ETH zürich

Tampering with the Delivery of Blocks and Transactions in Bitcoin

Arthur Gervais*, Hubert Ritzdorf*,
Ghassan O. Karame’, Srdjan Capkun*

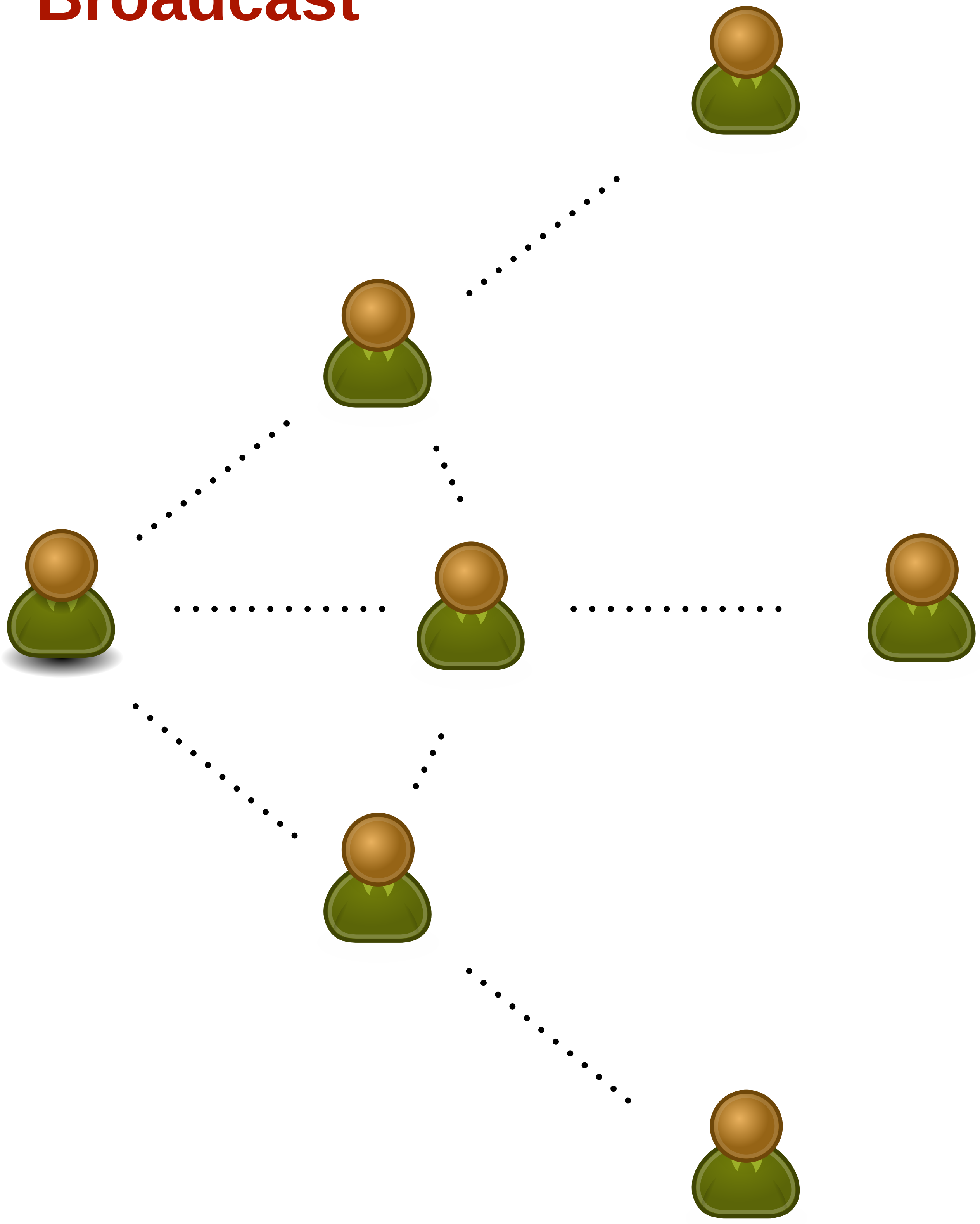
*ETH Zurich, ’NEC Laboratories Europe

Scaling Bitcoin Phase 2 - 2015

Bitcoin

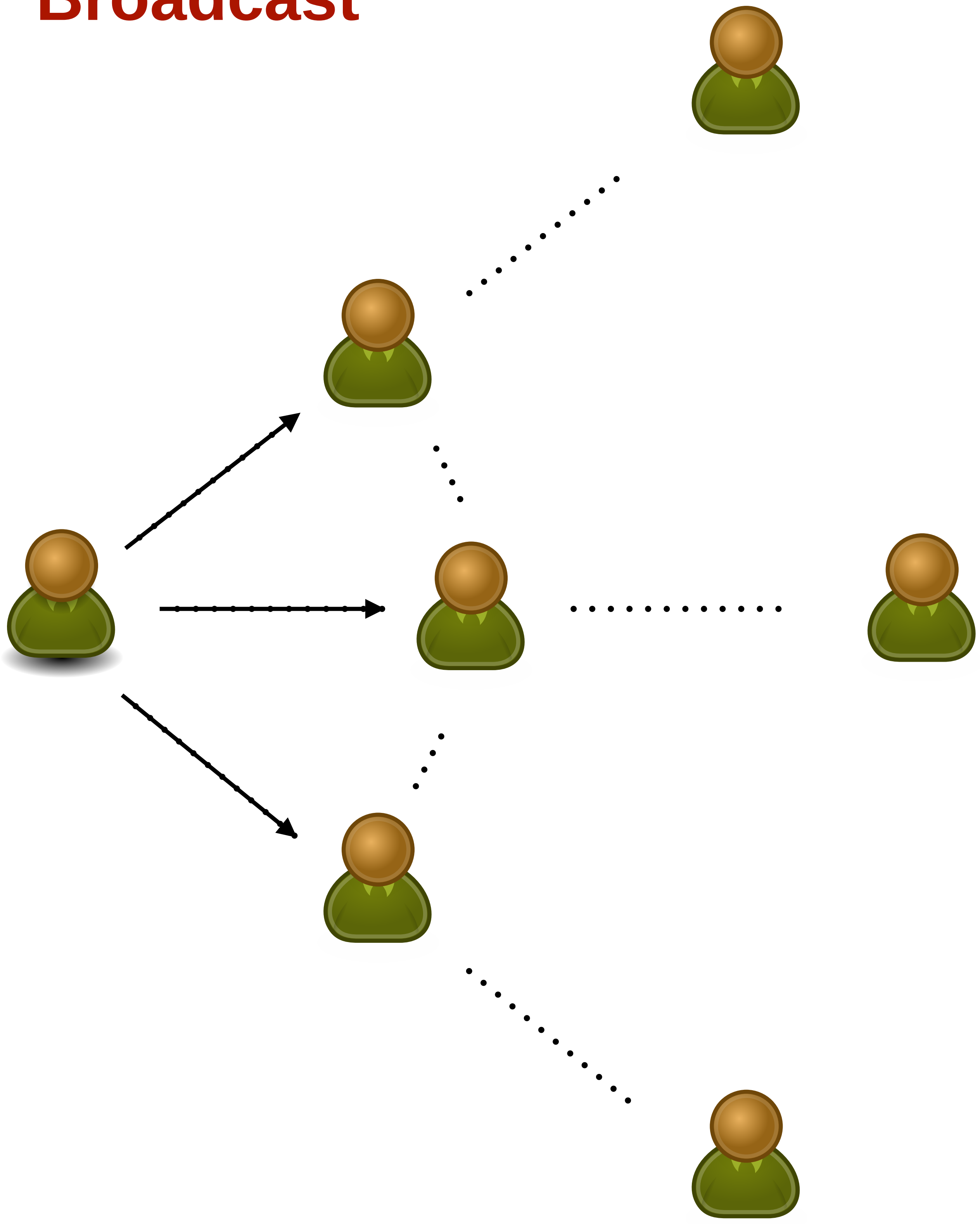
- Peer-to-peer decentralized currency
- No trusted third parties
- Blockchain: distributed DB
  - Transactions
  - Blocks
- Broadcast protocol

Broadcast



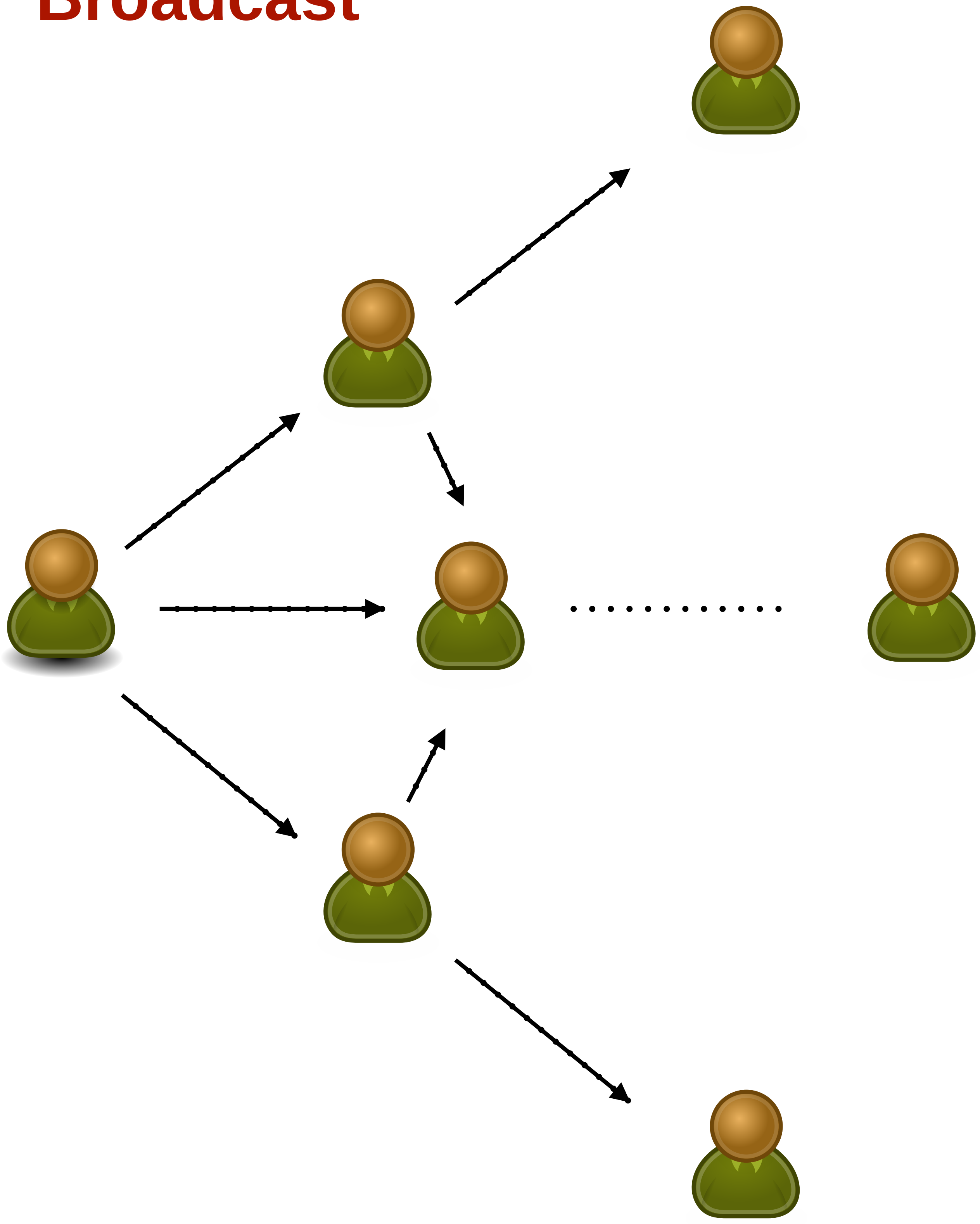
Broadcast of information

Broadcast



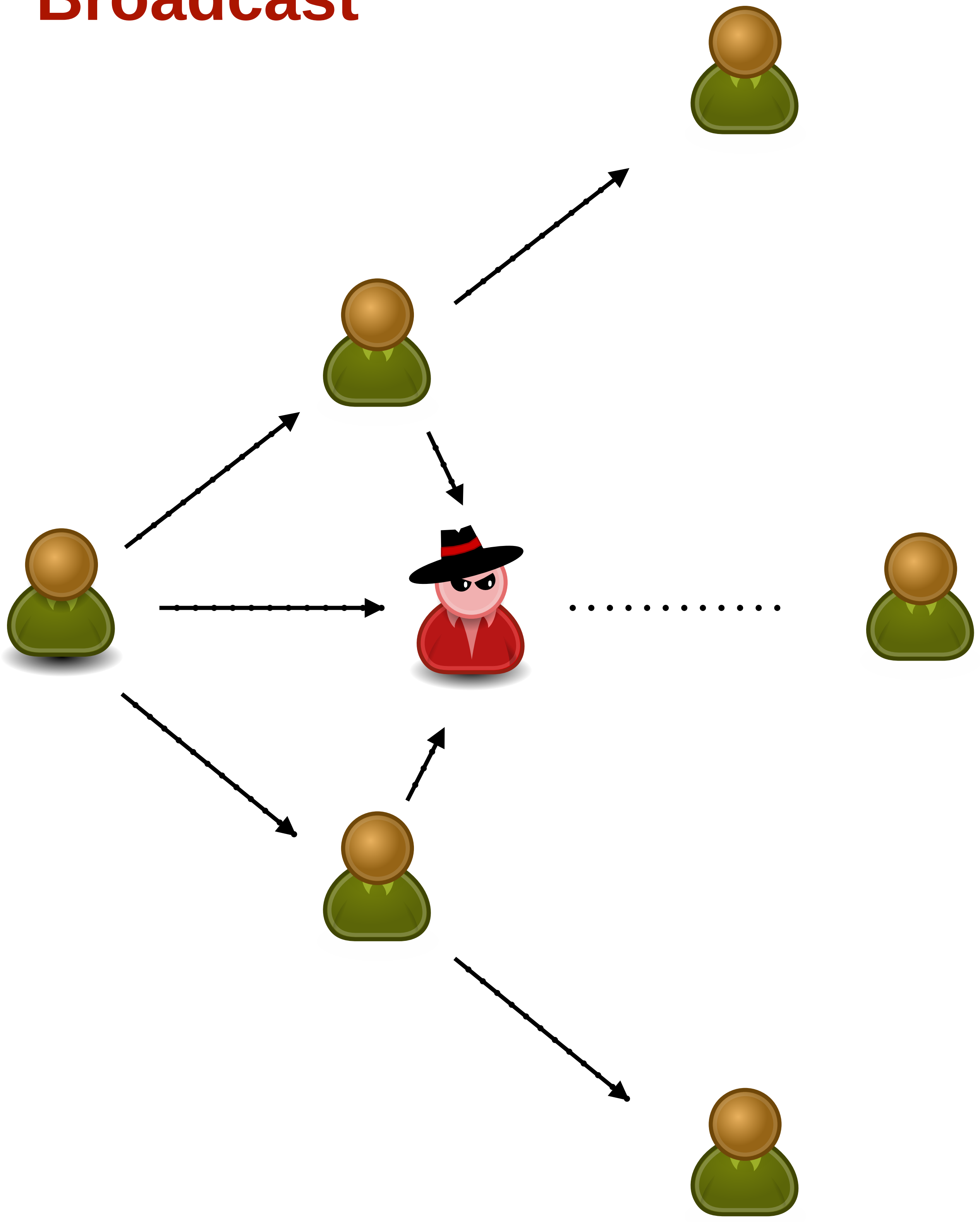
Broadcast of information

Broadcast

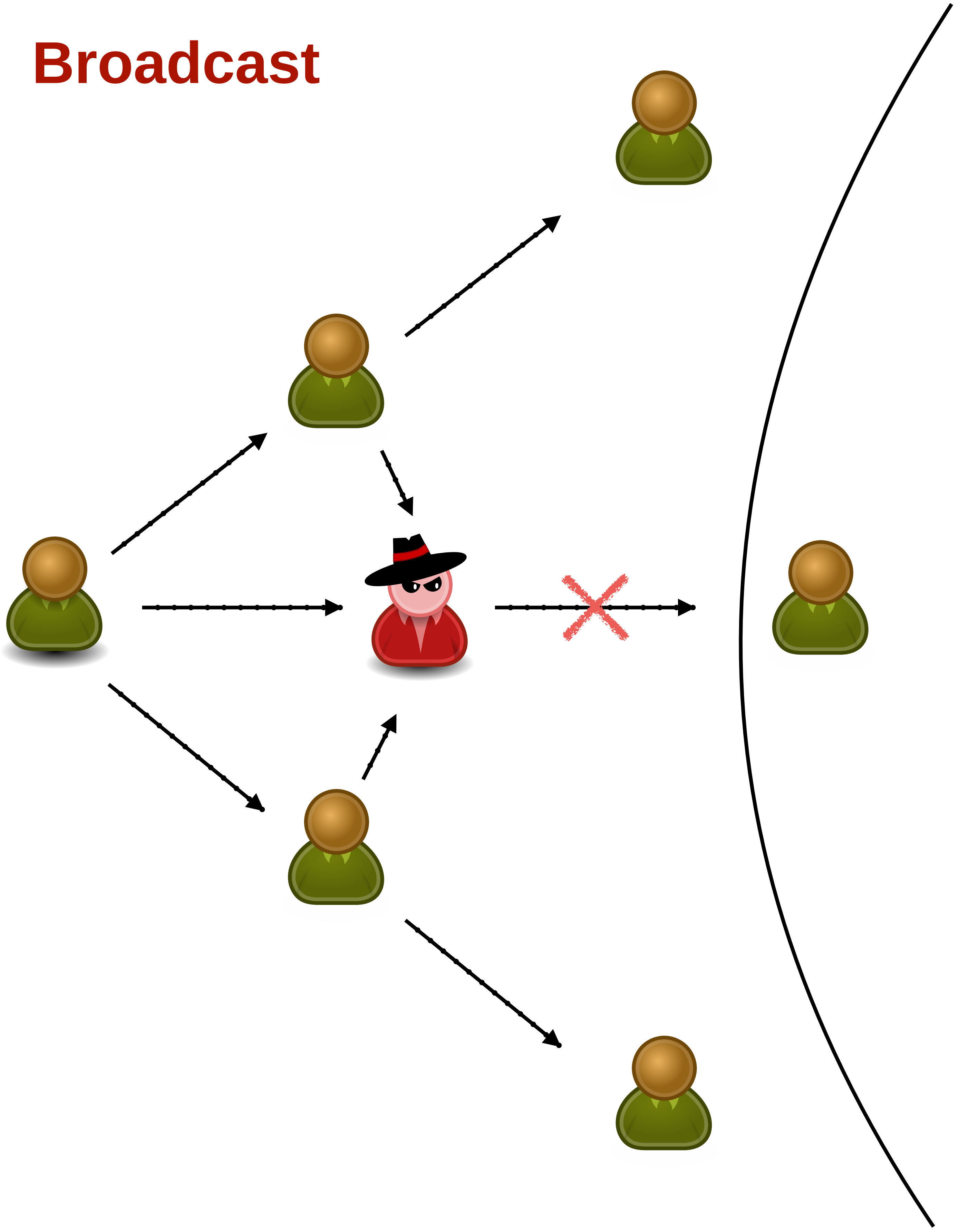


Broadcast of information

Broadcast

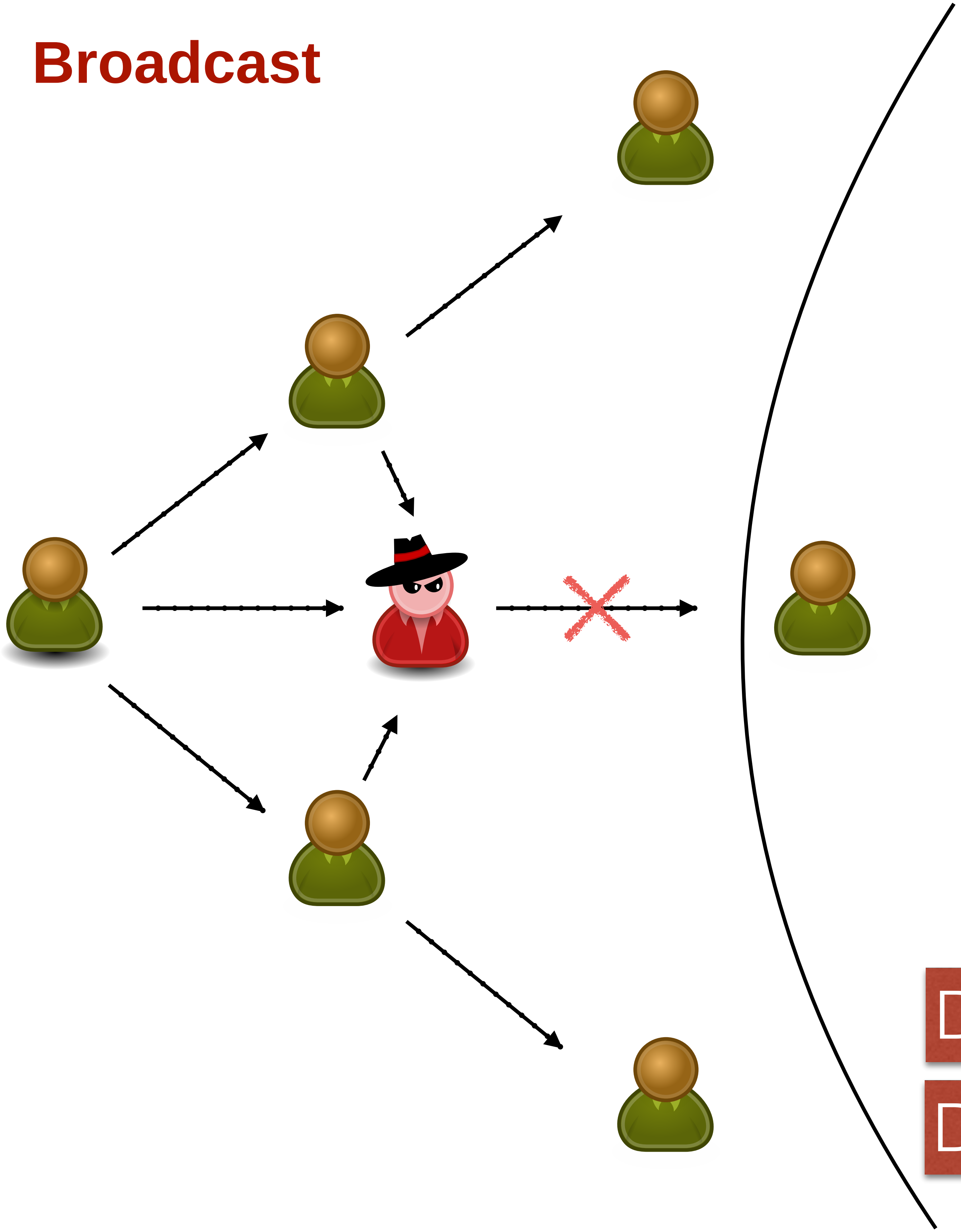


Broadcast



Broadcast of information

Broadcast

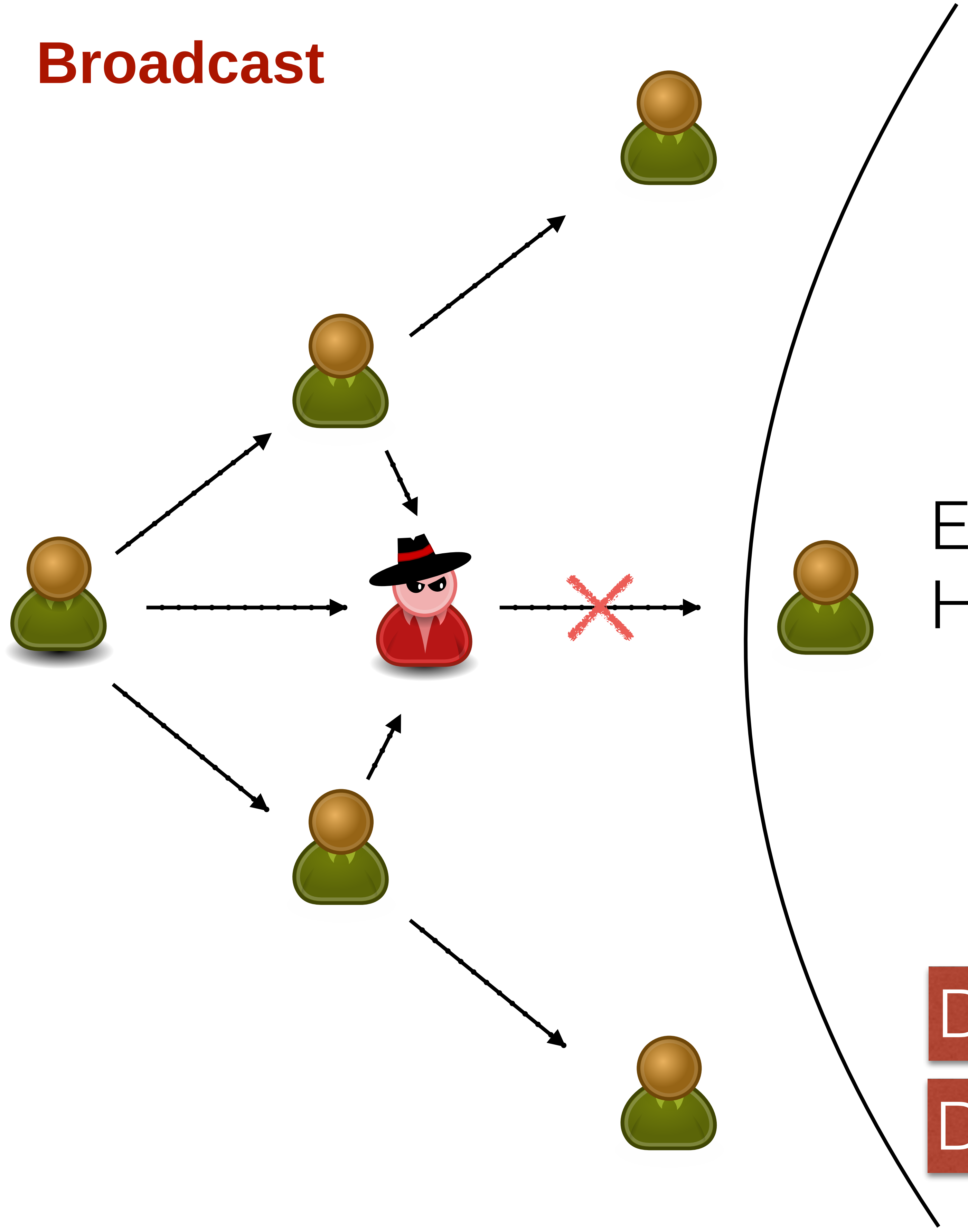


Denial of Service

Double Spending

Broadcast of information

Broadcast



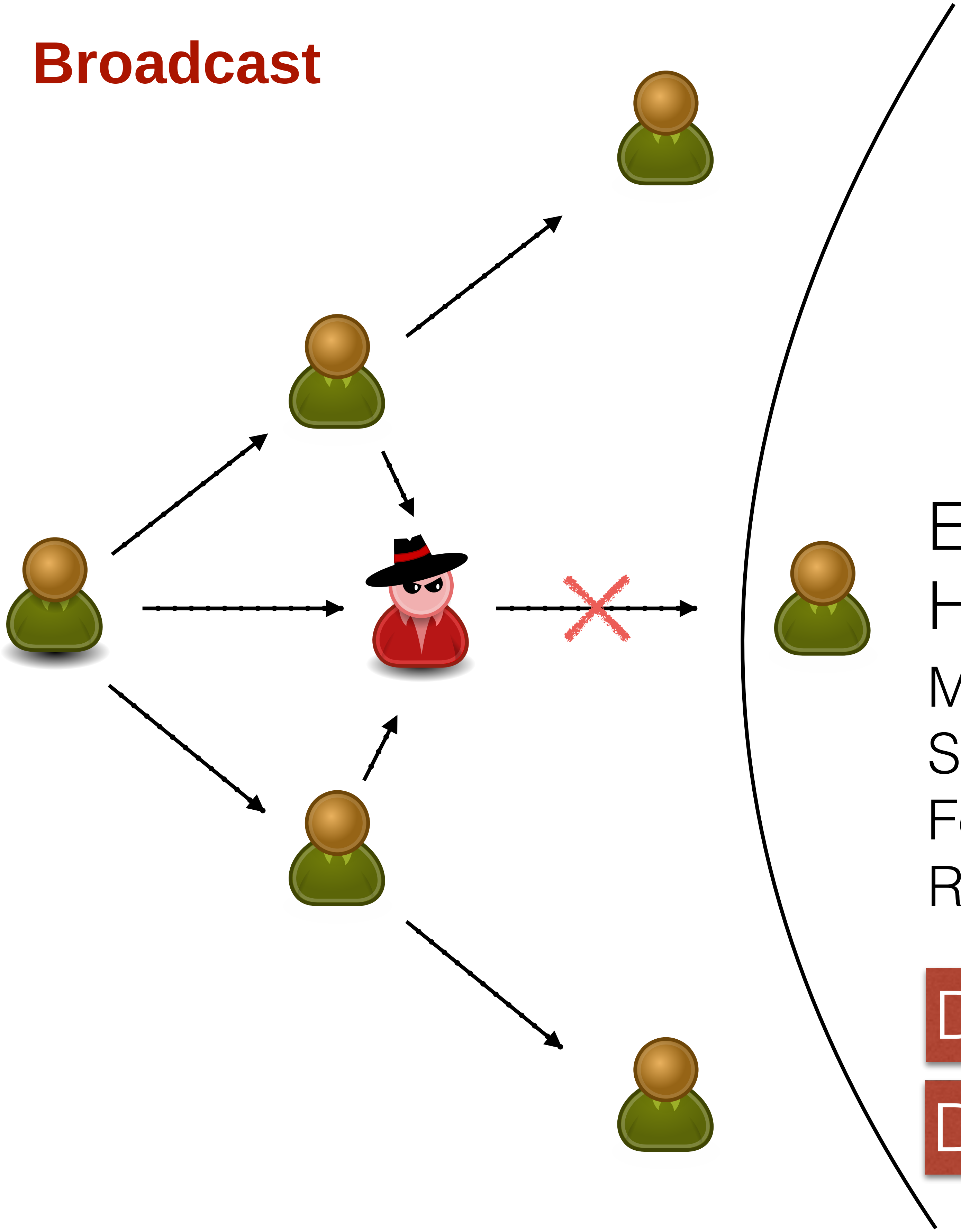
Eclipse attacks
Heilman et al., Usenix ’15

Denial of Service

Double Spending

Broadcast of information

Broadcast



Eclipse attacks
Heilman et al., Usenix ’15

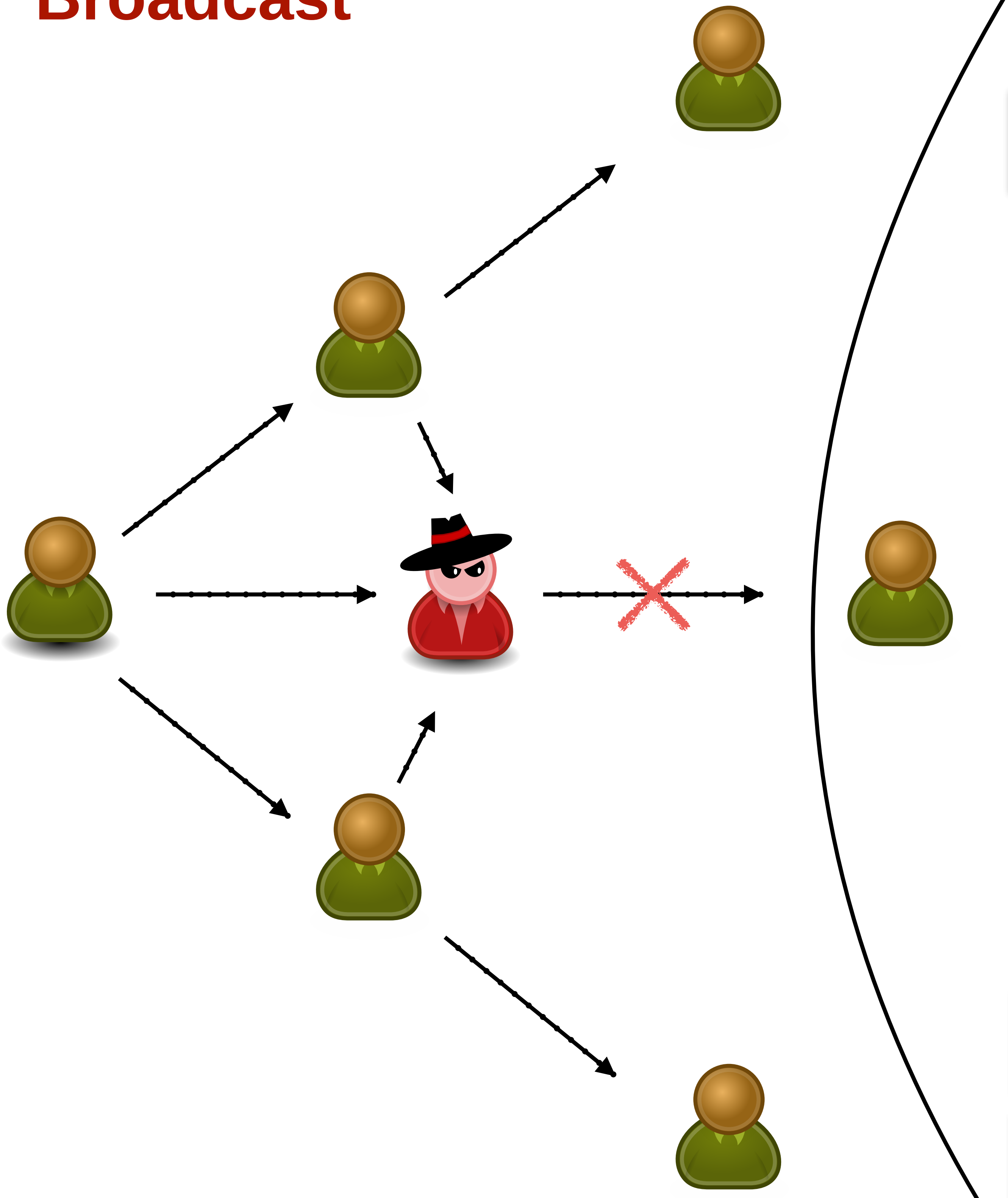
Monopolize connections
Spamming addresses
Forcing node restart
Requires many bots

