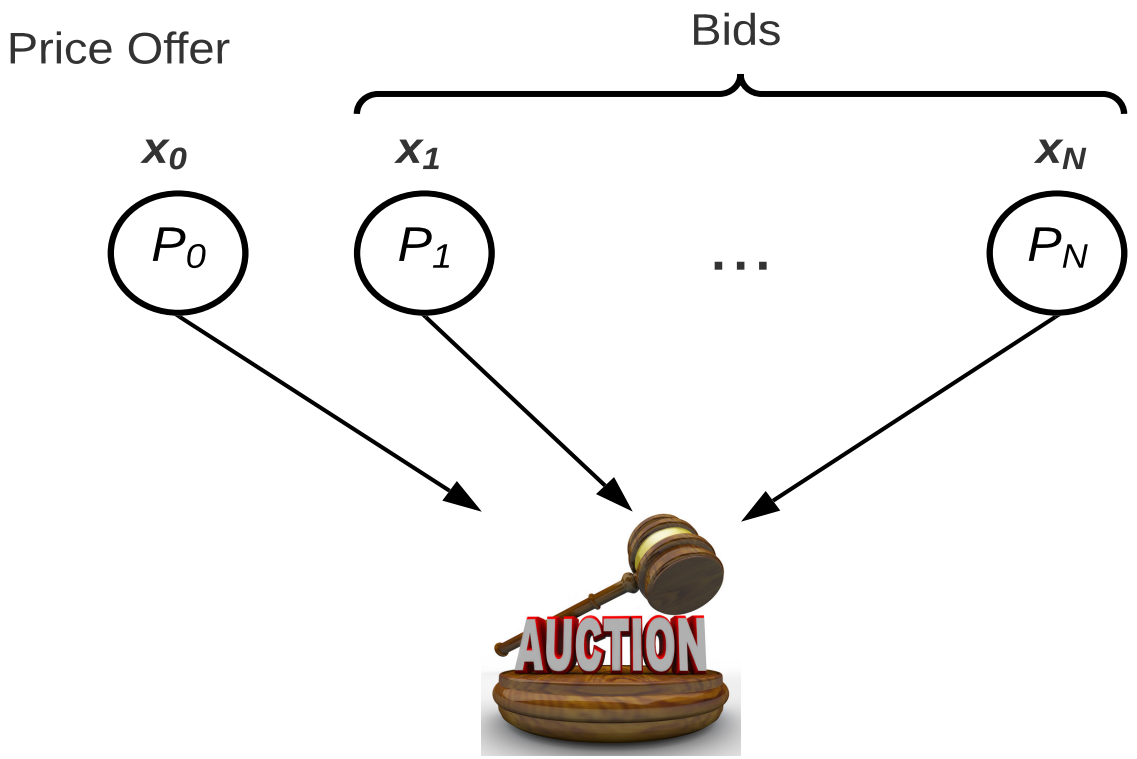
Gage MPC: Bypassing Residual Function Leakage for Non-Interactive MPC

Ghada Almashaqbeh[1] Fabrice Benhamouda[2] Seungwook Han[3]
Daniel Jaroslawicz[3] Tal Malkin[3] Alex Nicita[3] Tal Rabin[4,2]
Abhishek Shah[3] Eran Tromer[3,5]

