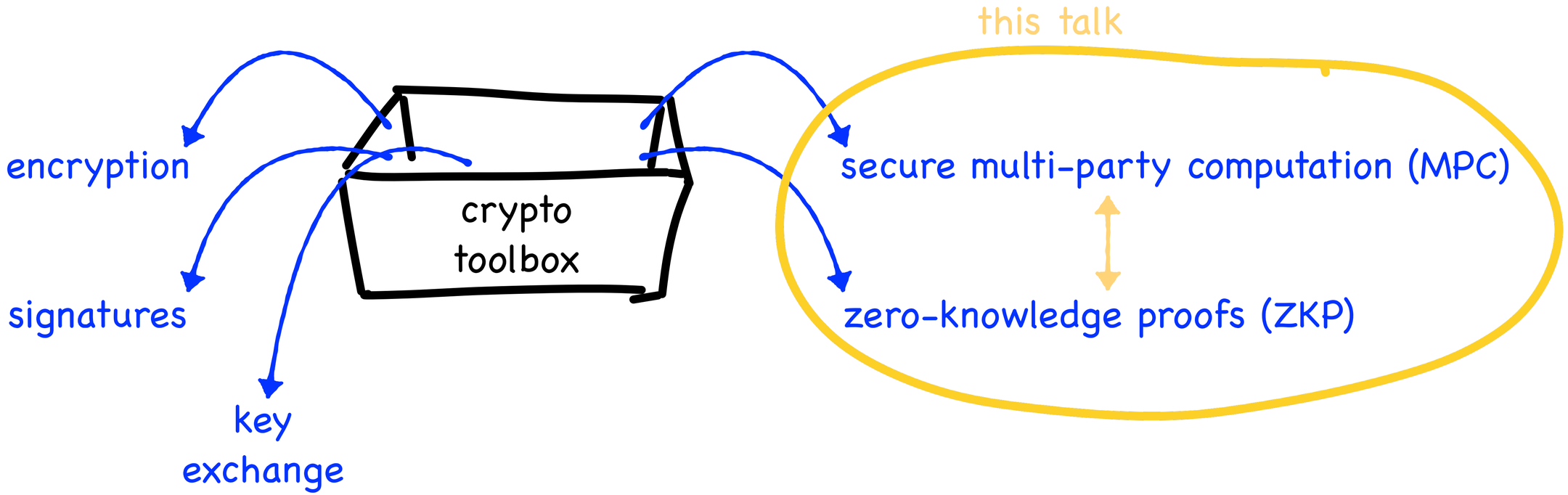
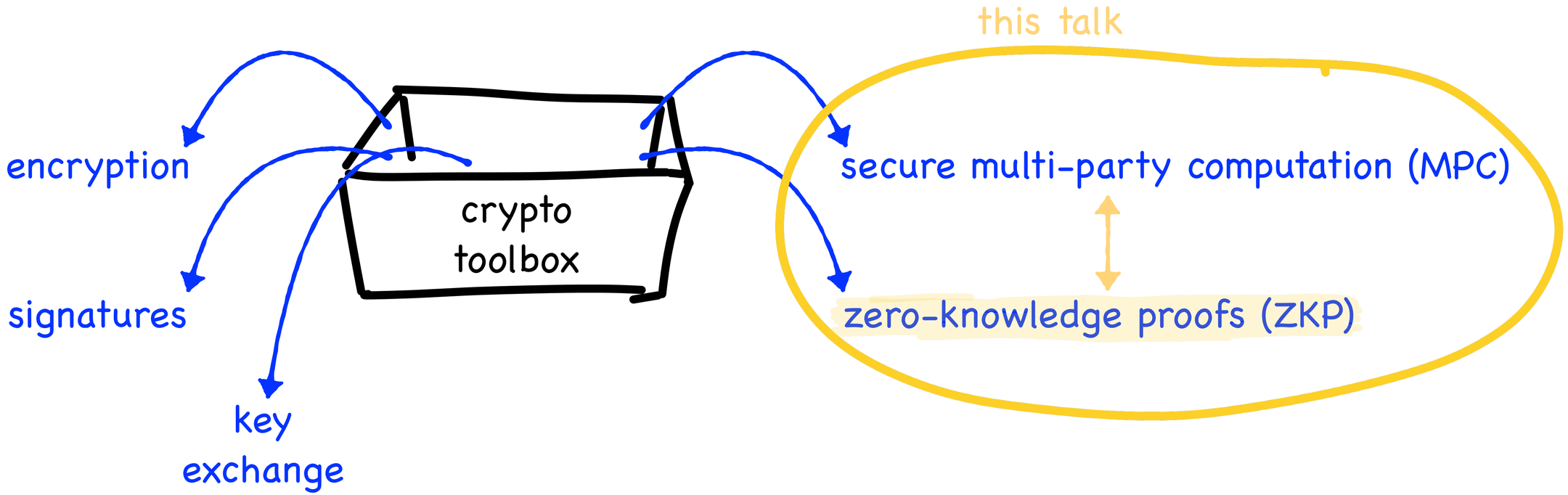
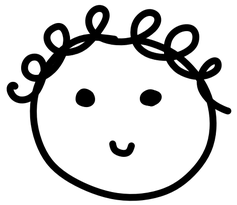
Zero Knowledge Proofs (ZKP)
and
Secure Multiparty Computation (MPC)

Sophia Yakoubov


this talk
encryption
signatures
key
exchange
crypto
toolbox
secure multi-party computation (MPC)
zero-knowledge proofs (ZKP)


this talk
encryption
signatures
key
exchange
crypto
toolbox
secure multi-party computation (MPC)
zero-knowledge proofs (ZKP)

Proof of Sudoku Solvability



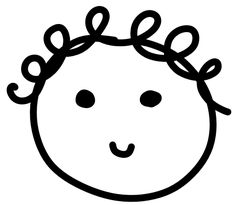
Alice

	7	5		9				6
	2	3		8			4	
8					3			1
5			7		2			
	4		8		6		2	
			9		1			3
9			4					7
	6			7		5	8	
7				1		3	9	

Constraints:

- row contains 1-9
- column contains 1-9
- sub-square contains 1-9

Proof of Sudoku Solvability



Alice

1	7	5	2	9	4	8	3	6
6	2	3	1	8	7	9	4	5
8	9	4	5	6	3	2	7	1
5	1	9	7	3	2	4	6	8
3	4	7	8	5	6	1	2	9
2	8	6	9	4	1	7	5	3
9	3	8	4	2	5	6	1	7
4	6	1	3	7	9	5	8	2
7	5	2	6	1	8	3	9	4

Constraints:

- row contains 1-9
- column contains 1-9
- sub-square contains 1-9

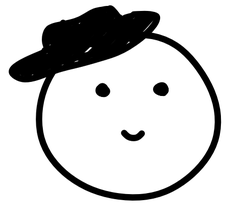
Proof of Sudoku Solvability

Do I just...
give the answer away?

Alice

Can you solve
this sudoku?

	7	5		9				6
	2	3		8			4	
8					3			1
5			7		2			
	4		8		6		2	
			9		1			3
9			4					7
	6			7		5	8	
7				1		3	9	



Dani

No - it's unsolvable,
you're messing with me!

Constraints:
- row contains 1-9
- column contains 1-9
- sub-square contains 1-9

Proof of Sudoku Solvability

Goal:
- convince Dani
- without giving anything away

Alice

	7	5		9				6
	2	3		8			4	
8					3			1
5			7		2			
	4		8		6		2	
			9		1			3
9			4					7
	6			7		5	8	
7				1		3	9	

Constraints:
- row contains 1-9
- column contains 1-9
- sub-square contains 1-9

Proof of Sudoku Solvability

Goal:
- convince Dani
- without giving anything away

Alice

1	7	5	2	9	4	8	3	6
6	2	3	1	8	7	9	4	5
8	9	4	5	6	3	2	7	1
5	1	9	7	3	2	4	6	8
3	4	7	8	5	6	1	2	9
2	8	6	9	4	1	7	5	3
9	3	8	4	2	5	6	1	7
4	6	1	3	7	9	5	8	2
7	5	2	6	1	8	3	9	4

random permutation:

$1 \to 2$
$2 \to 6$
$3 \to 5$
$4 \to 9$
$5 \to 1$
$6 \to 7$
$7 \to 8$
$8 \to 4$
$9 \to 3$

2	8	1	6	3	9	4	5	7
7	6	5	2	4	8	3	9	1
4	3	9	1	7	5	6	8	2
1	2	3	8	5	6	9	7	4
5	9	8	4	1	7	2	6	3
6	4	7	3	9	2	8	1	6
3	5	4	9	6	1	7	2	8
9	7	2	5	8	3	1	4	6
8	1	6	7	2	4	5	3	9

if the original sudoku was "valid", so is the new one!

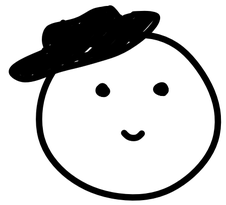
Constraints:
- row contains 1-9
- column contains 1-9
- sub-square contains 1-9

Proof of Sudoku Solvability

Goal:
- convince Dani
- without giving anything away

Alice

2	8	1	6	3	9	4	5	7
7	6	5	2	4	8	3	9	1
4	3	9	1	7	5	6	8	2
1	2	3	8	5	6	9	7	4
5	9	8	4	1	7	2	6	3
6	4	7	3	9	2	8	1	6
3	5	4	9	6	1	7	2	8
9	7	2	5	8	3	1	4	6
8	1	6	7	2	4	5	3	9



Dani

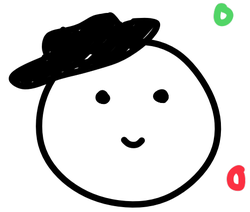
Constraints:
- row contains 1-9
- column contains 1-9
- sub-square contains 1-9

Proof of Sudoku Solvability

Goal:
✗ convince Dani
✓ without giving anything away

Alice

1	2	3	8	5	6	9	7	4



Dani

I didn't learn anything!

... but maybe the stuff under the remaining post-its is bogus.

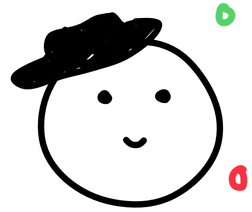
Constraints:
- row contains 1-9
- column contains 1-9
- sub-square contains 1-9
- initial conditions

Proof of Sudoku Solvability

Goal:
✗ convince Dani
✓ without giving anything away

Alice

1	2	3	8	5	6	9	7	4



Dani

I didn't learn anything!

... but maybe the stuff under the remaining post-its is bogus.

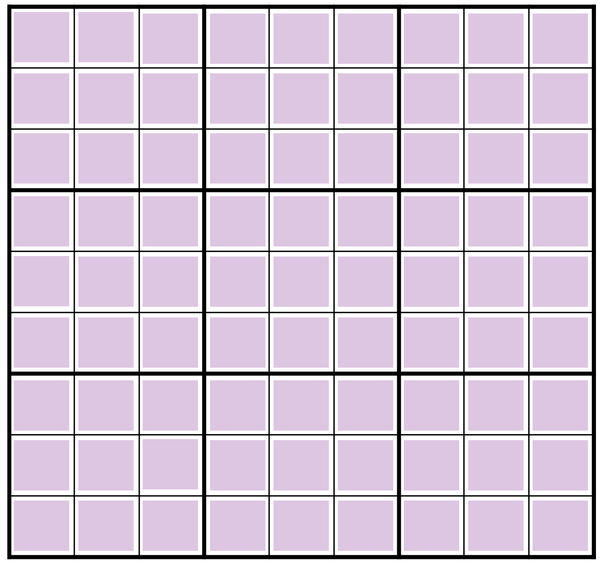
Constraints:
- row contains 1-9
- column contains 1-9
- sub-square contains 1-9
- initial conditions

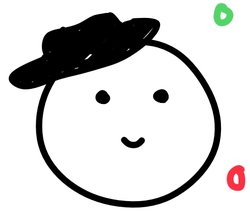
} 9+9+9+1 = 28 constraints. If all satisfied, solution is valid.

Proof of Sudoku Solvability

Goal:
✓ convince Dani
✓ without giving anything away

Alice





Dani

I didn't learn anything!

... but if Alice cheated, she might get away with it with prob ≤ 27/28.

