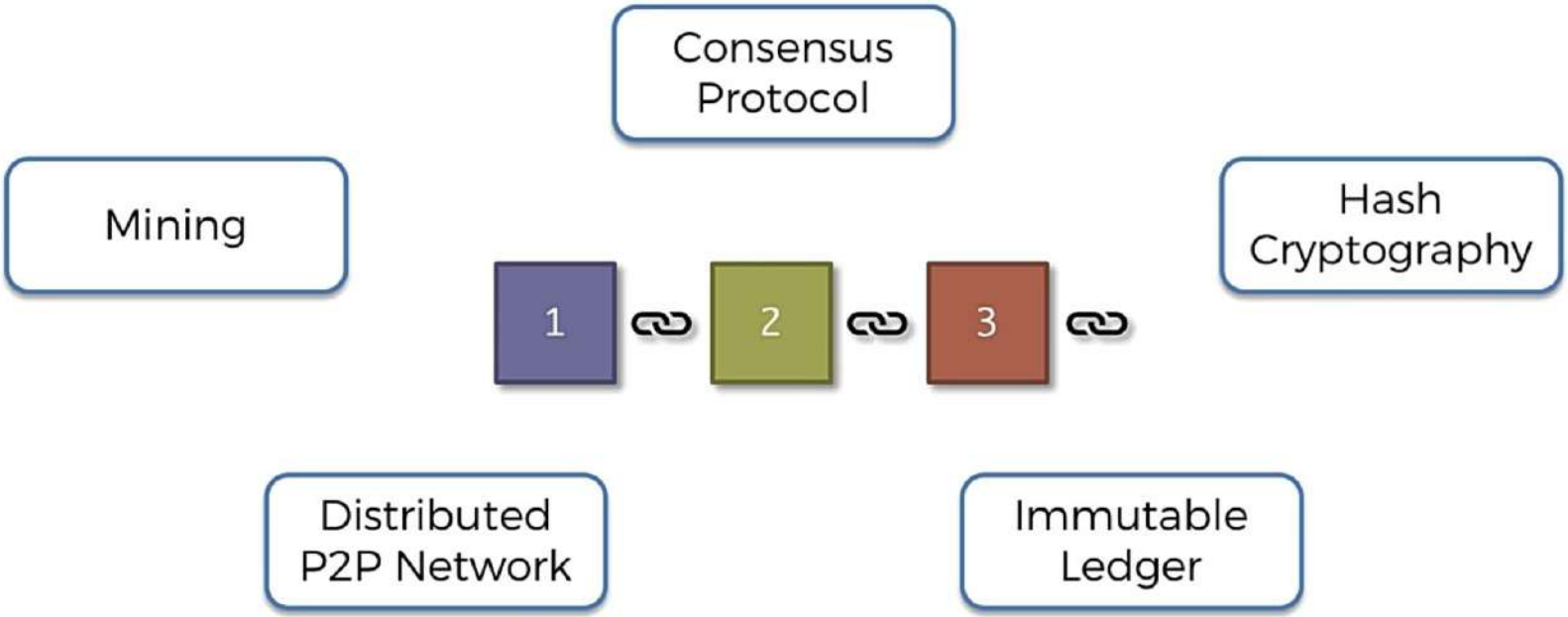
Module 2: Consensus & Mining

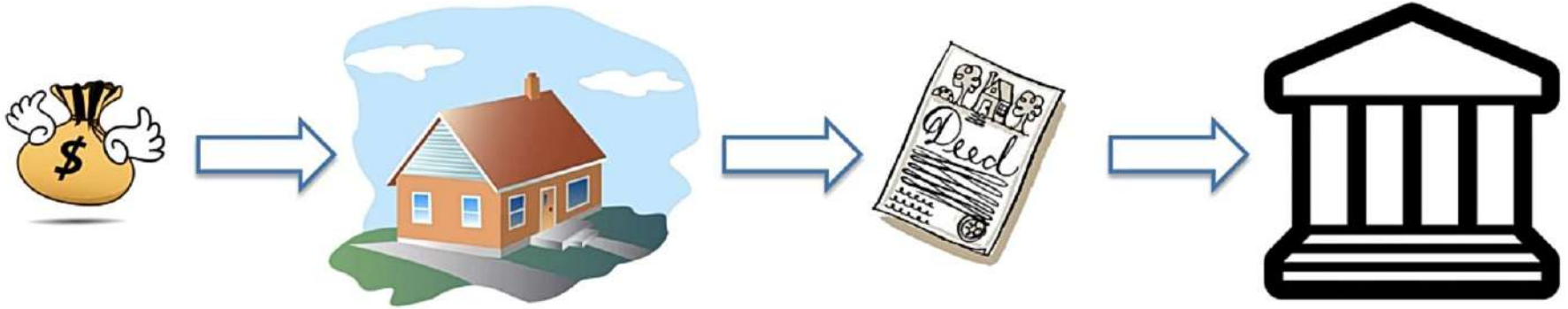
Agenda

- Decentralized Consensus
- Byzantine Generals Problem
- Independent Verification of Transactions
- Mining Nodes
- Aggregating Transactions into Blocks
- Constructing Block Header
- Mining the Block
- Successful Mining of Block
- Validating a new Block
- Assembling and Selecting Chains of Blocks
- Blockchain Forks

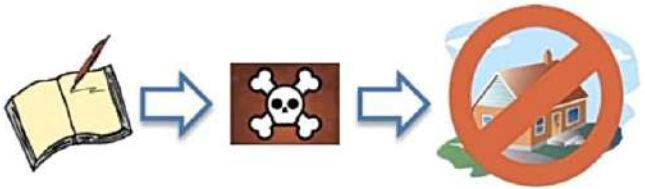
Blockchain


Consensus Protocol
Mining
Hash Cryptography
1
2
3
Distributed P2P Network
Immutable Ledger

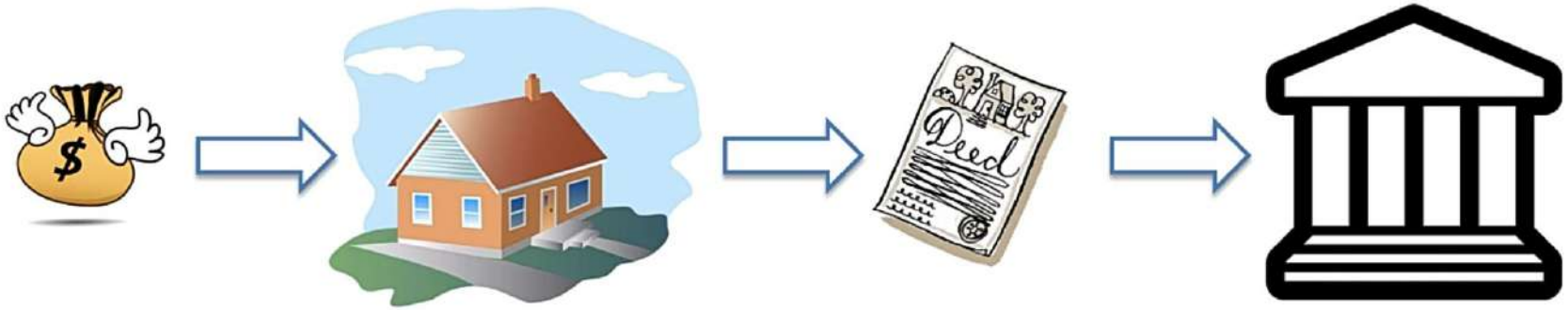
Immutable Ledger



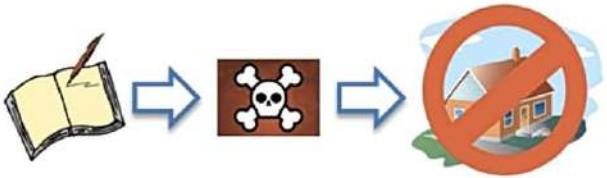
Traditional Ledger



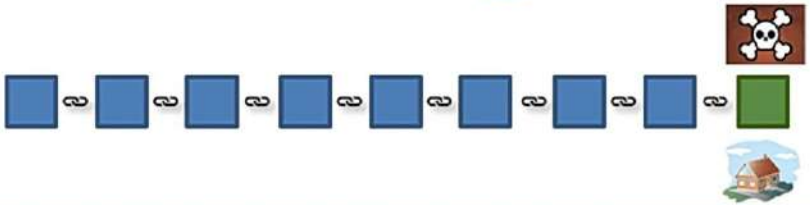
Immutable Ledger



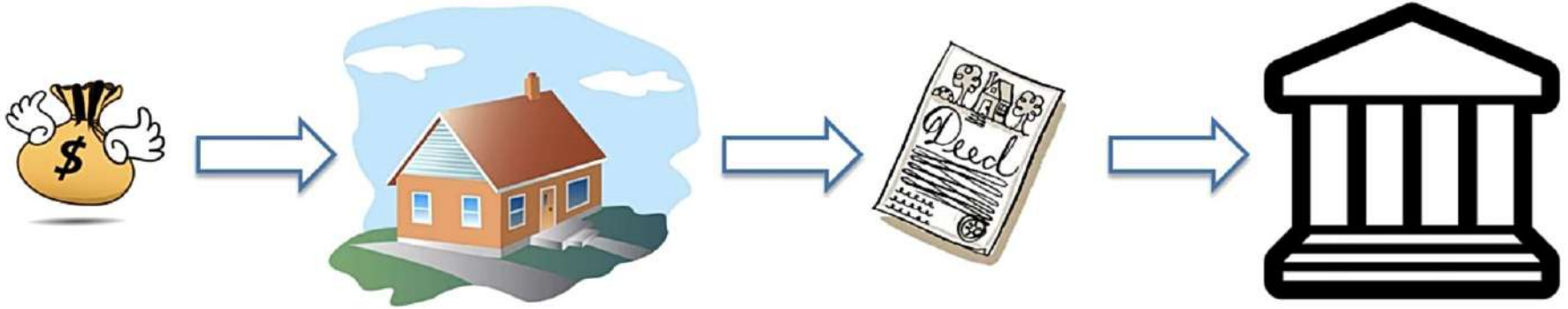
Traditional Ledger



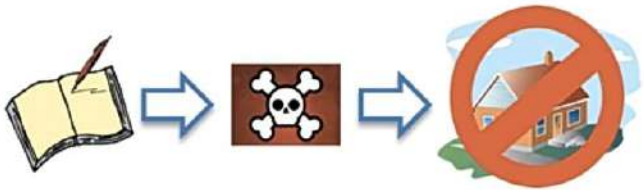
Blockchain



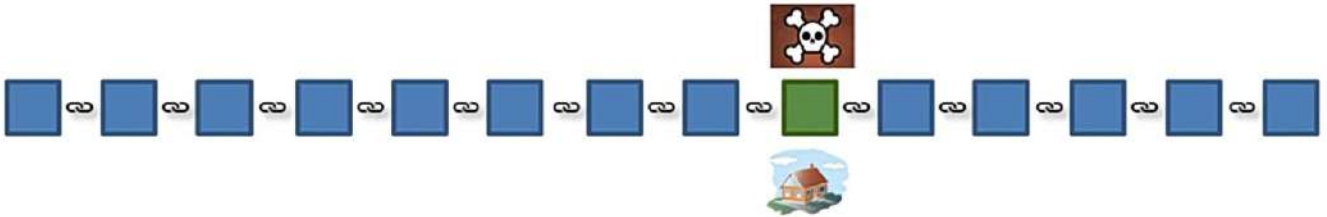
Immutable Ledger



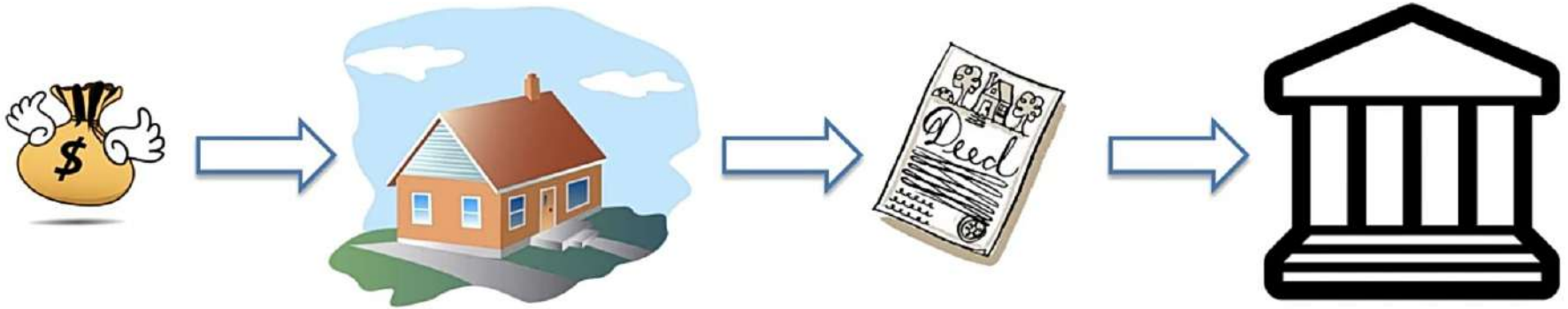
Traditional Ledger



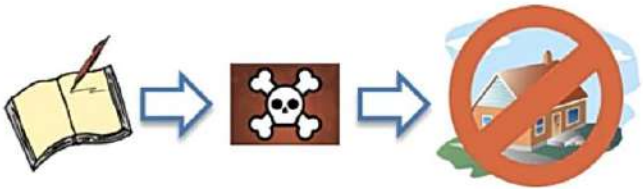
Blockchain



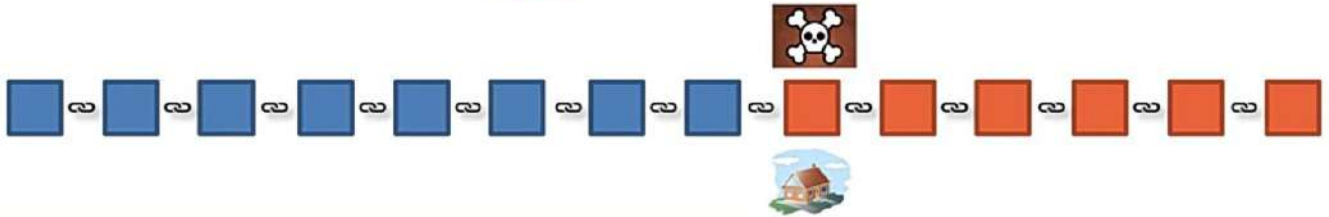
Immutable Ledger



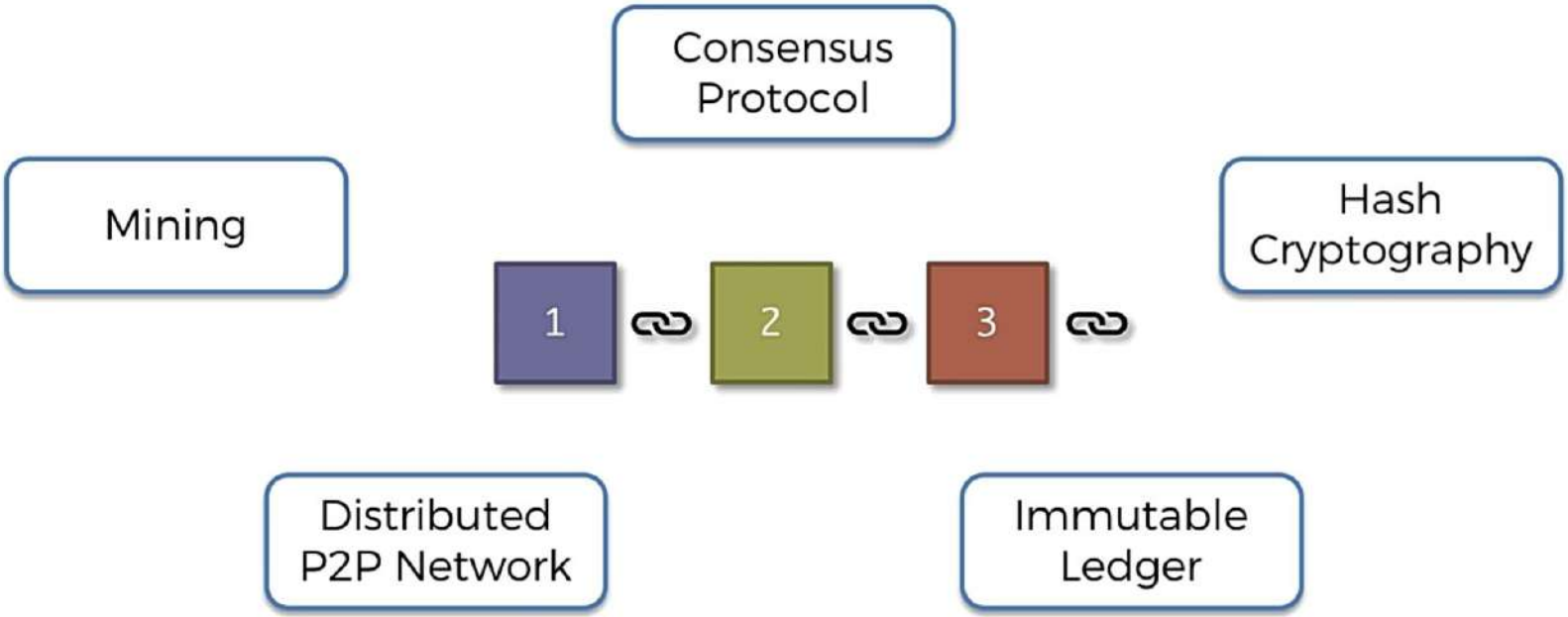
Traditional Ledger



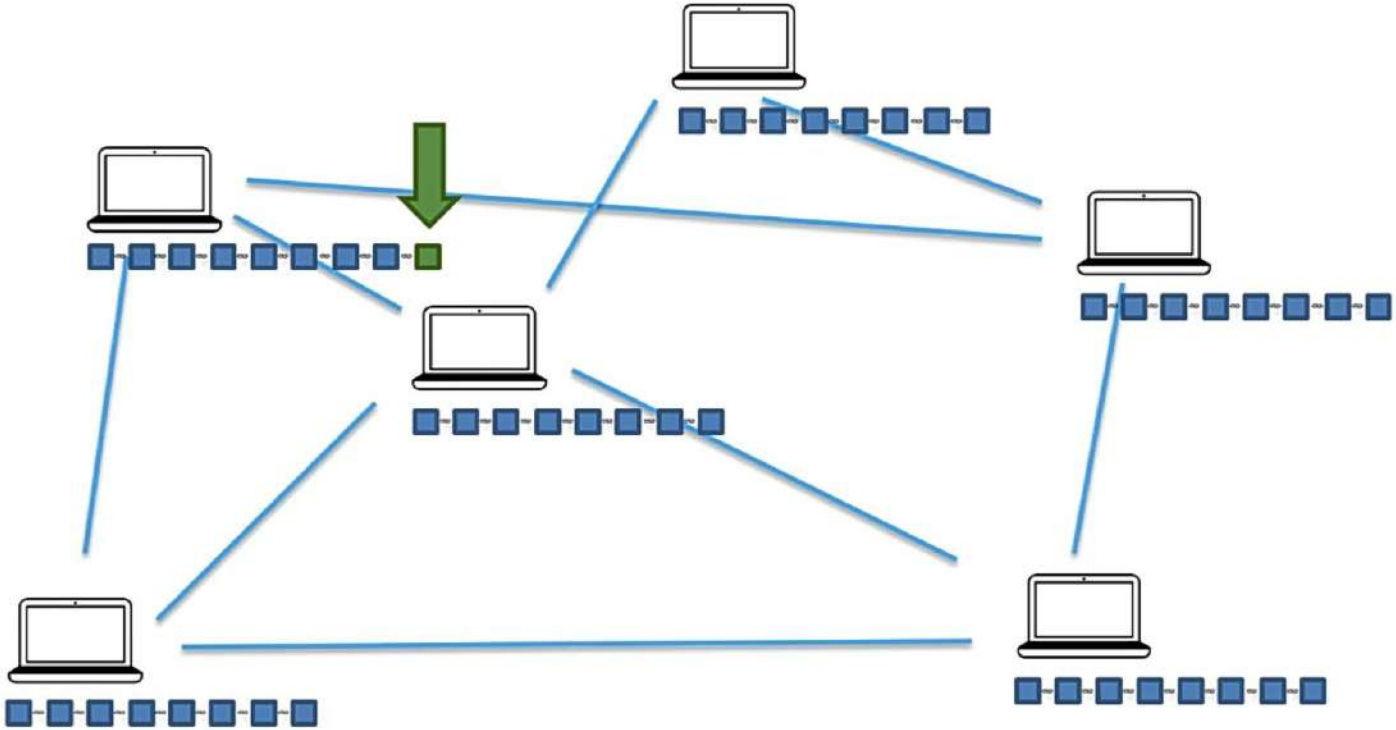
Blockchain



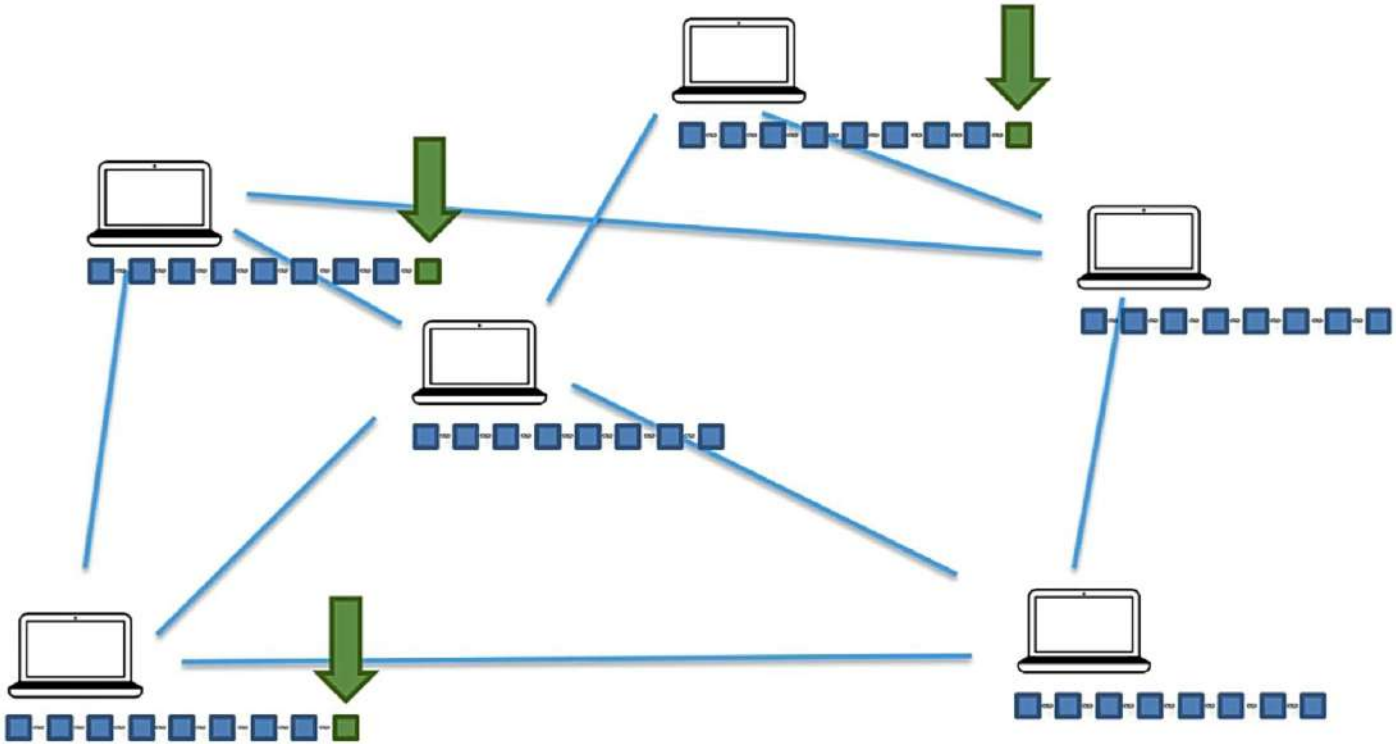
Blockchain


Consensus Protocol
Mining
Hash Cryptography
1
2
3
Distributed P2P Network
Immutable Ledger

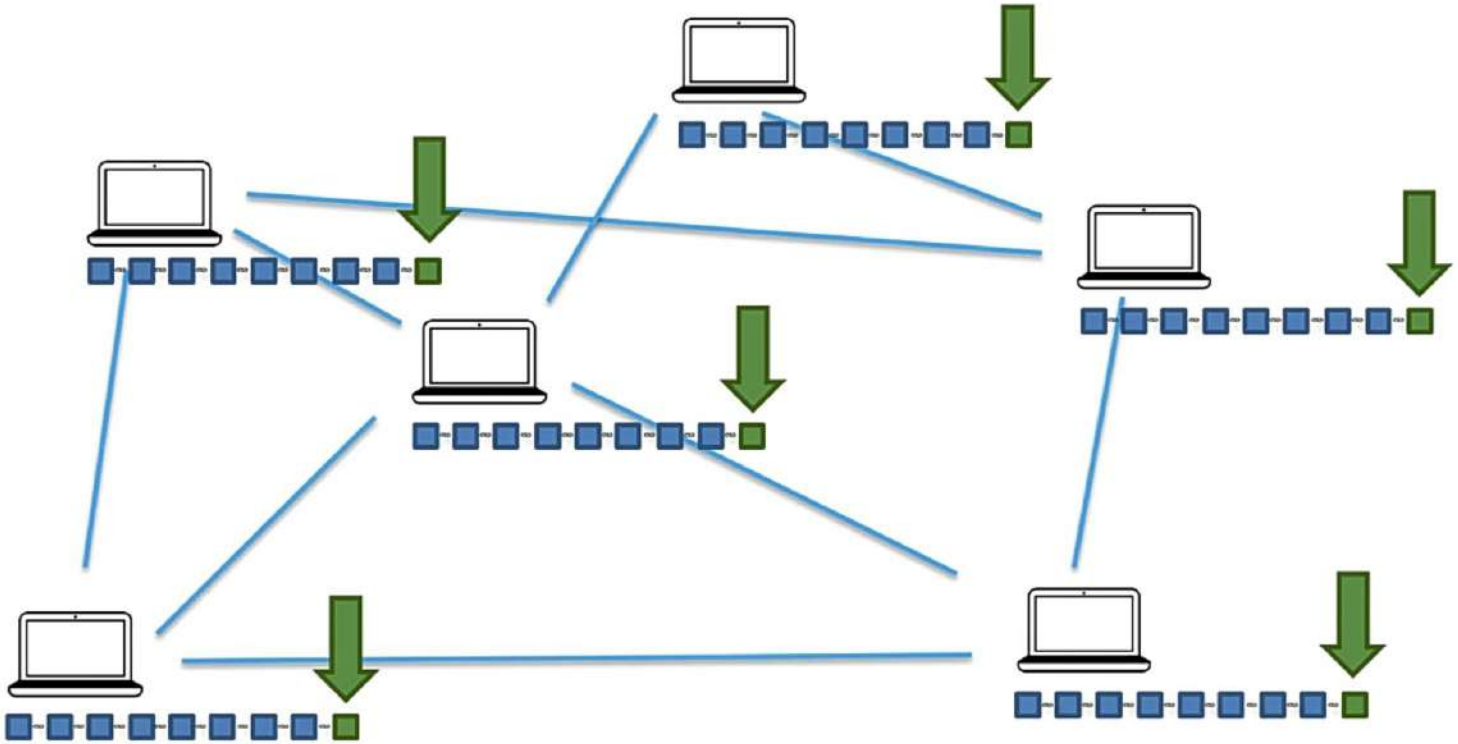
Distributed P2P Network



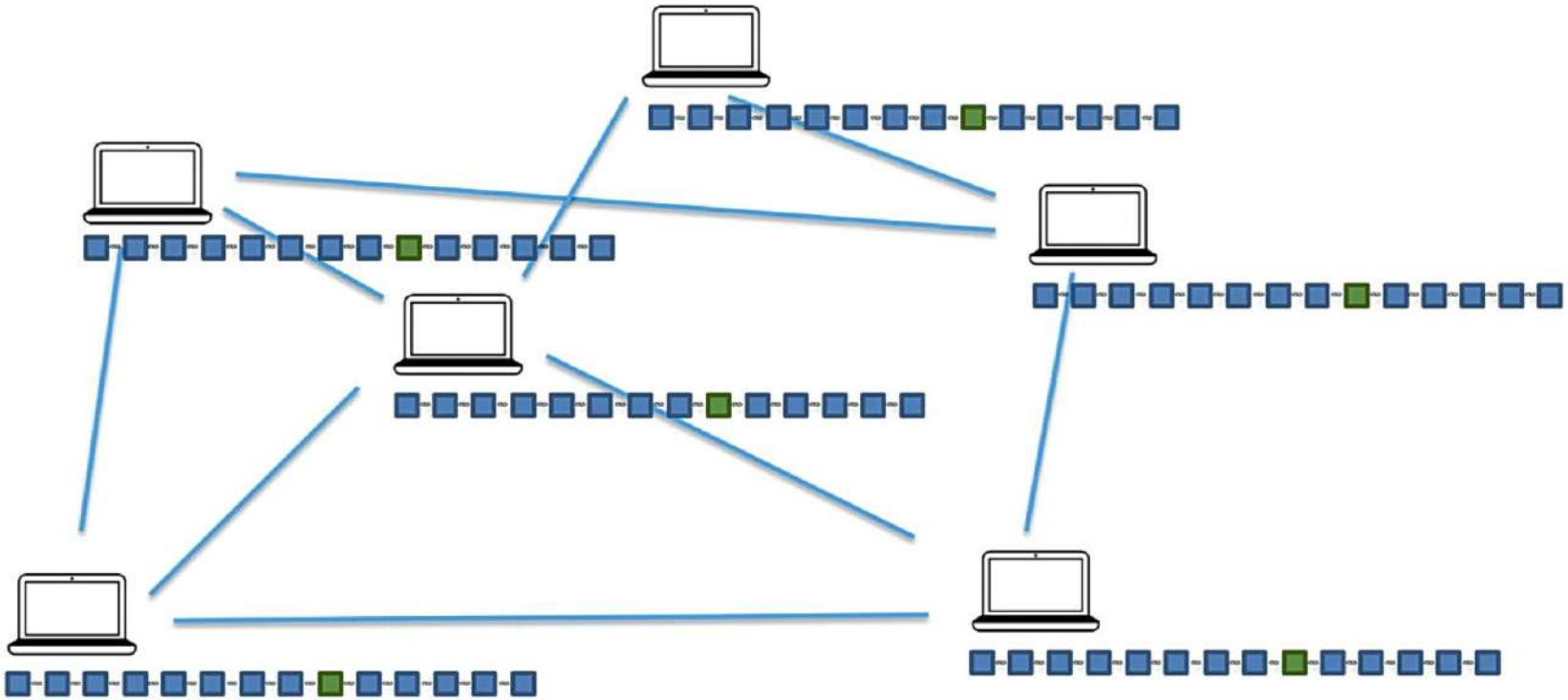
Distributed P2P Network



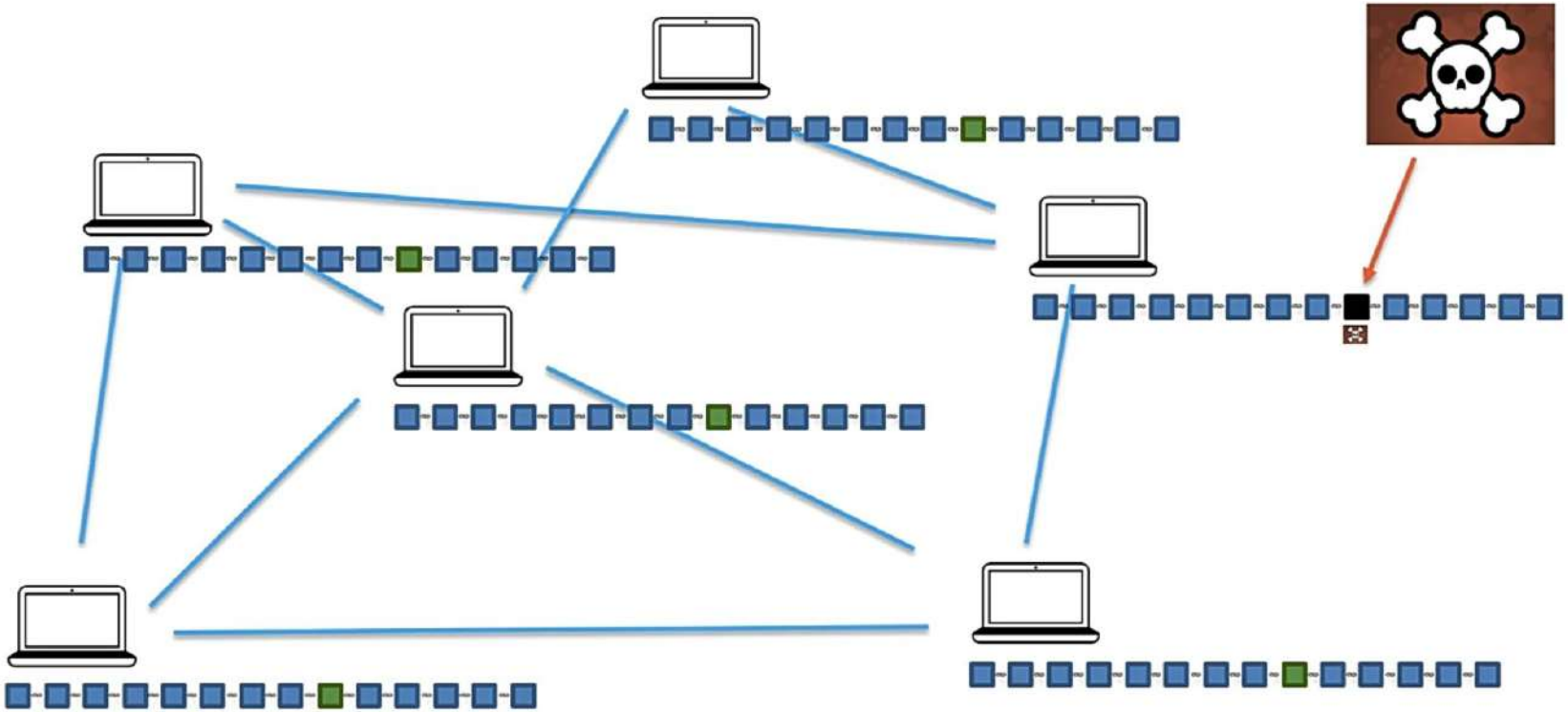
Distributed P2P Network



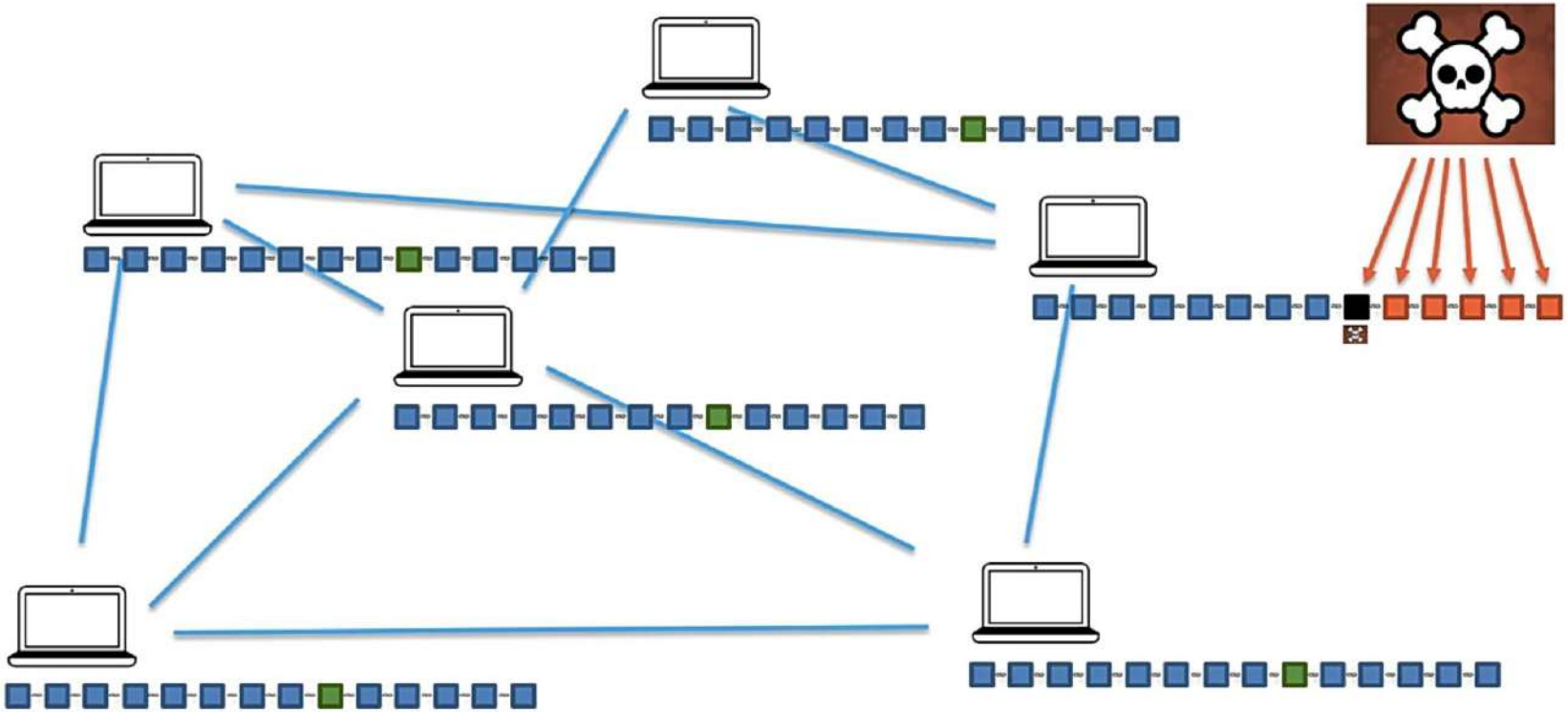
Distributed P2P Network



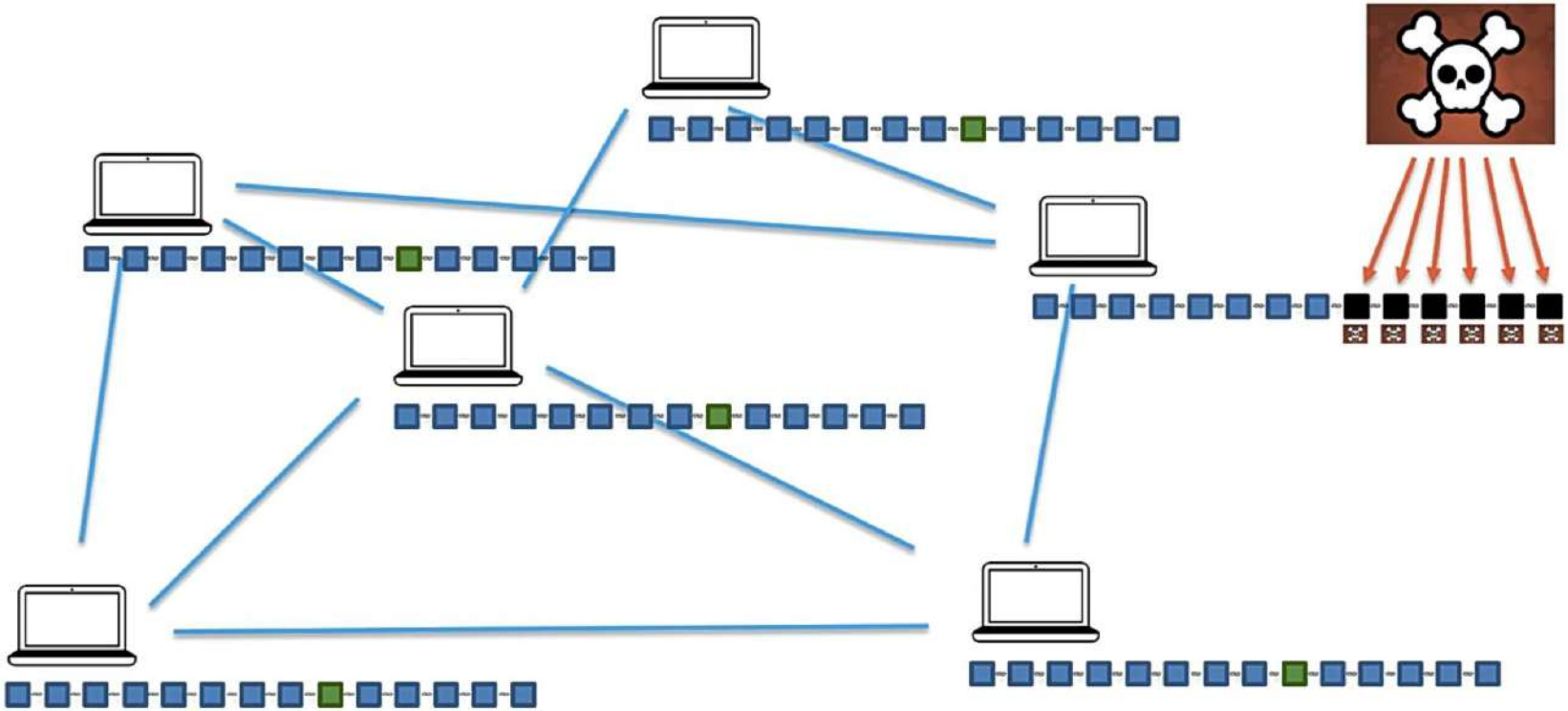
Distributed P2P Network



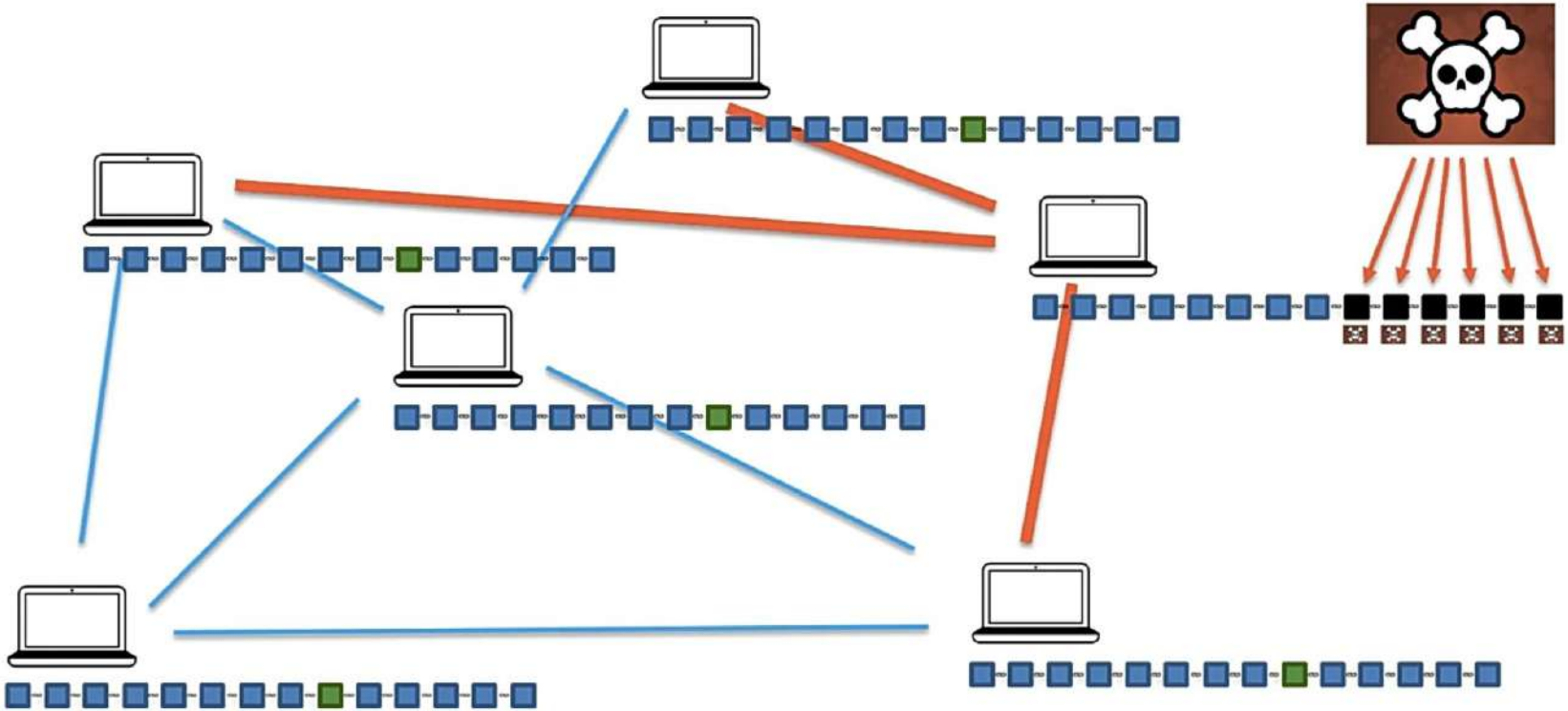
Distributed P2P Network



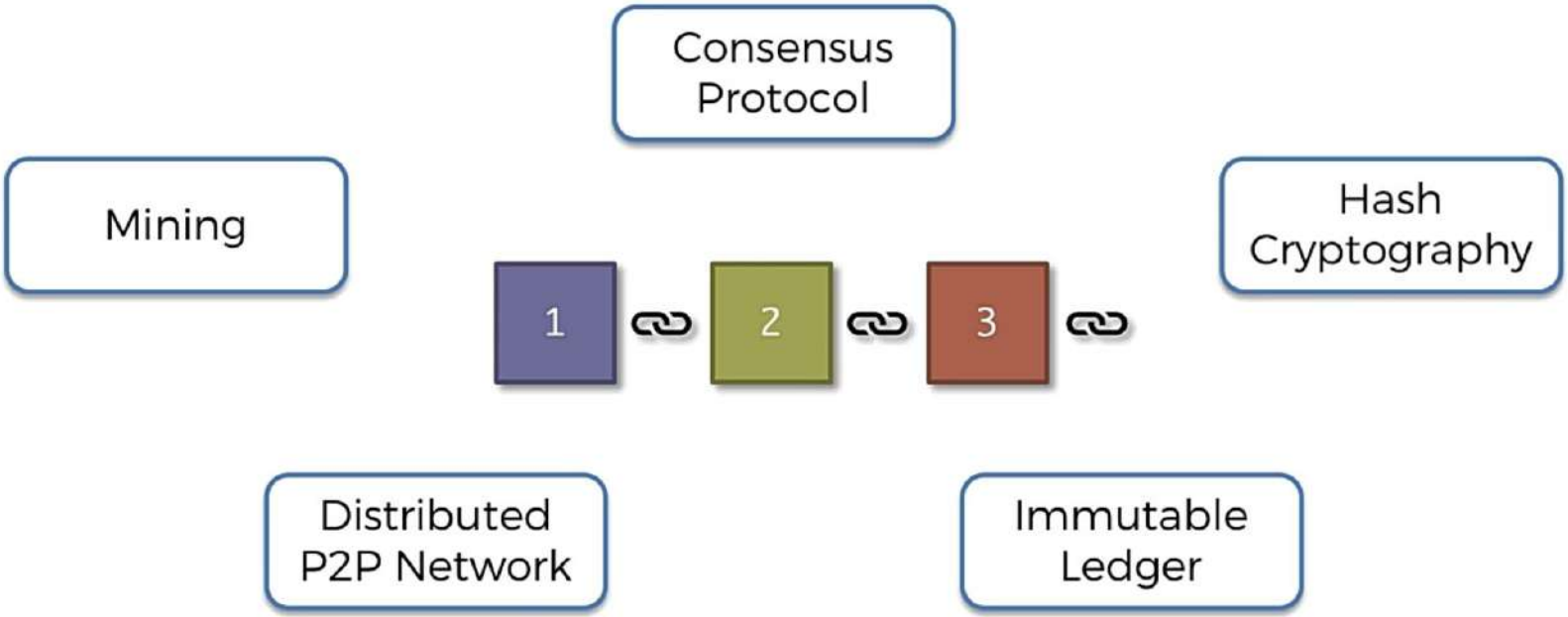
Distributed P2P Network



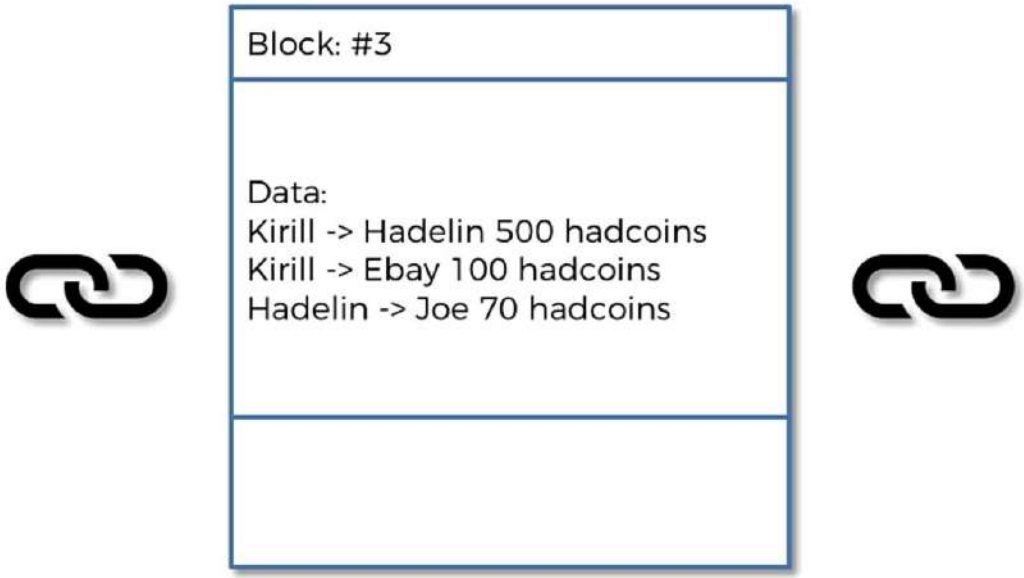
Distributed P2P Network



Blockchain


Consensus Protocol
Mining
Hash Cryptography
1
2
3
Distributed P2P Network
Immutable Ledger

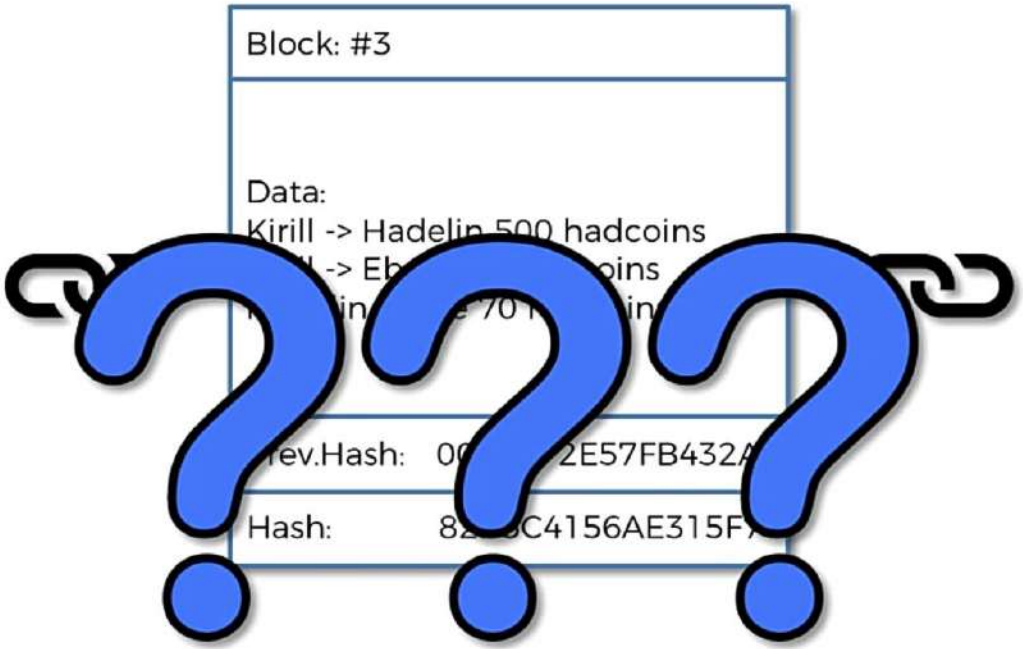
How Mining Works ?



How Mining Works ?

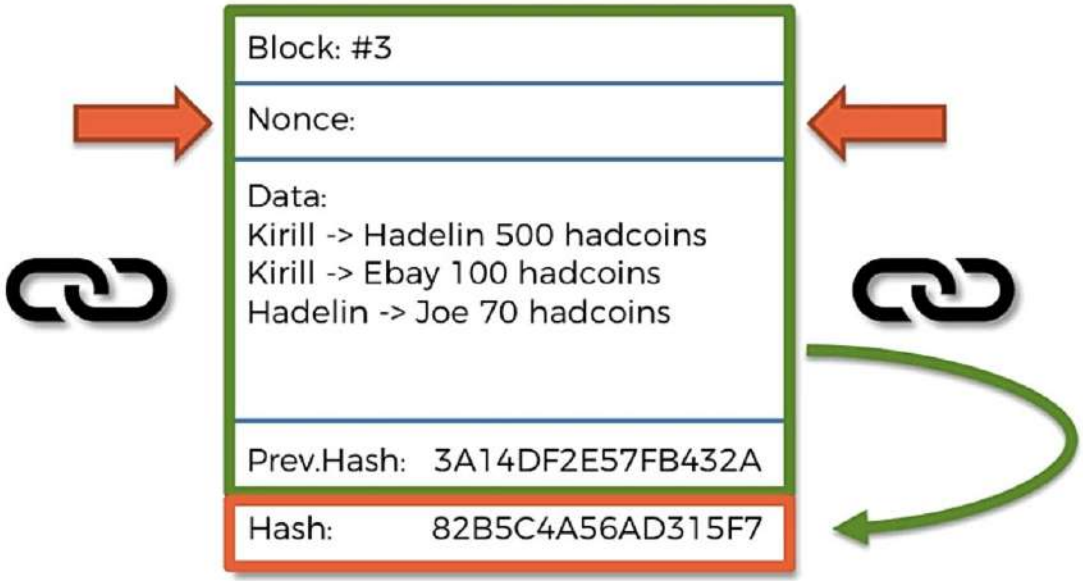
How Mining Works ?

How Mining Works ?

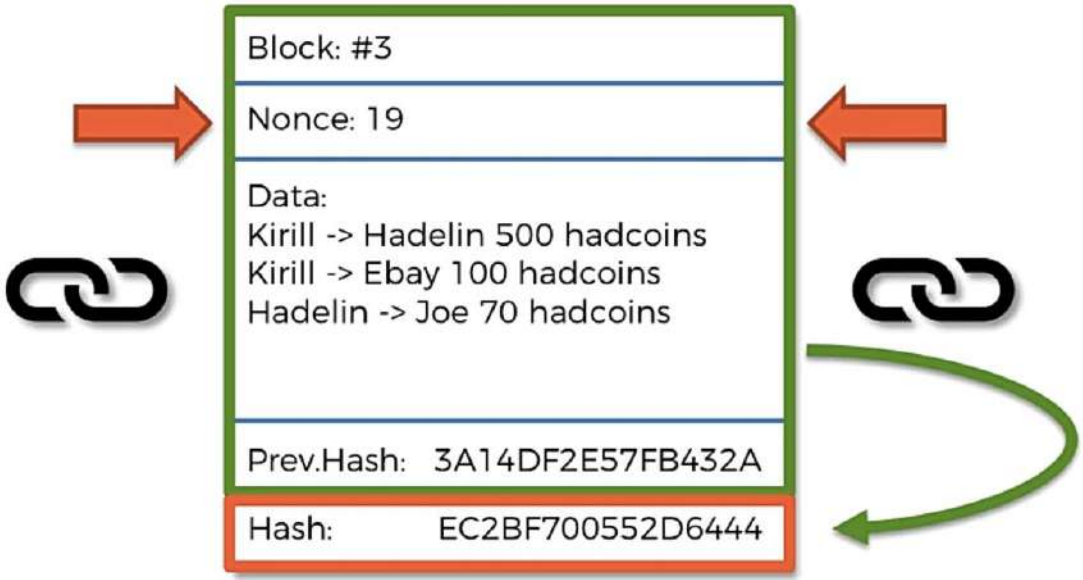


How Mining Works ?

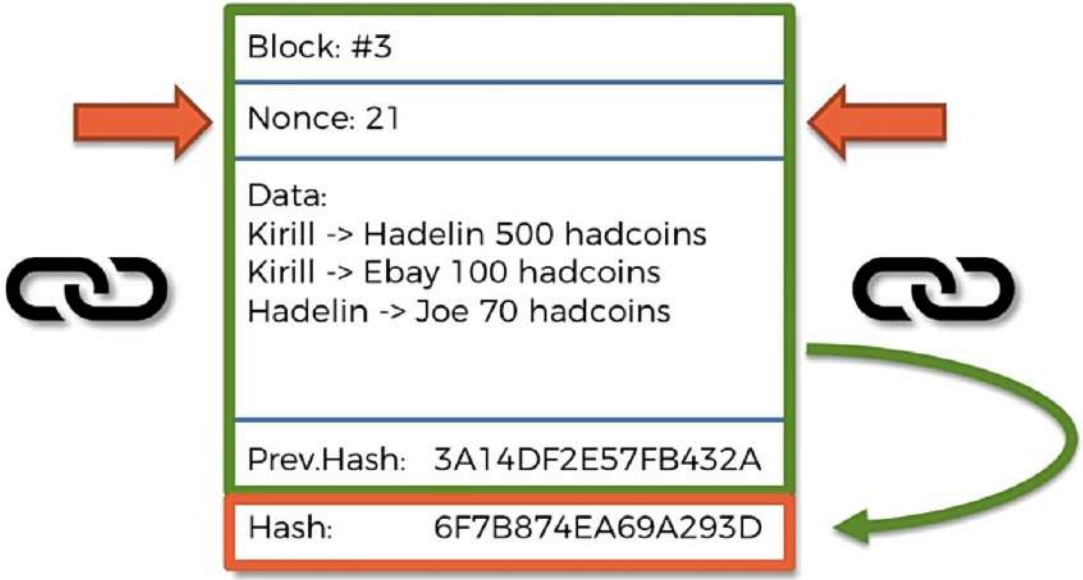
How Mining Works ?



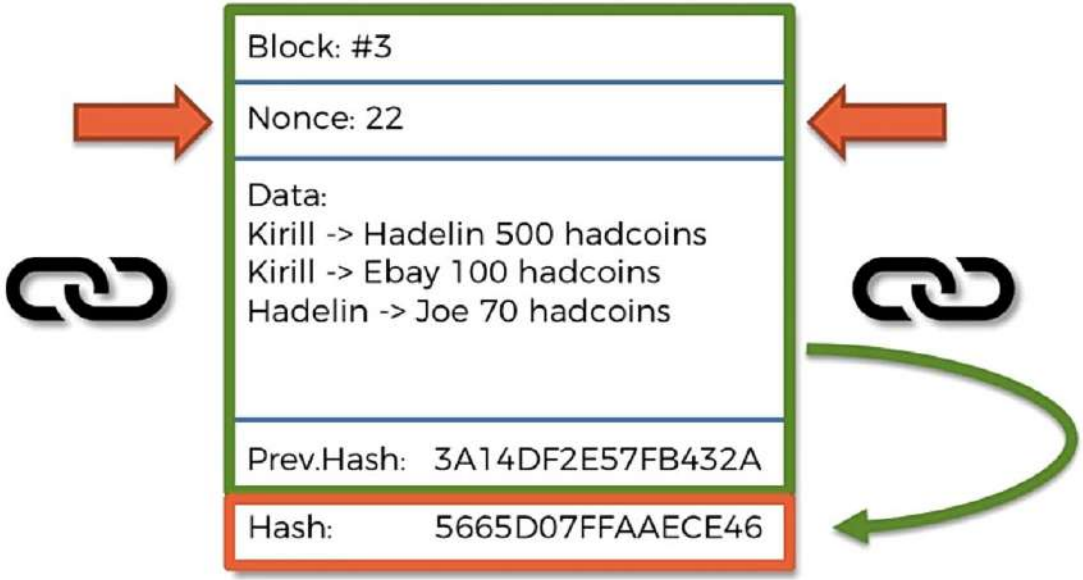
How Mining Works ?



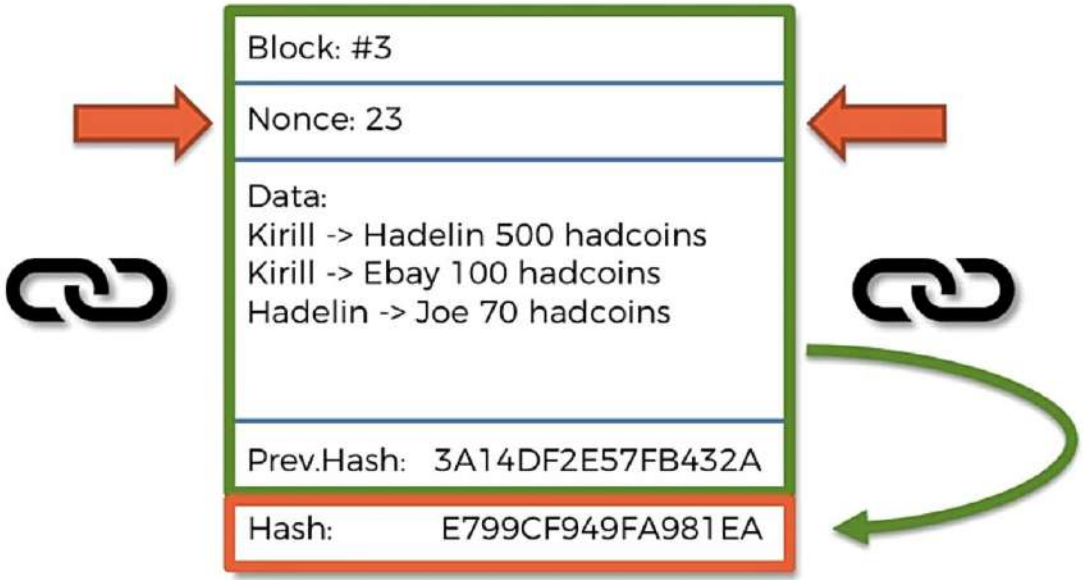
How Mining Works ?



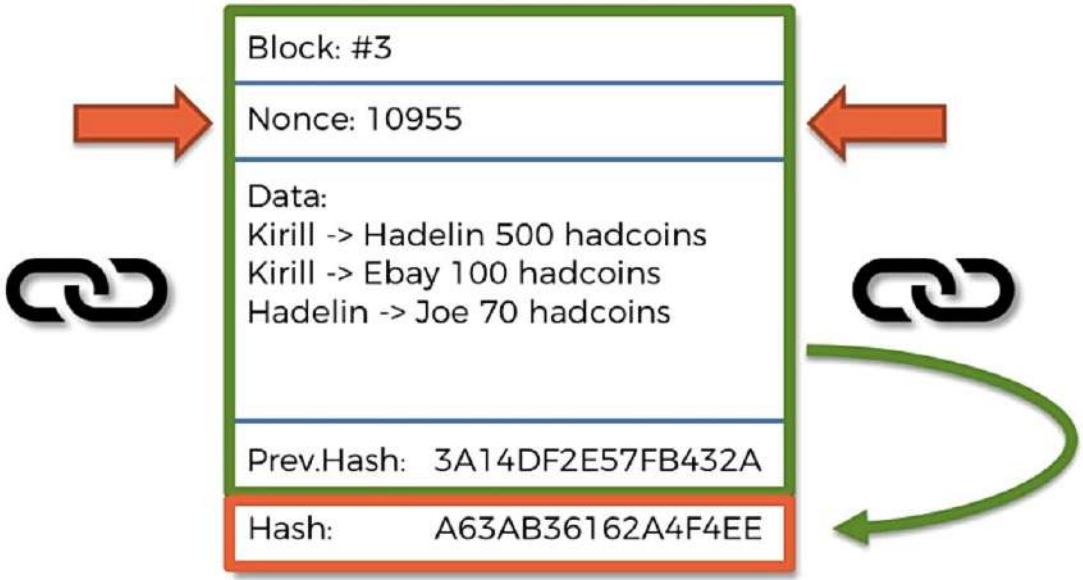
How Mining Works ?



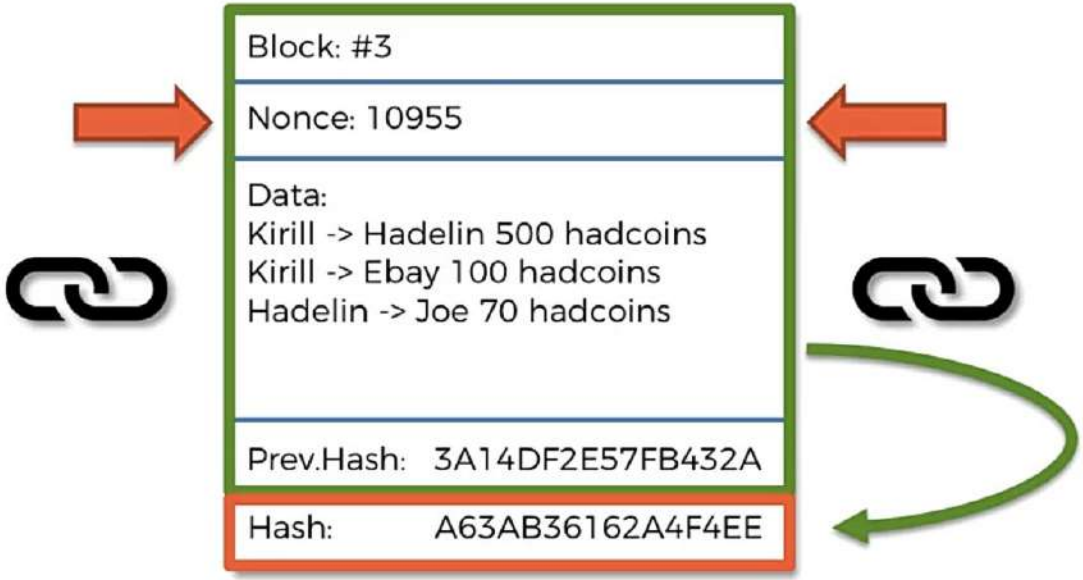
How Mining Works ?



How Mining Works ?



How Mining Works ?



How Mining Works ?