Denial of Service

Double Spending

Broadcast of information

Broadcast



This paper

1 connection sufficient
No victim restart necessary

Eclipse attacks
Heilman et al., Usenix ’15

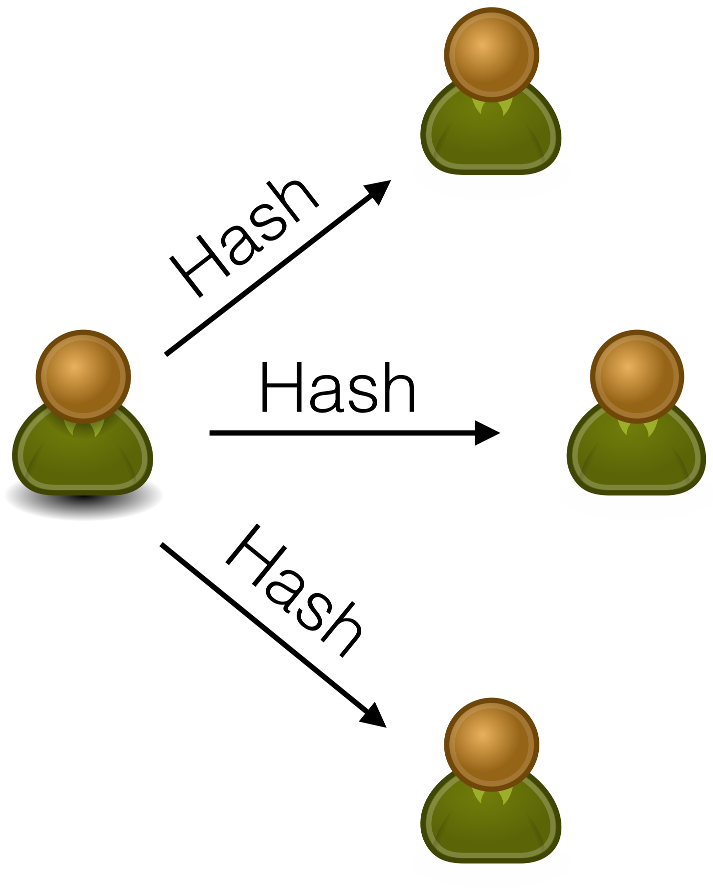
Monopolize connections
Spamming addresses
Forcing node restart
Requires many bots

Denial of Service

Double Spending

Transaction/Block advertisement

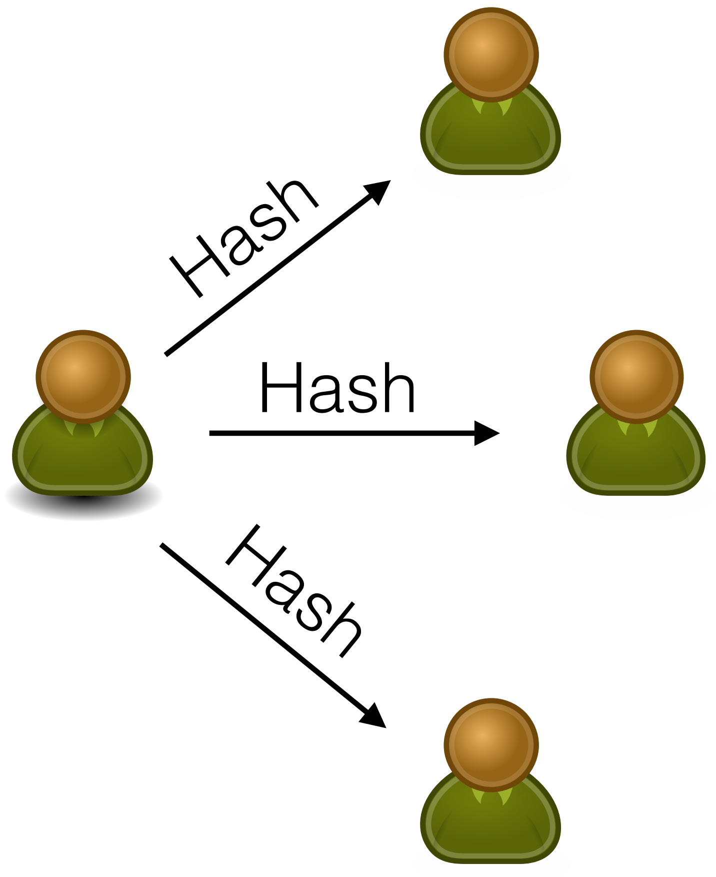
1. Transaction/Block hash broadcast



Broadcast

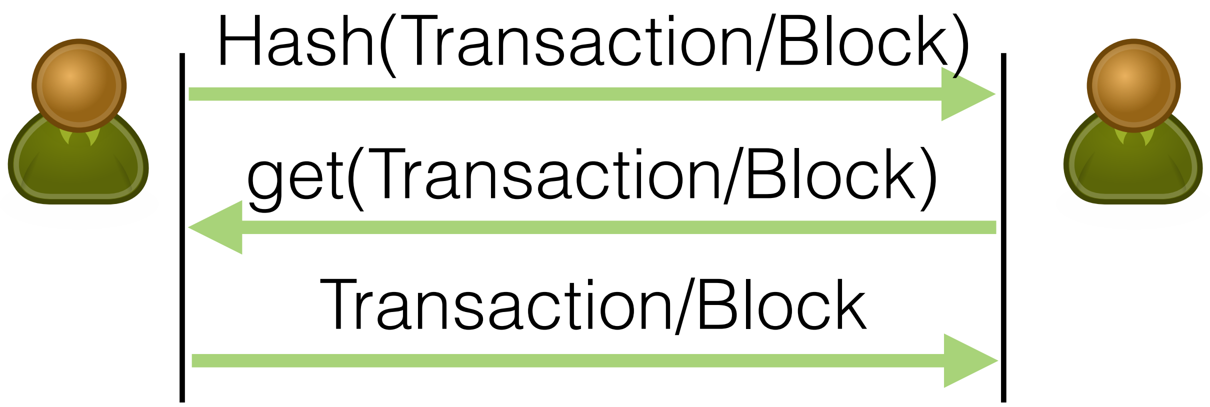
Transaction/Block advertisement

1. Transaction/Block hash broadcast



Broadcast

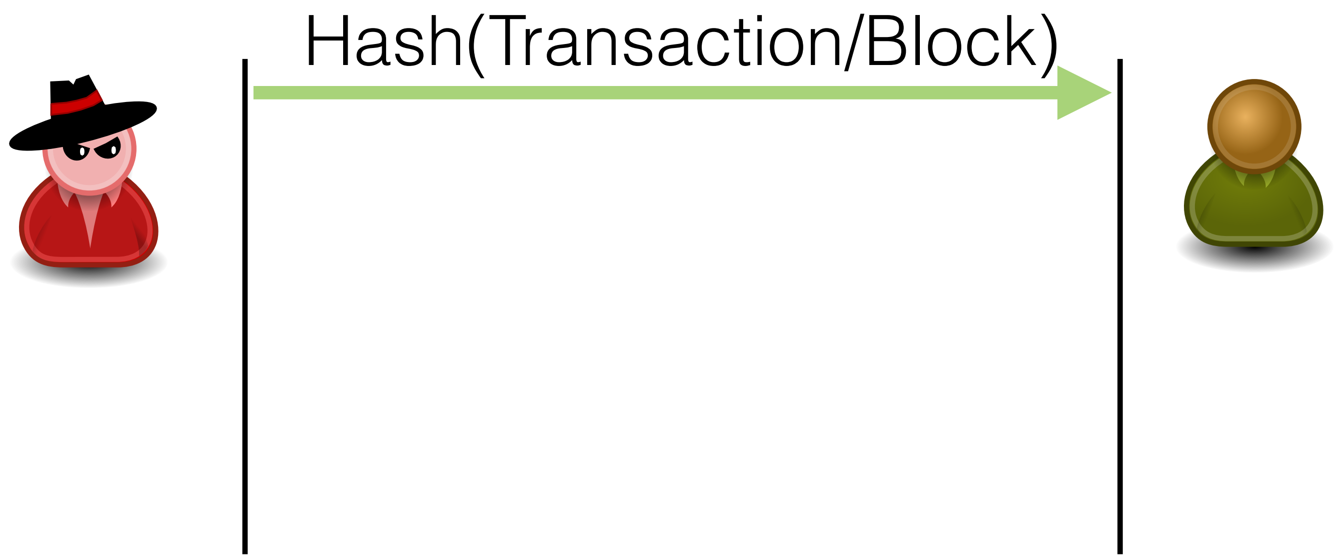
2. Transaction/Block request



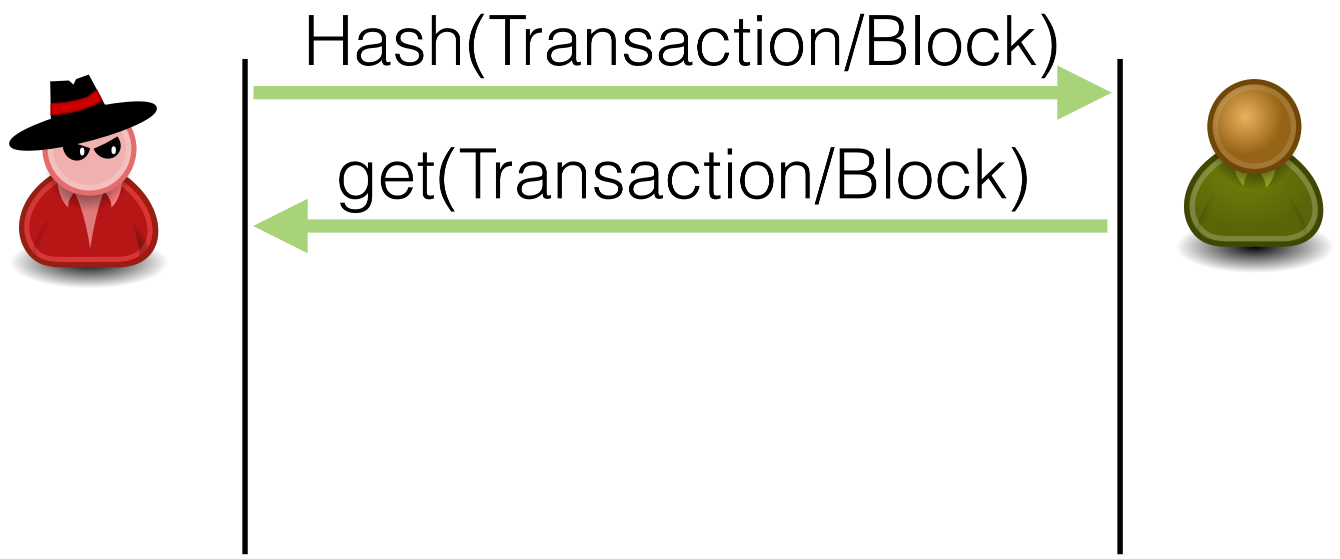
Request from only 1 peer!

Request timeouts

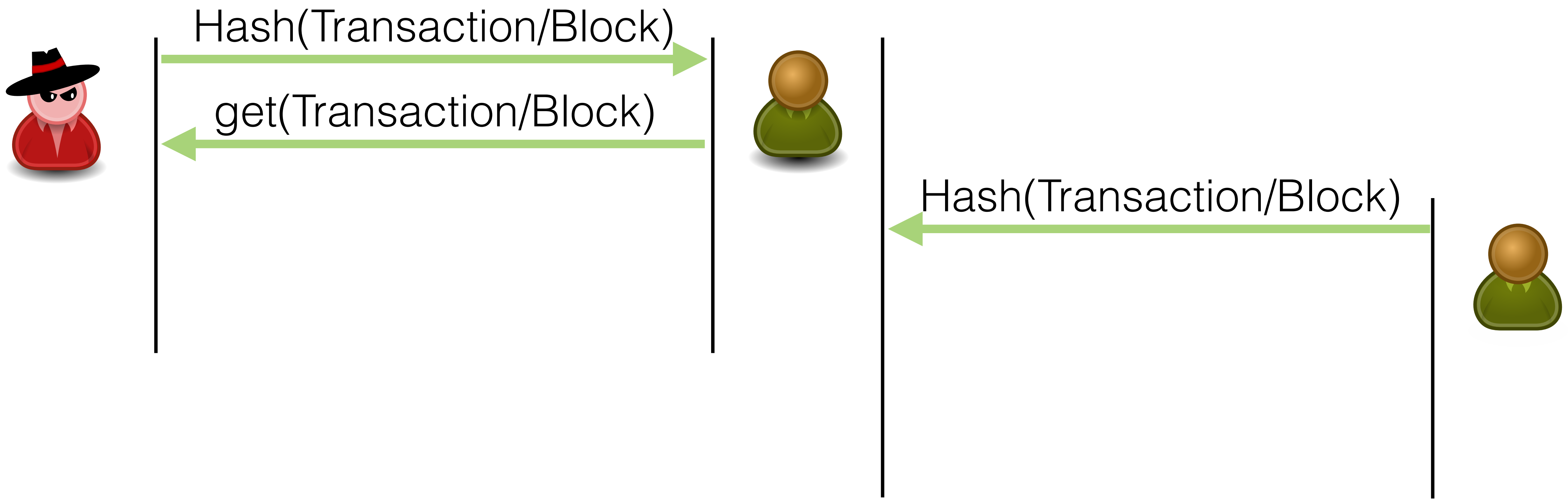
Request timeouts



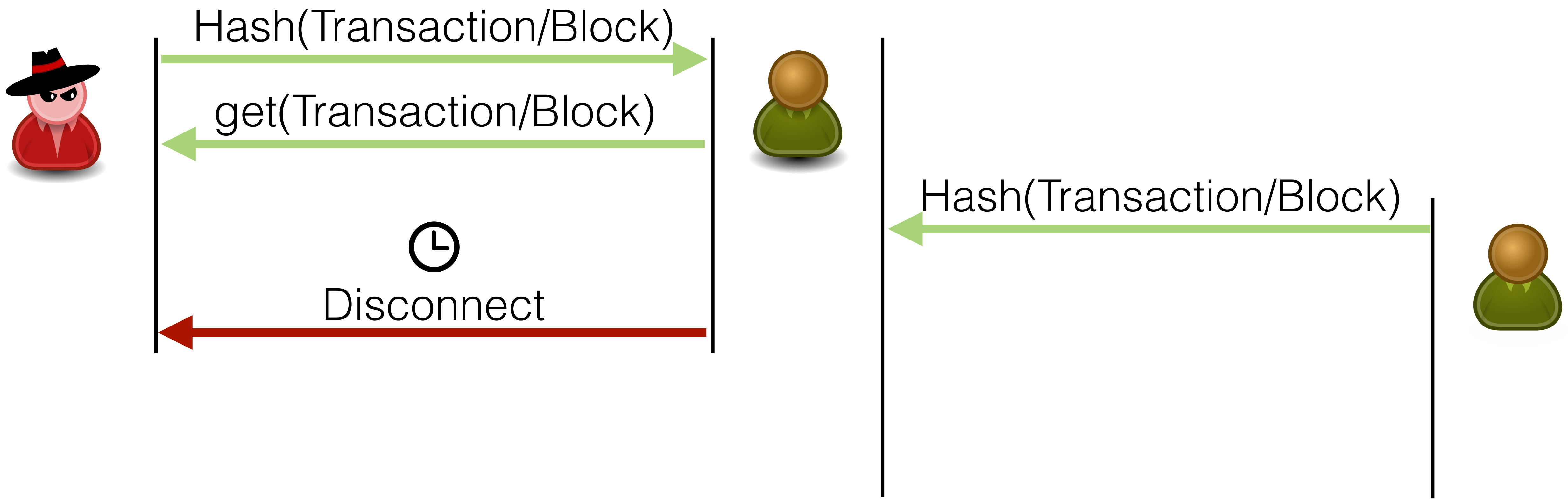
Request timeouts



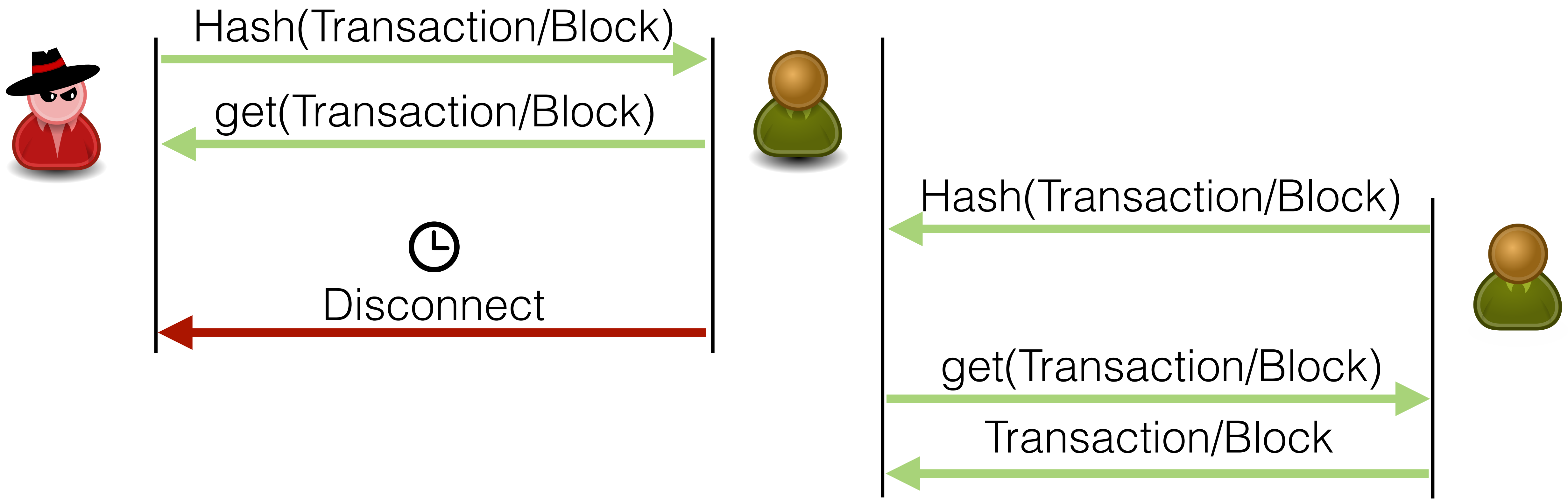
Request timeouts



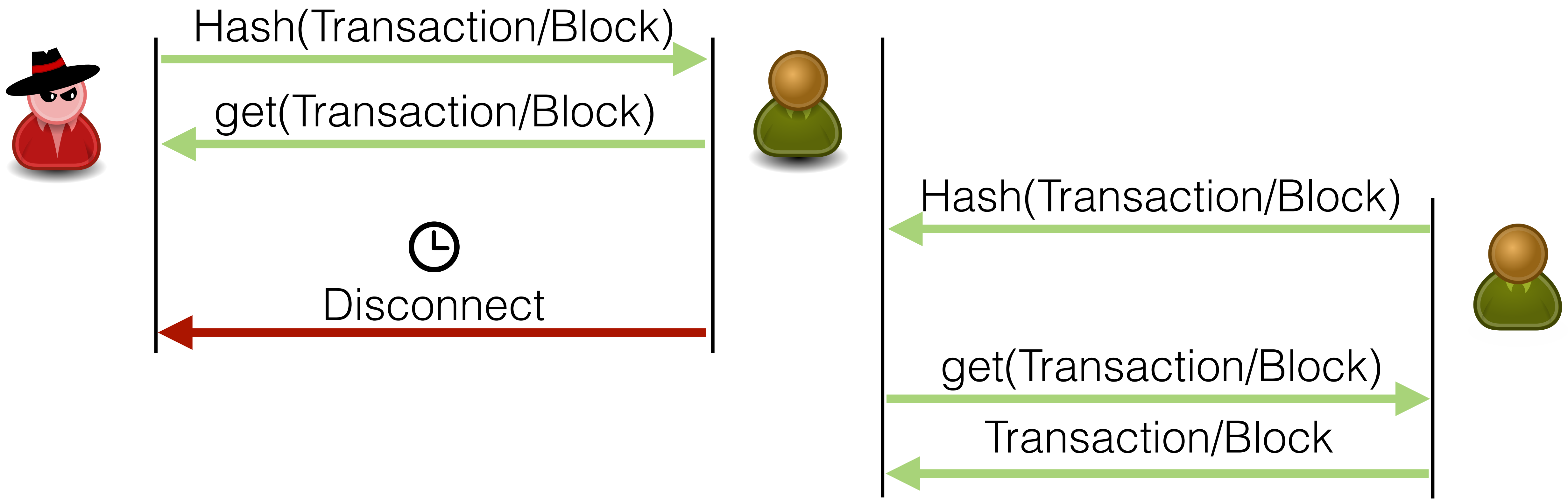
Request timeouts



Request timeouts



Request timeouts



Hash(Transaction/Block)

get(Transaction/Block)

Disconnect

Hash(Transaction/Block)

get(Transaction/Block)

Transaction/Block

Block timeout: 20 minutes
Transaction timeout: 2 minutes

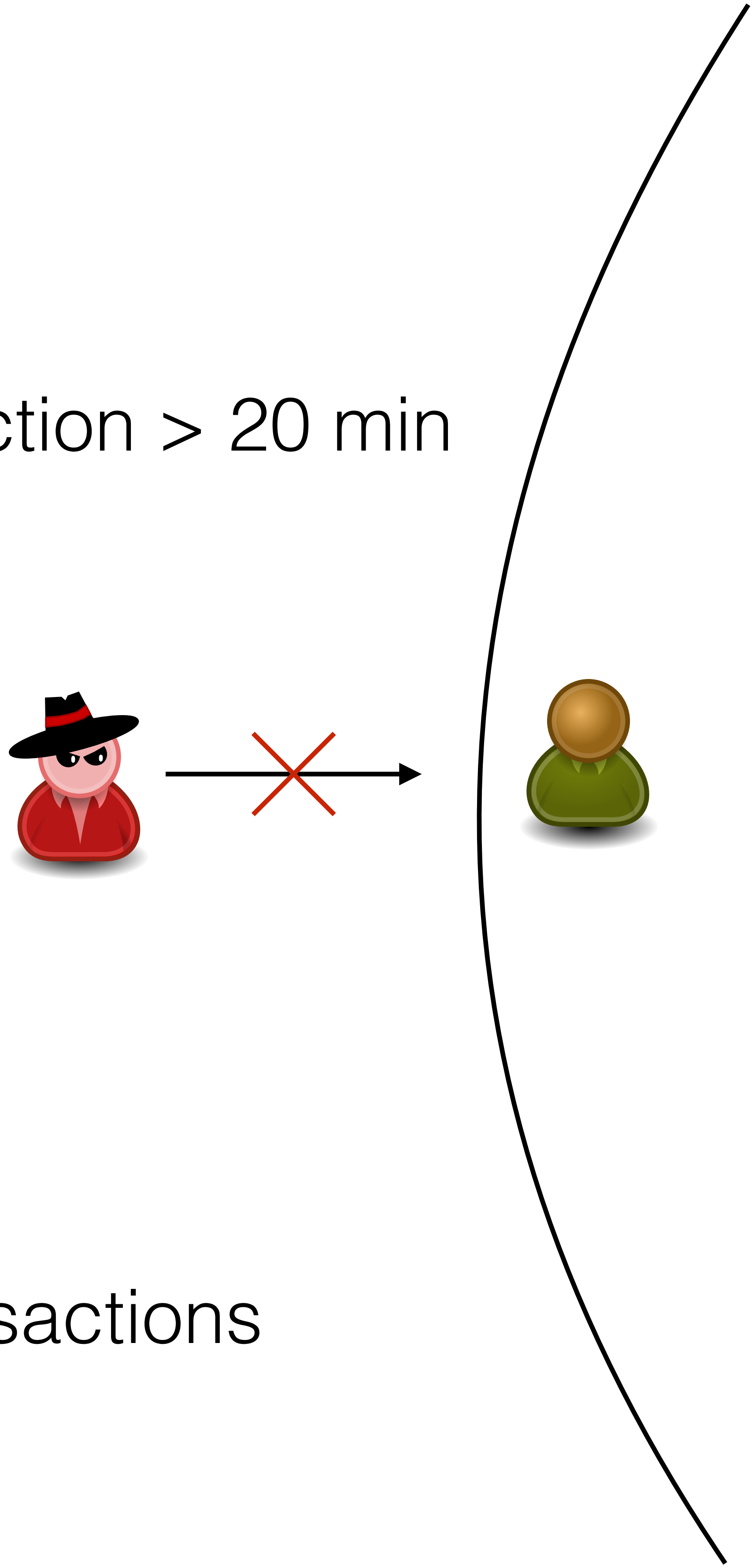
Contributions

Adversary

- Blinds victim from blocks and transaction > 20 min
- Experimental validation

Impact

- **Double spend transactions**
- Aggravated selfish mining
- **Network wide Denial of Service**



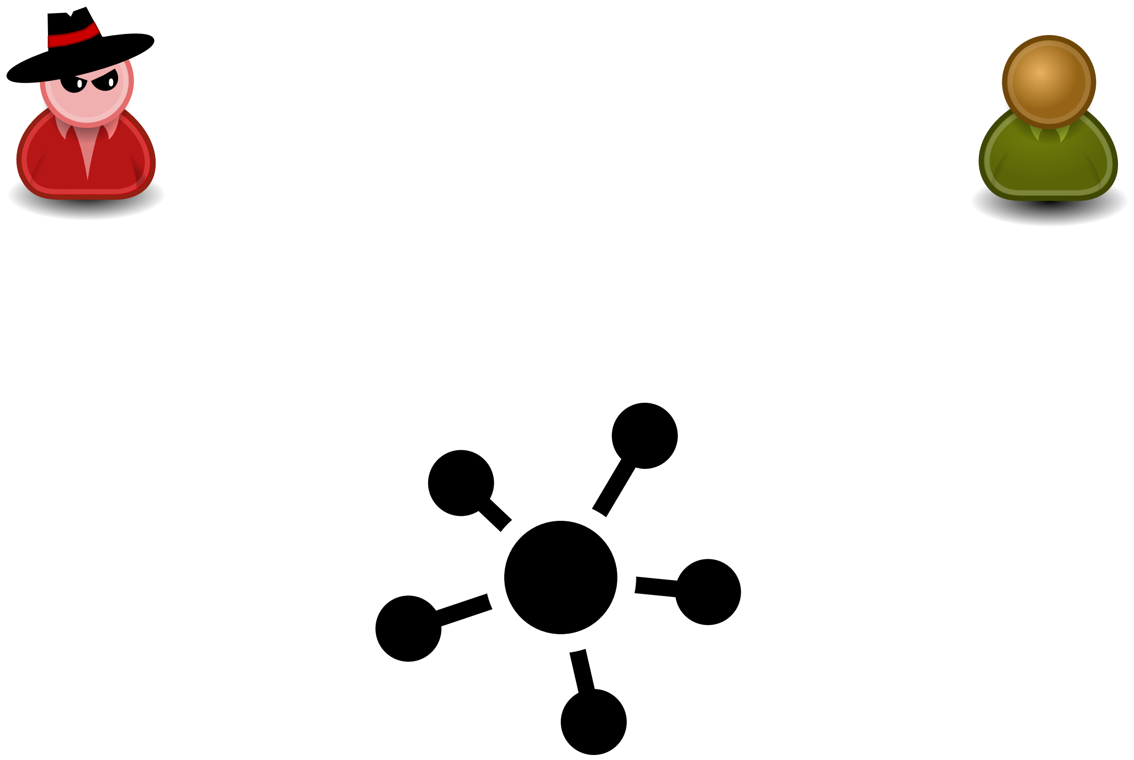
Mitigations

- **Hardening measures**
- Estimate waiting time for secure transactions

Delaying information delivery

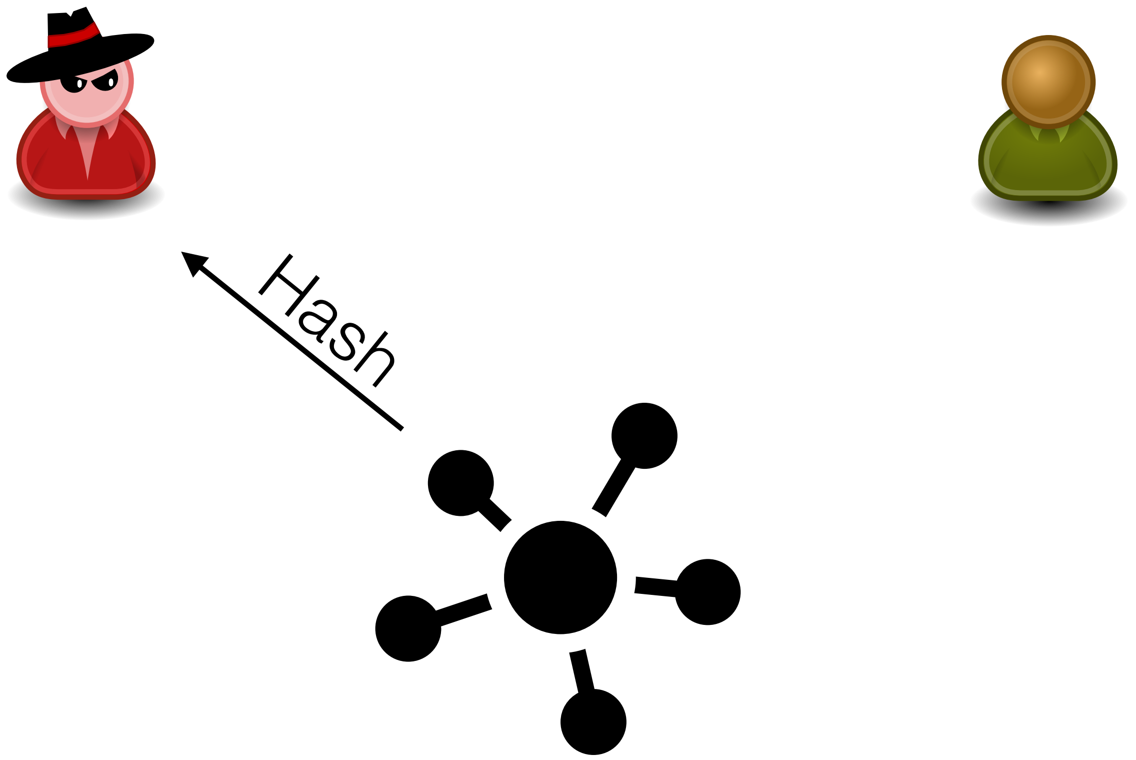
Necessary requirements

1. Must be **first** peer to advertise Transaction/Block



Necessary requirements

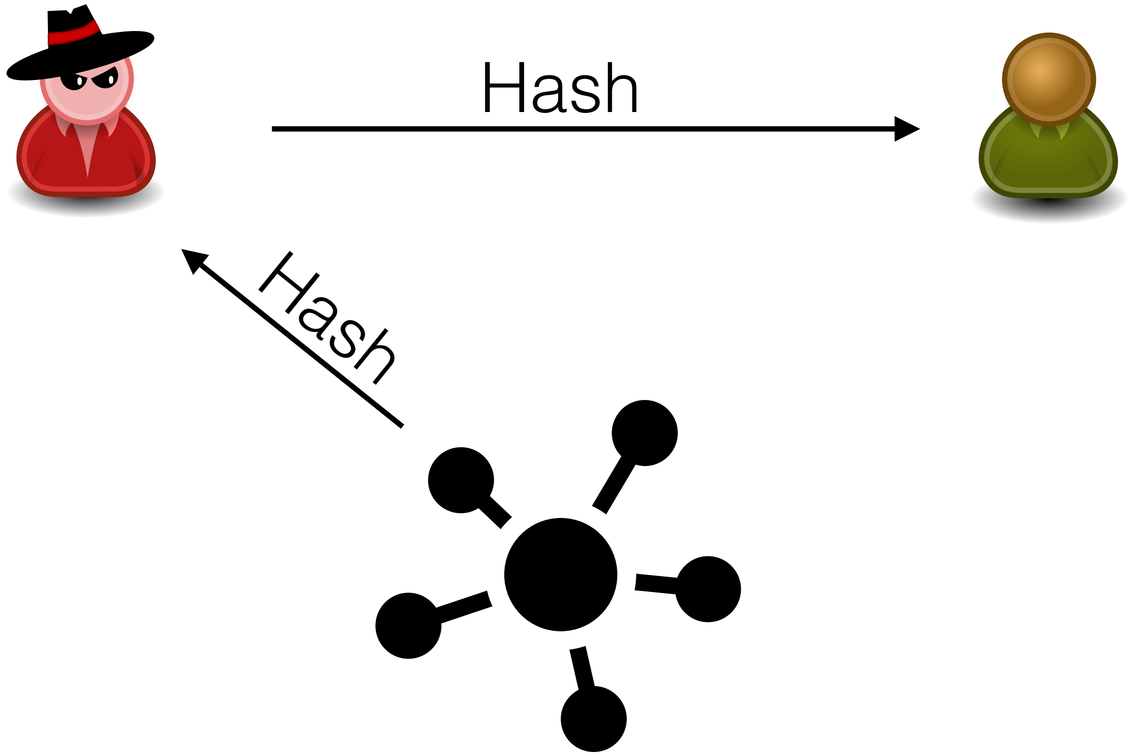
1. Must be **first** peer to advertise Transaction/Block



