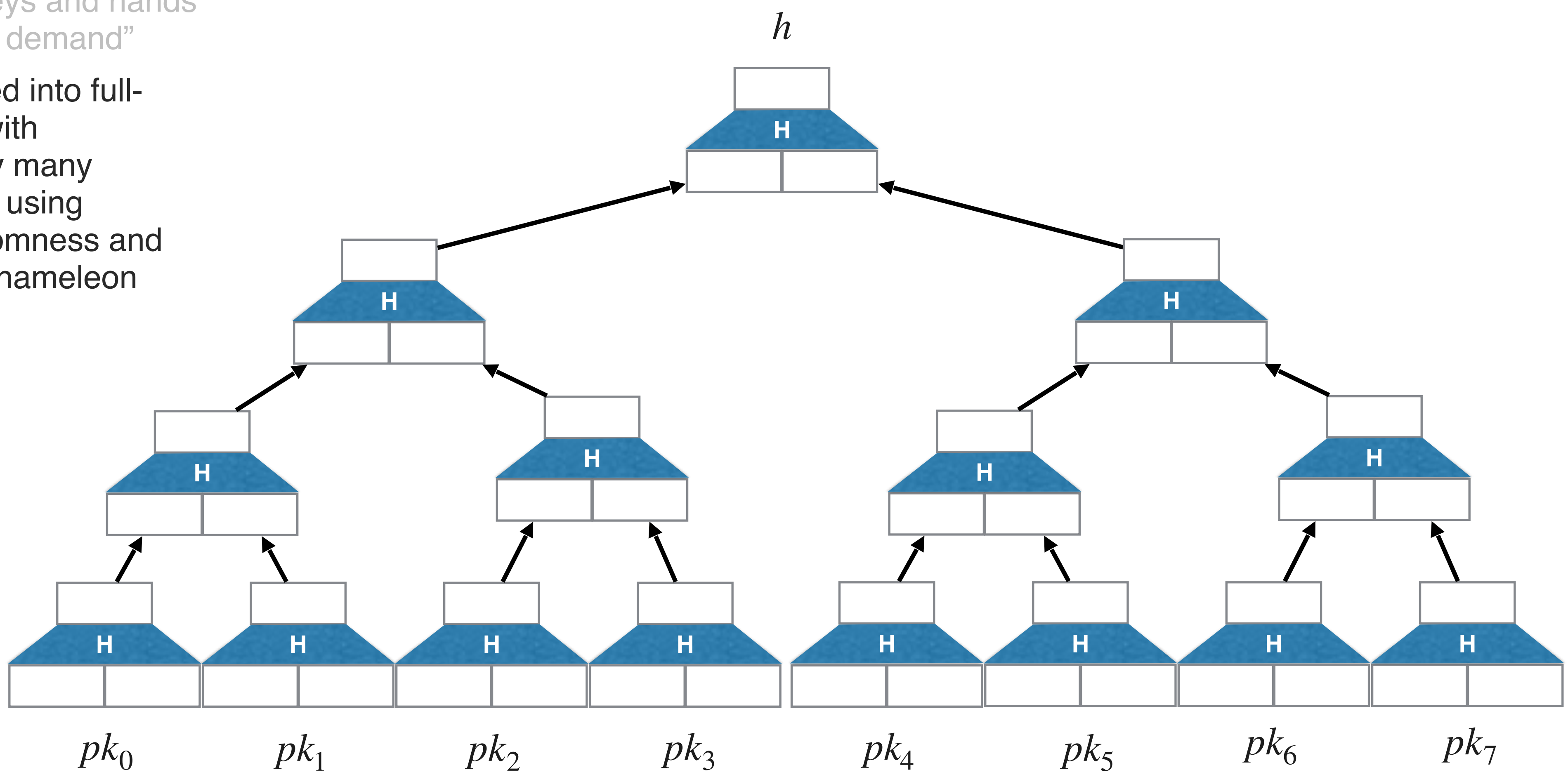
Identity-Based Encryption [DG17]: Setup

- Simpler Setting: Key generator has "pre-generated" a polynomial number of keys and hands them out "on demand"

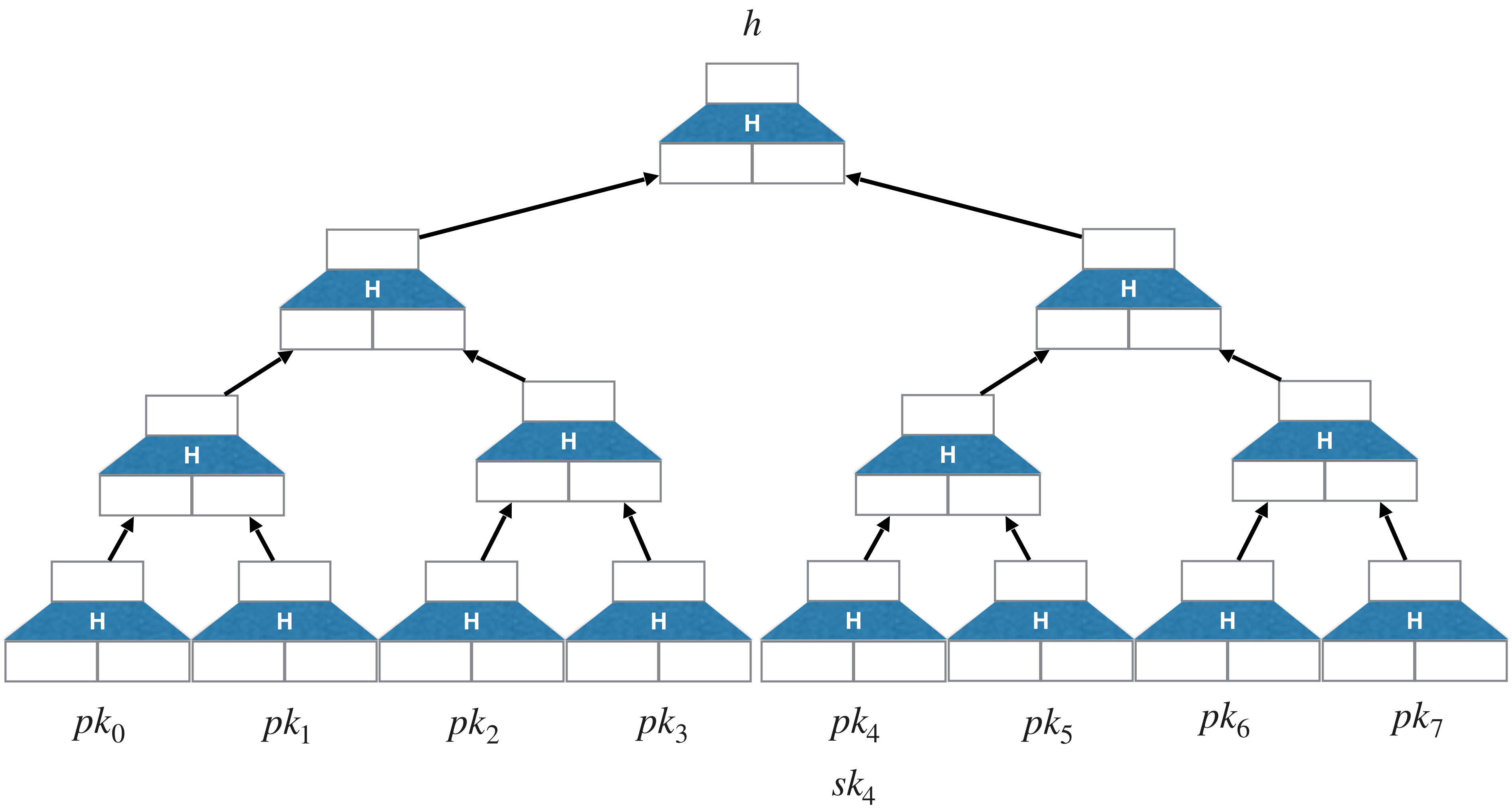


Identity-Based Encryption [DG17]: Setup

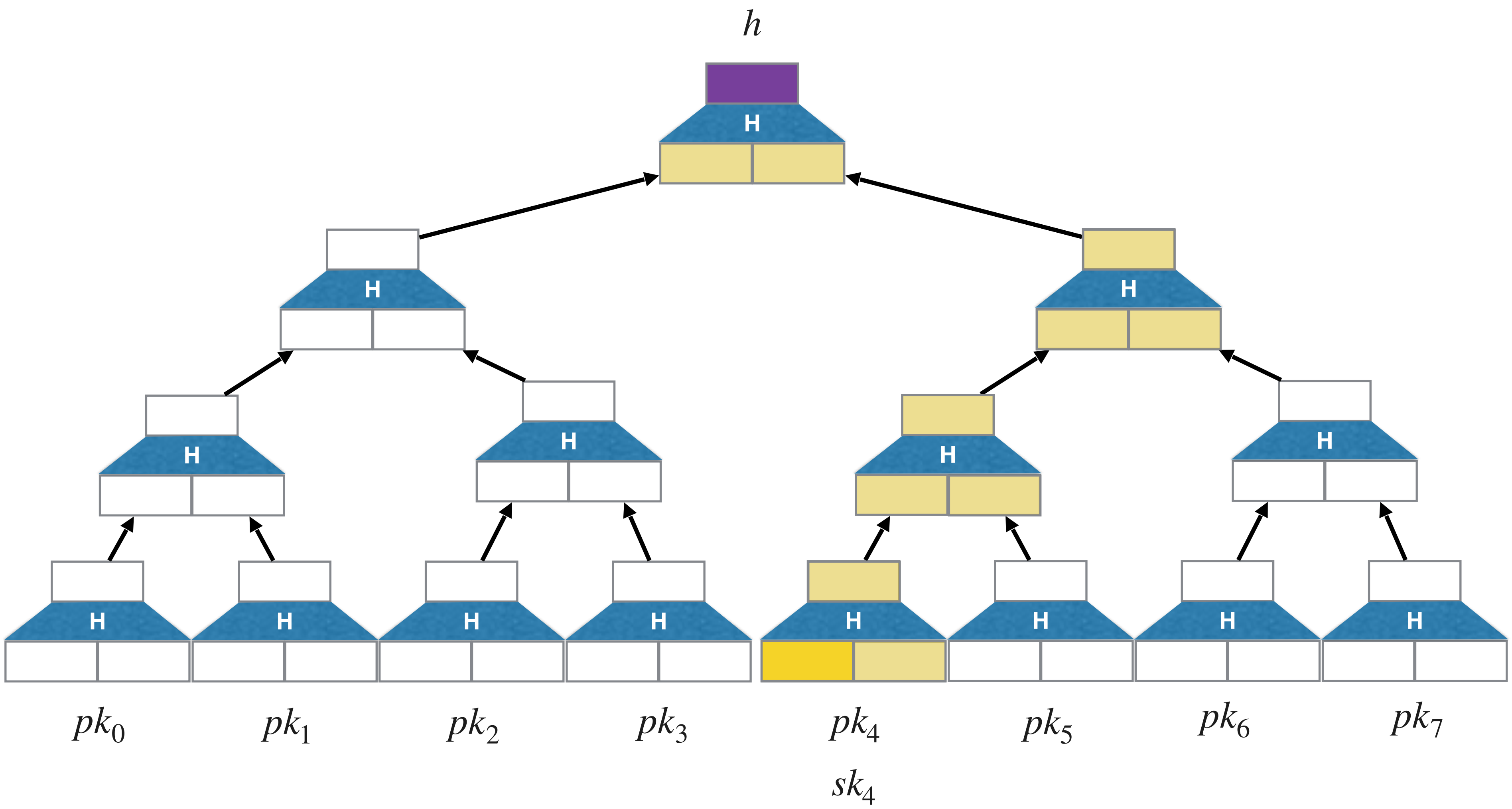
- Simpler Setting: Key generator has "pre-generated" a polynomial number of keys and hands them out "on demand"
- Can be turned into full-blown IBE (with exponentially many identities) by using pseudorandomness and trapdoors (Chameleon encryption)



Identity-Based Encryption [DG17]: KeyGen

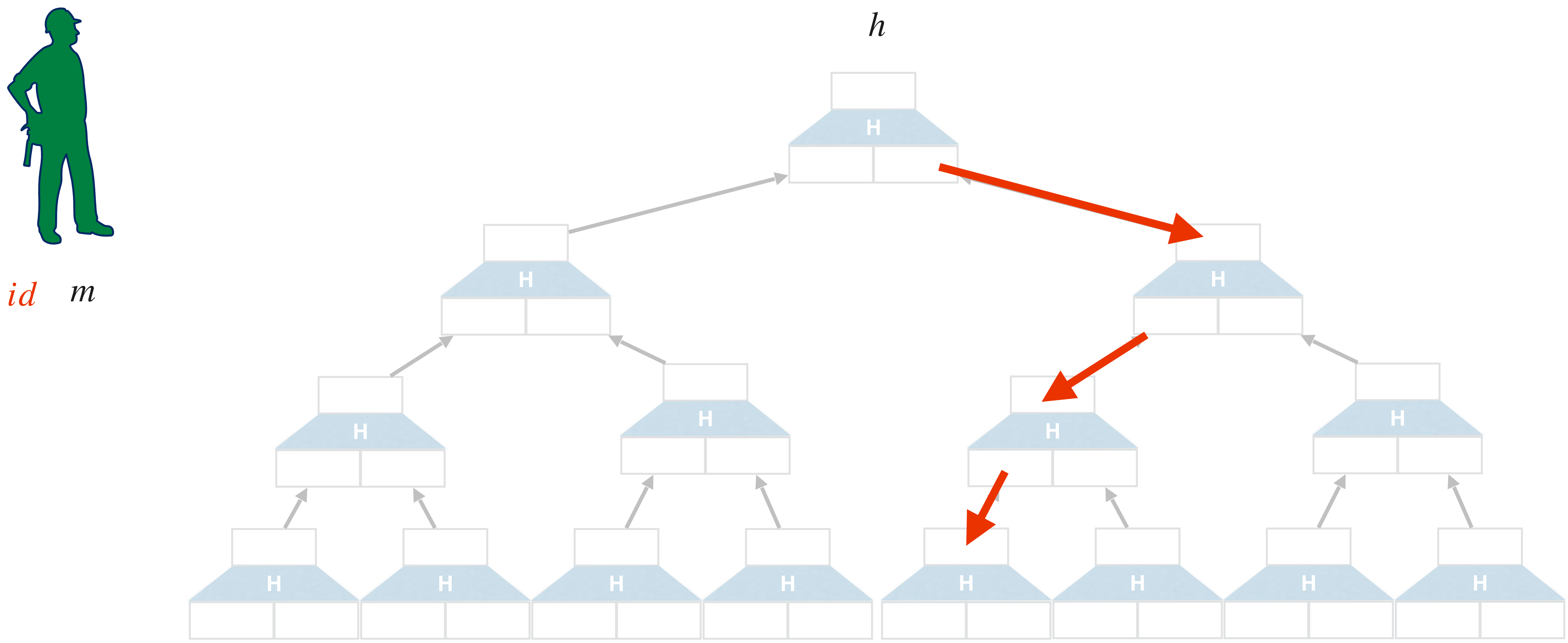


Identity-Based Encryption [DG17]: KeyGen



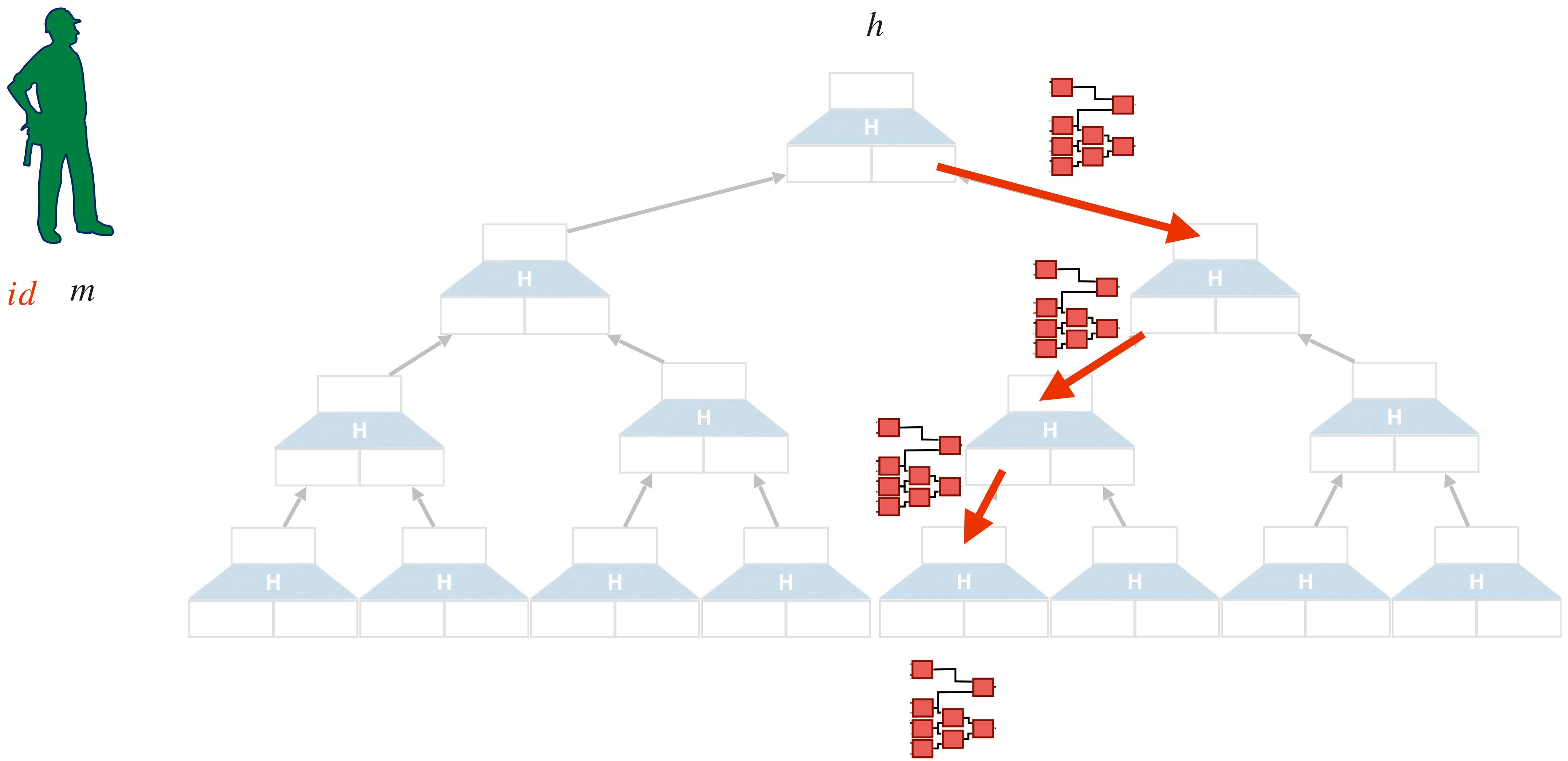
Identity-Based Encryption [DG17]: Encryption

