Two Round Information-Theoretic MPC with Malicious Security

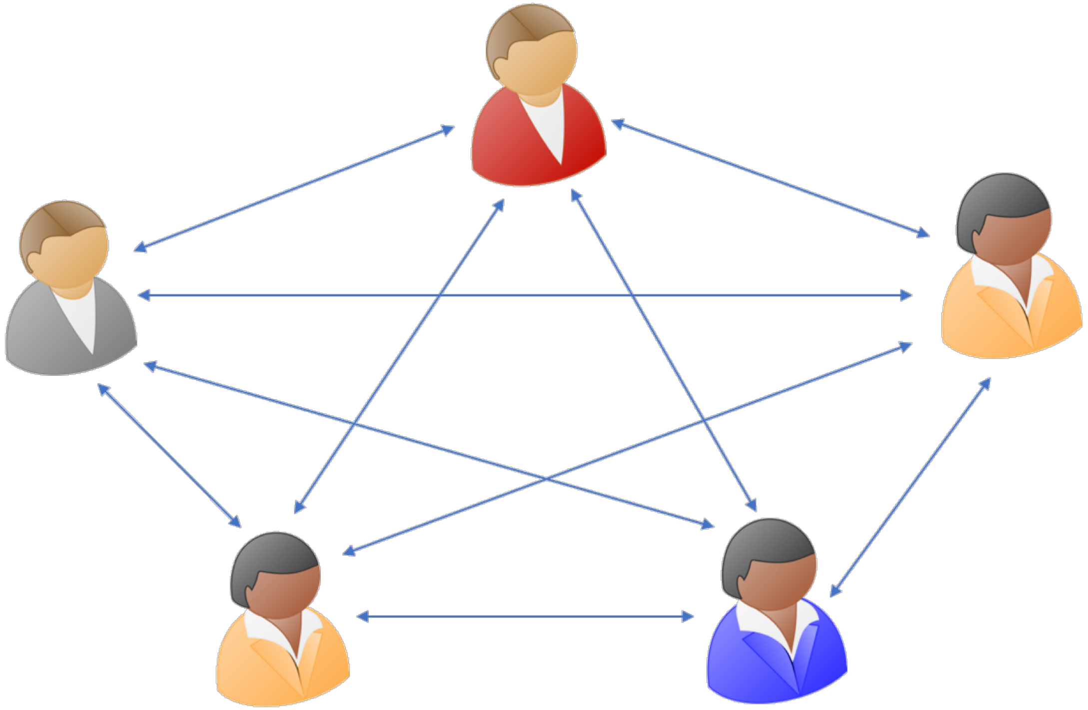
Prabhanjan Ananth   Arka Rai Choudhuri   Aarushi Goel   Abhishek Jain

Massachusetts Institute of Technology

JOHNS HOPKINS UNIVERSITY

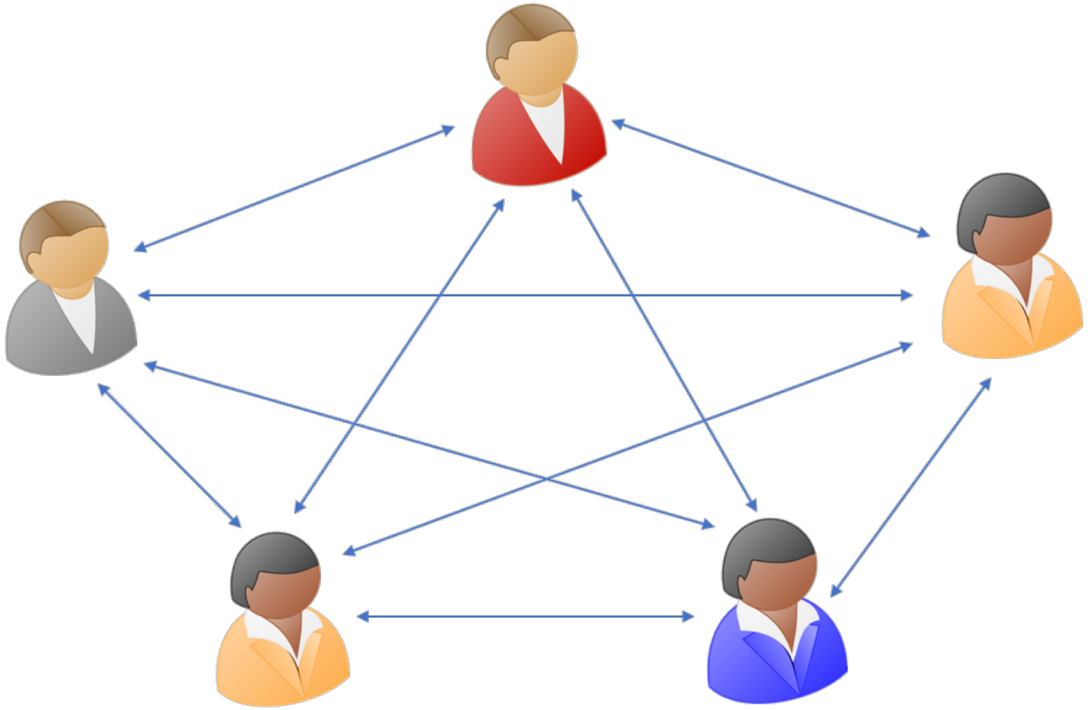
TPMPC 2019

Adversarial Model



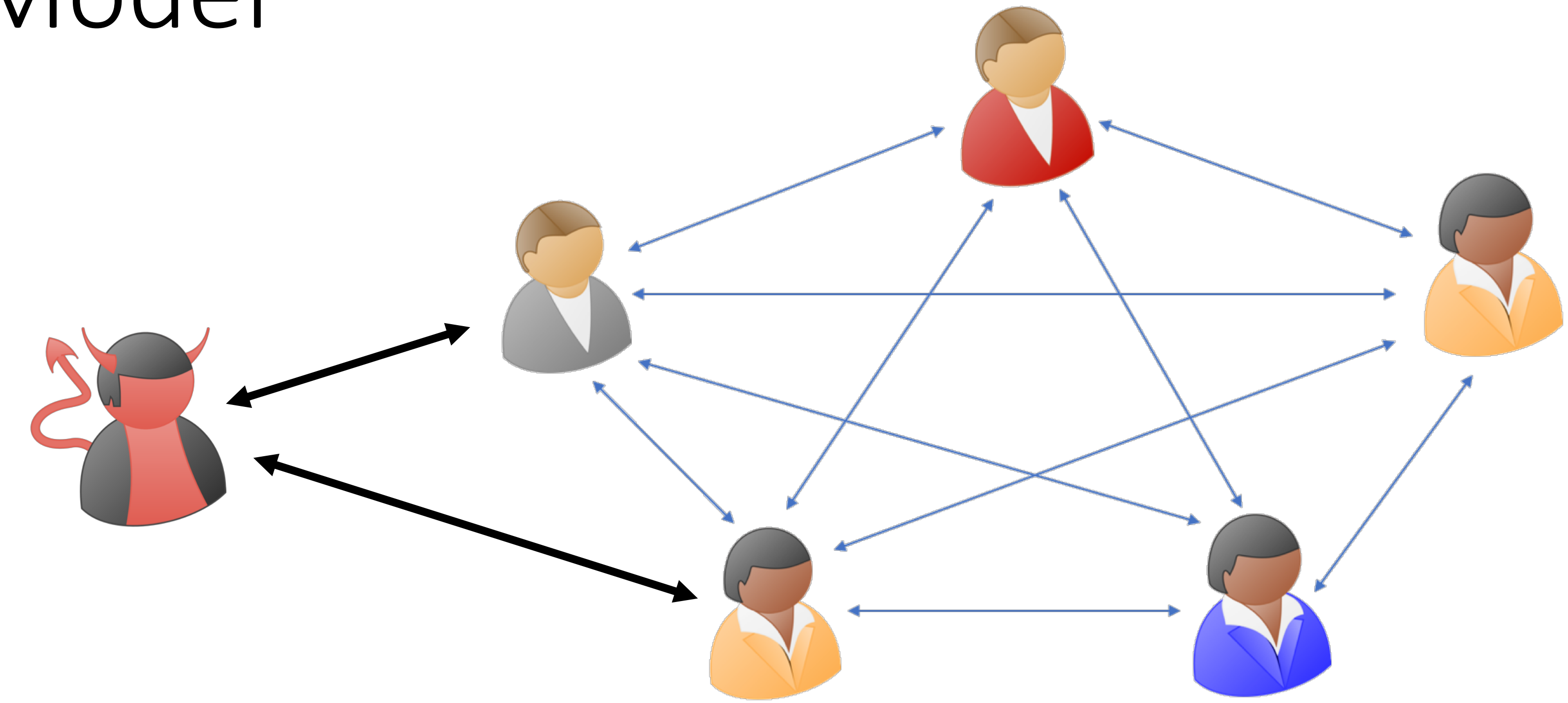
Adversarial Model

Malicious Adversary



Adversarial Model

Malicious Adversary



Corrupts $< n/2$ parties (Honest Majority)

Honest Majority MPC

Honest Majority MPC

Information-Theoretic security is possible.

[Ben-Or, Goldwasser, Widgerson’88]

Honest Majority MPC

Information-Theoretic security is possible.

[Ben-Or, Goldwasser, Widgerson'88]

Typically UC secure

Simulation proofs are typically straight-line

Honest Majority MPC

Information-Theoretic security is possible.

[Ben-Or, Goldwasser, Widgerson’88]

Typically UC secure

Simulation proofs are typically straight-line

Round complexity lower bounds for dishonest majority do not apply

4 rounds necessary for dishonest majority in the plain model

[Garg- Mukherjee-Pandey-Polychroniadou16]

Honest Majority MPC

Information-Theoretic security is possible.

[Ben-Or, Goldwasser, Widgerson'88]

Typically UC secure

Simulation proofs are typically straight-line

Round complexity lower bounds for dishonest majority do not apply

4 rounds necessary for dishonest majority in the plain model

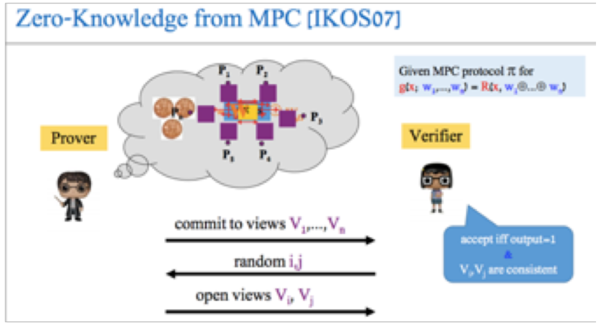
[Garg- Mukherjee-Pandey-Polychroniadou16]

Clean Constructions

Use lightweight tools such as garbling and secret-sharing

Honest Majority MPC: Applications

Efficient Zero-Knowledge
[IKOS'07,…]

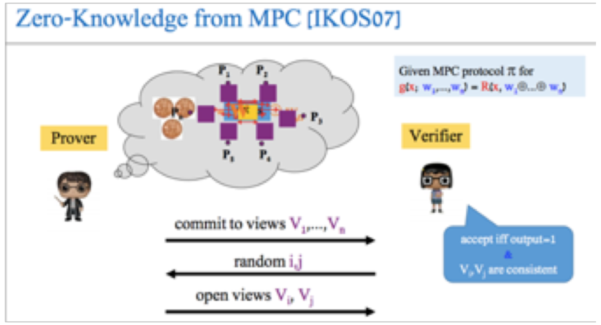


(Courtesy: Carmit Hazay's talk)

Useful for constructing efficient ZK-protocols.

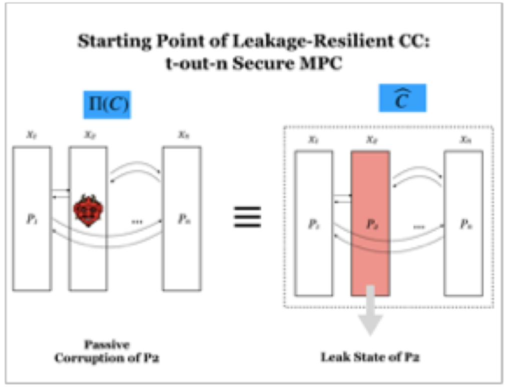
Honest Majority MPC: Applications

Efficient Zero-Knowledge
[IKOS’07,…]

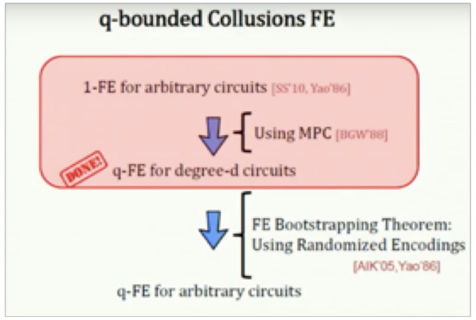


(Courtesy: Carmit Hazay’s talk)

Leakage-Resilient Circuit Compilers
[ISW03,FKKNV10,AIS18]



Bounded-Key Functional Encryption
[GVW12,AV18]



(Courtesy: Sergey Gorbunov’s talk)

History of IT-MPC

	Round Complexity	Class of Functions	Corruption Threshold	Adversary
[BGW’88]	> # of multiplications	P/Poly	t<n/2	Malicious
[BB’89, IK’00, AIK’06]	constant	NC^1	t<n/2	Malicious
[IKP’10]	2	NC^1	t<n/3	Malicious
[GIS’18, ABT’18]	2	NC^1	t<n/2	Semi-honest

Security with selective abort

Our Results

Round Complexity	Class of Functions	Corruption Threshold	Adversary
2	NC^1	$t<n/2$	Malicious

Security with Abort over
Broadcast + P2P

Security with Selective Abort over
P2P

Our Results

Round Complexity	Class of Functions	Corruption Threshold	Adversary
2	NC^1	$t<n/2$	Malicious

Security with Abort over Broadcast + P2P

Security with Selective Abort over P2P

Concurrent Work [ABT19]

Consider security with selective abort.

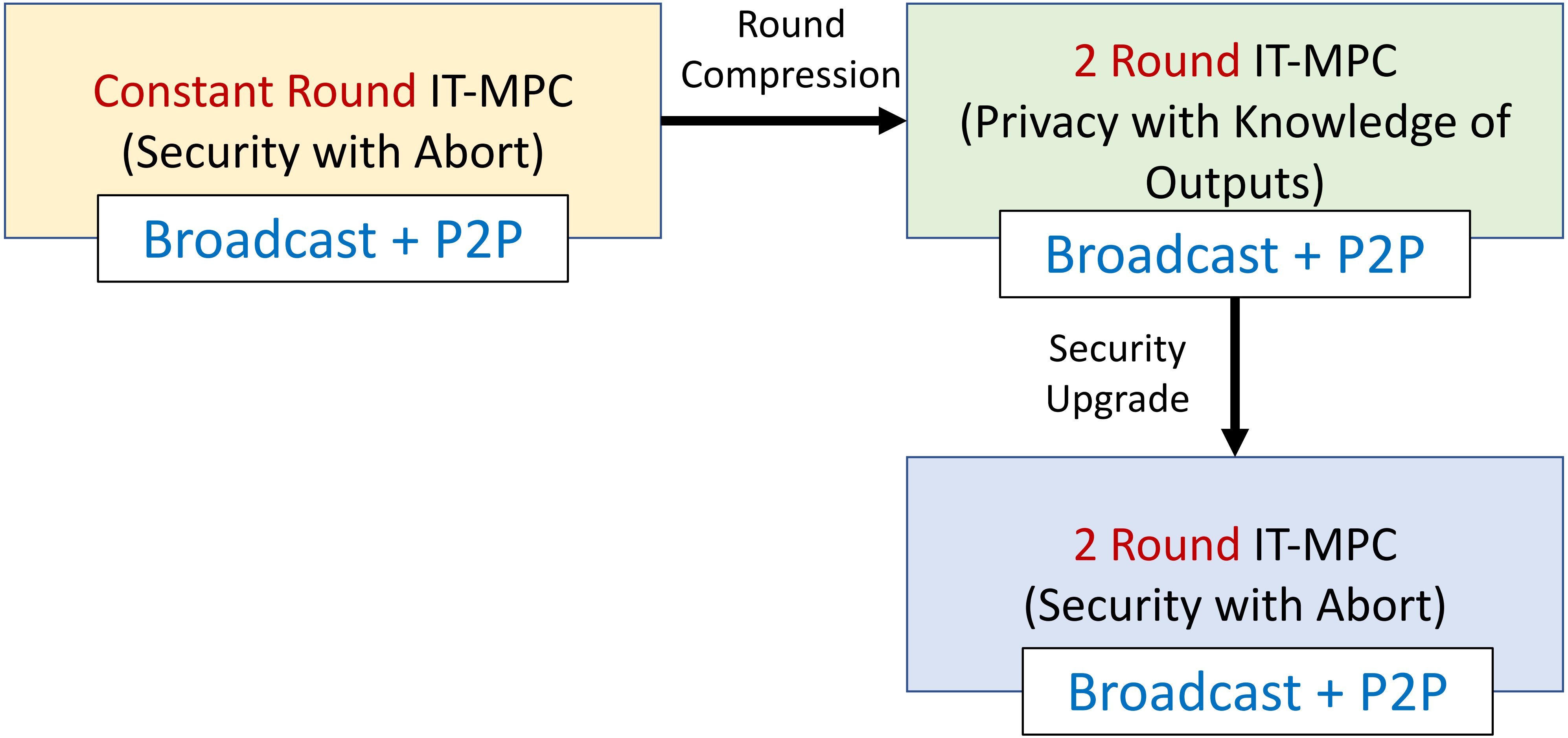
This Talk

Round Complexity	Class of Functions	Corruption Threshold	Adversary
2	NC^1	$t<n/2$	Malicious

