Proof of Work and Blockchains

Ittay Eyal

Computer Science, Cornell University
The Initiative for Cryptocurrencies and Contracts (IC3)

Distributed Cryptocurrencies and Consensus Ledgers, July 2017
Proof of Work

- Challenger provides puzzle
- Solver expends resources to solve puzzle
Proof of Work

A variety of uses [Jakobbson+Juels’99]

• Spam protection [Dwork+Naor’92]
• construction of digital time capsules [Goldschlag+Stubblebine’89, Rivest+’96]
• Server access metering [Franklin+Malkhi’97]
• (D)DoS protection [Juels+Brainard’99]
• Digital money minting [Rivest+Shamir’01]
• Sybil protection [Apsnes’15]

... but botnets?
How Hard?

Phone? Laptop? Server? Datacenter?
PoW for Blockchains

• Bitcoin [Nakamoto’08]:
  PoW for Sybil protection,
  With a trick:
    direct monetary compensation

• The result:
  Wildly successful and incredibly robust
  But also:
    some surprising properties
A Replicated State Machine

$A_1 \rightarrow B_1$

$A_1 \rightarrow A_2$

$B_1 \rightarrow C_1$

Log
The Blockchain

Log

Block

header

Blockchain
PoW for Blockchains

- Log in blocks
- Solve puzzle to add block
- Get prize per block
- On a fork (a natural event), stronger side wins
Basic Operation

• Puzzle is a function of current and previous block. (e.g., their hash smaller than target)

• Real-world participation cost
• Burn real-world resources, committing to a state machine history
PoW in a Blockchain

• Block every set interval (10min, 15sec)
• Automatically adjusting difficulty
  ==> a lottery of sorts
  ==> bustling mining industry
PoW in a Blockchain

• Block every set interval (10min, 15sec)
• Automatically adjusting difficulty
  ==> a lottery of sorts
  ==> bustling mining industry

Bitcoin
prize decay ==> FOMO at work
Also finite supply, deflation
Waste?

- Real-world waste
  - Compute power ($\text{sha256}^2$)
    - Really power (Watts)
  - Less useless (Primecoin)
  - Storage [Miller+’14]
  - Hardware (PoET)

- No real-world waste
  - Permissioned (Hyperledger, Stellar), or
  - Pending formal discussion (Proof of Stake)
Resilience

• Surprisingly stable
  • Strategic mining
    (Selfish mining etc. not seen in the wild)

• Few blockchain alternatives
  • GHOST +variants (Ethereum, DECOR)
    [Sompolinsky+Zohar’15, Lewenberg+’15]
  • Bitcoin-NG +variants (Hybrid consensus, Byzcoin)
Pooled Mining
Blockchain Mining

Proof of Work

money

Blockchain

Constant rate: globally updated Difficulty
Pooled Mining

Many miners
Constant PoW rate \(\Rightarrow\) Long time to win

Miners form pools
Pooled Mining

Many miners
Constant PoW rate \[\implies\] Long time to win

Miners form pools

Blockchain
Pooled Mining

Blockchain

Full PoW

money
Pooled Mining

Blockchain

Money

Partial PoW

Full PoW

money
Open Pools and Centralization

- Miners form pools
- Largest are **open pools**
- Lead to centralization

A threat to the blockchain’s basic premise
Pool Block Withholding
Oakland’15
Pool Block Withholding
Pool Block Withholding

Attacker:
- Registers as standard miner
- Uses some miners as moles
Pool Block Withholding

Attacker:

- Registers as standard miner
- Uses some miners as moles
- Drops full PoW
Pool Block Withholding

Attacker:

- Registers as standard miner
- Uses some miners as moles
- Drops full PoW

Sabotage?
Factors influencing revenue

Less direct mining power

Money

Less miners ==> reduced difficulty
The Pool Game

Goal

Maximize *revenue density*

Round

One pool updates infiltration rates
The Pool Game

Goal
Maximize revenue density

Round
One pool updates infiltration rates

Analysis
- Stable state (equilibrium)
- Generic (any pool size)
Analysis
One Attacker

Game progress:
One round – attacker optimizes $r_1(x_{1,2})$

Dominant strategy: Attack
Honest pool mining is not an equilibrium

In general:
Honest pool mining is not an equilibrium

(For any two pools, one should attack)
Two Attackers

Game progress
Repeatedly:
1. Pool 1 optimizes $r_1(x_{1,2}, x_{2,1})$
2. Pool 2 optimizes $r_2(x_{2,1}, x_{1,2})$

A single feasible equilibrium point
The Miner’s Dilemma

When both pools are minorities of any size:

<table>
<thead>
<tr>
<th></th>
<th>NO ATTACK</th>
<th>ATTACK</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>pool 1</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO ATTACK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ATTACK</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Pool 2*
# The Miner’s Dilemma

When both pools are minorities of any size:

<table>
<thead>
<tr>
<th></th>
<th>pool 1</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NO ATTACK</td>
<td>ATTACK</td>
</tr>
<tr>
<td>pool 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO ATTACK</td>
<td><img src="image1.png" alt="Face" /></td>
<td><img src="image2.png" alt="Face" /></td>
</tr>
<tr>
<td>ATTACK</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
One Attacker