Reverse Delegation as in Laconic OT



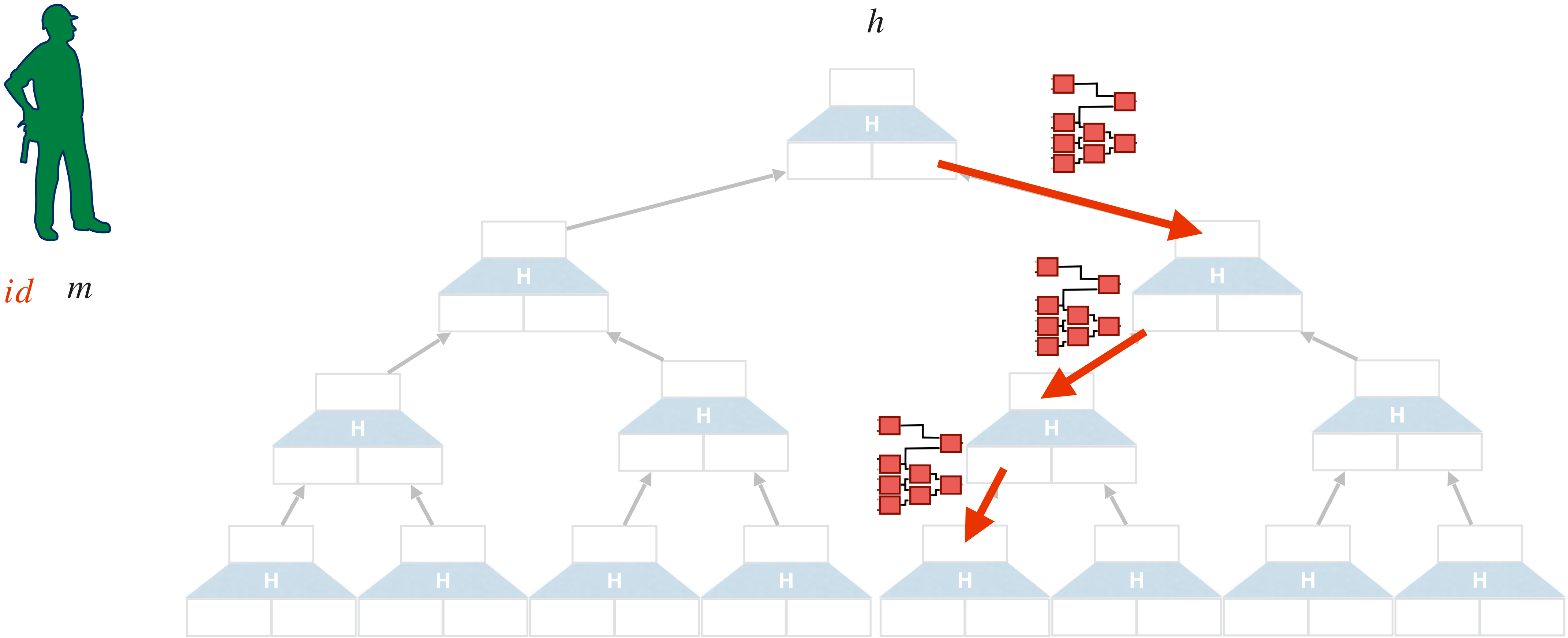
Identity-Based Encryption [DG17]: Encryption

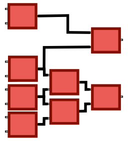
Reverse Delegation as in Laconic OT



Identity-Based Encryption [DG17]: Encryption

Reverse Delegation as in Laconic OT

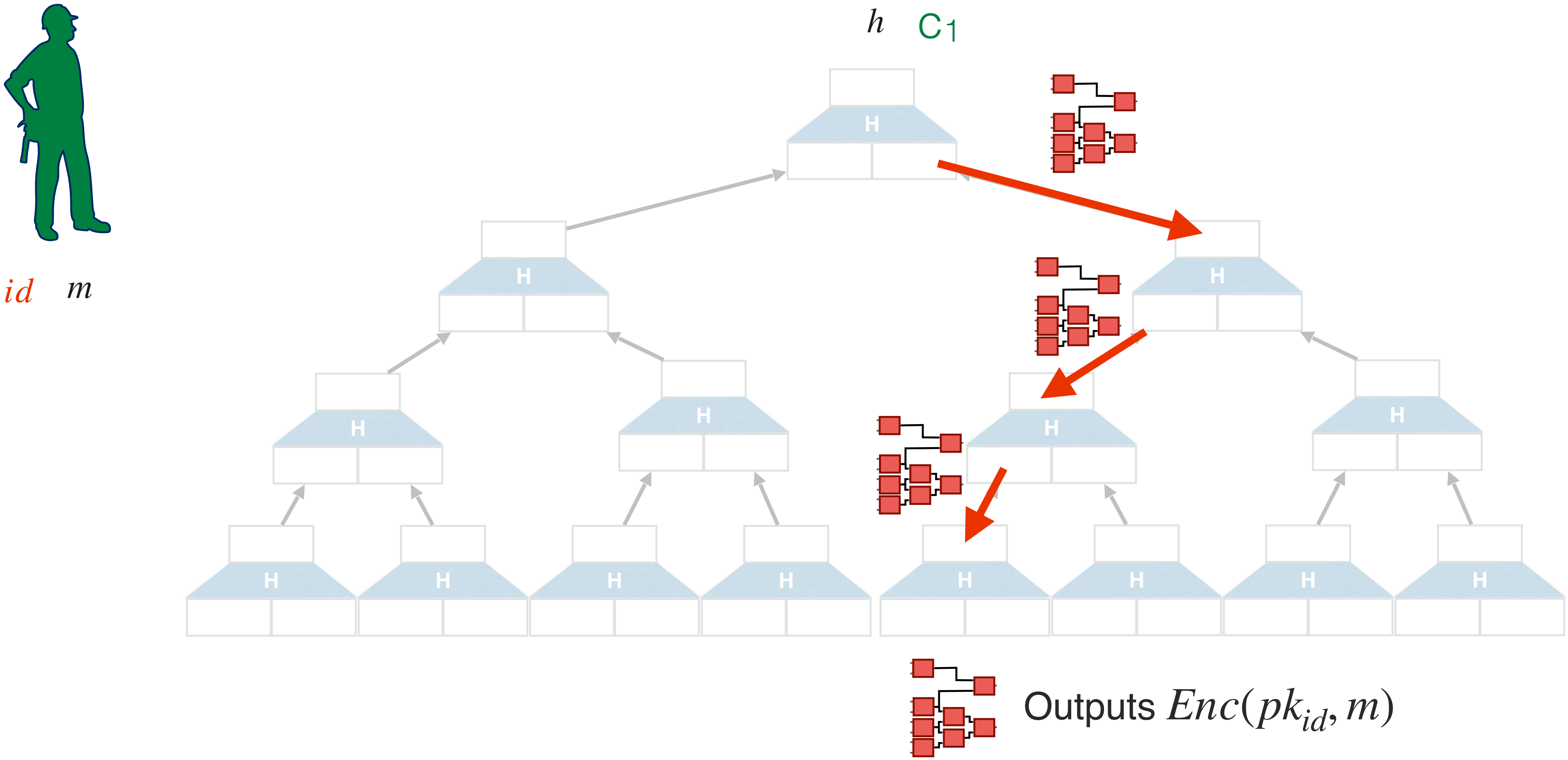




Outputs $Enc(pk_{id}, m)$

Identity-Based Encryption [DG17]: Encryption

Reverse Delegation as in Laconic OT

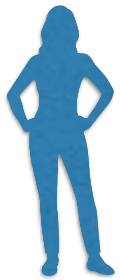


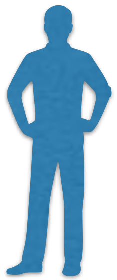
Registration-based Encryption [GHMR18]

$(pk, sk) \leftarrow KeyGen()$

Registration-based Encryption [GHMR18]

- Master Secret Key msk is single point of failure in IBE

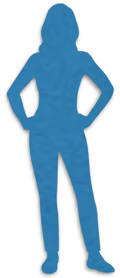


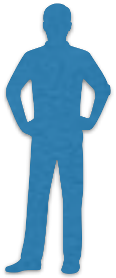


$(pk, sk) \leftarrow KeyGen()$

Registration-based Encryption [GHMR18]

- Master Secret Key msk is single point of failure in IBE
- Idea: Replace Key-Authority with Key-Curator

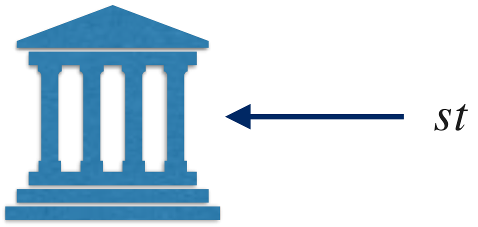


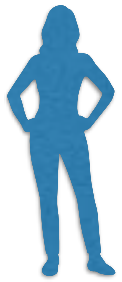


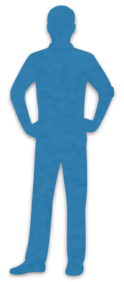
$(pk, sk) \leftarrow KeyGen()$

Registration-based Encryption [GHMR18]

- Master Secret Key msk is single point of failure in IBE
- Idea: Replace Key-Authority with Key-Curator

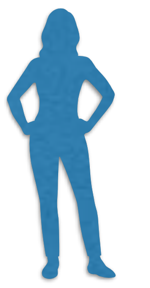


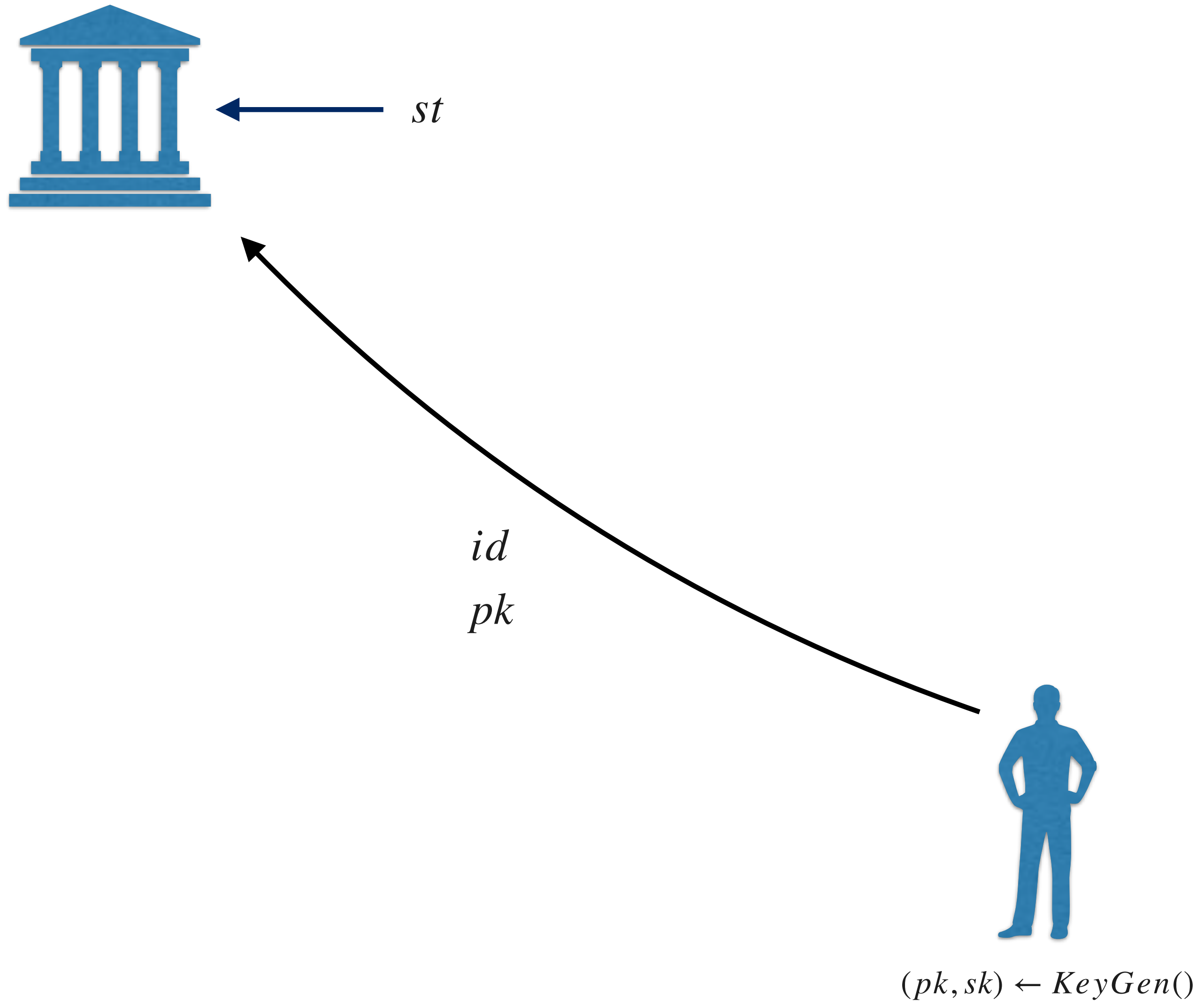




Registration-based Encryption [GHMR18]

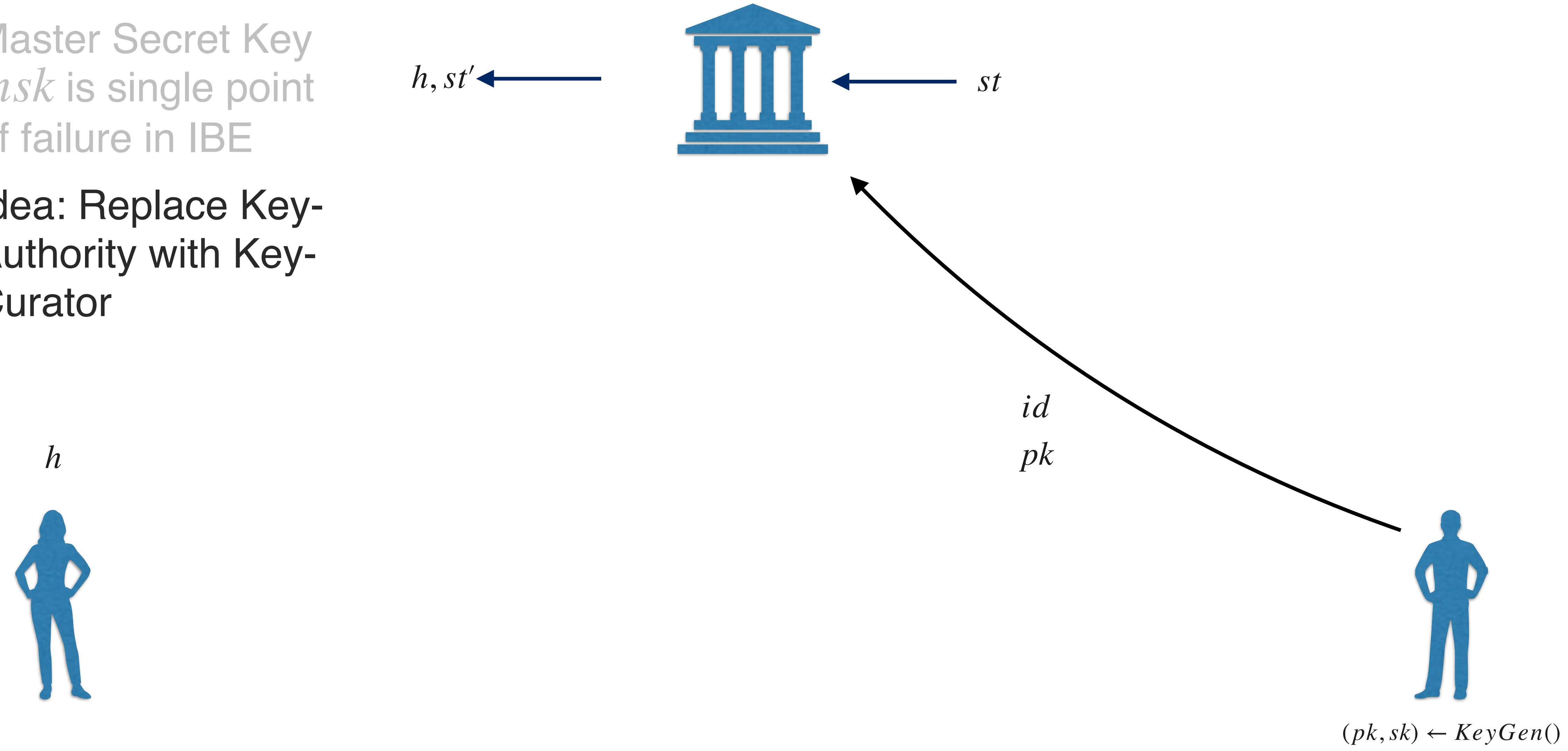
- Master Secret Key msk is single point of failure in IBE
- Idea: Replace Key-Authority with Key-Curator