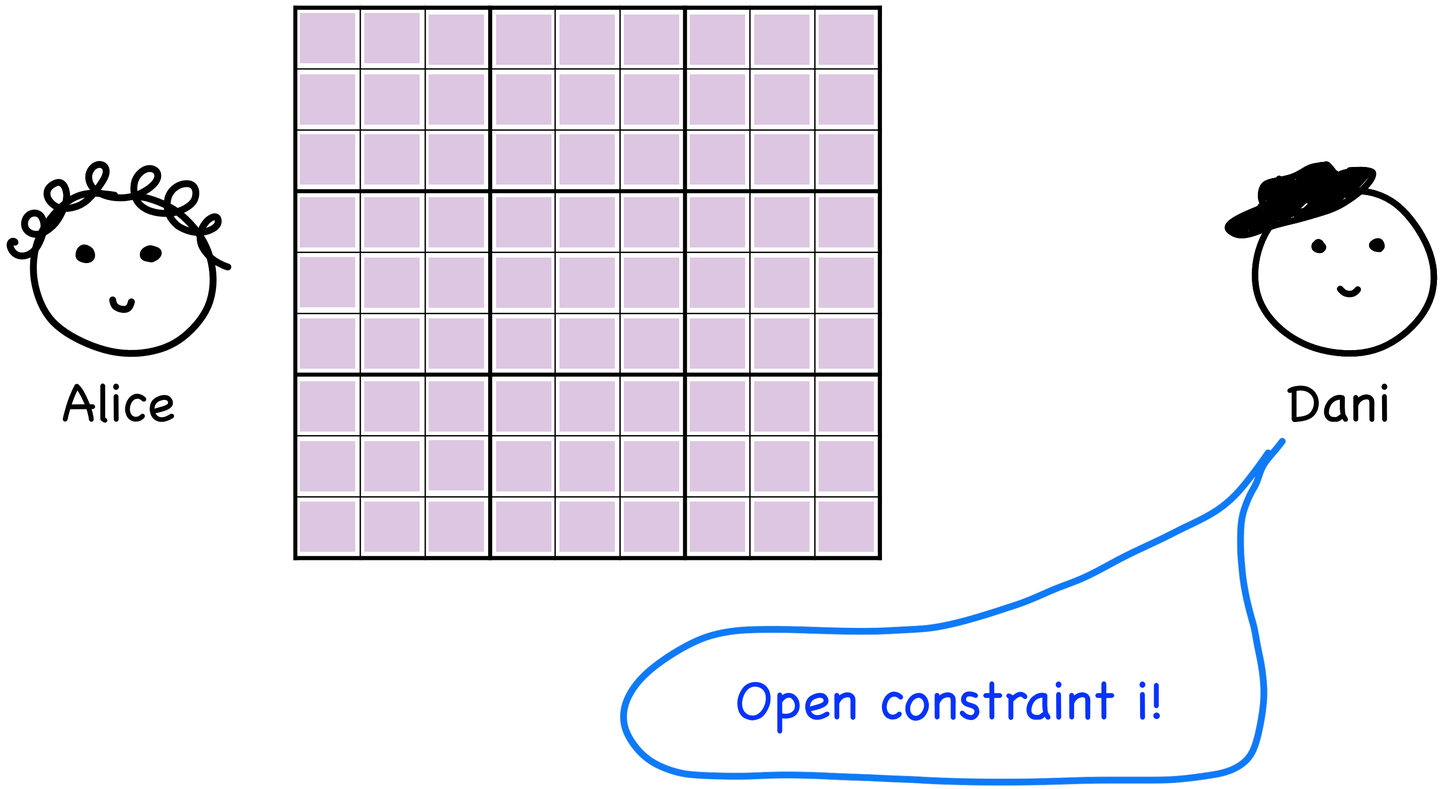
Open constraint i!

Constraints:
- row contains 1-9
- column contains 1-9
- sub-square contains 1-9
- initial conditions

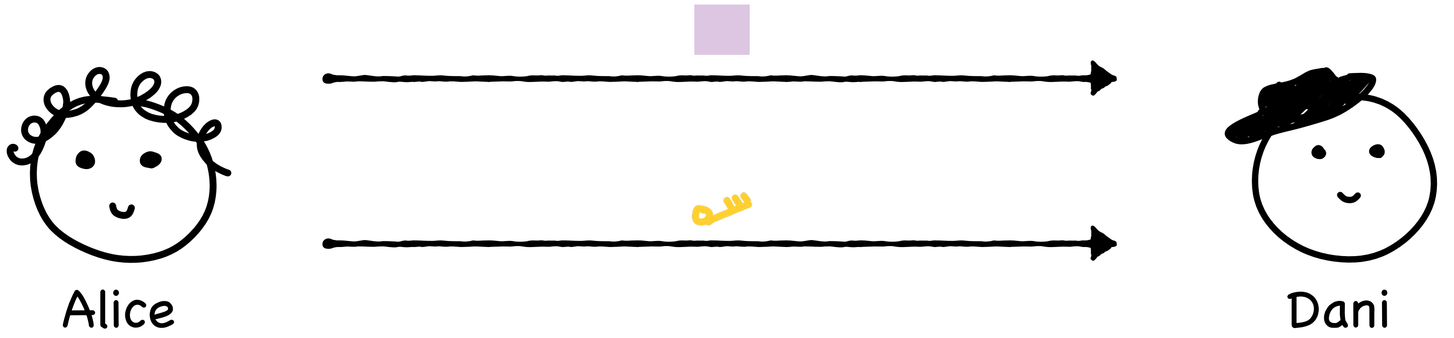
} 9+9+9+1 = 28 constraints. If all satisfied, solution is valid.

Solution: repeat k times, s.t $(27/28)^k$ is small enough.

Proof of Sudoku Solvability… online



Tool: Commitments

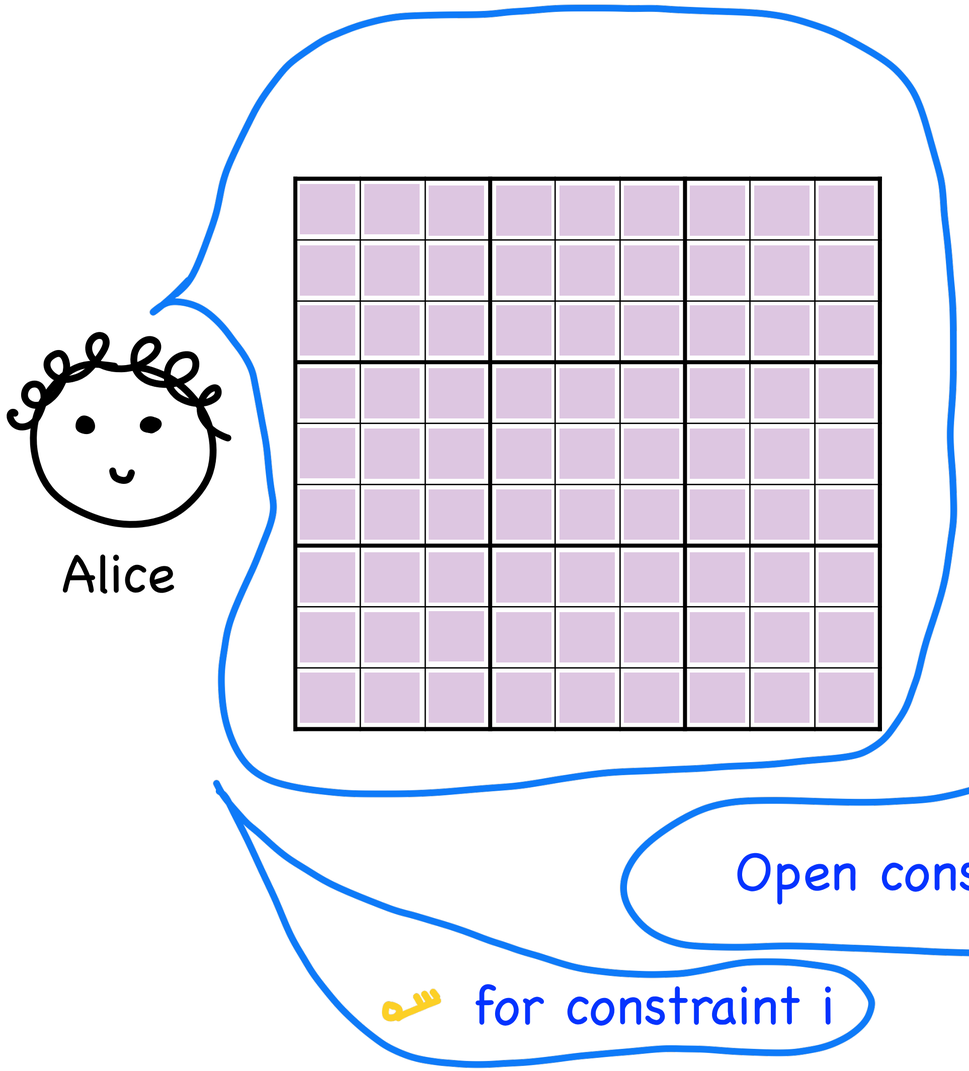


Commit(x) → (, سه)

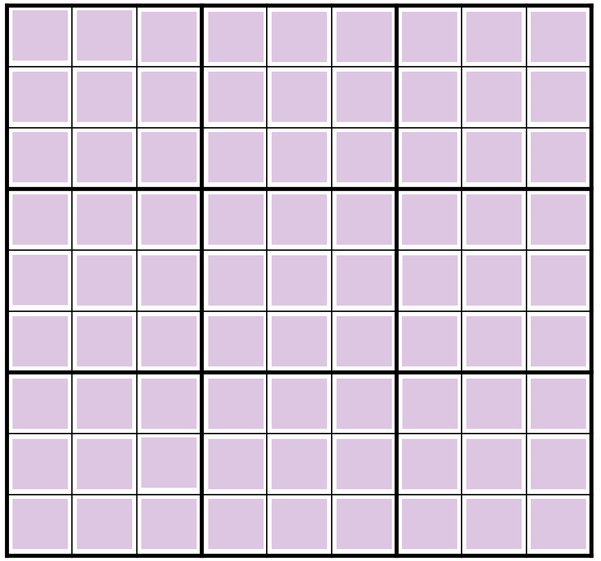
Open(, سه) → x

- Hiding: reveals nothing about what's inside, without سه
- Binding: can only be opened to one thing

Proof of Sudoku Solvability… online



Open(,) → x



Commitments:
- Binding: can only be opened to one thing
- Hiding: reveals nothing about what's inside, without

Zero Knowledge Proofs (ZKP)

ZKP: Prove a claim without revealing any information apart from the claim itself!

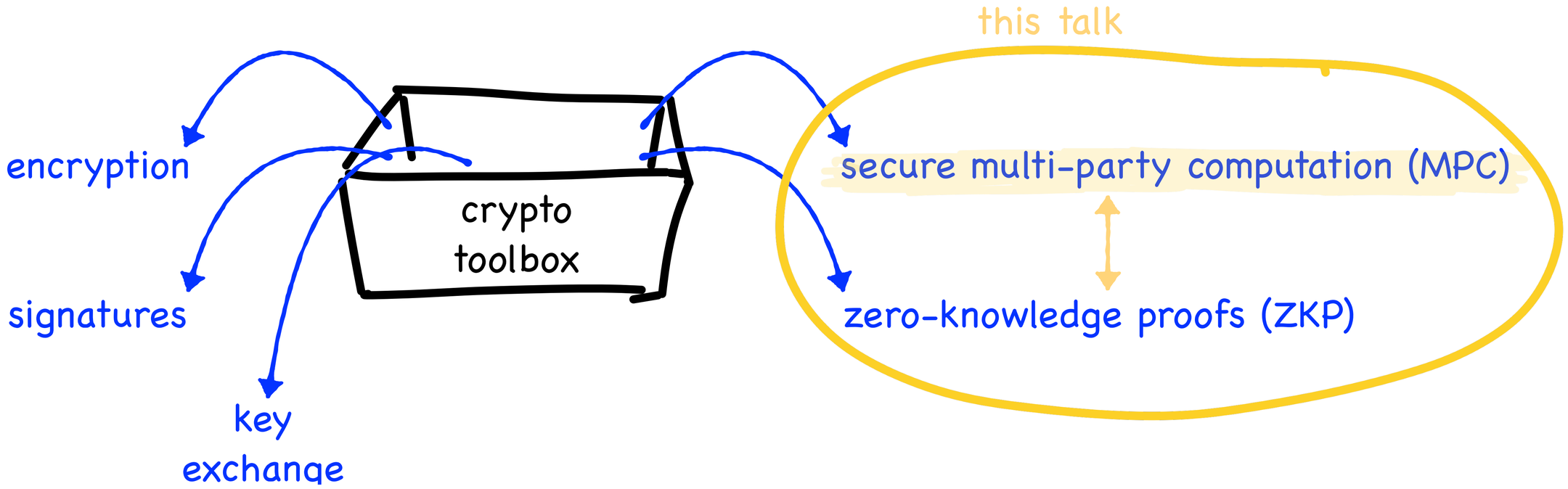
In practice, we want to prove things like...

