# Secret Sharing for General Access Structures

Benny Applebaum

Tel Aviv University

BIU Winter-School of Information-Theoretic Cryptography

February 2020

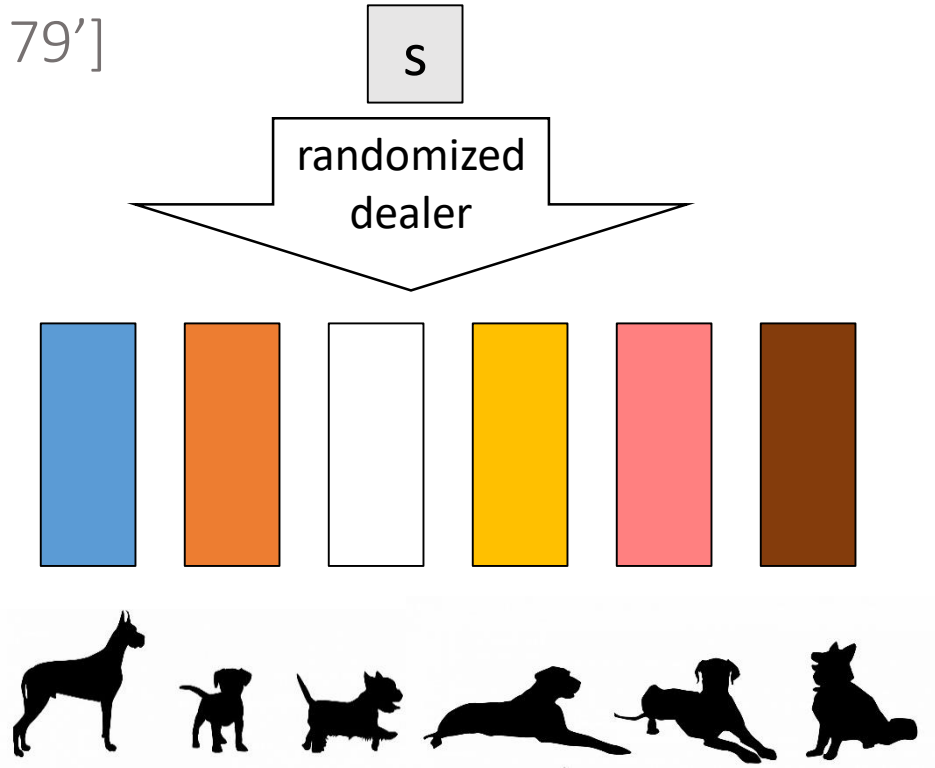
# Threshold Secret Sharing [Shamir 79, Blakley 79']

**(t-out-of-n)-SS:** Split a secret $s$ to $n$ shares such that:

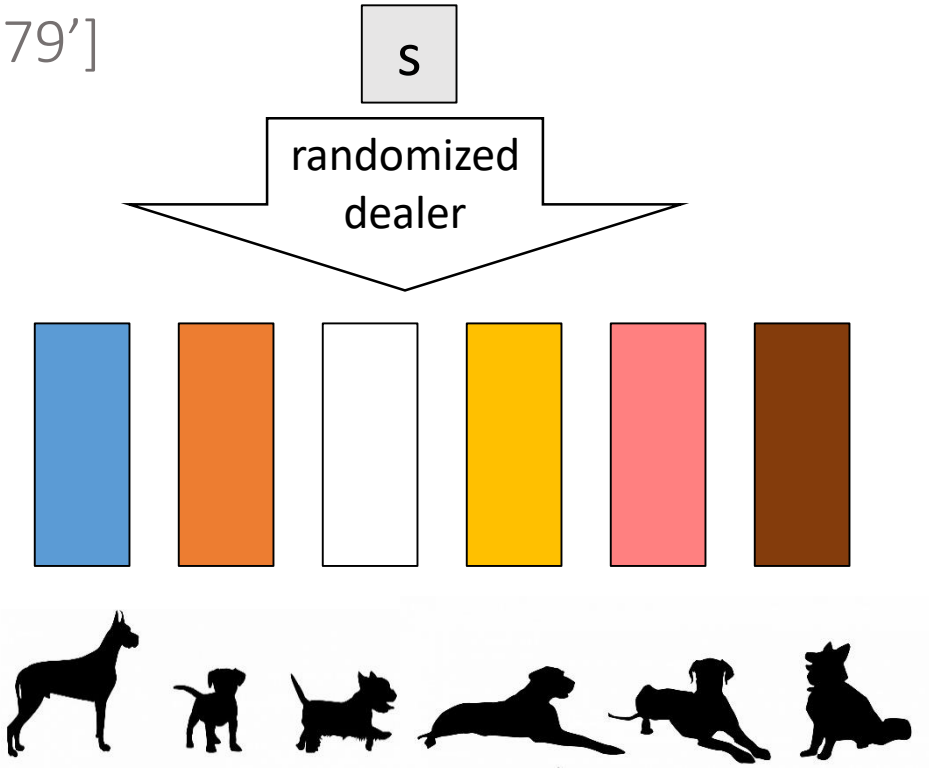
- $\geq t$ shares are enough to reconstruct the secret
- $< t$ shares – parties learn nothing about the secret

Basic primitive in information theoretic cryptography

Huge number of applications



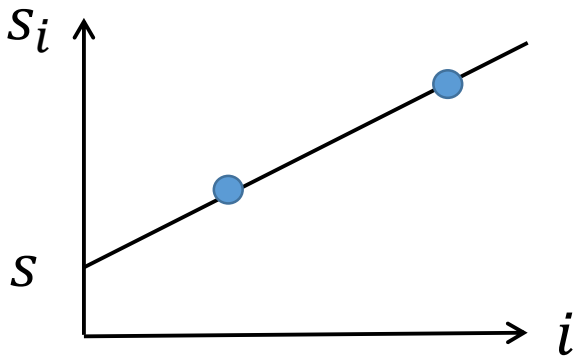
# Threshold Secret Sharing [Shamir 79, Blakley 79']




**Example**: 2-out-of-n secret sharing

- $s \in \{0,1\}$
- Sample $a \in \{1, \ldots, p-1\}$
  where $p > n$ is prime (e.g., $p = 7$)
- Set $s_i = s + a * i$

**Correctness**: Any pair can recover $s$ (via interpolation)
**Privacy**: Any singleton learns nothing

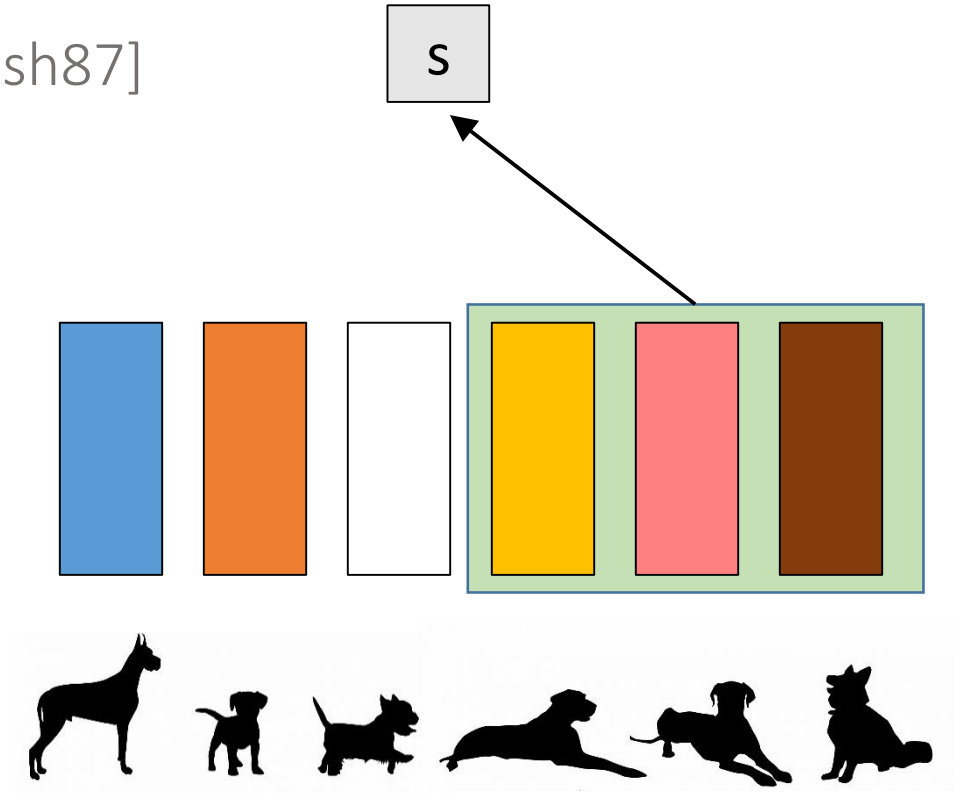


# Secret Sharing: Generalization [IttSaiNish87]

**Access structure** - A list $\mathcal{A}$ of authorized coalitions

Formally: SS scheme for access structure $\mathcal{A}$ satisfies:

**Correctness**: If $A \in \mathcal{A}$ is authorized then
∃ algorithm Recover: $(s_i: \forall i \in A)$ output $s$



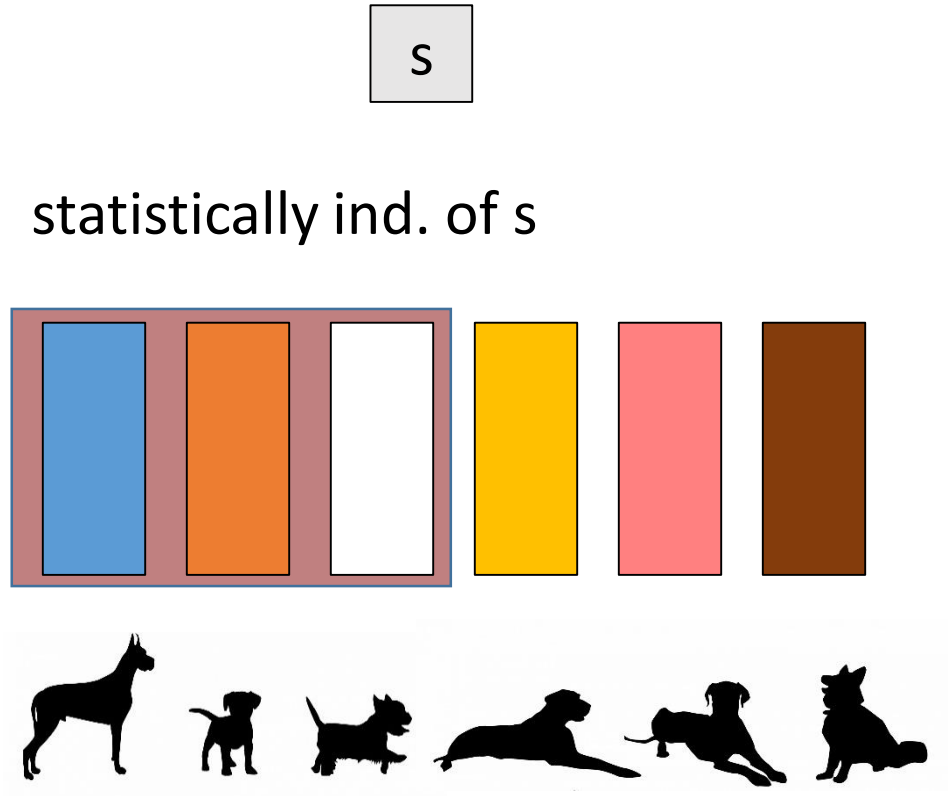
# Secret Sharing: Formalization

**Access structure** - A list $\mathcal{A}$ of authorized coalitions

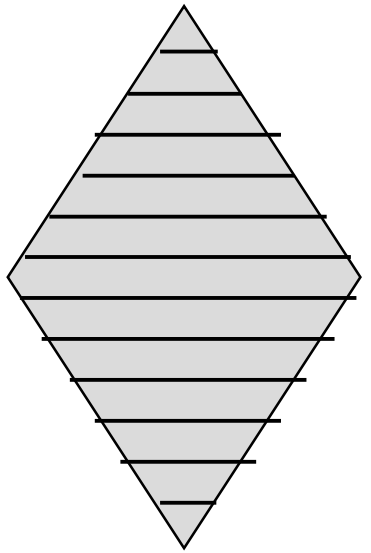
Formally: SS scheme for access structure $\mathcal{A}$ satisfies:

**Correctness**: If $A \in \mathcal{A}$ is authorized then
∃ algorithm Recover: $(s_i: \forall i \in A)$ output $s$

**Privacy:** If A is unauthorized
the tuple $(s_i: \forall i \in A)$ is distributed independently of $s$



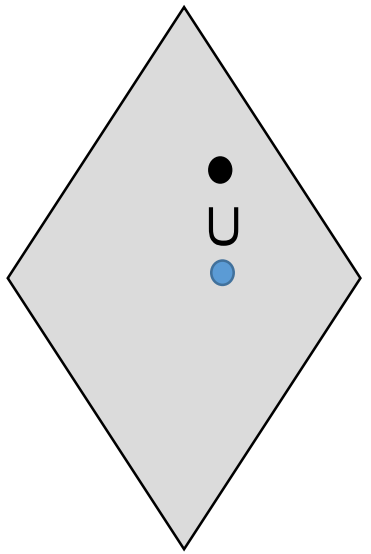
# Access Structures




**Lattice of subsets**

# Access Structures

**Monotone:** A super-set of an authorized set is also authorized




ALL={1,…,n}

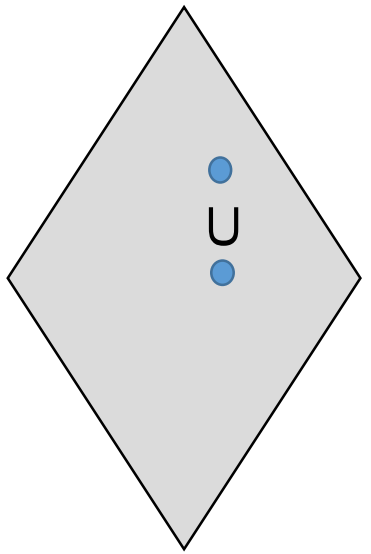
n/2-subsets

Singletons

Empty set

# Access Structures

**Monotone:** A super-set of an authorized set is also authorized




ALL={1,…,n}

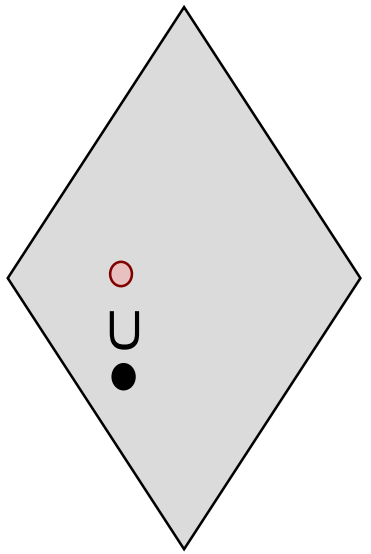
n/2-subsets

Singletons

Empty set

# Access Structures

**Monotone:** A sub-set of an unauthorized set is also unauthorized




ALL={1,…,n}

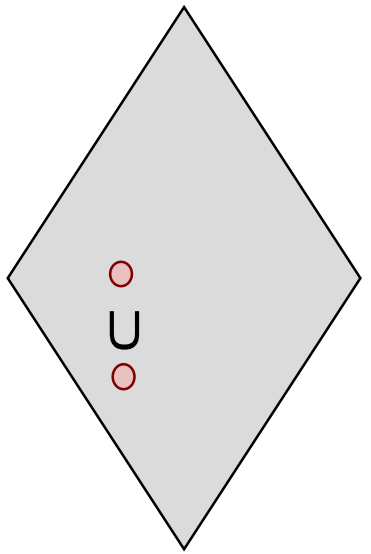
n/2-subsets

Singletons

Empty set

# Access Structures

**Monotone:** A sub-set of an unauthorized set is also unauthorized




ALL={1,…,n}

n/2-subsets

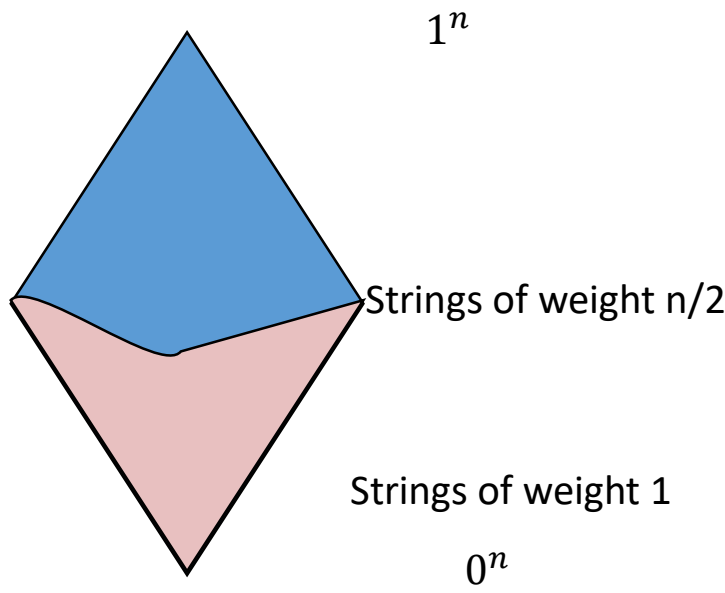
Singletons

Empty set

# Access Structures

**Monotone:** The characteristic function of $\mathcal{A}$ is **monotone**

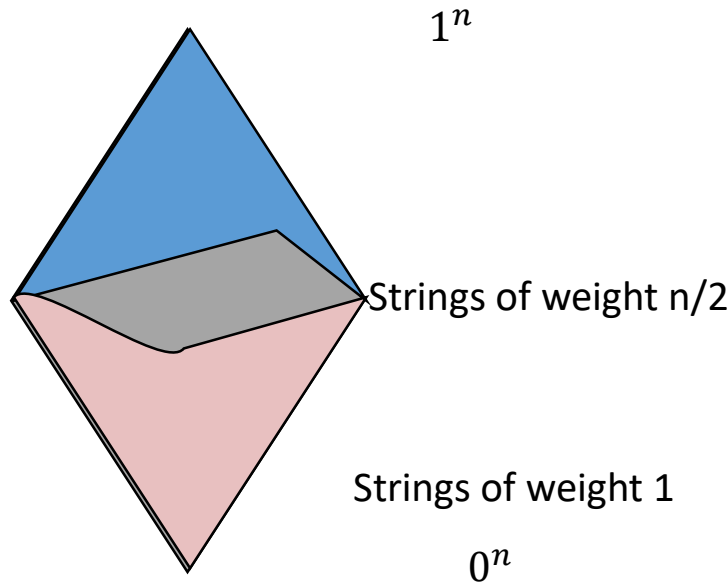
- $f_A: \{0,1\}^n \rightarrow \{0,1\}$



# Access Structures

**Monotone:** The characteristic function of $\mathcal{A}$ is **monotone**

- $f_A: \{0,1\}^n \rightarrow \{0,1\}$




**"Promise"** access structure: For some sets "don't care"

- $\mathcal{A}$ is given by a **partial monotone** function

**Complexity($\mathcal{A}$):** Minimal total length of all share size among all schemes that realize $\mathcal{A}$

- Complexity(n-out-of-n) $= 1$ bits
- Complexity(t-out-of-n) $\leq \log(n)$ bits (one field element per party)

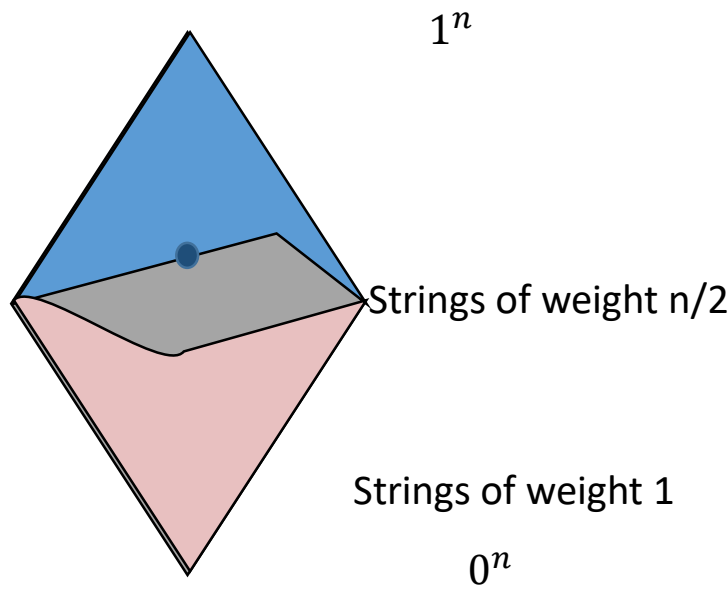
**Big open problem: Complexity of General Access Structures?**

# Simple constructions: DNF

**DNF:** For every minimal authorized set **A**

share **s** via |A|-out-of-|A| sharing

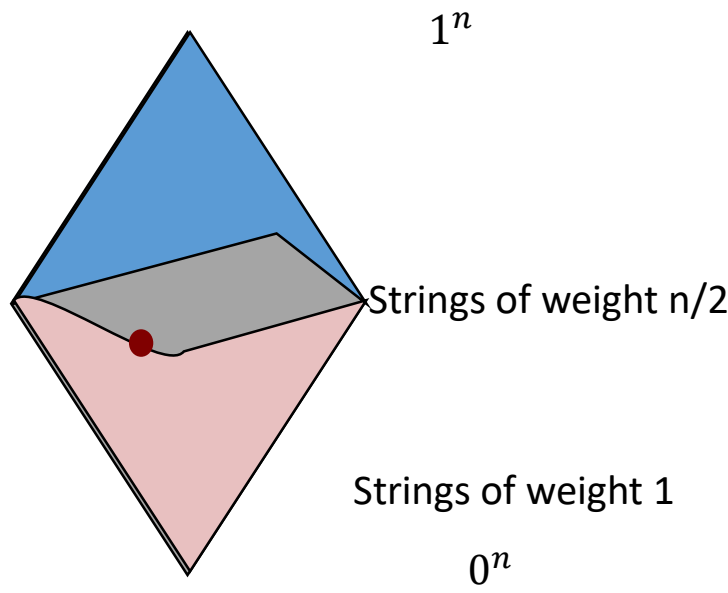
- Complexity: #min-authorized sets
- Worse-case: $2^n$



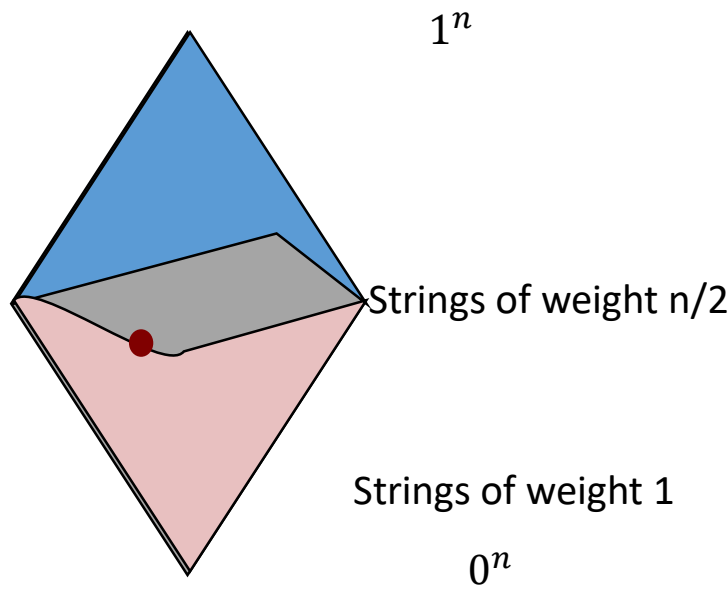
# Simple constructions: CNF

**CNF:** Share **s** among all maximal un-authorized sets **A**, $\boldsymbol{s = \sum_i s_A}$
give $s_A$ to all parties outside **A**

- Complexity: #max-unauthorized sets
- Worse-case: $2^n$



# Simple constructions: Monotone Formulas




**Write f as a monotone formula and SS recursively**

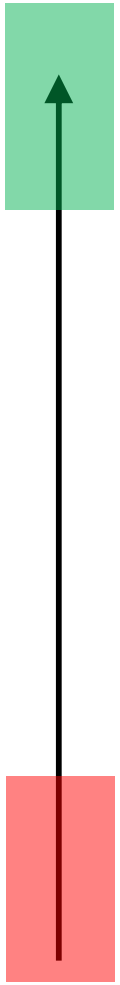
- Easy to handle AND gates and OR gates
- Complexity: Formula-size(f)
- Worse-case: $2^n$

Ex: Prove that DNF/CNF can be described as a special case of Formula construction.

**Q: Can we beat the $2^n$ upper-bound??**

# Complexity of Secret Sharing

The share size for n-party **general** access structure



**Upper Bounds:**

$2^n$ [IttSaiNish87]

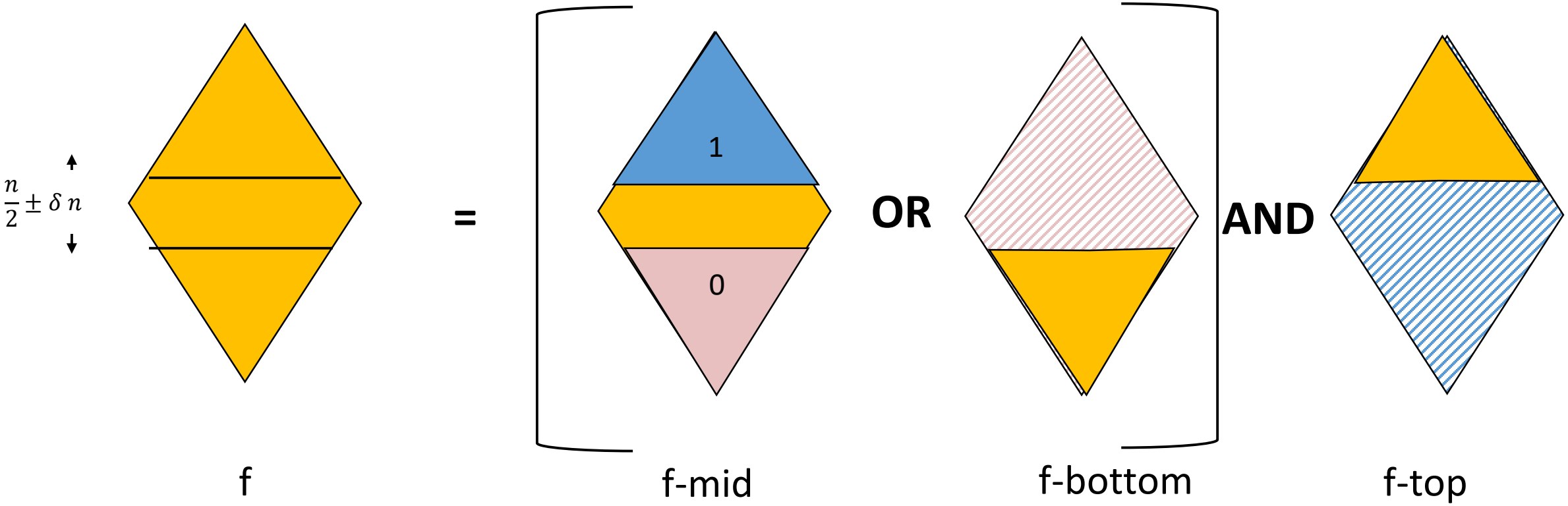
$2^{0.994n}$ [LiuVai18]

$2^{0.897n}$ [A-BieFarNirPet19]

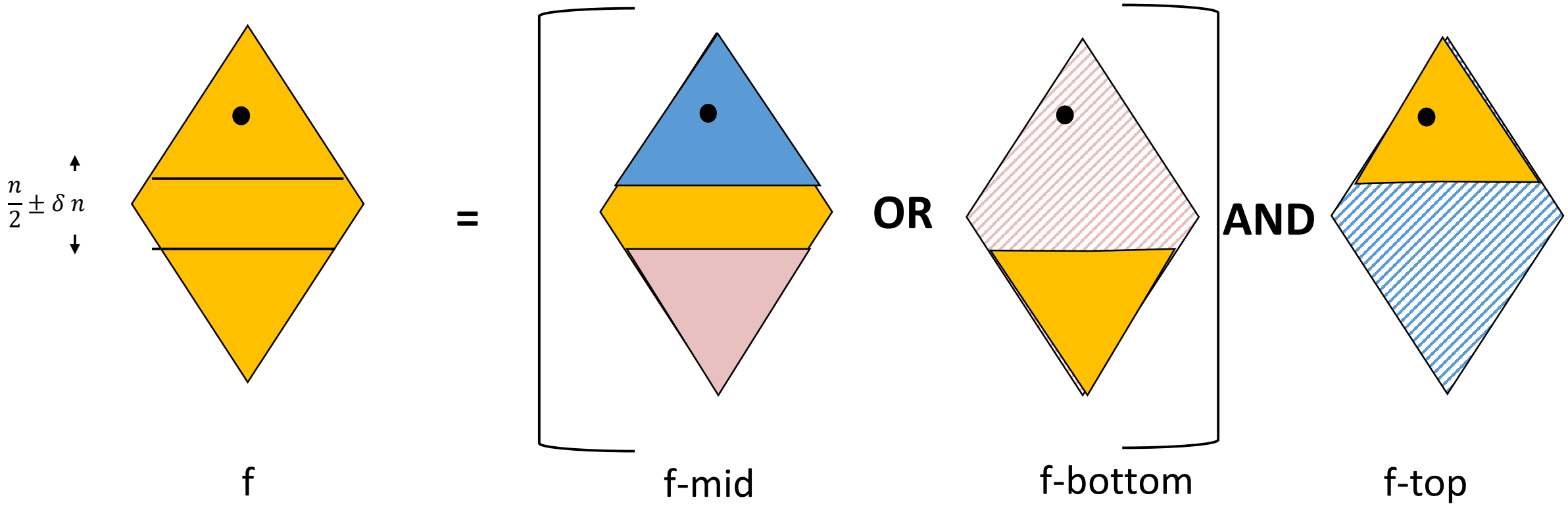
$2^{0.64n}$ [A-BieNirPet20]