[1]UConn, [2]Algorand, [3]Columbia, [4]UPenn, [5]Tel-Aviv University

PETS 2021

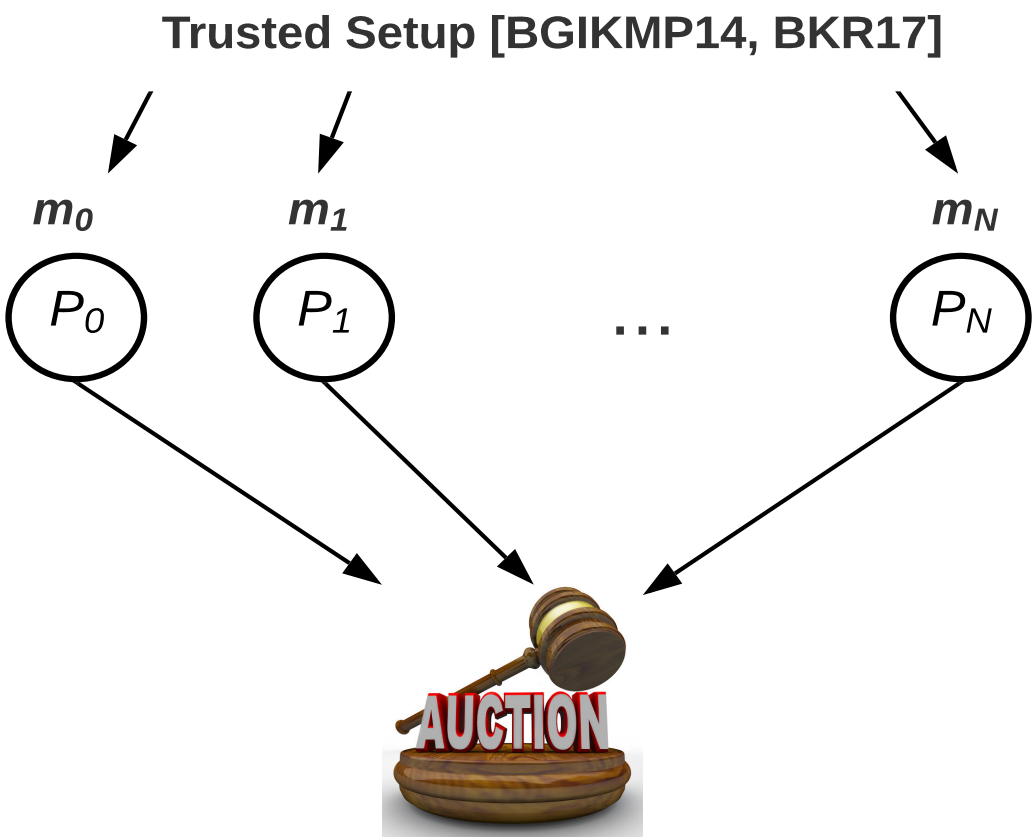
NIMPC — Auction



$f(x_0, x_1, ..., x_N)$

The winner is ...

NIMPC — Auction

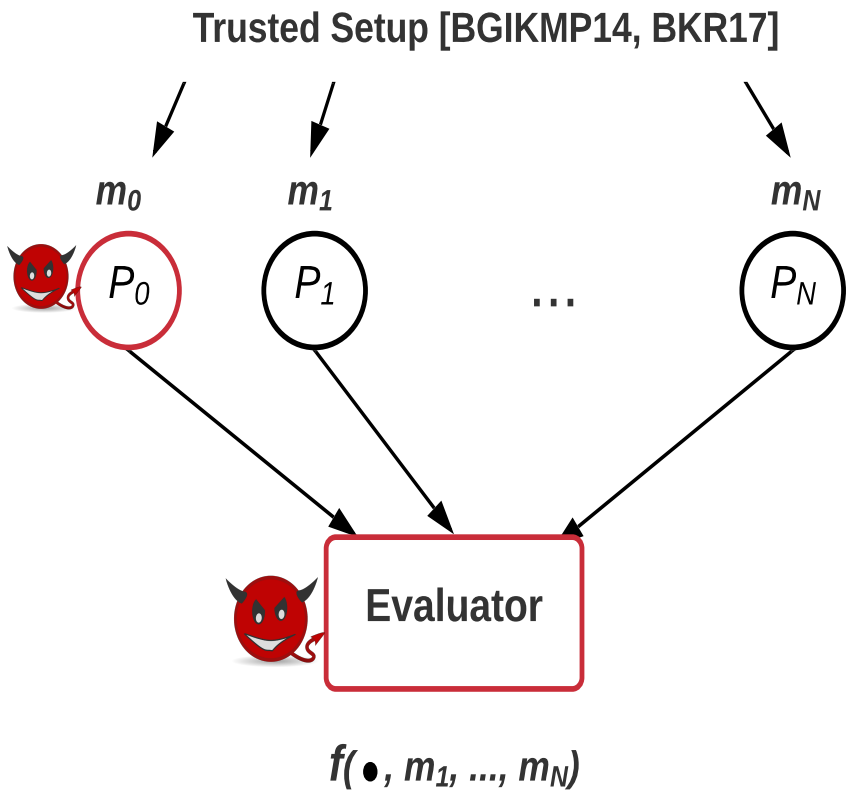


$f(m_0, m_1, ..., m_N)$

The winner is ...

