The Round Complexity of Secure Computation Against Covert Adversaries

Arka Rai Choudhuri
Johns Hopkins University

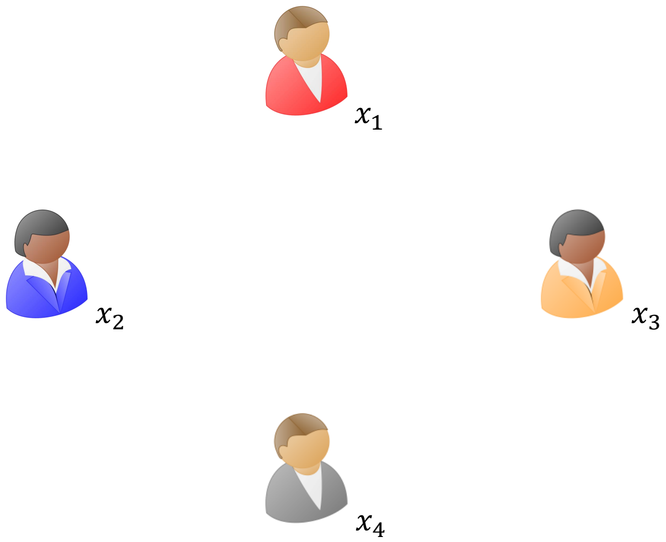
Vipul Goyal
Carnegie Mellon University

Abhishek Jain
Johns Hopkins University

SCN 2020

Multiparty Computation (MPC)

[Yao'86, Goldreich-Micali-Wigderson'87]



x_1

x_2

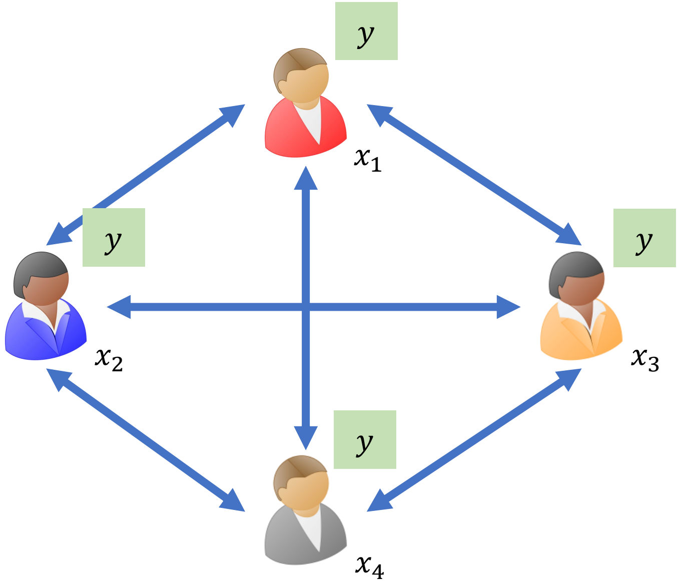
x_3

x_4

$y = f(x_1, x_2, x_3, x_4)$

Multiparty Computation (MPC)

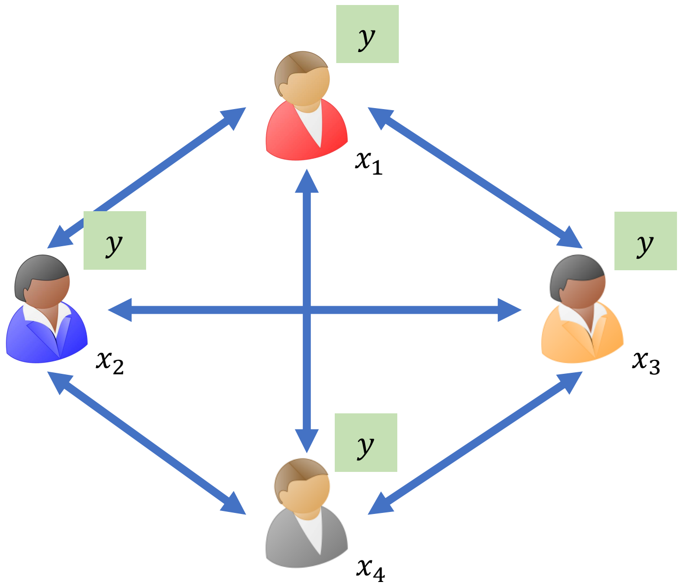
[Yao'86, Goldreich-Micali-Wigderson'87]



$y = f(x_1, x_2, x_3, x_4)$

Multiparty Computation (MPC)

[Yao'86, Goldreich-Micali-Wigderson'87]

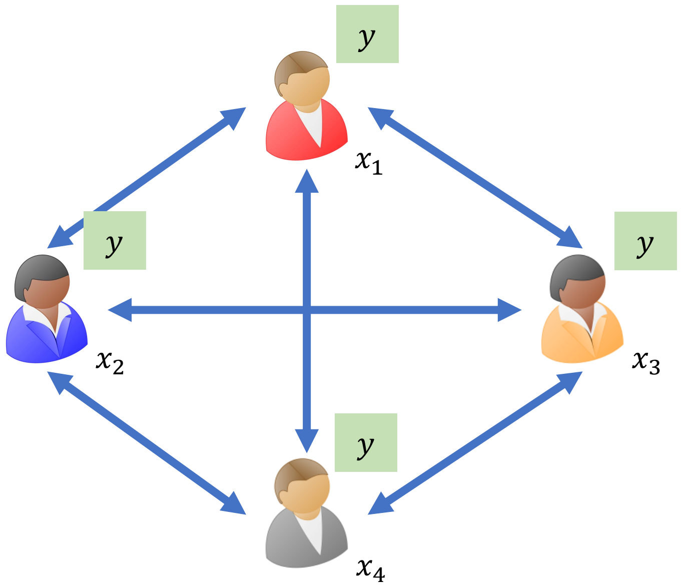


$y = f(x_1, x_2, x_3, x_4)$

A **round** constitutes of every participant sending a message.

Multiparty Computation (MPC)

[Yao'86, Goldreich-Micali-Wigderson'87]

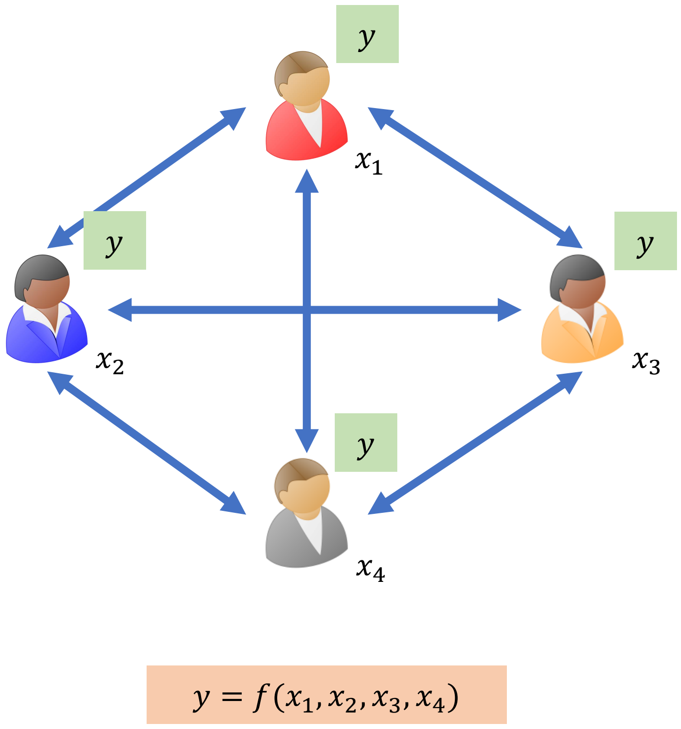


$y = f(x_1, x_2, x_3, x_4)$

A **round** constitutes of every participant sending a message.

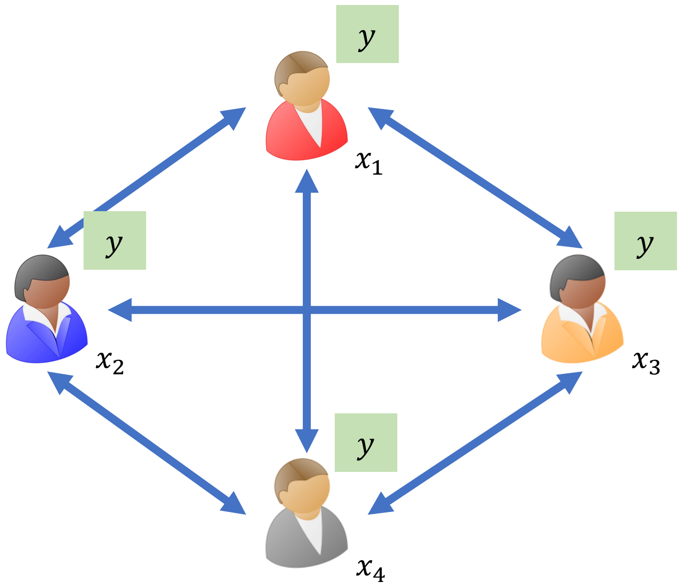
Goal: For efficiency, **minimize rounds of interaction**.

Security



Misbehaving participants should not learn anything beyond the output of the function.