- Identity, or possession of credentials
- Correct computation
- Generally: knowledge/existence of x s.t. $f(x) = 1$

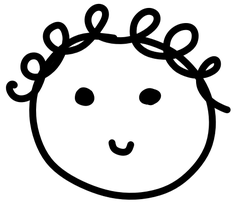
How do we do this?

- Sudoku is NP-complete - we can tu[illegible]g into a sudoku!

inefficient

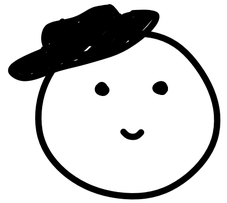

this talk
encryption
signatures
key
exchange
crypto
toolbox
secure multi-party computation (MPC)
zero-knowledge proofs (ZKP)

Secure Multi-Party Computation (MPC): A First Example



Alice

$x_A \in [1,\ldots,10]$



Dani

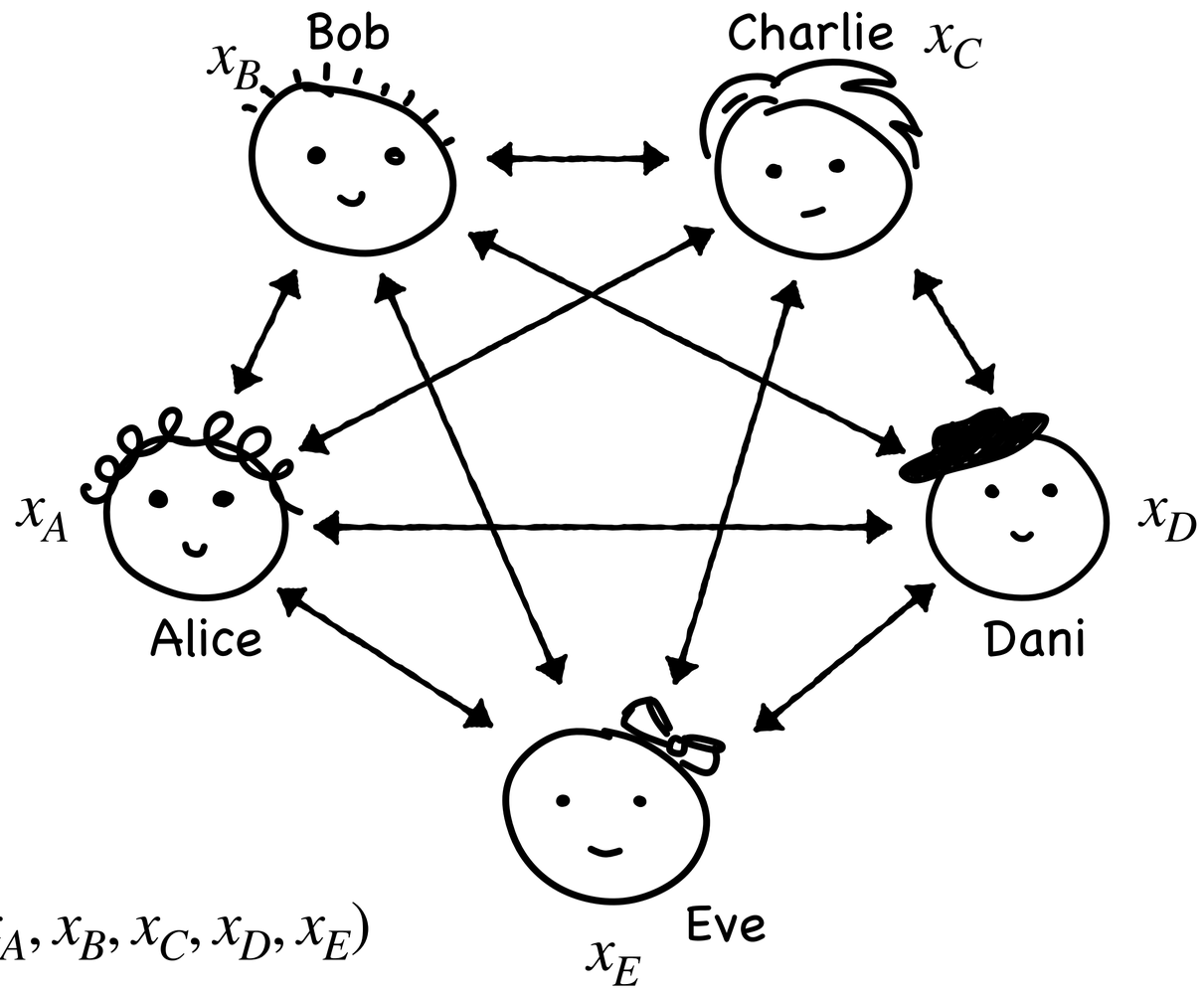
$x_D \in [1,\ldots,10]$

We want privacy:
if $x_A \neq x_D$, that is all they learn

$$f(x_A, x_B) = \begin{cases} 1 & \text{if } x_A = x_B \\ 0 & \text{otherwise} \end{cases}$$

we could ask someone to help us...
but there is no-one we both trust!

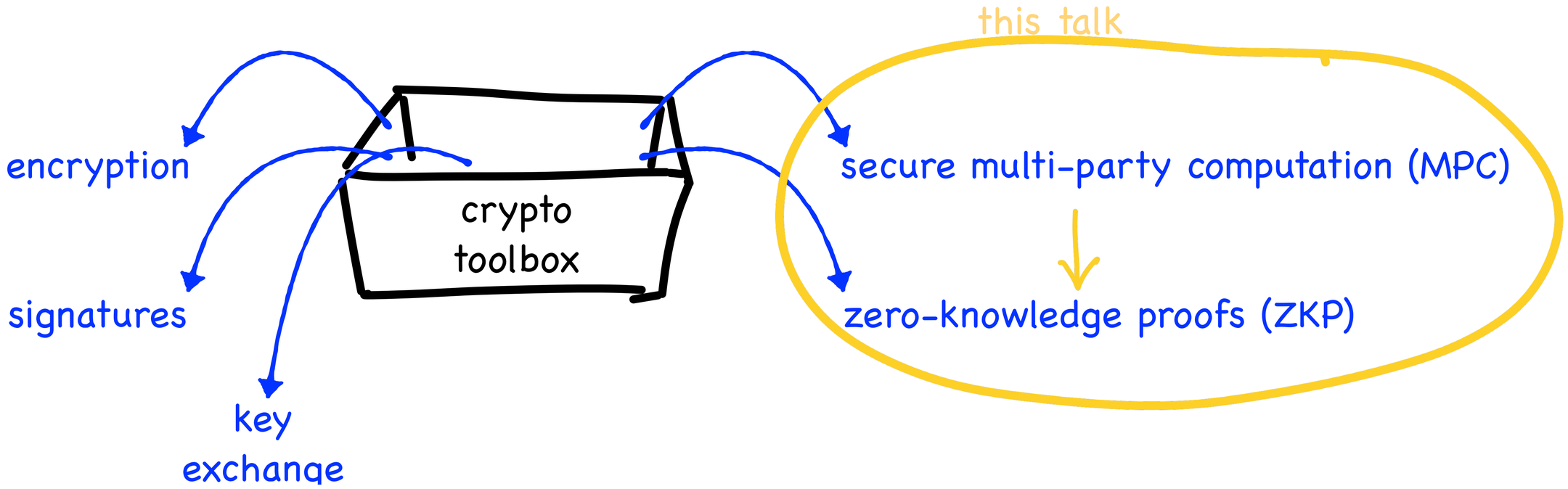
Secure Multi-Party Computation (MPC)



$y = f(x_A, x_B, x_C, x_D, x_E)$

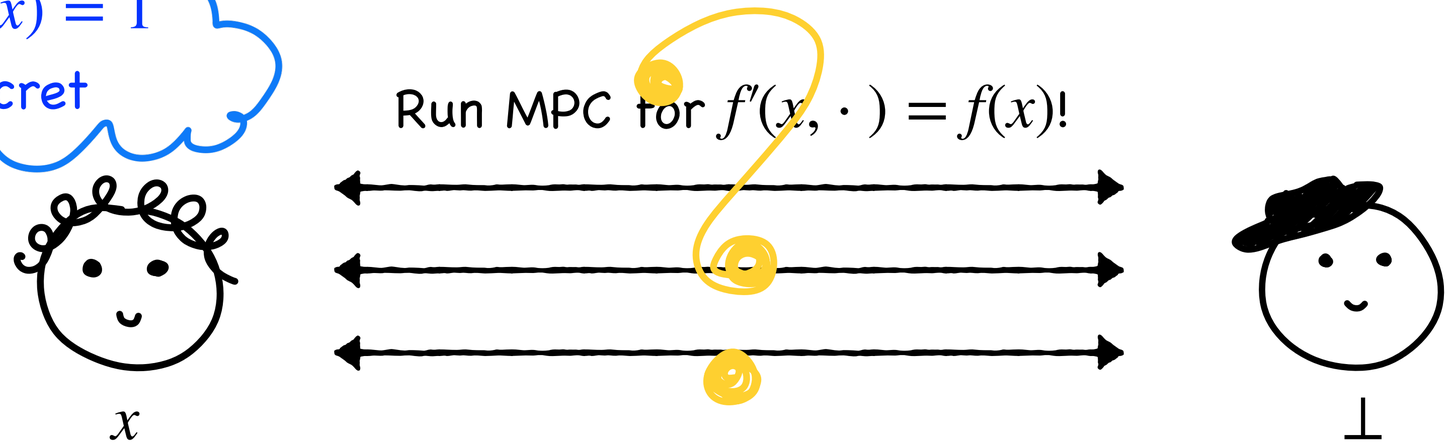
We want:

- correctness
- t-privacy: the combined views of t or fewer participants reveal nothing other than y


this talk
encryption
signatures
key
exchange
crypto
toolbox
secure multi-party computation (MPC)
zero-knowledge proofs (ZKP)

ZKP from MPC: Attempt 1

Goal:
- convince Dani of $f(x) = 1$
- while keeping x secret



	Communication Complexity	Tools
Reduce to Sudoku (or something...)	poly(|f|)	lightweight (commitments)
Run 2PC	O(|f|)	heavyweight (i.e. "public key" operations)

MPC from Lightweight Tools

Workarounds:

- More participants …

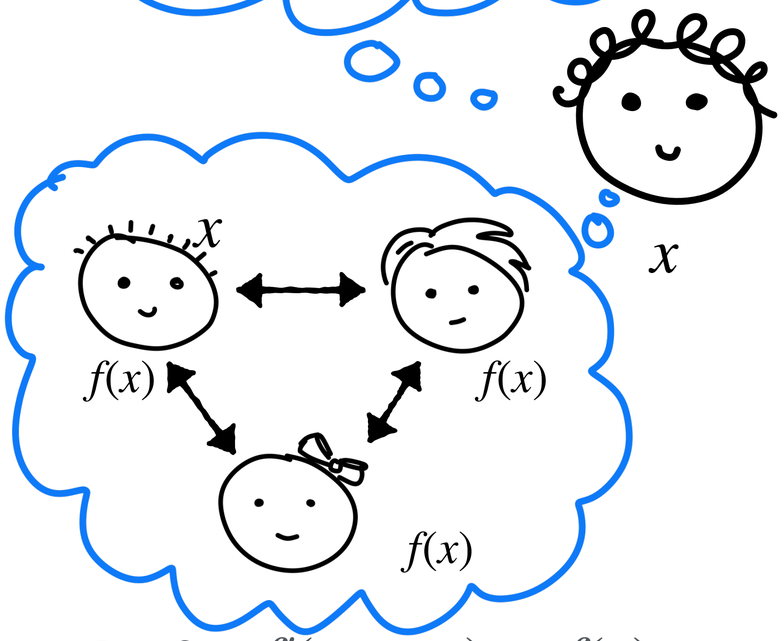
  we can get t-privacy for $t < \frac{n}{2}$ using only lightweight tools