NIMPC

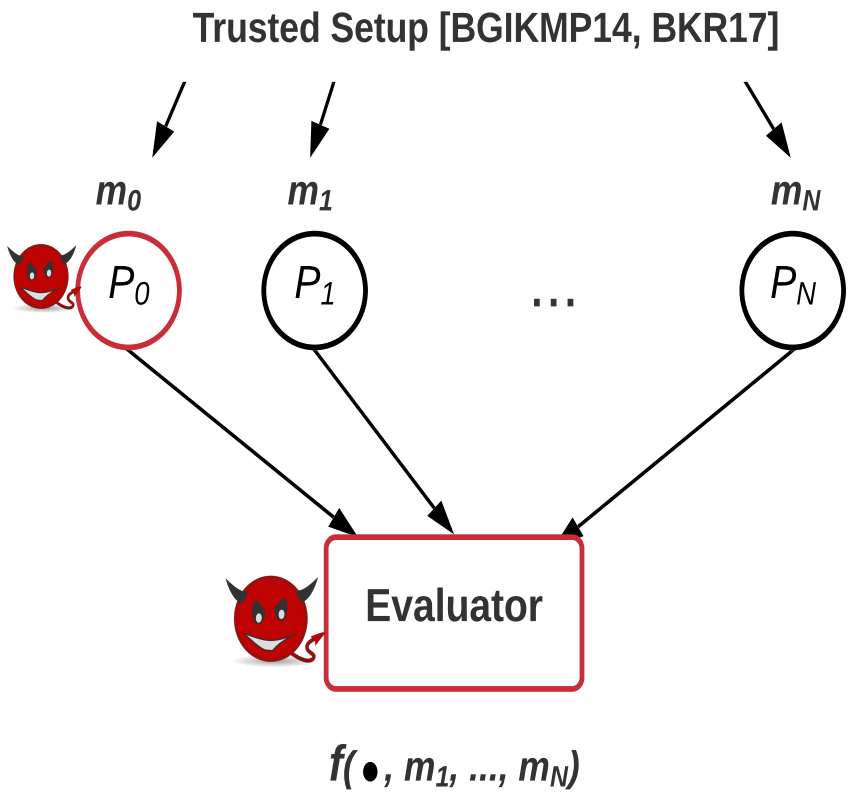
- Leakage of the *Residual Function* is inherent.
  Evaluator and say P_0 can compute $f(\bullet, m_1, \ldots, m_N)$.



NIMPC

- Leakage of the *Residual Function* is inherent.
  Evaluator and say P_0 can compute $f(\bullet, m_1, \ldots, m_N)$.

- Robustness to collusion.
  Only the residual function is leaked!

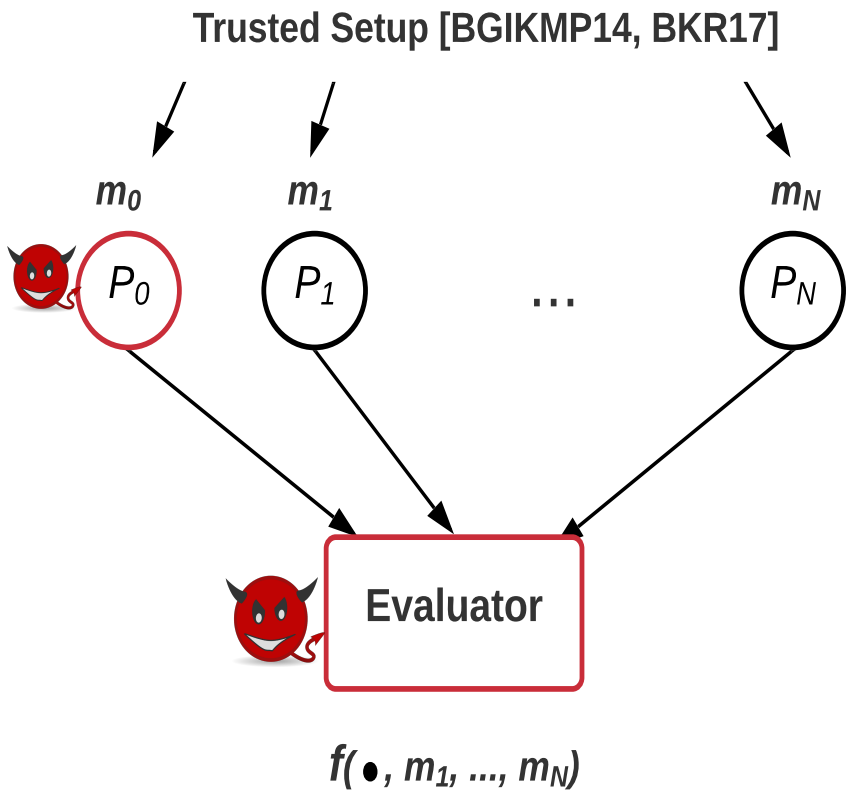


NIMPC

- Leakage of the *Residual Function* is inherent.
  Evaluator and say P_0 can compute $f(\bullet, m_1, \ldots, m_N)$.

- Robustness to collusion.
  Only the residual function is leaked!

- A trusted setup requirement.



MPC and Blockchain

- **Gen I**. A blockchain implements a broadcast channel.