Security with Abort over Broadcast + P2P

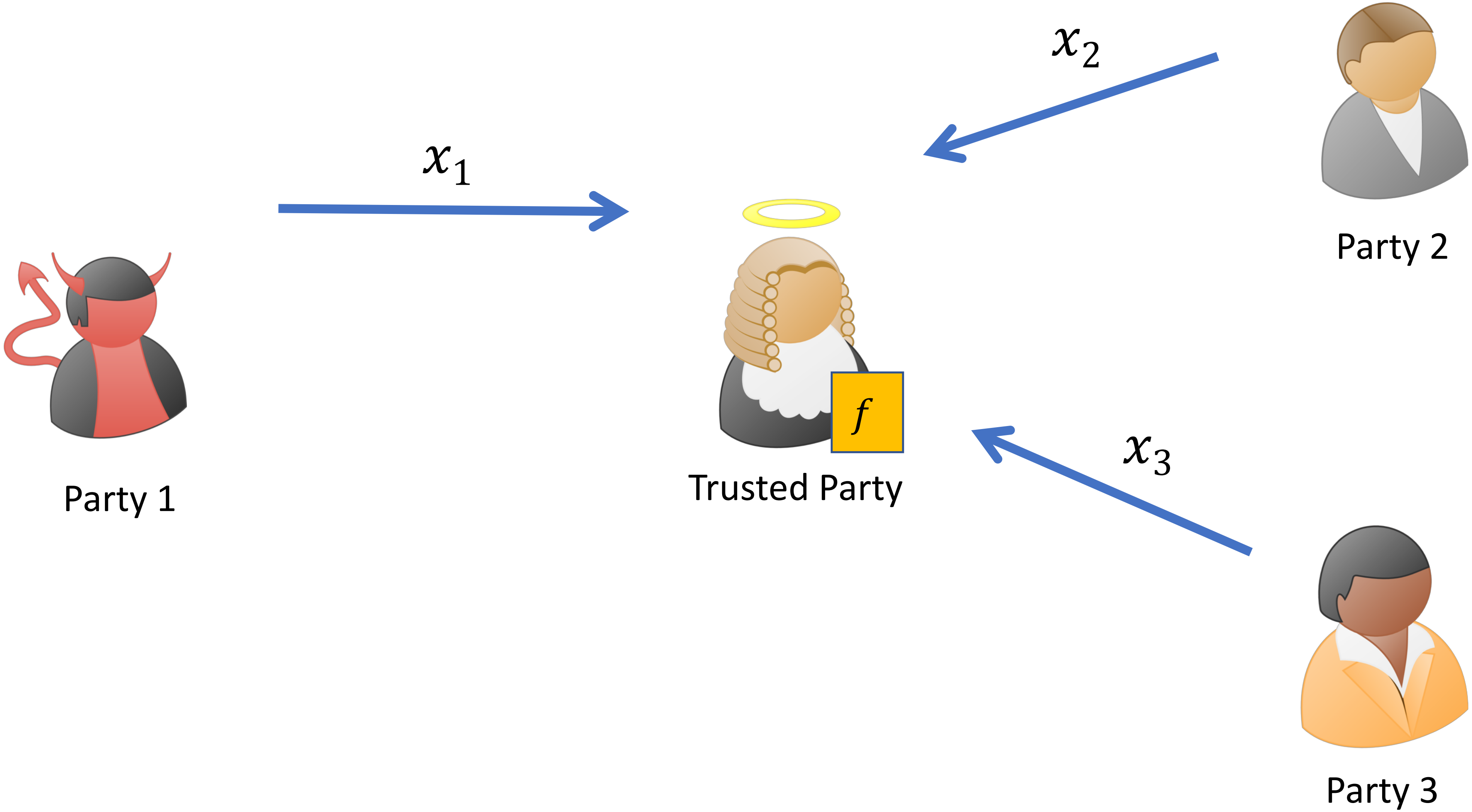
Security with Selective Abort over P2P

Our Strategy

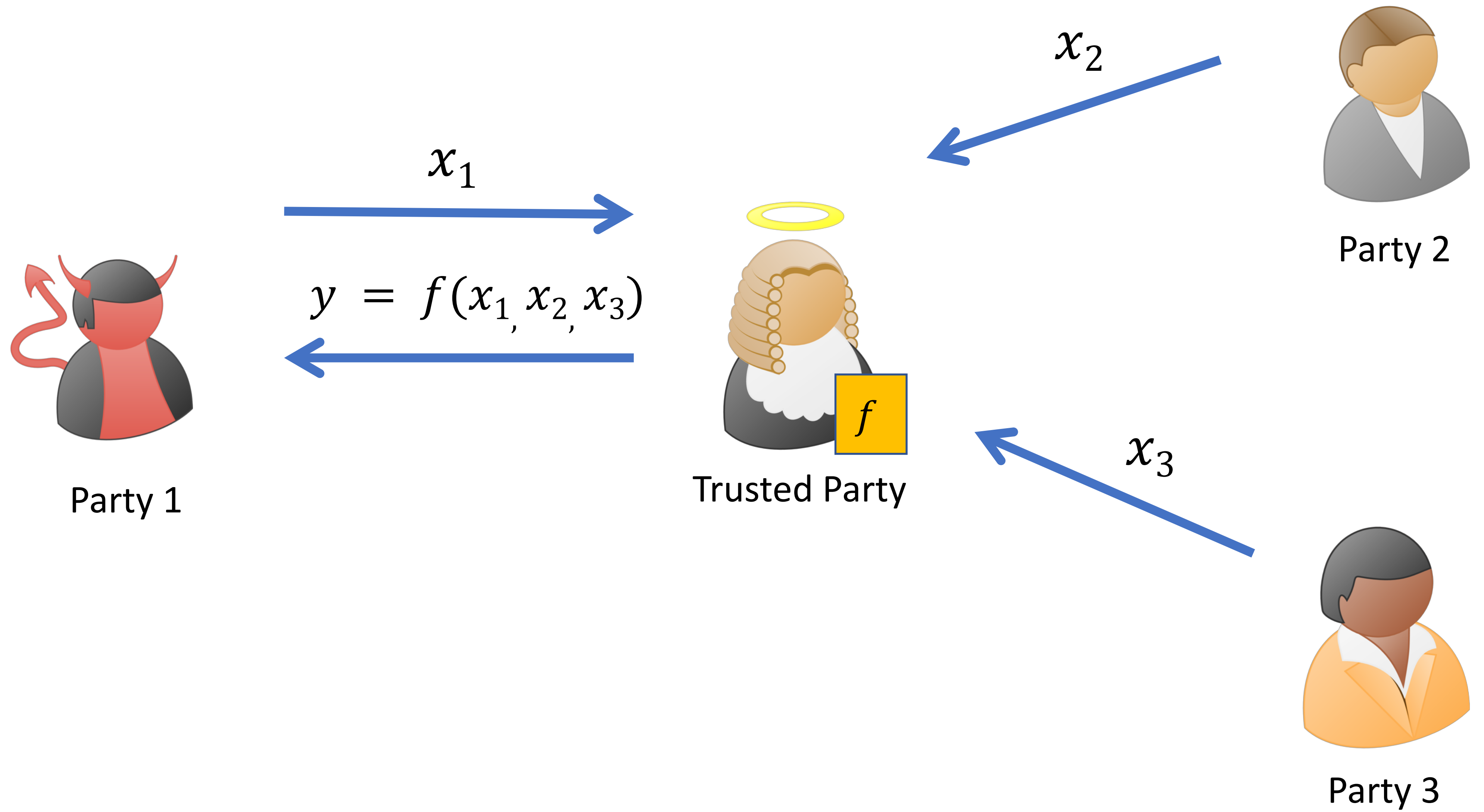


Security with Abort

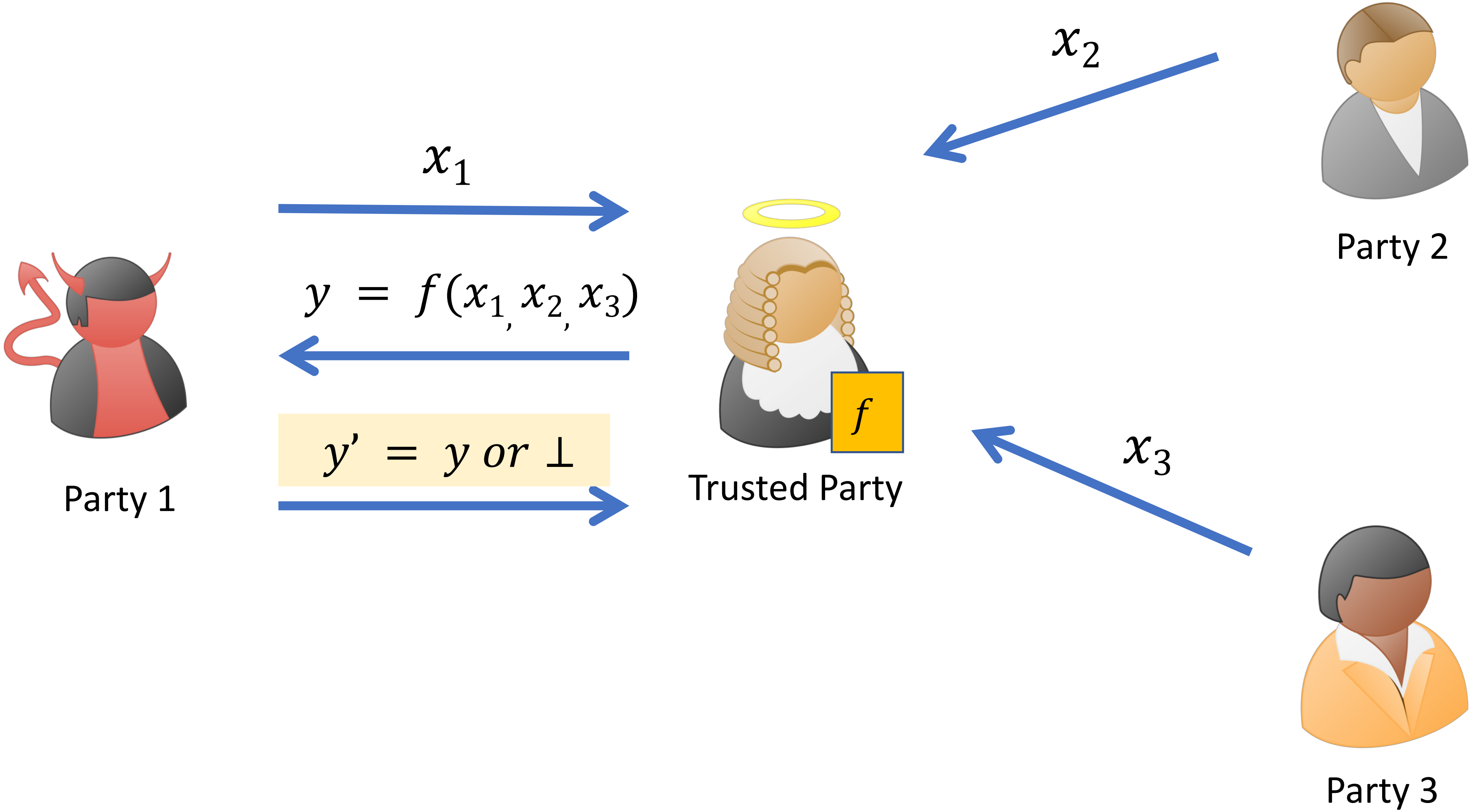
Security with Abort



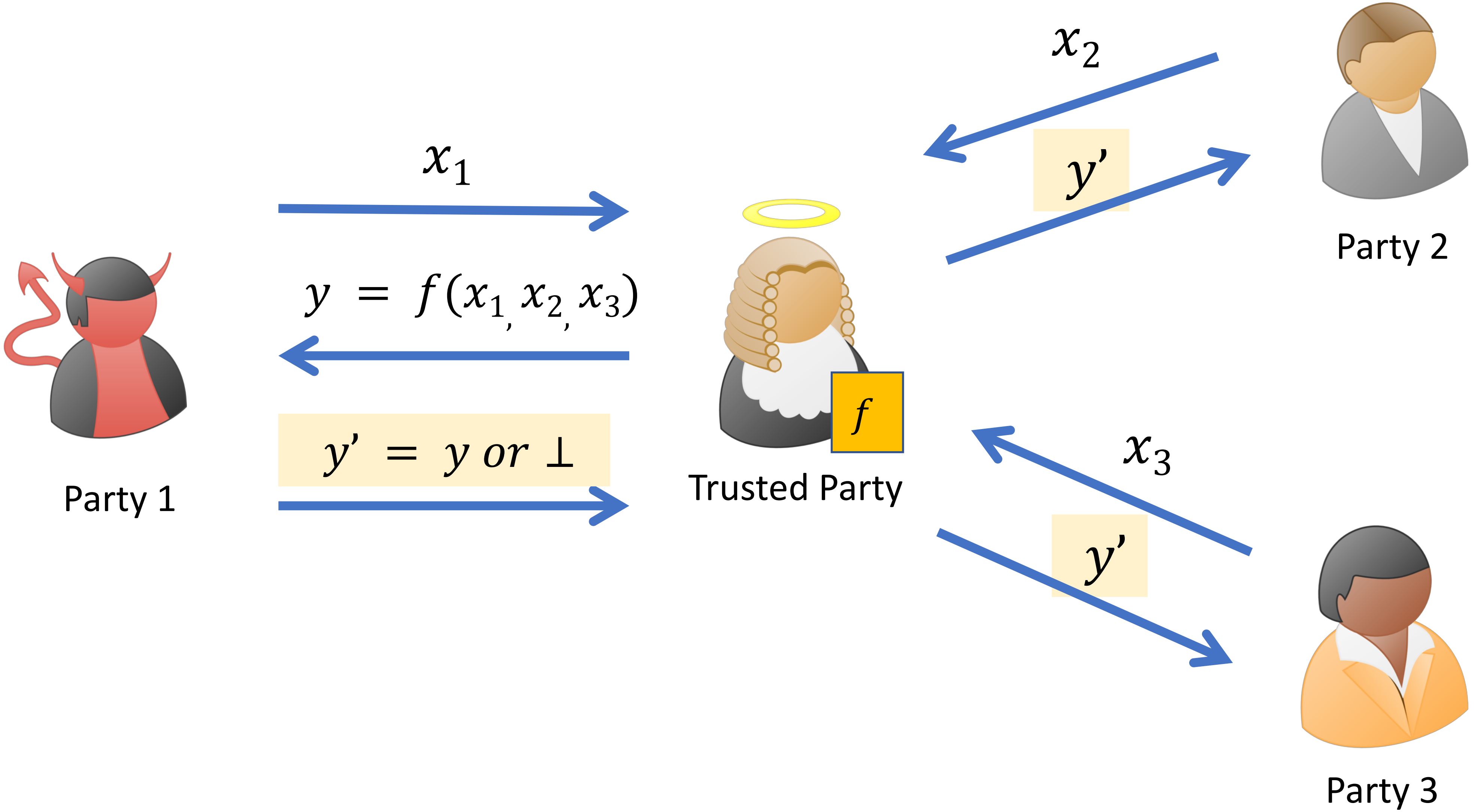
Security with Abort



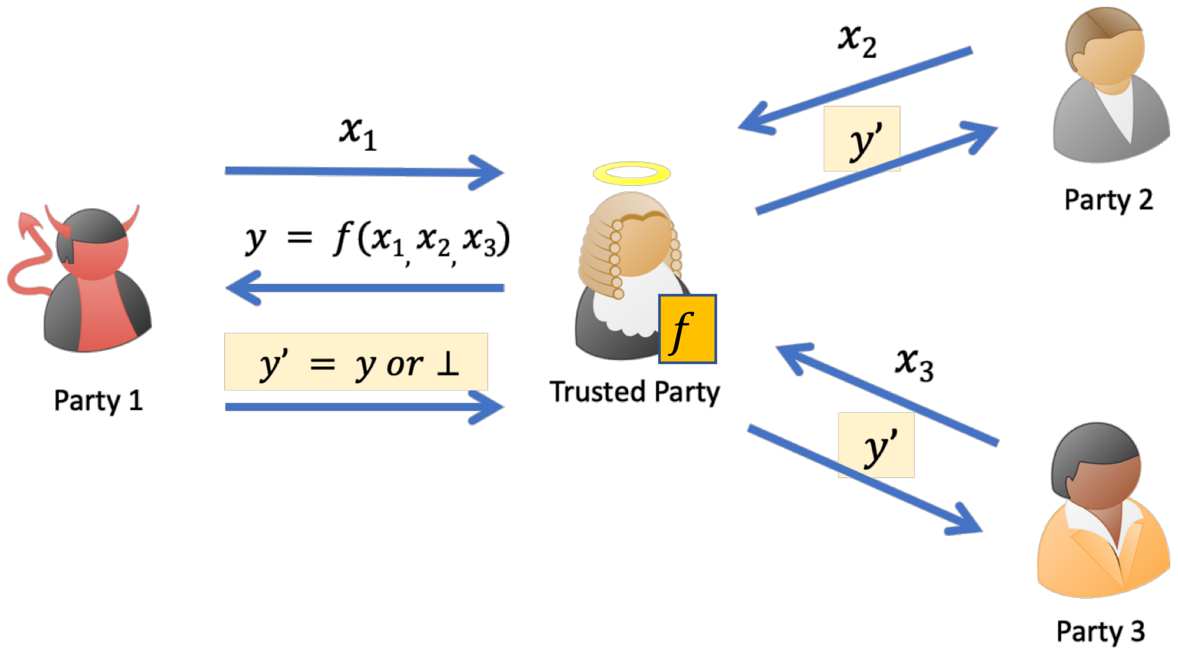
Security with Abort



Security with Abort



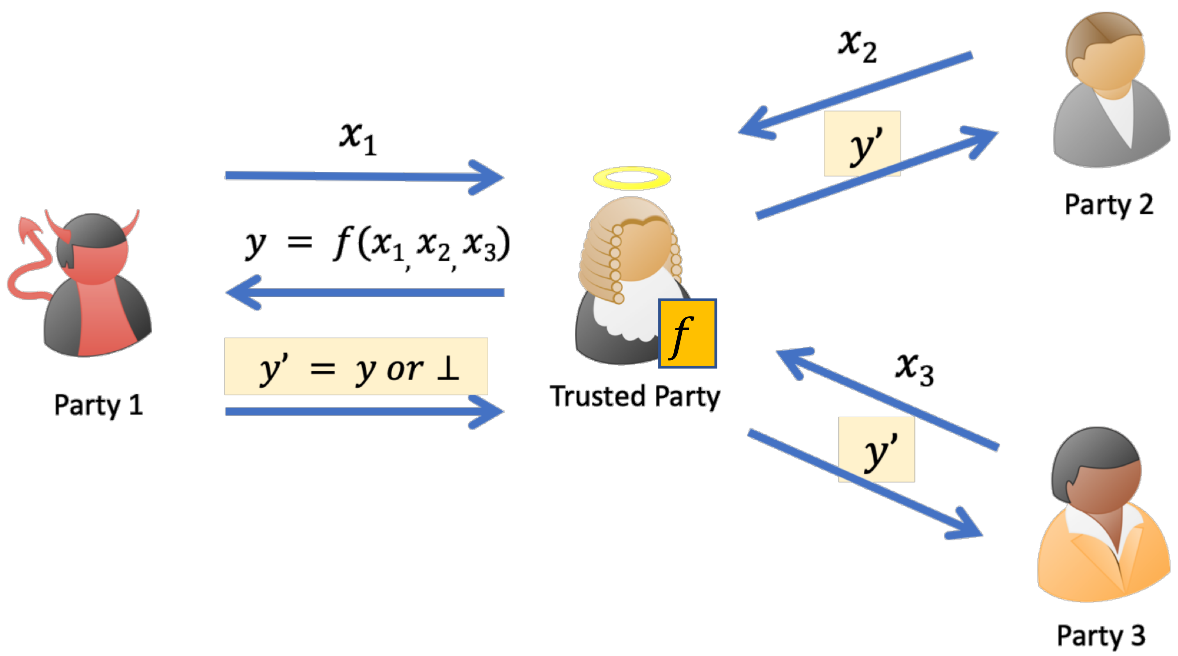
Security with Abort



Privacy

x_2 and x_3 remain hidden

Security with Abort



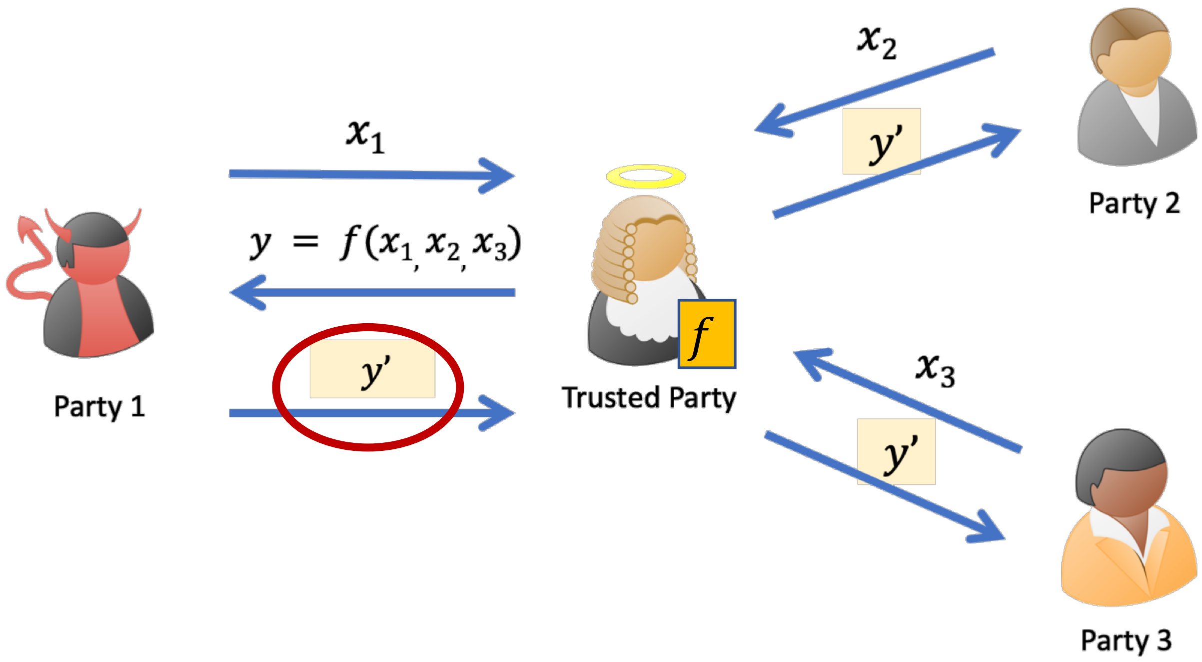
Privacy

x_2 and x_3 remain hidden

Output Correctness

Honest Parties either output $f(x_1, x_2, x_3)$ or $\perp$

Privacy with Knowledge of Outputs