  e.g.: $n = 3, t = 1$

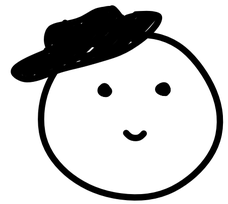
- Correlated randomness

ZKP from MPC: Attempt 2

Goal:
- convince Dani of $f(x) = 1$
- while keeping x secret

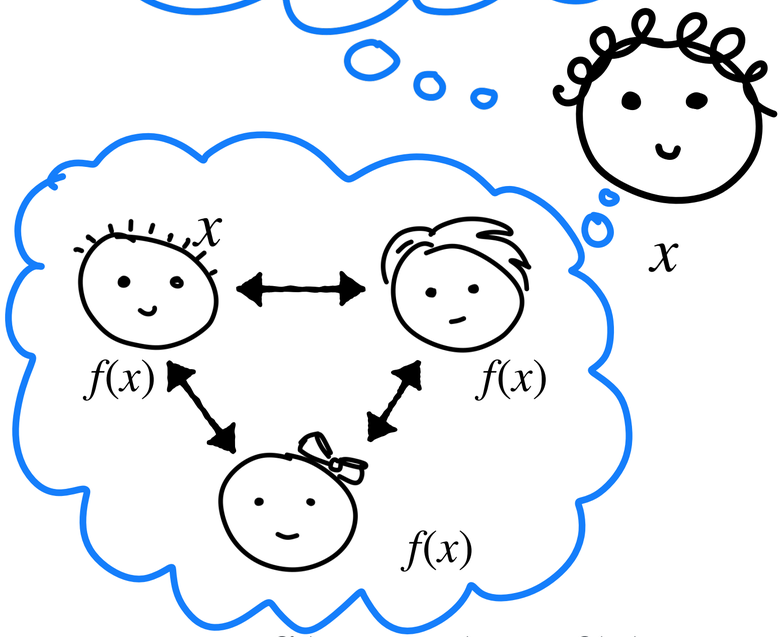


MPC for $f'(x, \cdot, \cdot) = f(x)$

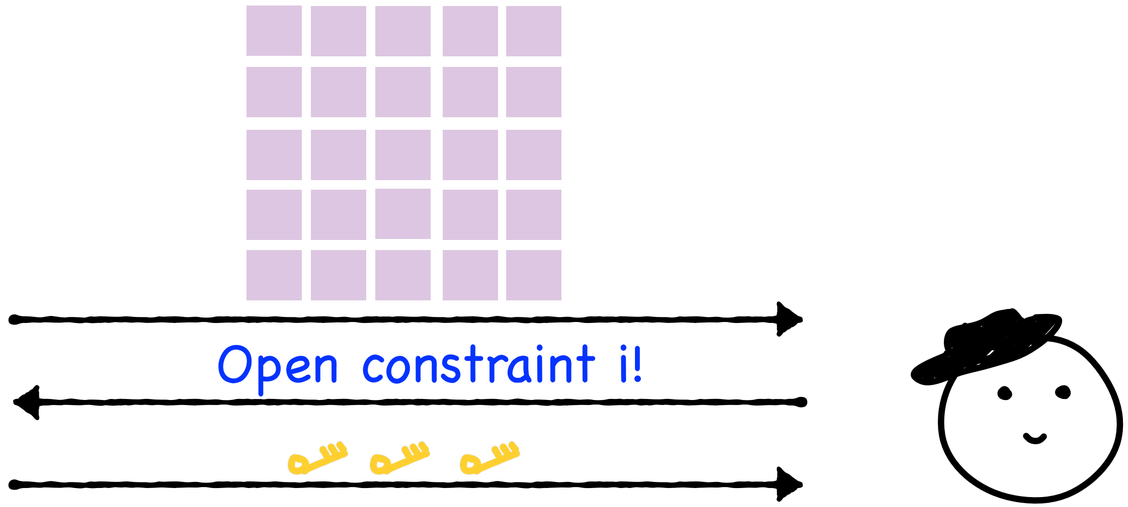


Recall our first ZKP...

Goal:
- convince Dani of $f(x) = 1$
- while keeping x secret



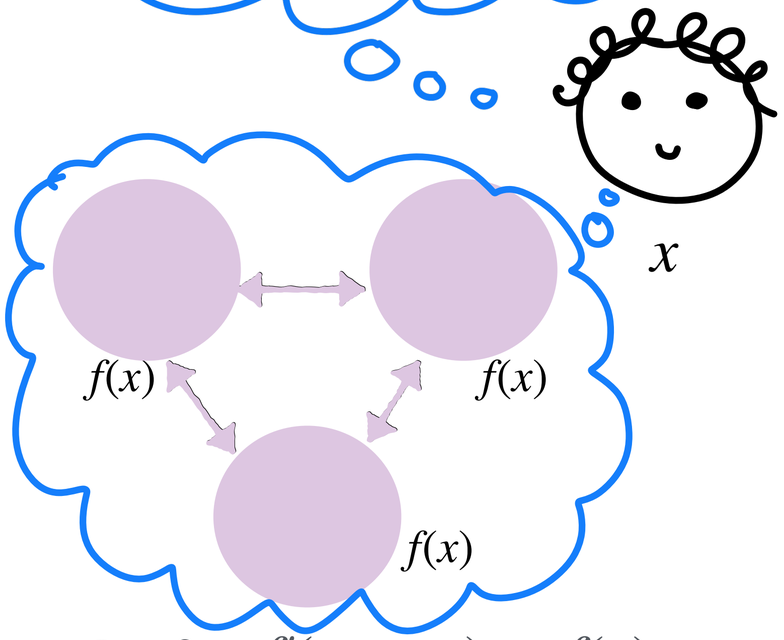
MPC for $f'(x, \cdot, \cdot) = f(x)$



Constraints s.t.
- one reveals nothing
- if all of them hold, the claim is true

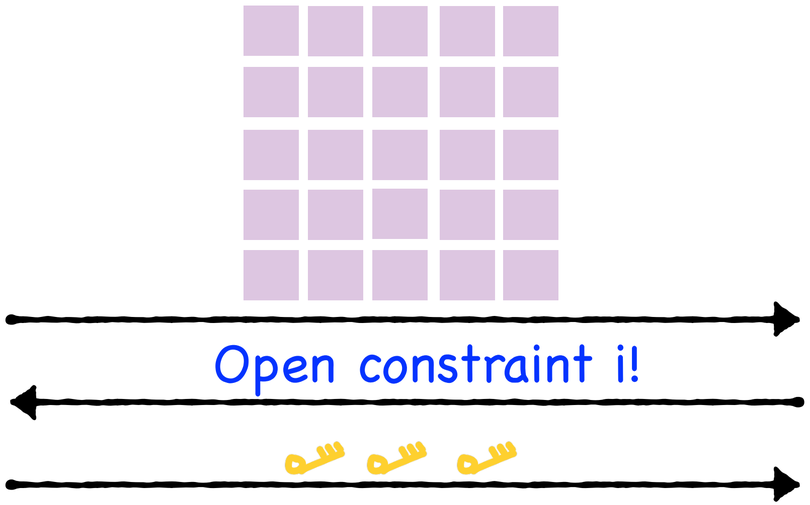
ZKP from MPC: Attempt 2

Goal:
- convince Dani of $f(x) = 1$
- while keeping x secret



MPC for $f'(x, \cdot, \cdot) = f(x)$

x

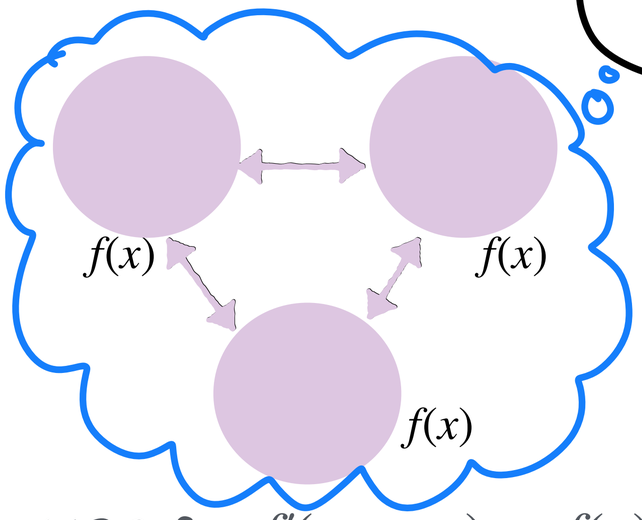


Constraints s.t.
- one reveals nothing
- if all of them hold, the claim is true

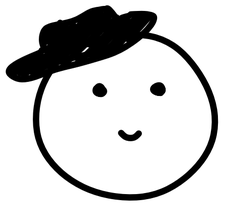
ZKP from MPC: Attempt 2

Goal:
- convince Dani of $f(x) = 1$
- while keeping x secret

x



MPC for $f'(x, \cdot, \cdot) = f(x)$



Open(,) → party i's choices
Open(,) → messages
Open(,) → messages

party i did not cheat,
and $f(x) = 1$

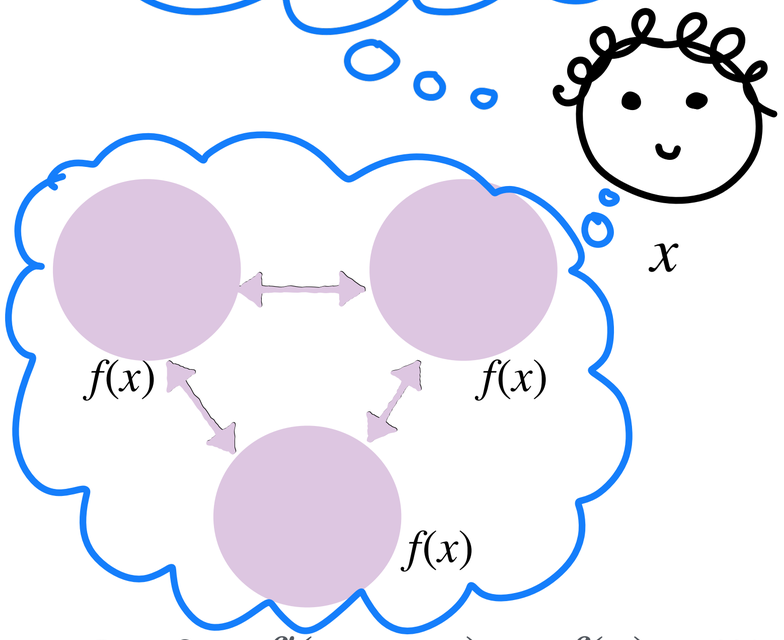
Constraints s.t.
- one reveals nothing
- if all of them hold, the claim is true

ZKP from MPC: Attempt 2

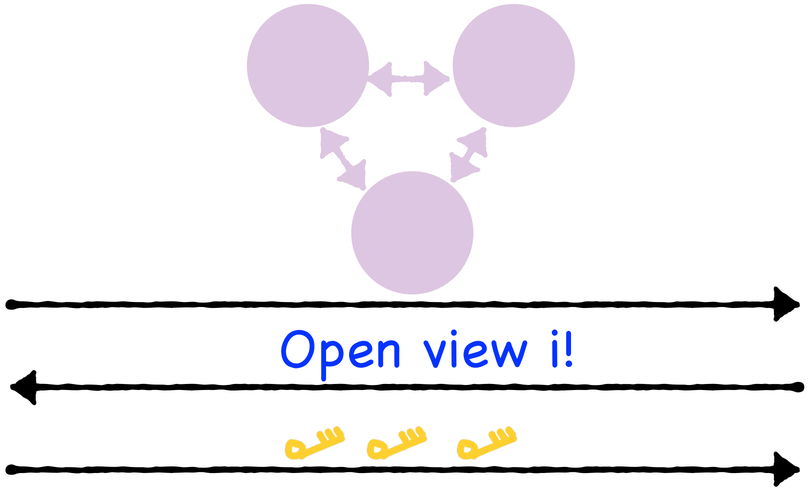
Goal:

✓ convince Dani of $f(x) = 1$

- while keeping x secret



x



if Alice cheated, she can get away with it with probability 2/3

repeat k times, s.t $(2/3)^k$ is small enough.

MPC for $f'(x, \cdot, \cdot) = f(x)$ with:

- 1-privacy

✓ perfect correctness

party i did not cheat, and $f(x) = 1$

Constraints s.t.

