Sybil Deterrence Via Relay Friction

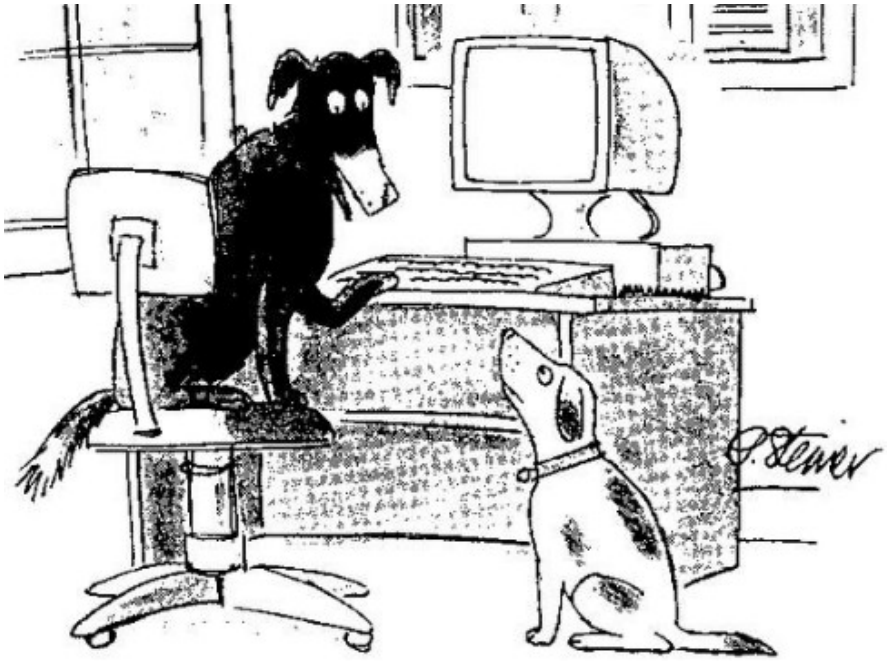
MAK
27-06-2019

Agenda

- Identity on the internet
- The sybil problem
- Understanding the threat
- Building towards a solution
- Final solution
- Questions

Identity on the Internet

- Lack of reliable identity on the internet
- Instead we have pseudo-identifiers
- Creating new identities
  - Easy
  - Cheap



"On the Internet, nobody knows you're a dog."

The Sybil problem

- Decentralization
  - Multiple participants
  - No central authority
  - Use “votes”
- Without a good robust identity system
  - Single person – multiple “votes”

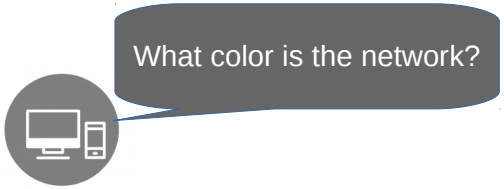
Only one of these is real

Understanding the threat

- Blockchains provide consensus
- Important to agree on a single version of history
  - Avoid forking
  - Resolve forks quickly when they occur
- Fork resolution in PoW
  - Depends on computation power

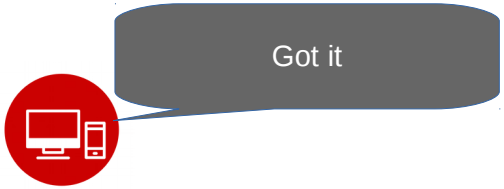
Understanding the threat

- Blockchains provide consensus
- Important to agree on a single version of history
  - Avoid forking
  - Resolve forks quickly when they occur
- Fork resolution in PoW
  - Depends on computation power