**Lower Bound:** $\Omega\left(\frac{n}{\log(n)}\right)$ [C97]

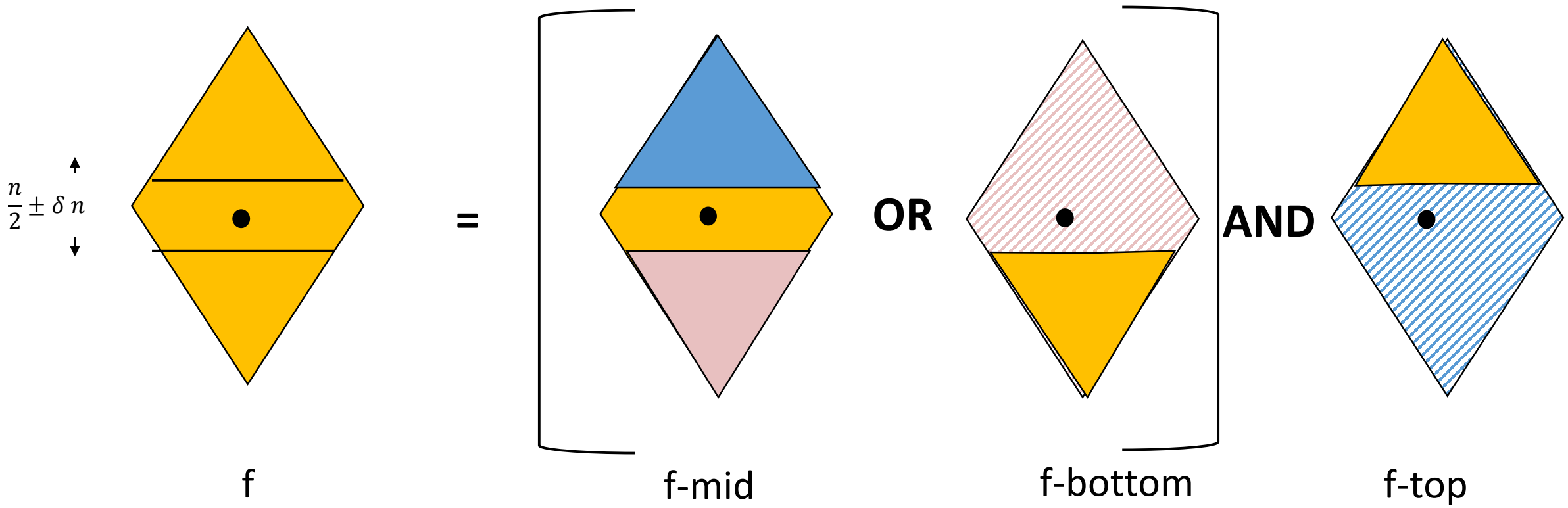
# The LV-decomposition



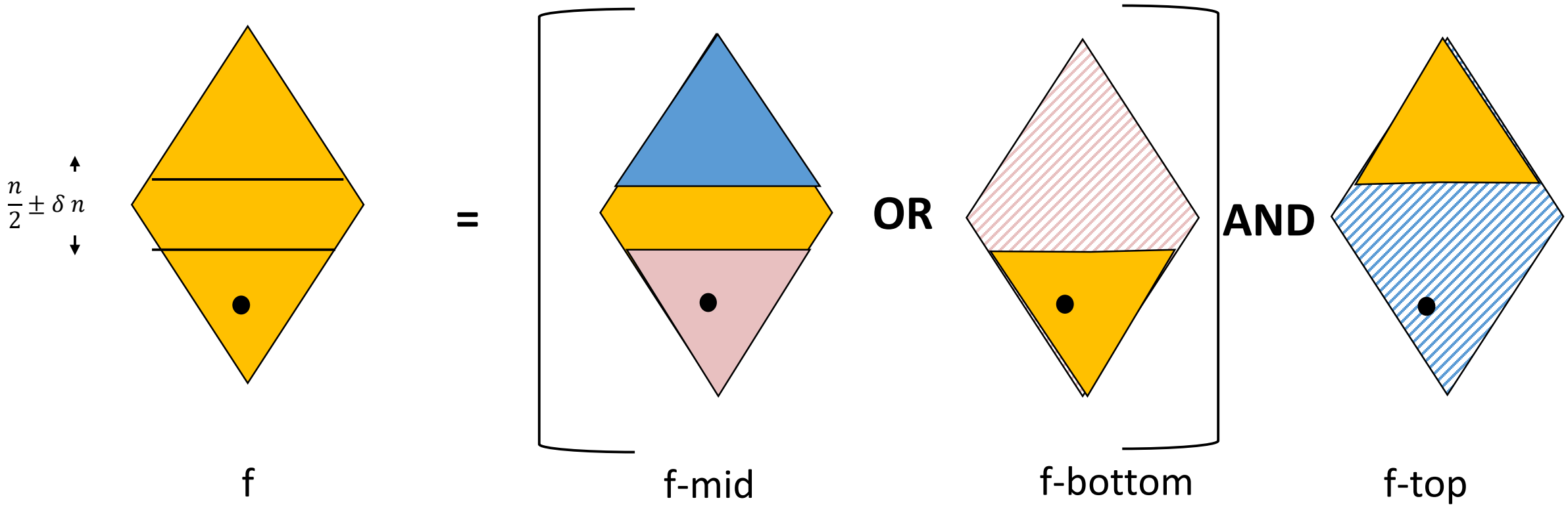
# The LV-decomposition



# The LV-decomposition

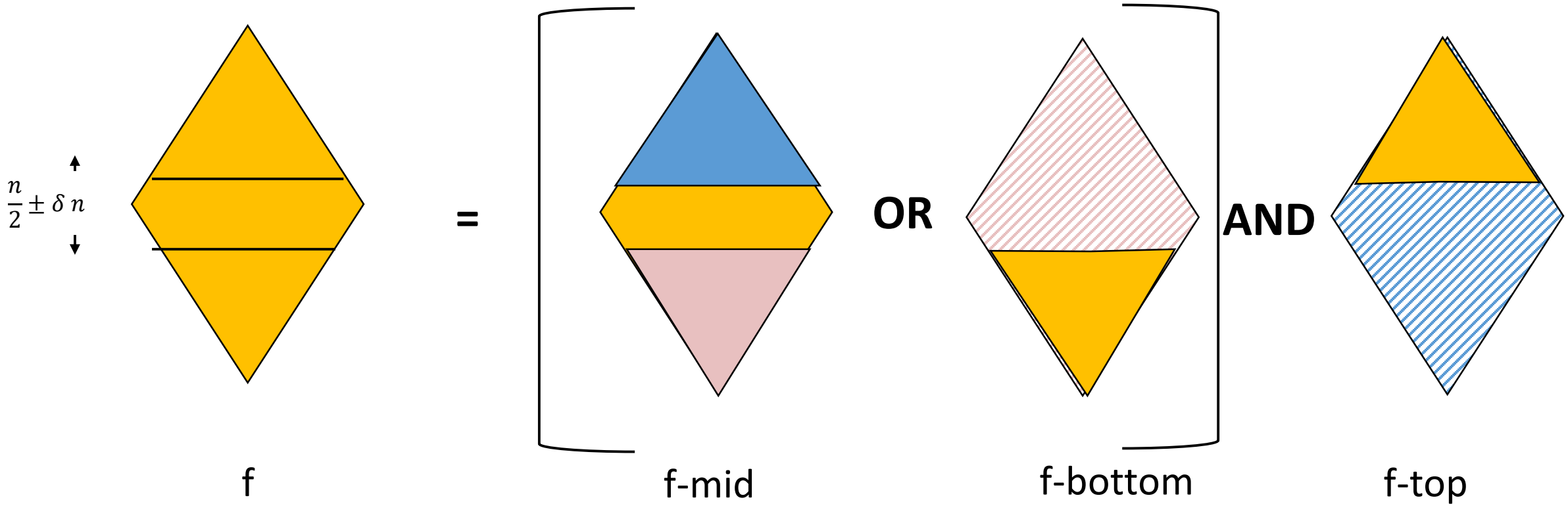


# The LV-decomposition



# The LV-decomposition

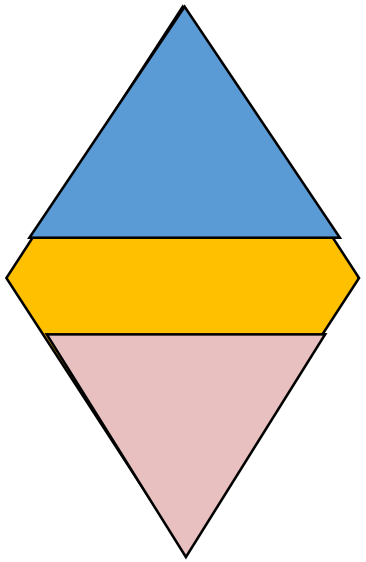
Extreme slices can be realized with non-trivial exponent



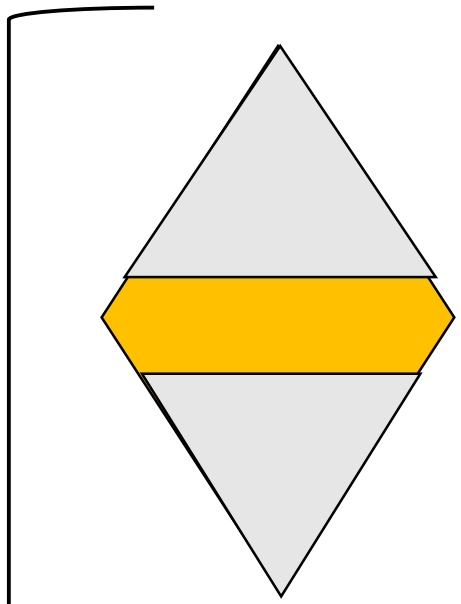
# Focus on mid-slice

Threshold access structures
(realized via Shamir)