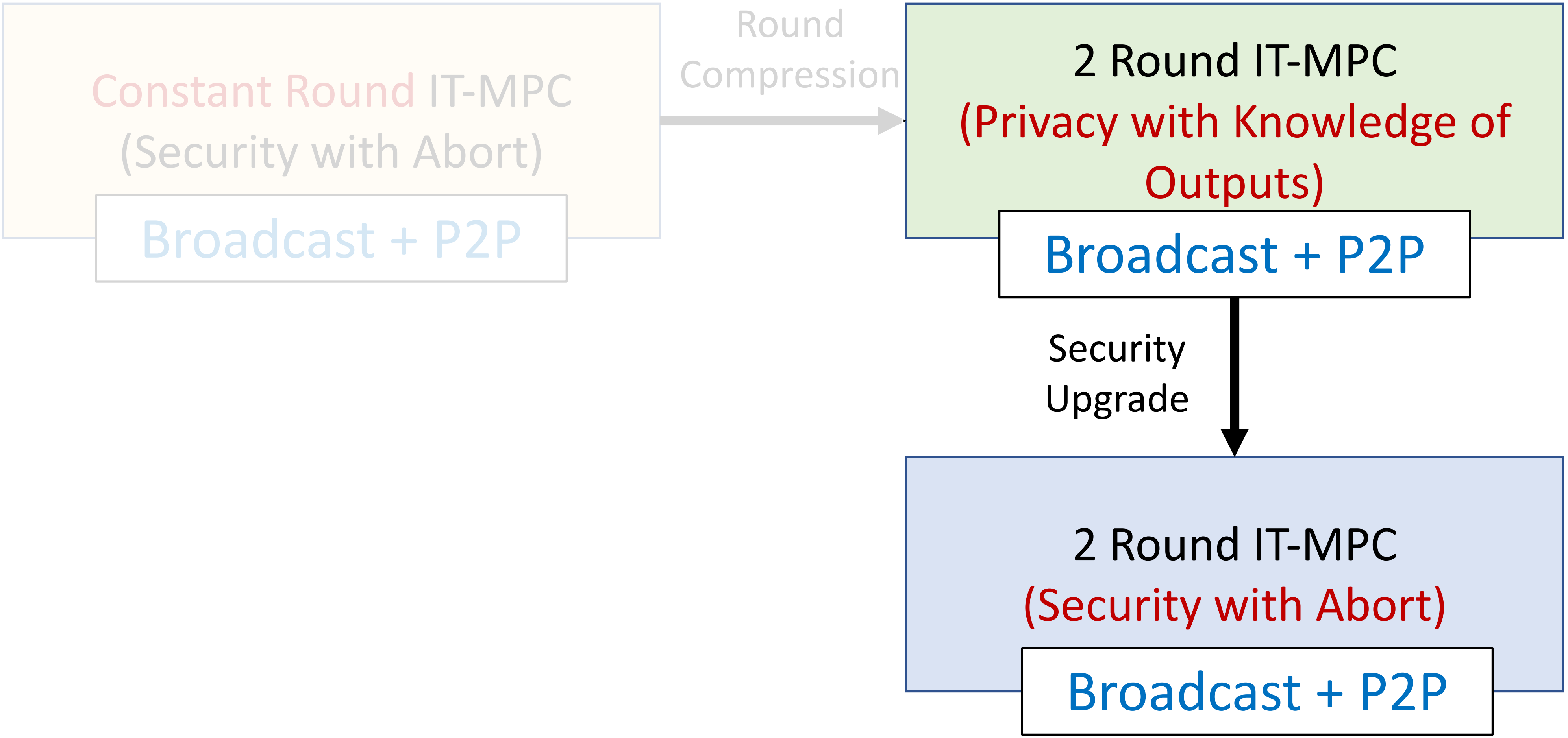
Privacy

x_2 and x_3 remain hidden

~~Output Correctness~~

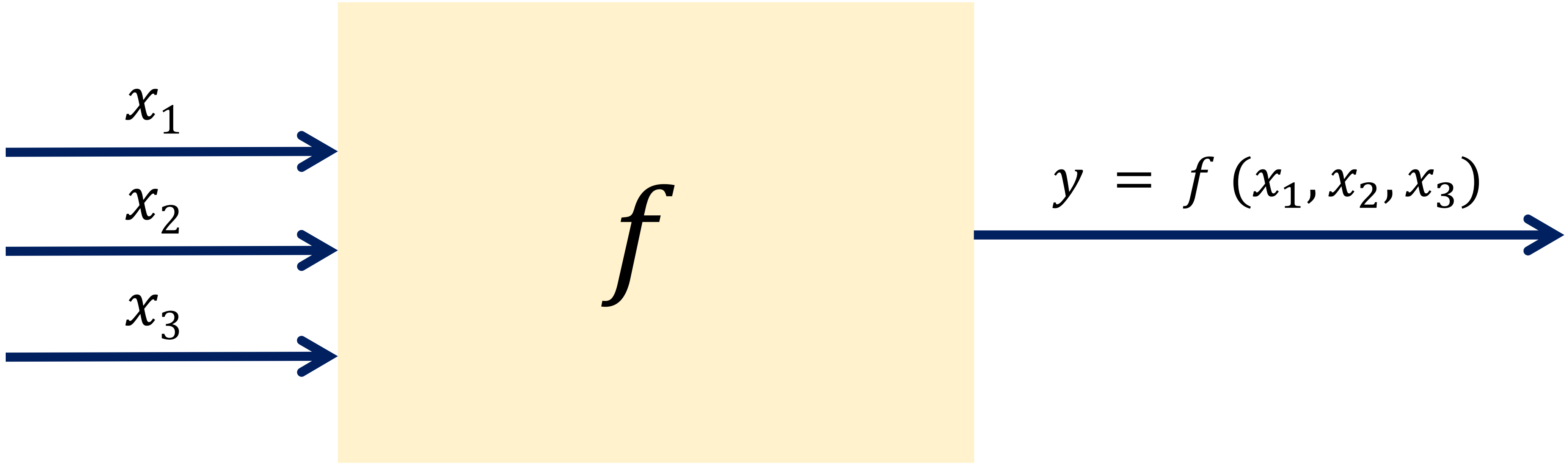
~~Honest Parties either output $f(x_1, x_2, x_3)$ or $\perp$~~

First Step

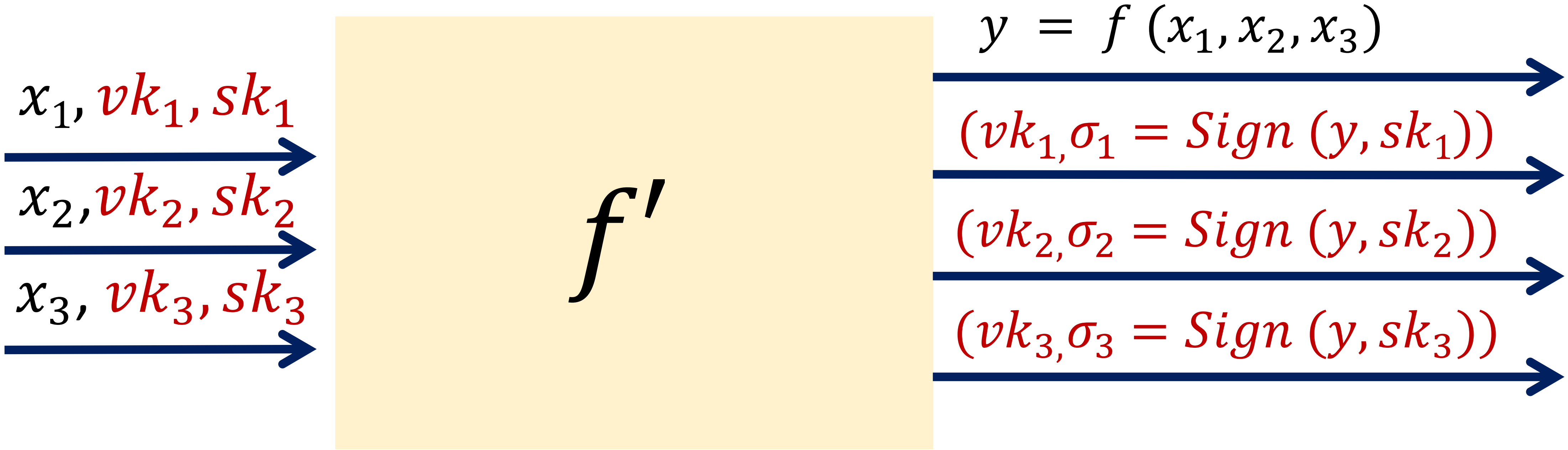


Constant Round IT-MPC
(Security with Abort)
Broadcast + P2P
Round
Compression
2 Round IT-MPC
(Privacy with Knowledge of Outputs)
Broadcast + P2P
Security
Upgrade
2 Round IT-MPC
(Security with Abort)
Broadcast + P2P

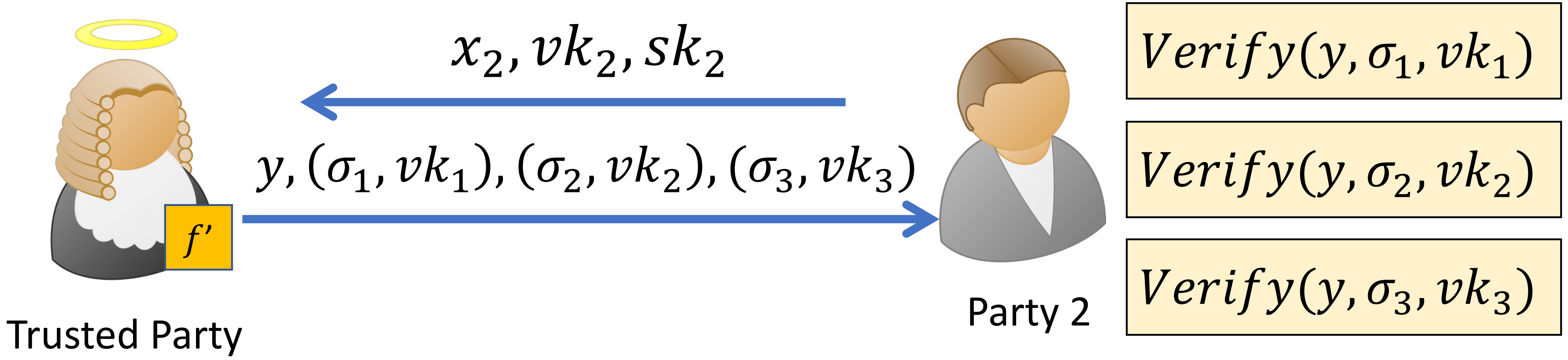
Using Signed Outputs [IKP10]



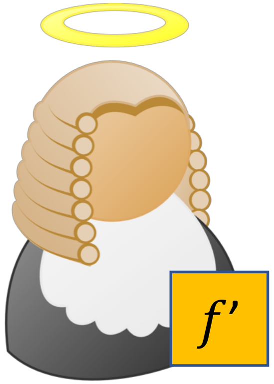
Using Signed Outputs [IKP10]



Security with abort: Using Signed Outputs



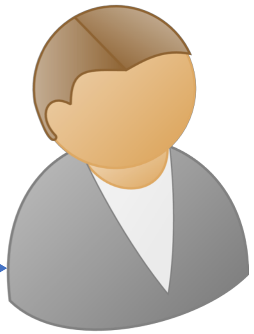
Security with abort: Using Signed Outputs