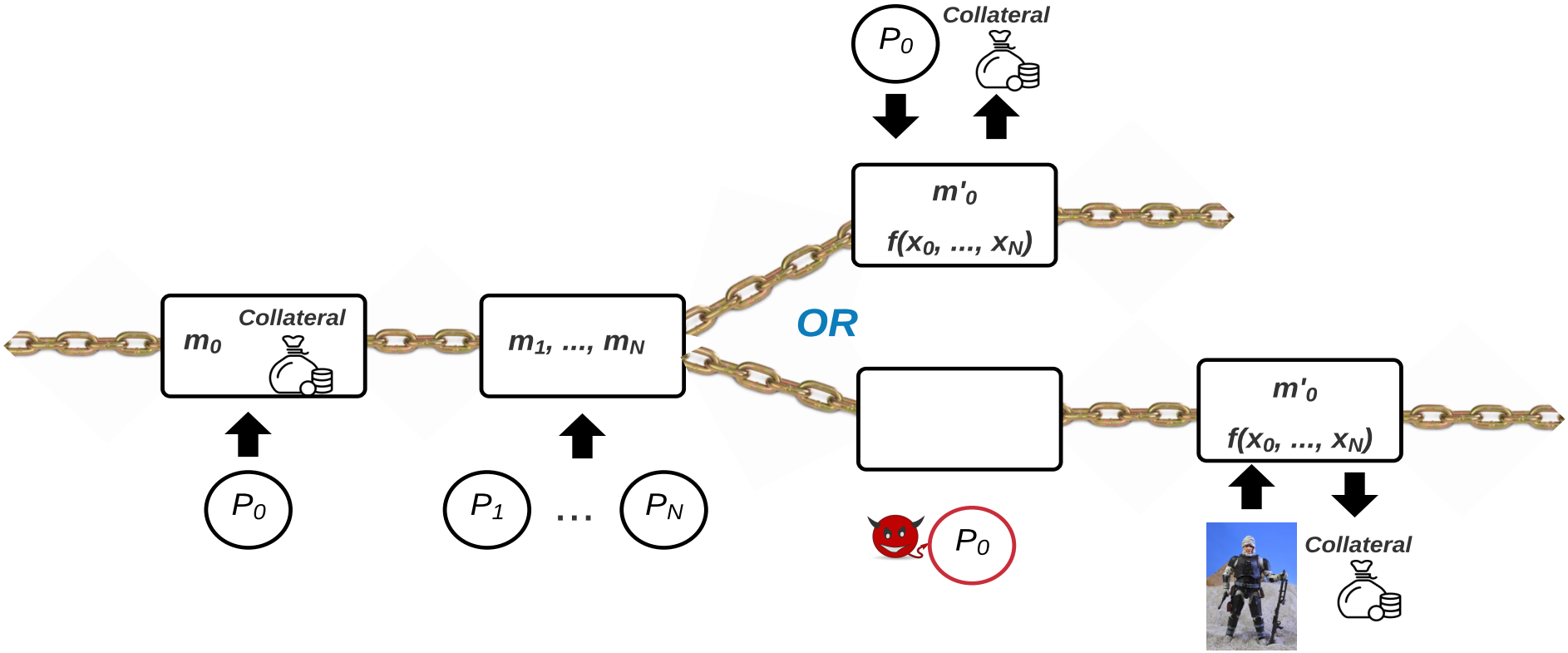
MPC and Blockchain

- **Gen I**. A blockchain implements a broadcast channel.
- **Gen II**. Payments are incorporated into MPC.

