

Cheater Identification on a Budget: MPC with Identifiable Abort from Pairwise MACs

Carsten Baum

DTU

Nikolas Melissaris

Aarhus University

Rahul Rachuri

Visa Research

Peter Scholl

Aarhus University

MPC with IA via Vindicating Release

Ran
Cohen



Jack
Doerner



Yashvanth
Kondi



abhi
shelat



<https://ia.cr/2023/1136>

Cheater Identification on a Budget: MPC with Identifiable Abort from Pairwise MACs

Carsten Baum, Nikolas Melissaris, Rahul Rachuri, Peter Scholl

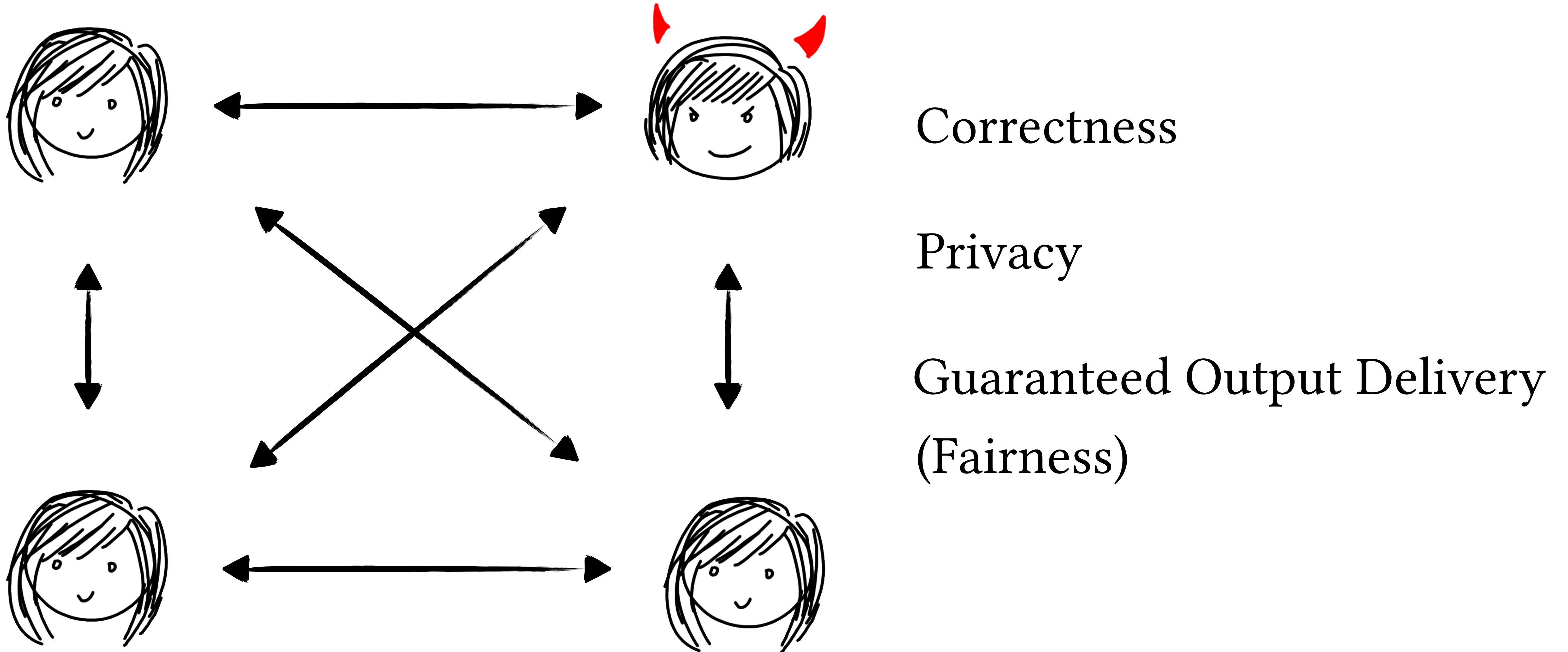
DTU

Aarhus University

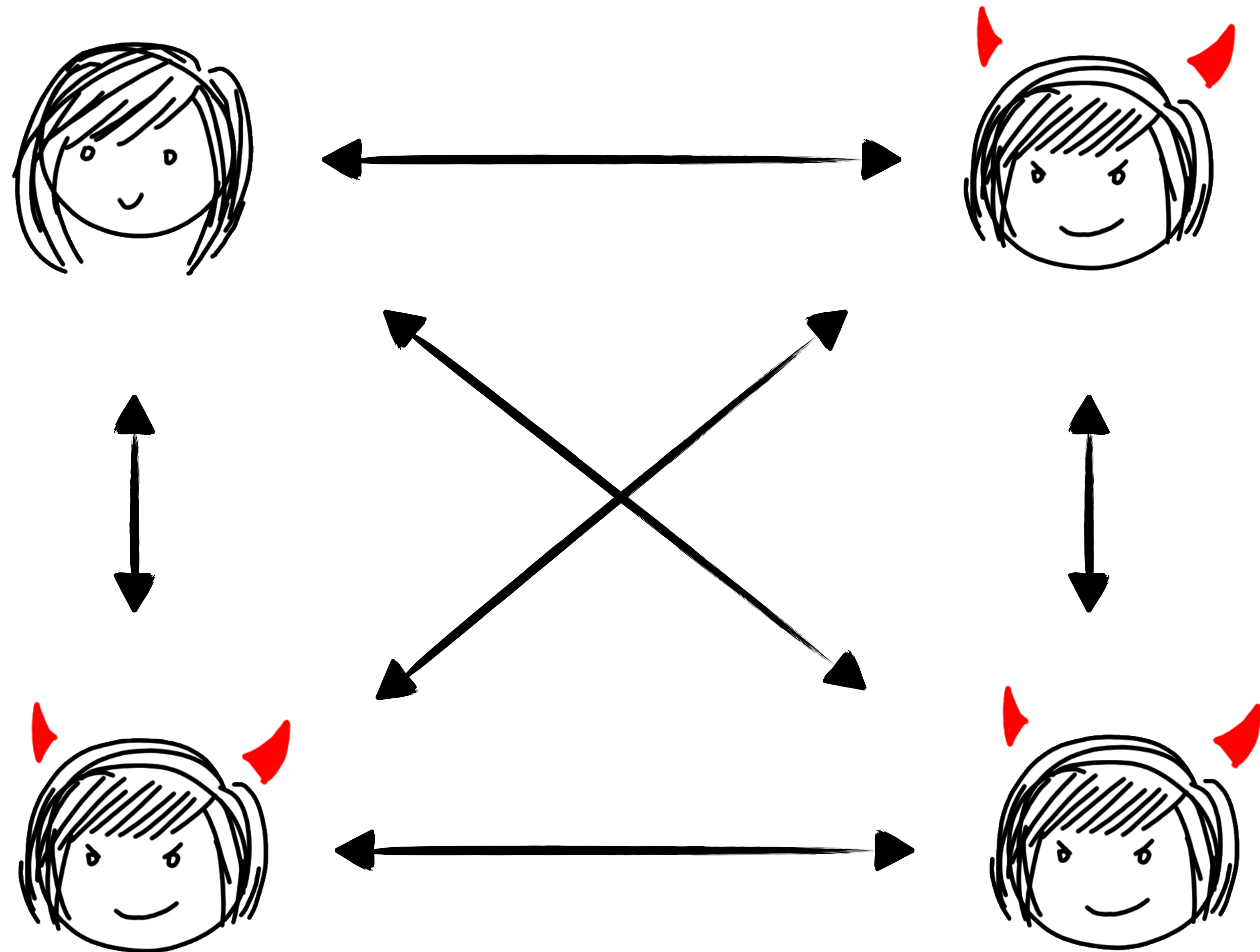
Visa Research

Aarhus University

Multiparty Computation (MPC)



Multiparty Computation (MPC)