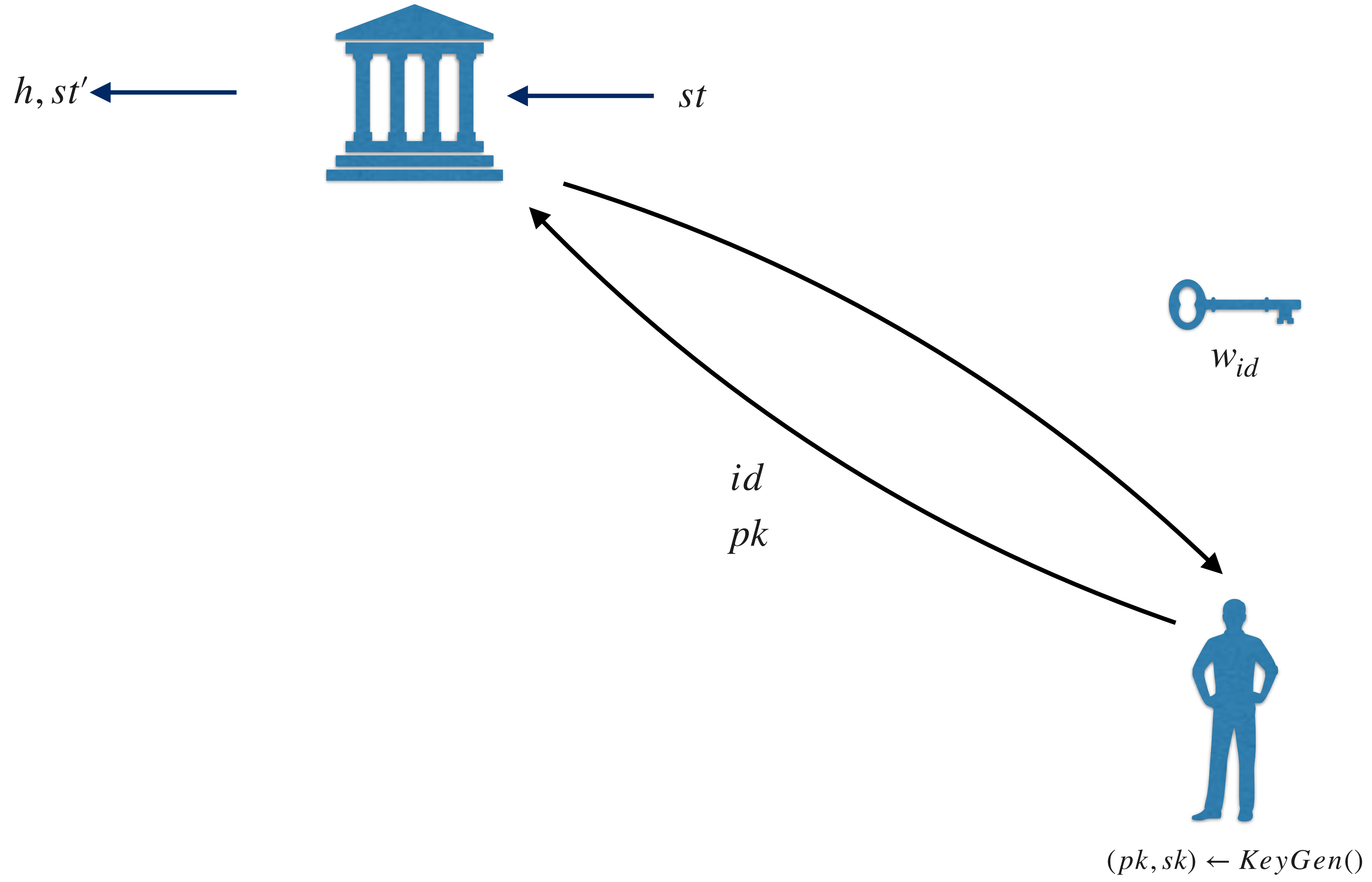
Registration-based Encryption [GHMR18]

- Master Secret Key msk is single point of failure in IBE
- Idea: Replace Key-Authority with Key-Curator



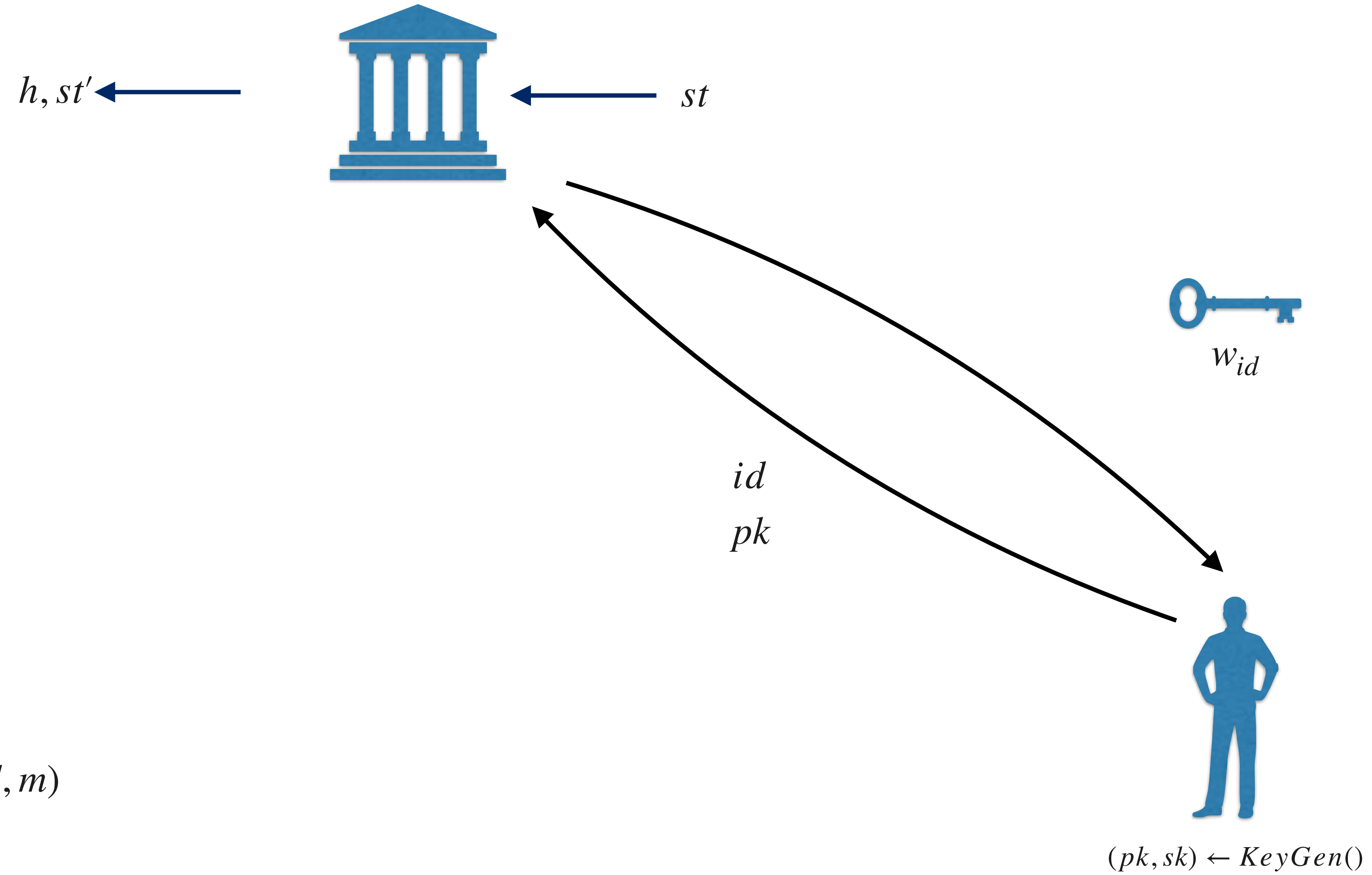
Registration-based Encryption [GHMR18]

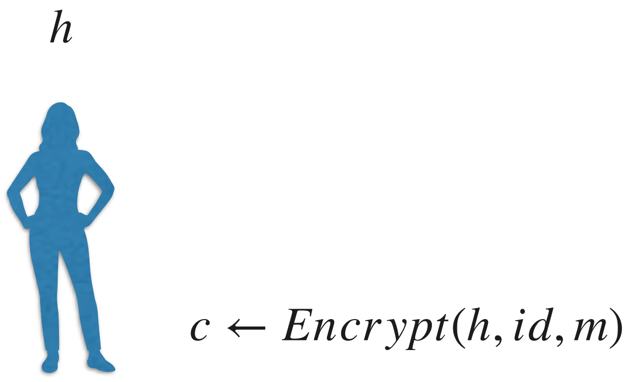
- Master Secret Key msk is single point of failure in IBE
- Idea: Replace Key-Authority with Key-Curator



Registration-based Encryption [GHMR18]

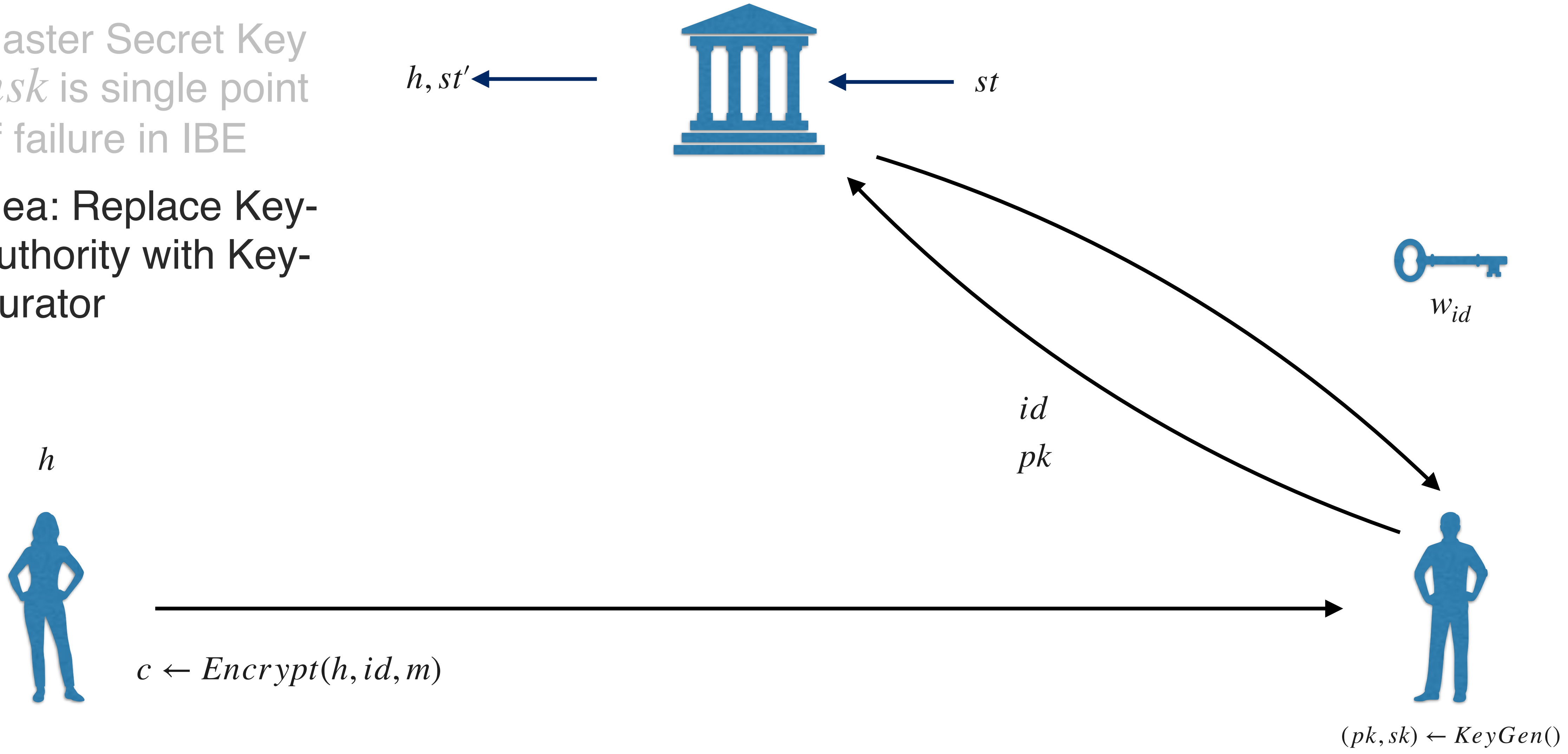
- Master Secret Key msk is single point of failure in IBE
- Idea: Replace Key-Authority with Key-Curator





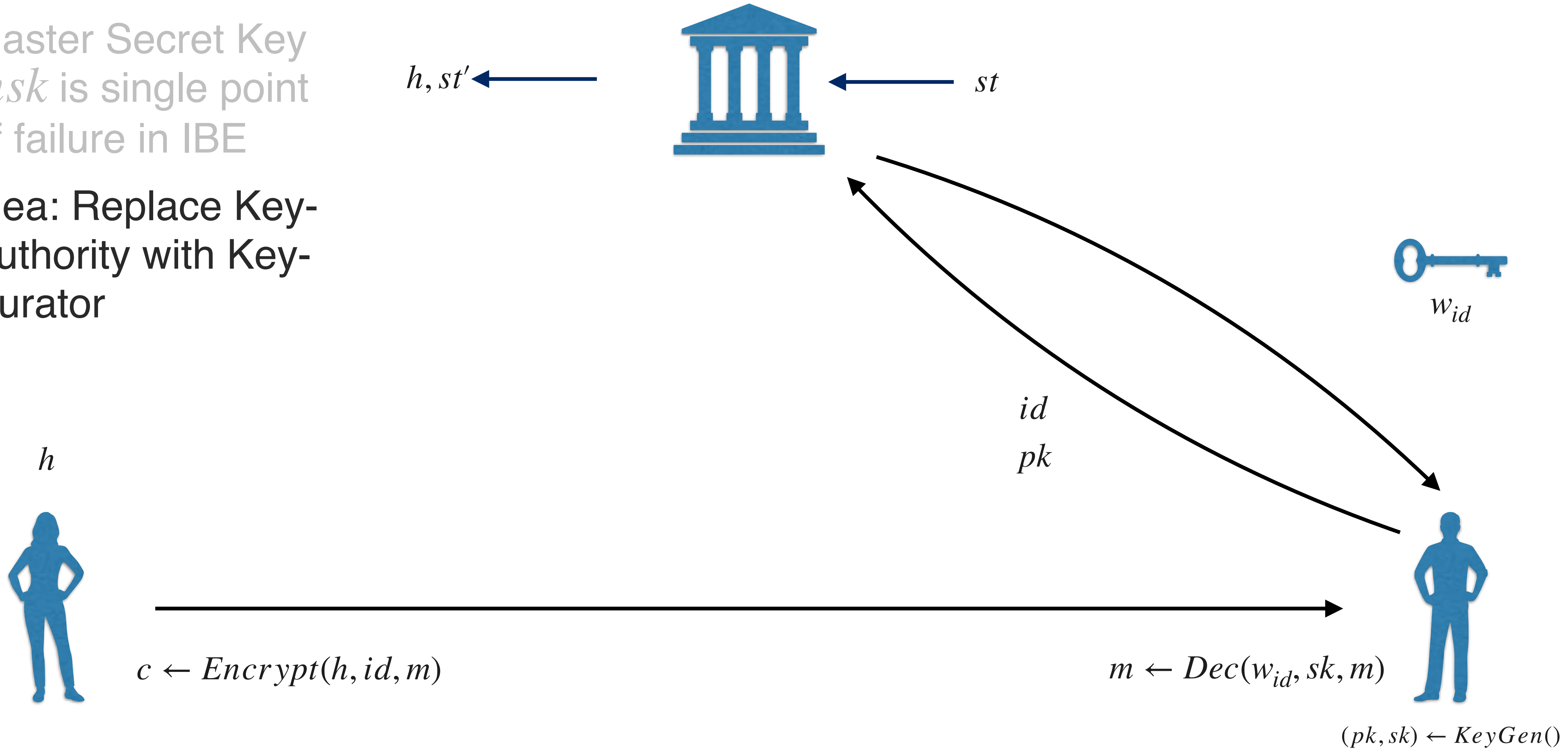
Registration-based Encryption [GHMR18]

- Master Secret Key msk is single point of failure in IBE
- Idea: Replace Key-Authority with Key-Curator