A Hash is a Number

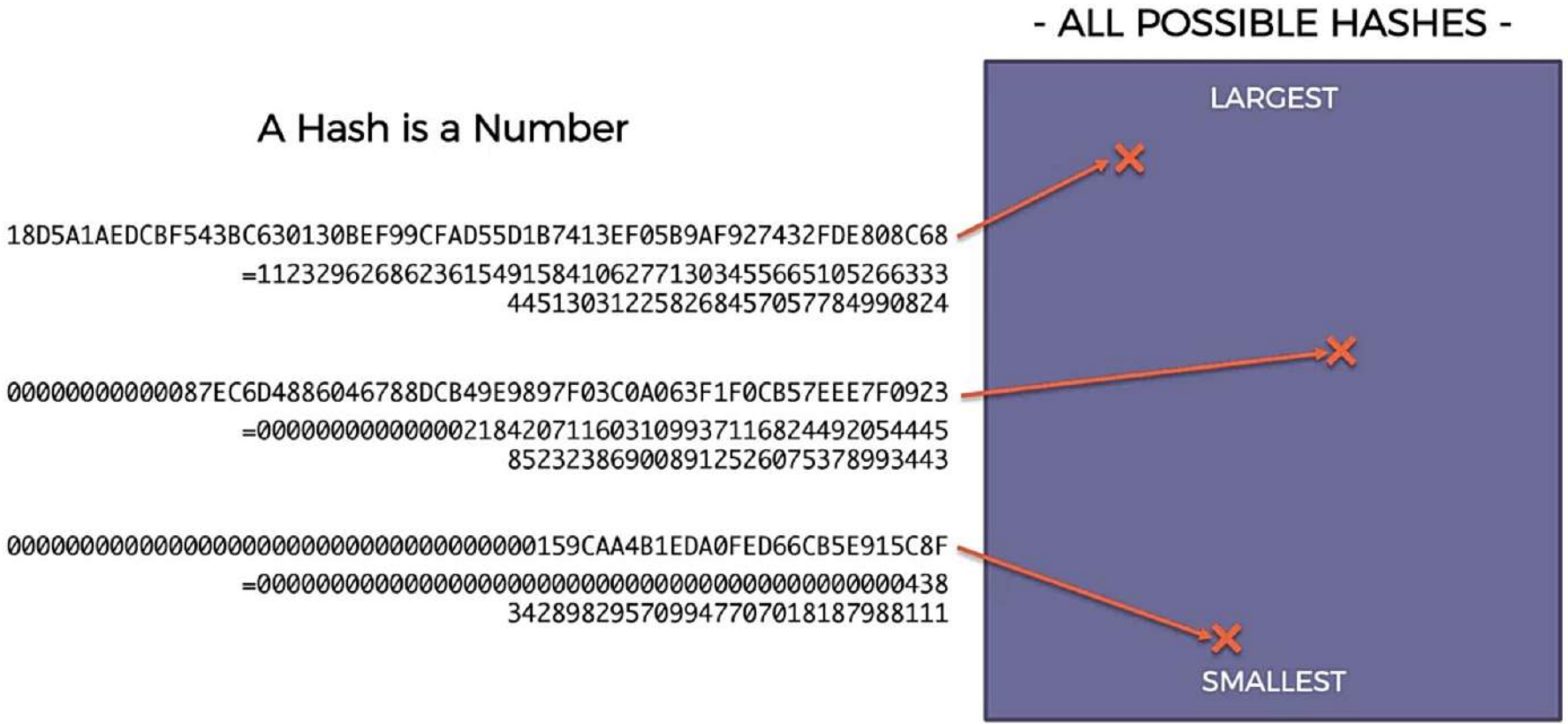
18D5A1AEDCBF543BC630130BEF99CFAD55D1B7413EF05B9AF927432FDE808C68
=112329626862361549158410627713034556651052663334451303122582684570577849908 24

How Mining Works ?

A Hash is a Number

```
18D5A1AEDCBF543BC630130BEF99CFAD55D1B7413EF05B9AF927432FDE808C68
                =112329626862361549158410627713034556651052663334451303122582684570577849908240000000000000087EC6D4886046788DCB49E9897F03C0A063F1F0CB57EEE7F0923
                =000000000000000218420711603109937116824492054445852323869008912526075378993443000000000000000000000000000000000159CAA4B1EDA0FED66CB5E915C8F
                =000000000000000000000000000000000000000000000000438342898295709947707018187988111
```

How Mining Works ?

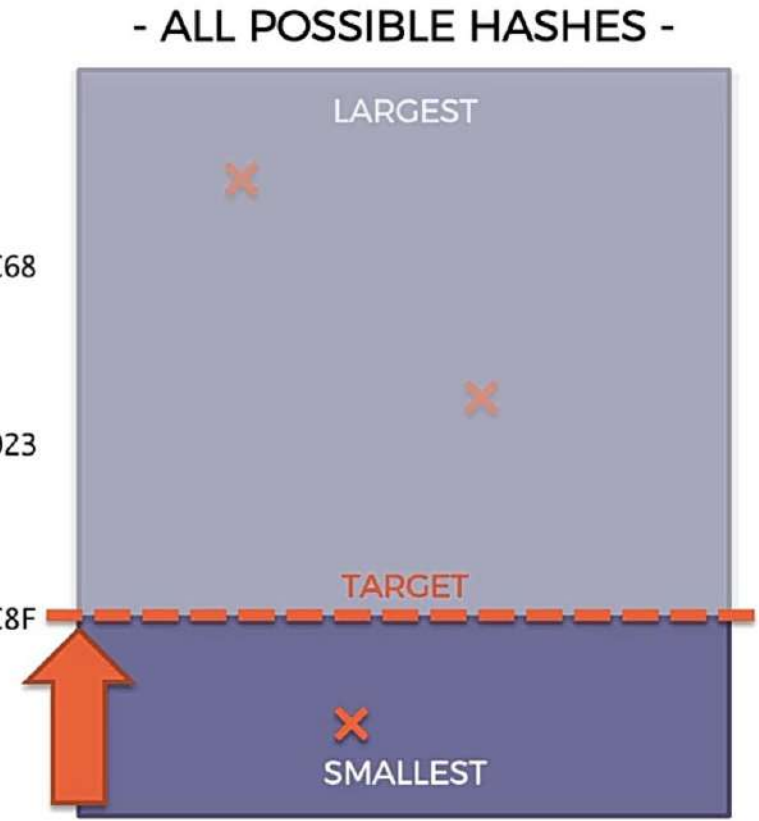


How Mining Works ?

18D5A1AEDCBF543BC630130BEF99CFAD55D1B7413EF05B9AF927432FDE808C68

00000000000087EC6D4886046788DCB49E9897F03C0A063F1F0CB57EEE7F0923

0000000000000000000000000000000000000159CAA4B1EDA0FED66CB5E915C8F

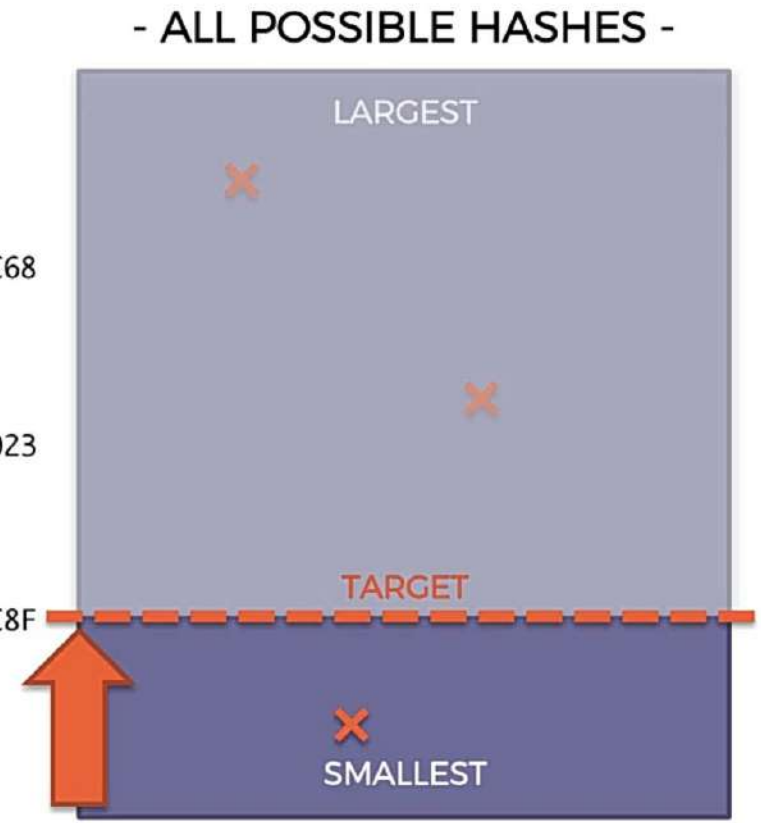


How Mining Works ?

18D5A1AEDCBF543BC630130BEF99CFAD55D1B7413EF05B9AF927432FDE808C68

00000000000087EC6D4886046788DCB49E9897F03C0A063F1F0CB57EEE7F0923

00000000000000000000000000000000000000159CAA4B1EDA0FED66CB5E915C8F



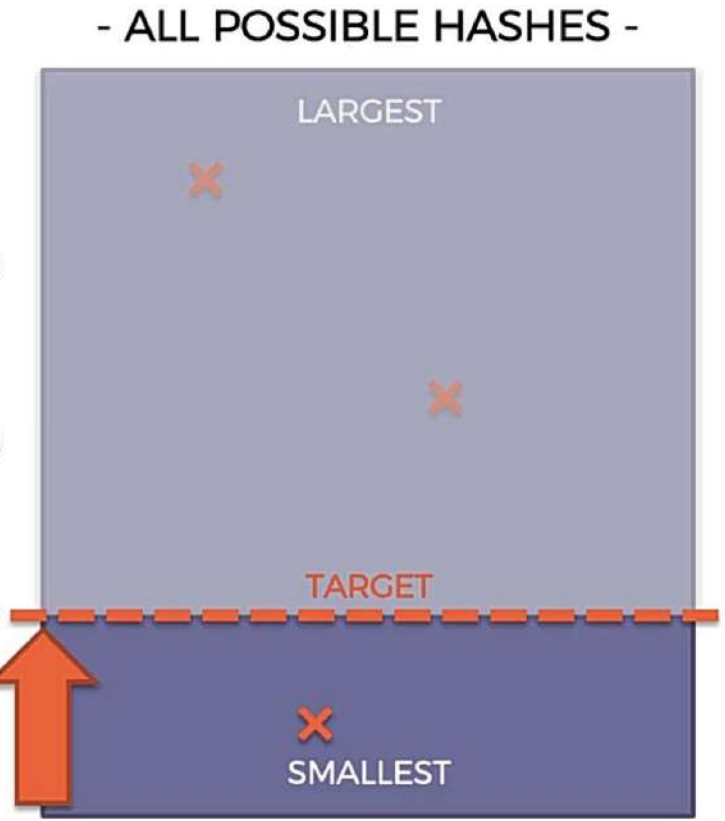
How Mining Works ?

18D5A1AEDCBF543BC630130BEF99CFAD55D1B7413EF05B9AF927432FDE808C68

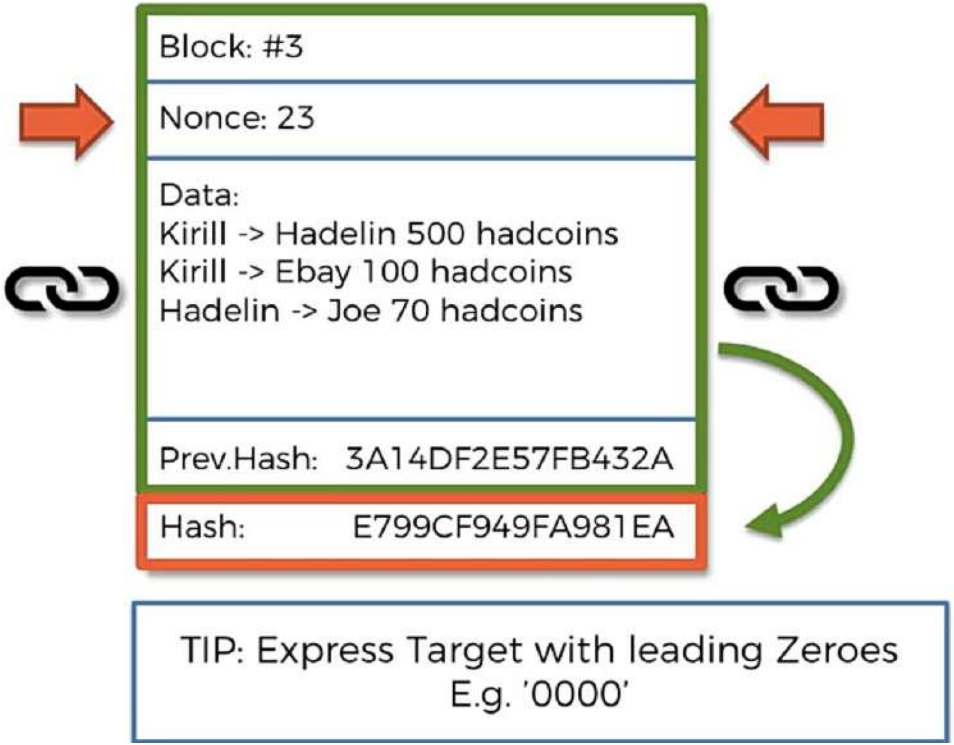
00000000000087EC6D4886046788DCB49E9897F03C0A063F1F0CB57EEE7F0923

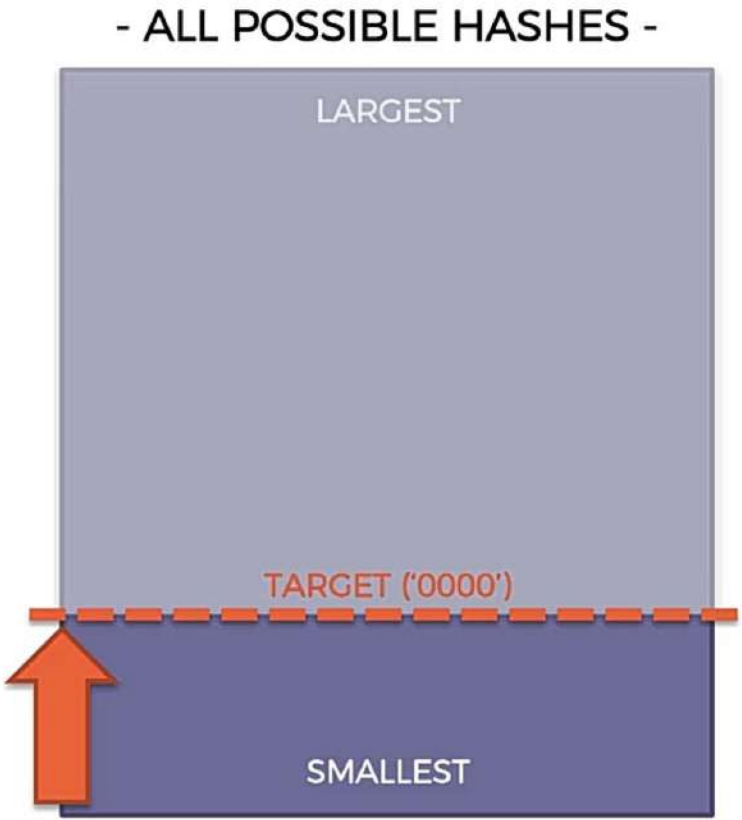
00000000000000000000000000000000000000159CAA4B1EDA0FED66CB5E915C8F

TIP: Express Target with leading Zeroes
E.g. '0000'

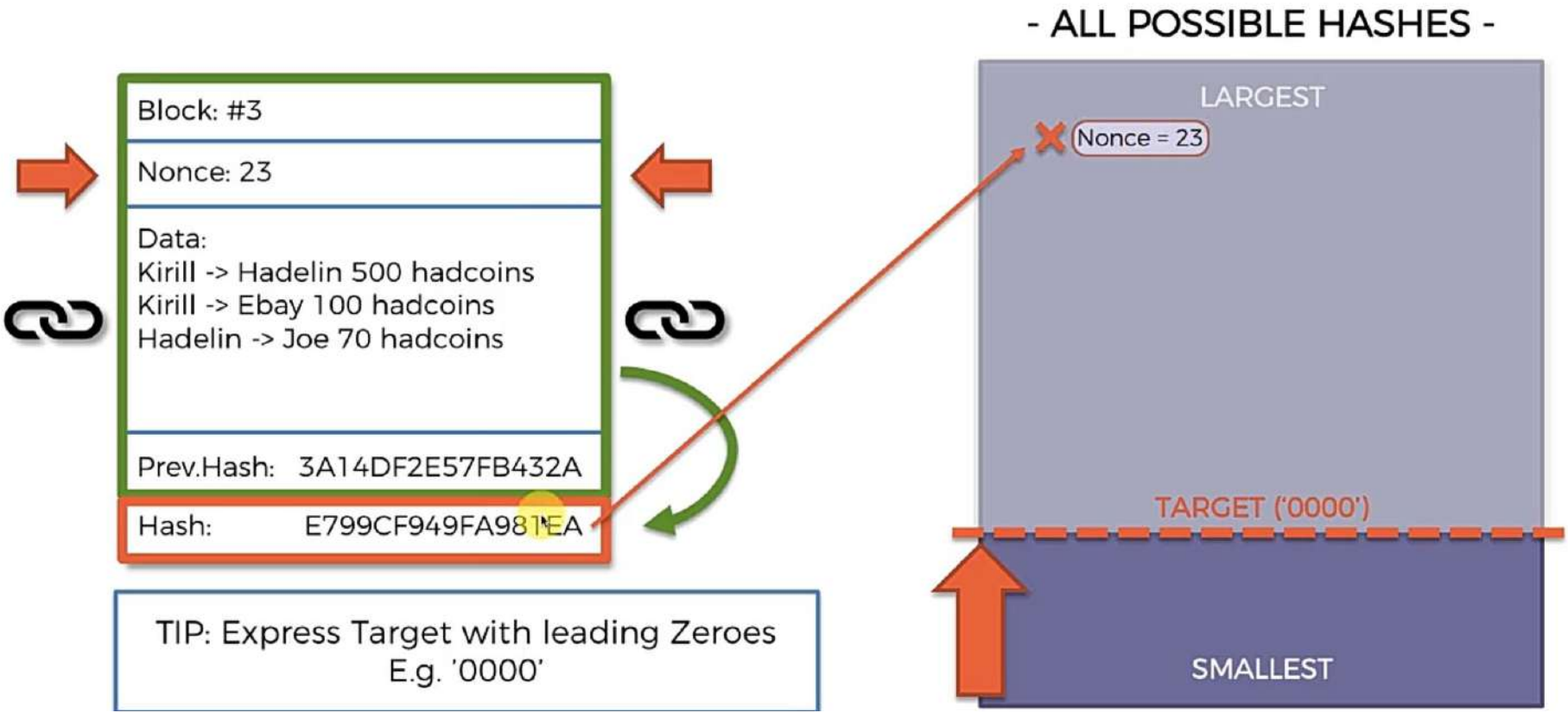


How Mining Works ?

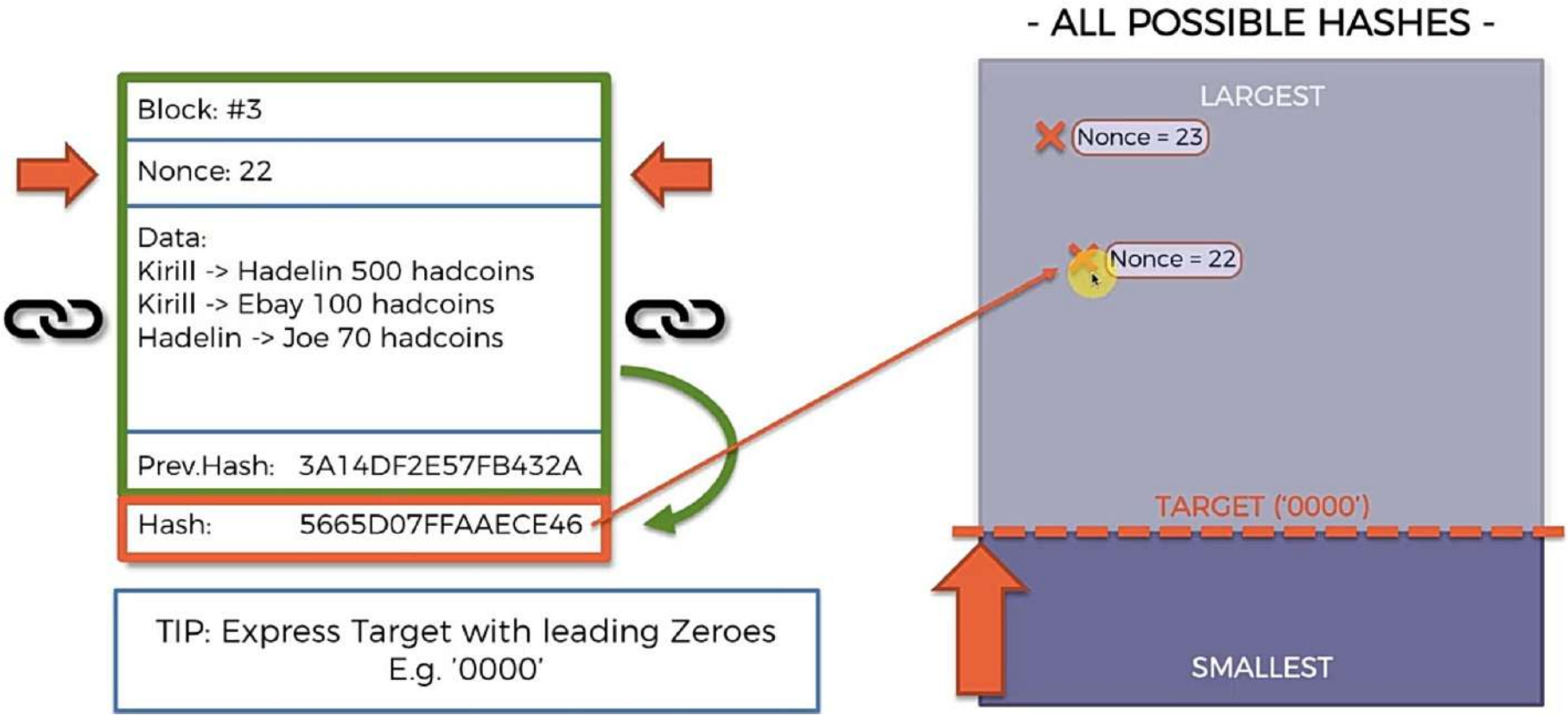




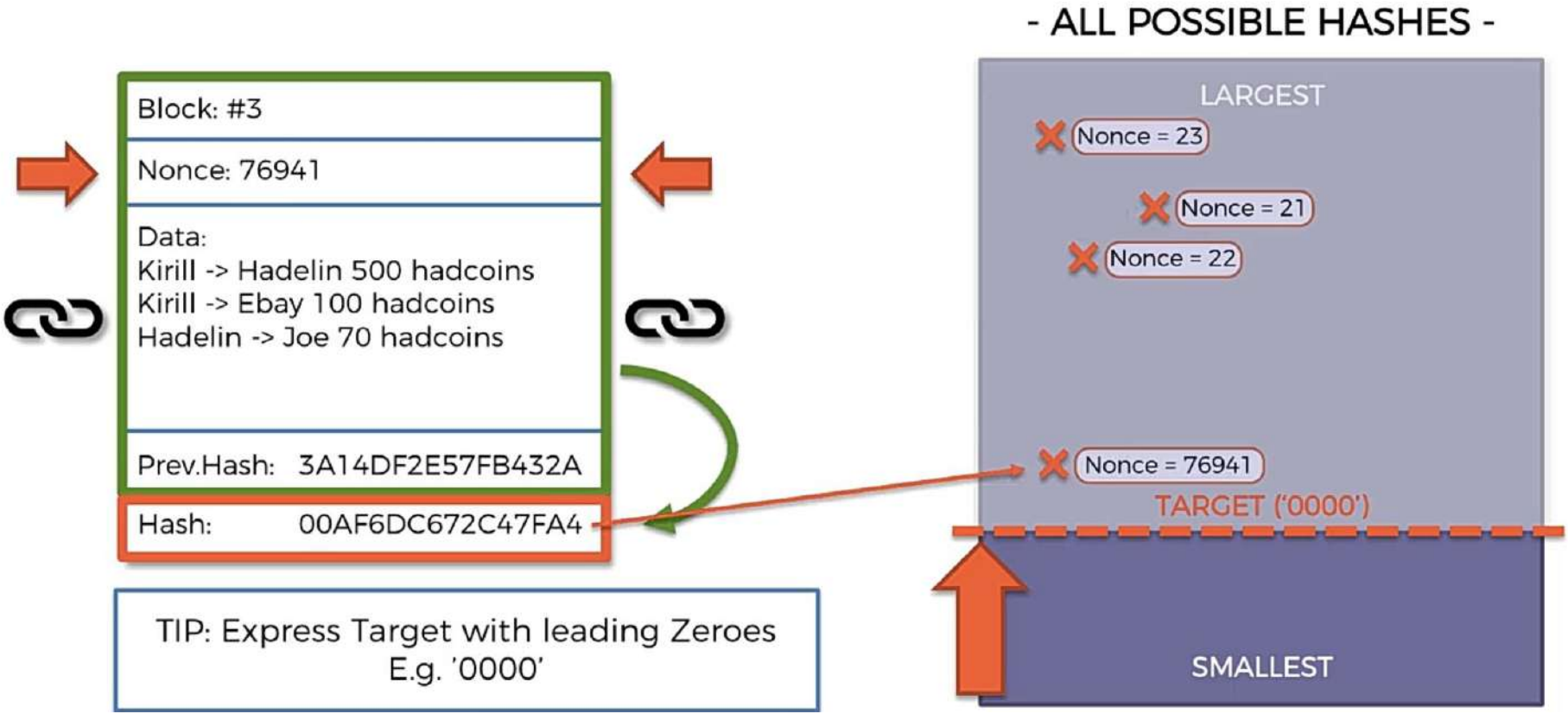
How Mining Works ?



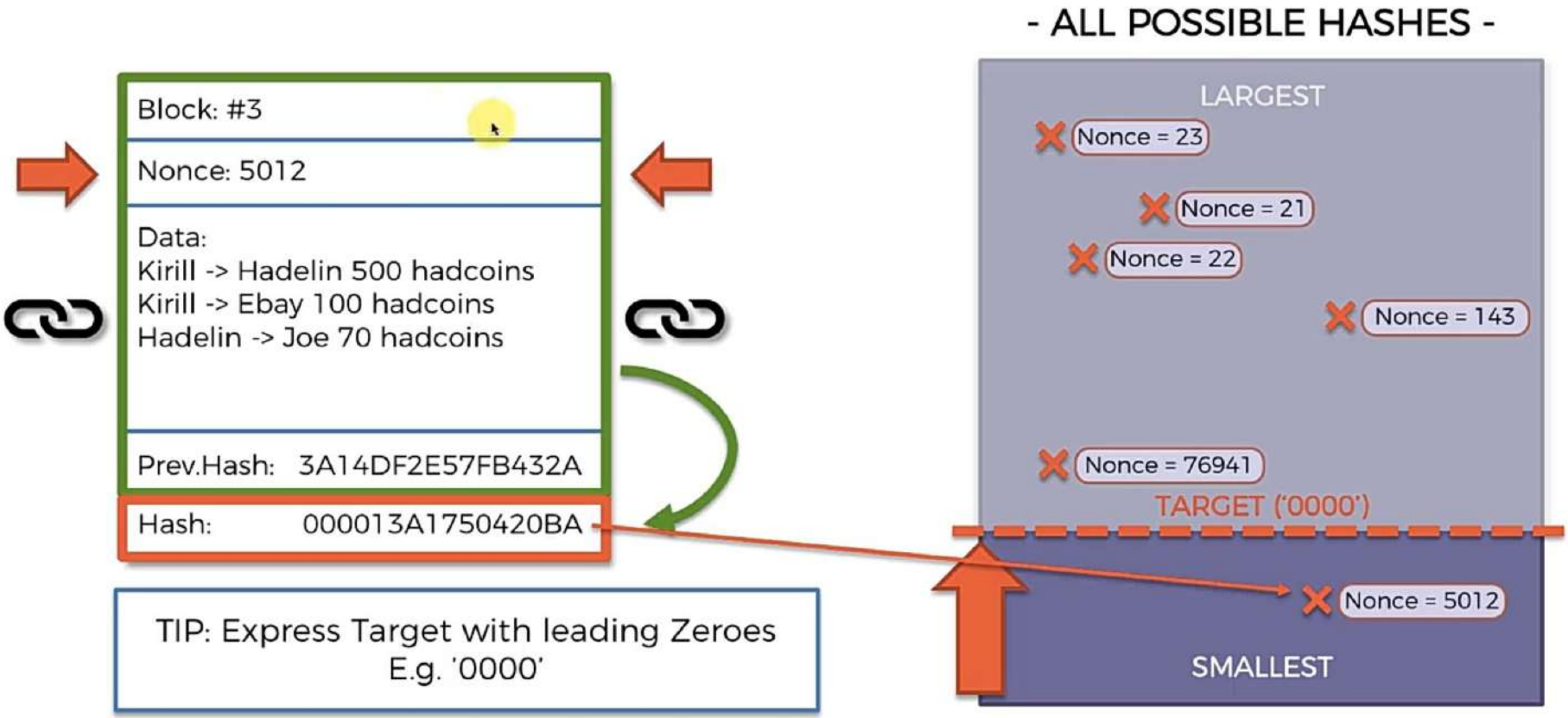
How Mining Works ?



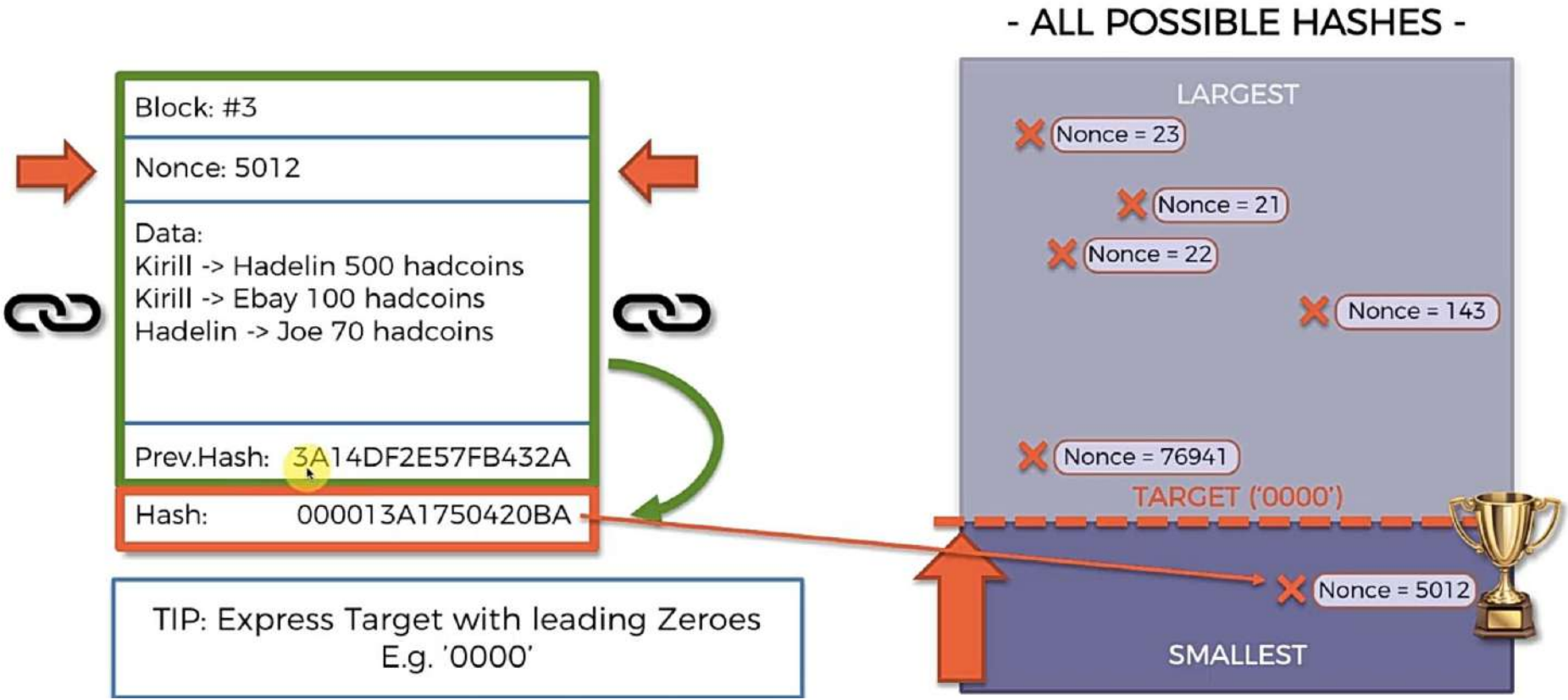
How Mining Works ?



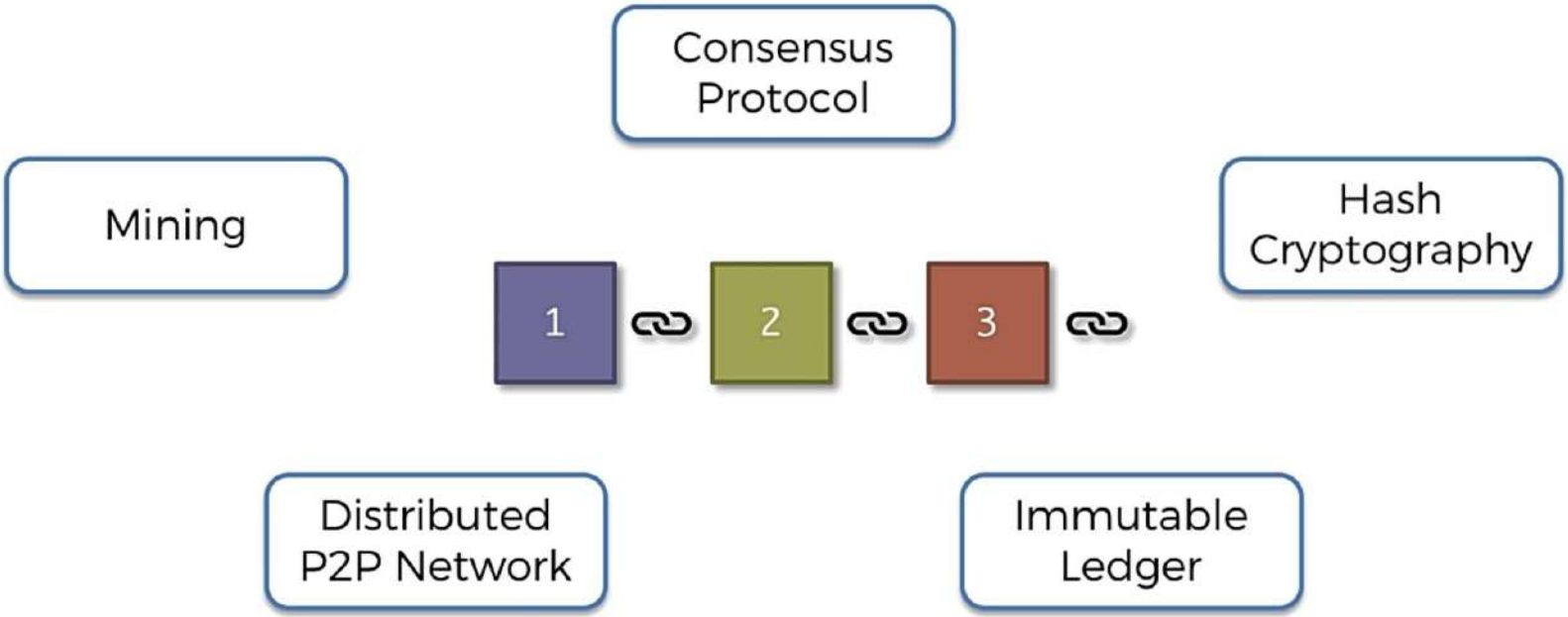
How Mining Works ?



How Mining Works ?



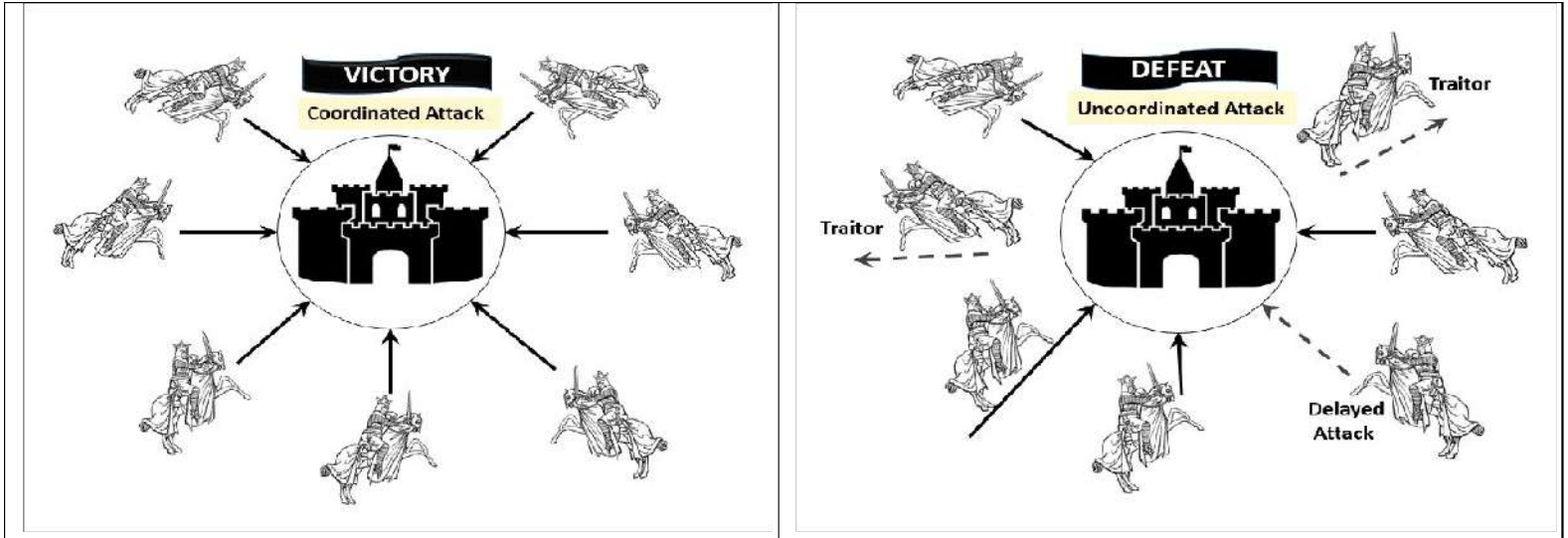
Blockchain


Consensus Protocol
Mining
Hash Cryptography
1
2
3
Distributed P2P Network
Immutable Ledger

What is Consensus?

- As per Webster dictionary, a consensus is a **general agreement or opinion shared by all the people in a group.**
- A protocol is a **system of standard rules that are acceptable by all parties** to control the exchange of information in a network. Thus, a **consensus protocol** in Blockchain can be defined as **a set of rules and procedures for attaining a unified agreement (consensus) between the participating nodes** on the status of the network.
- The consensus protocol **aims to overcome the classic problem of a distributed computing system known as the Byzantine Generals Problem**

