Cristina Nita-Rotaru

7610: Distributed Systems

Bitcoin.

Acknowledgements

- Based on slides from David Evans, Aviv Zohar, Lefteris Kokoris-Kogias, Ittay Eyal, Alina Oprea

1: Money, money, money.

What is Currency?

- Medium of exchange
  - May or may not have intrinsic value (e.g. gold)
  - Value is based on future exchanges
    - Trust is essential
- Store of value
  - Allows one to store "value" rather than objects
  - Facilitates lending, debt, investing, and other financial innovations

Ancient Coinage

- Ancient coinage was based on precious metals
  - Typically copper, silver, and gold
- Value of currency based the intrinsic value of the metal
- Advantages:
  - Largely obviated the problem of counterfeiting
  - Lack of advanced mining techniques limited inflation
- Disadvantages
  - Cumbersome to move and store large amounts of currency
  - Limited capability to mint new coins (requires raw materials)
  - Currency's value may fluctuate wildly, i.e. through speculation

Paper Currency

- Paper currency began to replace coinage around 600 BC
  - Paper was exchangeable for a commodity, e.g. gold, copper, salt
- Advantages
  - Easier to carry, mint large denomination bills
  - Gives the issuing authority flexibility to inflate or deflate the currency
- Disadvantages
  - Counterfeiting
  - Runaway inflation – issuing authority may print too much paper (e.g. to pay debts)
  - Bank runs – gold supply may run out during a panic if everyone tries to convert

Fiat Currency

- Modern physical currency is based on fiat
  - **Value is backed by the government that issued it**
  - Not linked to a hard commodity, based entirely on trust
- Why does fiat currency have "value"?
  - Social contract – everyone accepts the currency, therefore it has value
  - Centralized power – the government has the power to enforce taxation, and they accept currency as a means to pay taxes
- All the advantages of commodity-backed paper currency, without the need to stockpile commodities
- All the disadvantages of commodity-backed paper currency, but even more trust must be placed in the issuing authority

Physical Currency

Advantages

- Easily portable
- Cannot double-spend
  - Spend the same piece of paper >1 times
- Cannot repudiate payment
  - Once you've given the paper, you can't get it back
- Semi-anonymous
  - Modulo tracking serial numbers
- Issuing authority can inflate/deflate as necessary

Disadvantages

- Easy to steal
  - Paper is a bearer token
- Hard to monitor/tax transactions
  - Paper is semi-anonymous
- Requires trust in the centralized issuing authority
- Doesn't work online
  - I can't email you a scan of the paper :(

What About Electronic Currency?

- The rise of telecommunications networks gives rise to a need for electronic forms of currency
- Credit cards, Paypal, bank e-checks
  - Computers store the amount of money held by each individual/company
  - Transactions move money between parties
- Why do we trust the electronic money system?
  - Denominated in a physical, fiat currency
  - Rules are enforced by strict regulation and audits
    - E.g. Paypal can't just decide to mint a trillion dollars for themselves
    - Trust is centralized in the issuing authority, i.e. the government
- What is the incentive to participate in the electronic money system?
  - Transaction fees
  - E.g. Visa charges 1.51% of each transaction plus $0.10

Electronic Currency

Advantages

- Works online
- Easy for issuing authority to monitor/tax/control
  - Strict regulations and auditing
- Cannot double-spend
  - All transactions are monitored
- Transactions can be repudiated
  - Credit card chargebacks

Disadvantages

- Requires trust in the issuing authority and third-parties
  - E.g. Visa, Paypal
  - Manual oversight and auditing
- Issuing authority may prohibit transactions and/or third-parties
  - Took Paypal years to become legitimized
- High transaction fees
- No privacy
  - All transactions are monitored

Towards Non-fiat Electronic Currency

- Goal: e-cash without a centralized issuing authority
  - Store and transfer value without a reliance on commodities or central banks
  - Anyone can join the system and be party to transactions
- Bitcoin is the first viable non-fiat electronic currency
  - Invented by Satoshi Nakamoto in 2009
    - Released as a whitepaper plus open-source implementation
    - Name is a pseudonym, true author is unknown
  - Built on top of an unstructured P2P system
    - Participants validate transactions and mint currency
    - System works as long as the majority of users (i.e. a quorum) are honest
  - Elegantly solves challenges in computer science and economics

2: A basic P2P coin

Why is P2P Currency Hard?

- The most obvious challenge is determining ownership
  - Who owns a given unit of currency?
- Without strong ownership…
  - Forgery – a user can mint arbitrary currency
  - Double spending – how do you validate and enforce transactions?
  - Theft – impossible to separate true and false claims about ownership

Build neucoin, our P2P coin

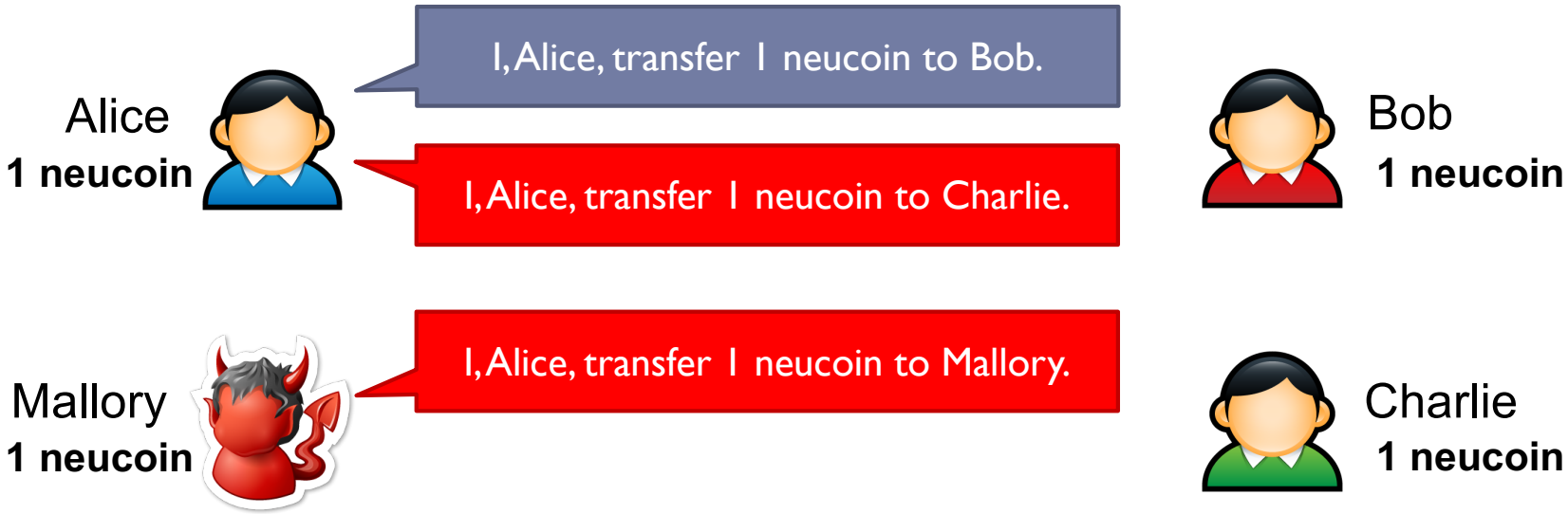
Requirements

- No centralized control
  - Central banks, governments, police
- No "strong identities"
  - ID cards, passports
  - Impossible to enforce without centralized control
  - Ideally, we would like anonymity (like physical paper cash)
- Entirely electronic
  - Not backed by commodities

Expectations

- Clear ownership of each neucoin
  - Cannot generate money you don't have
  - Cannot double spend
  - Cannot steal arbitrarily
- No repudiation

Motivating Example



Can transactions be forged?

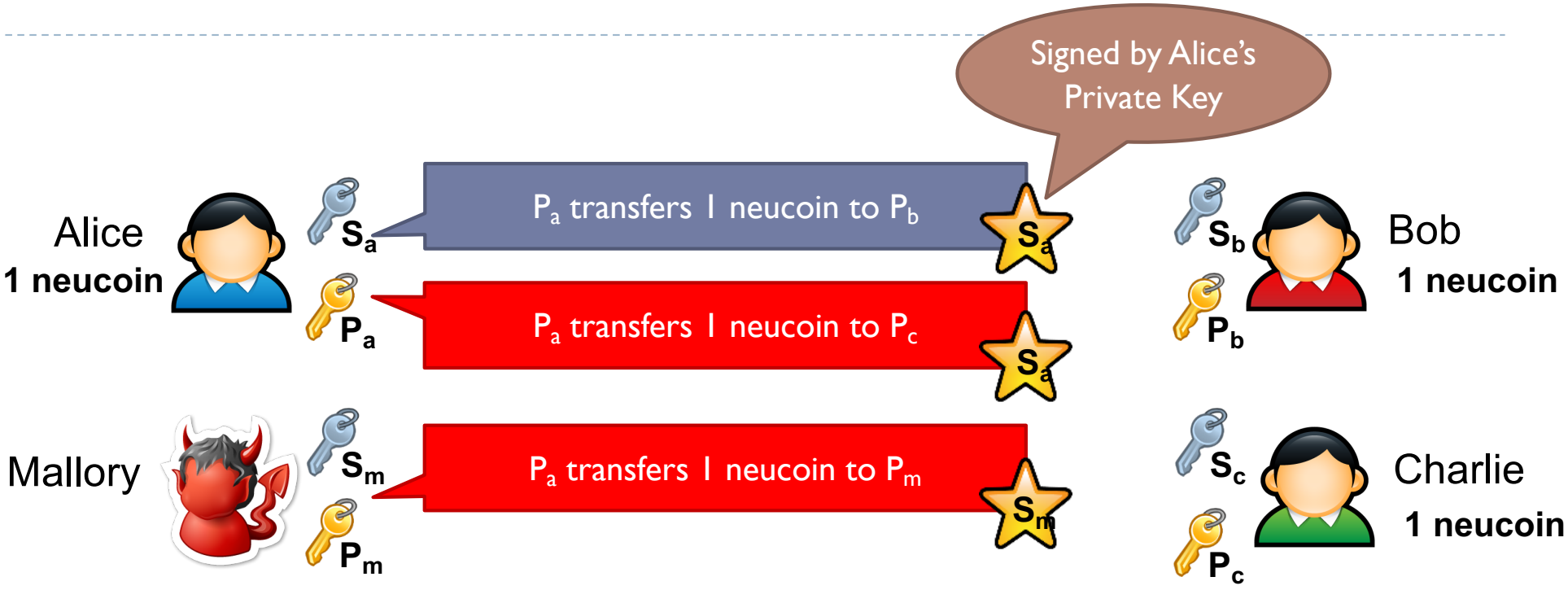
Can neucoins be stolen?

Can users double spend?

Introducing Cryptography

- Each entity in neucoin is defined by a public/private keypair
  - Simply referred to as wallets
  - Knowledge of private key gives ownership of neucoins in the associated wallet
- Coins are transferred from one public key to another public key

Example v2



Can transactions be forged?

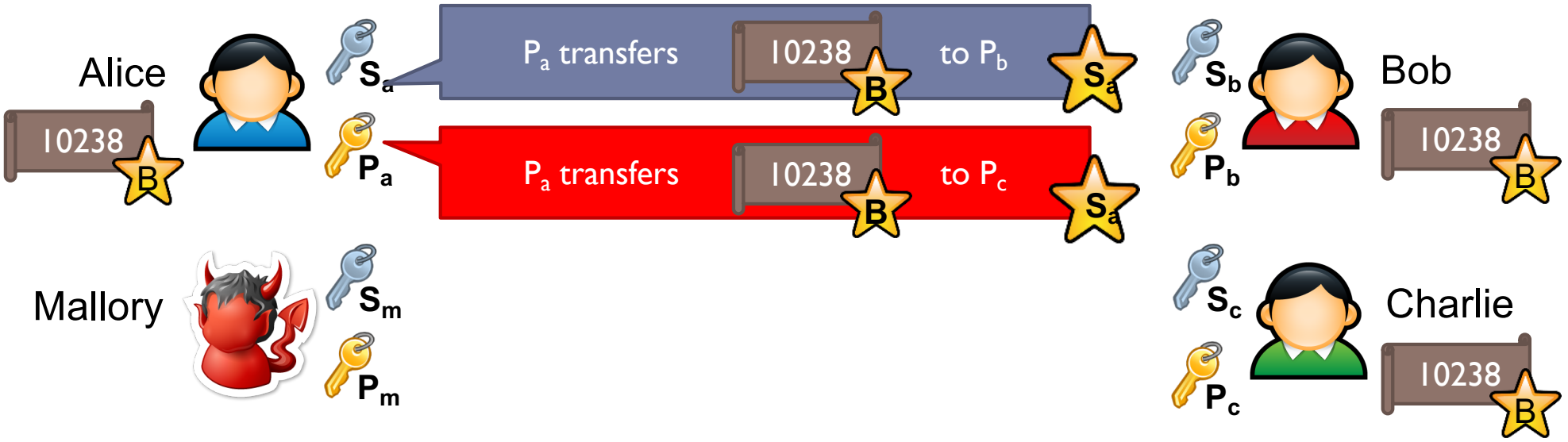
Can neucoins be stolen?

Can users double spend?