Trusted Party

x_2, vk_2, sk_2

$y, (\sigma_1, vk_1), (\sigma_2, vk_2), (\sigma_3, vk_3)$



Party 2

$Verify(y, \sigma_1, vk_1)$

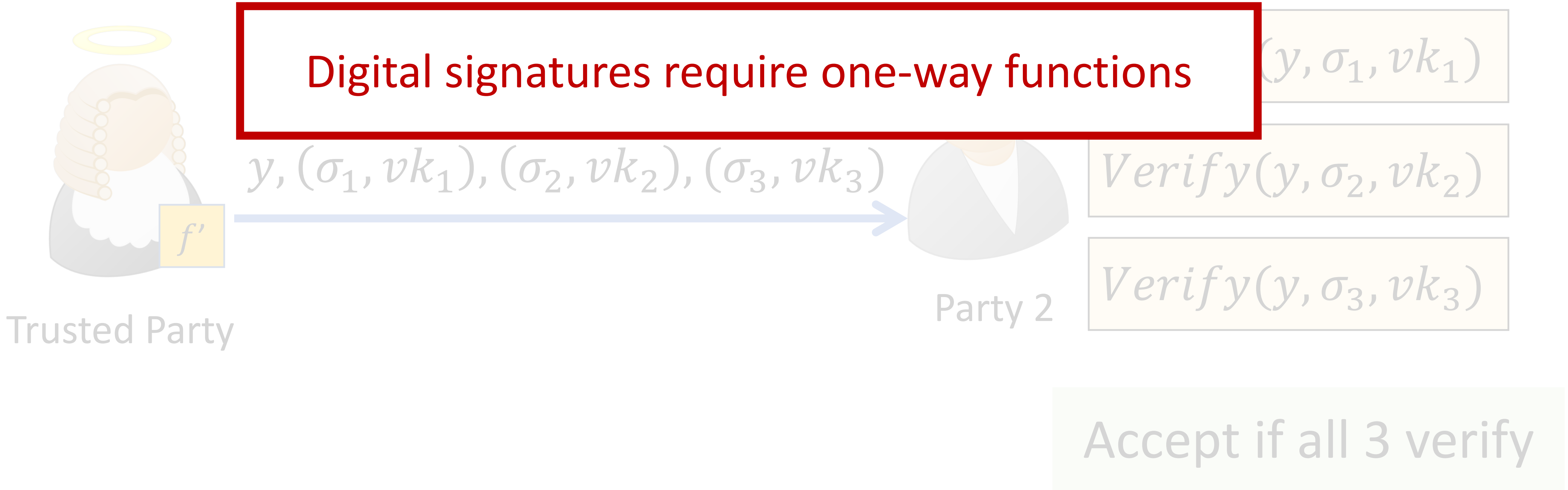
$Verify(y, \sigma_2, vk_2)$

$Verify(y, \sigma_3, vk_3)$

Accept if all 3 verify

Security with abort: Using Signed Outputs

Digital signatures require one-way functions



Security with abort: Using Signed Outputs

Digital signatures require one-way functions

MACs are not sufficient

Trusted Party

$Verify(y, \sigma_2, vk_2)$

Accept if all 3 verify

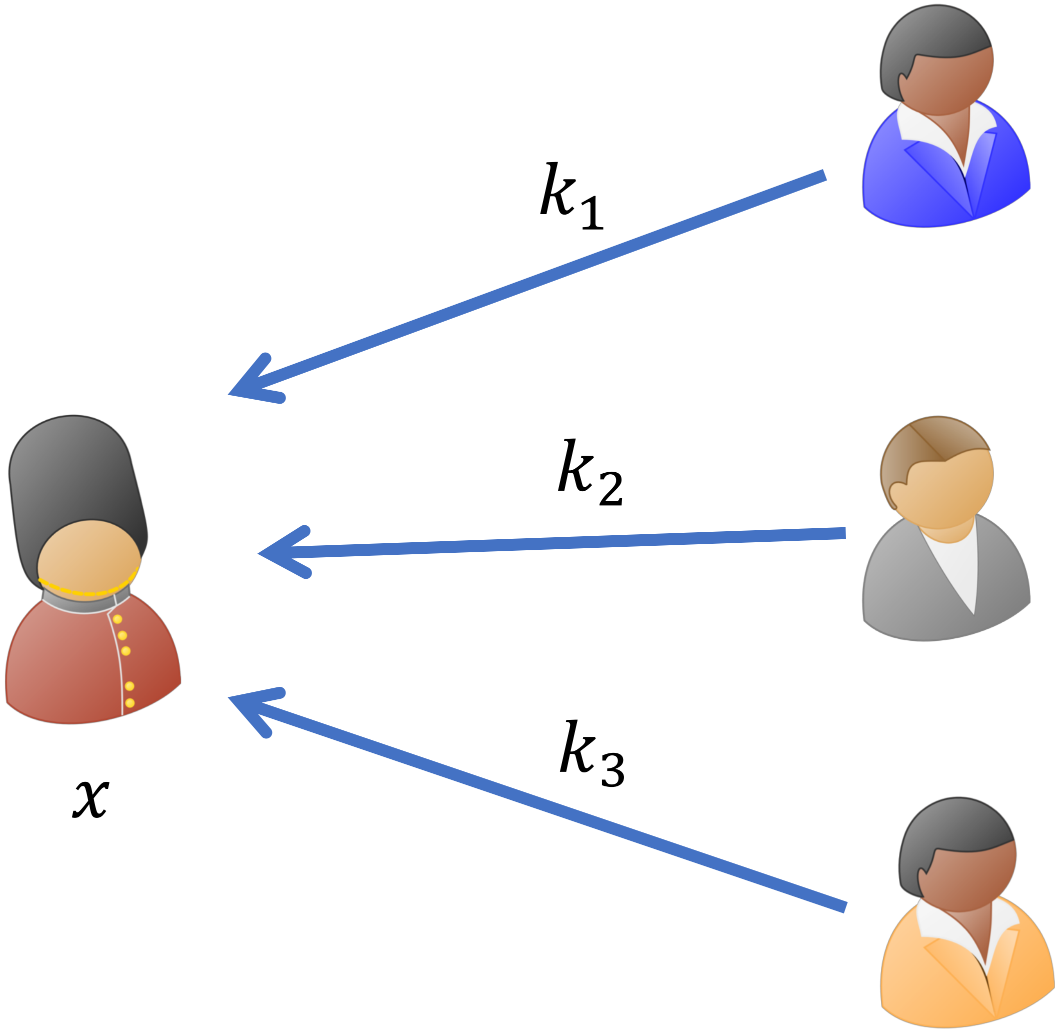
Security with abort: Using Signed Outputs

Digital signatures require one-way functions

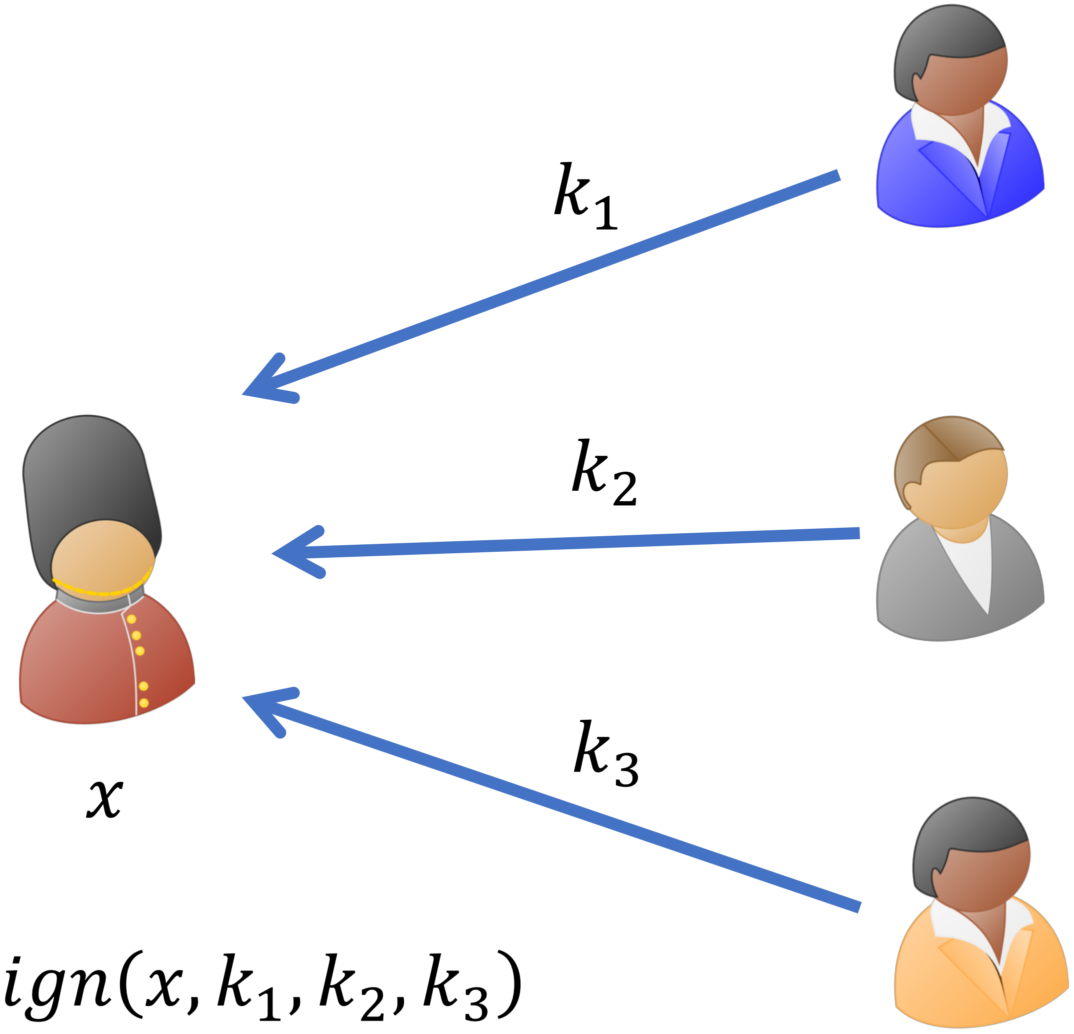
MACs are not sufficient

How can we do it information theoretically?

Our Tool: Multi-Key MAC

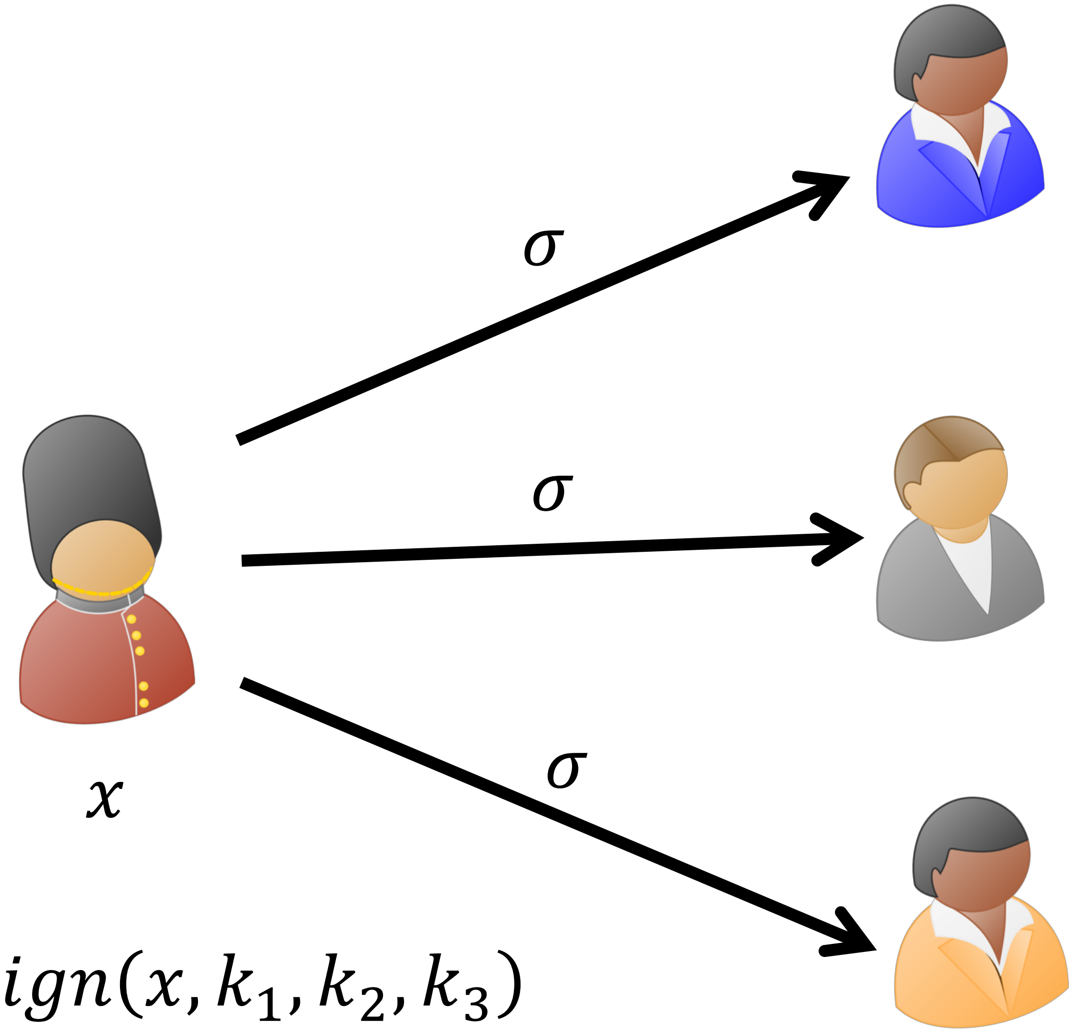


Our Tool: Multi-Key MAC



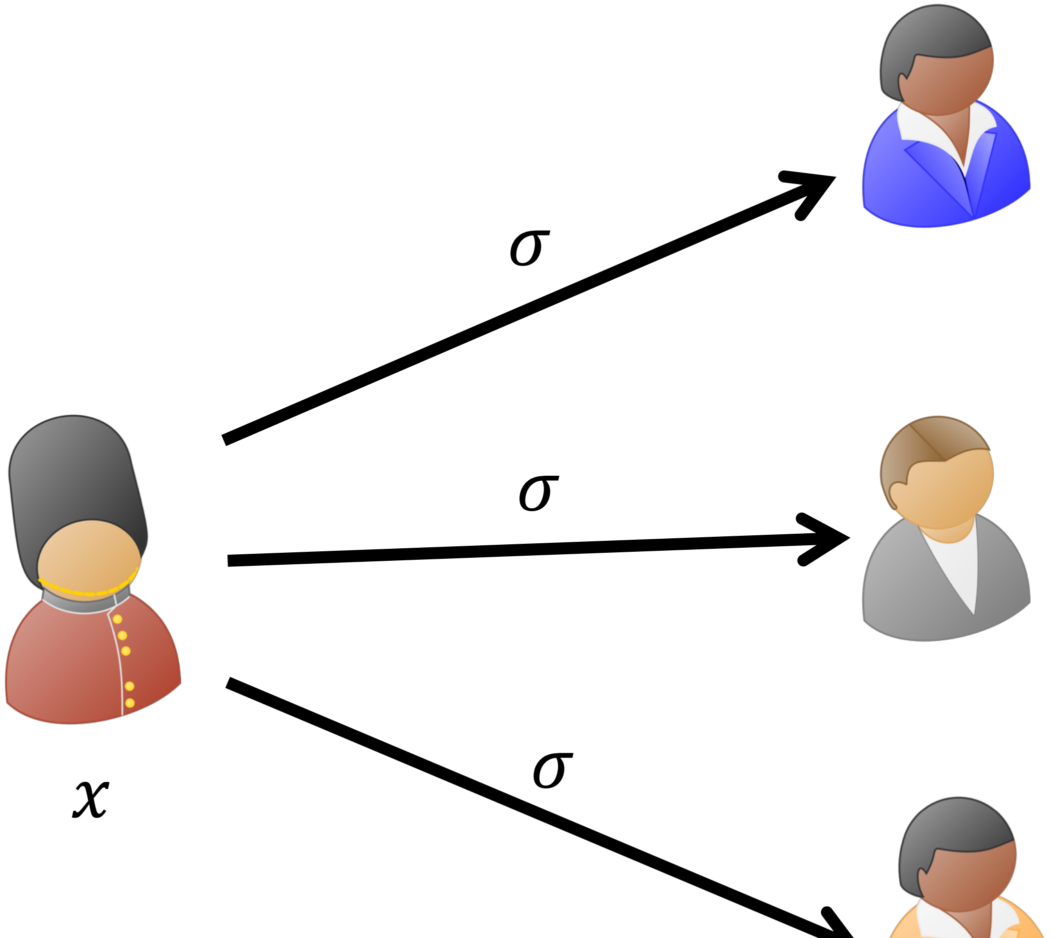
$$\sigma = M.Sign(x, k_1, k_2, k_3)$$

Our Tool: Multi-Key MAC



$$\sigma = M.Sign(x, k_1, k_2, k_3)$$

Our Tool: Multi-Key MAC



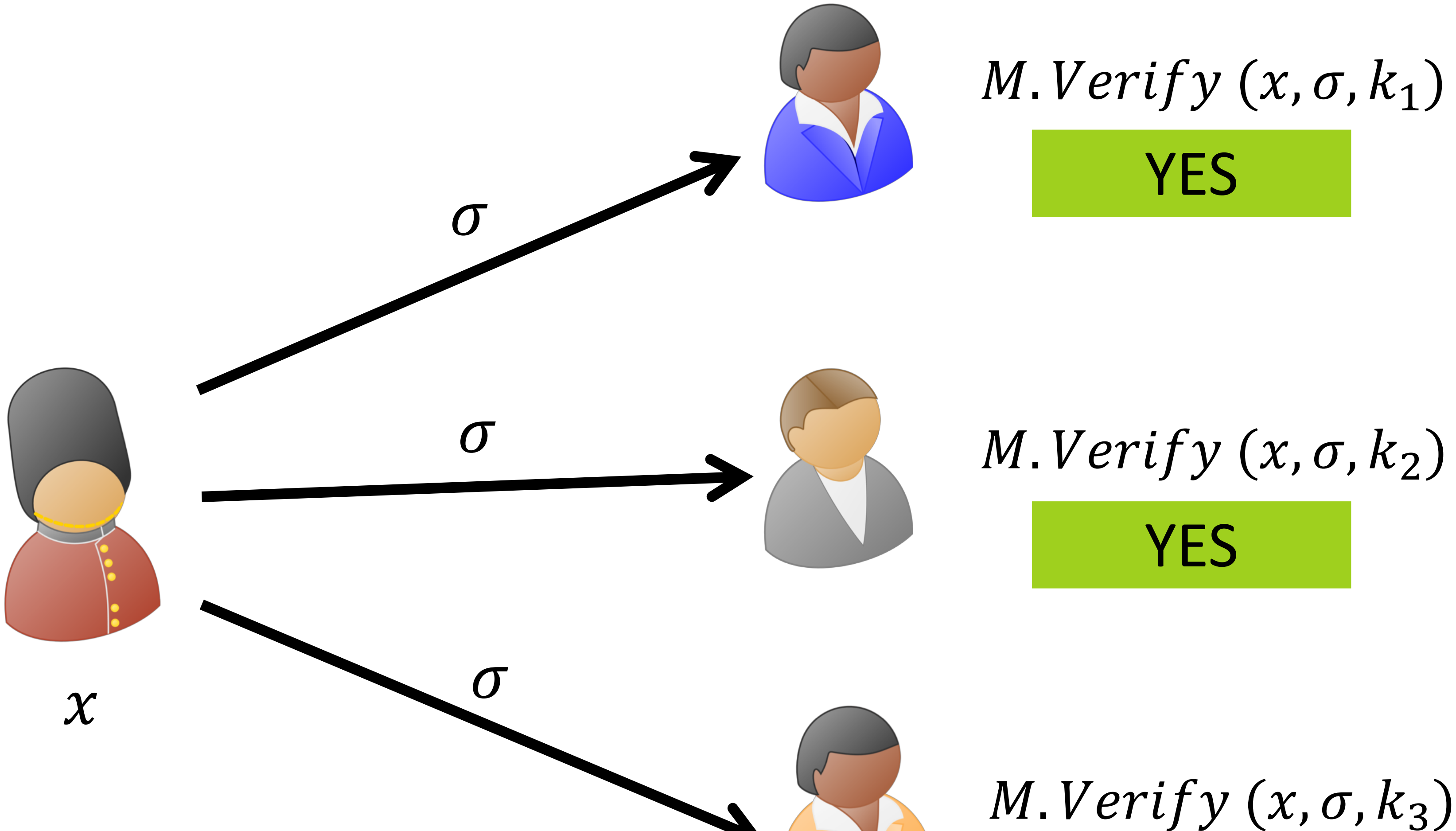
$M.Verify\,(x, \sigma, k_1)$

$M.Verify\,(x, \sigma, k_2)$

$M.Verify\,(x, \sigma, k_3)$

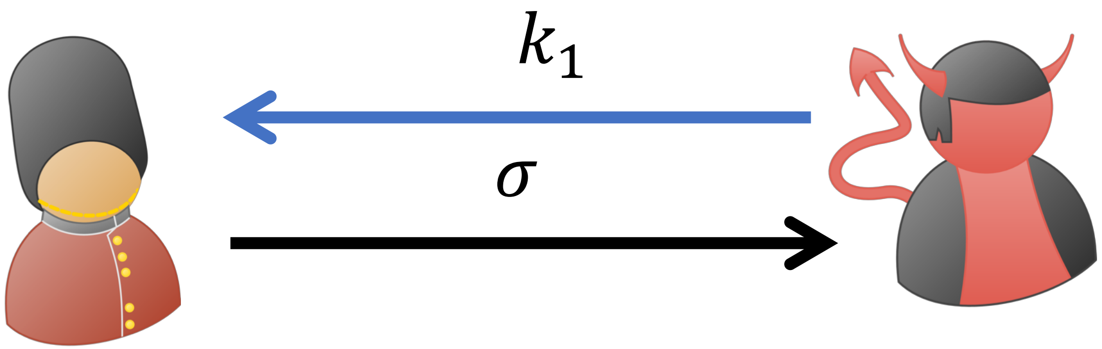
$\sigma = M.Sign(x, k_1, k_2, k_3)$

Our Tool: Multi-Key MAC (Correctness)



