Timechain

A Time Synchronization Protocol based on Distributed Network

LEUNG TSZ HIN (1155079351)
SUPERVISED BY PROF. LYU RUNG TSONG MICHAEL

Time. Why it matters?

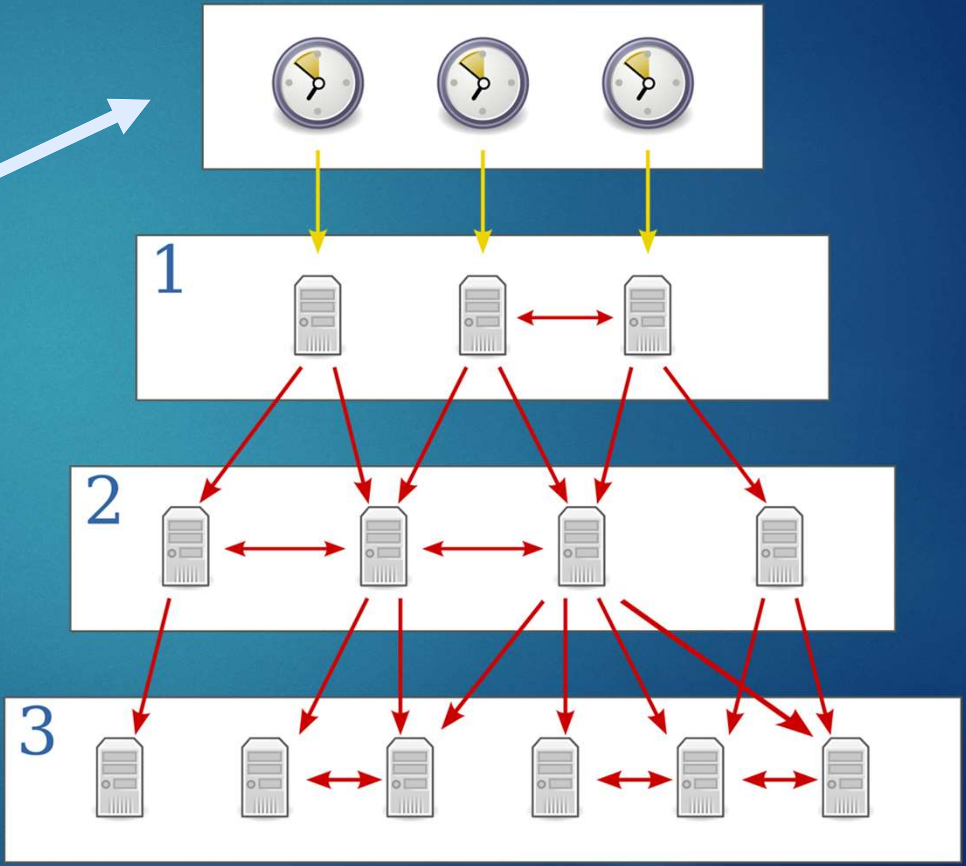
- TLS Certificates
  - 6.75% Chrome users have error >24 hours
- Authentication
- Bitcoin

Network Time Protocol

- Developed in 1980s
- NTPv4
- UDP Port 123
- Maintained by Network Time Foundation

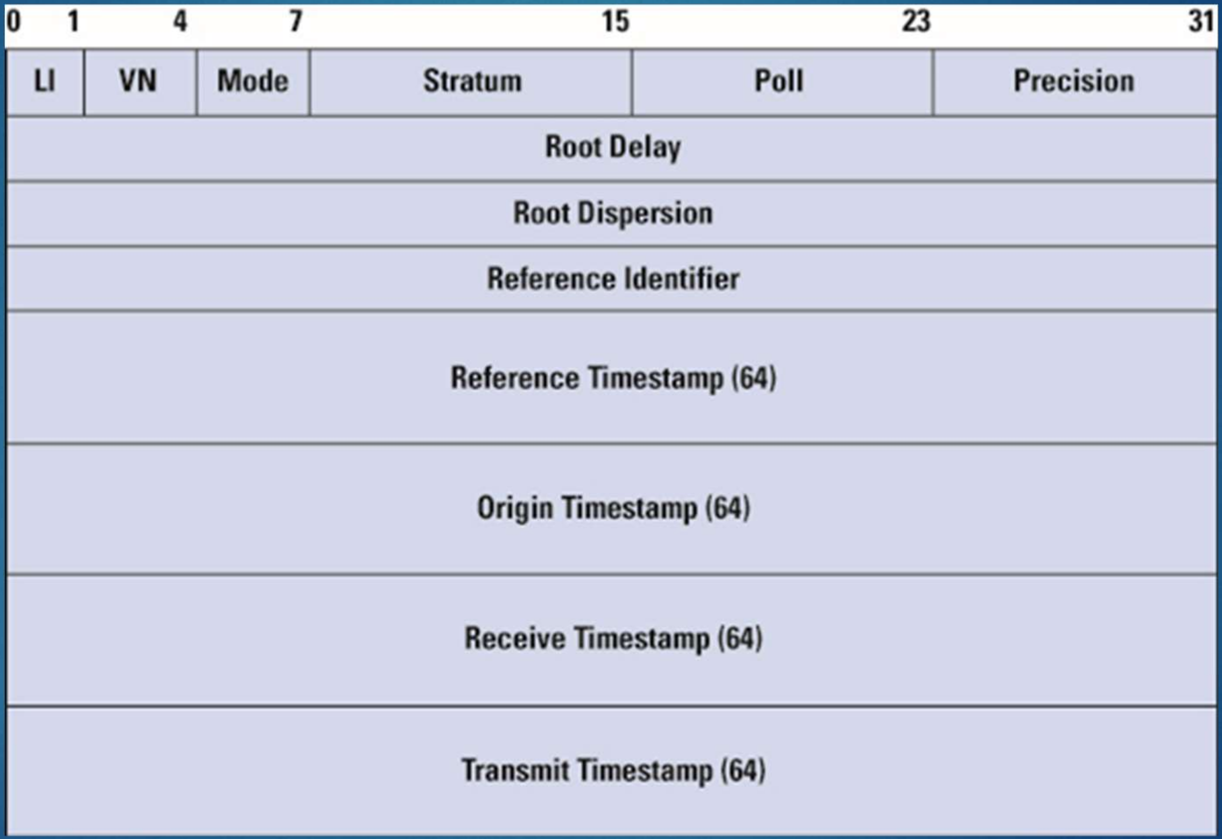
How NTP works?