Byzantine General Problem



Coordinated attack leading to victory	Uncoordinated attack leading to defeat

Figure 2.12: Byzantine Generals Problem

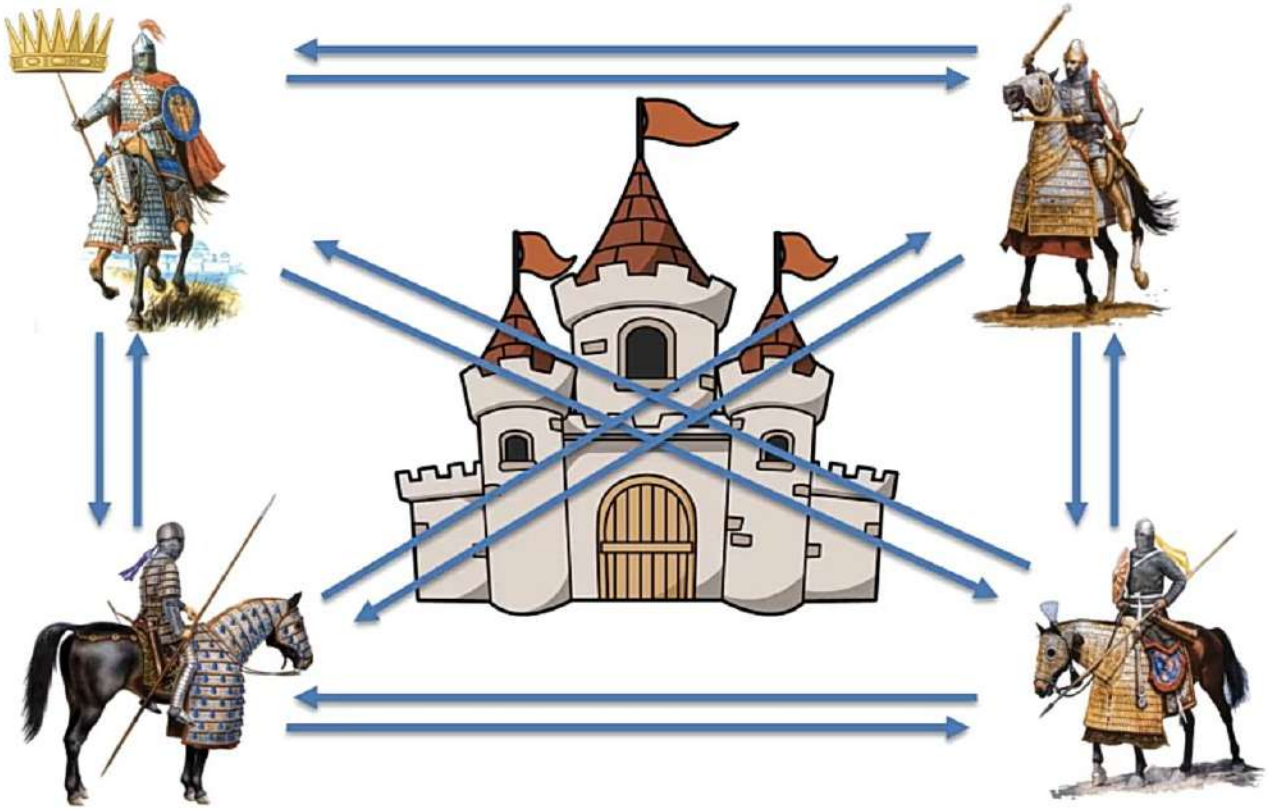
Objectives of Consensus Protocol

Byzantine Fault Tolerance

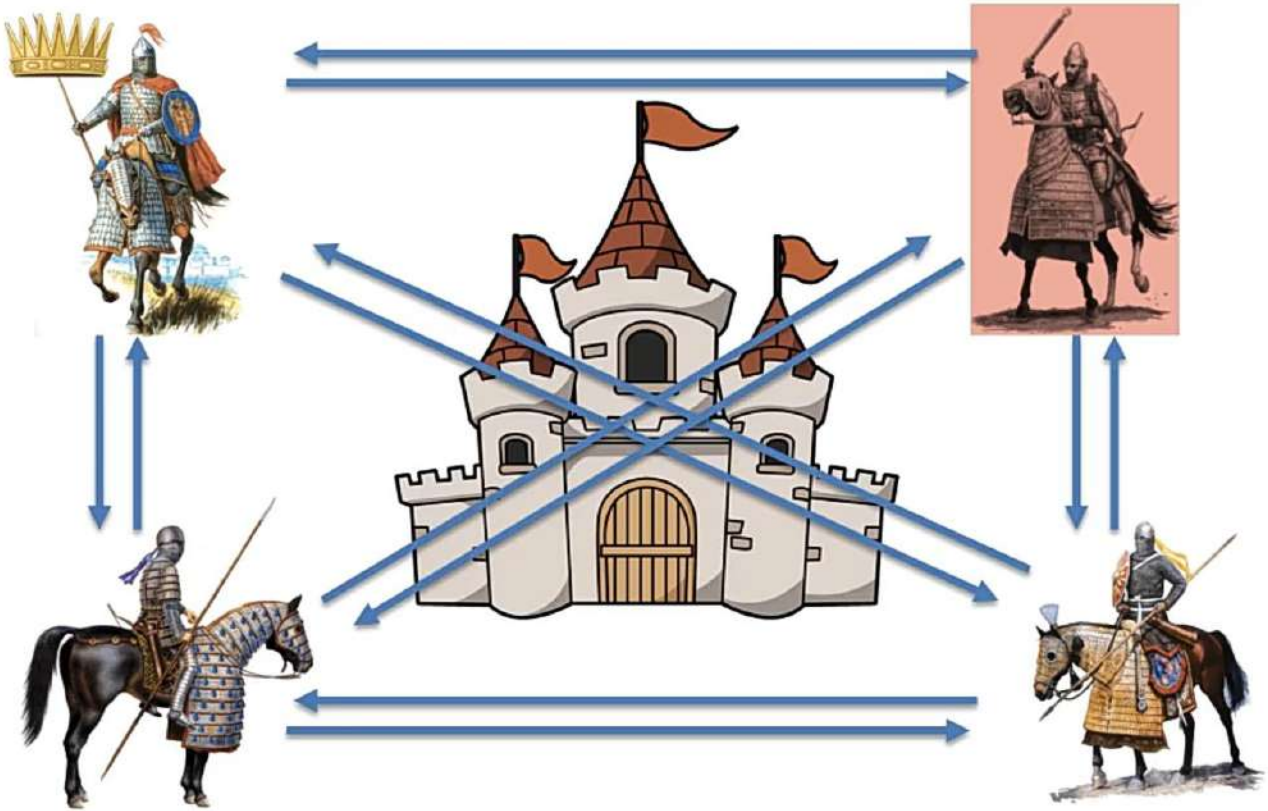
Byzantine Fault Tolerance

Byzantine Fault Tolerance

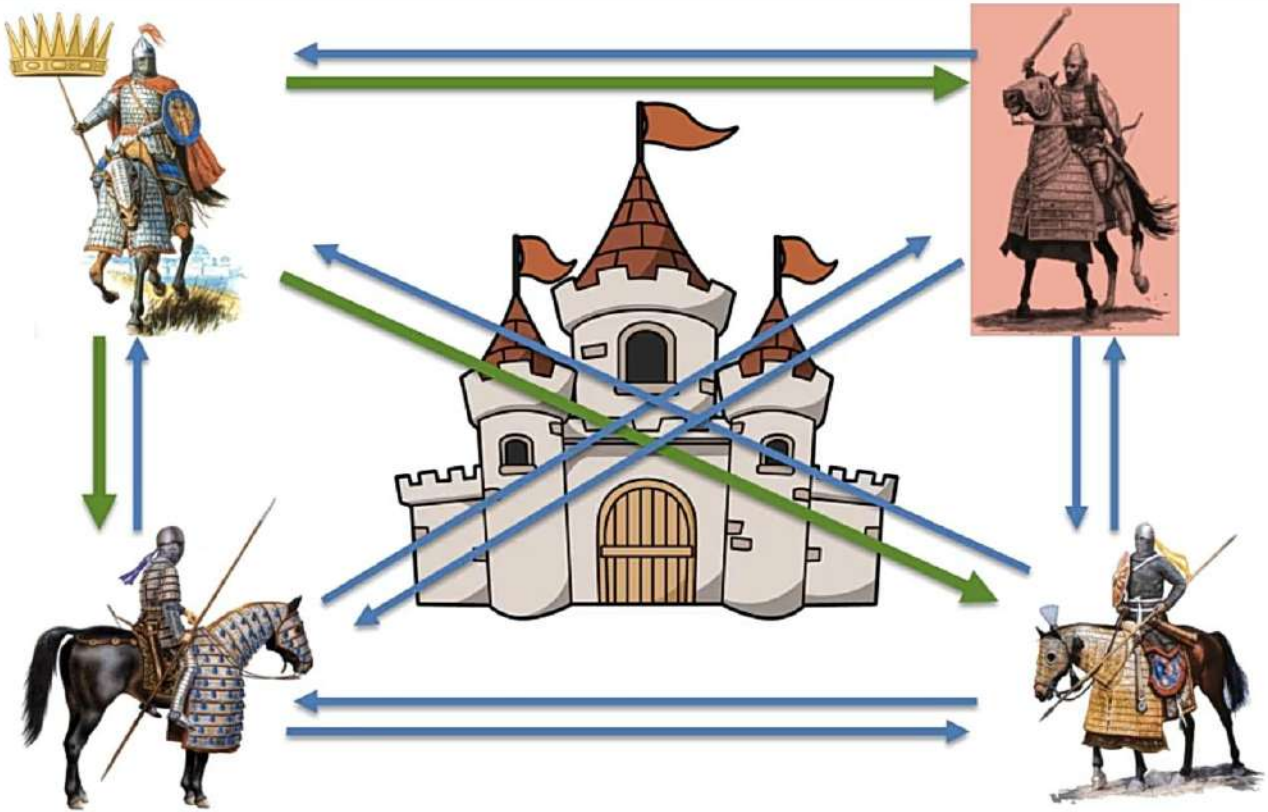
Byzantine Fault Tolerance



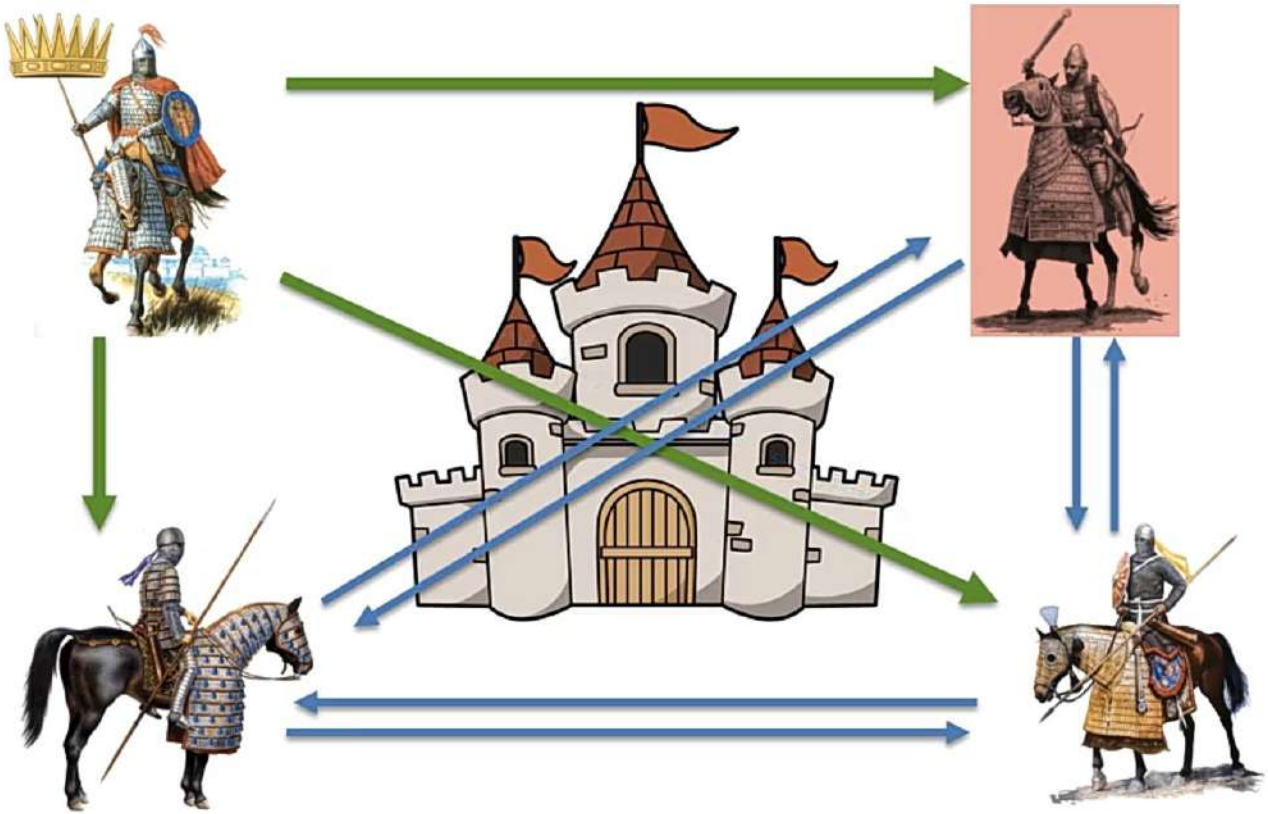
Byzantine Fault Tolerance



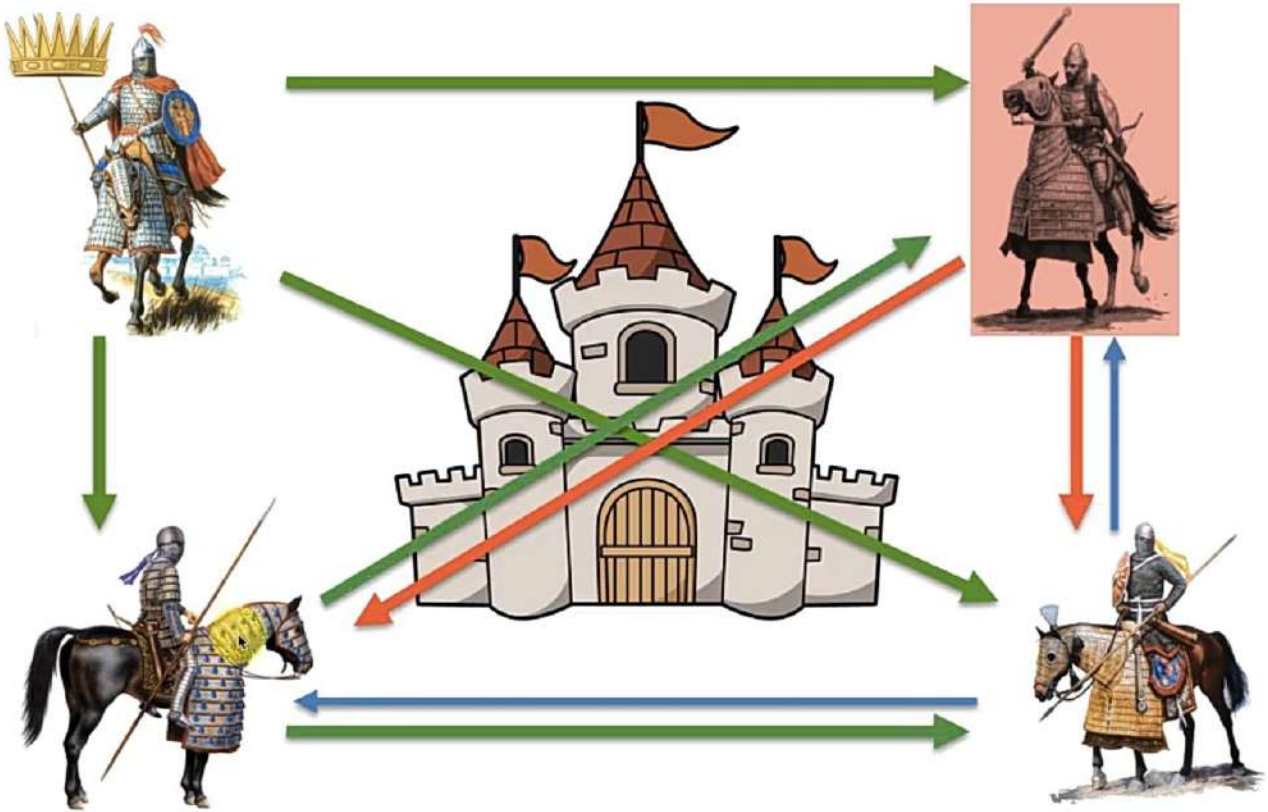
Byzantine Fault Tolerance



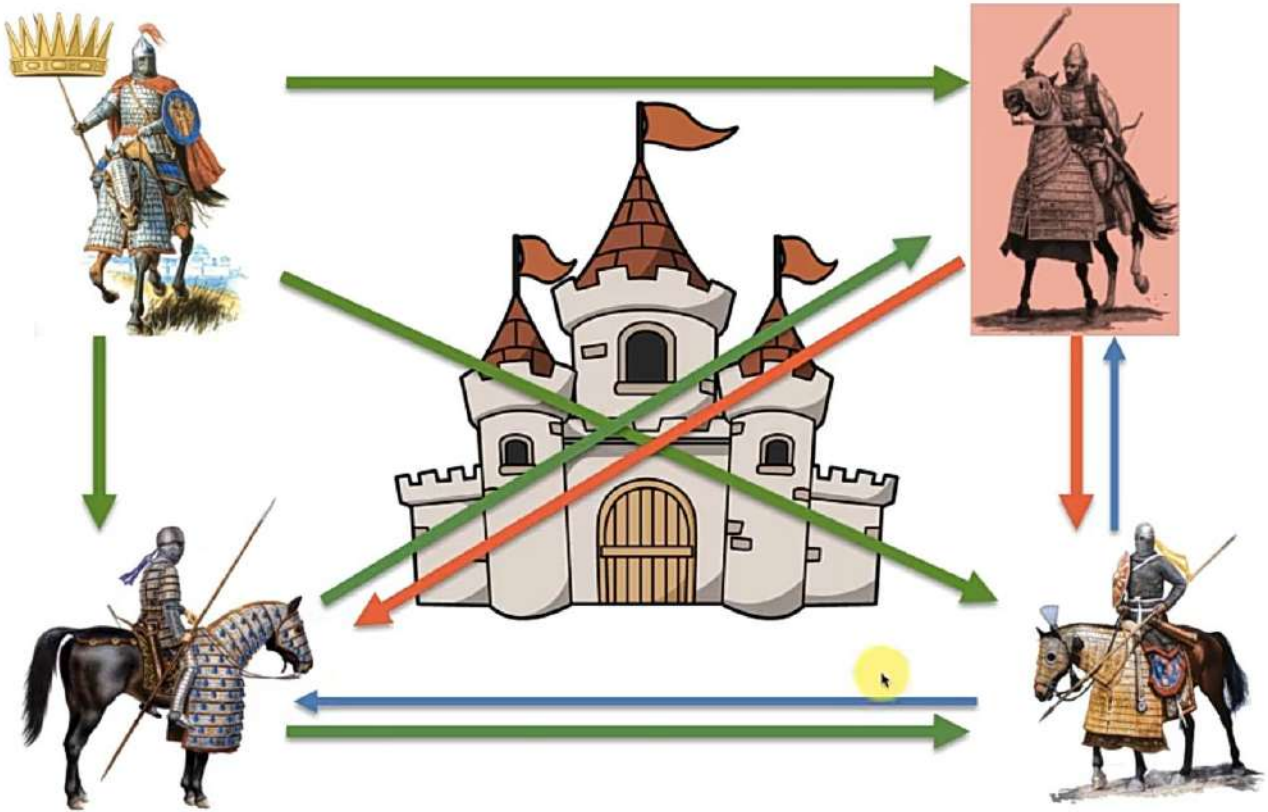
Byzantine Fault Tolerance



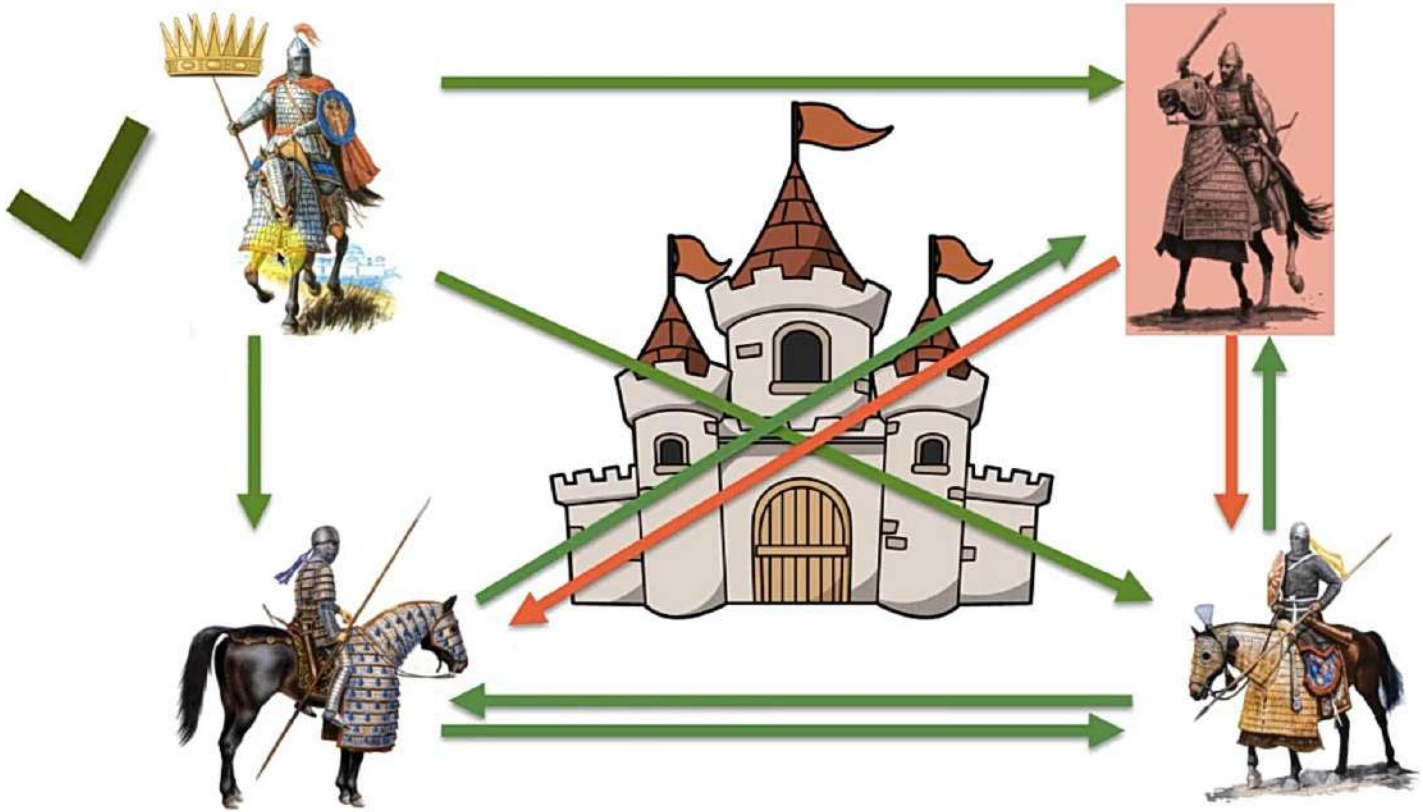
Byzantine Fault Tolerance



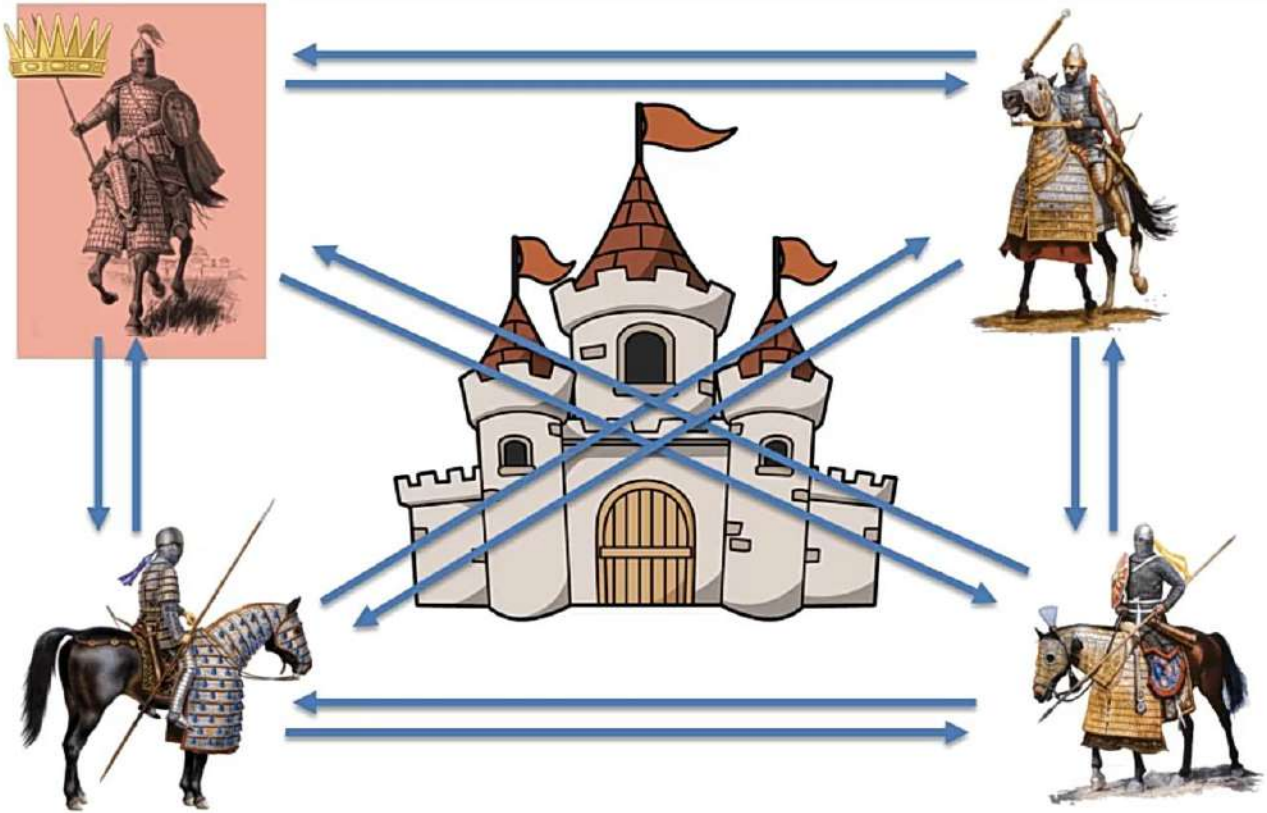
Byzantine Fault Tolerance



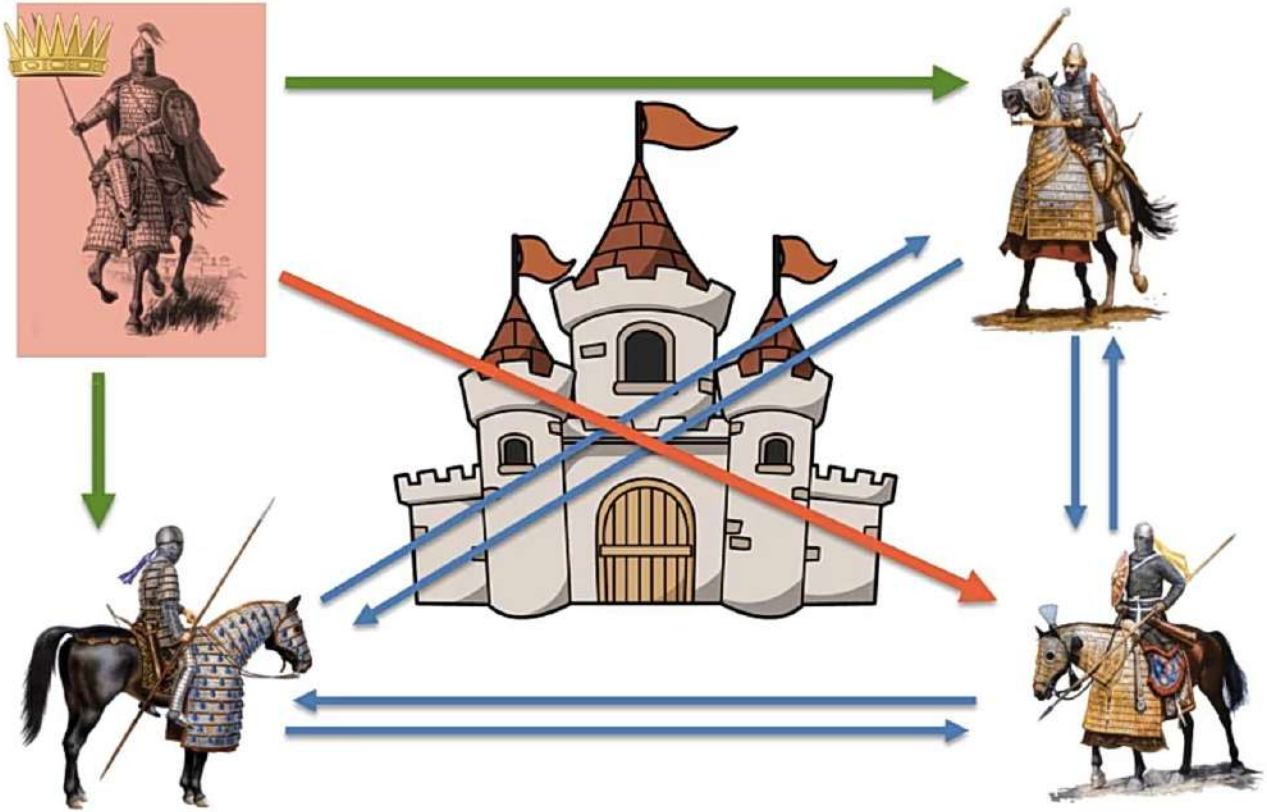
Byzantine Fault Tolerance



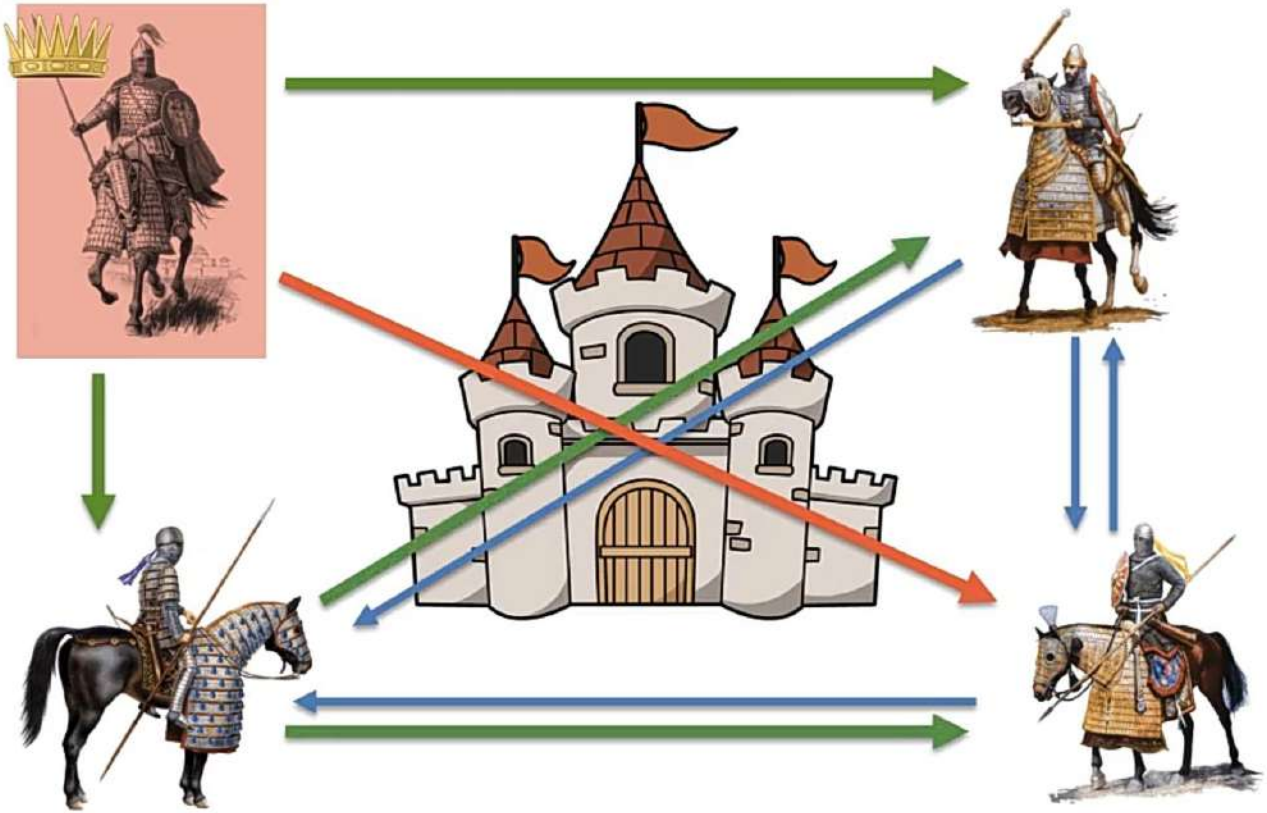
Byzantine Fault Tolerance



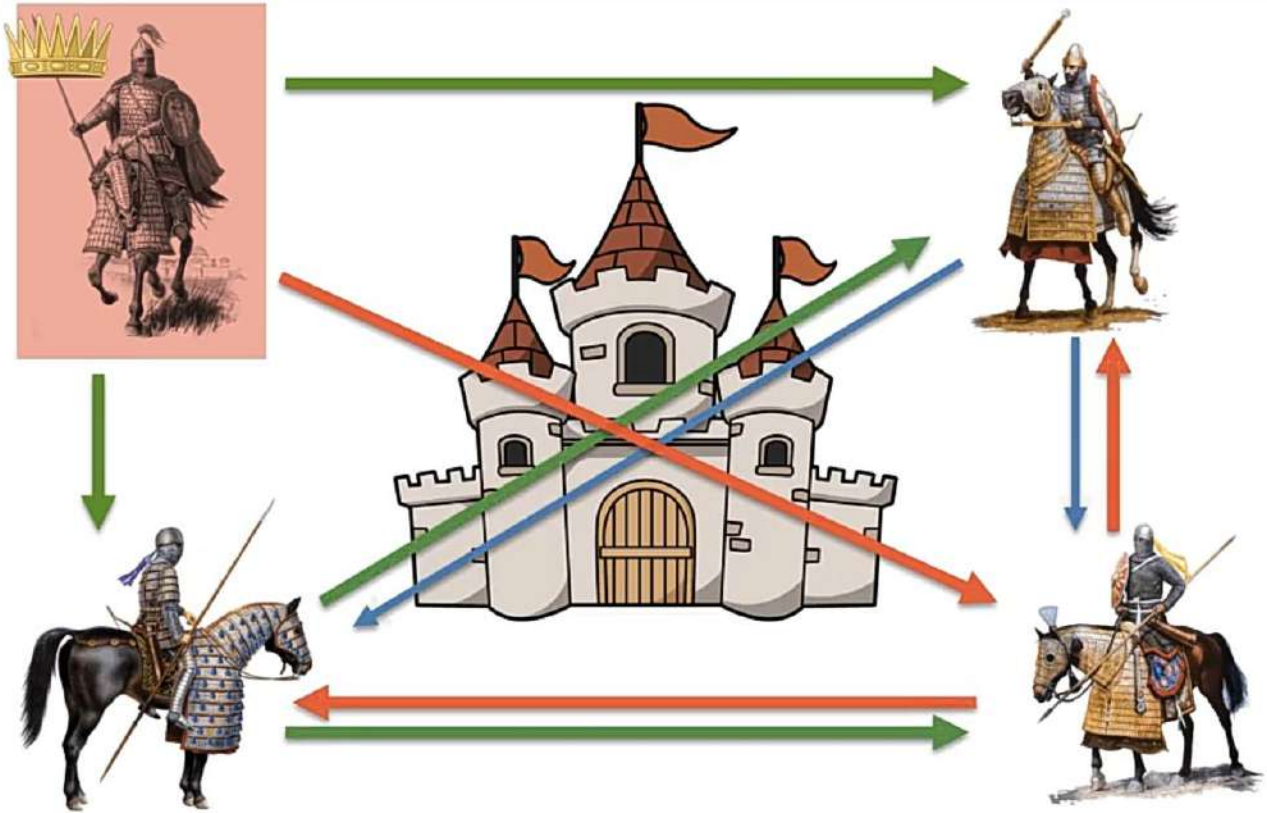
Byzantine Fault Tolerance



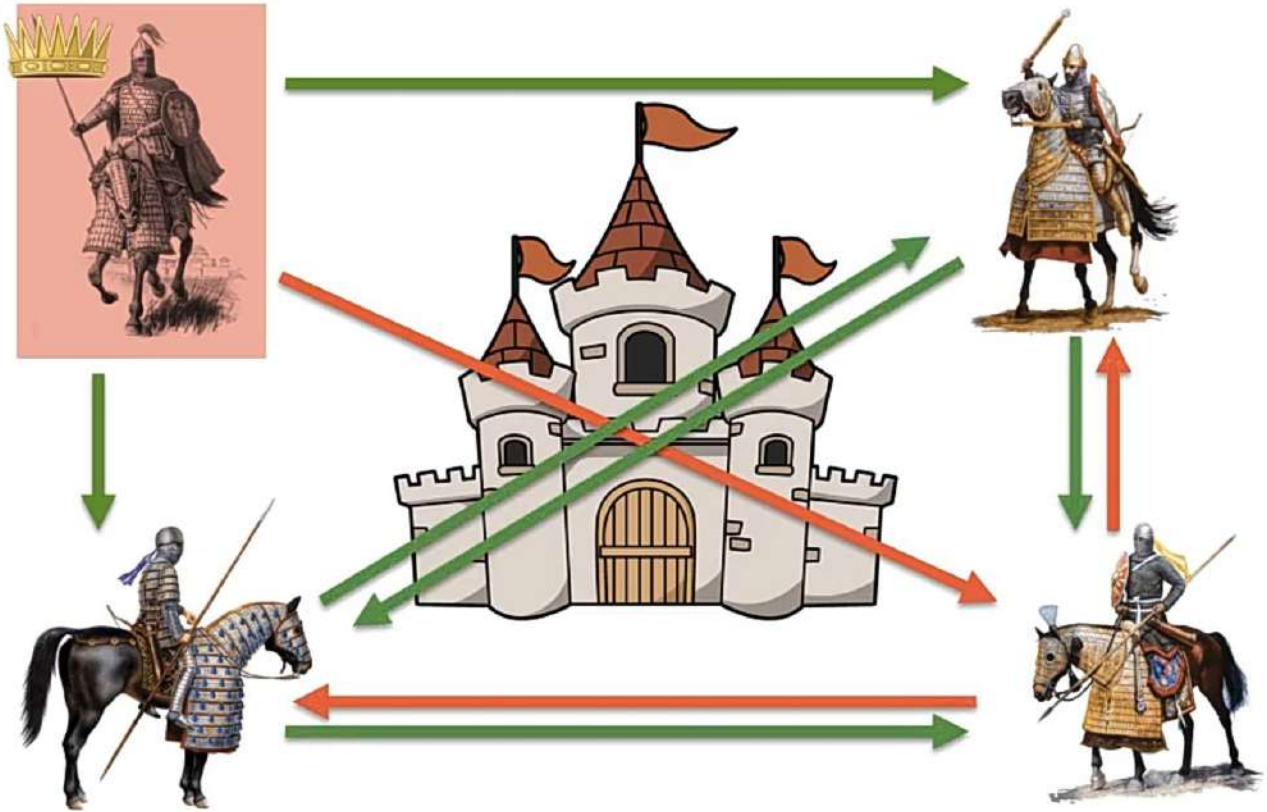
Byzantine Fault Tolerance



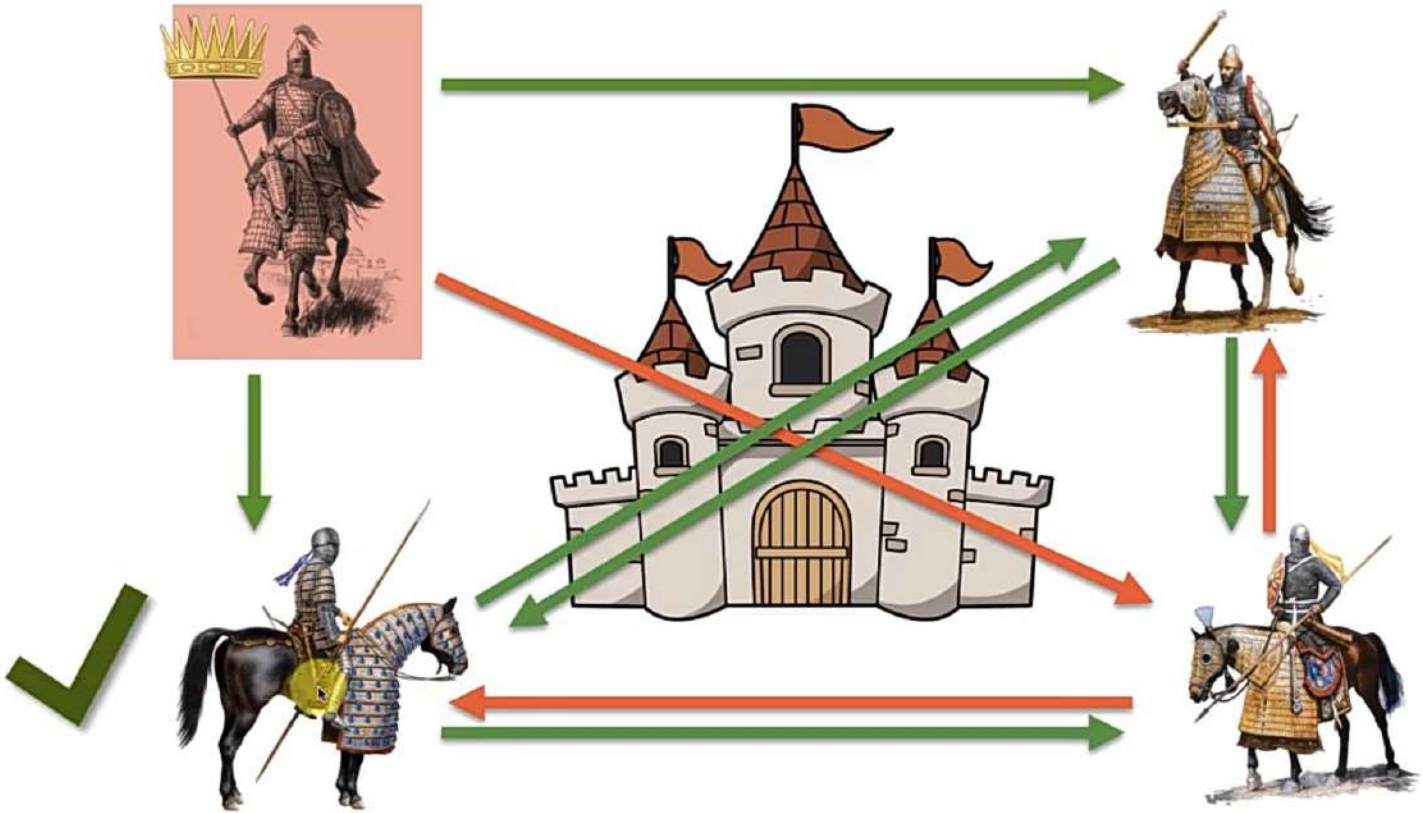
Byzantine Fault Tolerance



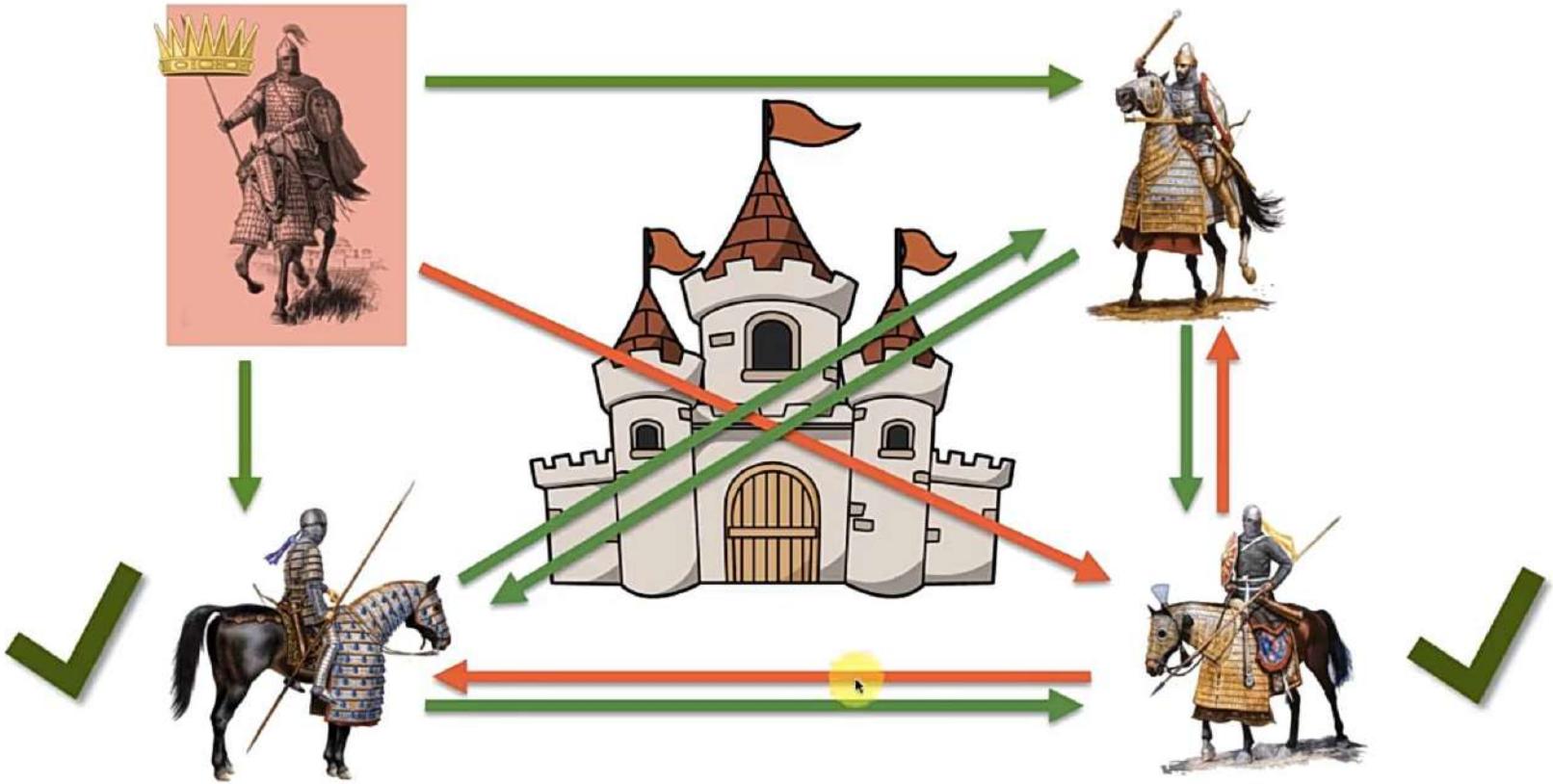
Byzantine Fault Tolerance



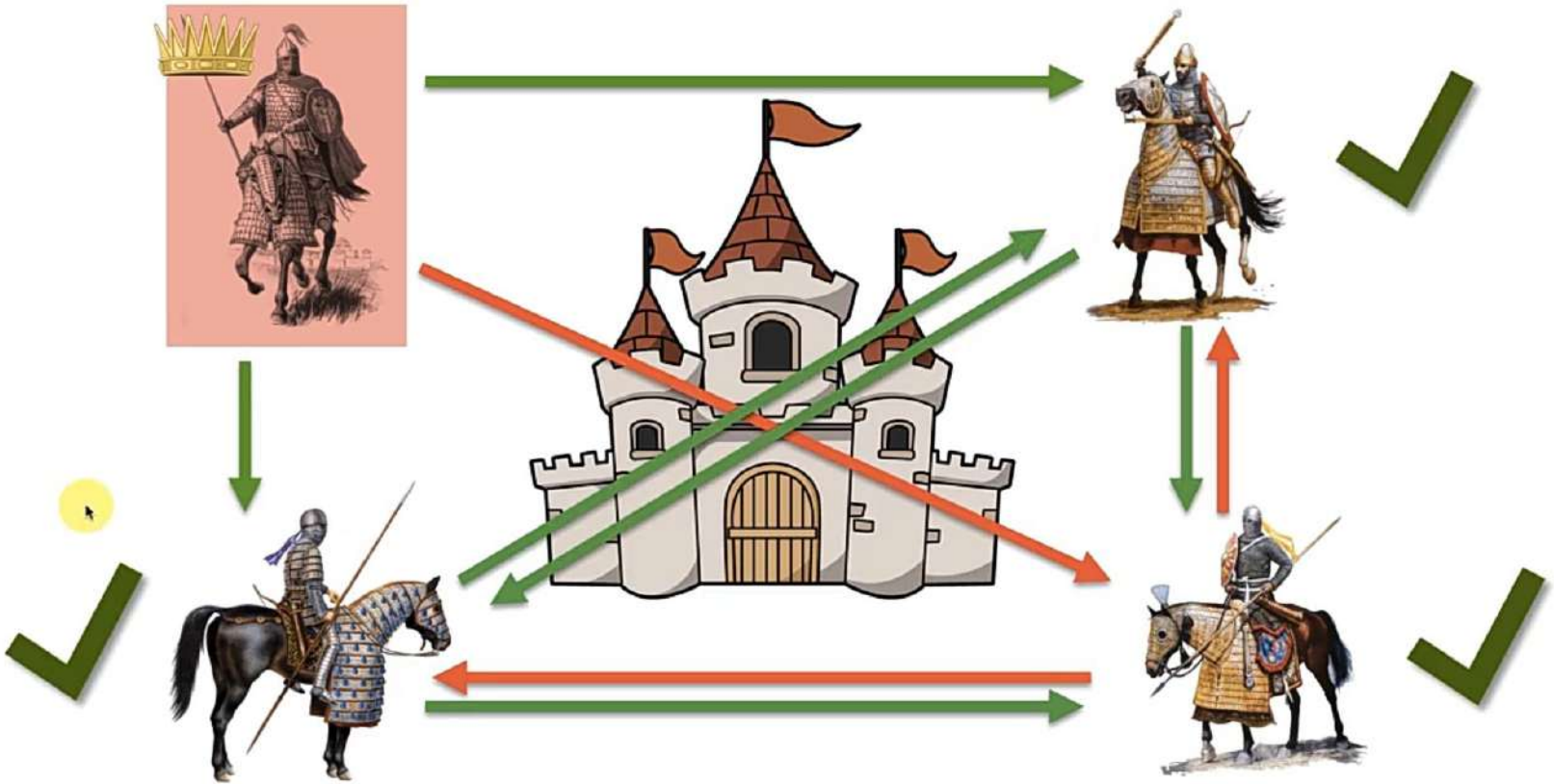
Byzantine Fault Tolerance



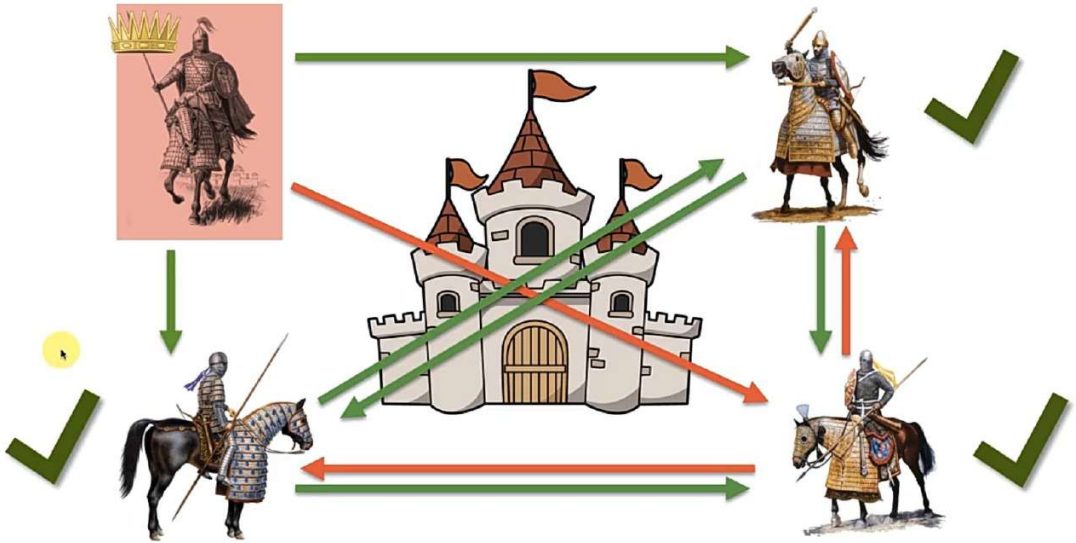
Byzantine Fault Tolerance



Byzantine Fault Tolerance

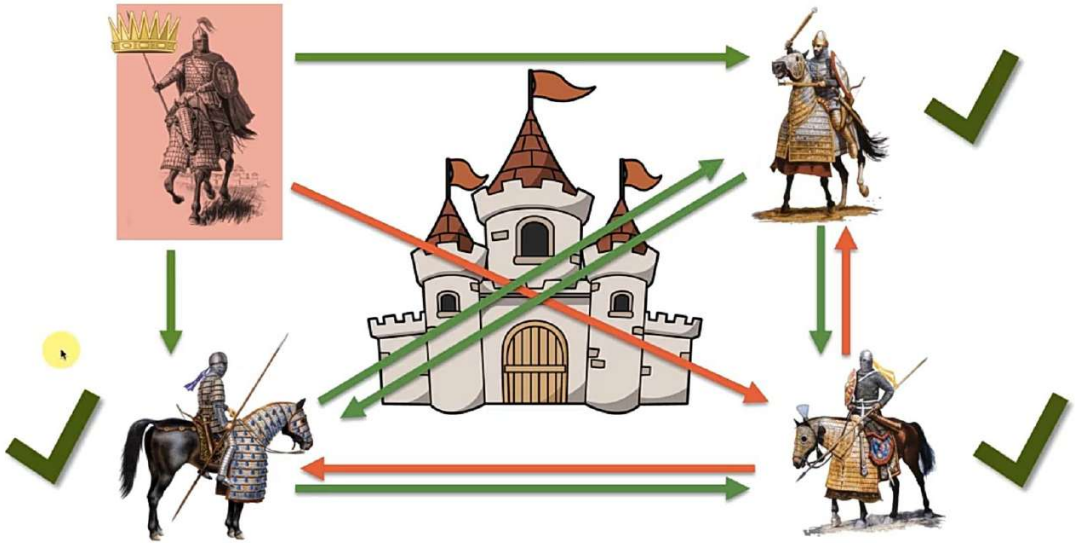


Byzantine Fault Tolerance



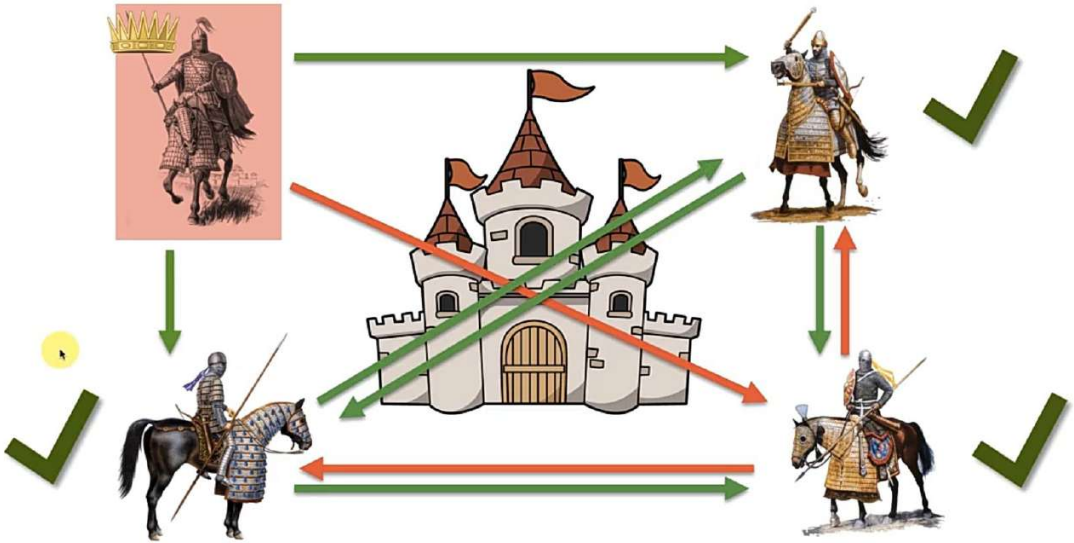
- What is the level of tolerance ?
- What if there are 2 traitors in this network ?

Byzantine Fault Tolerance



- What is the level of tolerance ?
- What if there are 2 traitors in this network ?
- **Not more than** ⅓ in the Army can be traitors.

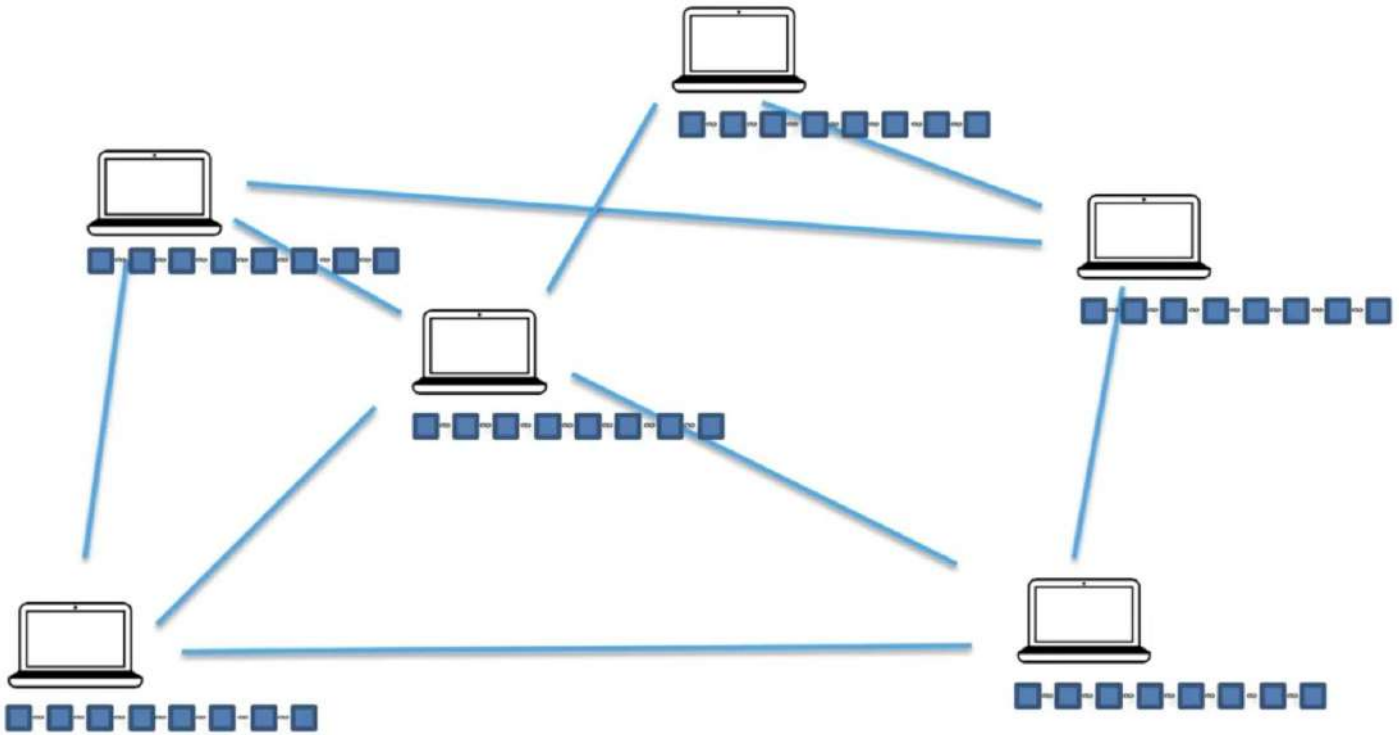
Byzantine Fault Tolerance



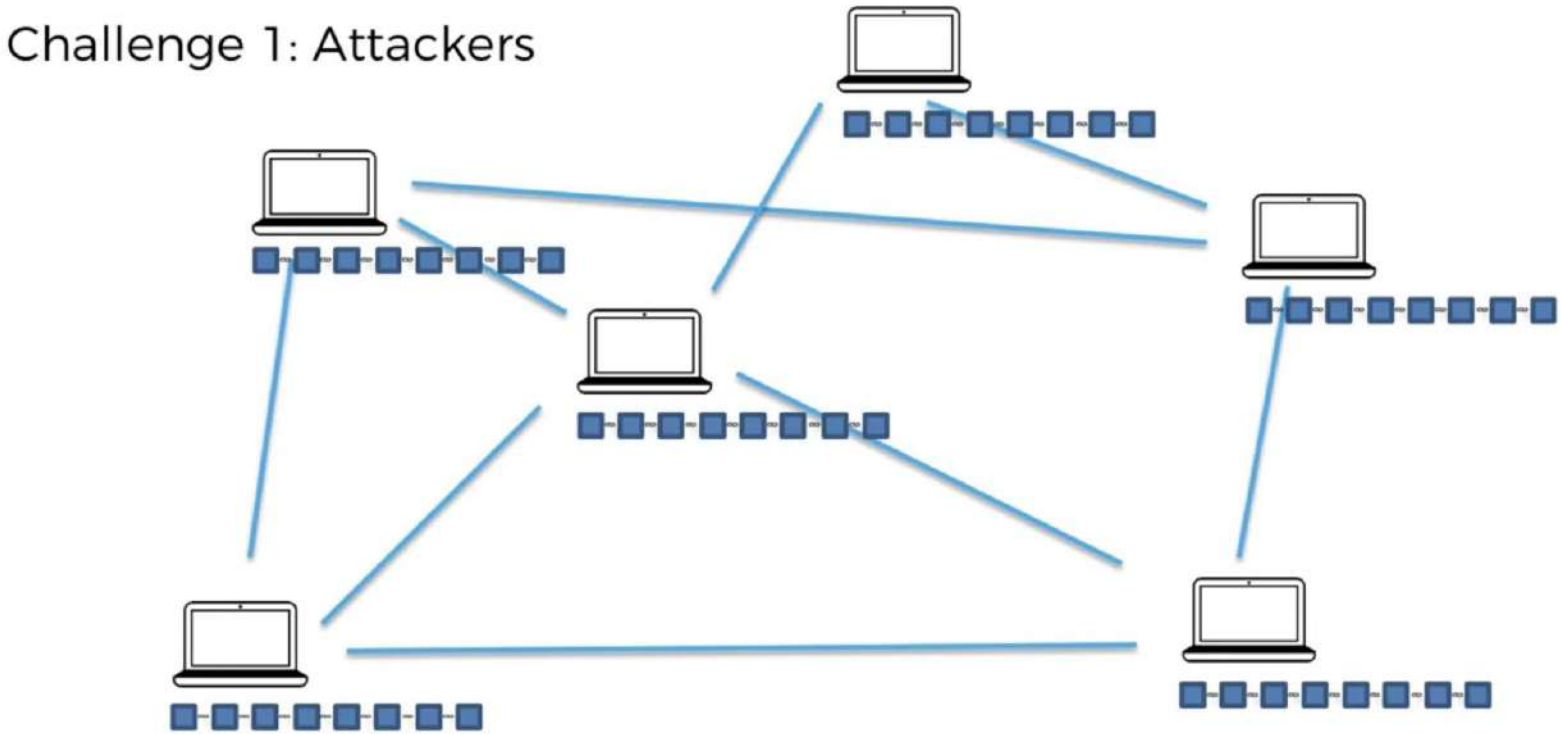
Applications of BFT

- Blockchain
- Aeroplane Circuits
- Nuclear Power Plants
- Rockets, etc …

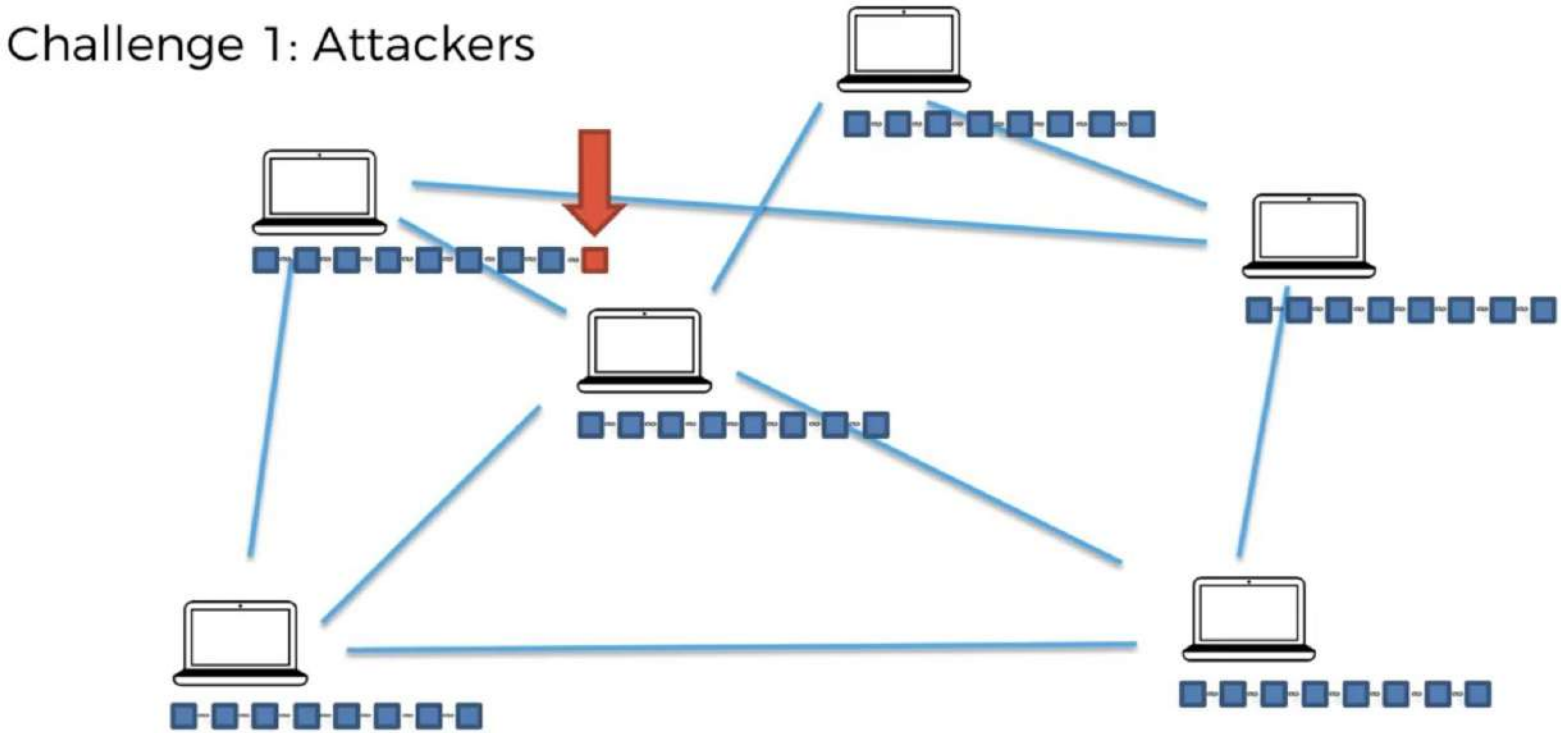
Challenges Addressed by Consensus Protocol



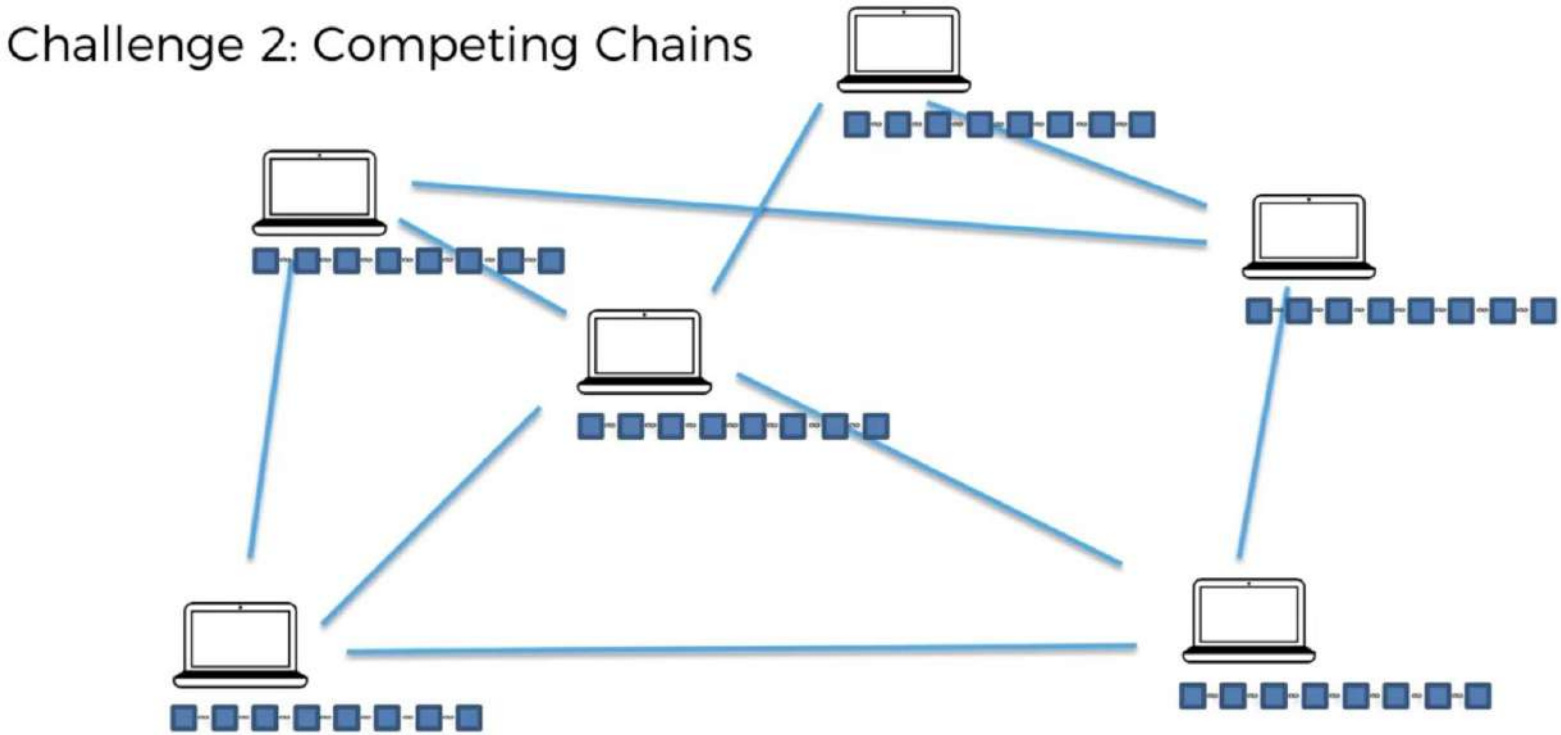
Challenges Addressed by Consensus Protocol



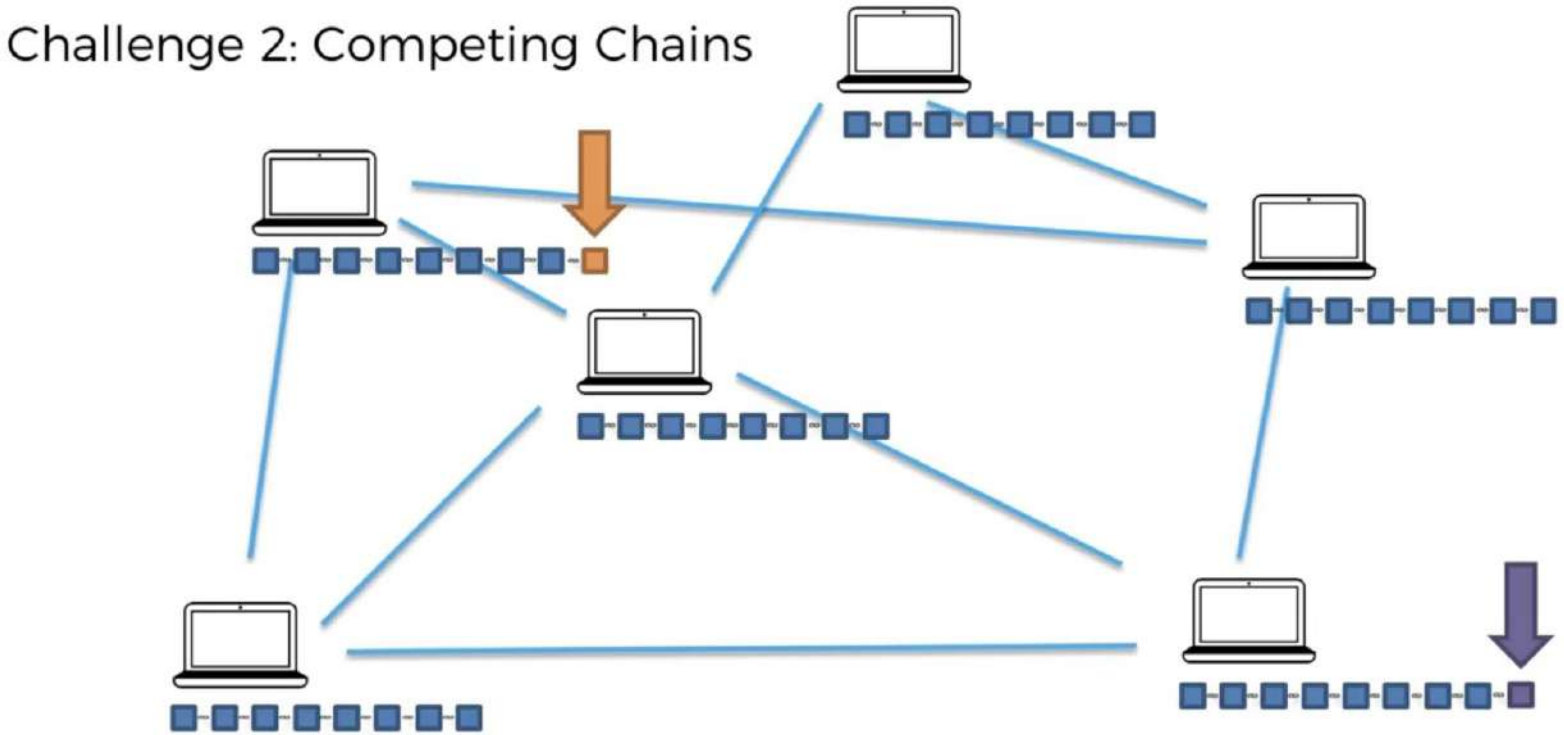
Challenges Addressed by Consensus Protocol



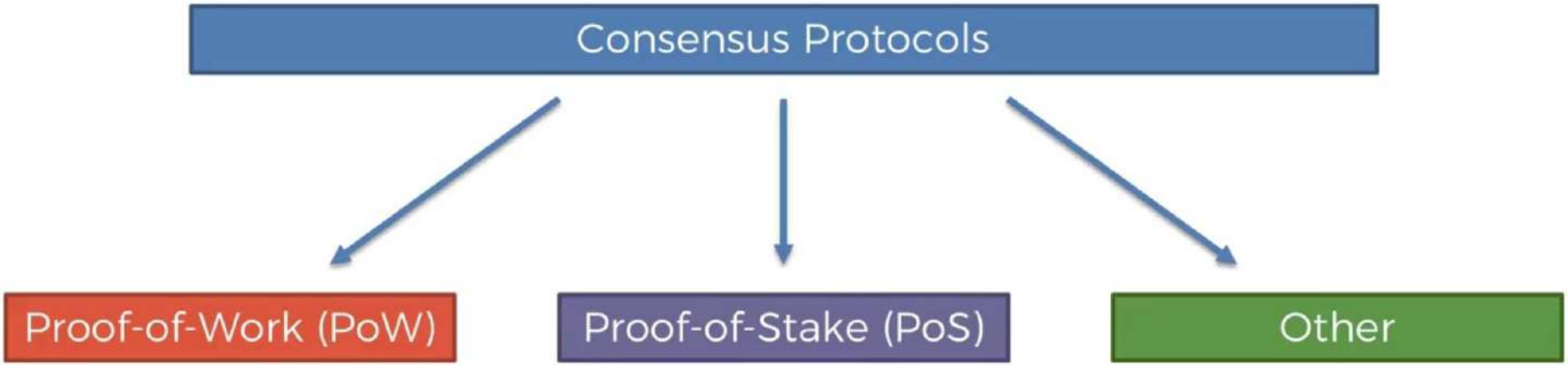
Challenges Addressed by Consensus Protocol



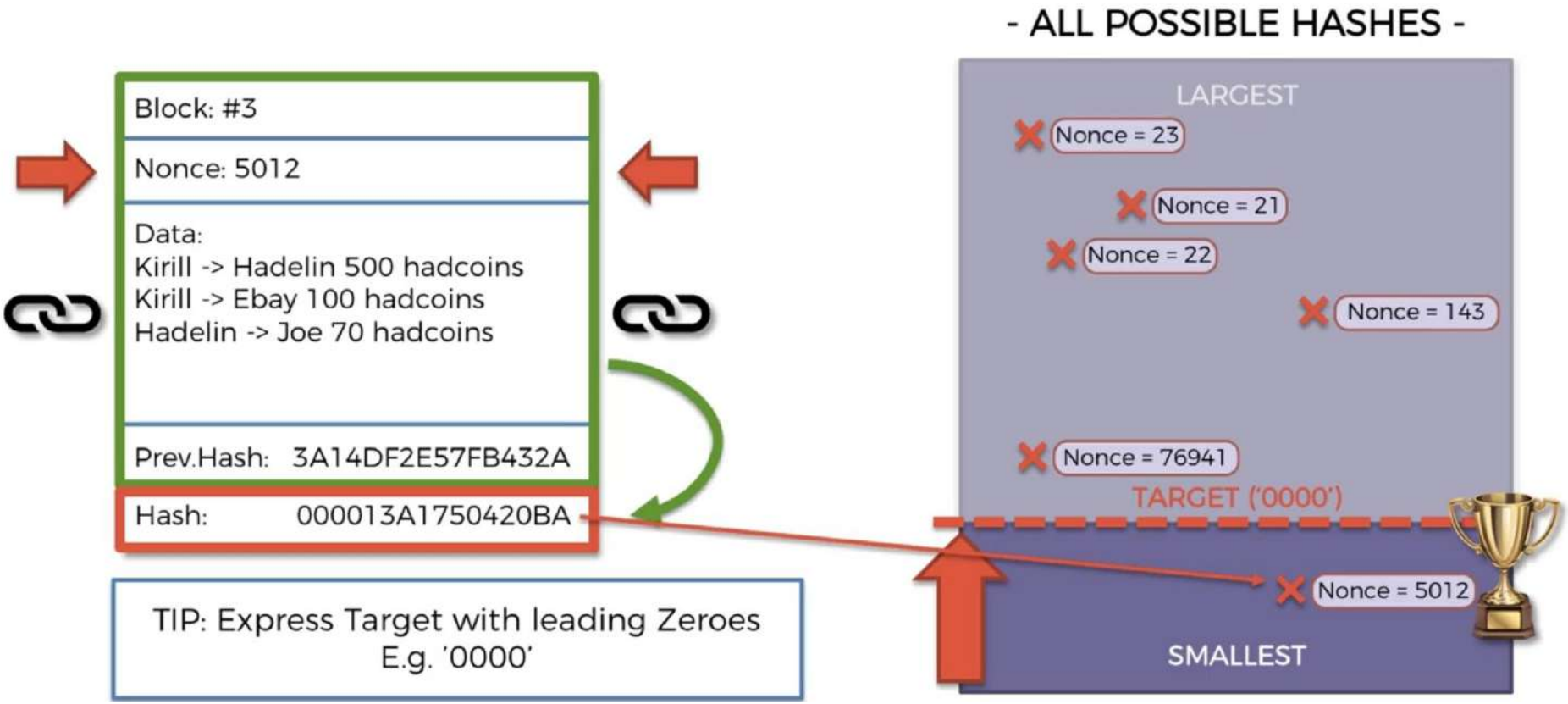
Challenges Addressed by Consensus Protocol



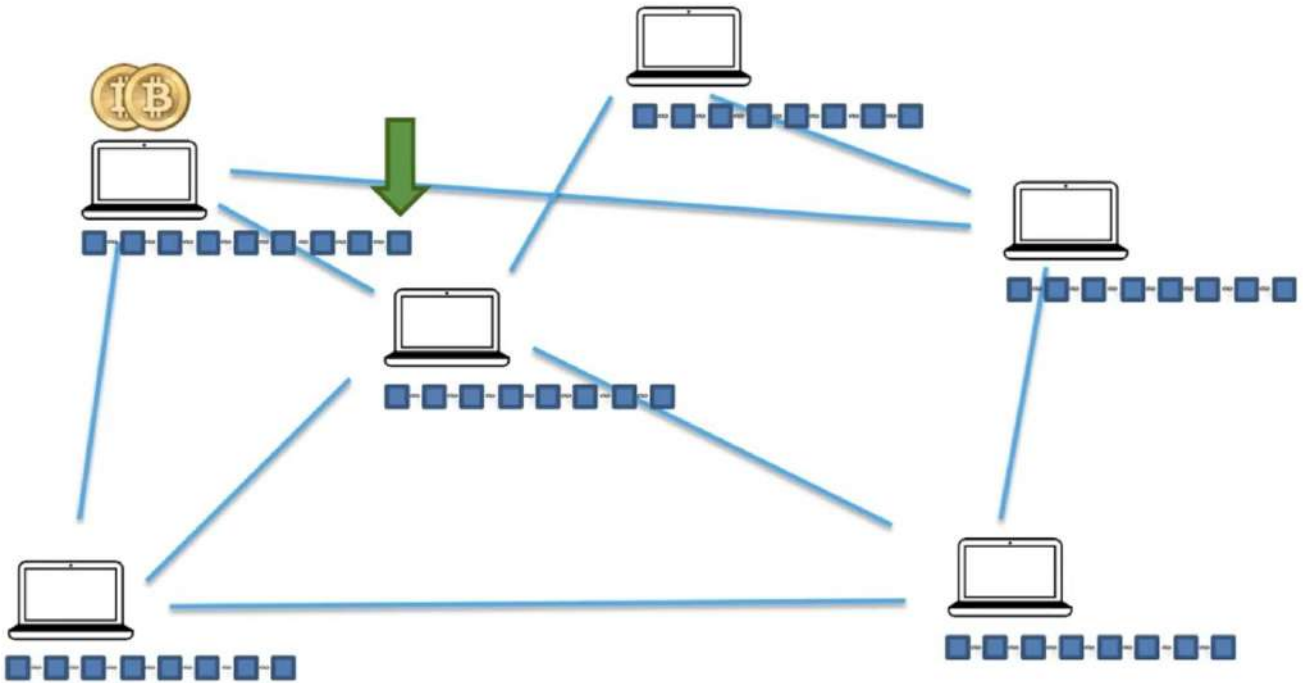
Consensus Protocol



Consensus Protocol - Cryptographic Challenge



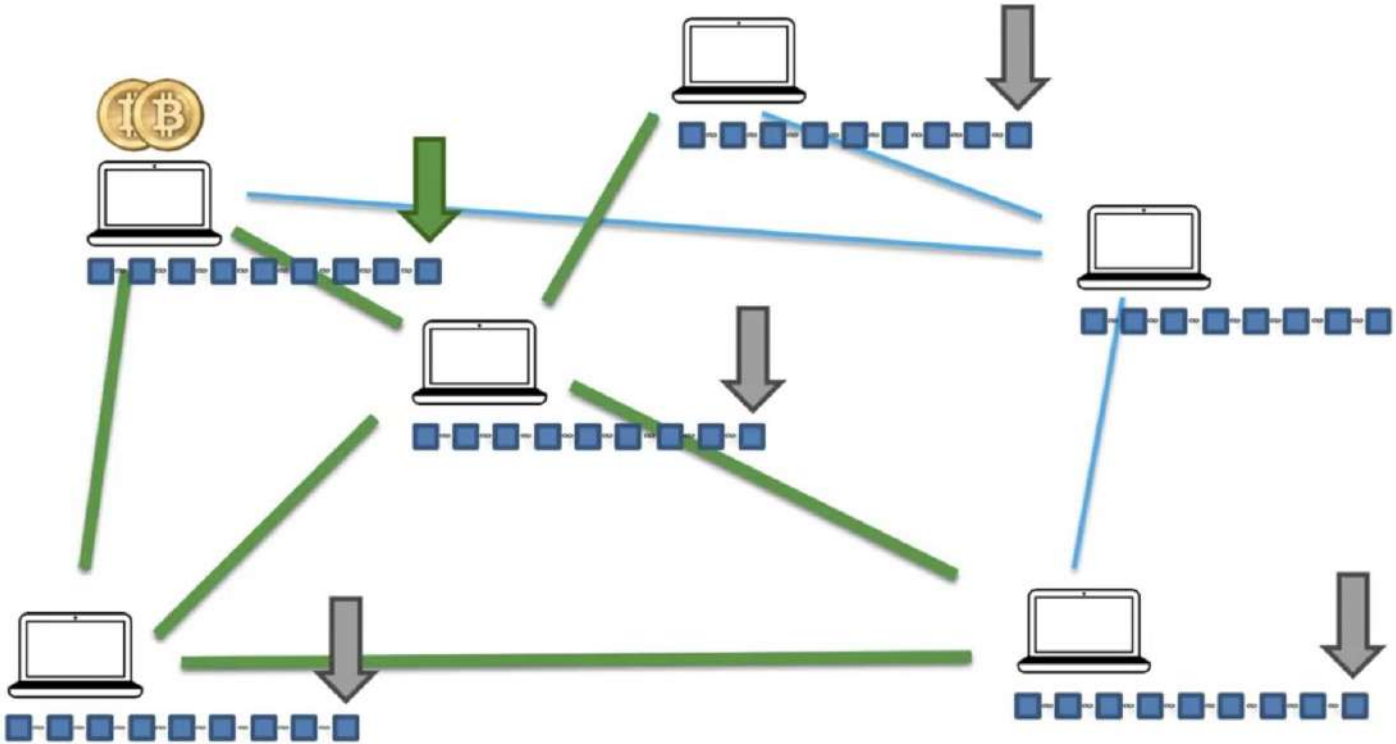
Consensus Protocol - Attackers Challenge



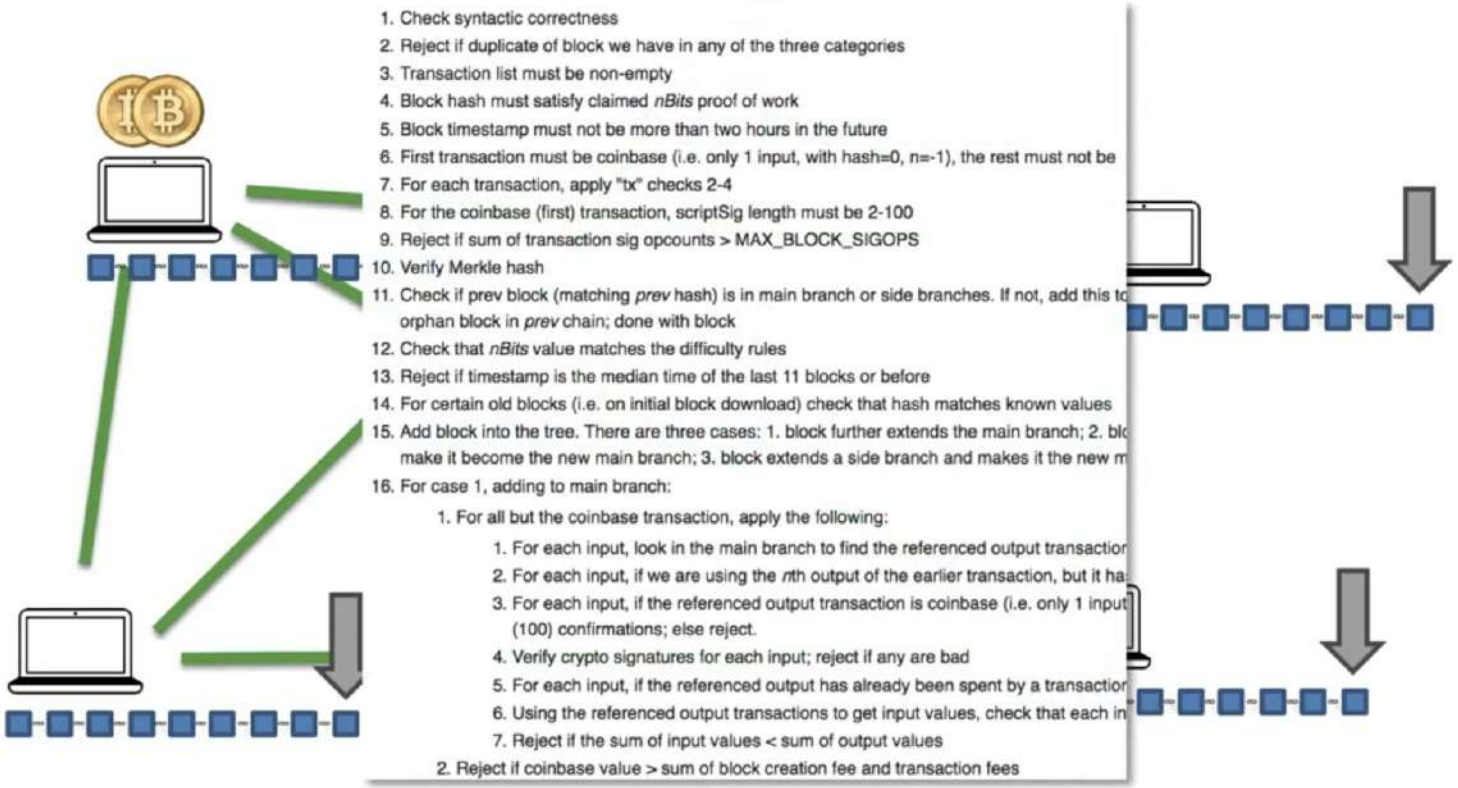
Miners get incentives for :