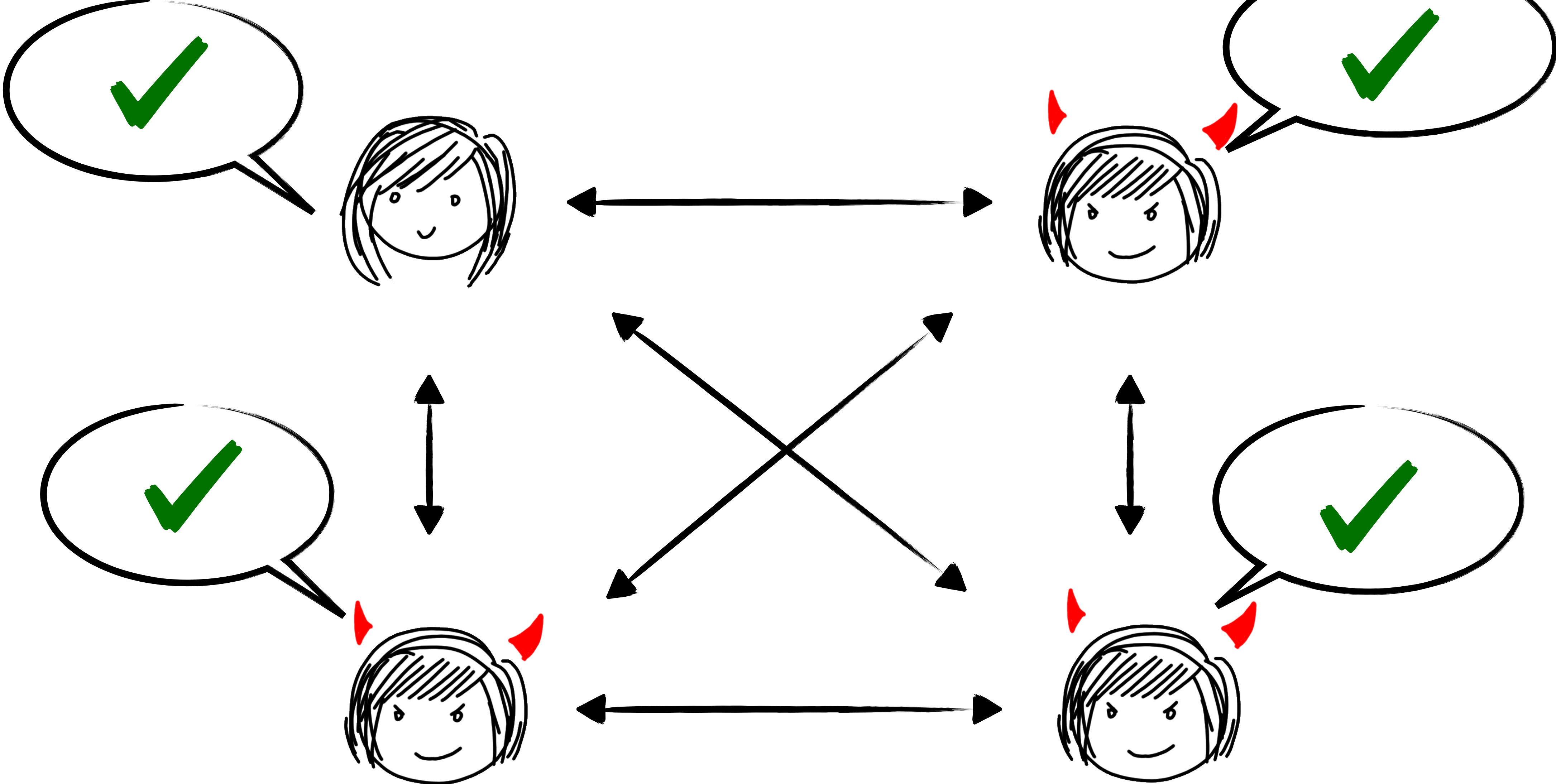
Correctness

Privacy

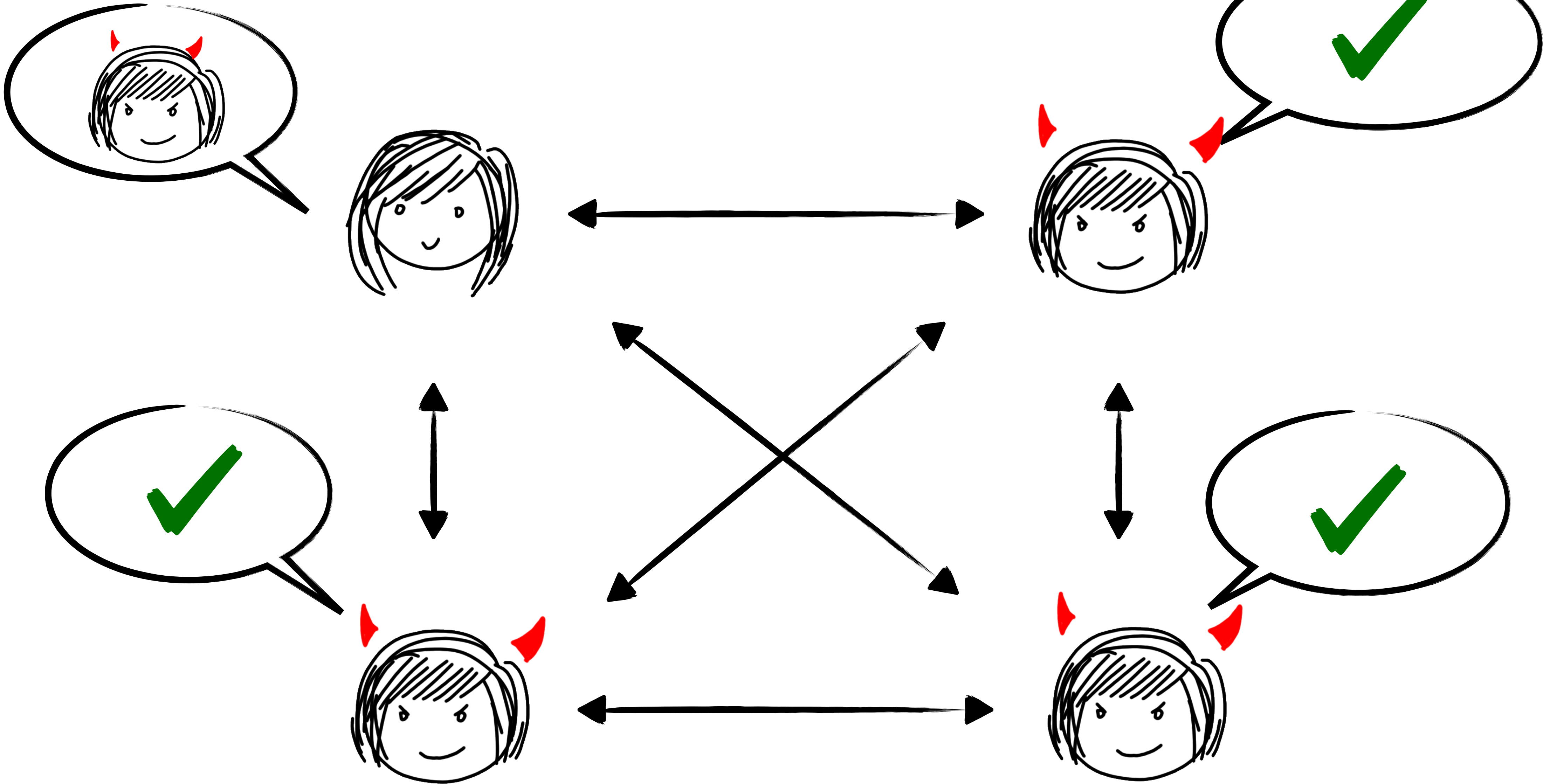
~~Guaranteed Output Delivery~~
(Fairness)

[Cleve86]

MPC with Identifiable Abort

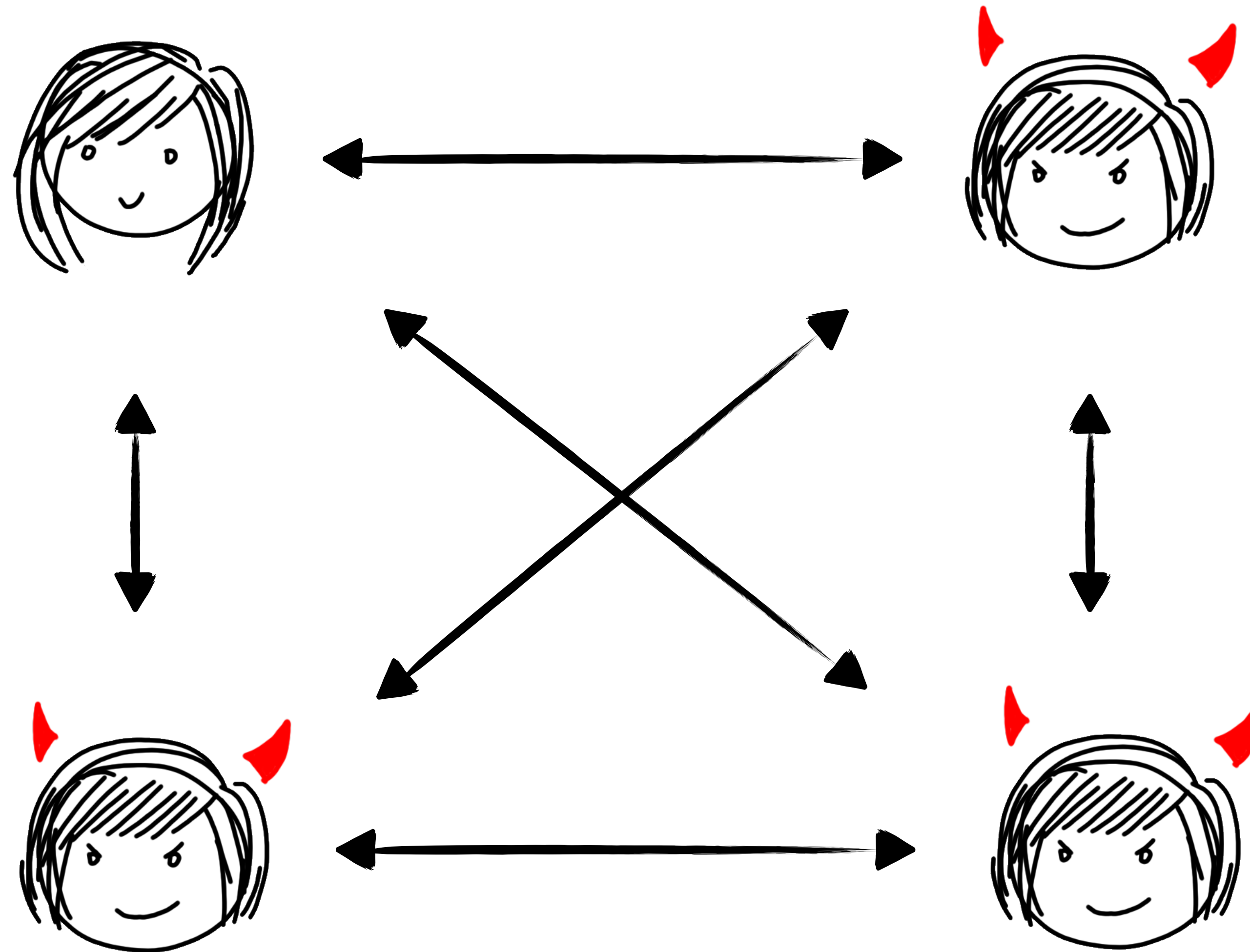


MPC with Identifiable Abort



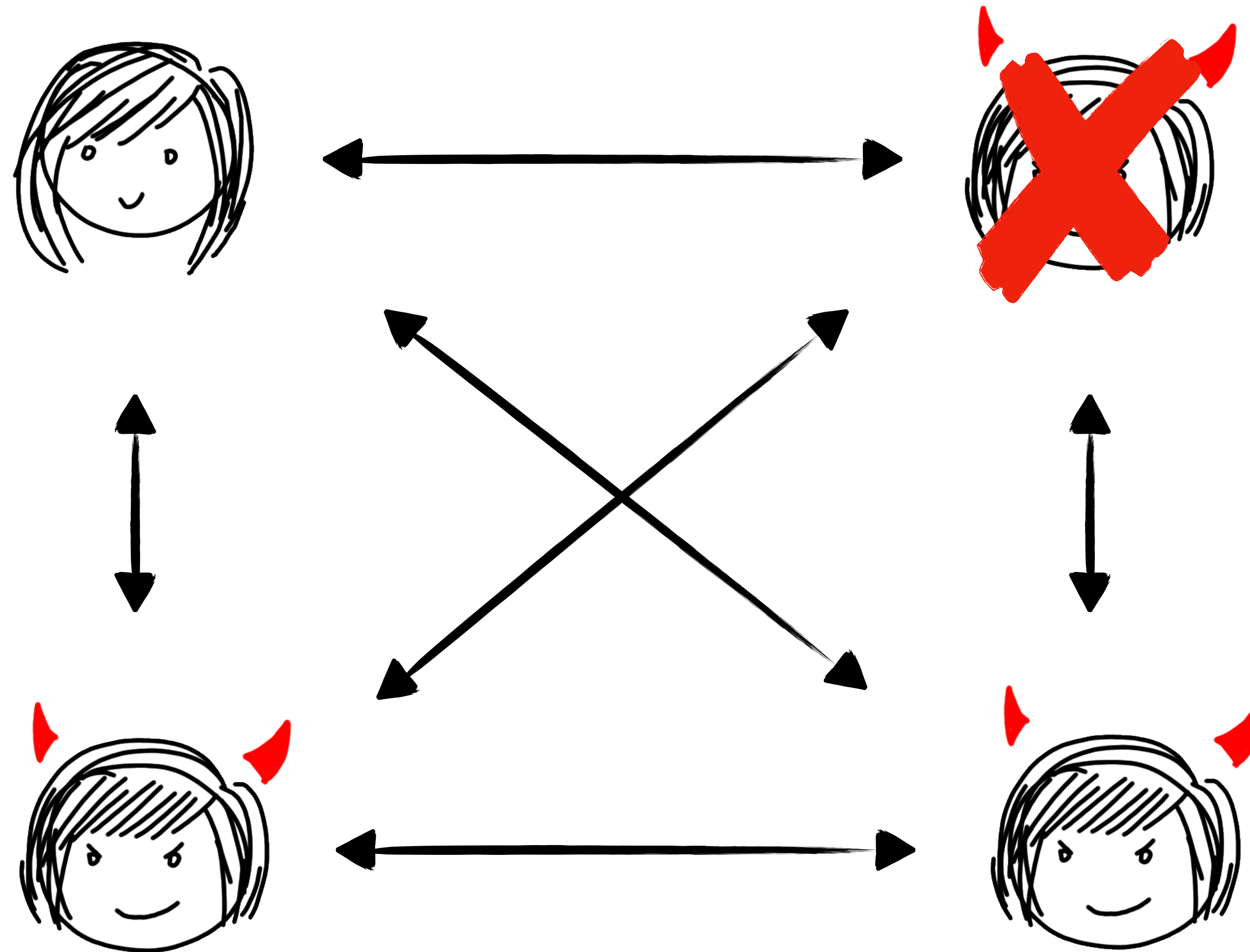
MPC with Identifiable Abort

Why do we care?



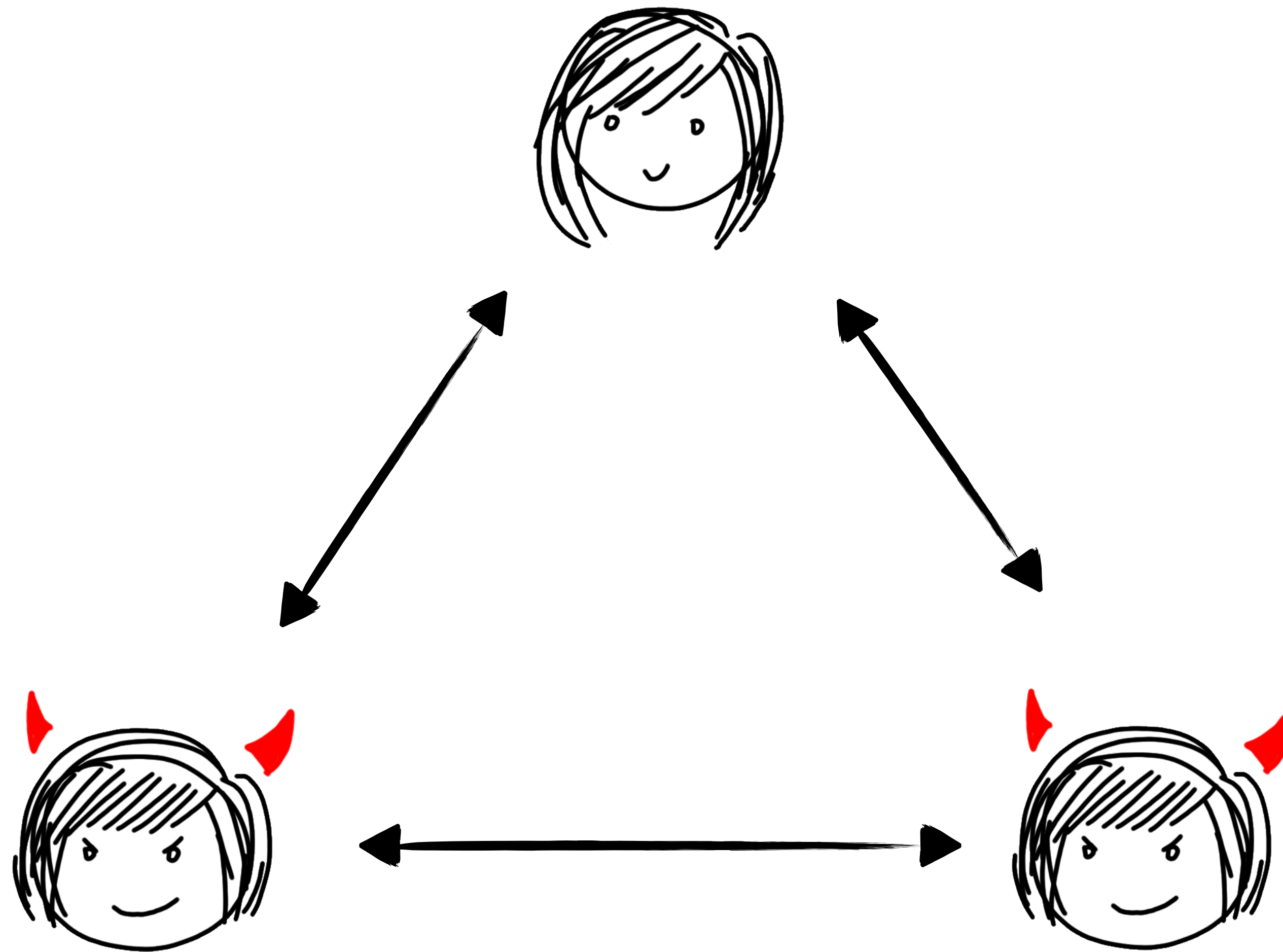
MPC with Identifiable Abort

Why do we care?