Serial Numbers

- How do we prevent users from minting arbitrary neucoins?
- Add a trusted third-party that issues serial numbers
  - Also known as a bank
  - Let's assume bank is trusted, has a well known public key

Example v3



Can transactions be forged? No

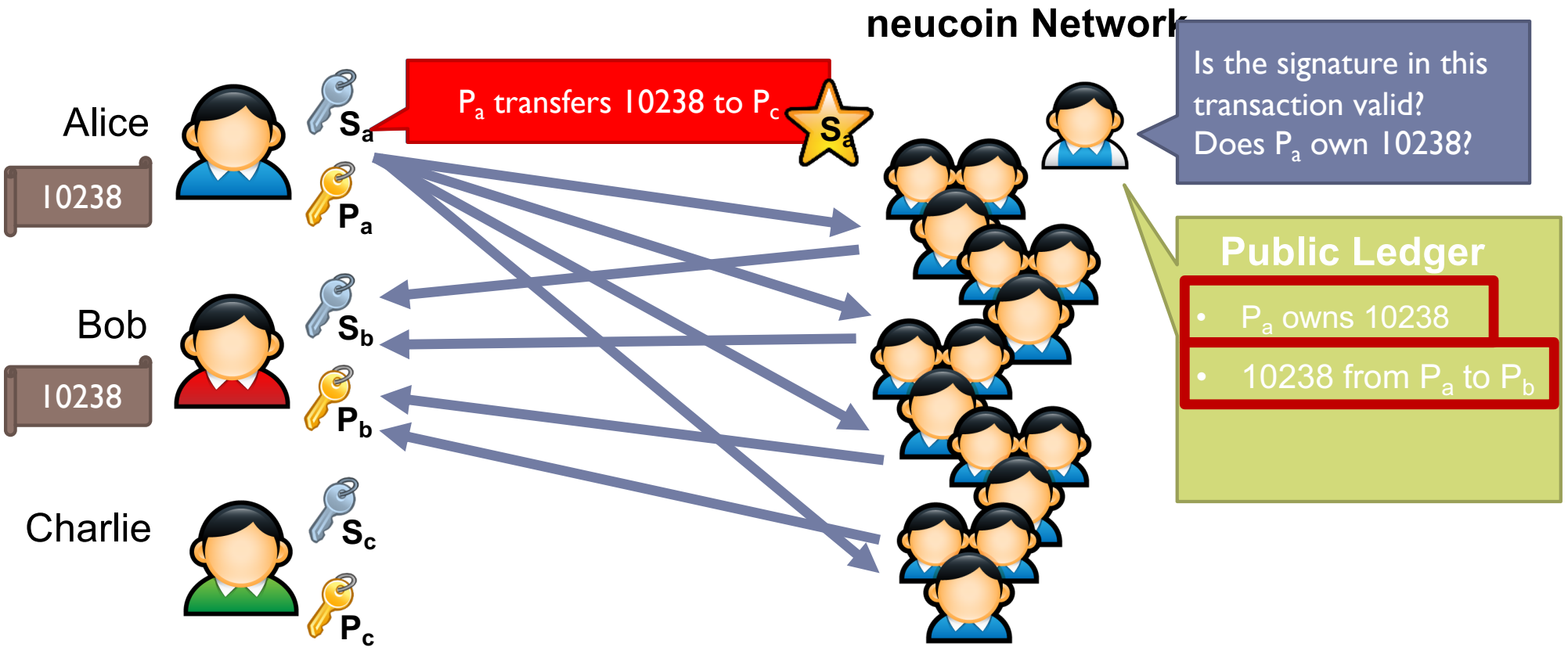
Can neucoins be stolen? No, assuming the private key isn't stolen

Can users double spend?

Preventing Double Spending

- What if the bank also tracked who owns each neucoin?
  - Bank would have a ledger, serve as official record of ownership
  - Charlie can contact the bank, verify that Alice owns a given coin
- Problems?
  - Centralized ledger totally defeats the purpose of neucoin
- **Instead, the network is the bank**
  - **Participants in neucoin collectively keep track of all transactions**
  - **Known as the public ledger**
  - **To verify that Alice isn't double spending, Charlie can check the public ledger**

Example v4


neucoin Network
Alice
10238
Bob
10238
Charlie
S_a
P_a
S_b
P_b
S_c
P_c
P_a transfers 10238 to P_c
S_a
Is the signature in this transaction valid?
Does P_a own 10238?
Public Ledger
• P_a owns 10238
• 10238 from P_a to P_b

3: Bitcoin

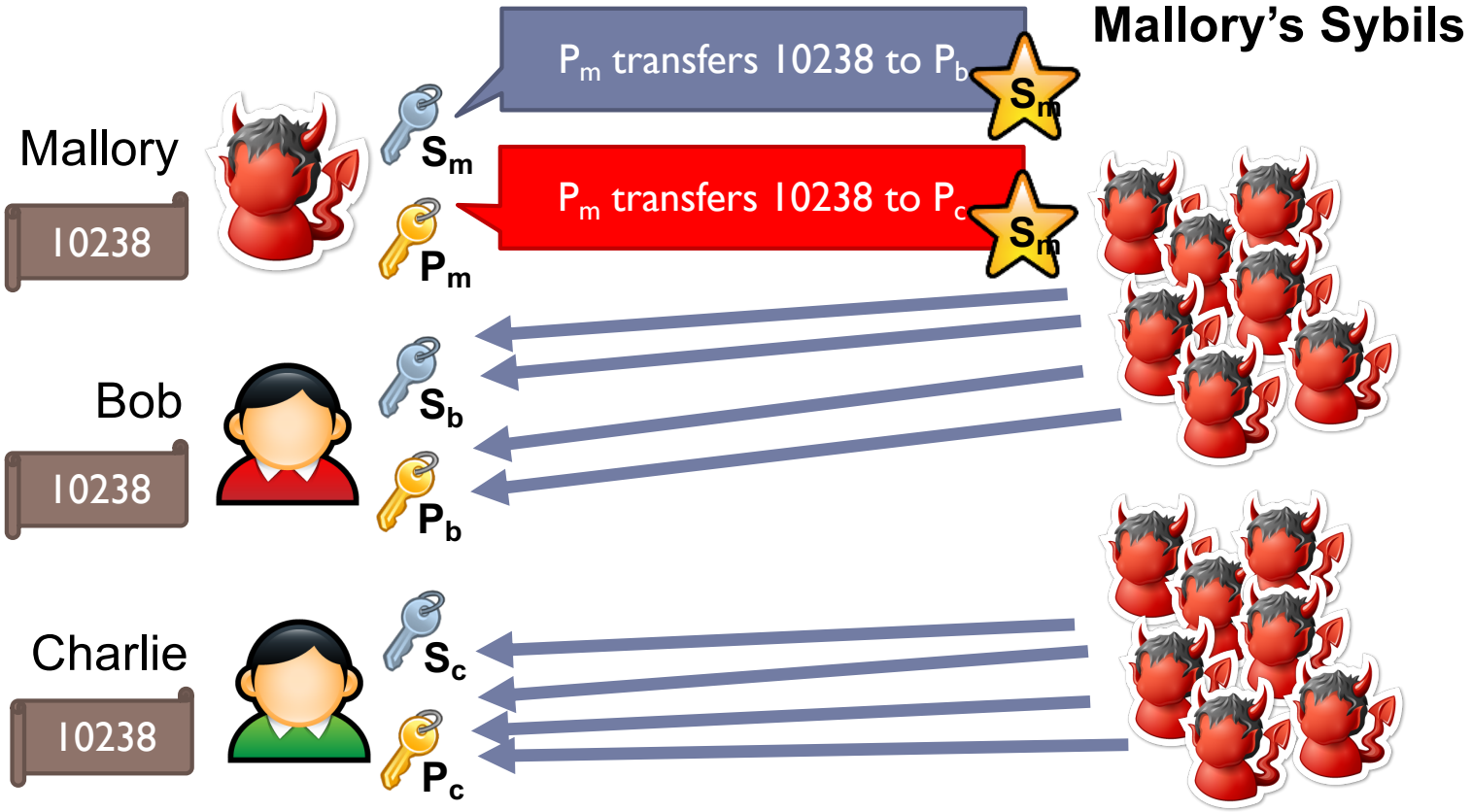
Bitcoin at a High-Level

- Bitcoin is a P2P network of nodes (Bitcoin clients)
  - Each node keeps a log (ledger) of all Bitcoin transactions, ever
  - This log is known as the blockchain
- Transactions are broadcast to nodes in the P2P network
  - Nodes validate transactions (Correct signatures and correct ownership of coins)
  - Valid transactions are added to the blockchain and broadcast to other peers
  - Transaction is considered accepted once it appears in the blockchain
- Goal: all transactions are known and agreed upon by the network, i.e. we want global consensus of events in log

Open Questions

- How much of the network must commit a transaction to their local blockchain for it to be considered accepted?
  - $\lfloor N/2 \rfloor + 1$ may not be feasible if *N* is large
- Suppose there is a fork in the blockchain. How do you reconcile?
- How to defend against Sybil attacks?
- What is the incentive for users to help maintain the blockchain?
  - Bandwidth, CPU, and storage aren't free

Sybil Attack



- Classic attack that all P2P systems are vulnerable to
- Mallory can introduce many fake nodes into the network
- N/2 respond to Bob, N/2 respond to Charlie
- Implication: one node = one vote doesn't work

Proof-of-Work

- Need to tie voting to a resource that is hard to obtain
  - In the past, we would have used a commodity (e.g. gold)
  - Identity (e.g. passports) would work, but defeats the purpose
- Key idea: tie voting to control of **computational resources**
  - To add a transaction to the ledger, you must present a **proof-of-work**
  - Proof-of-work can be generated by solving a cryptopuzzle
- Why is this a good idea?
  - Cryptopuzzles prove that a node expended significant effort
  - Obviates the need for Sybil prevention

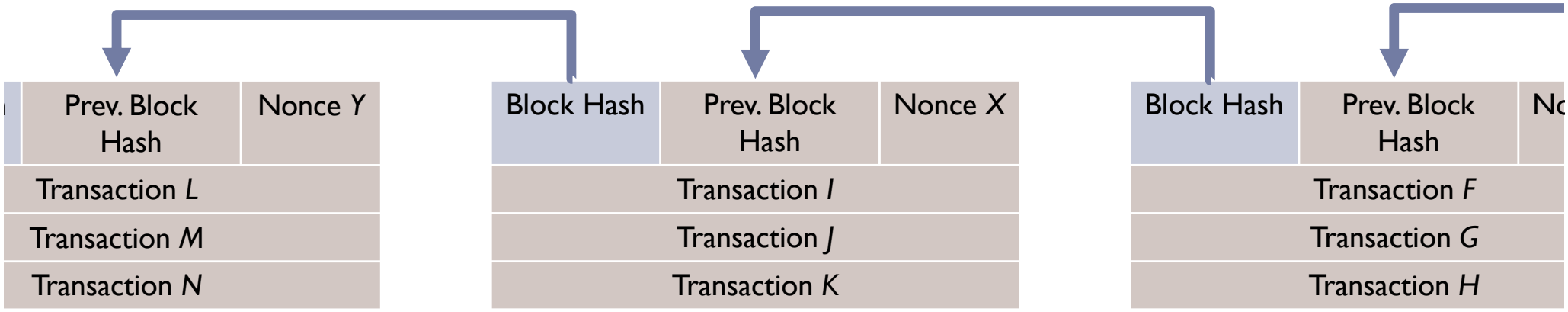
Cryptopuzzles

- Cryptographic hash functions, e.g. MD5, SHA-1, etc.
  - Deterministic, high entropy, and collision resistant
    - Locating A' such that hash(A) = hash(A') takes a long time
    - Example: 2^{21} tries for MD5
  - A hash function is cryptographically secure iff it is hard to find a pre-image (i.e. a collision) given a hash value *H*
- Implement cryptopuzzles in neucoin as follows:
  - Find *V* such that *Hash(V + [some other fixed data]) < target*
  - Nodes in the network have no choice but to brute force *V*
  - Difficulty can be adjusted by making *target* larger or smaller

Proof-of-work in Bitcoin

- Key idea: a node can only add an entry to the blockchain if it solves a cryptopuzzle
  - Other nodes can easily validate new blocks to ensure the puzzle has been solved
  - Changes "one node/one vote" to "one CPU/one vote"
    - To dominate the network, Mallory must control significant CPU resources
- Implementing proof-of-work in Bitcoin
  - Nodes receive transactions, group them into blocks
  - Nodes attempt to solve a cryptopuzzle based on the previous block and the new block
    - Blocks are linked, hence the name blockchain

Mining Blocks



- Block hash = Hash(Prev Block Hash | Nonce X | <Transactions>)
  - Concatenate all the fields in gray
- Nonce *X* is the number chosen to make the *block hash* < *target*
  - Changing the nonce changes the output of the hash function unpredictably
  - All nodes are competing to identify a nonce that solves the puzzle
- Creation of new blocks is known as mining