$$\sigma = M.Sign(x, k_1, k_2, k_3)$$

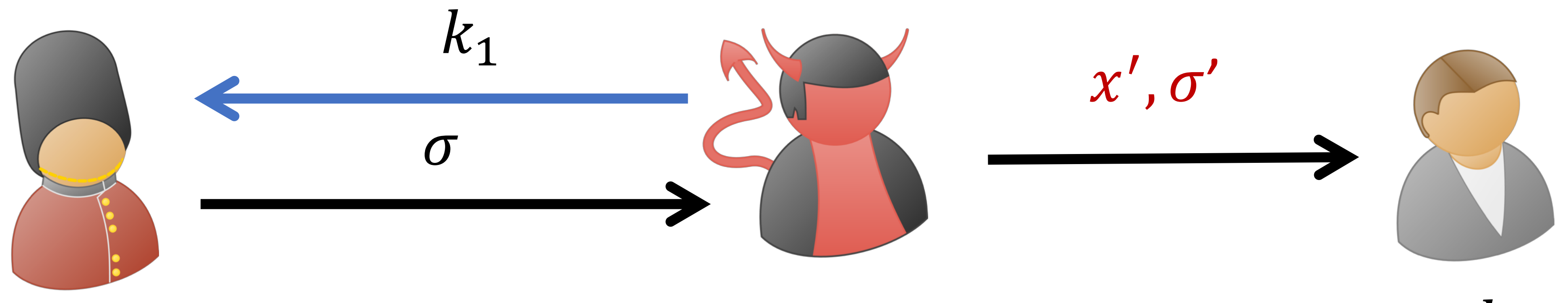
Our Tool: Multi-Key MAC (Security)



x, k_2, k_3

$$\sigma = M.Sign(x, k_1, k_2, k_3)$$

Our Tool: Multi-Key MAC (Security)

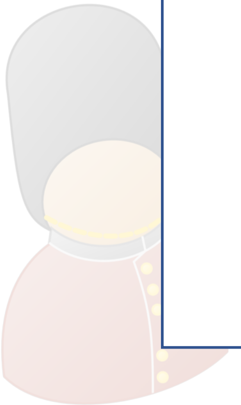


$\sigma = M.Sign(x, k_1, k_2, k_3)$

$M.Verify\,(x', \sigma', k_2)$

NO

Our Tool: Multi-Key MAC (Security)



An adversary cannot output any valid message-signature pair other than the one it received

k_2

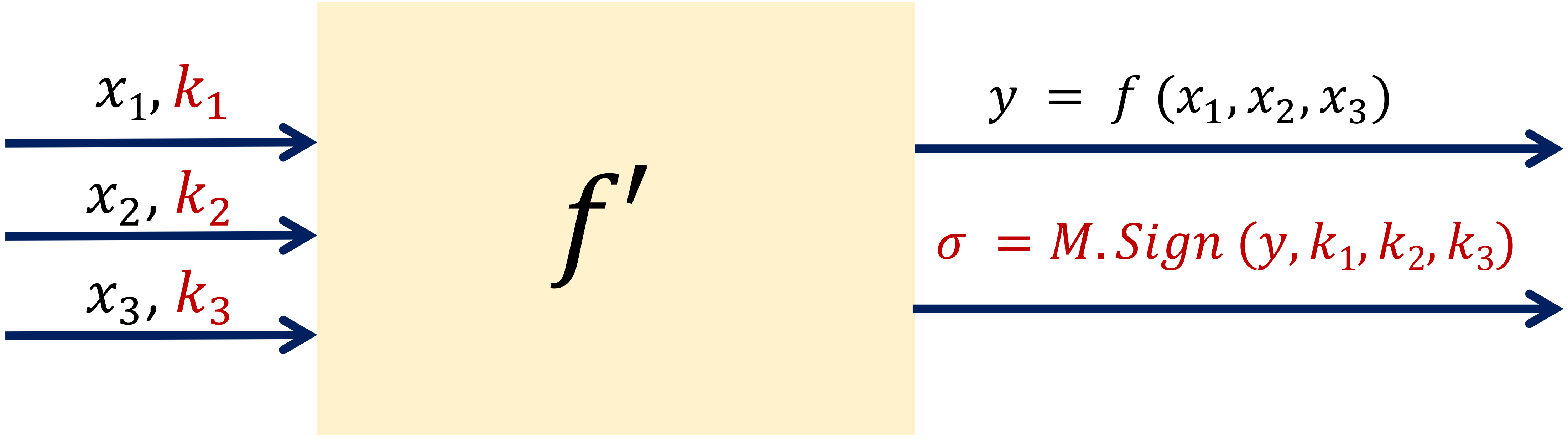
x, k_2, k_3

$M.Verify\,(x', \sigma', k_2)$

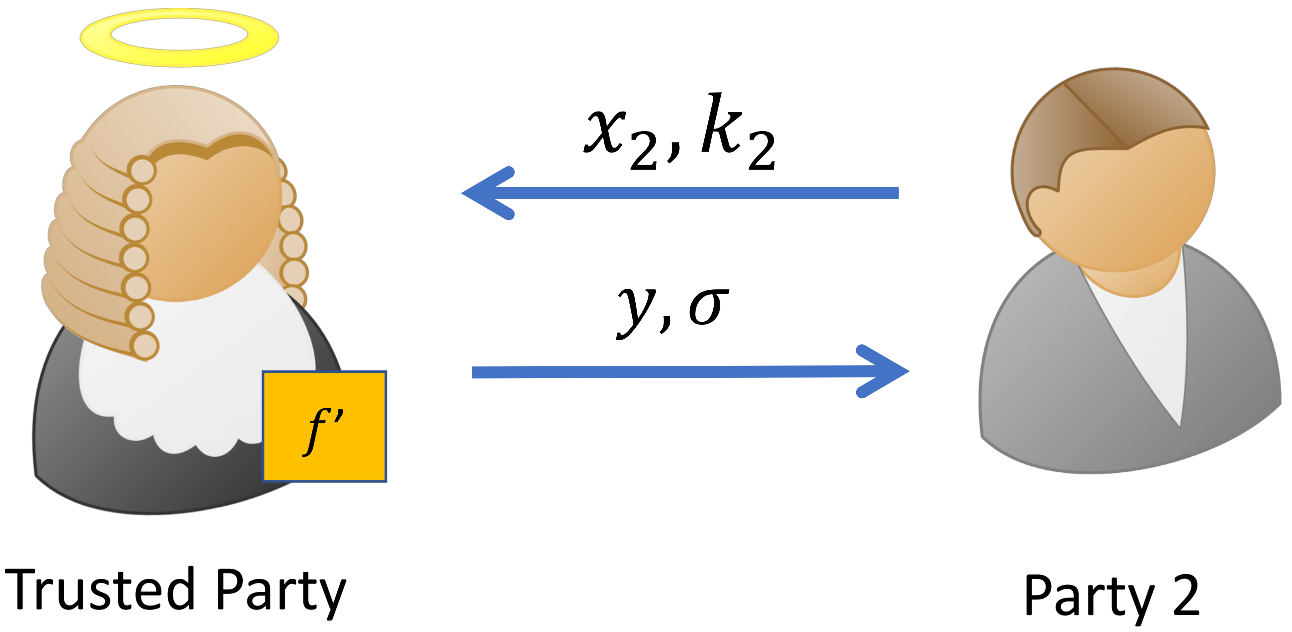
$\sigma = Sign(x, k_1, k_2, k_3)$

NO

Security with Abort: Using Multi-Key MAC

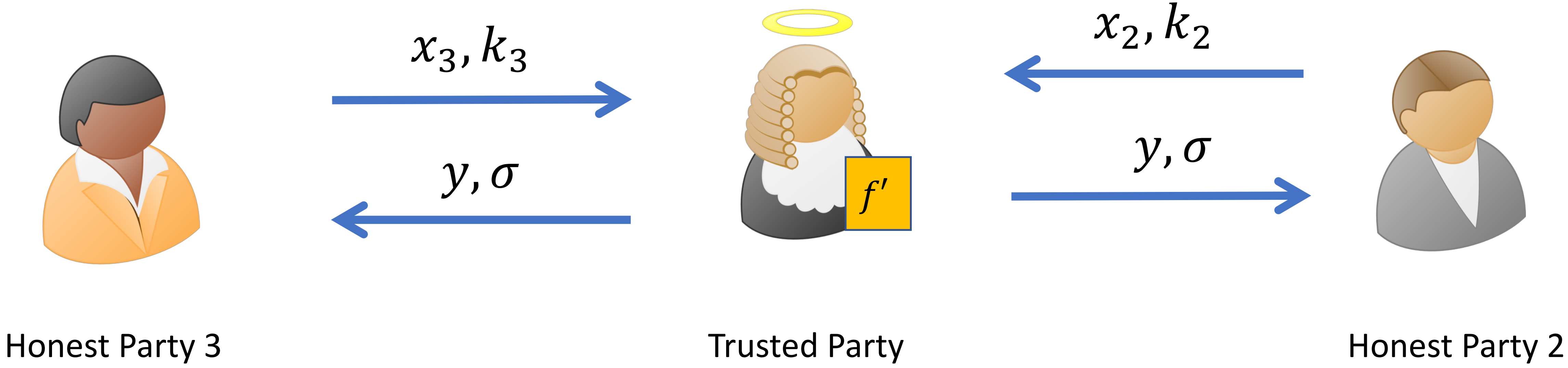


Security with Abort: Using Multi-Key MAC



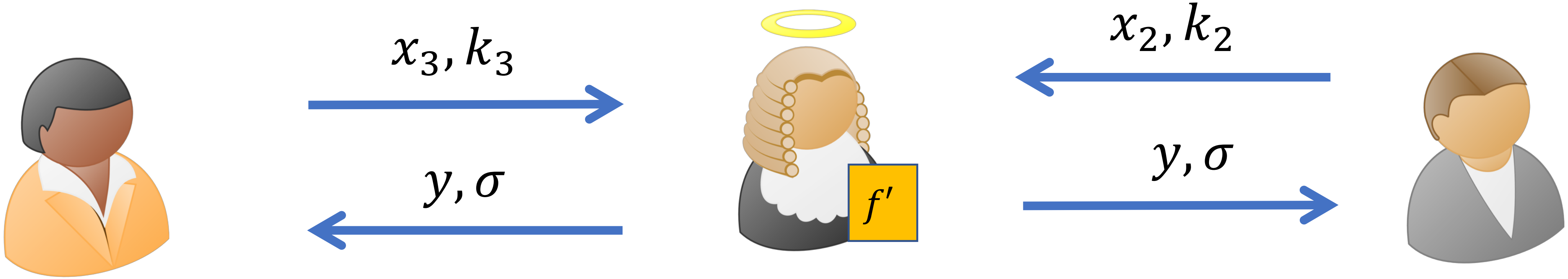
$$M.Verify(y, \sigma, k_2)$$

Security with abort: Using Multi-Key MAC



IF $y, \sigma = f'((x_1 k_1), (x_2, k_2), (x_3, k_3))$

Security with abort: Using Multi-Key MAC



Honest Party 3

Trusted Party

Honest Party 2

$M.Verify(y, \sigma, k_3)$

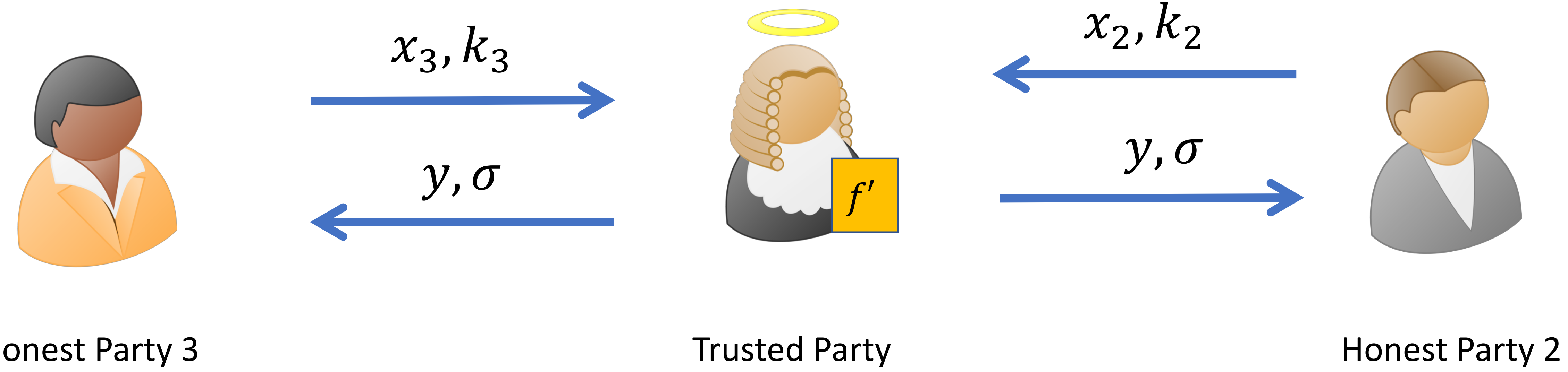
IF $y, \sigma = f'((x_1 k_1), (x_2, k_2), (x_3, k_3))$