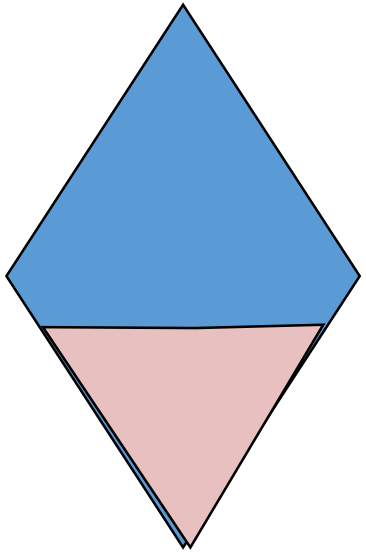
$\frac{n}{2} \pm \delta\, n$



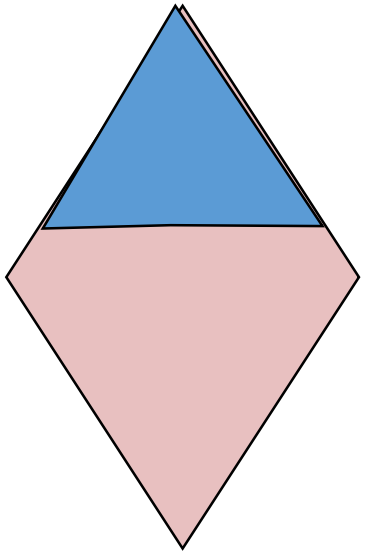
=



**AND**



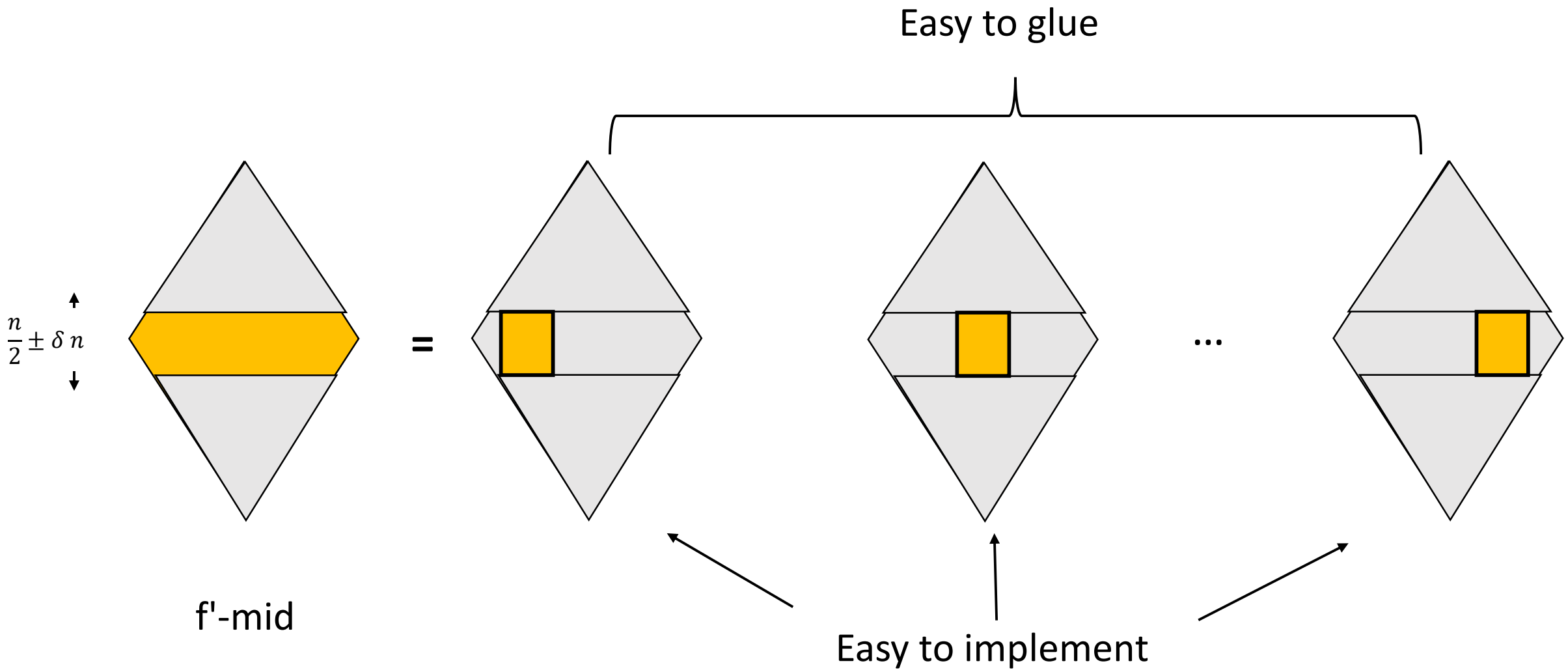
**OR**



f-mid

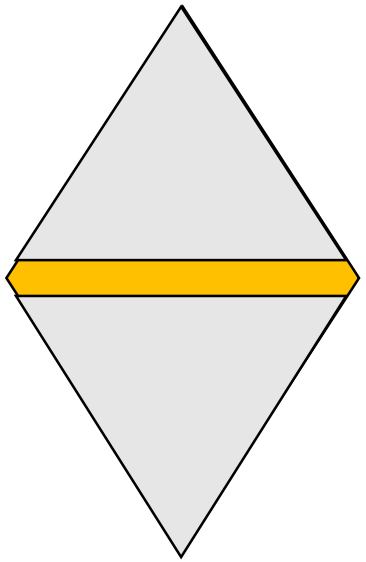
f'-mid

# What next? More partitions



# Focusing on a single slice

$\frac{n}{2}$

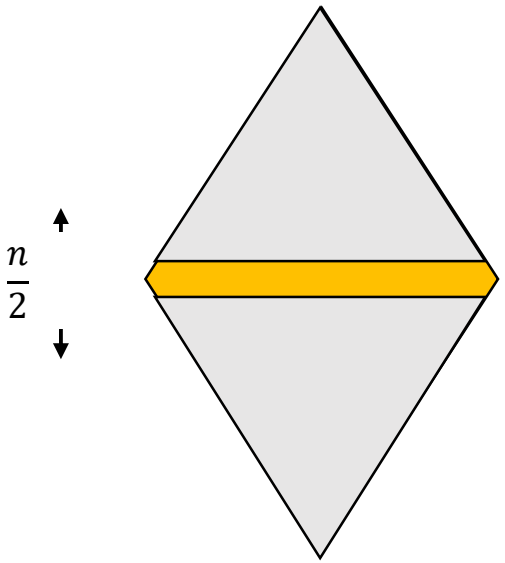


f'-mid

Care only about n/2-subsets

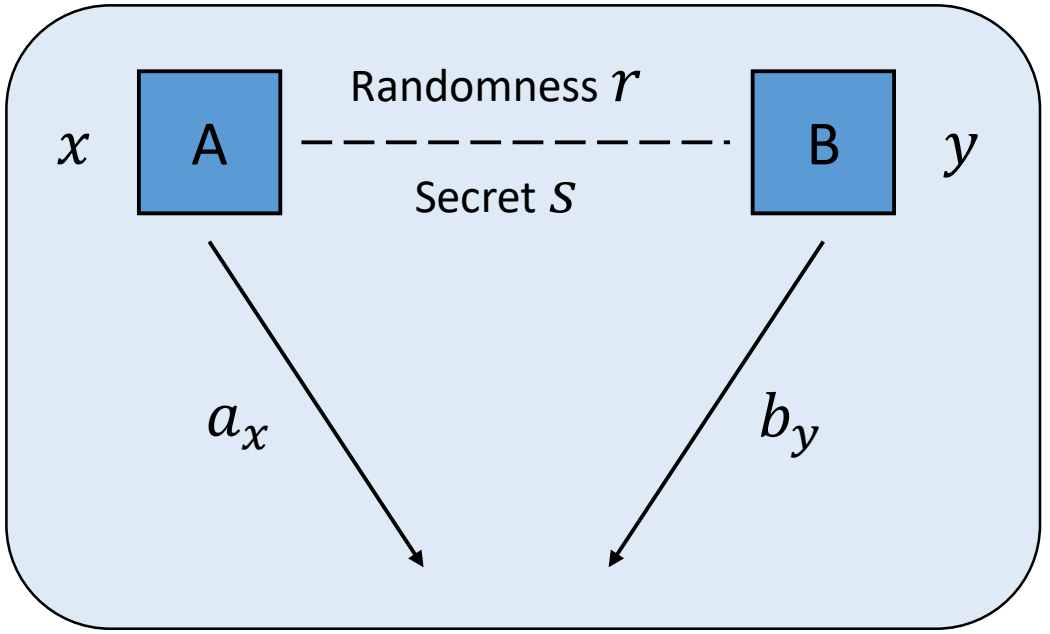
n parties

# Realizing n/2-uniform access structure via CDS




f'-mid

$$f:\{0,1\}^{n/2}\times\{0,1\}^{n/2}\to\{0,1\}$$



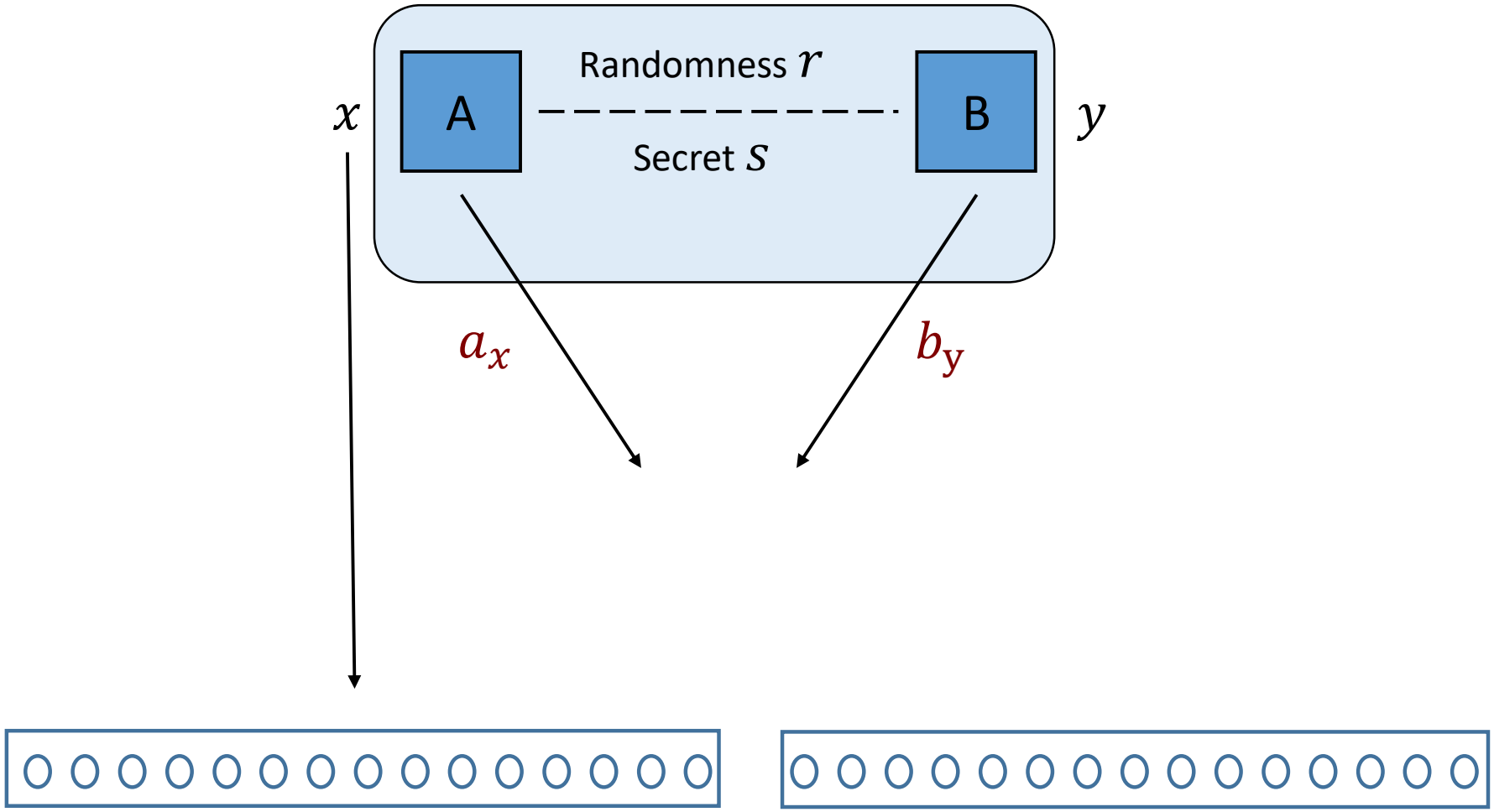

Messages $(a_x, b_y)$ reveal $s$ iff $f(x,y)=1$

**Reminder:** Linear-CDS: $2^{n/4}$

Non-linear $2^{o(n)}$ even for k parties

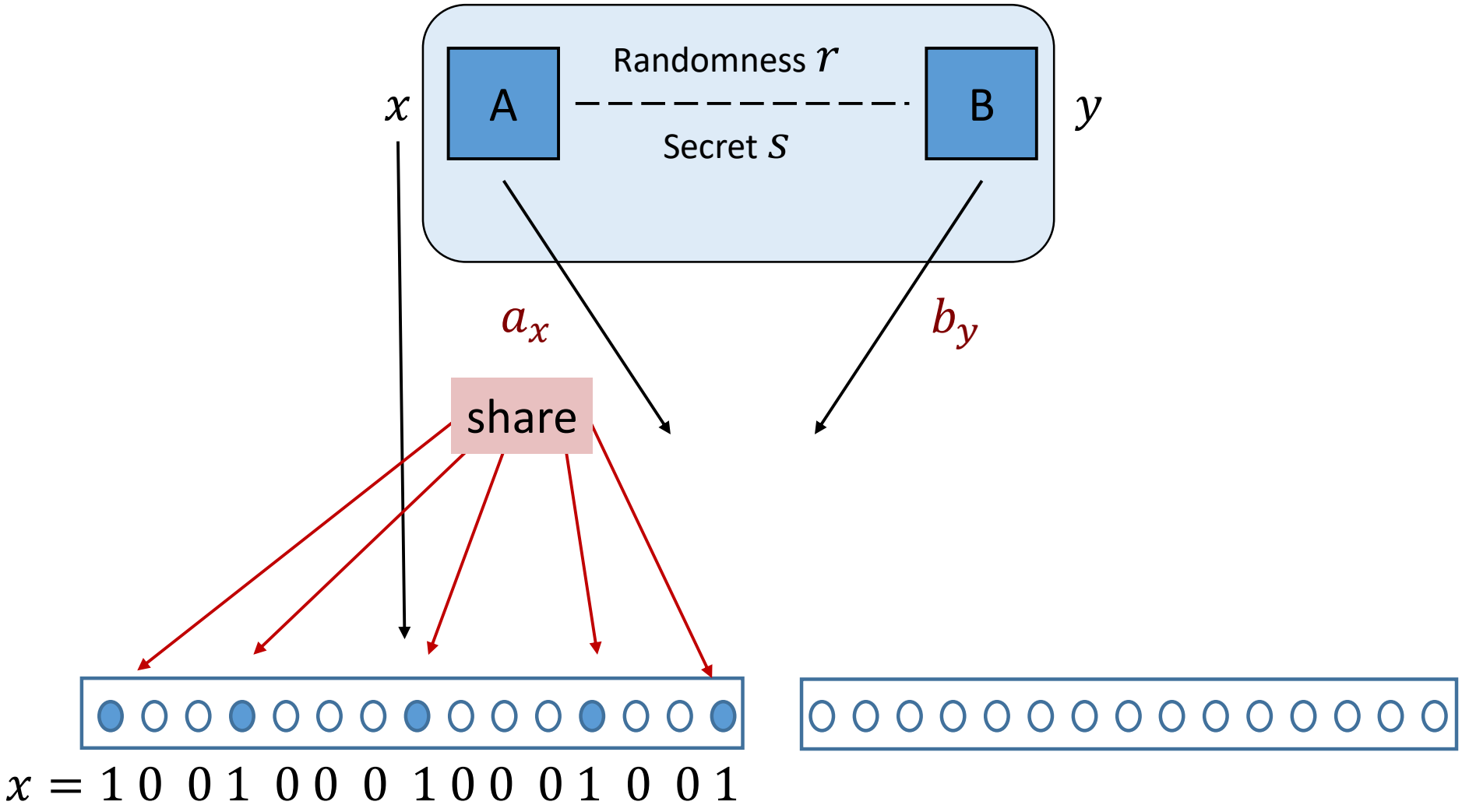
# Realizing n/2-uniform access structure via CDS

$$f: \{0,1\}^{n/2} \times \{0,1\}^{n/2} \rightarrow \{0,1\}$$



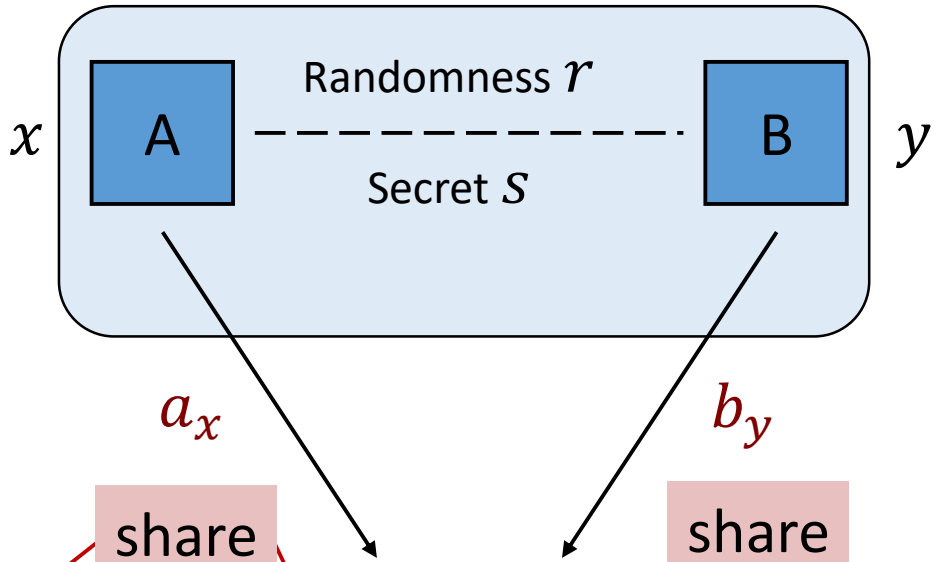
# Realizing n/2-uniform access structure via CDS

$$f: \{0,1\}^{n/2} \times \{0,1\}^{n/2} \to \{0,1\}$$



# Realizing n/2-uniform access structure via CDS

$f: \{0,1\}^{n/2} \times \{0,1\}^{n/2} \to \{0,1\}$




Good news:

Correctness holds

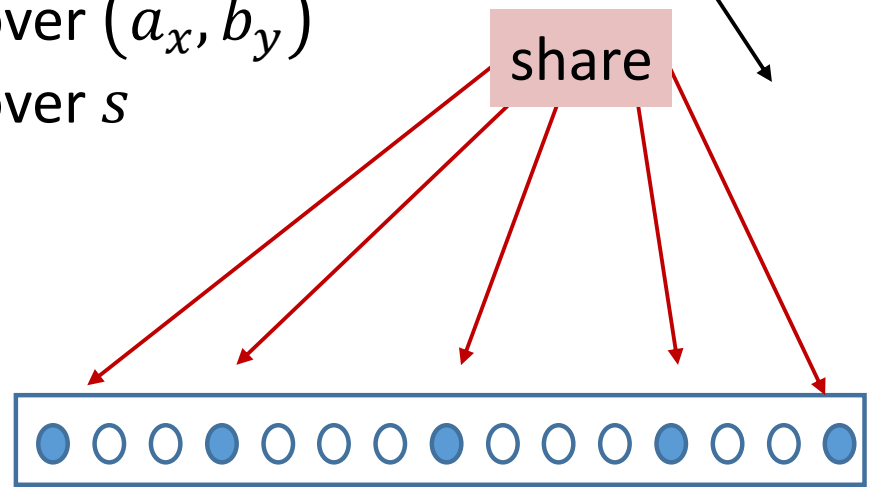
If $(x, y)$ is authorized

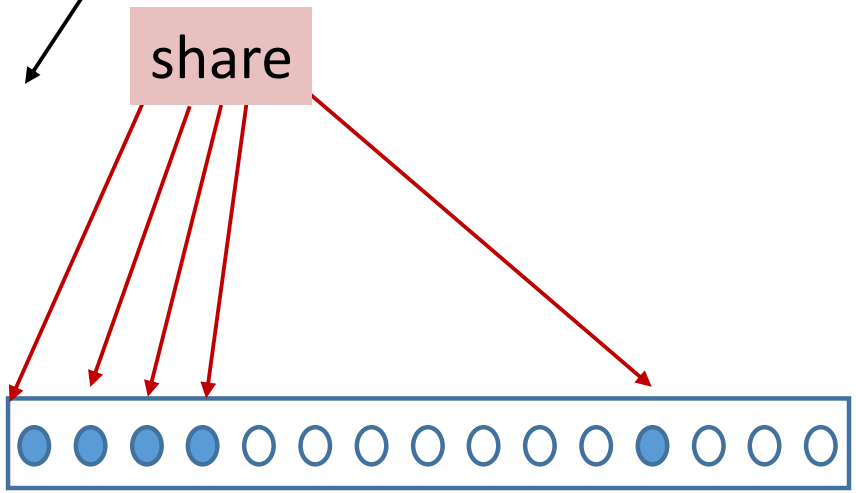
$\Rightarrow$can recover $(a_x, b_y)$

$\Rightarrow$can recover $s$

$a_x$

$b_y$



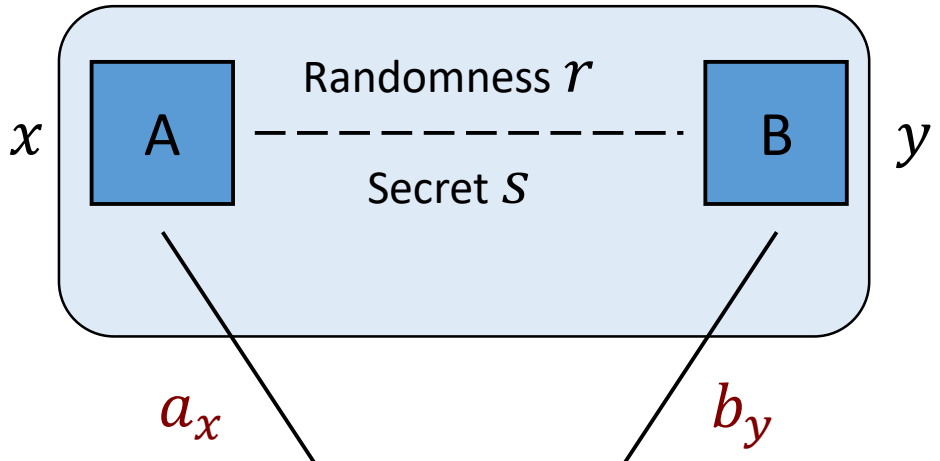




$x = 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1$

$1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0 = y$

# Realizing n/2-uniform access structure via CDS

$$f: \{0,1\}^{n/2} \times \{0,1\}^{n/2} \to \{0,1\}$$




**Privacy**:

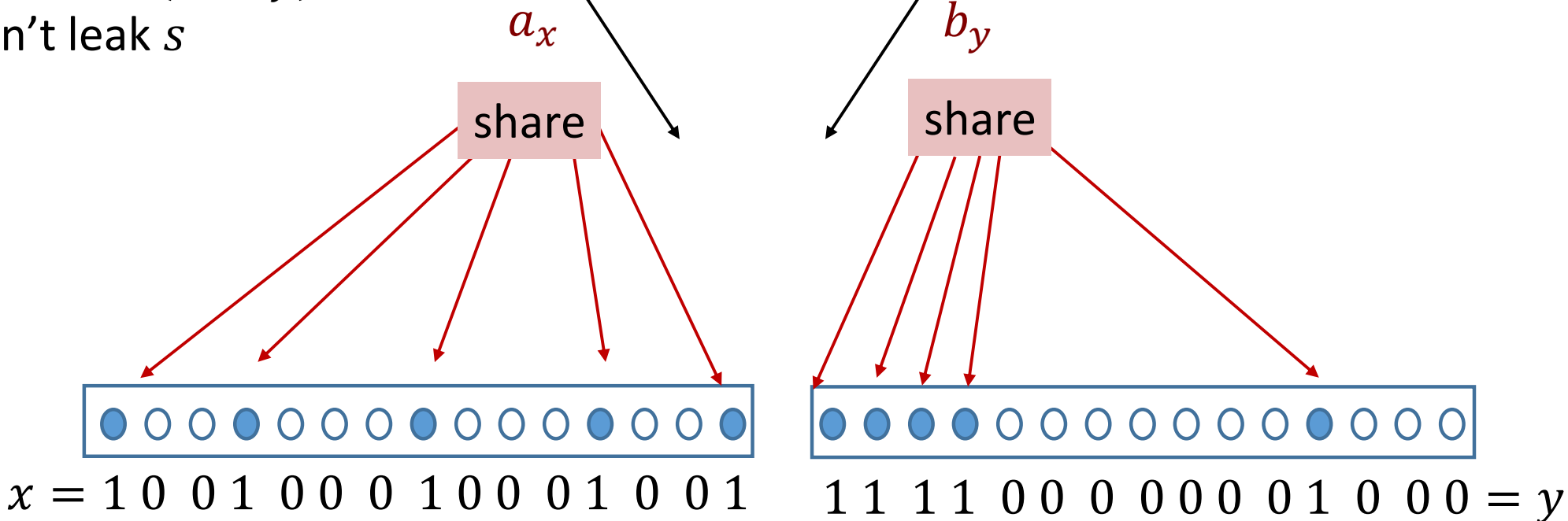
Suppose that $f(x, y) = 0$

$\Rightarrow$can recover $(a_x, b_y)$

$\Rightarrow$doesn't leak $s$

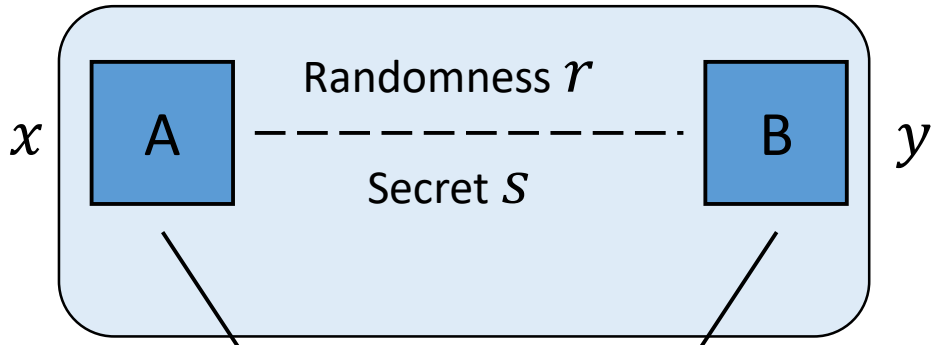
QED

?