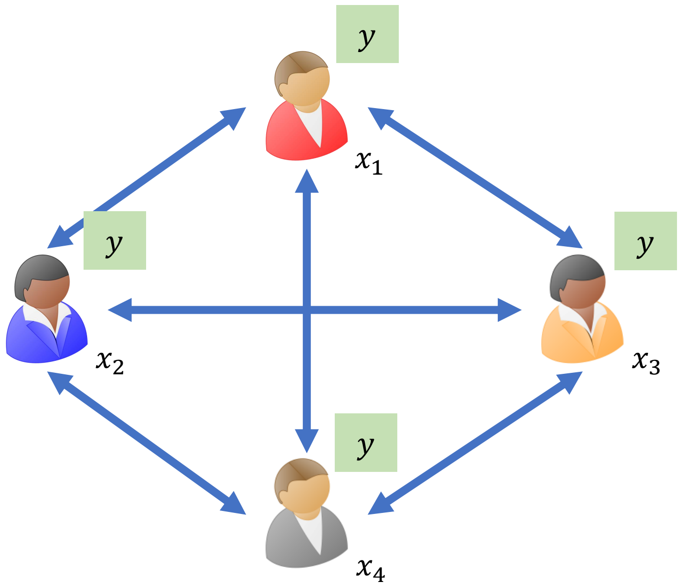
Security



$$y = f(x_1, x_2, x_3, x_4)$$

Security

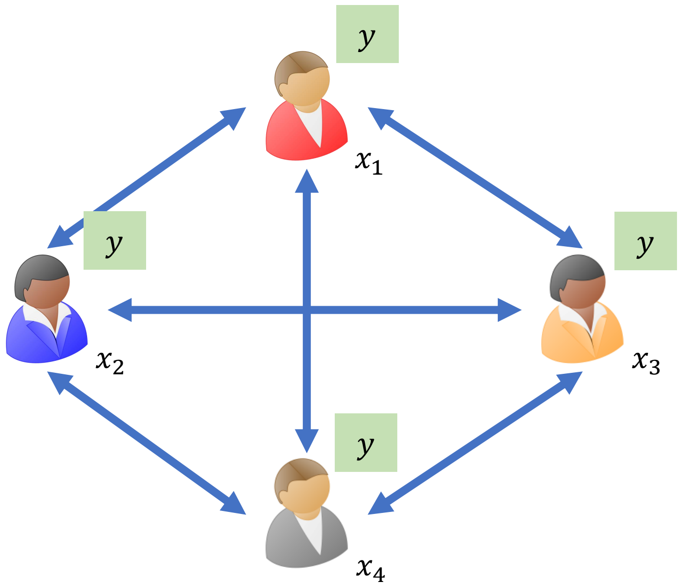
real world



$y = f(x_1, x_2, x_3, x_4)$

Security

real world

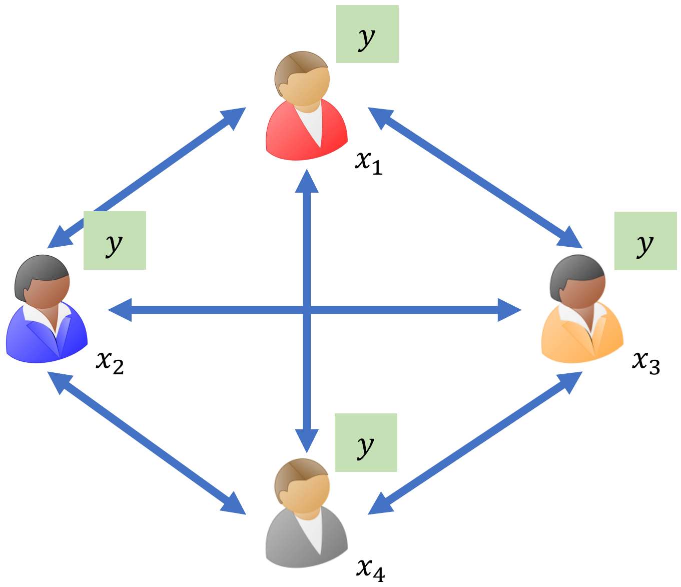


$y = f(x_1, x_2, x_3, x_4)$

ideal world

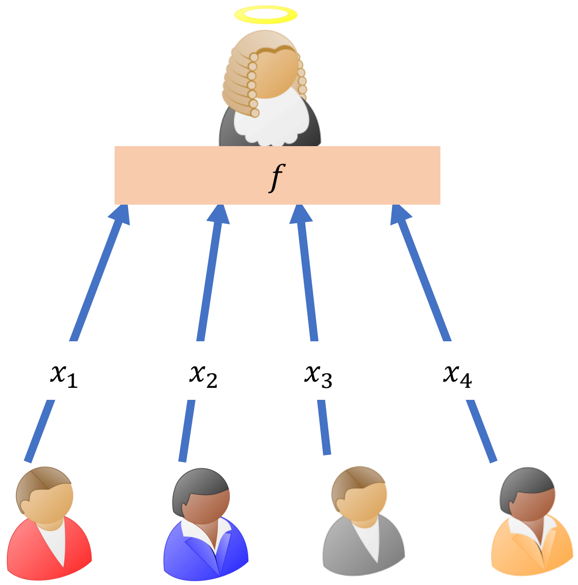
Security

real world



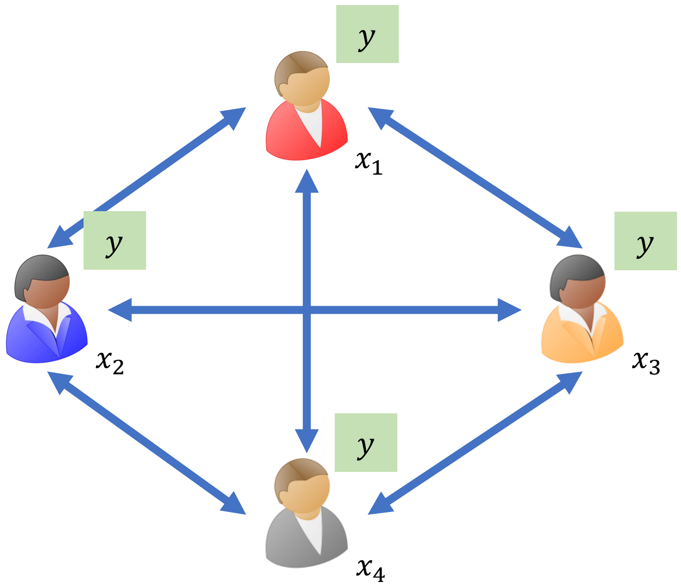
$$y = f(x_1, x_2, x_3, x_4)$$

ideal world



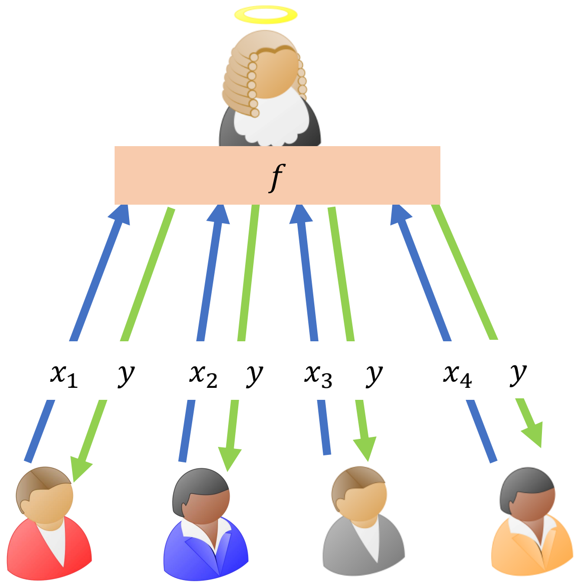
Security

real world



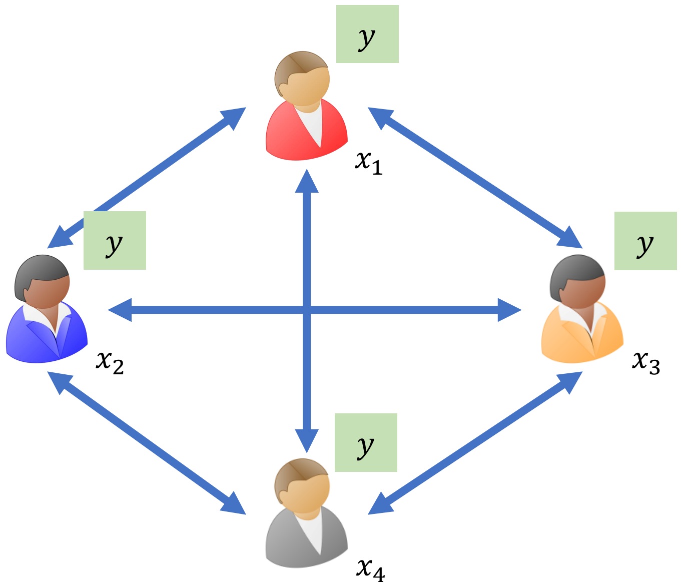
$y = f(x_1, x_2, x_3, x_4)$

ideal world



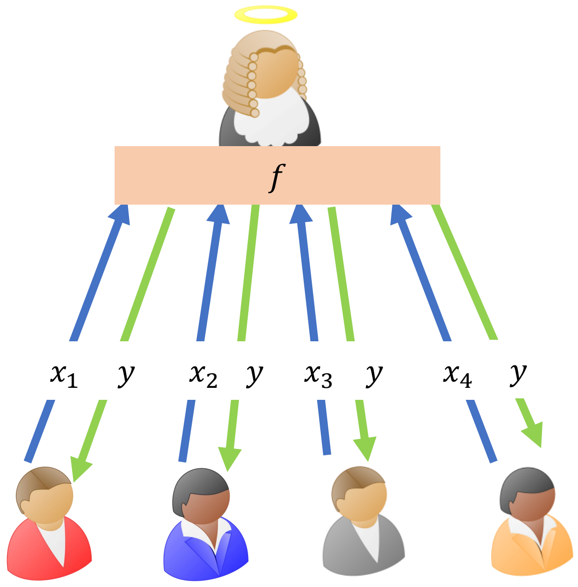
Security

real world



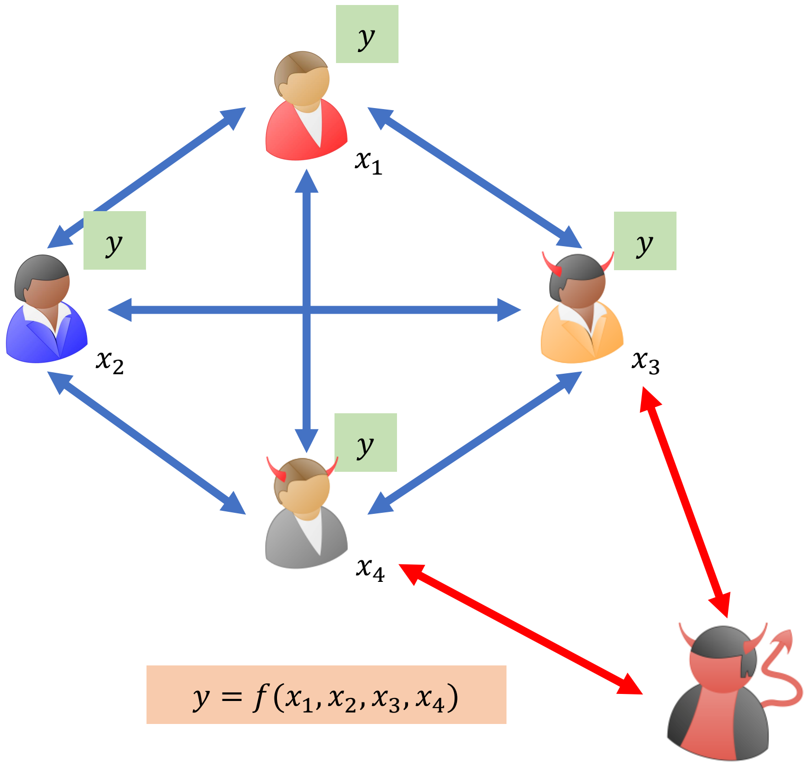
$y = f(x_1, x_2, x_3, x_4)$

ideal world

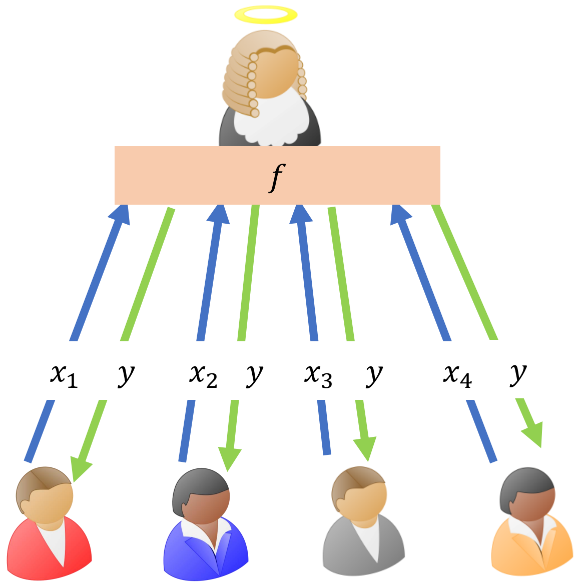


Security

real world

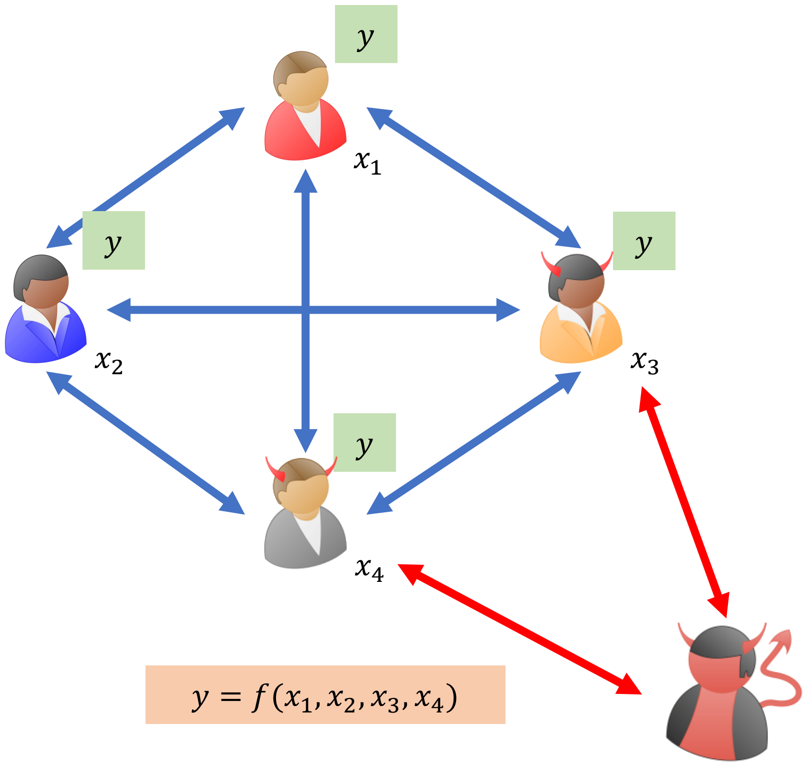


ideal world

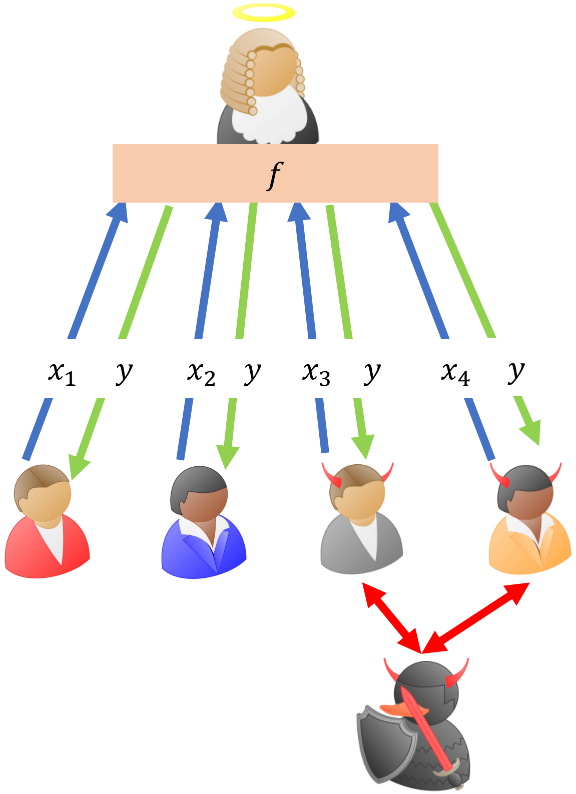


Security

real world



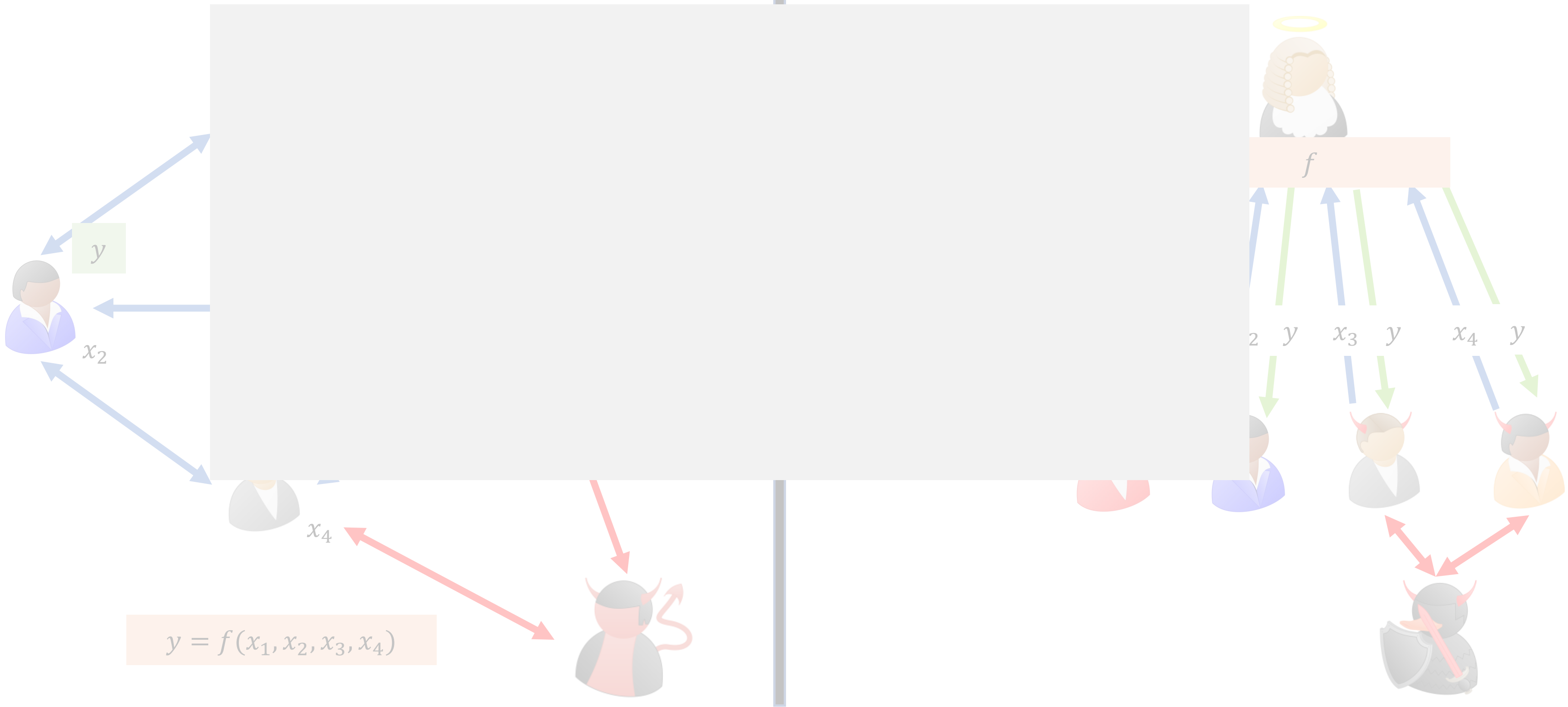
ideal world



Security

real world

ideal world



Security

real world

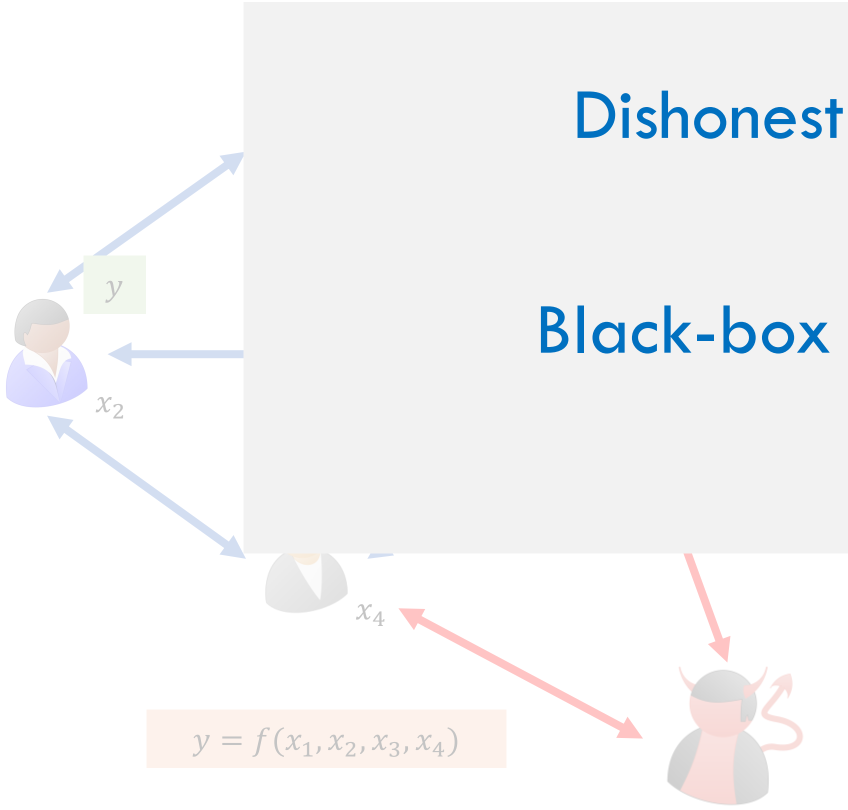
ideal world

y
x_2
x_4
$y = f(x_1, x_2, x_3, x_4)$
f
y
x_3
y
x_4
y

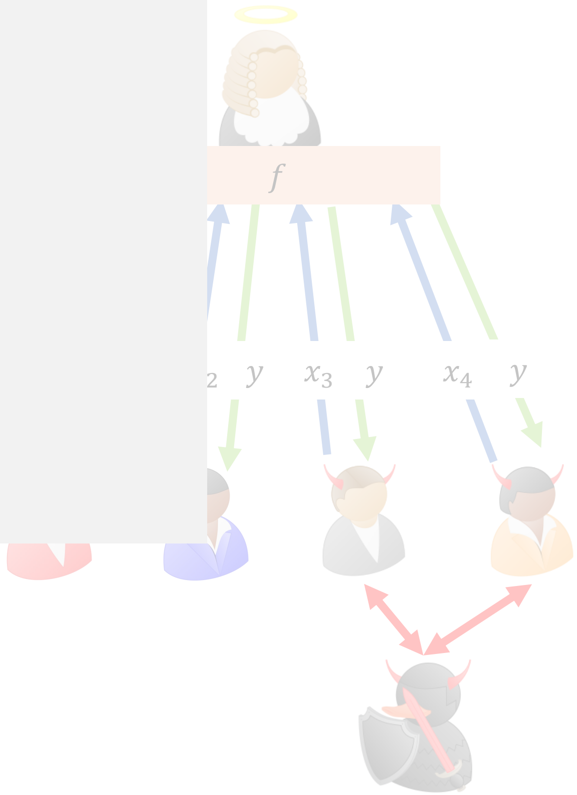
Dishonest majority.

Security

real world



ideal world



Dishonest majority.

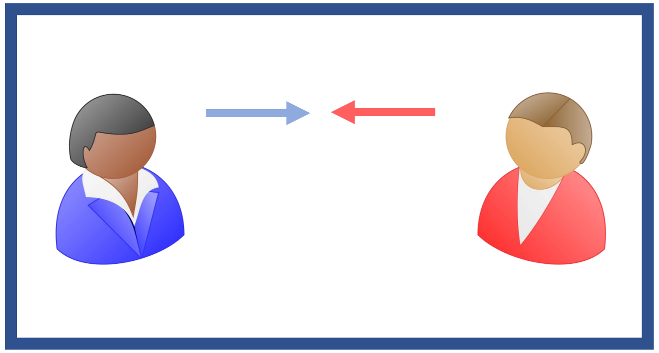
Black-box simulation.

Known Round Complexity Bounds

(Semi-Honest)

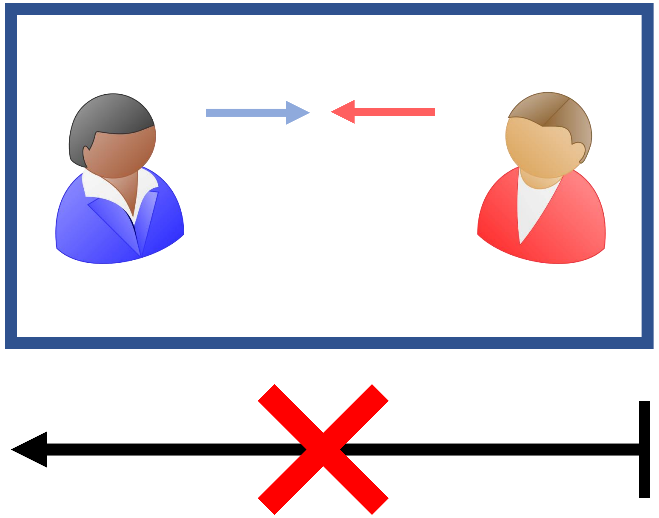
Known Round Complexity Bounds

(Semi-Honest)



Known Round Complexity Bounds

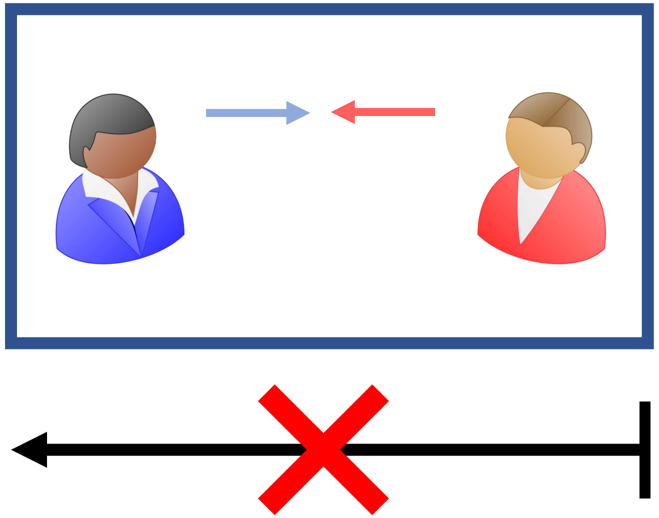
(Semi-Honest)



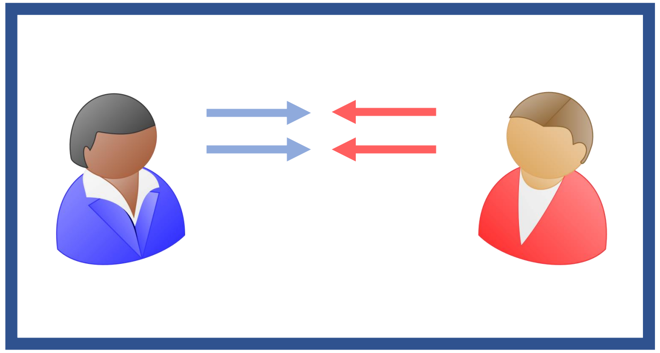
Impossible

Known Round Complexity Bounds

(Semi-Honest)

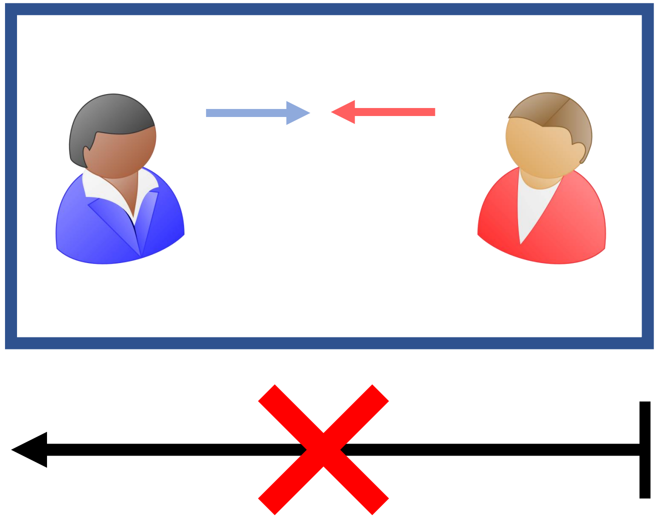


Impossible

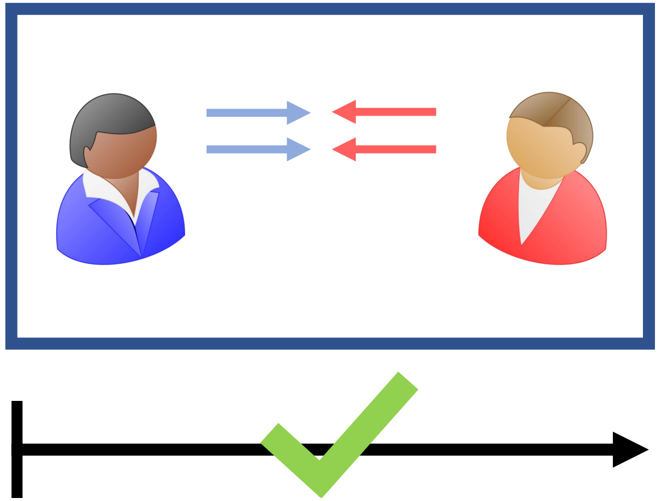


Known Round Complexity Bounds

(Semi-Honest)



Impossible



Possible

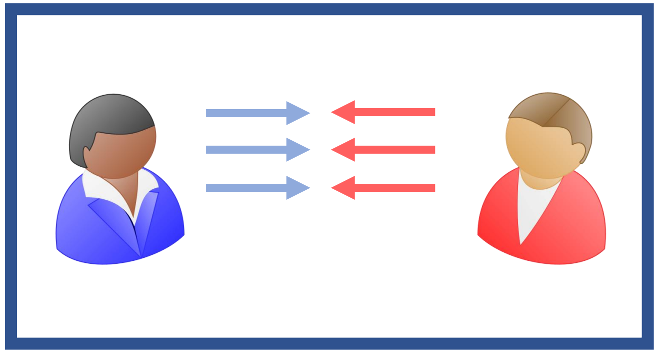
[Garg-Gentry-Halevi-Raykova'14]
[Mukherjee-Wichs'16]
[Garg-Srinivasan'17]
[Garg-Srinivasan'18]
[Benhamouda-Lin'18]