Delaying information delivery

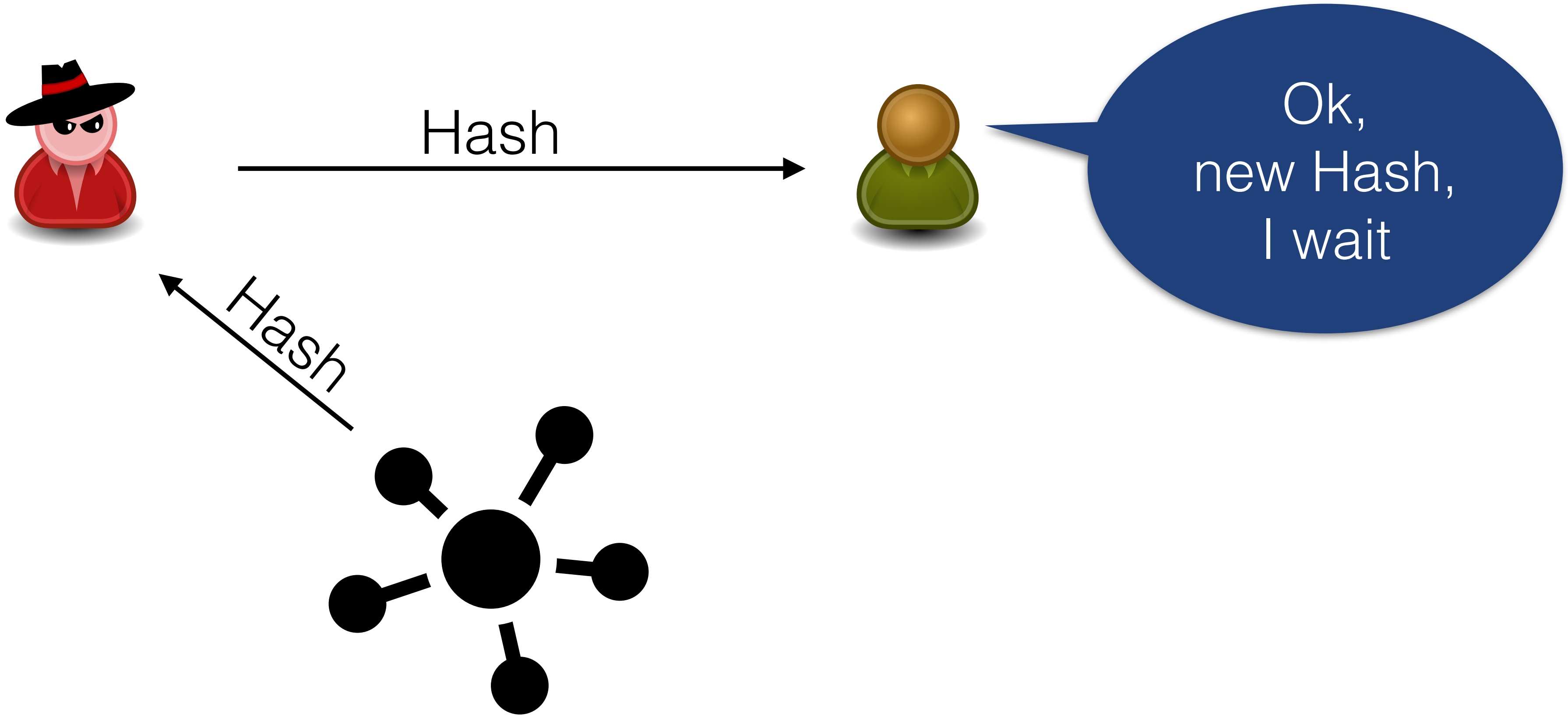
Necessary requirements

1. Must be **first** peer to advertise Transaction/Block



Necessary requirements

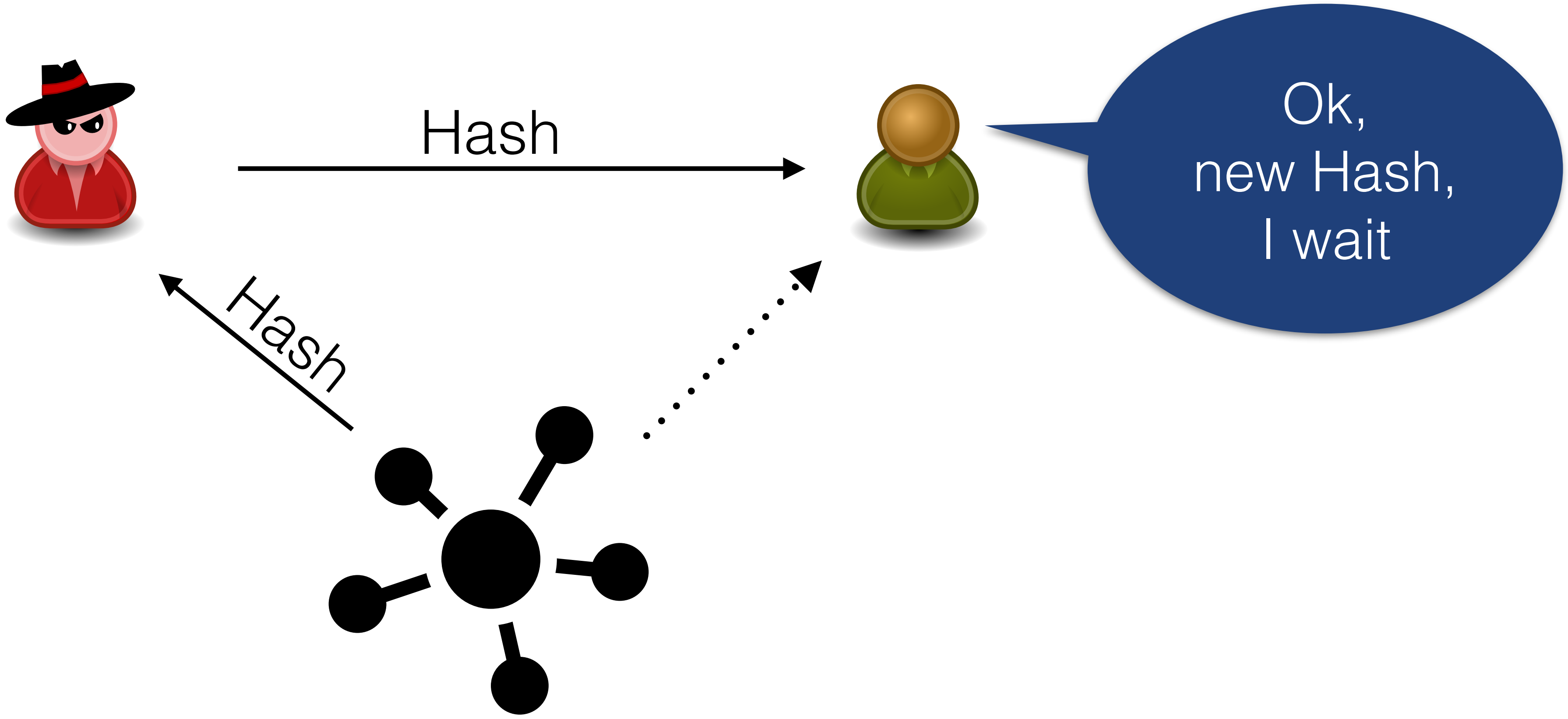
1. Must be **first** peer to advertise Transaction/Block



Delaying information delivery

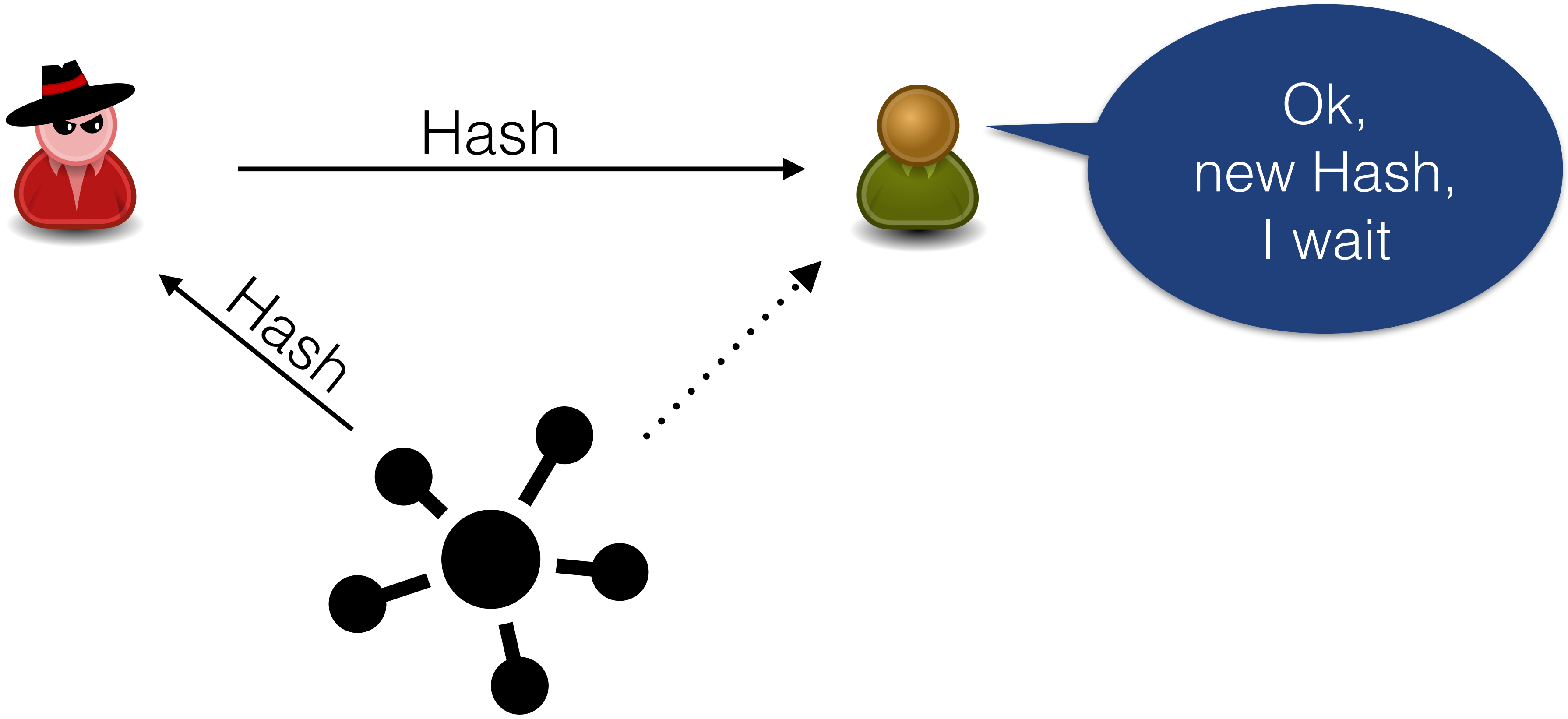
Necessary requirements

1. Must be **first** peer to advertise Transaction/Block



Necessary requirements

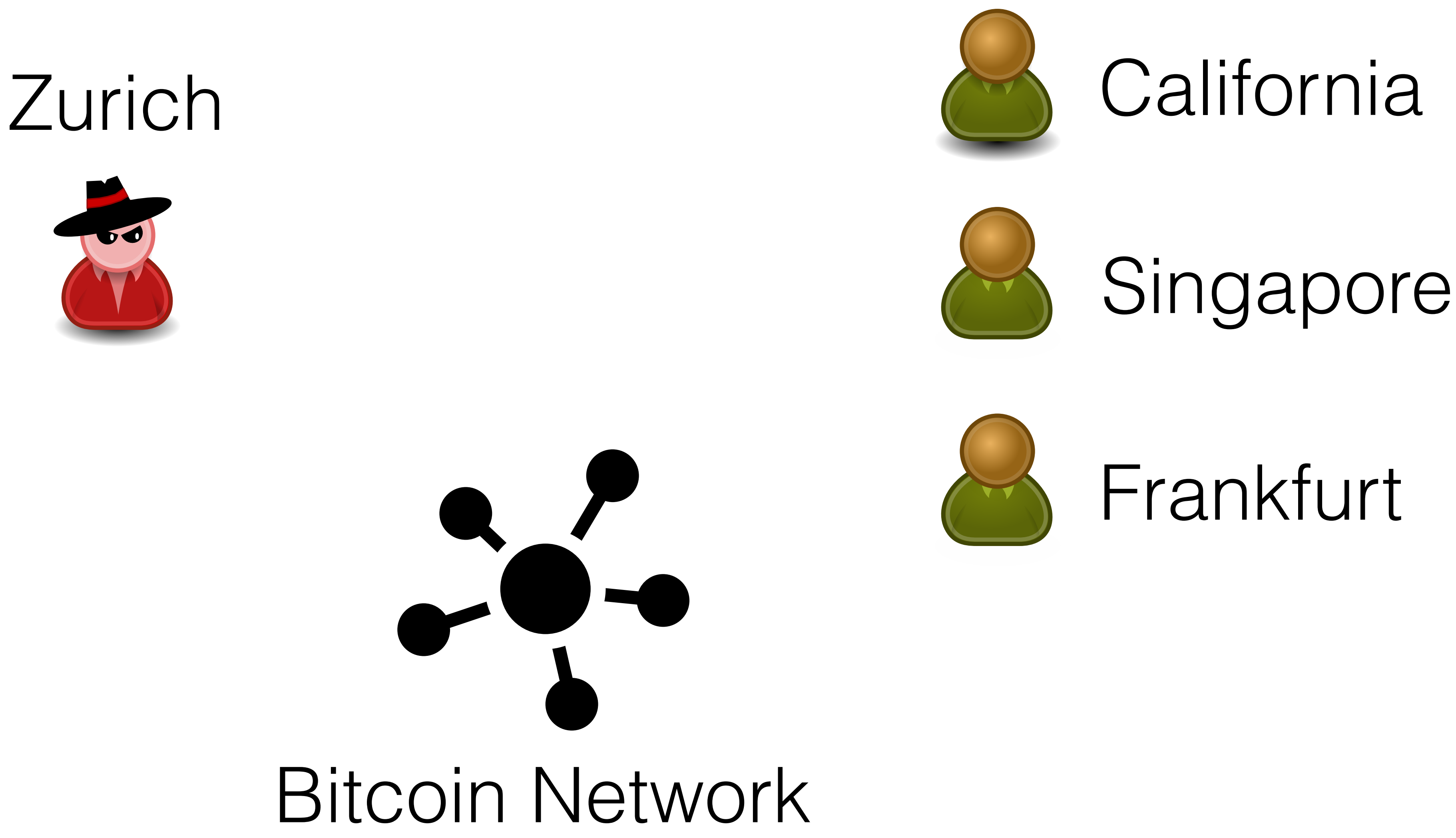
1. Must be **first** peer to advertise Transaction/Block



2. Victim should wait
   - Block timeout: 20 minutes
   - Transaction timeout: 2 minutes

Experimental validation

Being First



Zurich

California

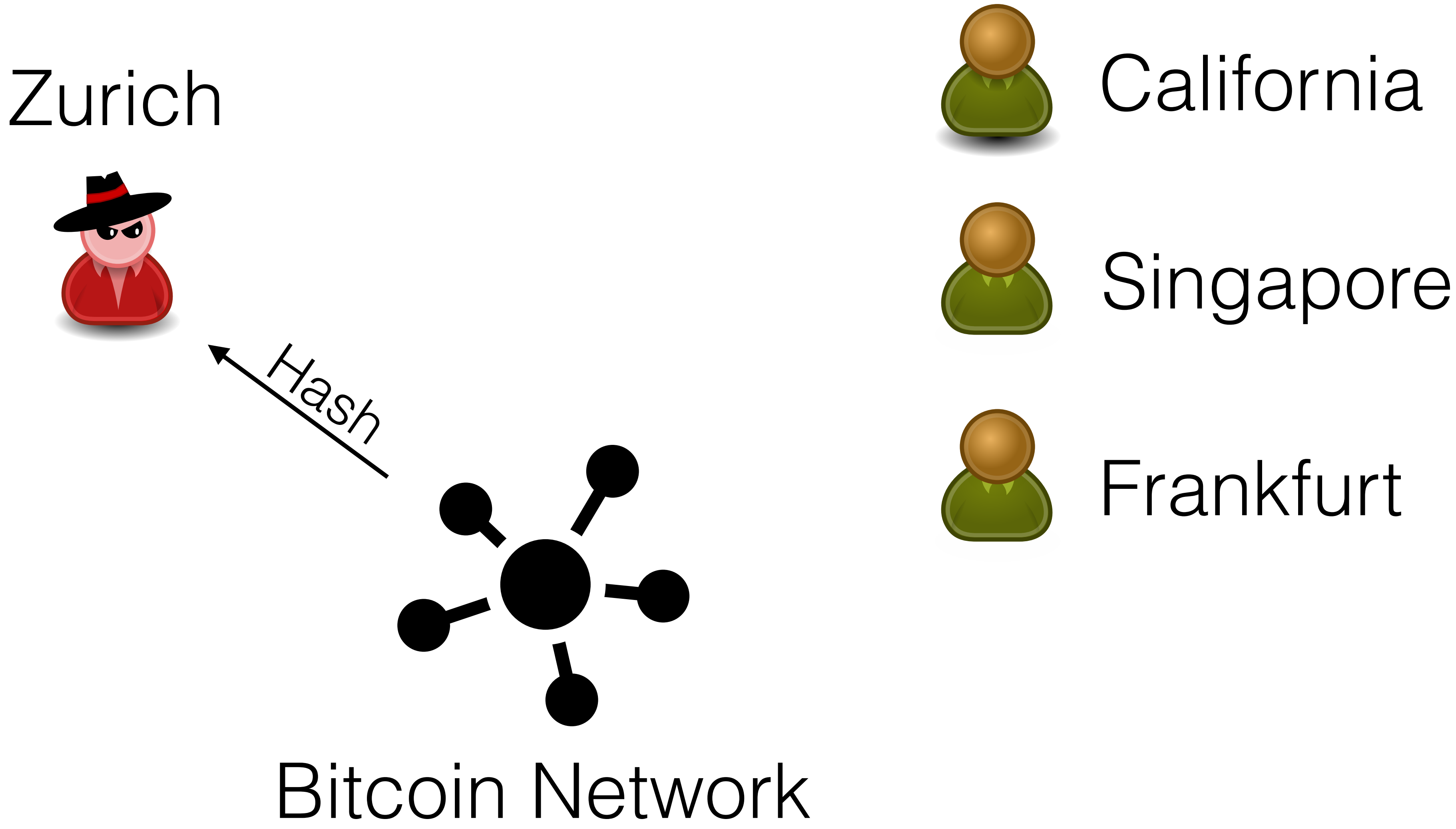
Singapore

Frankfurt

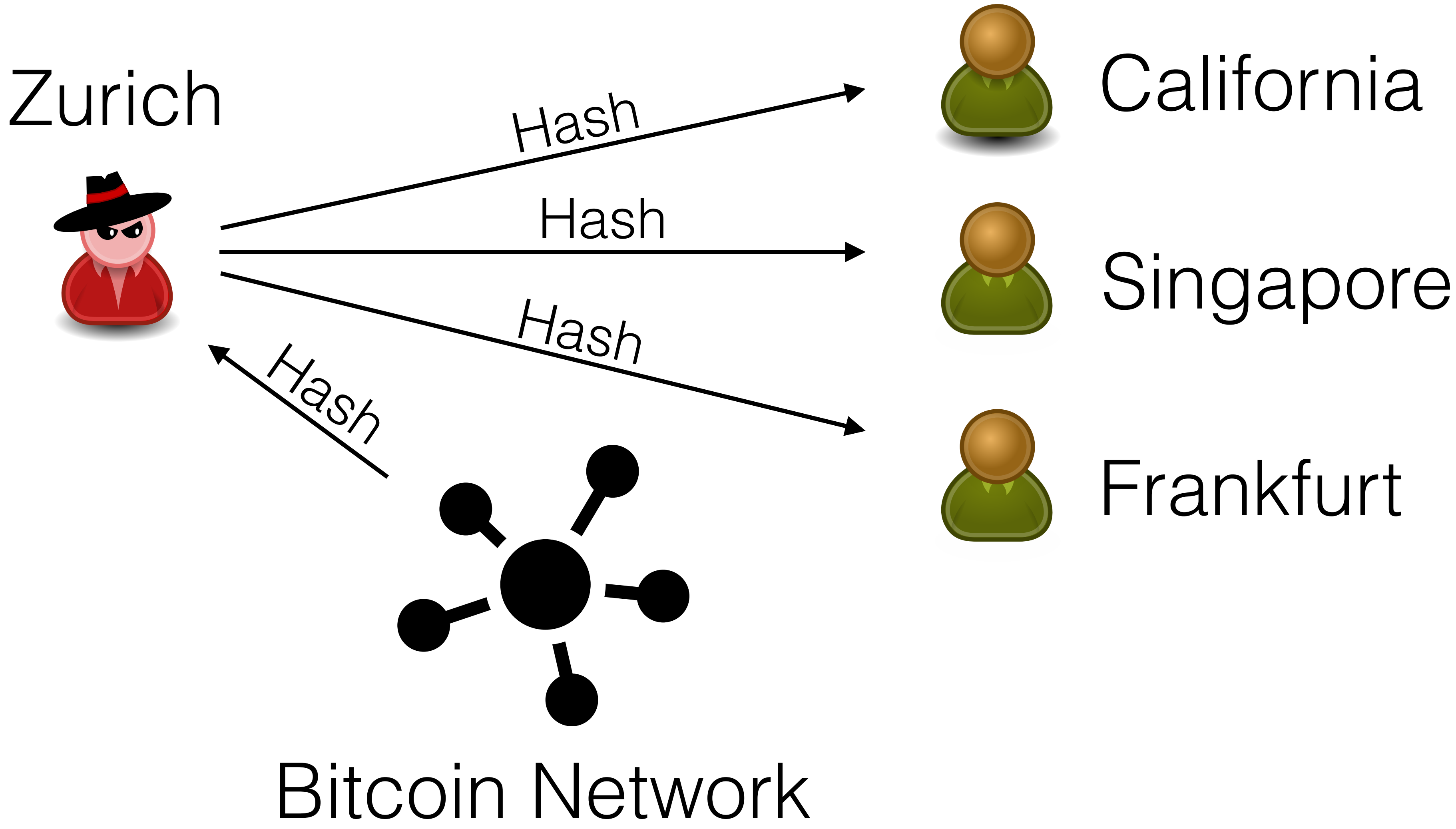
Bitcoin Network

Experimental validation

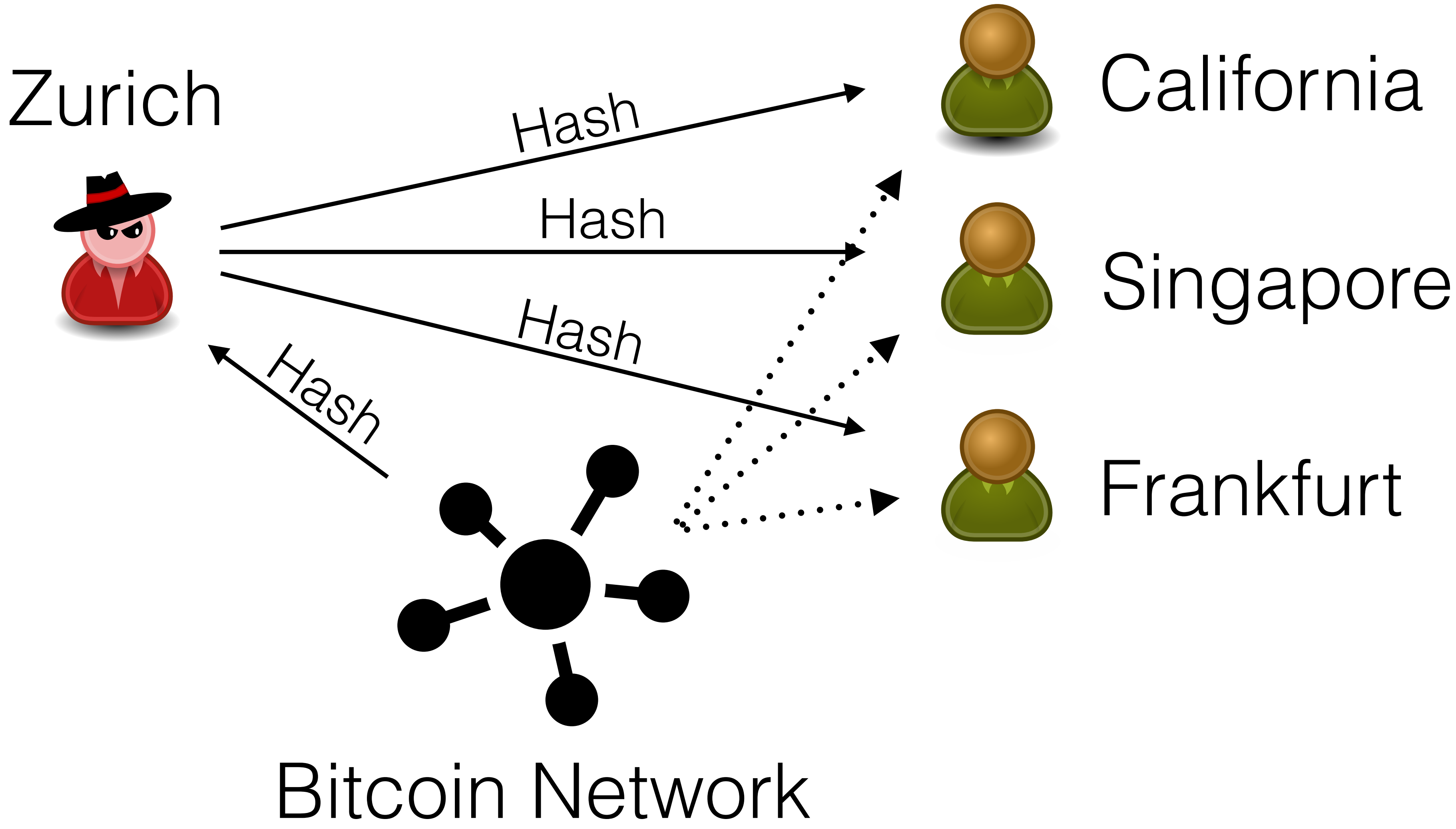
Being First



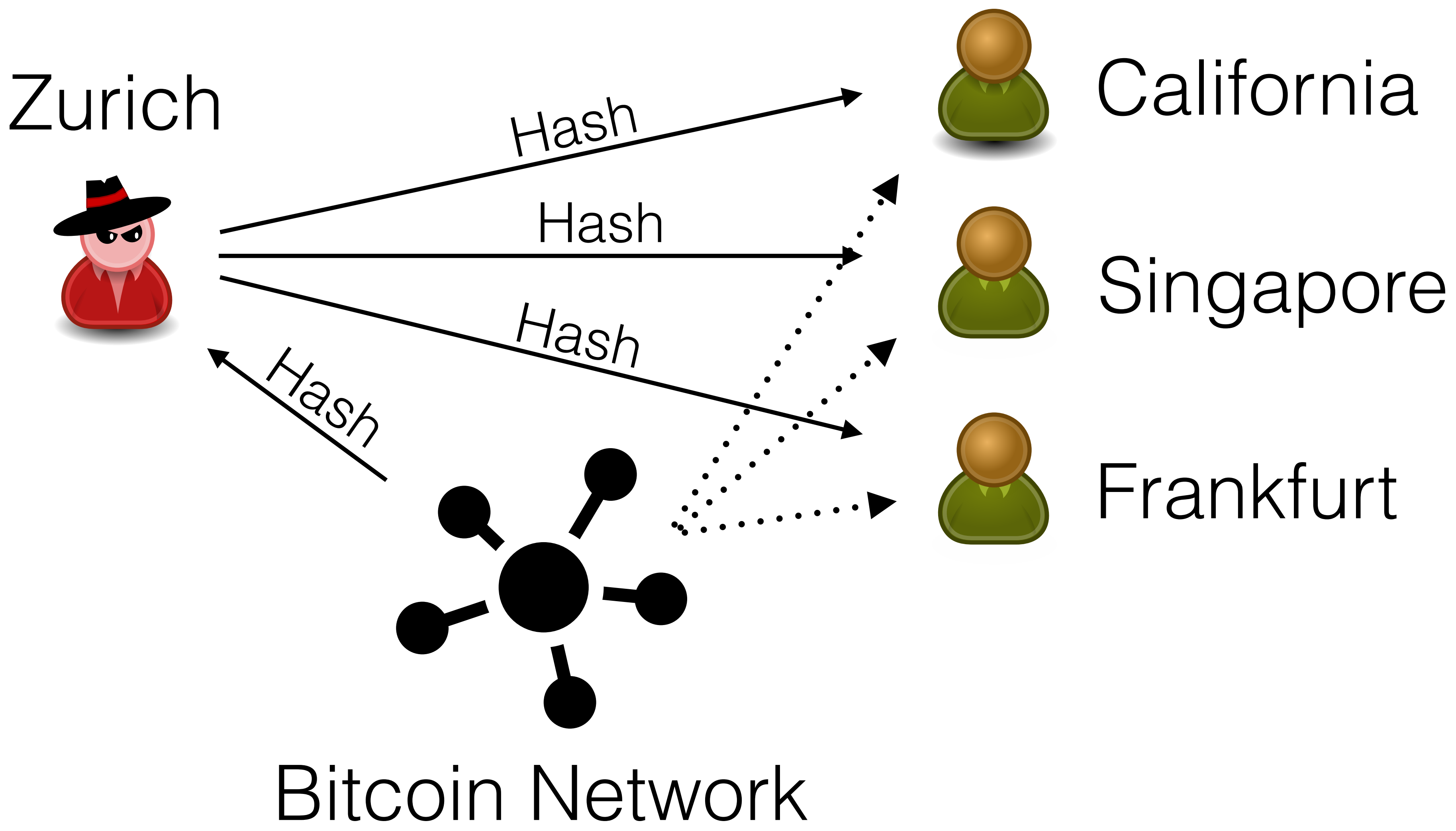
Being First


Zurich
Hash
Hash
Hash
Hash
California
Singapore
Frankfurt
Bitcoin Network

Being First


Zurich
Hash
Hash
Hash
Hash
California
Singapore
Frankfurt
Bitcoin Network

Being First



Connections of Adversary	40	80	200	800
Connections of Victim	40	40	40	40
Average success in being first	0.44± 0.14	0.57± 0.20	0.80± 0.14	**0.89± 0.07**

Experimental validation

ETH zürich

Waiting

Transactions

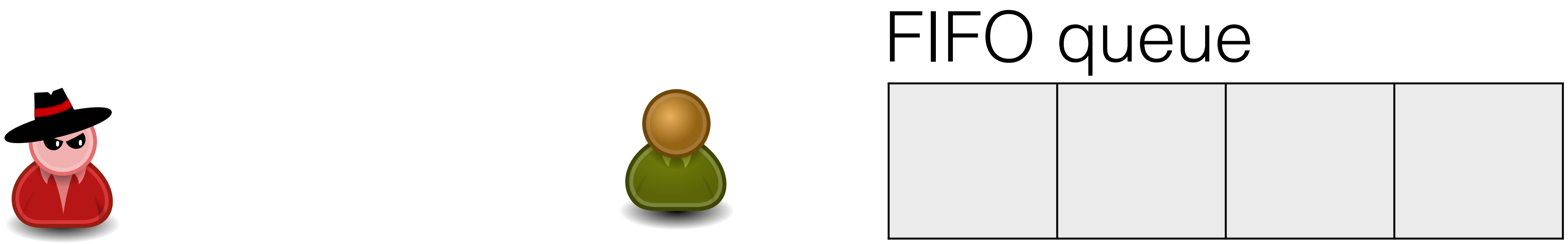
- After 2 minutes request from other peer (FIFO)

Experimental validation

Waiting

Transactions

- After 2 minutes request from other peer (FIFO)