$M.Verify(y, \sigma, k_2)$

YES

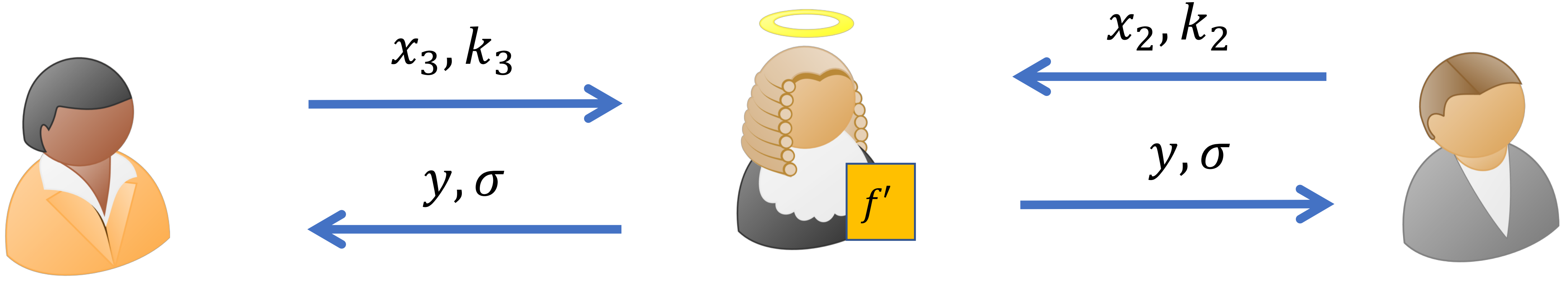
YES

Security with abort: Using Multi-Key MAC



IF $y, \sigma \neq f'((x_1 k_1), (x_2, k_2), (x_3, k_3))$

Security with abort: Using Multi-Key MAC



Honest Party 3

Trusted Party

Honest Party 2

$M.Verify(y, \sigma, k_3)$

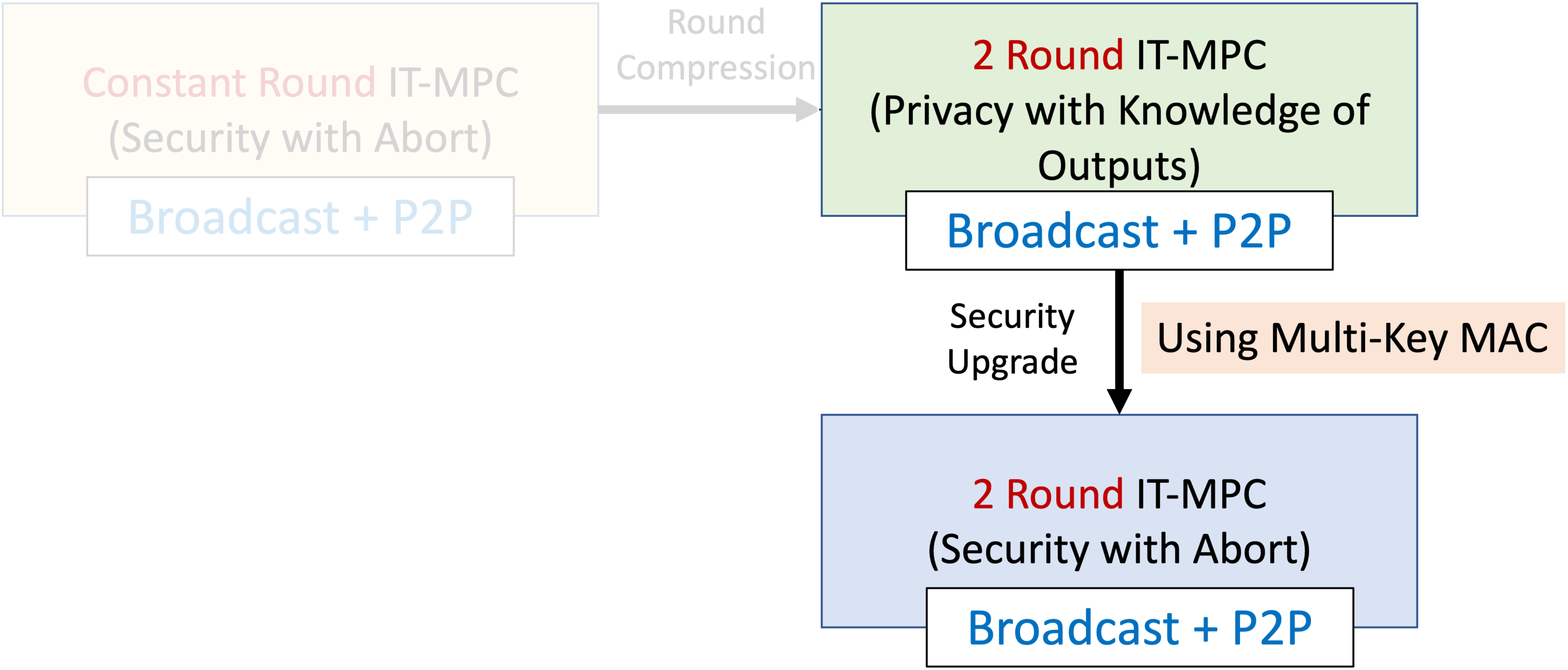
NO

IF $y, \sigma \neq f'((x_1 k_1), (x_2, k_2), (x_3, k_3))$

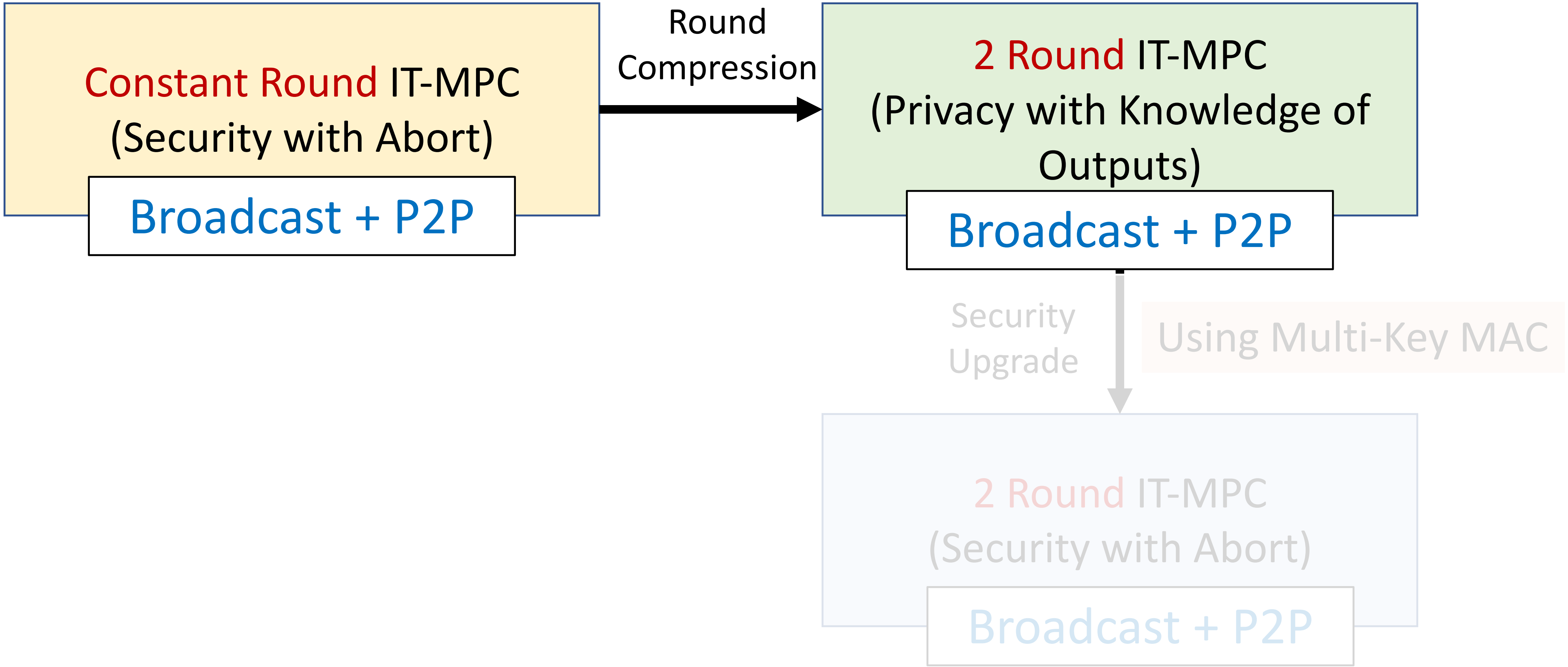
$M.Verify(y, \sigma, k_2)$

NO

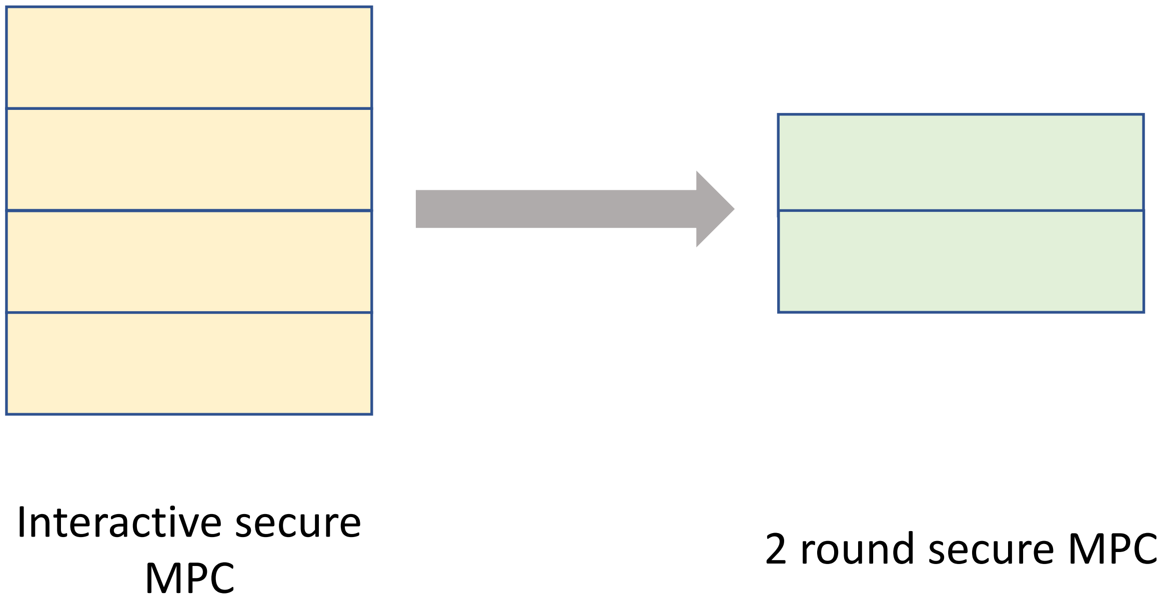
Recall: Our Strategy



Second Step


Constant Round IT-MPC
(Security with Abort)
Broadcast + P2P
Round
Compression
2 Round IT-MPC
(Privacy with Knowledge of
Outputs)
Broadcast + P2P
Security
Upgrade
Using Multi-Key MAC
2 Round IT-MPC
(Security with Abort)
Broadcast + P2P

Technique: Round Compression



[GGHR'13]
Indistinguishability Obfuscation

[GLS'15]
Witness Encryption + Garbled circuits

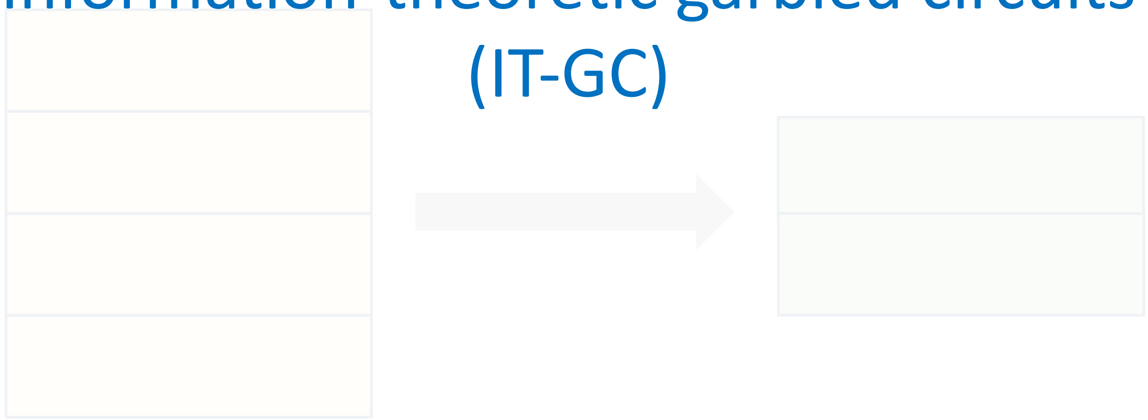
[GS'17]
Bilinear Maps + Garbled circuits

[GS'18, BL'18]
OT + Garbled Circuits

[ACGJ'18]
Garbled circuits

Initial Idea

Replace garbled circuits with
Information-theoretic garbled circuits
(IT-GC)



Interactive secure MPC

2 round secure MPC

[GGHR'13]
Indistinguishability Obfuscation

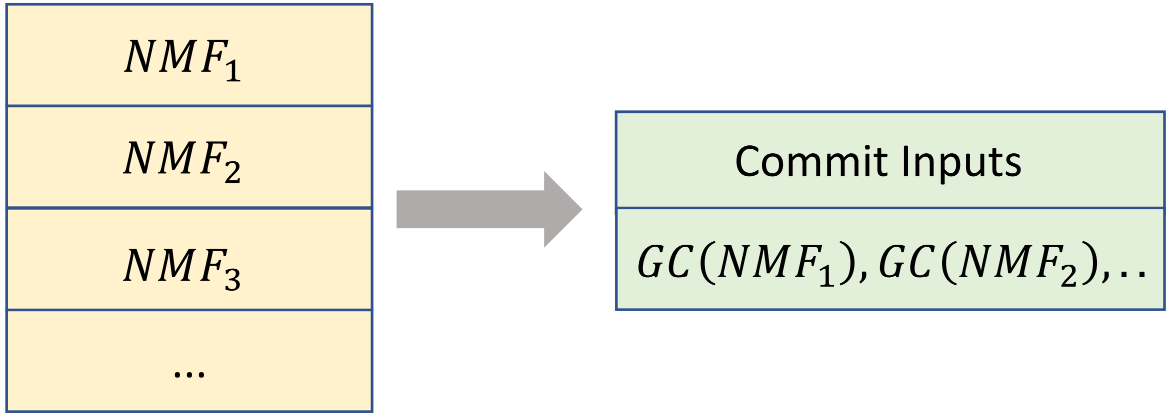
[GLS'15]
Witness Encryption + Garbled circuits

[GS'17]
Bilinear Maps + Garbled circuits

[GS'18, BL'18]
OT + Garbled Circuits

[ACGJ'18]
Garbled circuits

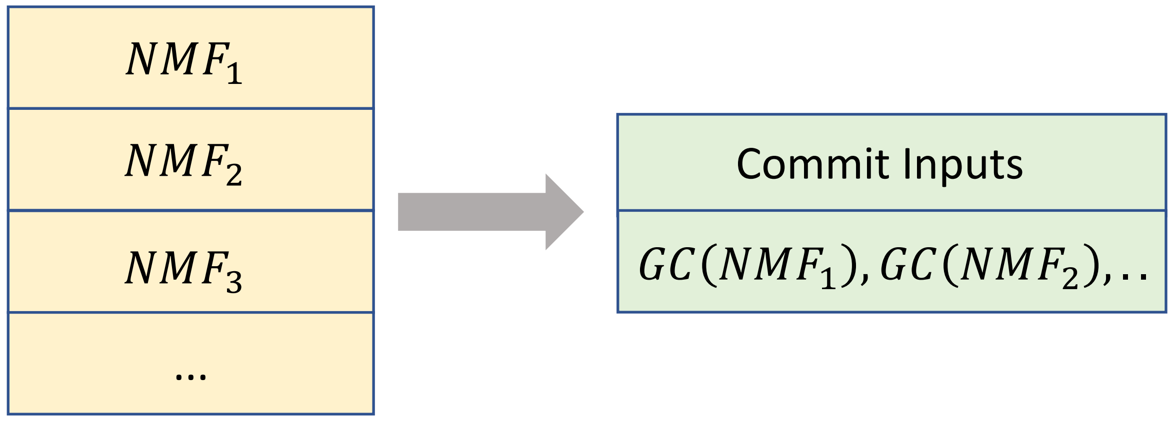
Round Compression Template



Interactive secure MPC

2 round secure MPC

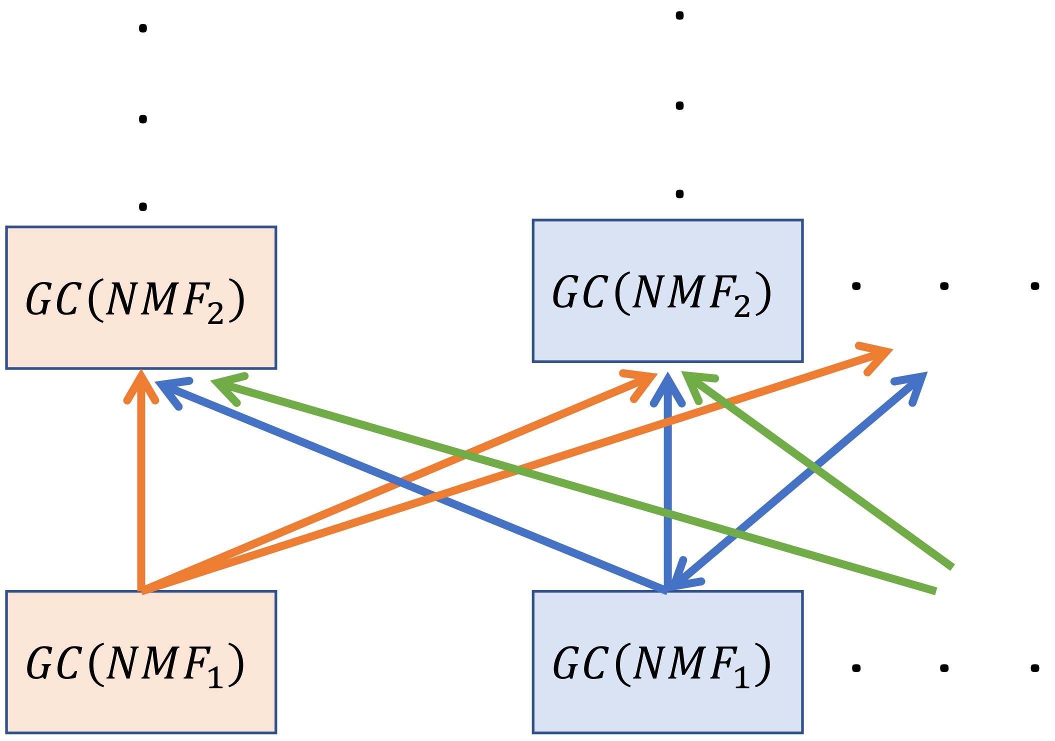
Round Compression Template



Interactive secure MPC

2 round secure MPC

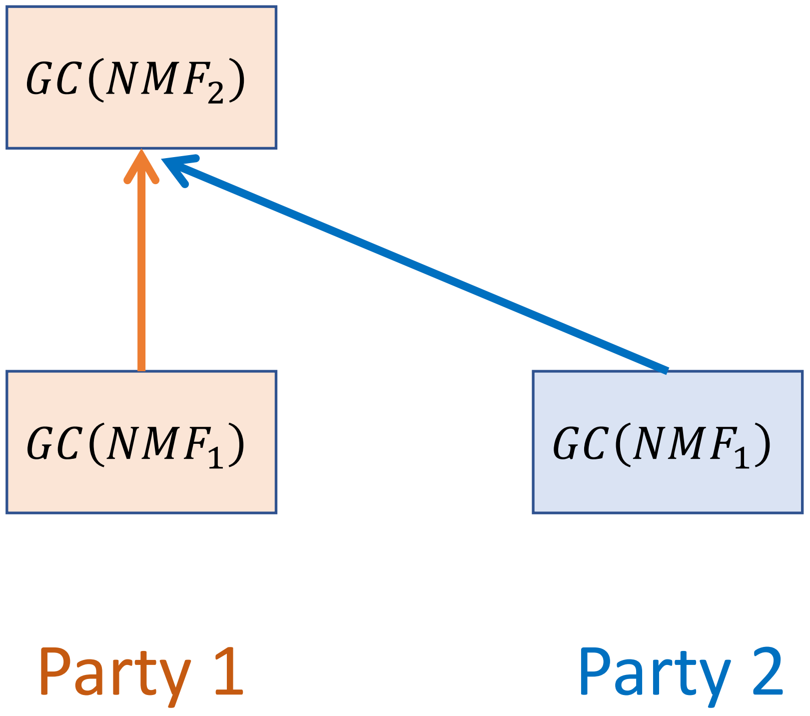
After Round 2



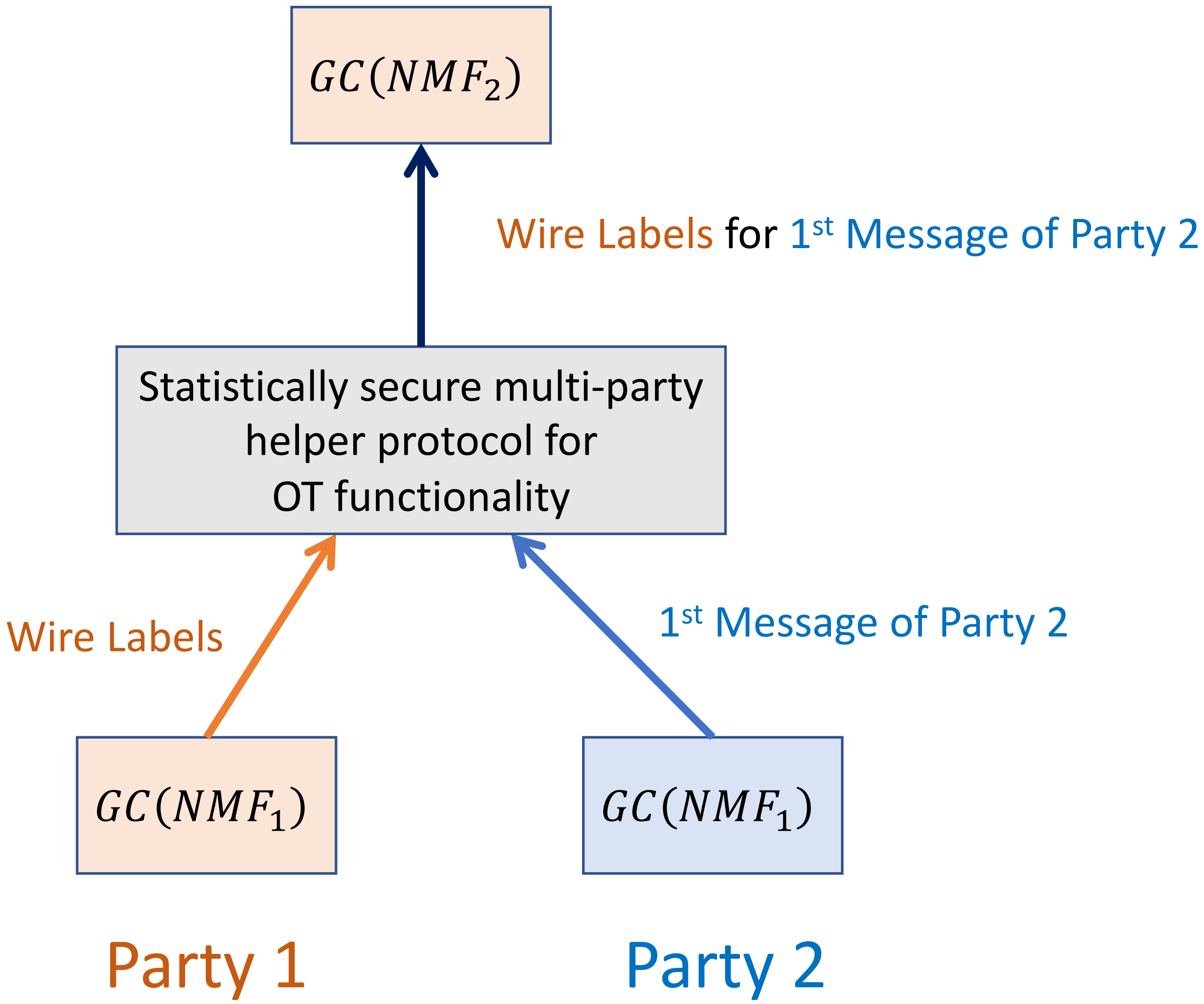
Party 1

Party 2

Round Compression Template: After Round 2



Round Compression Template: After Round 2



Initial Idea: Doesn't Work

Replace garbled circuits with
Information-theoretic garbled circuits
(IT-GC)