MPC with Identifiable Abort

Why do we care?



[IOZ14]

Take a protocol Π and add preprocessing.

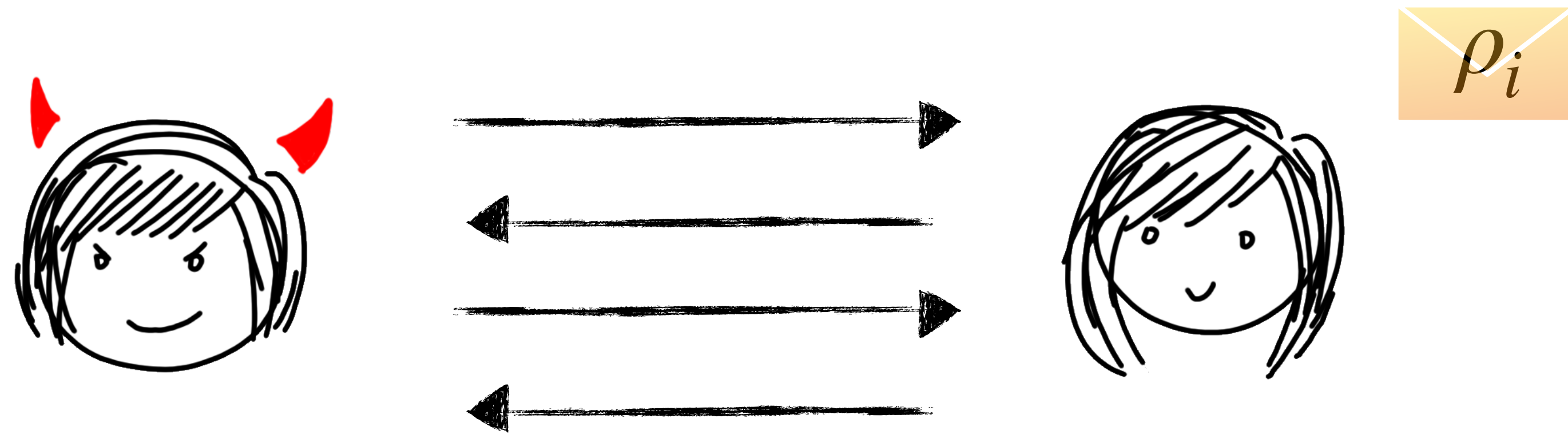
Preprocessing

1. Parties commit to their random tape.
2. Correlated randomness: preprocess ZK proofs.
3. Run all communication through a broadcast channel.
4. In case of a **complaint**: open commitments to random tapes.

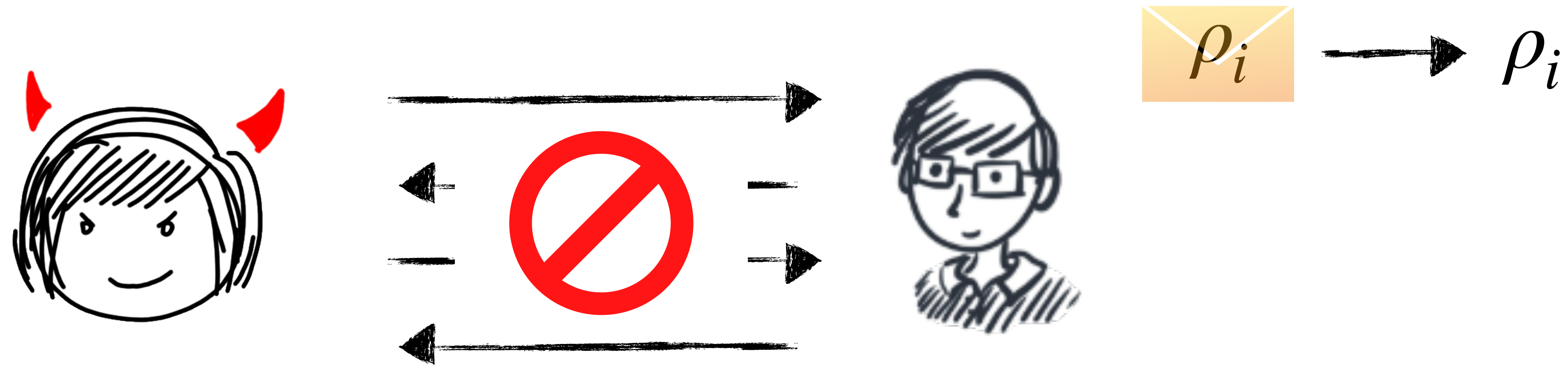
Online

Run Π and prove in ZK that all messages are well formed.

[IOZ14]



[IOZ14]



[IOZ14]



First construction that makes only black-box use of cryptographic primitives

[IOZ14]



First construction that makes only black-box use of cryptographic primitives

Yes, but

Adaptively secure OT



Proving every step, each round in ZK



Related Work

[BOS16, SF16, CFY17]

Avoid generic ZK, but still expensive preprocessing

[BOSS20]

Avoid ZK and adaptive OT, but only for (Boolean) garbled circuits

Our Contribution



Compiler for upgrading Sender - Receiver protocols to IA



Actively secure ID-MPC with small overhead for large prime fields



New technical tool in proof: [Online extractability](#)