Known Round Complexity Bounds

(Malicious)

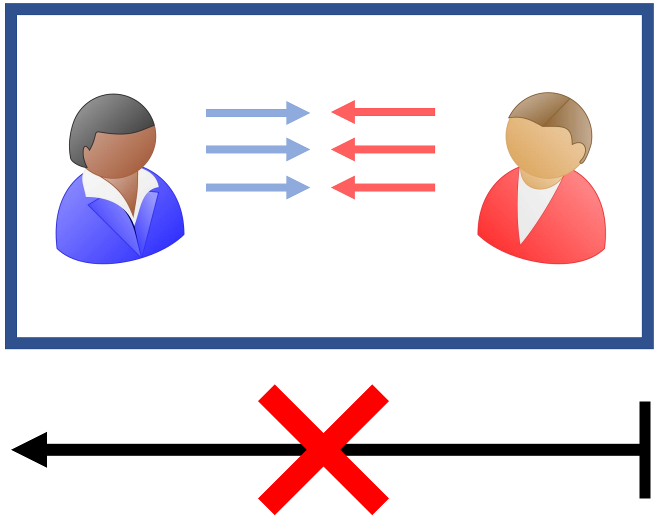
Known Round Complexity Bounds

(Malicious)



Known Round Complexity Bounds

(Malicious)

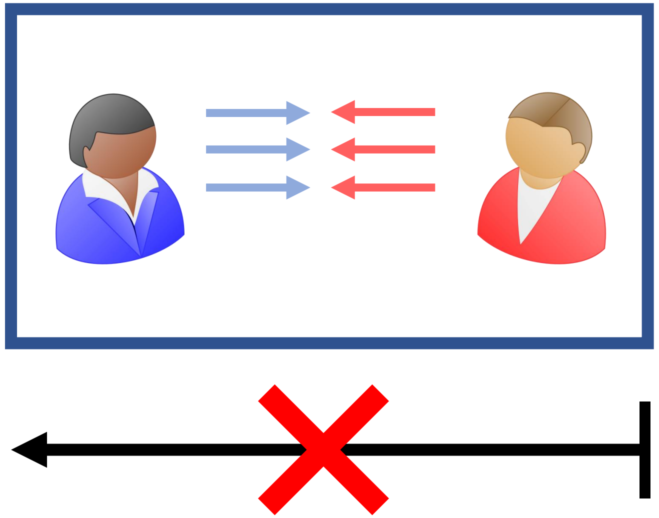


Impossible

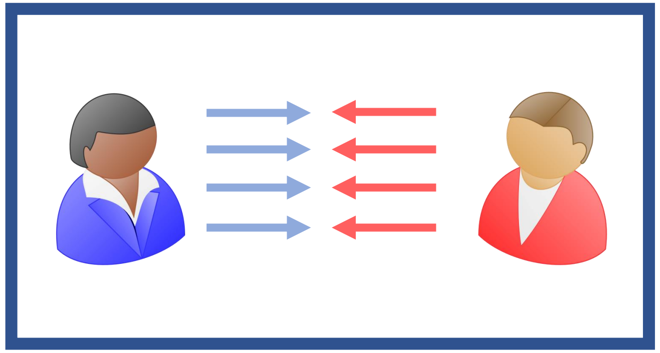
[Garg-Mukherjee-Pandey-Polychroniadou’16]

Known Round Complexity Bounds

(Malicious)

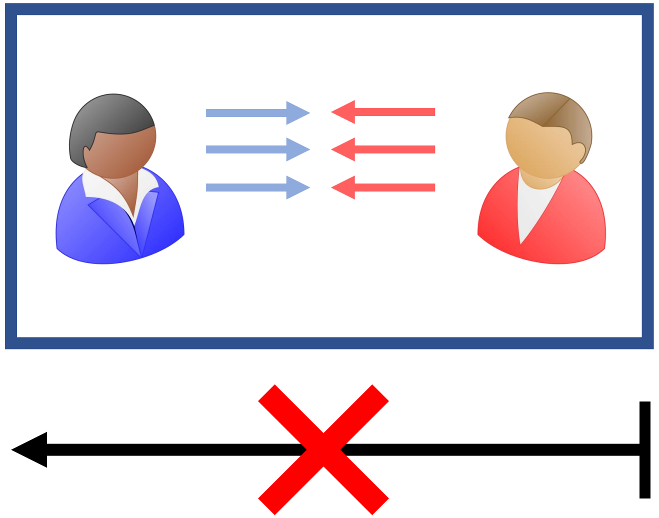


Impossible



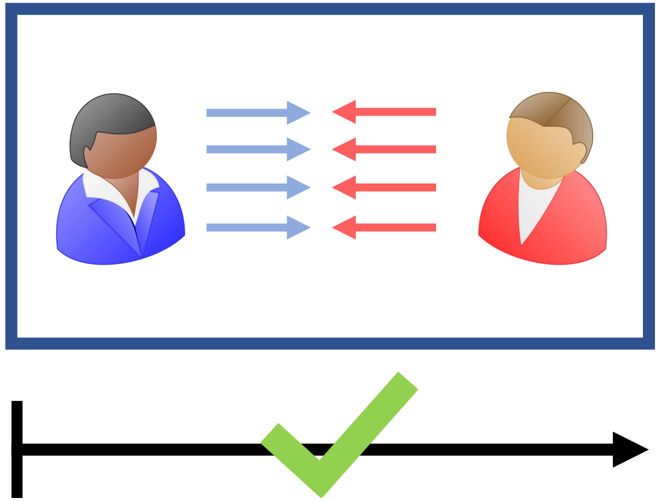
[Garg-Mukherjee-Pandey-Polychroniadou'16]

Known Round Complexity Bounds
(Malicious)



Impossible

[Garg-Mukherjee-Pandey-Polychroniadou'16]



Possible

[Ananth-**C**-Jain'17]
[Brakerski-Halevi-Polychroniadou'17]
[Badrinarayanan-Goyal-Jain-Kalai-Khurana-Sahai'18]
[Halevi-Hazay-Polychroniadou-Venkitasubramaniam18]
[**C**-Ciampi-Goyal-Jain-Ostrovsky'19]

Can we design a protocol that spans the spectrum of round complexity with an intermediate security notion?

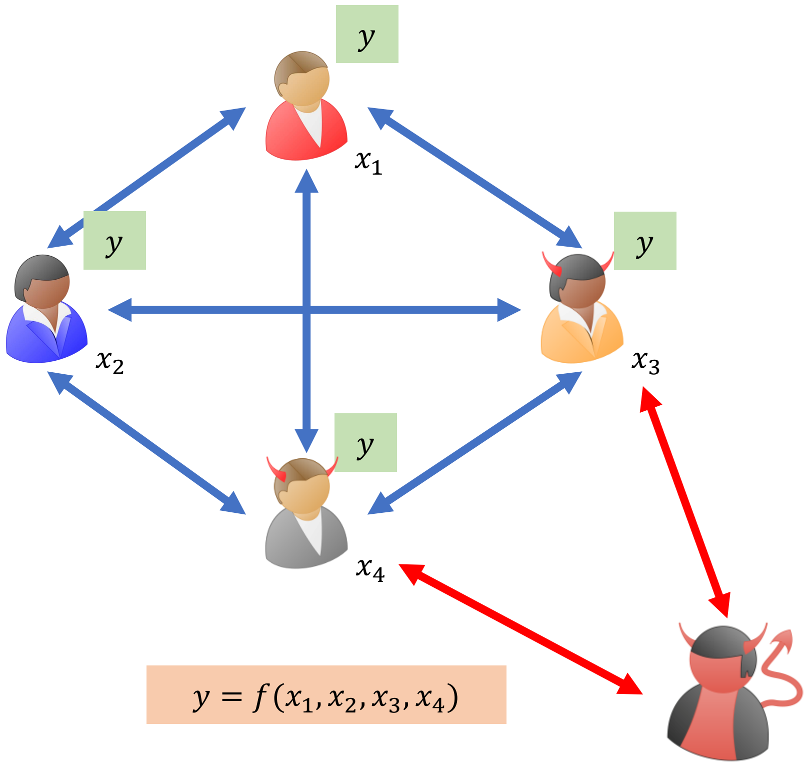
Can we design a protocol that spans the spectrum of round complexity with an intermediate security notion?

Covert Adversaries: Behaves maliciously if it can do so undetected.

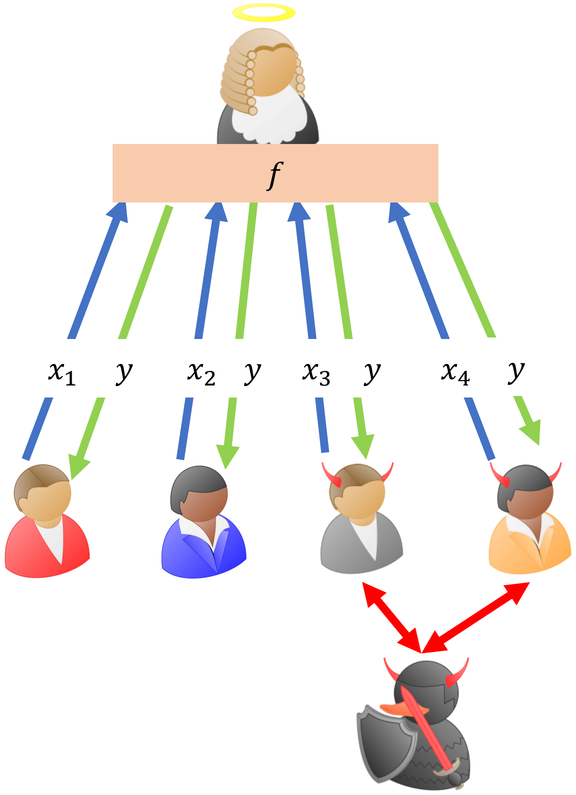
Covert Adversaries

[Aumann-Lindell'07]

real world



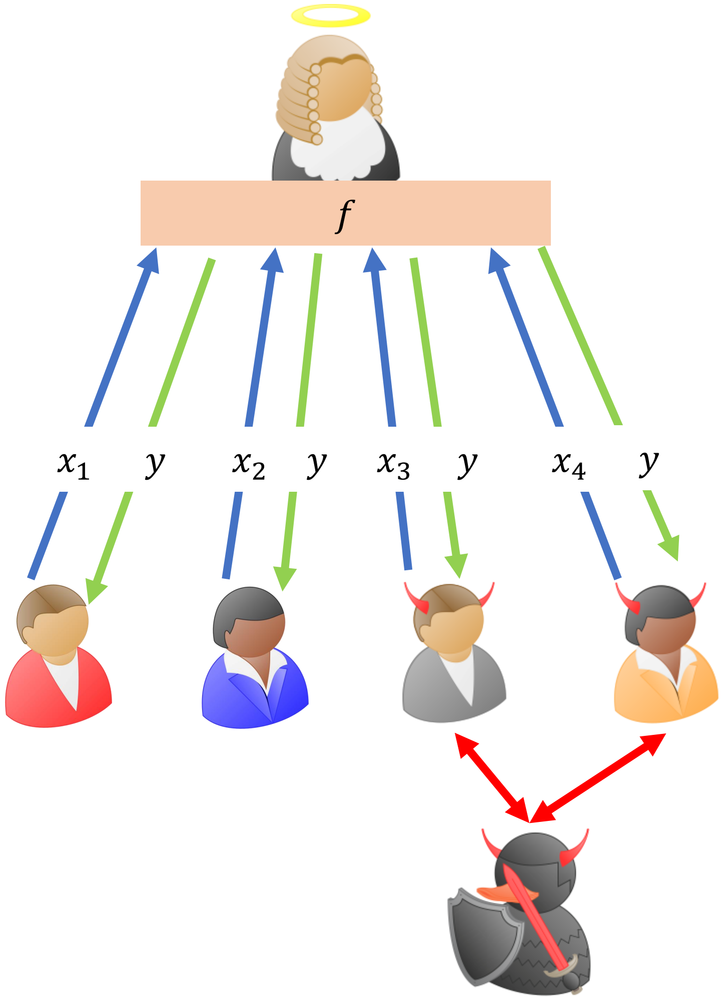
ideal world



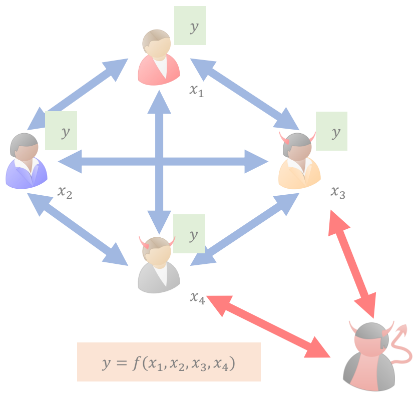
Covert Adversaries

[Aumann-Lindell'07]

ideal world



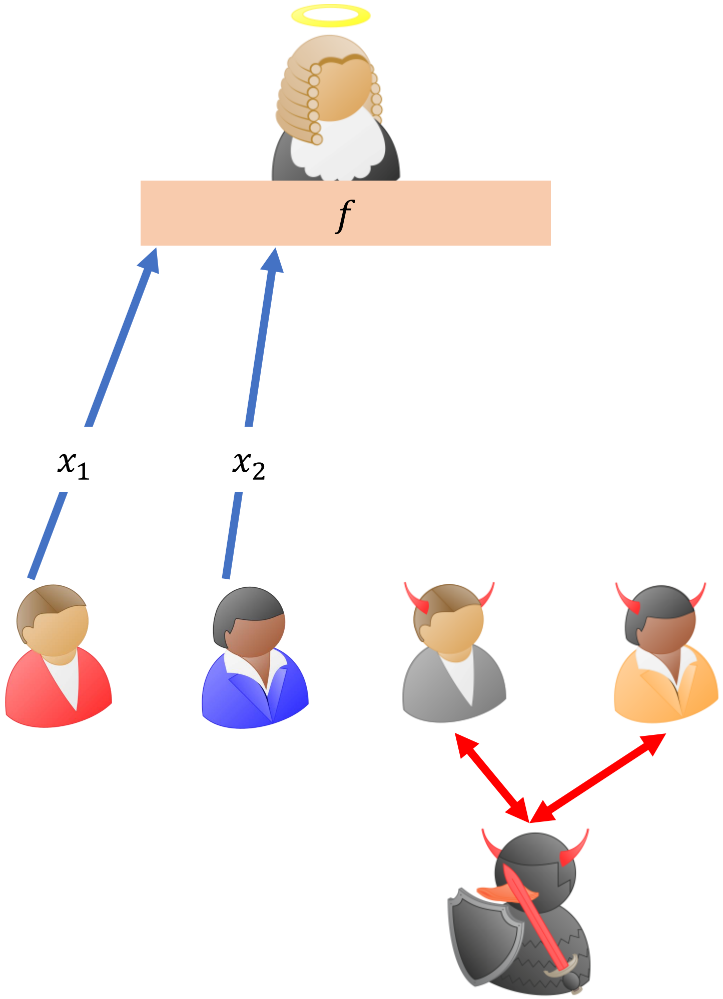
real world



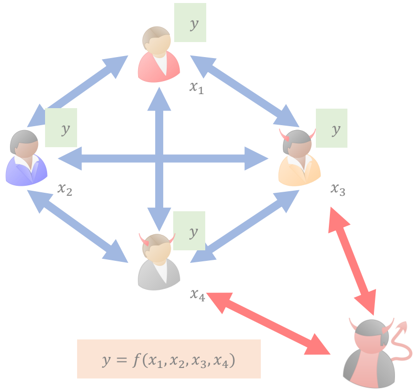
Covert Adversaries

[Aumann-Lindell'07]

ideal world



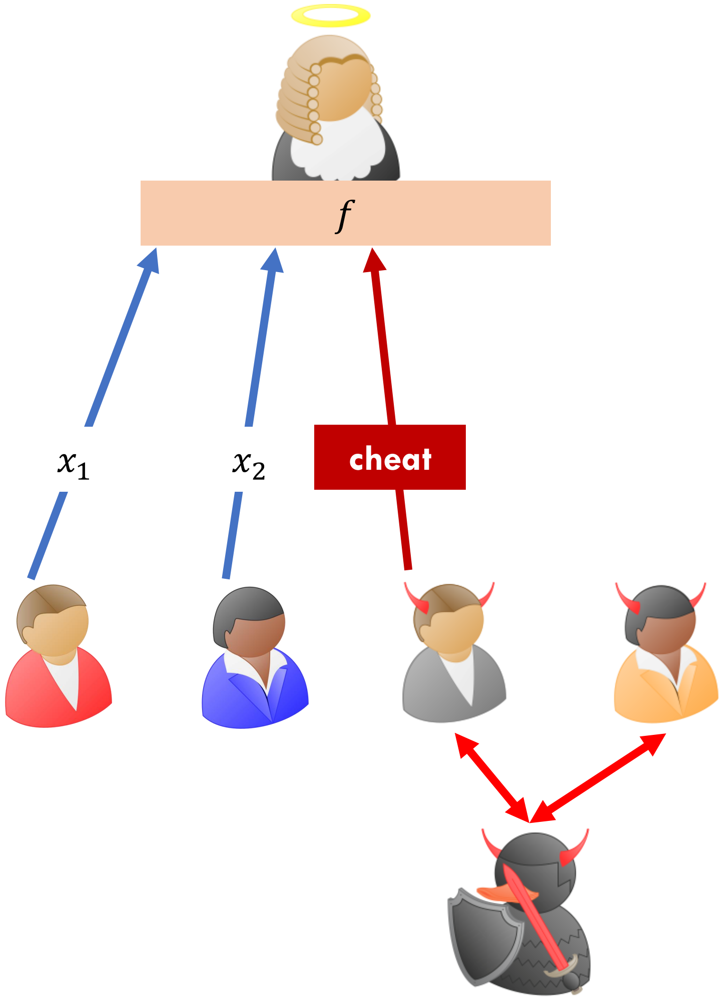
real world



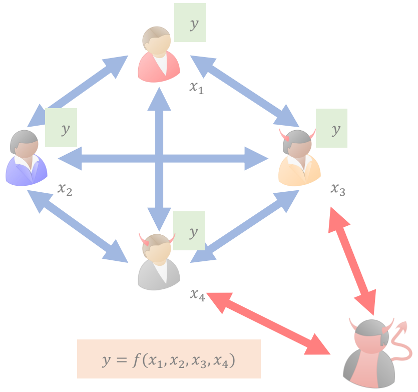
Covert Adversaries

[Aumann-Lindell'07]