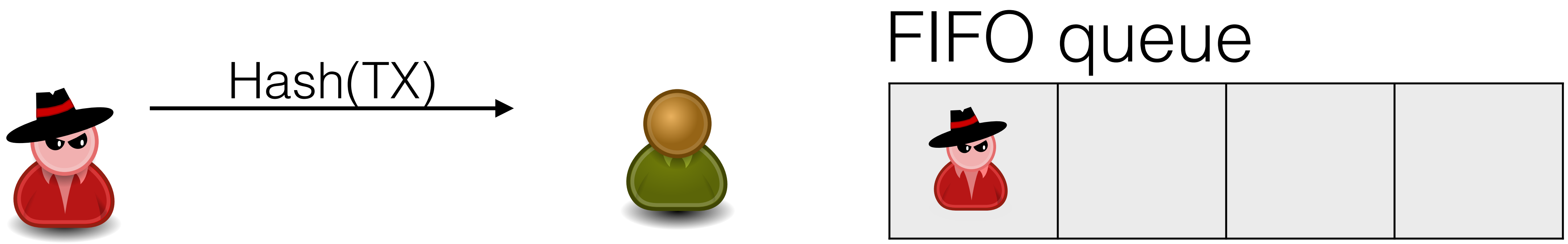
FIFO queue

Experimental validation

Waiting

Transactions

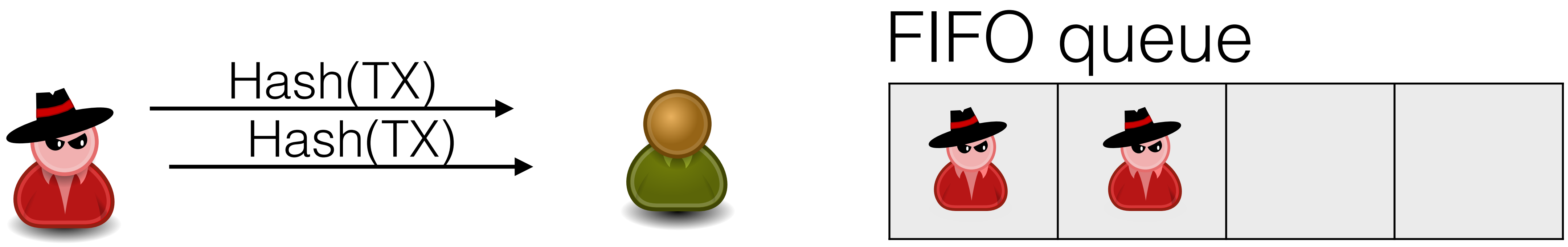
- After 2 minutes request from other peer (FIFO)



Waiting

Transactions

- After 2 minutes request from other peer (FIFO)

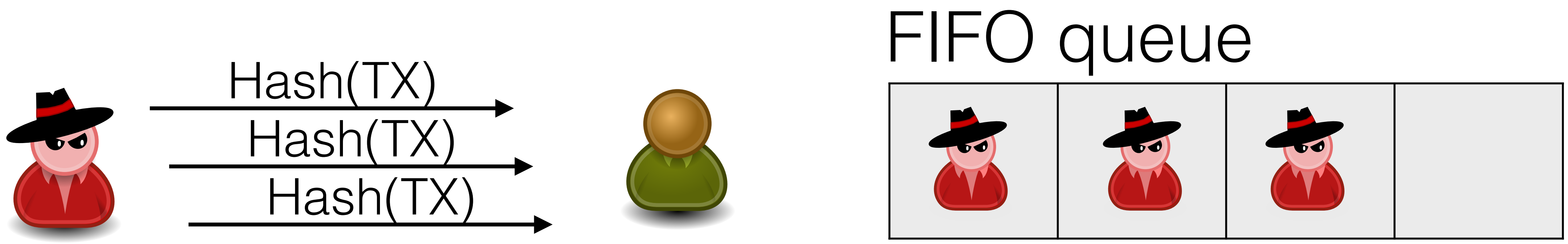


Experimental validation

Waiting

Transactions

- After 2 minutes request from other peer (FIFO)

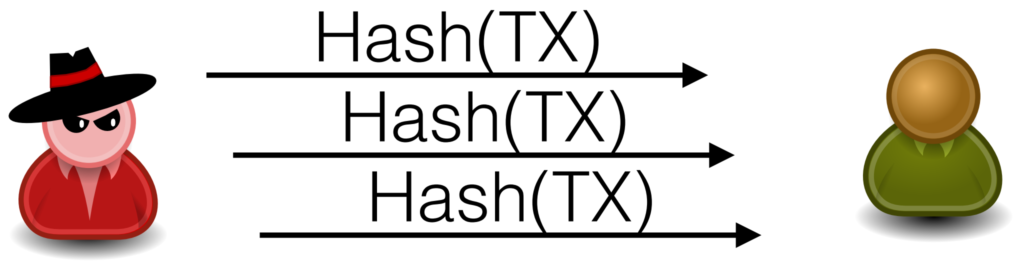


Experimental validation

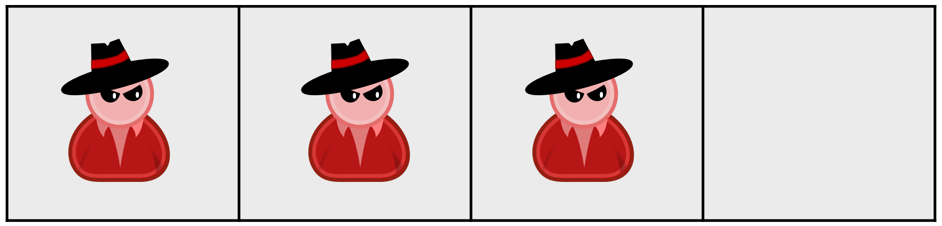
Waiting

Transactions

- After 2 minutes request from other peer (FIFO)



FIFO queue



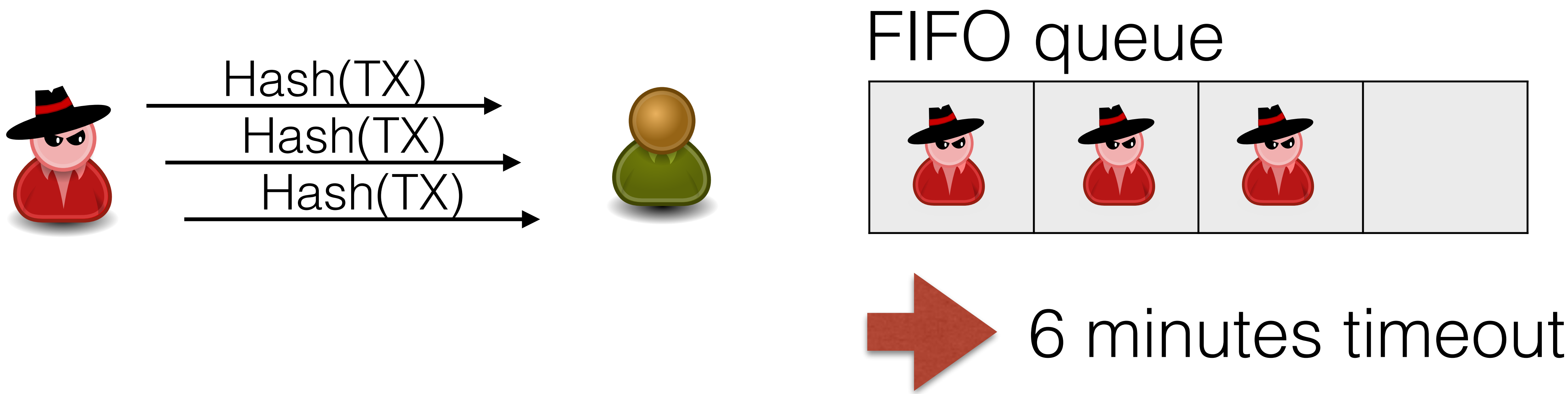
➡ 6 minutes timeout

Experimental validation

Waiting

Transactions

- After 2 minutes request from other peer (FIFO)



FIFO queue

6 minutes timeout

Blocks

- After 20 minutes disconnect and do nothing
- If received header, disconnect and request block from another peer

Delaying information delivery

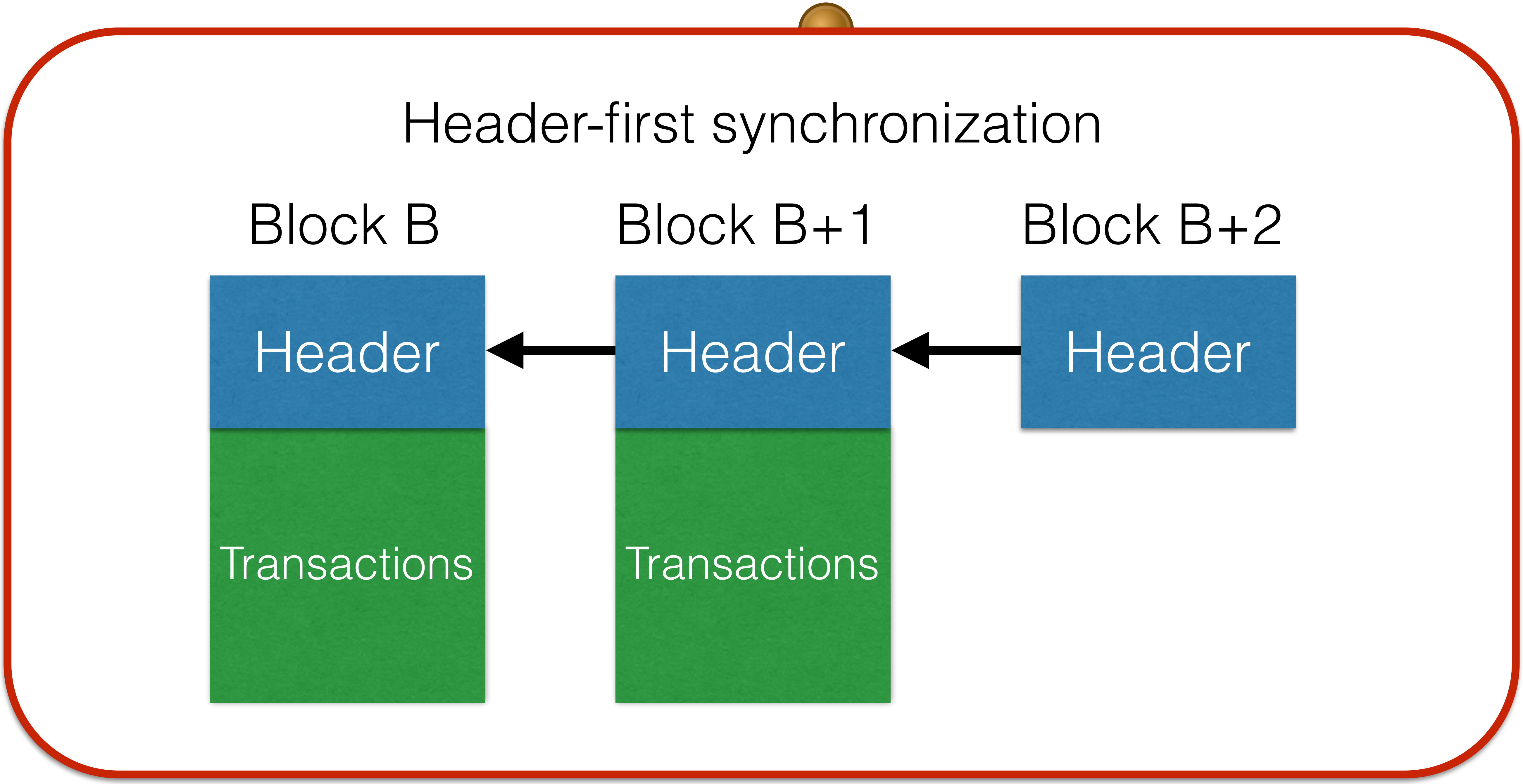
Block delivery time > 20 min

1. Requirement for victim

   - Must not receive block header

   - Must not receive version message

Block delivery time > 20 min



Delaying information delivery

Block delivery time > 20 min

1. Requirement for victim

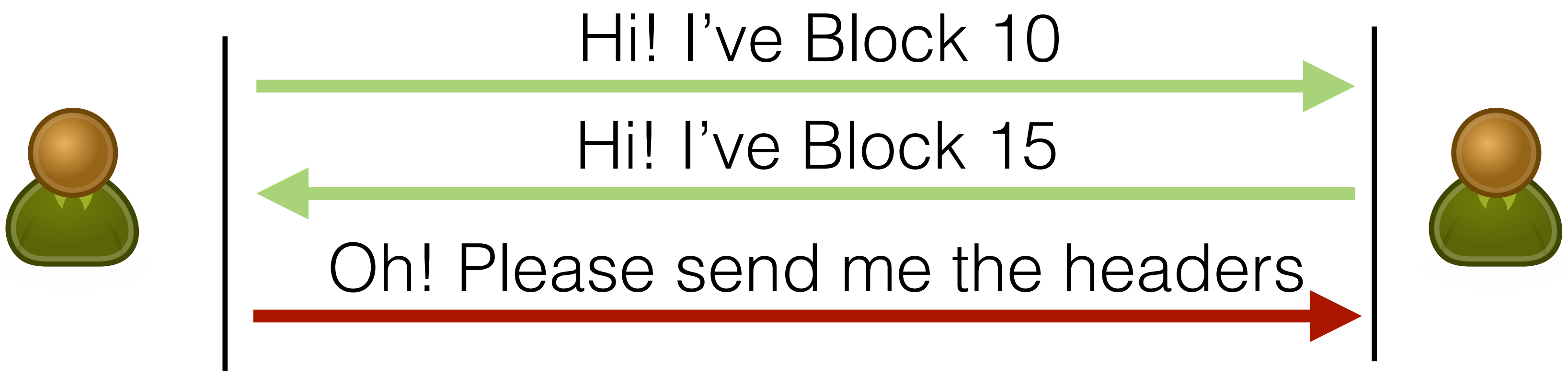
   - Must not receive block header

   - Must not receive version message

Delaying information delivery

Block delivery time > 20 min

Version message exchange
on connection initiation



Hi! I’ve Block 10

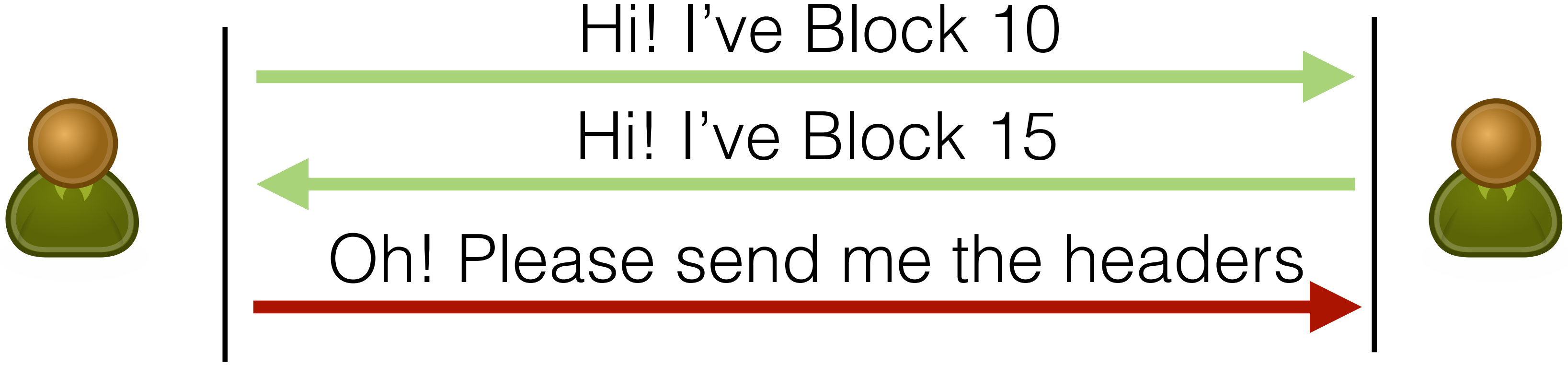
Hi! I’ve Block 15

Oh! Please send me the headers

Delaying information delivery

Block delivery time > 20 min

Version message exchange
on connection initiation



Occupy all open connection slots
No new connections

Delaying information delivery

Block delivery time > 20 min

1. Requirement for victim

   - Must not receive block header

   - Must not receive version message

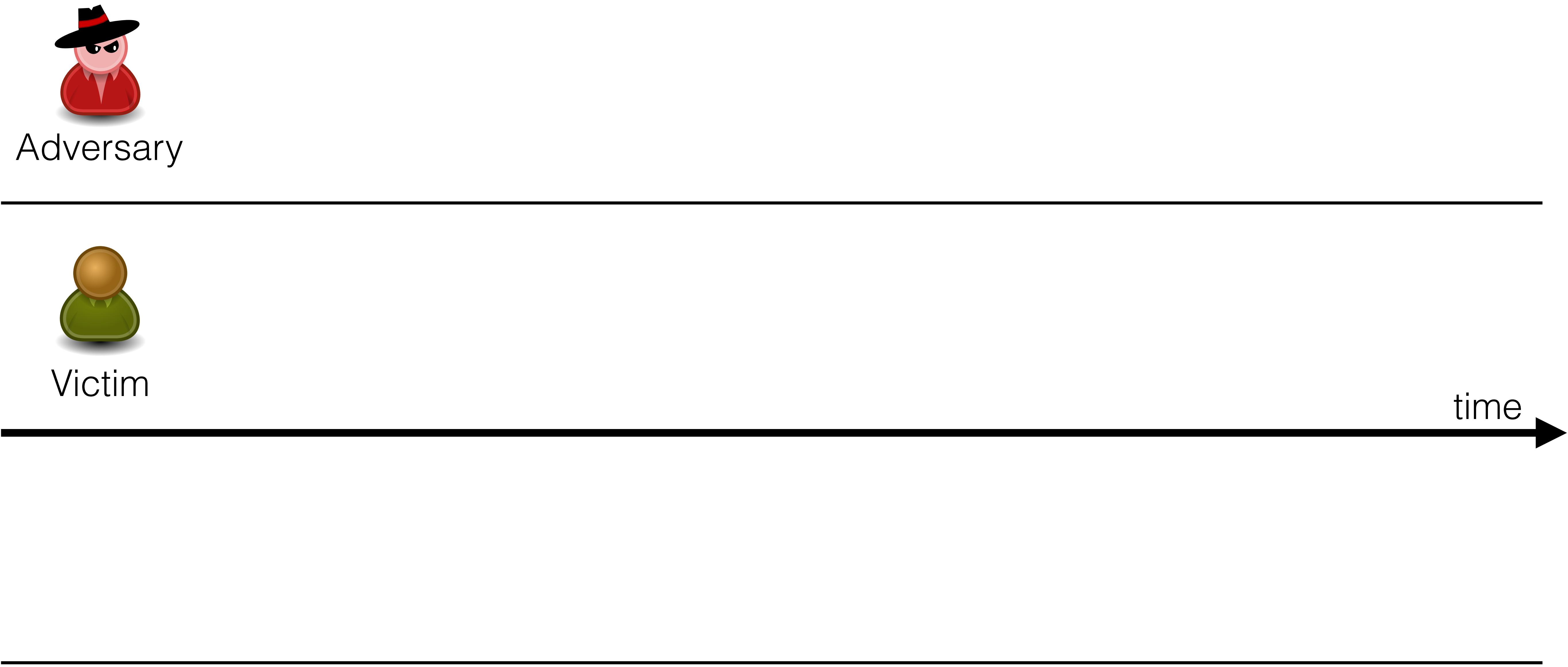
Delaying information delivery

Block delivery time > 20 min

1. Requirement for victim
   - Must not receive block header
   - Must not receive version message
2. Requirements for adversary
   - Must be first relayer for all blocks
   - Should perform connection depletion