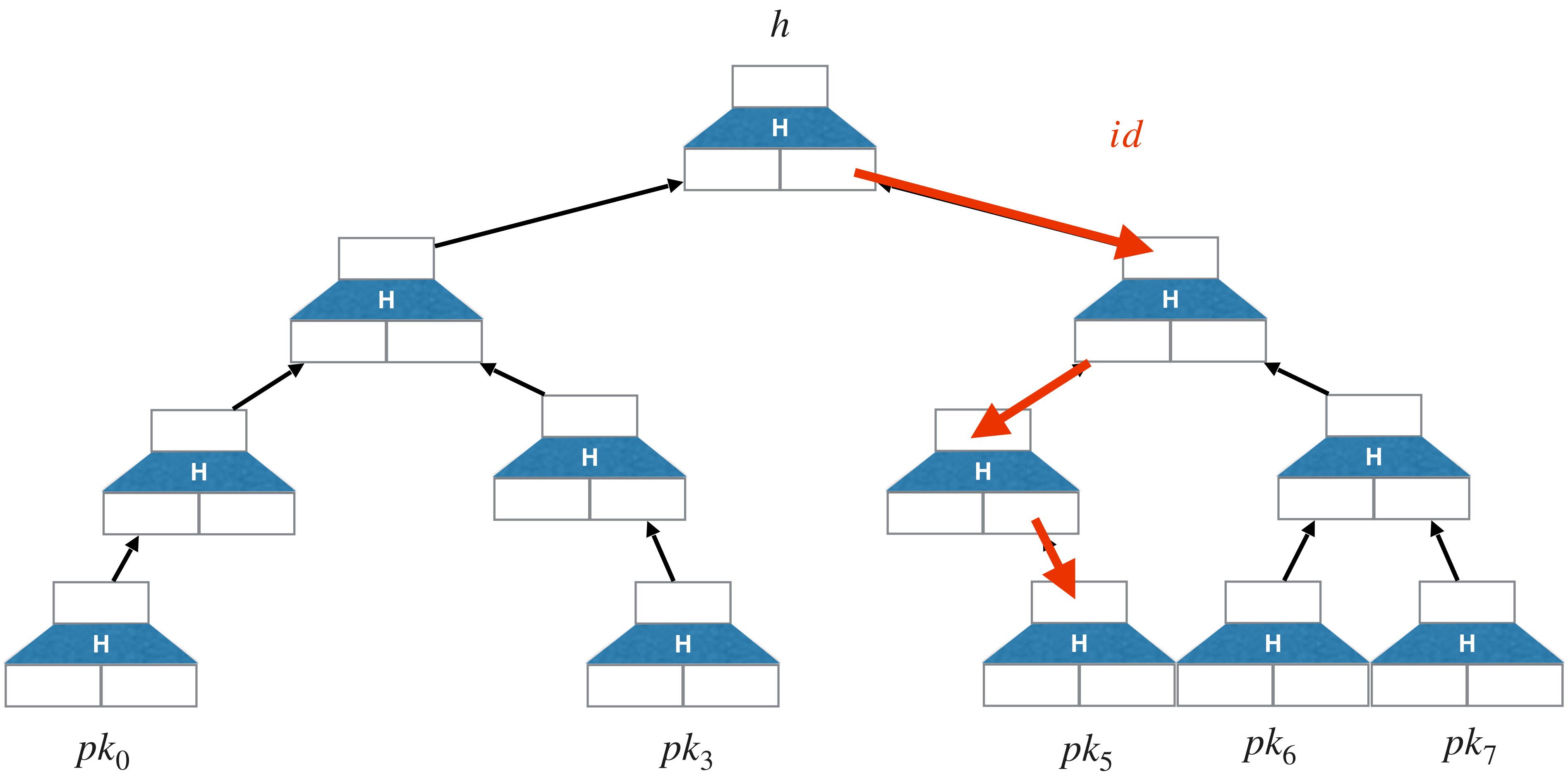


Registration-based Encryption [GHMR18]

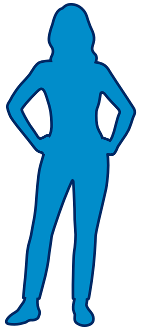
- Master Secret Key msk is single point of failure in IBE
- Idea: Replace Key-Authority with Key-Curator



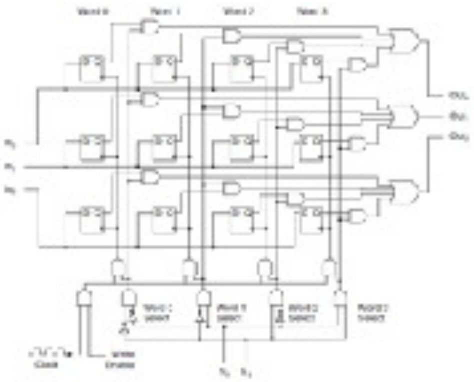
Registration-based Encryption: Registration



Laconic Function Evaluation [QWW18]



x

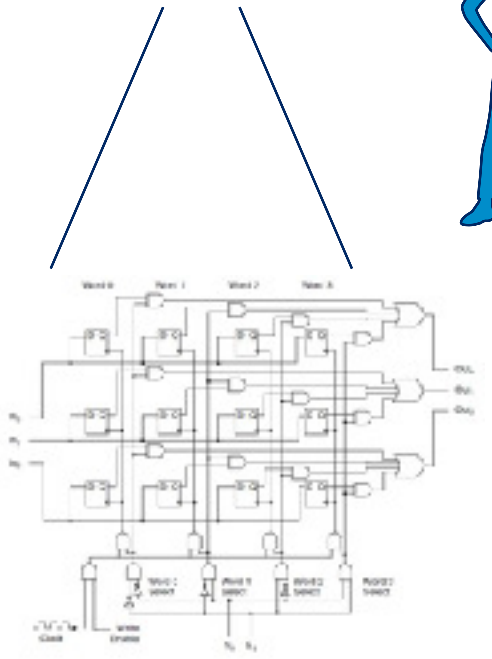


C

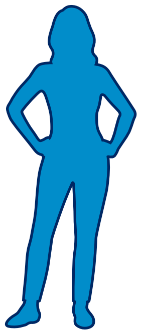
Laconic Function Evaluation [QWW18]

x

h



C

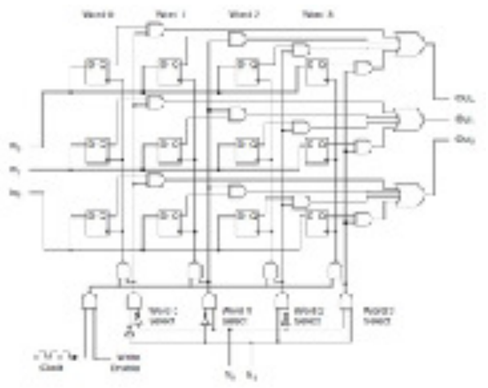


Laconic Function Evaluation [QWW18]

h

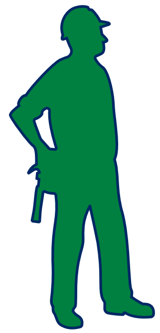
x

h



C

Laconic Function Evaluation [QWW18]

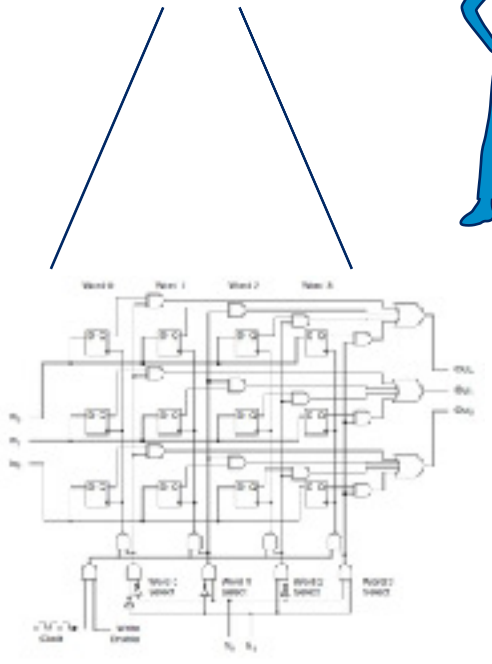


x

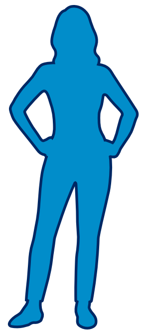
h

$c = Enc(h, x)$

h

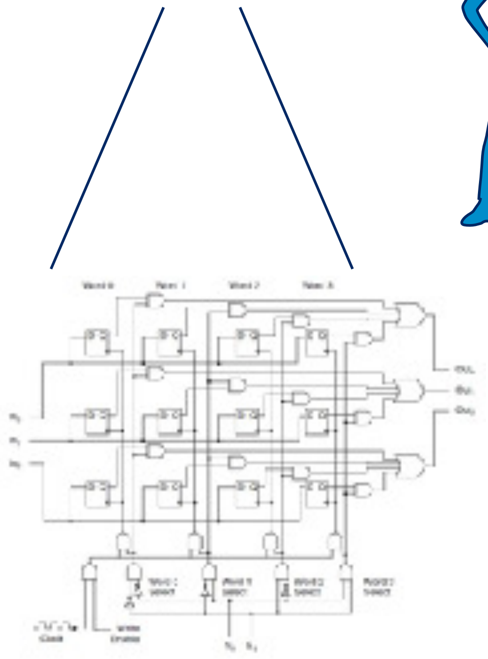


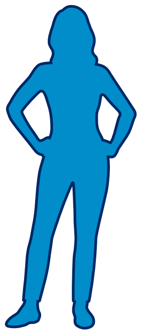
C



Laconic Function Evaluation [QWW18]

h





x

h

$c = Enc(h, x)$

C

c

Laconic Function Evaluation [QWW18]

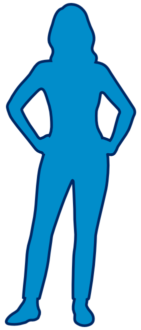
x

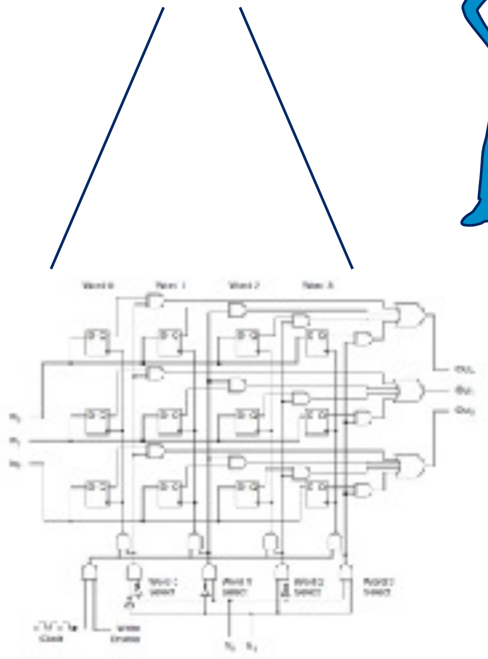
h

$c = Enc(h, x)$

c

h





C

$y = Dec(C, c) = C(x)$

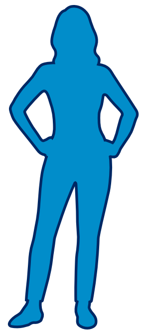
Laconic Function Evaluation [QWW18]

x

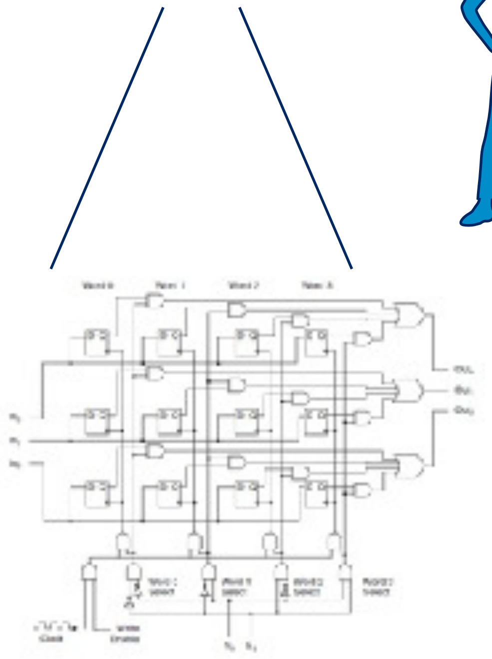
h

$c = Enc(h, x)$

c



h



C

$y = Dec(C, c) = C(x)$

Learns nothing about x except $C(x)$

Laconic Function Evaluation [QWW18]

x

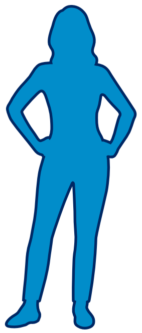
$c = Enc(h, x)$

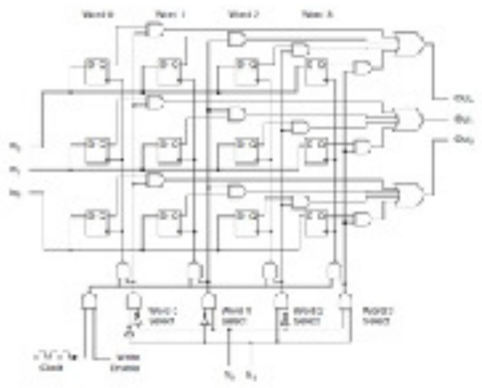
h

succinct

c

h



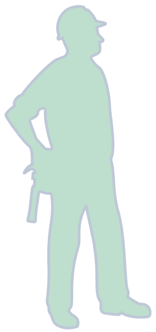


C

$y = Dec(C, c) = C(x)$

Learns nothing about x except $C(x)$

Laconic Function Evaluation [QWW18]



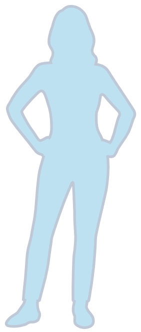
x

$c = Enc(h, x)$

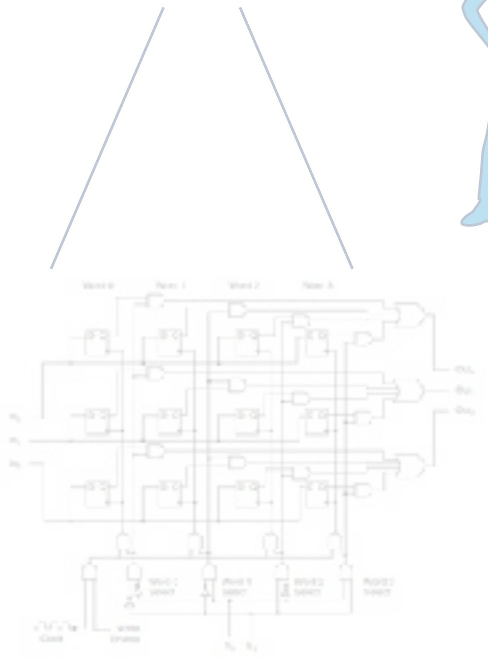
h

succinct

c



h

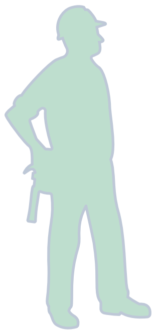


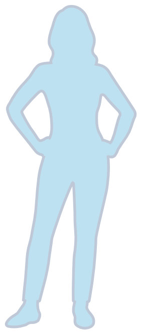
C

$$y = Dec(C, c) = C(x)$$

Learns nothing about x except $C(x)$

Laconic Function Evaluation [QWW18]



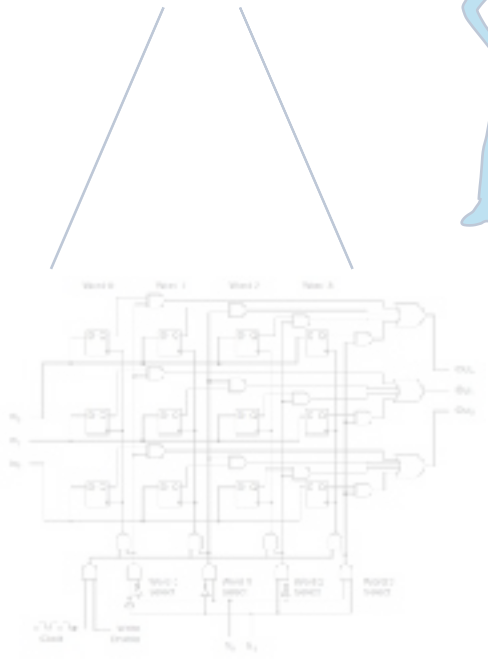


- Receiver commits to large function instead of database

x

h

h



$c = Enc(h, x)$

succinct

C

c

$y = Dec(C, c) = C(x)$

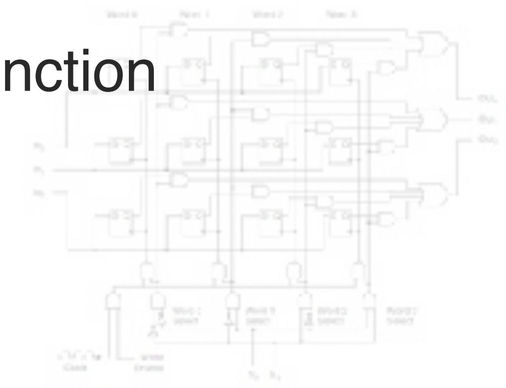
Learns nothing about x except $C(x)$

Laconic Function Evaluation [QWW18]

- Receiver commits to large function instead of database
- Laconic OT is a special case of LFE: Hashed function is selection function

x

h



C

succinct

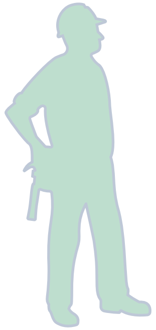
$c = Enc(h, x)$

c

$y = Dec(C, c) = C(x)$

Learns nothing about x except $C(x)$

Laconic Function Evaluation [QWW18]



x

- Receiver commits to large function instead of database
- Laconic OT is a special case of LFE: Hashed function is selection function
- [QWW18] construct LFE from LWE