ideal world



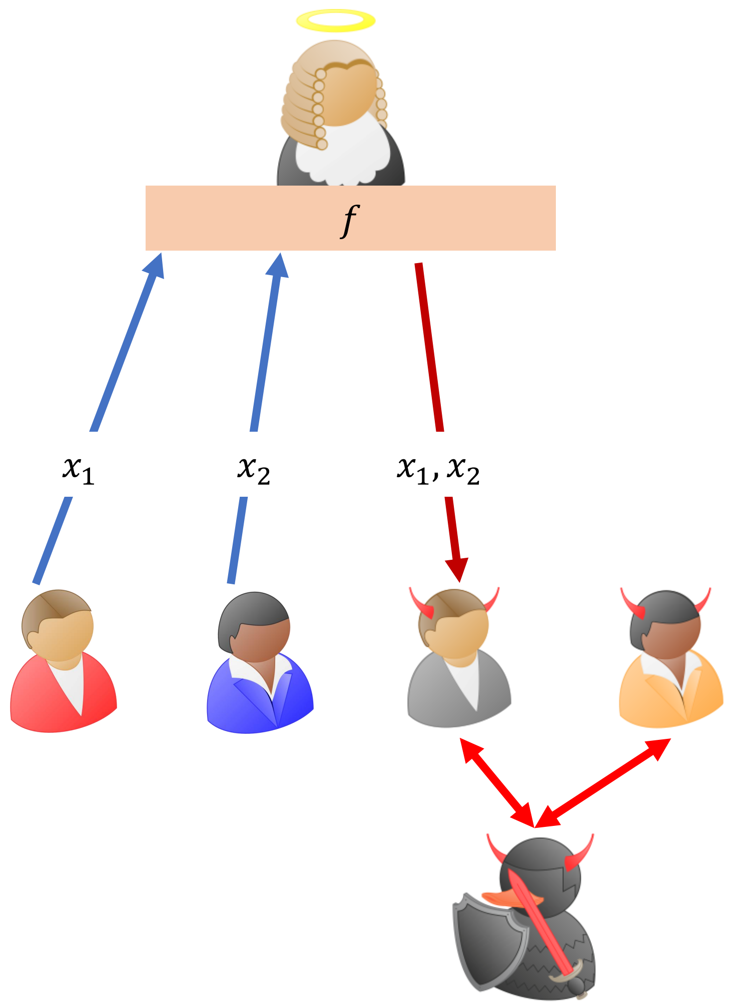
real world



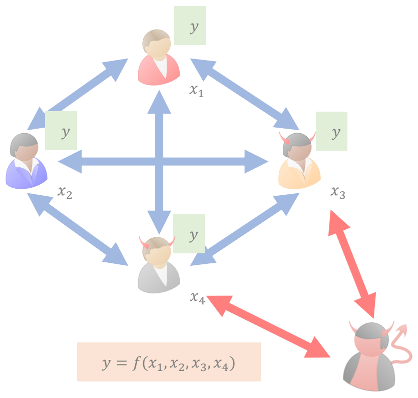
Covert Adversaries

[Aumann-Lindell'07]

ideal world



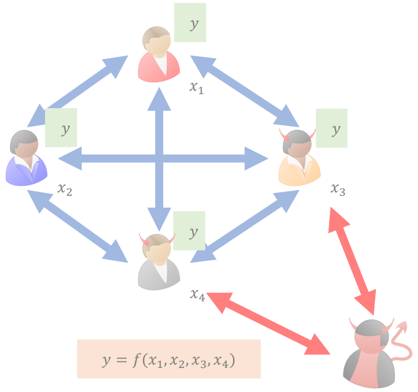
real world



Covert Adversaries

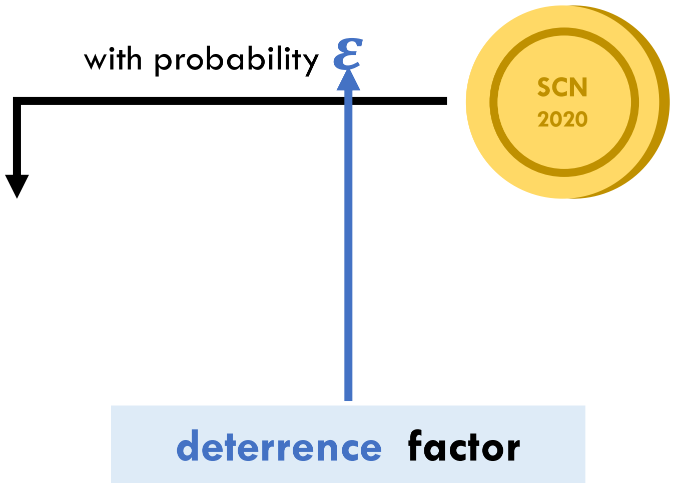
[Aumann-Lindell’07]

real world



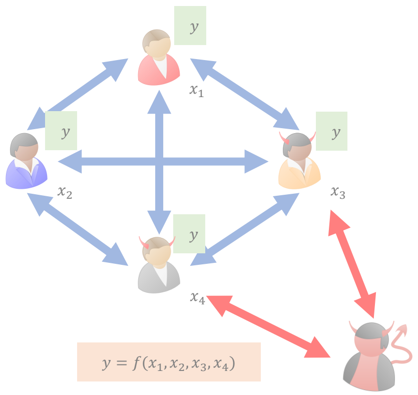
Covert Adversaries

[Aumann-Lindell’07]


with probability ε

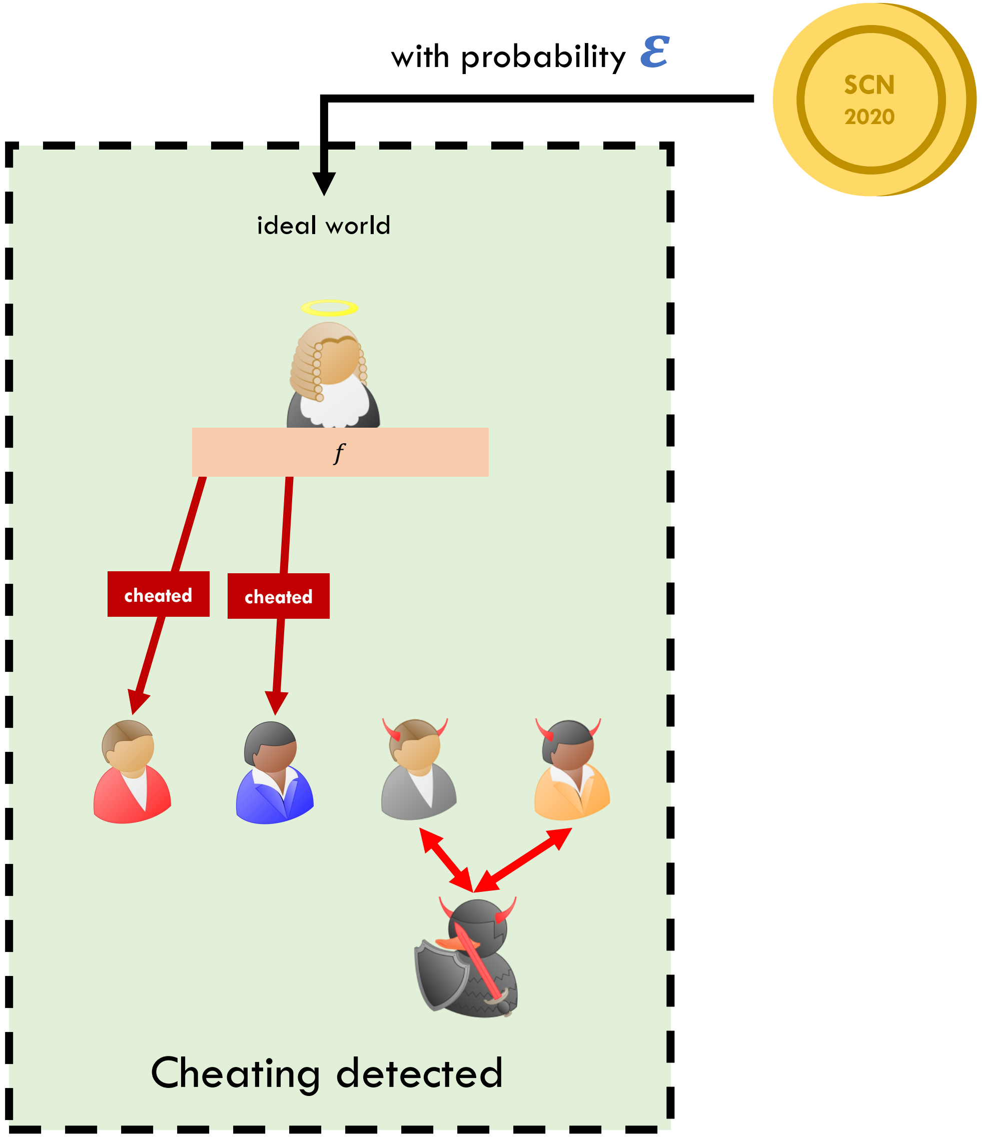
deterrence factor

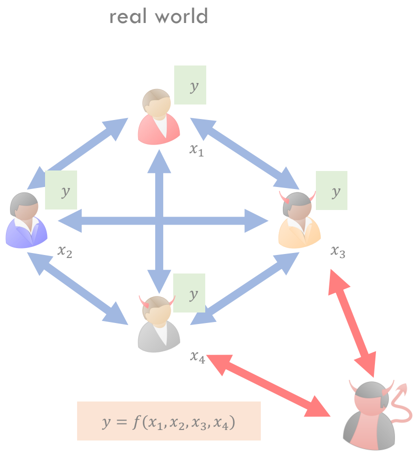
real world



Covert Adversaries

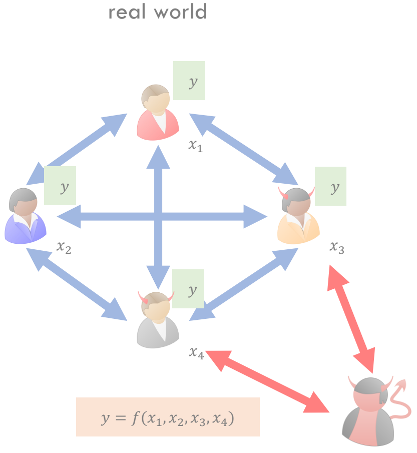
[Aumann-Lindell’07]

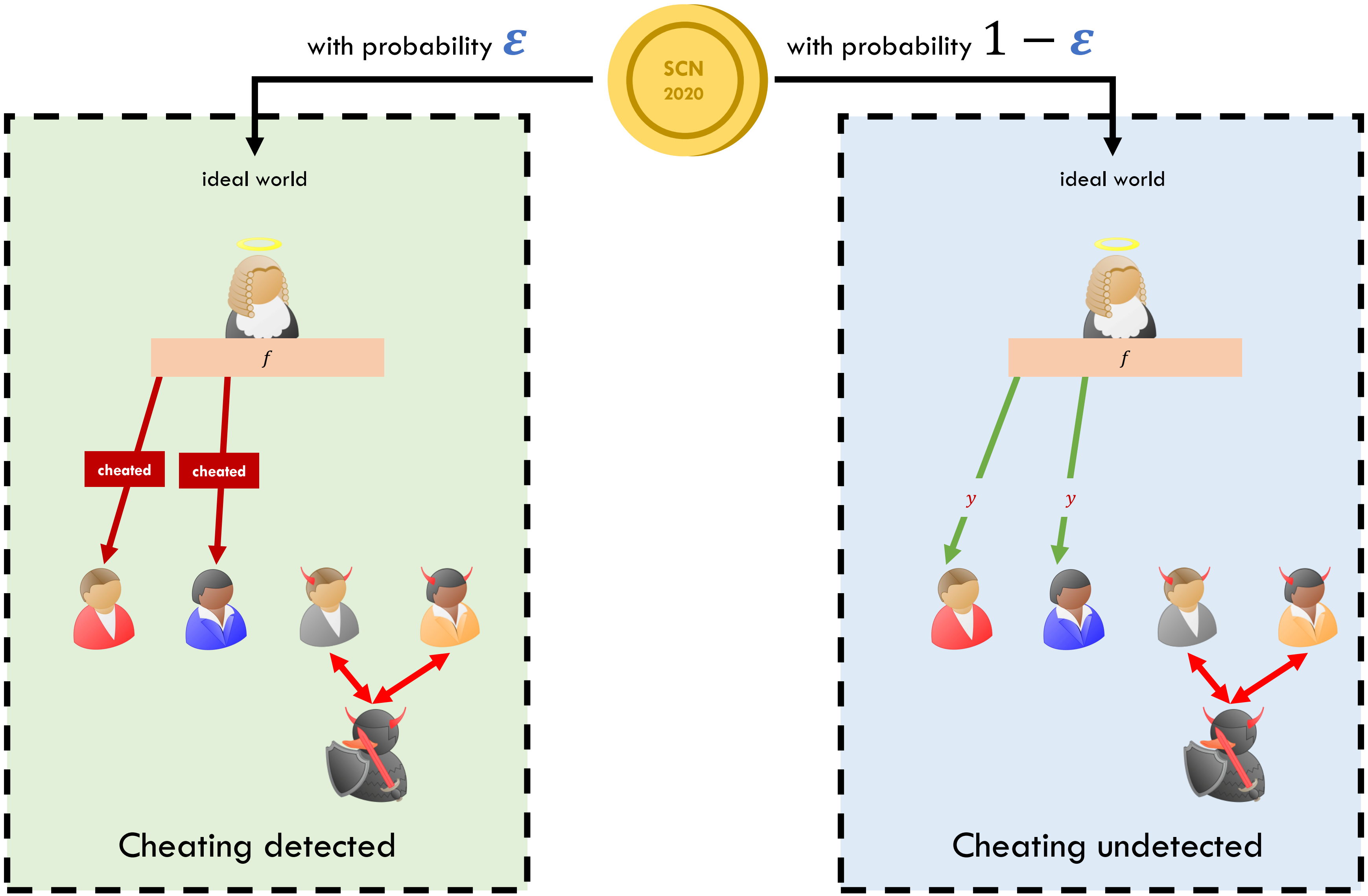




Covert Adversaries

[Aumann-Lindell'07]



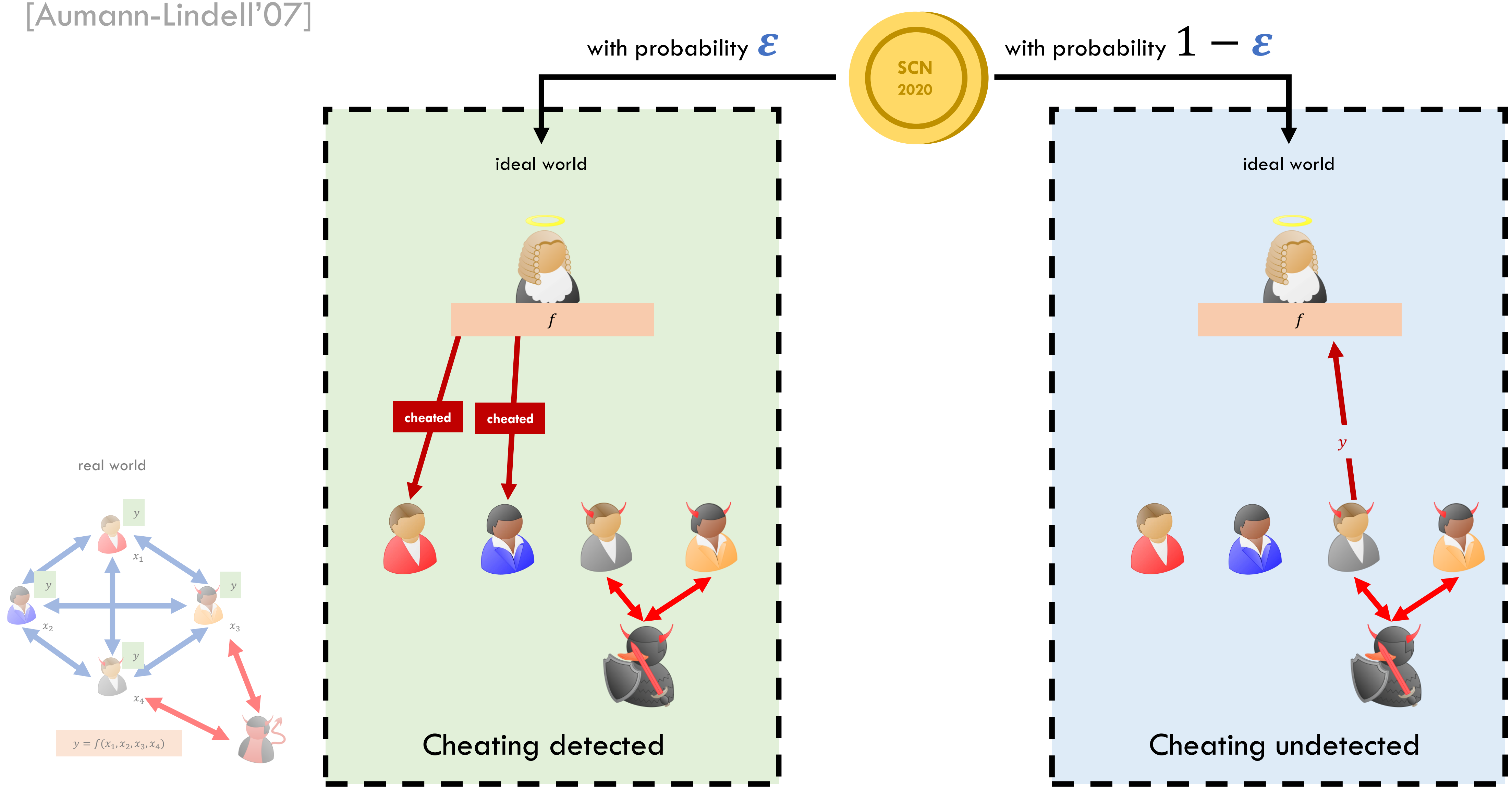

with probability ε
SCN
2020
with probability $1 - \varepsilon$
ideal world
f
cheated
cheated
ideal world
f
y
y

Cheating detected

Cheating undetected

Covert Adversaries

[Aumann-Lindell'07]



with probability ε

with probability $1-\varepsilon$

ideal world

f

cheated

cheated

Cheating detected

ideal world

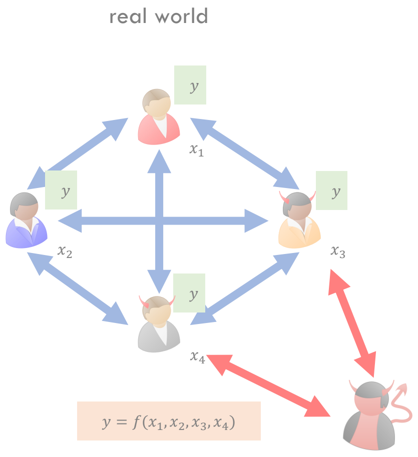
f

y

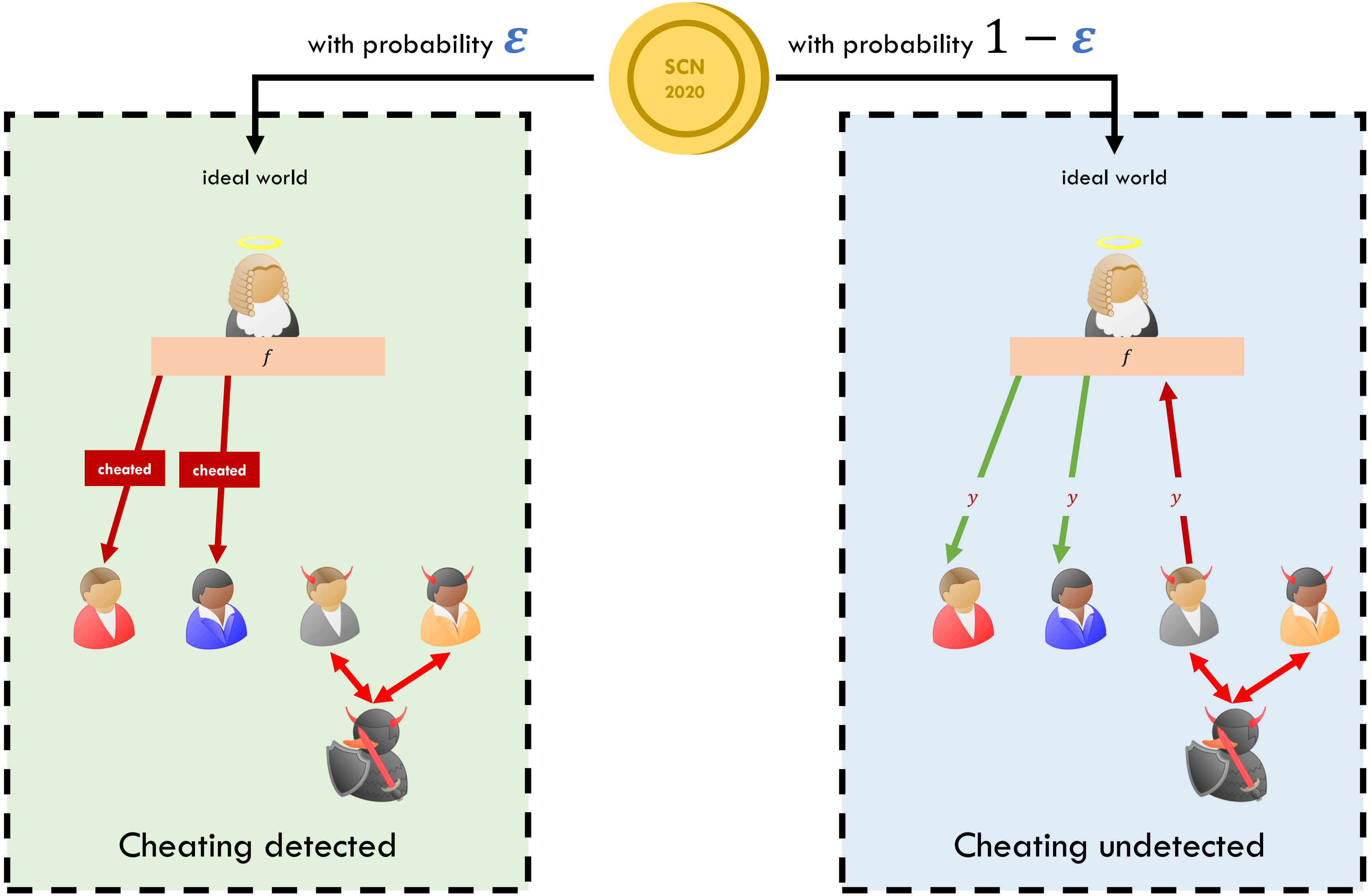
Cheating undetected

Covert Adversaries

[Aumann-Lindell'07]



real world



with probability ε

with probability $1 - \varepsilon$

ideal world

Cheating detected

ideal world

Cheating undetected

What is the precise round complexity of MPC protocols in the presence of covert adversaries?