- one reveals nothing

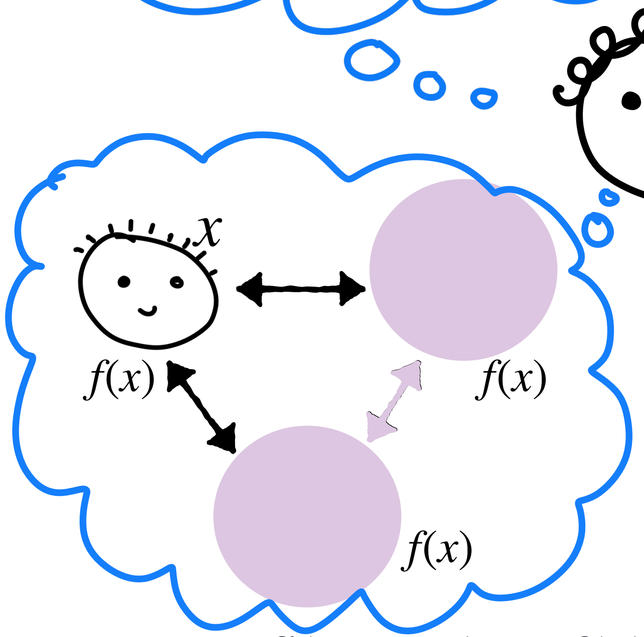
✓ if all of them hold, the claim is true

ZKP from MPC: Attempt 2

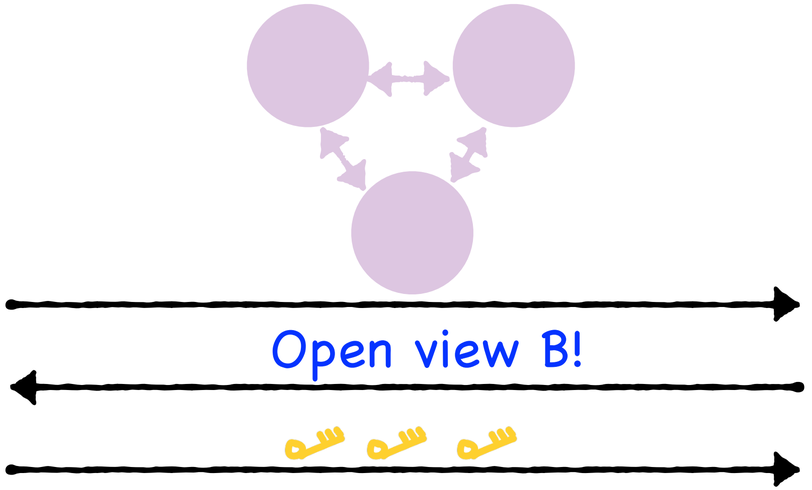
Goal:

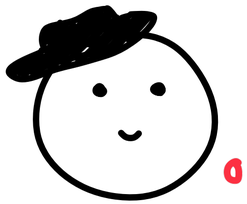
✓ convince Dani of $f(x) = 1$

× while keeping x secret



x





if Alice cheated, she can get away with it with probability 2/3

repeat k times, s.t $(2/3)^k$ is small enough.

MPC for $f'(x, \cdot, \cdot) = f(x)$ with:

- 1-privacy

✓ perfect correctness

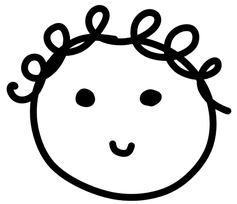
party i did not cheat, and $f(x) = 1$

Constraints s.t.

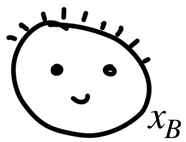
- one reveals nothing

✓ if all of them hold, the claim is true

Tool: Secret Sharing



pick random x_B, x_C
s.t. $x = x_B + x_C \mod m$



x

privacy: x_B, x_C alone look random

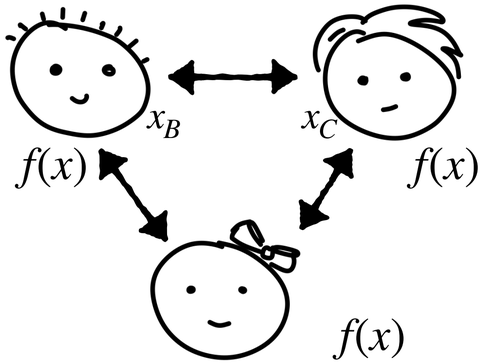
together, they determine x: $x = x_B + x_C \mod m$

ZKP from MPC: Attempt 2 (more parties)

Goal:
- ✓ convince Dani of $f(x) = 1$
- ✓ while keeping x secret

x

Open view B!



if Alice cheated, she can get away with it with probability 2/3

repeat k times, s.t $(2/3)^k$ is small enough.

MPC for $f'(x_B, x_C, \cdot) = f(x_B + x_C)$ with:

- ✓ 1-privacy → ✓ one reveals nothing
- ✓ perfect correctness → ✓ if all of them hold, the claim is true

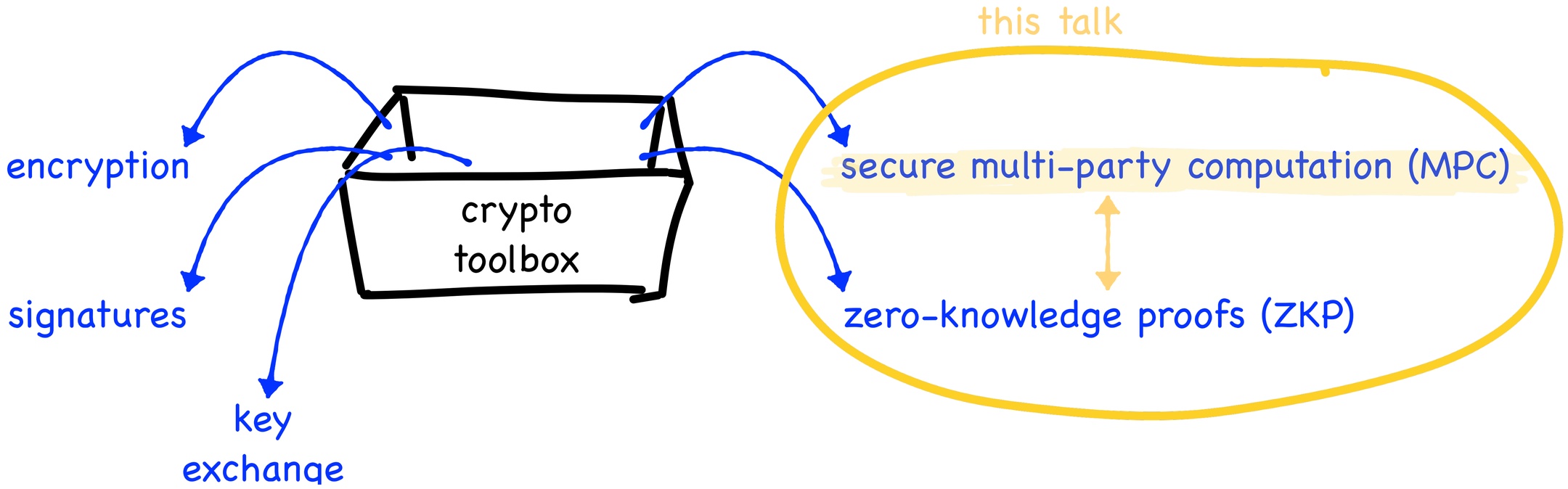
Constraints s.t.

ZKP from MPC

	Communication Complexity	Tools
Reduce to Sudoku (or something...)	poly(|f|)	lightweight (commitments)
Run MPC	O(|f|)	heavyweight (i.e. "public key" operations)
Run MPC in the Head	O(|f|)	

ZKP from MPC

	Communication Complexity	Tools
Reduce to Sudoku (or something...)	poly(|f|)	lightweight (commitments)
Run MPC	O(|f|)	heavyweight (i.e. "public key" operations)
Run MPC in the Head	O(|f|)	lightweight (commitments)


this talk
encryption
signatures
key
exchange
crypto
toolbox
secure multi-party computation (MPC)
zero-knowledge proofs (ZKP)

MPC from Lightweight Tools

Workarounds:

- More participants …

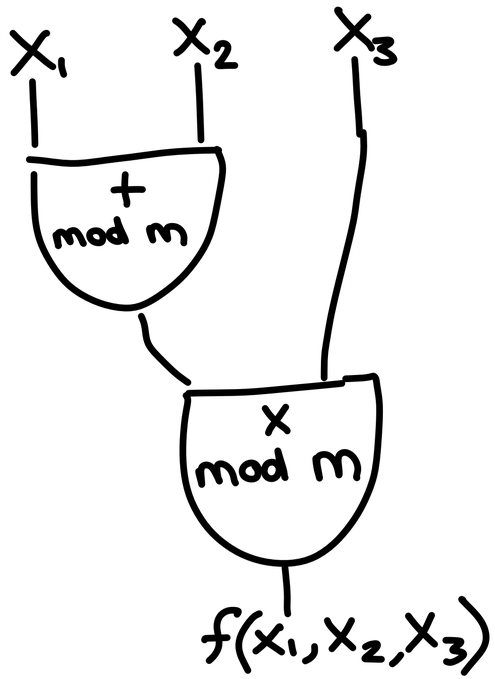
  s.t. we get t-privacy for $t < \frac{n}{2}$, n being the number of participants

  e.g.: $n = 3, t = 1$

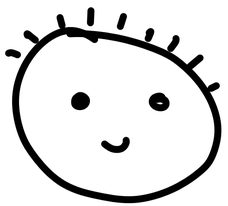
- Correlated randomness

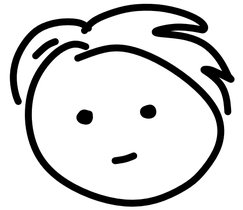
MPC from Correlated Randomness

Step 1: express f as a circuit



Invariant: for wire value w, we have $w = w_B + w_C \pmod m$





Input x: secret share x

Add x and y:

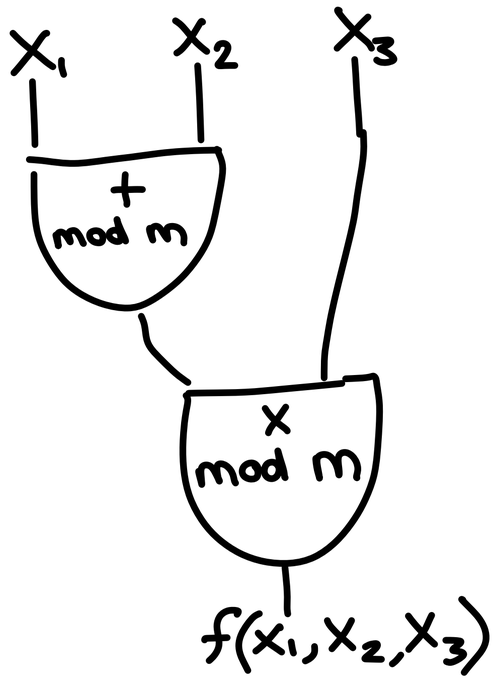
x_B, y_B — x_C, y_C

$z_B = x_B + y_B$ — $z_C = x_C + y_C$

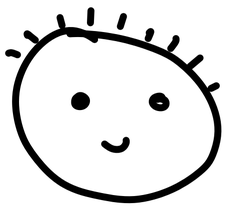
$$z_B + z_C = (x_B + y_B) + (x_C + y_C) = (x_B + x_C) + (y_B + y_C) = x + y \pmod m$$

