Towards a Game-Theoretic Security Analysis of Off-Chain Protocols

Sophie Rain, Zeta Avarikioti, Laura Kovacs, Matteo Maffei

Computer Security Foundations Symposium 2023
July 10, Dubrovnik
(Dis-)Proving **Incentives and Punishment Mechanisms** in Off-Chain Protocols do what they should - using **Game Theory**

Sophie Rain, Zeta Avarikioti, Laura Kovacs, Matteo Maffei
Structure

- Incentive and Punishment Mechanisms
- Game-Theoretic Security Properties
- Modeling Protocols as Games
- Results
Example: Lightning Network

- Opening phase. A, B (couple) lock 5 coins each, claimed by redistribution
  → Transaction on the blockchain
- can redistribute their 10 coins multiple times
- Lock = both signatures required
Example: Lightning Network

• **Opening phase.** A, B (couple) lock 5 coins each, claimed by redistribution → **Transaction on the blockchain**
  • can **redistribute** their 10 coins multiple times
  • **Lock** = both signatures required

• **Update phase.** E.g. B buys something for both (2 coins), A wants to give him her half.
  They agree on updating the redistribution state to 4 for Alice, 6 for Bob. → **off-chain**
  • many more updates follow...
Example: Lightning Network

2 cases for closing phase

Consensus (honest).

- A, B still happy, want to close channel
- publish latest update on the blockchain
- receive fair part of money
Example: Lightning Network

2 cases for closing phase

Consensus (honest).
- A, B still happy, want to close channel
- publish latest update on the blockchain
- receive fair part of money

Dispute (honest).
- horrible break-up, closing required
- A wants to do better than last update
- A publishes old distribution state on the blockchain
- B can prove state is outdated
- B receives 10 coins, A 0 coins
Example: Lightning Network

1) Incentives and Punishment Mechanisms

- honest behavior → fair split
- dishonest behavior → lose all money

Honest Behavior

- intended course of action in protocol

Is it always rational for cheated party to prove other published outdated state?
What is done already?

Cryptographic aspects of Blockchain protocols

- Universal Composability Framework:
  - cryptography = ideal functionality

... but what about rationality?

Incentive / Punishment mechanisms

rely on game-theoretic arguments

e.g. Lightning’s closing
What do we verify?

- 3 types of users

  - honest
  - rational
  - Byzantine
What do we verify?

- 3 types of users
  - honest
  - rational
  - Byzantine

No one has a reason to deviate!
What do we verify?

- 3 types of users
  - honest
  - rational
  - Byzantine

No one has a reason to deviate!

- done
- honest = best
- cannot harm honest
What do we verify?

1) Incentive-Compatibility
   “no profit from deviation”

2) Byzantine-Fault Tolerance
   • “even in presence of Byzantine users, honest ones not harmed”

Note: 1) + 2) enough
No assumption of honest/rational percentage
What do we verify *exactly*?

- 1) **Incentive-Compatibility**
  
  \[ \forall C, d_C. \sum_{R \in C} u_R(h_C, h_{-C}) \geq \sum_{R \in C} u_R(d_C, h_{-C}) \]

  **Collusion Resilience**

  **Practicality**

  always greedy choice

- 2) **Byzantine-Fault Tolerance**

  **Weak Immunity**

  \[ \forall r_{-H}. u_H(h_H, r_{-H}) \geq 0 \]
Introduction to Game Theory

Extensive Form Game

Finite set of players

Sequential actions

Pay-Off
Introduction to Game Theory

Extensive Form Game

Finite set of players
Sequential actions
Pay-Off

Game Tree
Modeling Lightning’s closing

A, B

A: publish latest state \((a, b)\)

publish old state \((a+d, b-d)\)

sign closing tx \((a,b)\), or \((a+c,b-c)\)
Modeling Lightning’s closing

A, B

A: publish latest state \((a,b)\)
   publish old state \((a+d, b-d)\)
   sign closing tx \((a,b)\), or \((a+c,b-c)\)

B: ignore \((a+d, b-d)\)
   prove it was old state \((0, a+b-f)\)
Modeling Lightning’s closing

A, B

A:
- publish latest state (a,b)
- publish old state (a+d, b-d)
- sign closing tx (a,b), or (a+c,b-c)

B:
- ignore (a+d, b-d)
- prove it was old state (0, a+b-f)

A: -a
B: a-f+alpha

symbolic, constraint, relative, infinitesimal, quantified
Full Model for Lightning’s Closing

3) Modeling Protocols as Games

[Diagram of a model with various states and transitions labeled with specific values and symbols, representing the states of a protocol game.]
Full Model for Lightning’s Closing
Full Model for Lightning’s Closing
Full Model for Lightning’s Closing
3) Example incl. Demo

3) Modeling Protocols as Games

Full Model for Lightning’s Closing
Full Model for Lightning’s Closing
3) Modeling Protocols as Games

Full Model for Lightning’s Closing
3) Modeling Protocols as Games

Full Model for Lightning’s Closing
3) Example incl. Demo

Full Model for Lightning’s Closing
Full Model for Lightning’s Closing

3) Modeling Protocols as Games

trillions of joint strategies
“Partial" Model for Lightning’s Routing

3) Modeling Protocols as Games

at least trillions of joint strategies
How do we verify it?

- protocol + honest behavior
- game + honest history
- security properties

4) Results

Authors: secure

not satisfied

not secure

satisfied

secure
A Protocol is Secure, if...

...its intended behavior satisfies IC and BFT.

Protocol → Extensive Form Game

Intended Behavior → “honest” terminal history h*

A game + h* are secure, if...

...there are strategies extending h*, which are weak immune, collusion resilient, practical.
Security Results for Closing and Routing

No unknown attacks found.
Security Results for Closing and Routing

No unknown attacks found.

Closing (a \rightarrow A, b \rightarrow B):

**Can honest participants be harmed?**  YES, if a,b < f

**Is the honest behavior rational?**
No, old state (a+d \rightarrow A, b – d \rightarrow B), where a+d < f
Security Results for Closing and Routing

![Image](image_url)

No unknown attacks found.

**Closing (a → A, b → B):**

**Can honest participants be harmed?** YES, if a,b < f

**Is the honest behavior rational?** No, old state (a+d → A, b – d → B), where a+d < f

**Routing:**

**Can honest participants be harmed?** YES

**Is the honest behavior rational?** NO
Take-Away

- protocol + honest behavior
- game + honest history
- security properties
- authors

not satisfied

not secure

satisfied

secure