Problem
Size of the input wire labels in IT-GC grows exponentially in the depth of the circuit being garbled.

Interactive secure MPC

2 round secure MPC

[GGHR'13]
Indistinguishability Obfuscation

[GLS'15]
Witness Encryption + Garbled circuits

[GS'17]
Bilinear Maps + Garbled circuits

[GS'18, BL'18]
OT + Garbled Circuits

[ACGJ'18]
Garbled circuits

Initial Idea: Doesn't Work

Replace garbled circuits with
Information-theoretic garbled circuits
(IT-GC)

Problem
Size of the input wire labels in IT-GC grows exponentially in the depth of the circuit being garbled.

Interactive secure MPC

2 round secure MPC

[GGHR'13]
Indistinguishability Obfuscation

No. of garbled circuits generated per-party $\geq |C|$

[GS'17]
Bilinear Maps + Garbled circuits

[GS'18, BL'18]
OT + Garbled Circuits

[ACGJ'18]
Garbled circuits

Initial Idea: Doesn't Work

Replace garbled circuits with
Information-theoretic garbled circuits
(IT-GC)

Problem
Size of the input wire labels in IT-GC grows exponentially in the depth of the circuit being garbled.

Interactive secure MPC

2 round secure MPC

[GGHR'13]
Indistinguishability Obfuscation

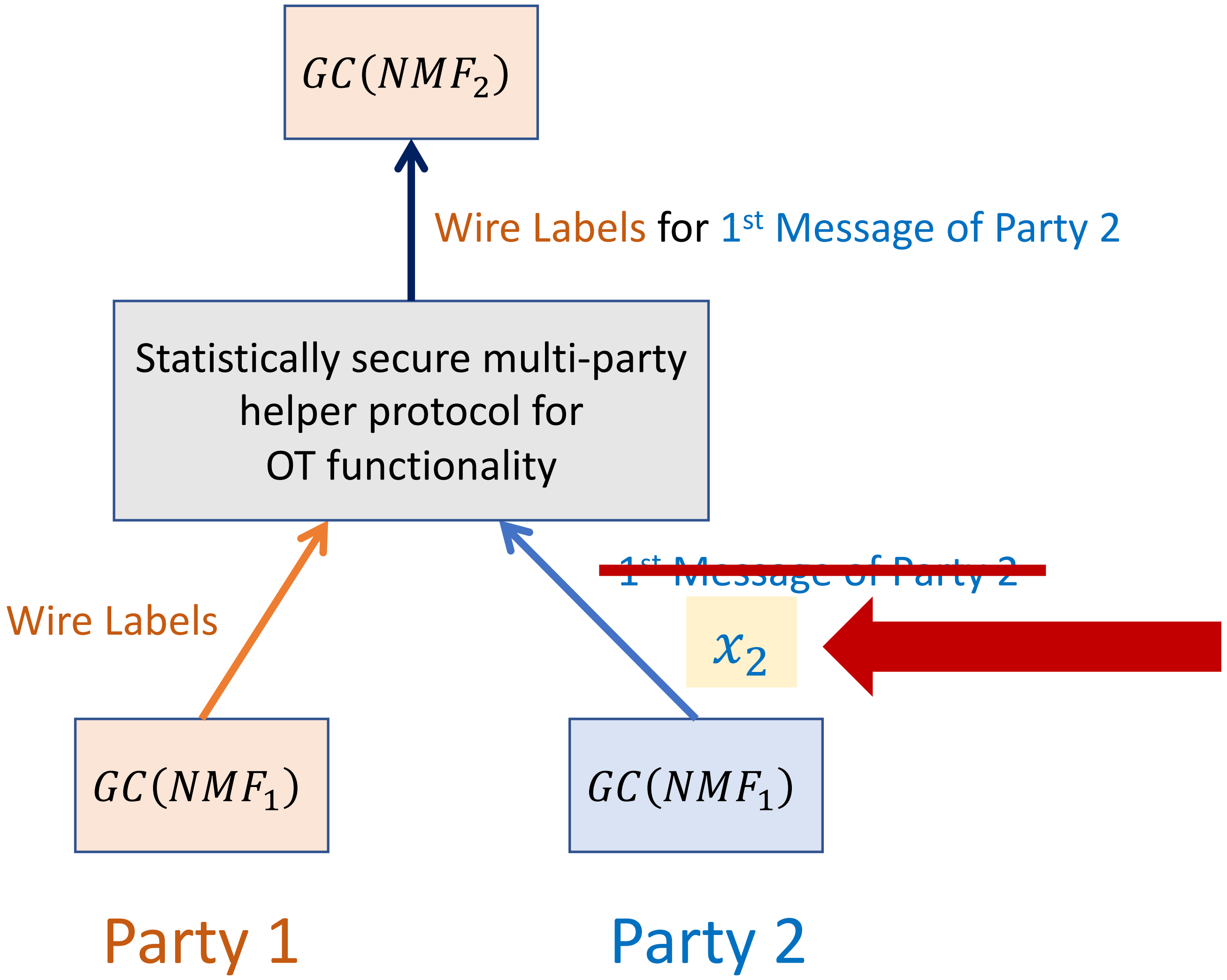
No. of garbled circuits generated per-party $\geq |C|$

Size of bottom-most garbled circuits is $\exp(|C|)$

[GS'18, BL'18]
OT + Garbled Circuits

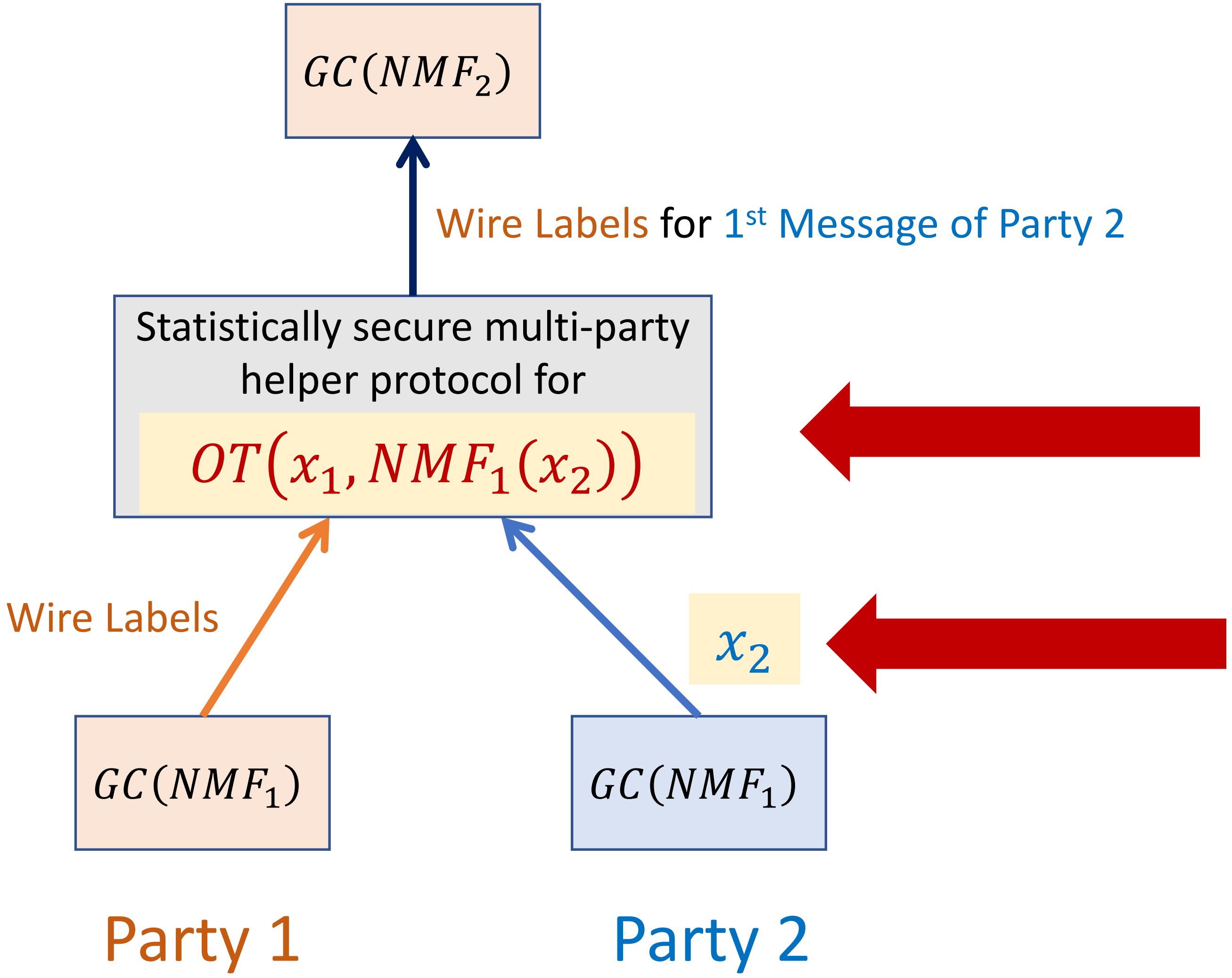
[ACGJ'18]
Garbled circuits

Our Approach



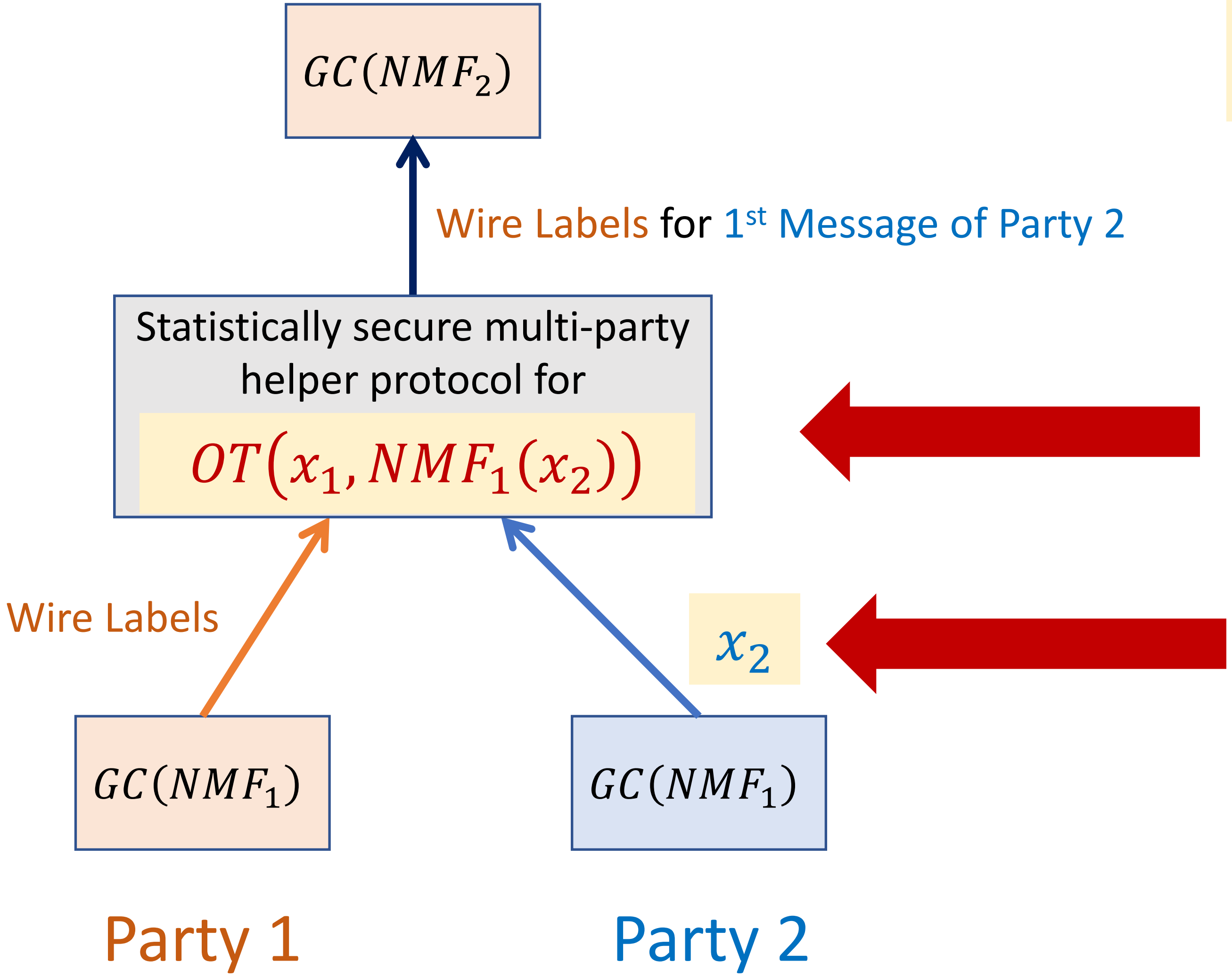
Inspired by the approach used in [BL'18]

Our Approach



Our Approach

Design a 2 round helper protocol for $OT(x_1, NMF_1(x_2))$



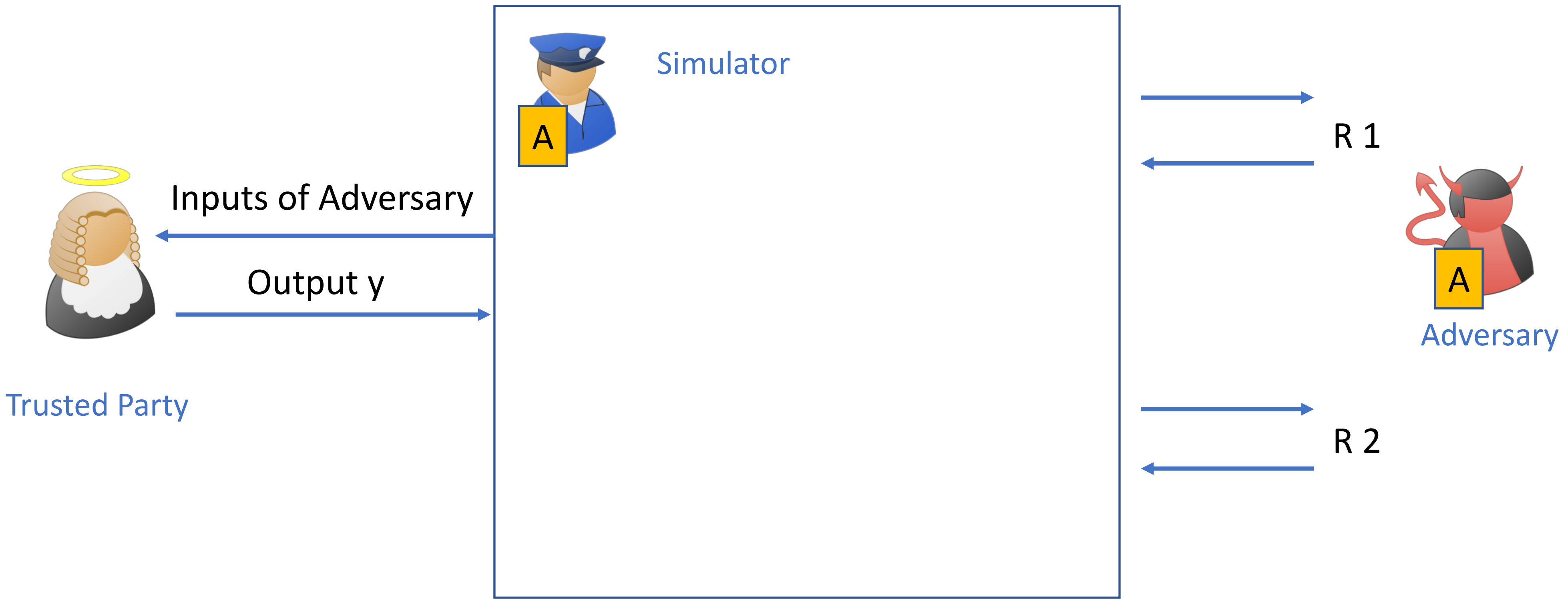
Challenges in Designing such a protocol

2 Round MPC Template using a 2 Round Helper Protocol

R 1	1st round of Helper Protocol (implicitly commits to inputs)
R 2	2nd round of Helper Protocol & $GC(NMF_1), GC(NMF_2), ..$