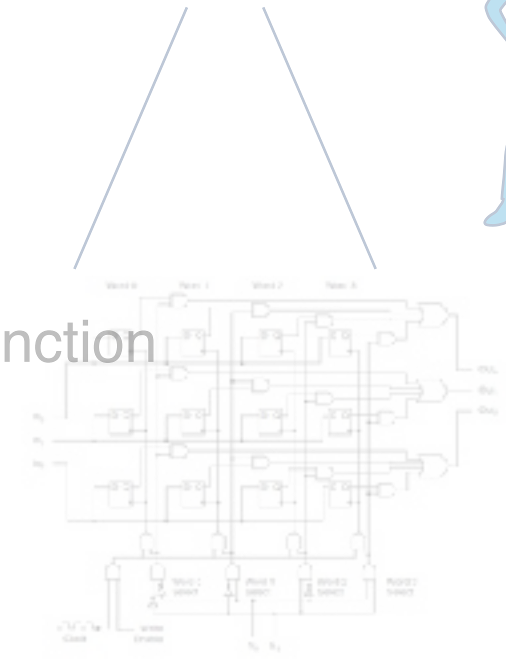
h

succinct

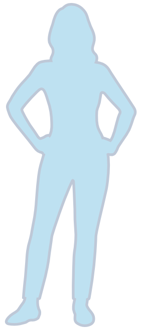
c

$c = Enc(h, x)$

h



C



$y = Dec(C, c) = C(x)$

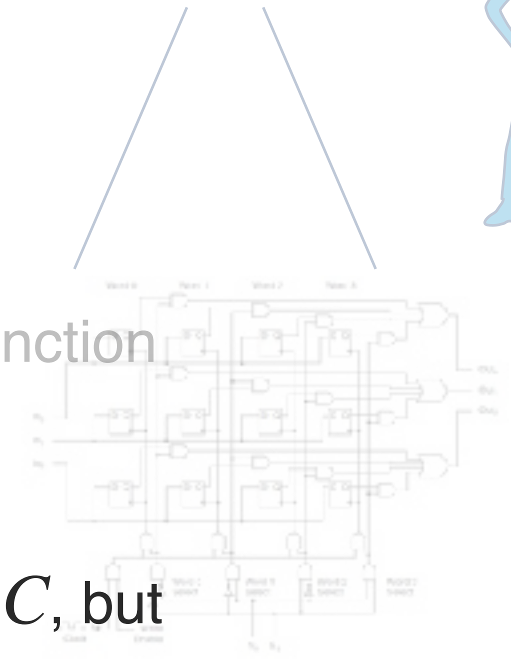
Learns nothing about x except $C(x)$

Laconic Function Evaluation [QWW18]

x

- Receiver commits to large function instead of database
- Laconic OT is a special case of LFE: Hashed function is selection function
- [QWW18] construct LFE from LWE
- Size of ciphertext c depends on depth of circuit C, but not on size

h



C

$c = Enc(h, x)$

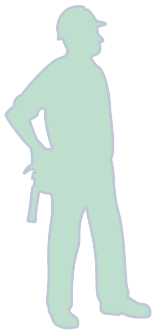
succinct

c

$y = Dec(C, c) = C(x)$

Learns nothing about x except $C(x)$

Laconic Function Evaluation [QWW18]



- Receiver commits to large function instead of database
- Laconic OT is a special case of LFE: Hashed function is selection function
- [QWW18] construct LFE from LWE
- Size of ciphertext c depends on depth of circuit C, but not on size

x

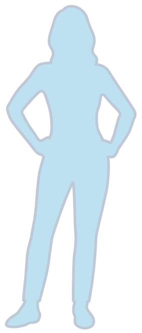
$c = Enc(h, x)$

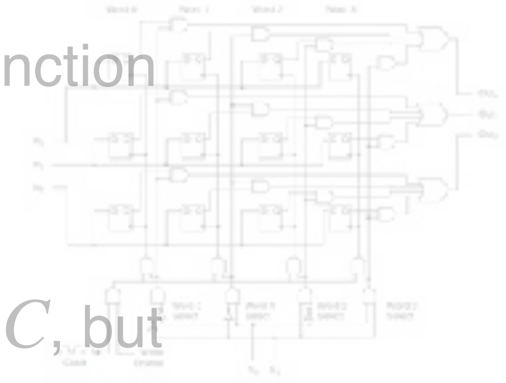
h

succinct

c

h



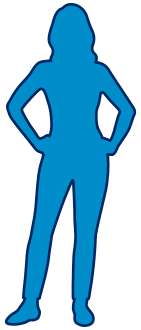


C

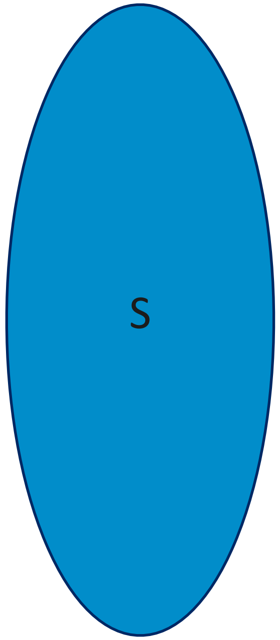
$y = Dec(C, c) = C(x)$

Learns nothing about x except $C(x)$

Laconic Private Set Intersection (LPSI) [ABDGHP21]

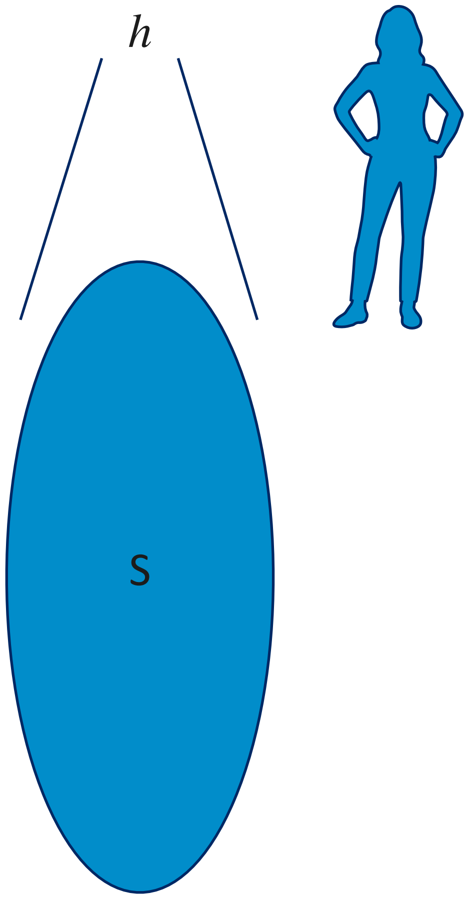


x

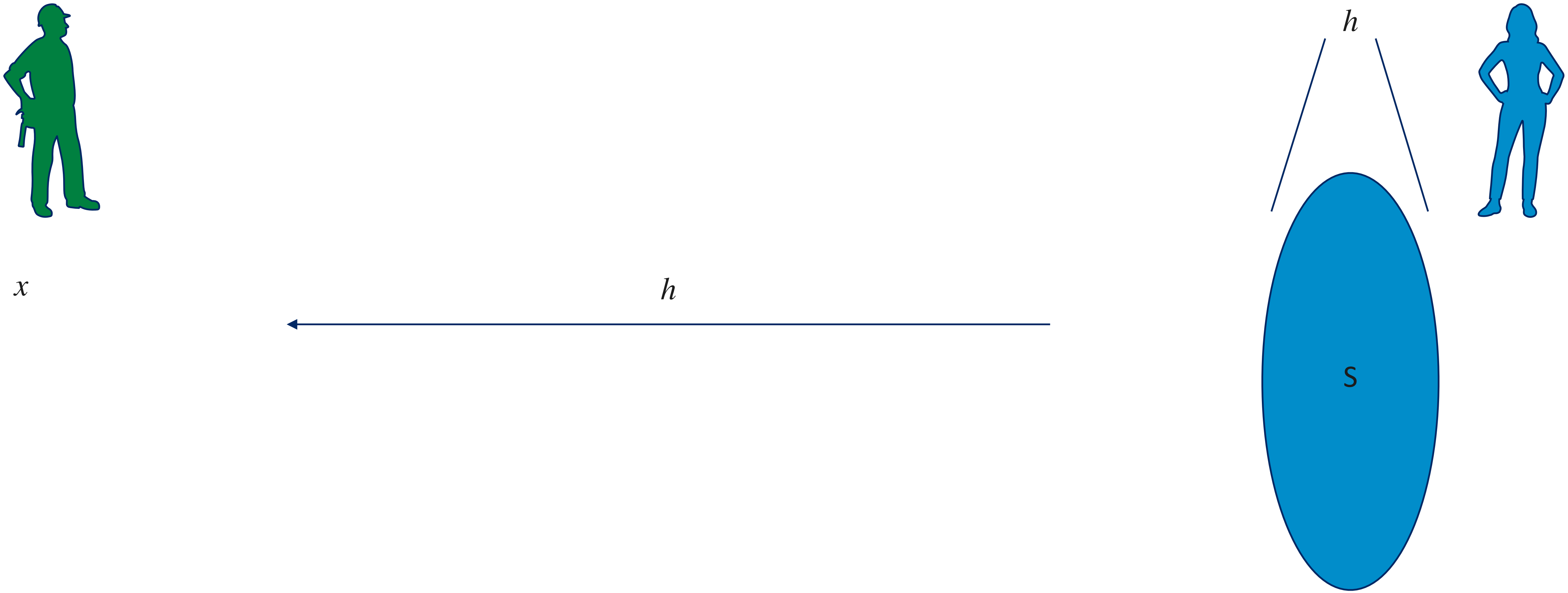


Laconic Private Set Intersection (LPSI) [ABDGHP21]

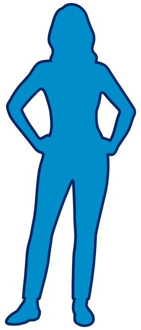
x



Laconic Private Set Intersection (LPSI) [ABDGHP21]



Laconic Private Set Intersection (LPSI) [ABDGHP21]



h

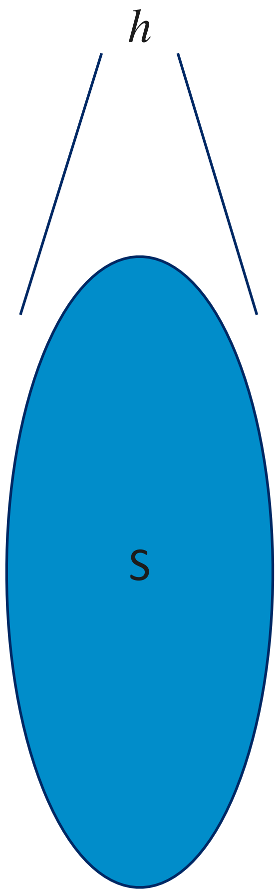
x

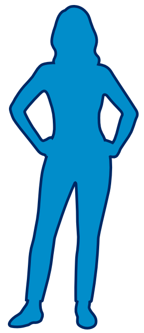
h

S

$c = Enc(h, x)$

Laconic Private Set Intersection (LPSI) [ABDGHP21]





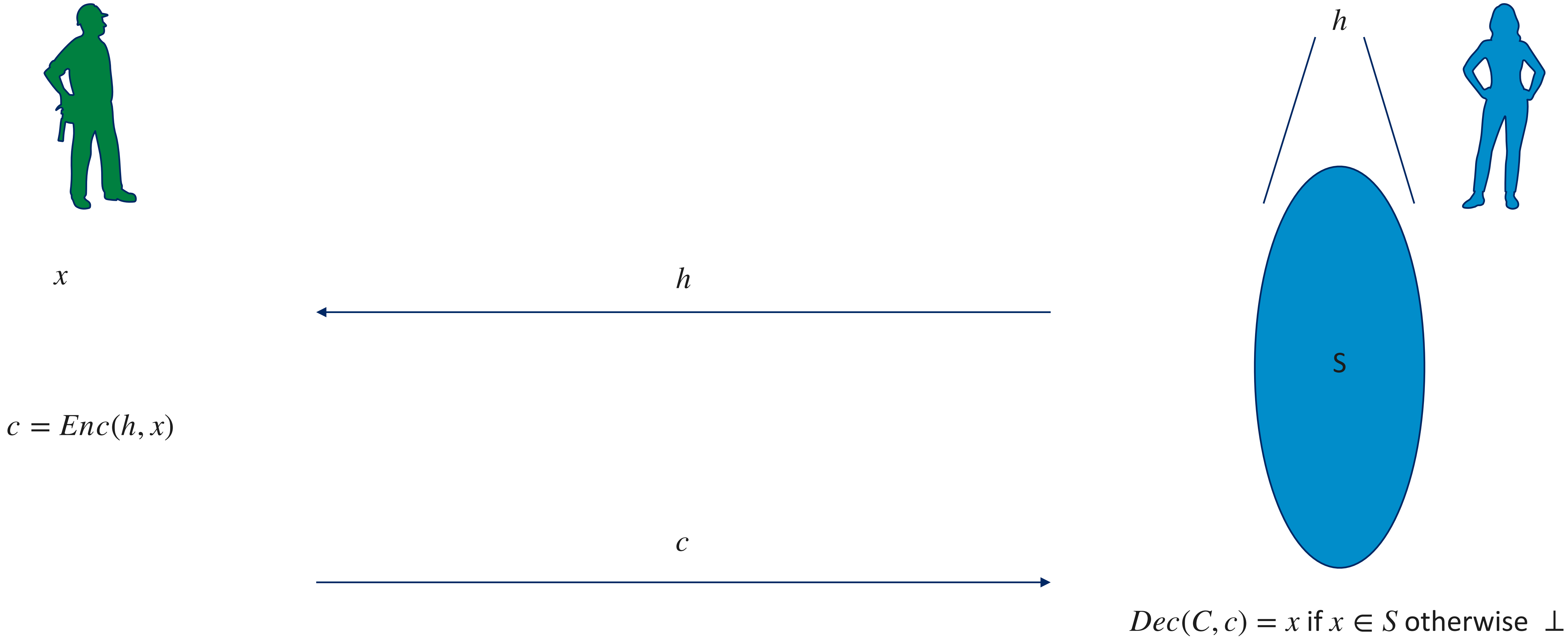
x

h (arrow from right to left)

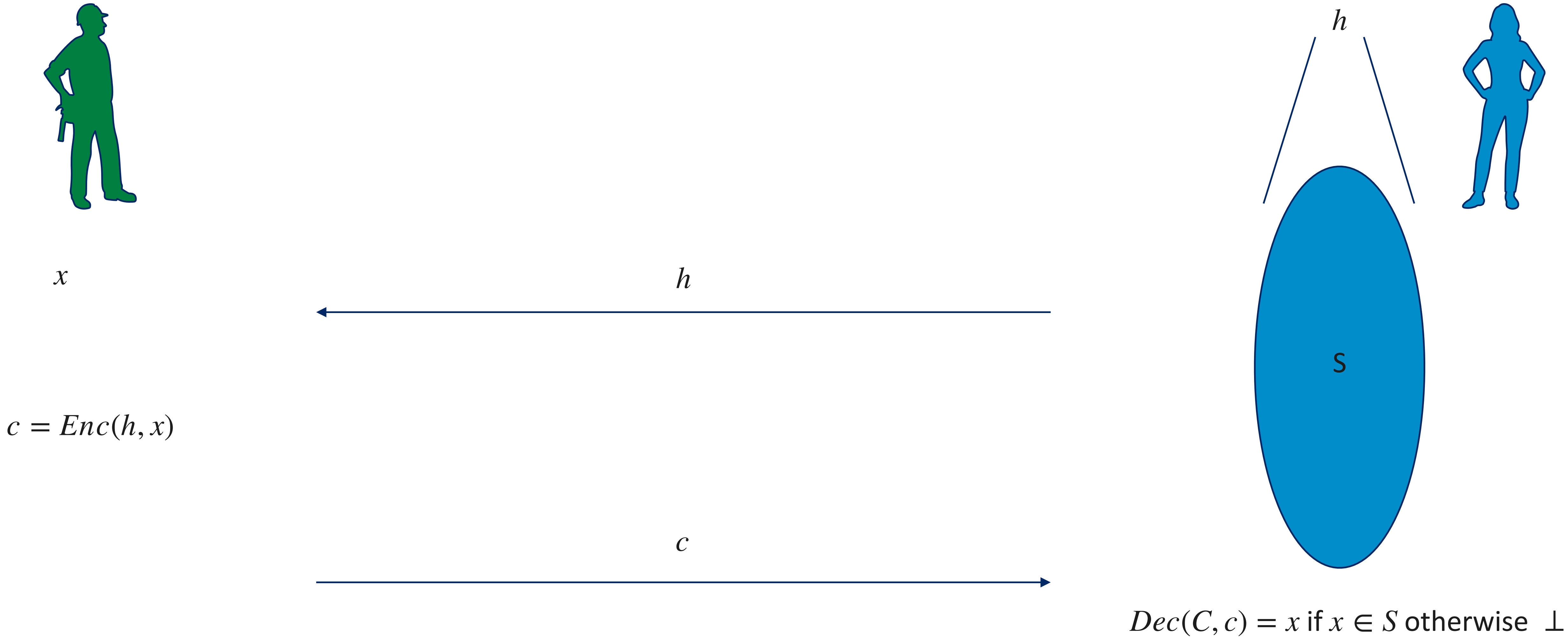
$c = Enc(h, x)$

c (arrow from left to right)