1. **Adding a block**
2. **To play fair**
3. **From the transaction fees**

Consensus Protocol - Attackers Challenge

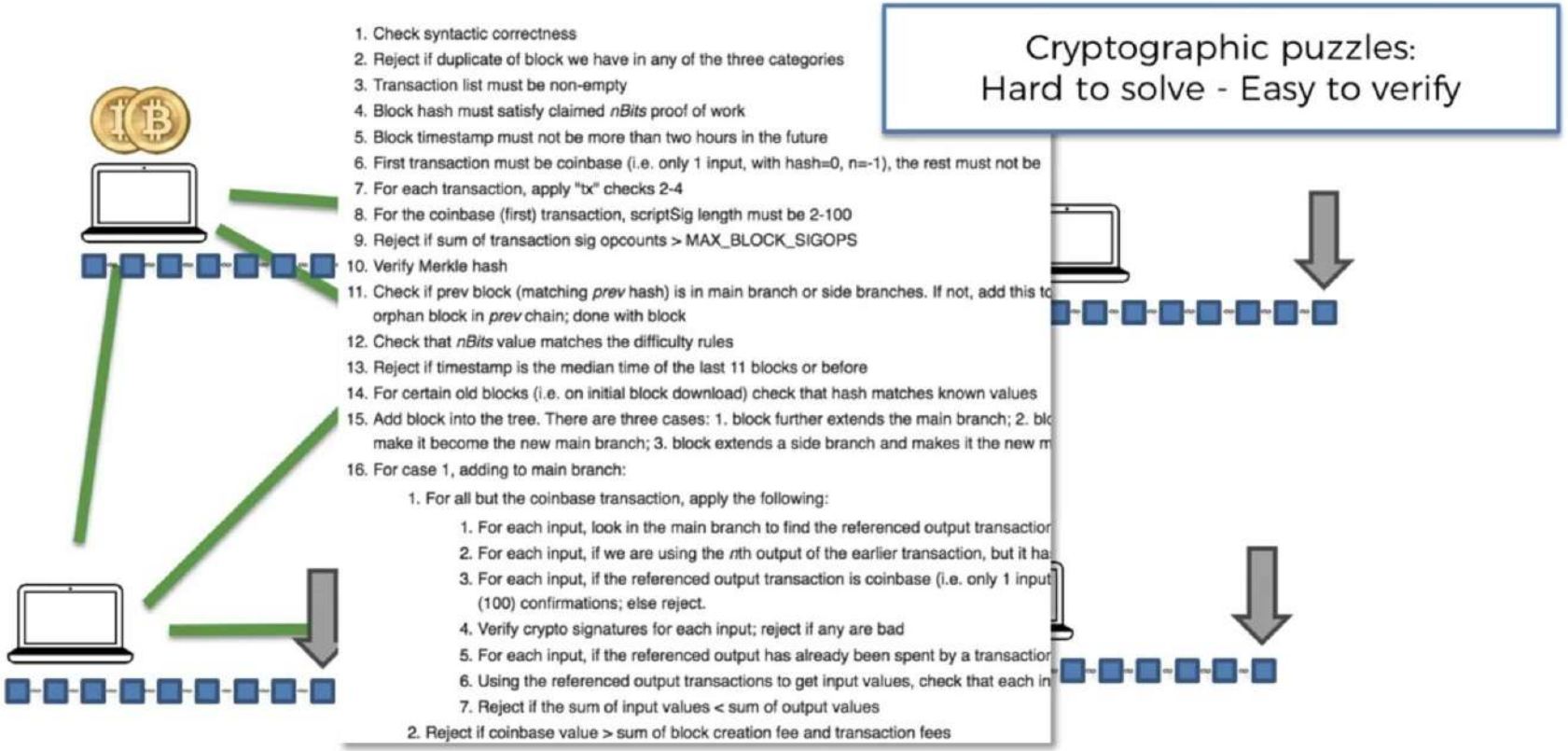


Consensus Protocol - Attackers Challenge

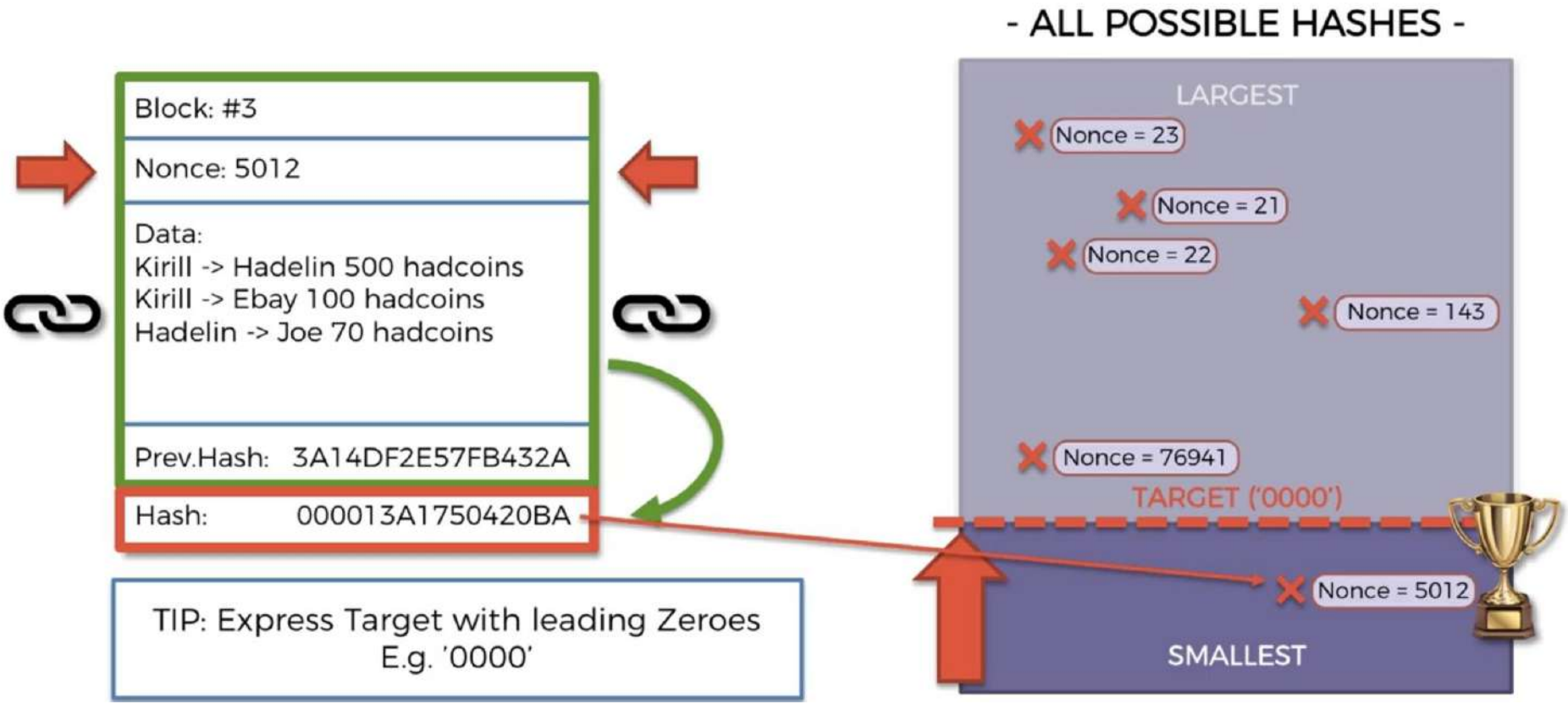


1. Check syntactic correctness
2. Reject if duplicate of block we have in any of the three categories
3. Transaction list must be non-empty
4. Block hash must satisfy claimed *nBits* proof of work
5. Block timestamp must not be more than two hours in the future
6. First transaction must be coinbase (i.e. only 1 input, with hash=0, n=-1), the rest must not be
7. For each transaction, apply "tx" checks 2-4
8. For the coinbase (first) transaction, scriptSig length must be 2-100
9. Reject if sum of transaction sig opcounts > MAX_BLOCK_SIGOPS
10. Verify Merkle hash
11. Check if prev block (matching *prev* hash) is in main branch or side branches. If not, add this to orphan block in *prev* chain; done with block
12. Check that *nBits* value matches the difficulty rules
13. Reject if timestamp is the median time of the last 11 blocks or before
14. For certain old blocks (i.e. on initial block download) check that hash matches known values
15. Add block into the tree. There are three cases: 1. block further extends the main branch; 2. bl make it become the new main branch; 3. block extends a side branch and makes it the new m
16. For case 1, adding to main branch:
    1. For all but the coinbase transaction, apply the following:
        1. For each input, look in the main branch to find the referenced output transactio
        2. For each input, if we are using the *n*th output of the earlier transaction, but it ha
        3. For each input, if the referenced output transaction is coinbase (i.e. only 1 input (100) confirmations; else reject.
        4. Verify crypto signatures for each input; reject if any are bad
        5. For each input, if the referenced output has already been spent by a transactio
        6. Using the referenced output transactions to get input values, check that each in
        7. Reject if the sum of input values < sum of output values
    2. Reject if coinbase value > sum of block creation fee and transaction fees

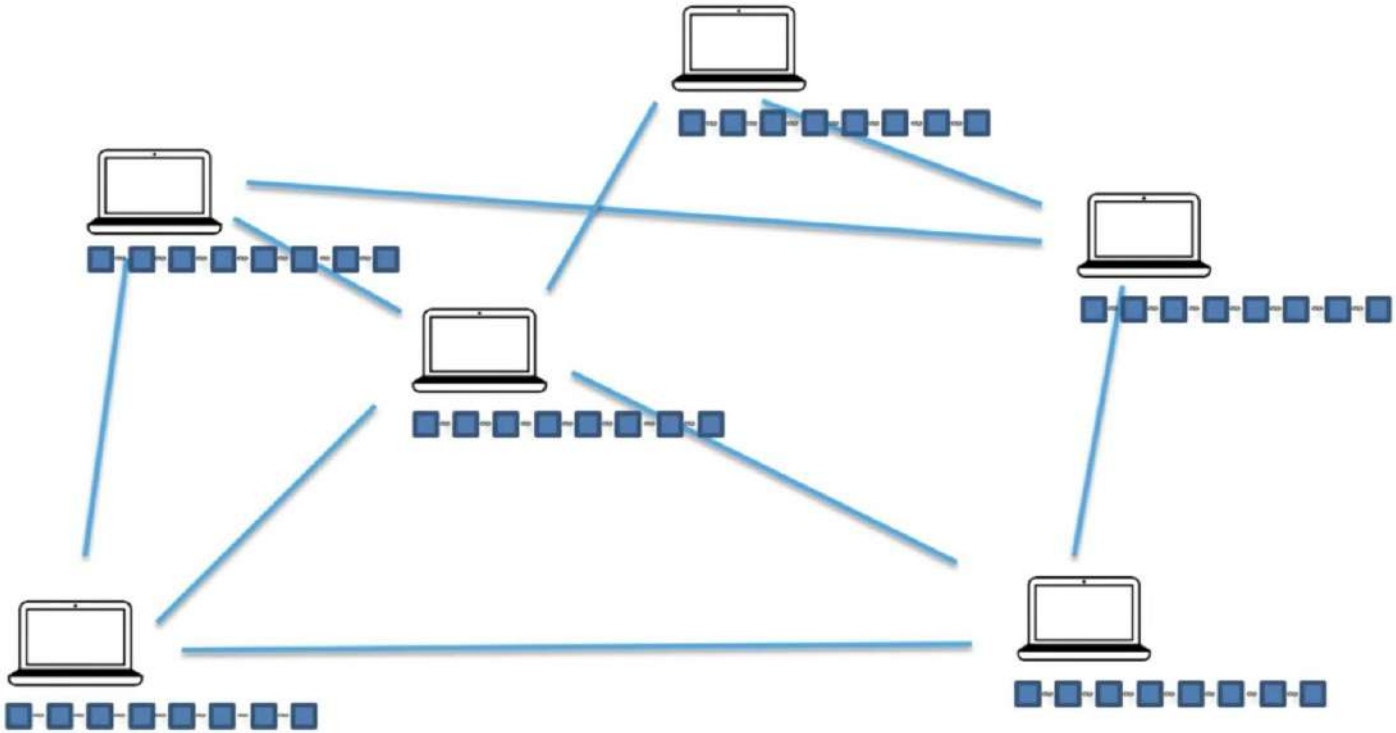
Consensus Protocol - Attackers Challenge



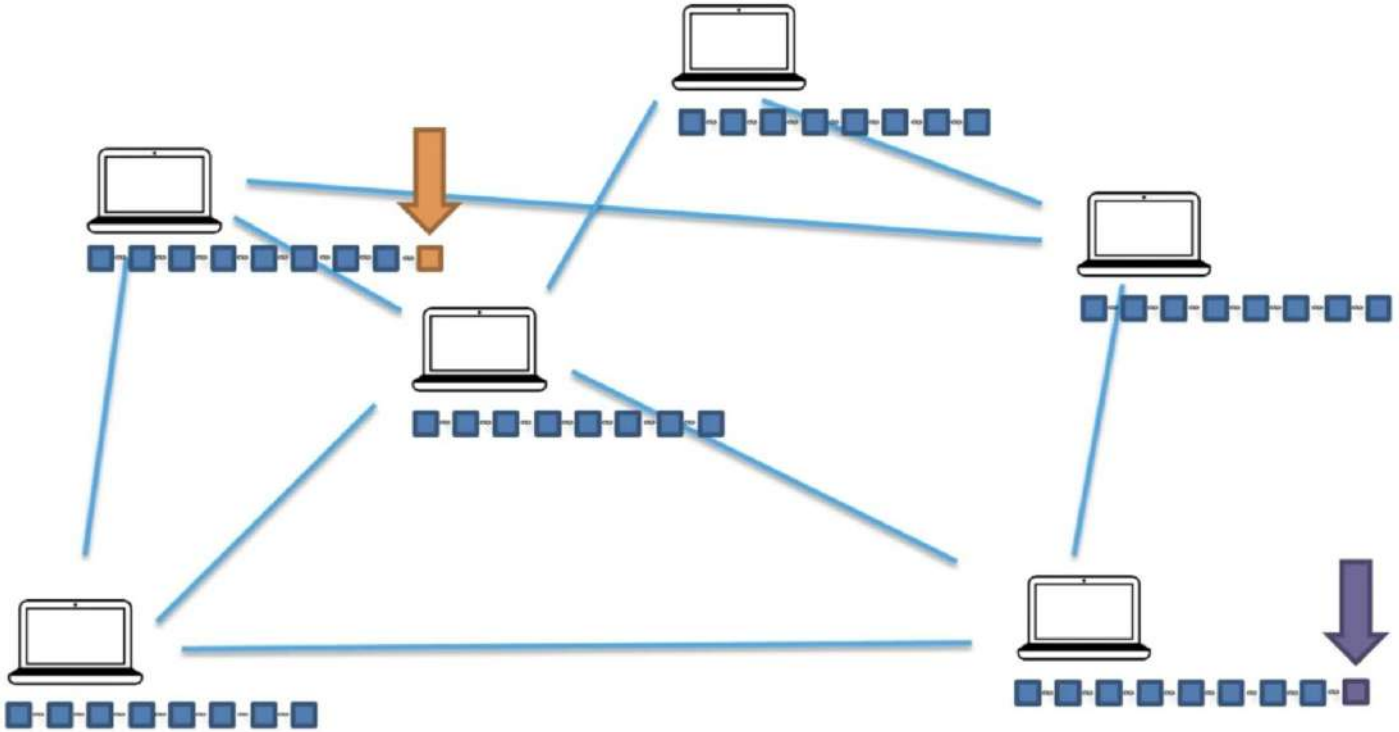
Consensus Protocol - Attackers Challenge



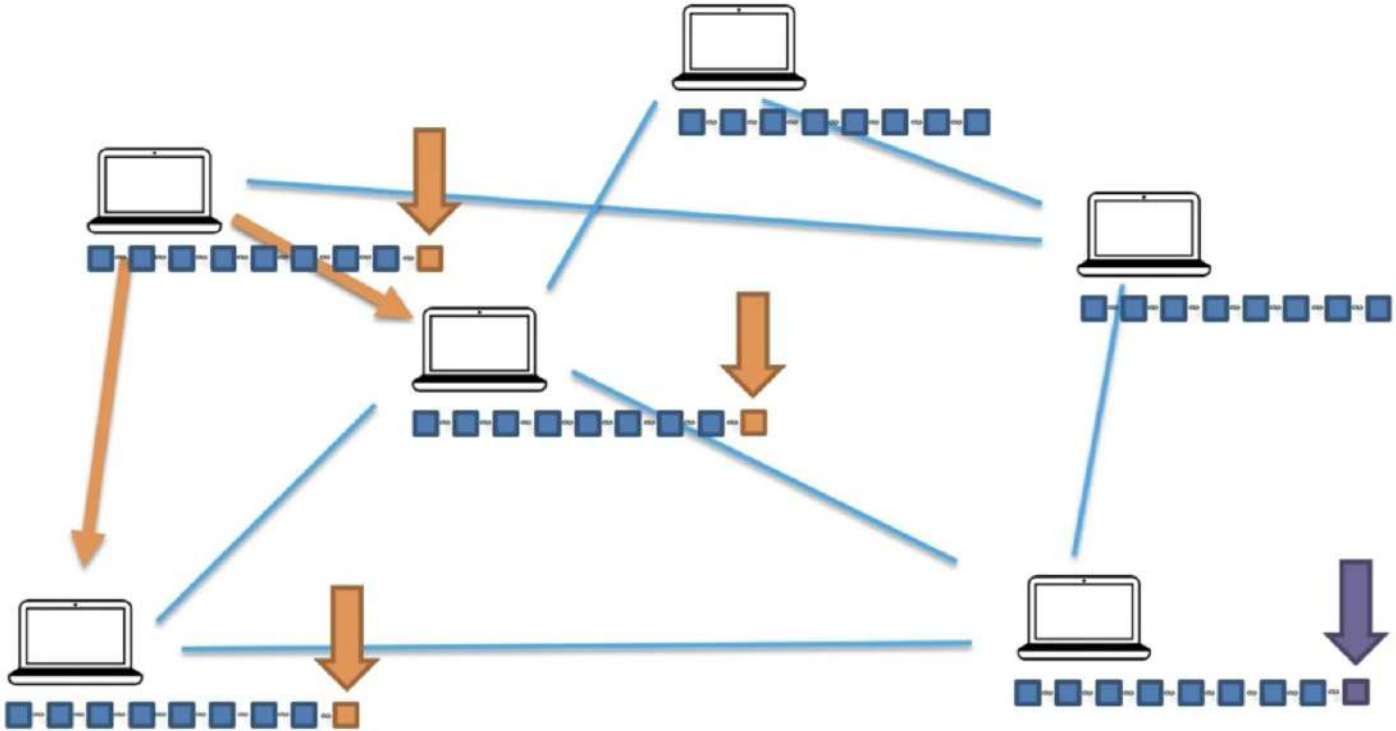
Consensus Protocol - Competing Chains



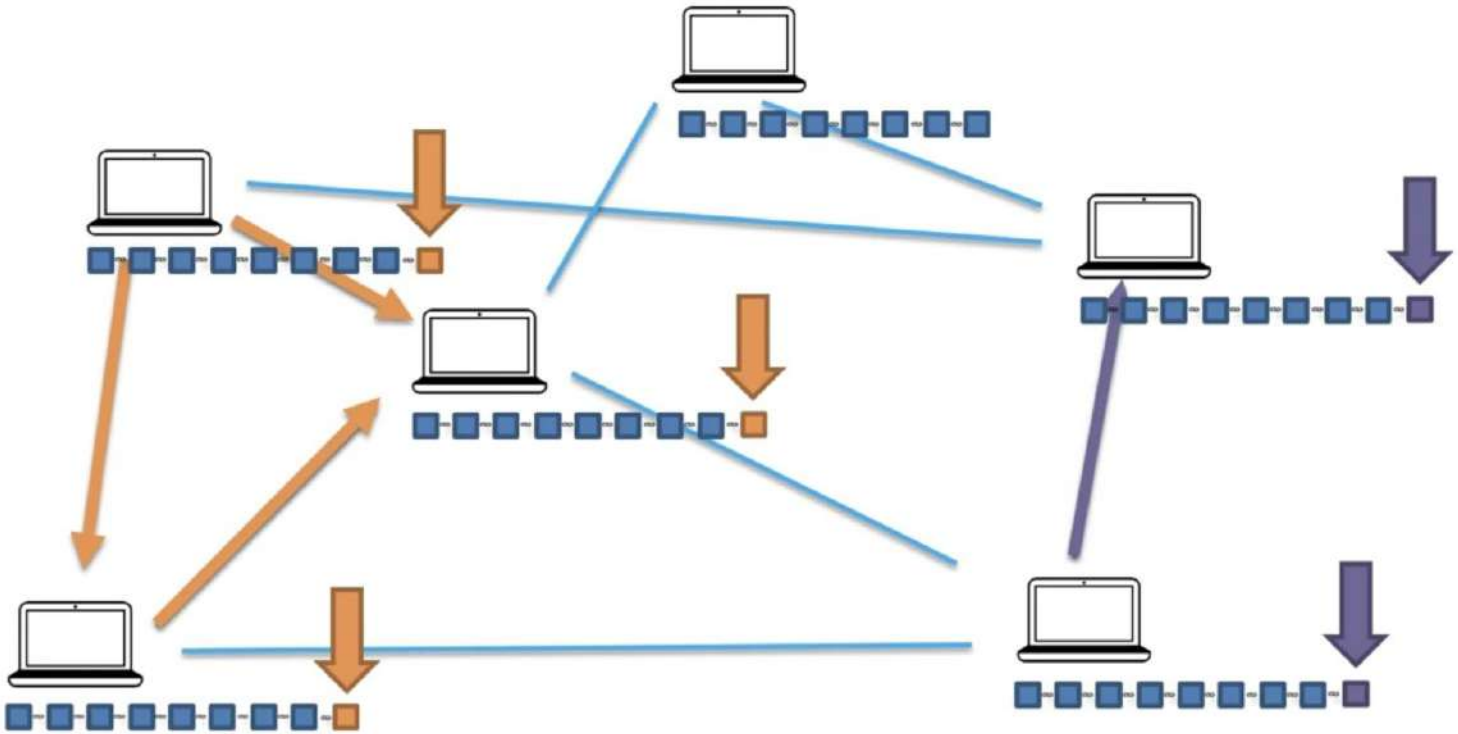
Consensus Protocol - Competing Chains



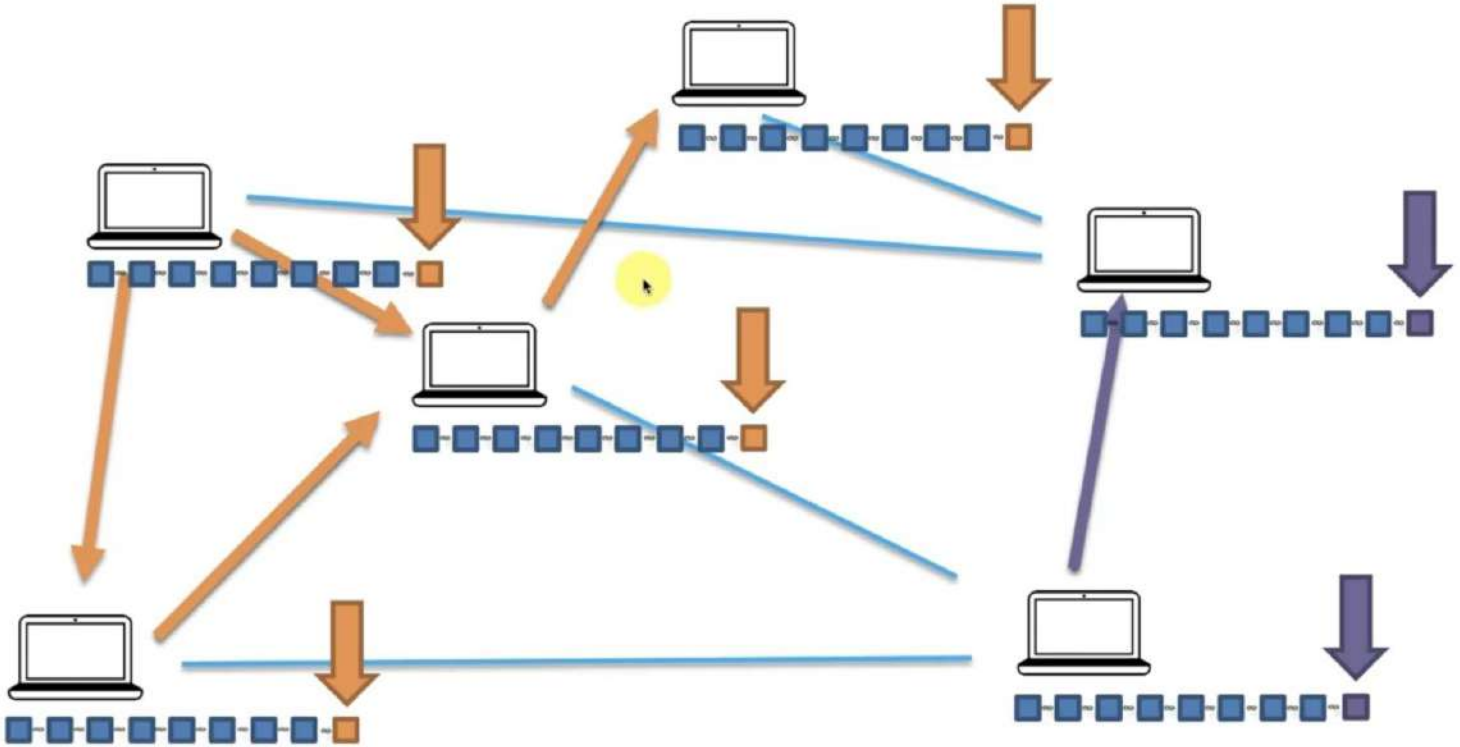
Consensus Protocol - Competing Chains



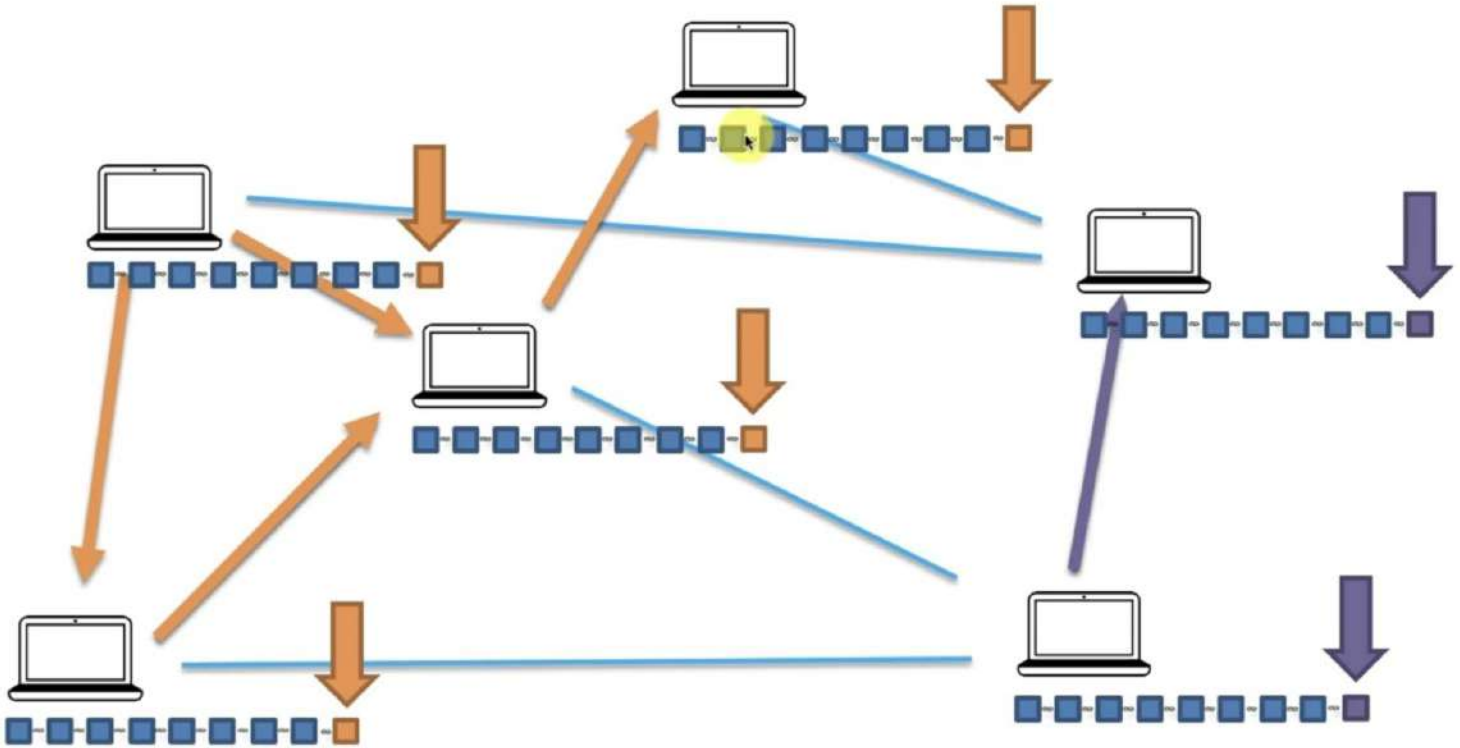
Consensus Protocol - Competing Chains



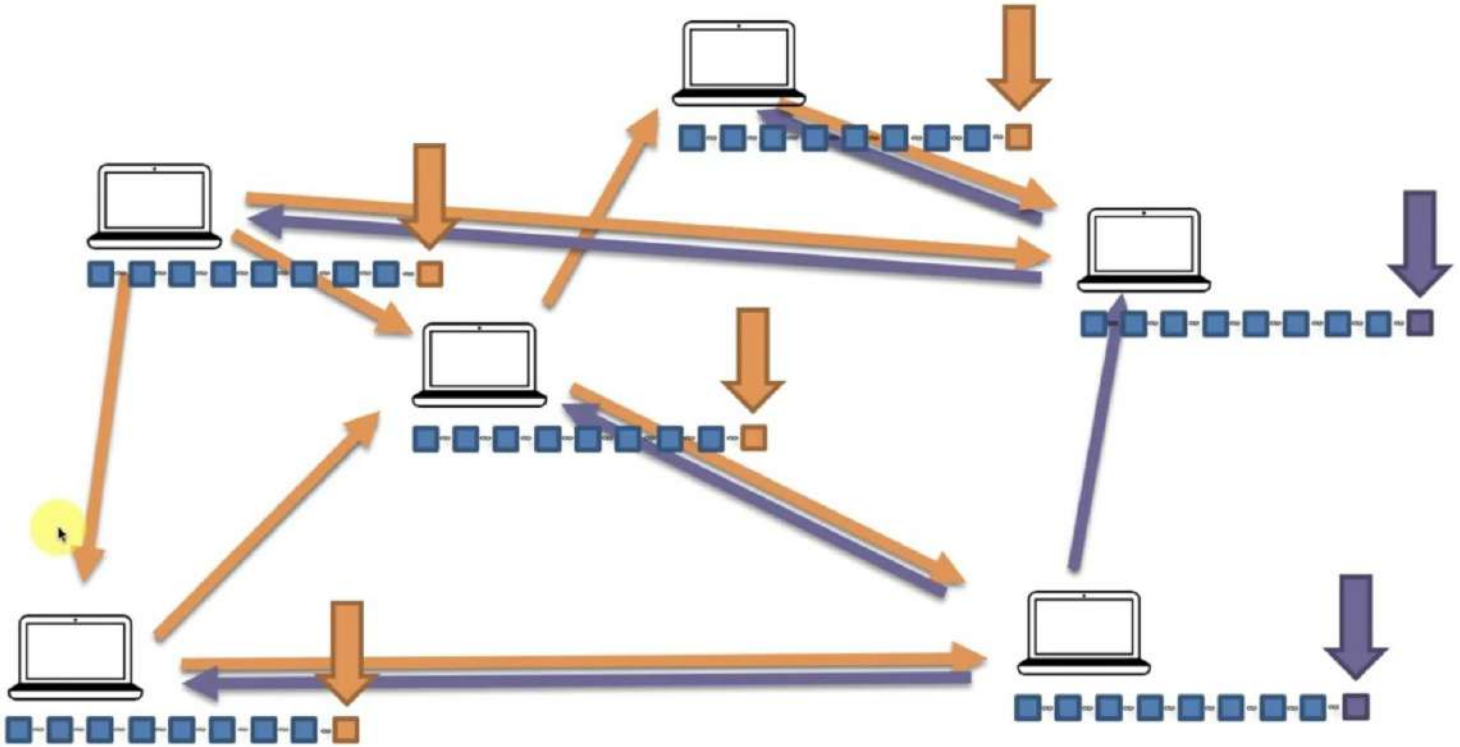
Consensus Protocol - Competing Chains



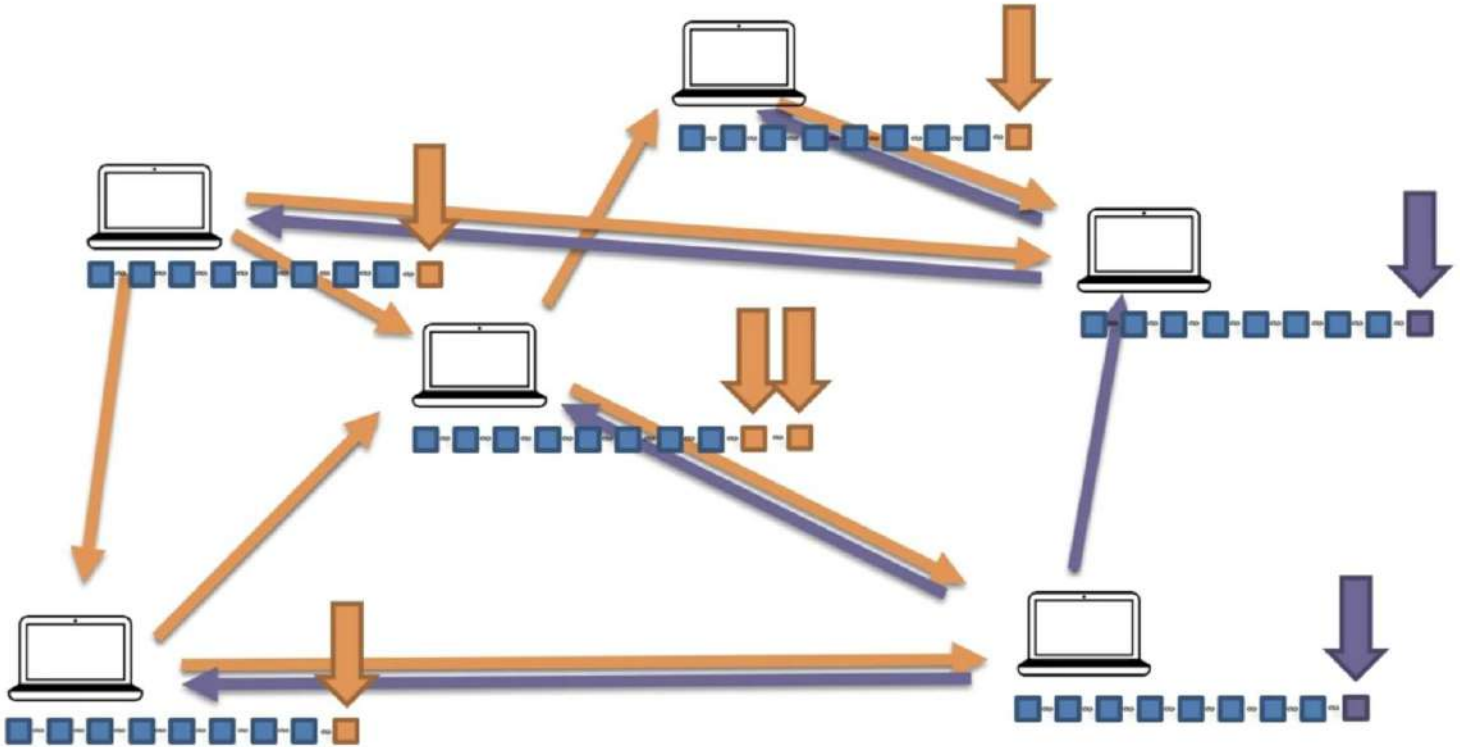
Consensus Protocol - Competing Chains



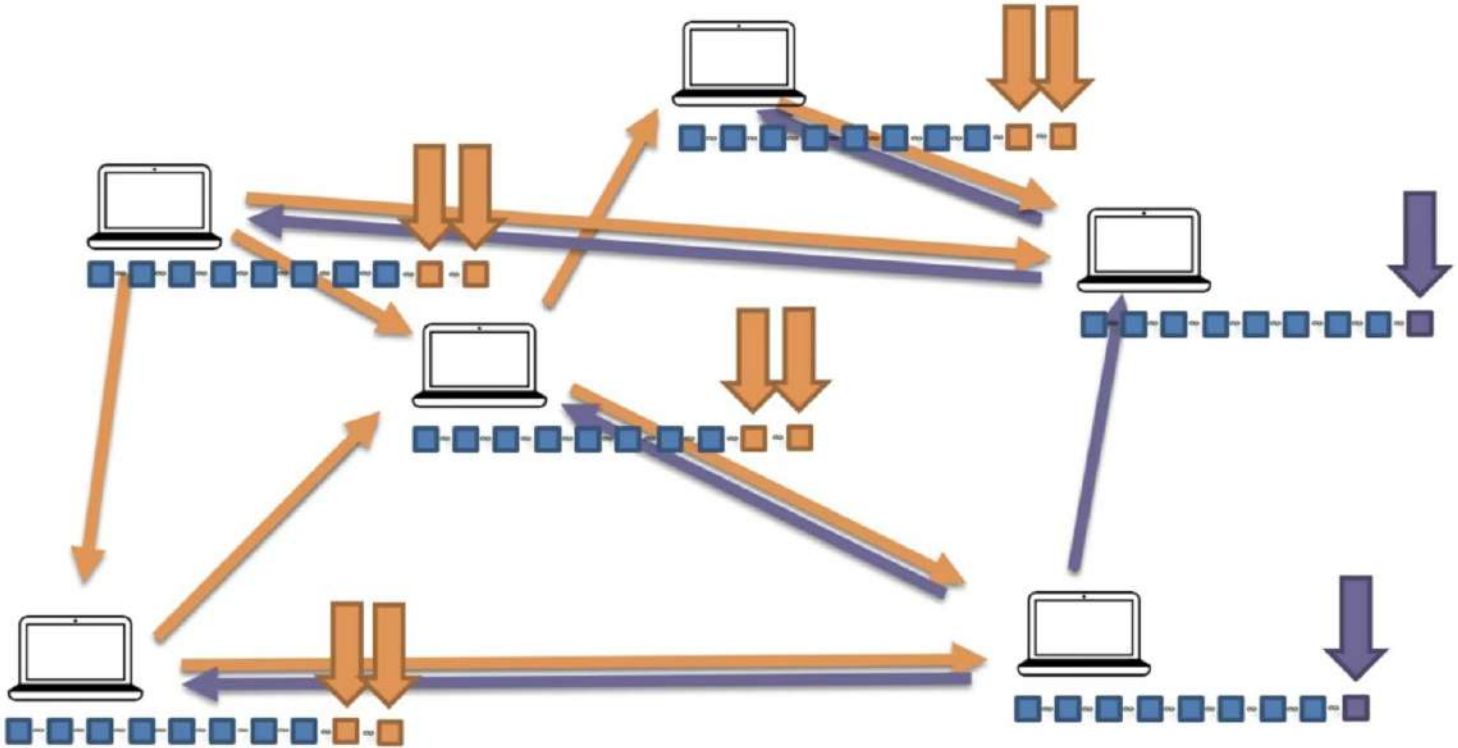
Consensus Protocol - Competing Chains



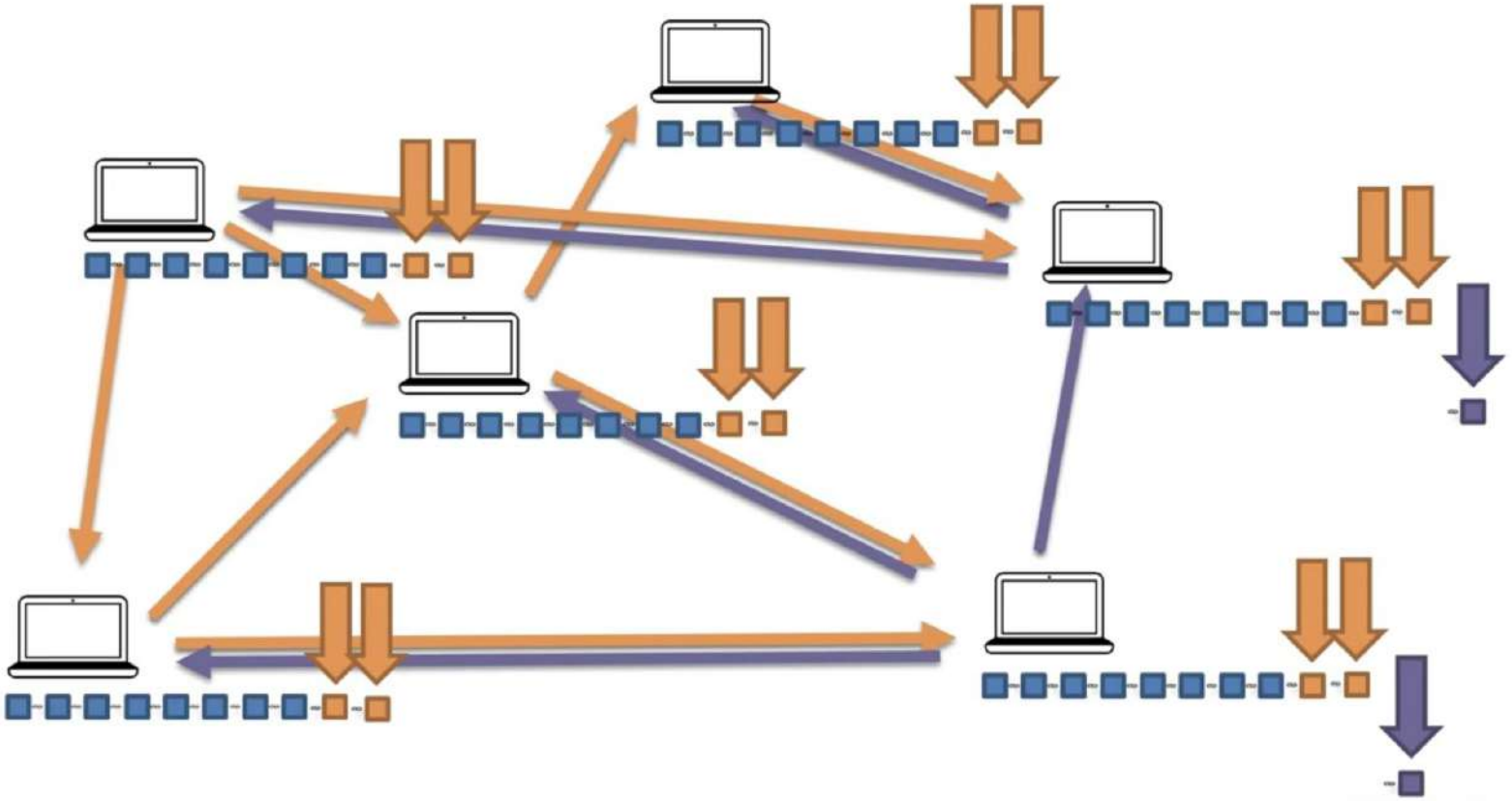
Consensus Protocol - Competing Chains



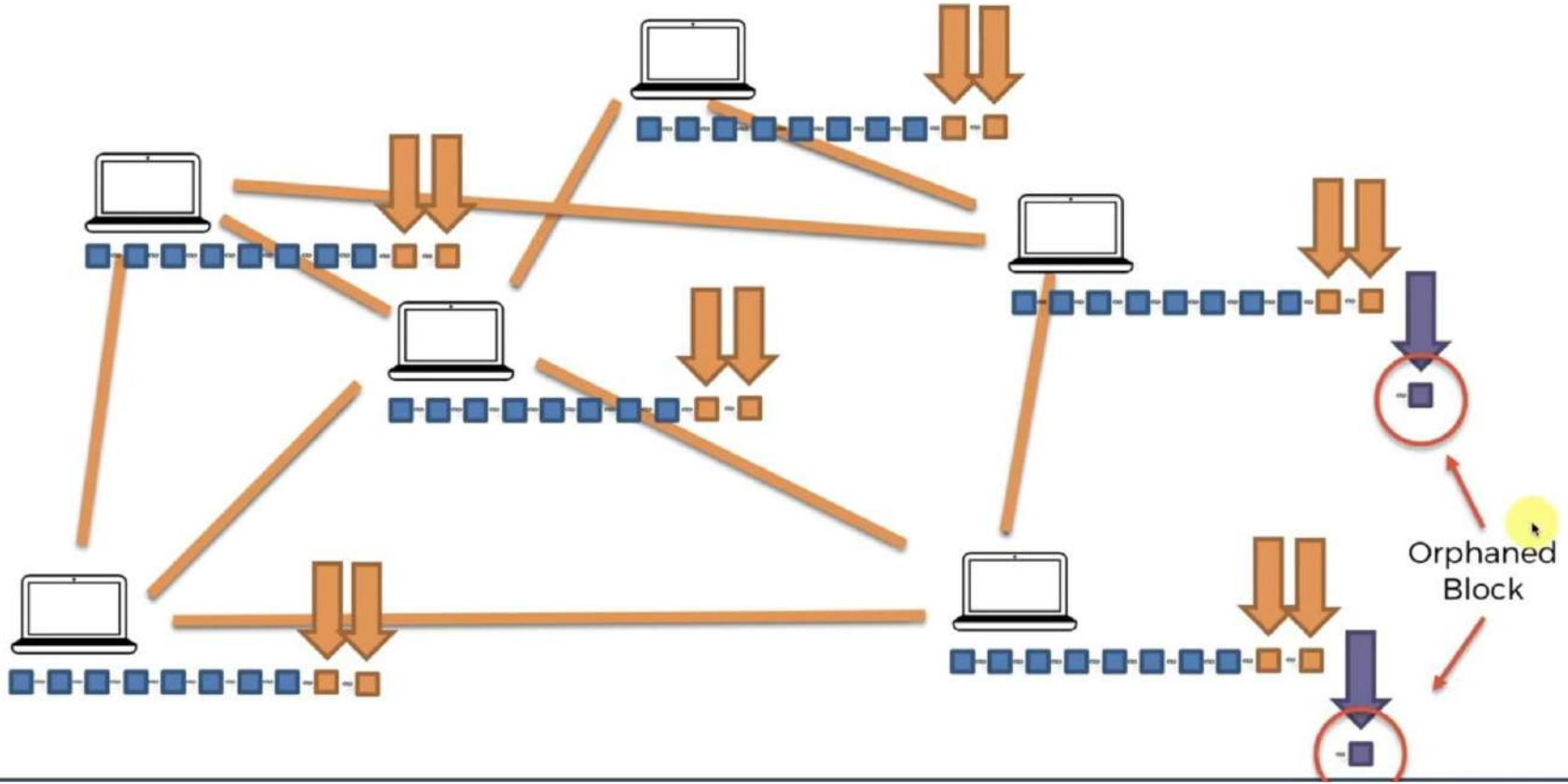
Consensus Protocol - Competing Chains



Consensus Protocol - Competing Chains

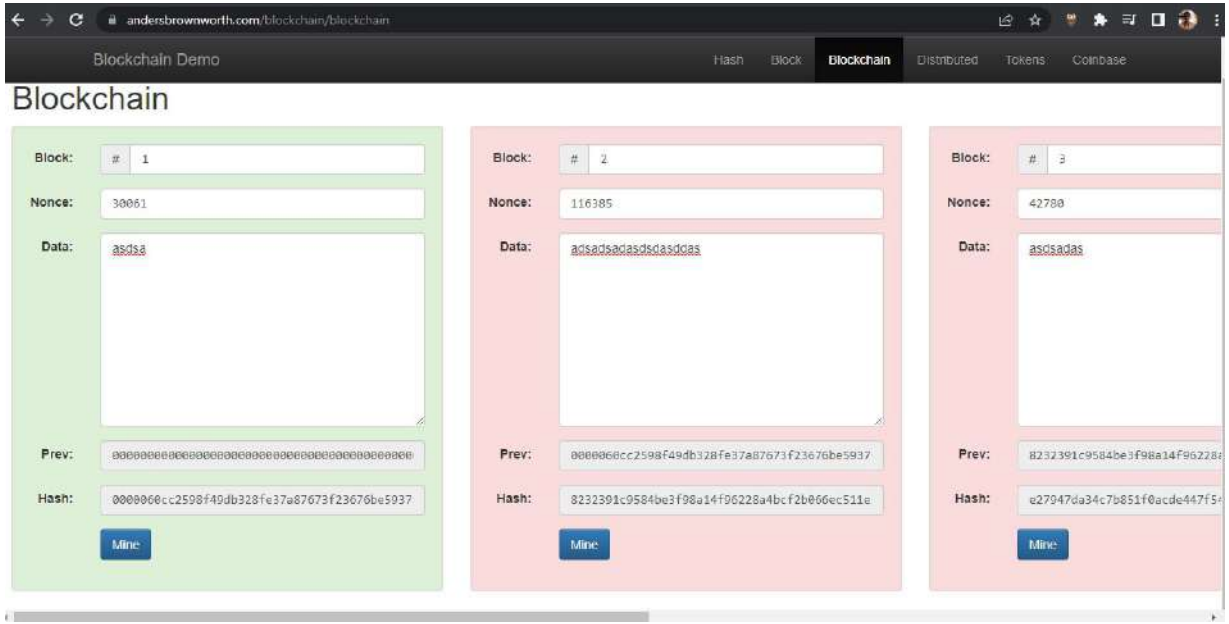


Consensus Protocol - Competing Chains



Blockchain - Demo

Courtesy : https://andersbrownworth.com/blockchain/blockchain



Understanding Mining Difficulty

Q1: What is the Current Target and how does that *feel*?

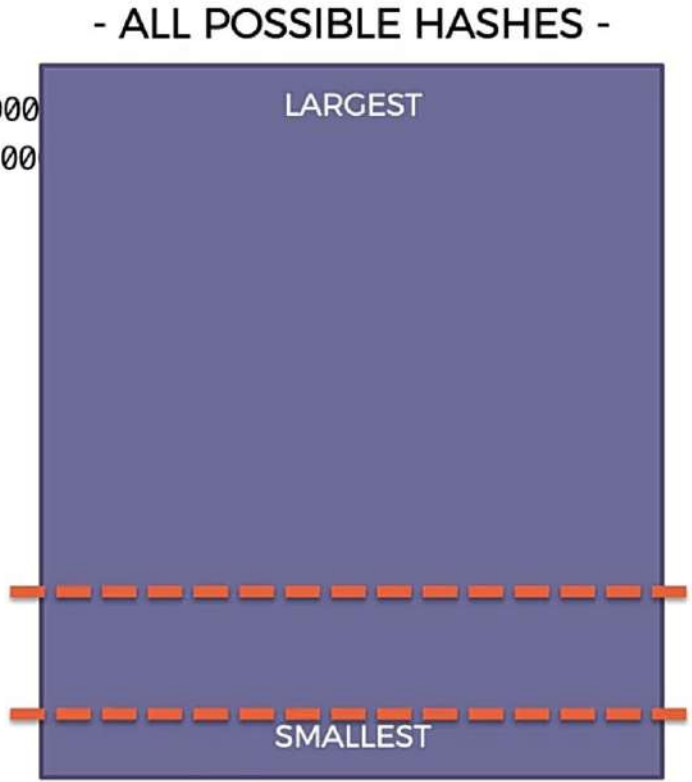
Understanding Mining Difficulty

Difficulty = current target / max target

Curr target = `00000000000000000005d97dc0000000000000000000`

Max target = `00000000FFFF0000000000000000000000000000000000`

Difficulty is adjusted every 2016 blocks (2 weeks)



Understanding Mining Difficulty

Difficulty = current target / max target

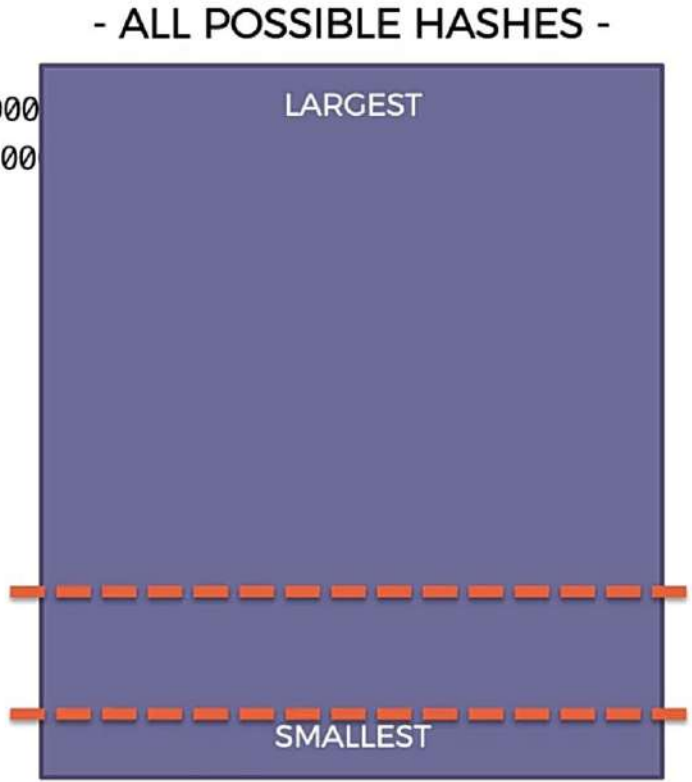
Curr target = 00000000000000000005d97dc0000000000000000000

Max target = 00000000FFFF0000000000000000000000000000000000

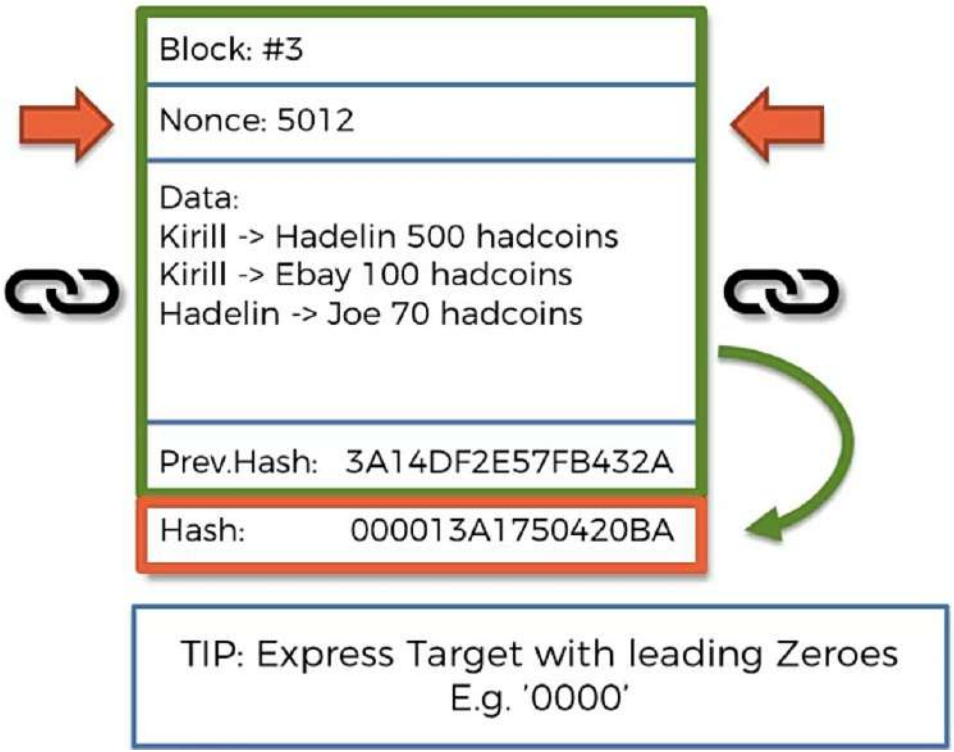
Difficulty is adjusted every 2016 blocks (2 weeks)

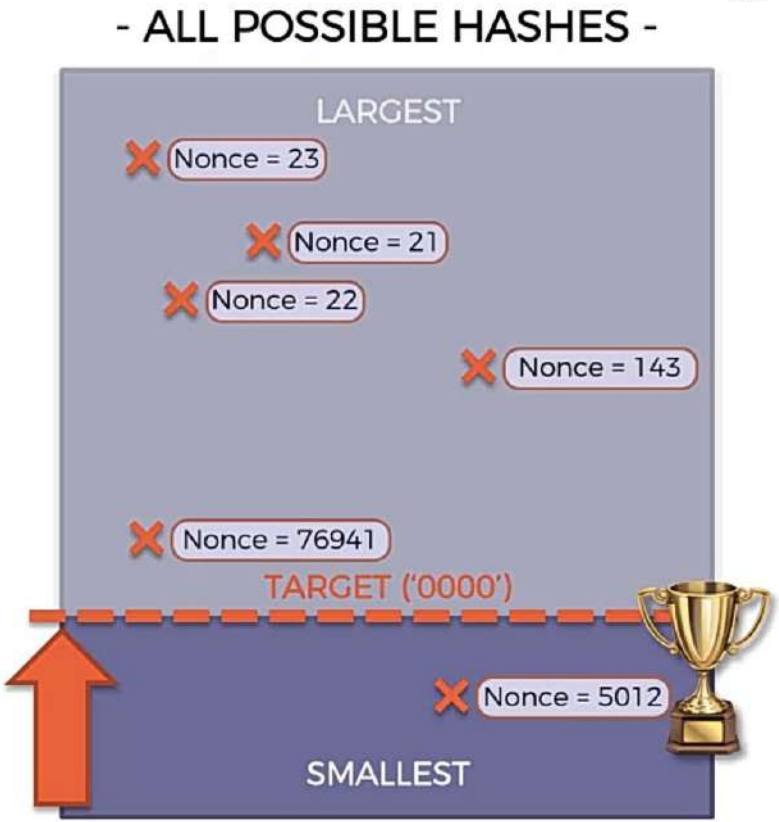
Why 2016 ?

 # Blocks generated in 2 weeks such that one block / 10 minutes

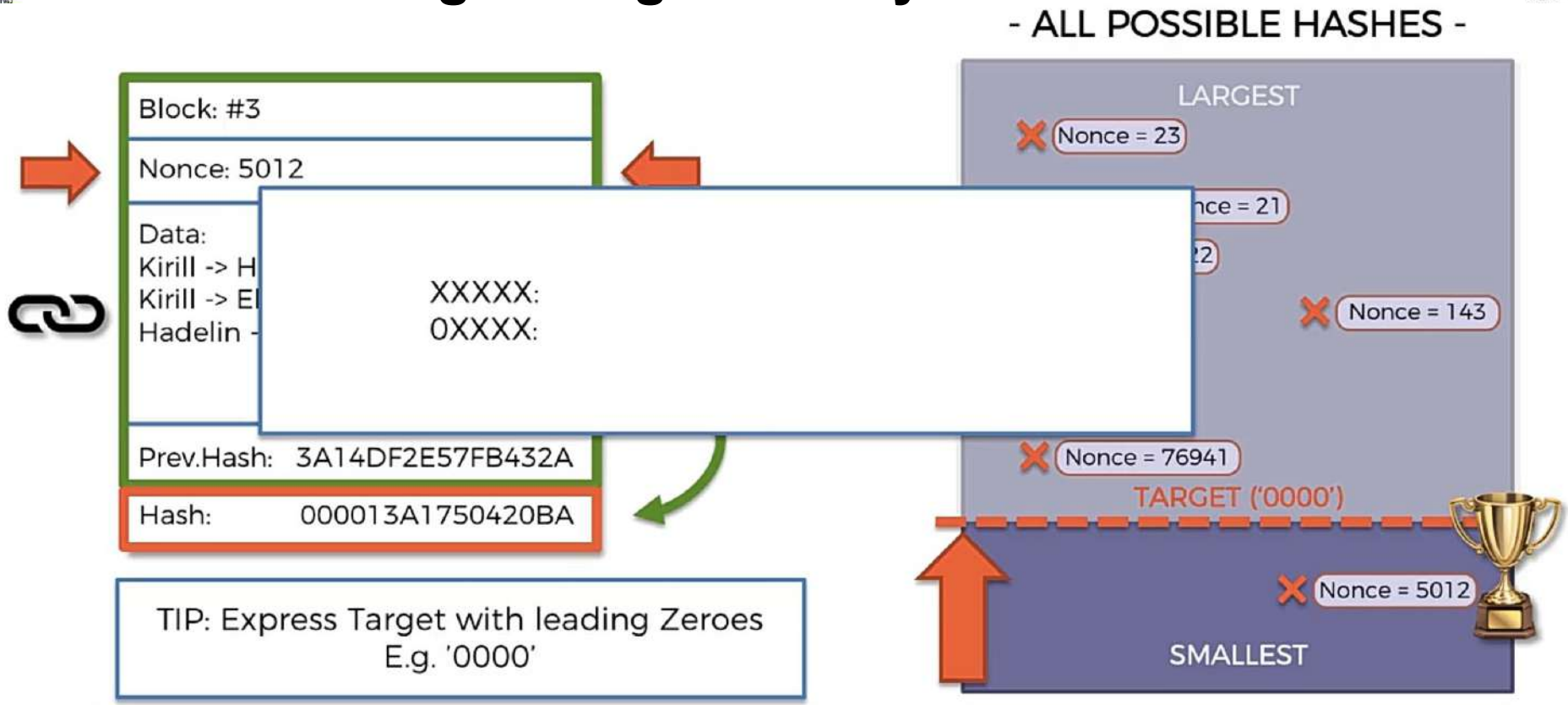


Understanding Mining Difficulty

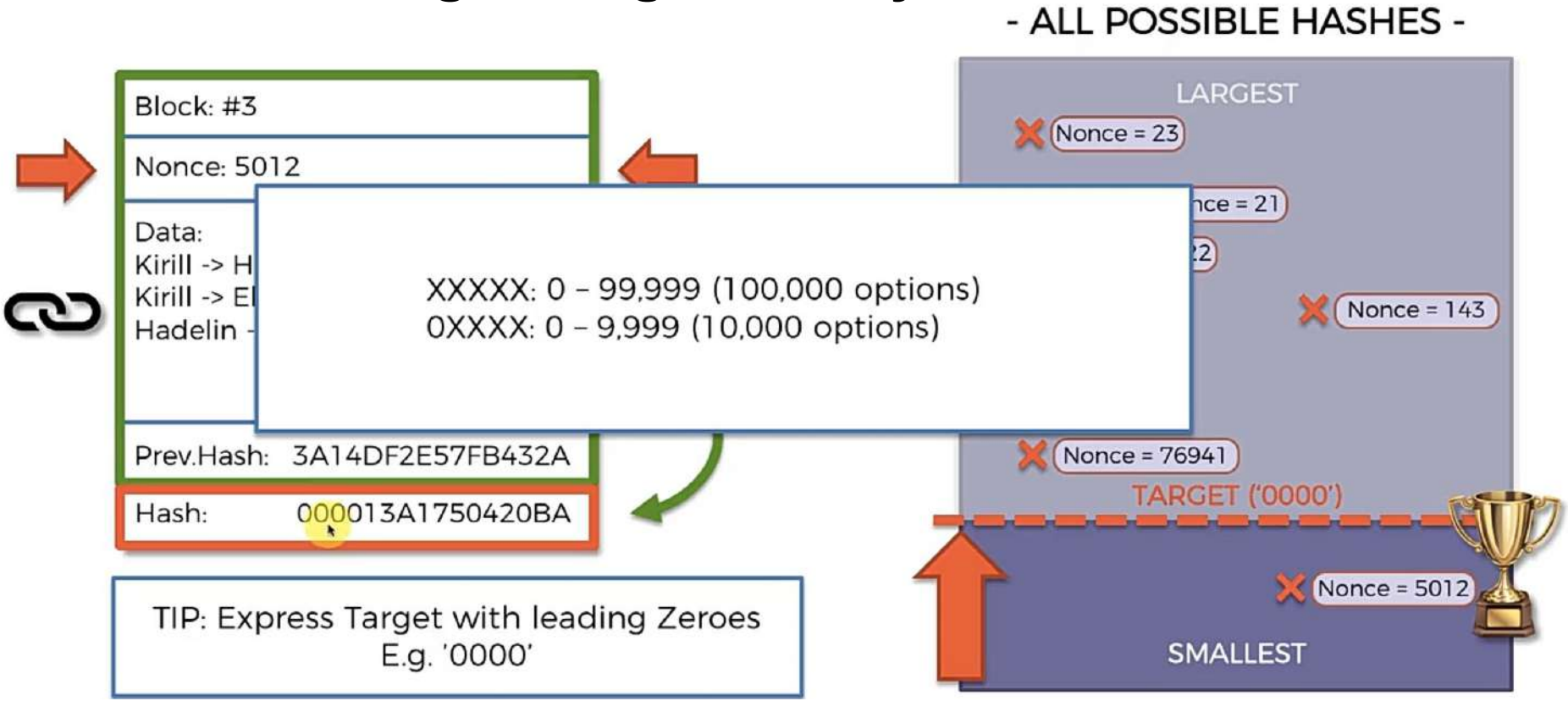




Understanding Mining Difficulty



Understanding Mining Difficulty



Understanding Mining Difficulty

Q2: How is “Mining Difficulty” calculated?