# Realizing n/2-uniform access structure via CDS

$f: \{0,1\}^{n/2} \times \{0,1\}^{n/2} \to \{0,1\}$




$x$- parties know:

$a_{1000\ldots0}$

$a_{0001\ldots0}$

$a_{00\ldots1\ldots0}$

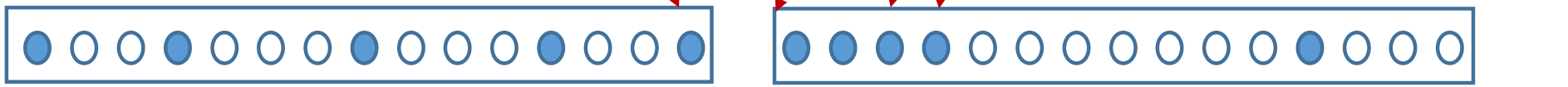
$a_{000\ldots01}$

$a_{10010\ldots0}$

…

$a_x$ $b_y$

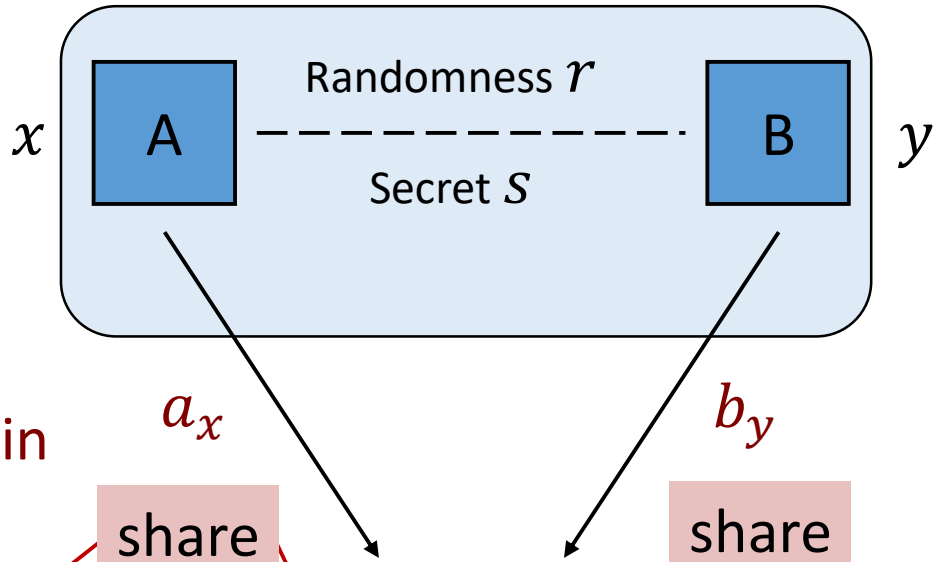
share share



$x = 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 0\ 0\ 1$ $1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 0\ 0\ 0\ 1\ 0\ 0\ 0 = y$

# Realizing n/2-uniform access structure via CDS

$f: \{0,1\}^{n/2} \times \{0,1\}^{n/2} \to \{0,1\}$

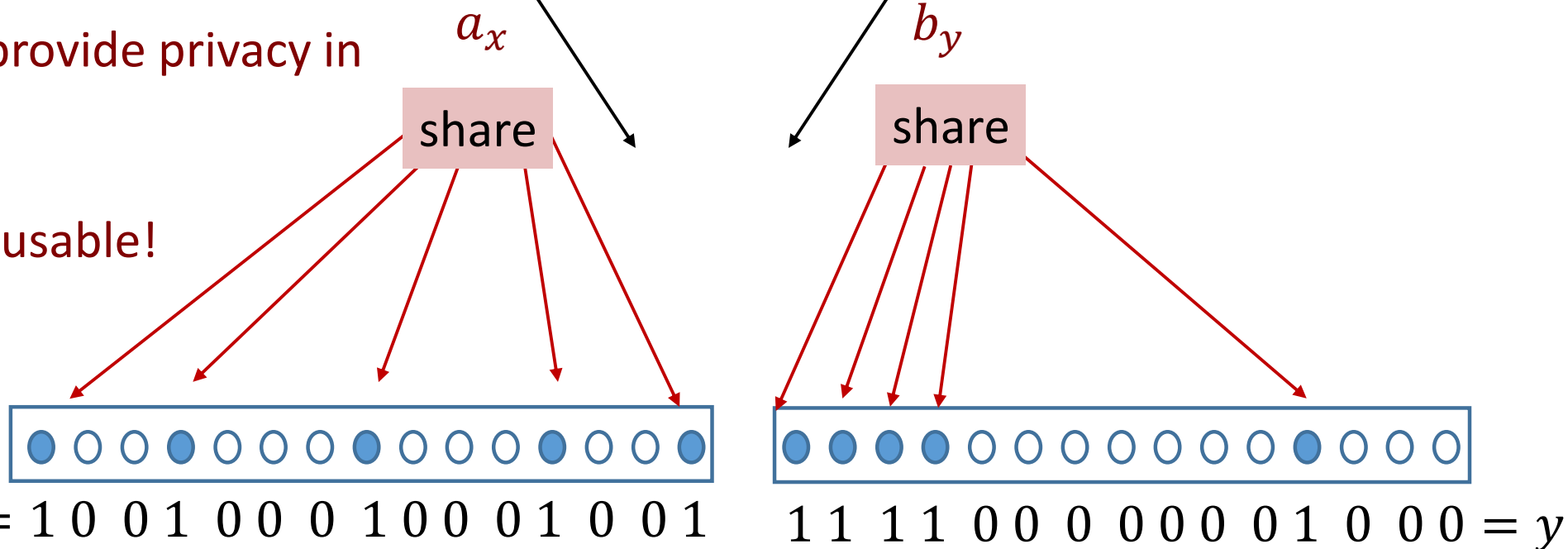



$x$- parties know:

$a_{x'}$ for every $x' \subseteq x$

CDS doesn't provide privacy in this case!

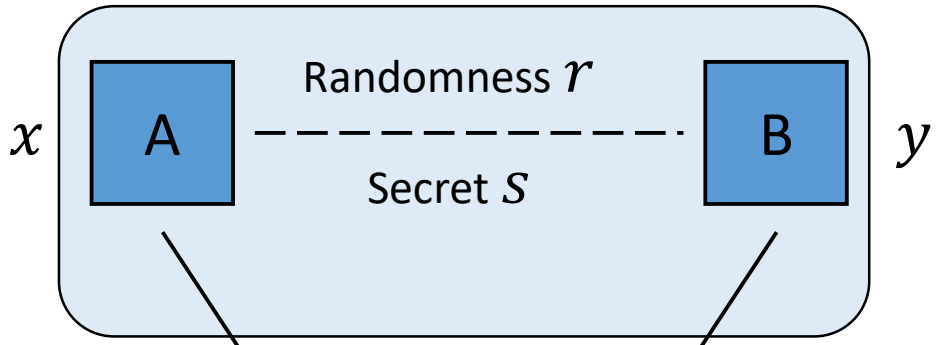
CDS is not re-usable!

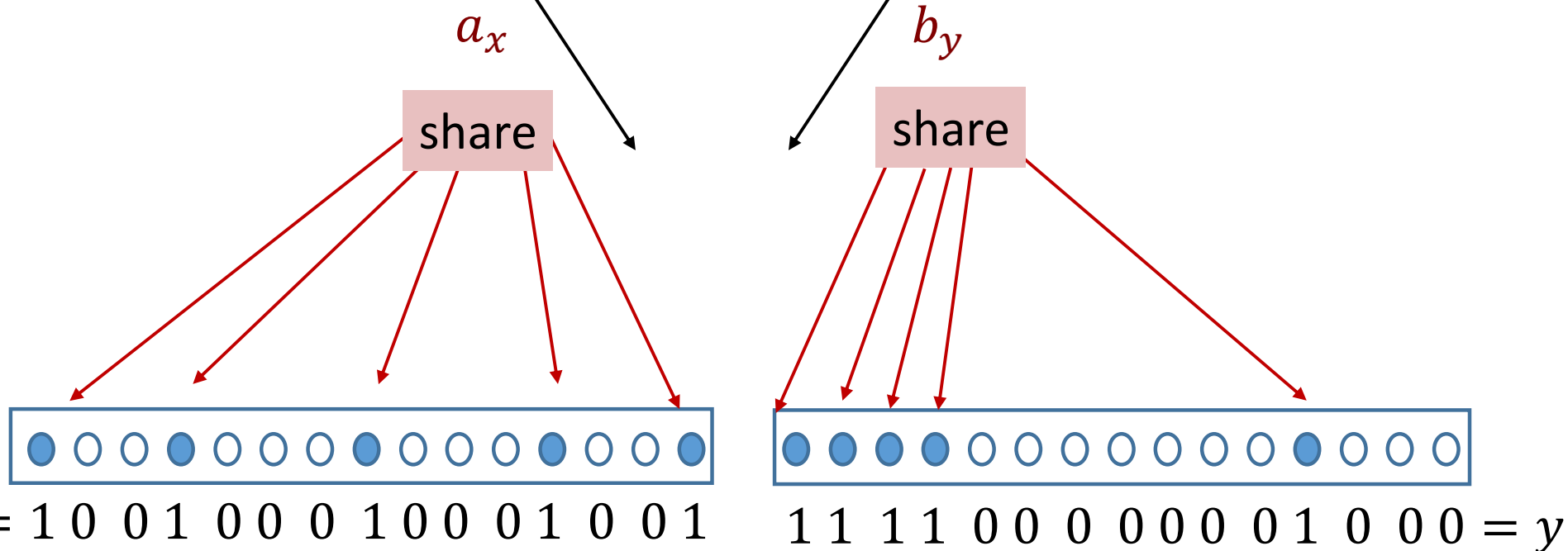


# Possible Sol: Restrict to sets of fixed size (anti-chain)

$$f: \{0,1\}^{n/2} \times \{0,1\}^{n/2} \to \{0,1\}$$

Avoid pairs $(x, x')$ for which $x' \subseteq x$



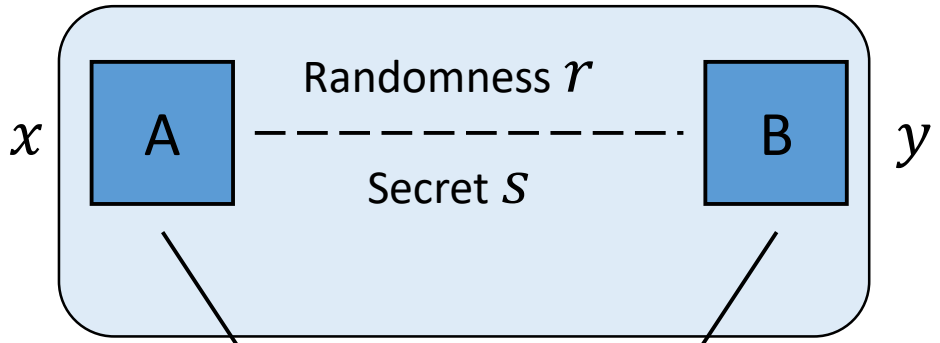



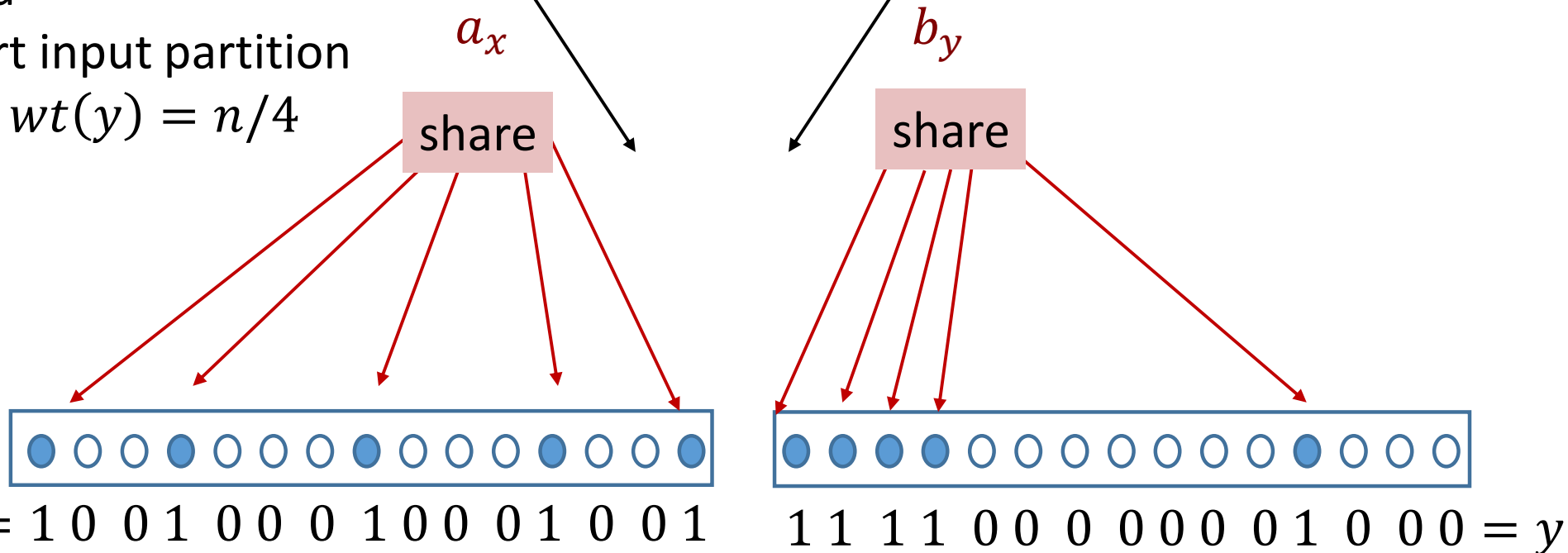
# Possible Sol: Restrict to sets of fixed size (anti-chain)

$$f: \{0,1\}^{n/2} \times \{0,1\}^{n/2} \to \{0,1\}$$

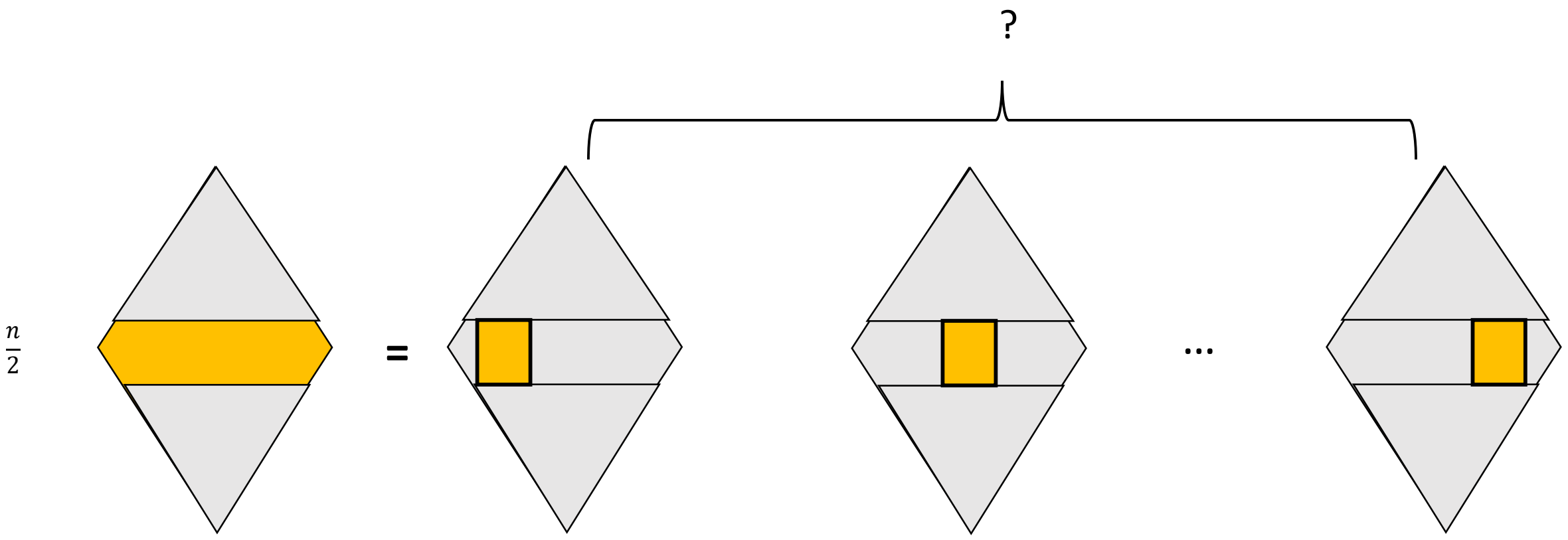
Apply only to $(x, y)$
of weight exactly $n/4$ each