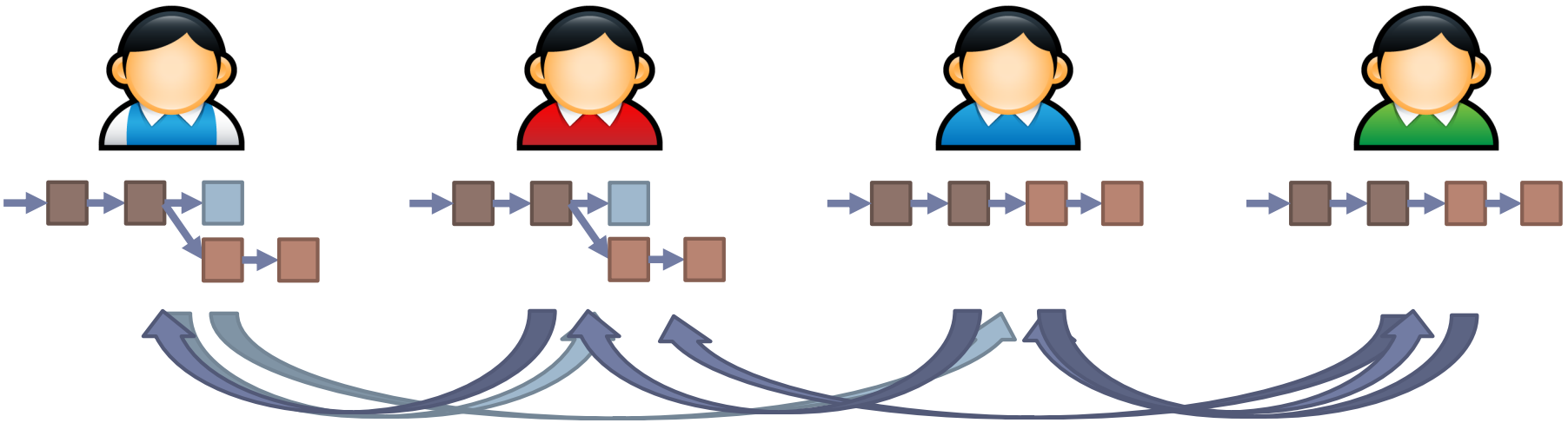
Building the Blockchain

- At any given time, all nodes are searching for the next block
  - **Searching = trying different nounces to solve the puzzle**
  - identify nonce *X* such that block hash < *target*
- Hashing power of the network grows over time
  - Bitcoin automatically adjusts *target* over time
  - Attempts to maintain 1 block/10 minutes on average
- Once a node finds a block, it broadcasts it to other peers
  - Other nodes validate (easy, simply recompute the hash)
  - Nodes begin working on the next block (with the new block as Prev Block)

Forking the Blockchain

- What happens if two nodes (Dave and Edgar) find different block simultaneously?
- Both Dave and Edgar broadcast
  - Some nodes will start working on Dave's fork, others on Edgar's
  - This is bad. Who is right? Dave and Edgar's blocks may contain different transactions
- In Bitcoin, nodes always accept the longest chain
  - The longest chain represents the most work
  - Eventually, either Dave or Edgar's fork will find the next block first
  - When this is broadcast, all nodes will switch to the longer chain

Forking Example



- In case of a fork, network searches for next block in both chains
- First chain to be lengthened "wins", all nodes eventually switch
  - Blocks in "loser" fork are ignored, transactions go back in the queue

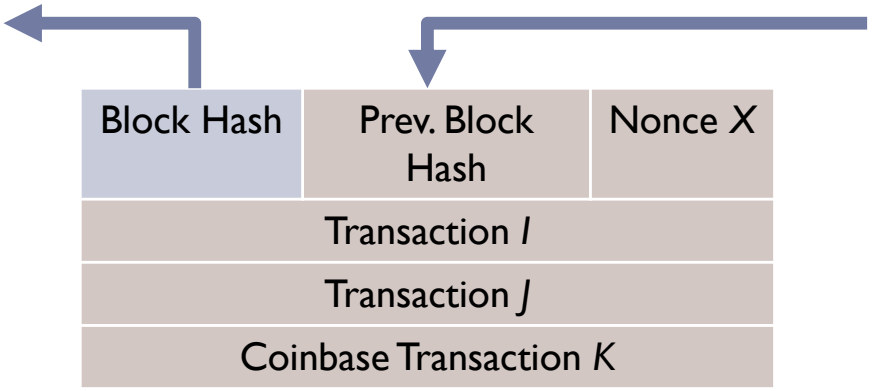
Open Questions, Revisited

- How to defend against Sybil attacks?
  - Require nodes to solve cryptopuzzles to commit transactions to the blockchain
- How do you reconcile if there is a fork in the blockchain?
  - Longest chain of blocks wins
- How much of the network must confirm a transaction for it to be considered accepted?
  - Only one node needs to solve the cryptopuzzle to create a new block
  - A single block may not be good enough for confirmation
- What is the incentive for users to help maintain the blockchain?

Creation of New Coins

- What is the incentive for users to act as Bitcoin nodes?
  - Computing hashes is CPU intensive
  - Bandwidth of communication with peers and users
  - Storage cost of storing the entire blockchain
- Bitcoin solves the incentive problem in two ways
  - Transactions may include a transaction fee
    - Paid to whoever "wins" first, i.e. mines a block that includes the transaction
  - New blocks mint new coins
    - Node who wins "mines" a fixed amount of coins as a prize
      - This is why it's called mining
    - Current reward is 12.5 BTC
    - Called a coinbase transaction

Coinbase Transactions



- Elegantly solves several problems
  - Where do bitcoins come from? How are they minted?
  - Who gets newly minted coins?
- Reward for mining halves every 210,000 blocks
  - Currently 12.5 BTC, was 50 BTC until 2012
  - Will become 0 in year 2140; 21 million total bitcoins
  - At this point, only transaction fees will incentivize miners

Do We Need Serial Numbers?

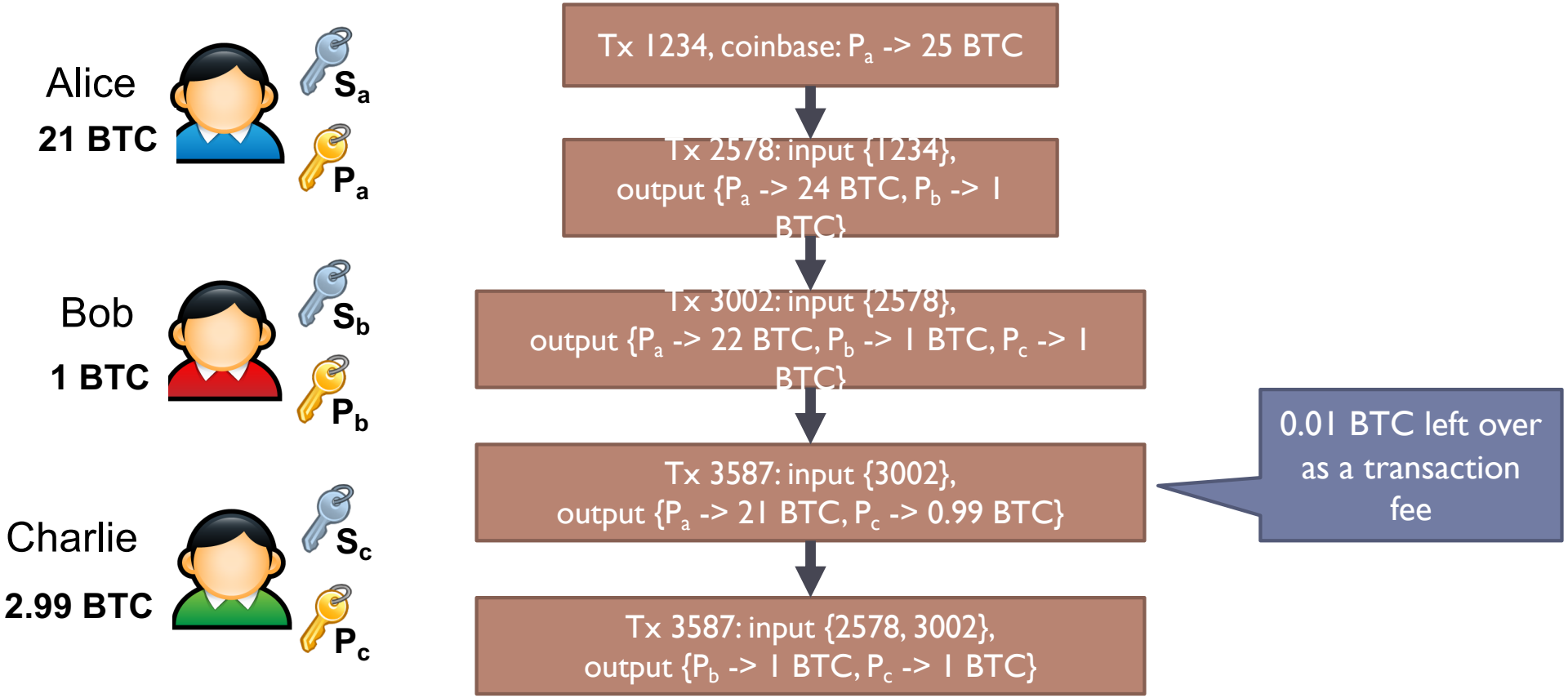
- Final annoyance: where do bitcoin serial numbers come from?
  - Trick question – bitcoins don't have serial numbers
- Idea: "bitcoins" don't matter, only transactions matter
  - All transactions have IDs, simply the hash of their fields
  - To transfer bitcoins
    - Simply state the transaction ID where you received the coins
    - Easy to verify that the transaction exists, as well as coin ownership (via the signature)
- What if you don't want to move all coins from a previous transaction?
  - Bitcoin transactions may have multiple inputs and outputs
  - Pay the change back to yourself ;)

Real Bitcoin Transactions

- Transactions have multiple inputs and outputs
  - Each input is the ID of a previous transaction
    - All value must be included
    - Nodes verify that no other transactions refer to the input IDs
  - Each output is an amount and a public key
    - Signed by owners private key
  - Implicit output: sum(input) – sum(output)
    - If it exists, can be claimed by the miner as the transaction fee

```
1.  {"hash":"7c4025...",
2.  "ver":1,
3.  "vin_sz":1,
4.  "vout_sz":1,
5.  "lock_time":0,
6.  "size":224,
7.  "in":[
8.    {"prev_out":
9.      {"hash":"2007ae...",
10.      "n":0},
11.    "scriptSig":"304502... 042b2d..."}],
12. "out":[
13.   {"value":"0.31900000",
14.    "scriptPubKey":"OP_DUP OP_HASH160 a7db6f C
```

Example Transactions



Is Bitcoin Secure?

- Can one fake a transaction? (i.e. steal your bitcoins)
  - No, he would need access to the victim's private key
- Can one edit the blockchain? (i.e. remove or modify old transactions)
  - No, all blocks are linked by their hashes
  - Changing historical block B_t would require changing all blocks $[B_t, B_{current}]$
- Can one repudiate a transaction? (i.e. deny that he paid you)
  - No, all of one's transactions are signed by his private key
  - Plus, one can't go back and change previous blocks
- Can one mint money out of thin air?
  - Yes, but only through legitimately solving a cryptopuzzle (i.e. a coinbase txn)
  - All peers can validate that the block (and the minted coins) are correct

What About Double Spending?

- Can I double spend?
  - Sort of – you could publish two new transactions with the same inputs
  - But, the network will only accept one of them *eventually*
- However, there may be a period of time where both new transactions are committed
  - If the blockchain forks, both transactions may exist concurrently
- When should a recipient conclude that a transaction is truly accepted?
  - Wait for *C* confirmations, i.e. blocks **after** the transaction in question
  - The longer the chain, the less likely it will be replaced by a fork
    - If a vendor requires 6 confirmations and an attacker controls 10% of the networks hashing power, the attack will succeed 0.02428% of the time

CPU Monopoly

- What if I control 51% of the networks hashing power?
  - All bets are off
  - Attacker can control all future blocks
    - Monopolize creation of new coins
    - Double spend
    - Deny arbitrary transactions
- Network needs to be diverse to prevent this kind of attack

Incentives, Revisited

- Why do nodes accept transactions?
  - Coinbase transactions and transaction fees
  - Essentially, monetary rewards

- Why do nodes "accept" a new block?
  - Couldn't they just ignore it and keep mining the old one?
  - No incentive: mining is guessing, so it's not like you are "close"
  - Also, all other nodes will switch to new block
    - To succeed, you now need to mine two blocks in a row

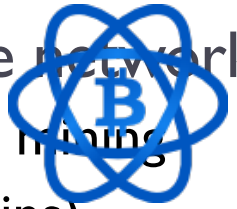
What About Anonymity?