Understanding Mining Difficulty

Difficulty = current target / max target

Curr target = 0000000000000000005d97dc0000000000000000000000000000000000000000

Max target = 00000000FFFF0000000000000000000000000000000000000000000000000000

Difficulty is adjusted every 2016 blocks (2 weeks)

Understanding Mining Difficulty

Current target = 00000000000000005d97dc0000000000000000000000000000000000000000

18 zeros

Let's do some estimations:

Probability:

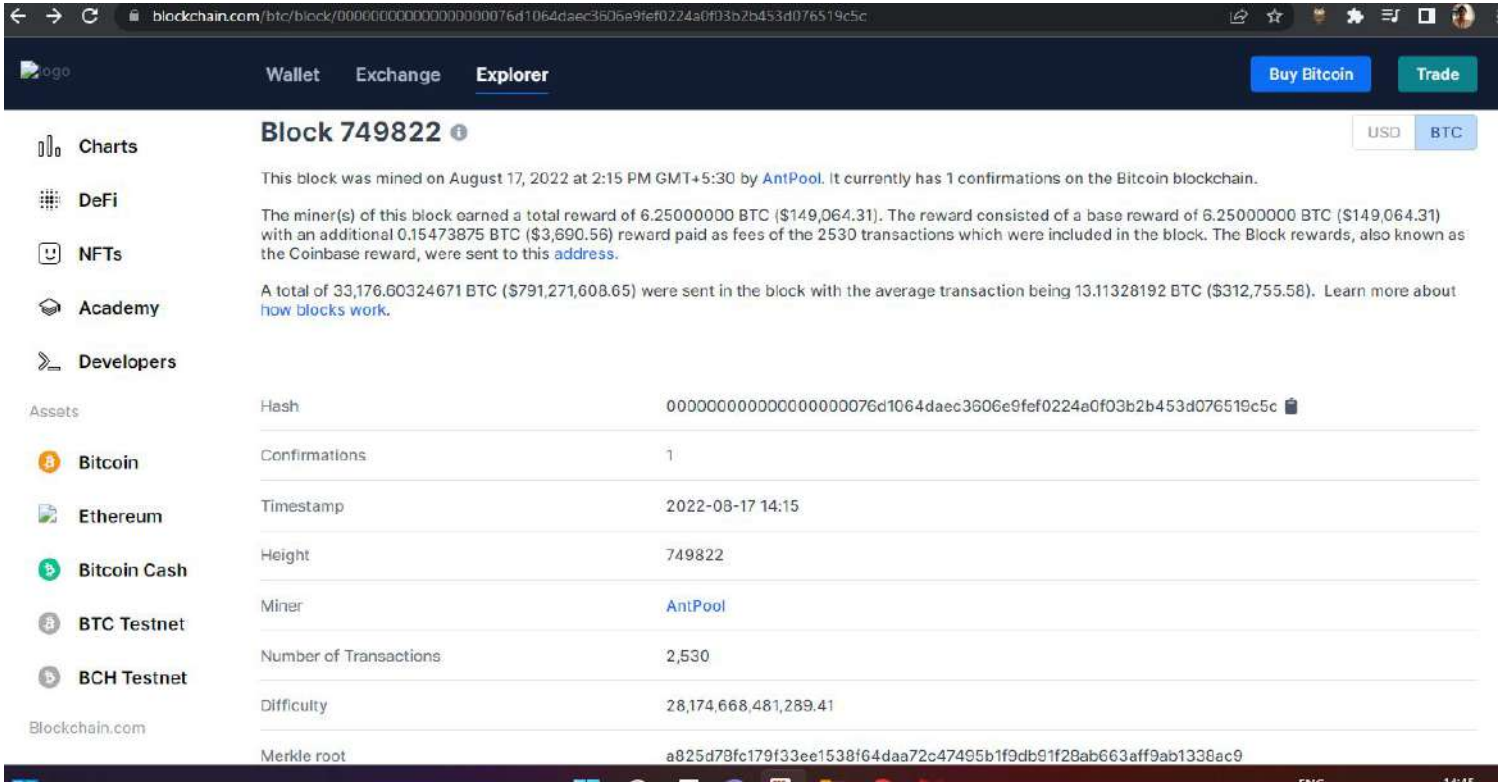
Total possible 64-digit hexadecimal numbers: $16 \times 16 \times \ldots \times 16 = 16^{64} \approx 1.1579 \times 10^{77} \approx 10^{77}$

Total valid hashes (with 18 leading zeros): $16 \times 16 \times \ldots \times 16 = 16^{64-18} \approx 2.4519 \times 10^{55} \approx 2 \times 10^{55}$

Probability that a Randomly picked hash is valid: $2 \times 10^{55} / 10^{77} = 2 \times 10^{-22} = 0.00000000000000000002\%$

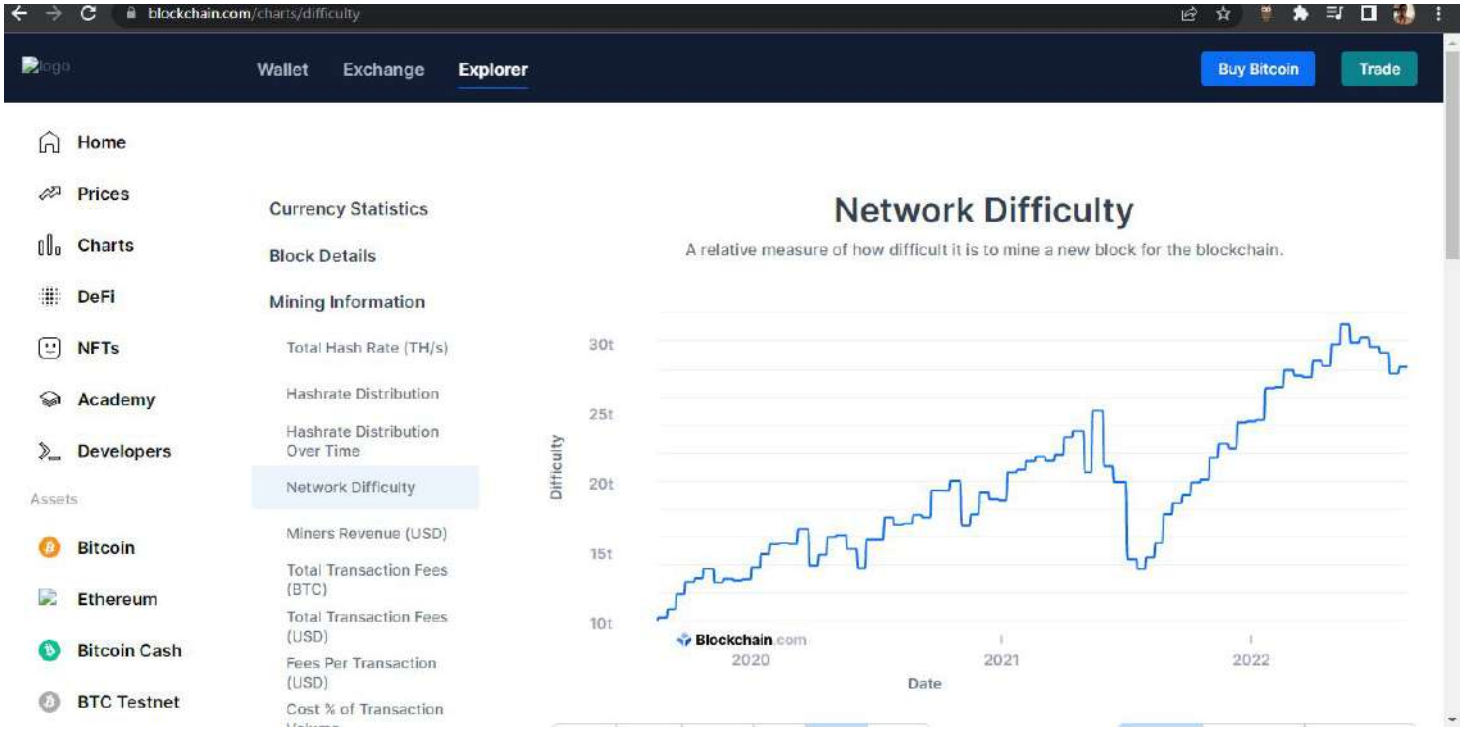
Mining Difficulty - Demo

Courtesy : https://www.blockchain.com/btc/block/0000000000000000000076d1064daec3606e9fef0224a0f03b2b453d076519c5c

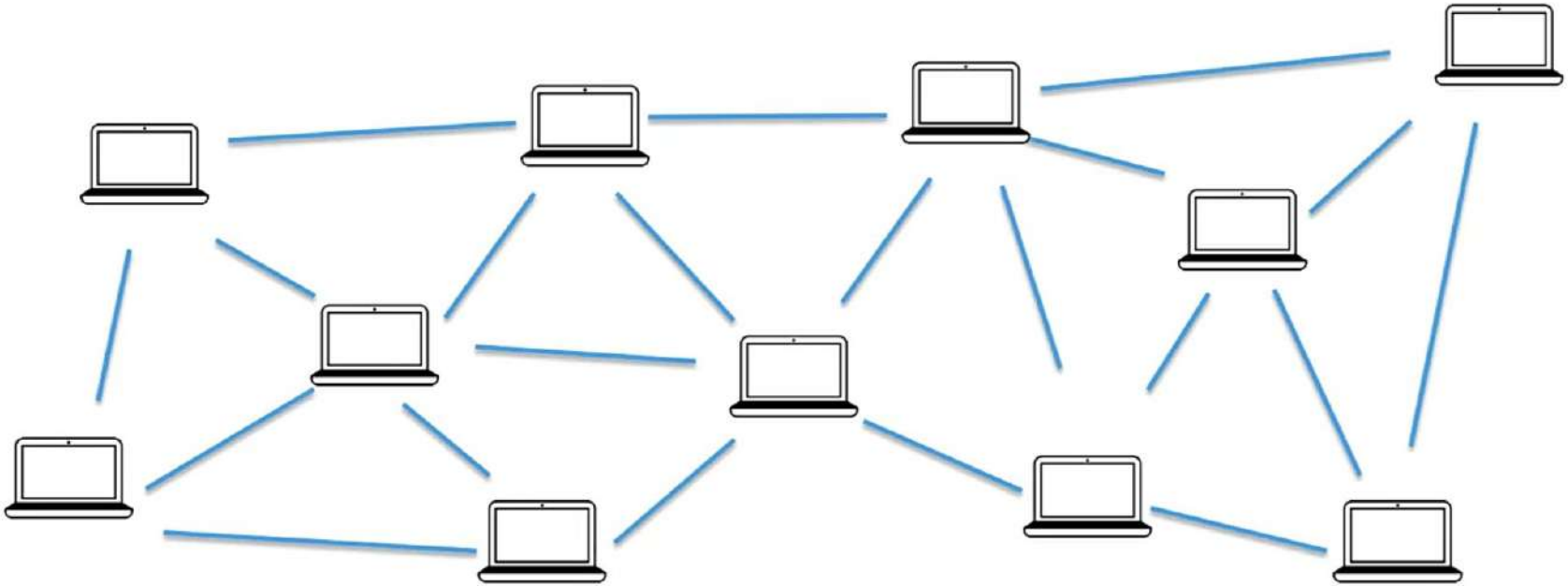


Mining Difficulty - Demo

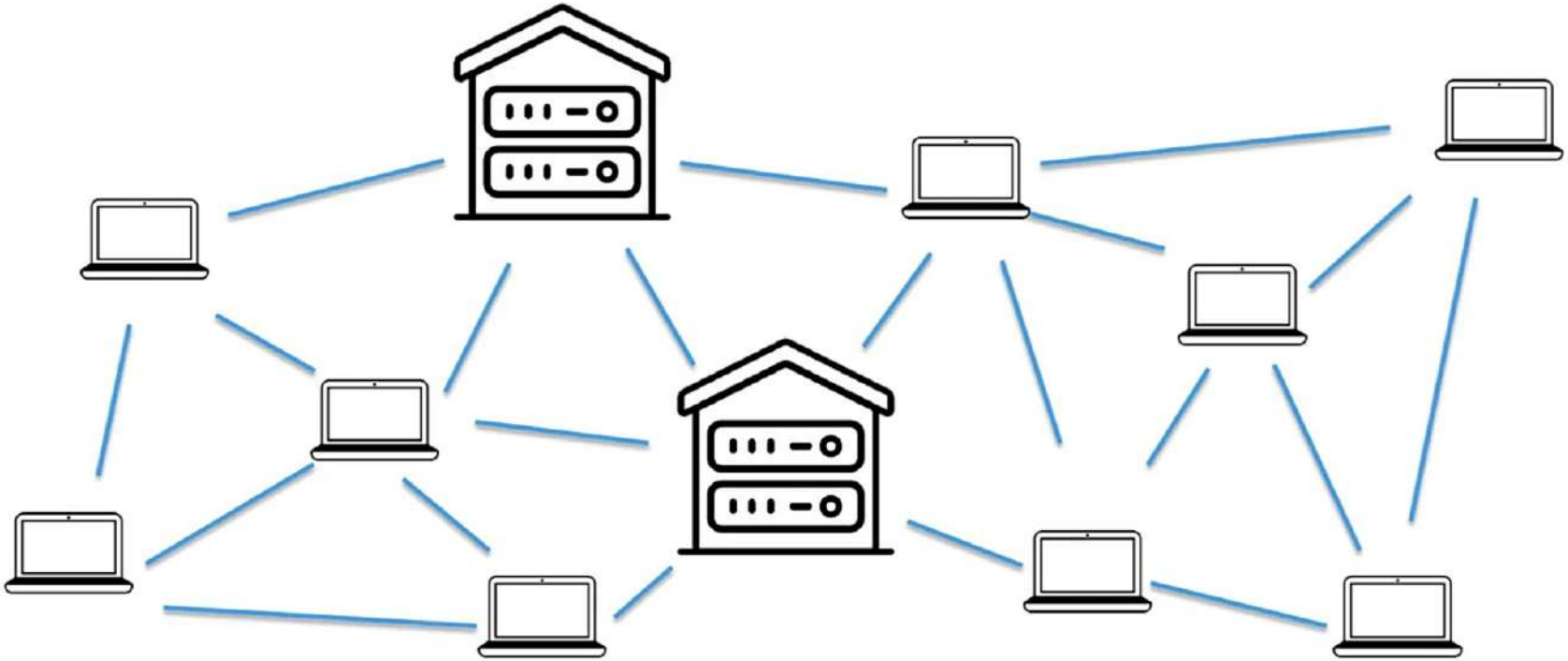
Courtesy : https://www.blockchain.com/charts/difficulty



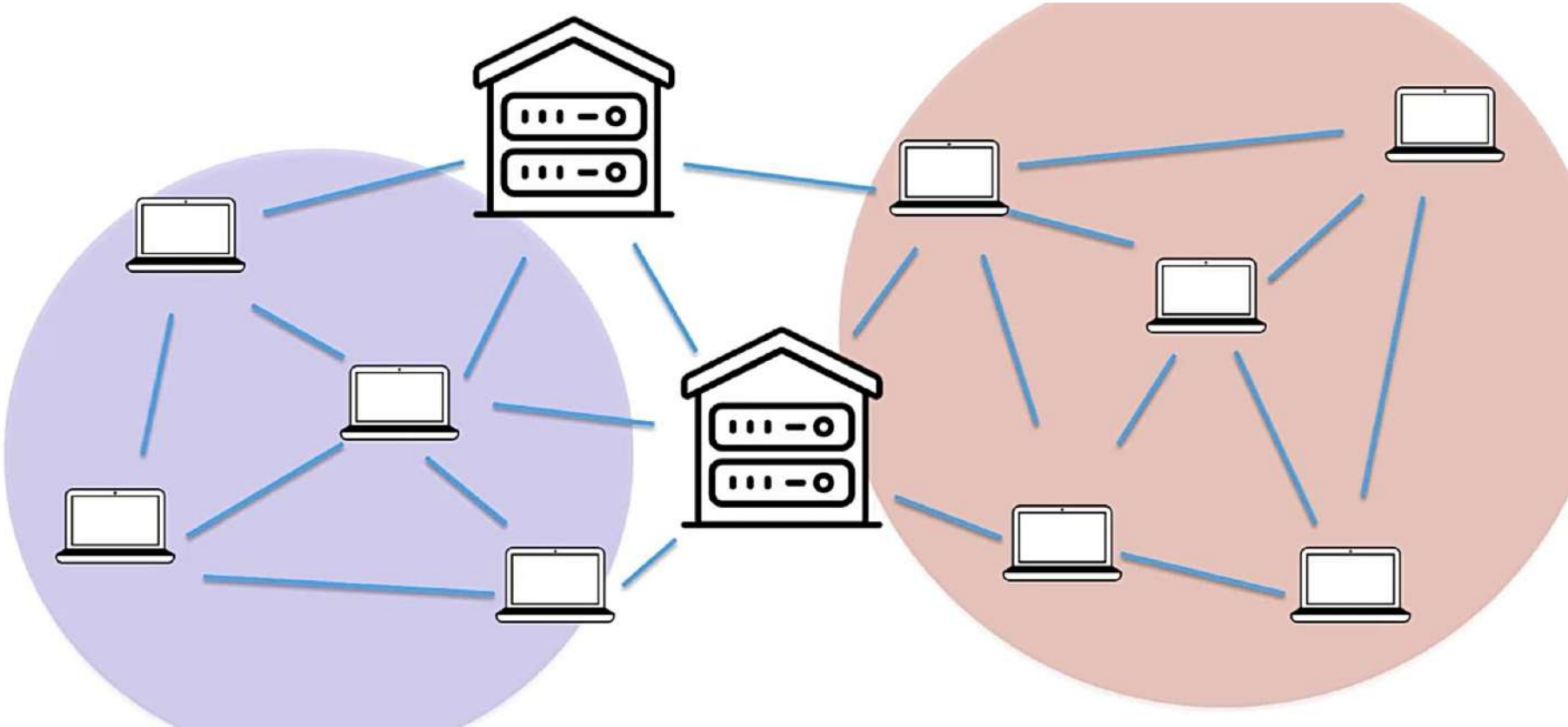
Mining Pools



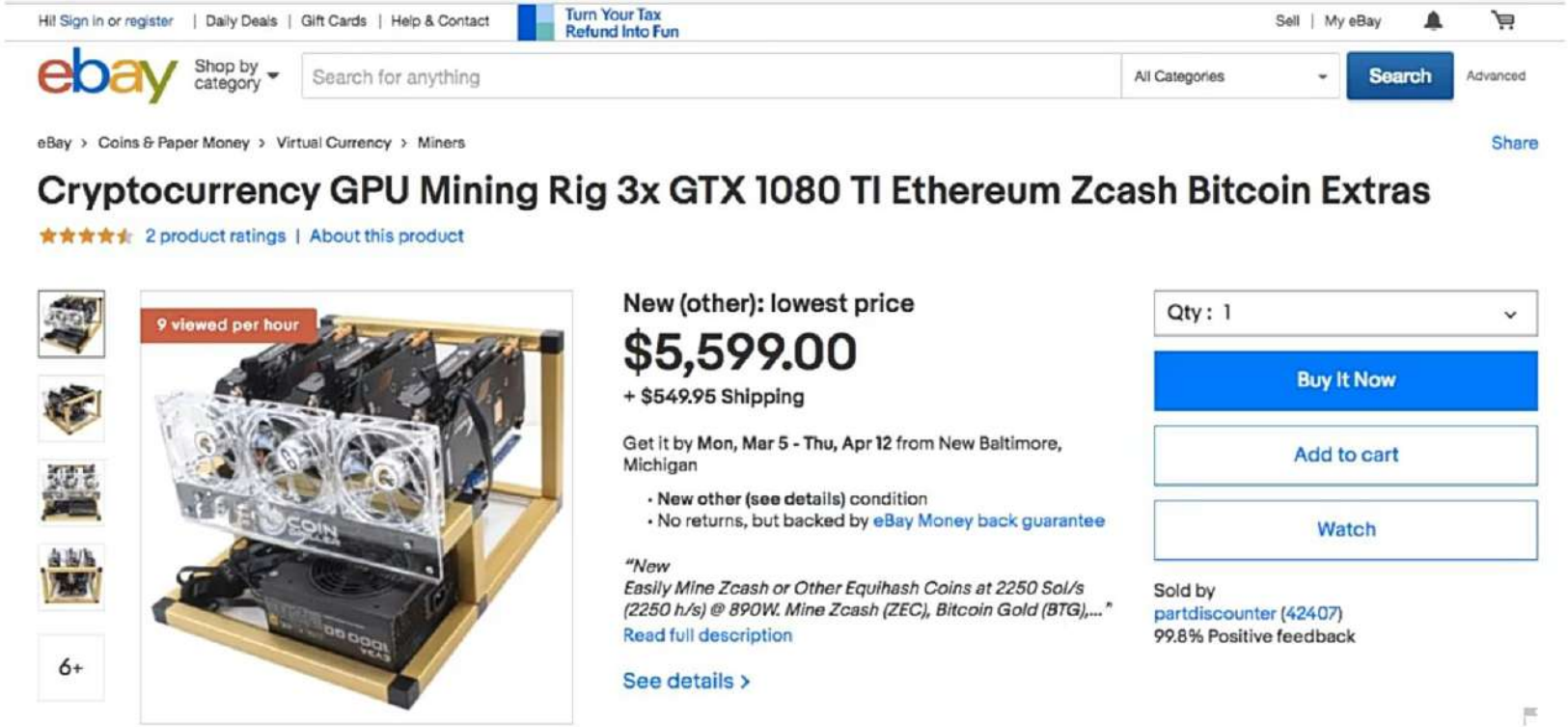
Mining Pools



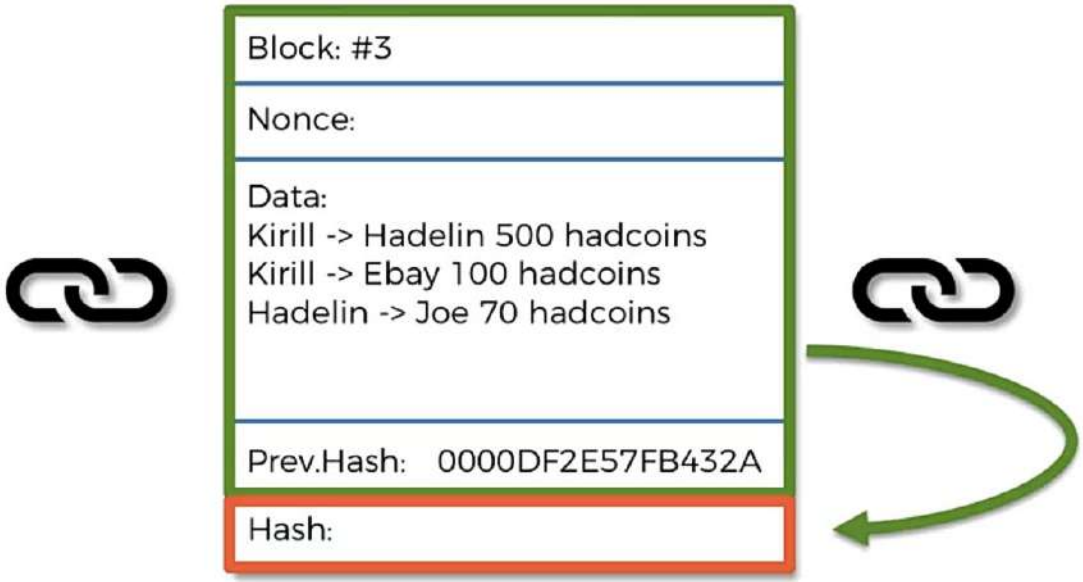
Mining Pools



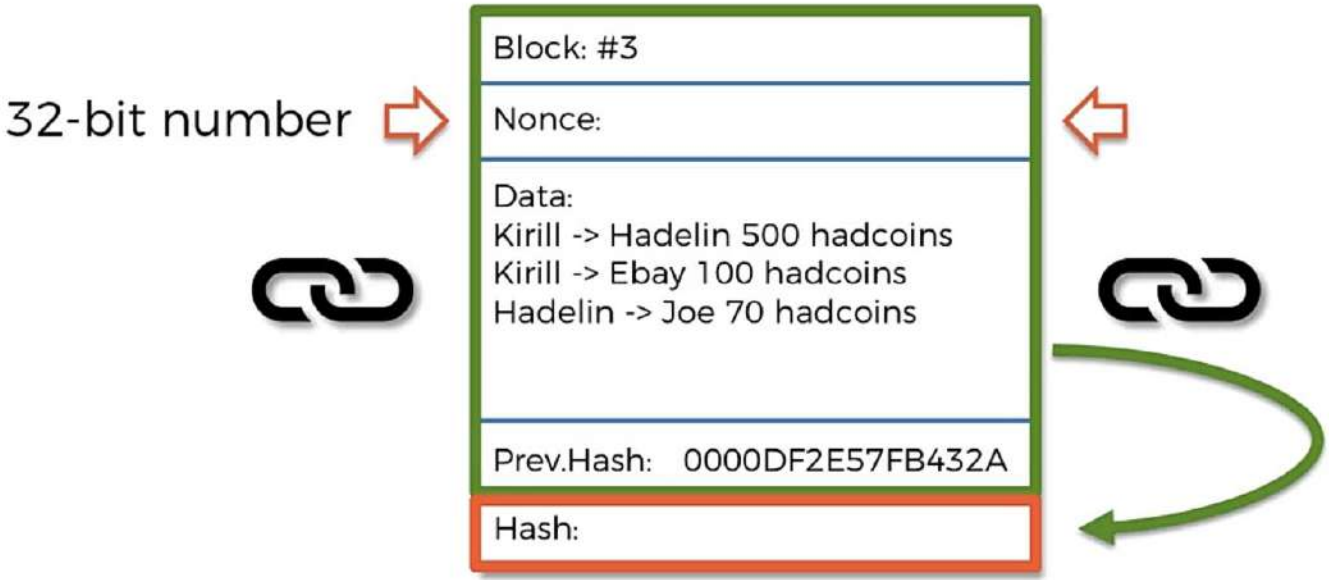
Mining Pools



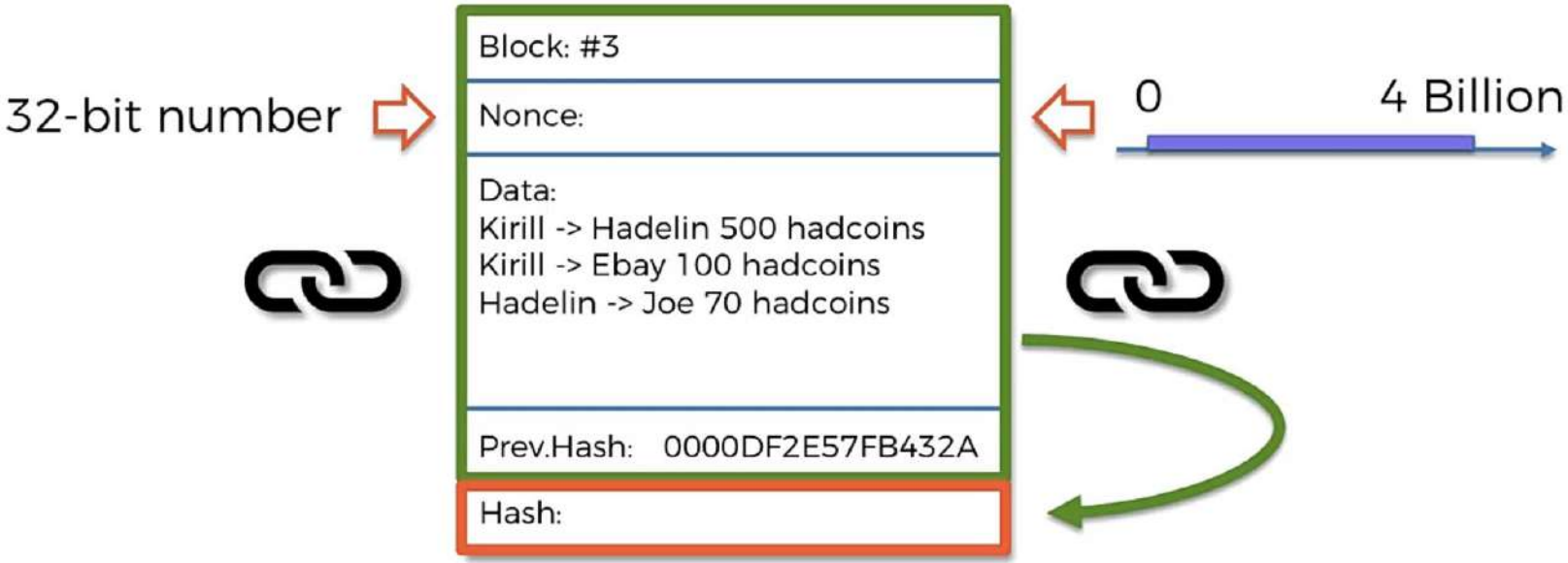
Nonce Range


Block: #3
Nonce:
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:

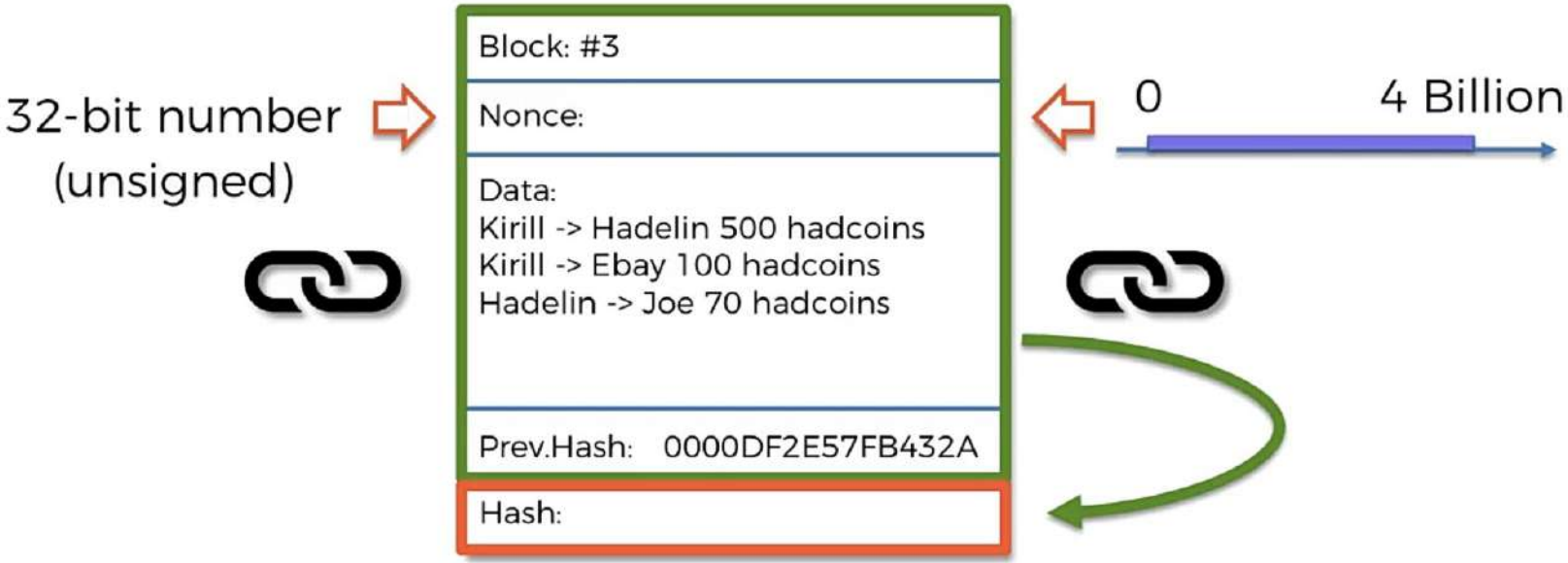
Nonce Range


32-bit number
Block: #3
Nonce:
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:

Nonce Range


32-bit number
Block: #3
Nonce:
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:
0
4 Billion

Nonce Range


32-bit number
(unsigned)
Block: #3
Nonce:
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:
0
4 Billion

Nonce Range

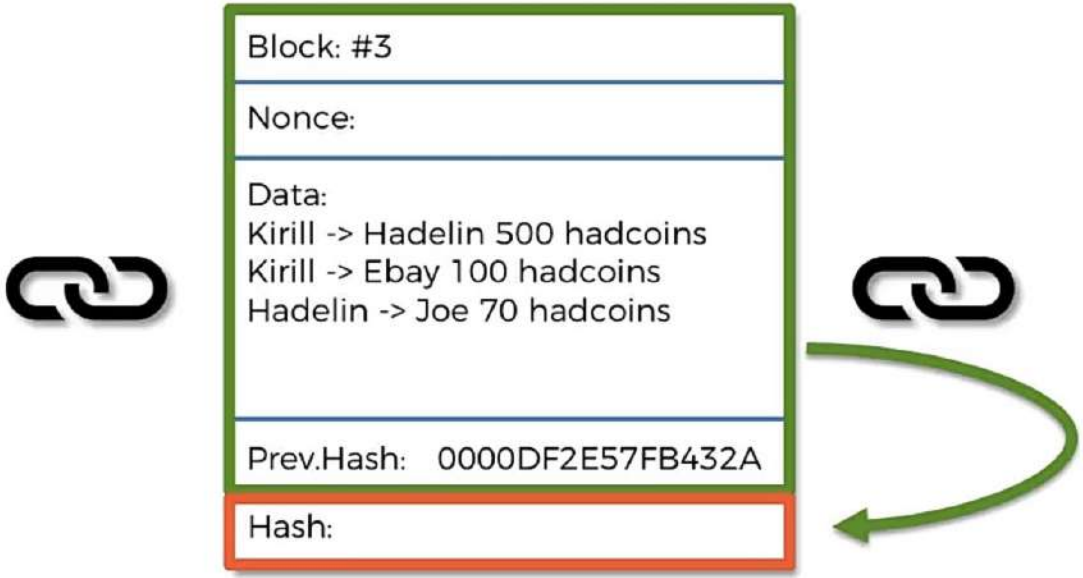
Let's do some estimations:

Difficulty:
Total possible 64-digit hexadecimal numbers: 16 x 16 x .... x 16 = $16^{64} \approx 10^{77}$
Total valid hashes (with 18 leading zeros): 16 x 16 x ... x 16 = $16^{64-18} \approx 2 \times 10^{55}$
Probability that a Randomly picked hash is valid: $2 \times 10^{55} / 10^{77} = 2 \times 10^{-22} = 0.00000000000000000002\%$

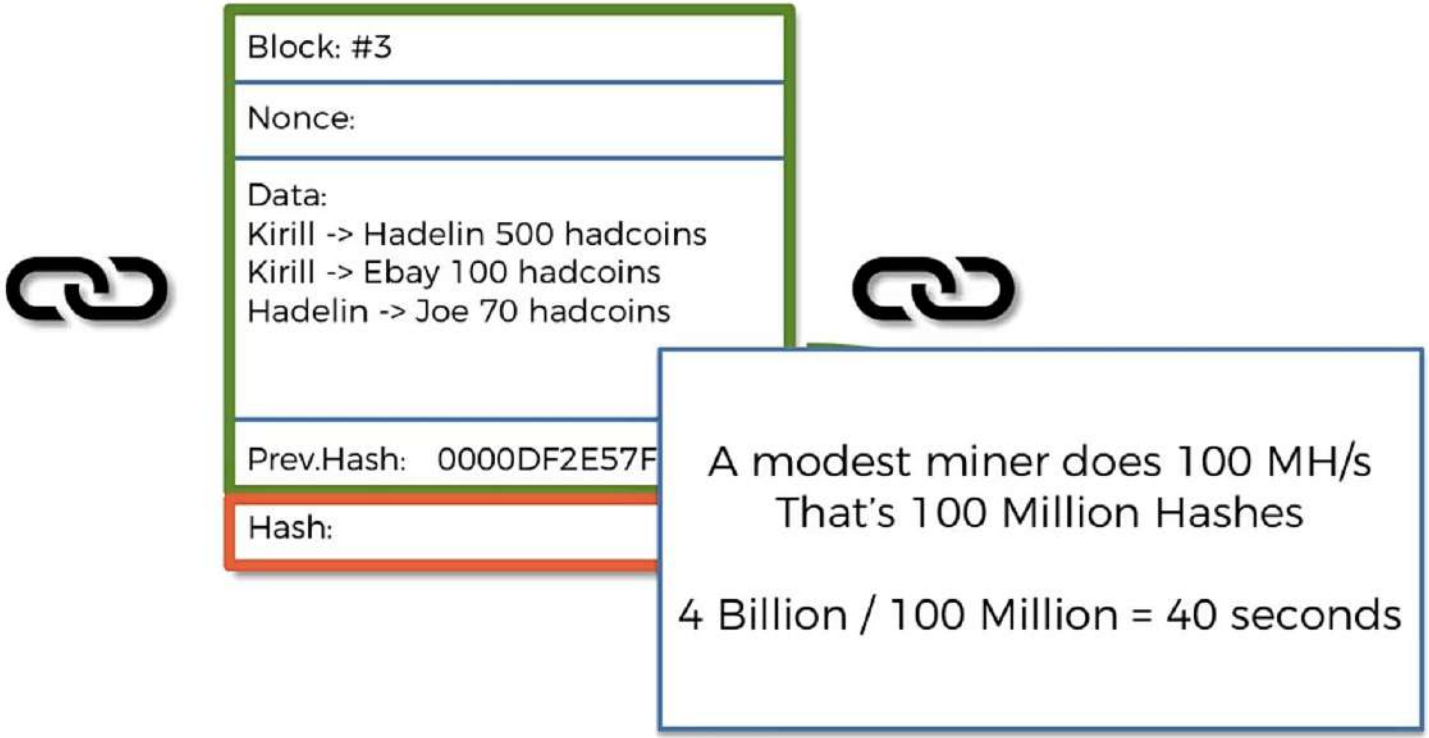
Nonce:
The Nonce is a 32-bit number, the Max Nonce = $2^{32} = 4{,}294{,}967{,}296 = 4 \times 10^{9}$
Assuming no collisions, this means 4×10^{9} different hashes
Probability that ONE of them will be valid: $4 \times 10^{9} \times 2 \times 10^{-22} = 8 \times 10^{-13} \approx 10^{-12} = 0.0000000001\%$

Conclusion: One Nonce Range is not enough

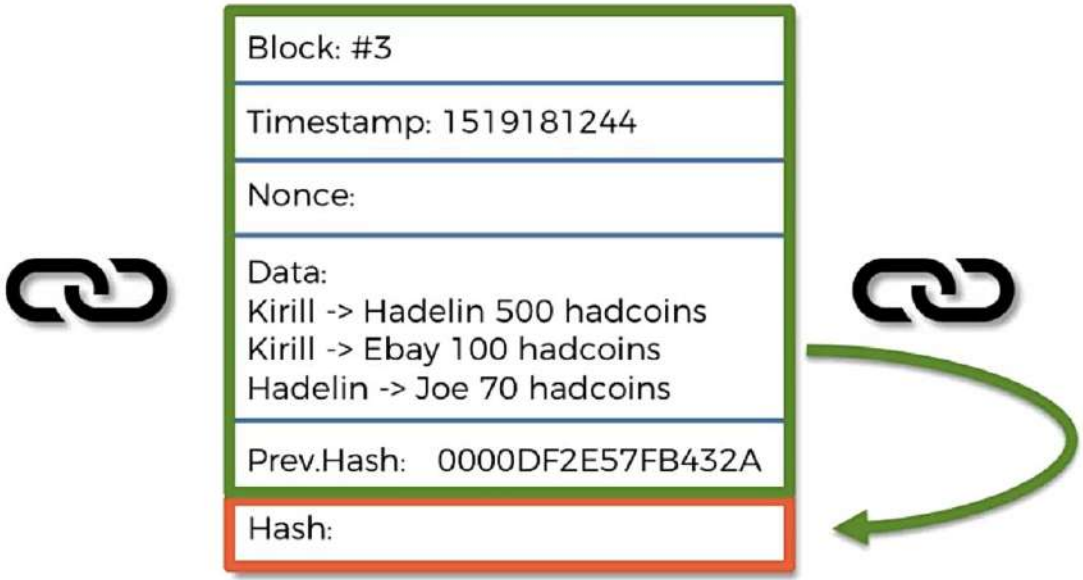
Nonce Range


Block: #3
Nonce:
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:

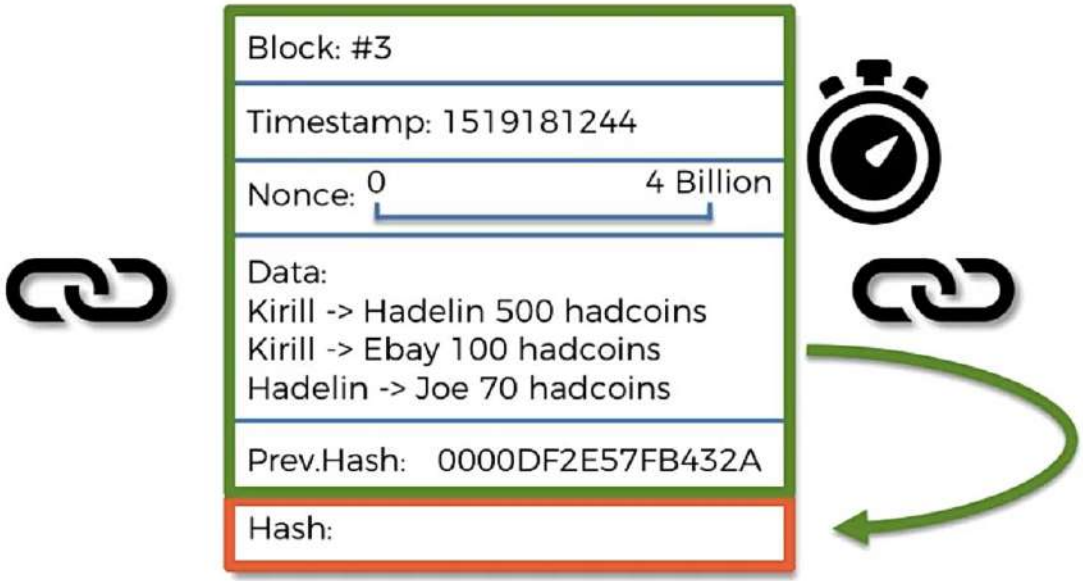
Nonce Range


Block: #3
Nonce:
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57F
Hash:
A modest miner does 100 MH/s
That's 100 Million Hashes
4 Billion / 100 Million = 40 seconds

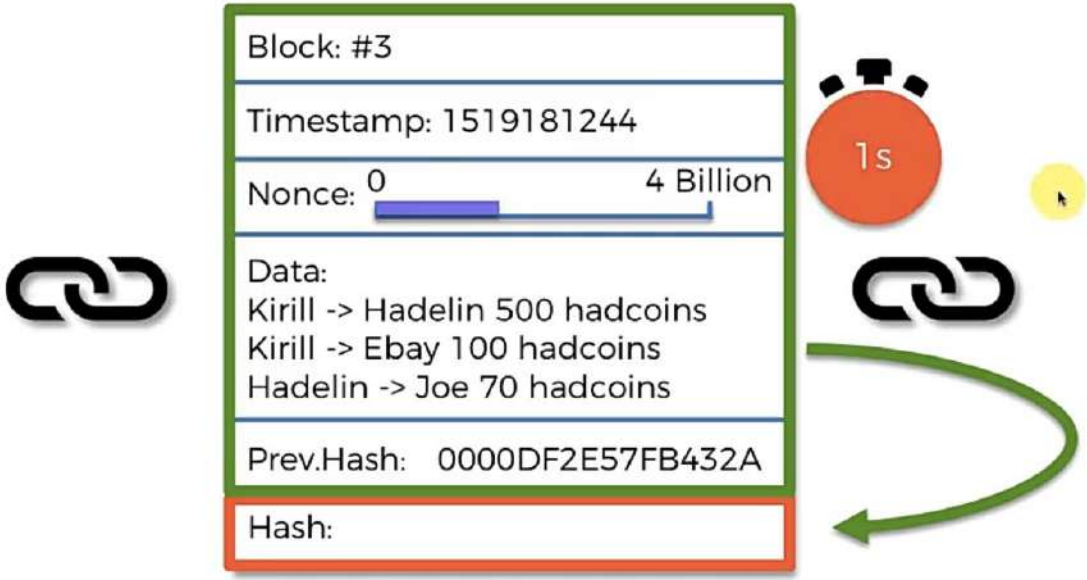
Nonce Range


Block: #3
Timestamp: 1519181244
Nonce:
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:

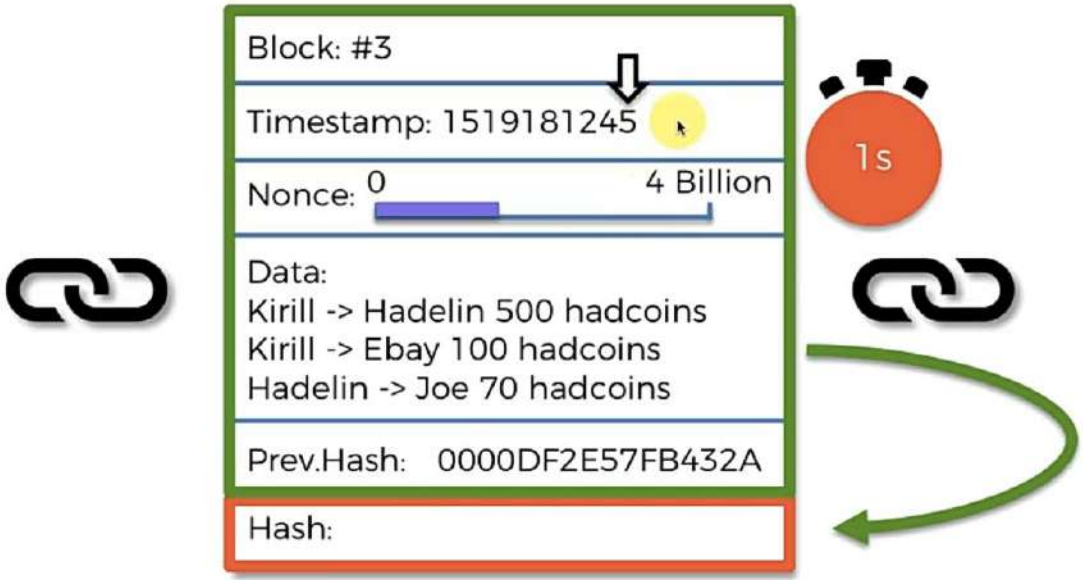
Nonce Range


Block: #3
Timestamp: 1519181244
Nonce: 0
4 Billion
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:

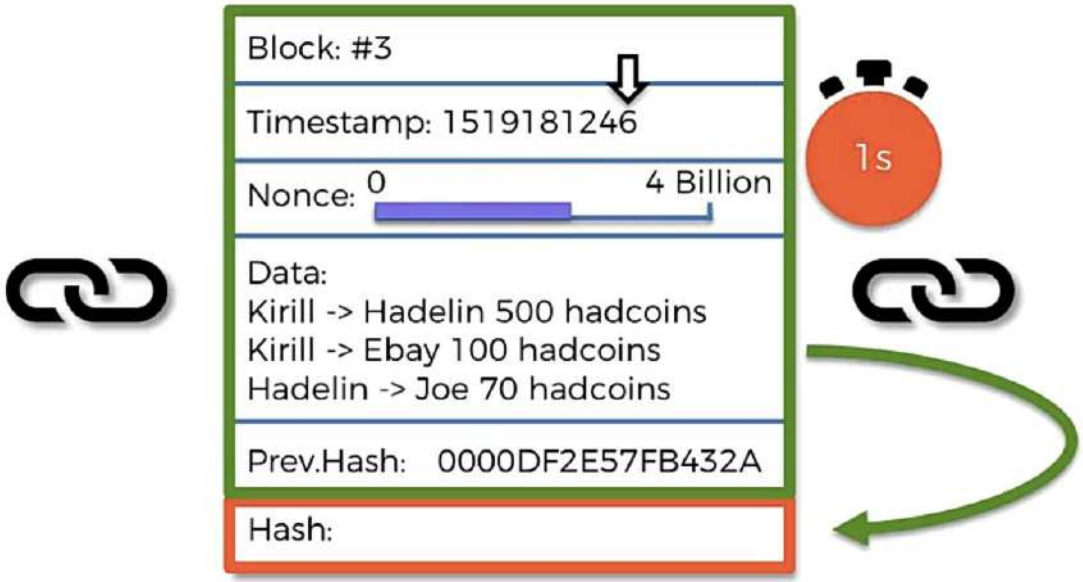
Nonce Range


Block: #3
Timestamp: 1519181244
Nonce: 0
4 Billion
1s
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:

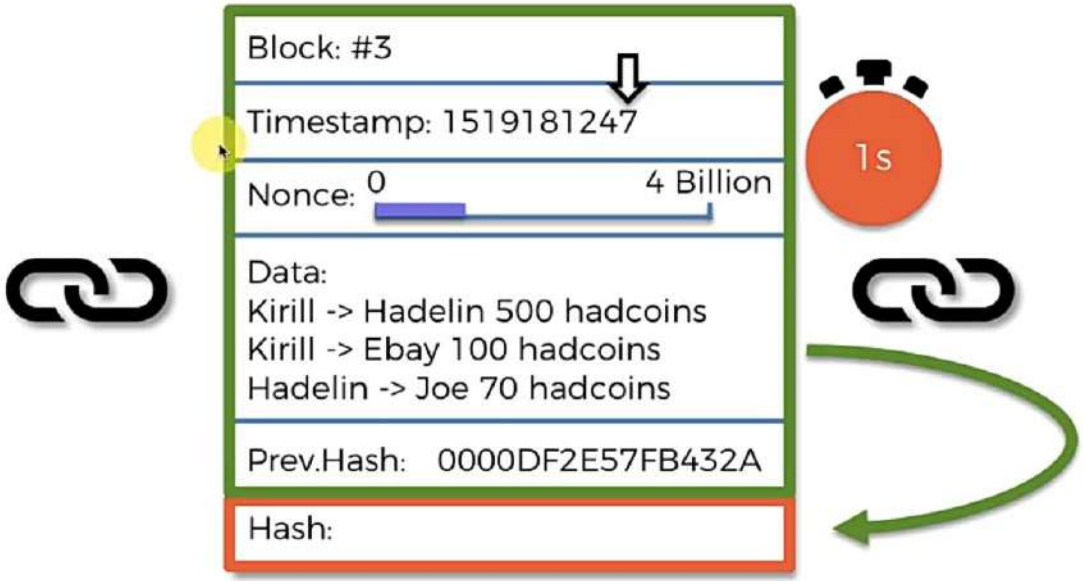
Nonce Range


Block: #3
Timestamp: 1519181245
Nonce: 0
4 Billion
1s
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:

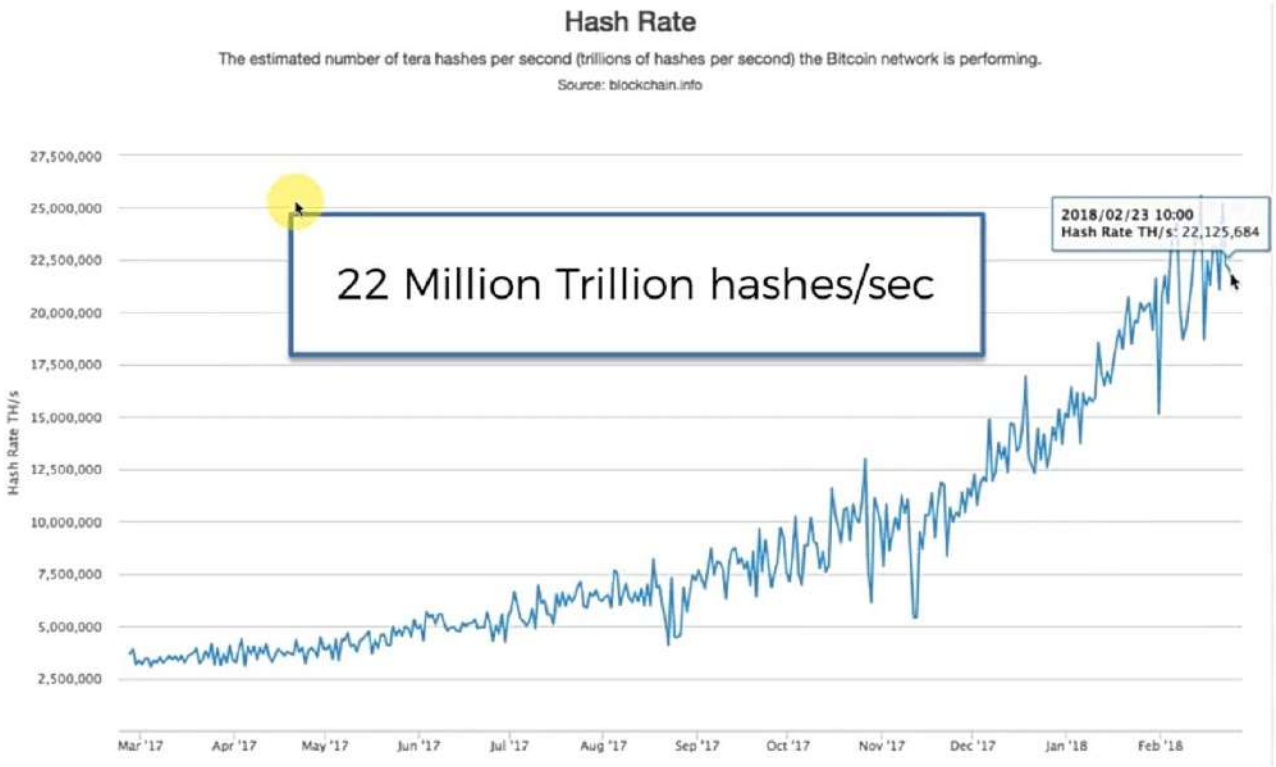
Nonce Range


Block: #3
Timestamp: 1519181246
Nonce: 0
4 Billion
1s
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:

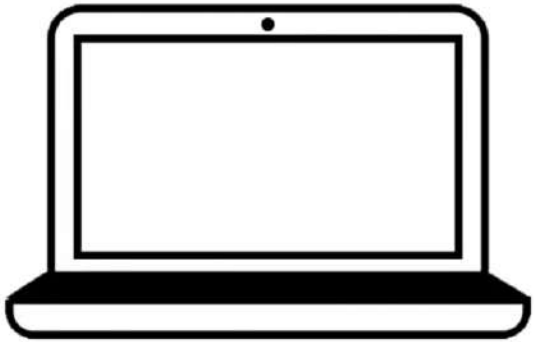
Nonce Range


Block: #3
Timestamp: 1519181247
Nonce: 0
4 Billion
1s
Data:
Kirill -> Hadelin 500 hadcoins
Kirill -> Ebay 100 hadcoins
Hadelin -> Joe 70 hadcoins
Prev.Hash: 0000DF2E57FB432A
Hash:

Nonce Range


Hash Rate
The estimated number of tera hashes per second (trillions of hashes per second) the Bitcoin network is performing.
Source: blockchain.info
22 Million Trillion hashes/sec
2018/02/23 10:00
Hash Rate TH/s: 22,125,684
Hash Rate TH/s
27,500,000
25,000,000
22,500,000
20,000,000
17,500,000
15,000,000
12,500,000
10,000,000
7,500,000
5,000,000
2,500,000
Mar '17
Apr '17
May '17
Jun '17
Jul '17
Aug '17
Sep '17
Oct '17
Nov '17
Dec '17
Jan '18
Feb '18

How Miners Pick Transactions ?



(Mining in Process)

Block: #500,112

Timestamp: 1519181244

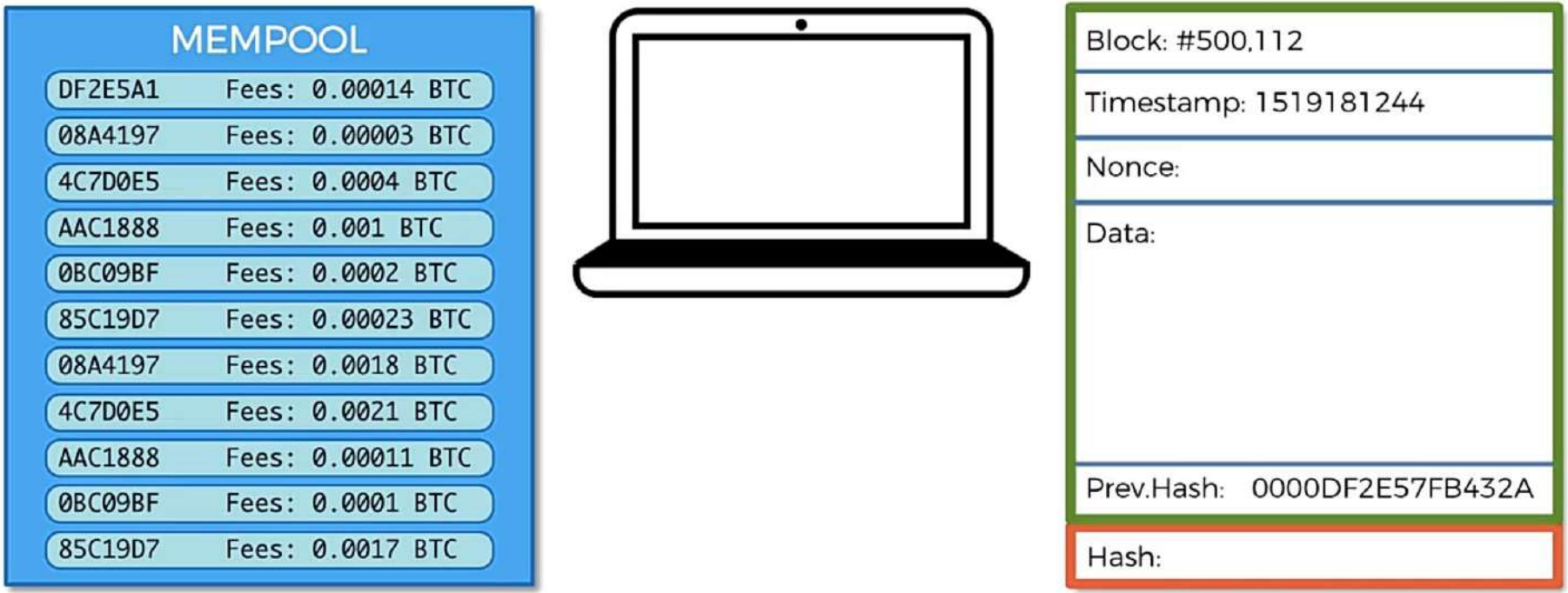
Nonce:

Data:

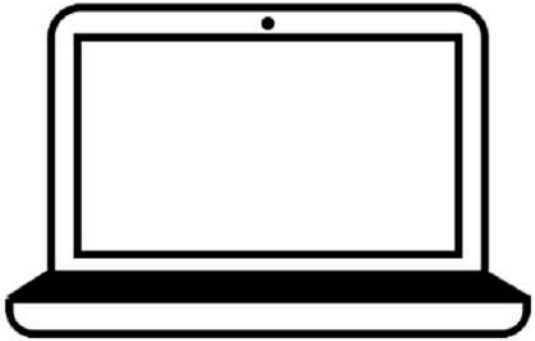
Prev.Hash: 0000DF2E57FB432A

Hash:

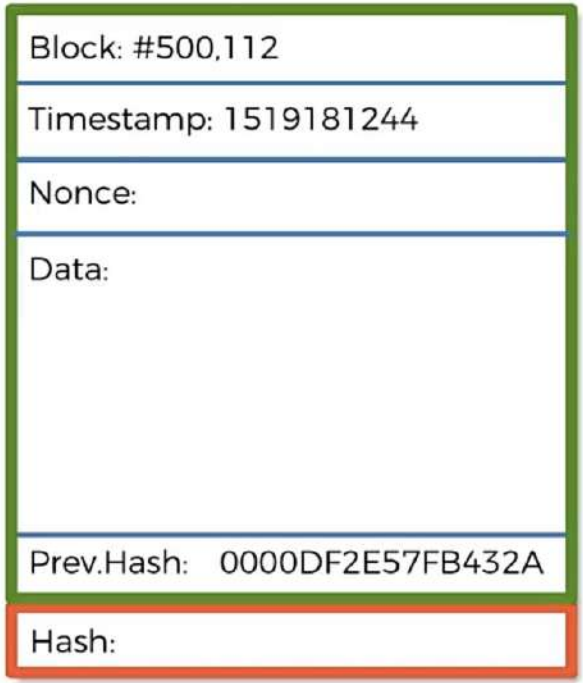
How Miners Pick Transactions ?



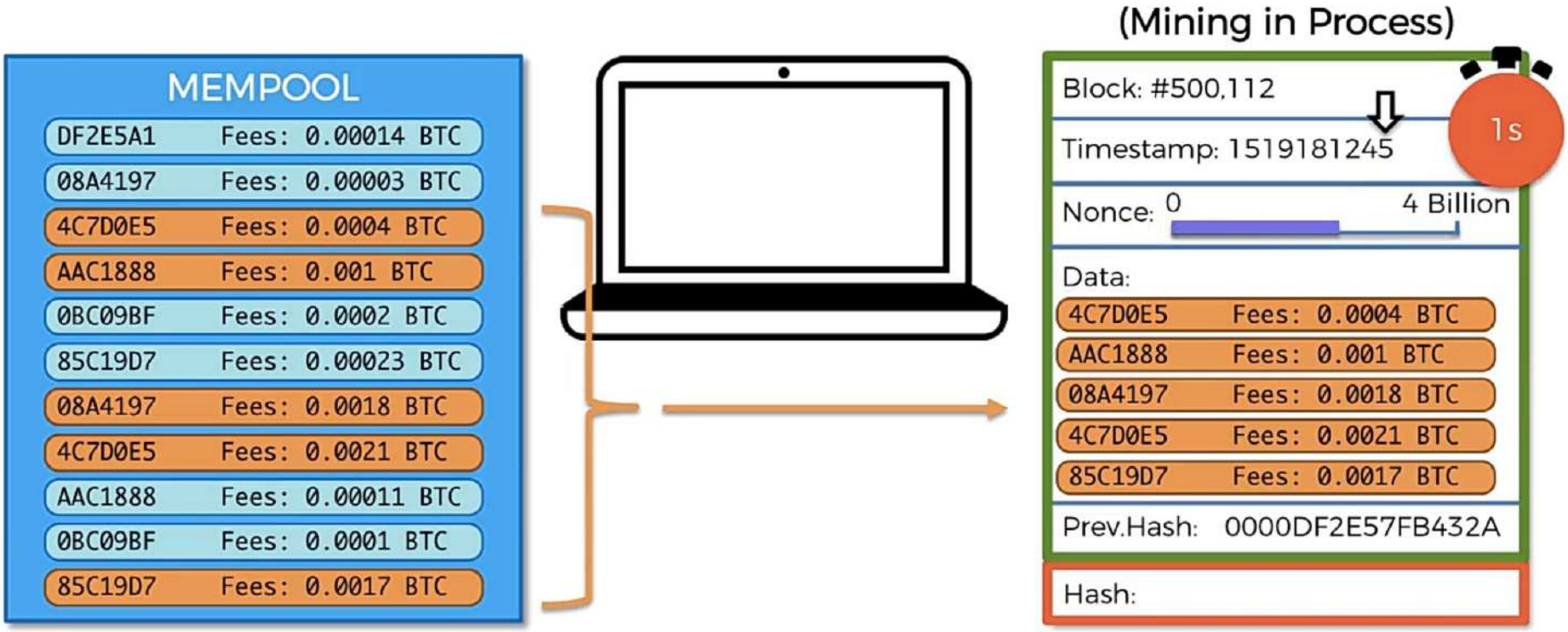
How Miners Pick Transactions ?



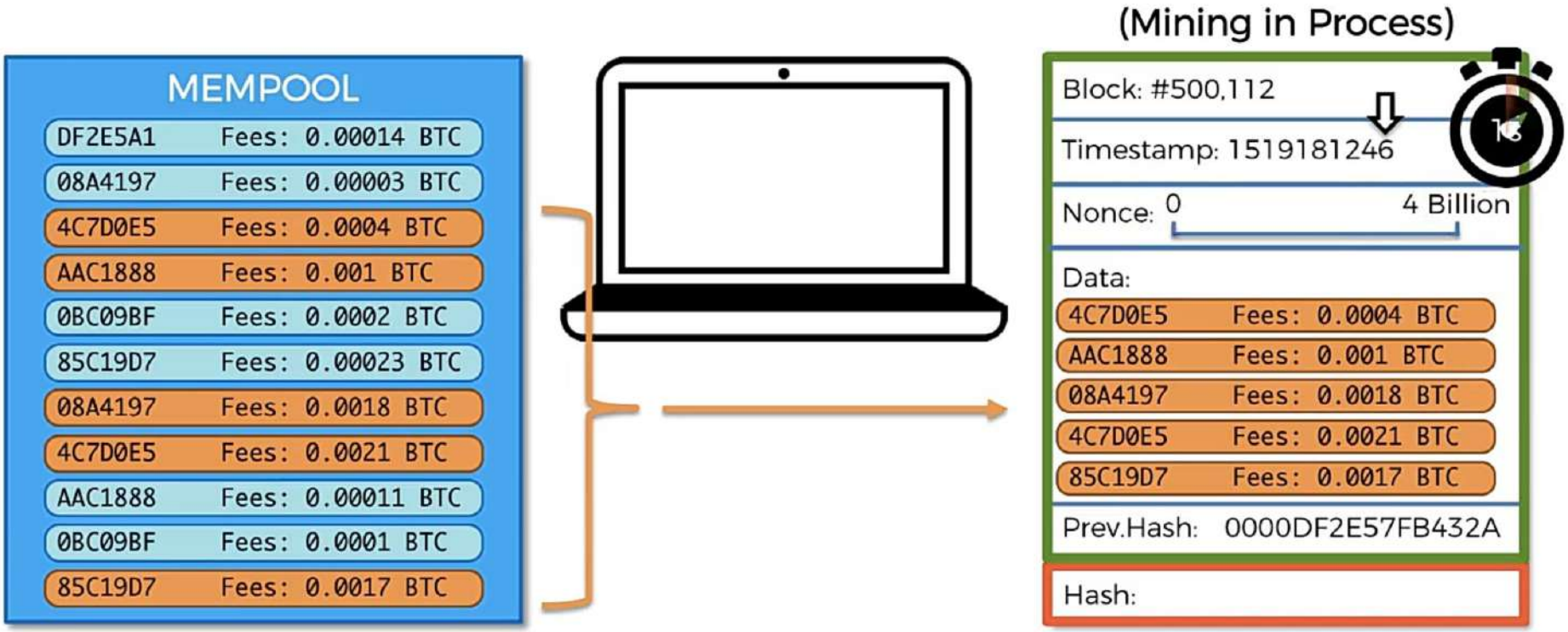
(Mining in Process)



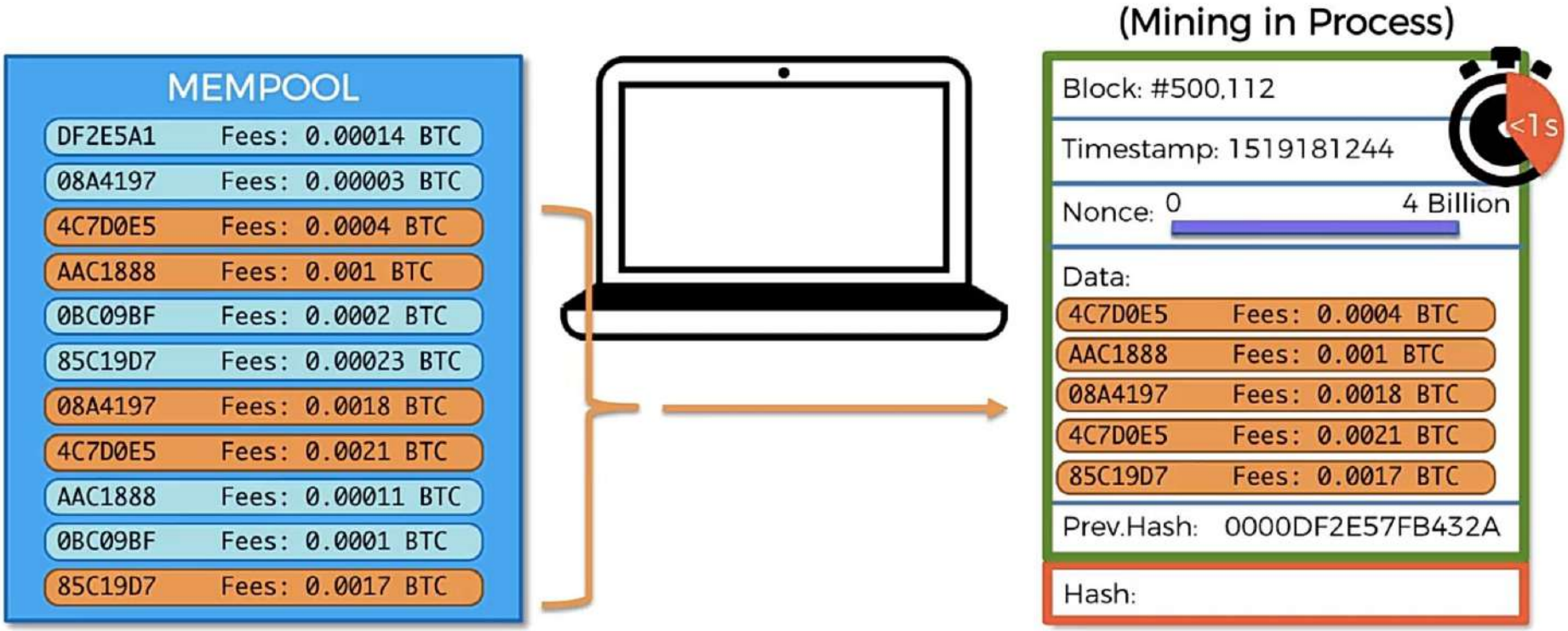
How Miners Pick Transactions ?



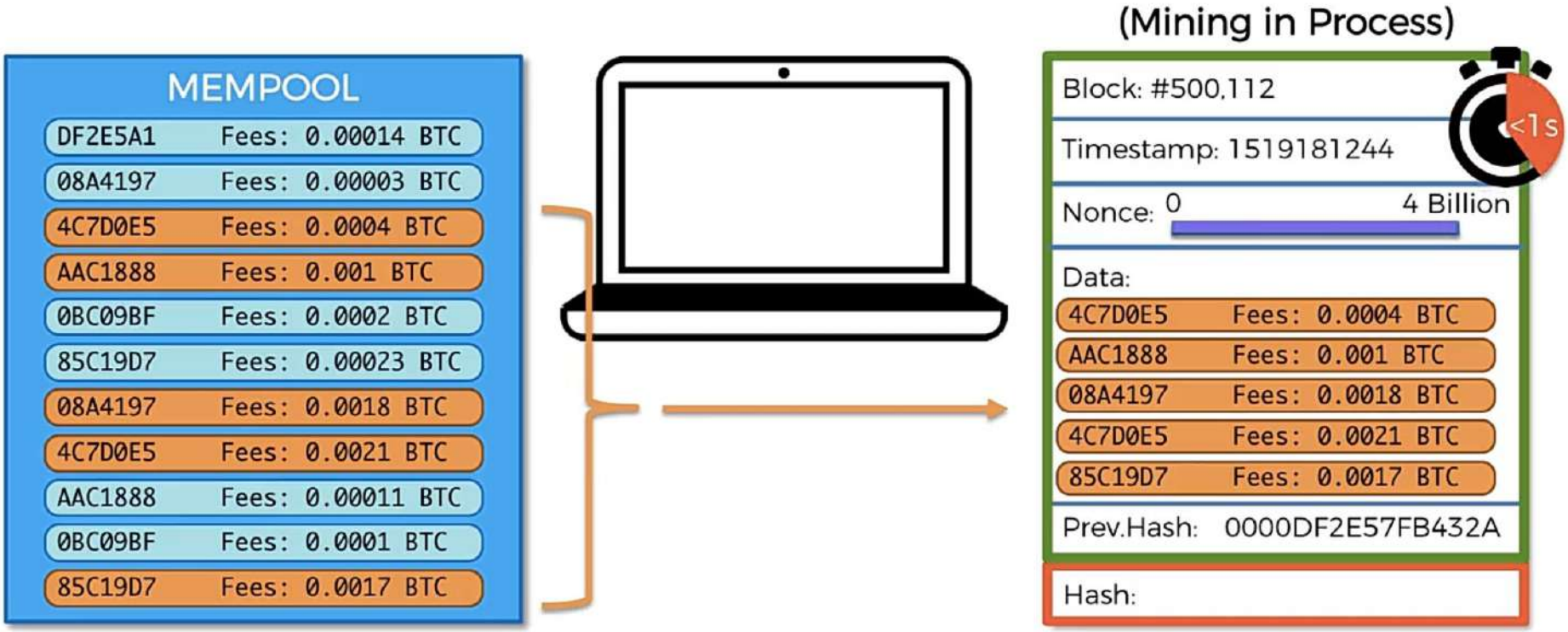
How Miners Pick Transactions ?