Better Identifiable Abort

Has input



Main Tool - $\mathcal{F}_{\text{HCom}}$

$\mathcal{F}_{\text{HCom}}$

Sender commits to x

For committed values x, y
sender can commit to $\alpha x + \beta y + \gamma$

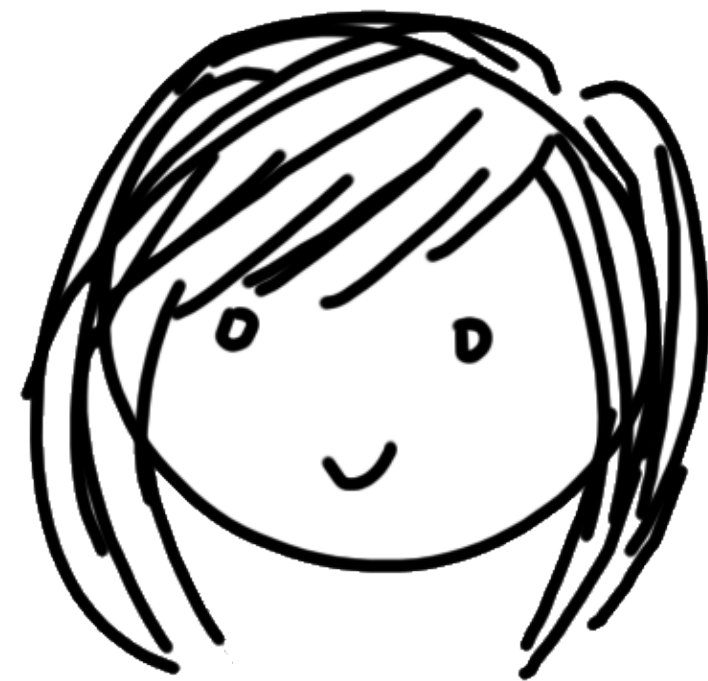
Sender can open commitment x ,
or abort



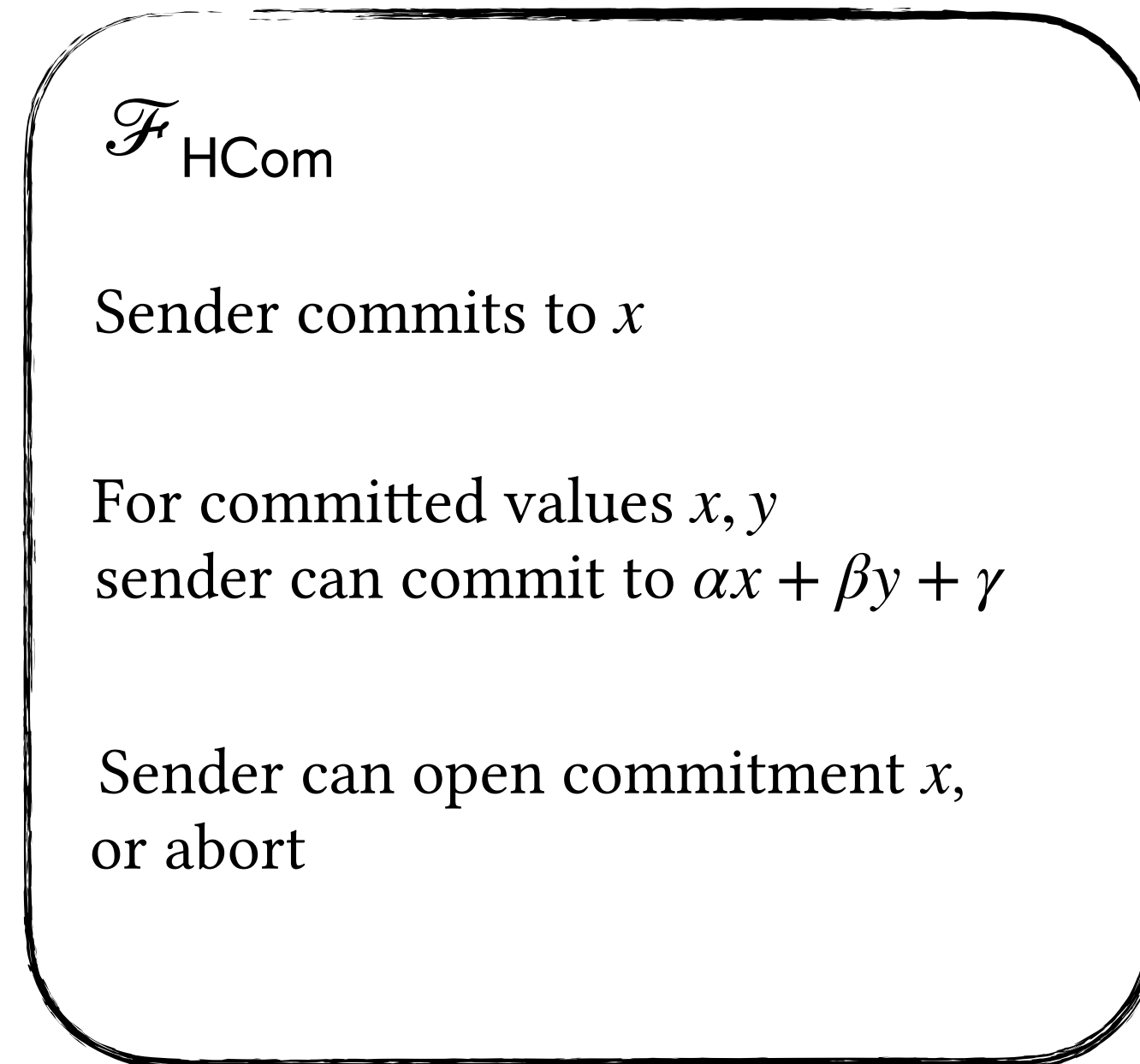
No input

Better Identifiable Abort

Has input



Main Tool - $\mathcal{F}_{\text{HCom}}$



No input

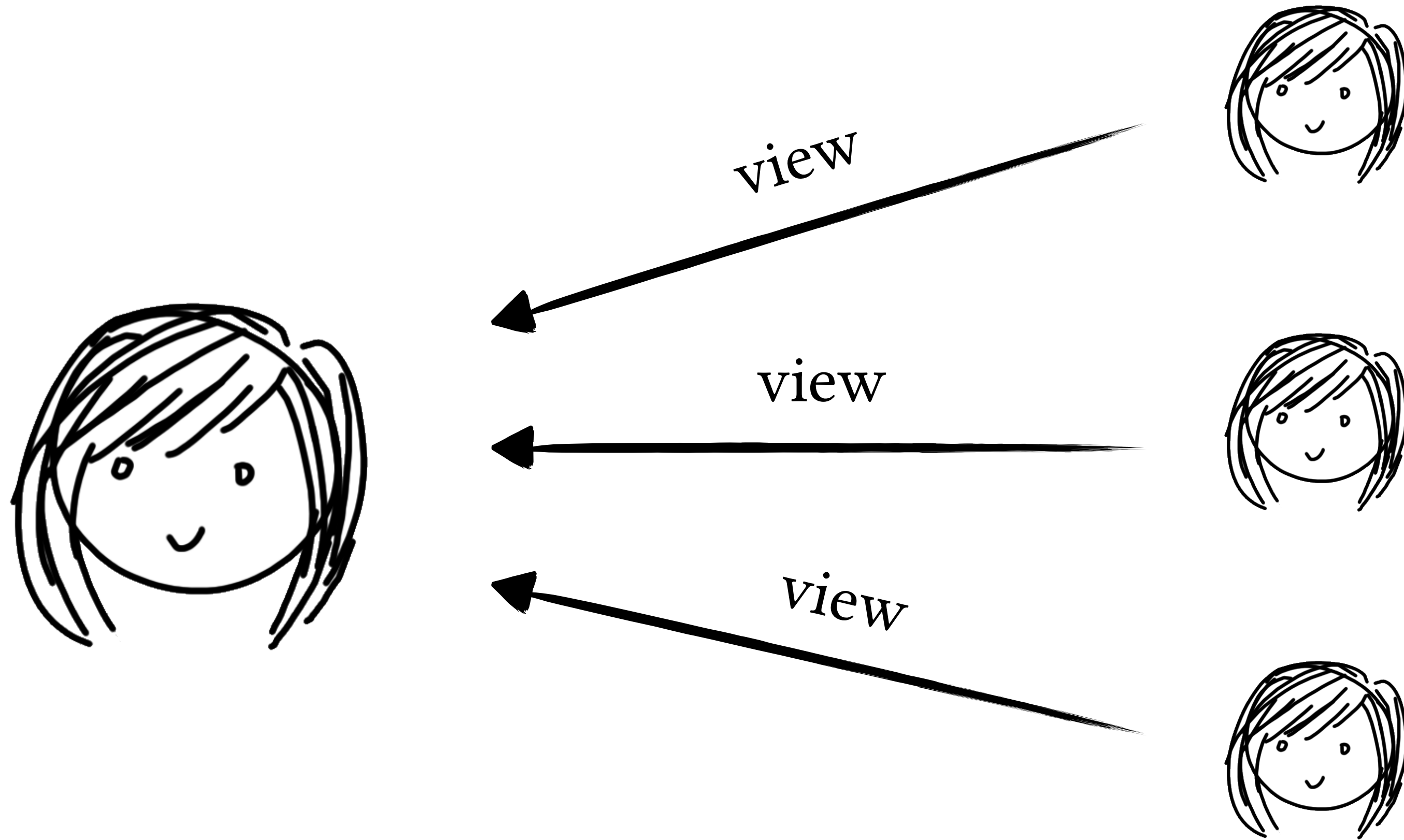
Better Identifiable Abort

Sender claims abort



Better Identifiable Abort

Sender claims abort

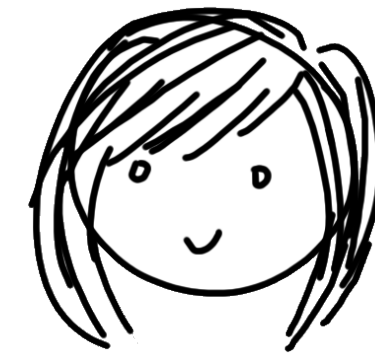
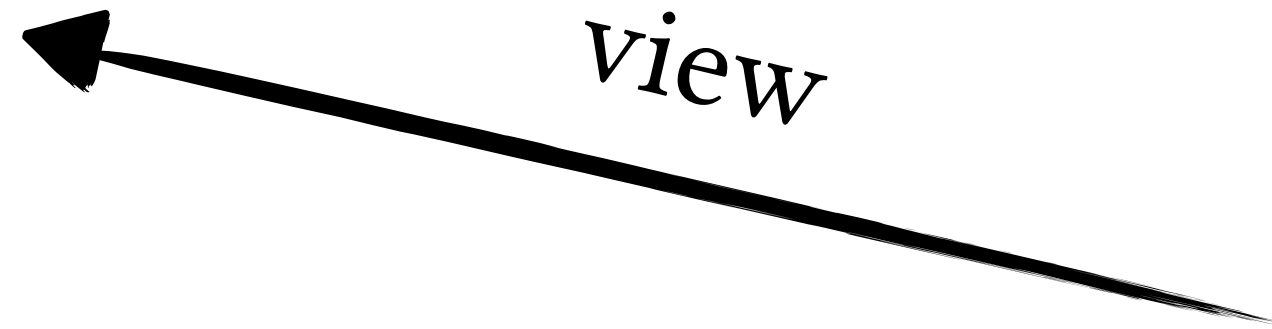
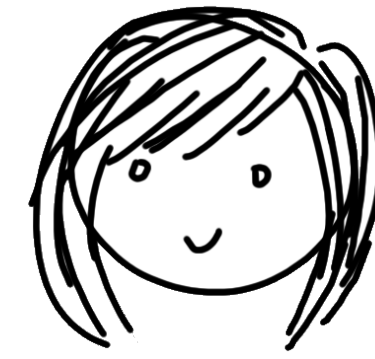
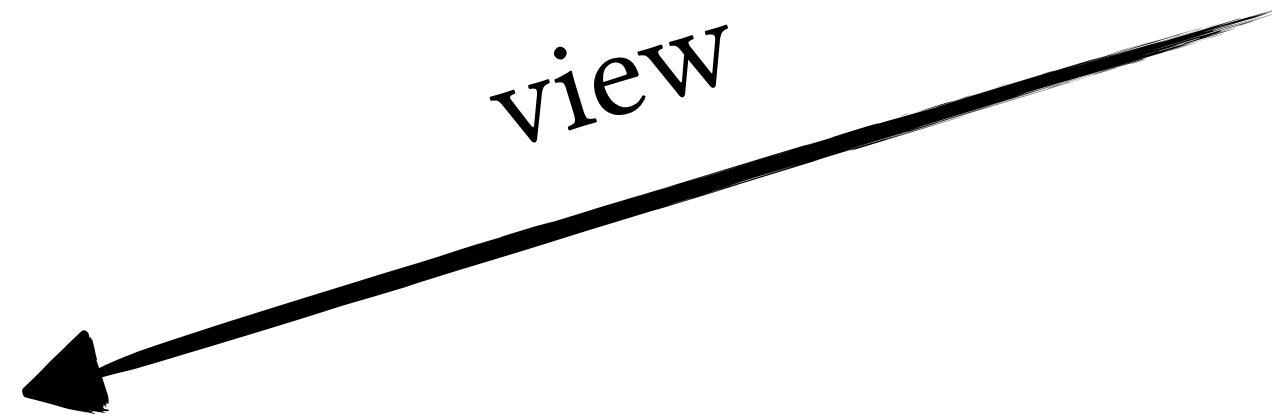


Better Identifiable Abort

Sender claims abort

This is who cheated

(but with proof)

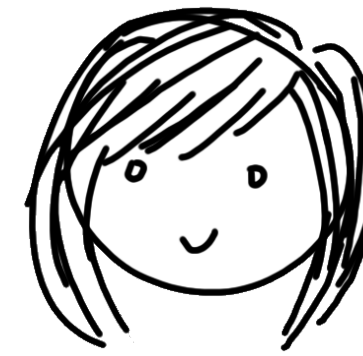
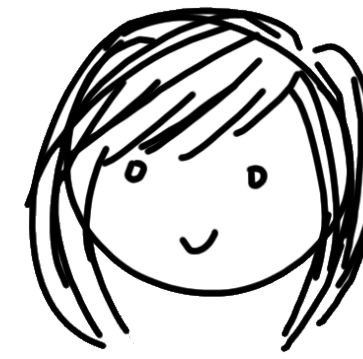


Better Identifiable Abort

Receiver claims abort

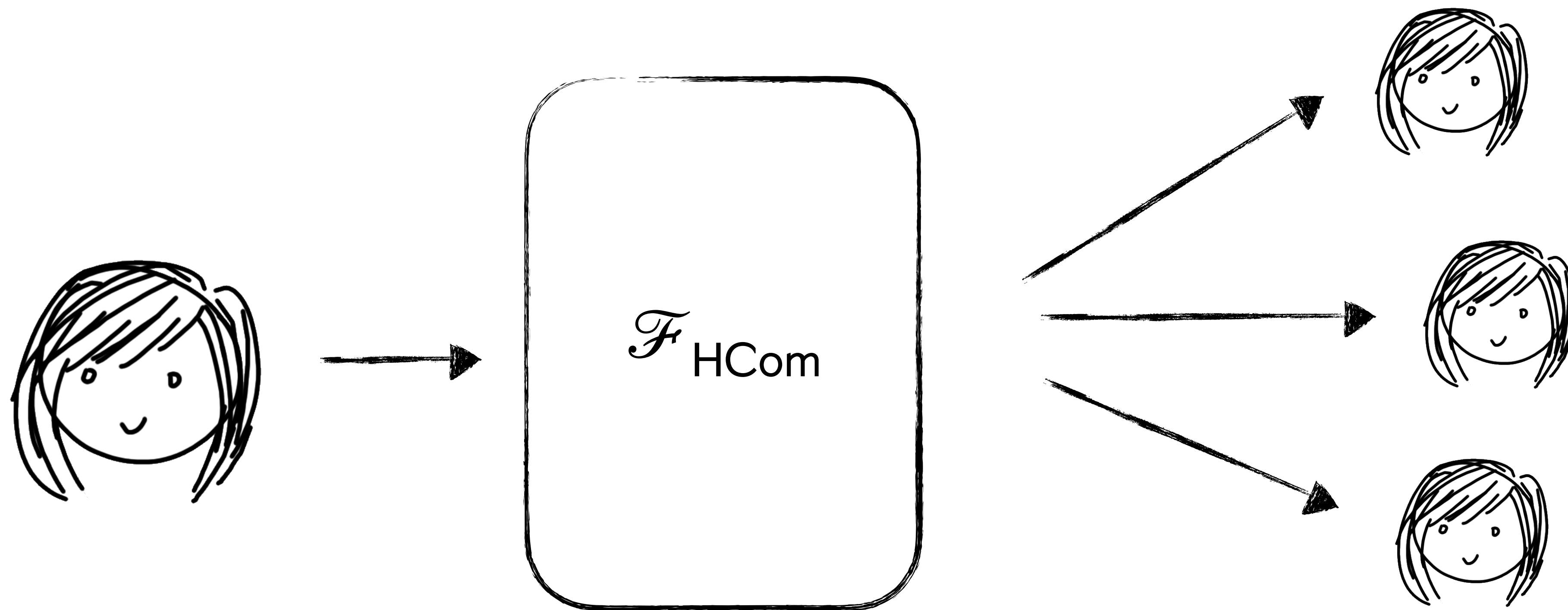


This is what I've seen



Better Identifiable Abort

Do we really need adaptive security?



Better Identifiable Abort Online Extractability

What is it?

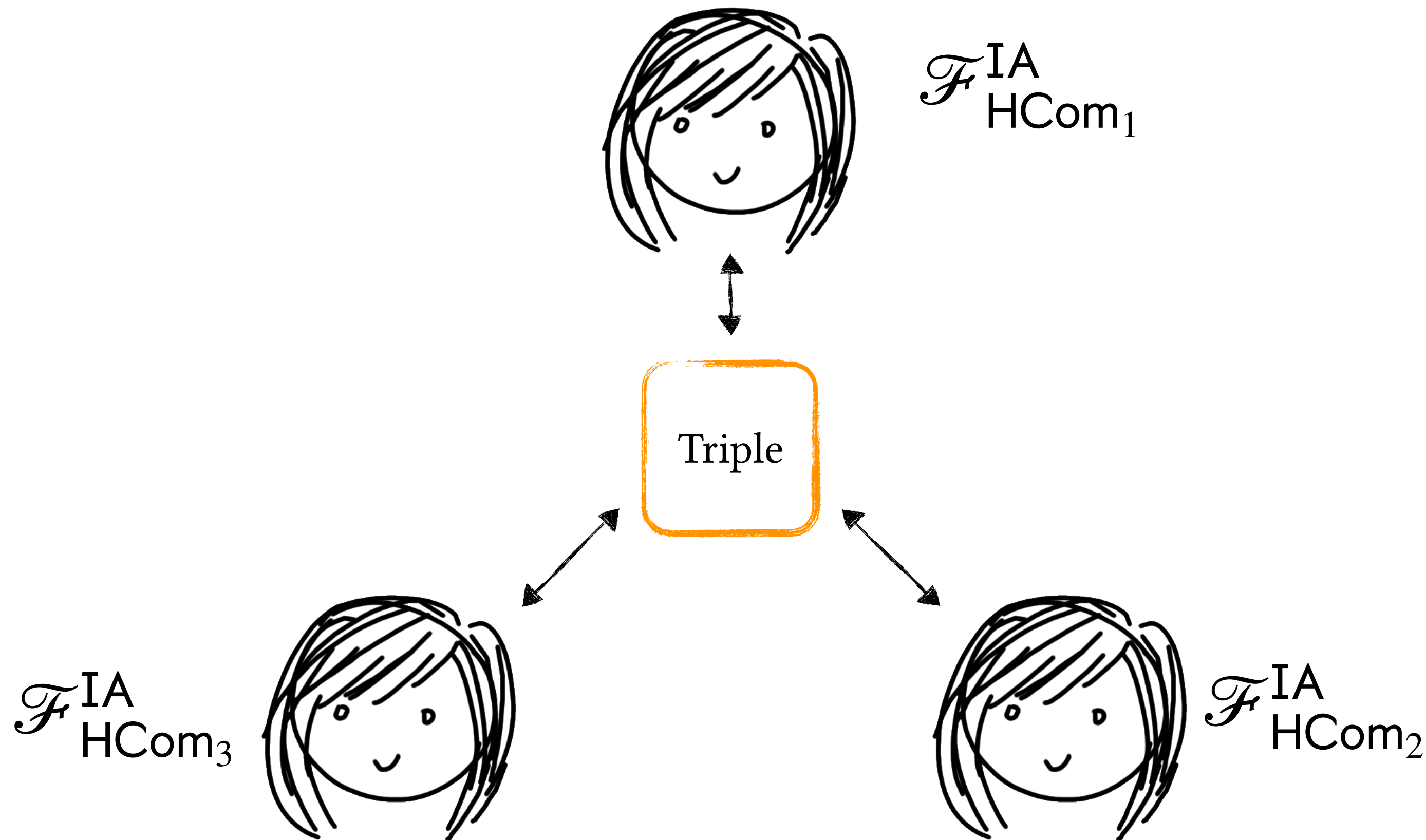
Special type of simulation

How does it work?