- Are bitcoin transactions truly anonymous?
  - No, transactions are pseudonymous
  - Everyone is identified by their public key(s)
- The blockchain (all transactions ever) is public
  - Current owner of any coin can be determined
  - The provenance of any bitcoin can be ascertained (past ownership)
- Related question: are bitcoins fungible?
  - *Fungible* – mutually interchangeable
  - Physical dollars are fungible, one is just as good as another
  - Since the provenance of bitcoins can be tracked, they may not be fungible
    - E.g. if coin *X* is stolen, user may refuse to accept *X* in future transactions in retaliation

4: Bitcoin in practice

Bitcoin Wallets

- Two options
  - Desktop client: You participate as node in the network
    - Direct access to peers and the blockchain, but no mining
    - Private key on your machine (lose it, lose your coins)
  - Online Service: You give your private key to a company/site
    - Log in to site to view "balance", make transactions; easy to use
    - They have your key



Electrum

- What's up with the stolen bitcoins?
  - All from Wallet sites
  - Hackers break in, get private keys, transfer bitcoins to themselves

Anonymity

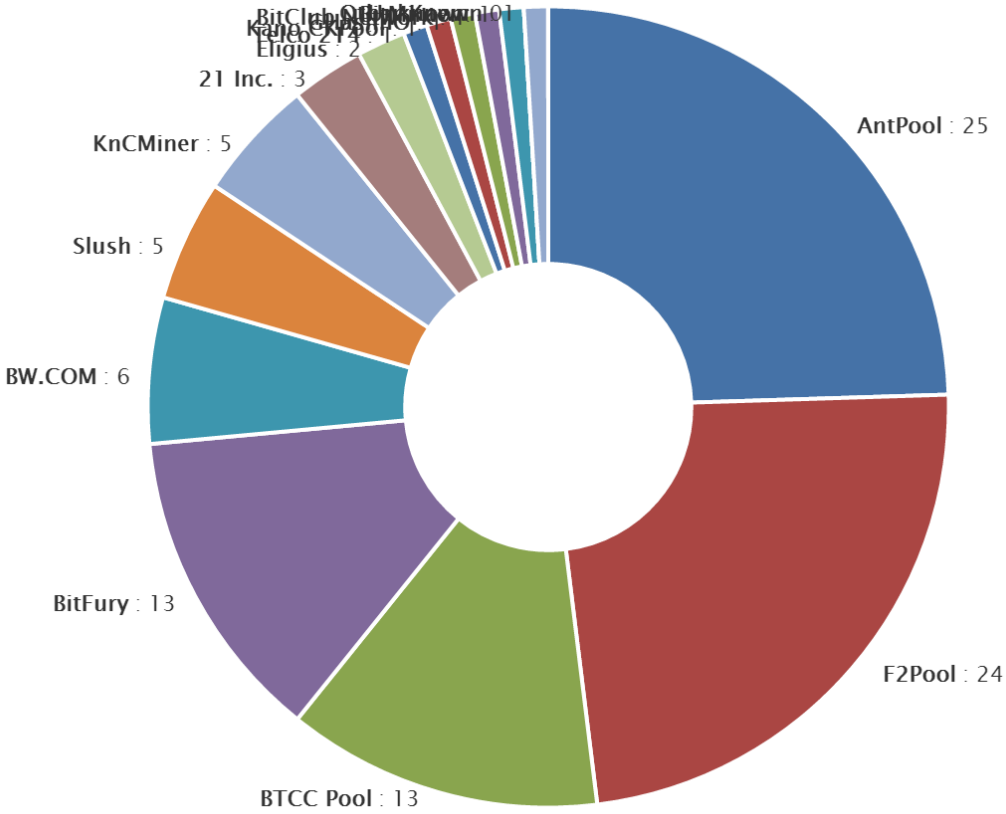
- Recall that bitcoin is pseudonymous
  - What if you want stronger anonymity?
- Best practice: generate a new wallet for every transaction
  - Users can have many wallet addresses (i.e. keypairs)
  - Makes linkability somewhat more difficult
- Even more anonymity: use a "mixing" service to launder your coins
  - Service uses a well known wallet address
  - Accepts bitcoins from any source
  - Return the coins after some time (minus a small fee)
  - Assuming volume is high, provides unlinkability of inputs and outputs

Mining

- In theory, anyone can download bitcoin and start mining
  - Your node will search for blocks
  - But in practice, you will *never win*
- Arms race: CPU < GPU < FPGA < ASICs
  - Real miners use thousands of chips custom designed to solve cryptopuzzles
  - Much faster and more power efficient than general purpose hardware
- Many miners operate in Iceland (cheap power and free cooling)

Mining Pools

- Problem: Bitcoin is a lottery
  - Very unlikely to win
  - Make it more "fair"?
- Solution: mining pools
  - Groups of nodes that work together
  - Split proceeds when any node finds the next block (more fair)
- Lots of mining pools today
  - Some represent up to 25% of mining capacity!



Proof-of-Work in Mining Pools

- How to evenly distribute coins in a mining pool?
  - How to determine what nodes "put in"?
  - Nodes could lie, say "I worked really hard!"
- Elegant solution: Nodes report "best hash" they found for block
  - i.e., they say "I didn't win, but here's the best I did"
  - Corresponds to amount of effort expended
- Payout distribution then based on how "hard" best hash was
  - Closer to target, more coins

Civil War

- Bitcoin functions because of its community
  - Now that bitcoin is popular, the community is large
  - To make changes, everyone must agree and adopt new software
- Blocksize and transactions
  - Currently, the max block size is 1 MB, and block rate is 1 every 10 minutes
  - This effectively caps the max capacity and speed of the bitcoin network
- To compete with Visa, bitcoin needs to handle more transactions at lower latencies
- How to solve these scalability challenges?
  - Bitcoin XT proposes to gradually increase the blocksize over time
  - Sidechains – push tiny transactions out of the main blockchain
  - Do nothing – let transaction fees rise for people seeking high-priority transactions

Altcoins

- P2P + cryptopuzzles + blockchain + incentives is a general formula
  - Can be applied to other distributed systems problems besides currency
  - Bitcoin is open-source
- Led to the creation of dozens of alternate coins
  - Litecoin – faster mining rate and scrypt instead of SHA-256 (harder to design ASICS)
  - NXT – designed for securities trading
  - Filecoin – designed for storing files in the blockchain
  - Namecoin – designed as a replacement for DNS
  - Dogecoin – litecoin + shiba inus

5: Bitcoin security

Bitcoin Protocol

- Each P2P node runs the following algorithm:
  - New transactions are broadcast to all nodes.
  - Each node (miner) collects new transactions into a block.
  - Each node works on solving proof-of-work (PoW) for its block
    - Use computational resources
  - When a node finds a solution, it broadcasts the block to all nodes.
  - Nodes accept the block only if all transactions are valid (digital signature checking) and coins not already spent (check transactions from public ledger).
  - Nodes express their acceptance by working on creating the next block in the chain
    - If multiple valid blocks are available, choose the longest chain and include transactions from discarded blocks in the queue
    - Include the hash of the accepted block as the previous hash.

Nodes eventually reach global consensus on all transactions

Bitcoin security

- Protection again *invalid transactions* (forgery)
  - Cryptographic (digital signature)
- Protection against *modification of blockchain* (remove or modify old transactions)
  - Cryptography (collision-resistant hash functions and digital signatures)
- *Non-repudiation of transactions*
  - Based on blockchain
- Protection against *double spending*
  - Enforced by consensus (correct majority)
  - One of the transactions (either one) will be eventually accepted
- Protection against *Sybil attacks*
  - PoW cryptographic puzzles
  - Assume that adversary does not control majority of CPU resources

Bitcoin: Security issues

- Consensus algorithm:
  - Is majority enough?
  - Can blocks be removed?

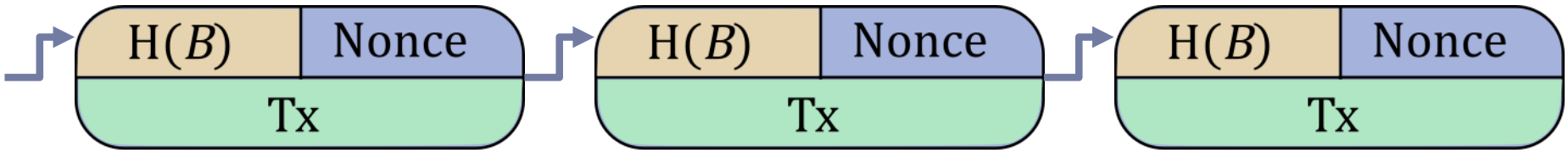
- P2P network:
  - What are the reliability and network connectivity requirements?
  - Are any of the attacks in P2P relevant in this context?

Selfish miners

Majority is not Enough: Bitcoin Mining is Vulnerable

Ittay Eyal, and Emin Gün Sirer

Mining

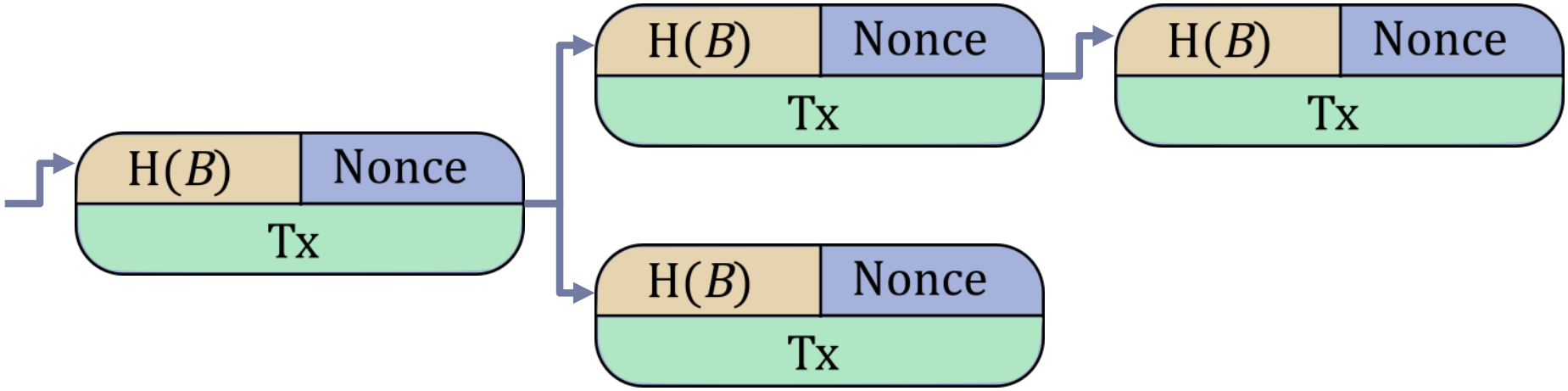


Why do we need miners?

Conflicting Blocks

Fork: multiple miner create blocks with the same preceding block.

- Longest chain wins with random tie-breakings.
- Accidental bifurcation happens once every 60 blocks.



Consensus

Majority of hashing power has voted for transactions on longest chain.

- It is costly to increase voting power
- Players are not motivated to cheat

The 51% attack!

If any party controls majority of hashing power, they can:

- Undo the past
- Deny mining rewards
- Undermine the currency

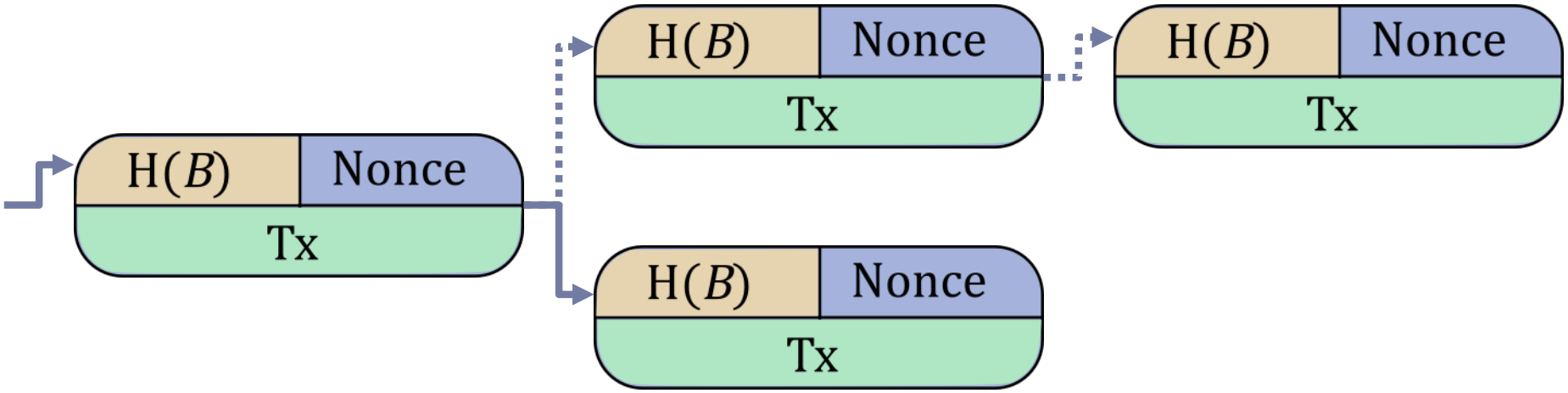
We have put a plan in place to see that 51% of all hashing power, will not be maintained by Ghash.IO by executing the following actions:

- We will temporarily stop accepting new independent mining facilities to the Ghash.IO pool.
- We will implement a feature, allowing CEX.IO users to mine bitcoins from other pools. So when they purchase GH/s they can put it towards any pool they choose.

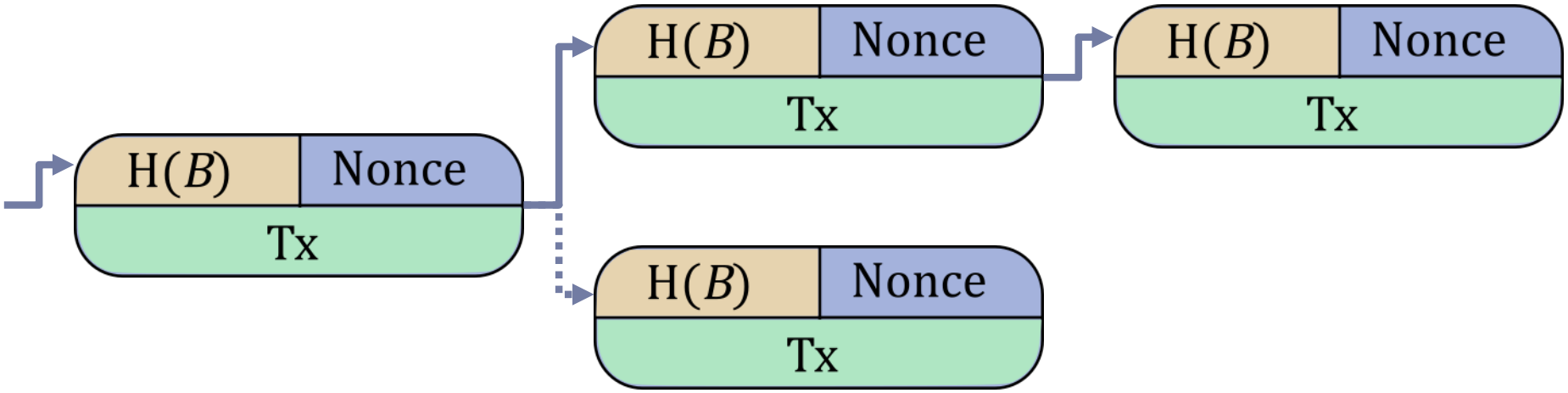
We will not be implementing a pool fee, as we believe the pool has to remain free.