What is the precise round complexity of MPC protocols in the presence of covert adversaries?

Best known results are those achieving malicious security, i.e. 4 rounds.

Prior works focus on computational complexity.

Our Results

Our Results

There does not exist a three round protocol in the presence of covert adversaries with respect to black-box simulation.

Our Results

There does not exist a three round protocol in the presence of covert adversaries with respect to black-box simulation.

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

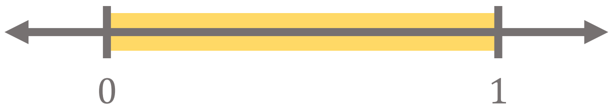
$$2 + 3 \cdot f(\varepsilon, n)$$

Our Results

There does not exist a three round protocol in the presence of covert adversaries with respect to black-box simulation.

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

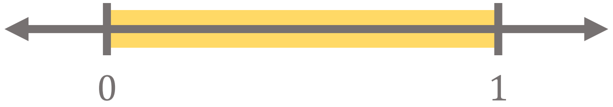
$$2 + 3 \cdot \underbrace{f(\varepsilon, n)}$$



Our Results

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot \underbrace{f(\varepsilon, n)}_{}$$

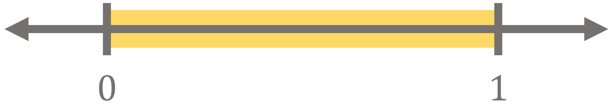


Our Results

n – number of 👤

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

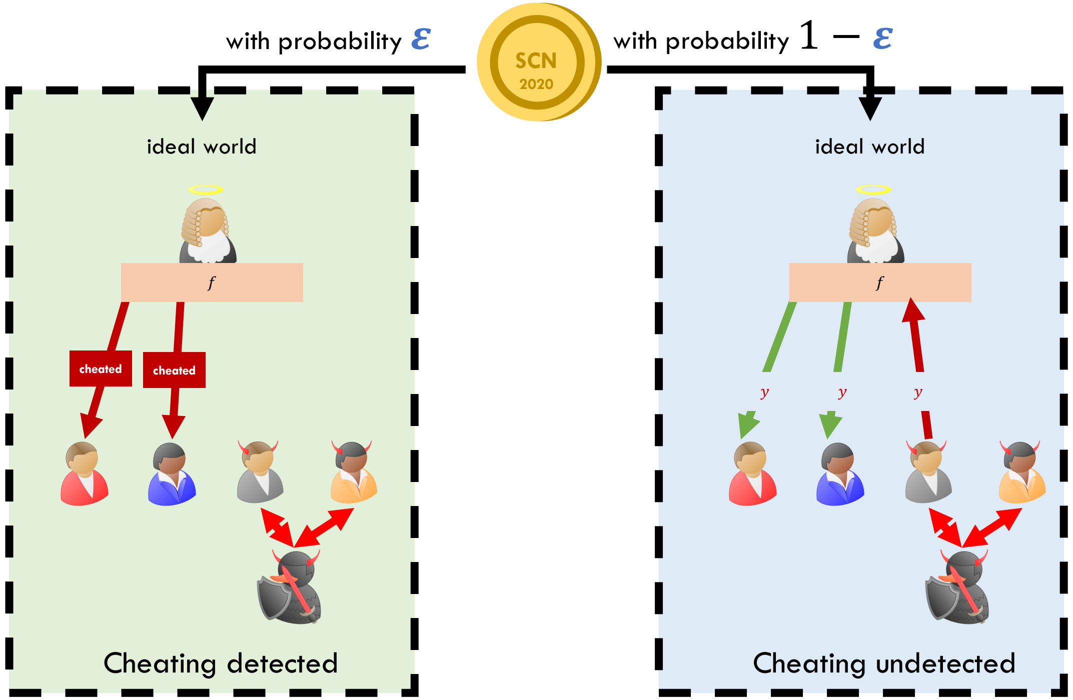
$$2 + 3 \cdot \underbrace{f(\varepsilon, n)}_{}$$



Our Results

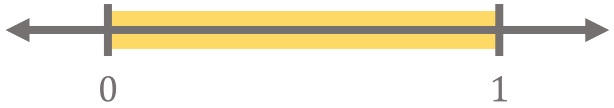
n – number of

ε – deterrence factor



There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2+3\cdot\underbrace{f(\varepsilon,n)}_{}$$



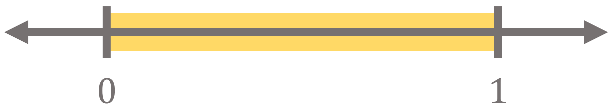
Our Results

n – number of 👤

ε – deterrence factor SCN 2020

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot \underbrace{f(\varepsilon, n)}_{}$$



Our Results

n – number of 👤

ε – deterrence factor 🪙 SCN 2020

Best case:

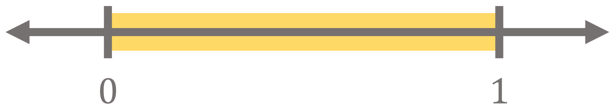
2 rounds

$\varepsilon = 0$

semi-honest

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot \underbrace{f(\varepsilon, n)}$$



Our Results

n – number of 👤

ε – deterrence factor 🪙

Best case:

2 rounds

$\varepsilon = 0$

semi-honest

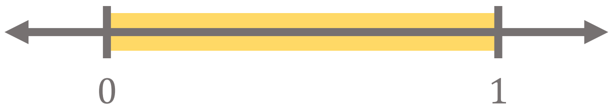
Worst case:

5 rounds

$\varepsilon \geq 1/2$

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot \underbrace{f(\varepsilon, n)}$$



Our Results

n – number of 👤

ε – deterrence factor SCN 2020

Best case:

2 rounds

$\varepsilon = 0$

Semi-honest

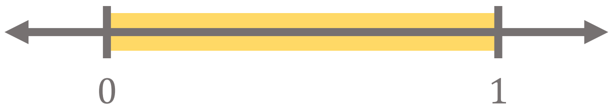
Worst case:

5 rounds

$\varepsilon \geq 1/2$

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot \underbrace{f(\varepsilon, n)}$$



Lower Bound

There does not exist a three round protocol in the presence of covert adversaries with respect to black-box simulation.

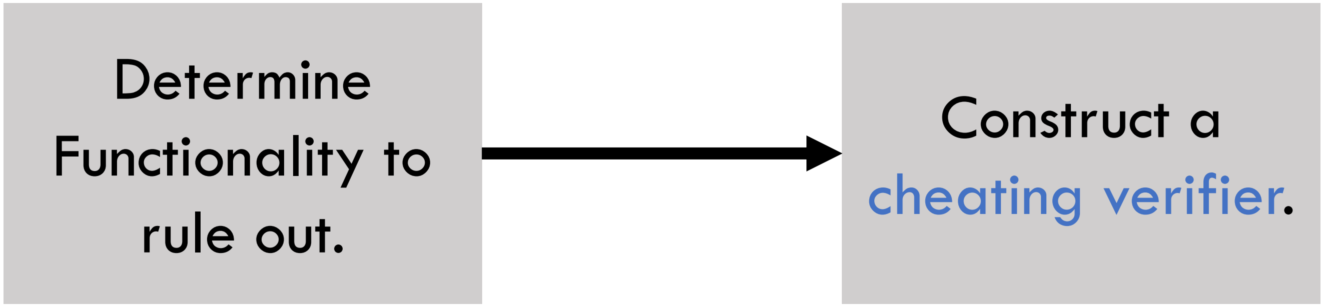
High Level Strategy

High Level Strategy

Determine Functionality to rule out.

Two Party **Zero-knowledge(ZK)** functionality

High Level Strategy

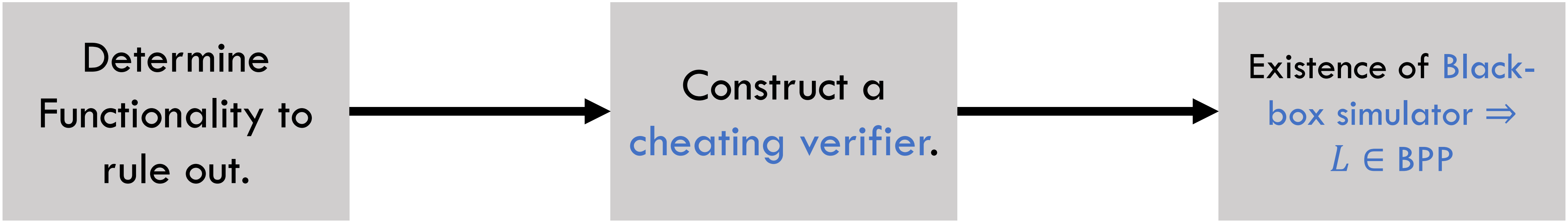


Determine Functionality to rule out.

Construct a cheating verifier.

Two Party **Zero-knowledge(ZK)** functionality

High Level Strategy



Determine Functionality to rule out.

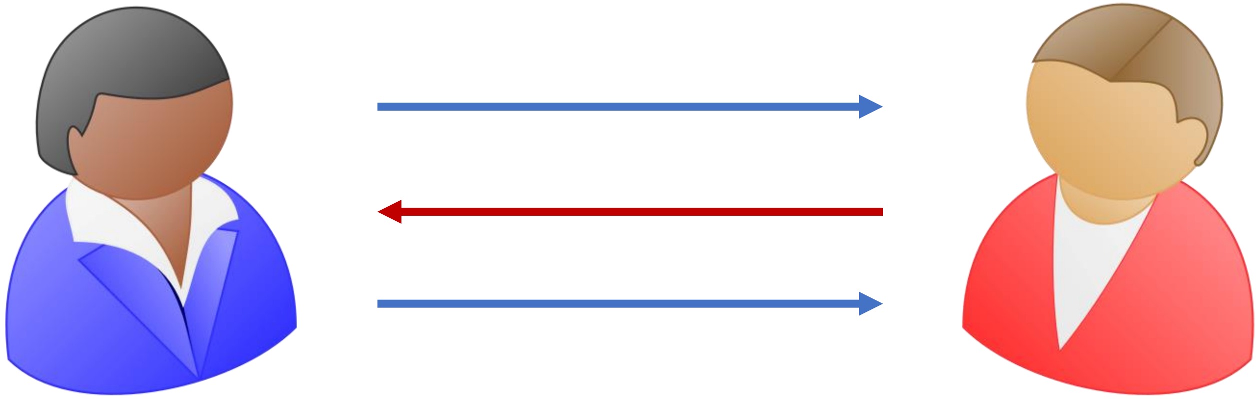
Construct a cheating verifier.

Existence of Black-box simulator ⇒ $L \in \text{BPP}$

Two Party **Zero-knowledge(ZK)** functionality

Only trivial languages

Two Party 3 round Zero-Knowledge protocol



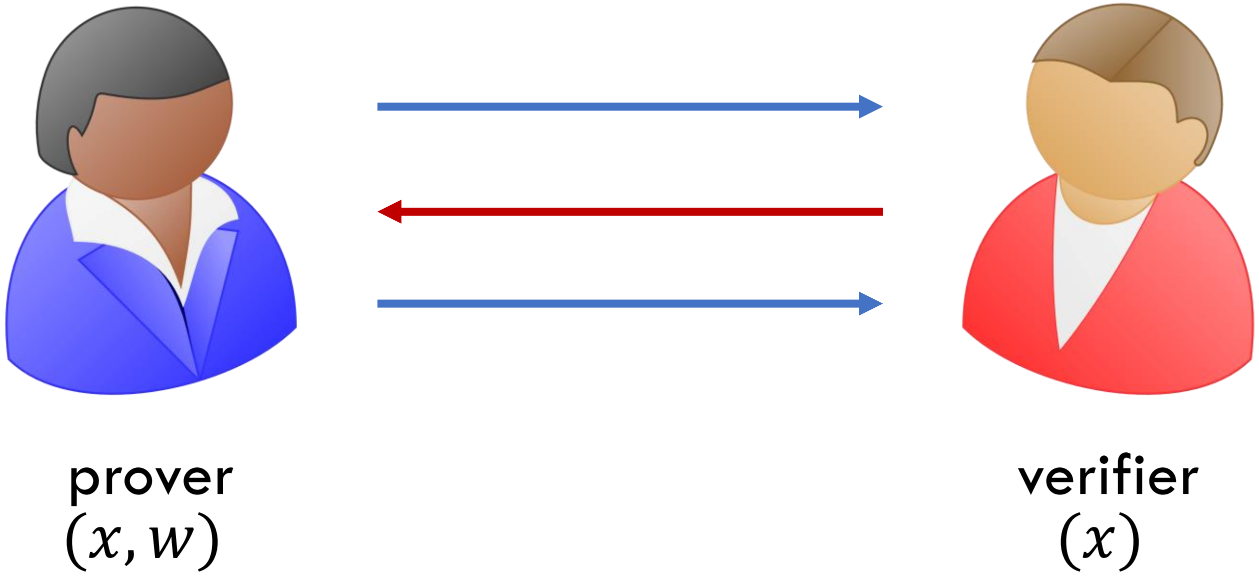
prover
(x, w)

verifier
(x)

Two Party 3 round Zero-Knowledge protocol

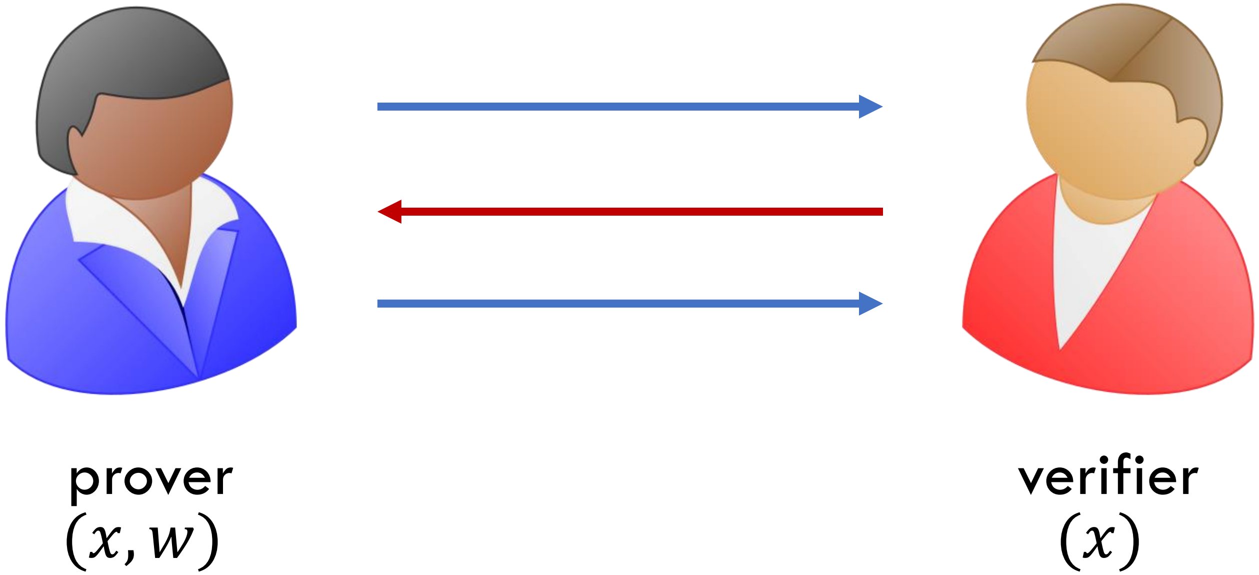
[Goldreich-Krawczyk'96]

3 round ZK does not exist



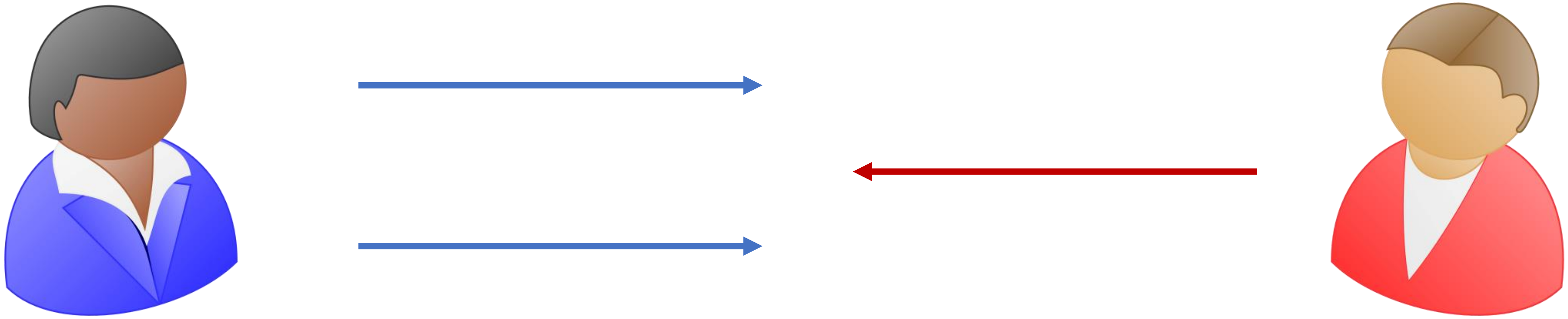
Two Party 3 round Zero-Knowledge protocol

[Goldreich-Krawczyk’96] 3 round ZK does not exist



Why doesn’t [Goldreich-Krawczyk’96] directly apply?