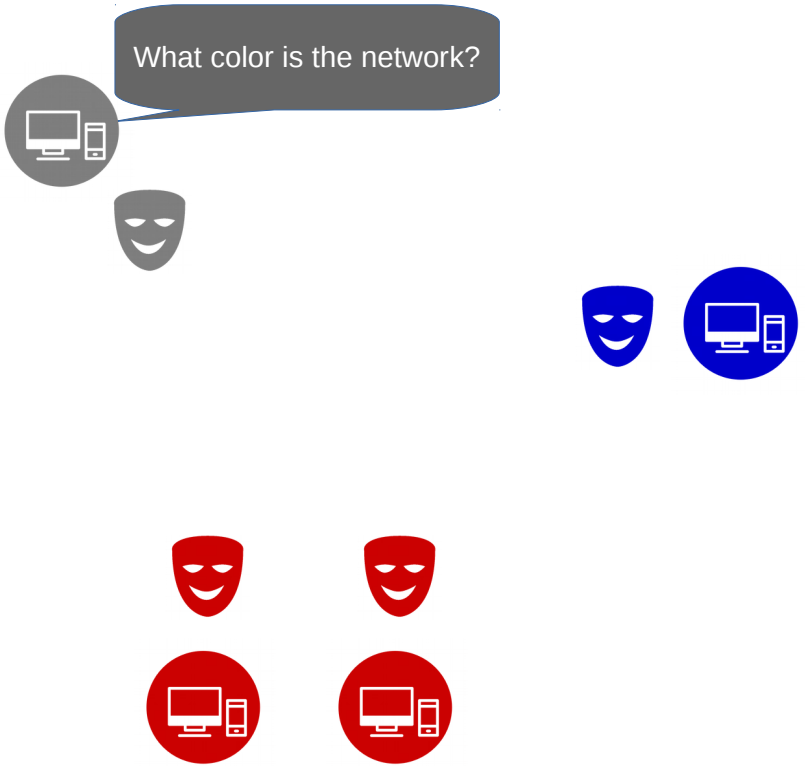
Understanding the threat

- Blockchains provide consensus
- Important to agree on a single version of history
  - Avoid forking
  - Resolve forks quickly when they occur
- Fork resolution in PoW
  - Depends on computation power
  - Verifiable delay ensures correctness



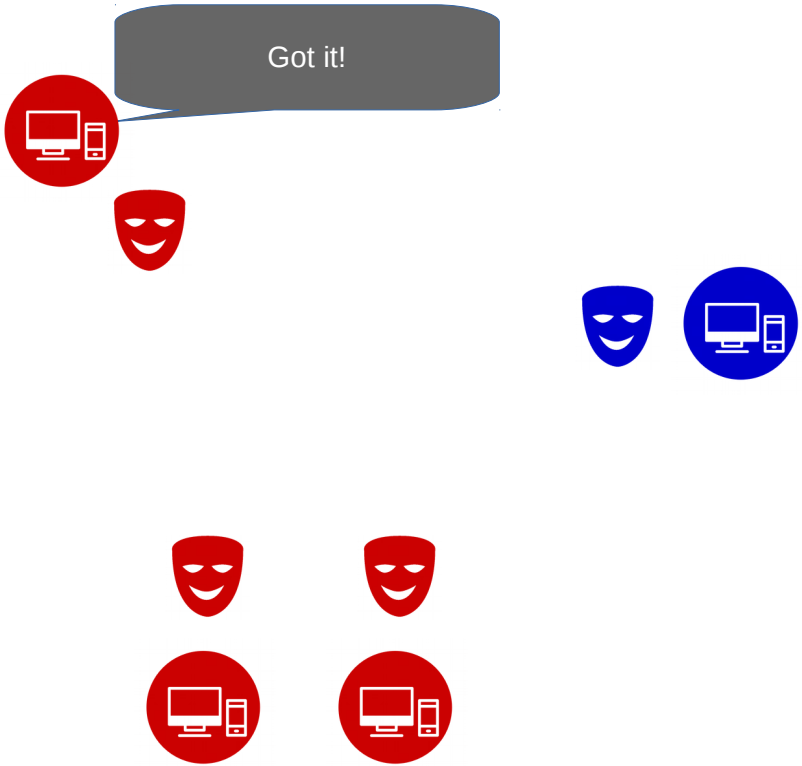
Understanding the threat

- Blockchains provide consensus
- Important to agree on a single version of history
  - Avoid forking
  - Resolve forks quickly when they occur
- Fork resolution in (D)PoS
  - Depends on votes by peers
  - No simple way to ensure correctness