Normal protocol execution suffices for adversarial input extraction

(with tiny changes to the CRS)

From $\mathcal{F}_{\text{HCom}}$ to ID-MPC



1. Commit to random shares using $\mathcal{F}_{\text{HCom}}^{\text{IA}}$
 2. Commit to tapes for triple generation
 3. Generate random triples from the random tapes
 4. Commit to inputs/outputs of triple generation using $\mathcal{F}_{\text{HCom}}^{\text{IA}}$
 5. Run triple sacrifice
- If success: commit to inputs via $\mathcal{F}_{\text{HCom}}^{\text{IA}}$ run online phase
If fail: Identify cheaters

Efficiency

Without Identifiable Abort

	Preprocessing	Online
Le Mans v1 [RS22]	$n^2 \cdot \text{OLE}$	$12n$
Le Mans v2 [RS22]	$n^2 \cdot \text{OLE} + O(n)$	$4n$

With Identifiable Abort

	Preprocessing	Online
Our work	$n^2 \cdot \text{OLE} + O(n^2)$	$O(n^2)$

Efficiency

Without Identifiable Abort

	Preprocessing	Online
Le Mans v1 [RS22]	$n^2 \cdot \text{OLE}$	$12n$
Le Mans v2 [RS22]	$n^2 \cdot \text{OLE} + O(n)$	$4n$

With Identifiable Abort

	Preprocessing	Online
Our work	$n^2 \cdot \text{OLE} + O(n^2)$	$\leq 2n^2$

Corrupt party can force broadcast



MPC with IA via Vindicating Release

Ran
Cohen



Jack
Doerner



Yashvanth
Kondi



abhi
shelat



<https://ia.cr/2023/1136>

Vindicating Release

The naïve approach:
if something goes wrong, open
your internal state to show that
you computed honestly.

(Typically requires adaptive
security - but not today!)

(Simplified) Summary of Techniques

Identifiable Abort

- [Goldreich Micali Wigderson 87]
ZK over underlying protocol
NBB use of crypto
- [Ishai Ostrovsky Zikas 14]
MPCitH + opening tape if prep fails
Adaptively secure OT protocol
- [Baum Orsini Scholl 16]
ZK over somewhat homomorphic encryption
- [Baum Orsini Scholl Soria-Vasquez 20]
Additive homomorphic commitments
+ OT protocol + CCRH for online phase
(boolean only)

(Simplified) Summary of Techniques

Identifiable Abort

- [Goldreich Micali Wigderson 87]
ZK over underlying protocol
NBB use of crypto
- [Ishai Ostrovsky Zikas 14]
MPCitH + opening tape if prep fails
Adaptively secure OT protocol
- [Baum Orsini Scholl 16]
ZK over somewhat homomorphic encryption
- [Baum Orsini Scholl Soria-Vasquez 20]
Additive homomorphic commitments
+ OT protocol + CCRH for online phase
(boolean only)

Construct (at least one)
Protocol Compiler

(Simplified) Summary of Techniques

Identifiable Abort

- [Goldreich Micali Wigderson 87]
ZK over underlying protocol
NBB use of crypto
- [Ishai Ostrovsky Zikas 14]
MPCitH + opening tape if prep fails
Adaptively secure OT protocol
- [Baum Orsini Scholl 16]
ZK over somewhat homomorphic encryption
- [Baum Orsini Scholl Soria-Vasquez 20]
Additive homomorphic commitments
+ OT protocol + CCRH for online phase
(boolean only)

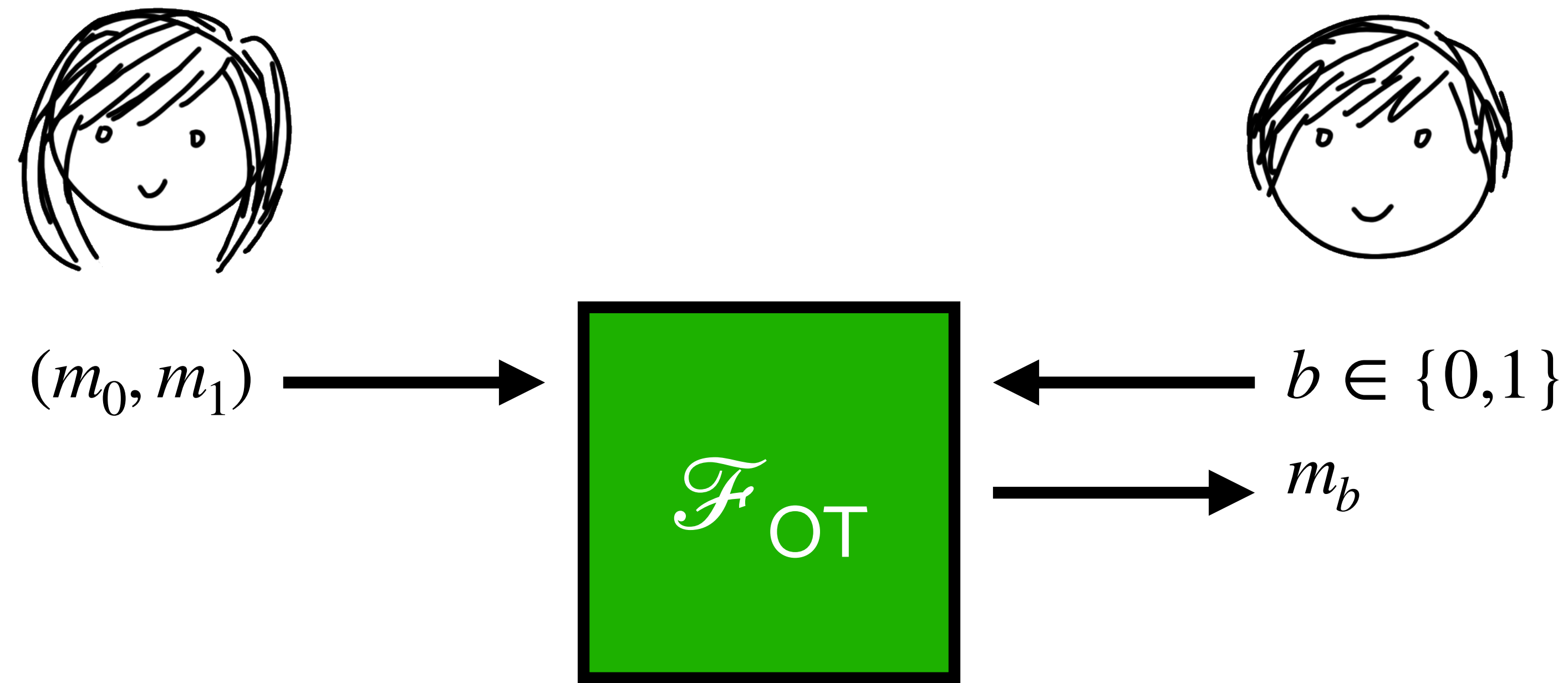
Non-Identifiable Abort

- Many advanced protocols using exotic primitives!
- Simple, widely-recognized
Fundamental Primitive:
Oblivious Transfer [Kilian 88, IPS 08]
- We can construct protocols that are IT-secure in the *OT-hybrid* model
 - Easy to understand
 - Easy to implement
 - Efficient enough for deployment
 - Often modular
- e.g. MASCOT [Keller Orsini Scholl 16]

Our Goal

1. Propose fundamental primitive
2. Construct generic MPC
 - IT only in hybrid model of fundamental primitive
 - Add IA to well-known constructions using Vindicating Release
 - Reusable modules (e.g. VOLE)
3. Don't use the words
 - “Non-black-box”
 - “Adaptive”
 - “Homomorphic”
 - “Compiler”
 - “Straight-line Extraction”