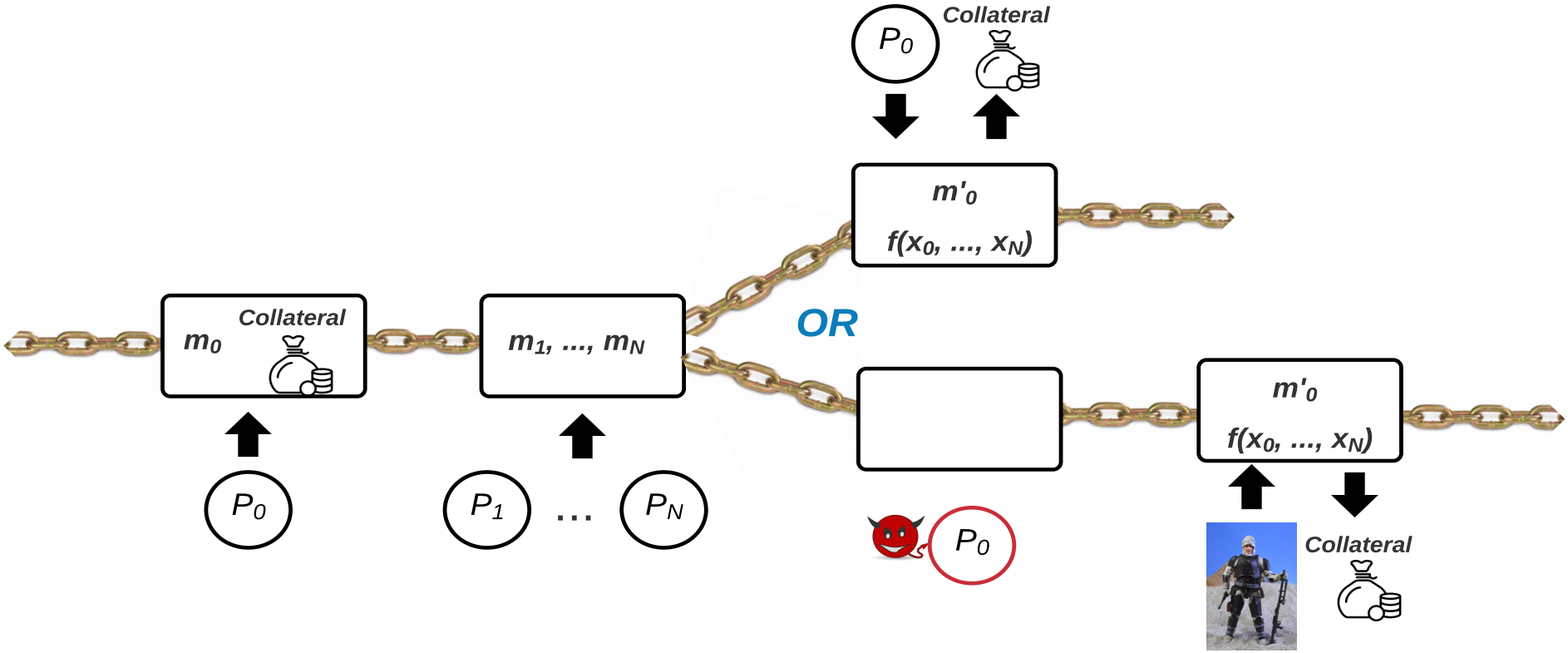
MPC and Blockchain

- **Gen I**. A blockchain implements a broadcast channel.
- **Gen II**. Payments are incorporated into MPC.
- **Gen III**. *This work; Gage MPC!* Smart contracts and miners are active participants in MPC.

Gage MPC


P_0
Collateral
m'_0
$f(x_0, ..., x_N)$
OR
m_0
Collateral
$m_1, ..., m_N$
P_0
P_1
...
P_N
P_0
m'_0
$f(x_0, ..., x_N)$
Collateral

Gage MPC



A monetary assumption. *An honest party can put a collateral of value much higher than what an adversary can expend on computation.*

On Circumventing the Lower Bounds

- Eliminate the leakage of the residual function.

On Circumventing the Lower Bounds

- Eliminate the leakage of the residual function.
- Eliminate setup assumptions (aka pre-shared correlated randomness).

On Circumventing the Lower Bounds

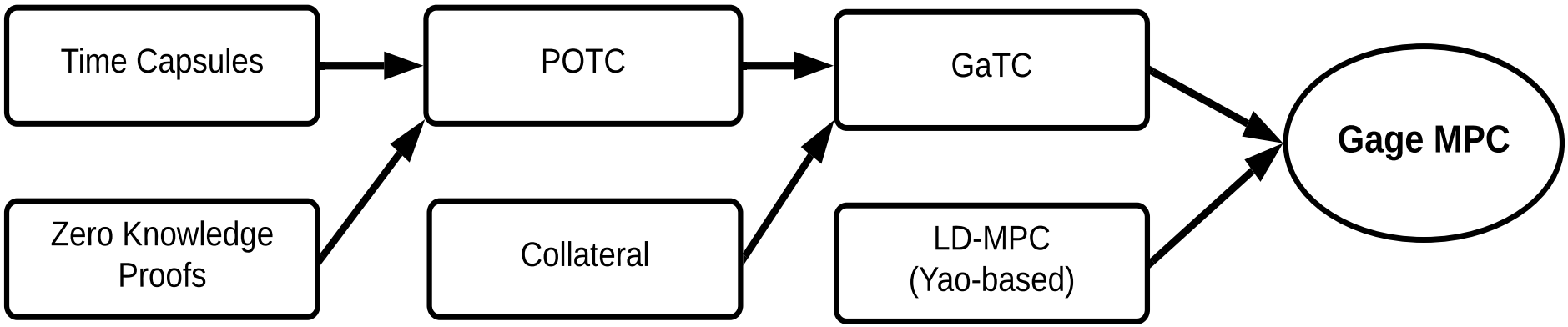
- Eliminate the leakage of the residual function.
- Eliminate setup assumptions (aka pre-shared correlated randomness).
- Eliminate the need for a dedicated online evaluator.