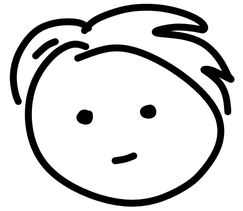
MPC from Correlated Randomness

Step 1: express f as a circuit



Invariant: for wire value w, we have $w = w_B + w_C \pmod m$





Input x: secret share x

Add x and y: $z_B = x_B + y_B$ $\quad z_C = x_C + y_C \pmod m$

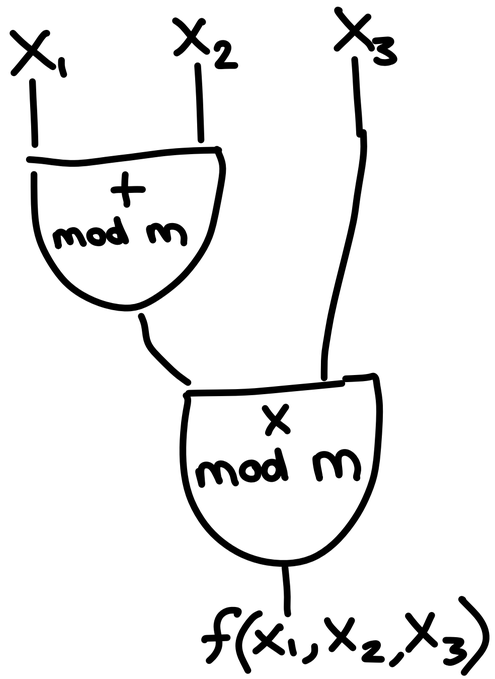
Open w: $w_B \rightleftarrows w_C$

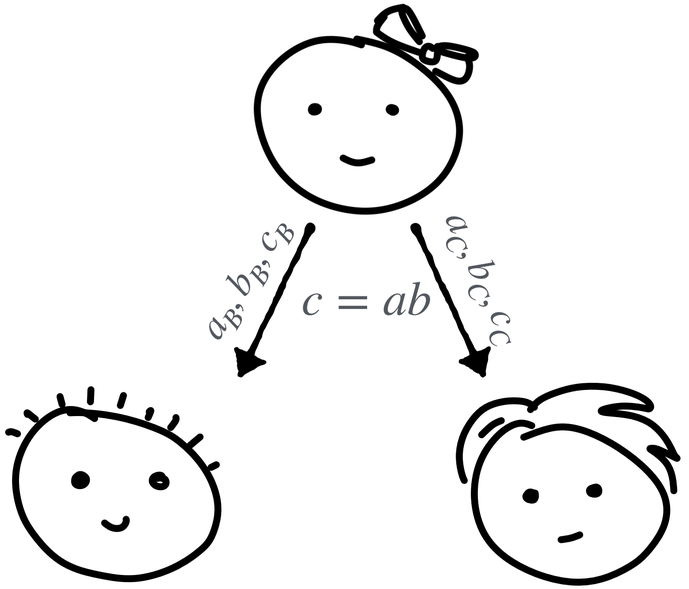
Mult x and y:

$$z = xy = (x_B + x_C)(y_B + y_C)$$
$$= x_B y_B + x_C y_C + x_B y_C + x_C y_B$$

MPC from Correlated Randomness

Step 1: express f as a circuit





Invariant: for wire value w, we have $w = w_B + w_C \pmod m$

Input x: secret share x

Add x and y: $z_B = x_B + y_B$ $\quad z_C = x_C + y_C \pmod m$

Open w: $w_B \rightleftarrows w_C$

Mult x and y: some openings and additions

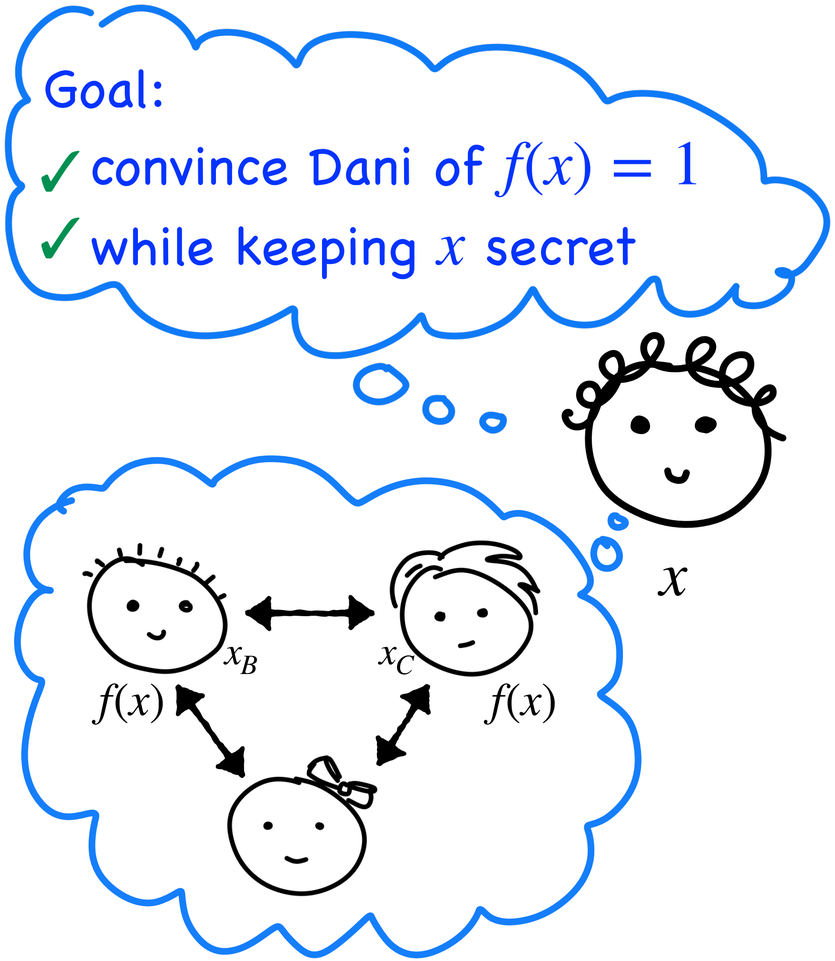
Done!

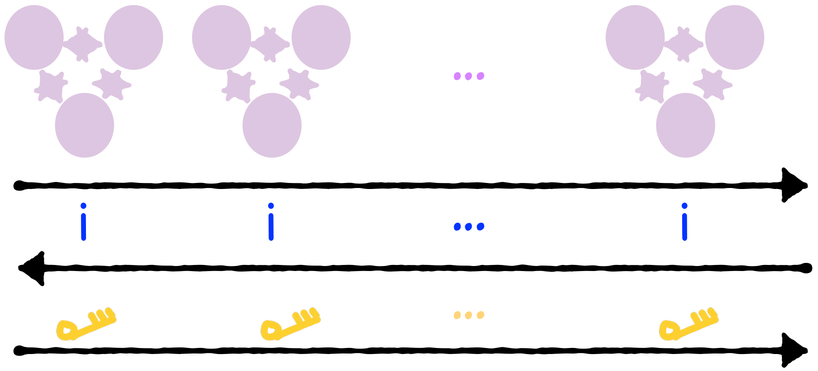
ZKP from MPC: Summary

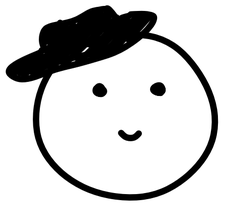
Goal:

✓ convince Dani of $f(x) = 1$

✓ while keeping x secret





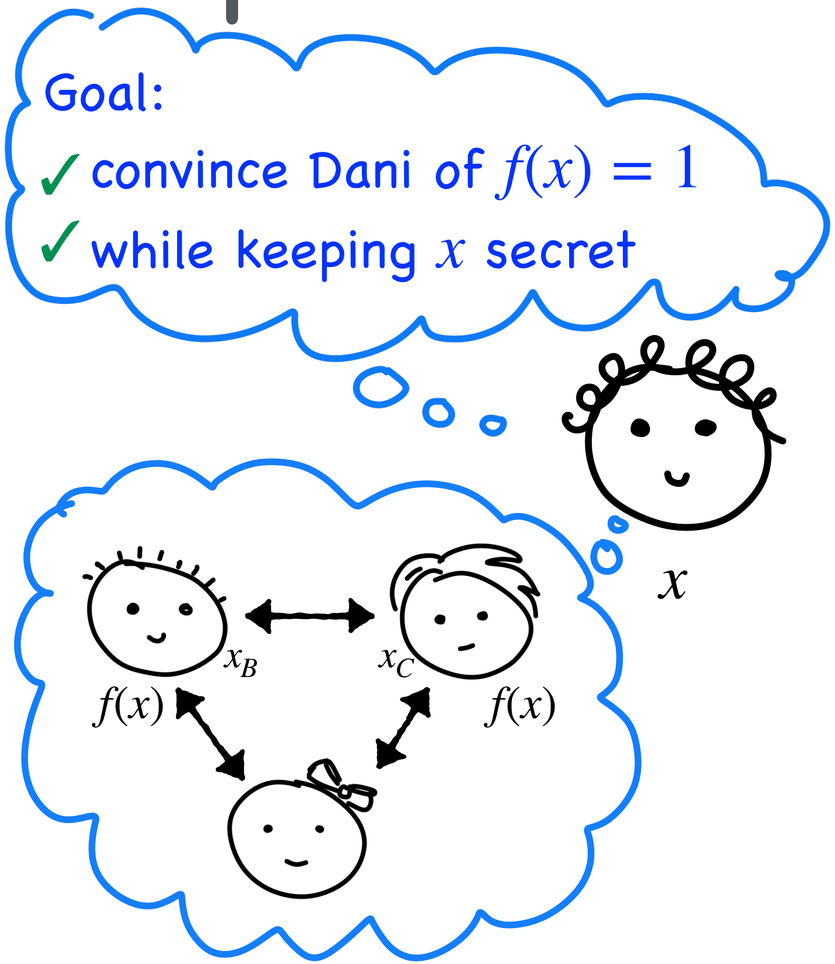


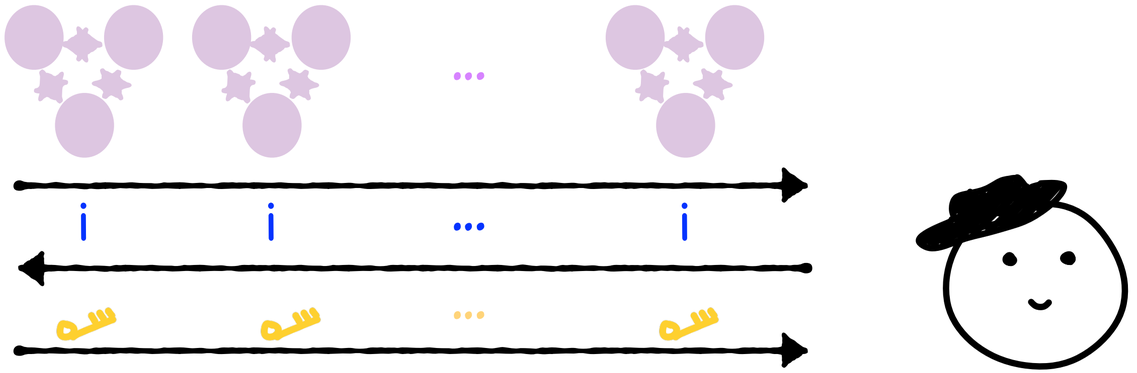
Optimization: from 3 rounds to 1

Goal:

✓ convince Dani of $f(x) = 1$

✓ while keeping x secret



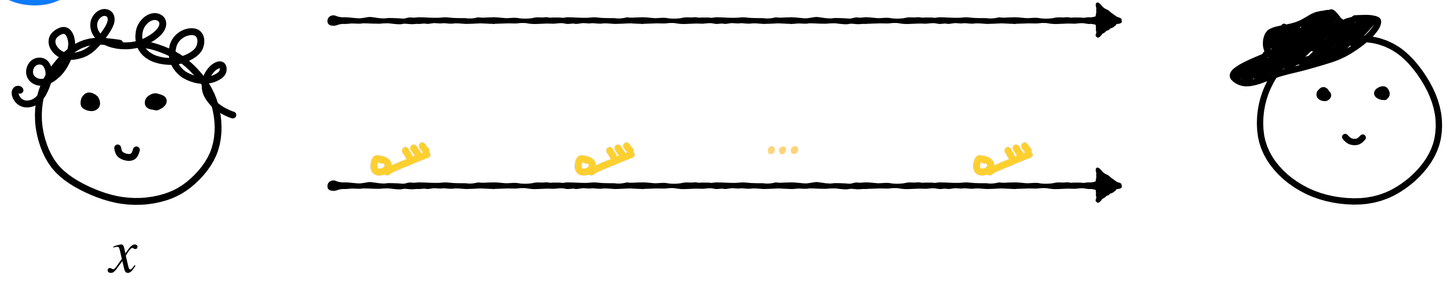


Optimization: from 3 rounds to 1

Goal:

✓ convince Dani of $f(x) = 1$

✓ while keeping x secret



x

i, i, …, i random

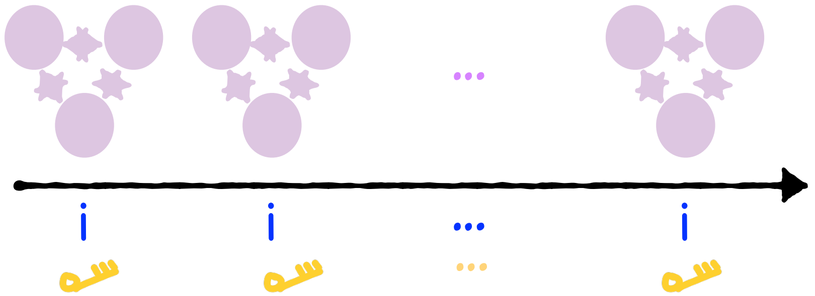
Optimization: from 3 rounds to 1

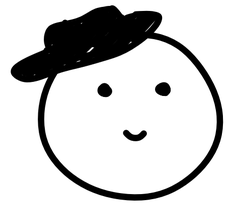
Goal:

✓ convince Dani of $f(x) = 1$

✓ while keeping x secret

x





i, i, …, i random

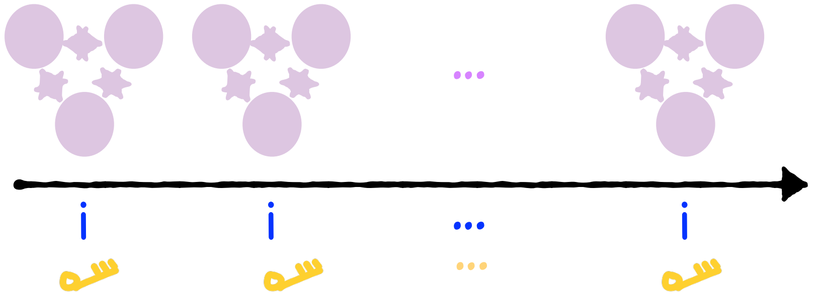
Optimization: from 3 rounds to 1

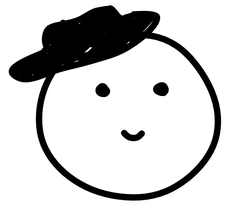
Goal:

✓ convince Dani of $f(x) = 1$

✓ while keeping x secret

x



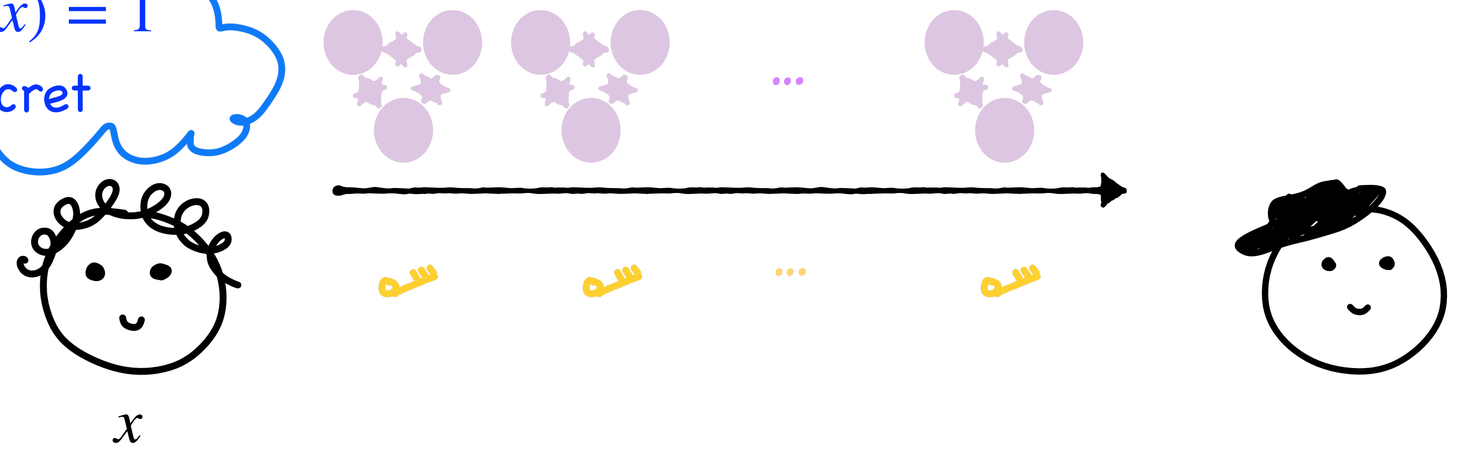


i, i, …, i = H( )

H is a one-way function; hard to sample i's before

Optimization: from 3 rounds to 1

Goal:
✓ convince Dani of $f(x) = 1$
✓ while keeping x secret



x

i, i, …, i = H( )

H is a one-way function; hard to sample i's before

i, i, …, i = H( )

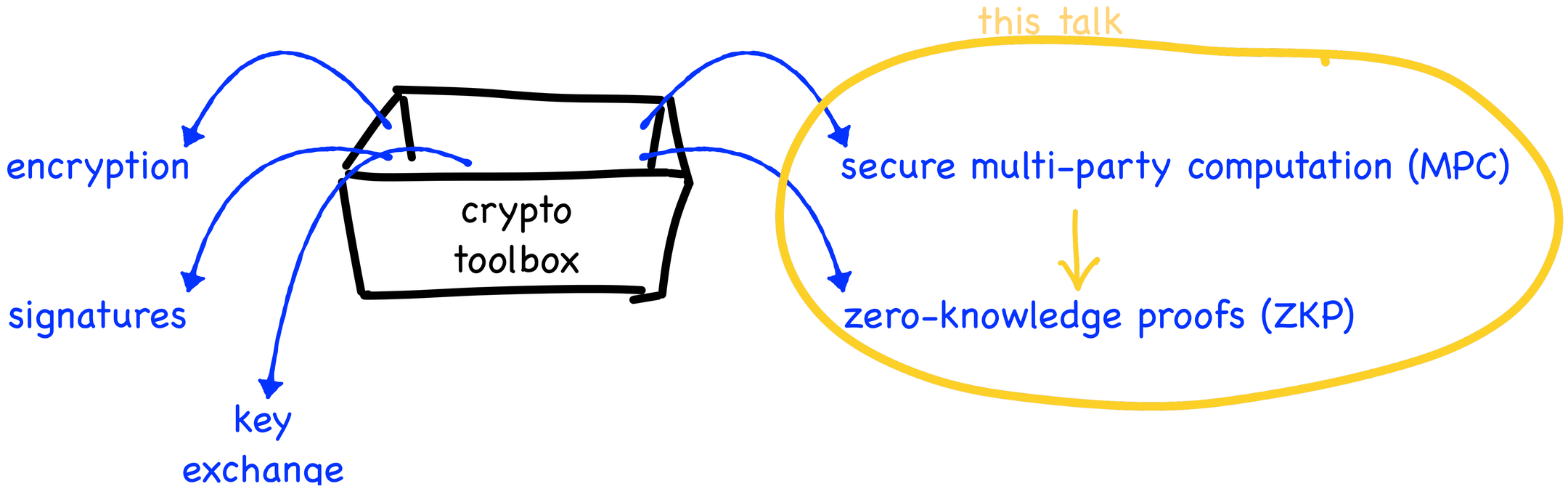
For each i:

Open(,) → party i's view

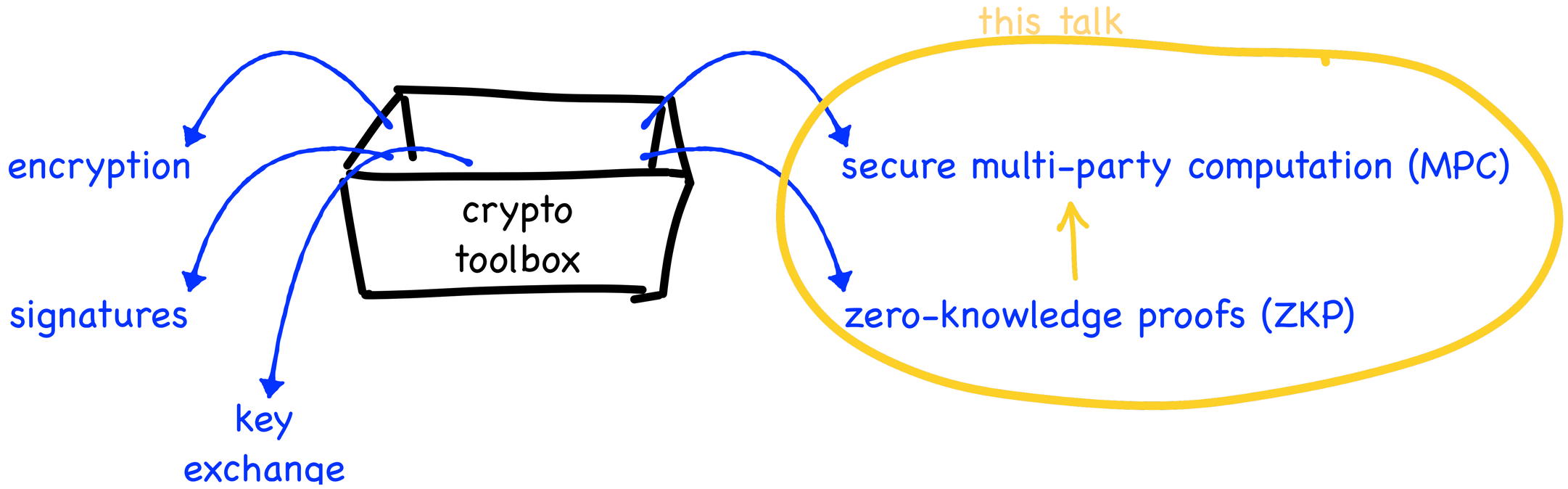
Check that:

- party i did not cheat, and
- $f(x) = 1$

Up til now...

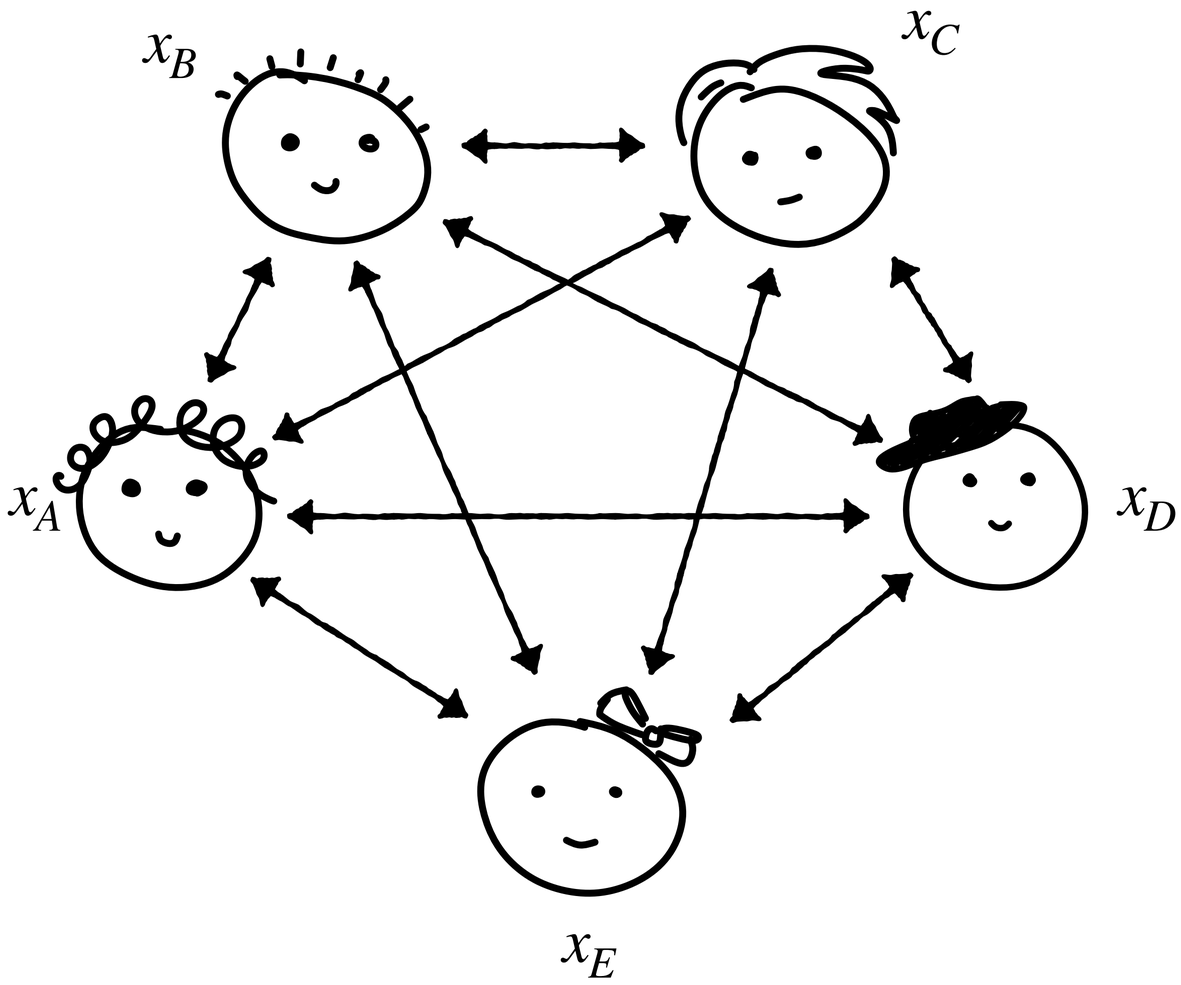

this talk
encryption
signatures
key
exchange
crypto
toolbox
secure multi-party computation (MPC)
zero-knowledge proofs (ZKP)

Briefly:


this talk
encryption
signatures
key
exchange
crypto
toolbox
secure multi-party computation (MPC)
zero-knowledge proofs (ZKP)

Back to MPC

What about "active" corruptions?