$(x, y)$ is good
if balance wrt input partition
$wt(x) = wt(y) = n/4$



# Handle single layer via many partitions




f'-mid

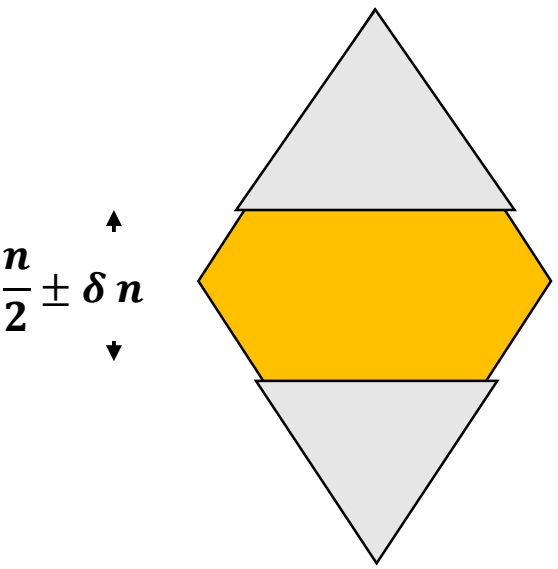
Inputs good for partition $(X_1, Y_1)$

Inputs good for partition $(X_2, Y_2)$

Inputs good for partition $(X_{\sqrt{n}}, Y_{\sqrt{n}})$

# From single-layer to many layers?

**Should treat inputs of different weights** $\mathbf{wt}(\mathbf{x}) \in (\mathbf{0.5} \pm \boldsymbol{\delta})\boldsymbol{n}$

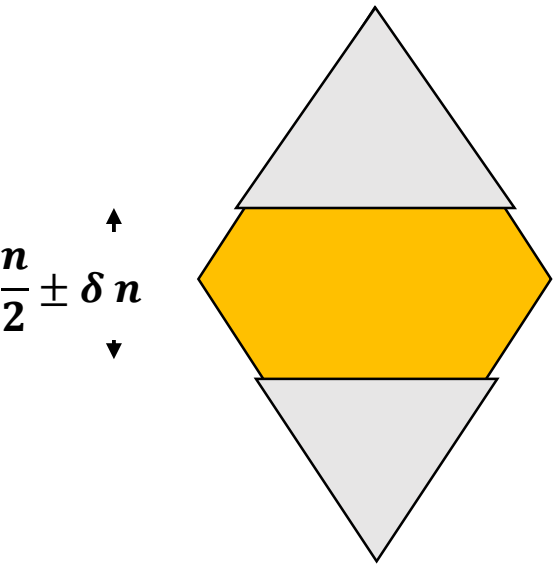



Solution 1 [LV'18]: More sophisticated decomposition

- Use k-multiparty CDS $k = n/5$
- Each block exactly half-occupied
- Special "overflow/underflow" block
- Exponential number of partitions

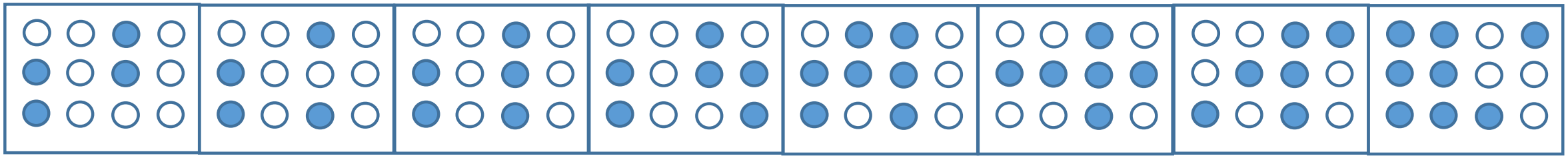
# From single-layer to many layers?

**Should treat inputs of different weights** $\mathbf{wt(x)} \in (\mathbf{0.5} \pm \boldsymbol{\delta})\boldsymbol{n}$



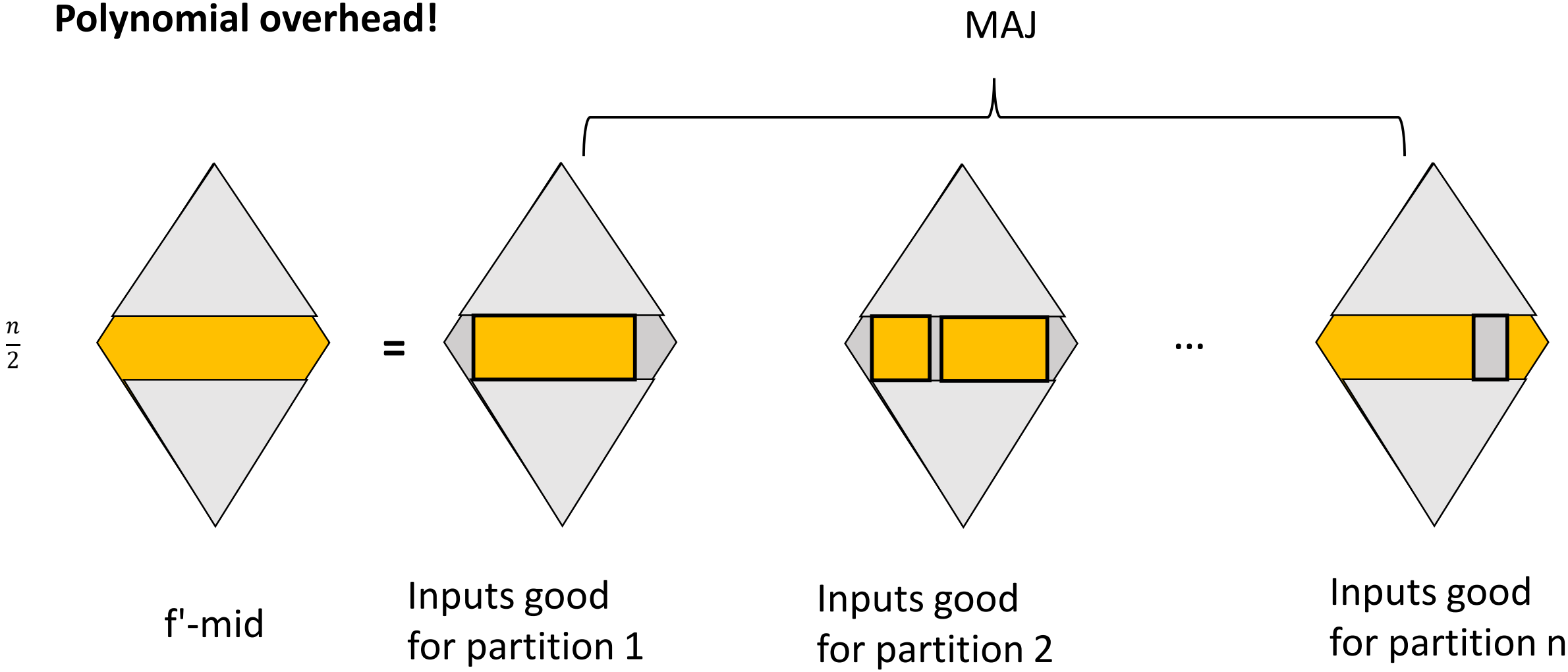

Solution 2 [ABNP'20]: Robust CDS

- Tolerates Limited re-usability
- Use k-multiparty CDS $k = \sqrt{n}$
- Each block should be $\left(\frac{1}{2} \pm \delta\right)\sqrt{n}$ occupied
- Linear number of partitions
- Easier gluing