On Circumventing the Lower Bounds

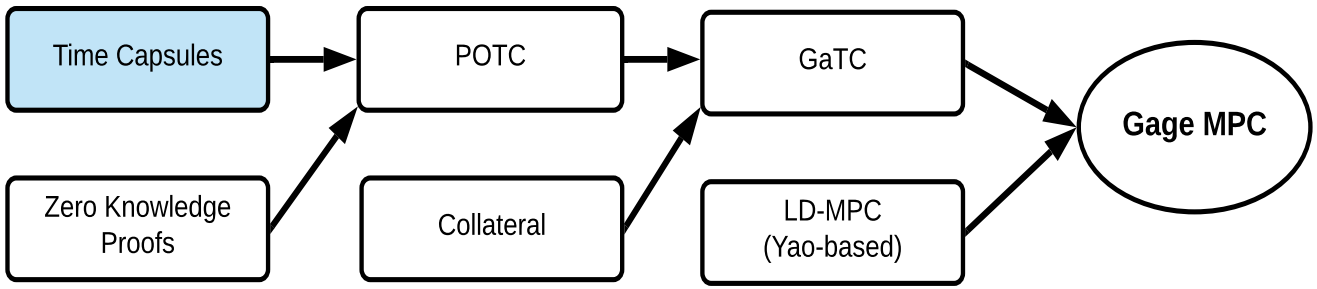
- Eliminate the leakage of the residual function.
- Eliminate setup assumptions (aka pre-shared correlated randomness).
- Eliminate the need for a dedicated online evaluator.

Gage MPC guarantees *short term* security!

Gage MPC: Our Construction



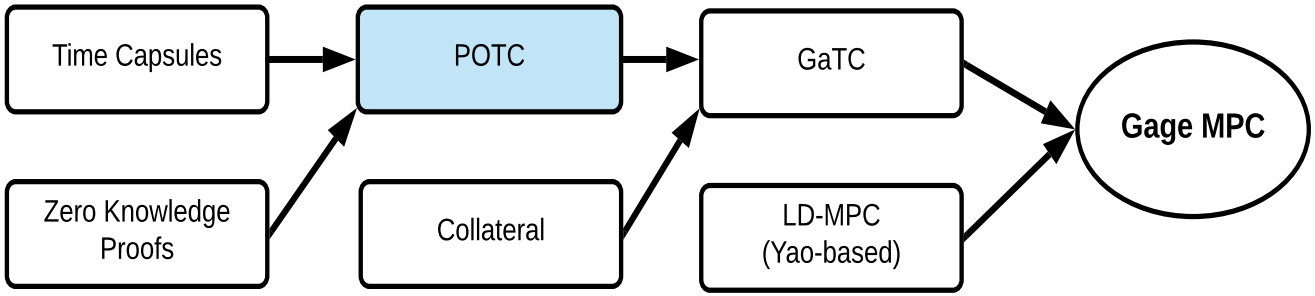
Our Construction — Time Capsules



Simply commit to a value that can be opened after expending a pre-specified amount of computation.

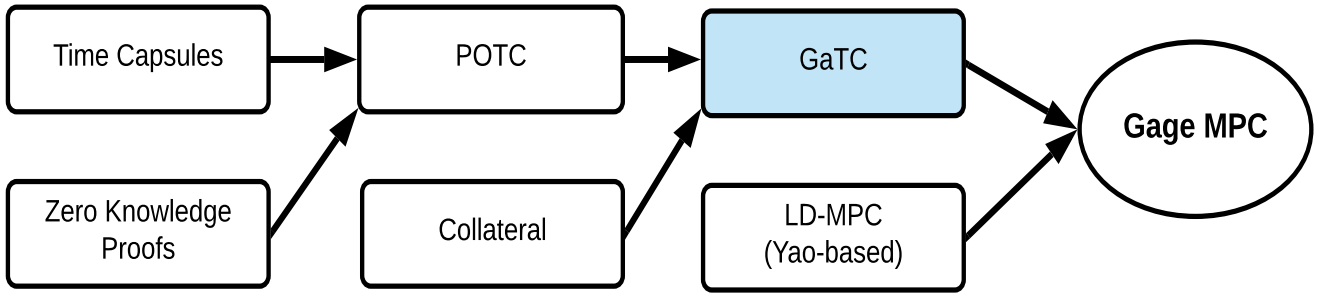
E.g., $h(s)$ where $s \leftarrow \{0,1\}^{\lambda^*}$

Our Construction — Proof of Time Capsules (POTC)