GHash.IO does not have any intentions to execute a 51% attack, as it will do serious damage to the Bitcoin community, of which we are part of. On the contrary, our plans are to expand the bitcoin community as well as utilise the hashing power to develop a greater bitcoin economic structure. If something happened to Bitcoin as a whole it could risk our investments in physical hardware, damage those who love Bitcoin and we see no benefit from having 51% stake in mining.

I'll keep these blocks for myself!



I'll keep these blocks for myself!



if we gain a lead:

withhold blocks
mine on private chain

else if lead shrinks, but is still at least 2:

reveal blocks to keep abreast with public chain

else if lead drops below 2:

reveal all blocks
mine on public chain

Worries

"Rational miners will prefer to join the selfish miners, and the colluding group will increase in size until it becomes a majority. At this point, the Bitcoin system ceases to be a decentralized currency."

Majority is not Enough: Bitcoin Mining is Vulnerable

Ittay Eyal, and Emin Gün Sirer

Detecting selfishness

- Orphaned blocks
  - i.e. valid **blocks** which are not part of the main **chain**:
    - occur naturally when two miners produce **blocks** at similar times
    - can be caused by an attacker (with enough hashing power) attempting to reverse transactions.
- Timing of successive blocks
  - Two blocks in close succession should be a rare occurence with the honest protocol, and more common when someone is quickly releasing selfishly mined blocks in order to squash the honest miners.

More at: *"How to detect selfish miners"* by Ittay Eyal, and Emin Gün Sirer, http://hackingdistributed.com/2014/01/15/detecting-selfish-mining/

Bitcoin_NG

Bitcoin-NG: A Secure, Faster, Better Blockchain

Authors
Ittay Eyal, Adem Efe Gencer, Emin Gün Sirer, and Robbert van Renesse

Bitcoin-NG [Eyal et all, NSDI '16]

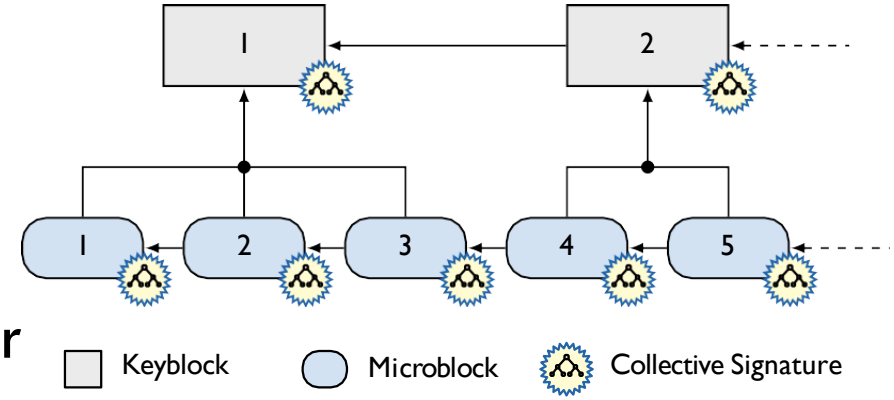
- Makes the observation that block mining implement two distinct functionalities
  - Transaction verification
  - Leader election
- But, Bitcoin-NG inherits many of Bitcoin's problems
  - Double-spending
  - Leader is checked after his epoch ends

Their proposal: using the mined blocks to elect a temporary leader, who can then mint many microblocks within the 10 minute interval; *leader verifies/signs incoming transactions almost instantaneously.*

Decoupling Transaction Verification from Leader Election

Two type of blocks:

- Key blocks:
  - PoW & share value
  - Leader election
  - Contains public key used for future microblocks
  - Leader wins 40% of the transactions's revenue
- Microblocks:
  - Validating client transactions
  - Issued by the leader



Proof of Stake

- Proof-of-Work: the algorithm rewards miners who solve mathematical problems with the goal of validating transactions and creating new blocks
- Proof-of-Stake, the creator of a new block is chosen in a deterministic way, depending on its wealth, also defined as stake; No block reward

Block persistence

Enhancing Bitcoin Security and Performance with Strong Consistency via Collective Signing

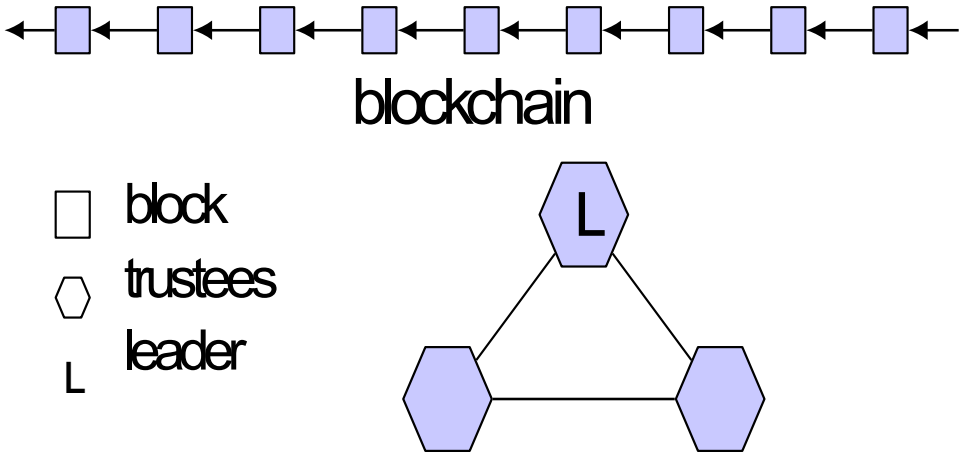
Authors:
Eleftherios Kokoris Kogias, Philipp Jovanovic, Nicolas Gailly, Ismail Khoffi, Linus Gasser, and Bryan Ford, *École Polytechnique Fédérale de Lausanne (EPFL)*

Block persistence

- A transaction is confirmed when it is buried "deep enough" 6 blocks, arbitrary number
- In Bitcoin there is no verifiable commitment of the system that a block will persist
  - Clients rely on probabilities to gain confidence.
  - Probability of successful fork-attack decreases exponentially

Strawman Design: PBFTCoin

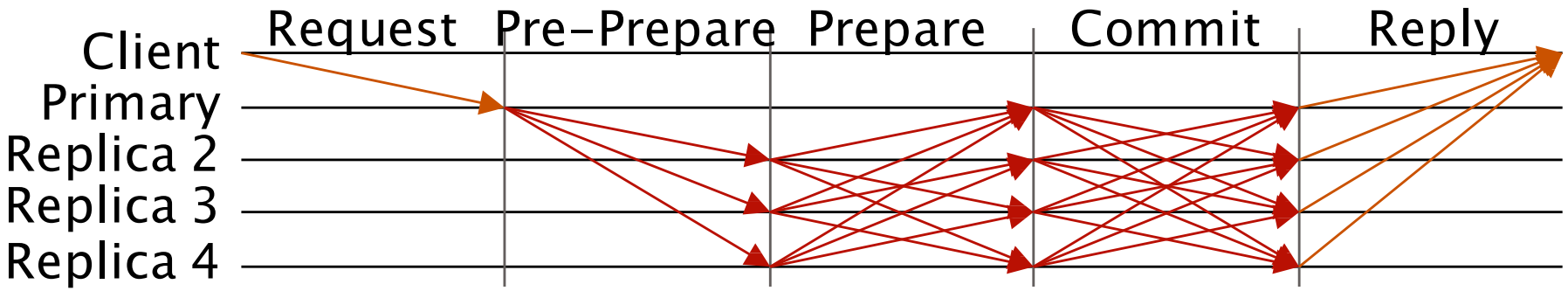
- **3f+1** fixed "trustees" running PBFT* to withstand **f** failures
- Non-probabilistic strong consistency
  - Low latency
  - No forks/inconsistencies
    - No double-spending



*Practical Byzantine Fault Tolerance [Castro/Liskov]

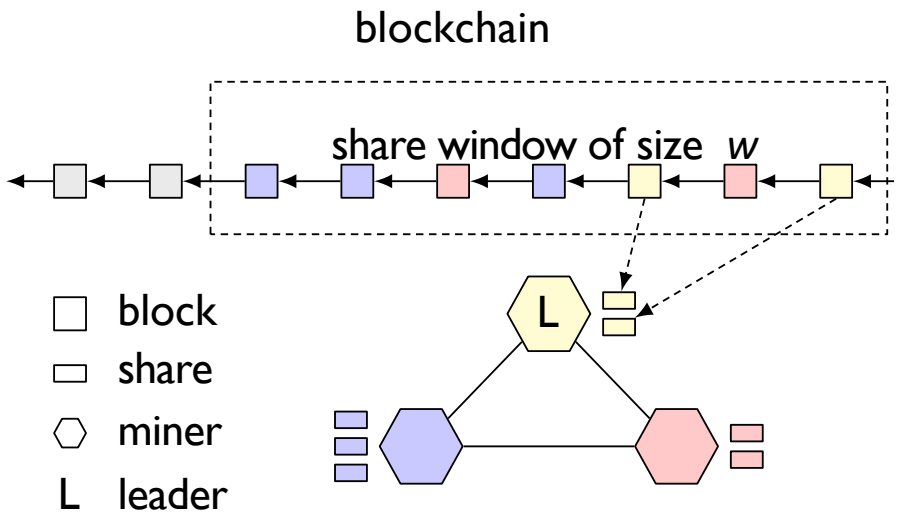
Strawman Design: PBFTCoin

- Problem: Needs a static consensus group
- Problem: Scalability
  - $O(n^2)$ communication complexity
  - $O(n)$ verification complexity
  - Absence of third-party verifiable proofs (due to MACs)



Opening the Consensus Group

- PoW against Sybil attacks
- One share per block
  - % of shares $\propto$ hash-power
- Window mechanism
  - Protect from inactive miners



From MACs to Signing

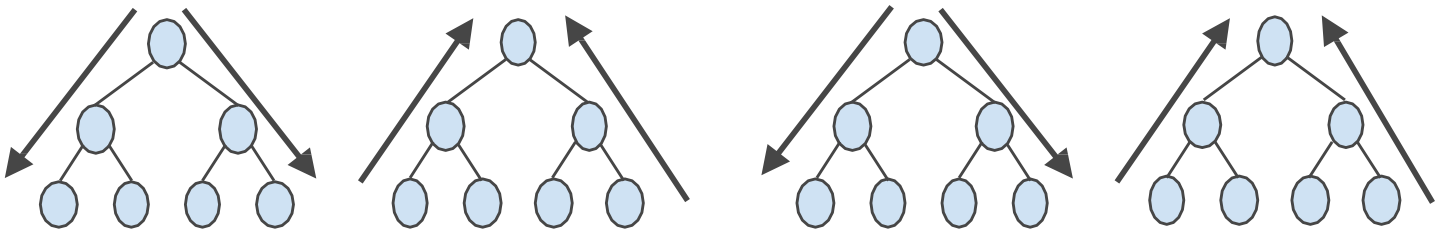
- Substitute MACs with public-key cryptography
  - ECDSA provides more efficiency
  - Third-party verifiable
  - PoW Blockchain as PKI
  - Enables sparser communication patterns (ring or star topologies)

From MACs to Collective Signing

- Can we do better than O(n) communication complexity?
  - Multicast protocols transmit information in O(log n)
  - Use trees!!
- Can we do better than O(n) complexity to verify?
  - Schnorr multisignatures could be verified in O(1)
  - Use aggregation!!
- Schnorr multisignatures + communication trees = Collective Signing [Syta et all, IEEE S&P'16]