# Approximately-Balanced Partitions

**Polynomial overhead!**

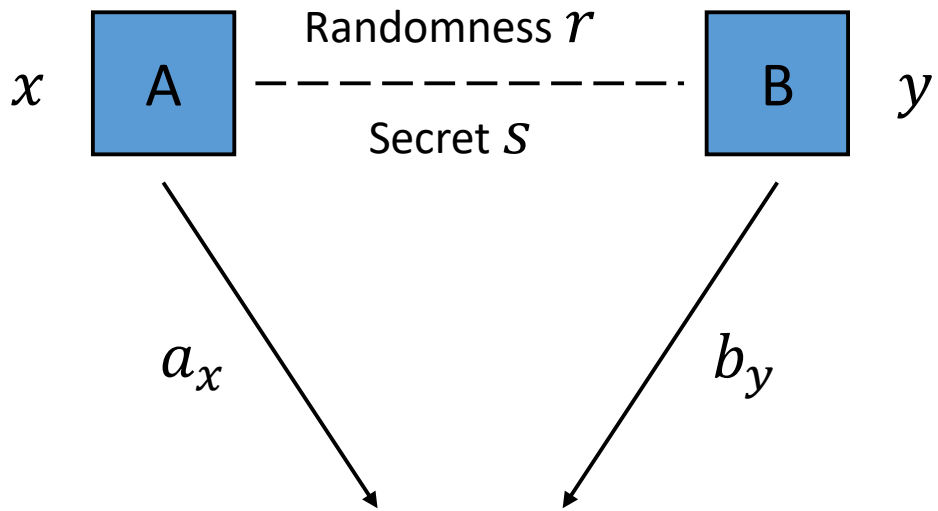


# Last missing component: Robust CDS

**General Transformation:**

- CDS => robust-CDS
- Exponential overhead
- Leads to best-known exponent

# Robust-CDS

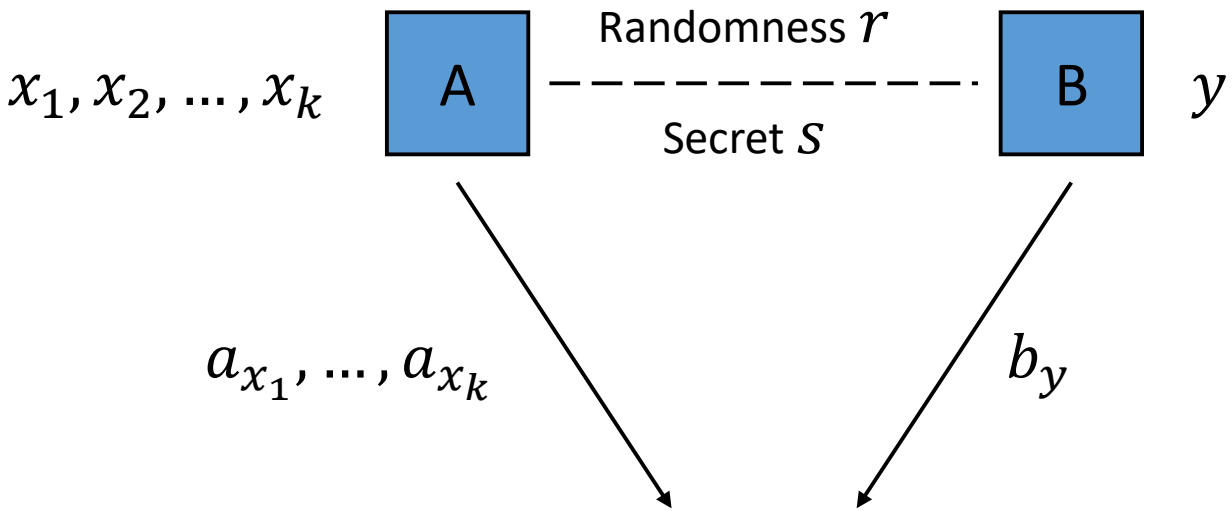



Messages $(a_x, b_y)$ reveal $s$ iff $f(x, y) = 1$

# Robust-CDS

**Params:**

- k= #simultaneous inputs
- L= #possible input vectors $\vec{x}$




**Robustness**: If $f(x_1, y) = \cdots = f(x_k, y) = 0$ secret remains hidden

- Need it for all parties simultaneously

# The Channel Immunization Problem

Sender

One-Time Secure

Receiver

N channels

# The Channel Immunization Problem

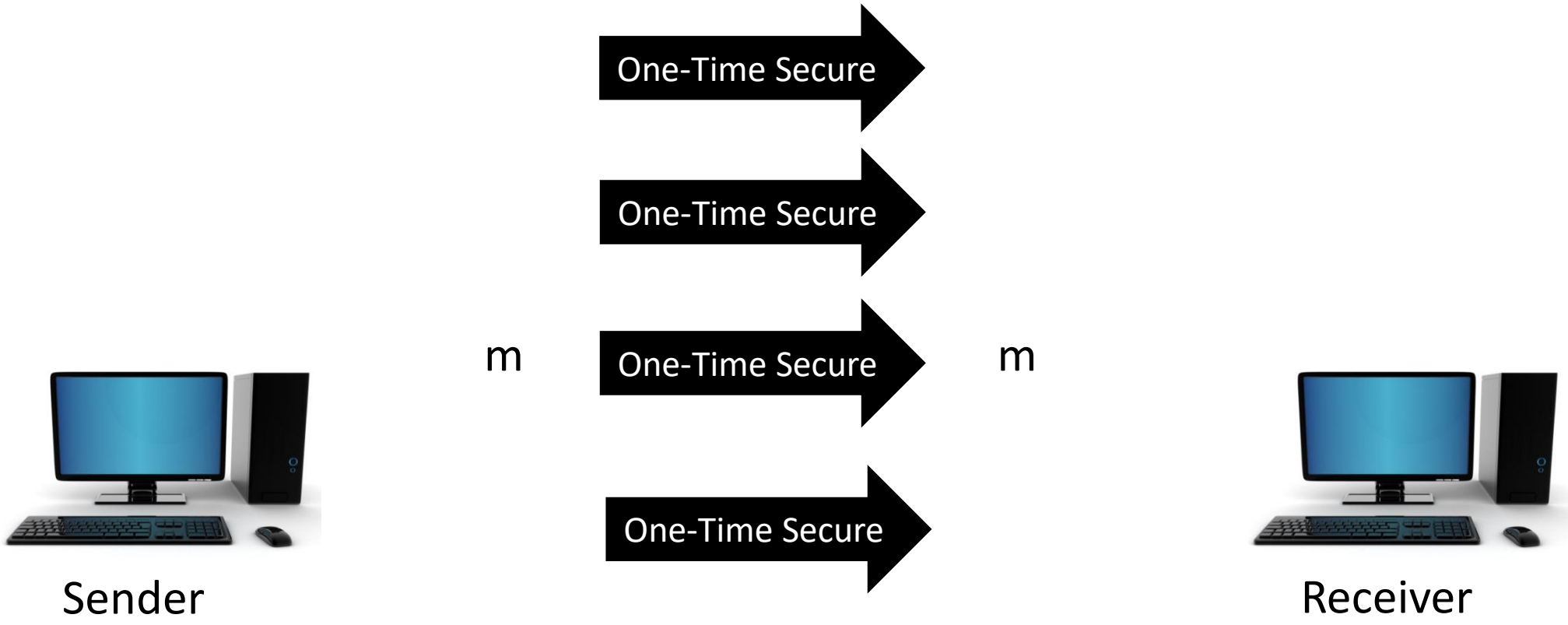
Sender

t-Time Secure

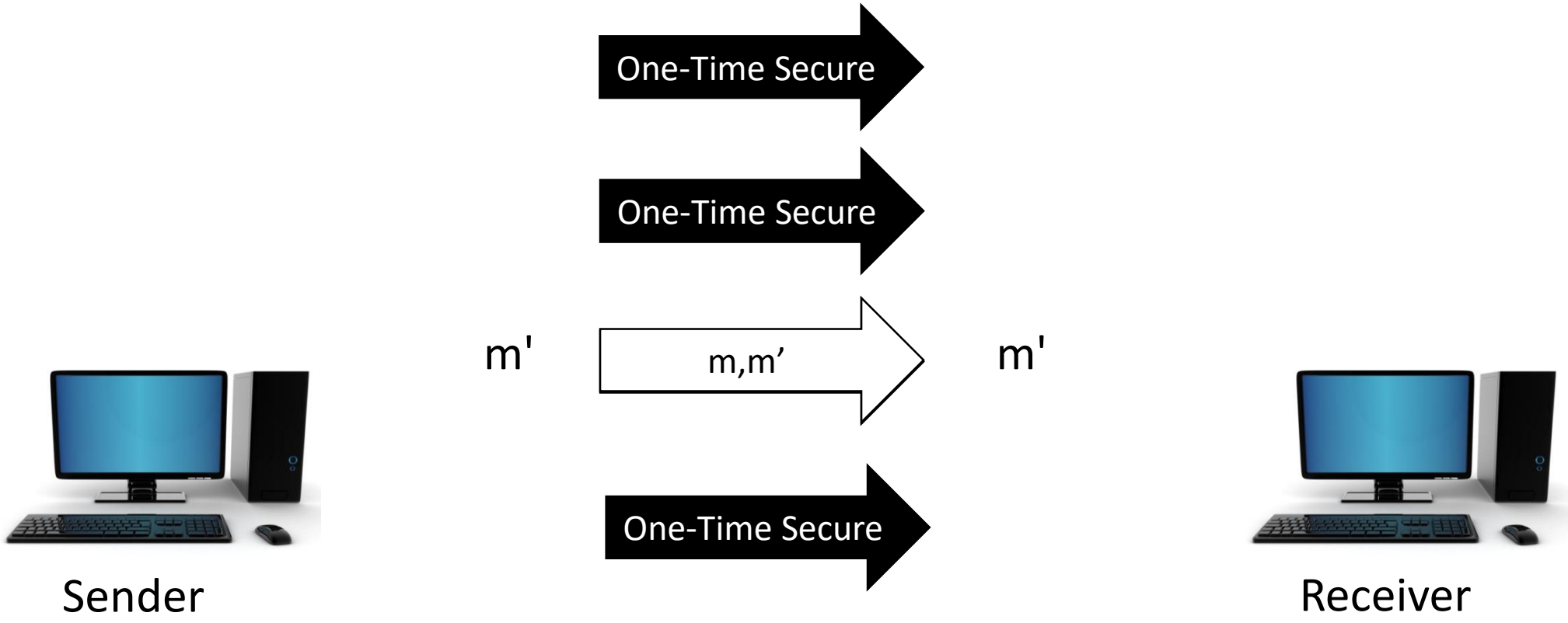
Receiver

N channels

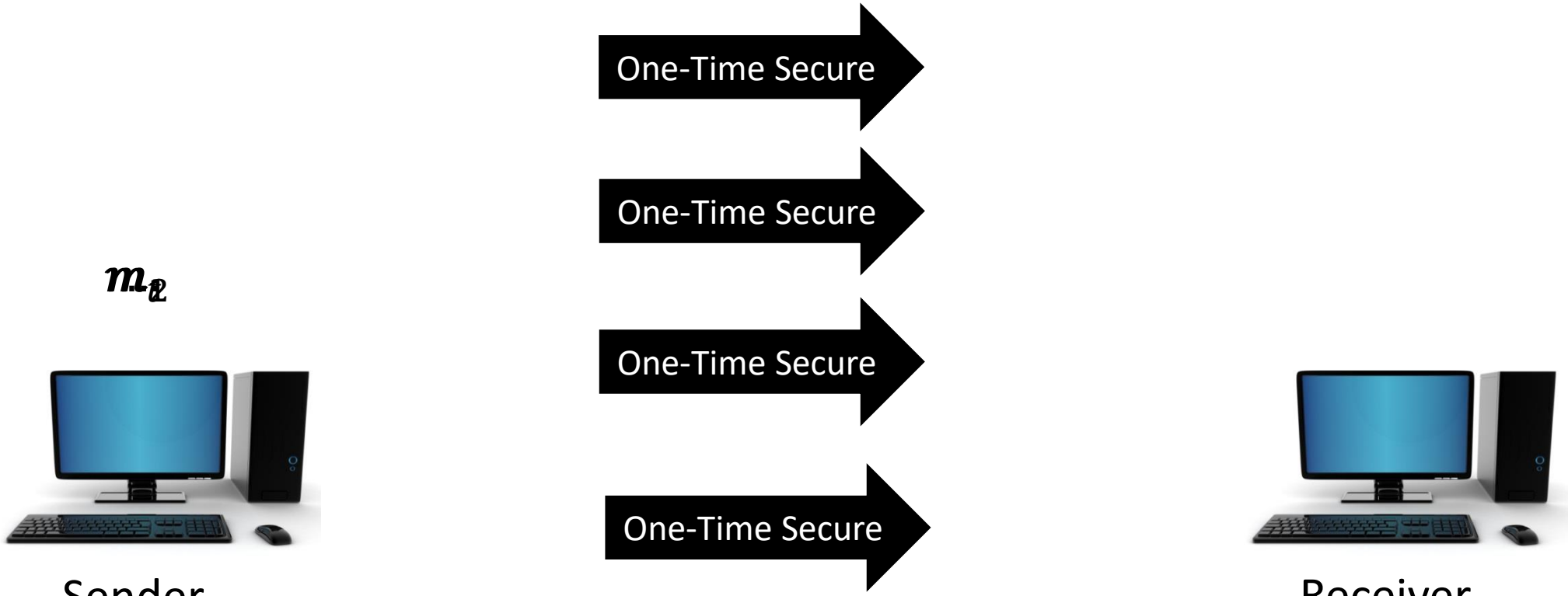
# The Channel Immunization Problem



# The Channel Immunization Problem

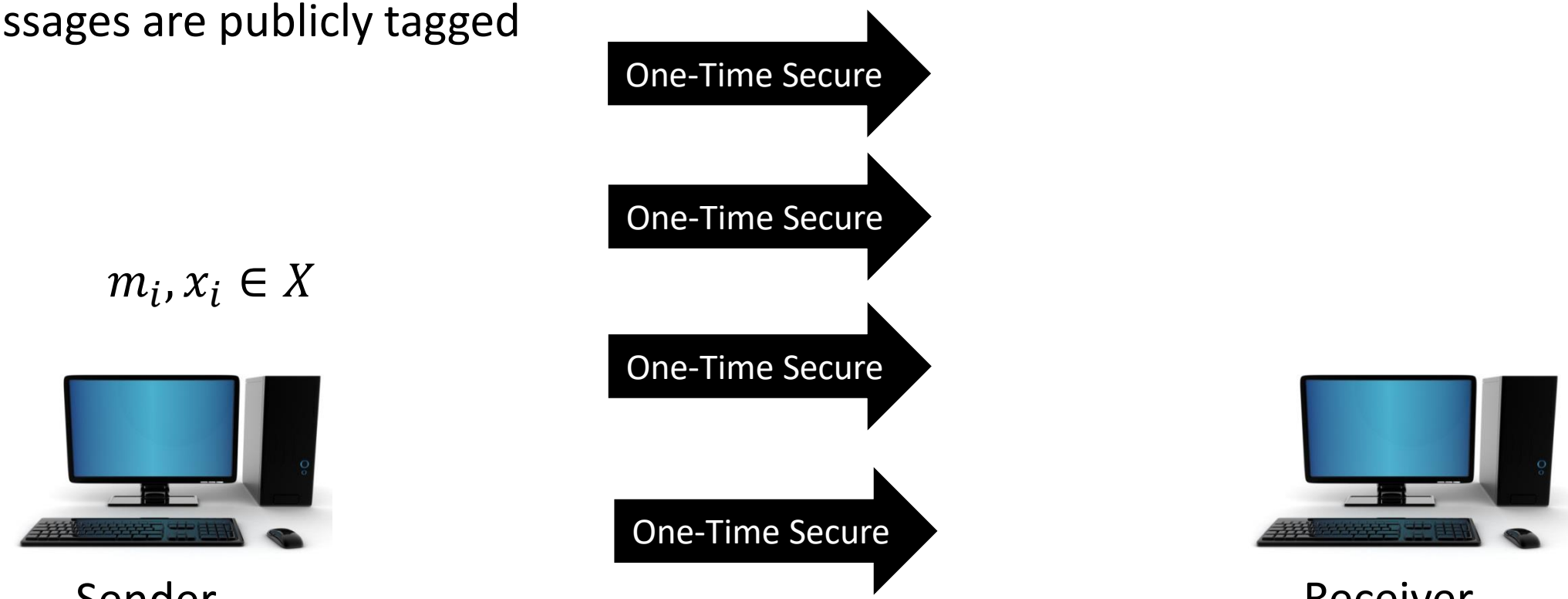


# The Channel Immunization Problem



# The Channel Immunization Problem

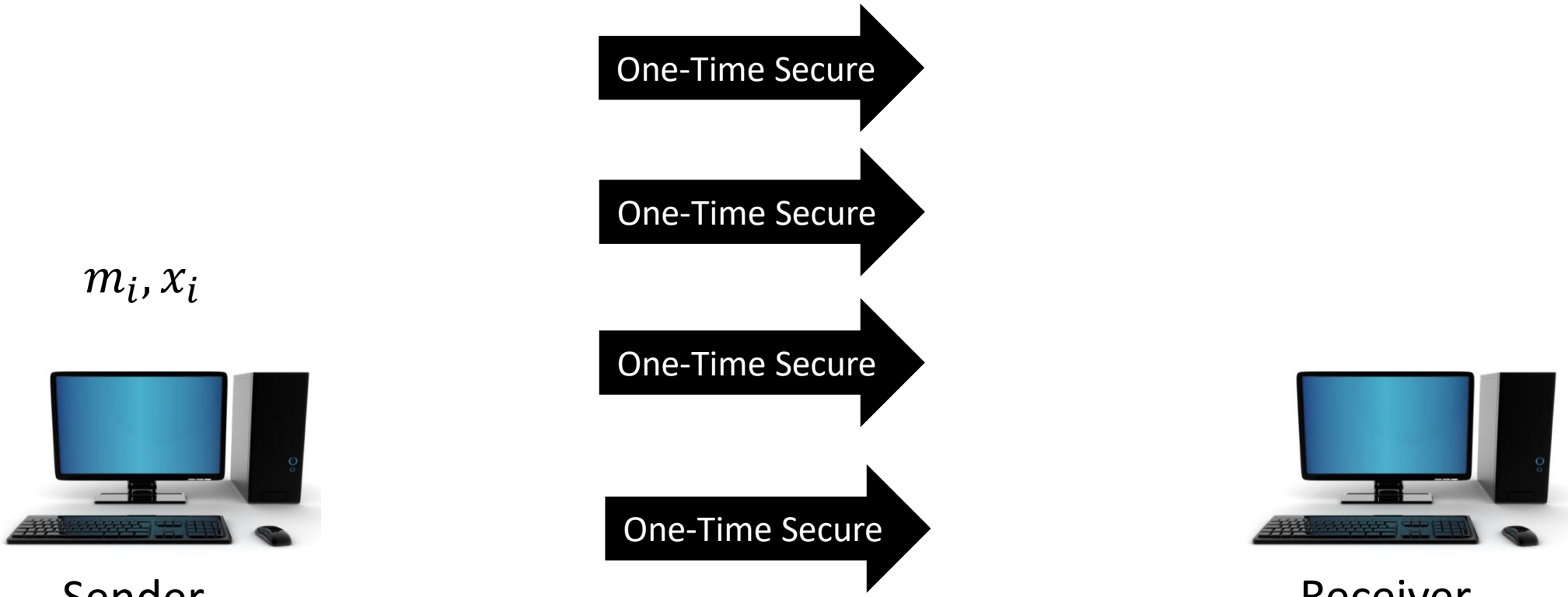
- t online messages
- Sender has no memory
- Messages are publicly tagged



# The Channel Immunization Problem

How many channels N are needed to deliver t-vectors with tag domain X ?