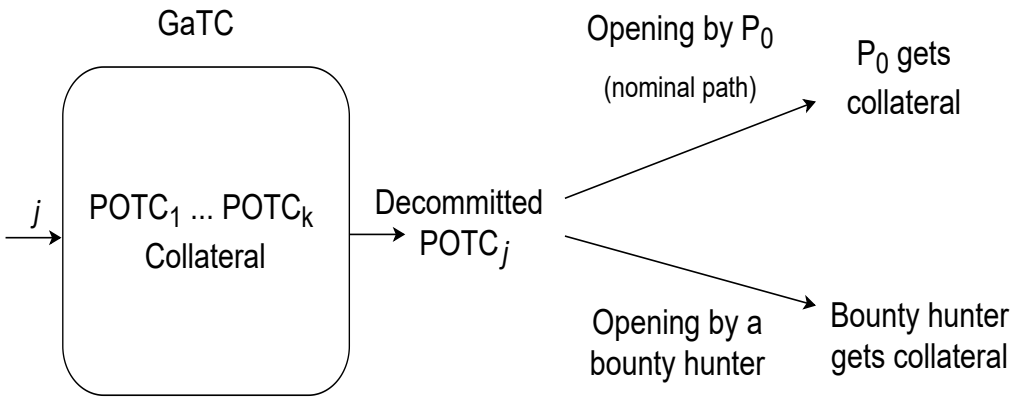


Instead of announcing the decommitment itself (i.e., s), prove in zero knowledge that the decommitment has been found.

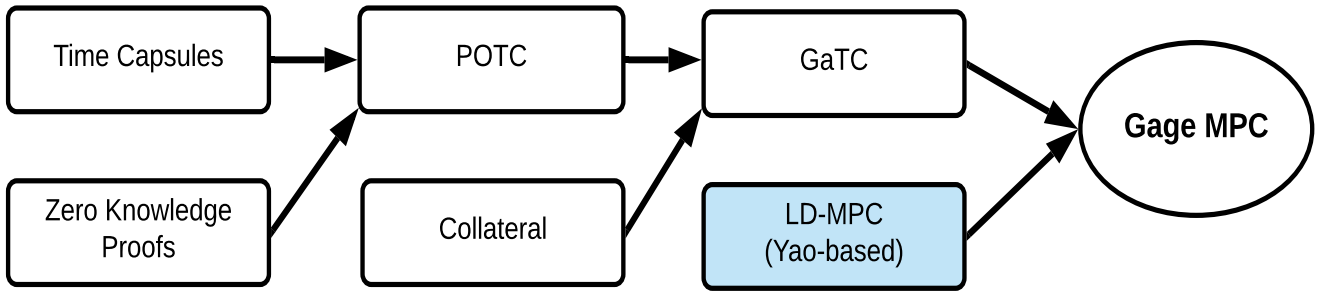
Our Construction — Gage Time Capsules (GaTC)



Bundle several POTCs together, and utilize a smart contract to provide a monetary incentive to open the intended POTC.



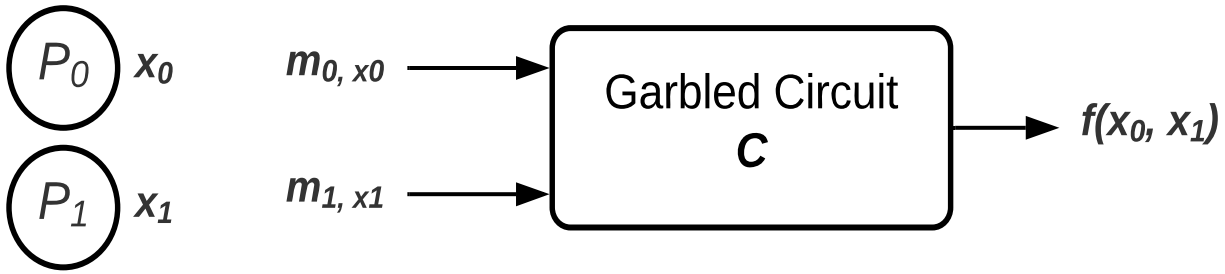
Our Construction — Label Driven MPC (LD-MPC)