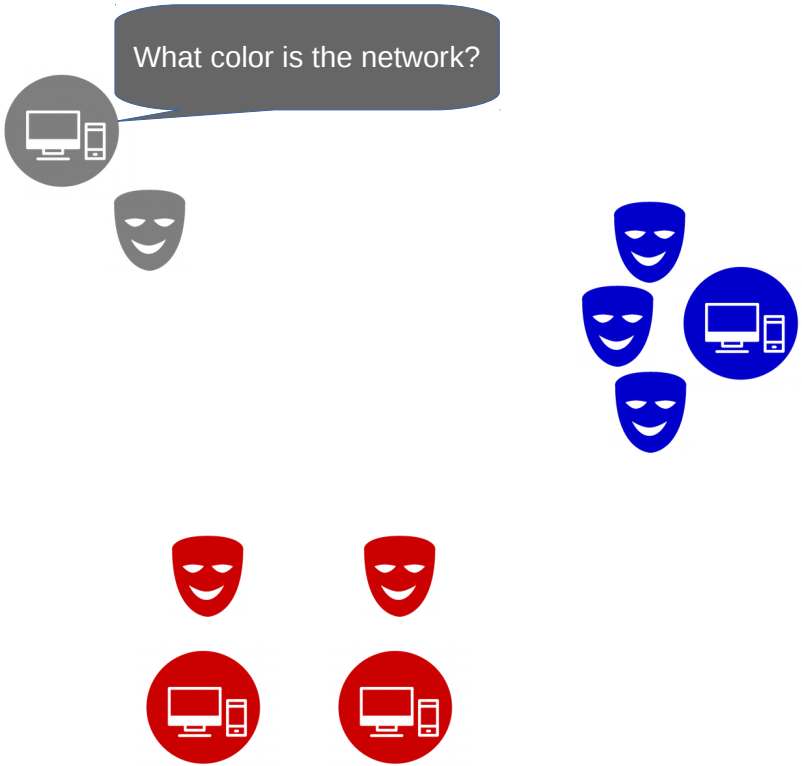
Understanding the threat

- Blockchains provide consensus
- Important to agree on a single version of history
  - Avoid forking
  - Resolve forks quickly when they occur
- Fork resolution in (D)PoS
  - Depends on votes by peers
  - No simple way to ensure correctness



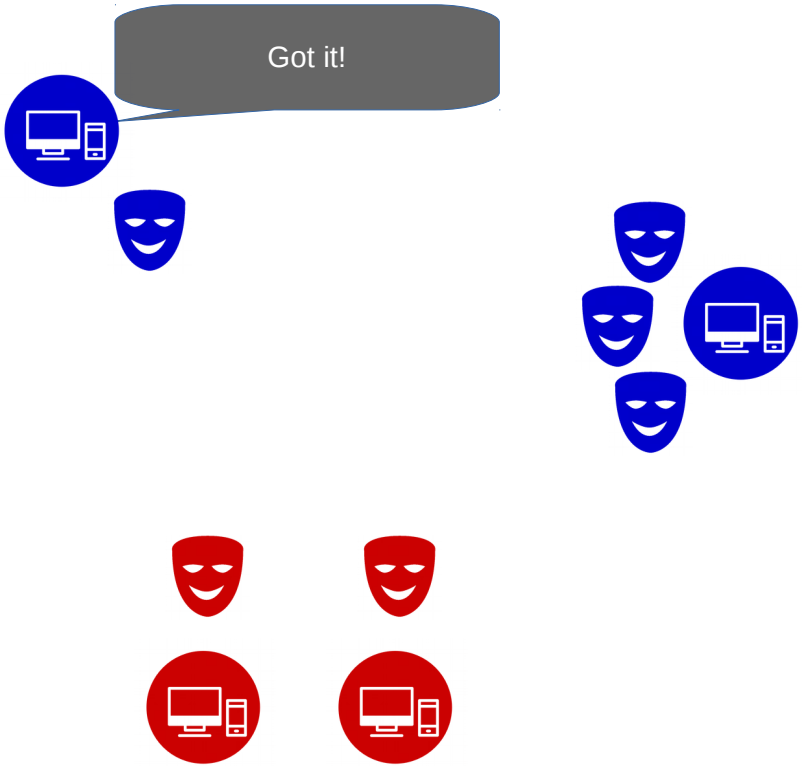
Understanding the threat

- Blockchains provide consensus
- Important to agree on a single version of history
  - Avoid forking
  - Resolve forks quickly when they occur
- Fork resolution in (D)PoS
  - Depends on votes by peers
  - No simple way to ensure correctness



Understanding the threat

- Blockchains provide consensus
- Important to agree on a single version of history
  - Avoid forking
  - Resolve forks quickly when they occur
- Fork resolution in (D)PoS
  - Depends on votes by peers
  - No simple way to ensure correctness



Goal

- To solve the Sybil problem in (D)PoS we want
  - expensive to maintain thousands of nodes
  - easy to detect the sybil nodes
  - easy to remove all the sybil nodes

Building up to a Solution

- Tie each relay to a small stake
  - Each relay operator must stake 5 ark
- If the stake is invalid or not present
  - ostracize peer from the network
- If relay provides bad information slash the stake

Shortfalls

- How can a SPV or API user check if the stake is valid?
- Solution:
  - Add a rare re-broadcast
  - One out of every N interactions with relay are broadcast out to the network

Shortfalls

- How would the relays know the data broadcasted is correct?
- Solution:
  - Relays must provide a signature with all the data they provide over API/SPV

Shortfalls

- How do we know the rebroadcast wasn’t censored by an eclipse attack or a sustained sybil attack?
- Solution:
  - Add PoW to make it difficult to sustain an eclipse attack