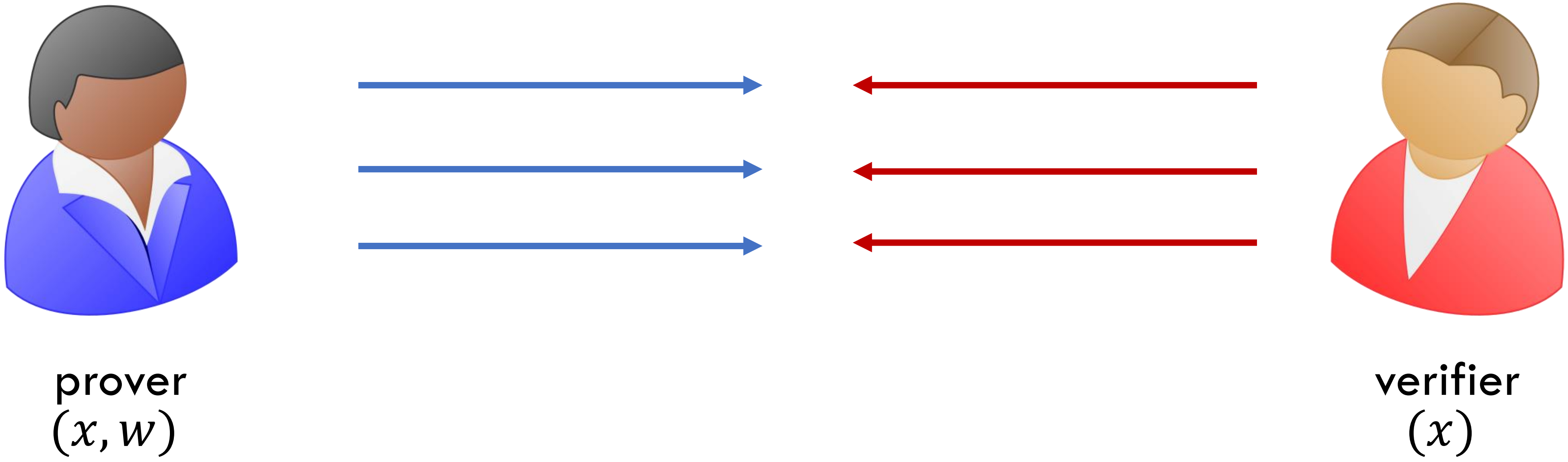
Challenge 1: Simultaneous Messages



prover
(x, w)

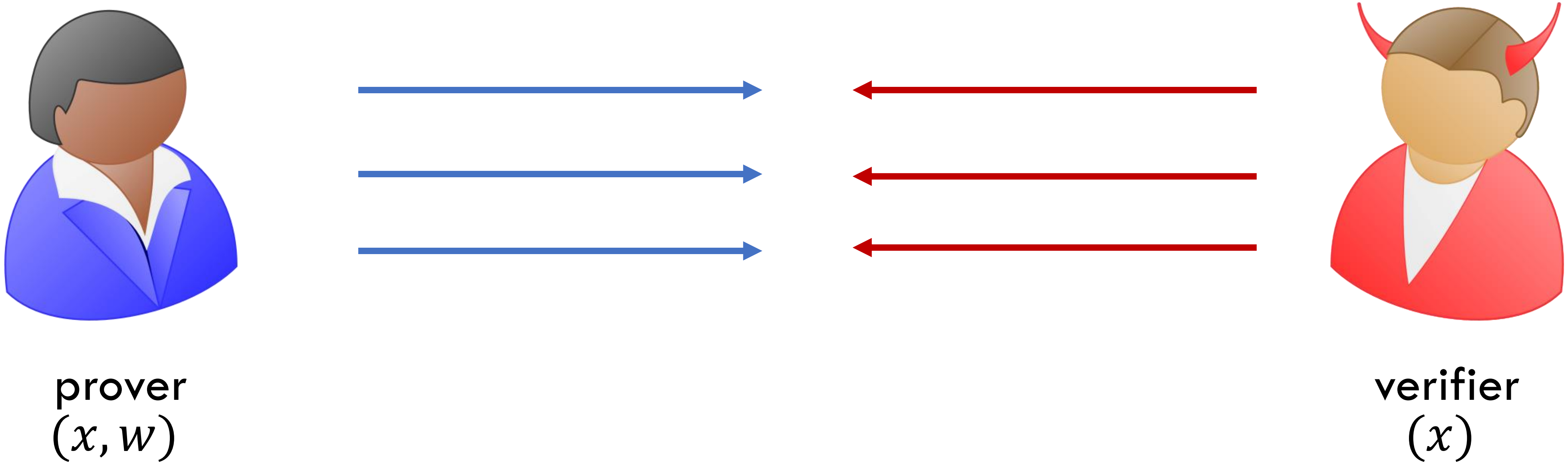
verifier
(x)

Challenge 1: Simultaneous Messages



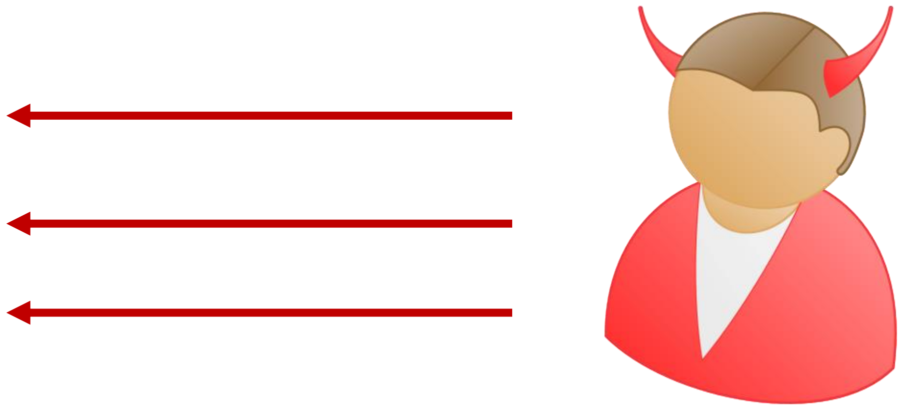
Simulator can potentially use all 3 messages sent by Bob to extract a trapdoor, and complete simulation.

Challenge 2: Covert Adversary



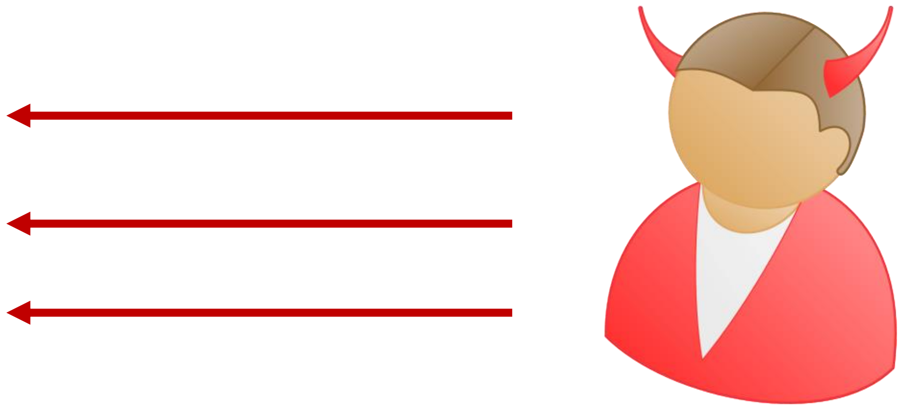
If Simulator determines Bob is cheating, can query with cheat message to get Alice's input, witness w.

The (Rushing) Cheating Verifier



verifier
(x)

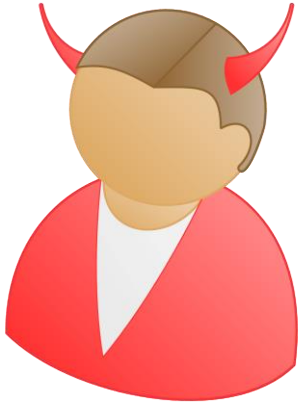
The (Rushing) Cheating Verifier



verifier
(x)

Cheating Bob can choose its i-th round message after receiving the i-th round message from the prover.

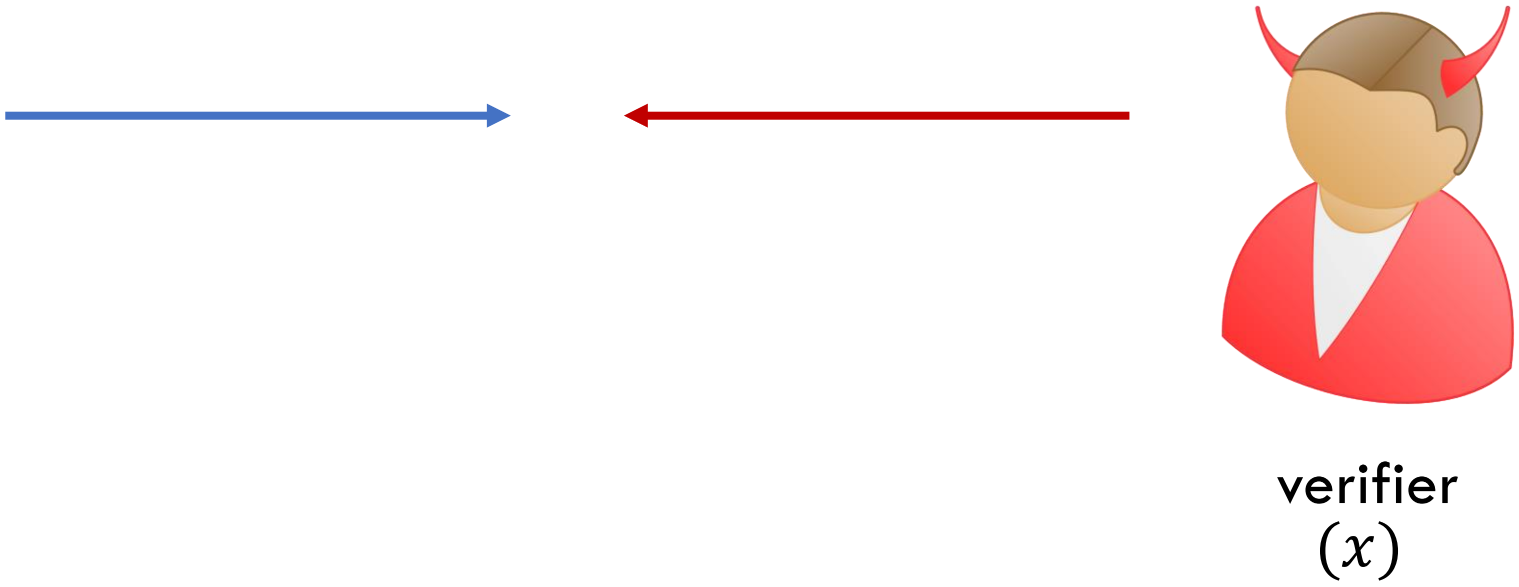
The (Rushing) Cheating Verifier



verifier
(x)

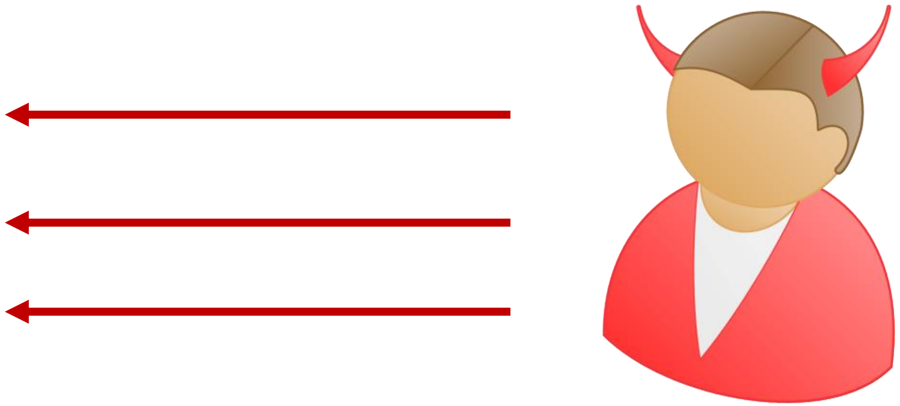
Cheating Bob can choose its i-th round message after receiving the i-th round message from the prover.

The (Rushing) Cheating Verifier



Cheating Bob can choose its i-th round message after receiving the i-th round message from the prover.

The (Rushing) Cheating Verifier



verifier
(x)

The (Rushing) Cheating Verifier

Intuition:

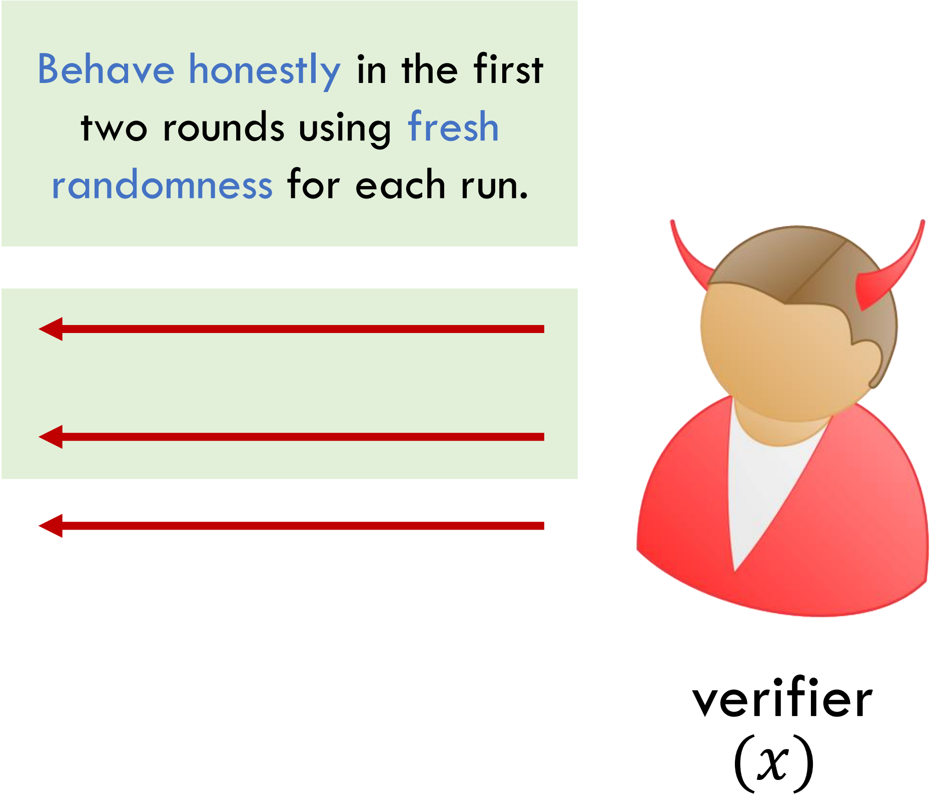
Behave honestly in the first two rounds using fresh randomness for each run.

verifier
(x)

The (Rushing) Cheating Verifier

Intuition:

1) Using fresh randomness prevents Simulator from rewinding first two rounds.

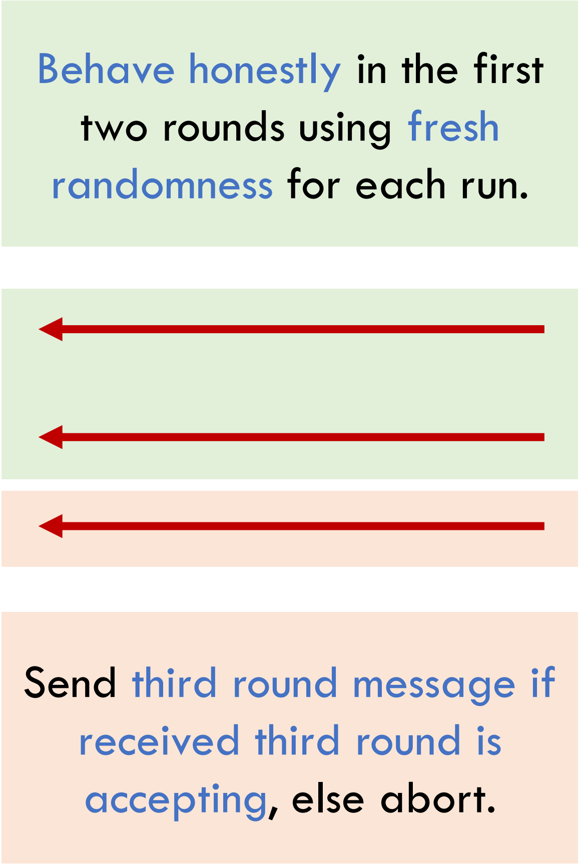


Behave honestly in the first two rounds using fresh randomness for each run.

The (Rushing) Cheating Verifier

Intuition:

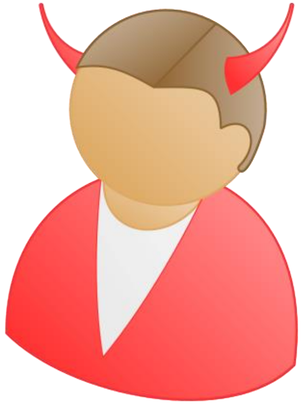
1) Using fresh randomness prevents Simulator from rewinding first two rounds.



Behave honestly in the first two rounds using fresh randomness for each run.

Send third round message if received third round is accepting, else abort.

Rushing Covert Verifier

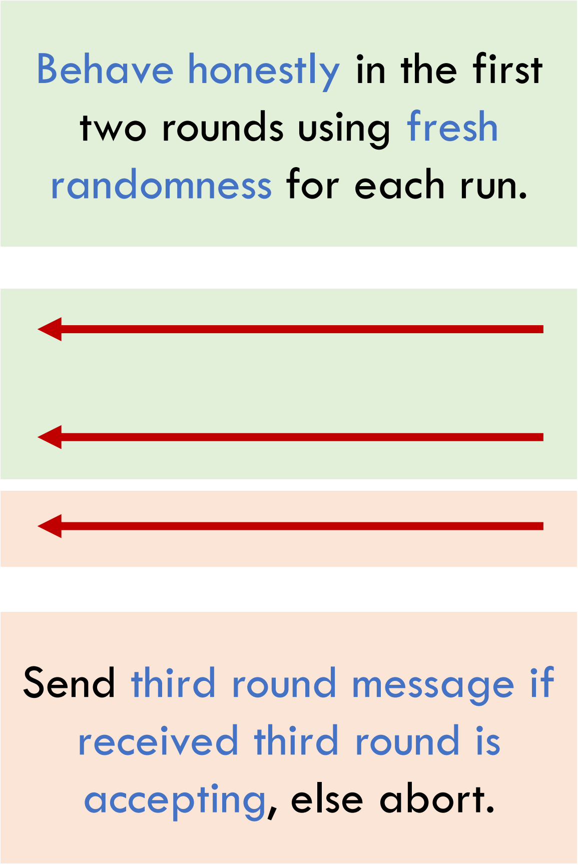


verifier (x)

The (Rushing) Cheating Verifier

Intuition:

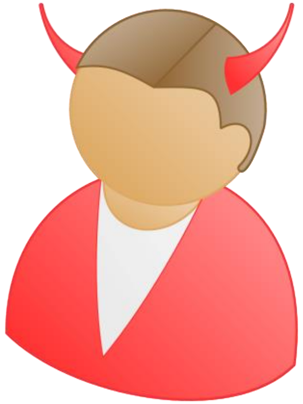
1) Using fresh randomness prevents Simulator from rewinding first two rounds.
2) Third round strategy prevents simulator from receiving Bob's third round unless it has an accepting transcript.



Behave honestly in the first two rounds using fresh randomness for each run.

Send third round message if received third round is accepting, else abort.

Rushing Covert Verifier

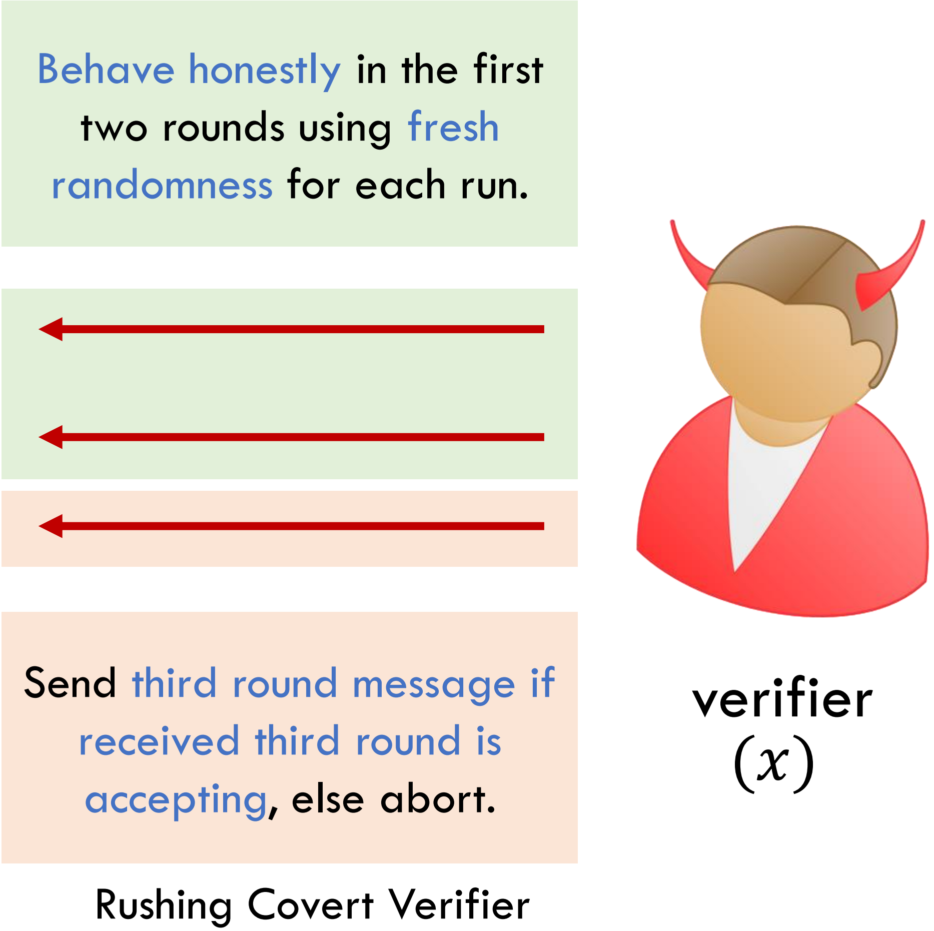


verifier (x)

The (Rushing) Cheating Verifier

Intuition:

1) Using fresh randomness prevents Simulator from rewinding first two rounds.
2) Third round strategy prevents simulator from receiving Bob’s third round unless it has an accepting transcript.
3) Bob will not cheat in an honest execution with the real prover $\Longrightarrow$ simulator cannot send cheat query.