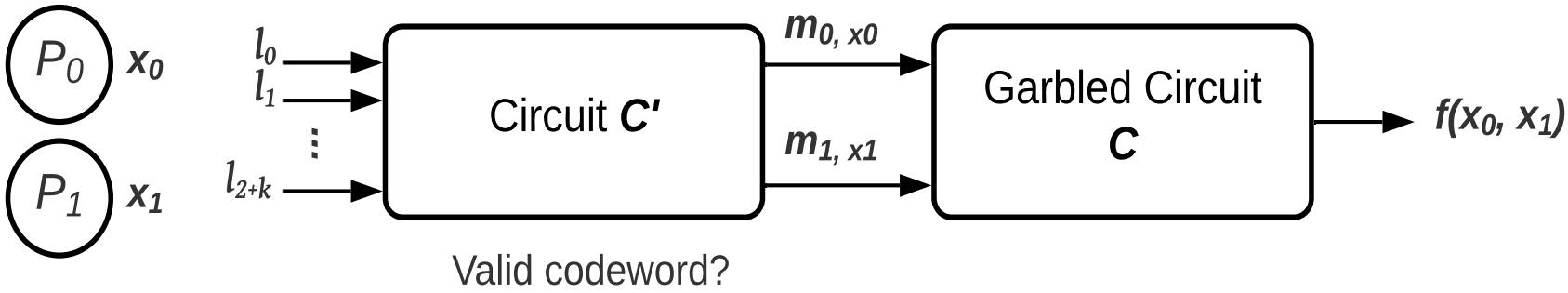
A generalization of Garbled Circuits that is robust to the exposure of additional labels.

Our Construction — Label Driven MPC (LD-MPC)

Conventional Yao; Exposure of any additional label compromise input privacy.



LD-MPC = Error Correcting Codes + Yao

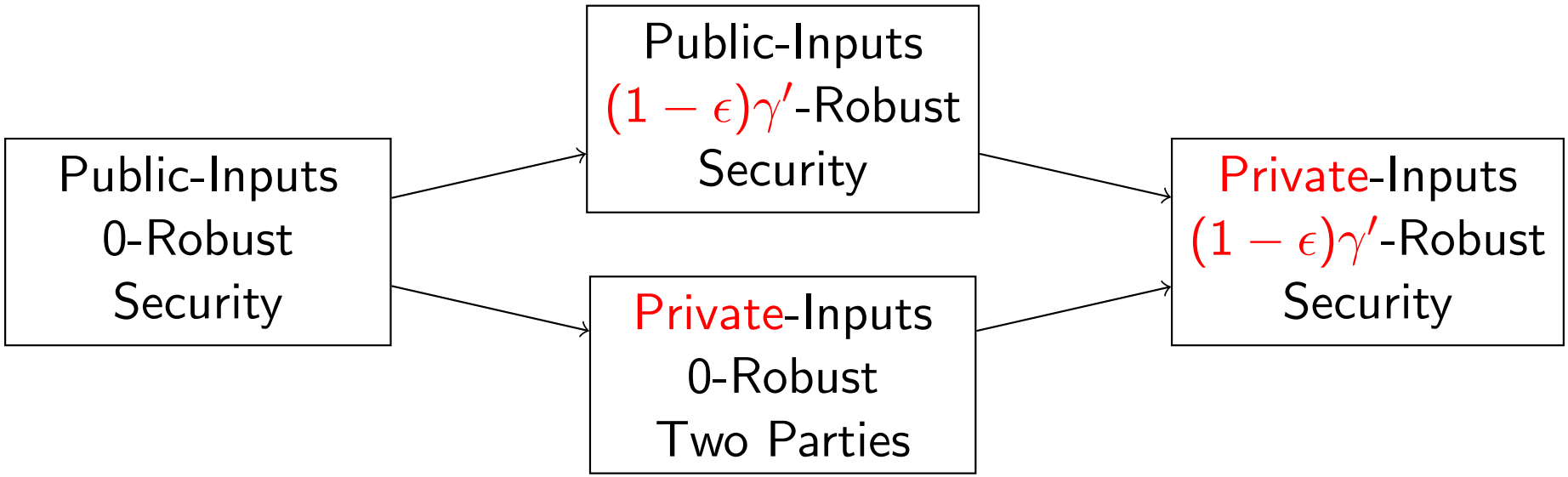


Main Result — Gage MPC

Combines LD-MPC with GaTC.
Simplest case; Only the input of P_0 is private.

- P_0 prepares a garbled circuit, GaTCs for input labels for $P_1, \ldots, P_N$, and a controller smart contract.
- $P_1, \ldots, P_N$ submit their inputs.
- Either P_0 will come back and open the corresponding labels, or bounty hunters will do.
- Smart contract evaluates the circuit over the labels and record the output.

Gage MPC — Transformations



The fully private input versions are for two party computation.

Conclusion

Main Result — Gage MPC

NIMPC for f leaking R and requiring $TS \rightarrow$ NIMPC with no R and TS

Gen III of MPC + blockchain

Side Result

Several new primitives (POTC, GaTC, and LD-MPC) that could be of independent interest.

A proof-of-concept implementation in Ethereum-like blockchain.

Thank you!

Questions?