Laconic Private Set Intersection (LPSI) [ABDGHP21]



Laconic Private Set Intersection (LPSI) [ABDGHP21]



Learns nothing about x if $x \notin S$

Laconic Private Set Intersection (LPSI) [ABDGHP21]

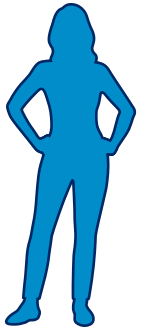
x

$c = Enc(h, x)$

h

succinct

c



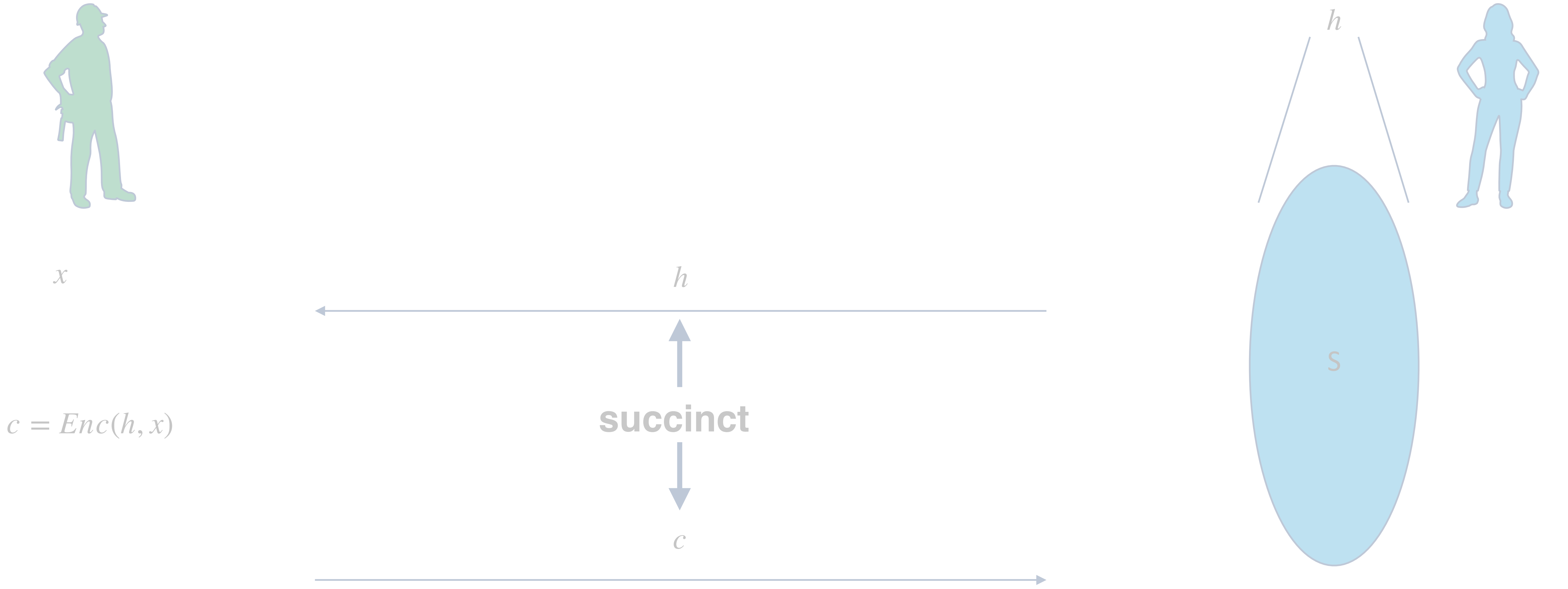
h

S

$Dec(C, c) = x$ if $x \in S$ otherwise $\perp$

Learns nothing about x if $x \notin S$

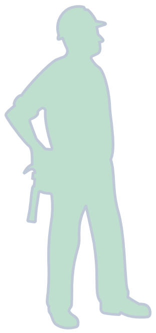
Laconic Private Set Intersection (LPSI) [ABDGHP21]



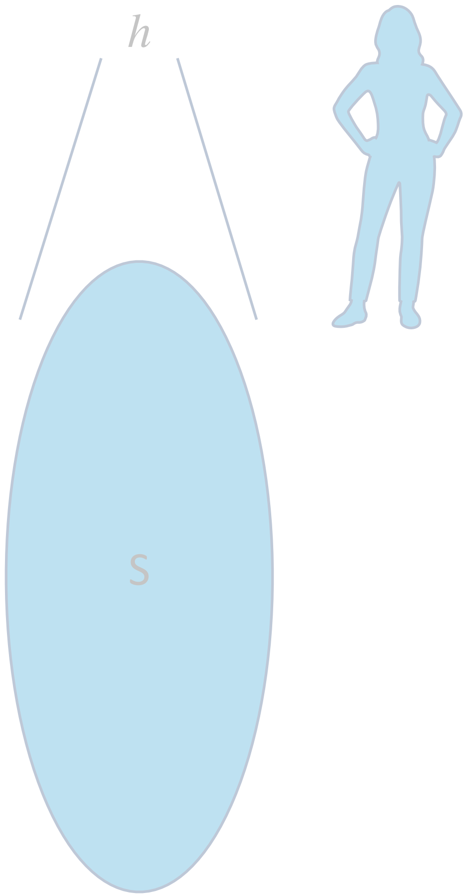
$Dec(C, c) = x$ if $x \in S$ otherwise $\perp$

Learns nothing about x if $x \notin S$

Laconic Private Set Intersection (LPSI) [ABDGHP21]



- [ABDGHP21] provides an efficient black-box construction of LPSI from number-theoretic assumptions



x

h

succinct

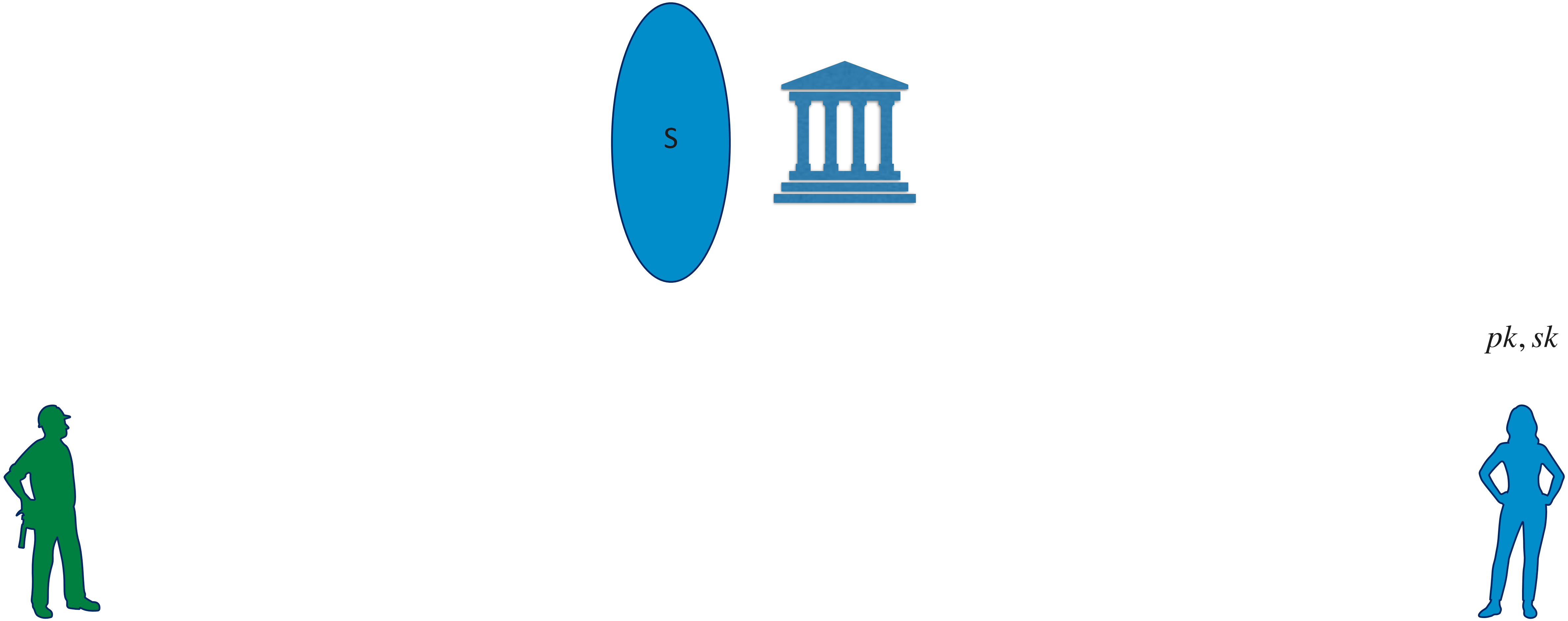
$c = Enc(h, x)$

c

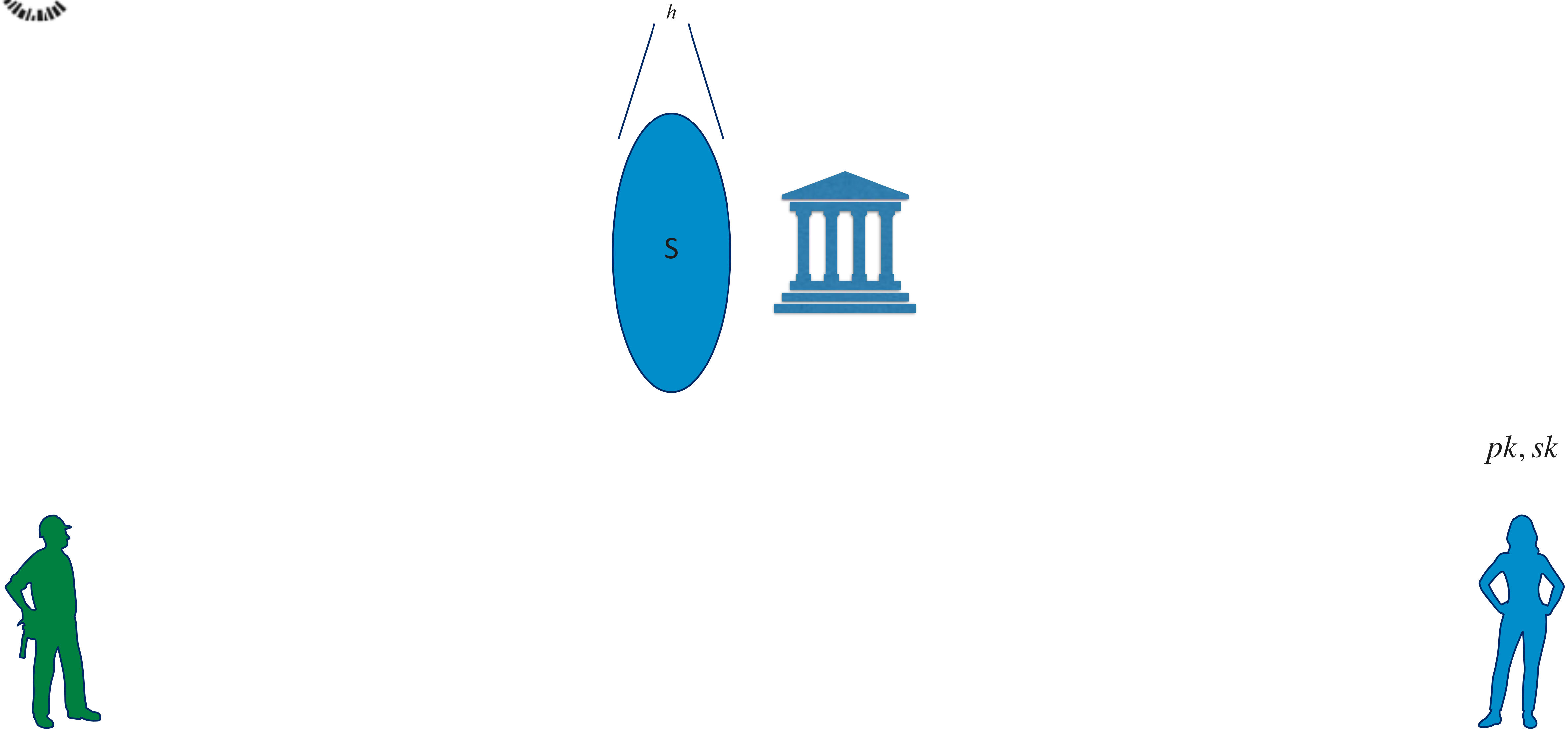
$Dec(C, c) = x$ if $x \in S$ otherwise $\perp$

$\boxed{\text{Learns nothing about } x \text{ if } x \notin S}$

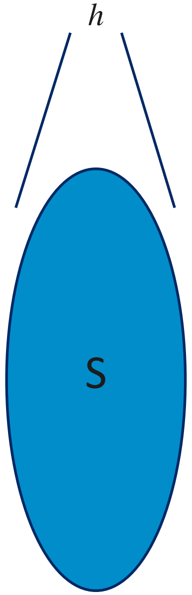
Application of LPSI: Self-Revealing Encryption



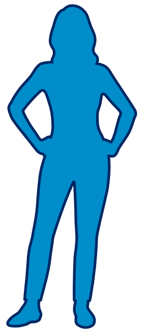
Application of LPSI: Self-Revealing Encryption