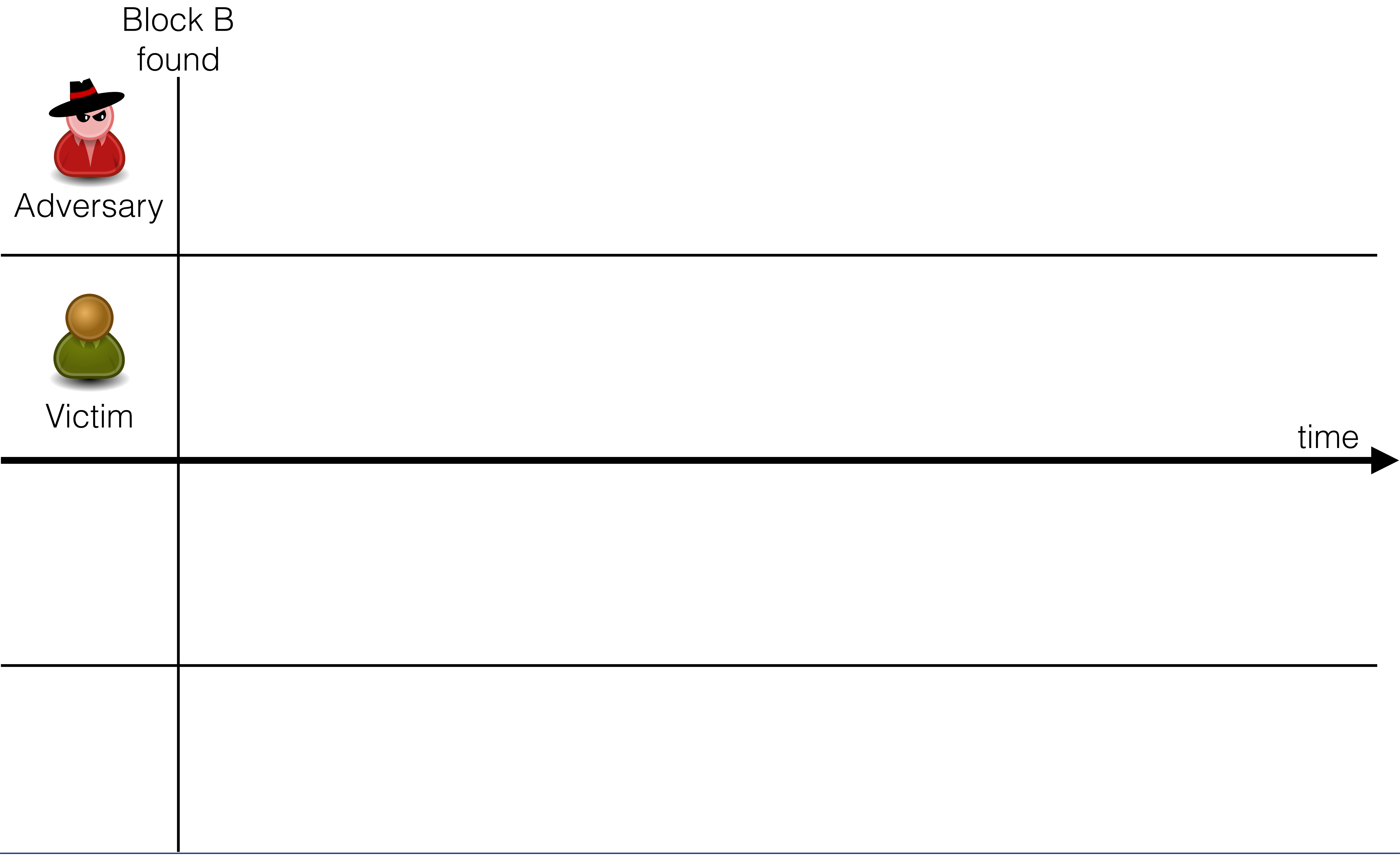
Delaying information delivery

Extending the block delivery time - Example

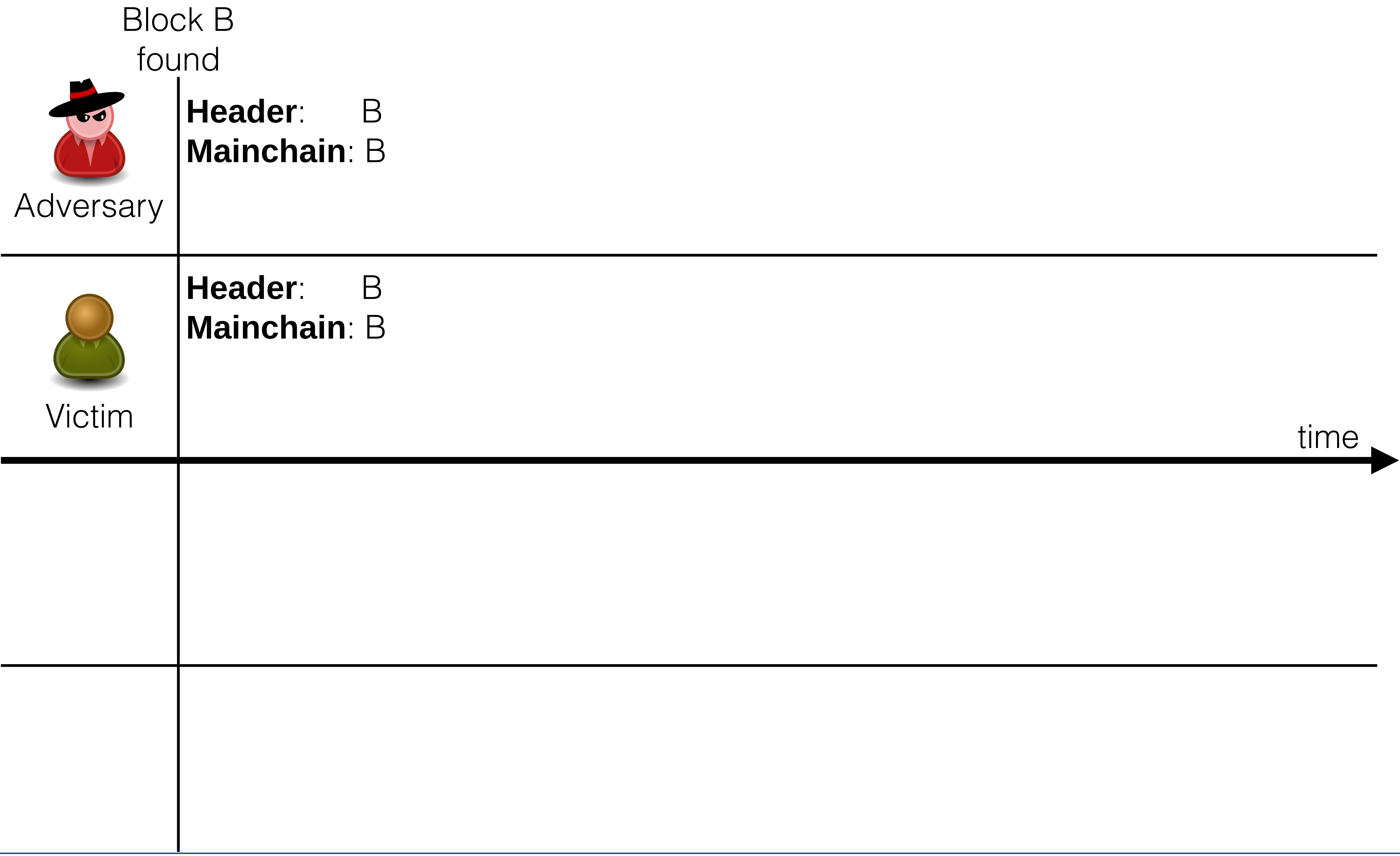


Delaying information delivery

Extending the block delivery time - Example



Extending the block delivery time - Example



Block B found

Adversary

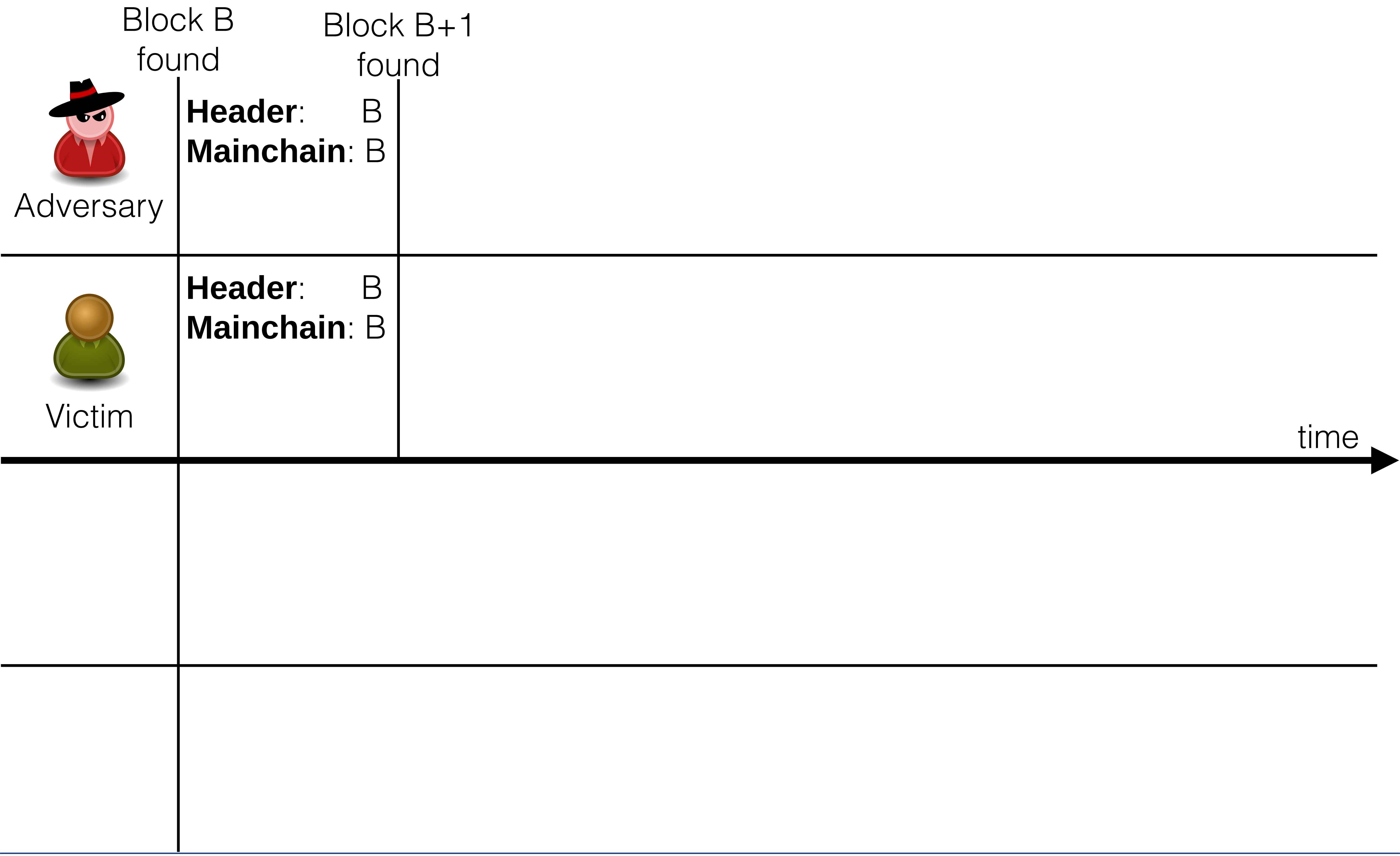
Header: B
Mainchain: B

Victim

Header: B
Mainchain: B

time

Extending the block delivery time - Example



Block B found

Block B+1 found

Adversary

Header: B
Mainchain: B

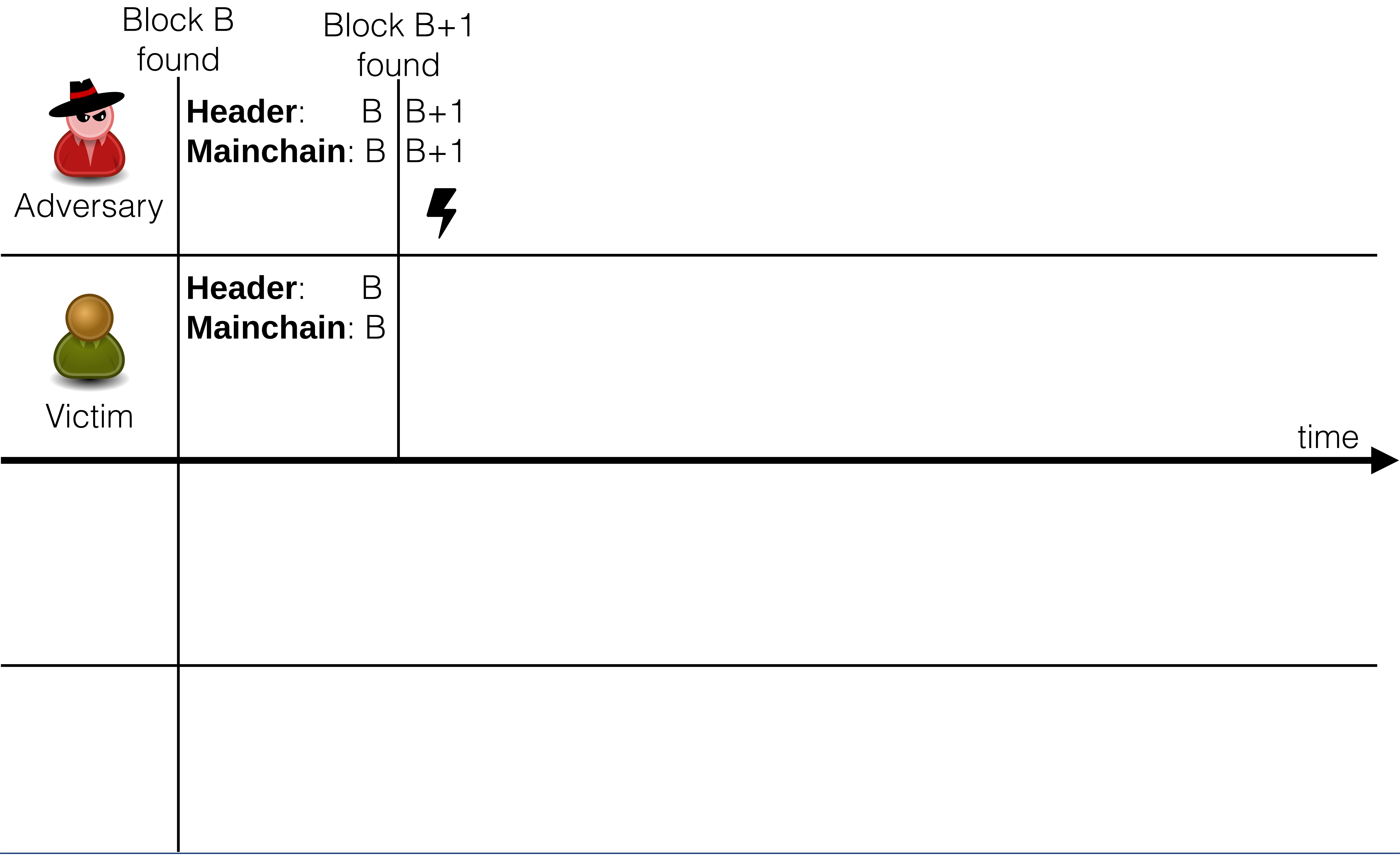
Victim

Header: B
Mainchain: B

time

Delaying information delivery

Extending the block delivery time - Example



Block B found

Block B+1 found

Adversary

Header: B | B+1

Mainchain: B | B+1

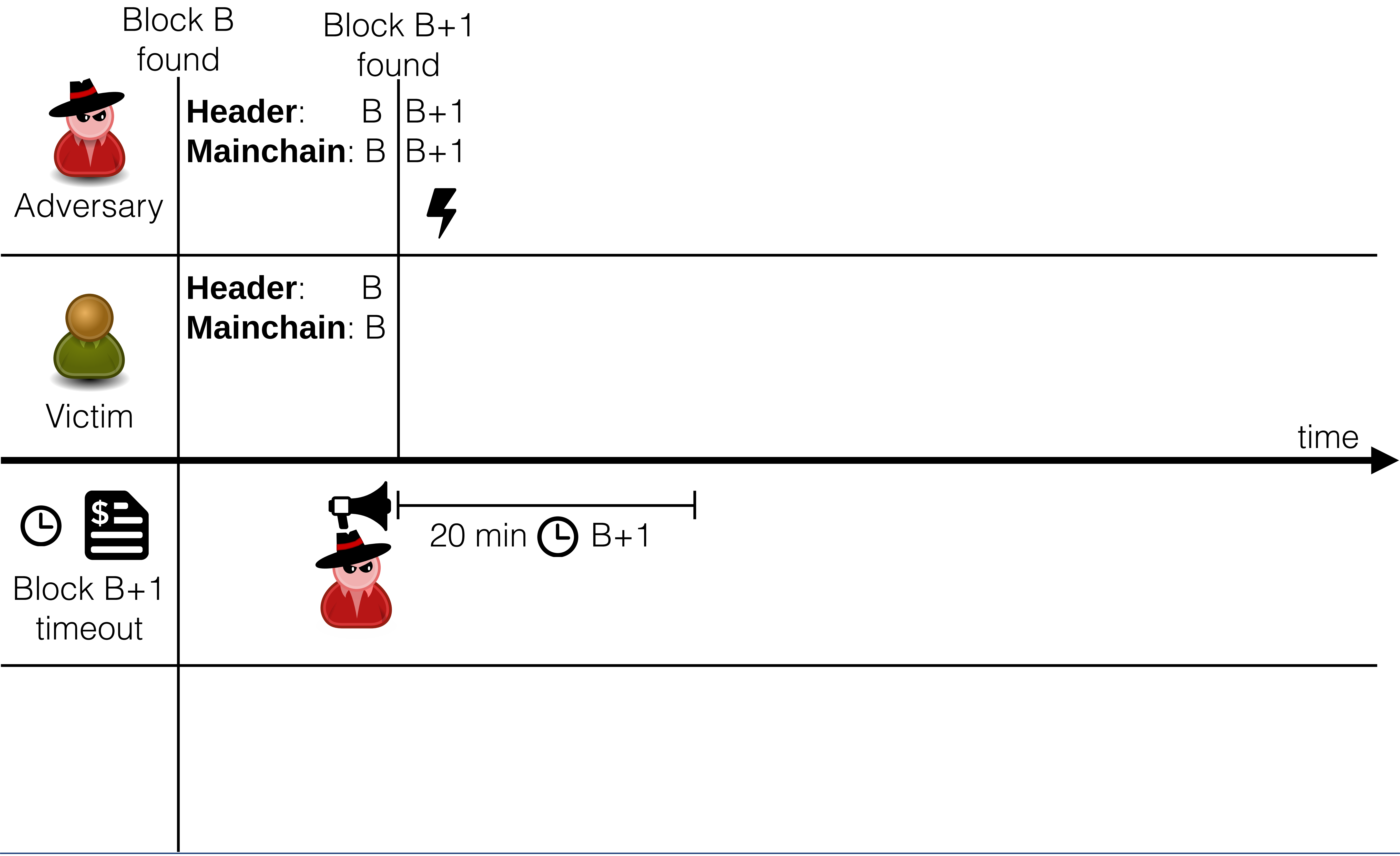
Victim

Header: B

Mainchain: B

time

Extending the block delivery time - Example



Block B found

Block B+1 found

Adversary

Header: B | B+1

Mainchain: B | B+1

Victim

Header: B

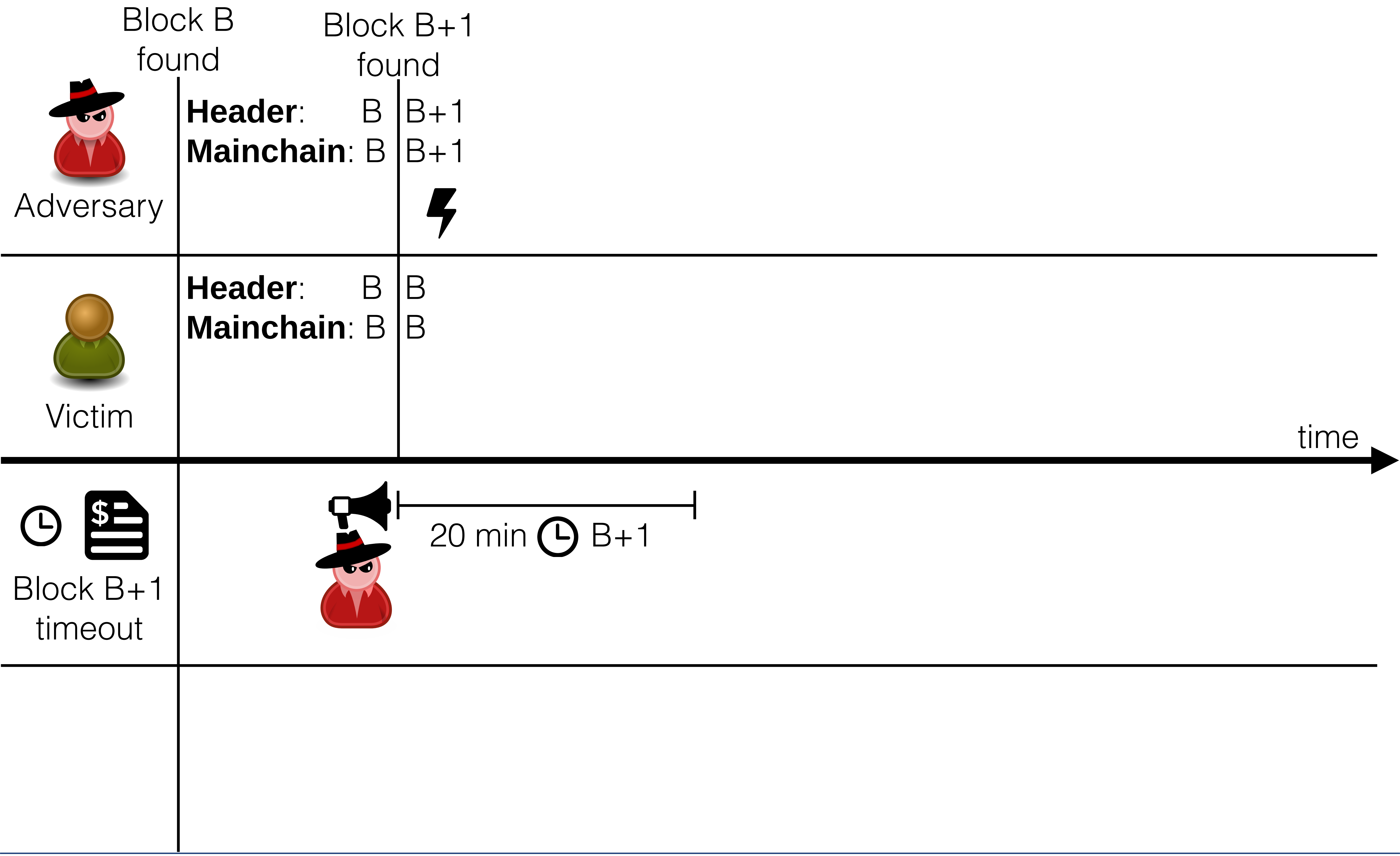
Mainchain: B

time

Block B+1 timeout

20 min B+1

Extending the block delivery time - Example



Block B found

Block B+1 found

Adversary

Header: B | B+1

Mainchain: B | B+1

Victim

Header: B | B

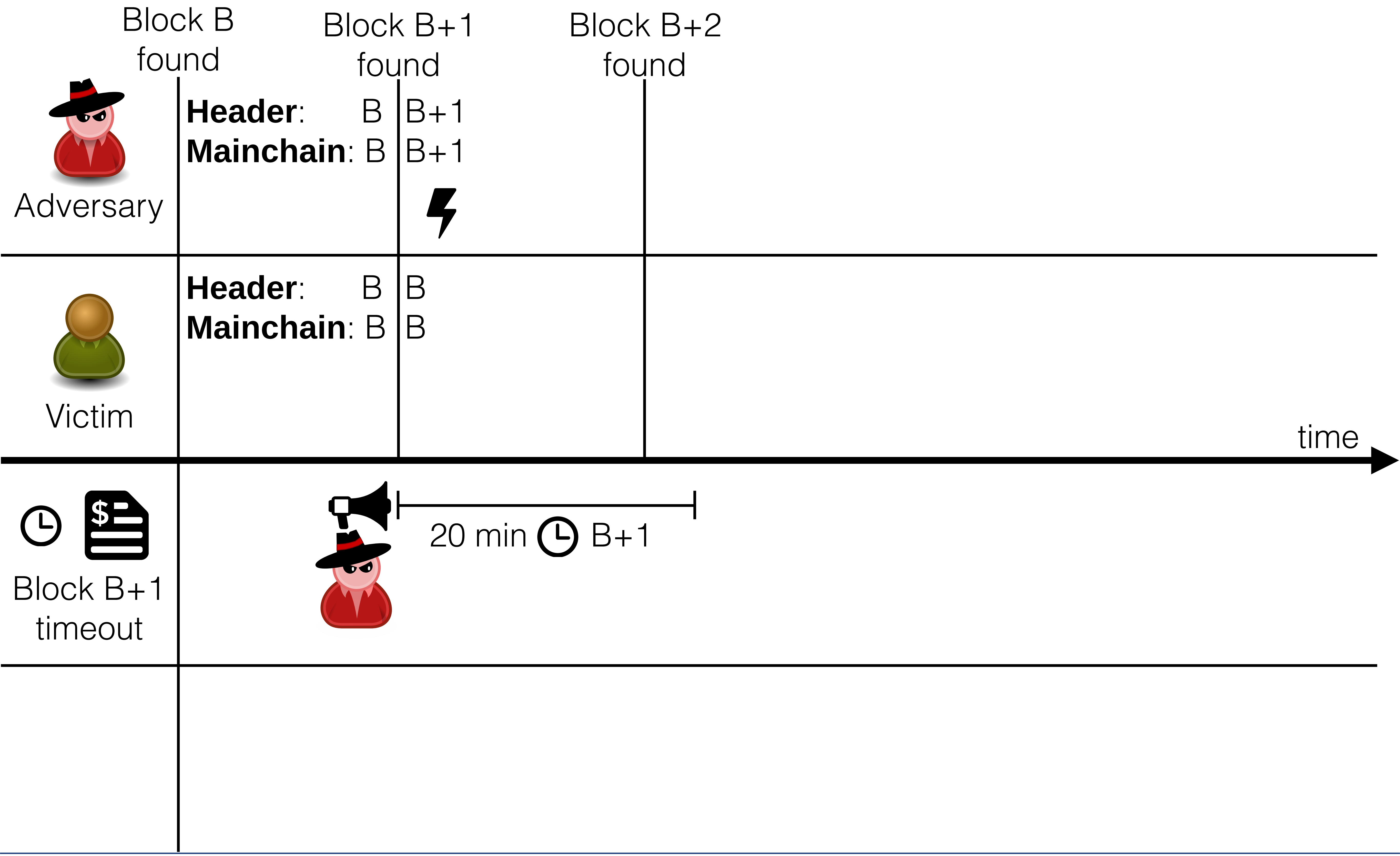
Mainchain: B | B

time

Block B+1 timeout

20 min B+1

Extending the block delivery time - Example



Block B found

Block B+1 found

Block B+2 found

Adversary

Header: B | B+1

Mainchain: B | B+1

Victim

Header: B | B

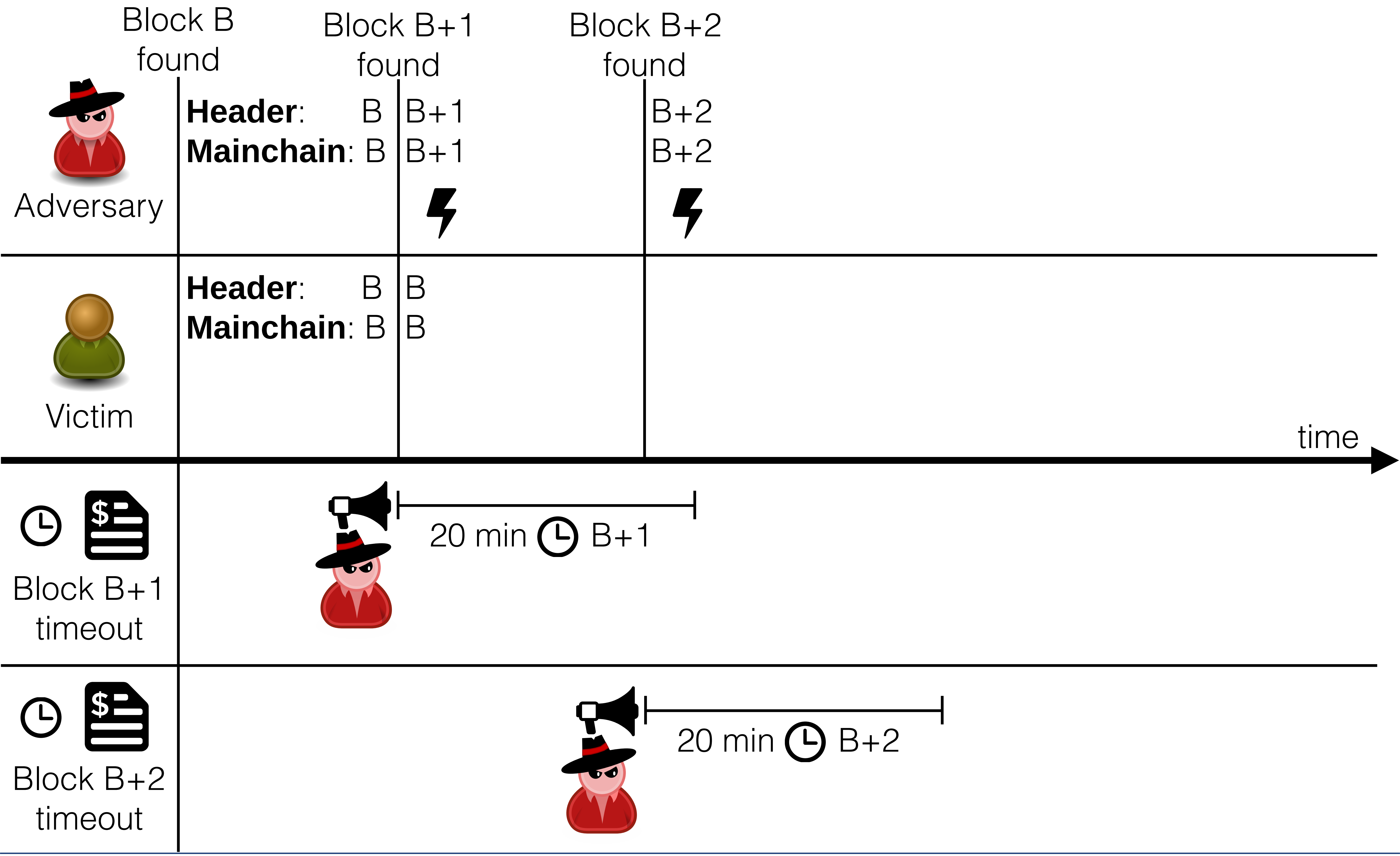
Mainchain: B | B

time

Block B+1 timeout

20 min B+1

Extending the block delivery time - Example



Block B found

Block B+1 found

Block B+2 found

Adversary

Header: B | B+1 | B+2

Mainchain: B | B+1 | B+2

Victim

Header: B | B

Mainchain: B | B

time

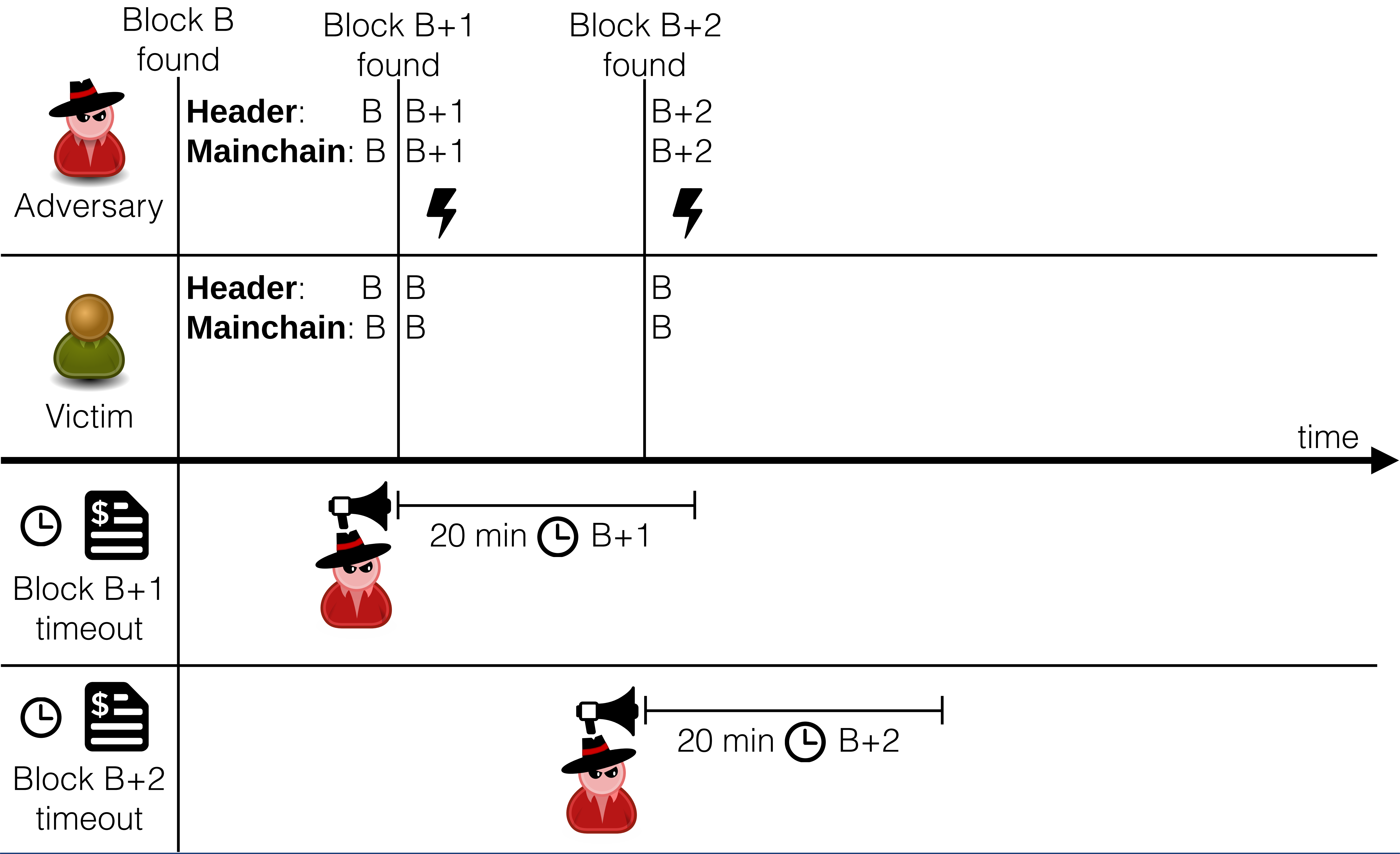
Block B+1 timeout

20 min B+1

Block B+2 timeout

20 min B+2

Extending the block delivery time - Example



Block B found

Block B+1 found

Block B+2 found

Adversary

Header: B | B+1 | B+2

Mainchain: B | B+1 | B+2

Victim

Header: B | B | B

Mainchain: B | B | B

time

Block B+1 timeout

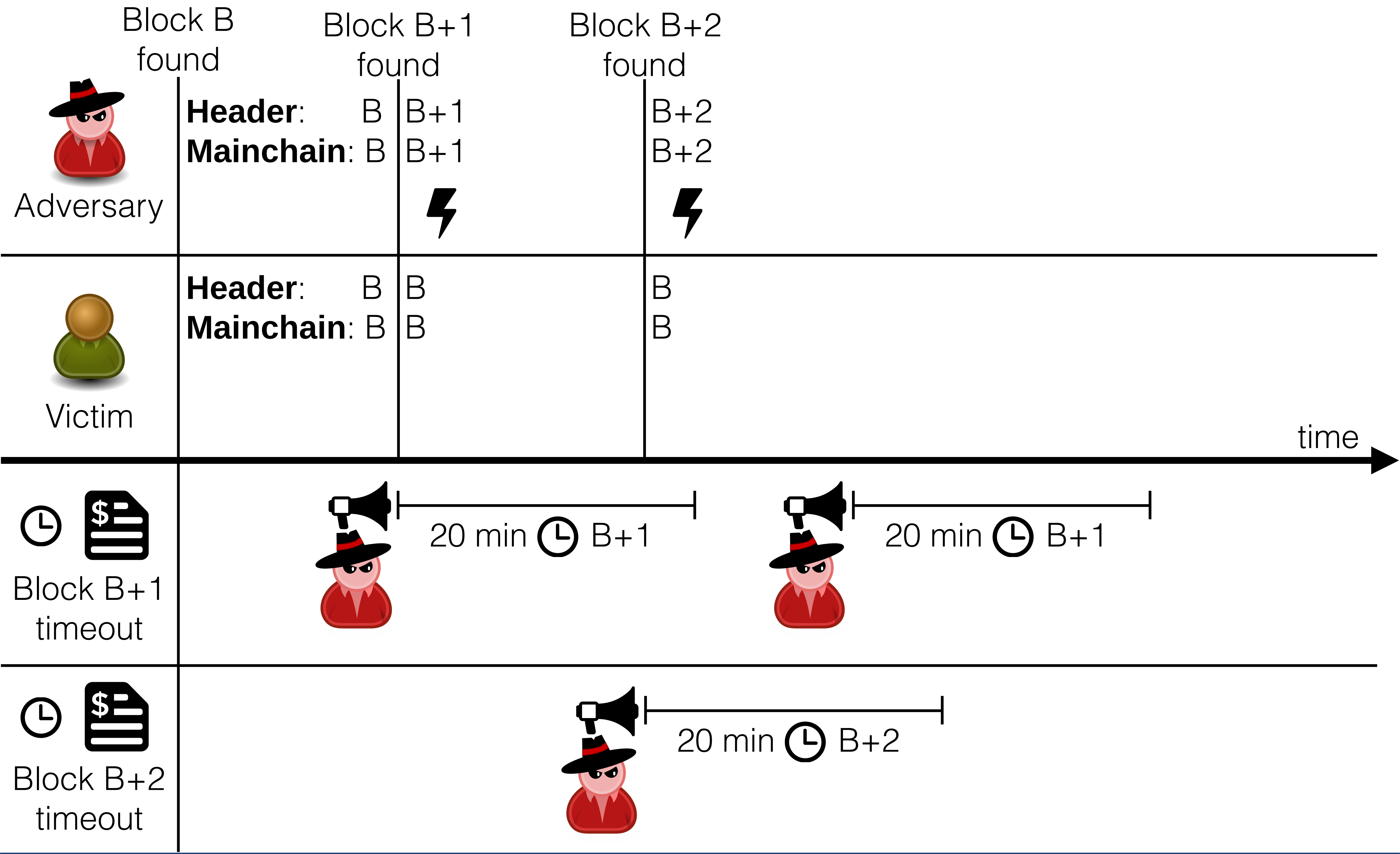
20 min B+1

Block B+2 timeout