Pool size: 24%
Infiltration rate: 25%
Revenue: +3%

Victim size

Attacker size

Attacker revenue density
**The Miner’s Dilemma**

When both pools are minorities of any size:

<table>
<thead>
<tr>
<th>pool 2</th>
<th>NO ATTACK</th>
<th>ATTACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO ATTACK</td>
<td><img src="neutral.png" alt="Neutral" /></td>
<td><img src="sad.png" alt="Sad" /></td>
</tr>
<tr>
<td>ATTACK</td>
<td><img src="upset.png" alt="Upset" /></td>
<td><img src="happy.png" alt="Happy" /></td>
</tr>
</tbody>
</table>

- **Pool 1**:
  - **NO ATTACK**: The miner is content.
  - **ATTACK**: The miner feels upset.

- **Pool 2**:
  - **NO ATTACK**: The miner is neutral.
  - **ATTACK**: The miner feels happy.

- **Decision**: min(A1, A2), where A1 and A2 are the Miners’ attacks in pools 1 and 2, respectively.
### The Miner’s Dilemma

When both pools are minorities of any size:

<table>
<thead>
<tr>
<th>pool 2</th>
<th>pool 1</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NO ATTACK</strong></td>
<td><strong>NO ATTACK</strong></td>
</tr>
<tr>
<td><strong>ATTACK</strong></td>
<td><strong>ATTACK</strong></td>
</tr>
</tbody>
</table>

- **NO ATTACK**  
  - **NO ATTACK**: Neutral face.  
  - **ATTACK**: Sad face.  

- **ATTACK**  
  - **NO ATTACK**: Sad face.  
  - **ATTACK**: Happy face.

*Ittay Eyal, July 16*
Two Attackers

Pool sizes: 24%, 13%
Infiltration rate: 8%, 12%
Revenue: -4%, -10%

Pool sizes:
- Pool 1 size
- Pool 2 size

Infiltration rate:
- Pool 1
- Pool 2

Revenue:
- Pool 1
- Pool 2
The Miner’s Dilemma

When both pools are minorities of any size:

<table>
<thead>
<tr>
<th></th>
<th>pool 1</th>
<th>pool 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO</td>
<td><img src="image" alt="No Attack" /></td>
<td><img src="image" alt="No Attack" /></td>
</tr>
<tr>
<td>ATTACK</td>
<td><img src="image" alt="Attack" /></td>
<td><img src="image" alt="Attack" /></td>
</tr>
</tbody>
</table>

- **NO ATTACK** results in a neutral outcome for both pools.
- **ATTACK** results in a negative outcome for the pool that attacks and a positive outcome for the attacked pool.
# The Miner’s Dilemma

When both pools are minorities of any size:

<table>
<thead>
<tr>
<th>pool 1</th>
<th>NO ATTACK</th>
<th>ATTACK</th>
</tr>
</thead>
<tbody>
<tr>
<td>pool 2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>NO ATTACK</td>
<td><img src="image1" alt="Emoji" /></td>
<td><img src="image2" alt="Emoji" /></td>
</tr>
<tr>
<td>ATTACK</td>
<td><img src="image3" alt="Emoji" /></td>
<td><img src="image4" alt="Emoji" /></td>
</tr>
</tbody>
</table>

This is good
## The Miner’s Dilemma

When both pools are minorities of any size:

<table>
<thead>
<tr>
<th></th>
<th>pool 1</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NO ATTACK</td>
<td>ATTACK</td>
</tr>
<tr>
<td>NO ATTACK</td>
<td><img src="image1" alt="Face" /></td>
<td><img src="image2" alt="Face" /></td>
</tr>
<tr>
<td>ATTACK</td>
<td><img src="image3" alt="Face" /></td>
<td><img src="image4" alt="Face" /></td>
</tr>
</tbody>
</table>

Iterated game with unbounded rounds $\Rightarrow$

Possible non-equilibrium stable state
Countermeasures

• Detection
  Does not work
Countermeasures

• Detection
  Does not work

• Bonus for full PoW / seniority
  Reduces revenue homogeneity
Countermeasures

- Detection
  Does not work

- Bonus for full PoW / seniority
  Reduces revenue homogeneity

- Honey pot
  Wastes resources
Countermeasures

• Detection
  Does not work

• Bonus for full PoW / seniority
  Reduces revenue homogeneity

• Honey pot
  Wastes resources

• Out of band enforcement
  Implies small trust circles
open pools \rightarrow \text{reduced eligibility} \rightarrow \text{smaller pools}
Conclusion

• Proof of work: cornerstone of open blockchains
  • Some waste
  • Effective security
    (being proven in retrospect)

• Architecture leads to surprising properties
  • The miner’s dilemma
  • Pooled mining
  • Industrial mining
  • Selfish mining
  • Non-standard proof-of-work
  • Proof of work outsourcing
  • Proof of work in face of chain forks