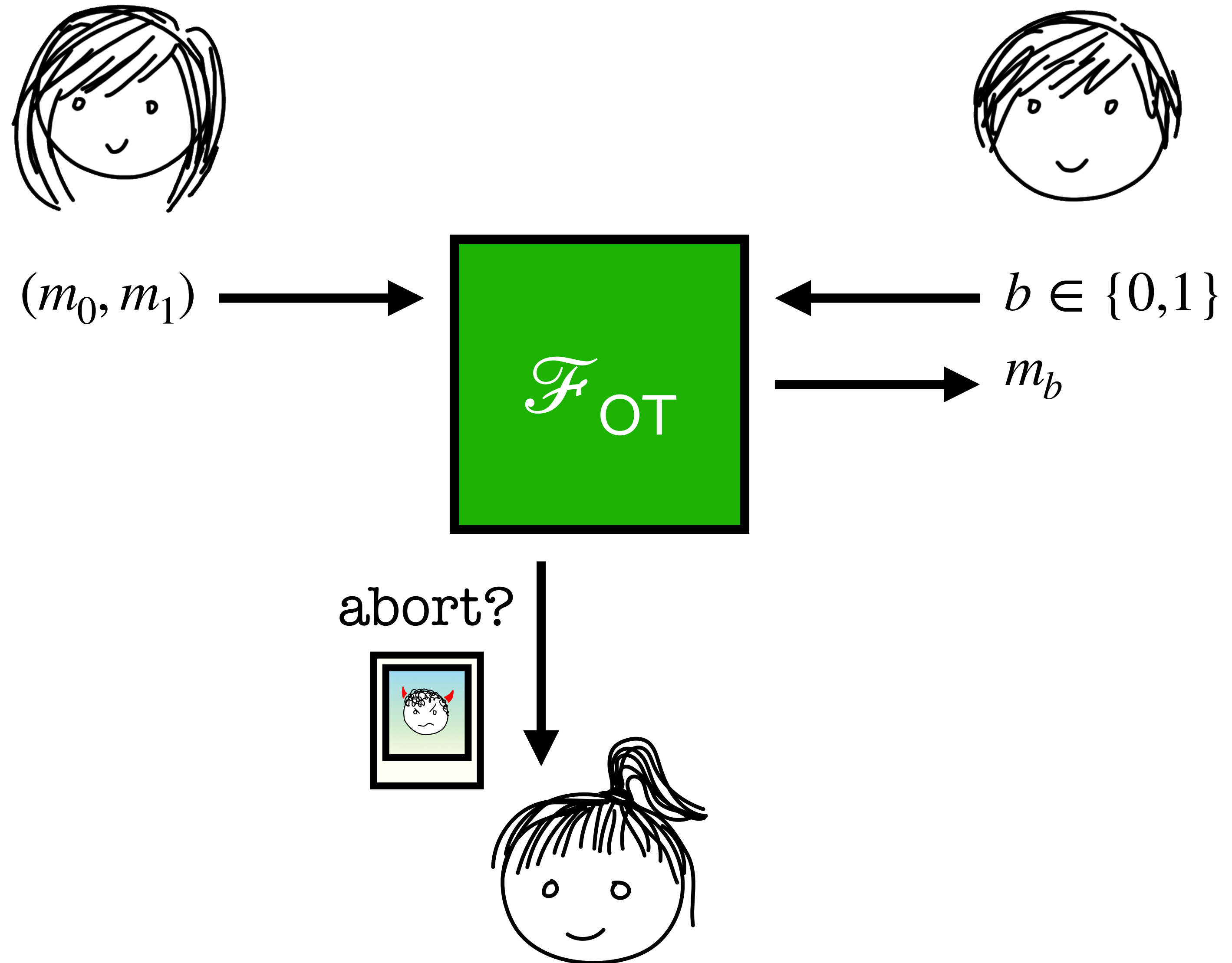
OT: the Fundamental Primitive of MPC



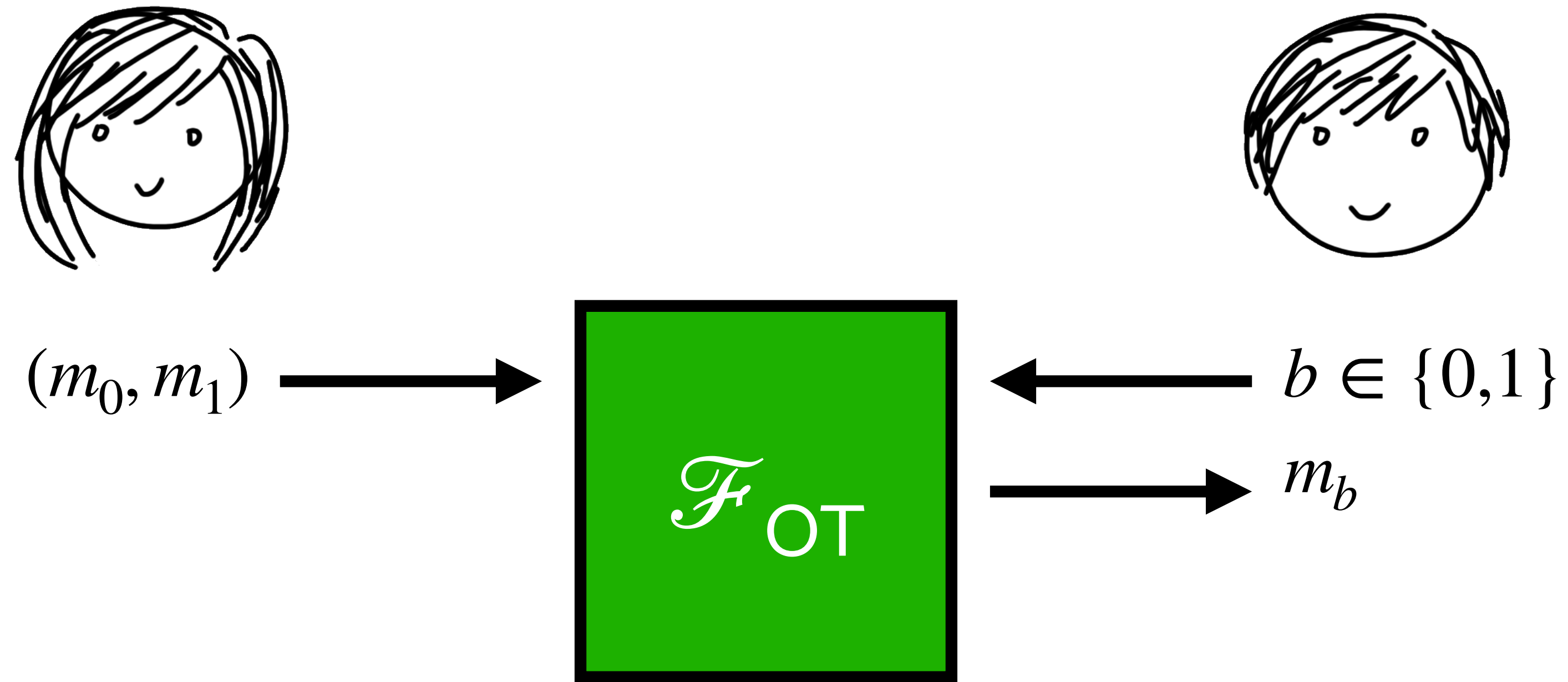
There is a two-party correlation that implies OT information-theoretically
[Beaver 95,96]

IA is separated from all two-party correlations!
[Ishai Ostrovsky Seyalioglu 12]

The simplest multiparty OT analog?

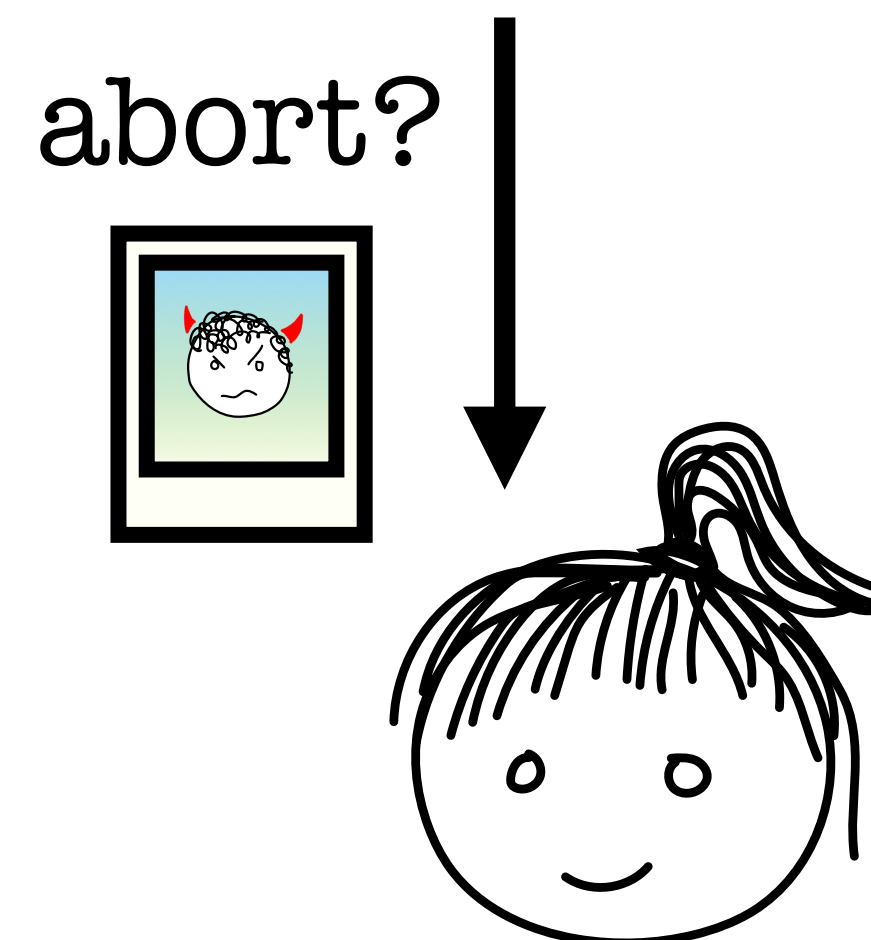


The simplest multiparty OT analog?

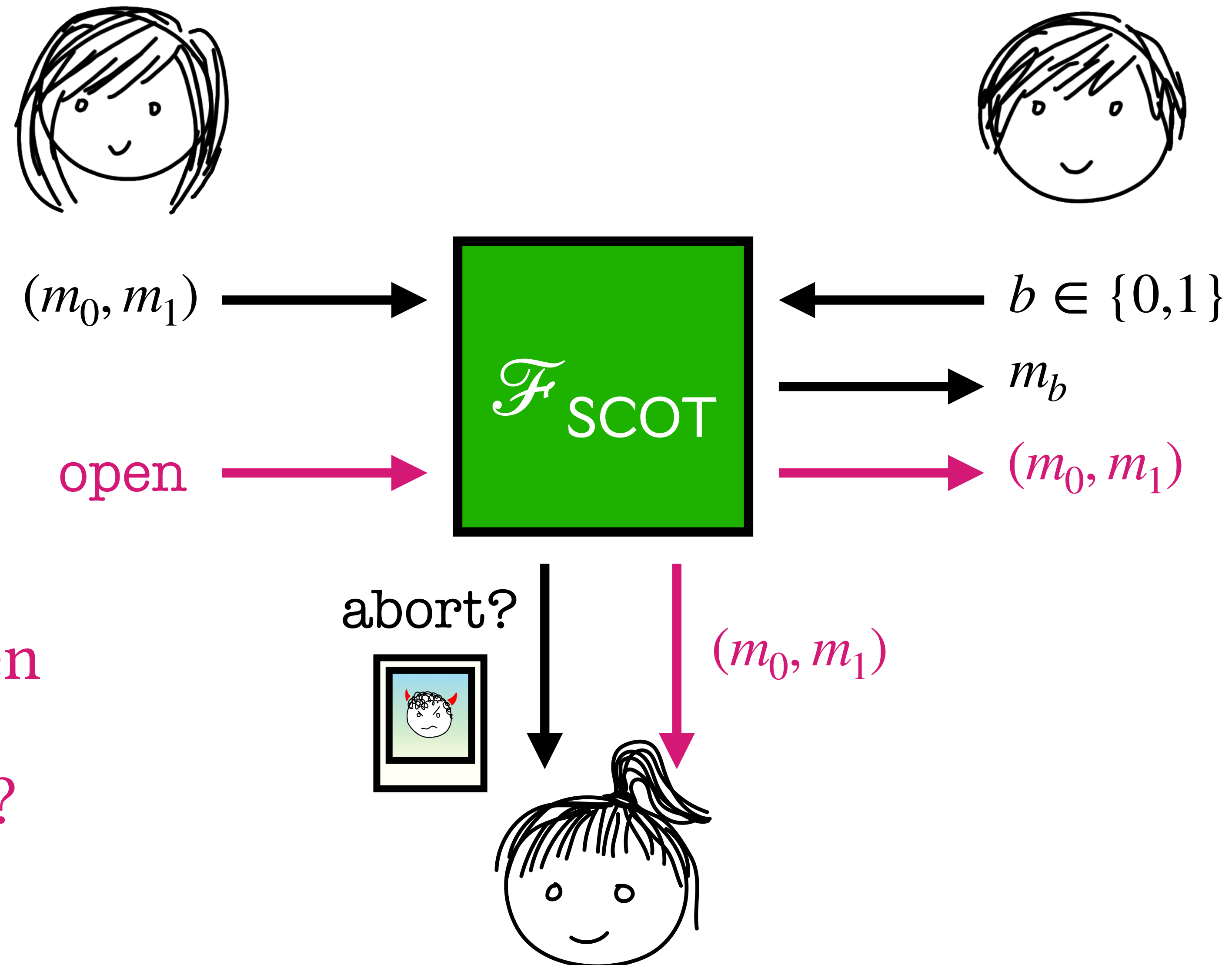


This is not enough...

What happens when
a protocol in the
 \mathcal{F}_{OT} -hybrid aborts?