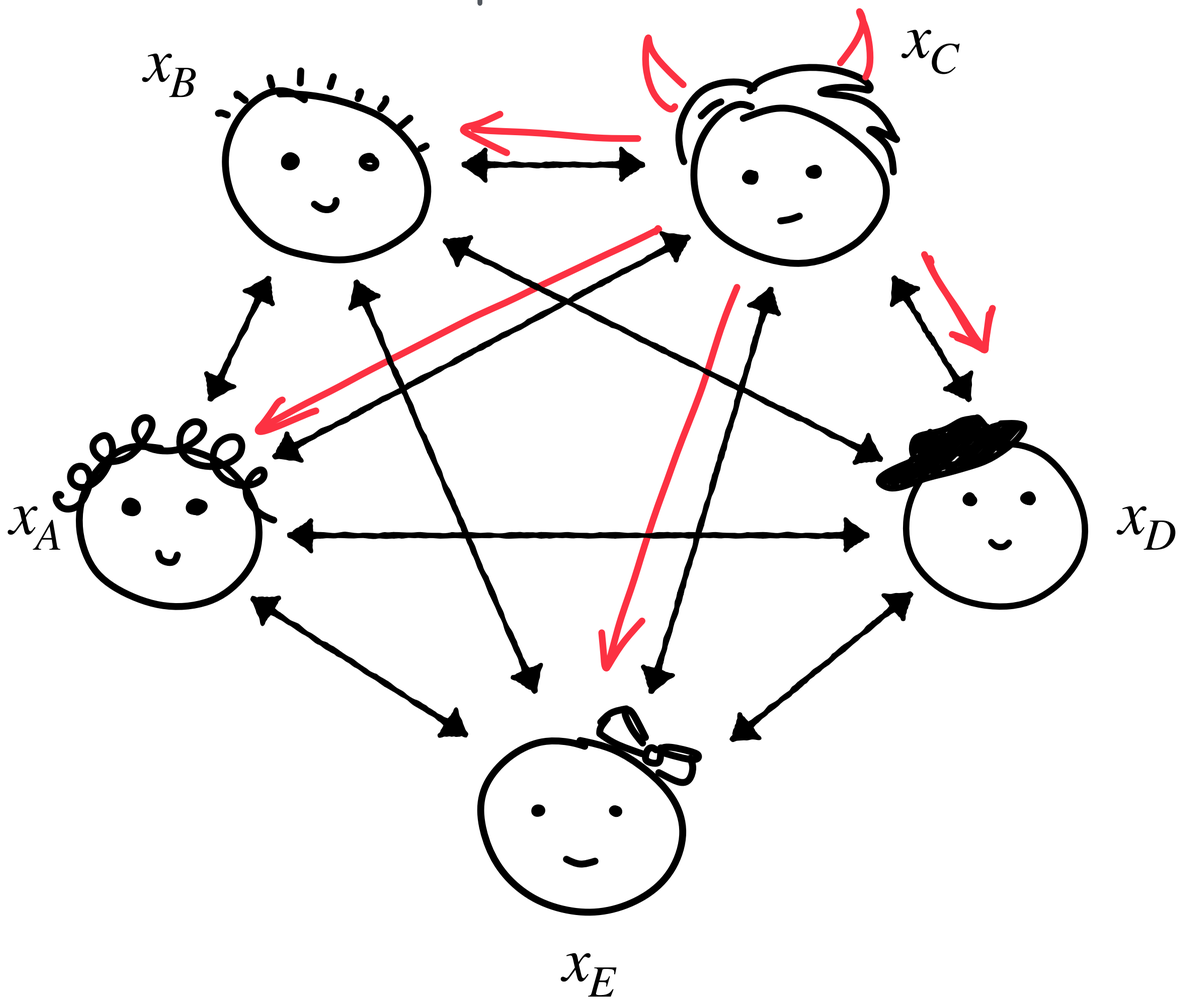
We have:

- correctness
- privacy

as long as everyone follows instructions.

Back to MPC

What about "active" corruptions?



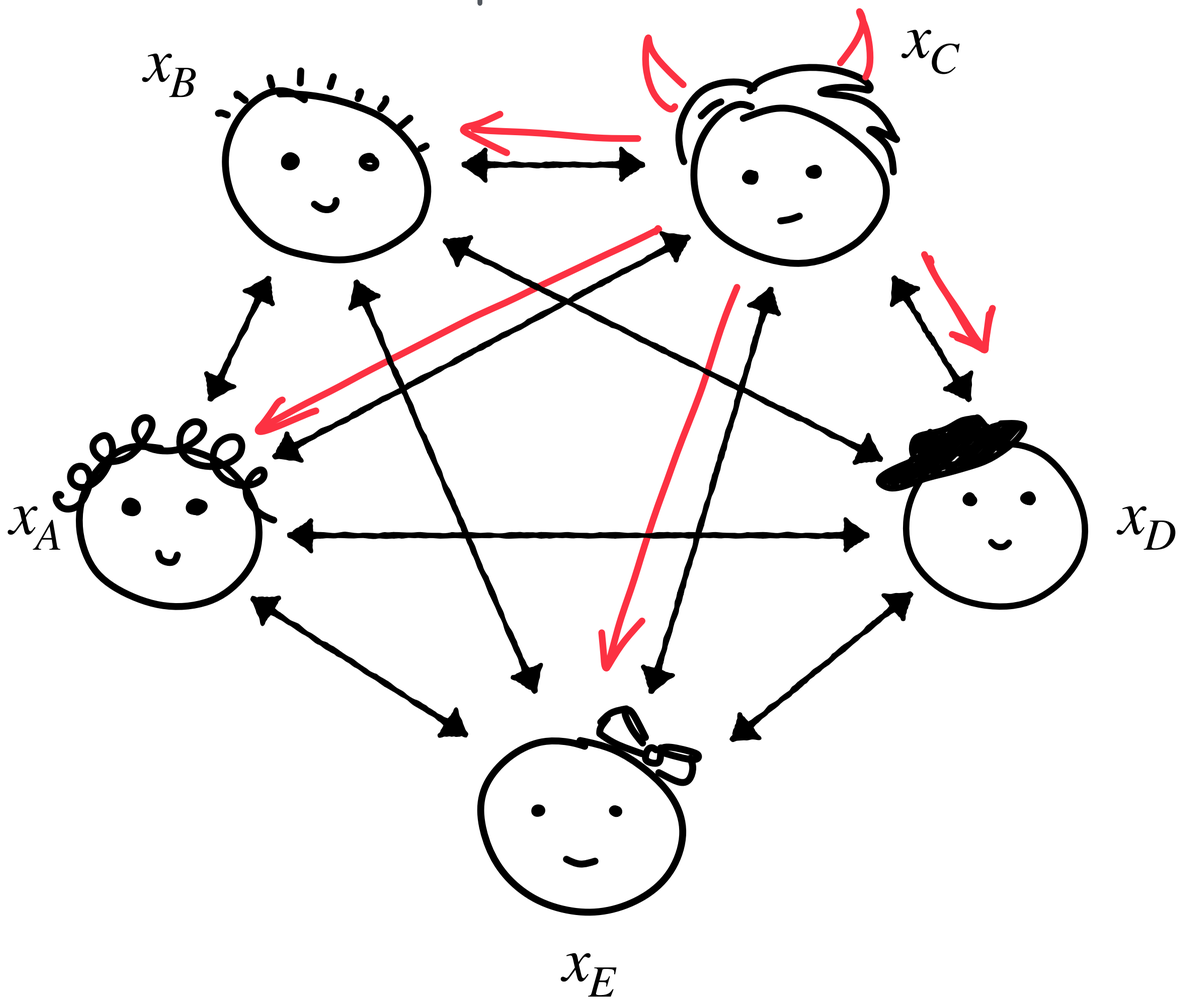
We have:

- correctness
- privacy

as long as everyone follows instructions.

Back to MPC

What about "active" corruptions?



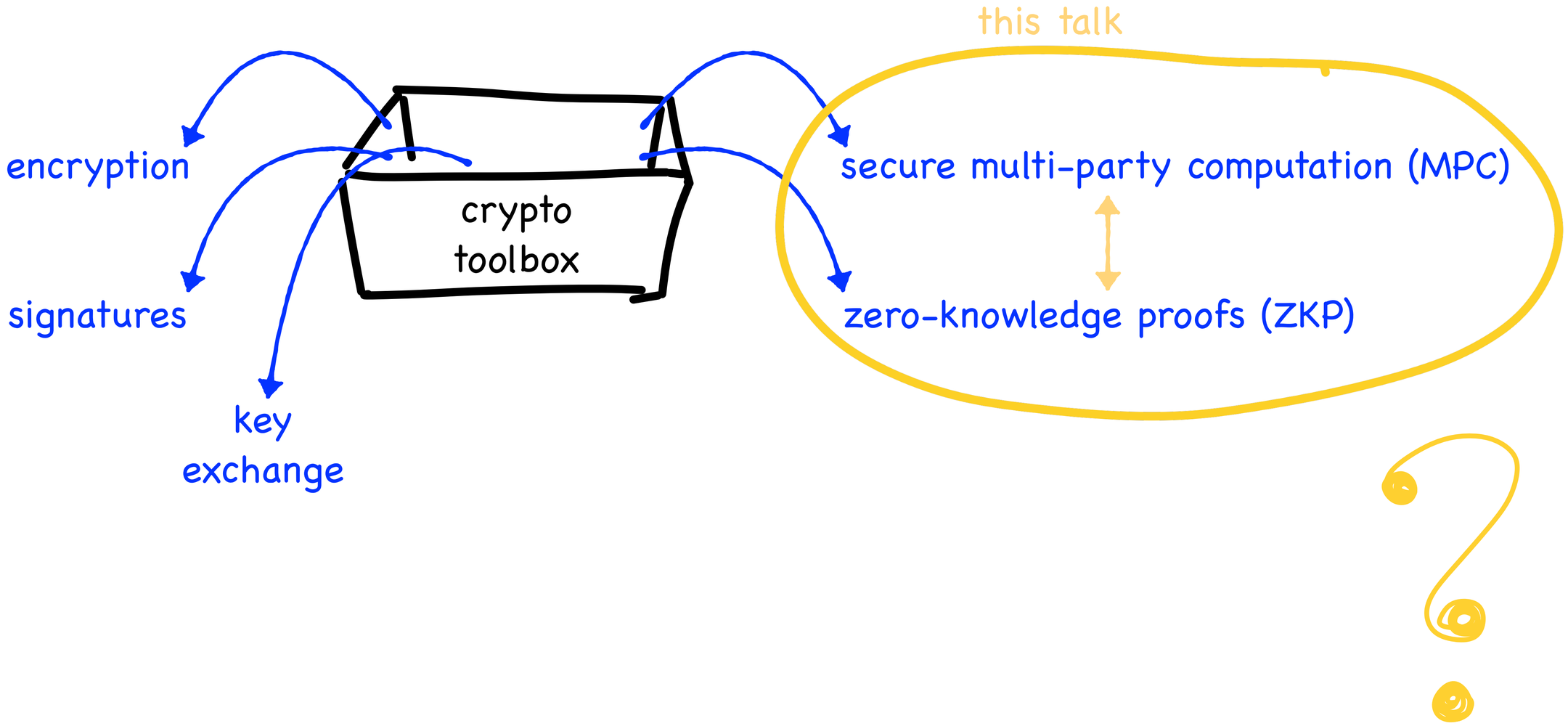
We want:

- correctness
- privacy

even if up to t participants cheat!

take a protocol secure against "passive" corruptions, and have each participant zero-knowledge-prove their correct behavior!

Done!


this talk
encryption
signatures
key
exchange
crypto
toolbox
secure multi-party computation (MPC)
zero-knowledge proofs (ZKP)