Behave honestly in the first two rounds using fresh randomness for each run.

Send third round message if received third round is accepting, else abort.

Rushing Covert Verifier

Upper Bound

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

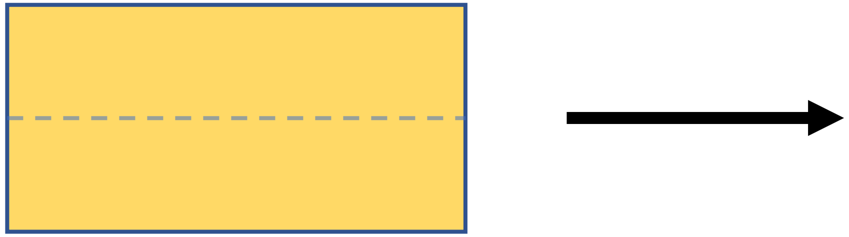
$$2 + 3 \cdot f(\varepsilon, n)$$

Upper Bound

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

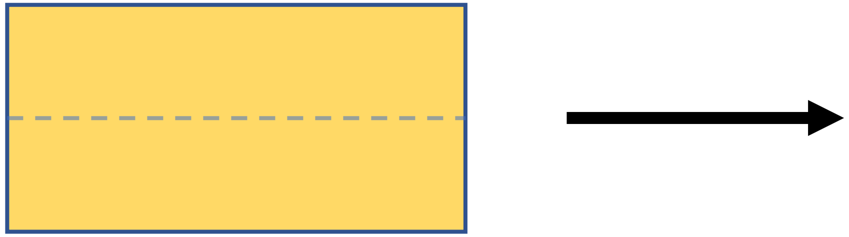
$$2 + 3 \cdot (1 - (1 - 2\varepsilon)^n)$$

Covert Adversary Compiler



Two round protocol secure against adversarial randomness

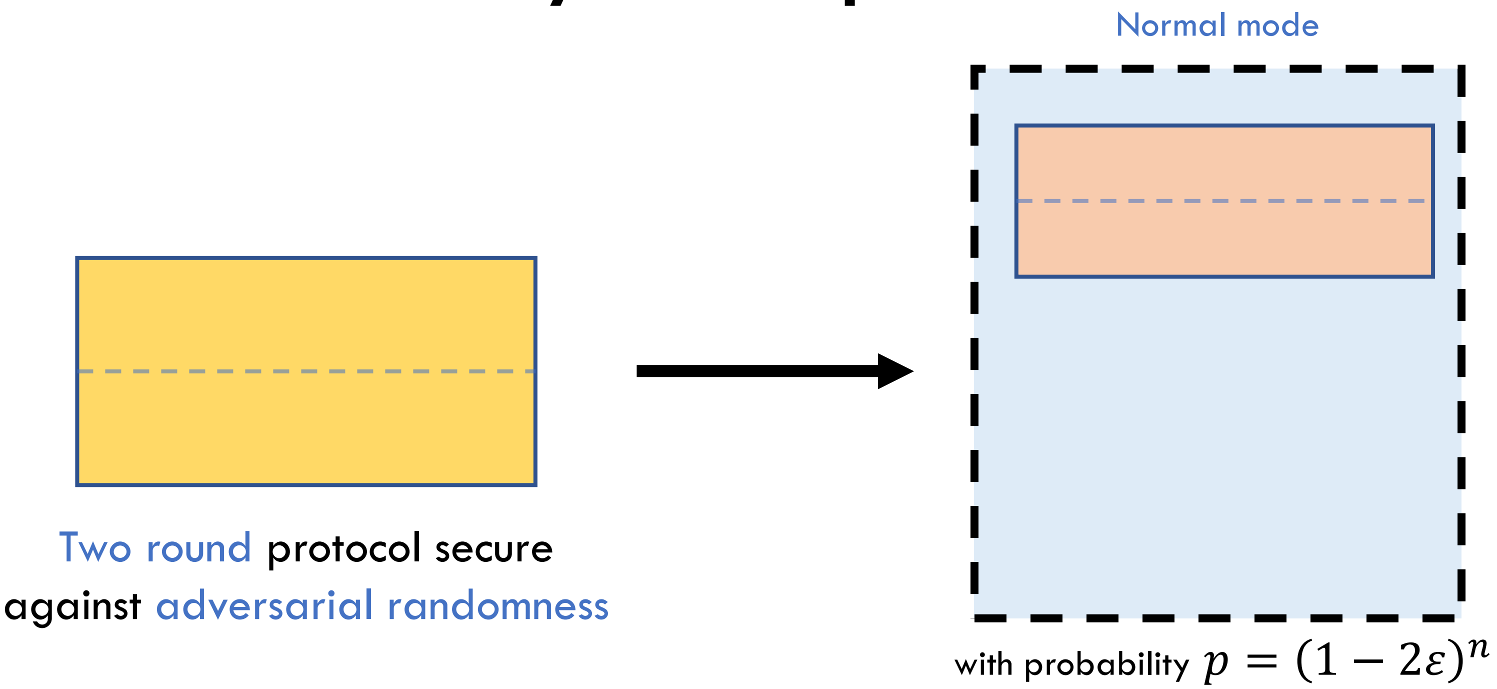
Covert Adversary Compiler



Two round protocol secure against adversarial randomness

Main Idea: Initiate check for honest behavior with certain probability.

Covert Adversary Compiler

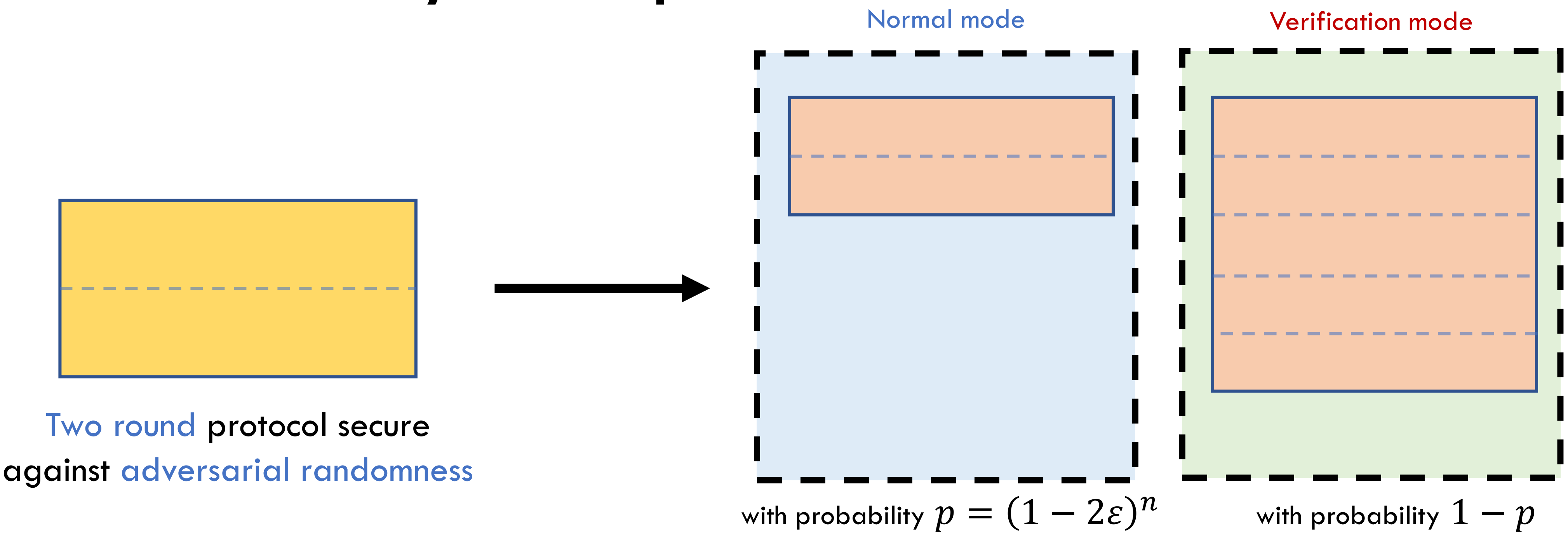


Two round protocol secure against adversarial randomness

with probability $p = (1 - 2\varepsilon)^n$

Main Idea: Initiate check for honest behavior with certain probability.

Covert Adversary Compiler



Two round protocol secure against adversarial randomness

Normal mode

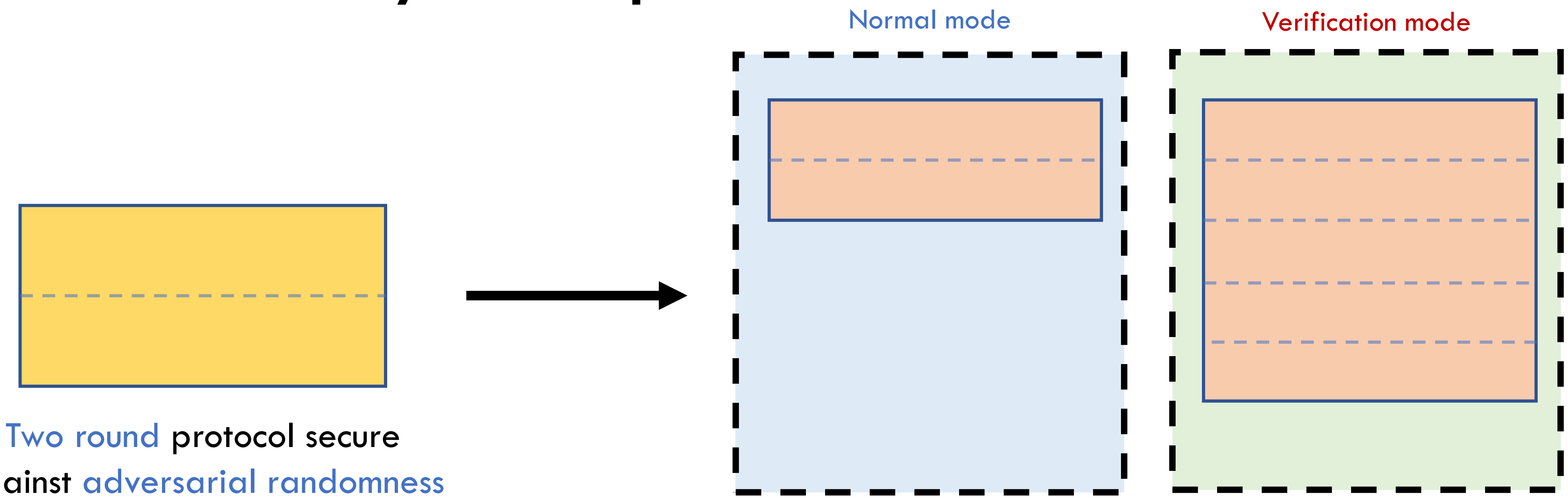
with probability $p = (1 - 2\varepsilon)^n$

Verification mode

with probability $1 - p$

Main Idea: Initiate check for honest behavior with certain probability.

Covert Adversary Compiler



Normal mode

Verification mode

Two round protocol secure against adversarial randomness

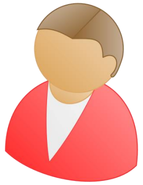
with probability $p = (1 - 2\varepsilon)^n$

with probability $\boxed{1 - p}$

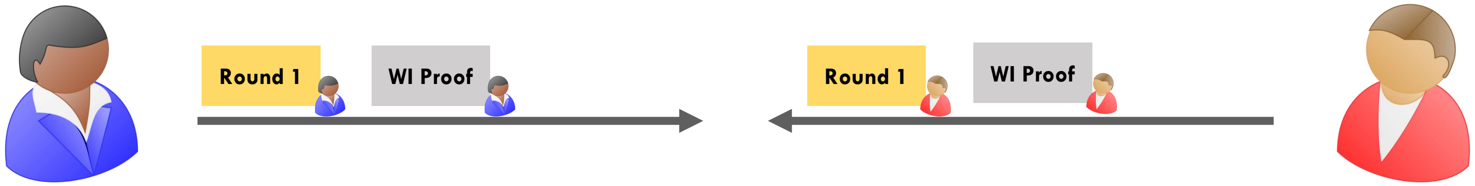
$\boldsymbol{f(\varepsilon, n)}$

Main Idea: Initiate check for honest behavior with certain probability.