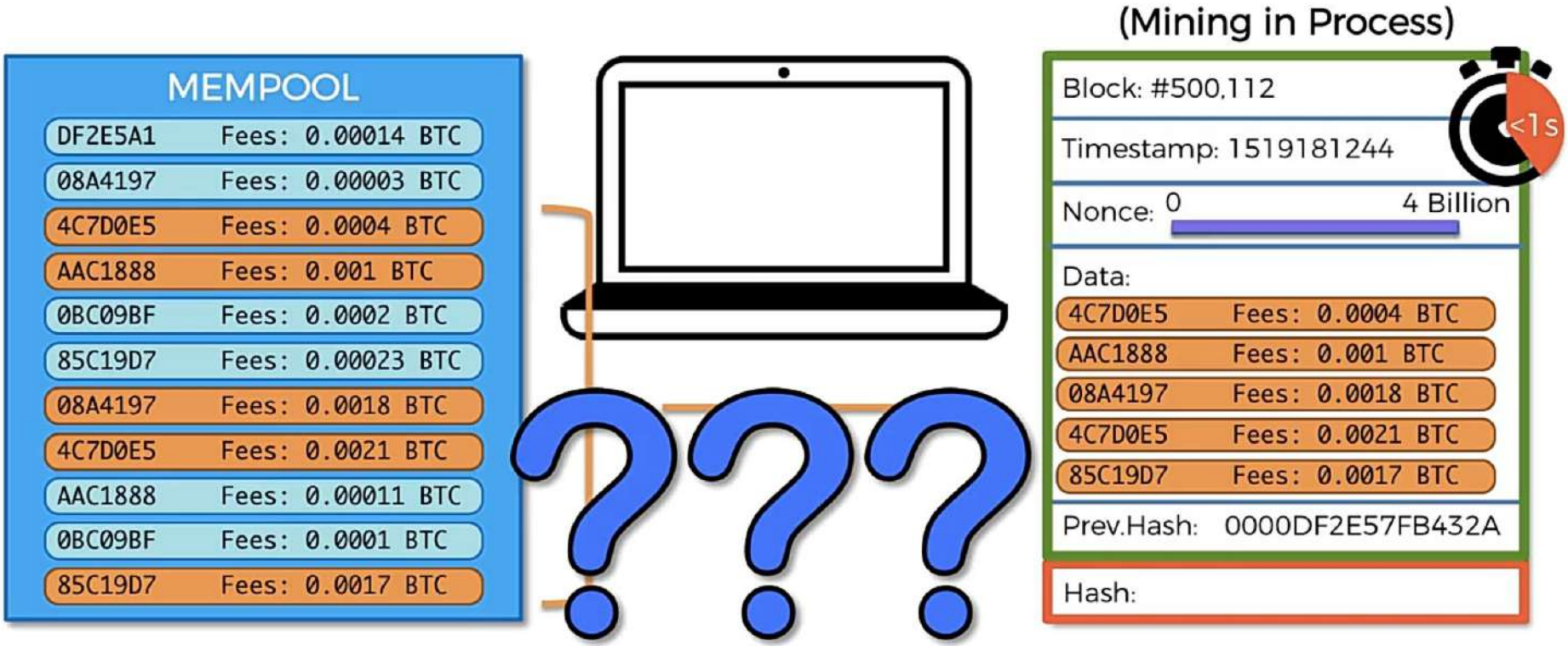
How Miners Pick Transactions ?



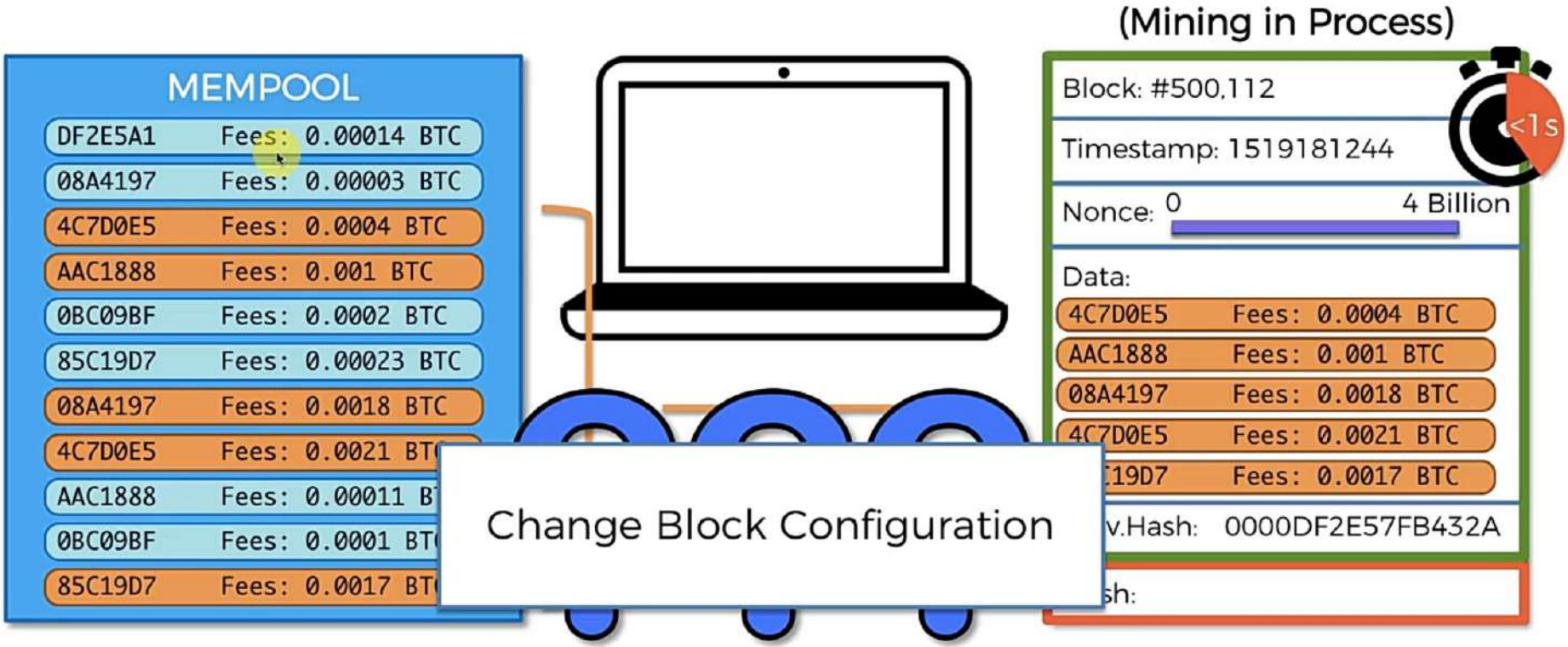
How Miners Pick Transactions ?



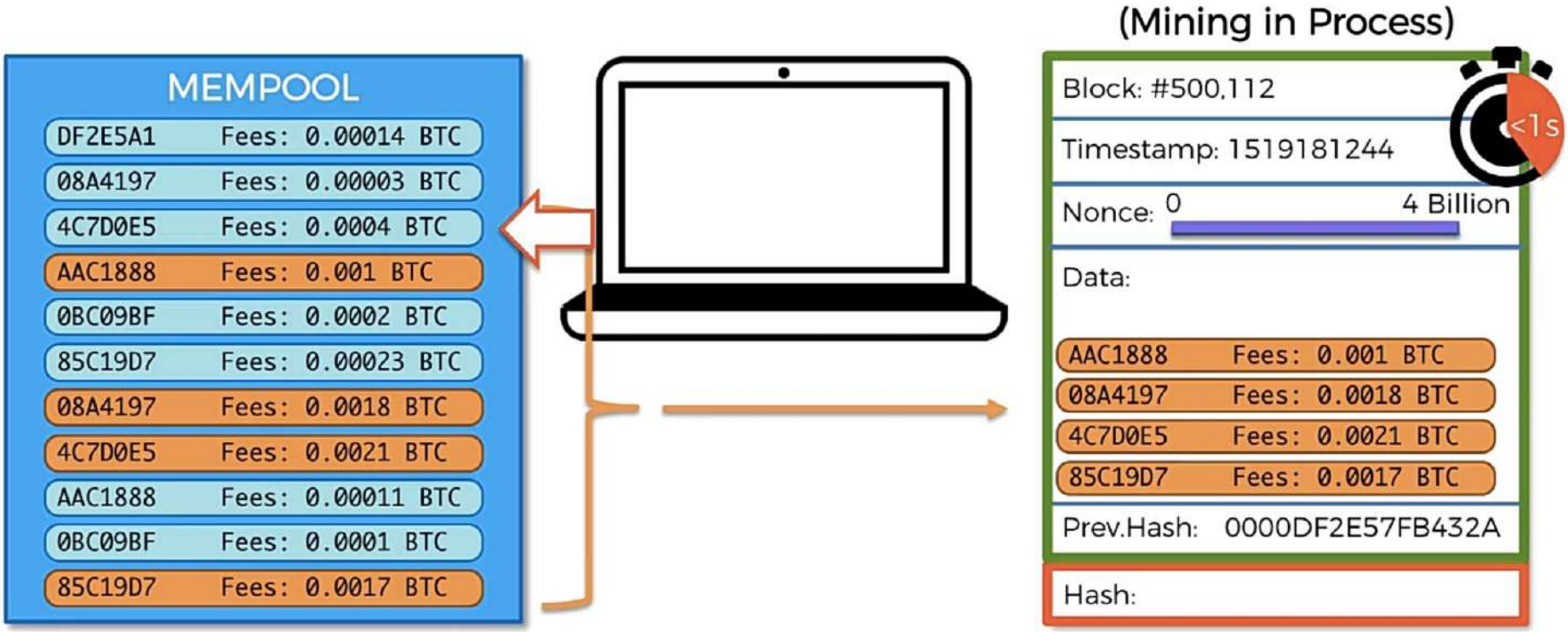
How Miners Pick Transactions ?



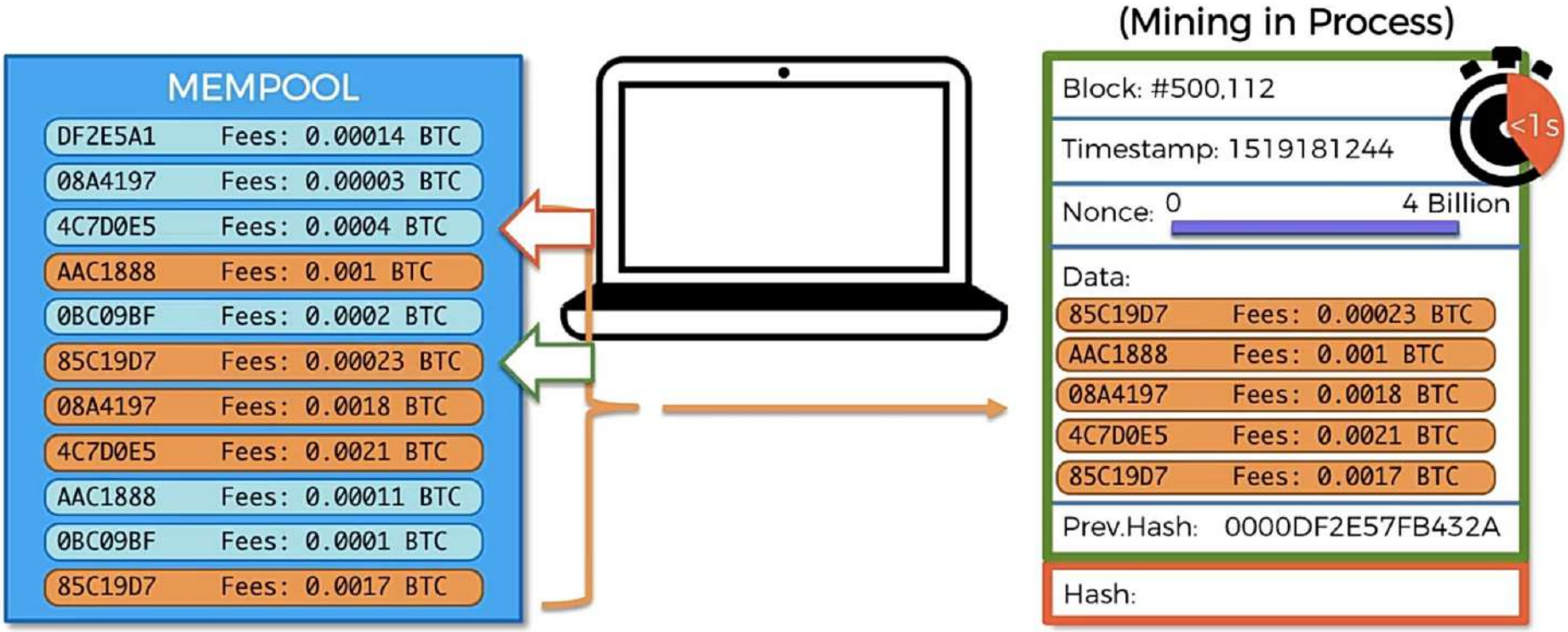
How Miners Pick Transactions ?



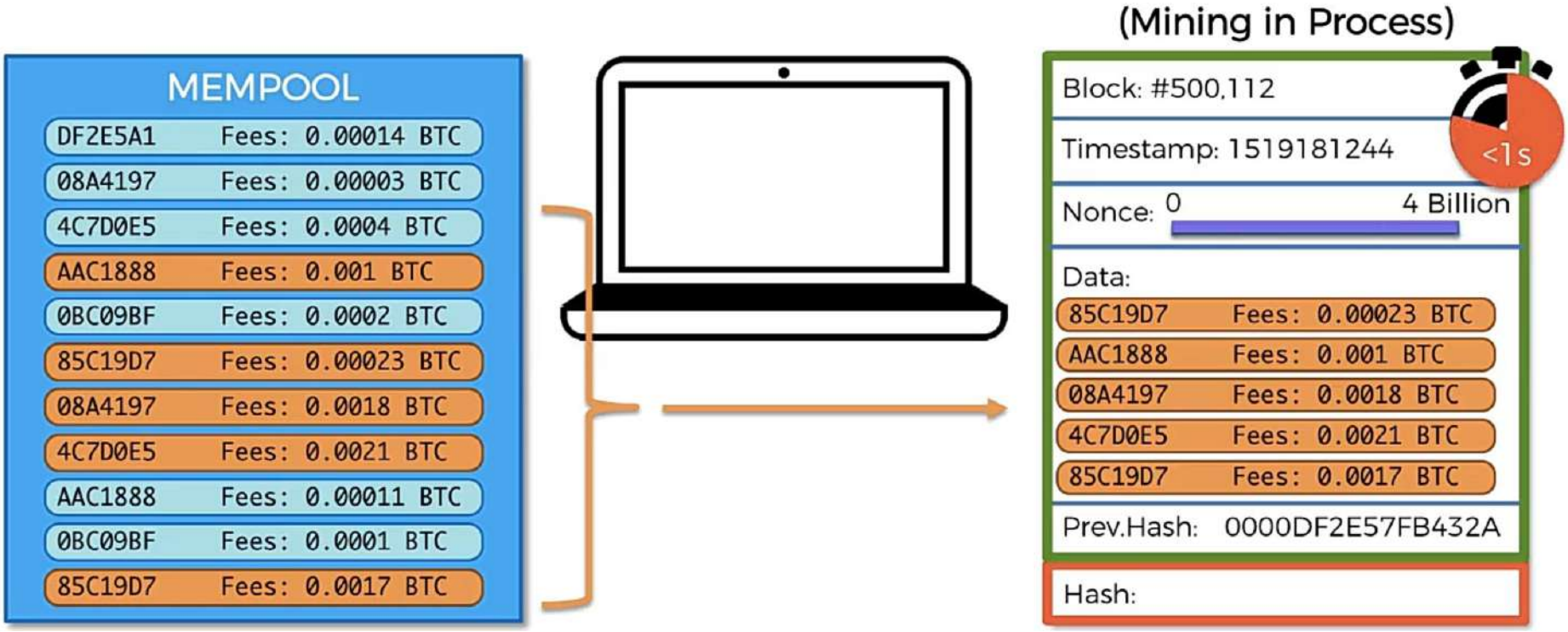
How Miners Pick Transactions ?



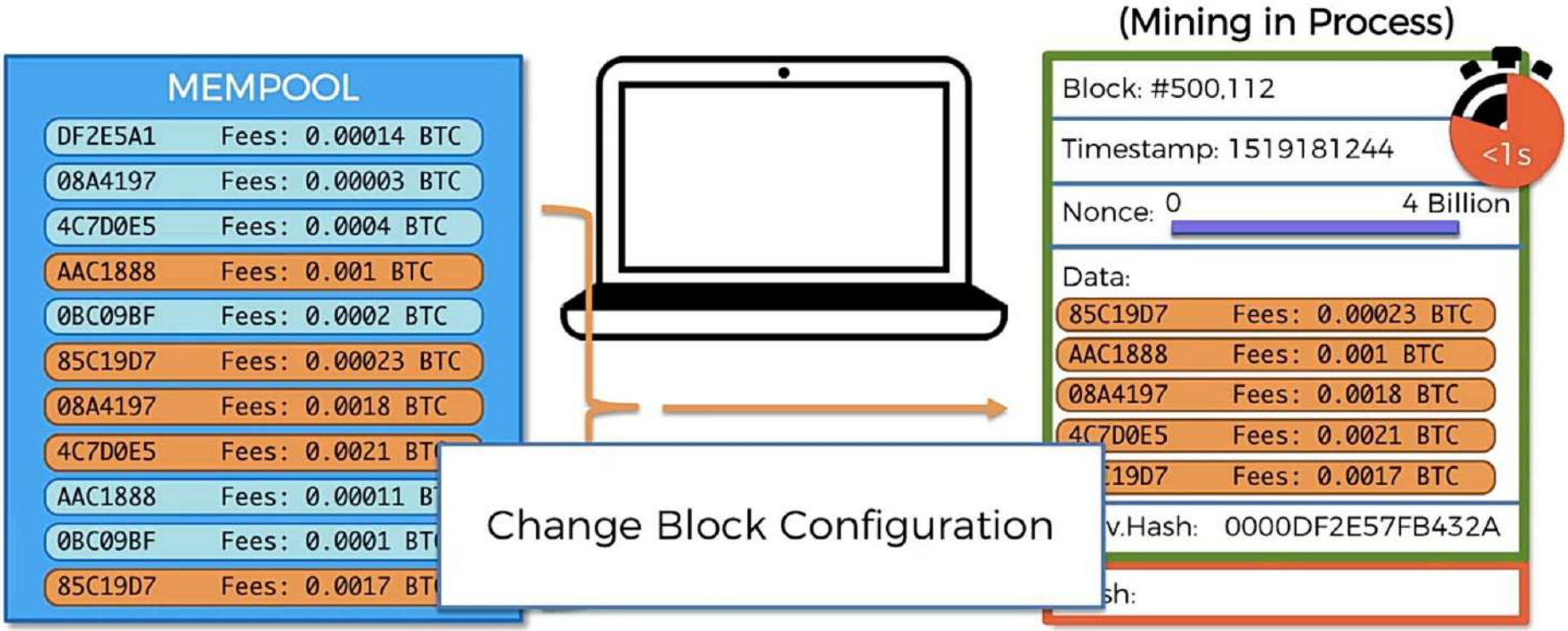
How Miners Pick Transactions ?



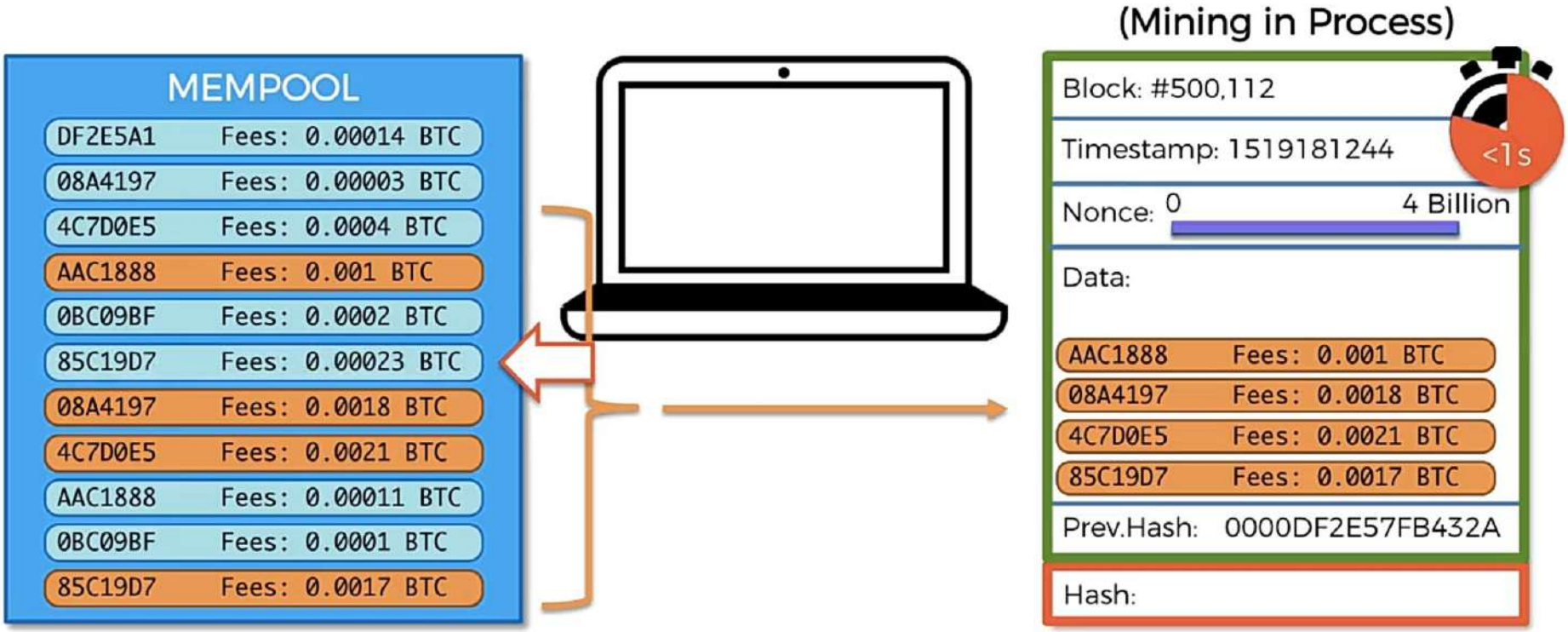
How Miners Pick Transactions ?



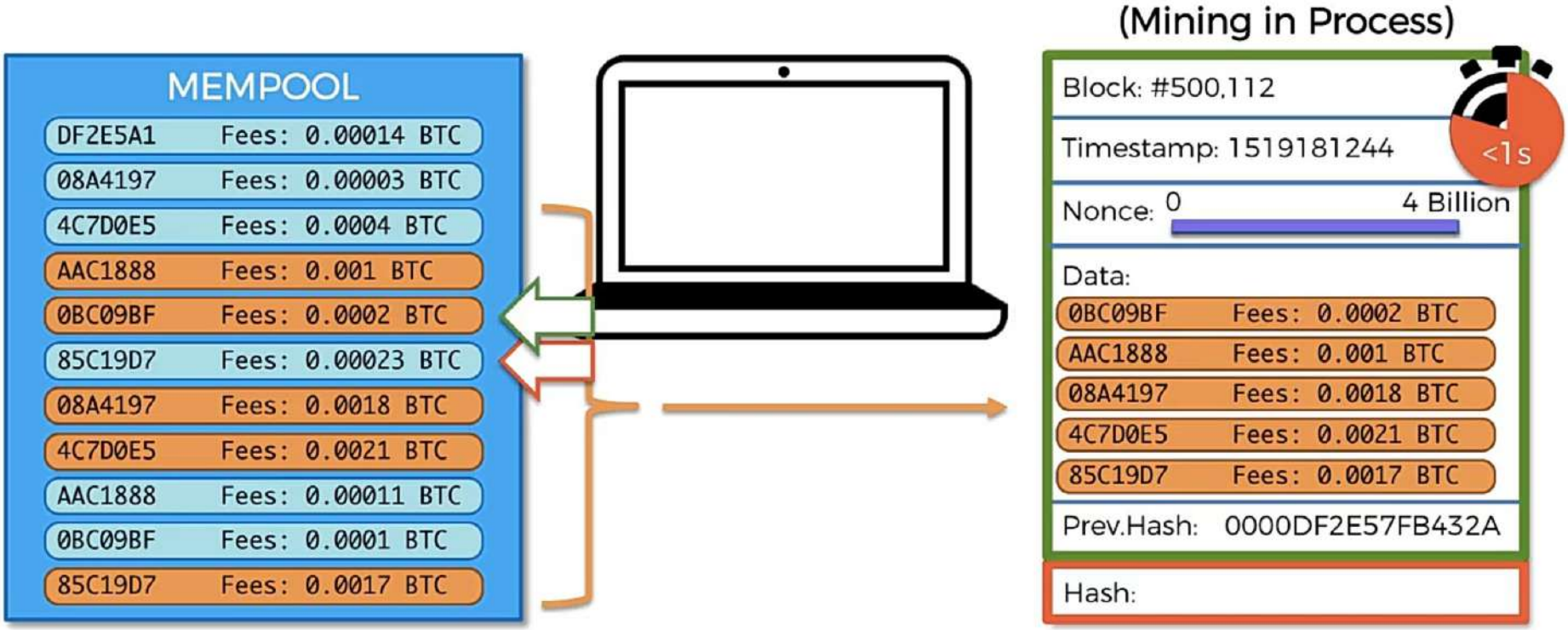
How Miners Pick Transactions ?



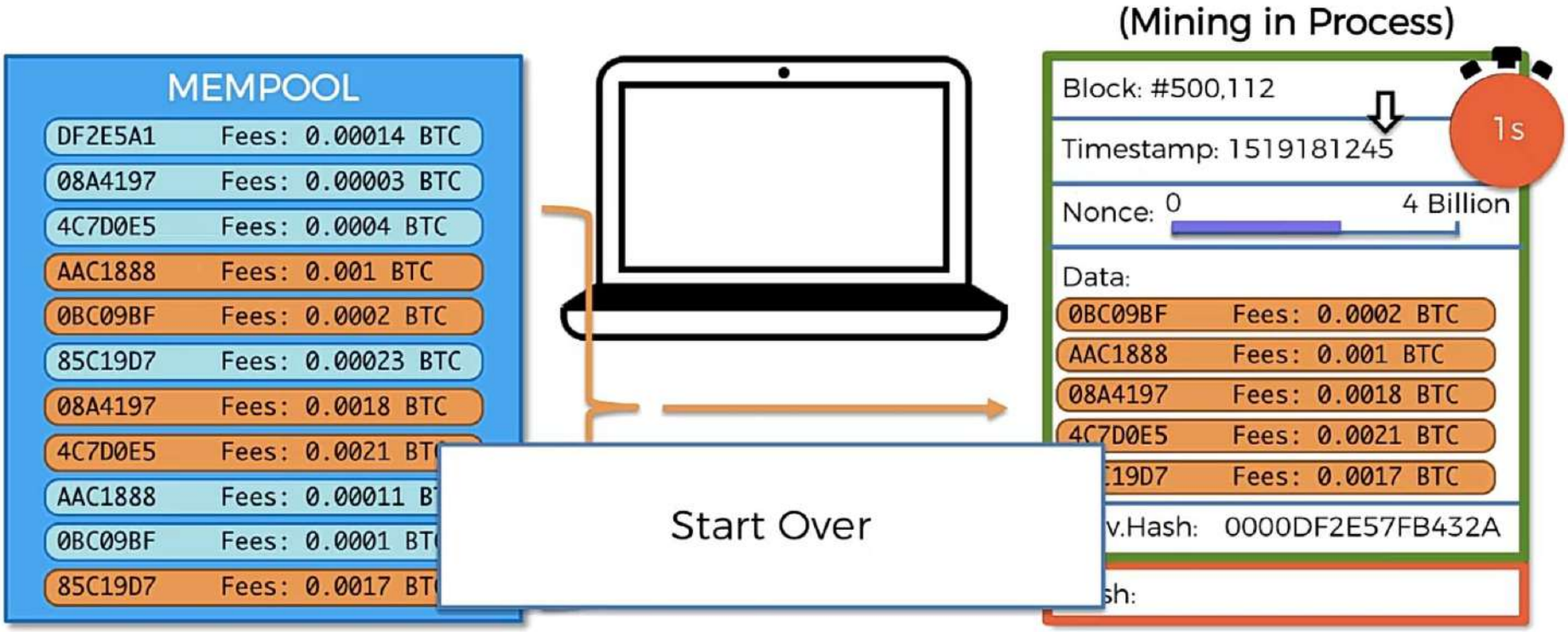
How Miners Pick Transactions ?



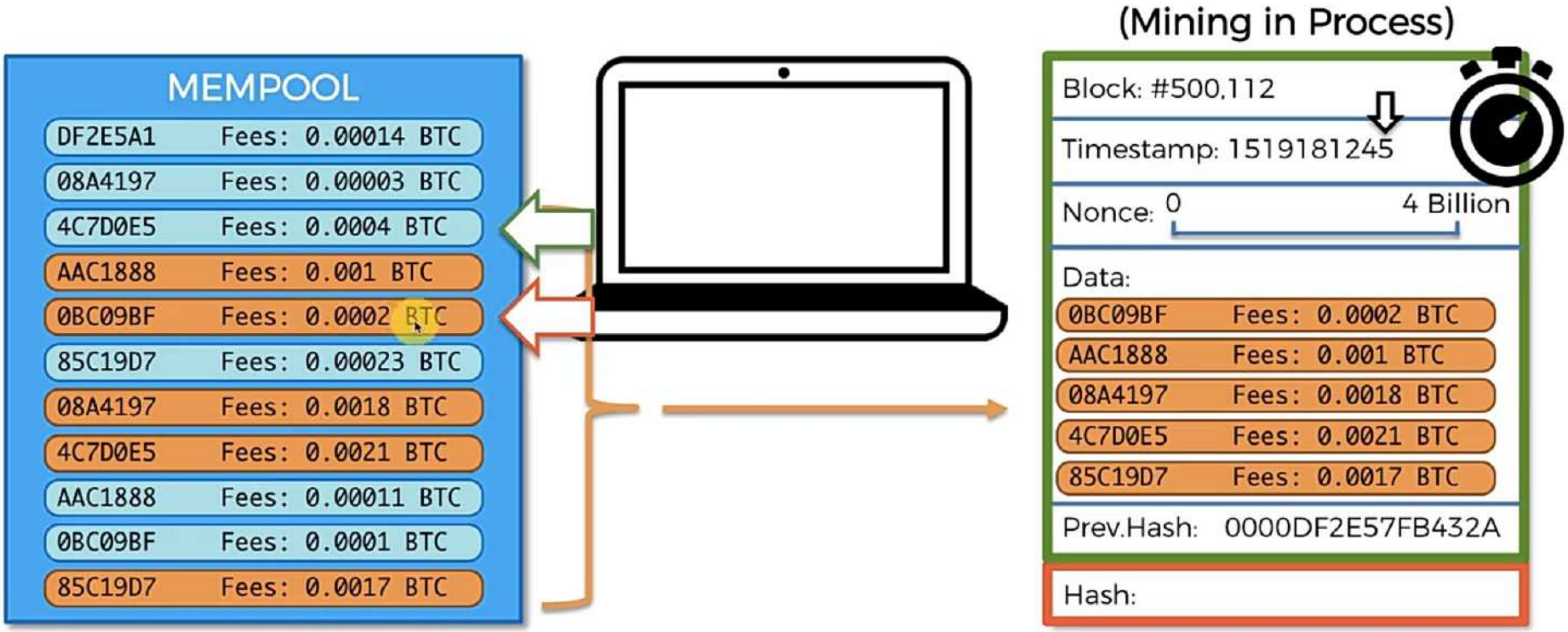
How Miners Pick Transactions ?



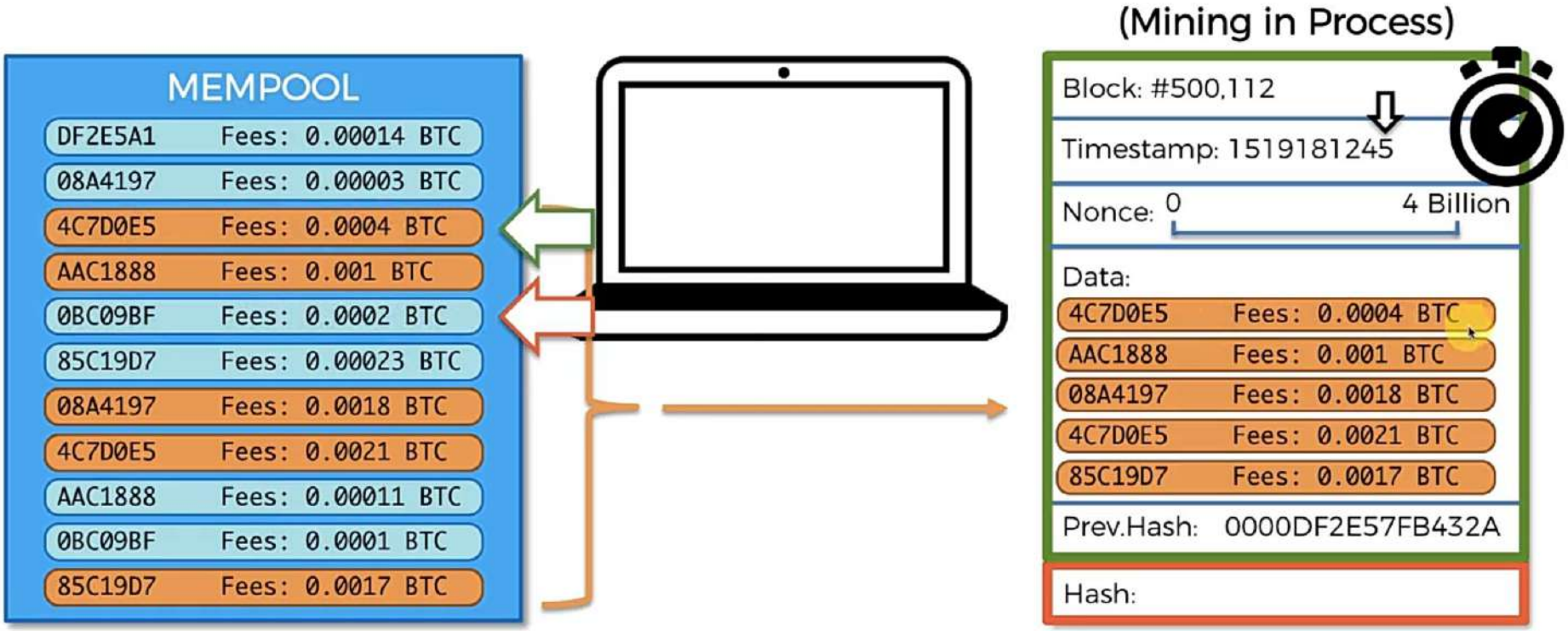
How Miners Pick Transactions ?



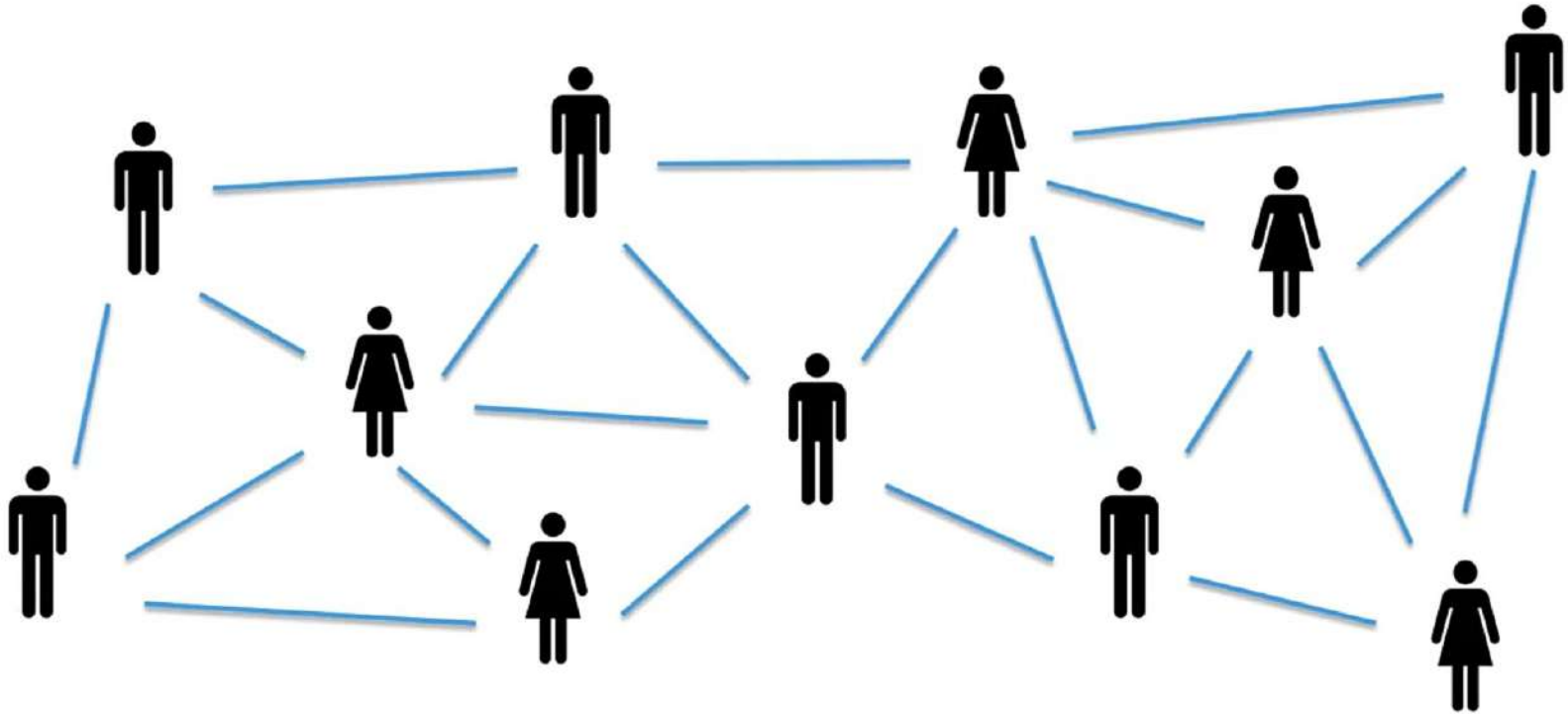
How Miners Pick Transactions ?



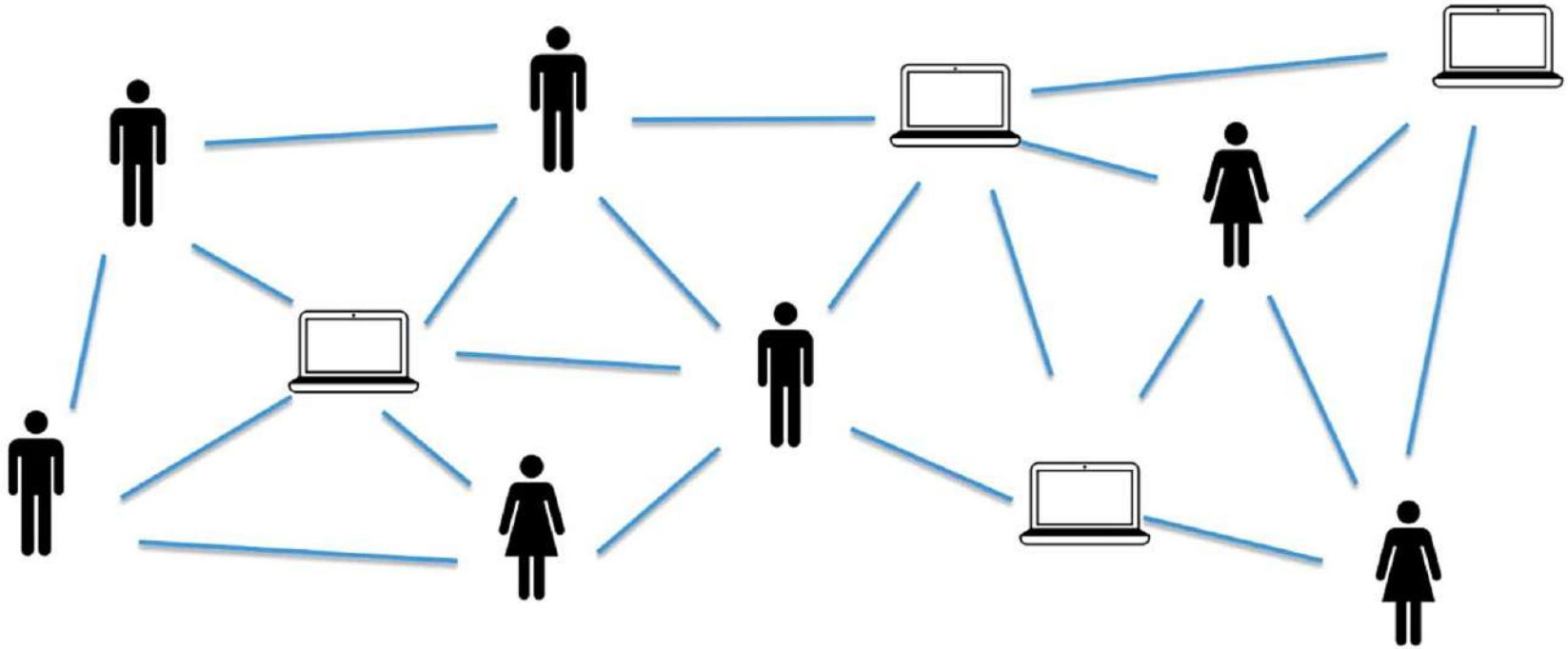
How Miners Pick Transactions ?



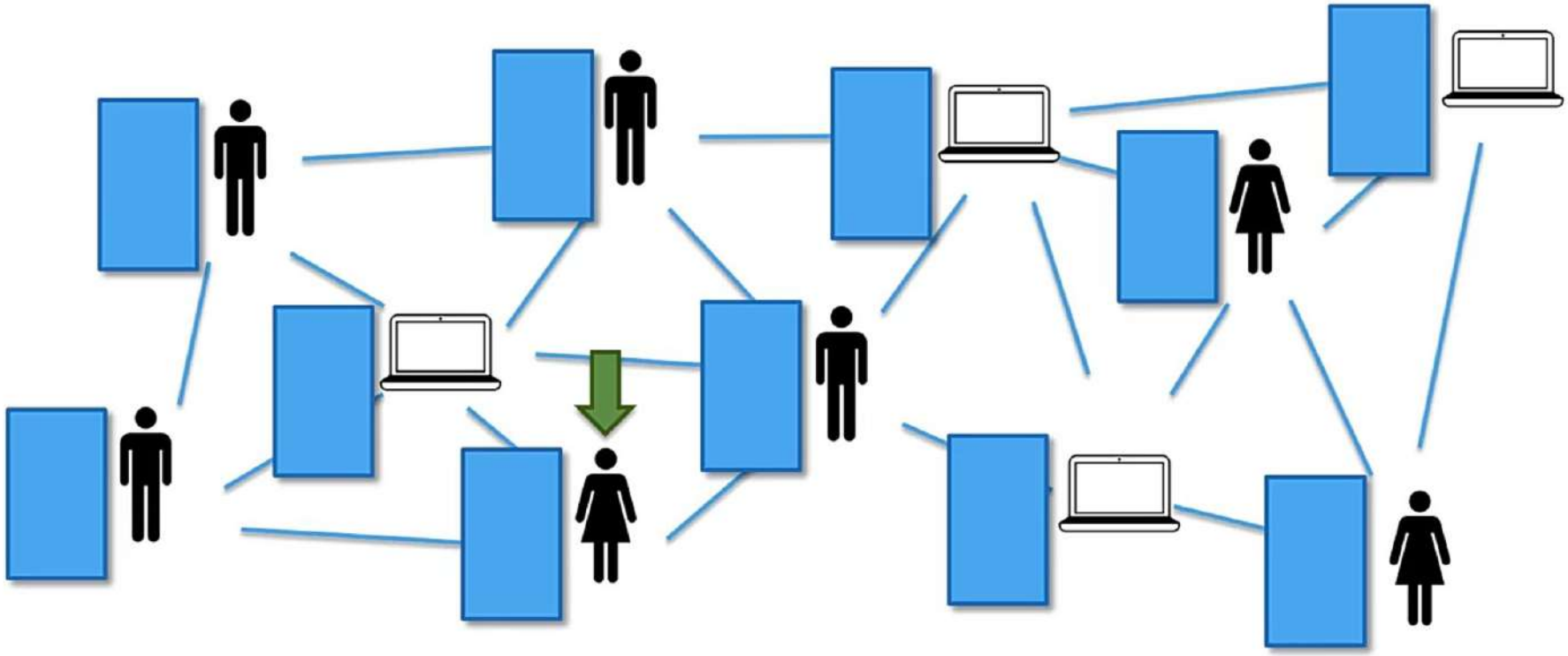
How do Mempools work?



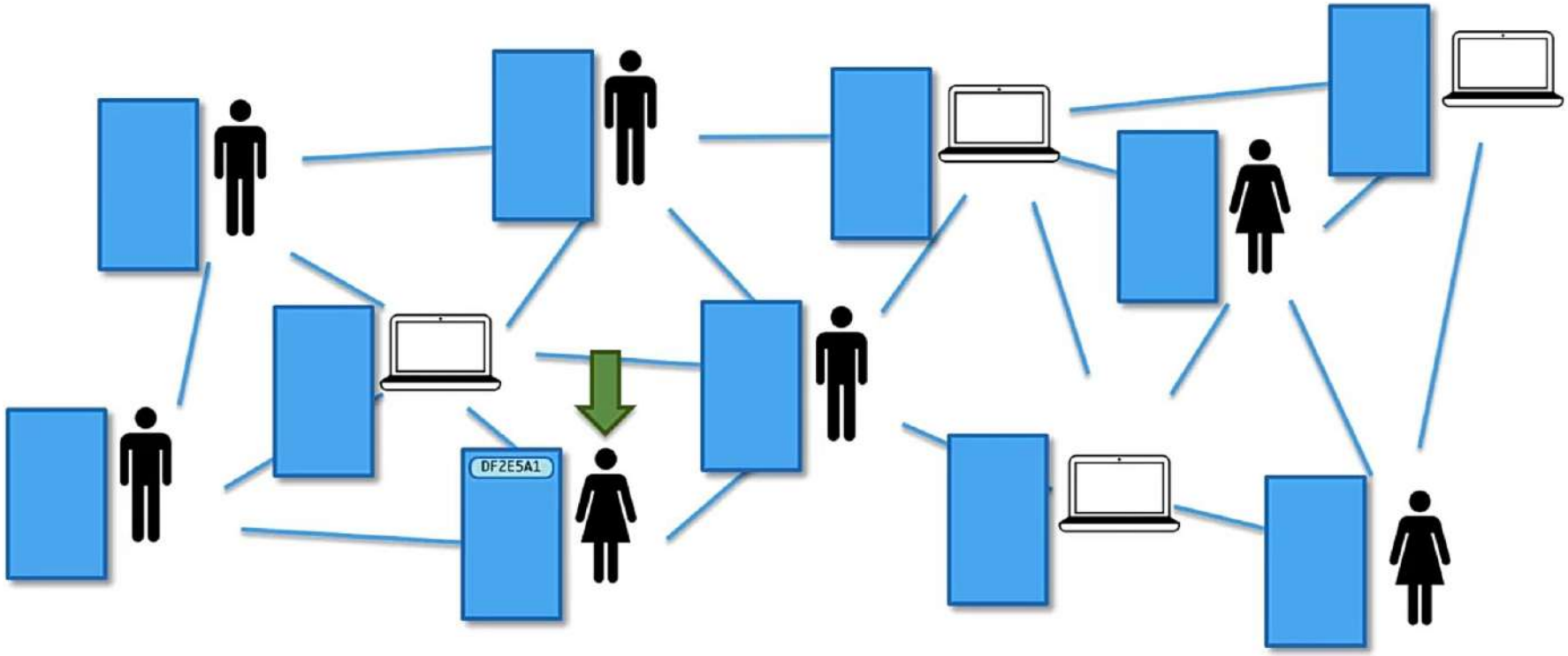
How do Mempools work?



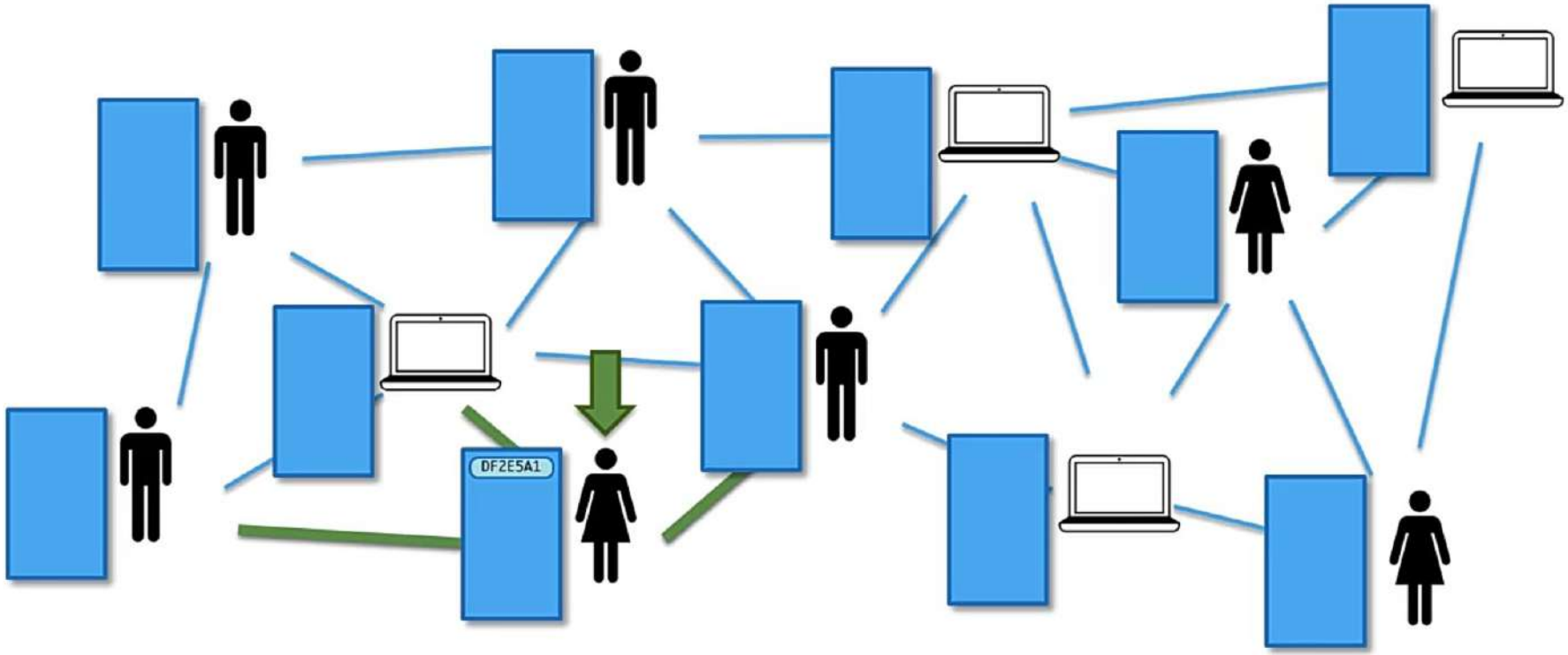
How do Mempools work?



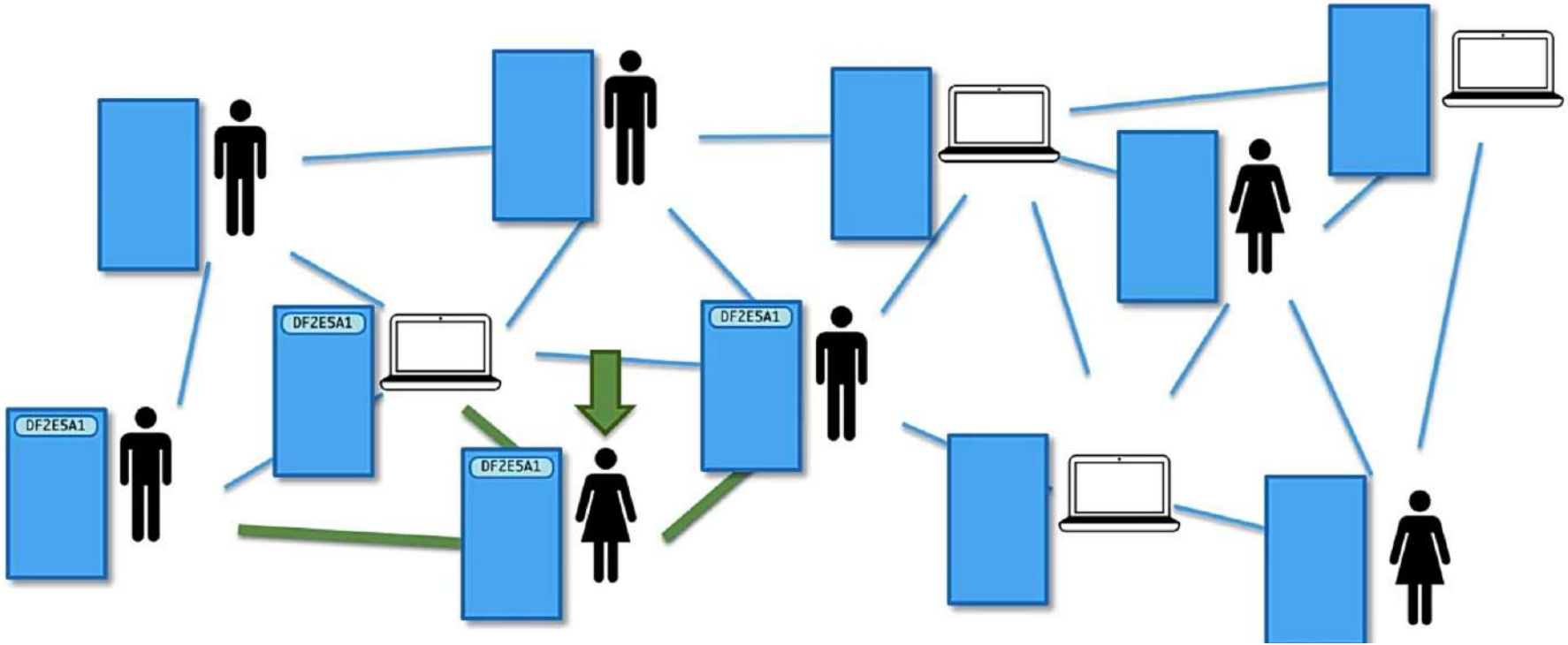
How do Mempools work?



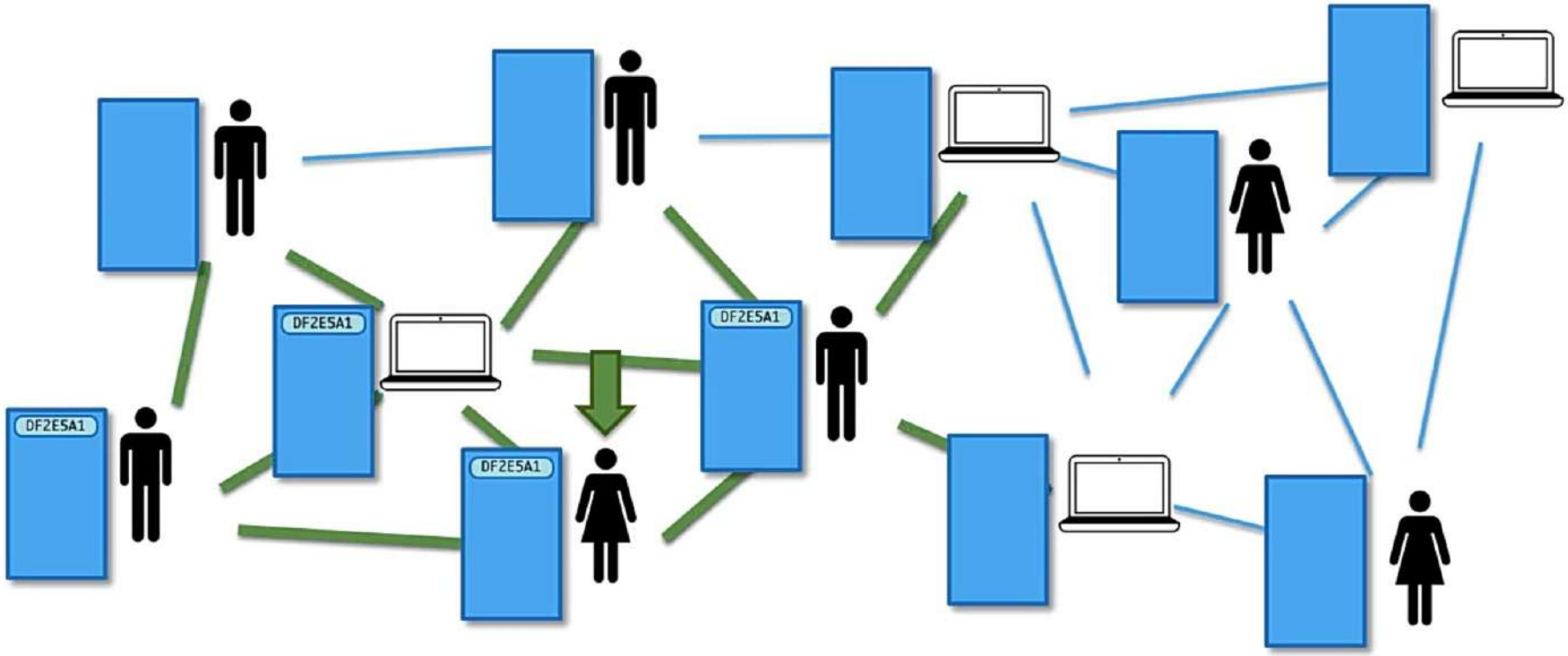
How do Mempools work?



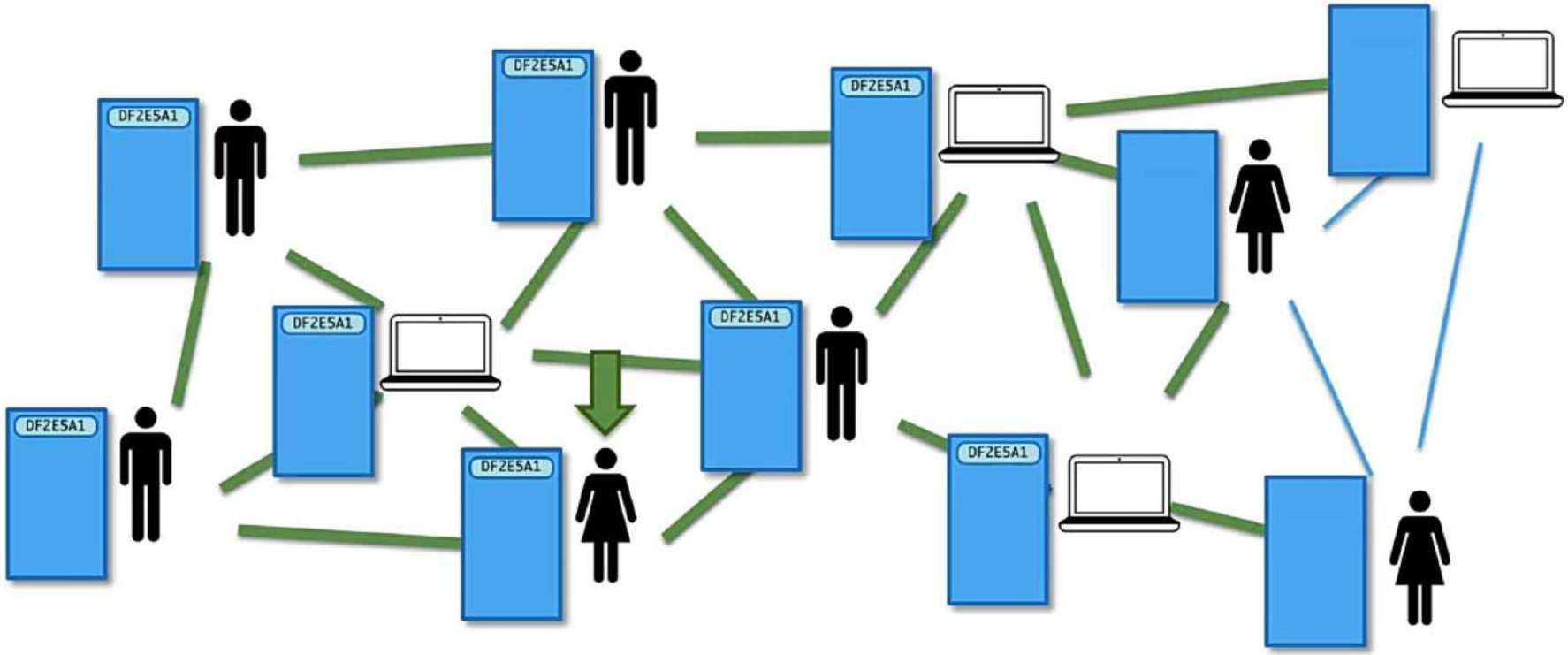
How do Mempools work?



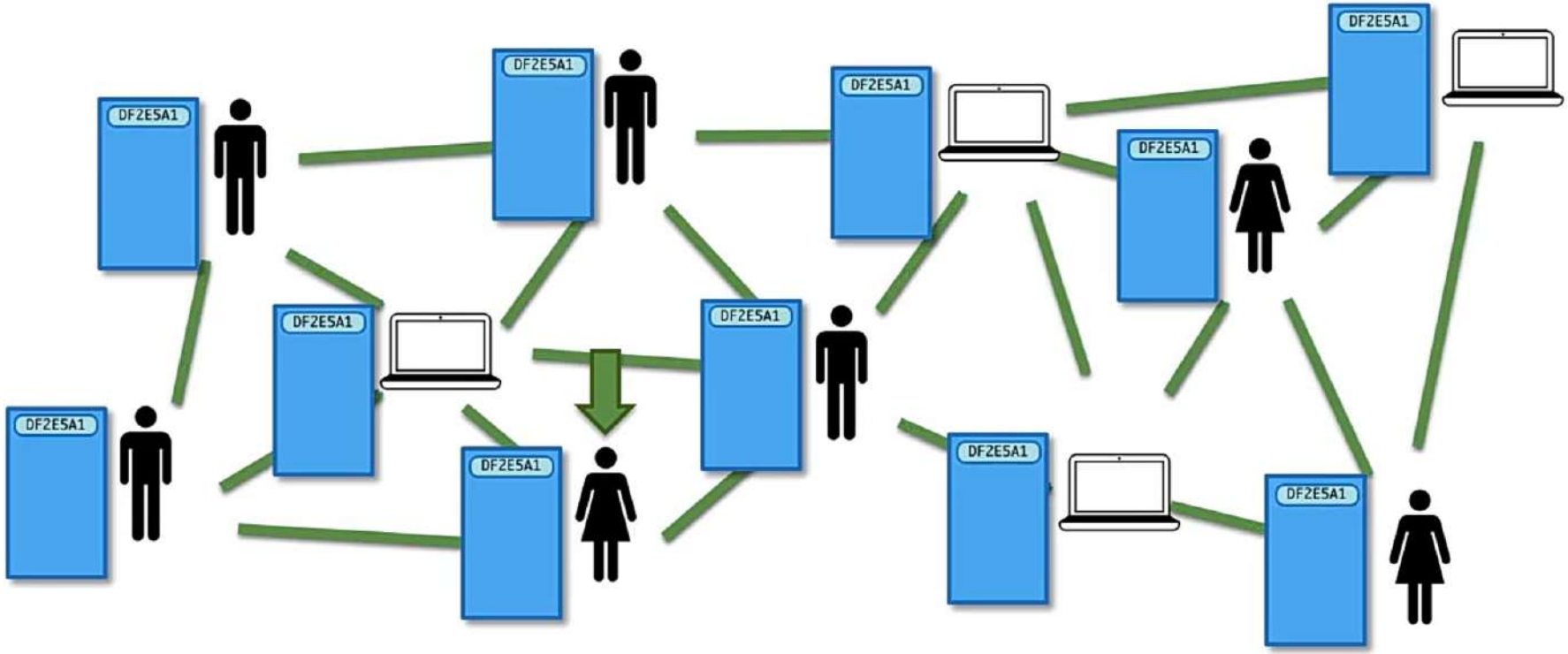
How do Mempools work?



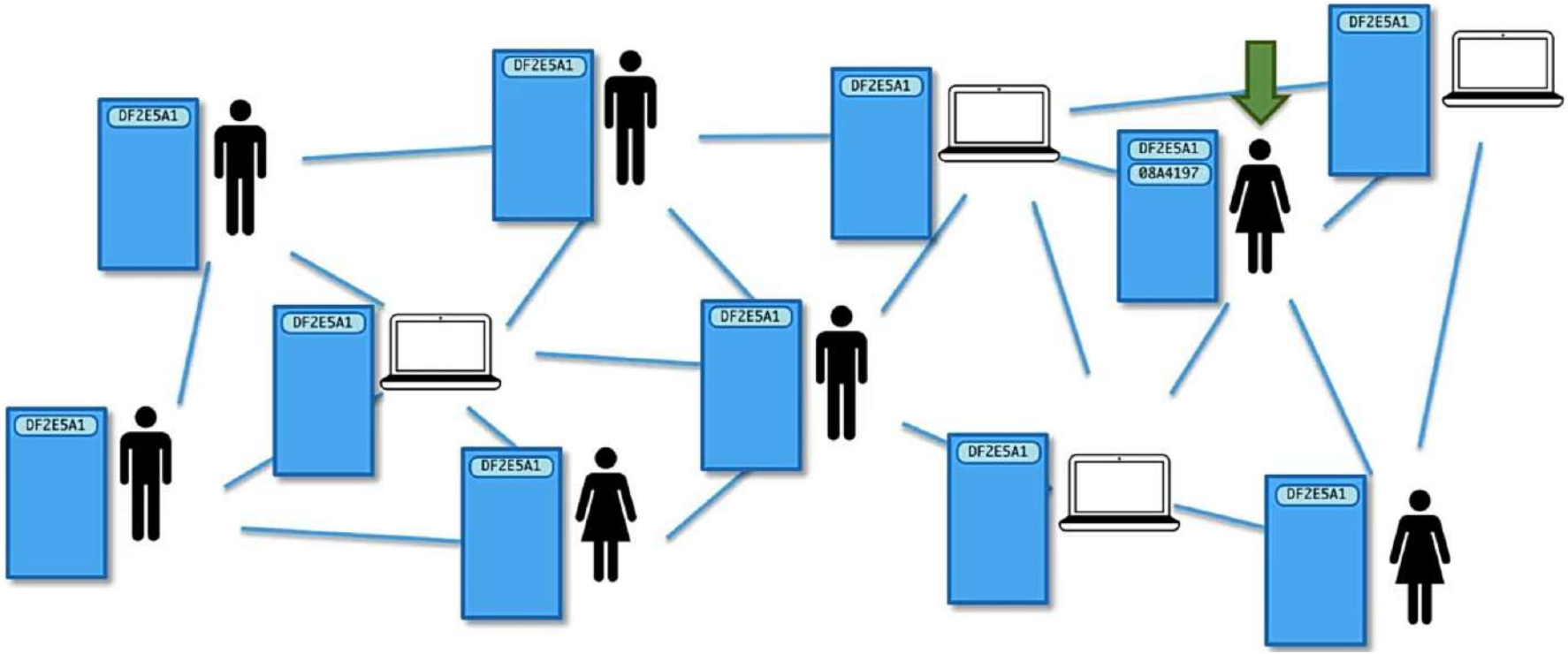
How do Mempools work?