20 min B+2

Delaying information delivery

Extending the block delivery time - Example



Block B found

Block B+1 found

Block B+2 found

Adversary

Header: B | B+1 | B+2

Mainchain: B | B+1 | B+2

Victim

Header: B | B | B

Mainchain: B | B | B

time

Block B+1 timeout

20 min B+1

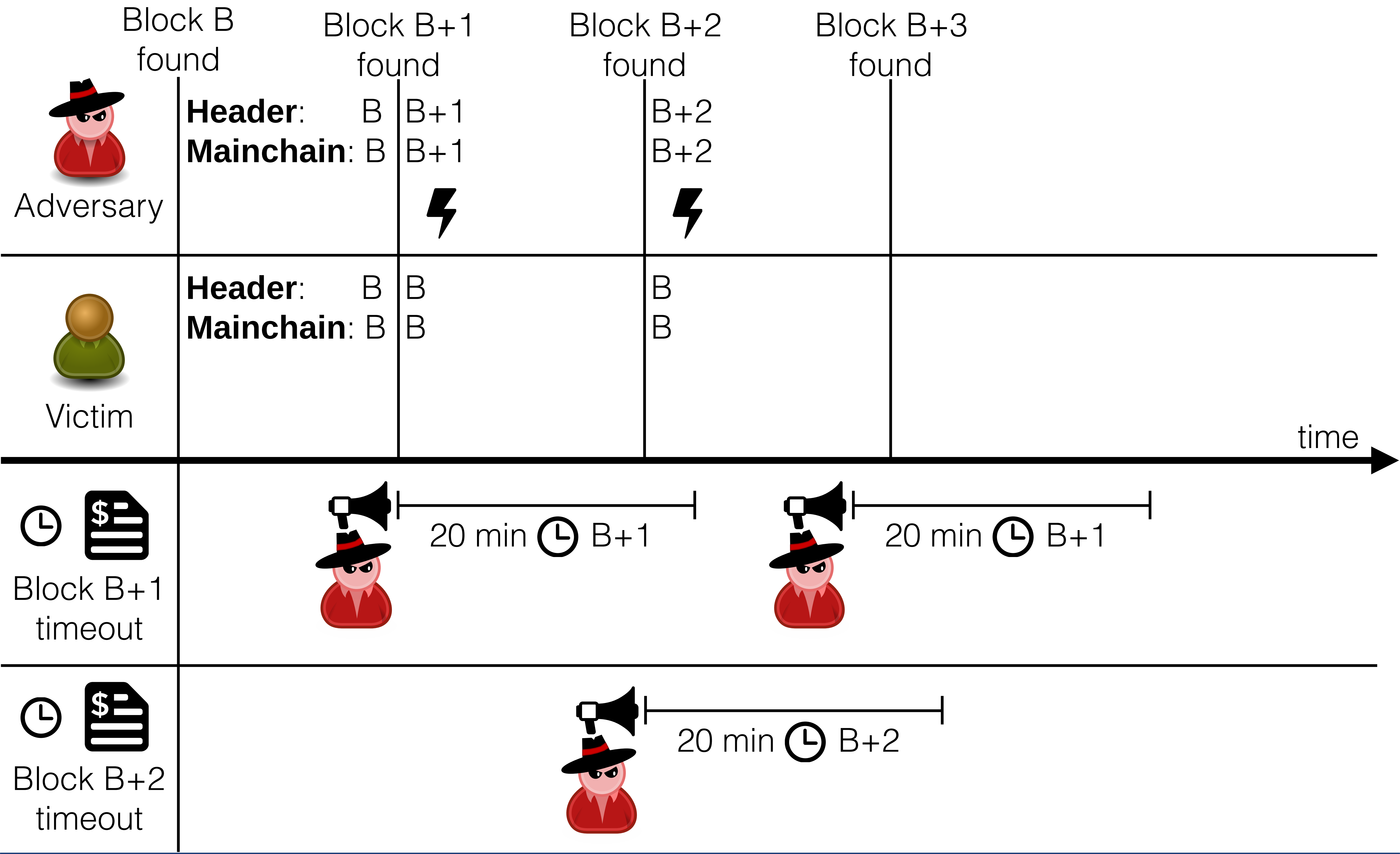
20 min B+1

Block B+2 timeout

20 min B+2

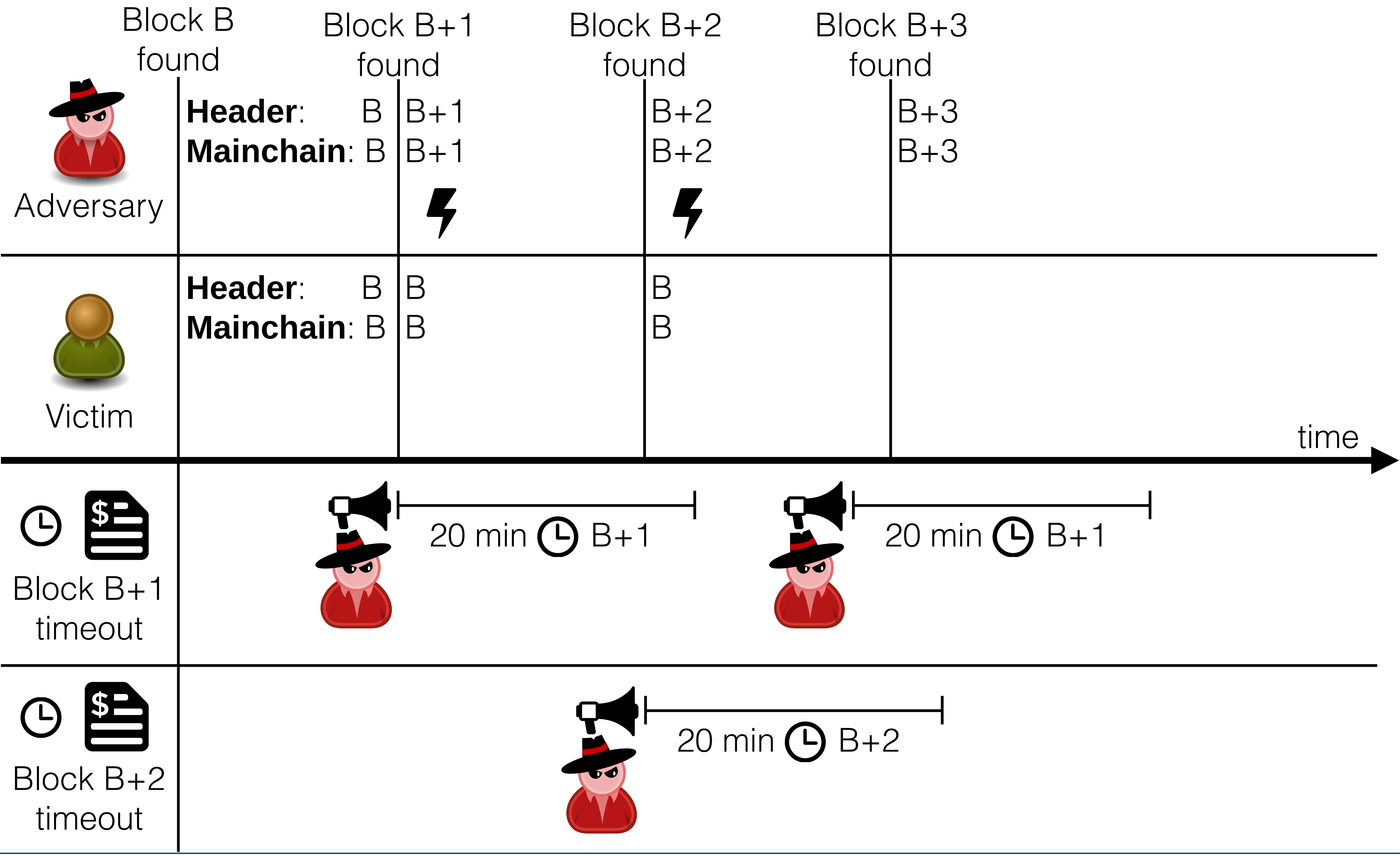
Delaying information delivery

Extending the block delivery time - Example



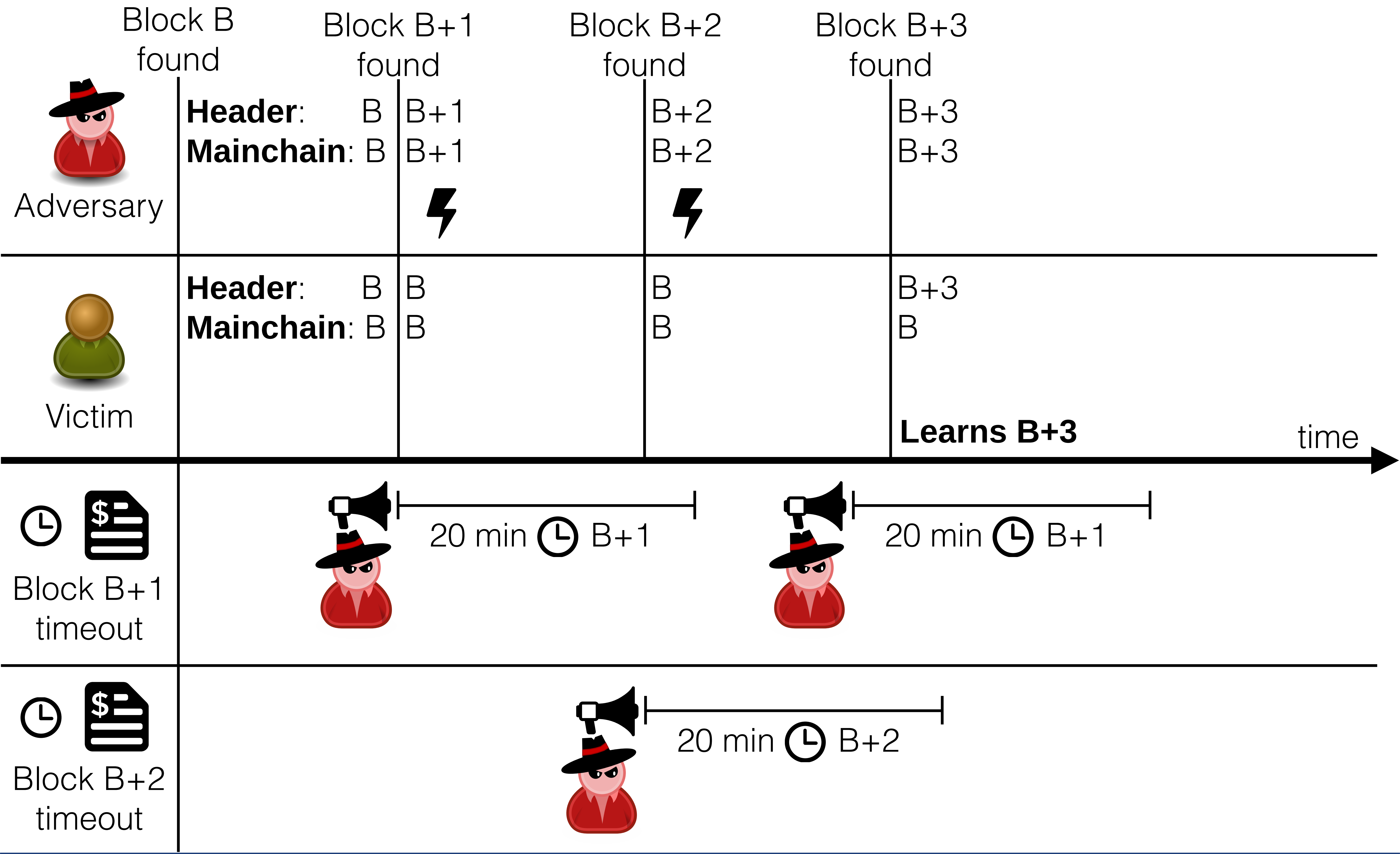
Delaying information delivery

Extending the block delivery time - Example



Delaying information delivery

Extending the block delivery time - Example



Block B found

Block B+1 found

Block B+2 found

Block B+3 found

Adversary

Header: B | B+1 | B+2 | B+3

Mainchain: B | B+1 | B+2 | B+3

Victim

Header: B | B | B | B+3

Mainchain: B | B | B | B

Learns B+3

time

Block B+1 timeout

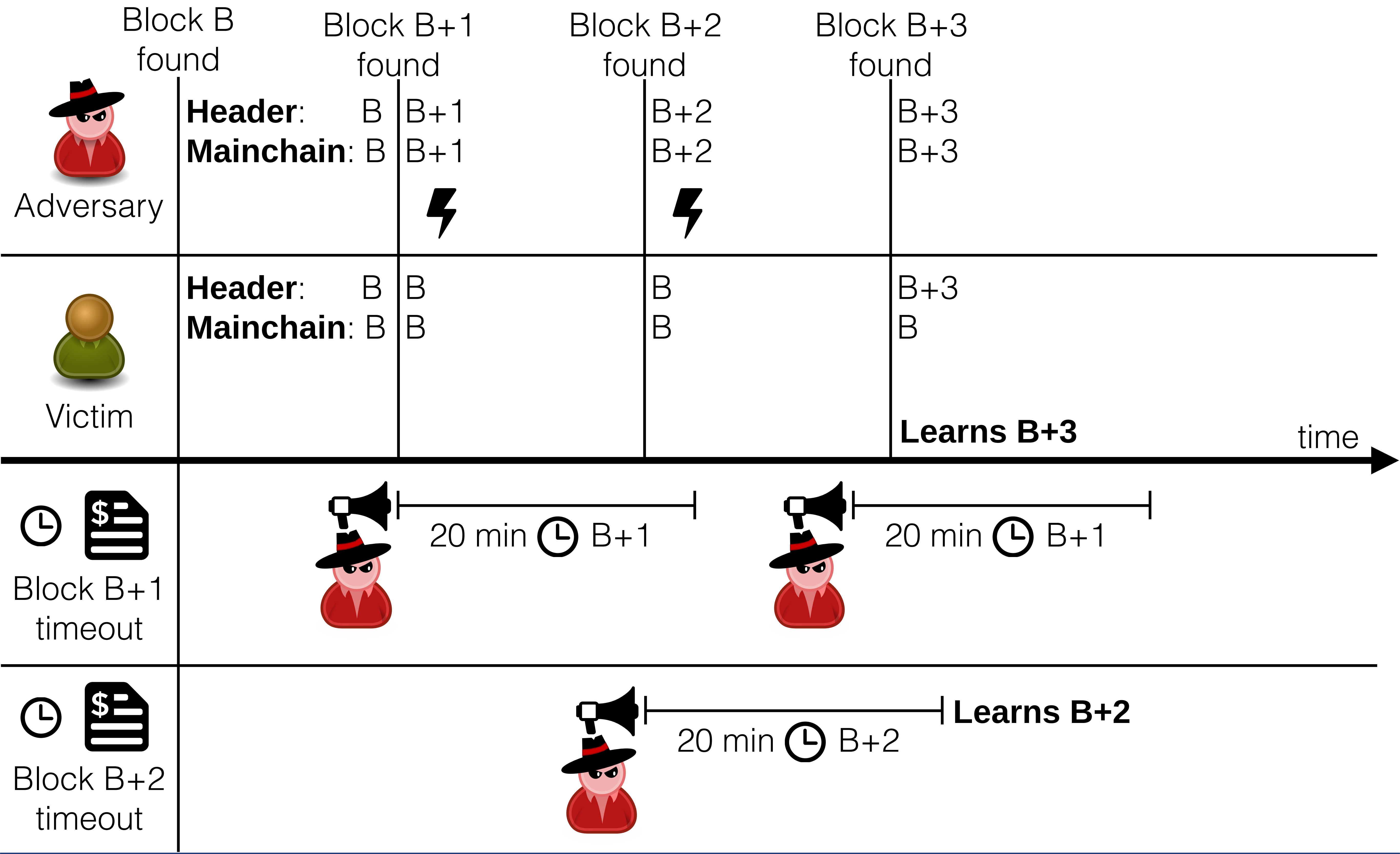
20 min B+1

20 min B+1

Block B+2 timeout

20 min B+2

Extending the block delivery time - Example



Block B found

Block B+1 found

Block B+2 found

Block B+3 found

Adversary

Header: B | B+1 | B+2 | B+3

Mainchain: B | B+1 | B+2 | B+3

Victim

Header: B | B | B | B+3

Mainchain: B | B | B | B

Learns B+3

time

Block B+1 timeout

20 min B+1

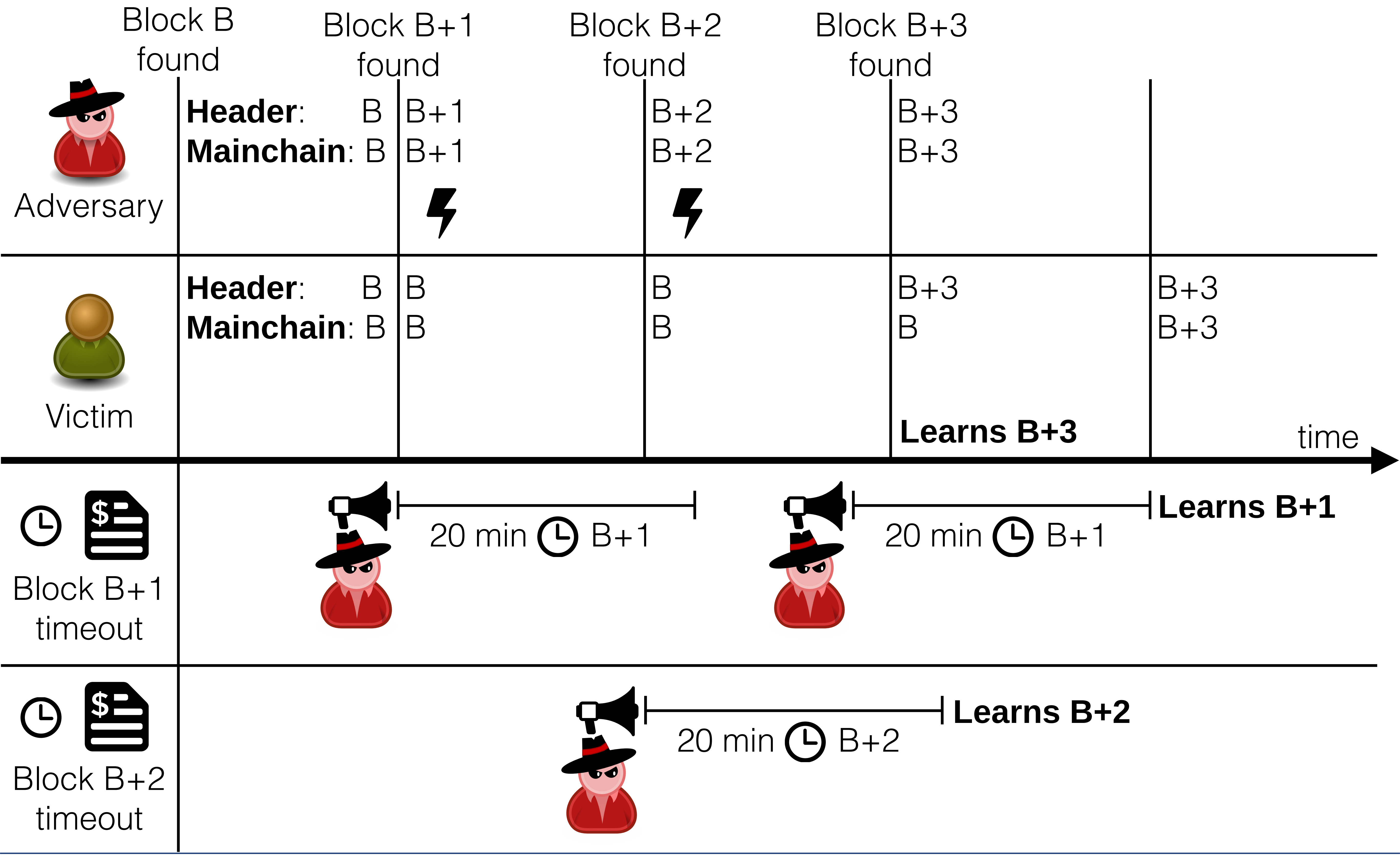
20 min B+1

Block B+2 timeout

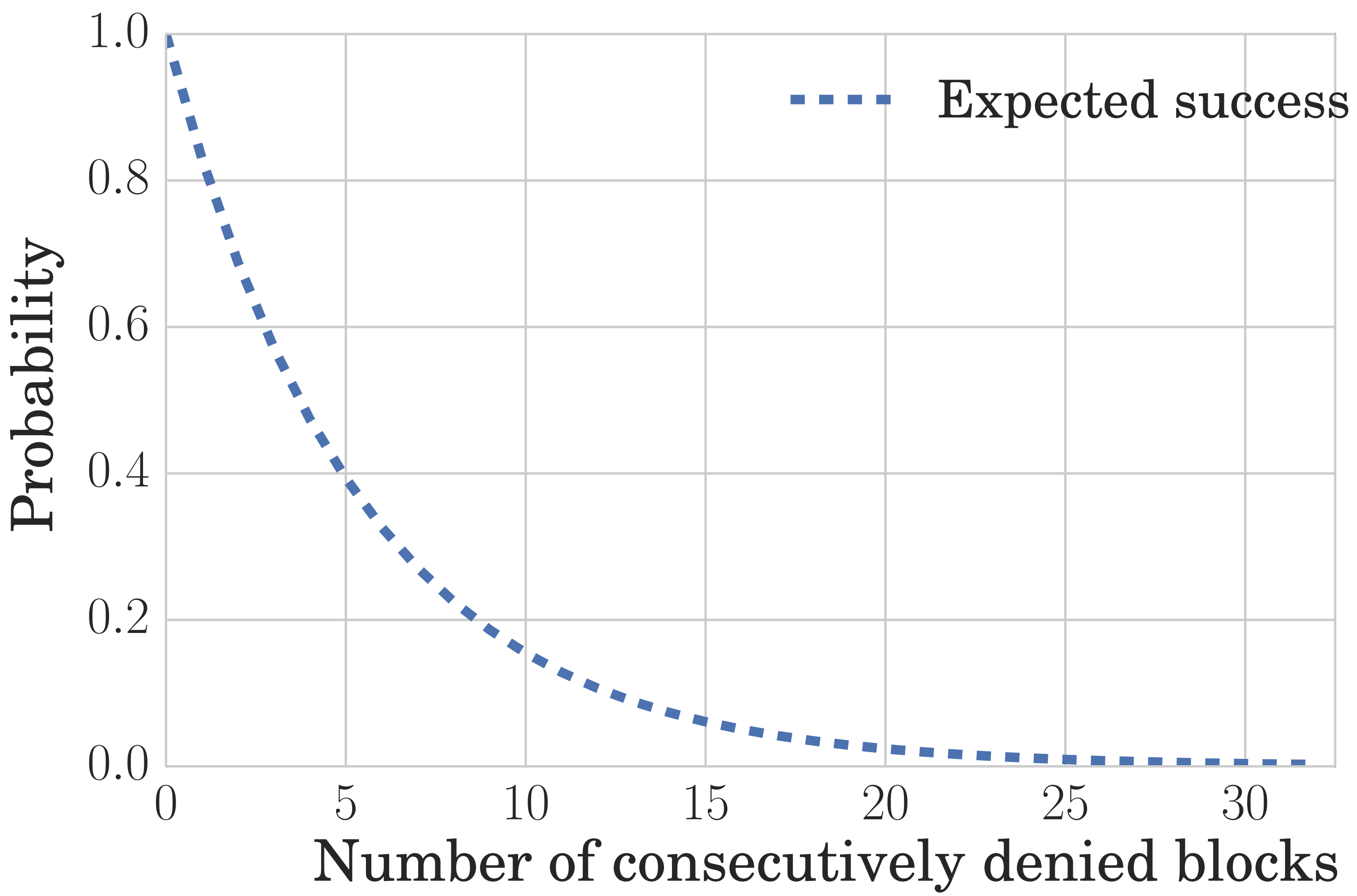
20 min B+2

Learns B+2

Extending the block delivery time - Example



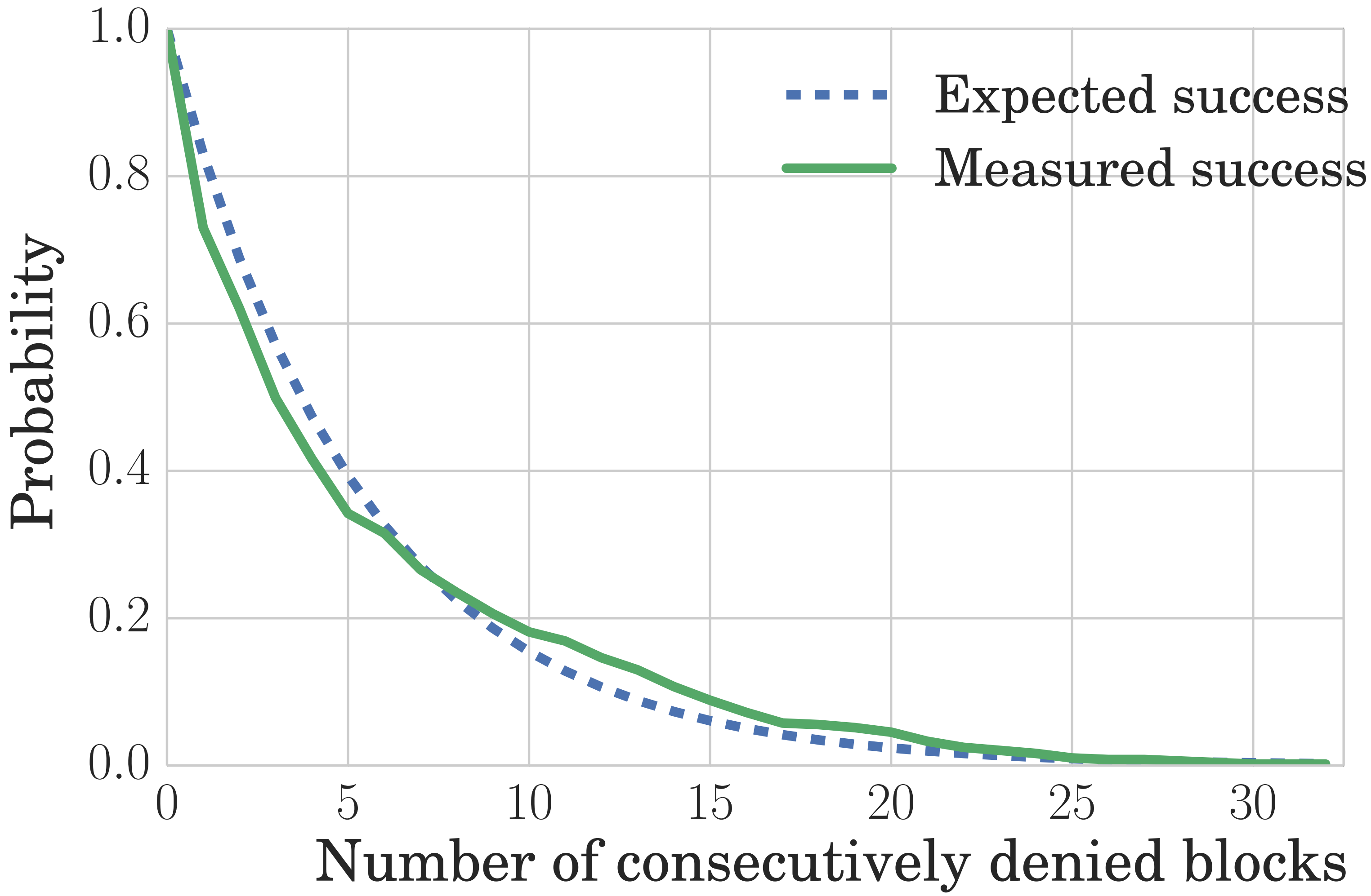
Experimental validation - Continuous block denial



Probability for N blocks = P^N

based on P = 0.83

Experimental validation - Continuous block denial



Probability for N blocks = P^N

based on P = 0.83

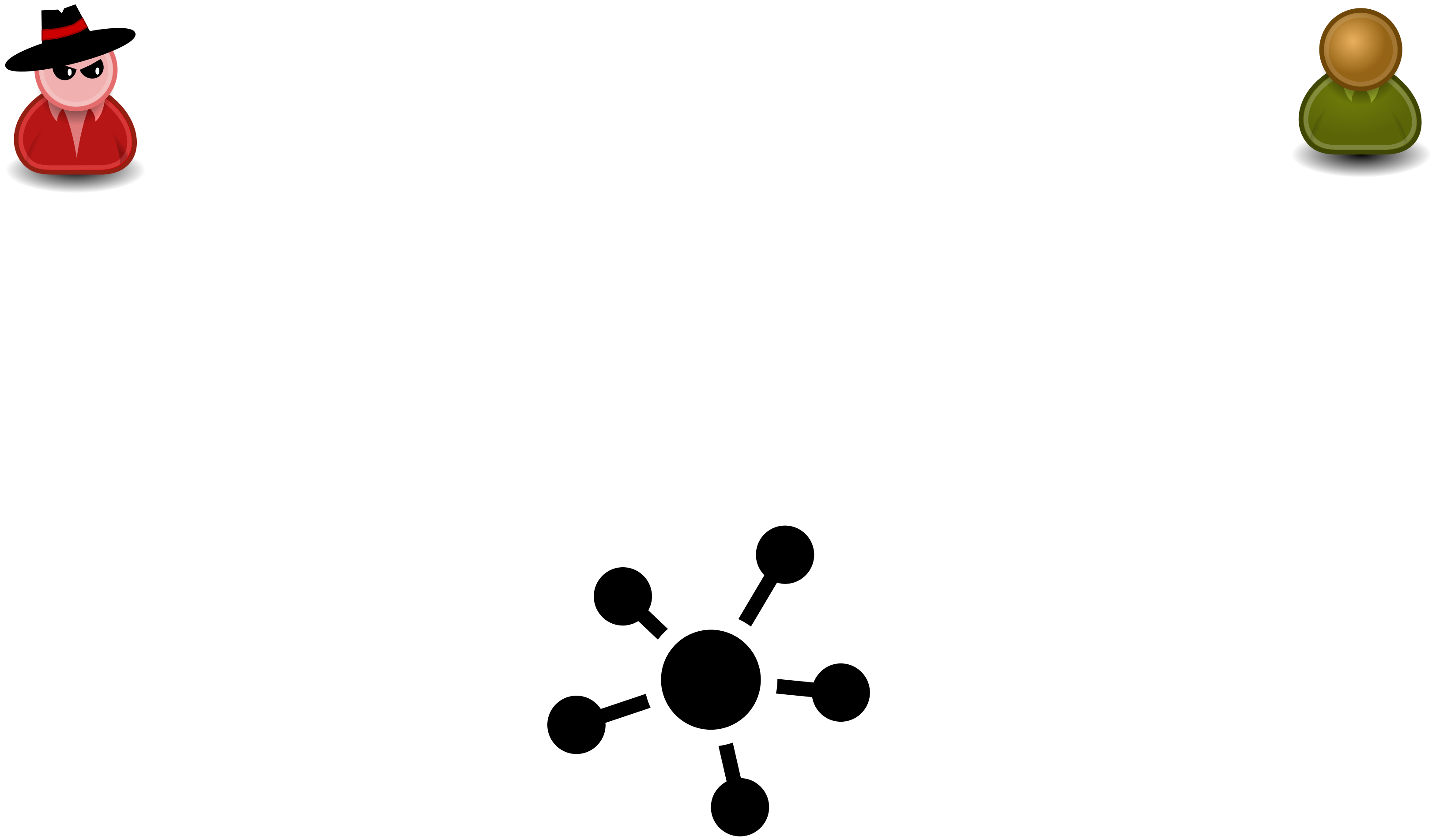
Implications

What is Double Spending?

Spending money more than once

$TX_{legitimate}$ - Pays the vendor

$TX_{doublespend}$ - Pays the adversary

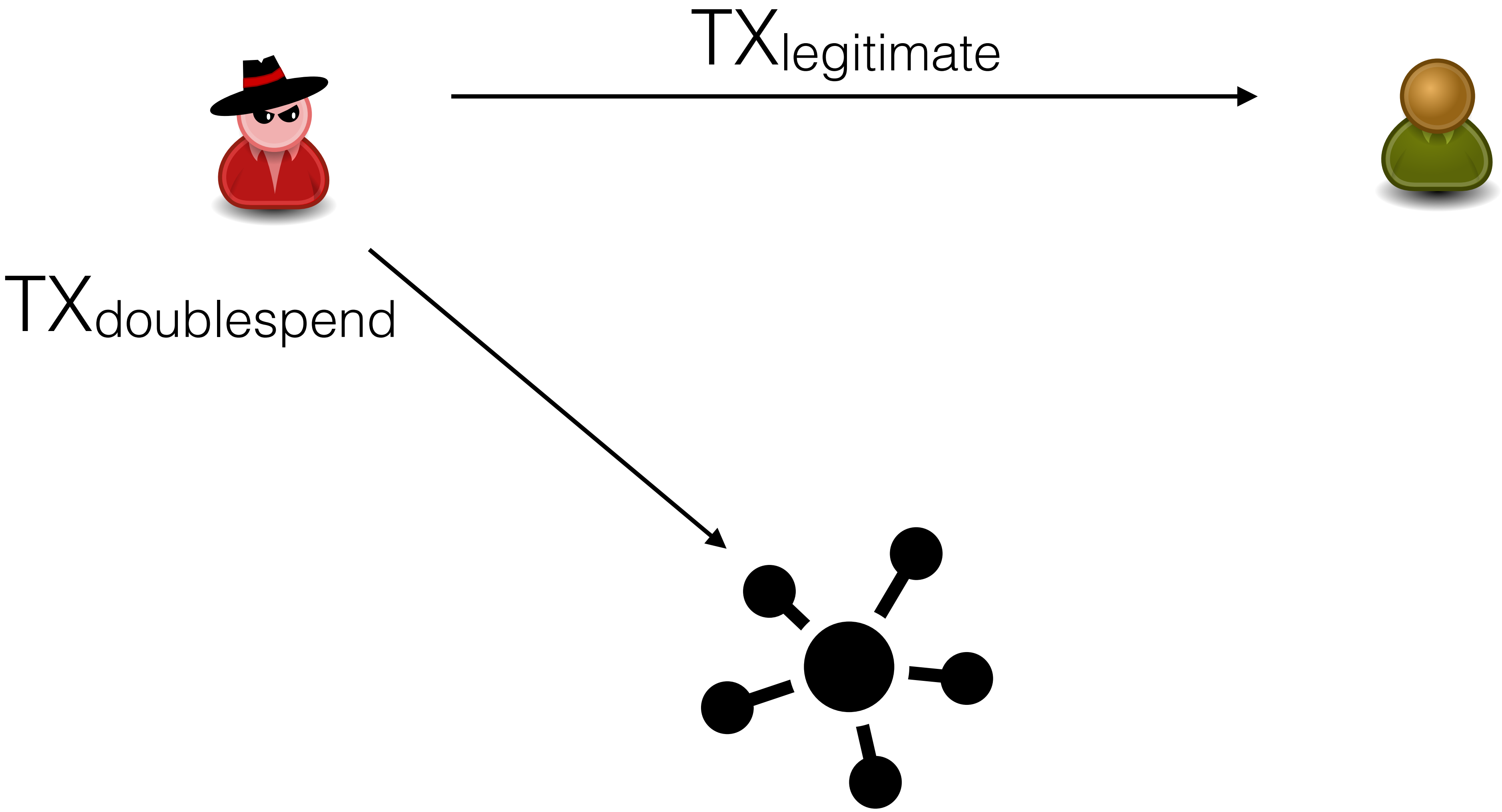


What is Double Spending?

Spending money more than once

$TX_{\text{legitimate}}$ - Pays the vendor

$TX_{\text{doublespend}}$ - Pays the adversary

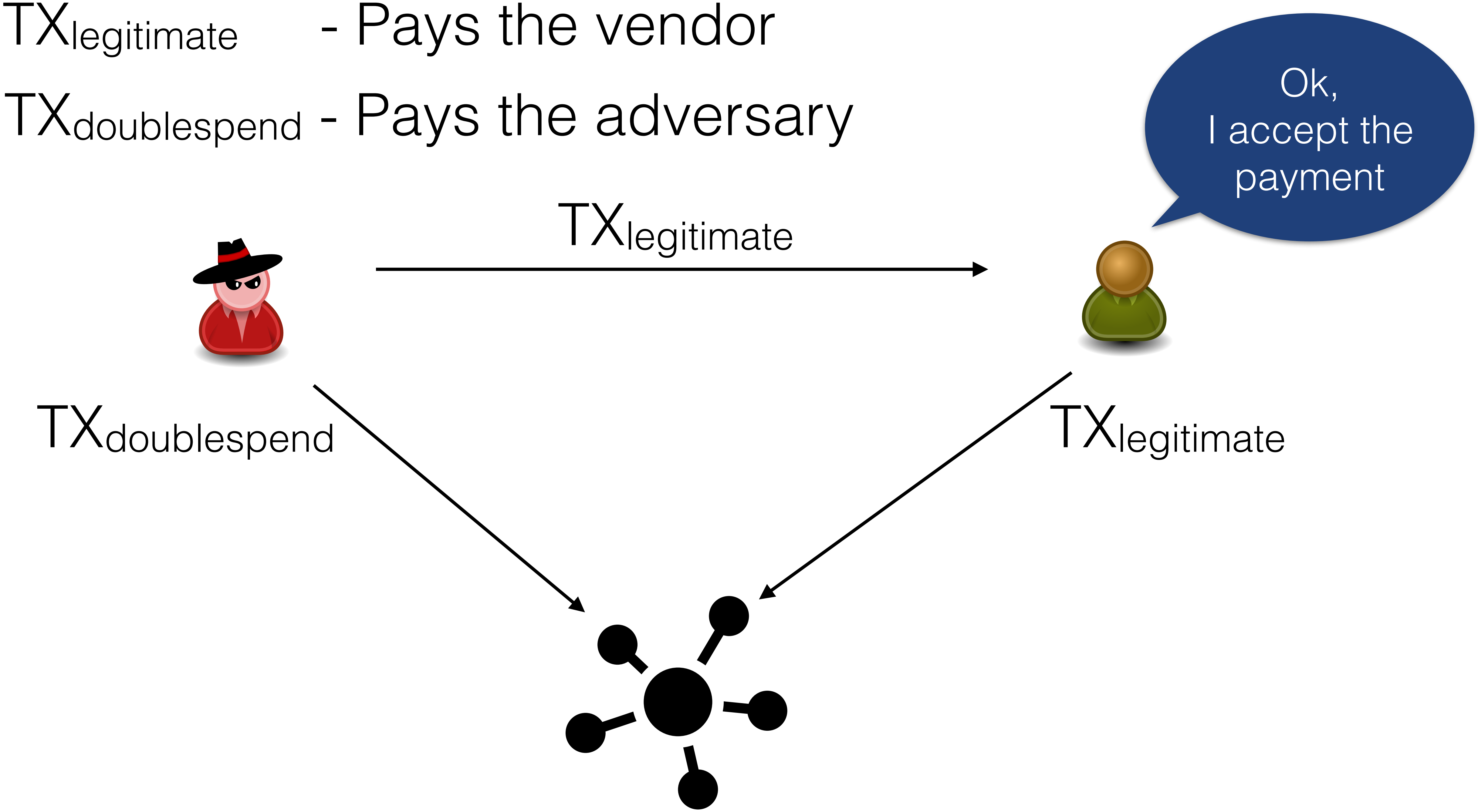


What is Double Spending?