Shortfalls

- How do we know this PoW isn’t going to be used for a DDoS attack?
- Solution:
  - Add a PoW to clientside as well so that they need to match power

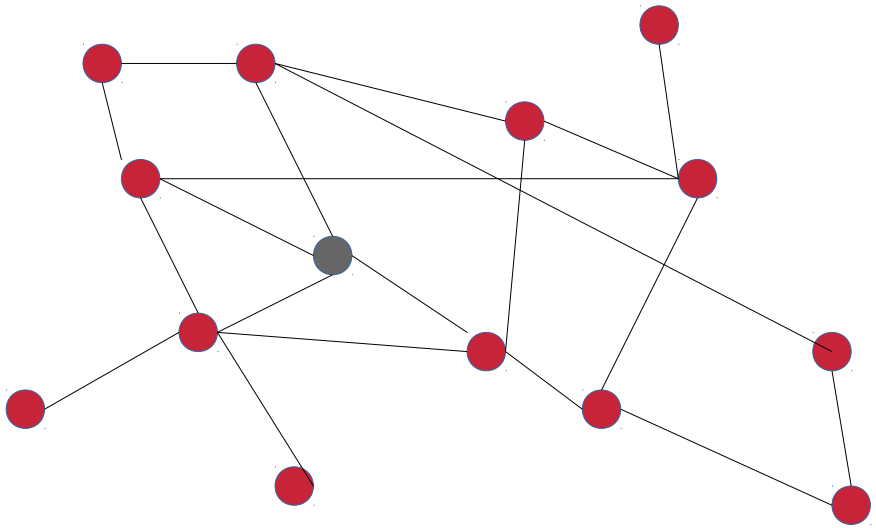
Solution

- Client connects to relay and both contribute to a nonce1 during handshake
- Client solves nonce1 to prove it is not trying a DDoS attack
- Relay creates required data with nonce2 that solves PoW and signs it
- Relay sends the data to client
- Client checks if the signature is valid
- If not valid then the response is discarded and client connects to a new peer
- Client checks if the PoW is valid

Solution (contd.)

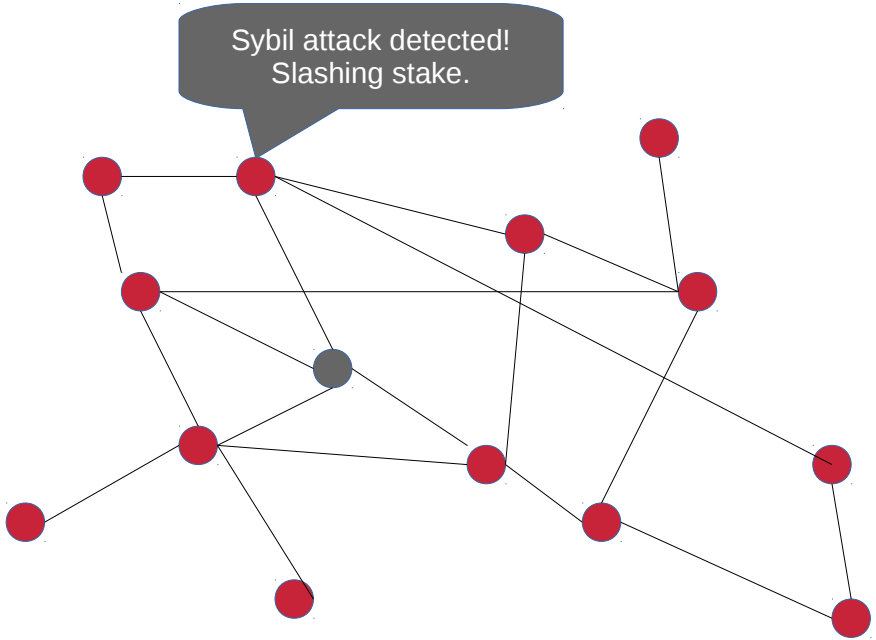
- If PoW is not valid then the data is broadcast to multiple nodes on the network
- If Pow is valid then one out of N chance that the data is broadcast to the network
- If Forging node receives a proof of invalid PoW or invalid data assume Sybil attack
- Slash stake
- Remove all nodes registered with the slashed stake