Application of LPSI: Self-Revealing Encryption

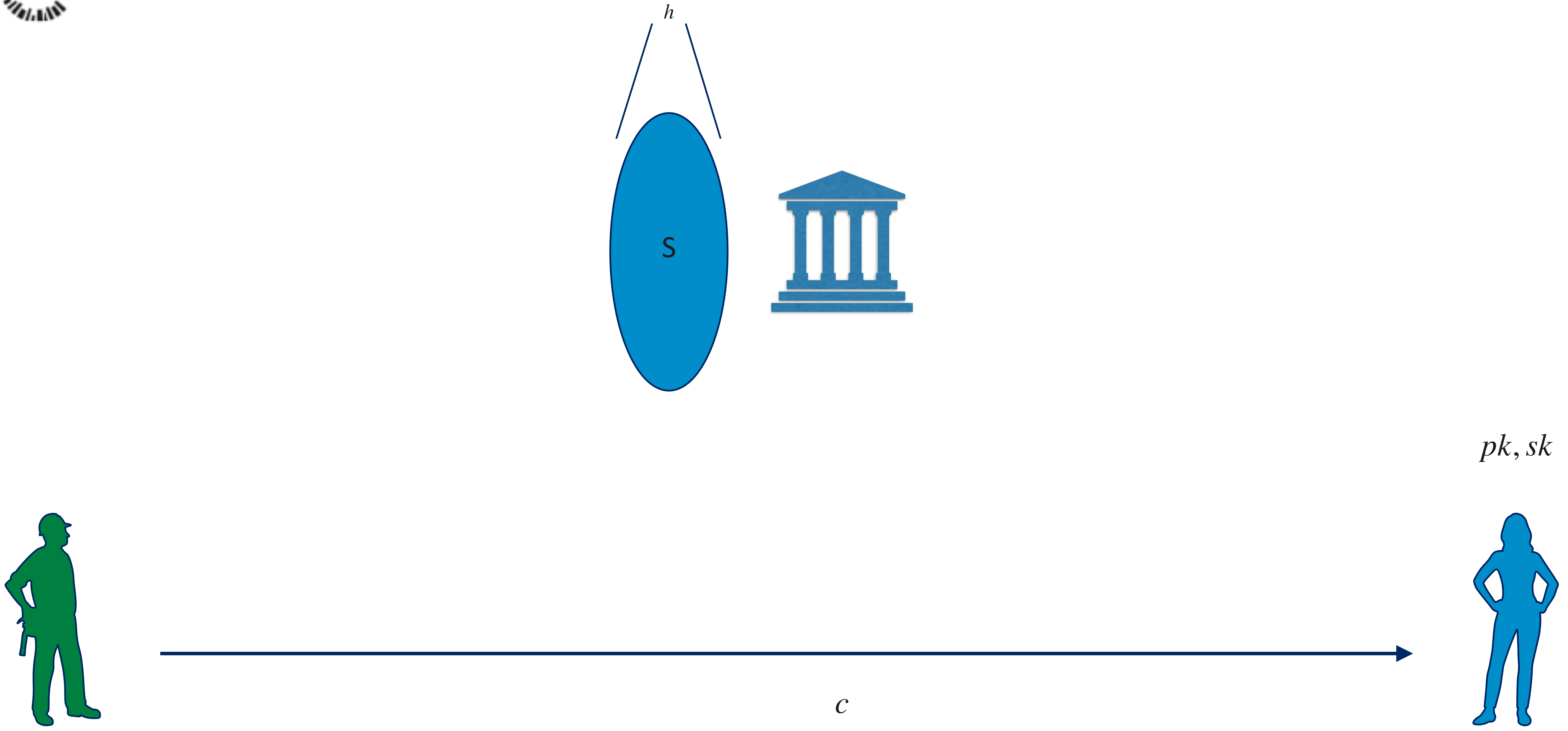


pk, sk



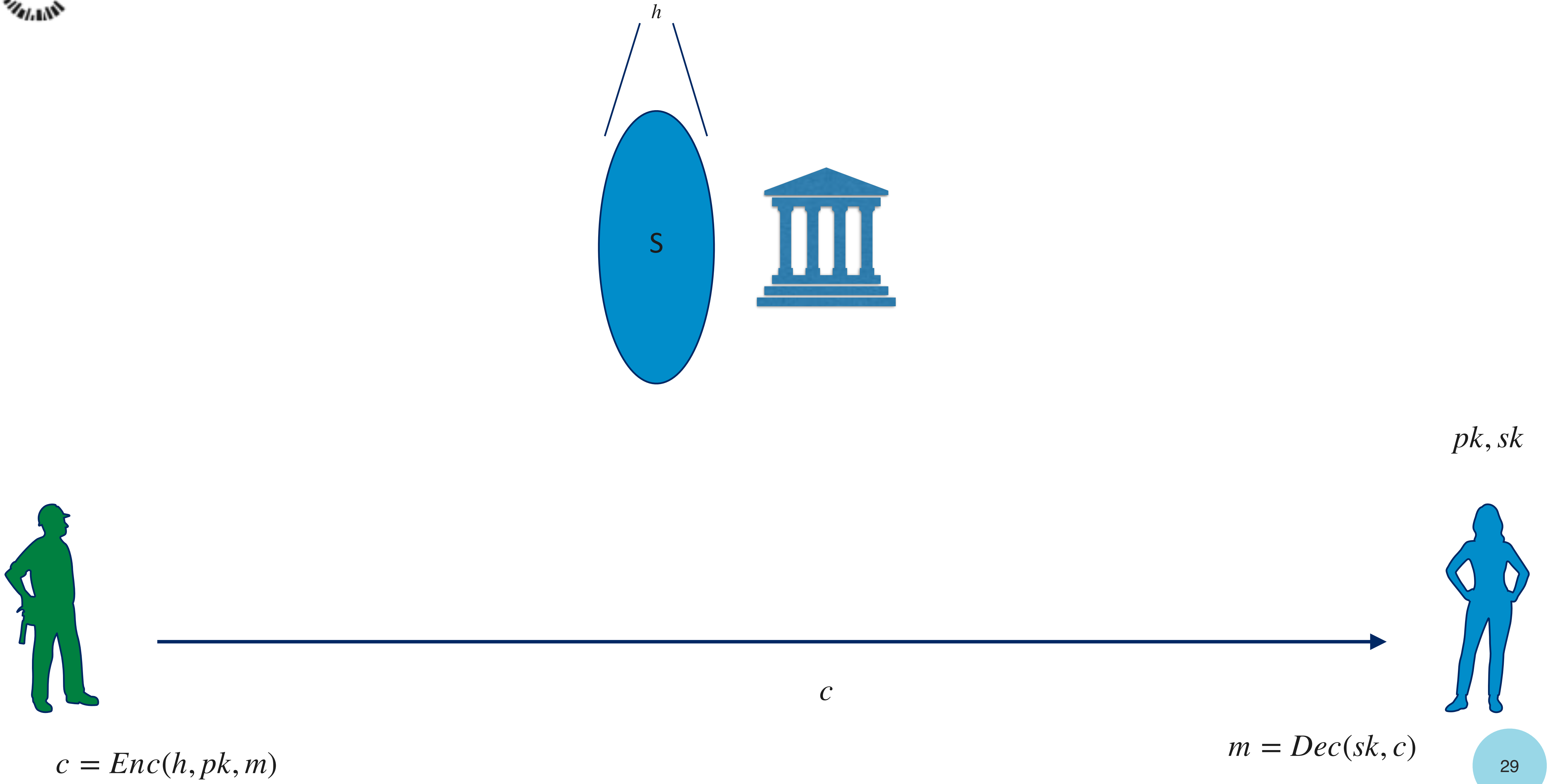
$c = Enc(h, pk, m)$

Application of LPSI: Self-Revealing Encryption

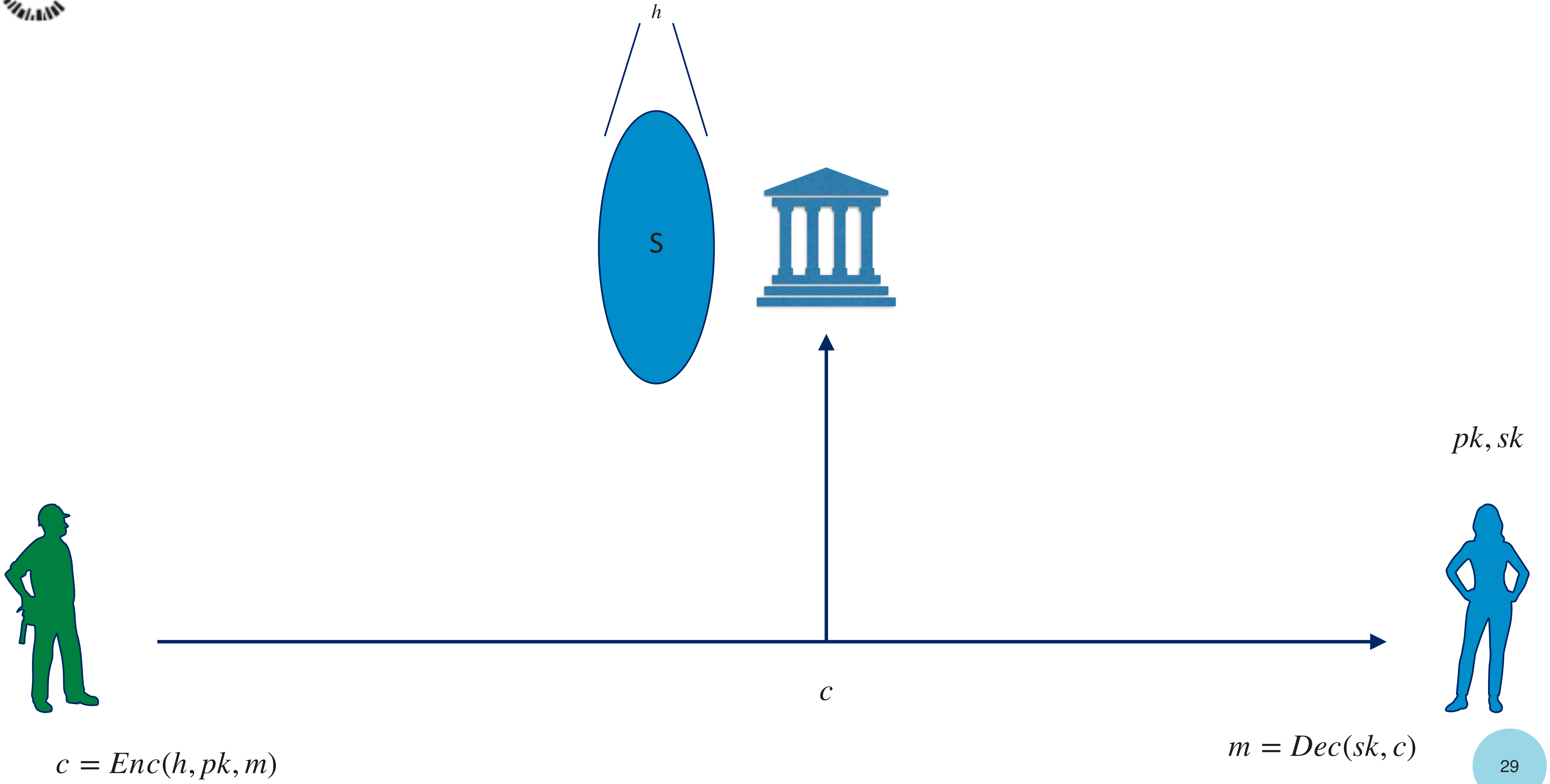


$c = Enc(h, pk, m)$

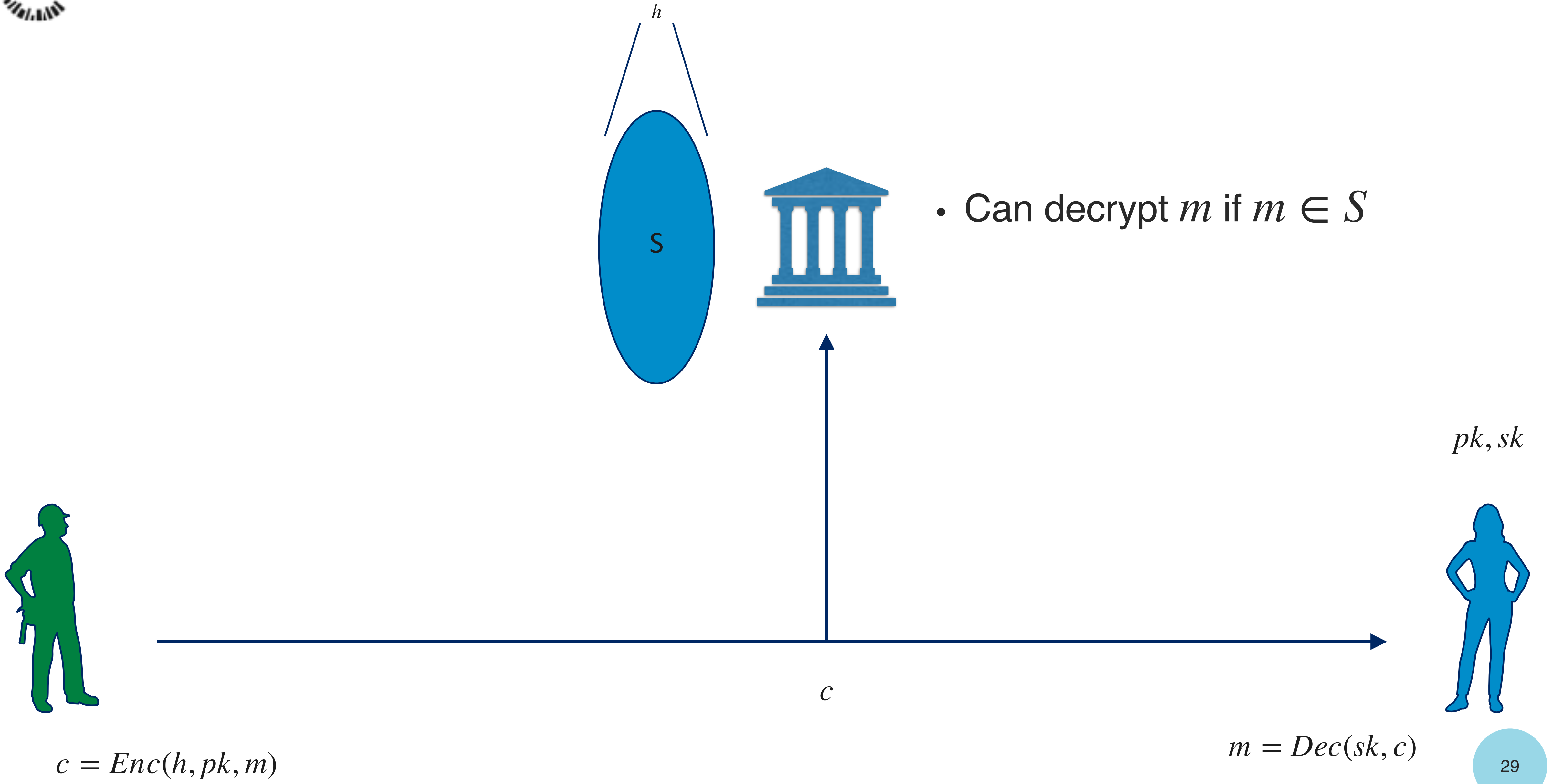
Application of LPSI: Self-Revealing Encryption



Application of LPSI: Self-Revealing Encryption



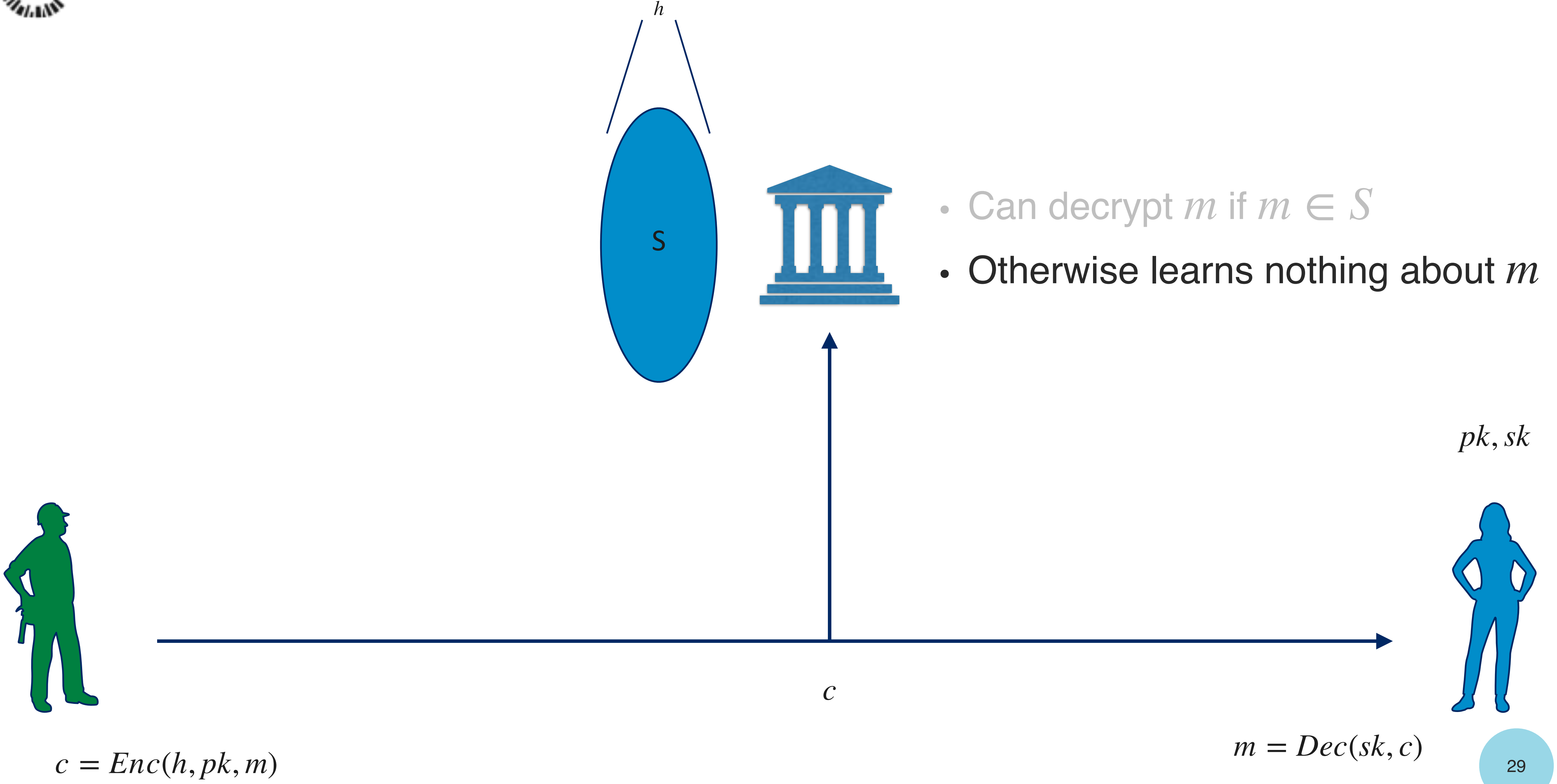
Application of LPSI: Self-Revealing Encryption



- Can decrypt m if $m \in S$

Application of LPSI: Self-Revealing Encryption

- Can decrypt m if $m \in S$
- Otherwise learns nothing about m



New Directions

New Directions

- [**D**KLLMR’22]: First Laconic Crypto Schemes **without** bootstrapping
- Key Insight: Lattice-based re-encryption gadget without intermediate decryption

New Directions

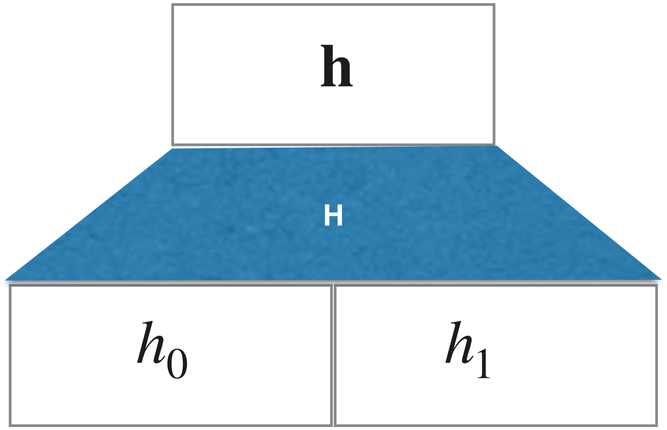
- [**D**KLLMR’22]: First Laconic Crypto Schemes **without** bootstrapping
- Key Insight: Lattice-based re-encryption gadget without intermediate decryption
- Practically efficient: Prototype Implementation with Single Digit Millisecond runtimes

New Directions

- [**D**KLLMR’22]: First Laconic Crypto Schemes **without** bootstrapping
- Key Insight: Lattice-based re-encryption gadget without intermediate decryption
- Practically efficient: Prototype Implementation with Single Digit Millisecond runtimes
- Applications: Registration-based Encryption, Laconic Oblivious Transfer, Private Set Intersection