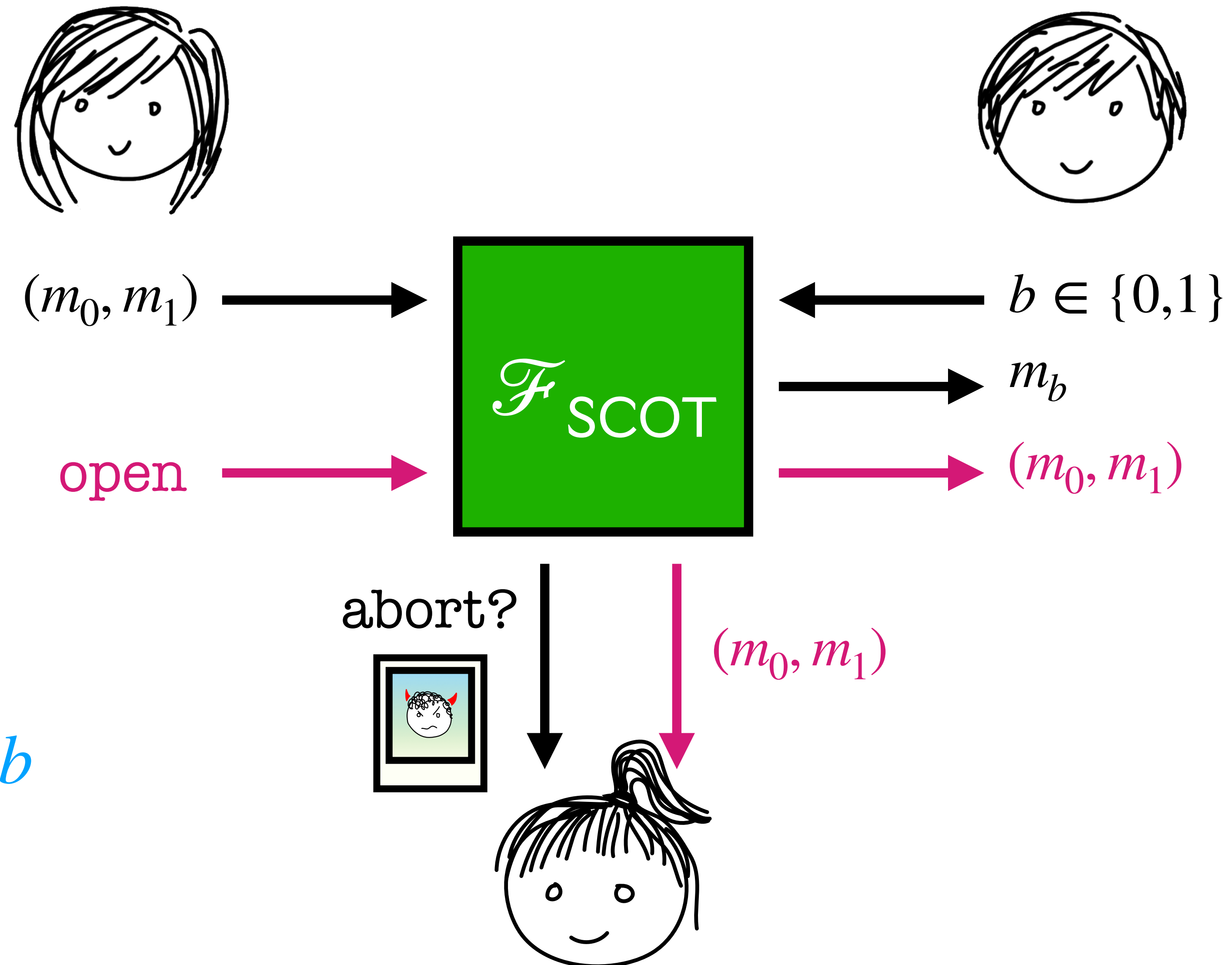


SCOT-IA: *Sender Committed* OT with IA



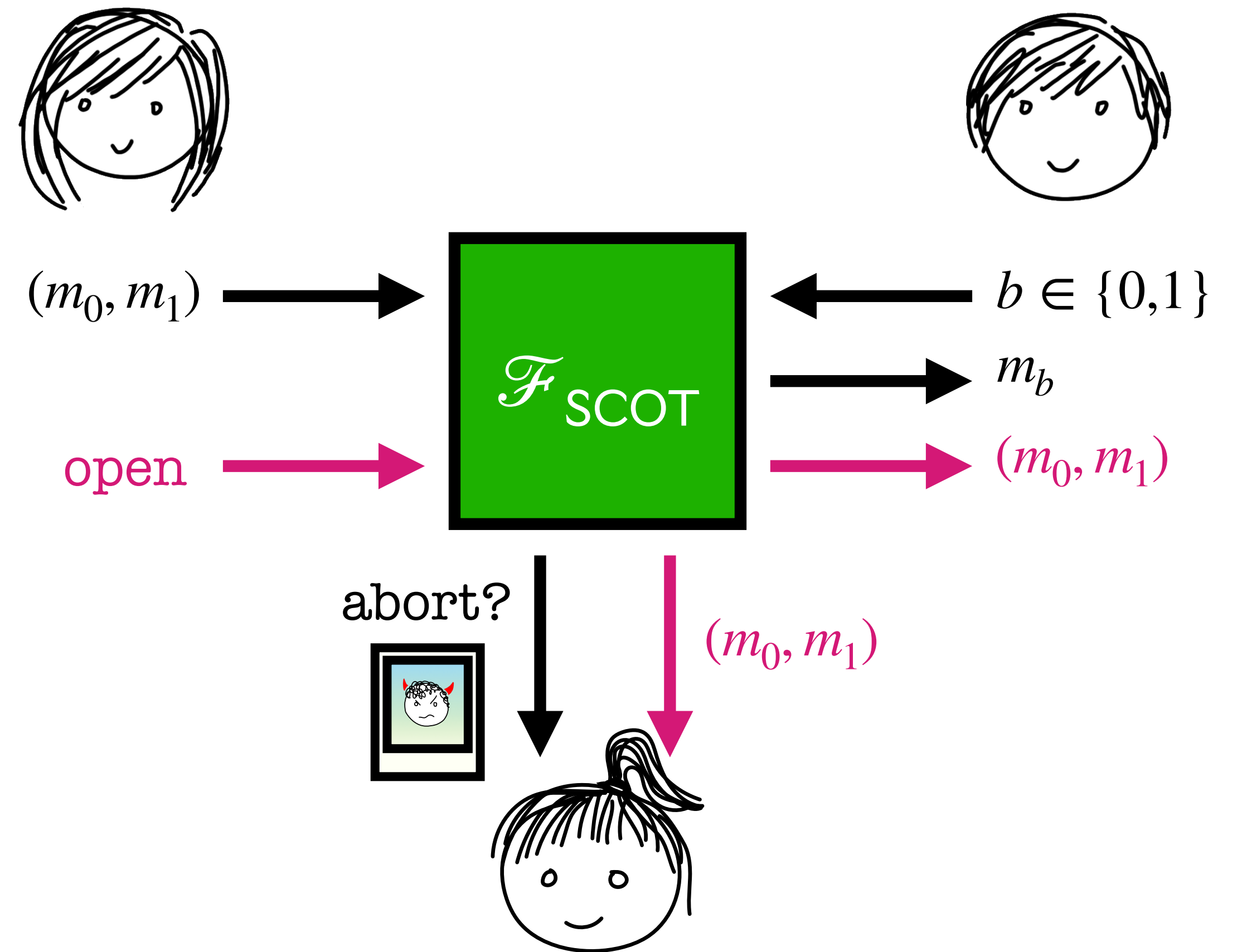
What happens when
a protocol in the
 \mathcal{F}_{OT} -hybrid aborts?

SCOT-IA: *Sender Committed* OT with IA





Notice that m_b is a decommitment for b

SCOT-IA: *Sender Committed* OT with IA



Important Notes:

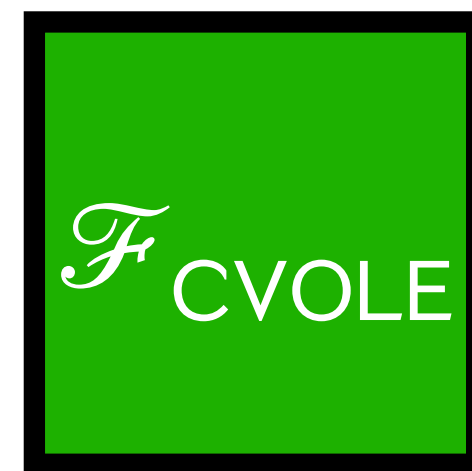
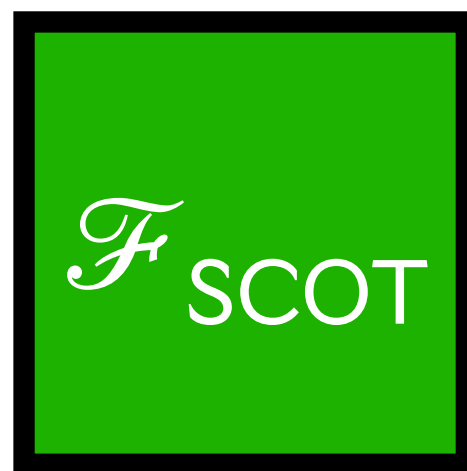
- Functionality is *reactive*. Opening is voluntary
- Functionality is *asymmetric*. Only  opens her inputs
-  only listens passively

Realizing SCOT-IA

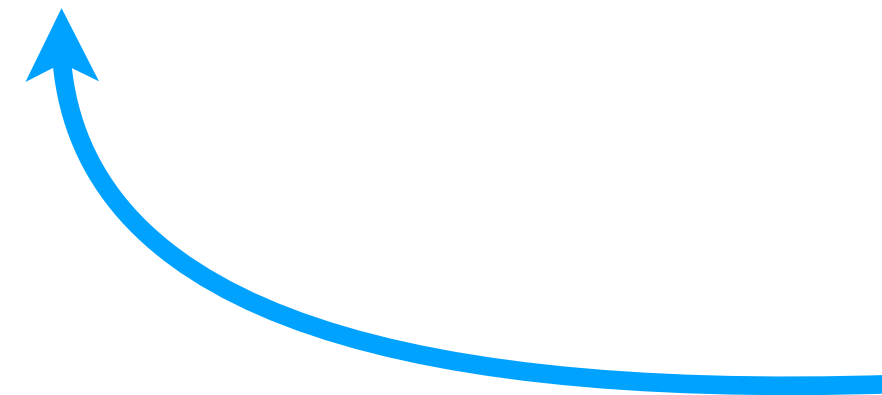
1. IT from simple depth-1 correlation
(OT correlation + MACs for Observers)
2. PVW [Peikert Vaikuntanathan Waters 08] over broadcast
+ Simple sigma protocol to open
 - Instantiable from same assumptions as normal PVW (DDH or LWE or DCR + QR).
 - Composable *without* Fischlin/Pass/Kondi-shelat.
3. Softspoken SCOT-extension
 - Technique: vindicating release in the SCOT-IA-hybrid model.
 - Minimal changes relative to protocol/proof of [Roy 22].
 - Number of public key ops independent of batch size.
 - Requires programmable RO :(

Roadmap of Constructions

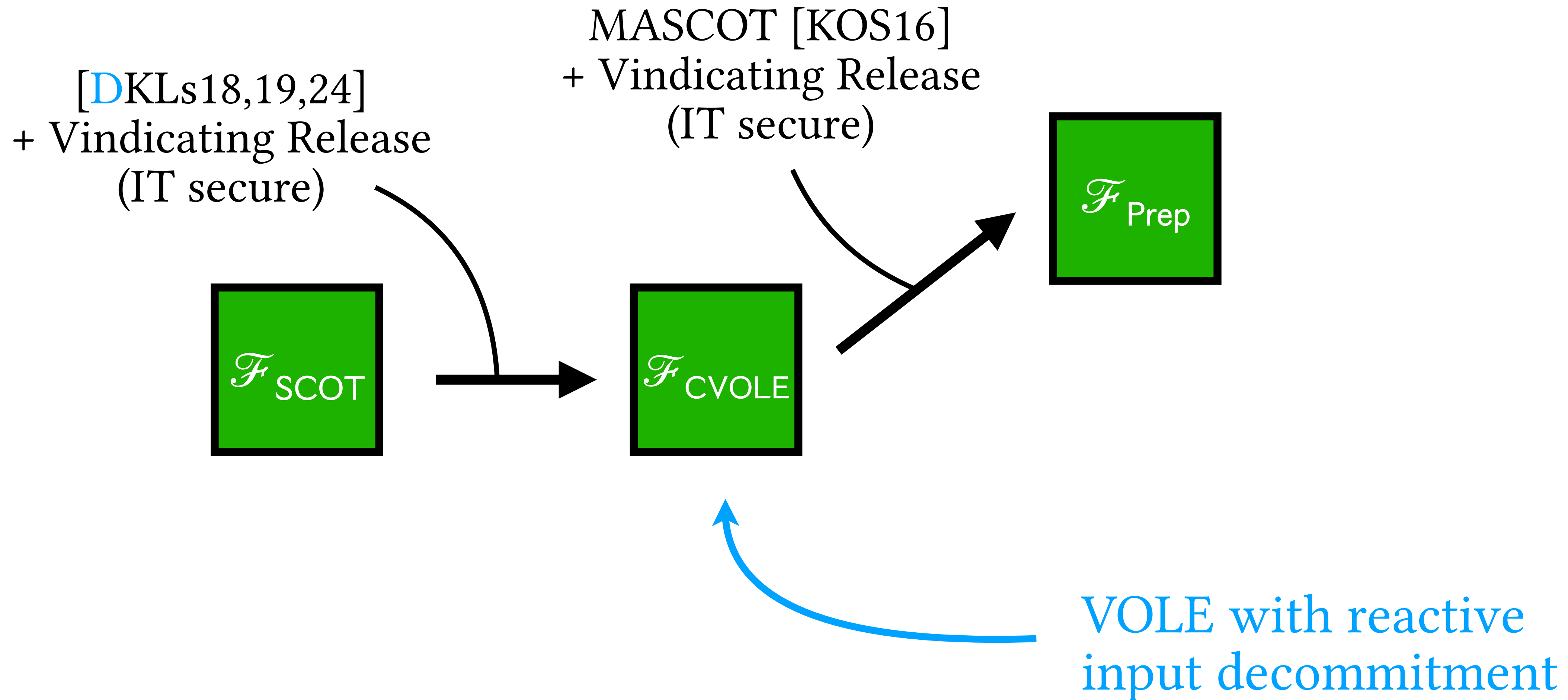
[DKLs18,19,24]
+ Vindicating Release
(IT secure)