Spending money more than once

$\text{TX}_{\text{legitimate}}$ - Pays the vendor

$\text{TX}_{\text{doublespend}}$ - Pays the adversary


Ok,
I accept the payment
$\text{TX}_{\text{legitimate}}$
$\text{TX}_{\text{doublespend}}$
$\text{TX}_{\text{legitimate}}$

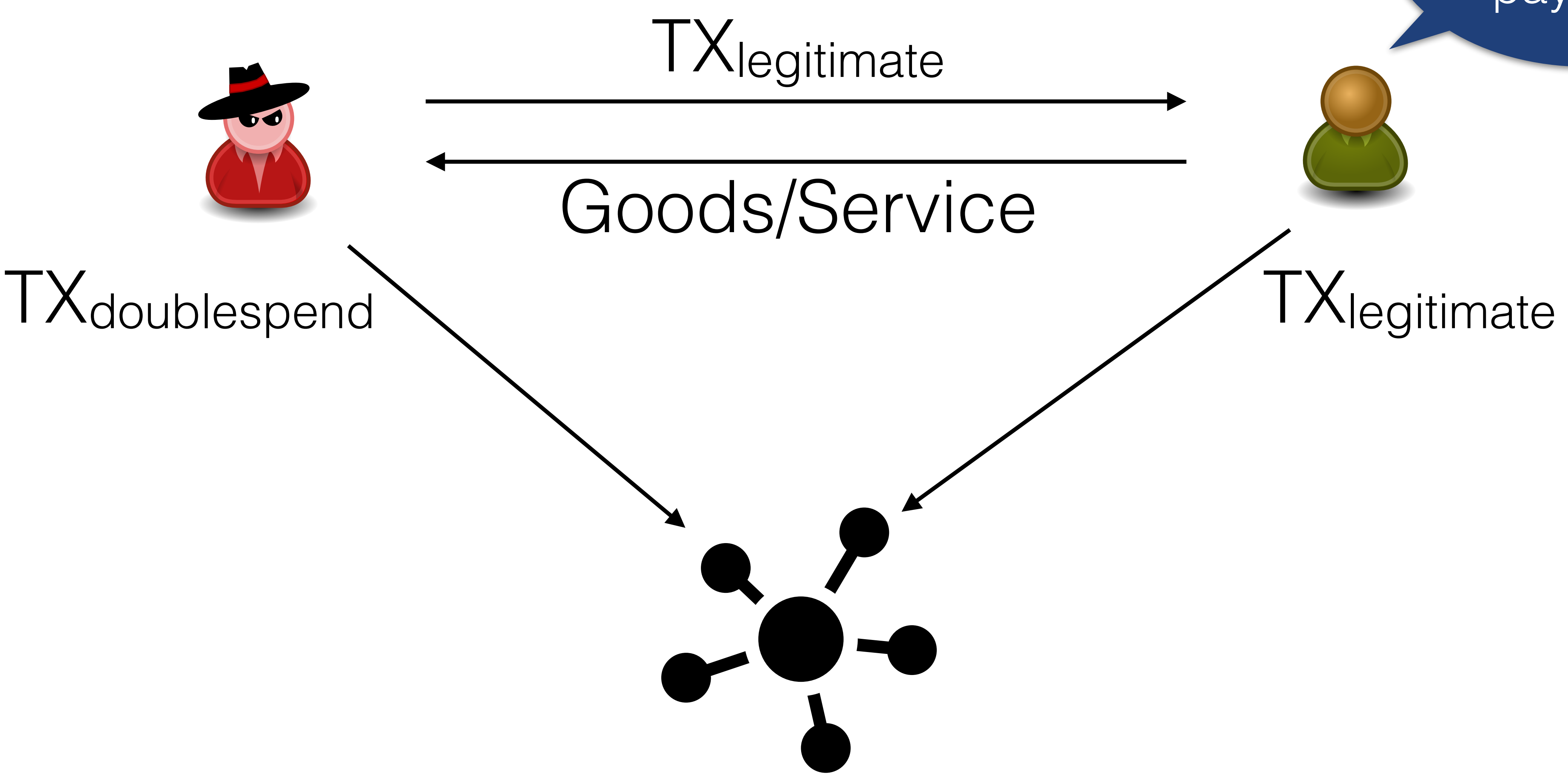
What is Double Spending?

Spending money more than once

$TX_{legitimate}$ - Pays the vendor

$TX_{doublespend}$ - Pays the adversary

Ok,
I accept the
payment

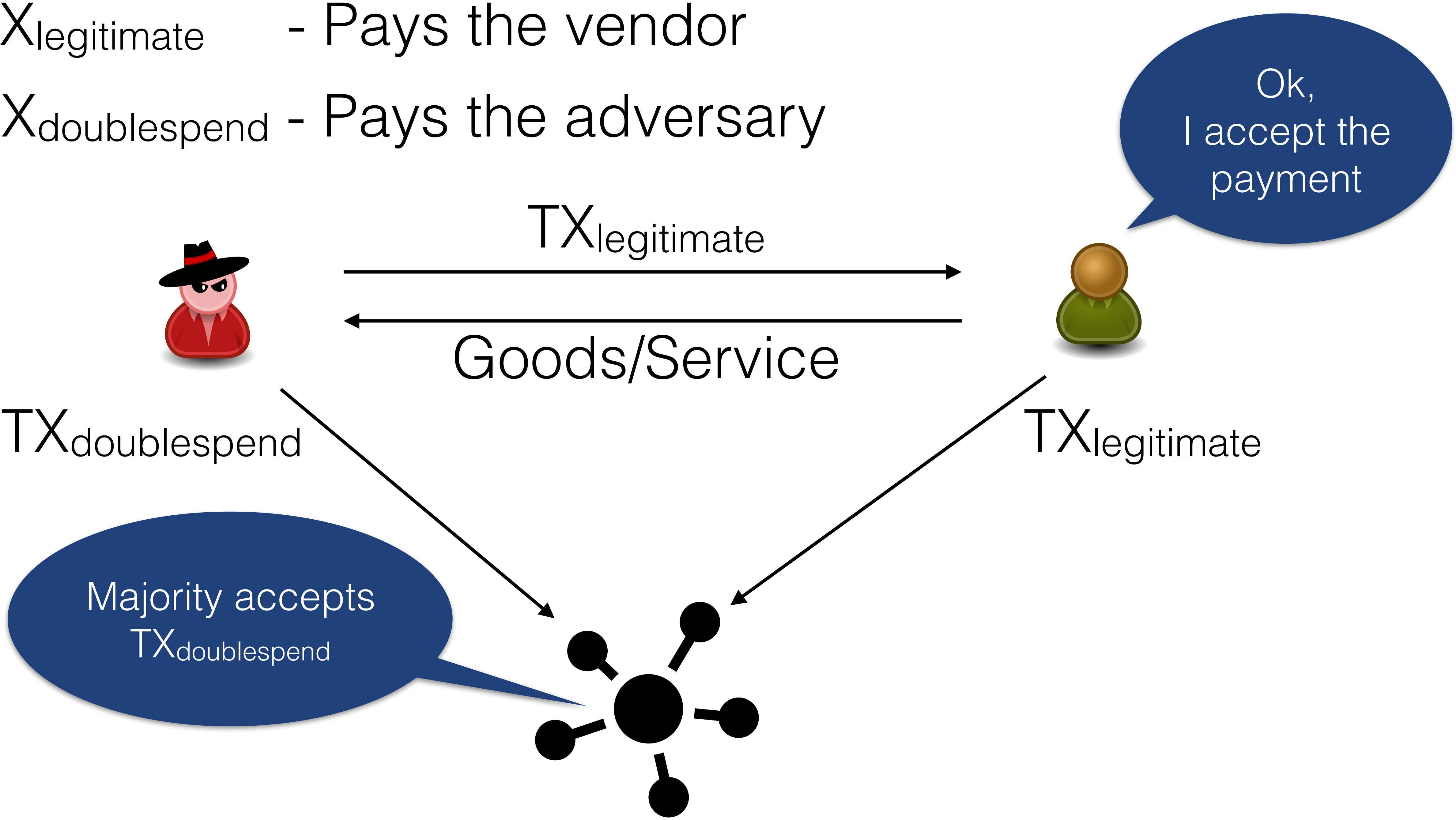


What is Double Spending?

Spending money more than once

$\text{TX}_{\text{legitimate}}$ - Pays the vendor

$\text{TX}_{\text{doublespend}}$ - Pays the adversary

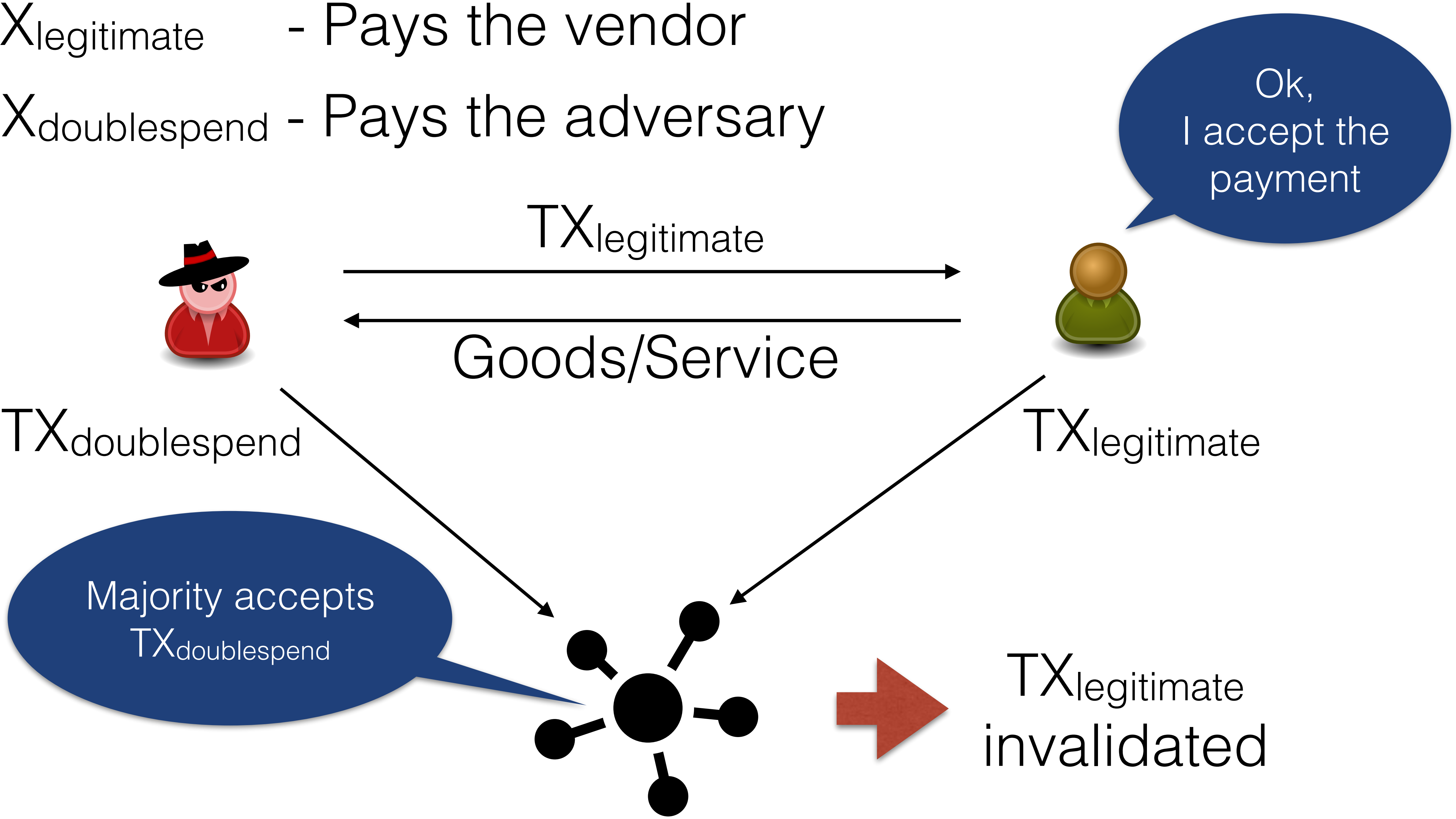

Ok,
I accept the payment
$\text{TX}_{\text{legitimate}}$
Goods/Service
$\text{TX}_{\text{doublespend}}$
$\text{TX}_{\text{legitimate}}$
Majority accepts $\text{TX}_{\text{doublespend}}$

What is Double Spending?

Spending money more than once

$TX_{legitimate}$ - Pays the vendor

$TX_{doublespend}$ - Pays the adversary



Ok, I accept the payment

$TX_{legitimate}$

Goods/Service

$TX_{doublespend}$

$TX_{legitimate}$

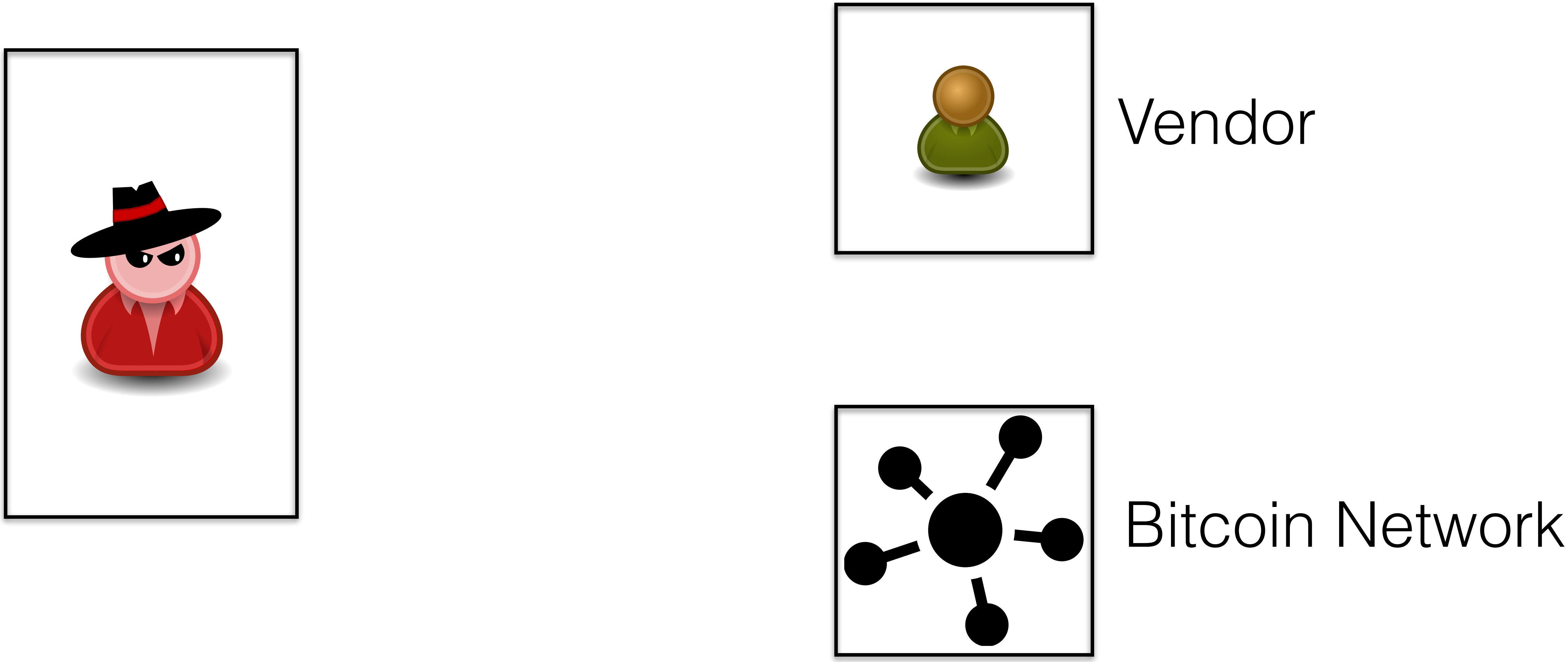
Majority accepts $TX_{doublespend}$

$TX_{legitimate}$ invalidated

Implications

Double Spending 0 Confirmation Transactions

- Very reliable attack
- Regardless of protection (double spend relay)



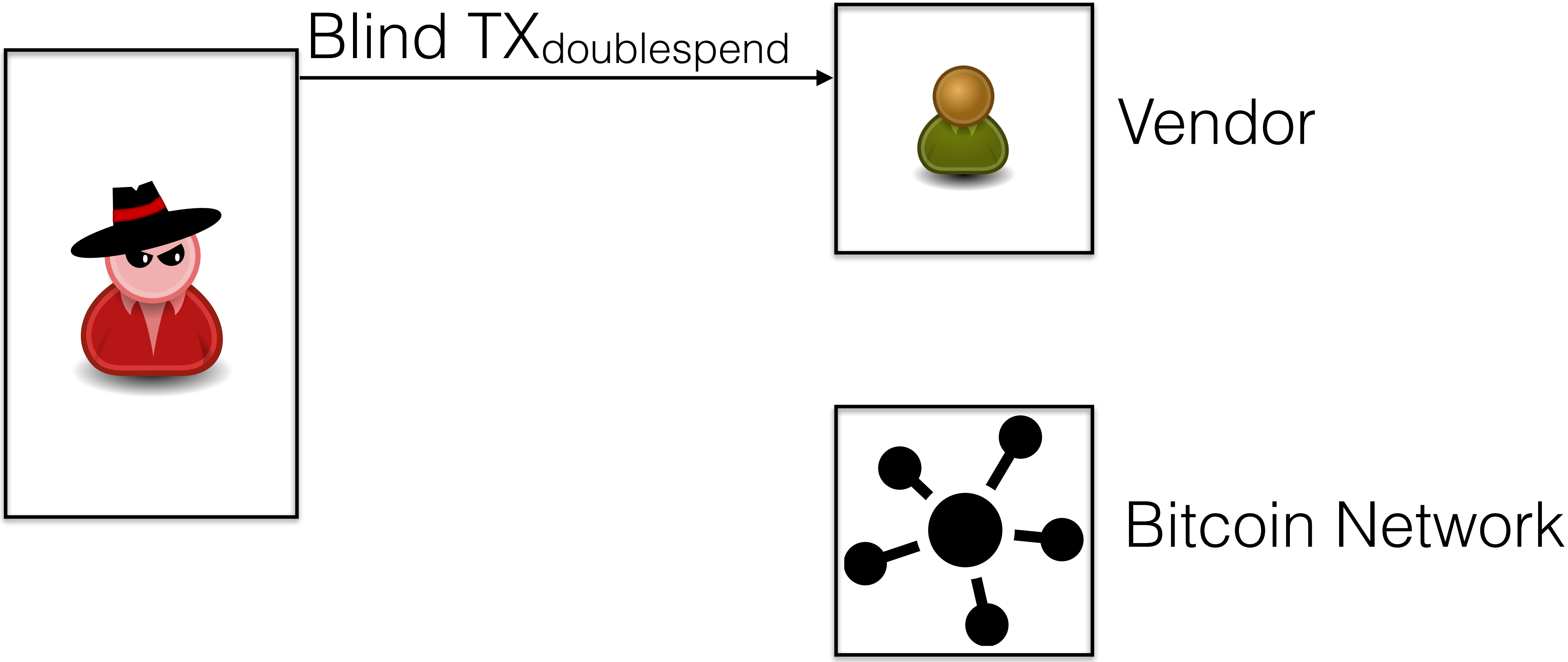
Vendor

Bitcoin Network

Implications

Double Spending 0 Confirmation Transactions

- Very reliable attack
- Regardless of protection (double spend relay)



Blind $\text{TX}_{\text{doublespend}}$

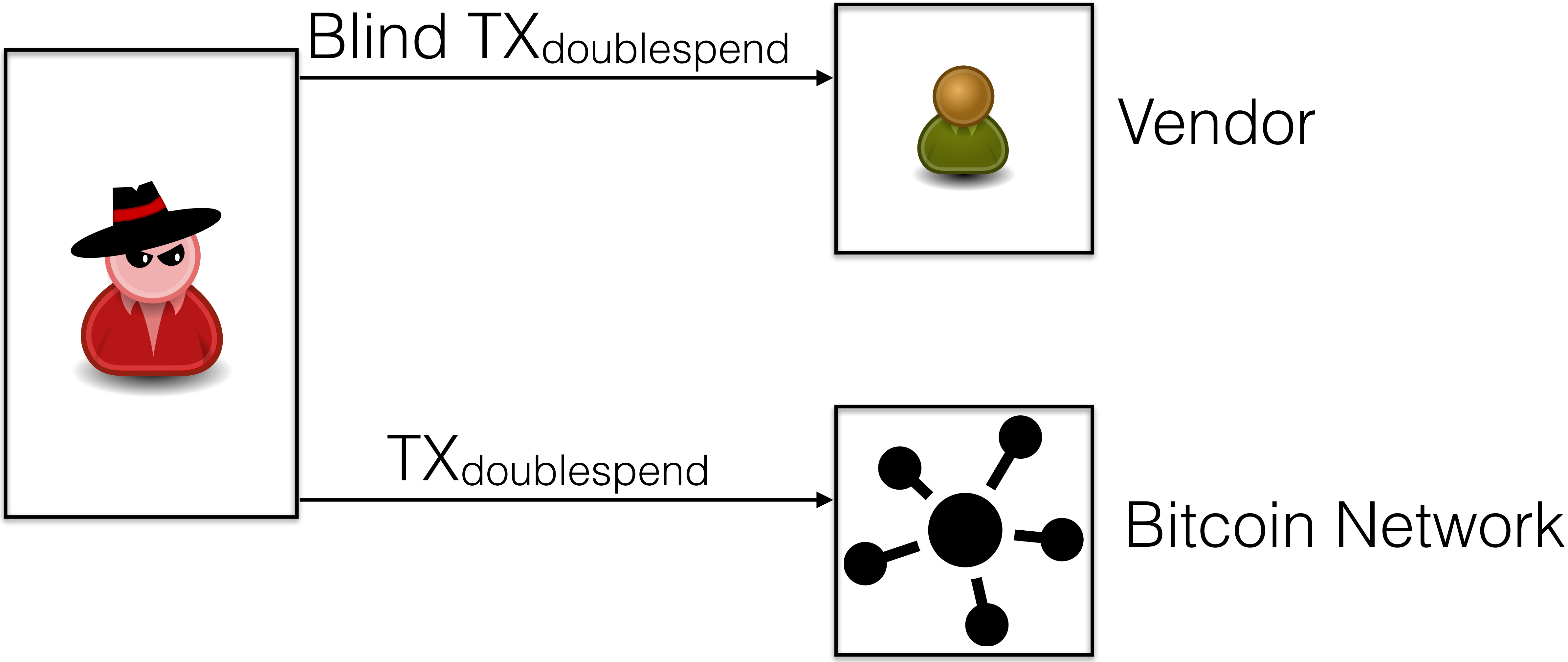
Vendor

Bitcoin Network

Implications

Double Spending 0 Confirmation Transactions

- Very reliable attack
- Regardless of protection (double spend relay)



Blind $TX_{doublespend}$

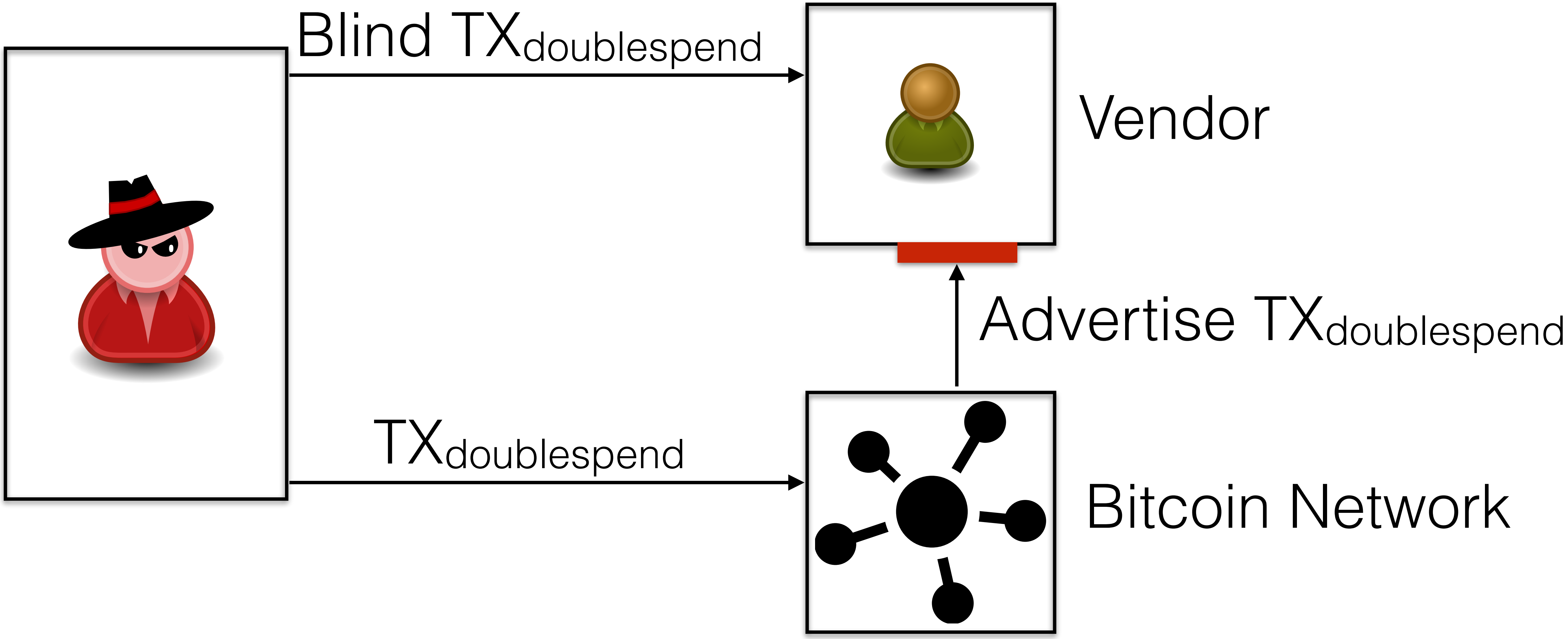
Vendor

$TX_{doublespend}$

Bitcoin Network

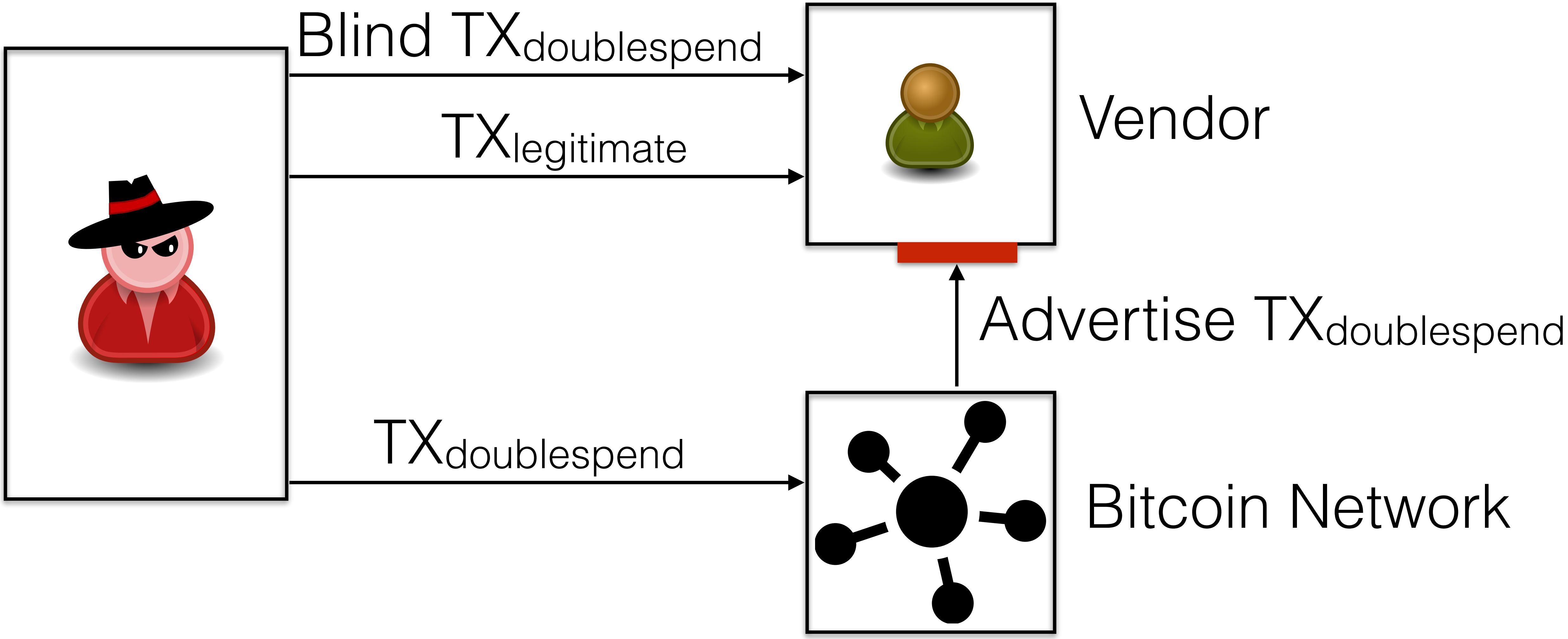
Double Spending 0 Confirmation Transactions

- Very reliable attack
- Regardless of protection (double spend relay)



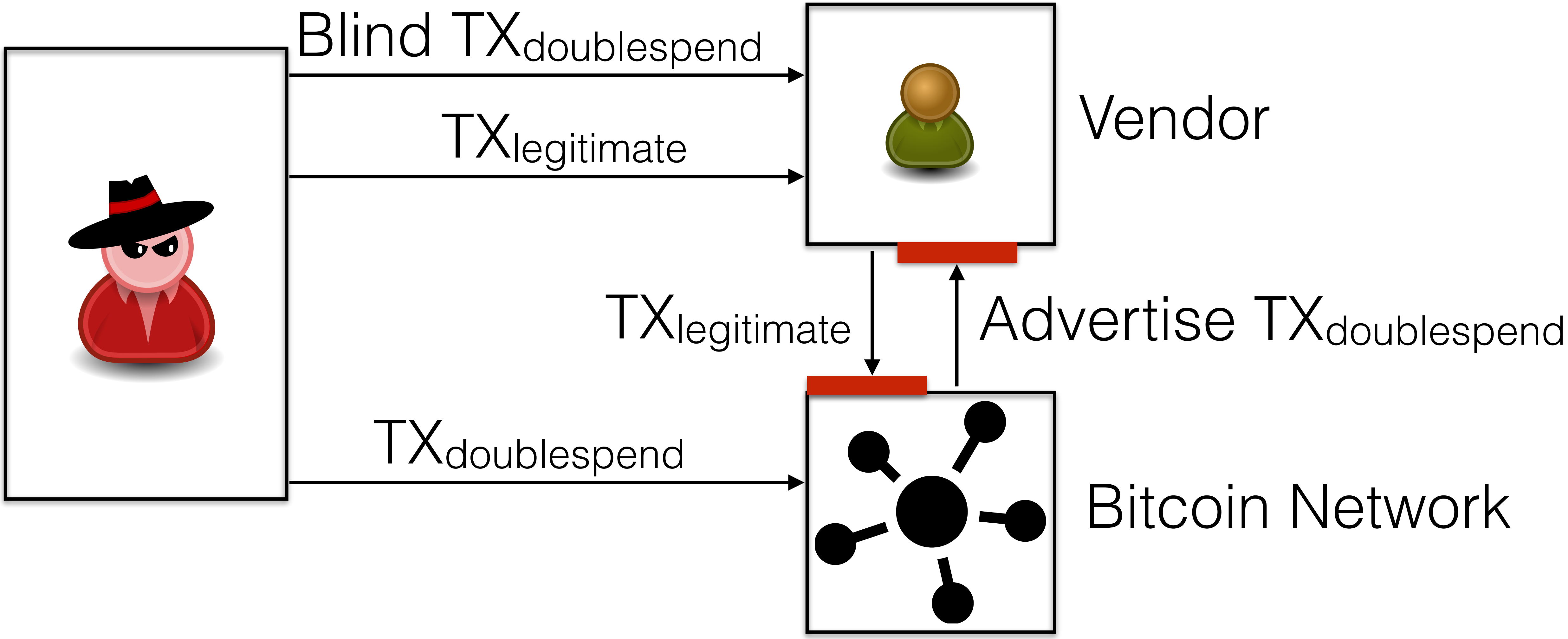
Double Spending 0 Confirmation Transactions

- Very reliable attack
- Regardless of protection (double spend relay)



Double Spending 0 Confirmation Transactions

- Very reliable attack
- Regardless of protection (double spend relay)


Blind $\text{TX}_{\text{doublespend}}$
$\text{TX}_{\text{legitimate}}$
Vendor
$\text{TX}_{\text{legitimate}}$
Advertise $\text{TX}_{\text{doublespend}}$
$\text{TX}_{\text{doublespend}}$
Bitcoin Network

Denial of Service

6000 reachable Bitcoin nodes

Preventing the delivery of blocks to these

- 450 000 TCP connections required
- 600 KB of advertisement / block / 20 min

Implications

Denial of Service

6000 reachable Bitcoin nodes

Preventing the delivery of blocks to these

- 450 000 TCP connections required
- 600 KB of advertisement / block / 20 min

➡ Network wide Denial of Service

Hardening the P2P overlay network

1. Why not smaller static timeouts?

2. Why not requesting from multiple peers?

3. What about alternative relay networks?

Security vs Scalability tradeoffs

Countermeasures

Hardening the P2P overlay network

1. Dynamic timeouts

Hardening the P2P overlay network

1. Dynamic timeouts

2. Handling Transaction Advertisements
   - **Filtering by IP address**
   - Randomly choosing sender

➡ First request from one peer, then two, then three…

Countermeasures

Hardening the P2P overlay network

1. Dynamic timeouts

2. Handling Transaction Advertisements
   - **Filtering by IP address**
   - Randomly choosing sender

➡ First request from one peer, then two, then three…

3. Updating Block Advertisements:
   - **Broadcast header instead of hash**
   - Keep track of block advertisers

Summary

Scalability measures impact security

Blind victim from blocks and transactions

- Minimum 20 minutes
- 1 connection sufficient per target

Summary

Scalability measures impact security

Blind victim from blocks and transactions

- Minimum 20 minutes
- 1 connection sufficient per target

Show impact

- Double Spending
- Aggravated selfish mining
- Affordable Denial of Service

Summary

Scalability measures impact security

Blind victim from blocks and transactions

- Minimum 20 minutes
- 1 connection sufficient per target

Show impact

- Double Spending
- Aggravated selfish mining
- Affordable Denial of Service

Proposal to harden the network

- Hardening measures
- Estimation of waiting time for secure transactions

Summary

Scalability measures impact security

Blind victim from blocks and transactions

- Minimum 20 minutes
- 1 connection sufficient per target

Thank you!