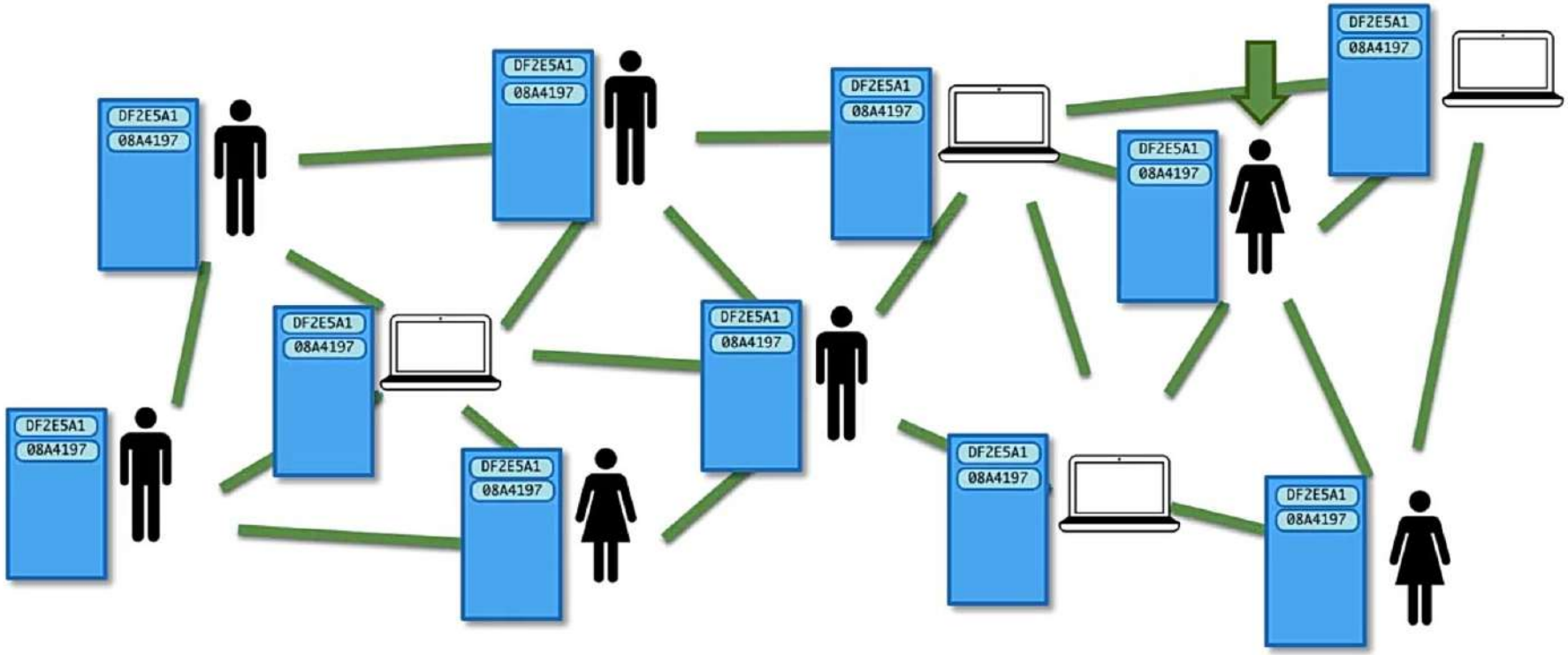
How do Mempools work?



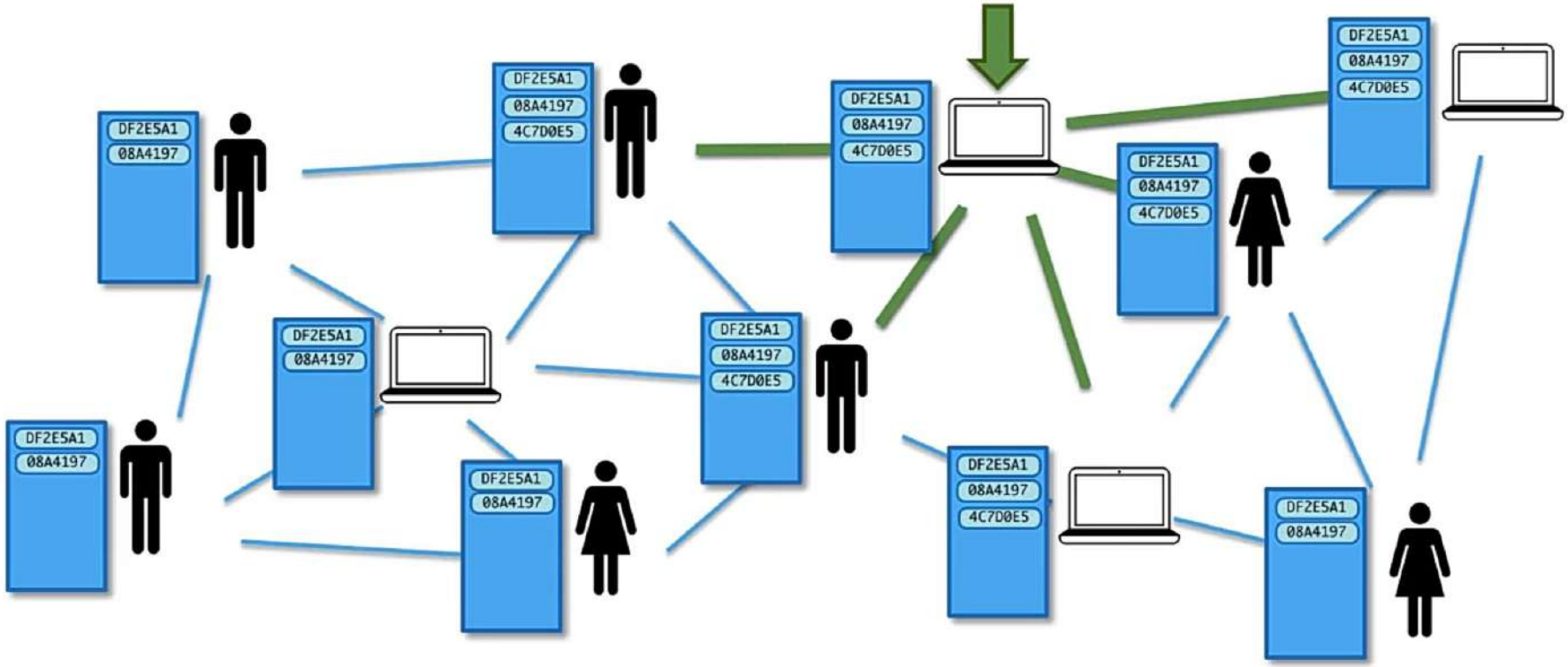
How do Mempools work?



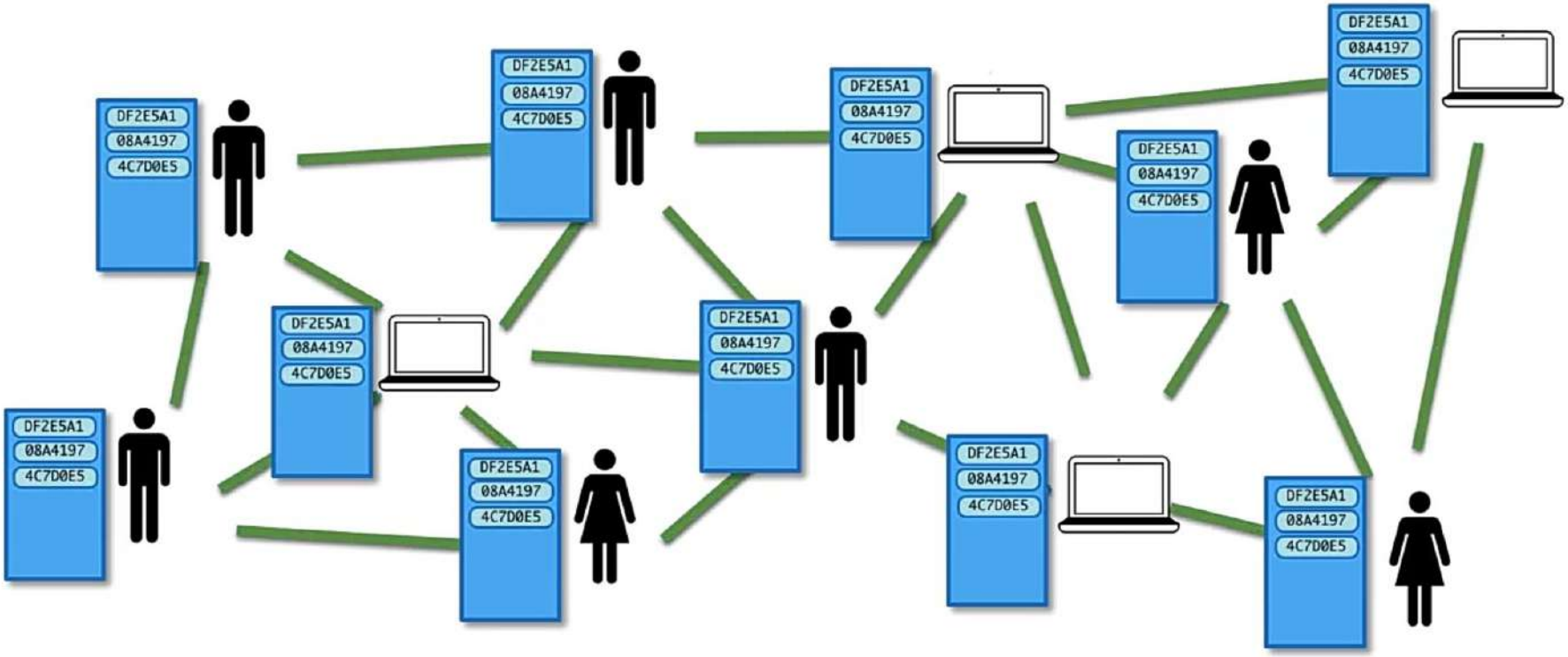
How do Mempools work?



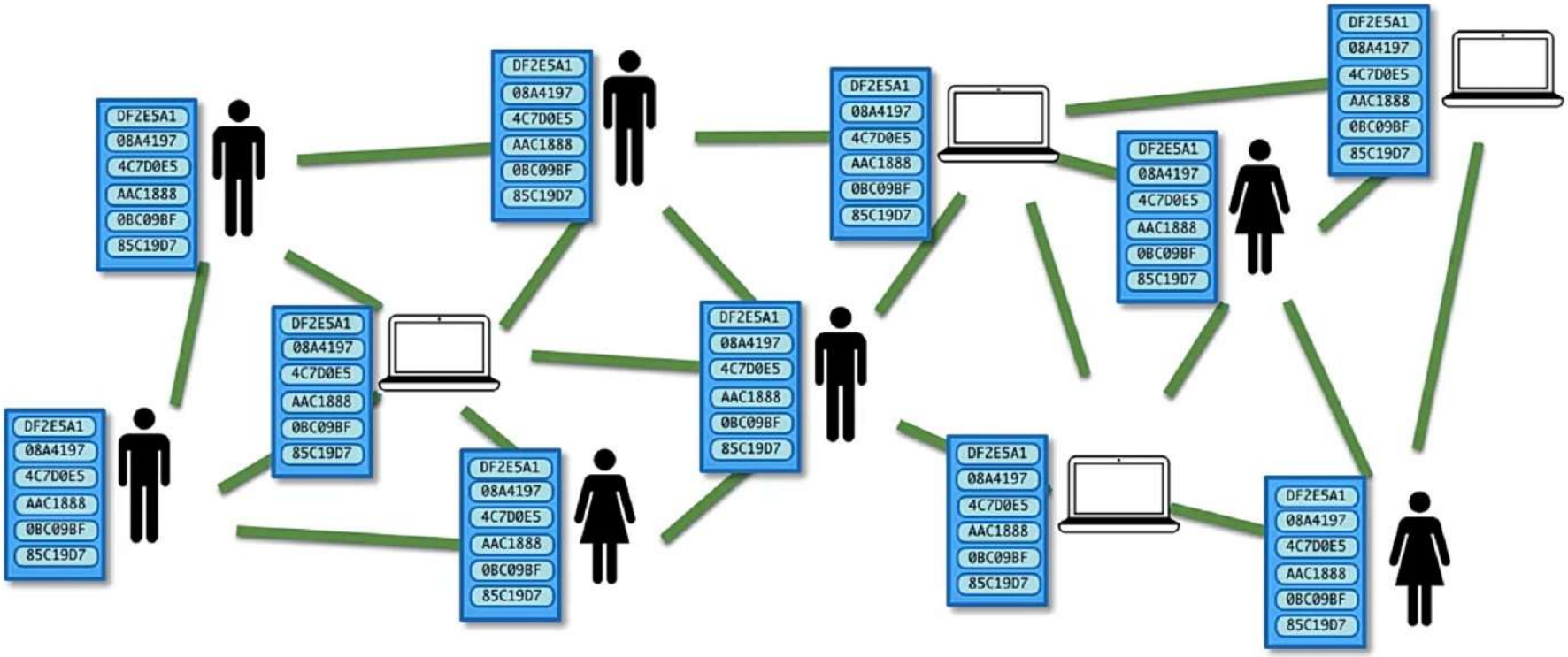
How do Mempools work?



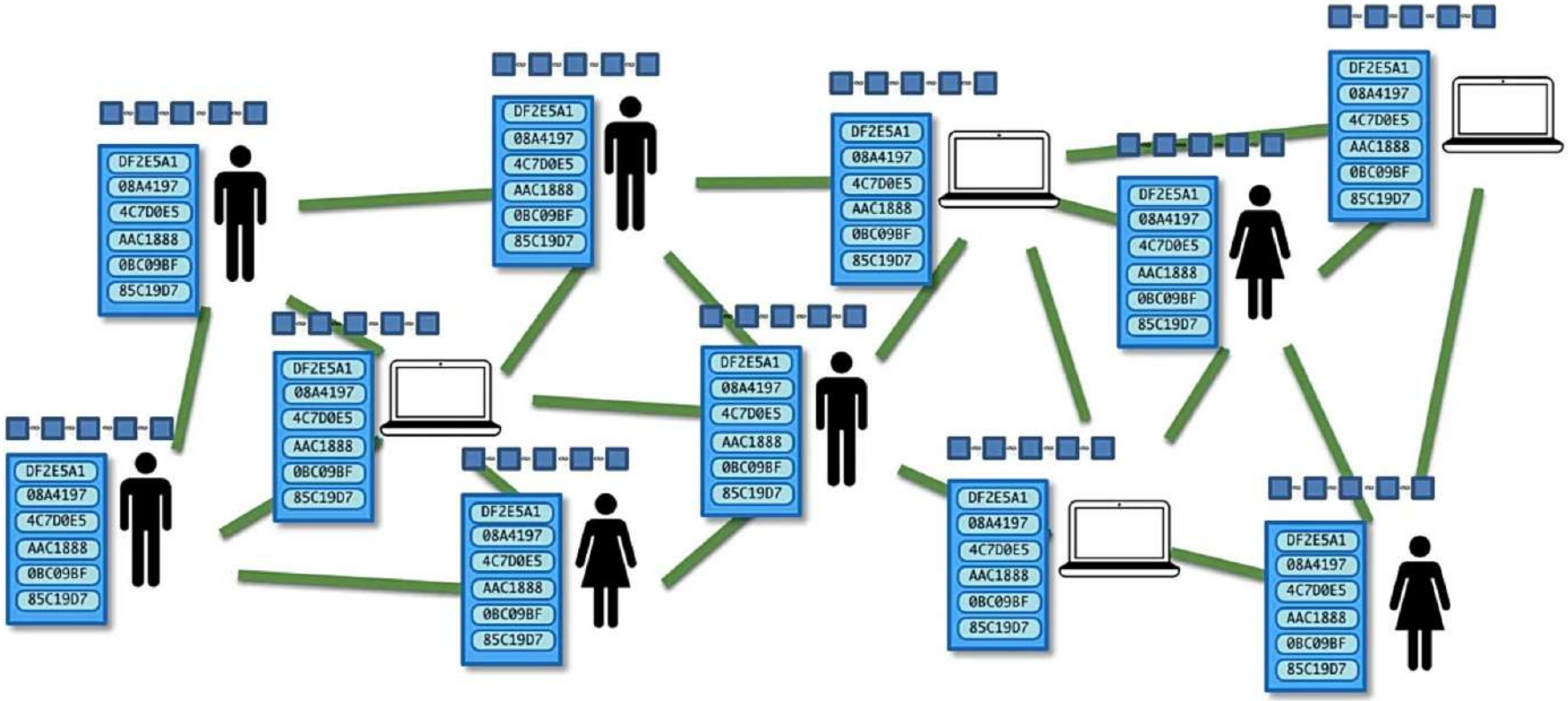
How do Mempools work?



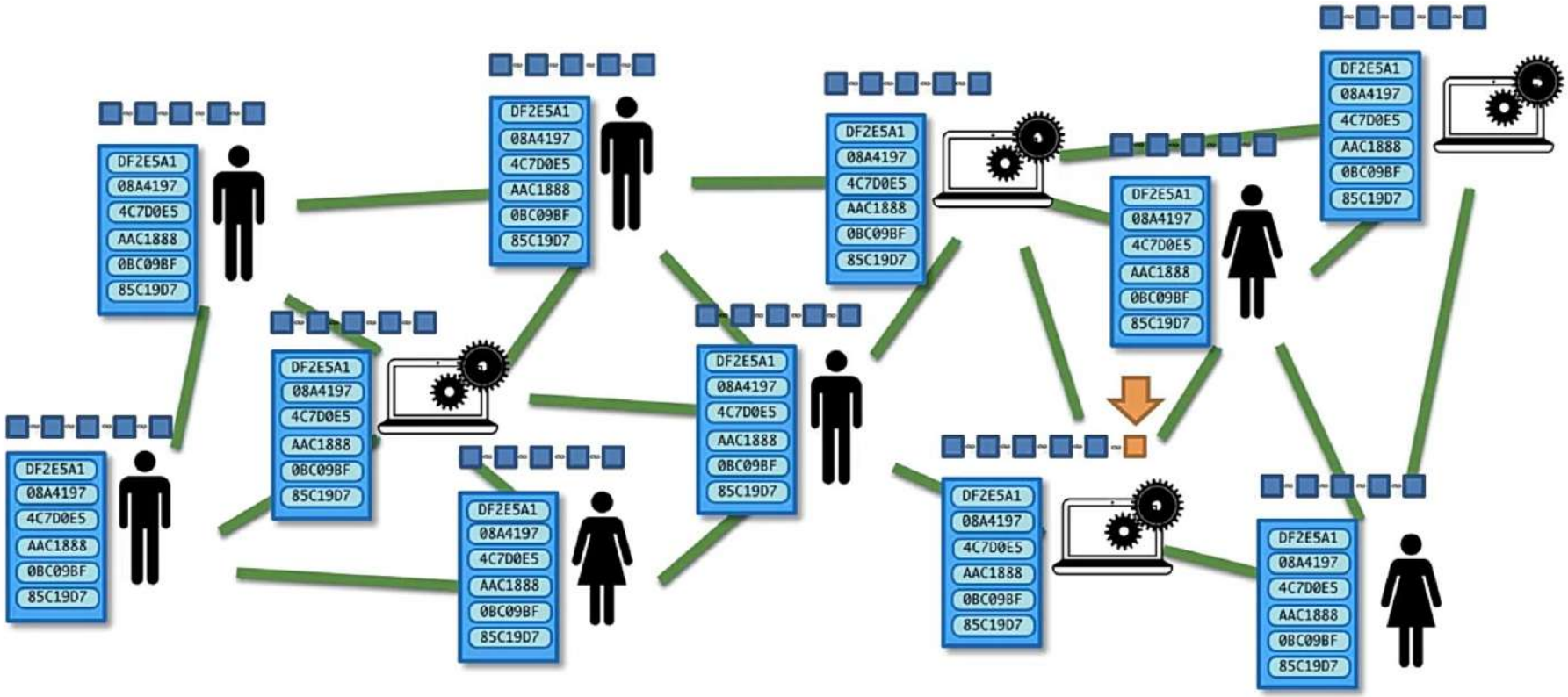
How do Mempools work?



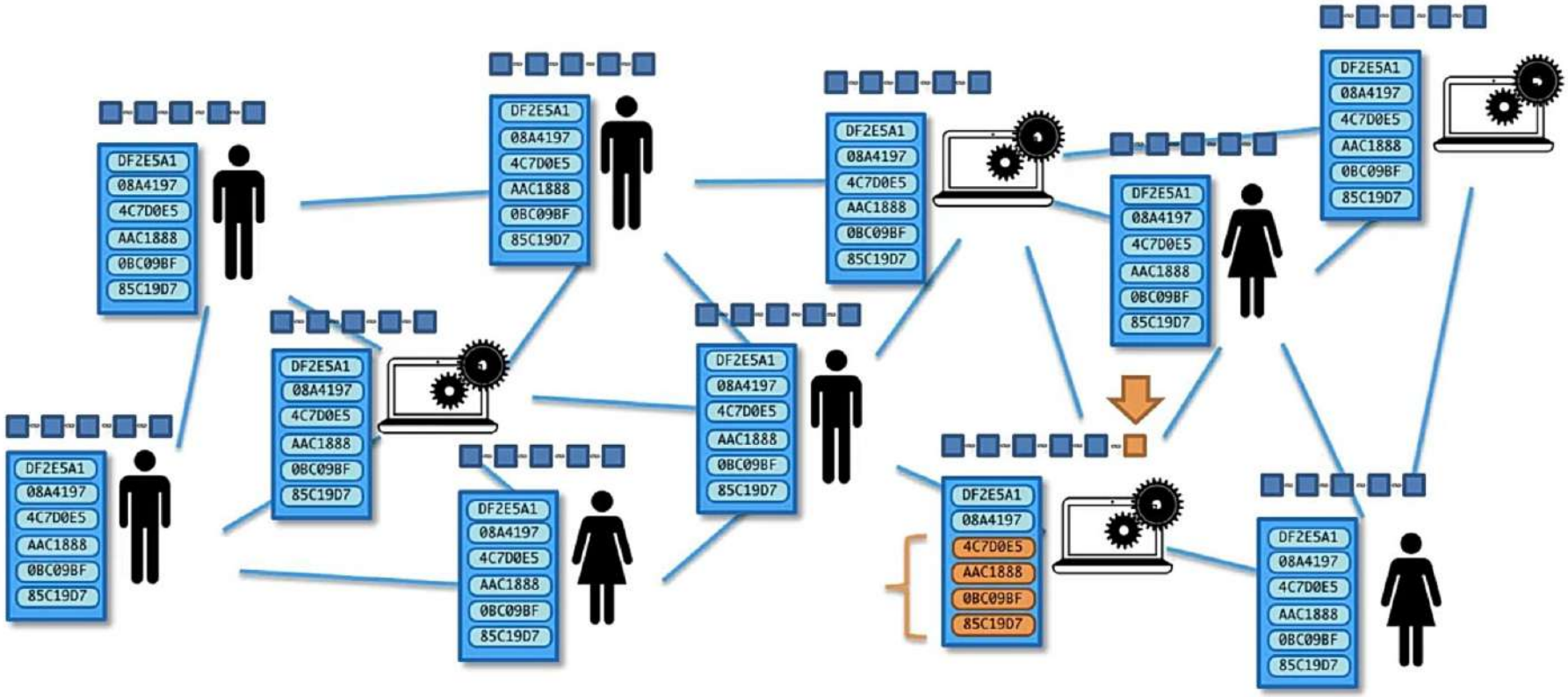
How do Mempools work?



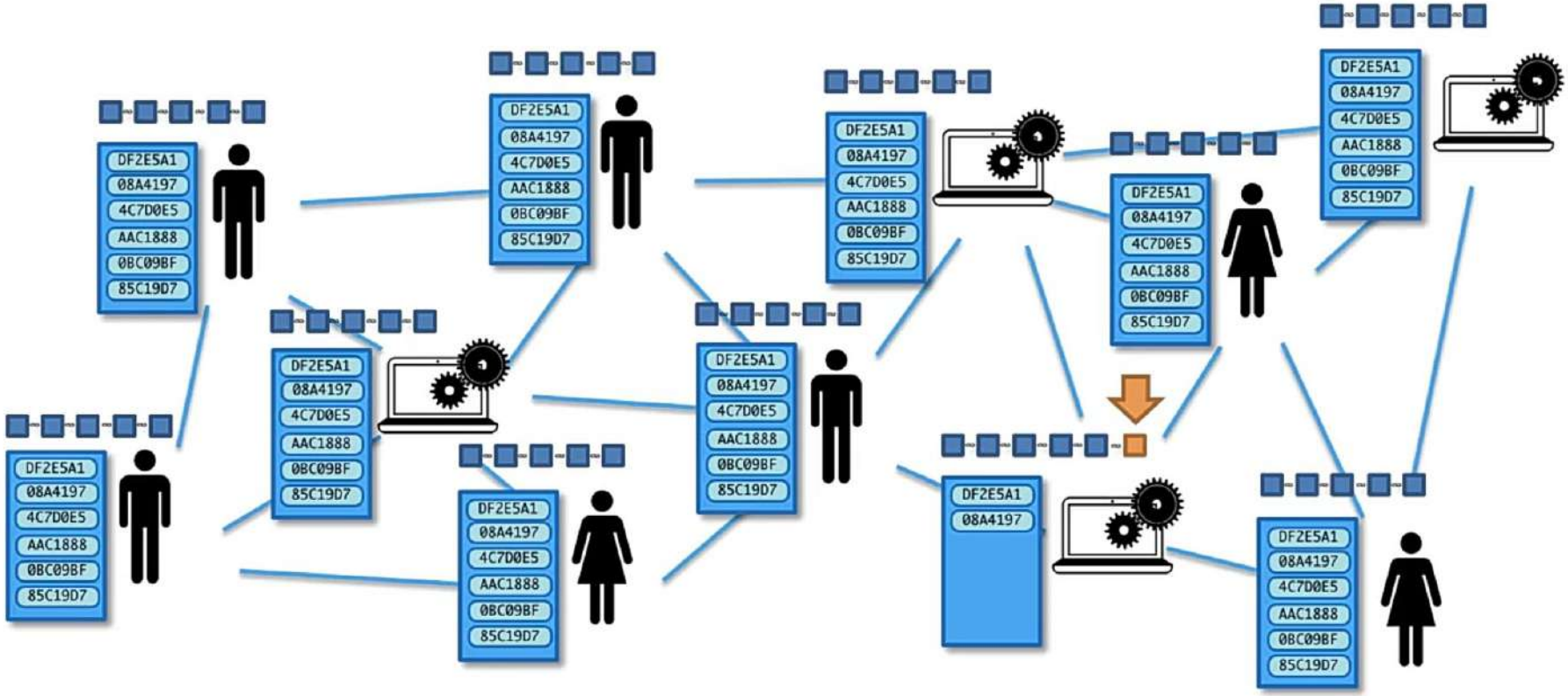
How do Mempools work?



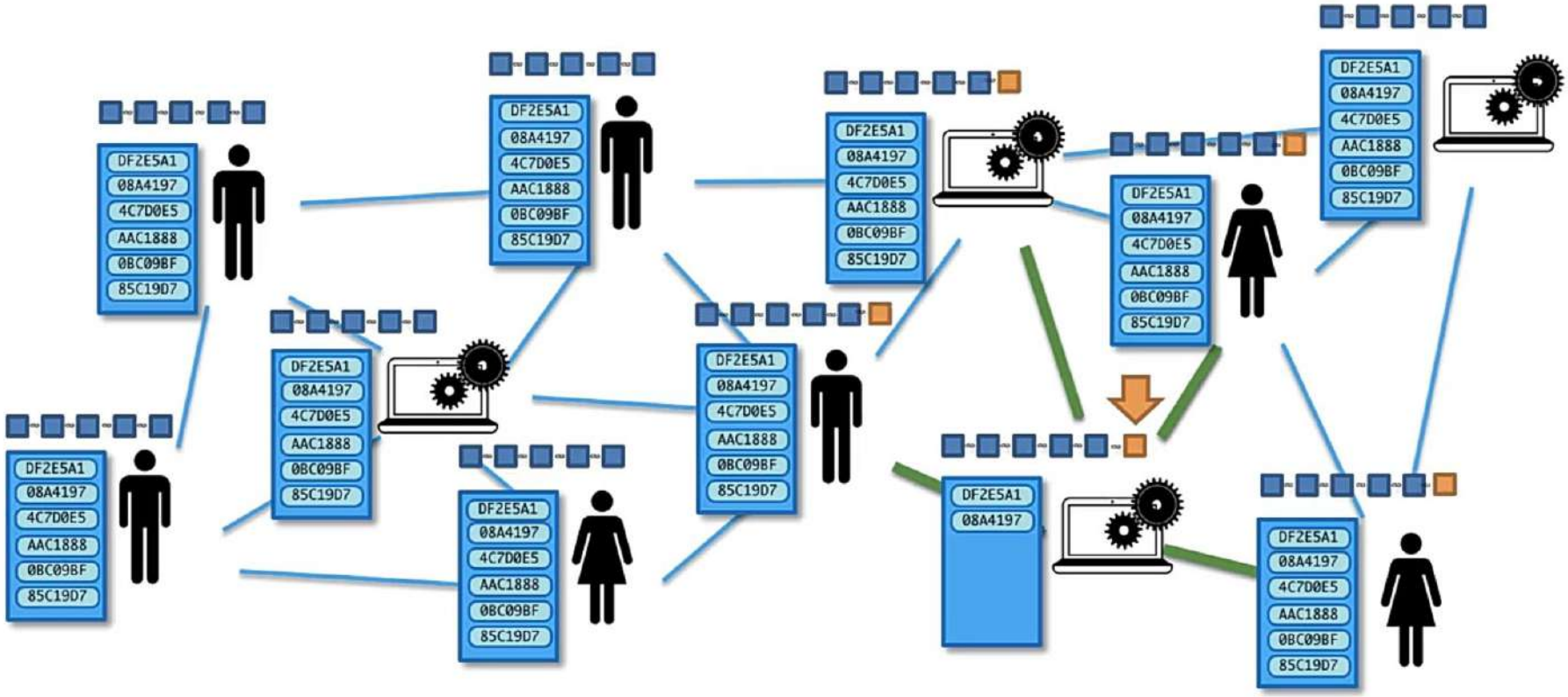
How do Mempools work?



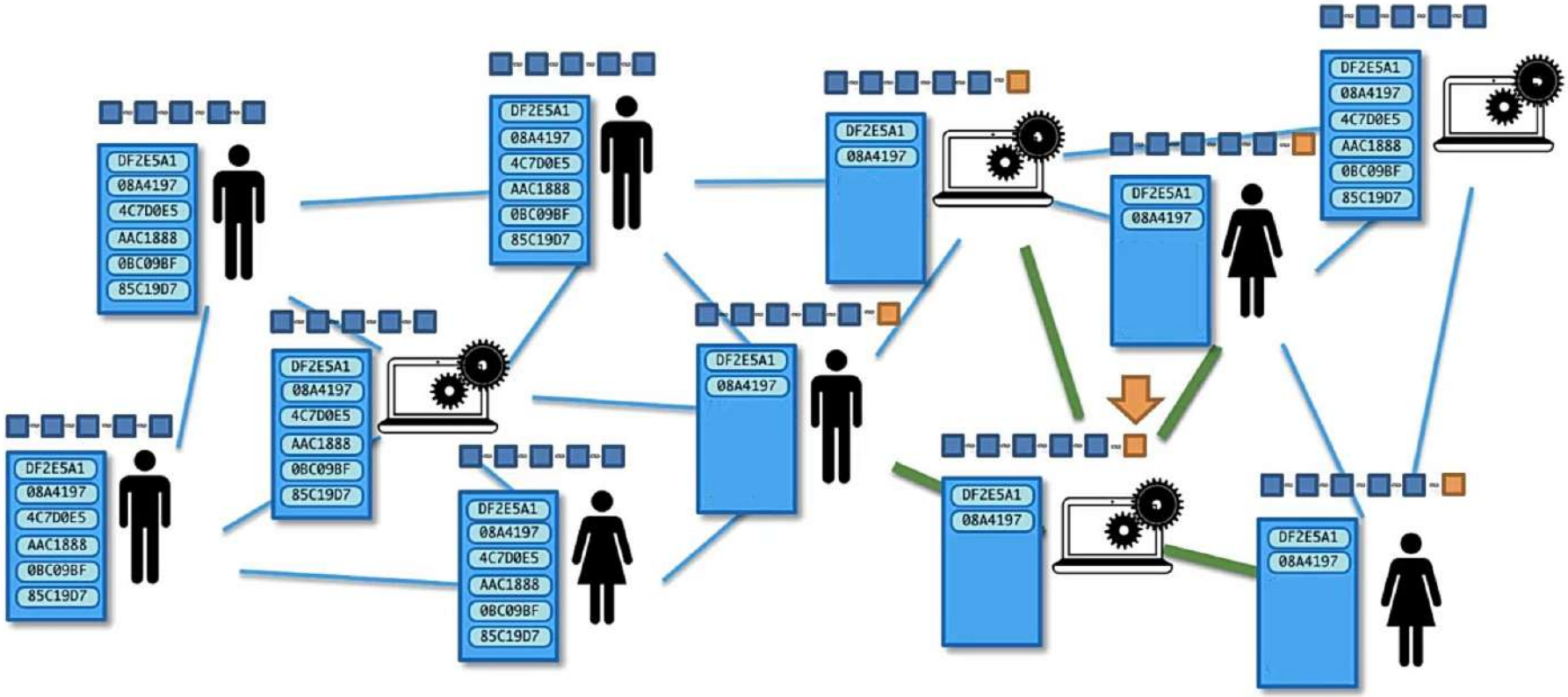
How do Mempools work?



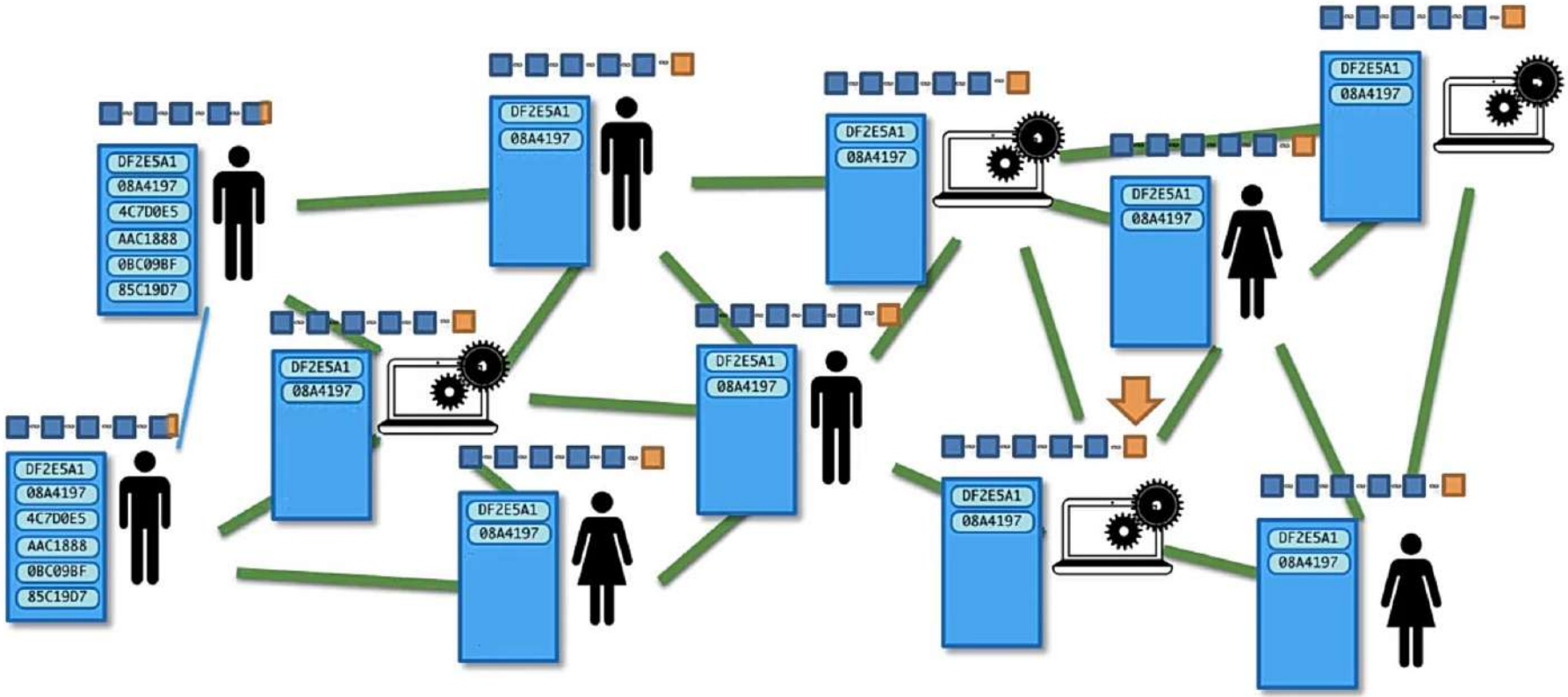
How do Mempools work?



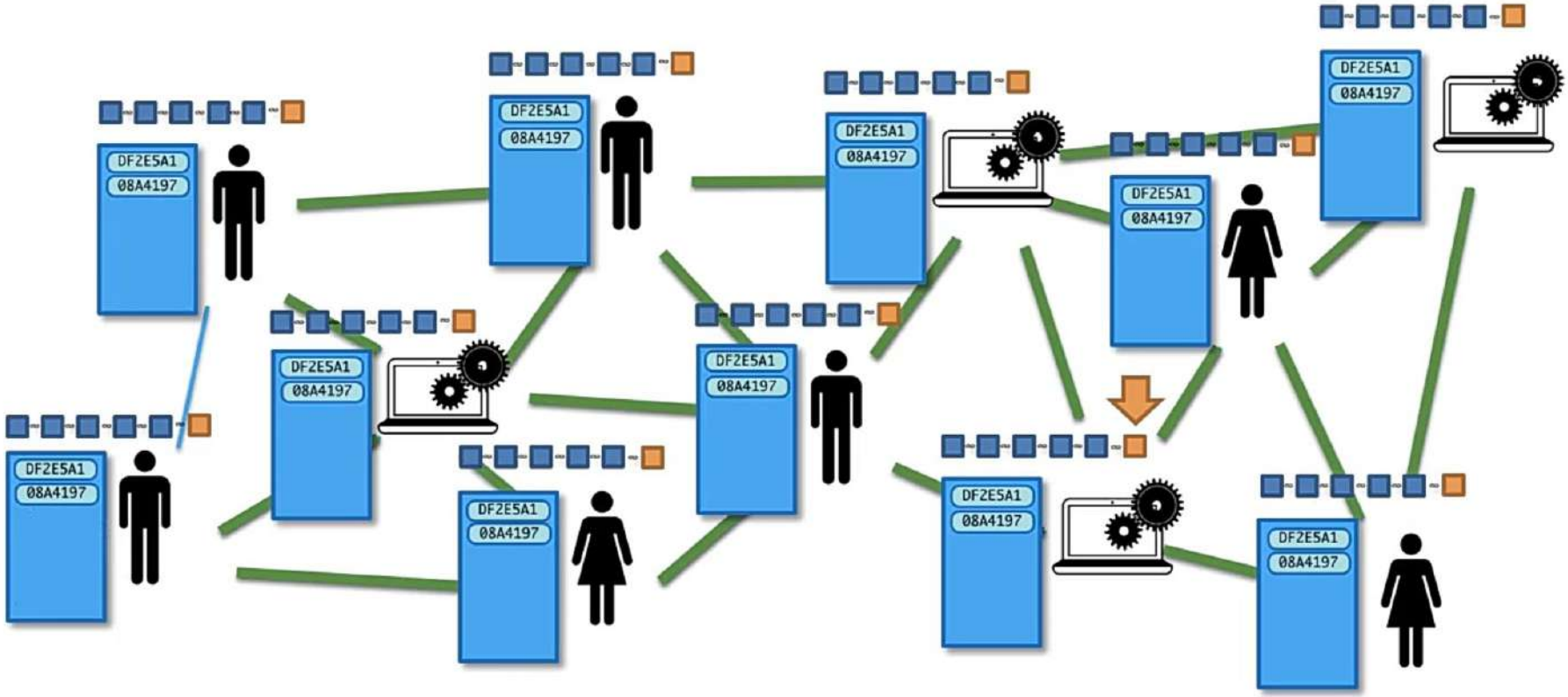
How do Mempools work?



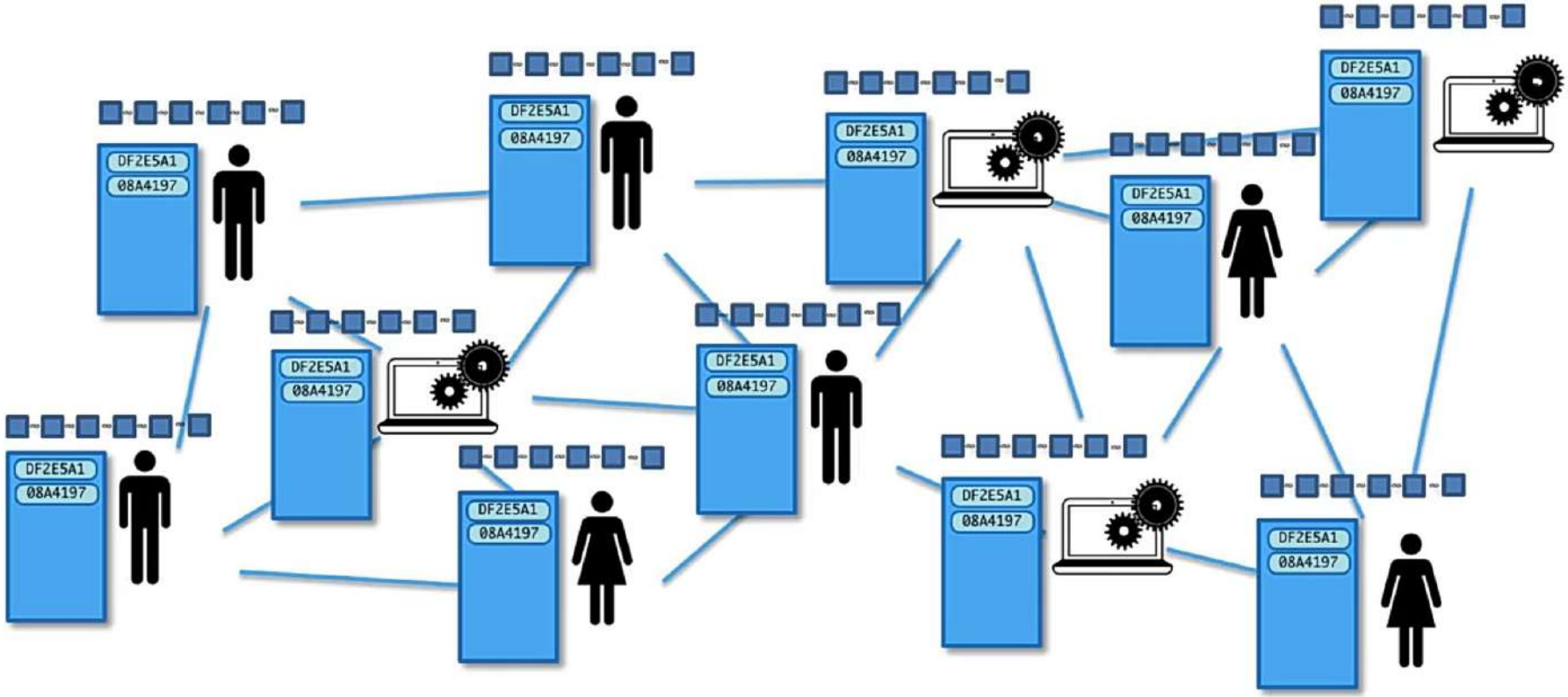
How do Mempools work?



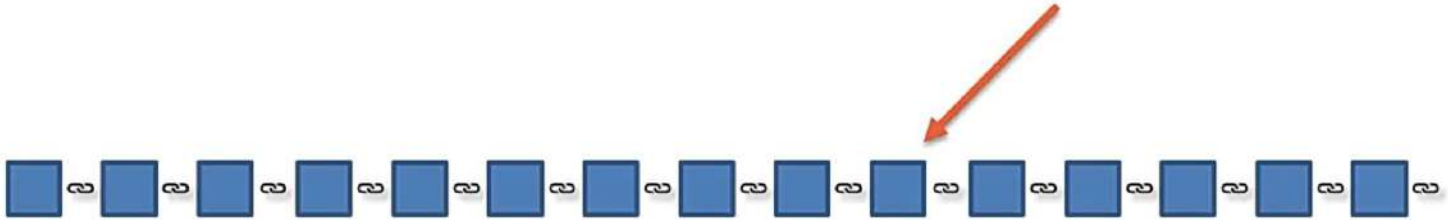
How do Mempools work?



How do Mempools work?

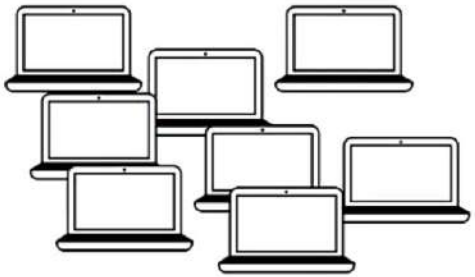


51% Attack

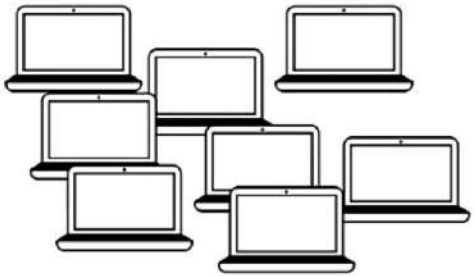


This is NOT the 51% attack

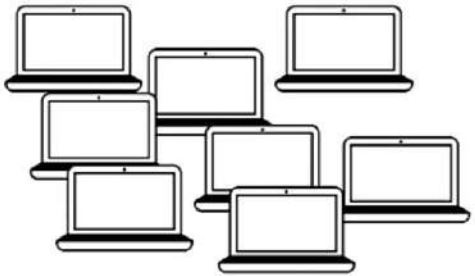
51% Attack

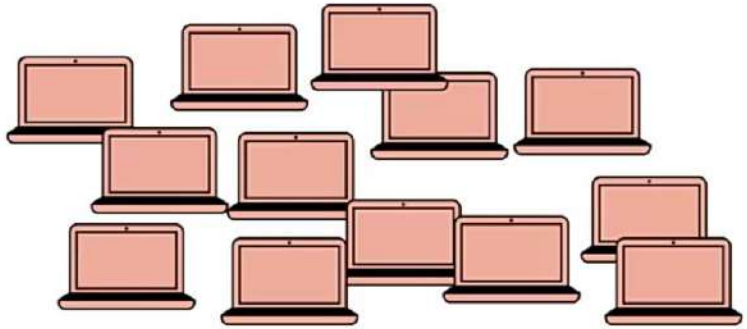


51% Attack

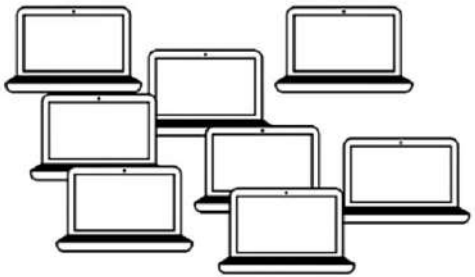


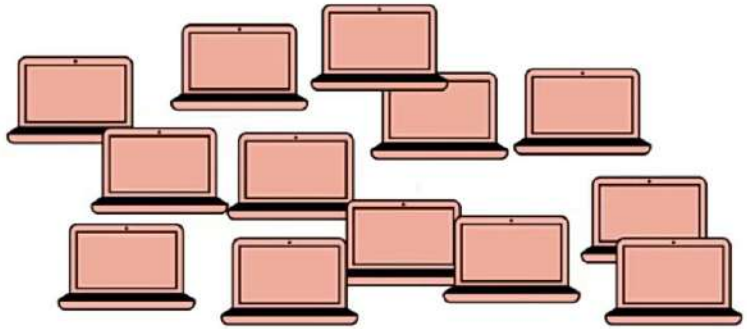
51% Attack



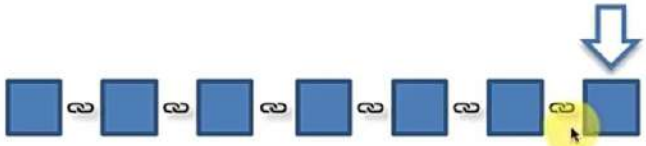


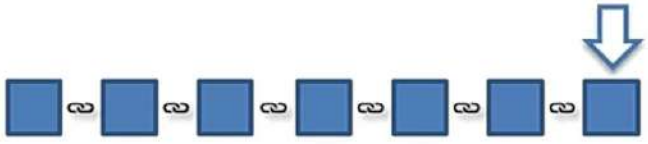
51% Attack

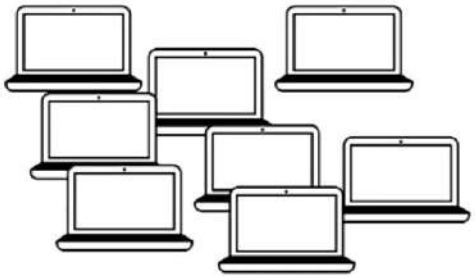


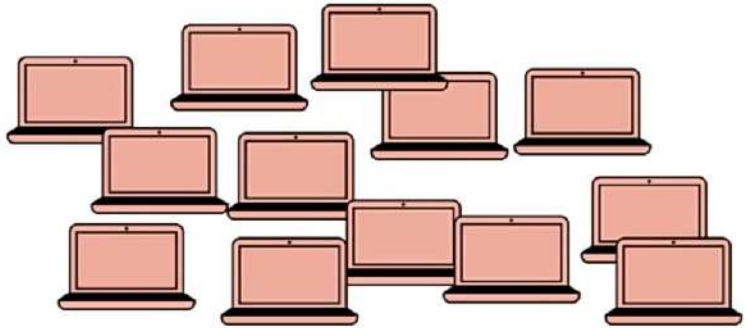


51% Attack

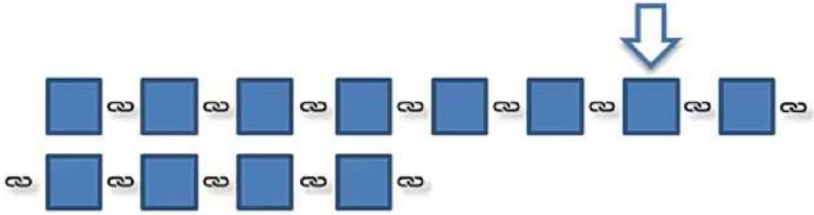


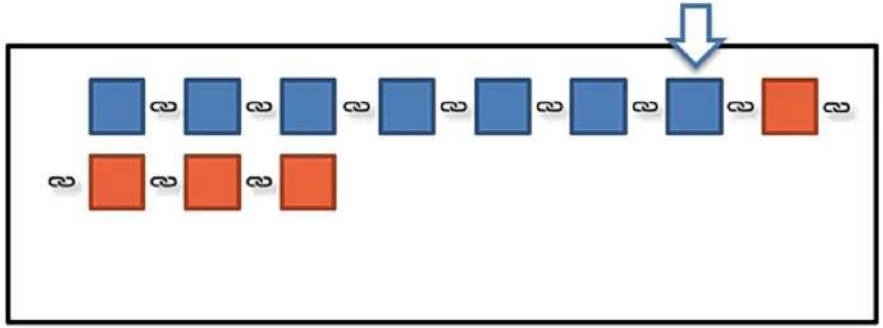


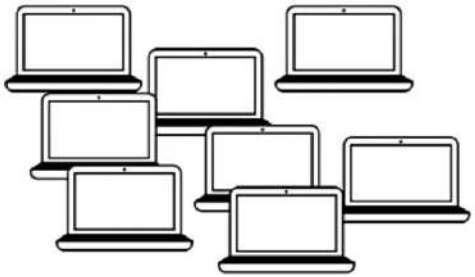


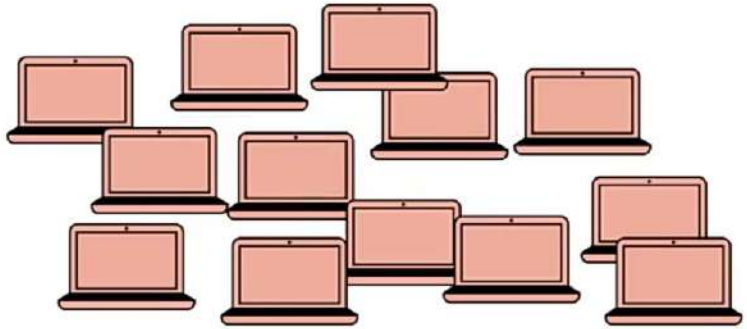


51% Attack

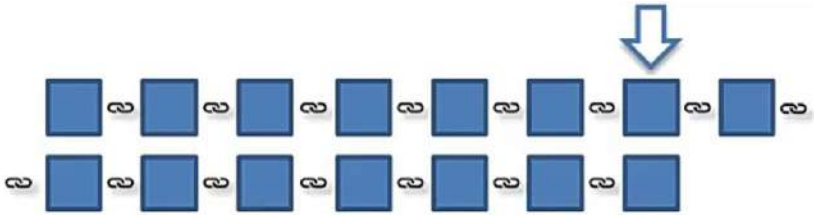


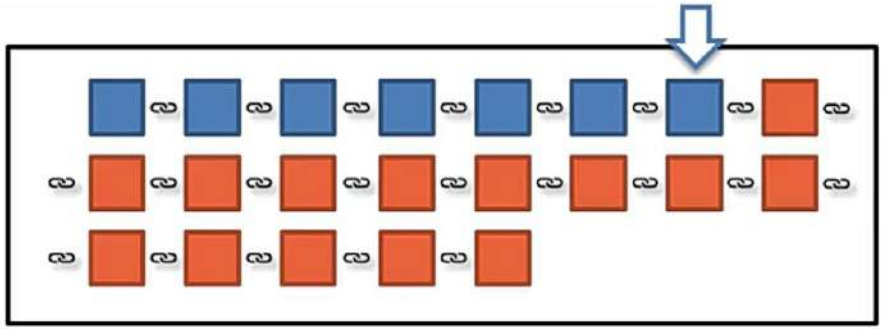


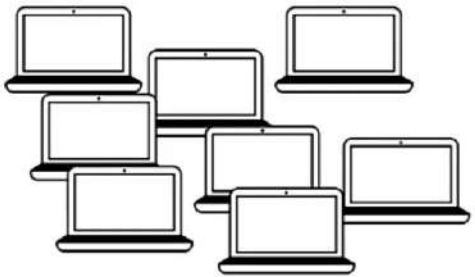


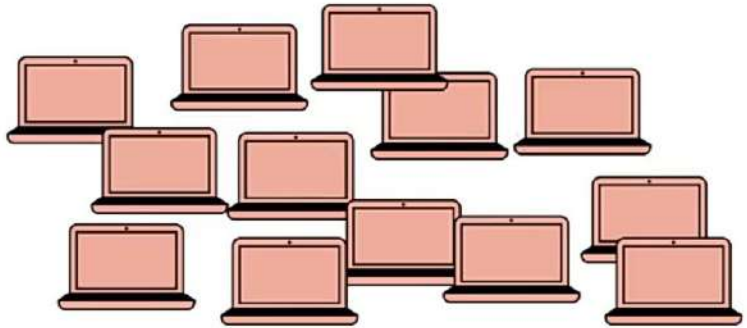


51% Attack

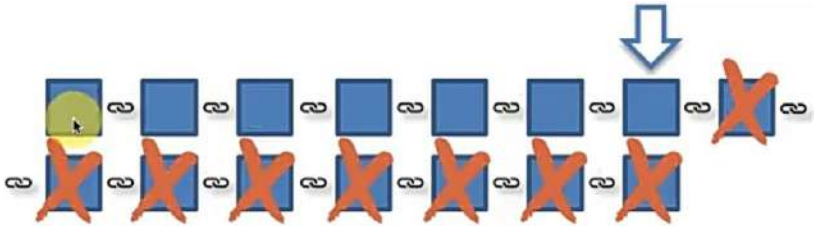


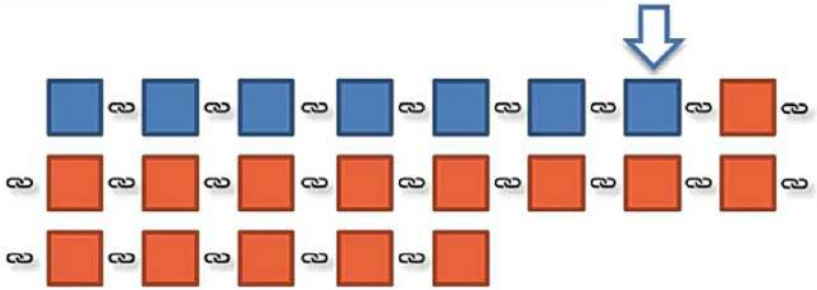


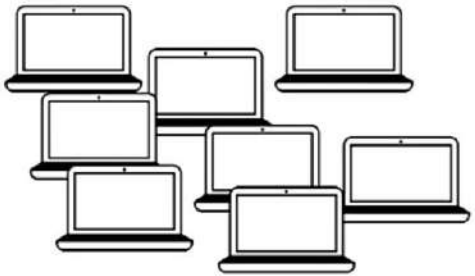


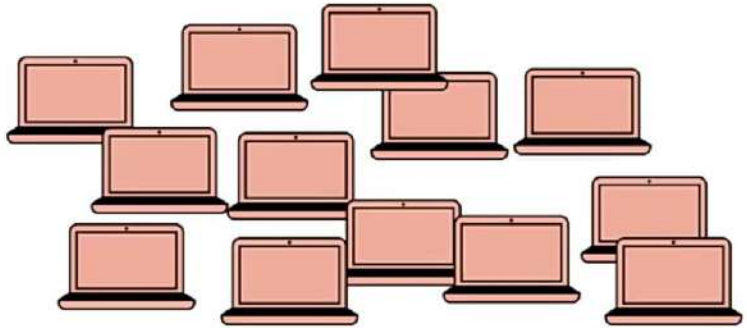


51% Attack

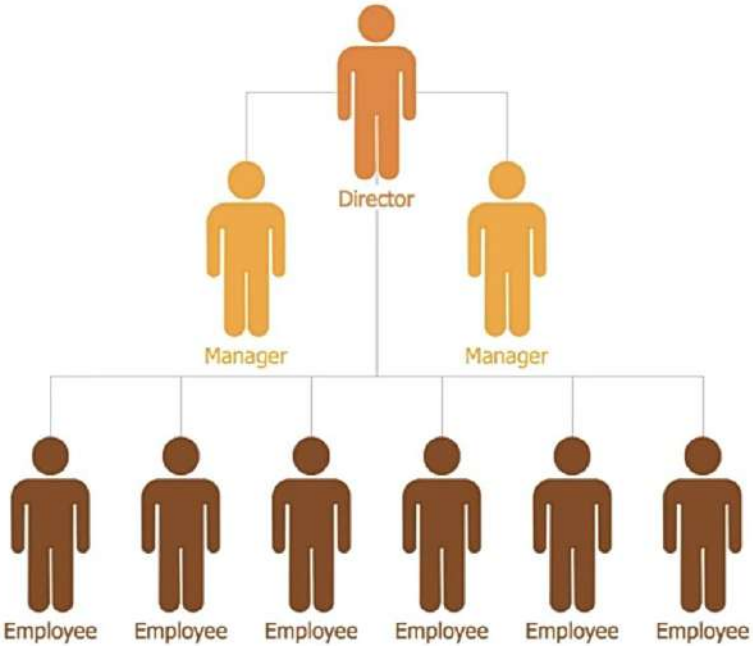




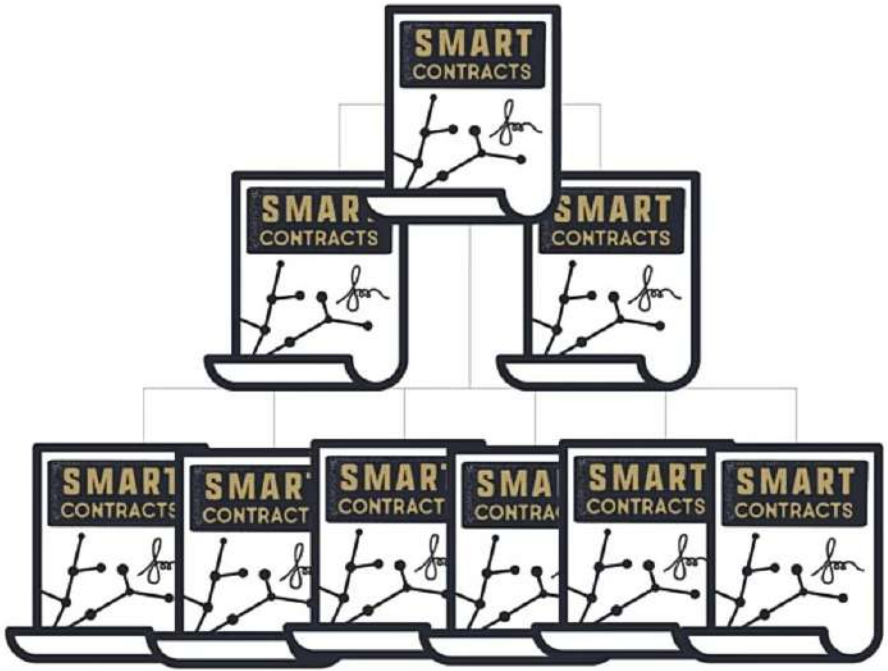




Decentralized Autonomous Organizations



Decentralized Autonomous Organizations



Decentralized Autonomous Organizations

Decentralized Autonomous Organizations

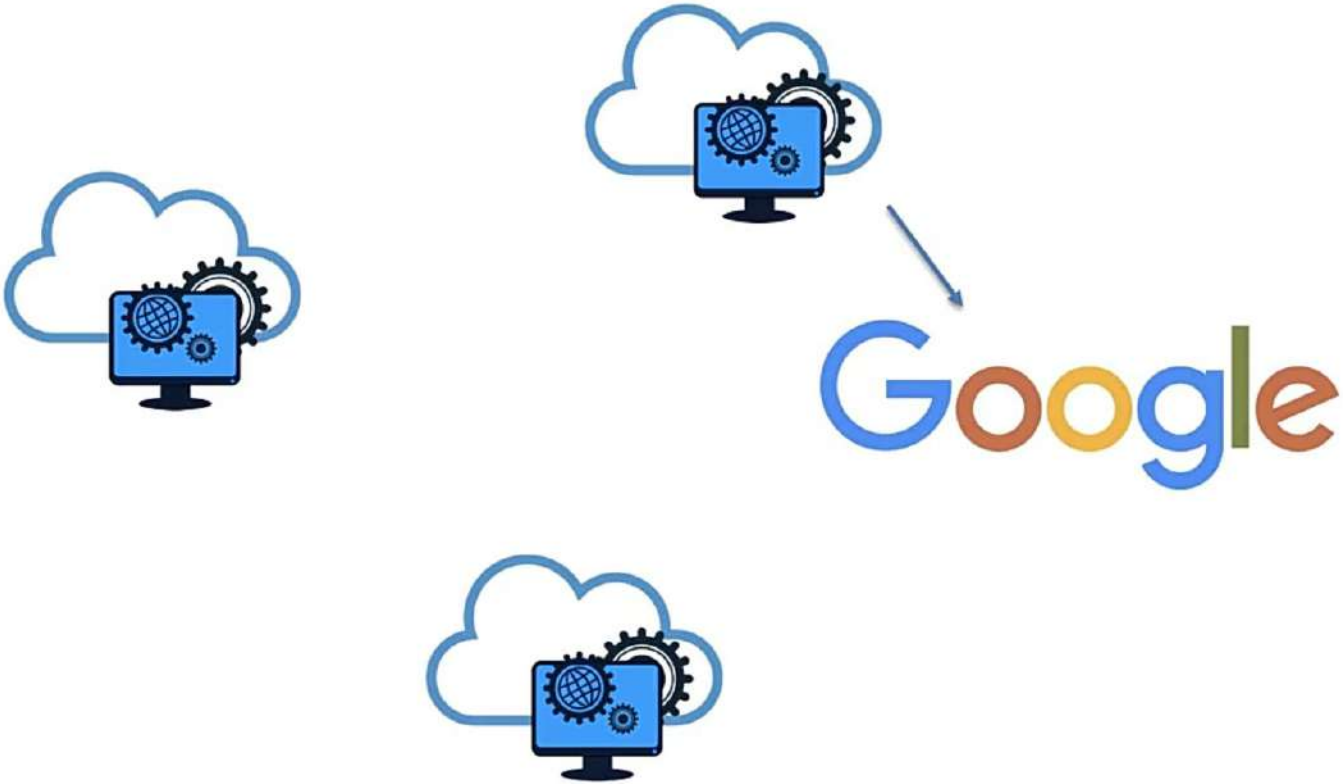
Decentralized Autonomous Organizations

Decentralized Autonomous Organizations

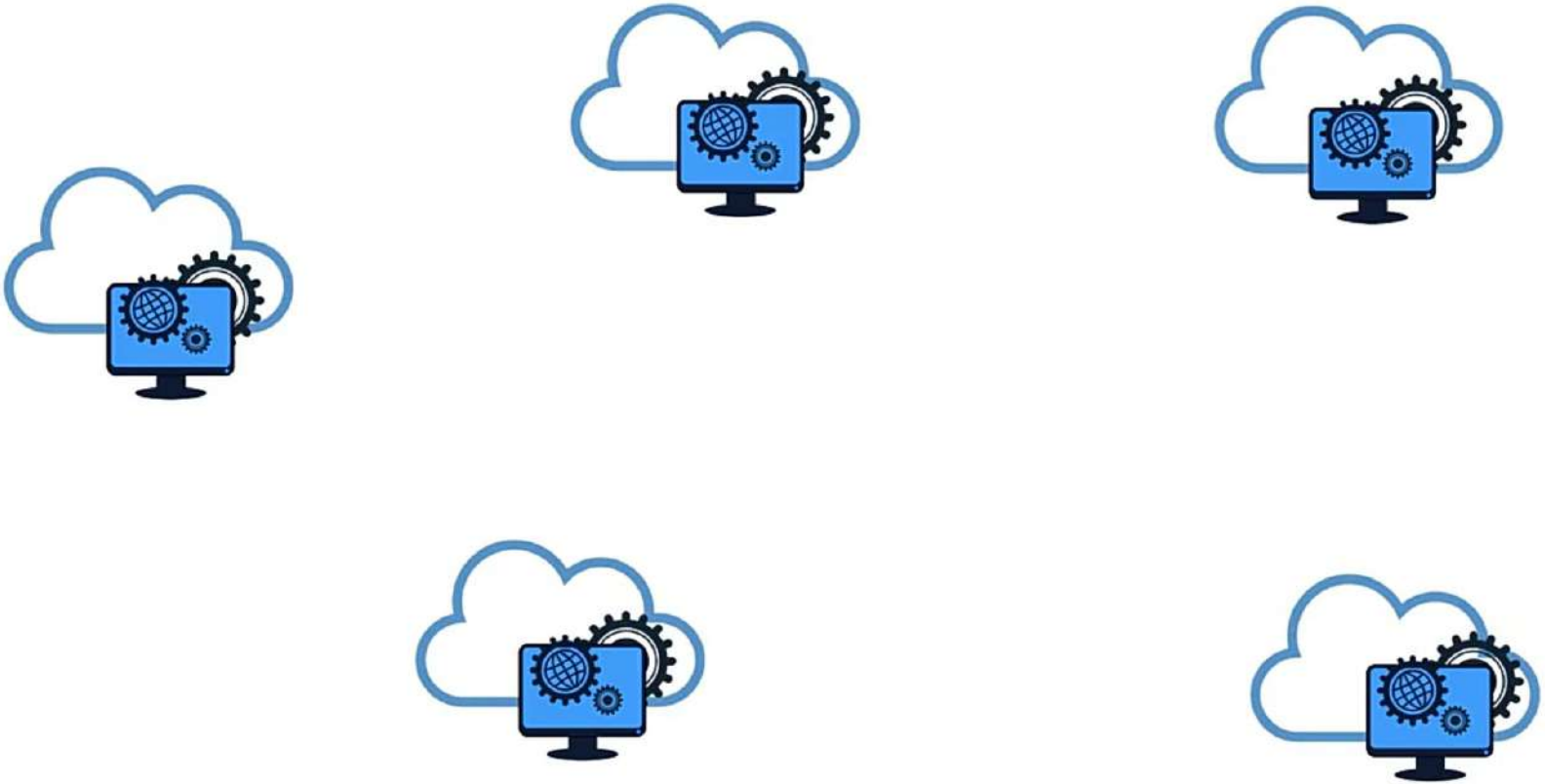
Decentralized Autonomous Organizations

Decentralized Autonomous Organizations

Decentralized Autonomous Organizations



Decentralized Autonomous Organizations

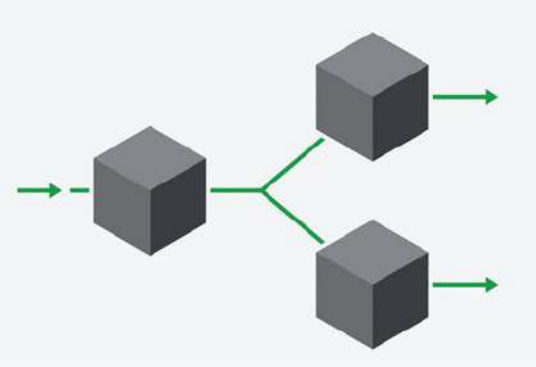


Decentralized Autonomous Organizations

2016
On Ethereum
Investor-directed venture capital fund
Stateless
May 2016 Crowdfunded ~$150,000,000
June 2016 Hacked for ~$50,000,000
Dilemma: *"Code Is Law?"*
Hard fork
Ethereum split into ETH and ETC
Hacker walked away with ~$67,000,000 in ETC
Problem in DAO code not Ethereum

Blockchain Forks

- A blockchain split that produces two competing branches.
- Can be **accidental** or **intentional**.
- **Accidental forks** are resolved by the blockchain.
- Intentional forks are used to implement new consensus rules.
    a. **Hard forks** require nodes to be upgraded to the new consensus rules or to rollback the state.
    b. **Soft forks** don't require nodes to be upgraded to the new consensus rules.