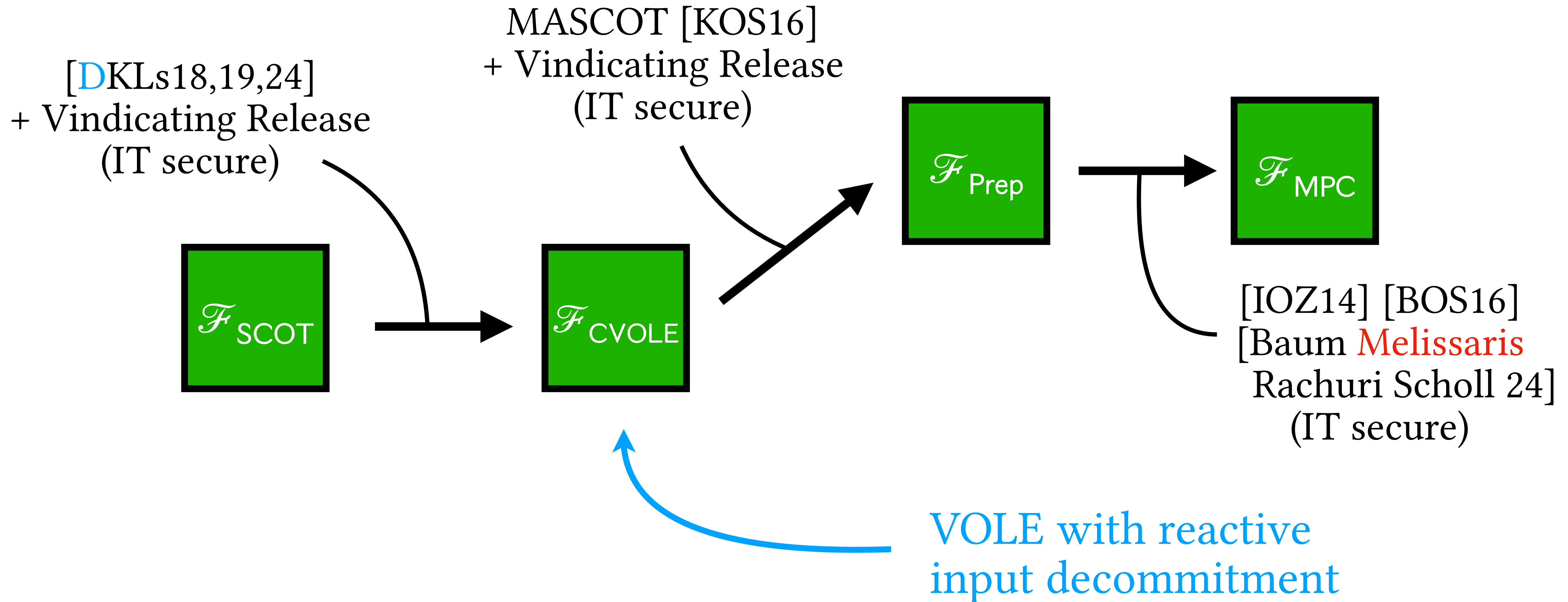
VOLE with reactive
input decommitment



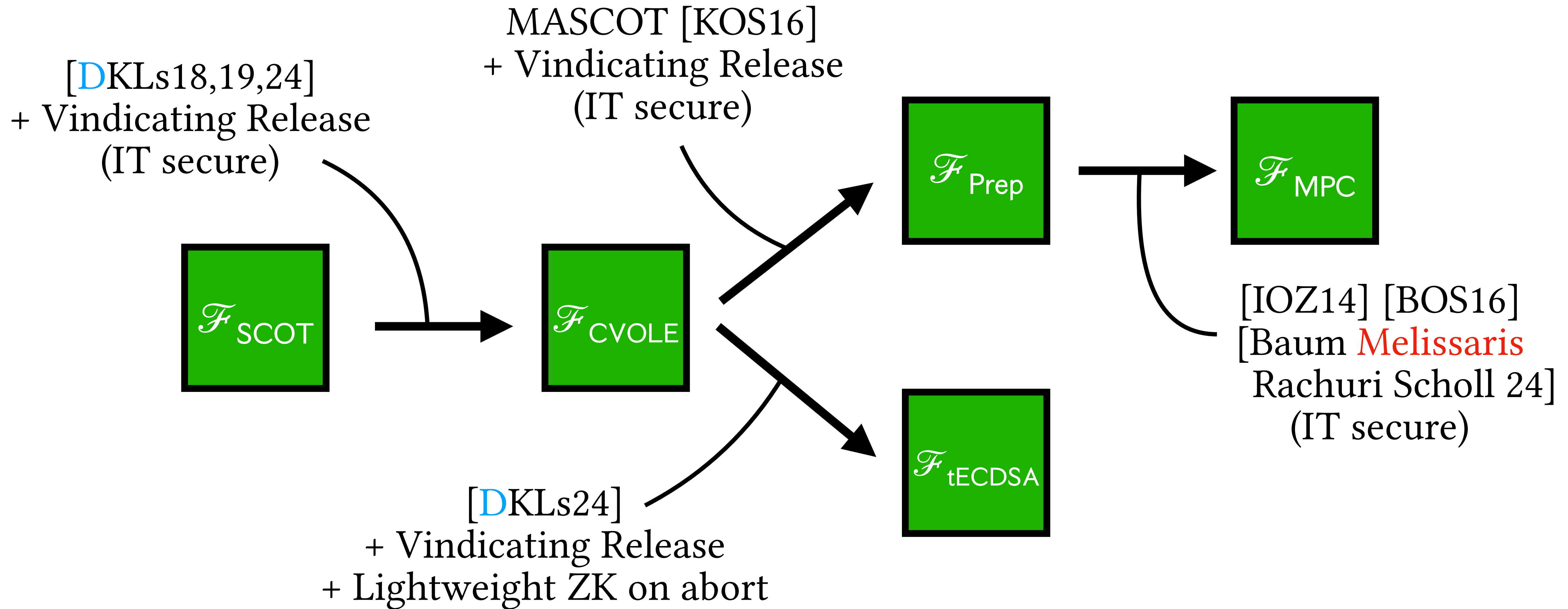
Roadmap of Constructions



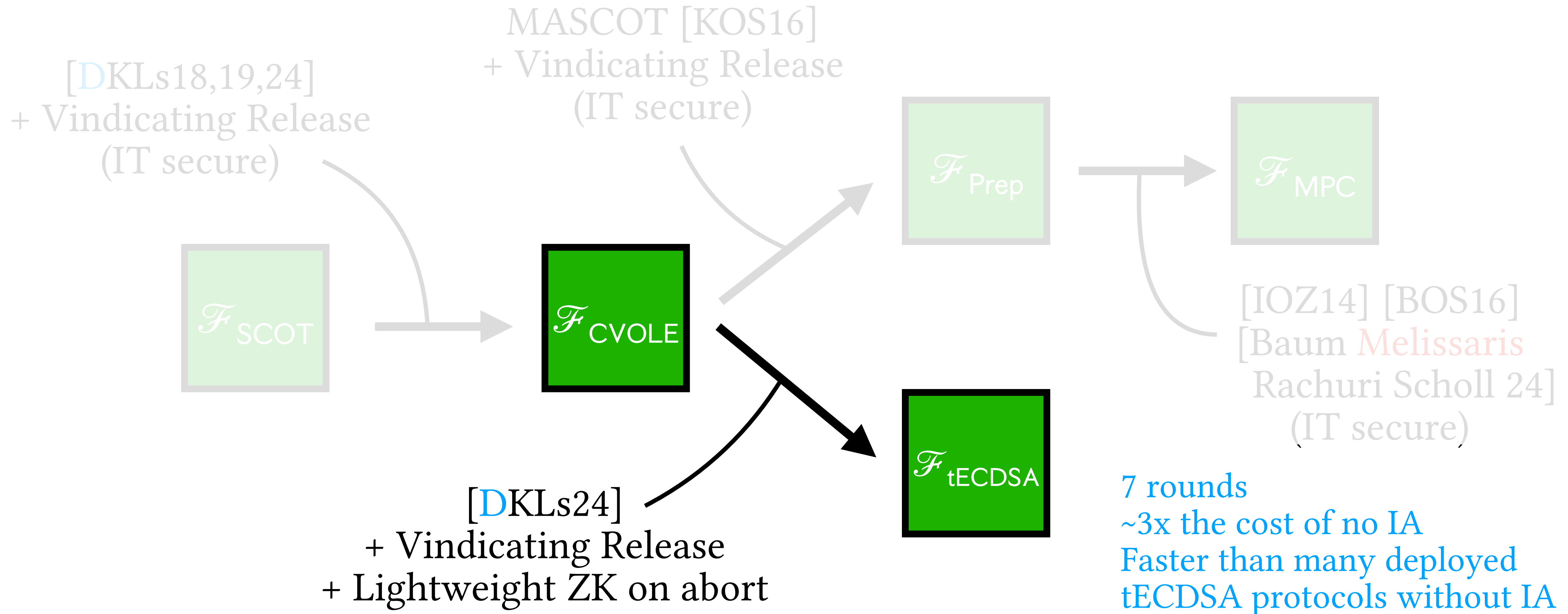
Roadmap of Constructions



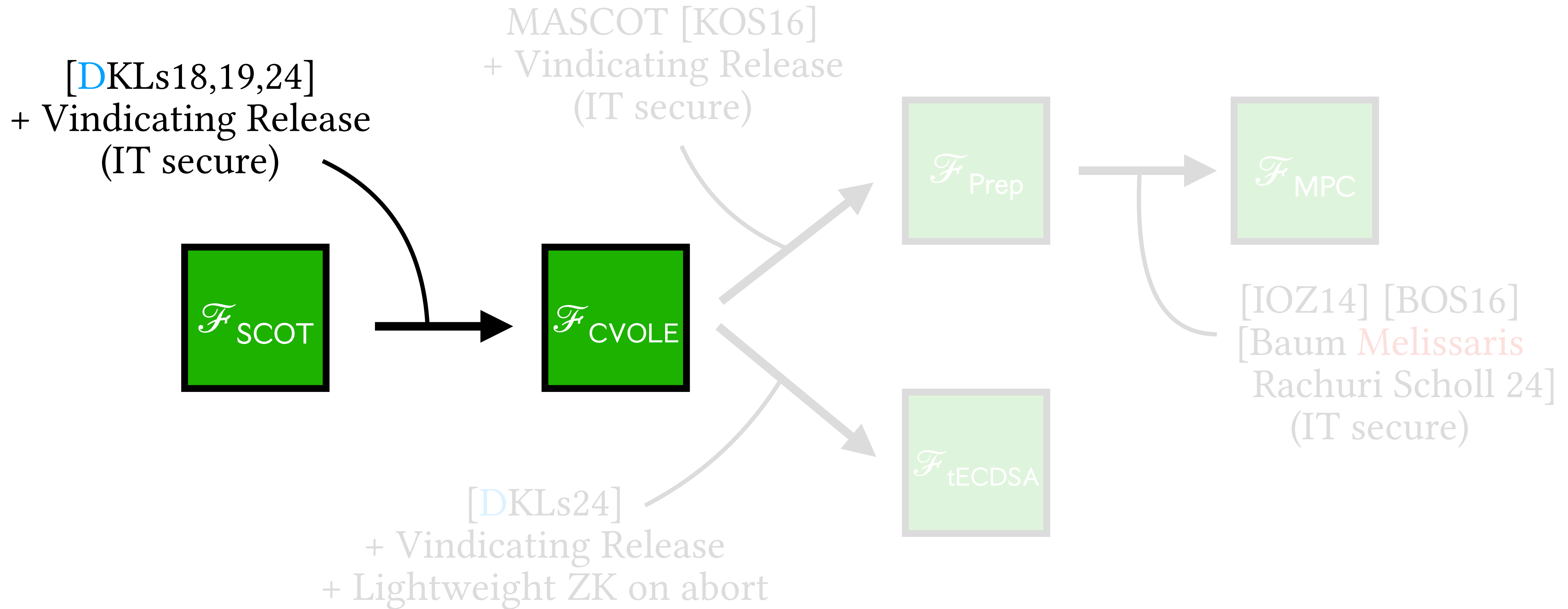
Roadmap of Constructions

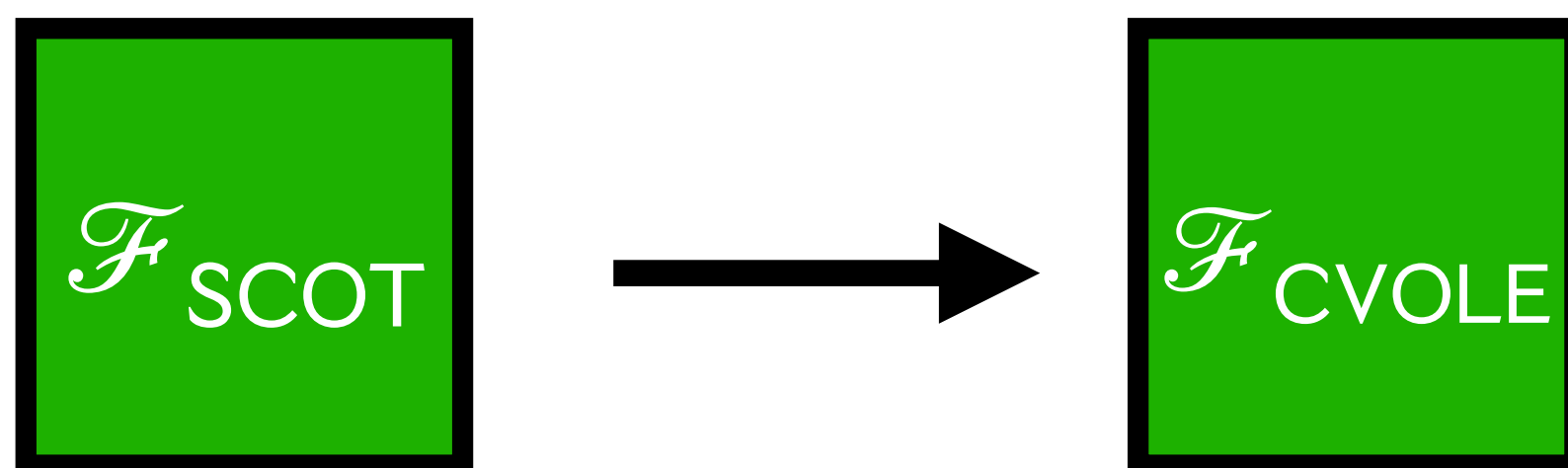


Evidence of Practicality



CVOLE: Distinguishing VR from Adaptive Sec







To open the protocol state of , the simulator must solve an instance of subset sum.

If  is honest, we always sample an easy instance.

If  cheats, the adversary can influence the instance.

The protocol is probably not adaptively secure, but because vindicating release is an active process, we can check for  cheats before opening the state of  which guarantees that simulation is efficient.

Our Goal

1. Propose fundamental primitive
2. Construct generic MPC
 - IT only in hybrid model of fundamental primitive
 - Add IA to well-known constructions using Vindicating Release
 - Reusable modules (e.g. VOLE)
3. Don't use the words
 - “Non-black-box”
 - “Adaptive”
 - “Homomorphic”
 - “Compiler”
 - “Straight-line Extraction”

Secure Multiparty Computation with Identifiable Abort via Vindicating Release

Ran
Cohen

Jack
Doerner

Yashvanth
Kondi

abhi
shelat

<https://ia.cr/2023/1136>

Cheater Identification on a Budget: MPC with Identifiable Abort from Pairwise MACs

Carsten
Baum

Nikolas
Melissaris

Rahul
Rachuri

Peter
Scholl

<https://ia.cr/2023/1548>