Example

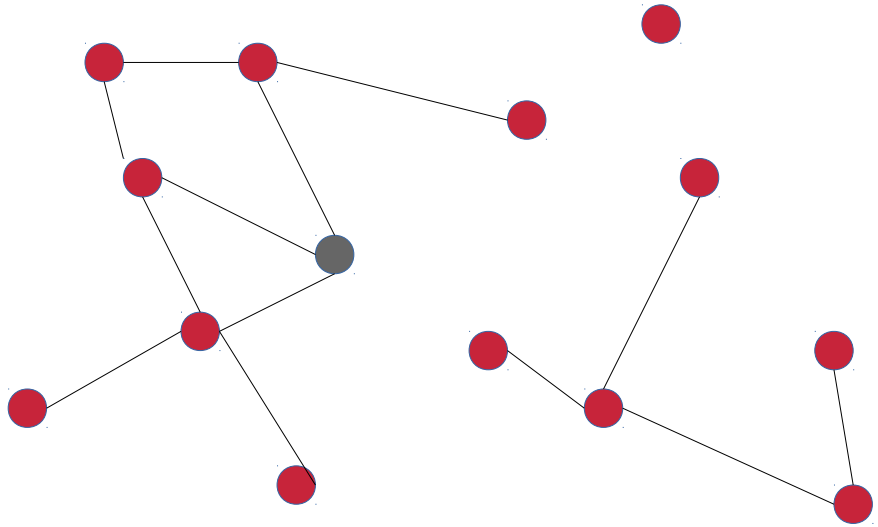


Is the seed server

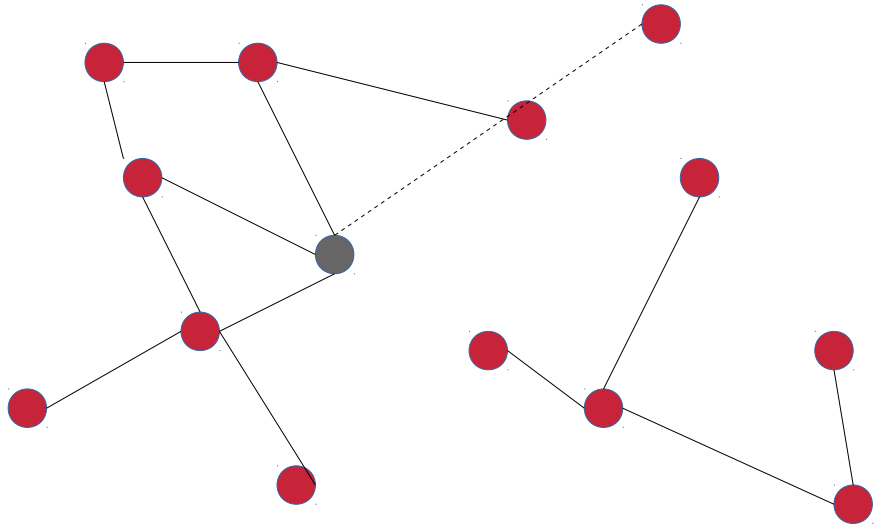
Example



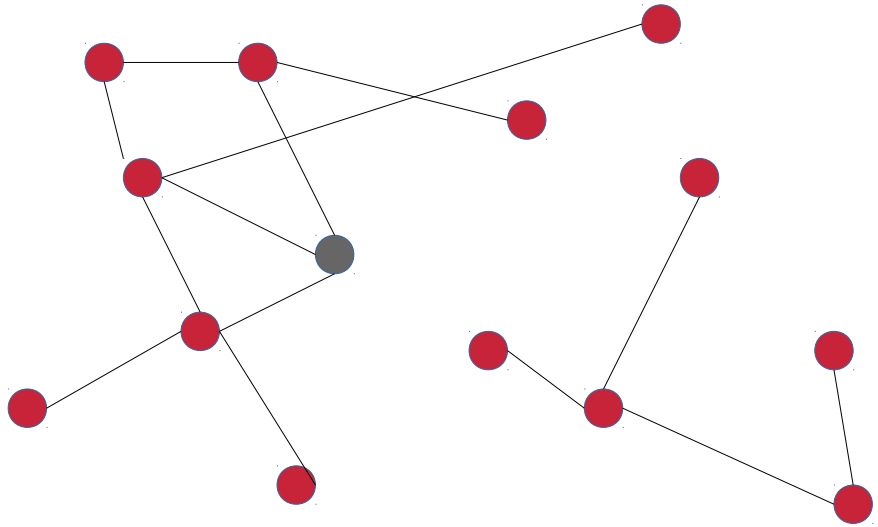
Example



Example



Example



Expected Results

- Network is more resistant to sybil attacks
- PoW doesn’t end up in an exponentially increasing race
- PoW requirement per request will go down as more nodes join the network
- Might need to provide incentive to run relays
- Fork resolution is as robust as a PoW driven chain (AIPs 25, 27, 85)

Questions?