Bootstrapping/Recryption

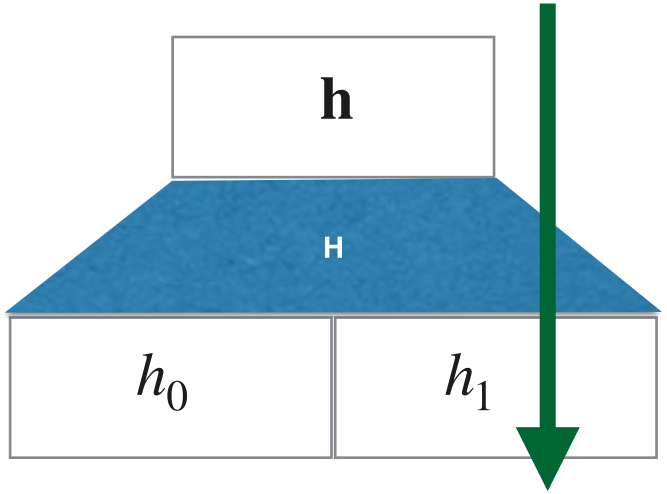
$$\mathbf{c} = \mathsf{Enc}(h, lab)$$



$$GarbleInput(lab, h_1)$$

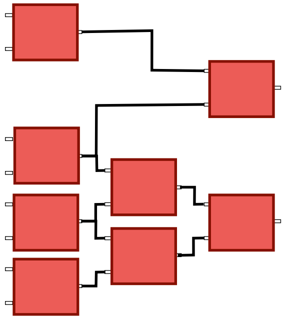
Bootstrapping/Recryption

$\mathbf{c} = \mathsf{Enc}(h, lab)$



$GarbleInput(lab, h_1)$

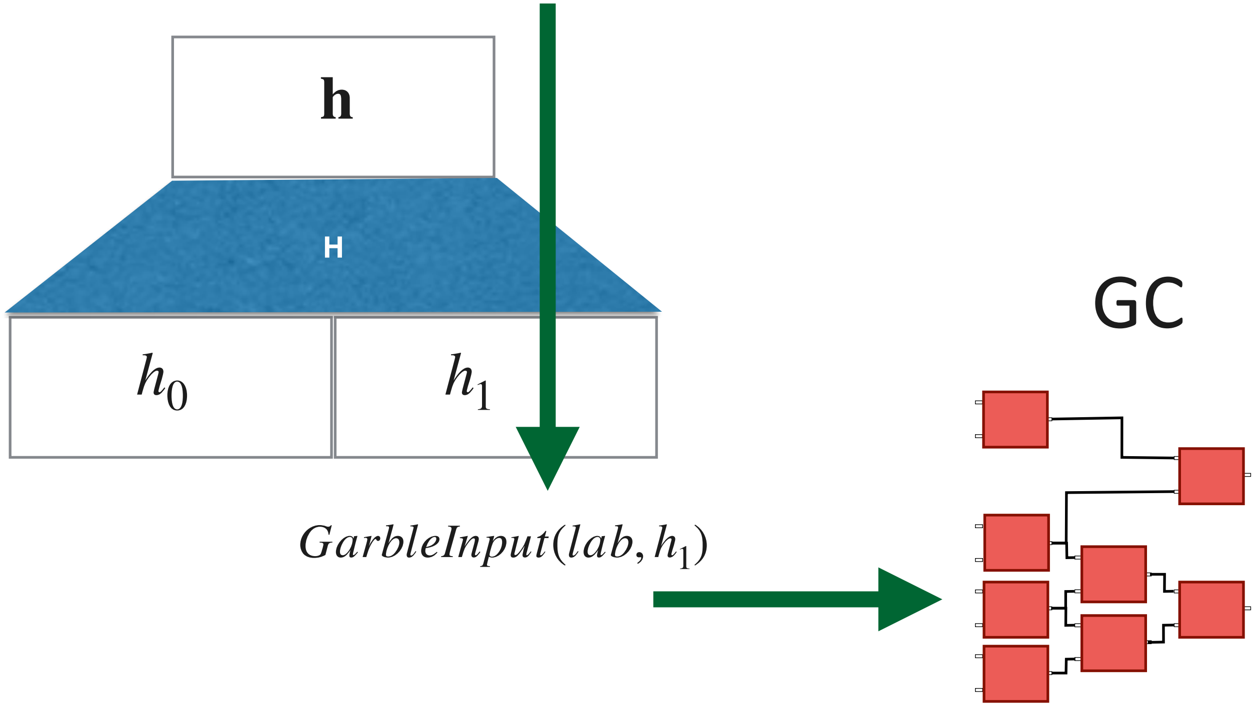
GC



$\mathbf{c}' = \mathsf{Enc}(h_1, m)$

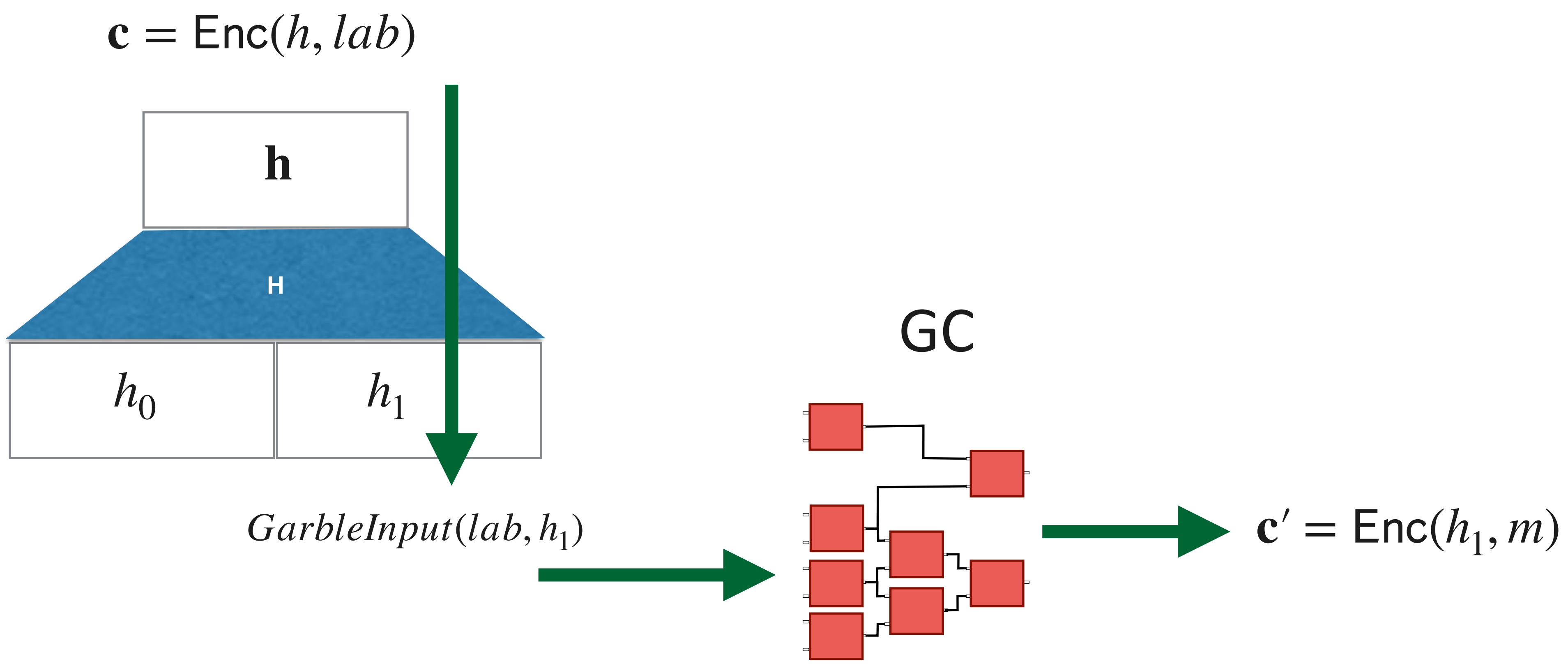
Bootstrapping/Recryption

$\mathbf{c} = \mathsf{Enc}(h, lab)$

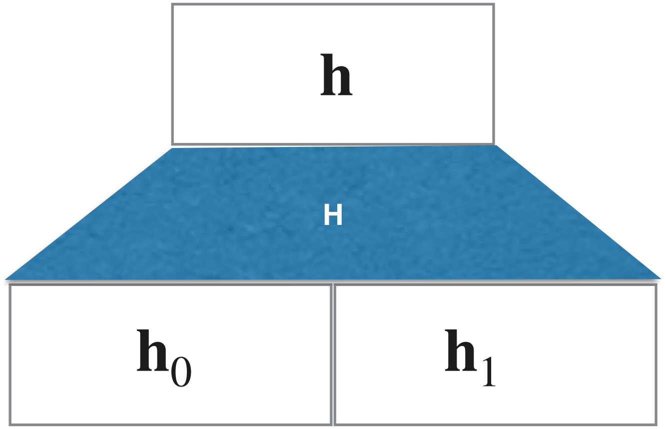


$\mathbf{c}' = \mathsf{Enc}(h_1, m)$

Bootstrapping/Recryption

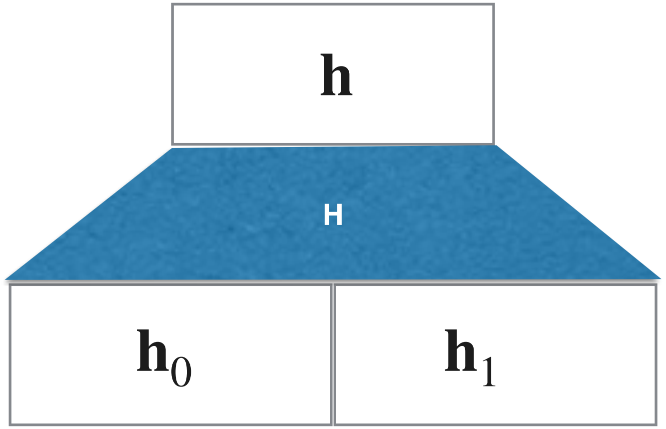


New Recryption Algorithm [DKLLMR'22]:



New Recryption Algorithm [DKLLMR'22]:

$$H(\mathbf{pk}_0, \mathbf{pk}_2) = \mathbf{h} = \mathbf{A} \cdot \begin{pmatrix} \mathbf{G}^{-1}(\mathbf{h}_0) \\ \mathbf{G}^{-1}(\mathbf{h}_1) \end{pmatrix}$$

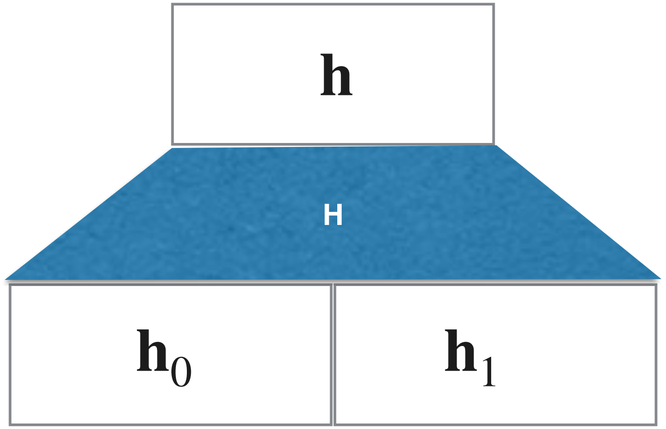


New Recryption Algorithm [DKLLMR'22]:

$$H(\mathbf{pk}_0, \mathbf{pk}_2) = \mathbf{h} = \mathbf{A} \cdot \begin{pmatrix} \mathbf{G}^{-1}(\mathbf{h}_0) \\ \mathbf{G}^{-1}(\mathbf{h}_1) \end{pmatrix}$$

$$\mathbf{c} \approx \mathbf{s} \cdot (\mathbf{A} + (\mathbf{G} \| 0))$$

$$c_1 \approx \mathbf{s} \cdot \mathbf{h} + \frac{q}{2} m$$

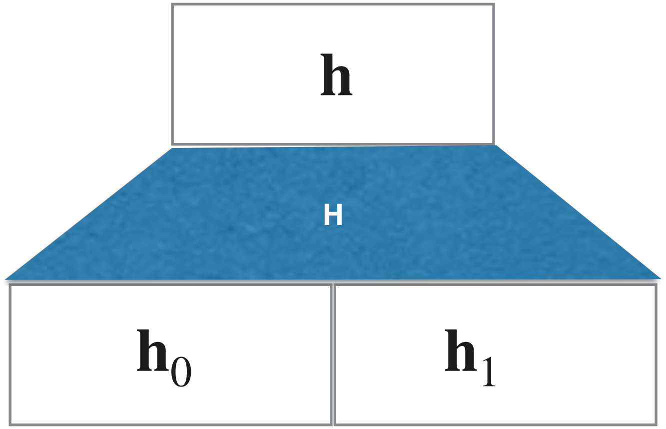


New Recryption Algorithm [DKLLMR'22]:

$$H(\mathbf{pk}_0, \mathbf{pk}_2) = \mathbf{h} = \mathbf{A} \cdot \begin{pmatrix} \mathbf{G}^{-1}(\mathbf{h}_0) \\ \mathbf{G}^{-1}(\mathbf{h}_1) \end{pmatrix}$$

$$\mathbf{c} \approx \mathbf{s} \cdot (\mathbf{A} + (\mathbf{G} \| 0))$$

$$c_1 \approx \mathbf{s} \cdot \mathbf{h} + \frac{q}{2} m$$

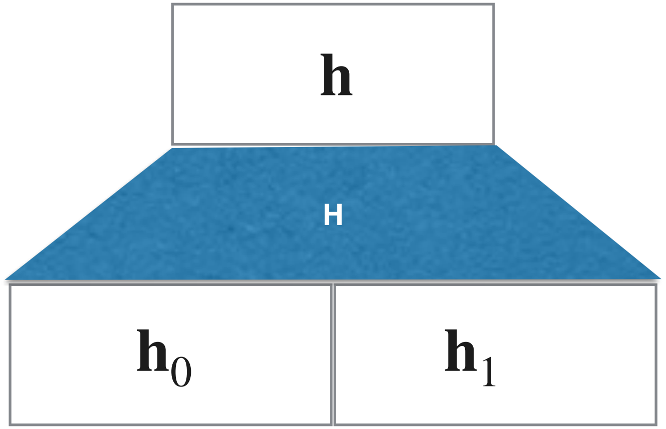


New Recryption Algorithm [DKLLMR'22]:

$$H(\mathbf{pk}_0, \mathbf{pk}_2) = \mathbf{h} = \mathbf{A} \cdot \begin{pmatrix} \mathbf{G}^{-1}(\mathbf{h}_0) \\ \mathbf{G}^{-1}(\mathbf{h}_1) \end{pmatrix}$$

$$\mathbf{c} \approx \mathbf{s} \cdot (\mathbf{A} + (\mathbf{G} \| 0))$$

$$c_1 \approx \mathbf{s} \cdot \mathbf{h} + \frac{q}{2} m$$



$$c_1' = c_1 - \mathbf{c} \begin{pmatrix} \mathbf{G}^{-1}(\mathbf{h}_0) \\ \mathbf{G}^{-1}(\mathbf{h}_1) \end{pmatrix} \approx \mathbf{s} \cdot \mathbf{h}_0 + \frac{q}{2} m$$

Efficient Private Laconic OT

Efficient Private Laconic OT

- Generally: Laconic OT ciphertext c either reveals query index i or decryption has linear complexity

Efficient Private Laconic OT

- Generally: Laconic OT ciphertext c either reveals query index i or decryption has linear complexity
- [DHMW24]: Private laconic OT (hiding query index i) and polylogerithmic decryption complexity

Efficient Private Laconic OT

- Generally: Laconic OT ciphertext c either reveals query index i or decryption has linear complexity
- [DHMW24]: Private laconic OT (hiding query index i) and polylogerithmic decryption complexity
- Leverages recent breakthrough on doubly efficient private information retrieval [LMW23]

Private Laconic OT with Preprocessing

Private Laconic OT with Preprocessing

- Preprocessing model: Sender and receiver compute and store a "correlations" before e.g. sender gets his input

Private Laconic OT with Preprocessing

- Preprocessing model: Sender and receiver compute and store a "correlations" before e.g. sender gets his input
- Emerging line of research in sublinear PIR with preprocessing following [CK20].

Private Laconic OT with Preprocessing

- Preprocessing model: Sender and receiver compute and store a “correlations” before e.g. sender gets his input
- Emerging line of research in sublinear PIR with preprocessing following [CK20].
- Very efficient, online phase uses only symmetric key crypto

Private Laconic OT with Preprocessing

- Preprocessing model: Sender and receiver compute and store a "correlations" before e.g. sender gets his input
- Emerging line of research in sublinear PIR with preprocessing following [CK20].
- Very efficient, online phase uses only symmetric key crypto
- [BDHL24]: private laconic OT with preprocessing. Also only using symmetric key crypto in online phase

Conclusion

Conclusion

- Laconic Cryptography: Secure computation on **LARGE** data with *small* communication in 2 messages

Conclusion

- Laconic Cryptography: Secure computation on **LARGE** data with *small* communication in 2 messages
- Beyond succinct communication: sublinear computation

Conclusion

- Laconic Cryptography: Secure computation on **LARGE** data with *small* communication in 2 messages
- Beyond succinct communication: sublinear computation
- Unexpected Applications: IBE, RBE, Self-Revealing Encryption

Conclusion

- Laconic Cryptography: Secure computation on **LARGE** data with *small* communication in 2 messages
- Beyond succinct communication: sublinear computation
- Unexpected Applications: IBE, RBE, Self-Revealing Encryption
- Until recently: Mostly theoretical progress

Conclusion

- Laconic Cryptography: Secure computation on **LARGE** data with *small* communication in 2 messages
- Beyond succinct communication: sublinear computation
- Unexpected Applications: IBE, RBE, Self-Revealing Encryption
- Until recently: Mostly theoretical progress
- Now: Breaking the wall to practical usefulness, new ideas such as preprocessing