CoSi

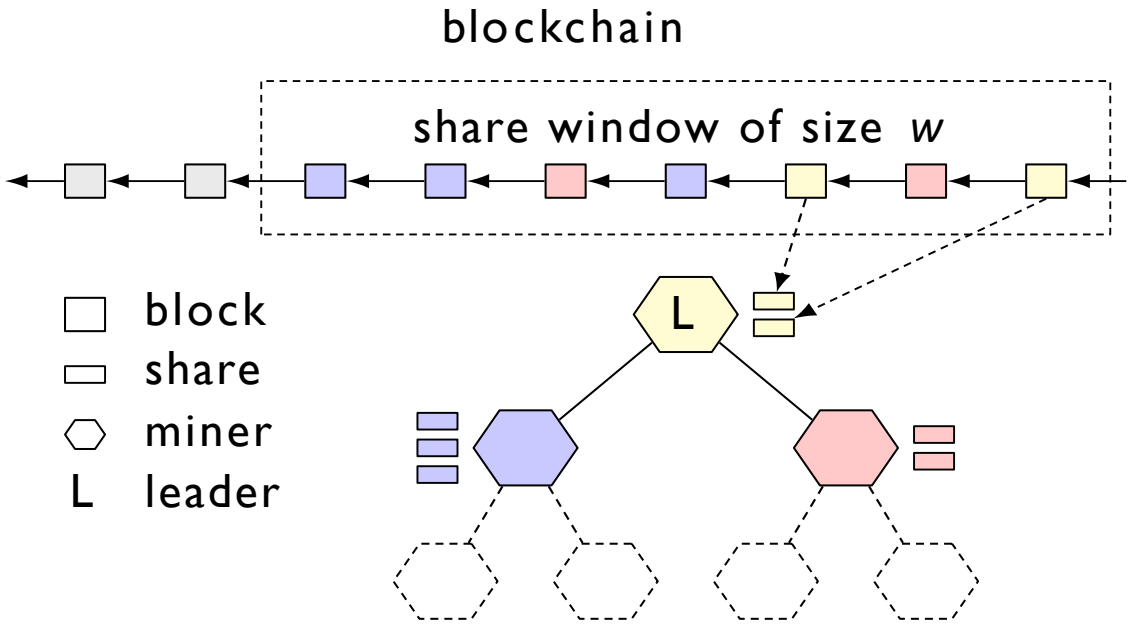
- Efficient collective signature, verifiable as a simple signature
  - 80 bytes instead of 9KB for 144* co-signers (Ed25519)



* Number of ~10-minute blocks in 1-day time window

Discussion

- CoSi is not a BFT protocol
- PBFT can be implemented over two subsequent CoSi rounds
  - Prepare round
  - Commit round



Problem Statement

1. In ~~Bitcoin~~ ByzCoin there is ~~no~~ a verifiable commitment of the system that a block will persist
2. Throughput is limited by forks
   - Increasing block size increases fork probability
   - Liveness exacerbation

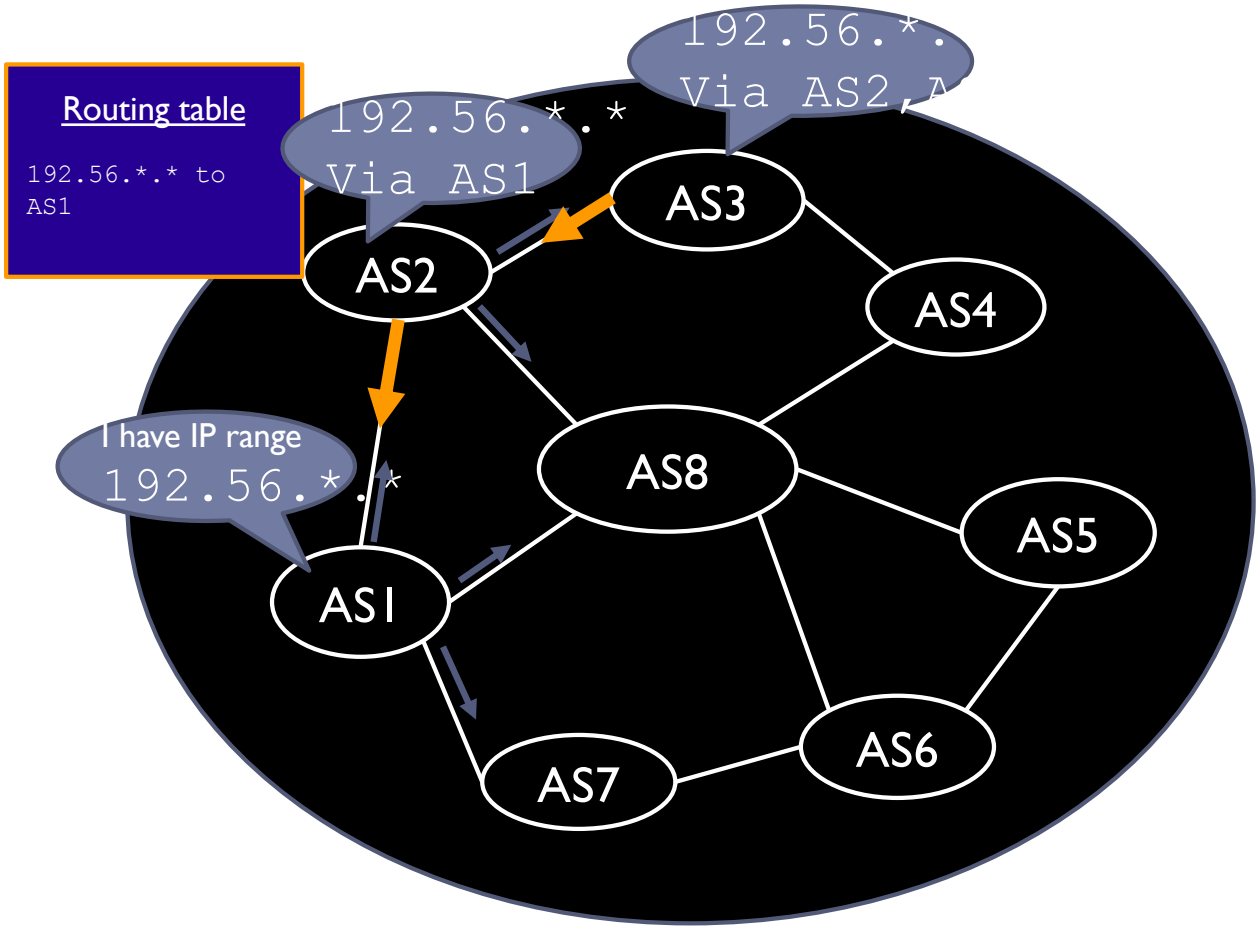
3: Eclipse attacks

Eclipse Attacks on Bitcoin's Peer-to-Peer Network
Ethan Heilman Alison Kendler Aviv Zohar Sharon Goldberg

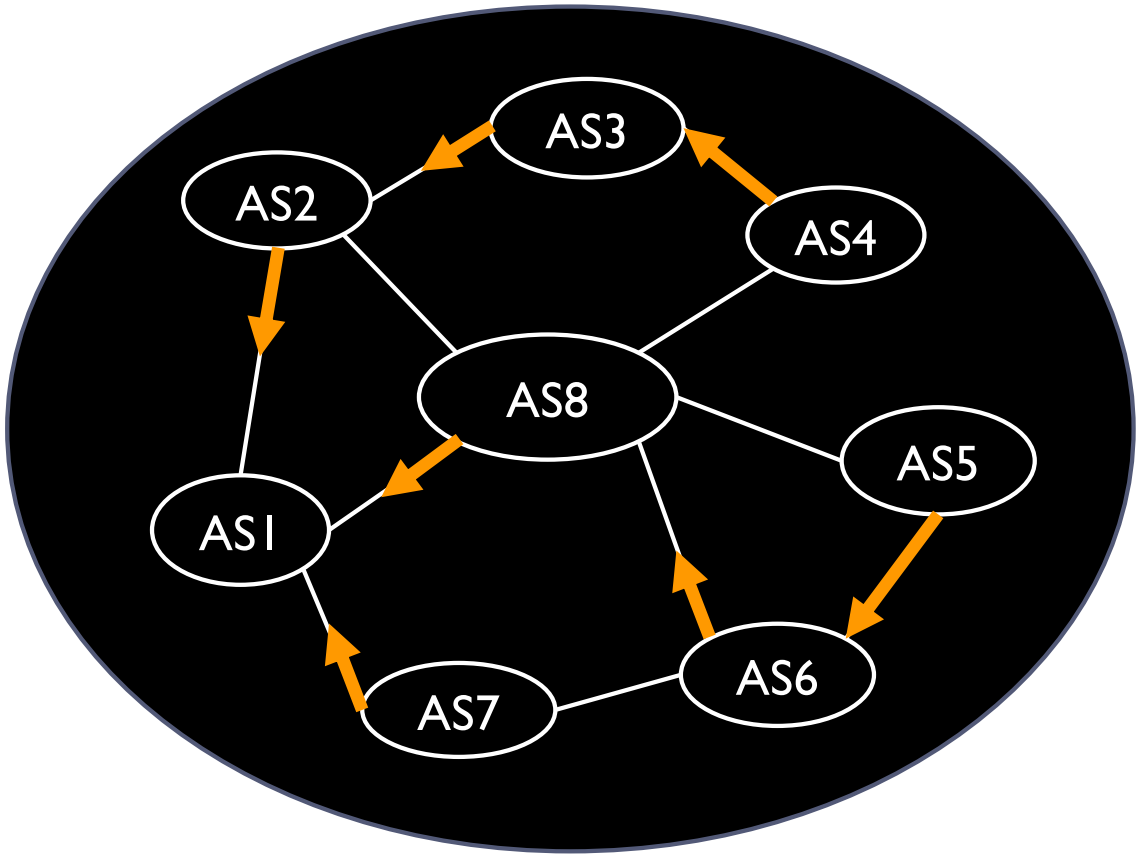
Eclipse attacks

- Eclipse attack: an attacker isolates the victim from the rest of the peers, i.e. controls all of the victim's incoming and outgoing connections
- Attackers
  - On-path attackers
  - Off-path attacker
- Implications for bitcoin:
  - Attacker can then filter the victim's view of the blockchain
  - Force the victim to waste compute power on obsolete views of the blockchain
  - Use the victim's compute power for its own purposes