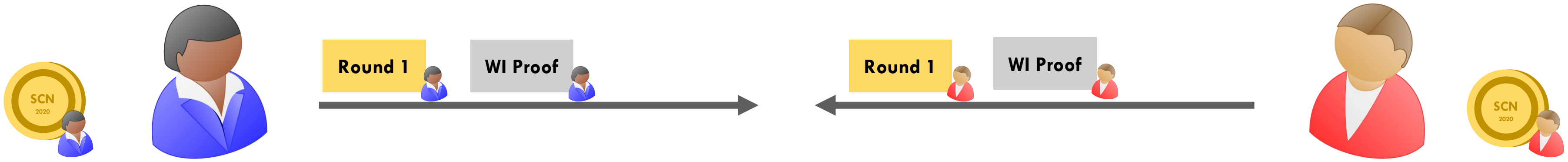
High Level Idea of the Protocol



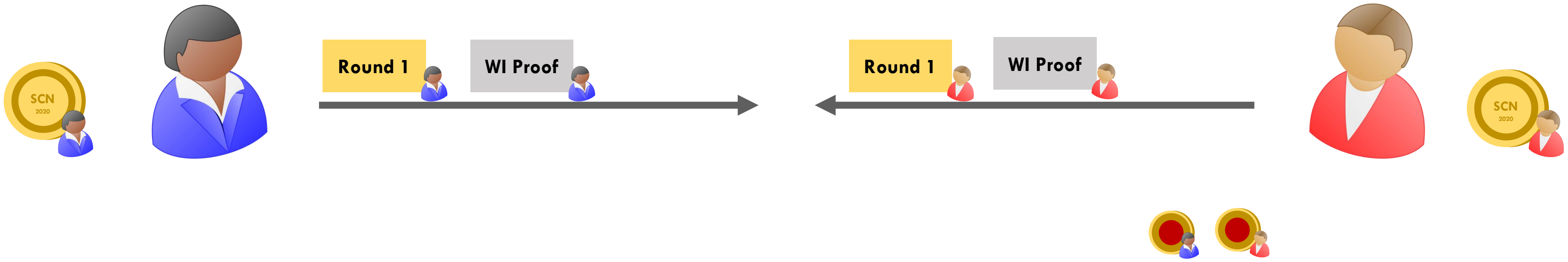
High Level Idea of the Protocol



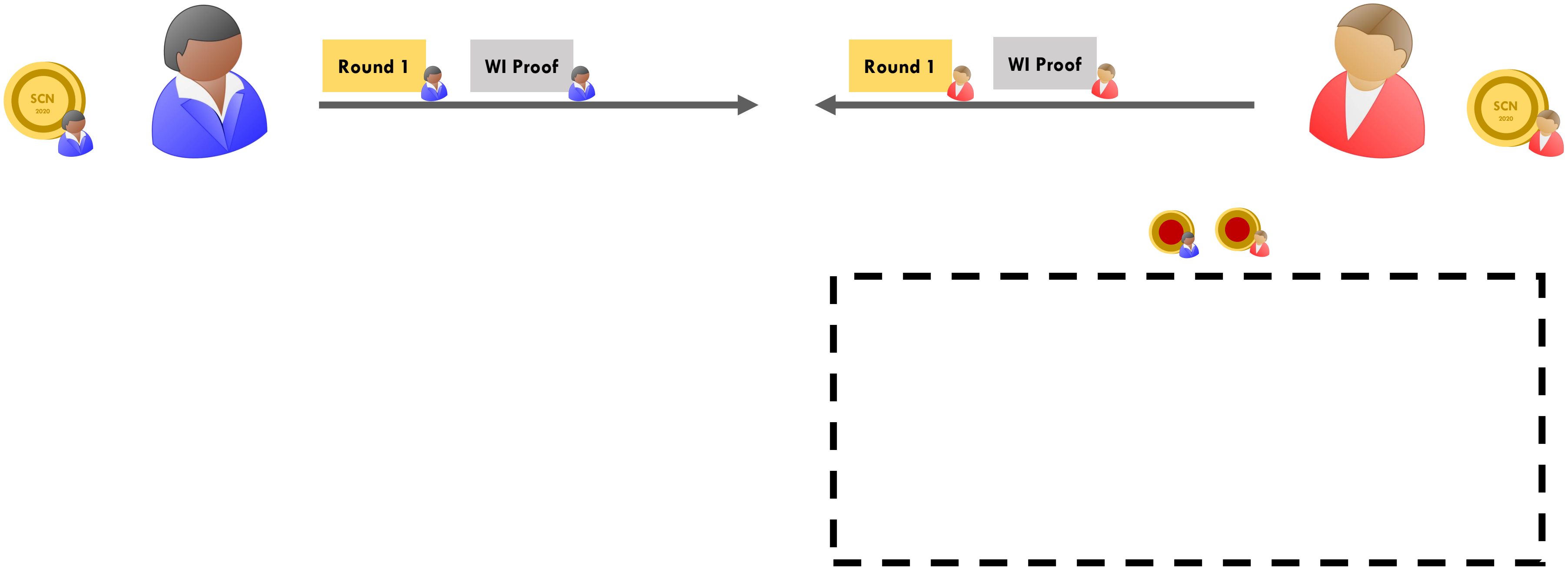
High Level Idea of the Protocol



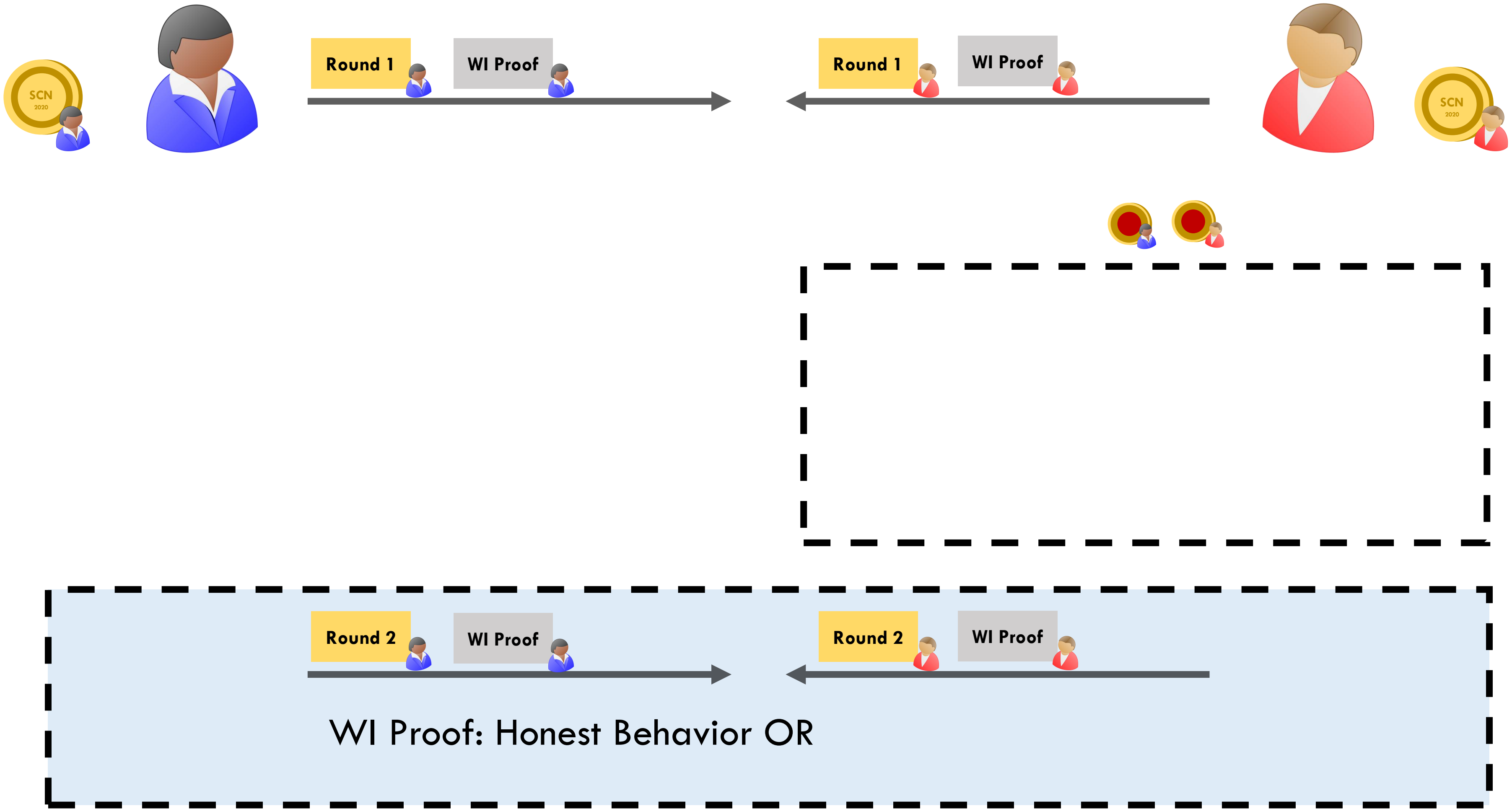
High Level Idea of the Protocol



High Level Idea of the Protocol

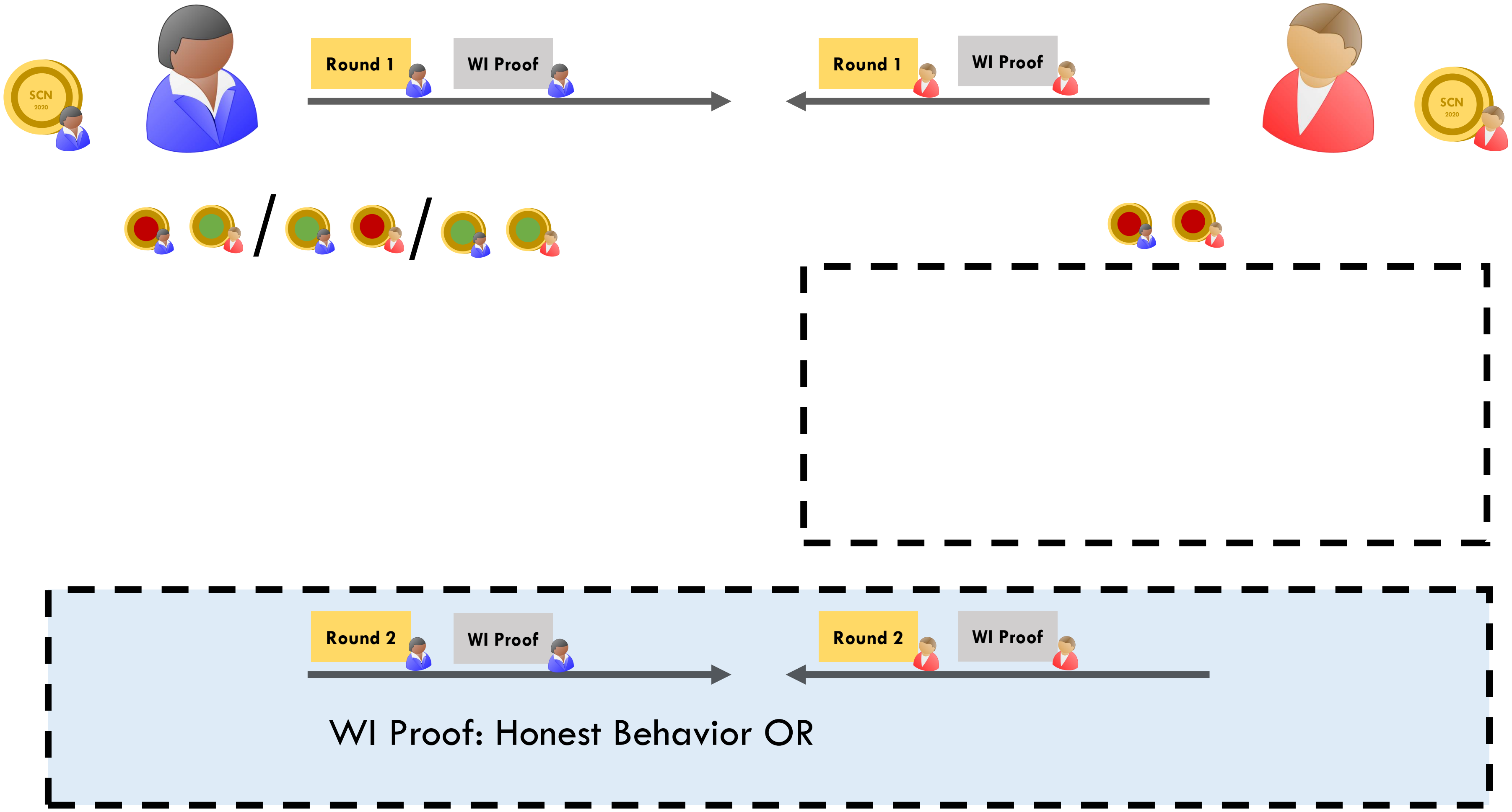


High Level Idea of the Protocol



WI Proof: Honest Behavior OR

High Level Idea of the Protocol



Round 1

WI Proof

Round 1

WI Proof

Round 2

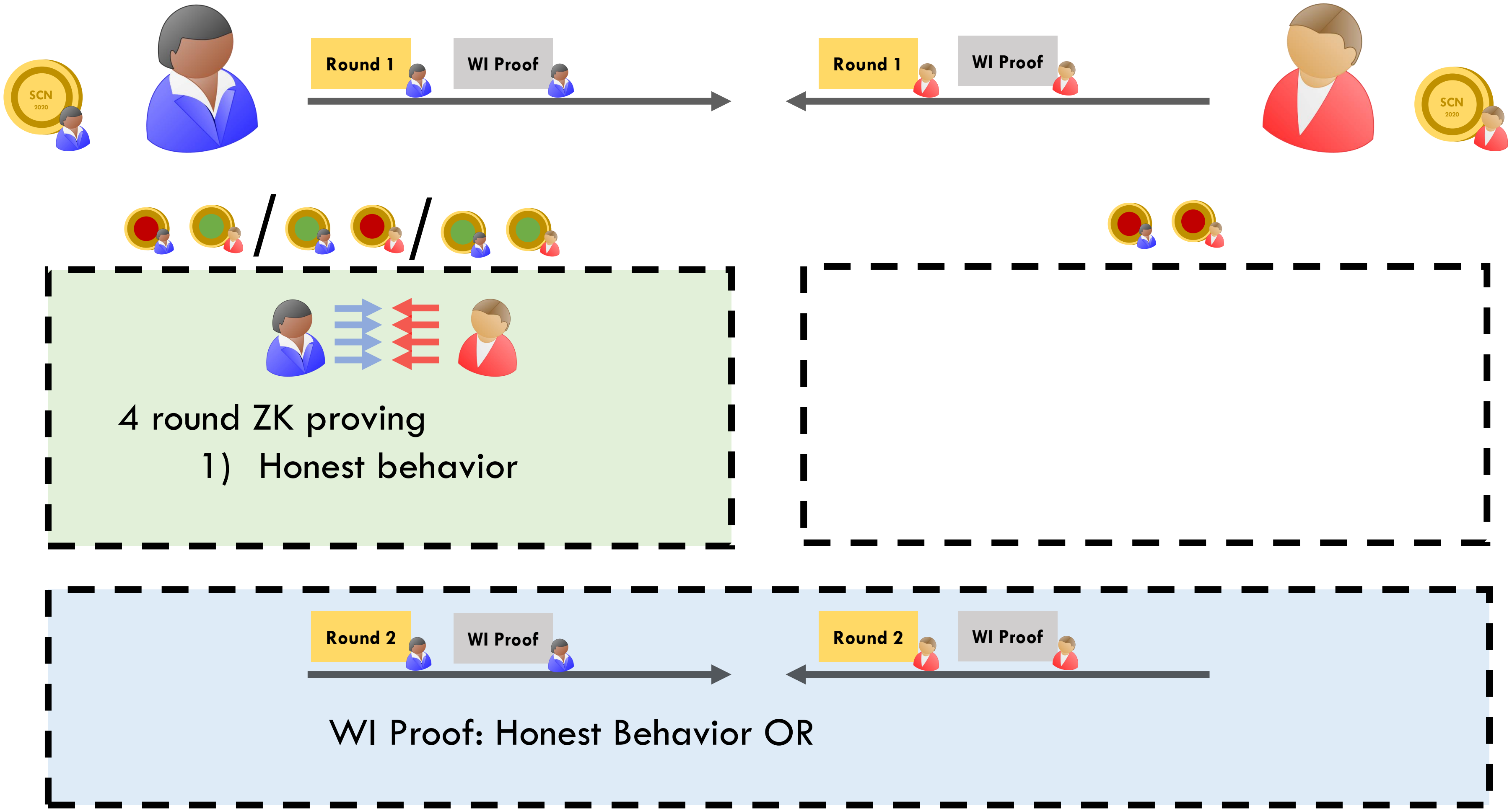
WI Proof

Round 2

WI Proof

WI Proof: Honest Behavior OR

High Level Idea of the Protocol

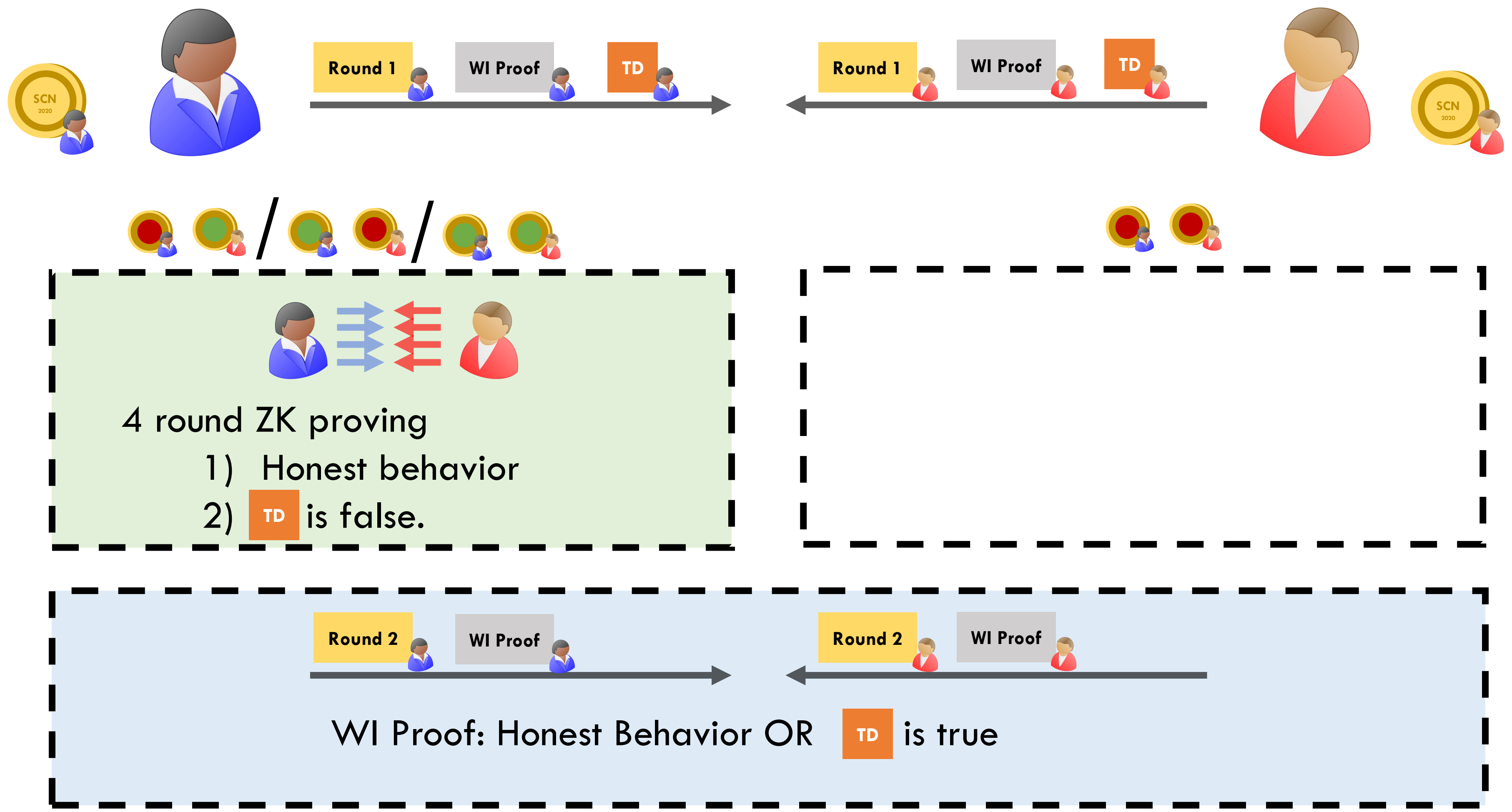


4 round ZK proving

1) Honest behavior

WI Proof: Honest Behavior OR

High Level Idea of the Protocol

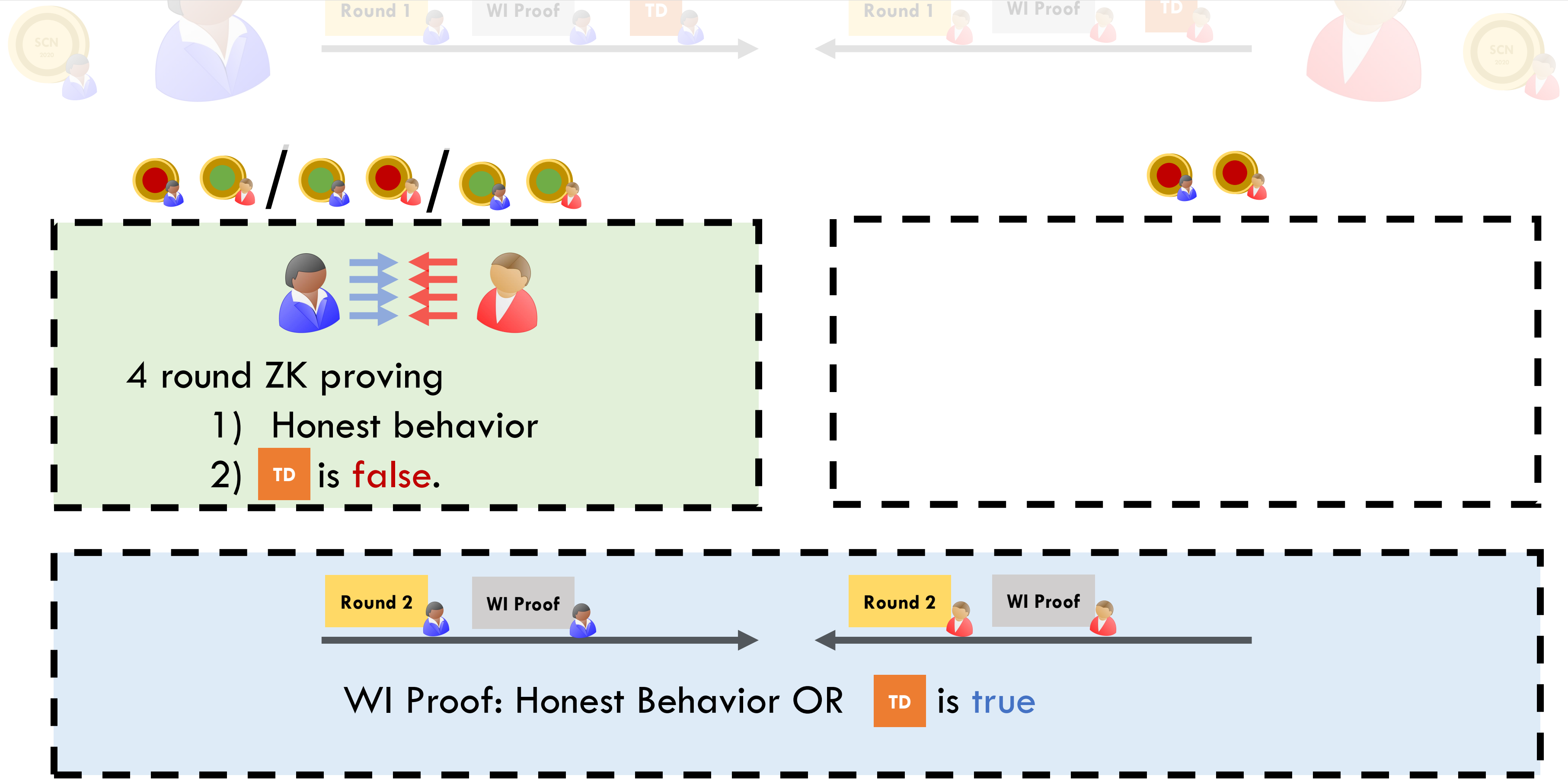


4 round ZK proving

1) Honest behavior
2) TD is false.

WI Proof: Honest Behavior OR TD is true

Intuition: Adversary must make a choice to set the trapdoor true to cheat in normal mode or risk being caught for incorrectly setting trapdoor in verification mode.



4 round ZK proving

1) Honest behavior

2) TD is false.

WI Proof: Honest Behavior OR TD is true

Several Simulation Challenges

For two rounds in best case:

Need stronger security from underlying protocol: free-simulation

Non-malleability issues

Skewed distribution of transcripts while checking if adversary is cheating

Worst case:

5 rounds

$\varepsilon \geq 1/2$

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot (1 - (1 - 2\varepsilon)^n)$$

Worst case 4 rounds?

Worst case:

5 rounds

$\varepsilon \geq 1/2$

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot (1 - (1 - 2\varepsilon)^n)$$

Worst case 4 rounds?

- Similar ideas might likely work
- Current 4 round protocols complex

Worst case:

5 rounds

$\varepsilon \geq 1/2$

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot (1 - (1 - 2\varepsilon)^n)$$

Worst case 4 rounds?

- Similar ideas might likely work
- Current 4 round protocols complex

Adversary forcing worst case?

Worst case:

5 rounds

$\varepsilon \geq 1/2$

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot (1 - (1 - 2\varepsilon)^n)$$

Worst case 4 rounds?

- Similar ideas might likely work
- Current 4 round protocols complex

Adversary forcing worst case?

- Won't be able to cheat covertly

Worst case:

5 rounds

$\varepsilon \geq 1/2$

There exists a variable round protocol in the presence of covert adversaries where the expected number of rounds are:

$$2 + 3 \cdot (1 - (1 - 2\varepsilon)^n)$$

There exists a protocol with expected number of rounds $2 + 3 \cdot (1 - (1 - 2\varepsilon)^n)$

There does not exist a three-round protocol.

There exists a protocol with expected number of rounds $2 + 3 \cdot (1 - (1 - 2\varepsilon)^n)$

There does not exist a three-round protocol.

Thank you. Questions?

Arka Rai Choudhuri

achoud@cs.jhu.edu

ia.cr/2020/1092