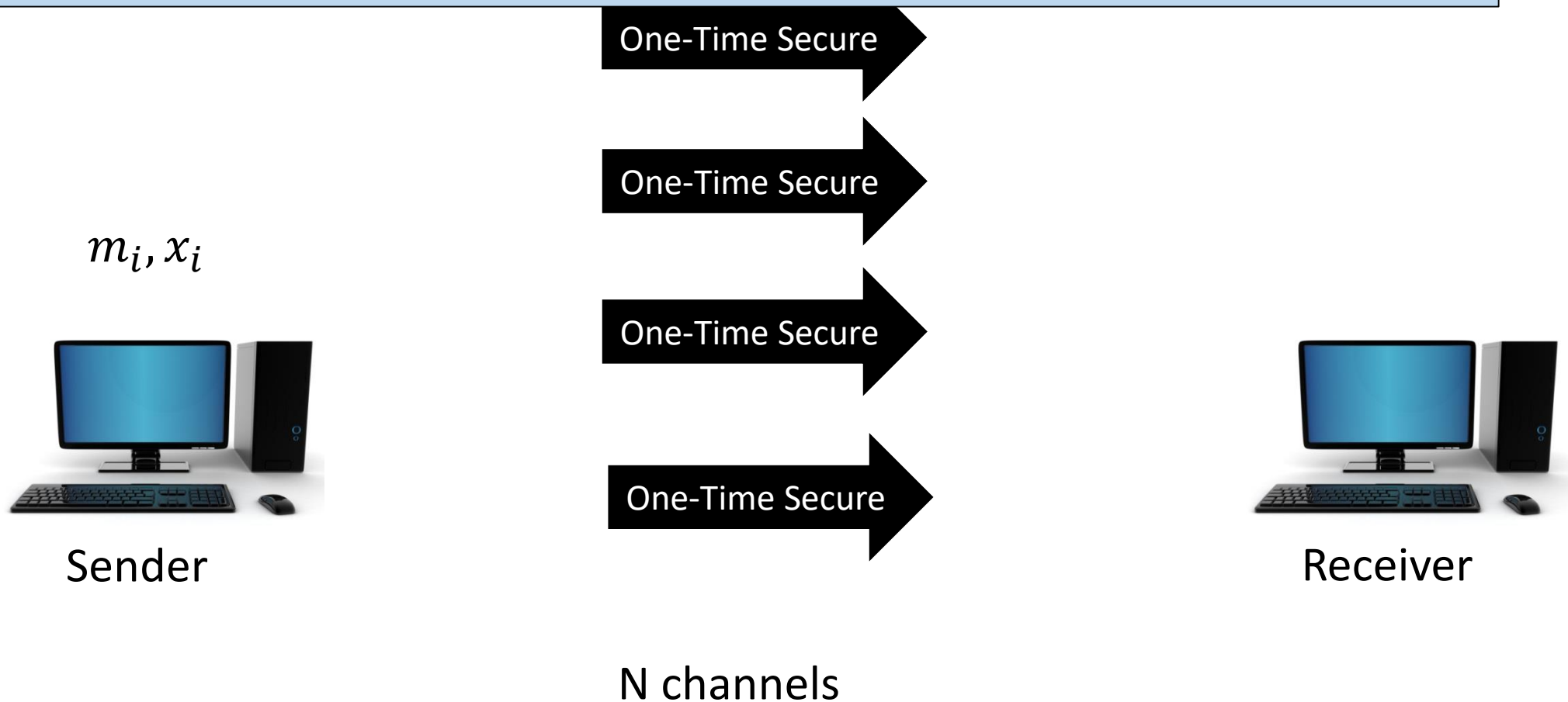
Clearly, $t \leq N \leq |X|$



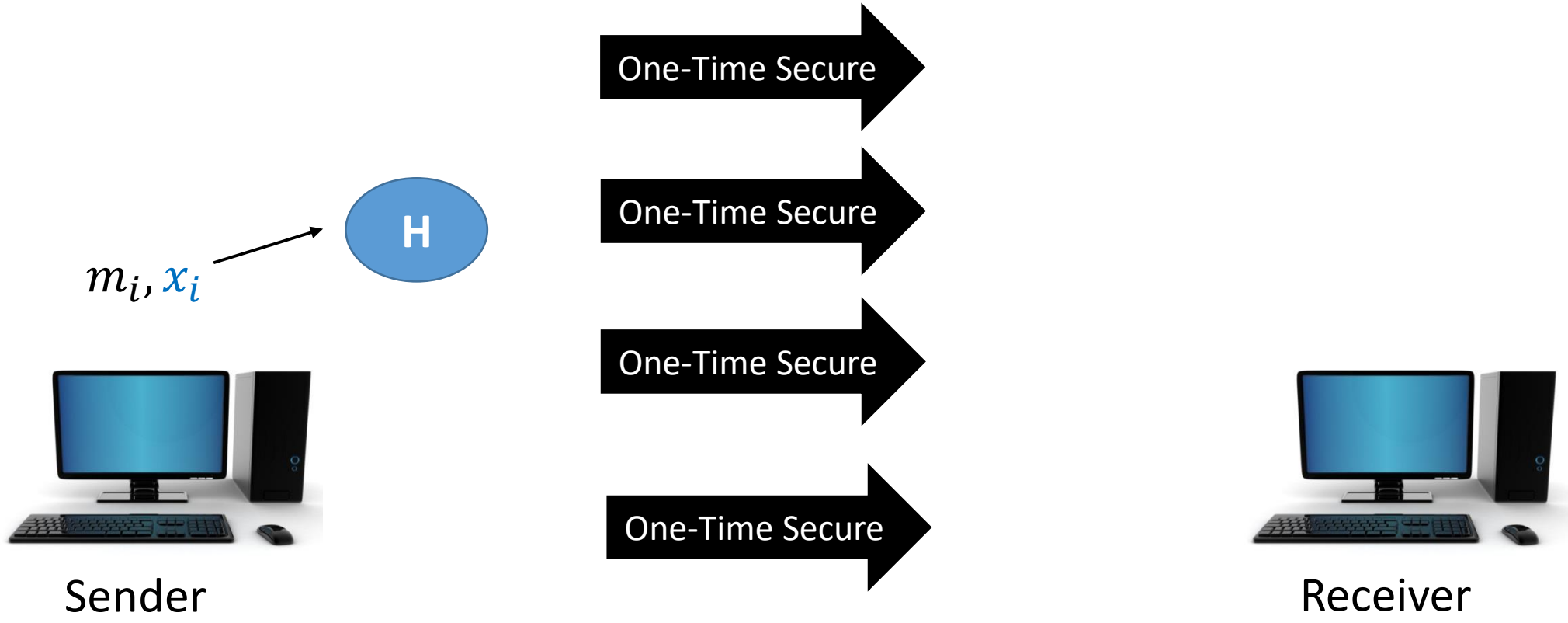
# The Channel Immunization Problem

**Thm.** $N \leq t\,\mathbf{polylog}(|X|^t)$

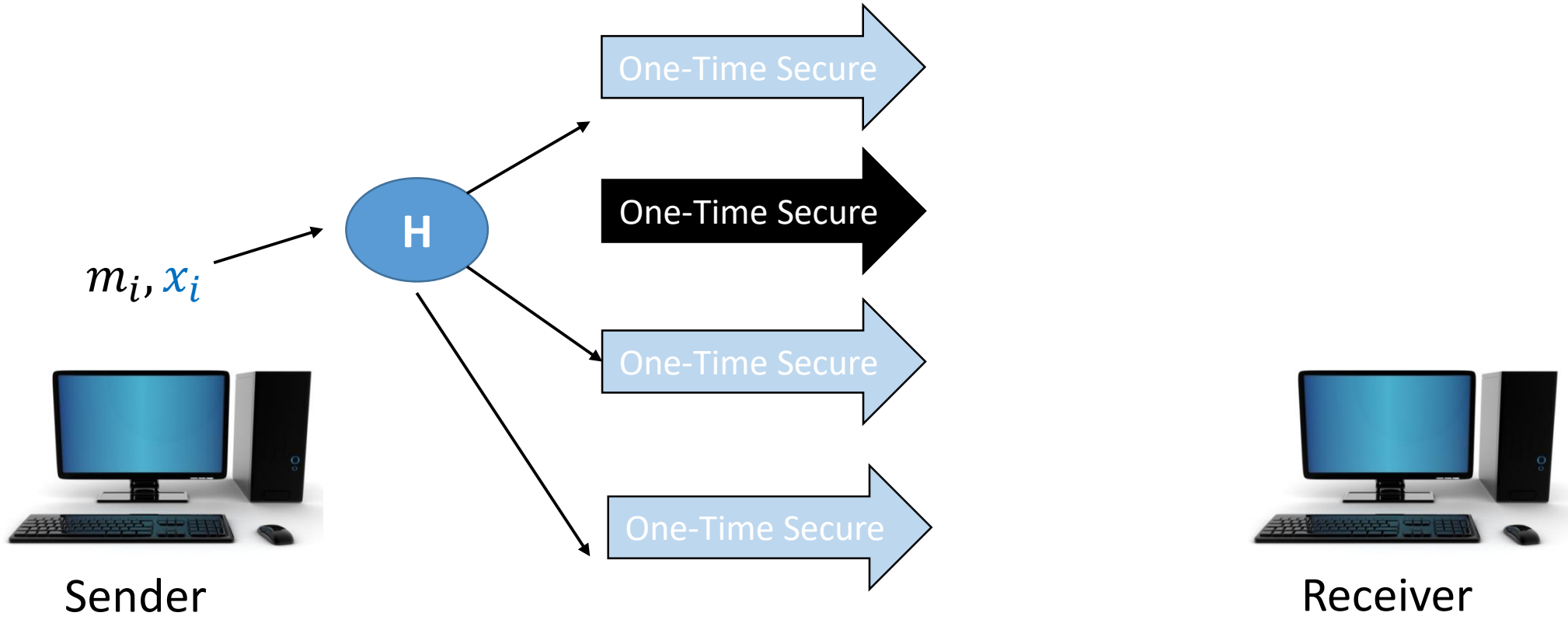
- Actually $N \leq t\,\mathrm{polylog}(L)$ where $L$ is the number of possible $t$-tuples



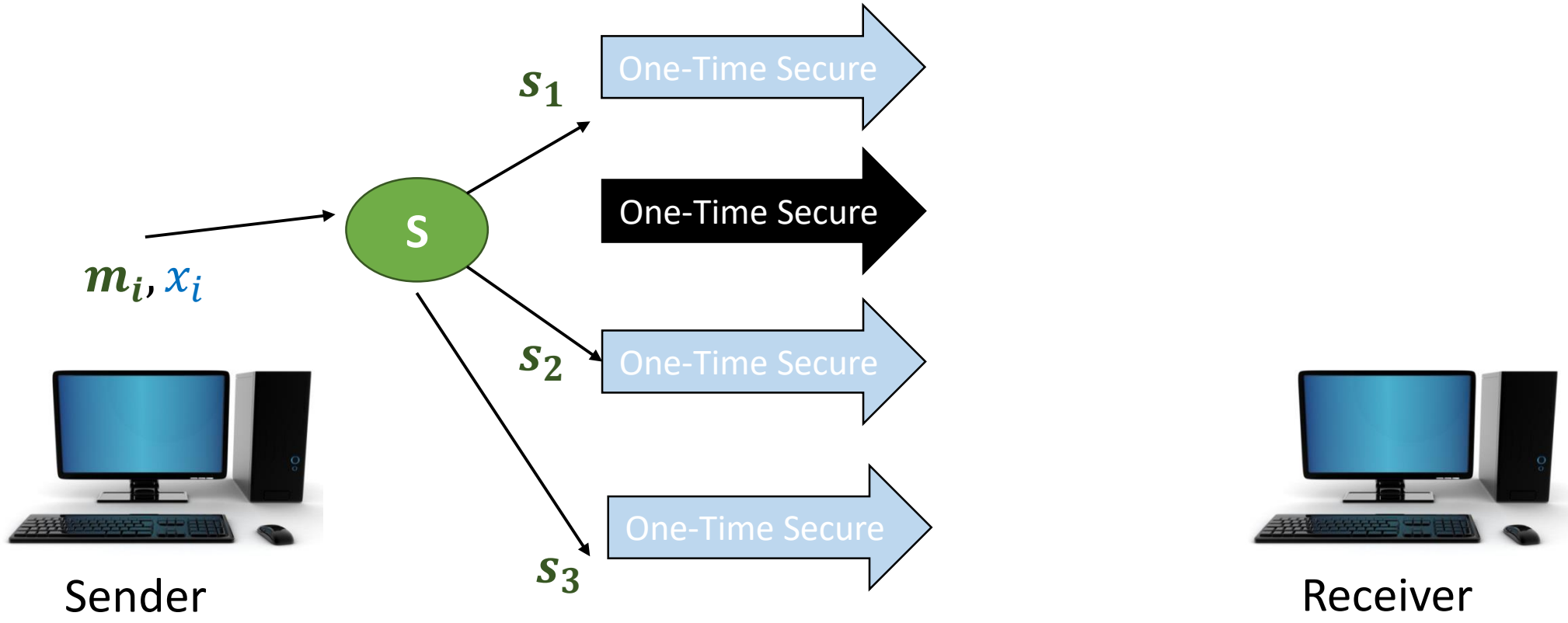
# General strategy: Select & Share



# General strategy: Select & Share



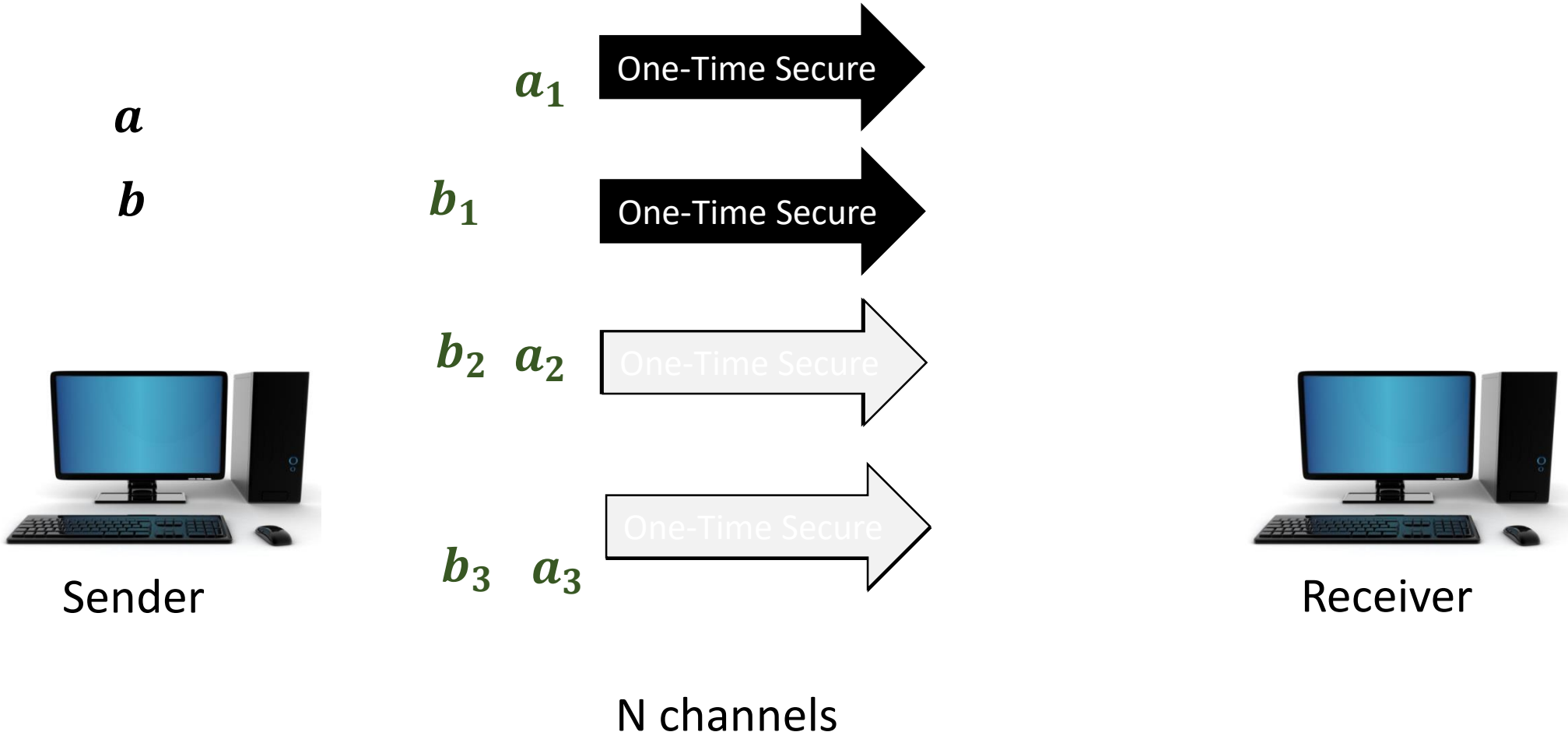
# General strategy: Select & Share



# General strategy: Select & Share

Security for an input tuple $(a, x), (b, y), \ldots$

if collision channels form unauthorized set



# General strategy: Select & Share

**Thm.** $N \leq t\,\mathbf{polylog}(|X|^t)$

- Two level solution:
  - 1-time security to $\log(t)$-security (quadratic overhead)
  - $\log(t)$-security to $t$-security (quasilinear overhead)

- Two instantiations inspired [ChorFiatNaorPinkas-00,GurVaiWee15]

Sender

One-Time Secure

One-Time Secure

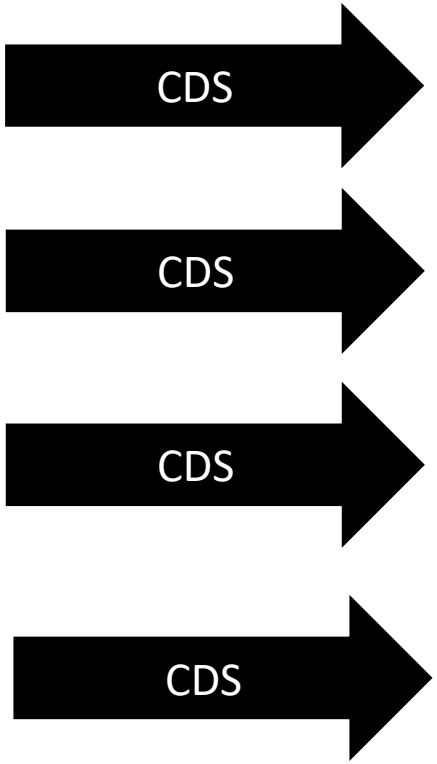
N channels

Receiver

# Robust CDS

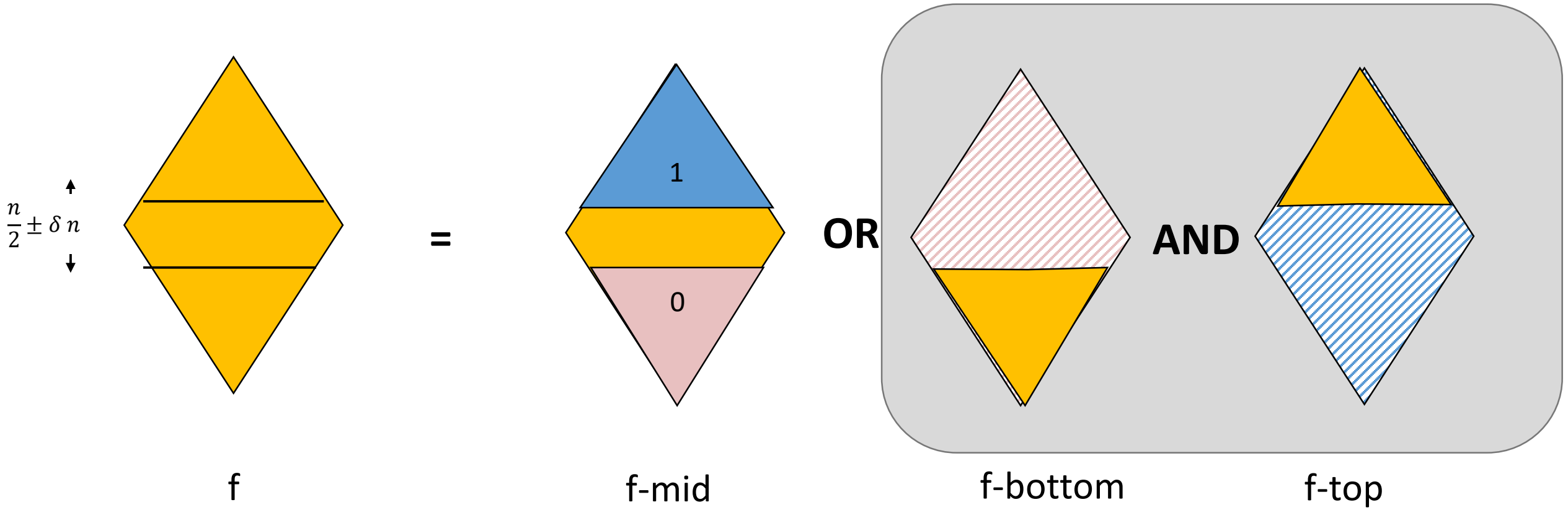
**Immunize each party separately**

- $t = \#\,(0.5 - \delta)n$ −subsets of fixed $(0.5 + \delta)n$ set

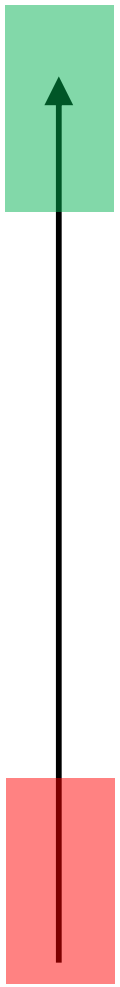


# Final optimizations: Recursively implementing the extreme slices

Based on combinatorial designs [ABFNP19]



# Conclusion



**Upper Bounds:**

$2^n$ [IttSaiNish87]

$2^{0.994n}$ [LiuVai18]

$2^{0.897n}$ [A-BieFarNirPet19]

$2^{0.64n}$ [A-BieNirPet20]

**Lower Bound:** $\Omega\left(\frac{n}{\log(n)}\right)$ [C97]

**OPEN**:

- Sub-exponential general SS?
  - Better Robust-CDS?
- Optimal Linear SS $2^{0.5n}$?
- Super-linear lower-bounds?
- Better Amortized SS?
- Better SS for circuits/monotone circuits?

Thank You !