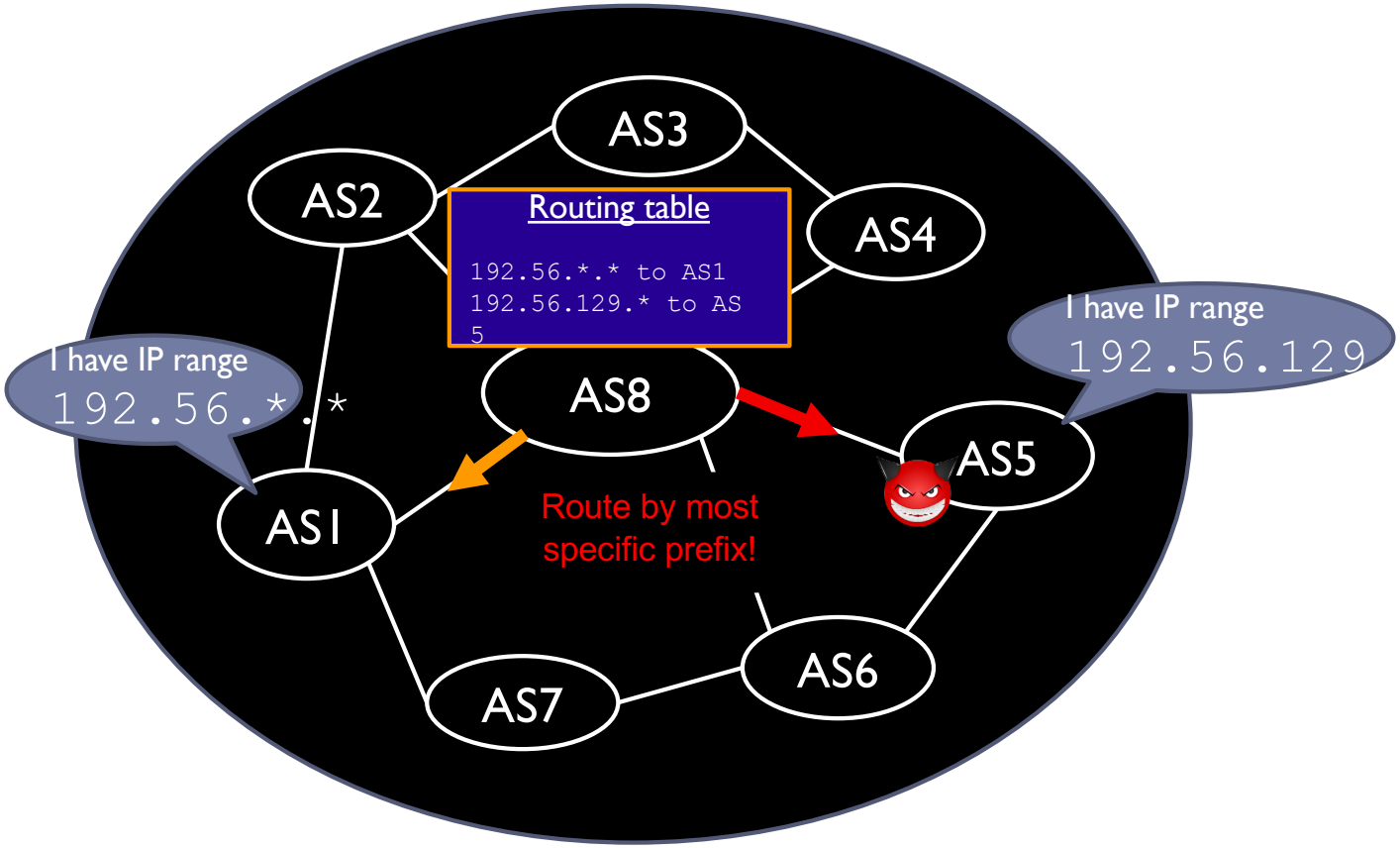
BGP and Routing



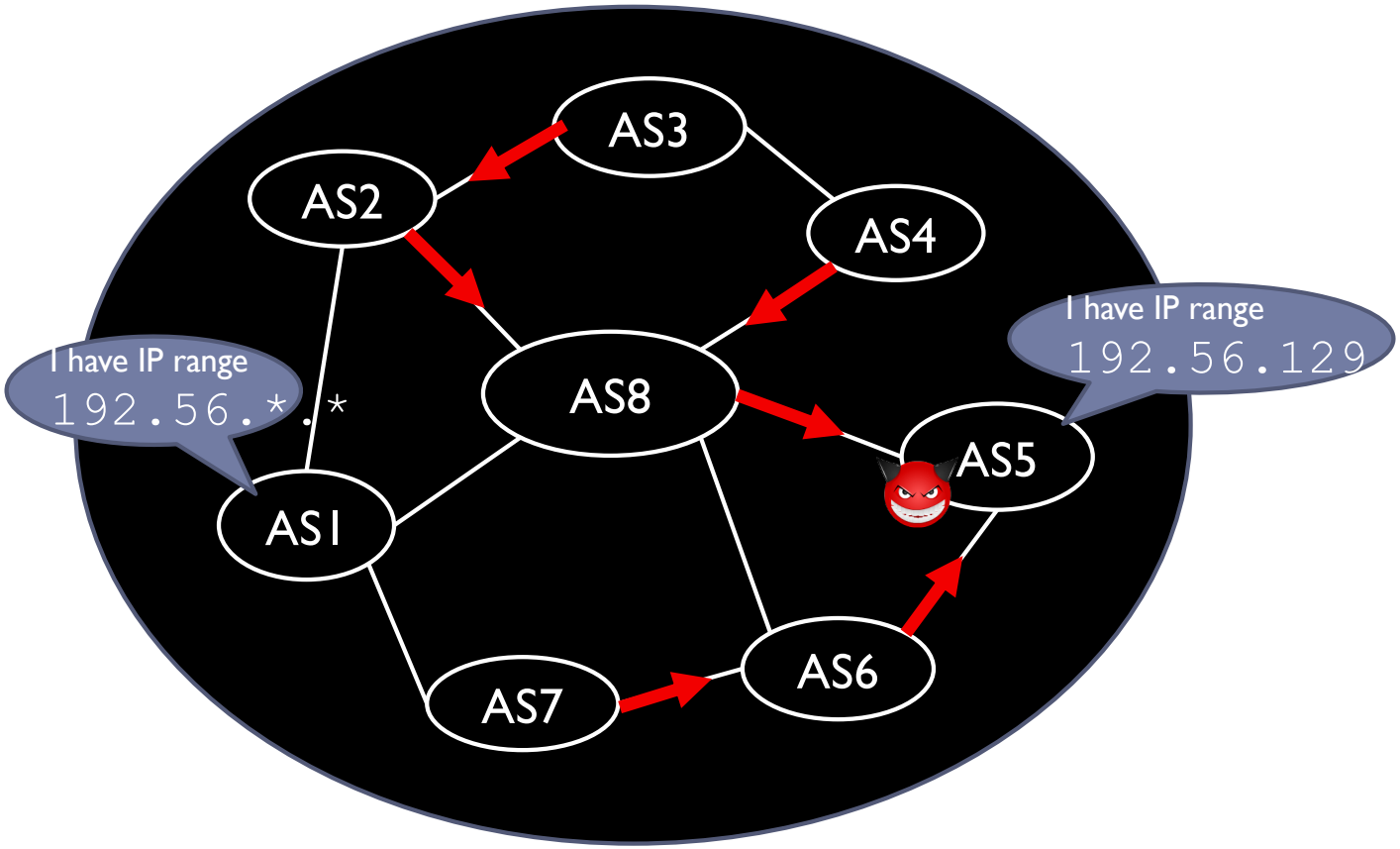
BGP and Routing



Prefix Hijacking

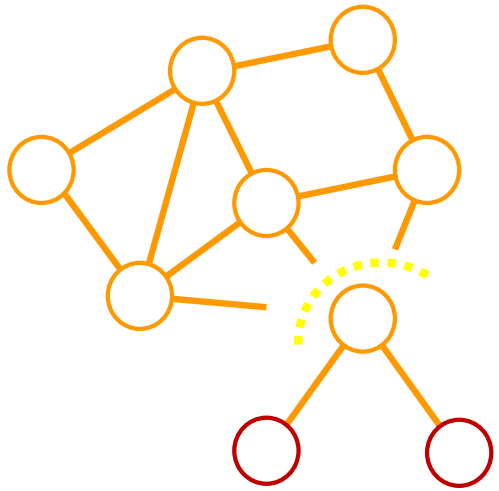


Prefix Hijacking

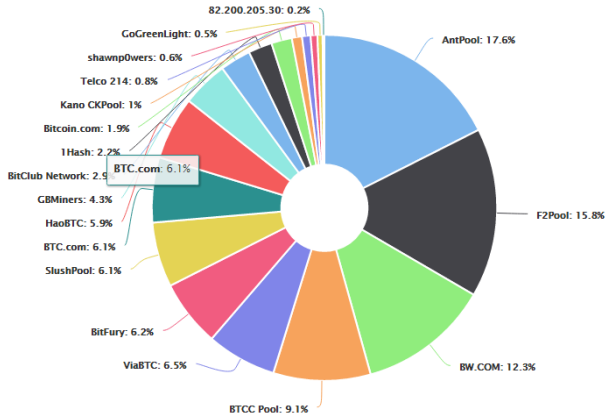


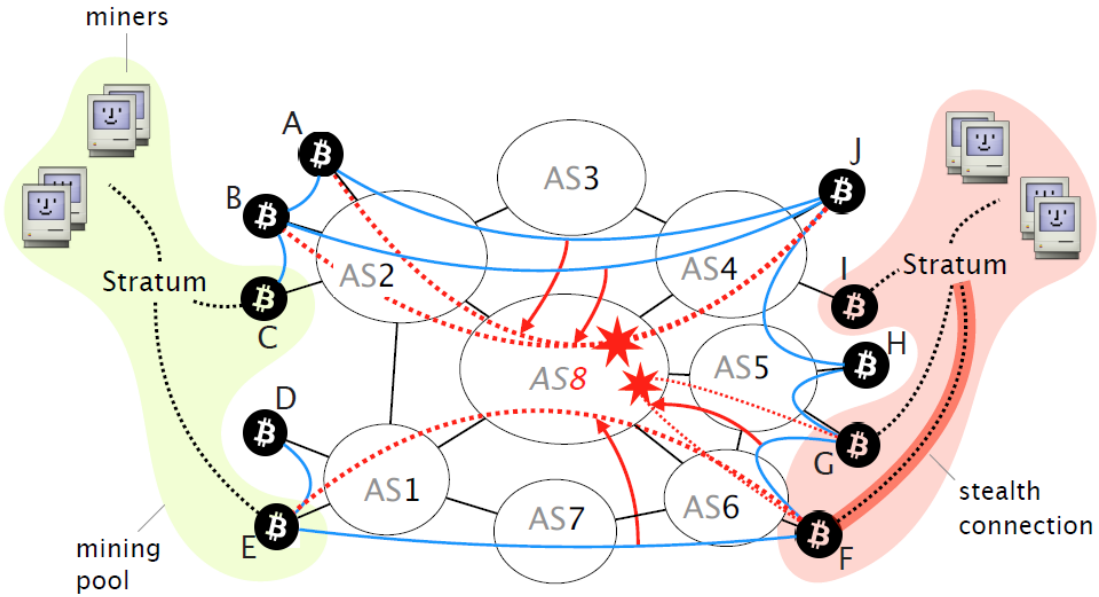
Consequences of disrupting connectivity

- Transactions cannot be sent (DoS)
- Pool rewards can be stolen
- Transactions on one side of the network are reversed
  - Miners lose revenue
  - Double spending attacks against merchants
- Mining power subverted to attack
  - double spend
  - selfish mining
  - Censorship via empty blocks



Mining pools





Attack 1: Partitioning Bitcoin

- Deduce gateway nodes for pools
  - Stratum servers
  - Block propagation data
- Combine with routing data

Factors that aid attacker:

- Mining power is held by few nodes
- Only 7% of nodes are advertised in /24 prefixes

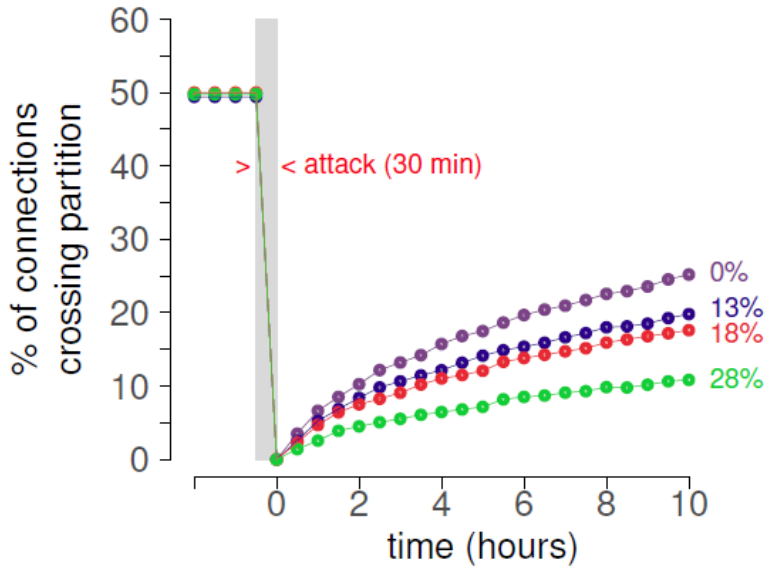
Isolated mining power	*Minimum # prefixes*	*Median # prefixes*	*# Feasible Partitions*
6%	2	86	20
7%	7	72	23
8%	32	69	14
30%	83	83	1
39%	32	51	11
40%	37	80	8
41%	44	55	3
45%	34	41	5
46%	78	78	1
47%	39	39	1

TABLE IV: This table lists all partitions that can be created based on our inferred topology. The leftmost column indicates the portion of mining power contained within the isolated set and the rightmost the number of different combinations of pools that could form it.

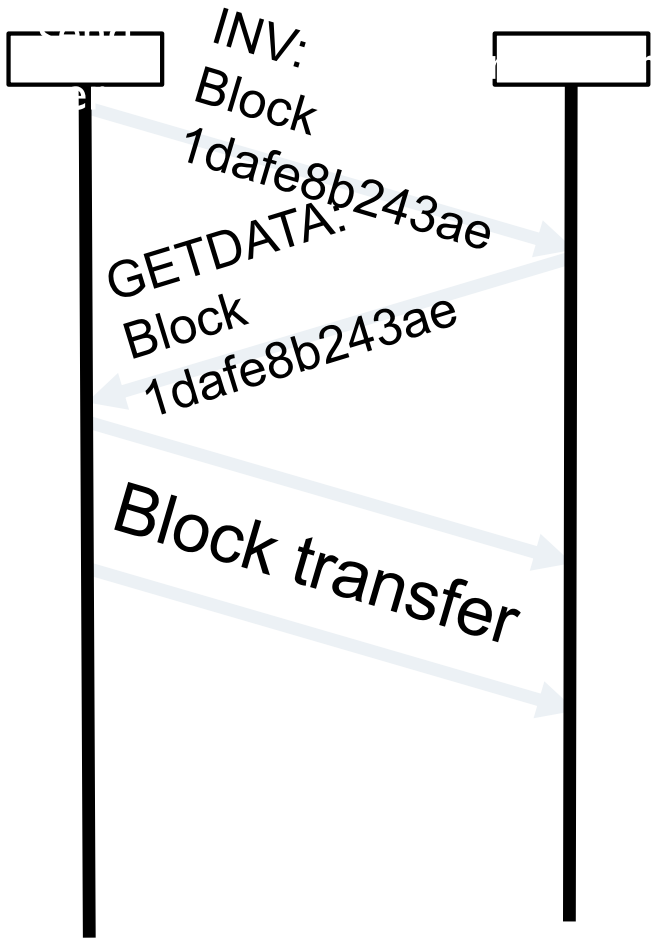
Partitions need to be perfect

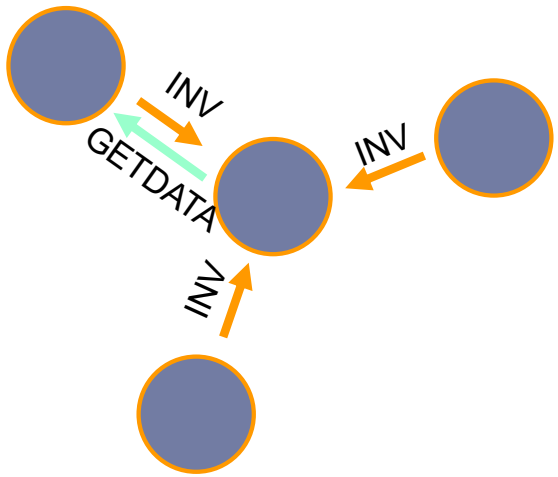
- 1050 bitcoind nodes running on VMs on emulated network.
  - With churn (as measured on network)

- Connections return slowly
- BUT a few connections suffice.