Show impact

- Double Spending
- Aggravated selfish mining
- Affordable Denial of Service

Proposal to harden the network

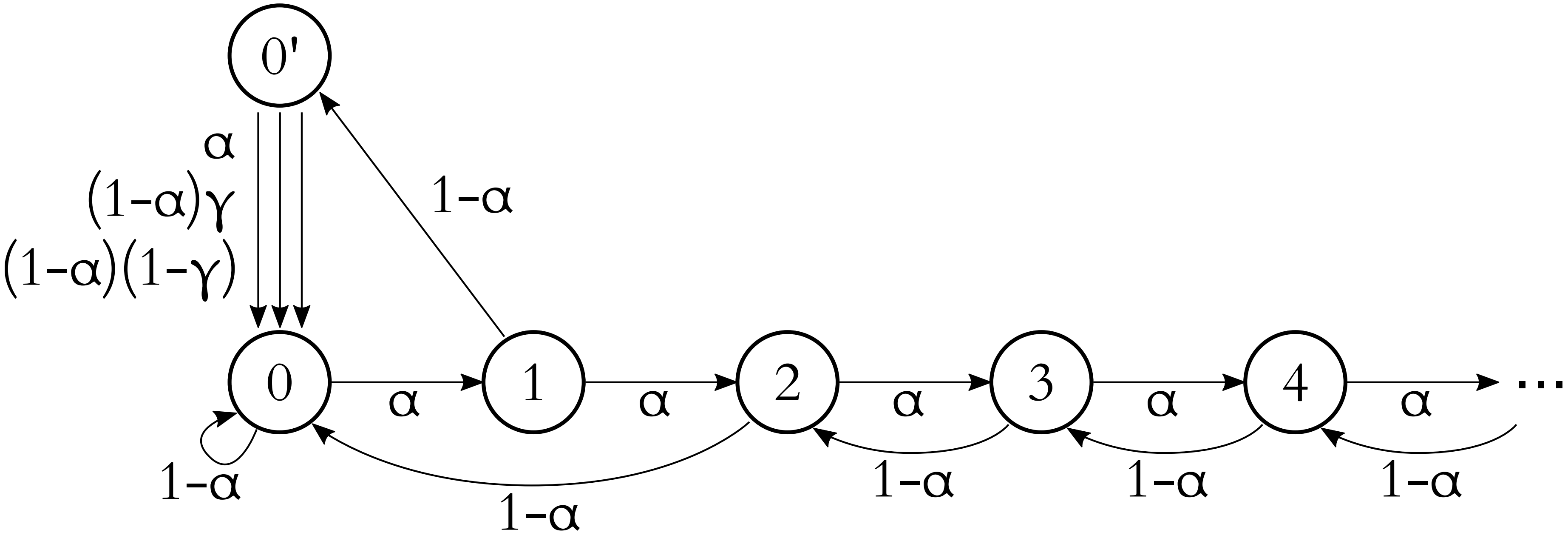
- Hardening measures
- Estimation of waiting time for secure transactions

Implications - Increasing Mining Advantage

Idea from Eyal et. al:

- Instead of publishing, keep a block private

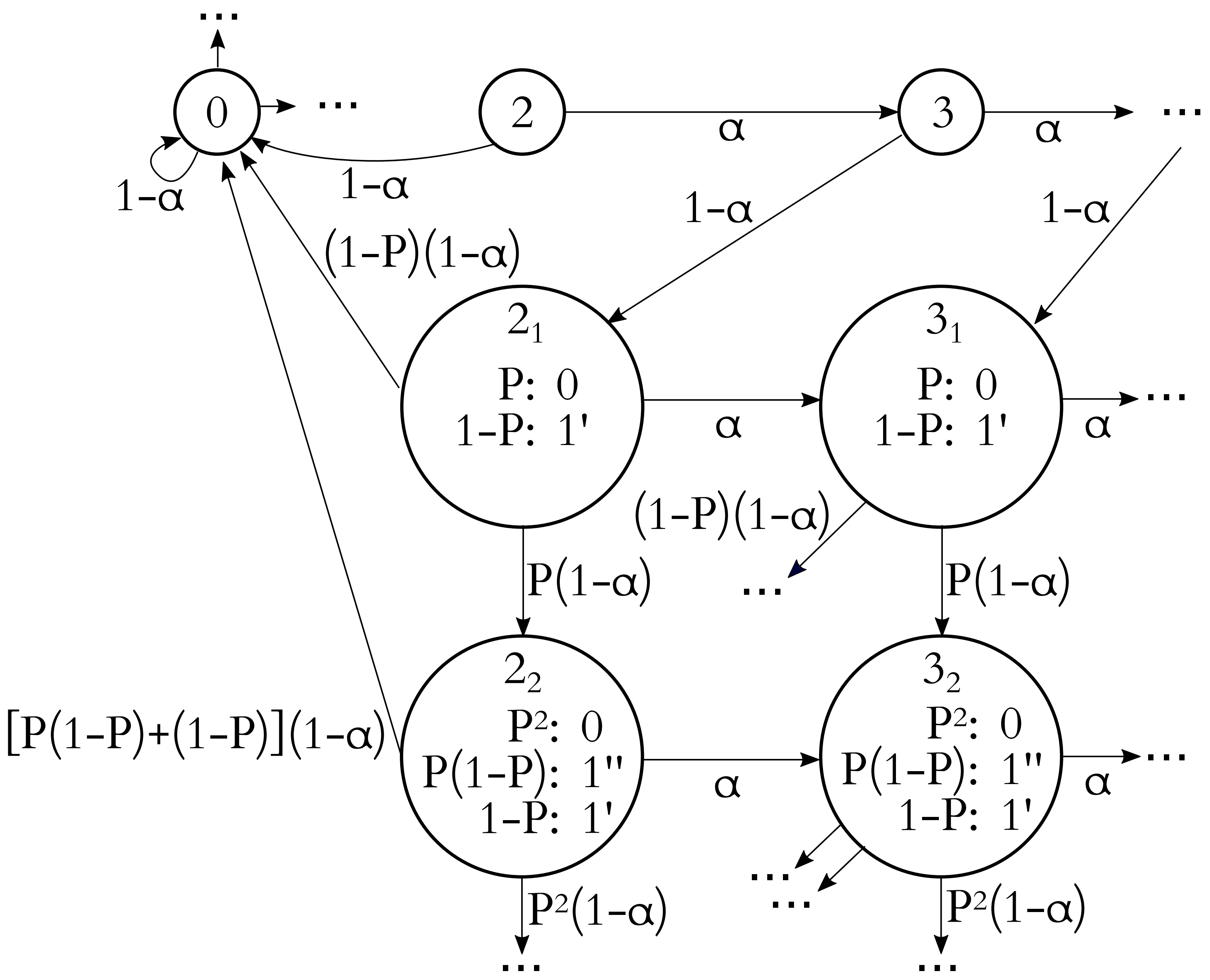
➡ Other miners will perform wasteful computations



α : hashing power of adversary

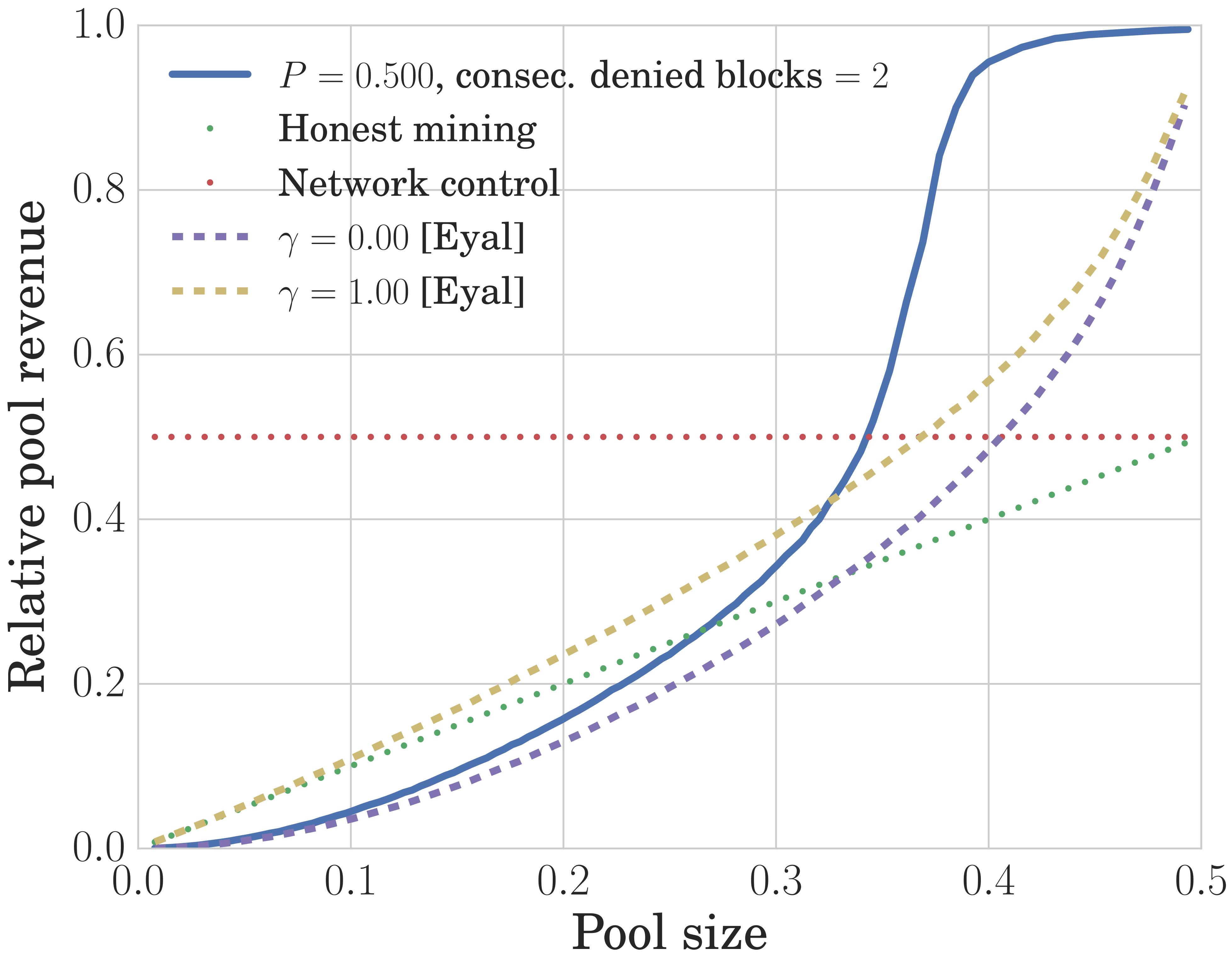
γ : propagation parameter

Implications - Increasing Mining Advantage

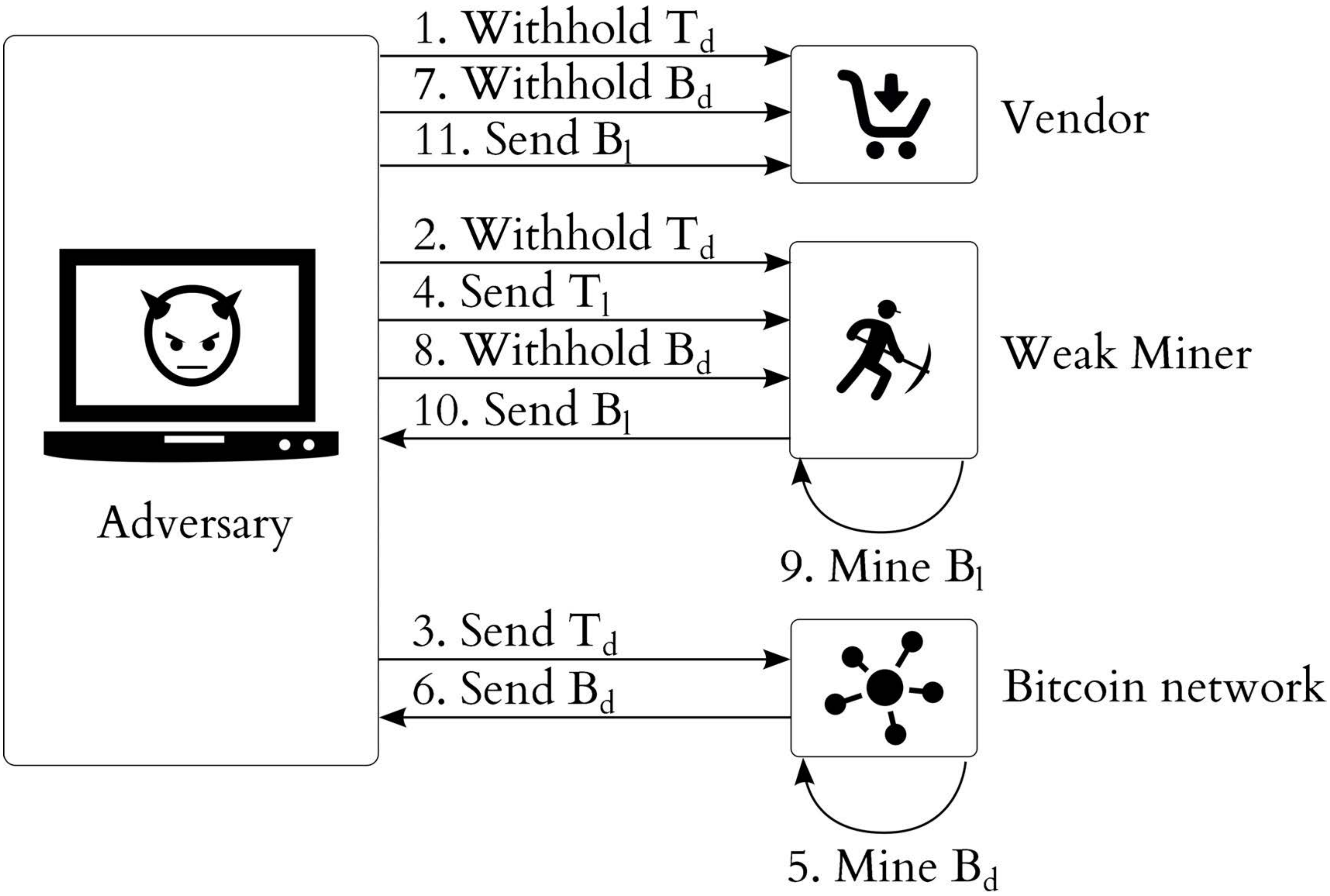


P: probability to deny a block to a miner

Implications - Increasing Mining Advantage



Implications - Double Spending 1 Confirmation Transactions



X**l** = legitimate
X**d** = double spend attempt

Hardening the P2P overlay network

1. Dynamic timeouts

2. Updating Block Advertisements:
   - **Broadcast header instead of hash**
   - Keep track of block advertisers

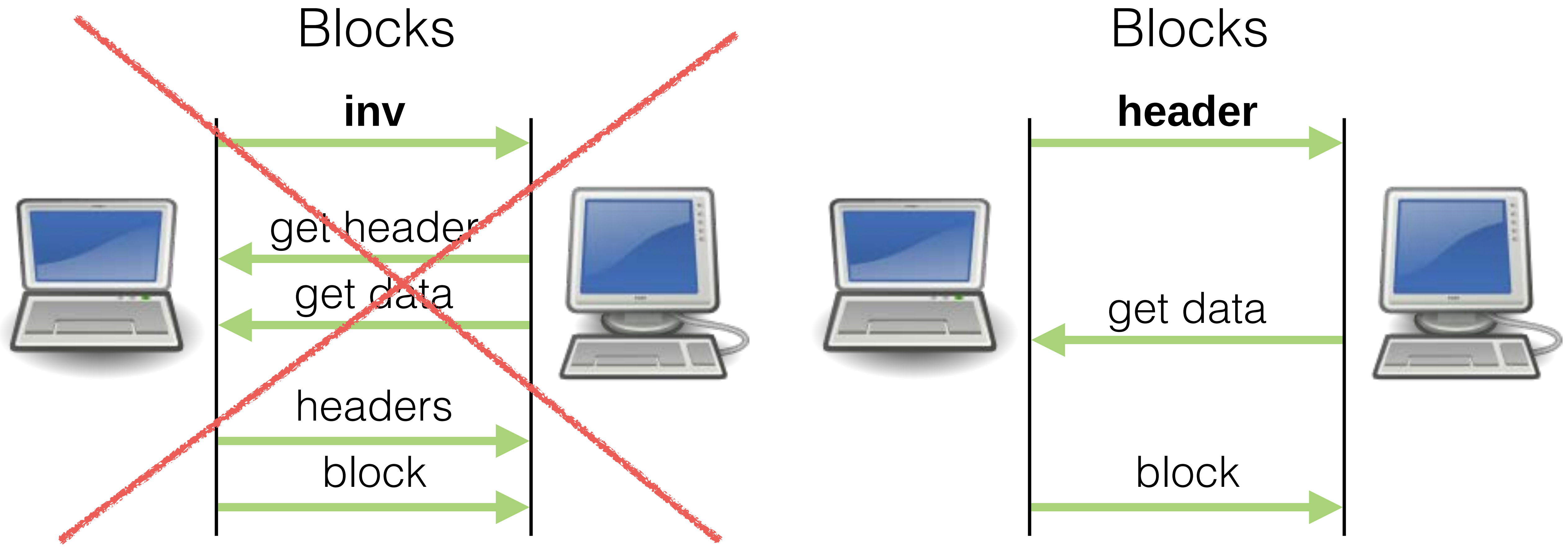
3. Handling Transaction Advertisements
   - **Filtering by IP address**
   - Randomly choosing sender

➡ First request from one peer, then two, then three…

Countermeasures

Hardening the P2P overlay network

Better Block request management



size(inv) = 36 bytes
size(header) = 80 bytes

Countermeasures

Hardening the P2P overlay network

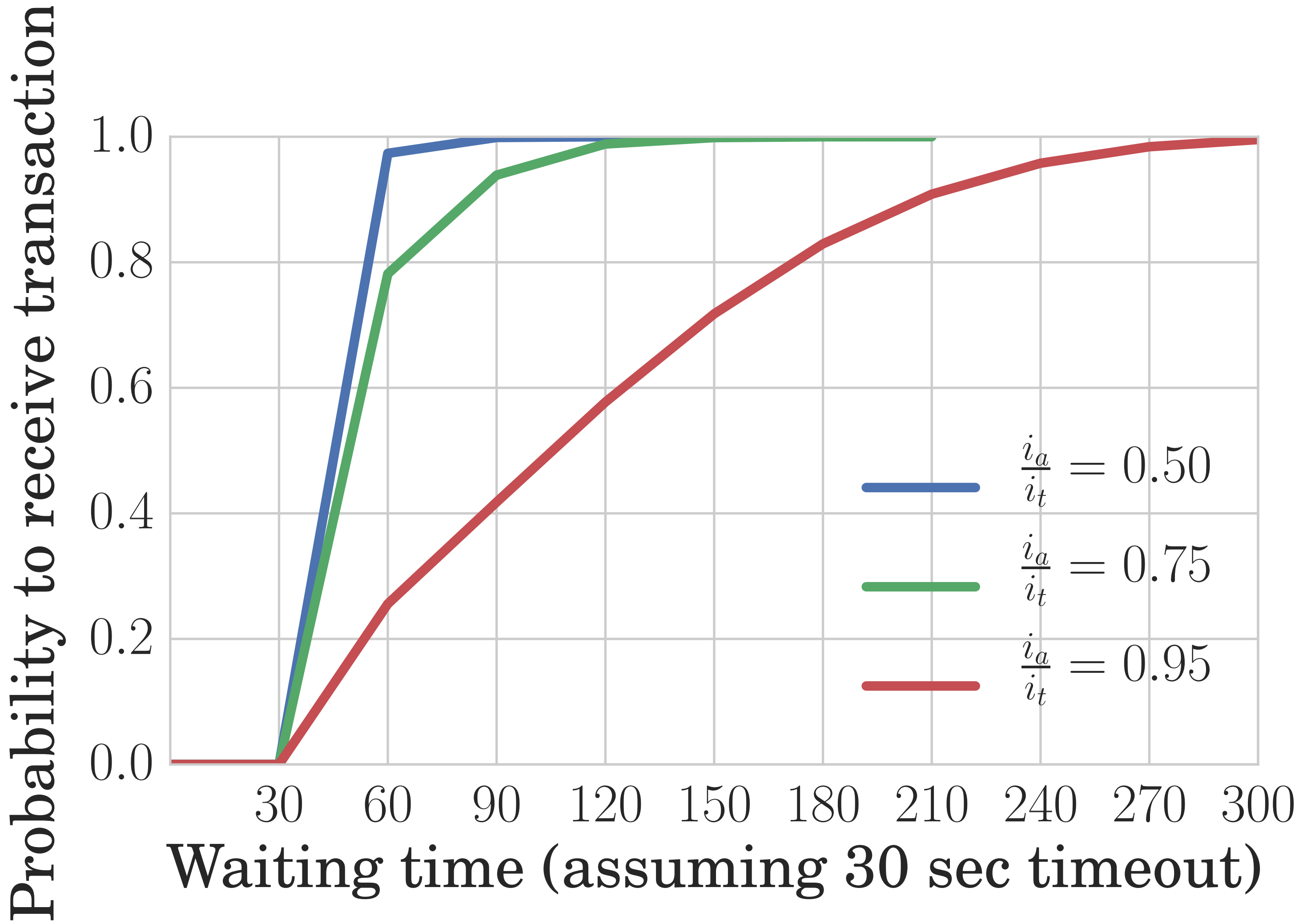
1. Dynamic timeouts

2. Updating Block Advertisements:
   - **Broadcast header instead of hash**
   - Keep track of block advertisers

3. Handling Transaction Advertisements
   - **Filtering by IP address**
   - Randomly choosing sender

➡ First request from one peer, then two, then three…

Hardening the P2P overlay network

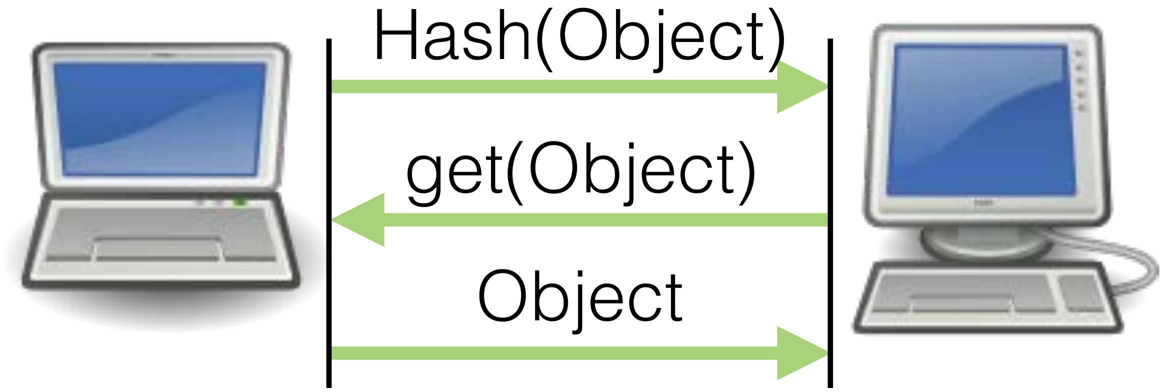


i_a = **inv** messages sent by adversary

i_t = total **inv** messages

After 5 minutes, transaction is received, even if the adversary controls 95% of the inv

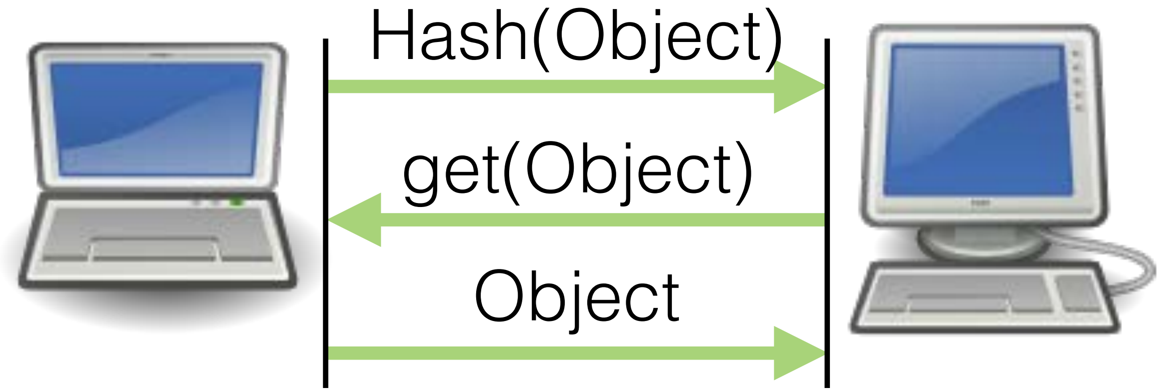
Advertisment-based request request management system



Blocks

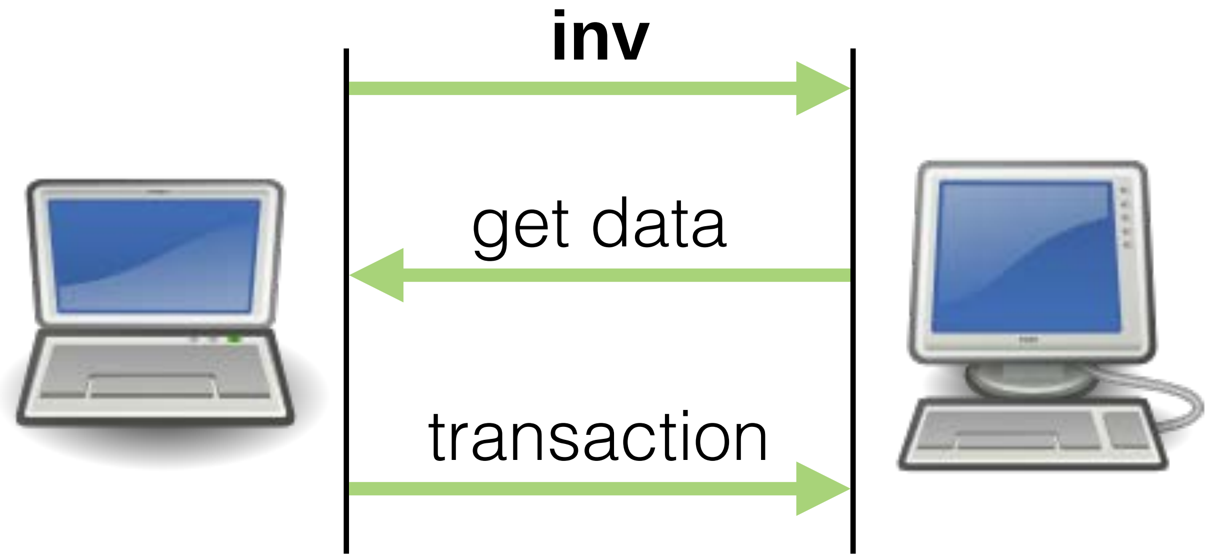
Transactions

Advertisment-based request request management system



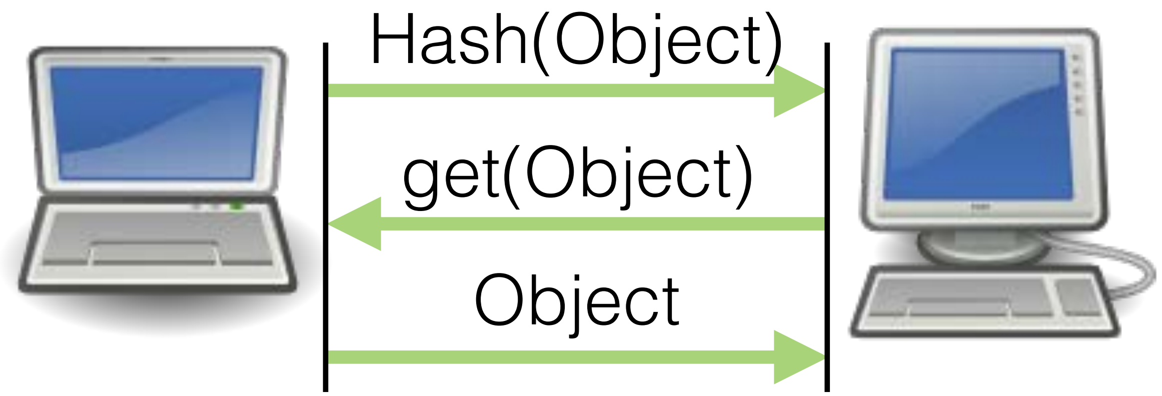
Blocks

Transactions

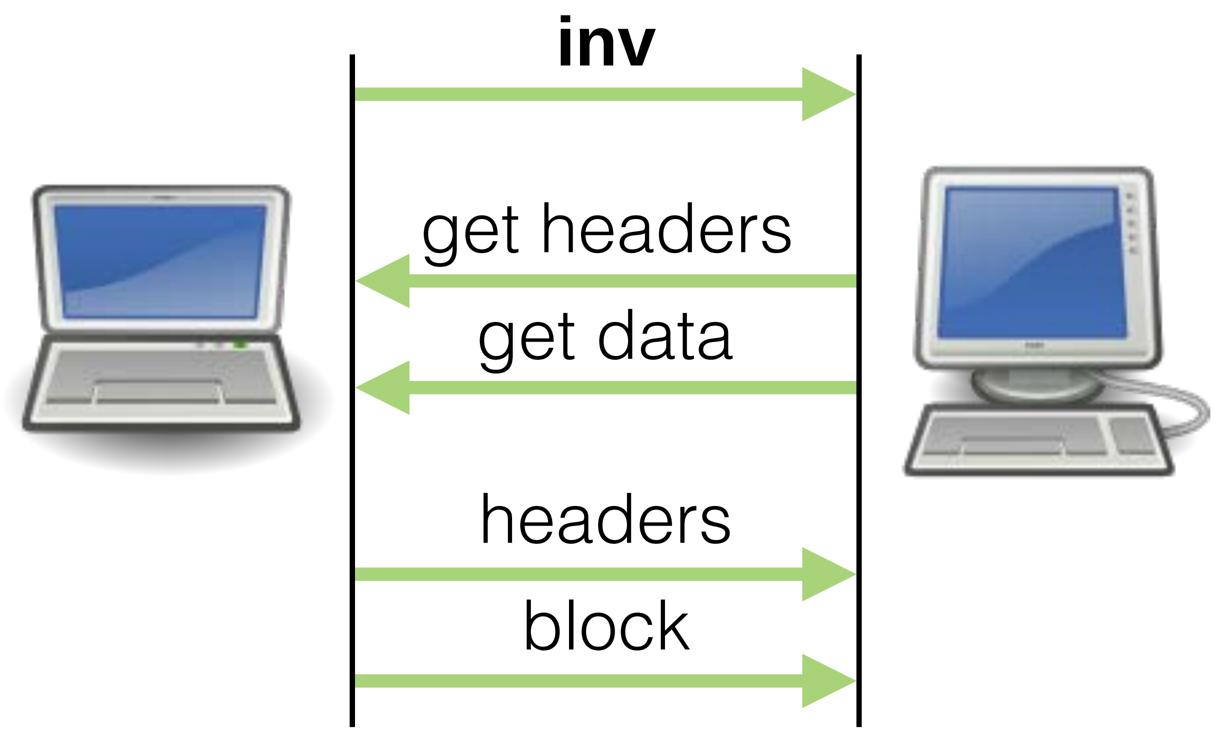


Main contributions

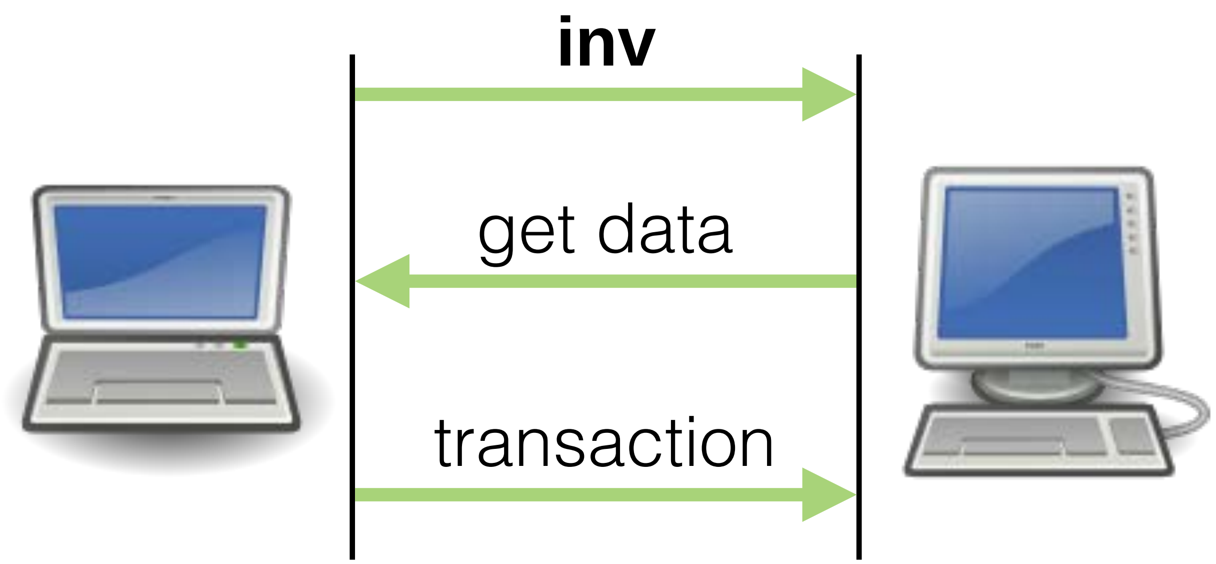
Advertisment-based request request management system



Blocks



Transactions


inv
get data
transaction