NB: Some cryptocurrencies, like **Bitcoin Cash**, are created using hard forks.

Courtesy : https://blog.bitstamp.net/post/what-are-blockchain-forks/

Blockchain Forks

Accidental Forks

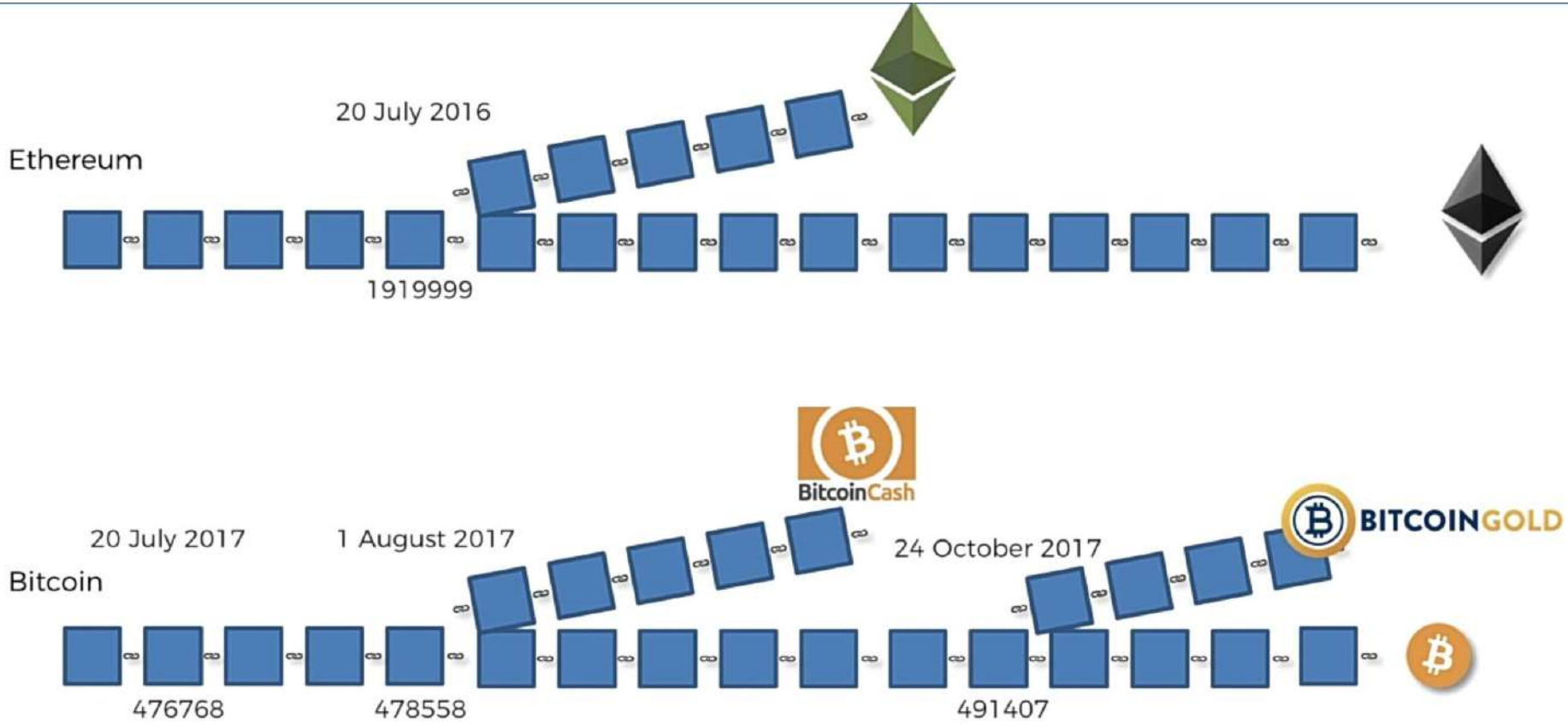
- At any given moment, thousands of miners are competing to create a new block.
- With so much mining going on at once, two or more miners sometimes mine a new block at the same time.
- When this happens, an **accidental fork** is created.
- The problem is solved when new blocks are added to one of the chains. When that happens, the network continues working on the longer chain and abandons the shorter one.

Blockchain Forks

Intentional forks

- When an **intentional fork** is made, the network doesn't reconverge on a single chain.
- This type of a fork is used by blockchain developers to implement changes to the protocol.
- For instance, developers may use an intentional fork to increase block size, reduce block time, or even implement an entirely new consensus algorithm.
- An intentional fork can be **hard** or **soft**. The two differ from each other in terms of compatibility with the other chain and their applications.

Soft & Hard Forks

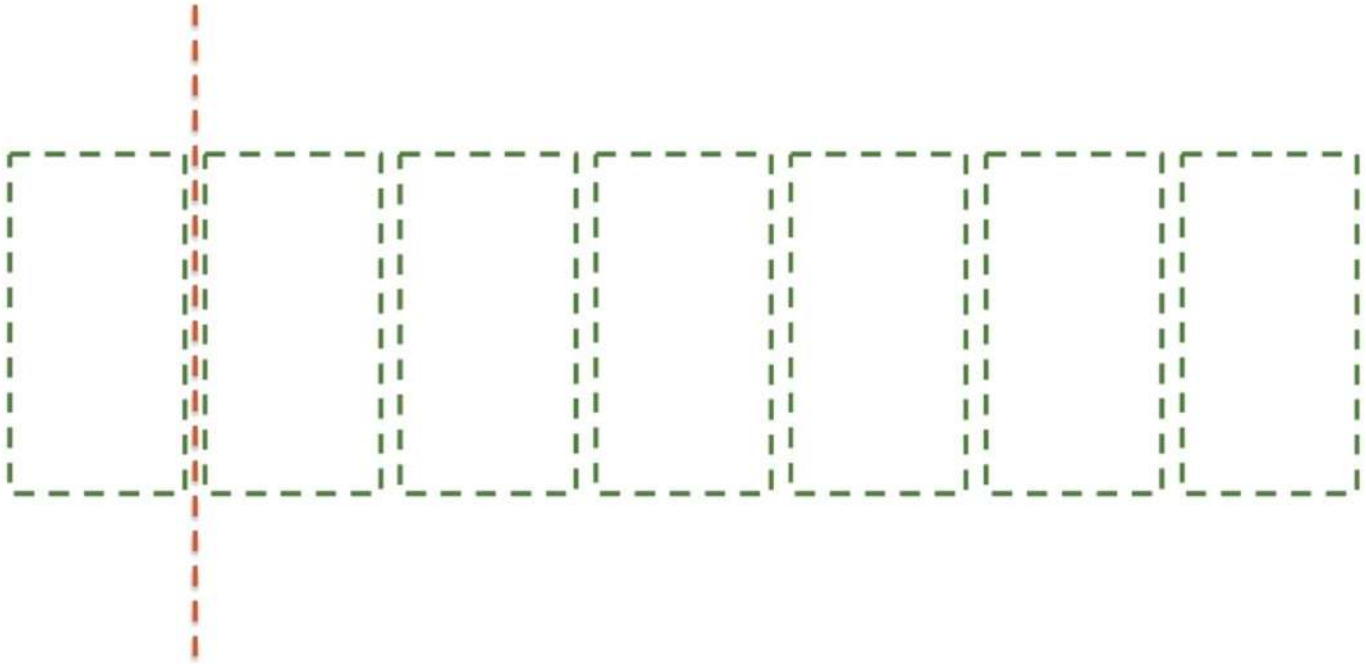


Soft & Hard Forks

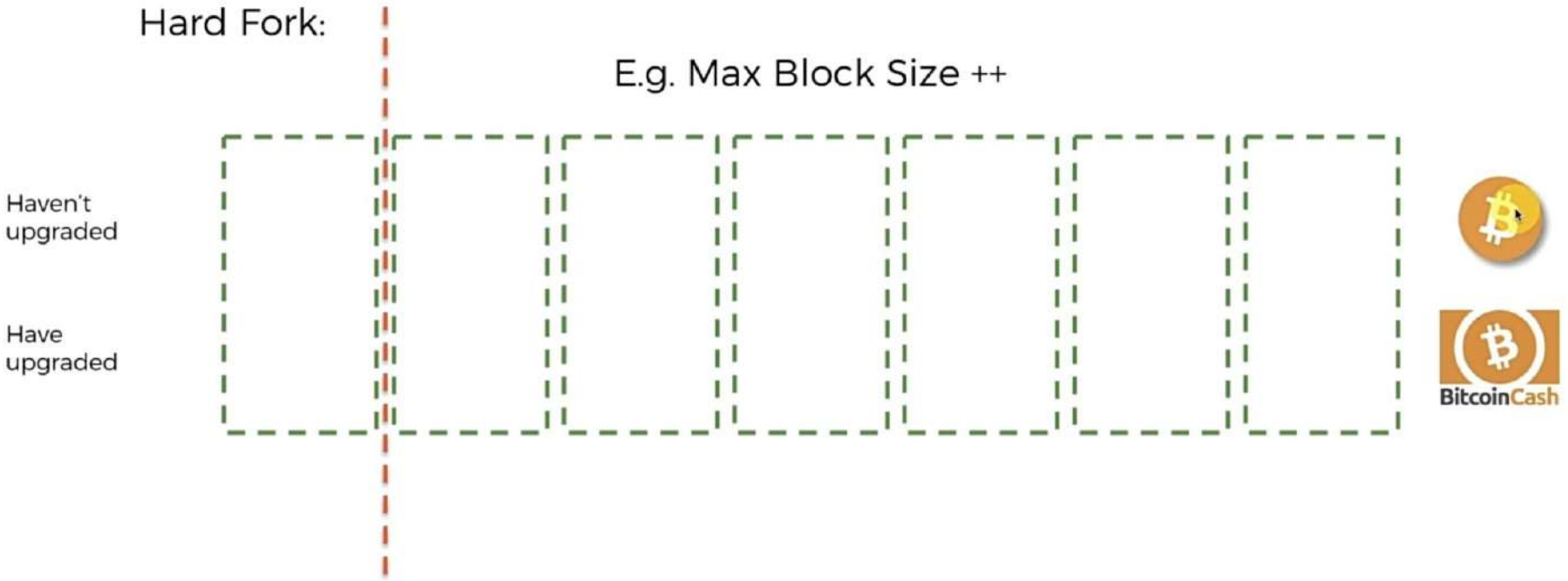
Hard Forks = Loosen Rules

Soft Forks = Tighten Rules

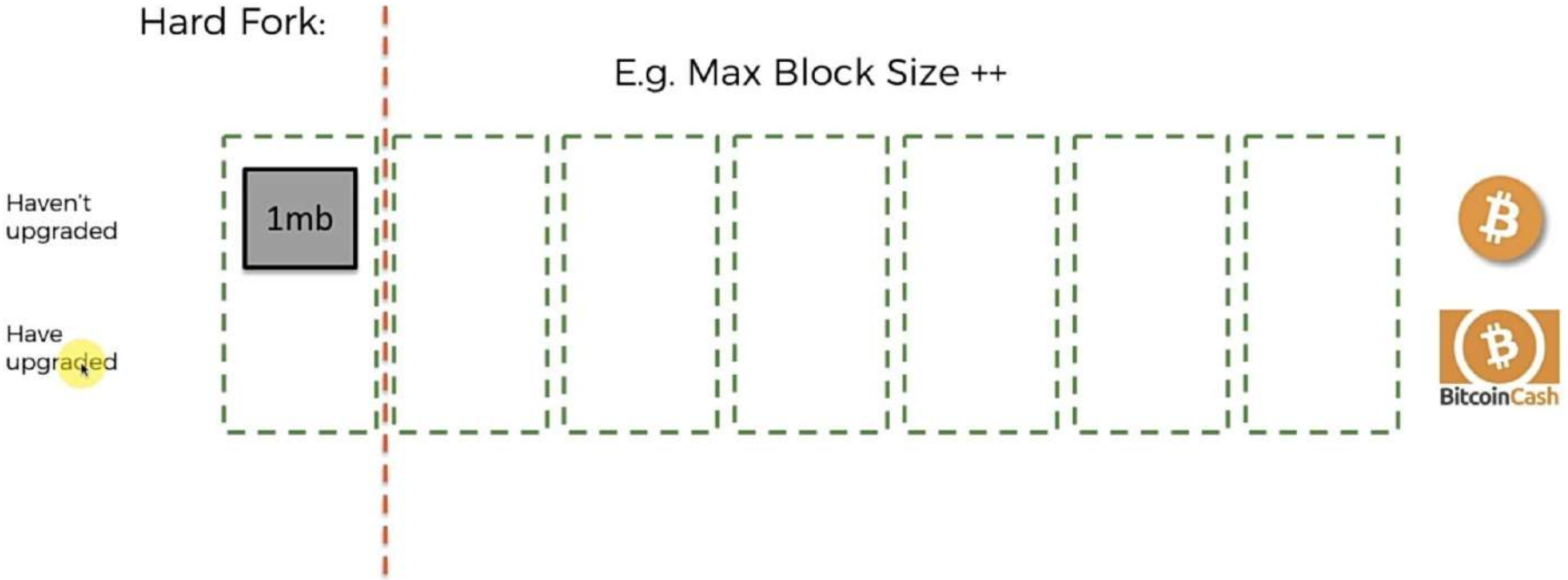
Soft & Hard Forks



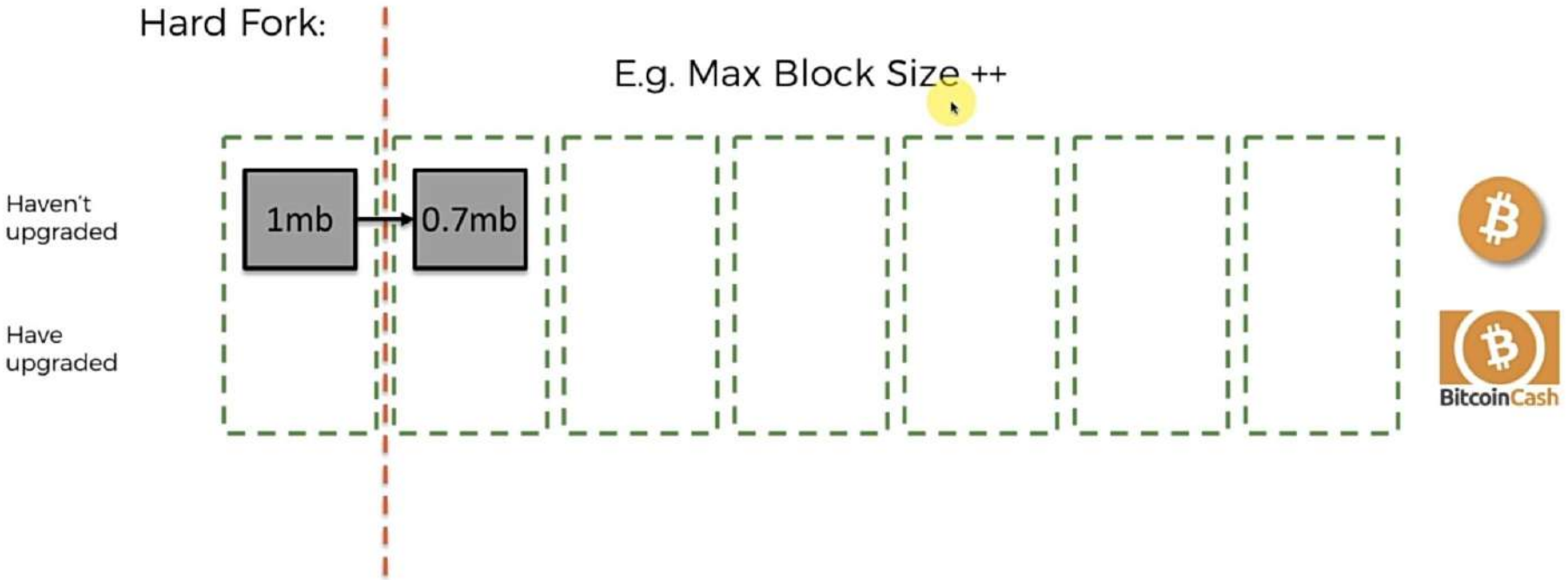
Soft & Hard Forks



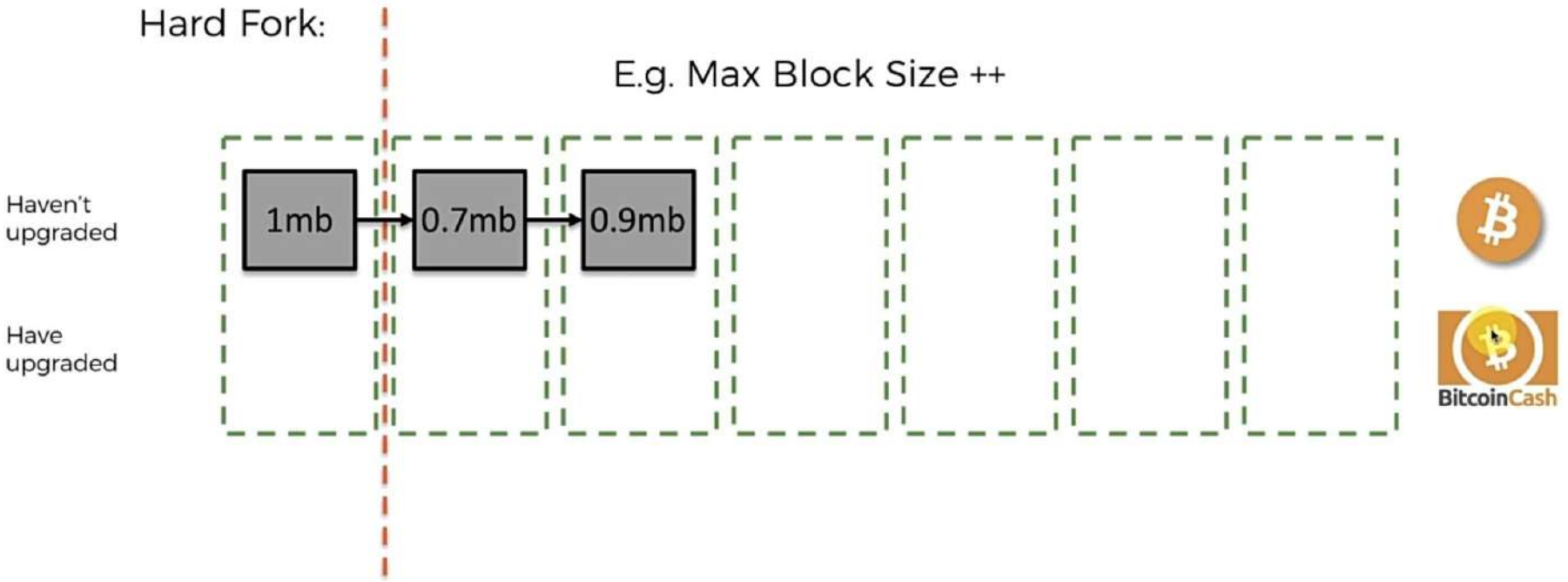
Soft & Hard Forks



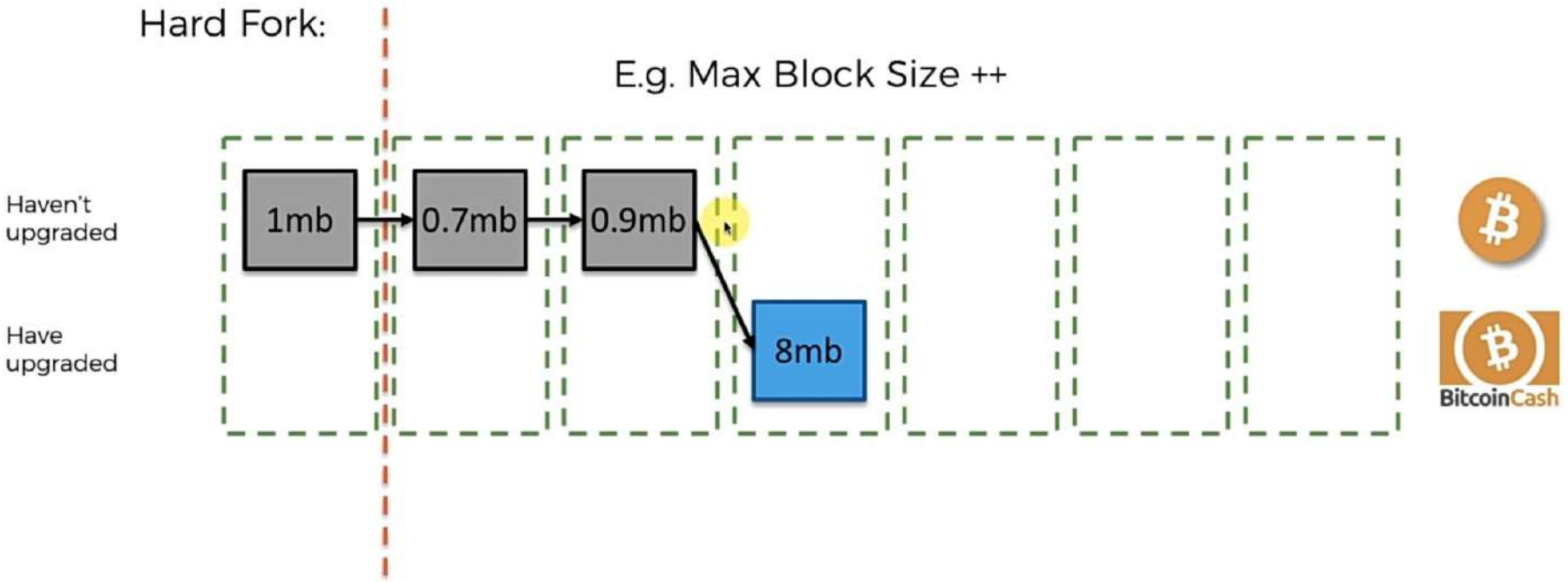
Soft & Hard Forks



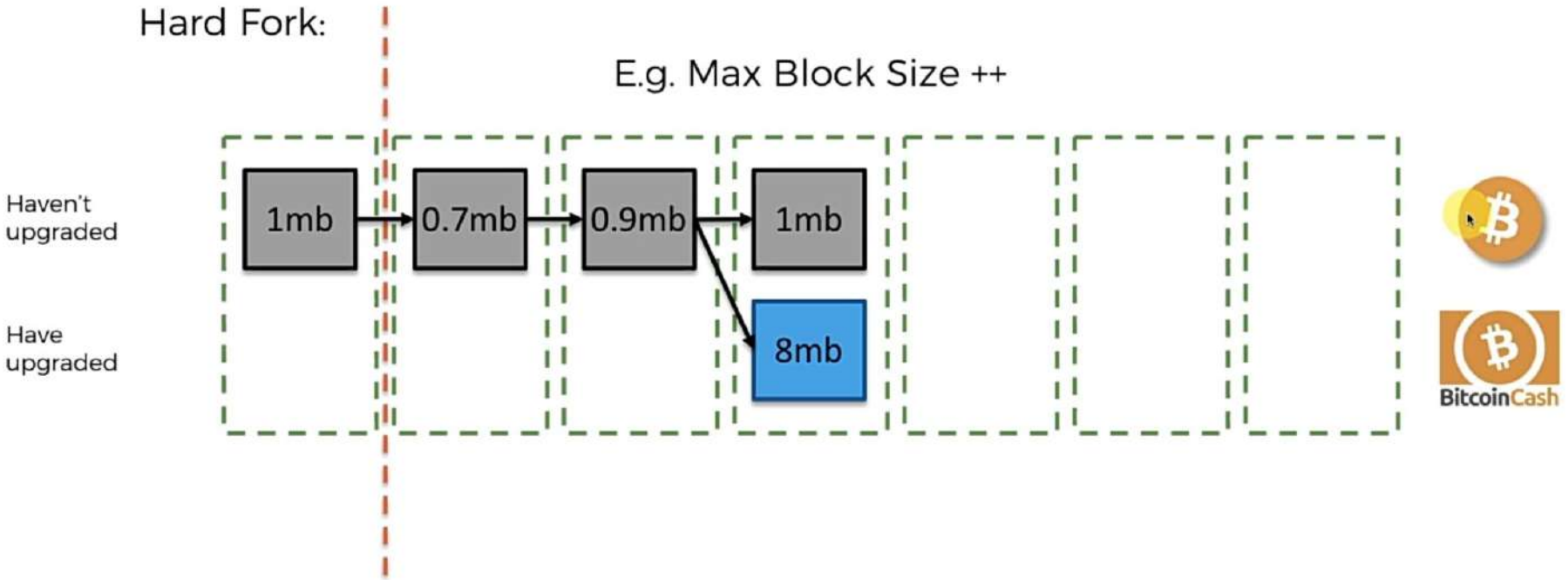
Soft & Hard Forks



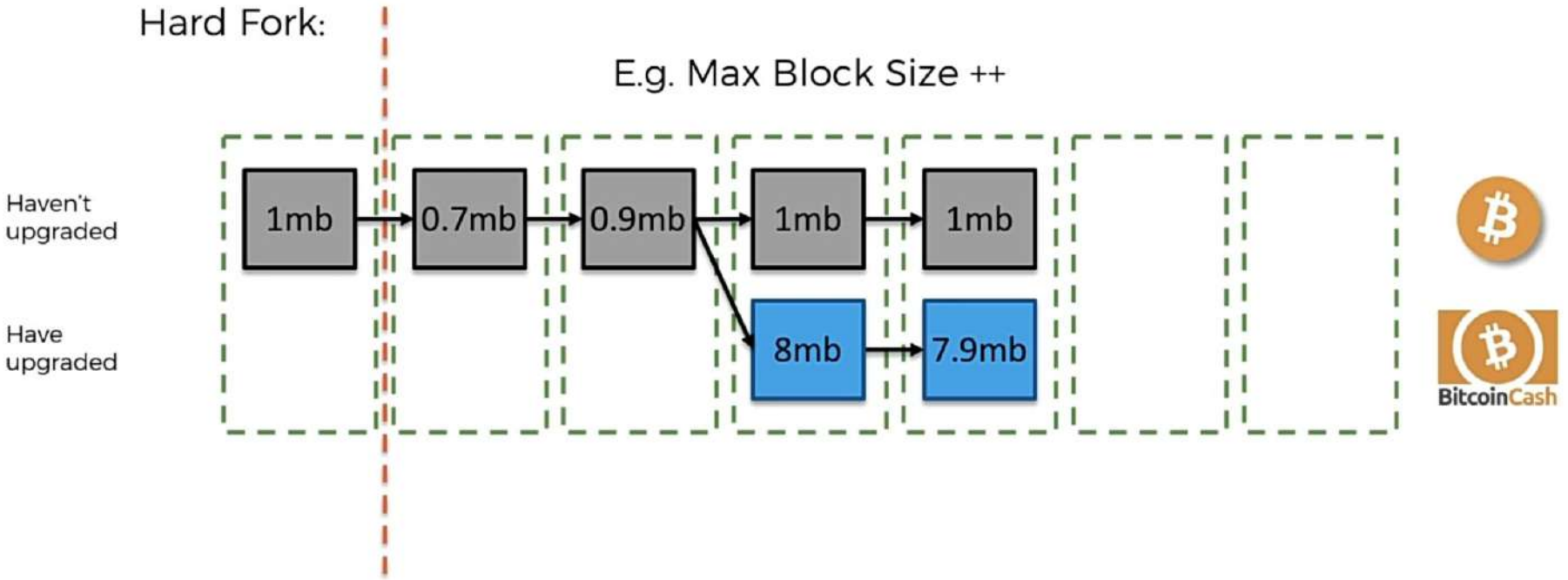
Soft & Hard Forks



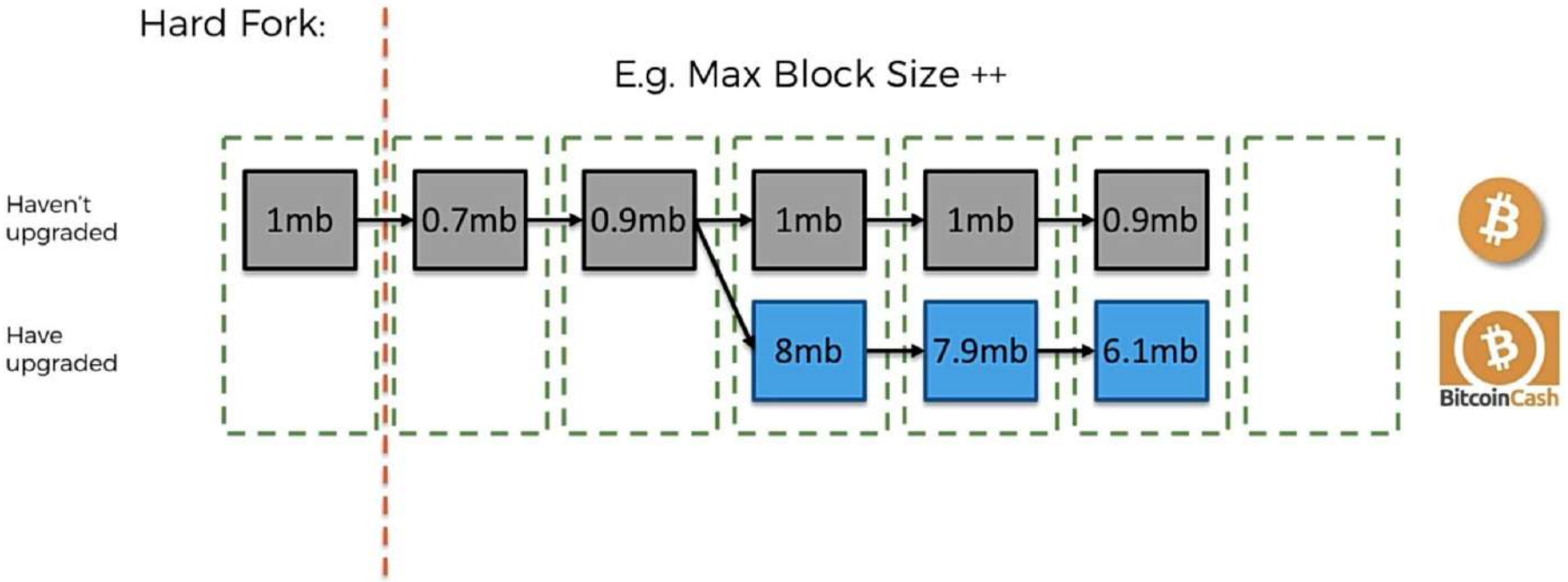
Soft & Hard Forks



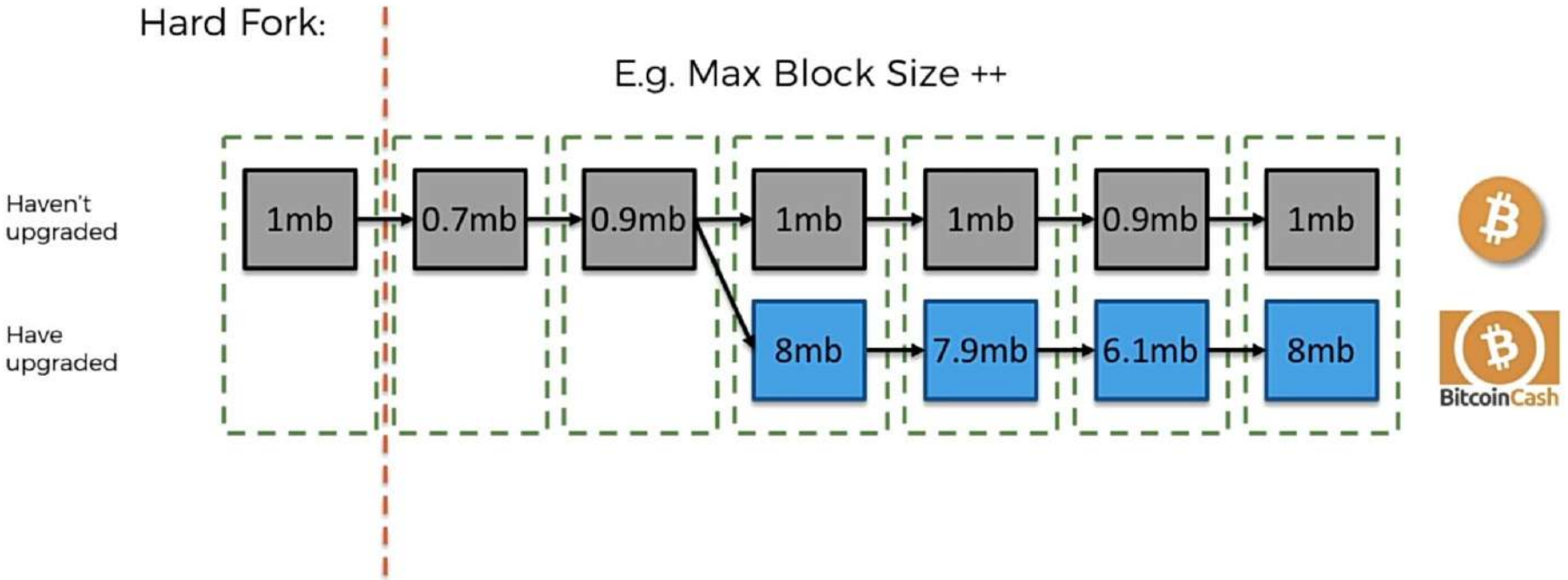
Soft & Hard Forks



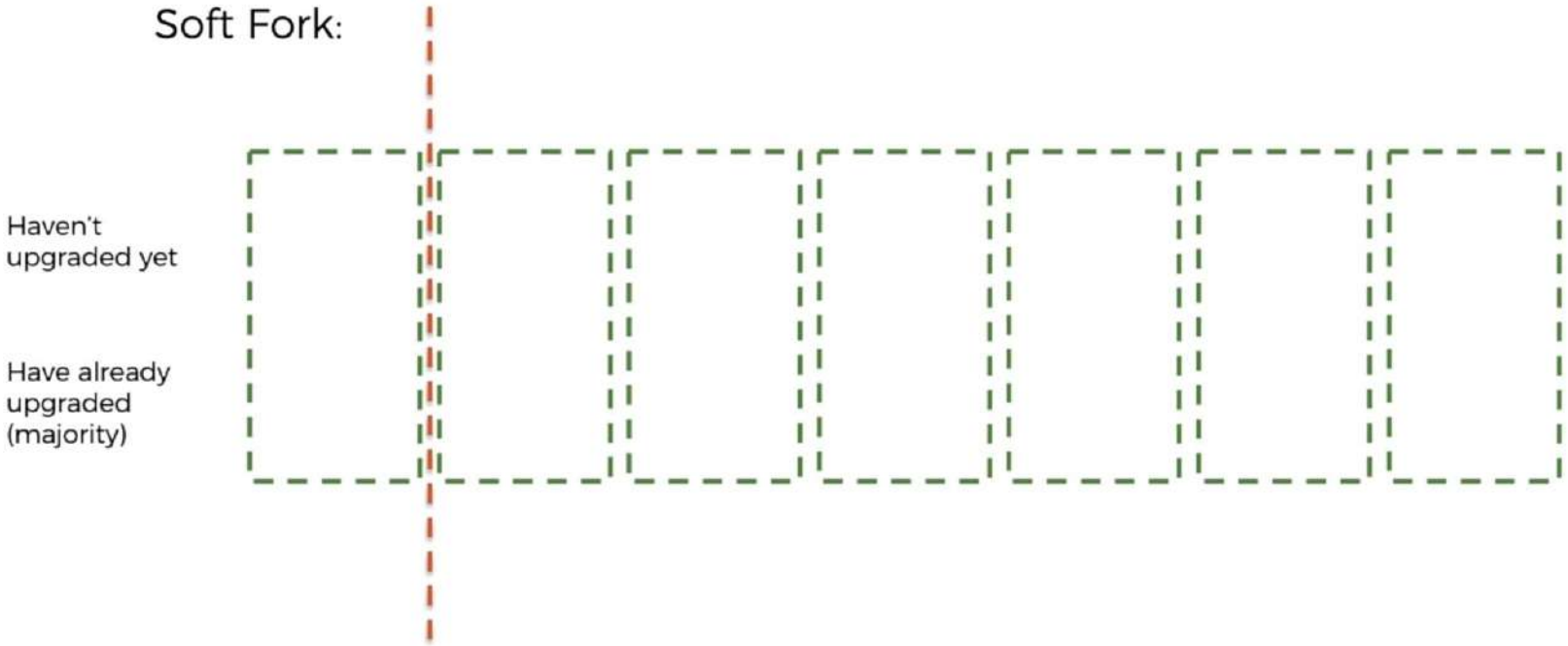
Soft & Hard Forks



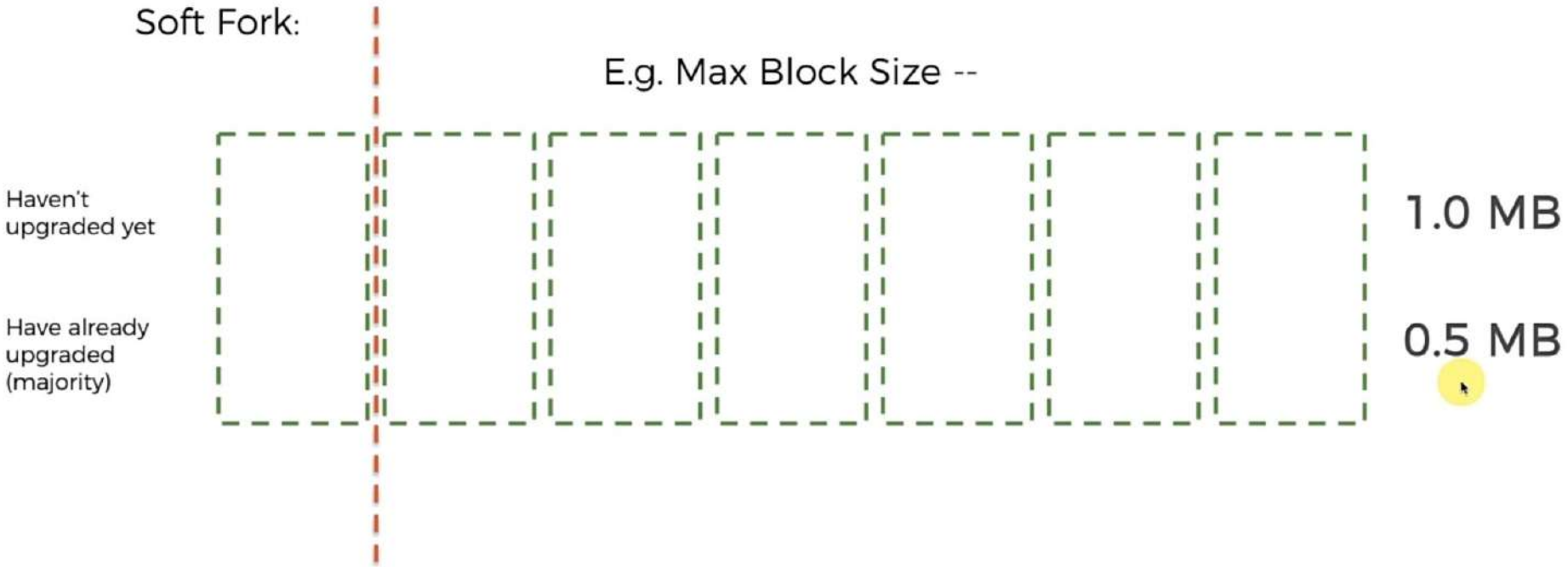
Soft & Hard Forks



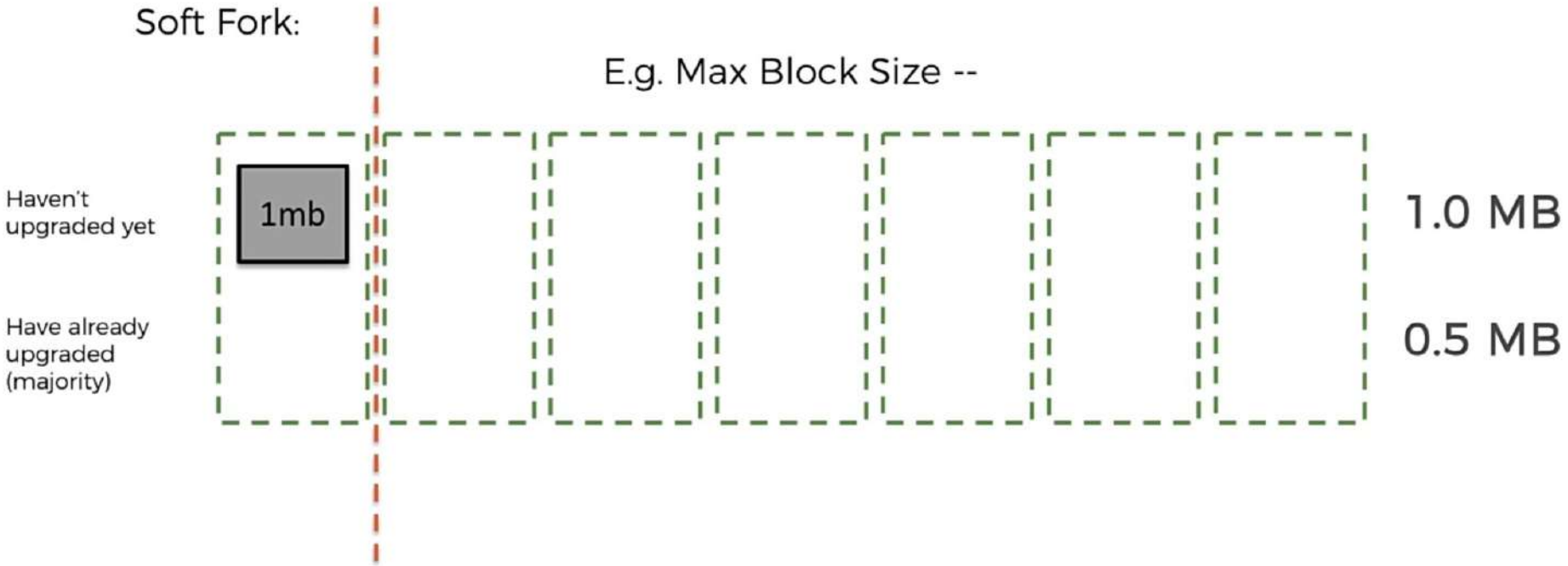
Soft & Hard Forks



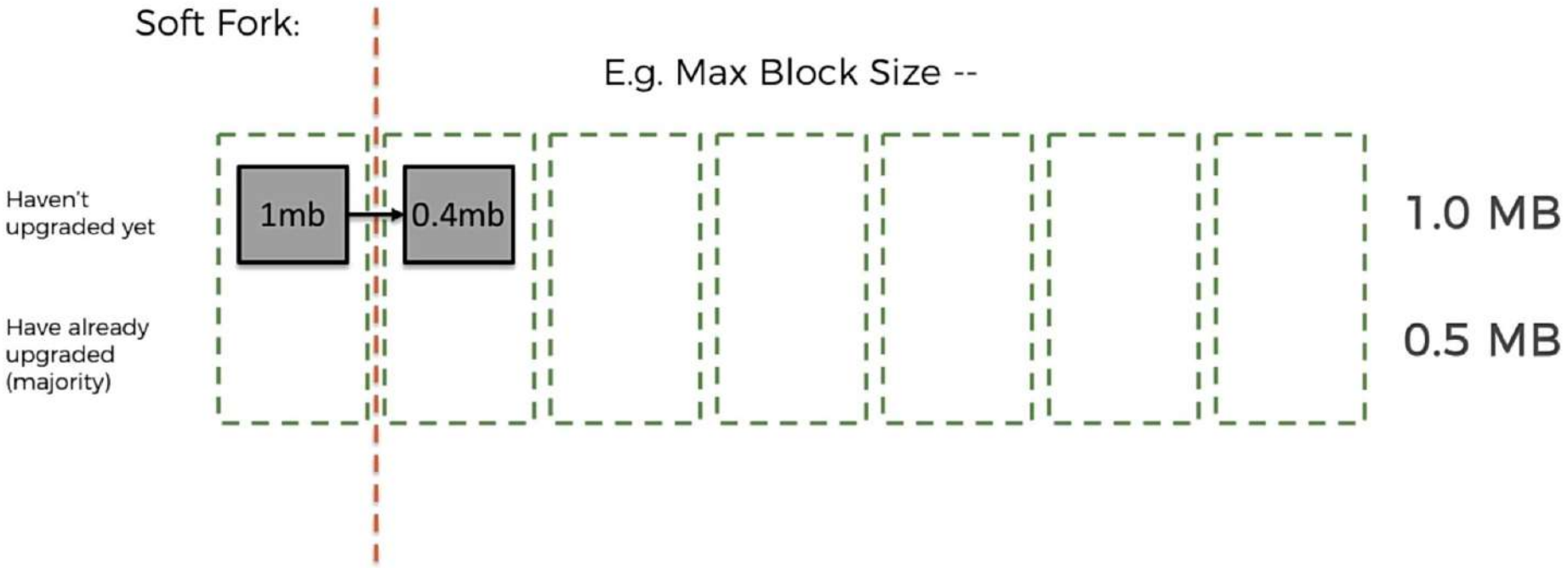
Soft & Hard Forks



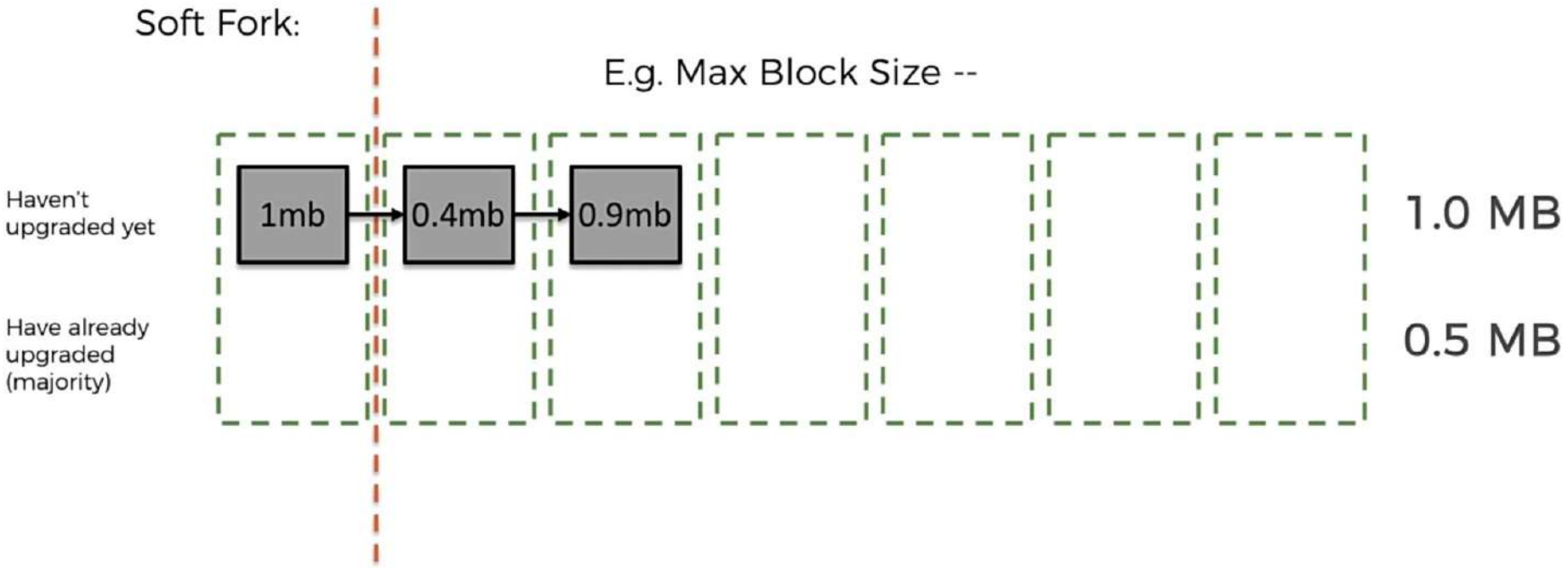
Soft & Hard Forks



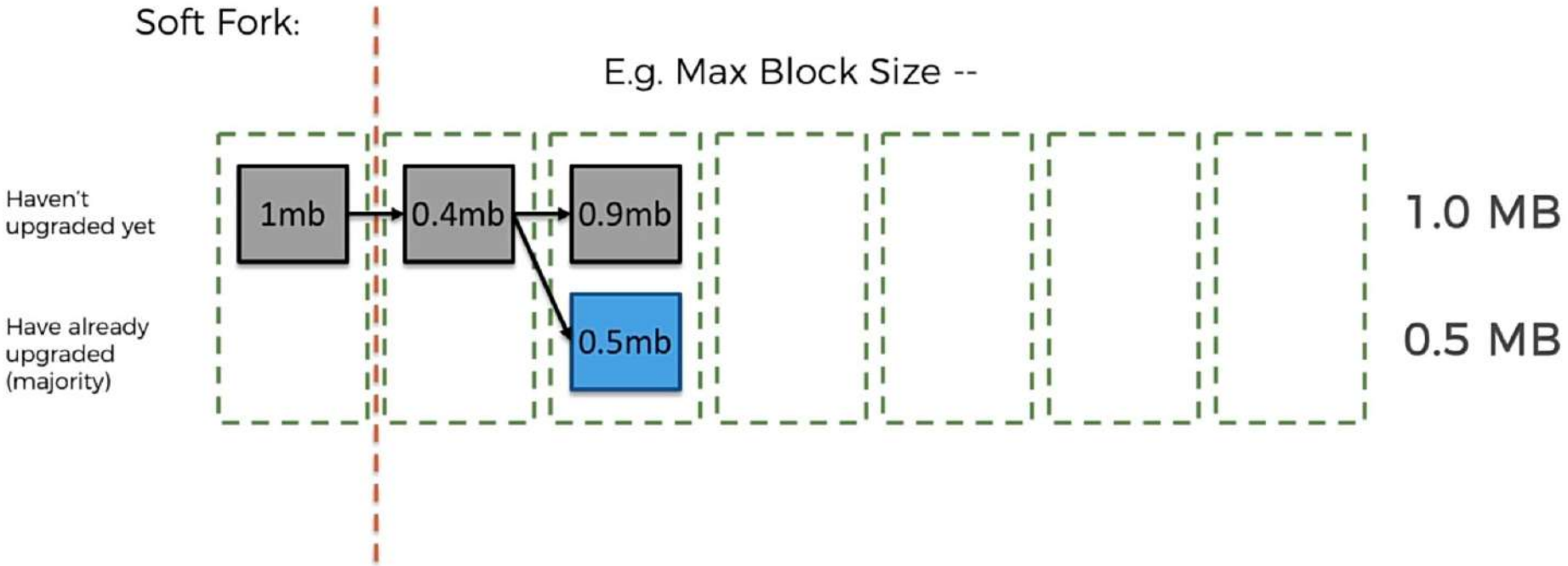
Soft & Hard Forks



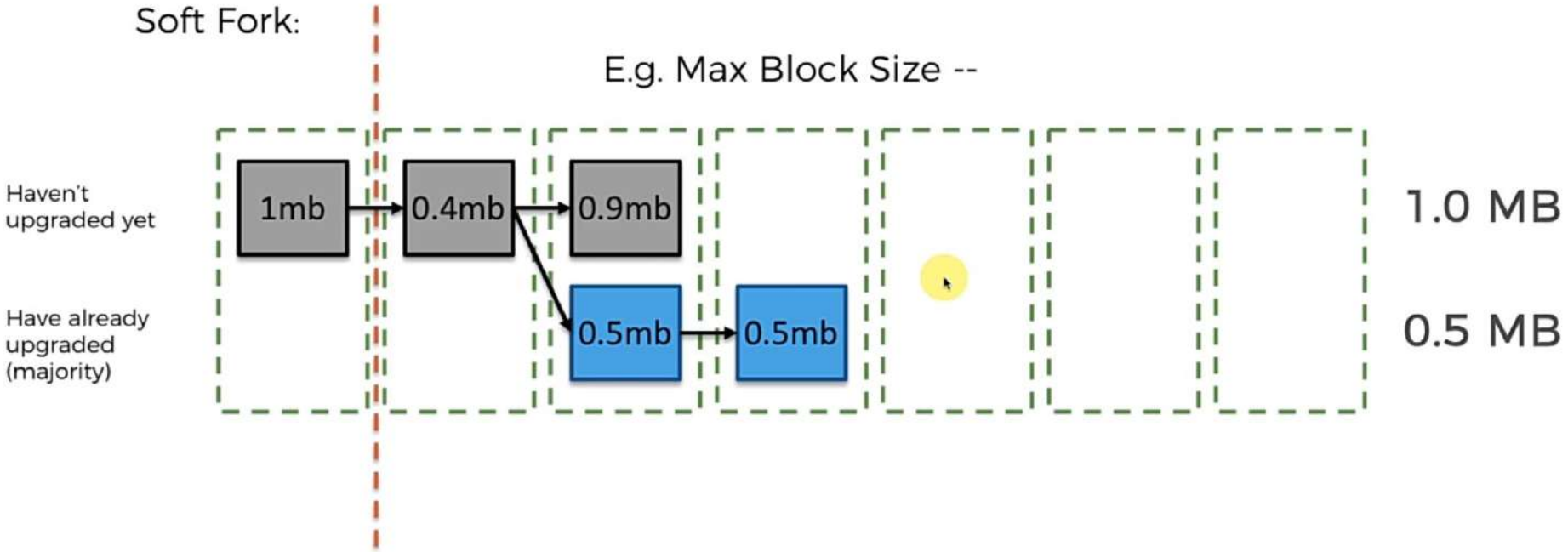
Soft & Hard Forks



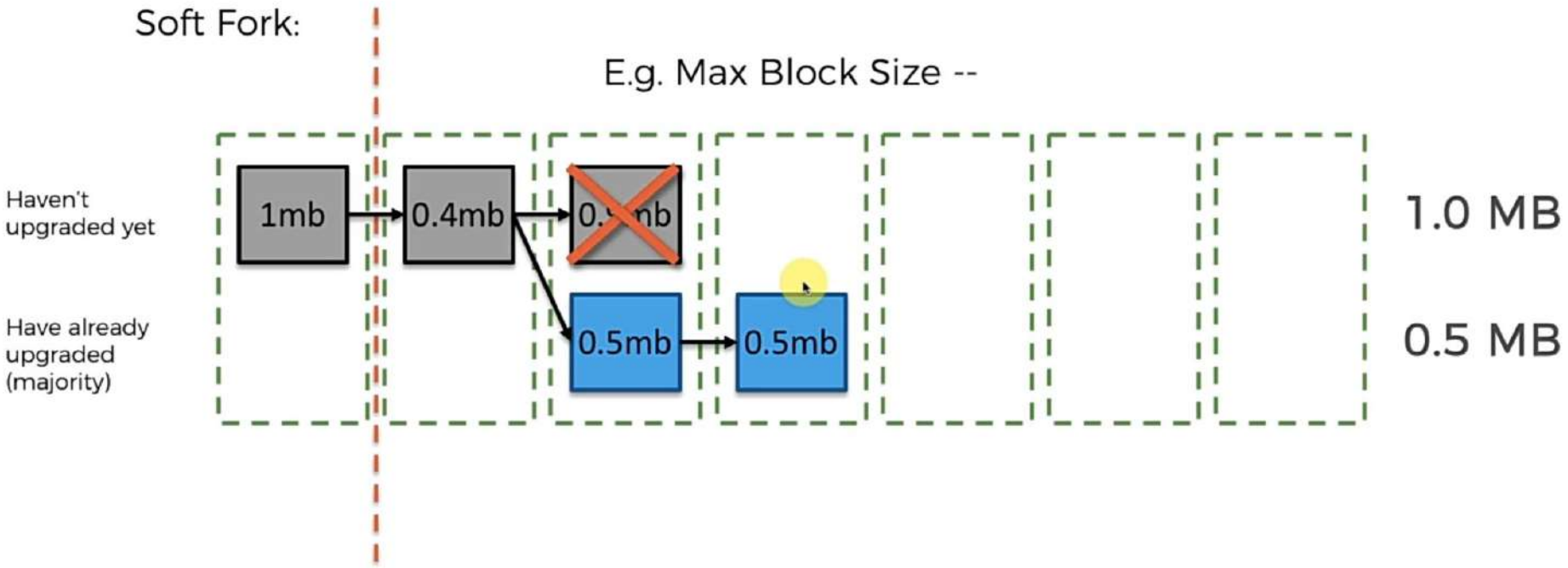
Soft & Hard Forks



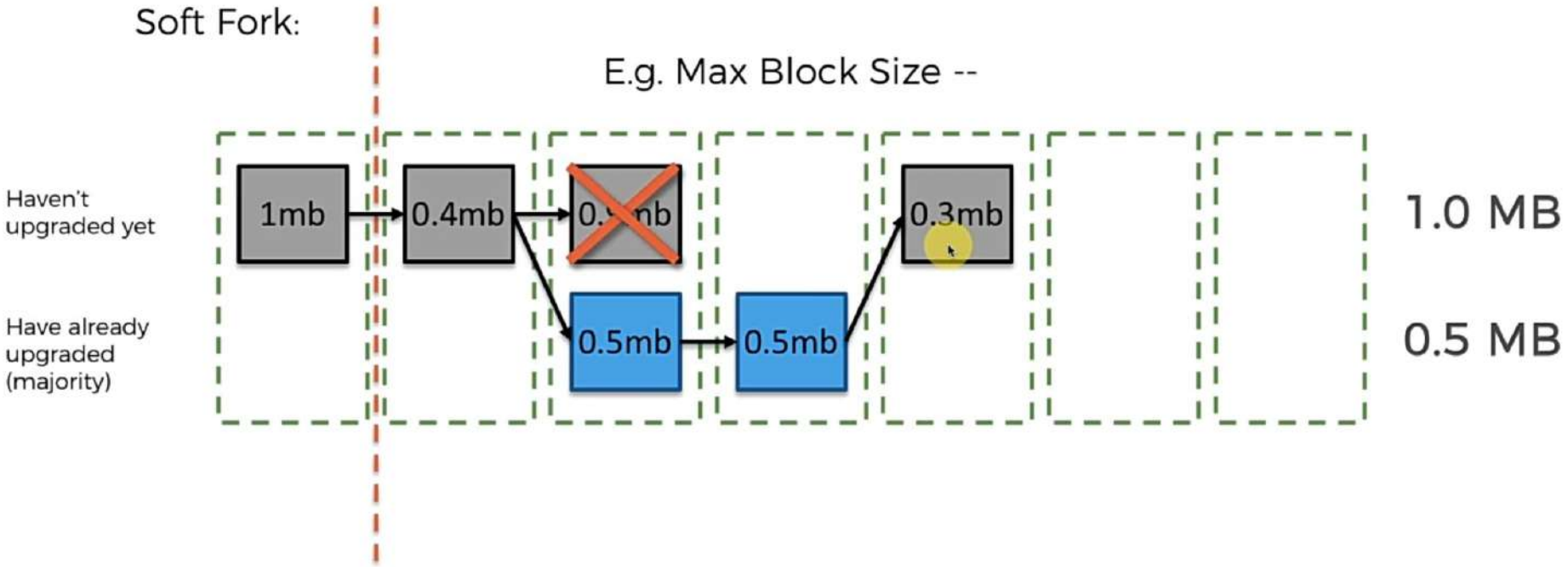
Soft & Hard Forks



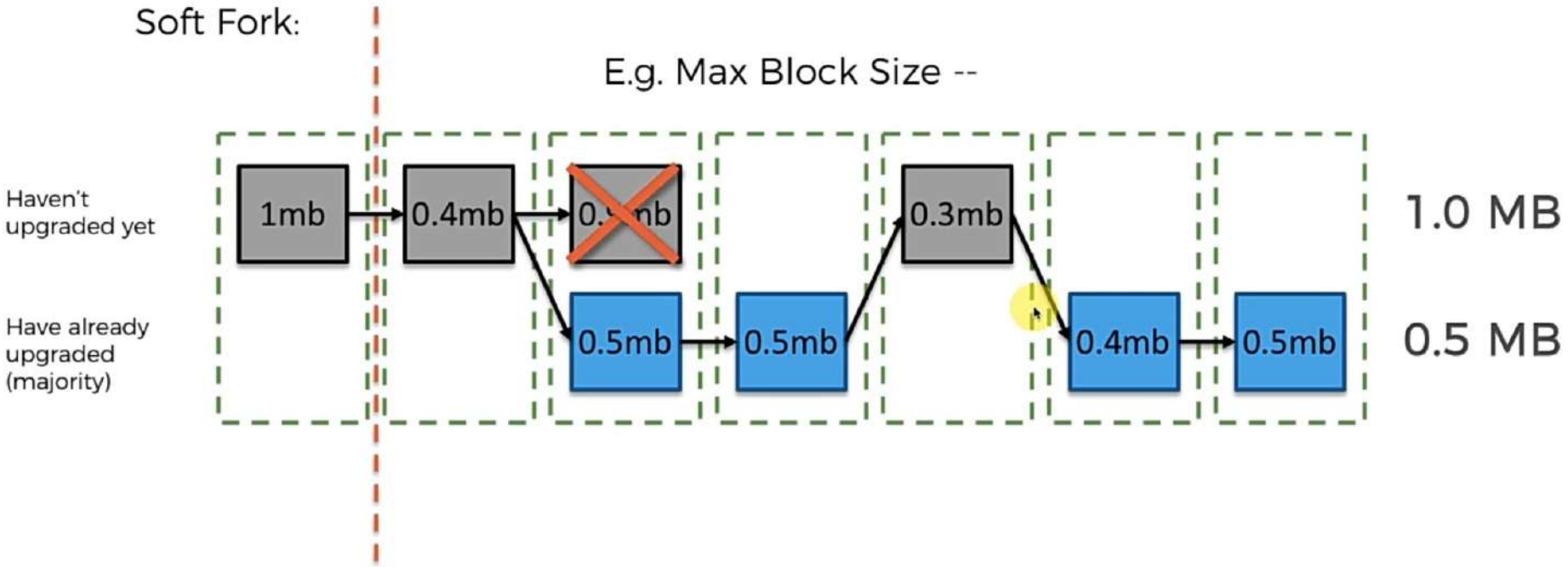
Soft & Hard Forks



Soft & Hard Forks



Soft & Hard Forks



Different Consensus Algorithms

1. **Proof of Work (PoW)**
2. Proof of Elapsed Time(PoET)
3. Proof of Stake (PoS)
4. Delegated Proof of Stake (DPoS)
5. Proof of Authority (PoA)
6. Practical Byzantine Fault Tolerance
7. RAFT

Other Consensus Algorithms

1. Proof of Stake Anonymous (PoSA):
2. Leased Proof of Stake (LPoS):
3. Proof of Importance (PoI):
4. Proof of Storage
5. Proof of Burn
6. Proof of Activity
7. Proof of Capacity
8. Directed Acyclic Graph (DAG)