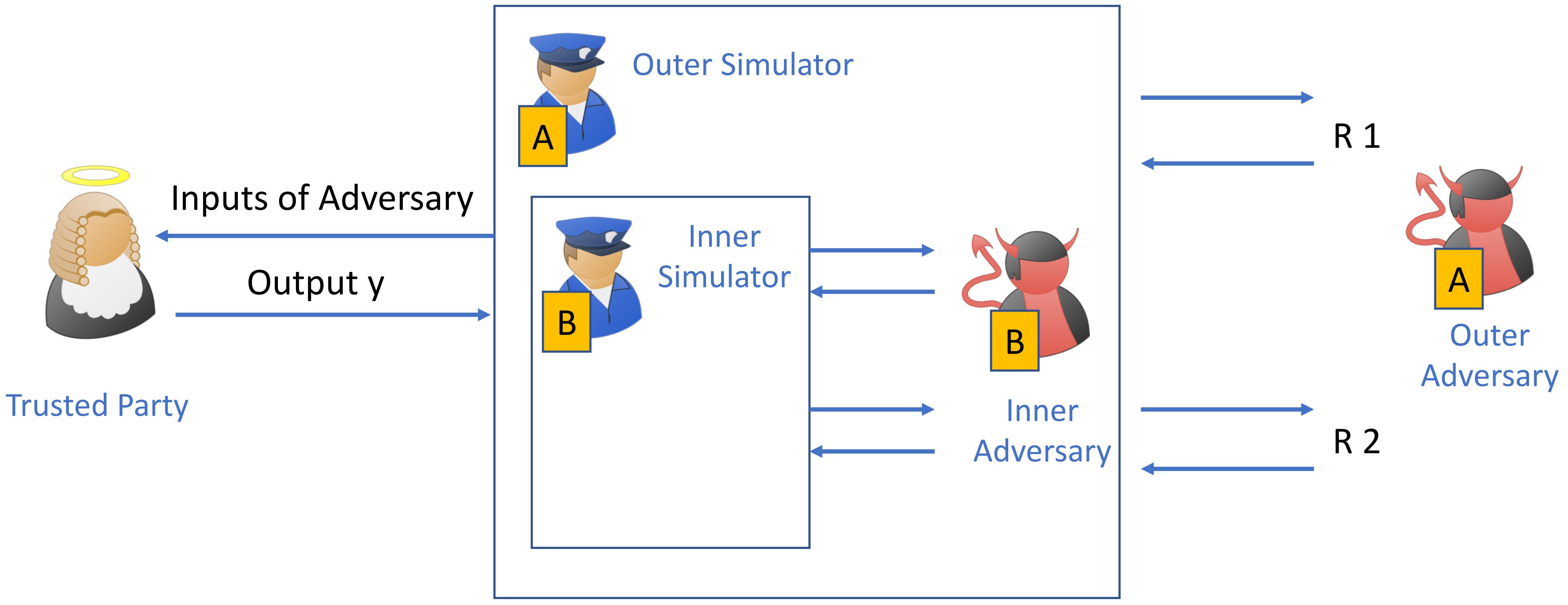
Challenges in Designing such a protocol

Malicious Security



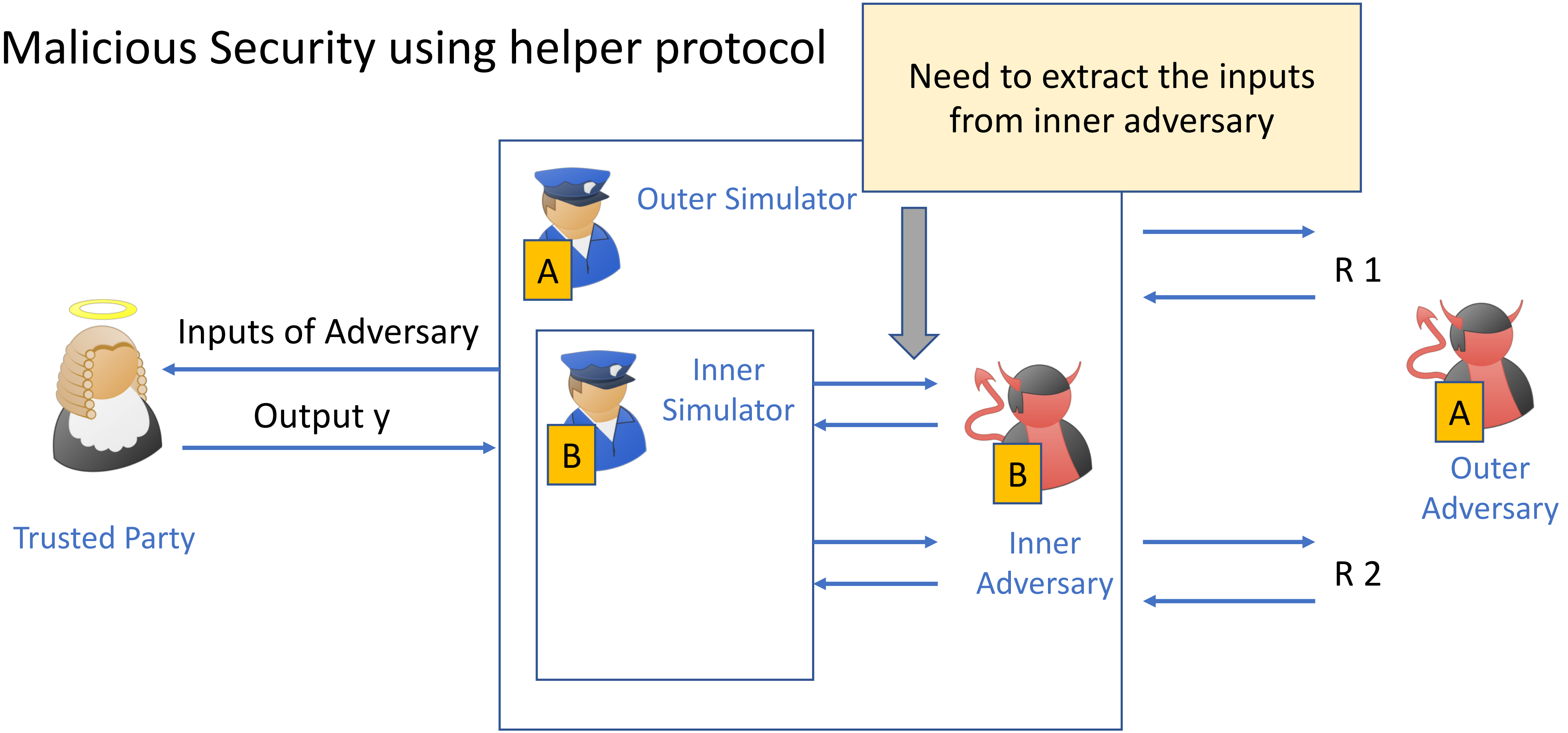
Challenges in Designing such a protocol

Malicious Security using helper protocol



Challenges in Designing such a protocol

Malicious Security using helper protocol

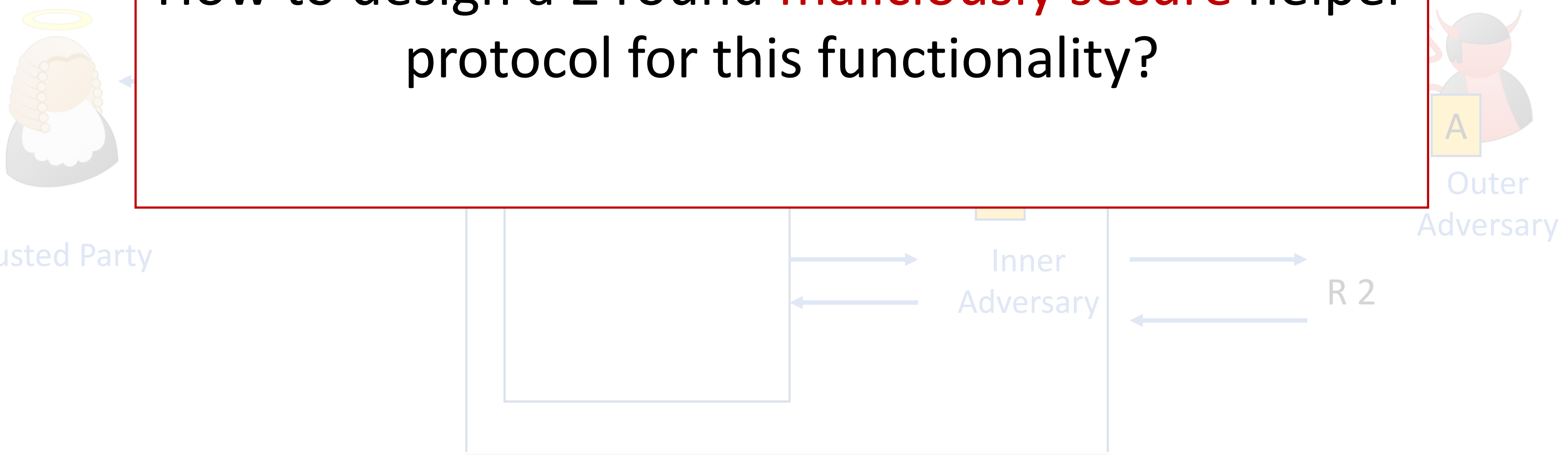


Challenges in Designing such a protocol

For Malicious Security

Need to extract the inputs from inner adversary

How to design a 2 round maliciously secure helper protocol for this functionality?



Our Solution

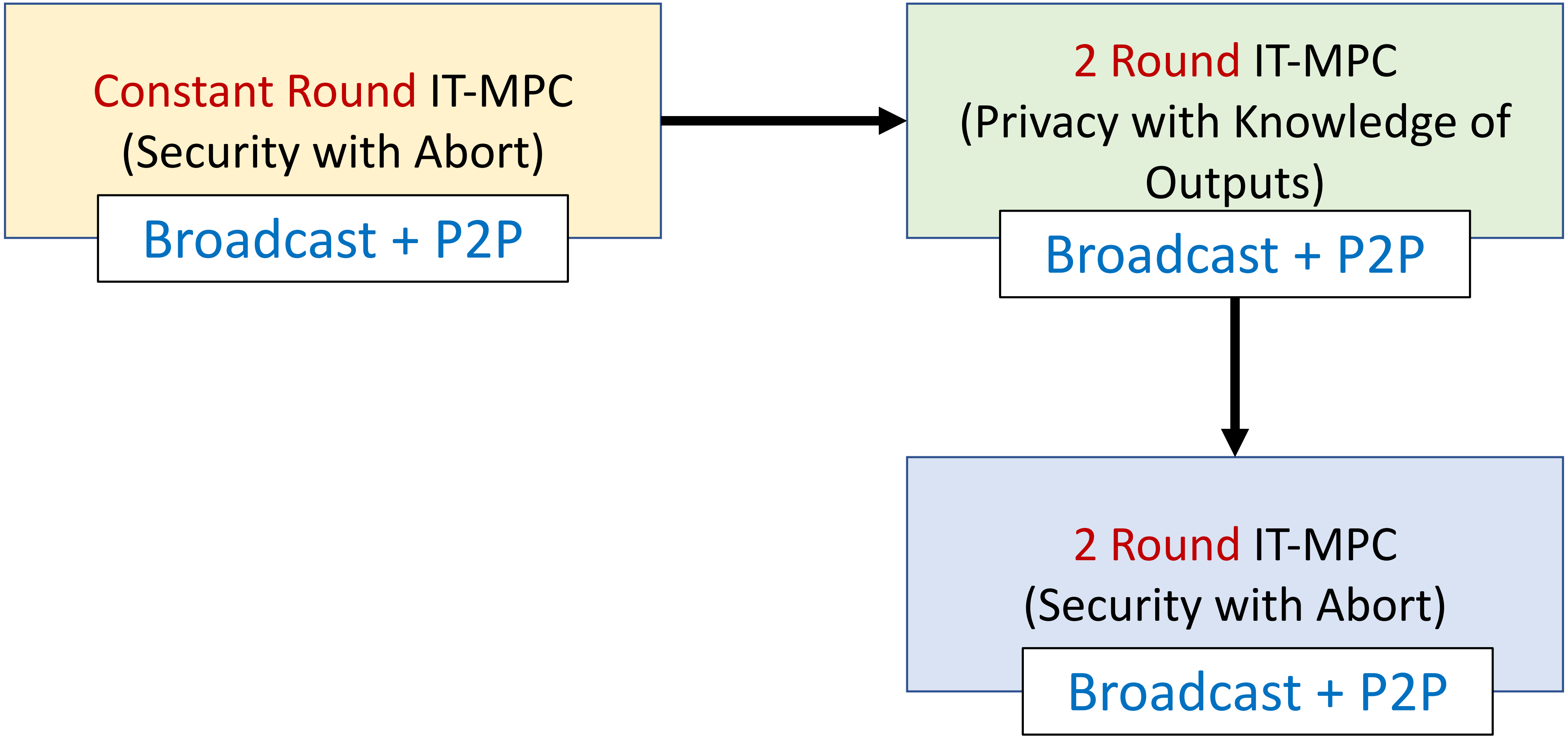
A two-round helper MPC protocol for 2 input delayed-function $OT\left(x_1, NMF_1(x_2)\right)$

NMF_2 is not known in the first round.

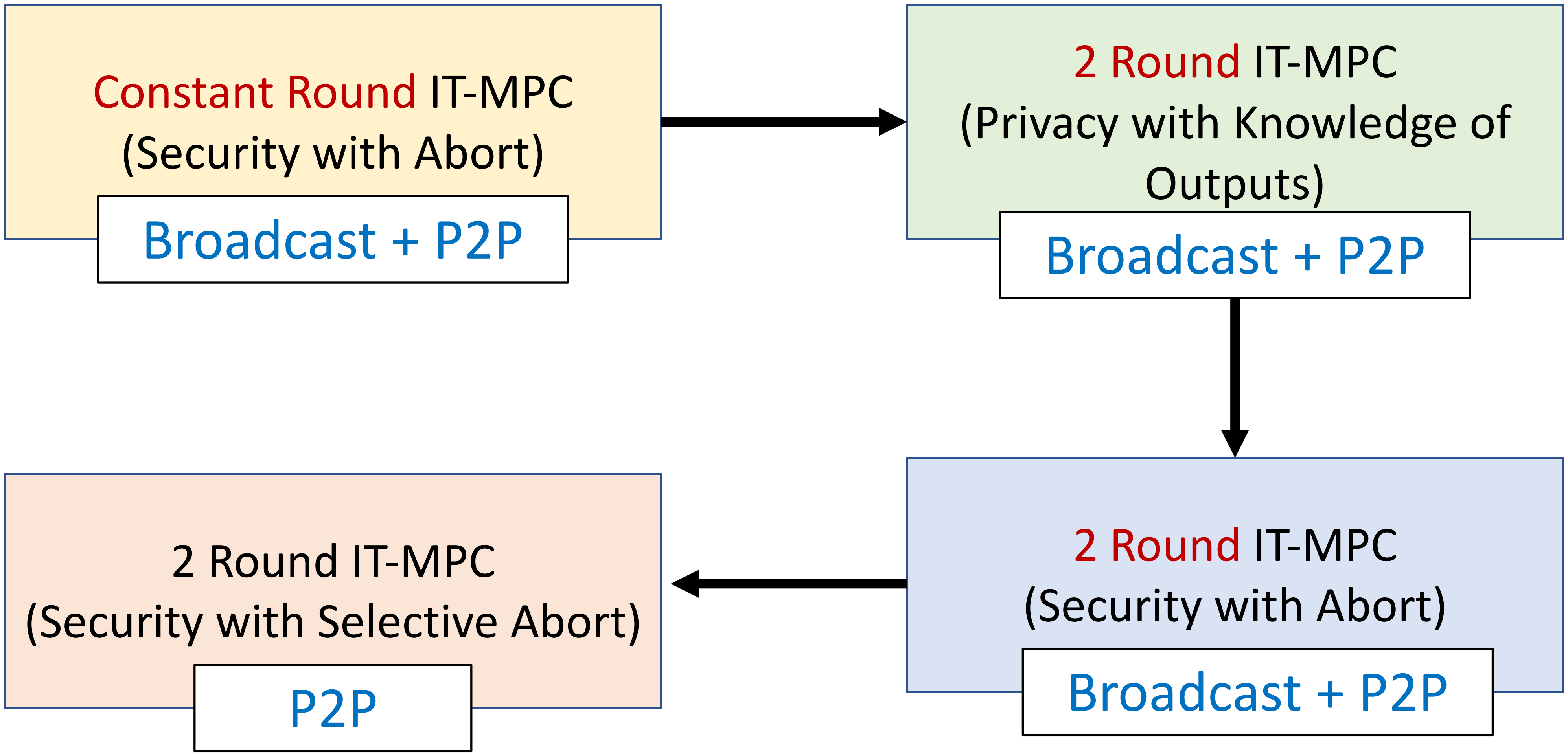
	Party 1	Party 2
HONEST	Nothing beyond the output is leaked	Nothing beyond $NMF_1(x_2)$ is leaked
CORRUPT	Simulator can extract x_1	Simulator can extract $NMF_1(x_2)$

This asymmetric weaker security suffices!

Conclusion


Constant Round IT-MPC
(Security with Abort)
Broadcast + P2P
2 Round IT-MPC
(Privacy with Knowledge of Outputs)
Broadcast + P2P
2 Round IT-MPC
(Security with Abort)
Broadcast + P2P

Conclusion


Constant Round IT-MPC
(Security with Abort)
Broadcast + P2P
2 Round IT-MPC
(Privacy with Knowledge of Outputs)
Broadcast + P2P
2 Round IT-MPC
(Security with Abort)
Broadcast + P2P
2 Round IT-MPC
(Security with Selective Abort)
P2P

Thank You!

https://eprint.iacr.org/2018/1078

aarushig@cs.jhu.edu