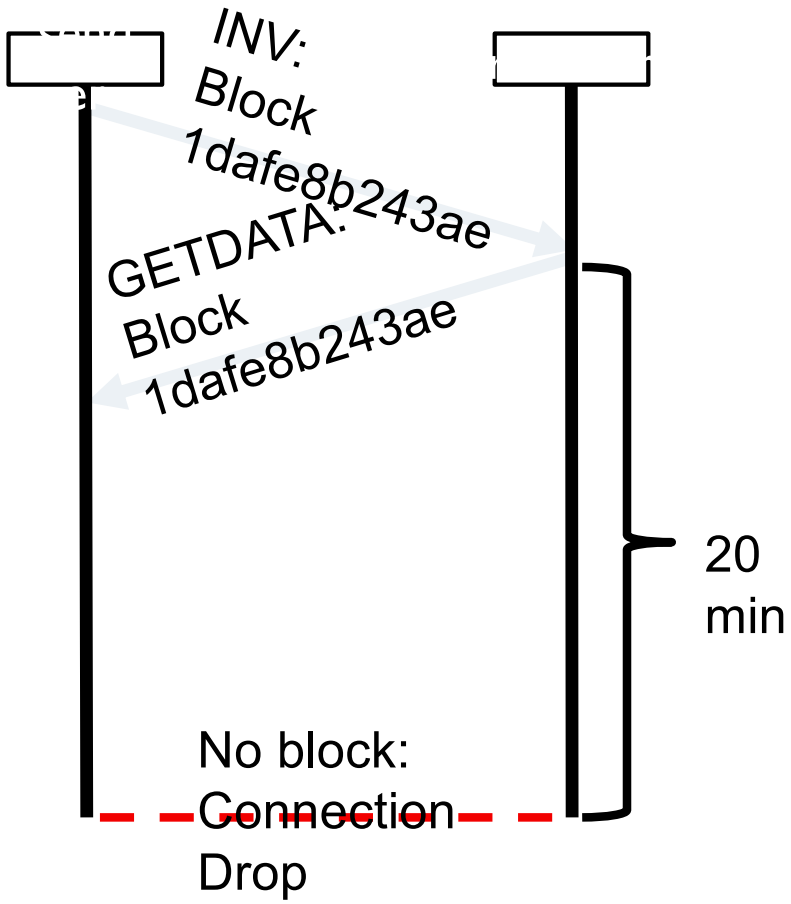
Blocks Propagation Mechanics



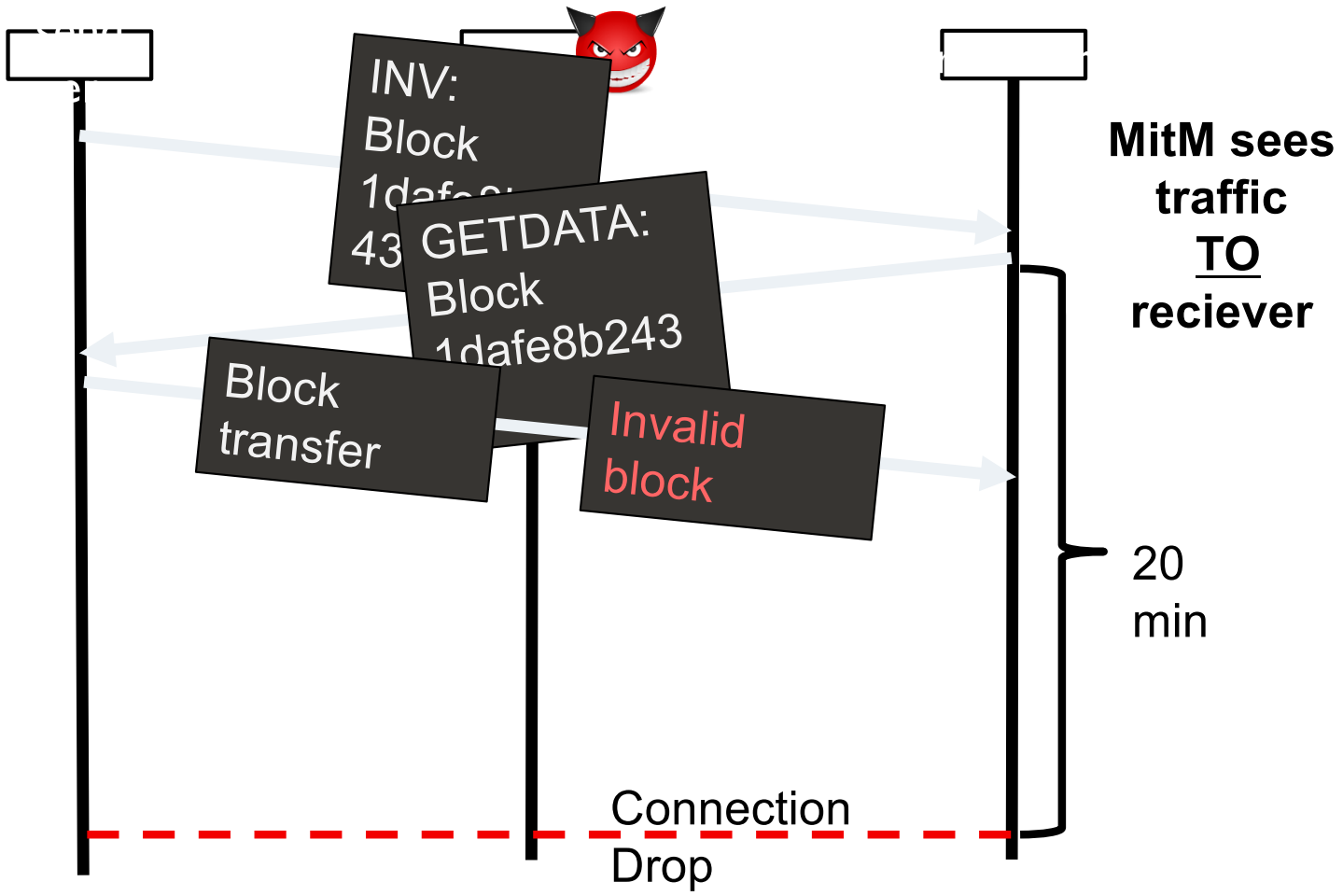


Traffic is not encrypted!

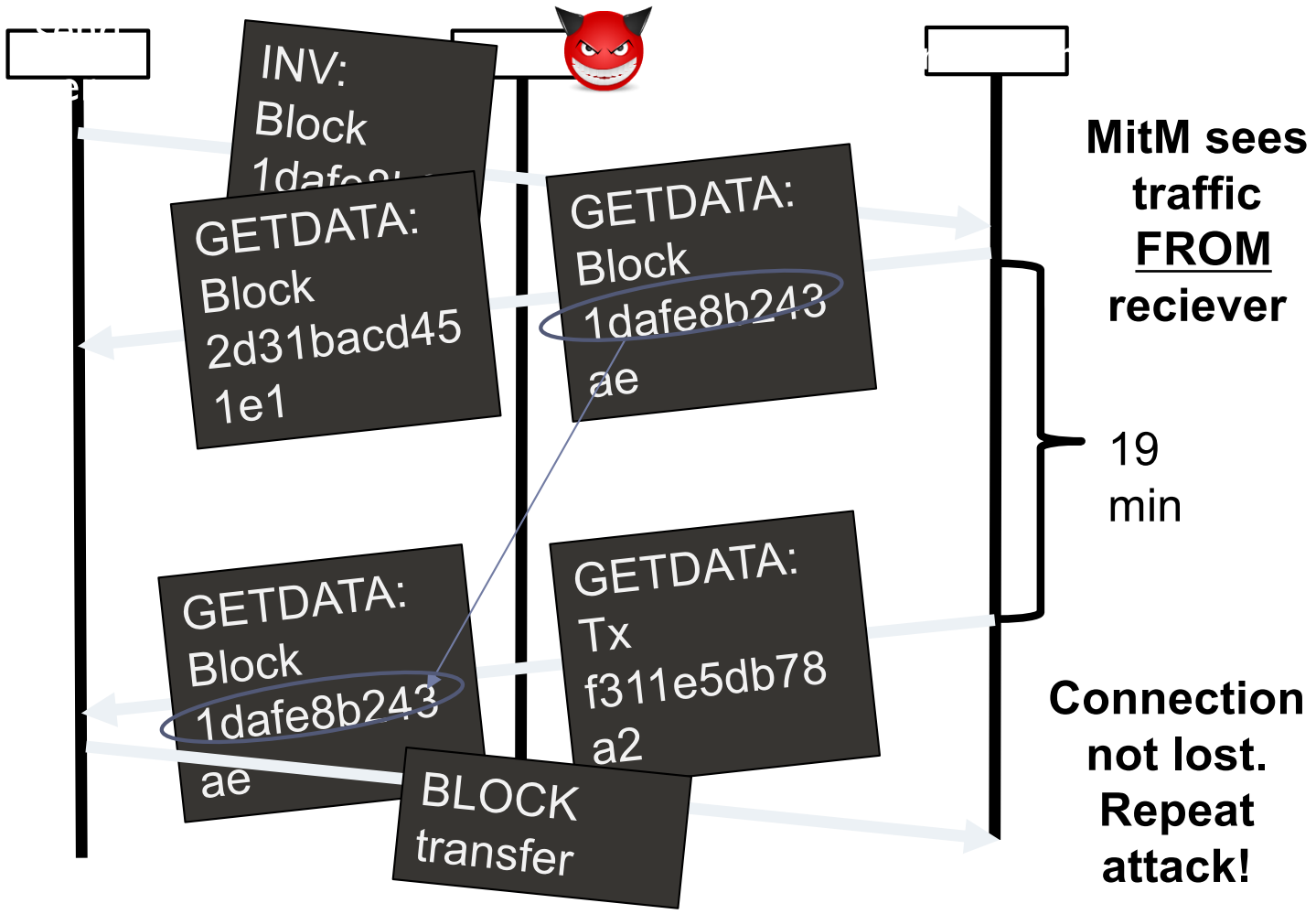
Blocks Propagation Mechanics



Attack 2a: MitM block delay attack



Attack 2b: MitM block delay attack

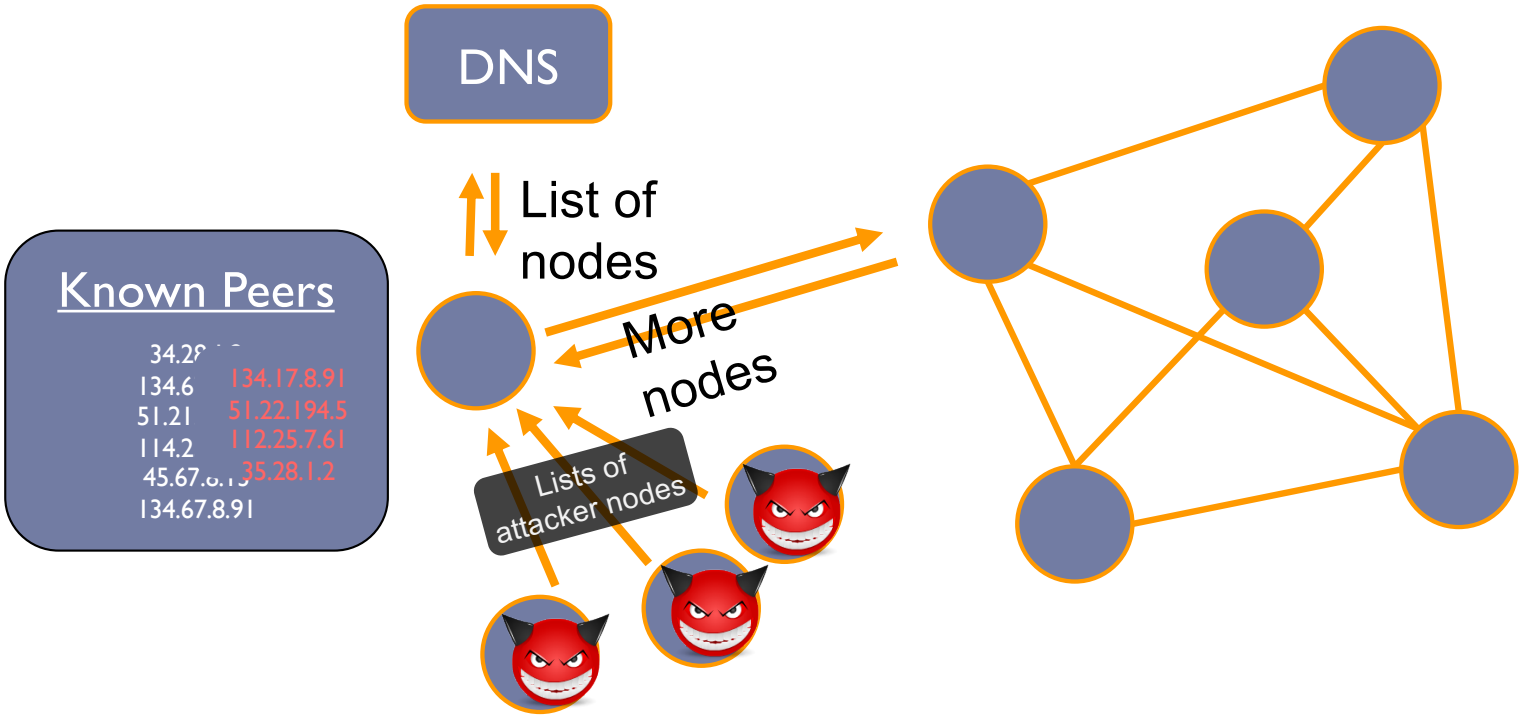


- We performed this MitM attack on our own node
- Passive AS (no hijacking)

% intercepted connections	50%	80%	100%
% time victim node is uniformed	63.21%	81.38%	85.45%
% total vulnerable Bitcoin nodes	67.9%	38.9%	21.7%

- Uninformed node wastes mining power
- Susceptible to 0-conf attacks

Eclipse attacks



Summary

- Bitcoin is considered secure as long as nodes can communicate
- Communication is easily disrupted
- Mitigation techniques in the papers
  - Much more needed!