Stratum 0 High-precision devices



Benjamin D. Esham [Public domain], via Wikimedia Commons

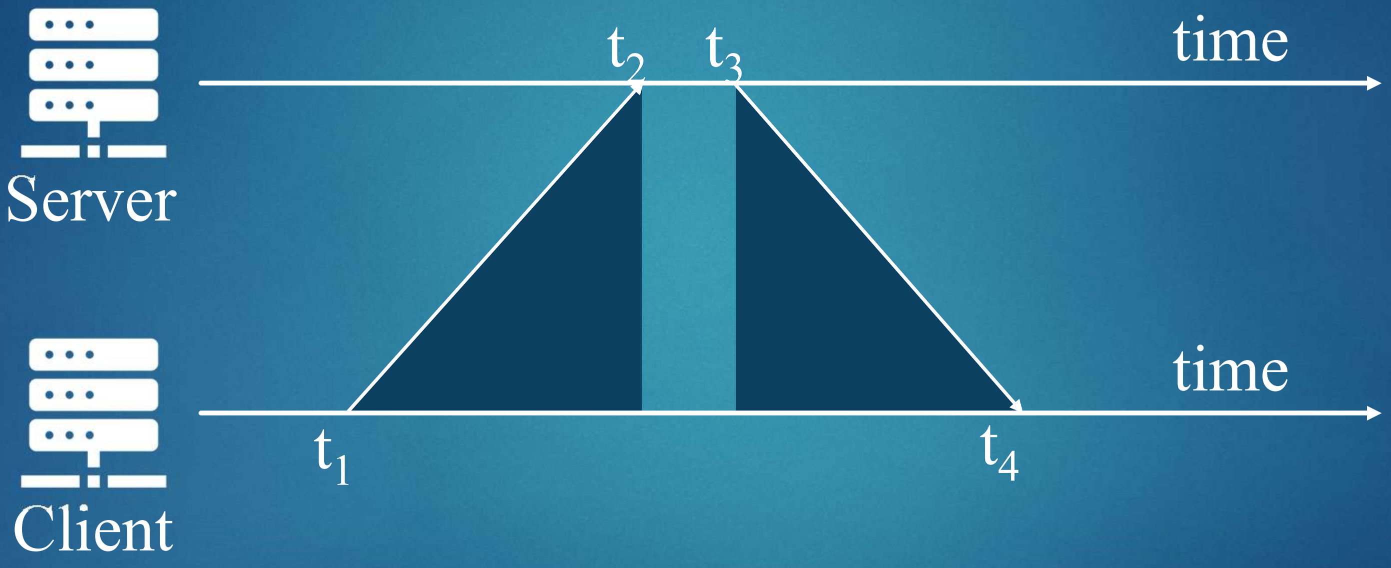
How NTP works?



0 1	4	7	15	23	31
LI	VN	Mode	Stratum	Poll	Precision
Root Delay					
Root Dispersion					
Reference Identifier					
Reference Timestamp (64)					
Origin Timestamp (64)					
Receive Timestamp (64)					
Transmit Timestamp (64)					

https://www.cisco.com/c/en/us/about/press/internet-protocol-journal/back-issues/table-contents-58/154-ntp.html

How NTP works?



$$\boldsymbol{\theta} = \frac{\mathbf{1}}{\mathbf{2}}[(\boldsymbol{t_2} - \boldsymbol{t_1}) + (\boldsymbol{t_3} - \boldsymbol{t_4})]$$

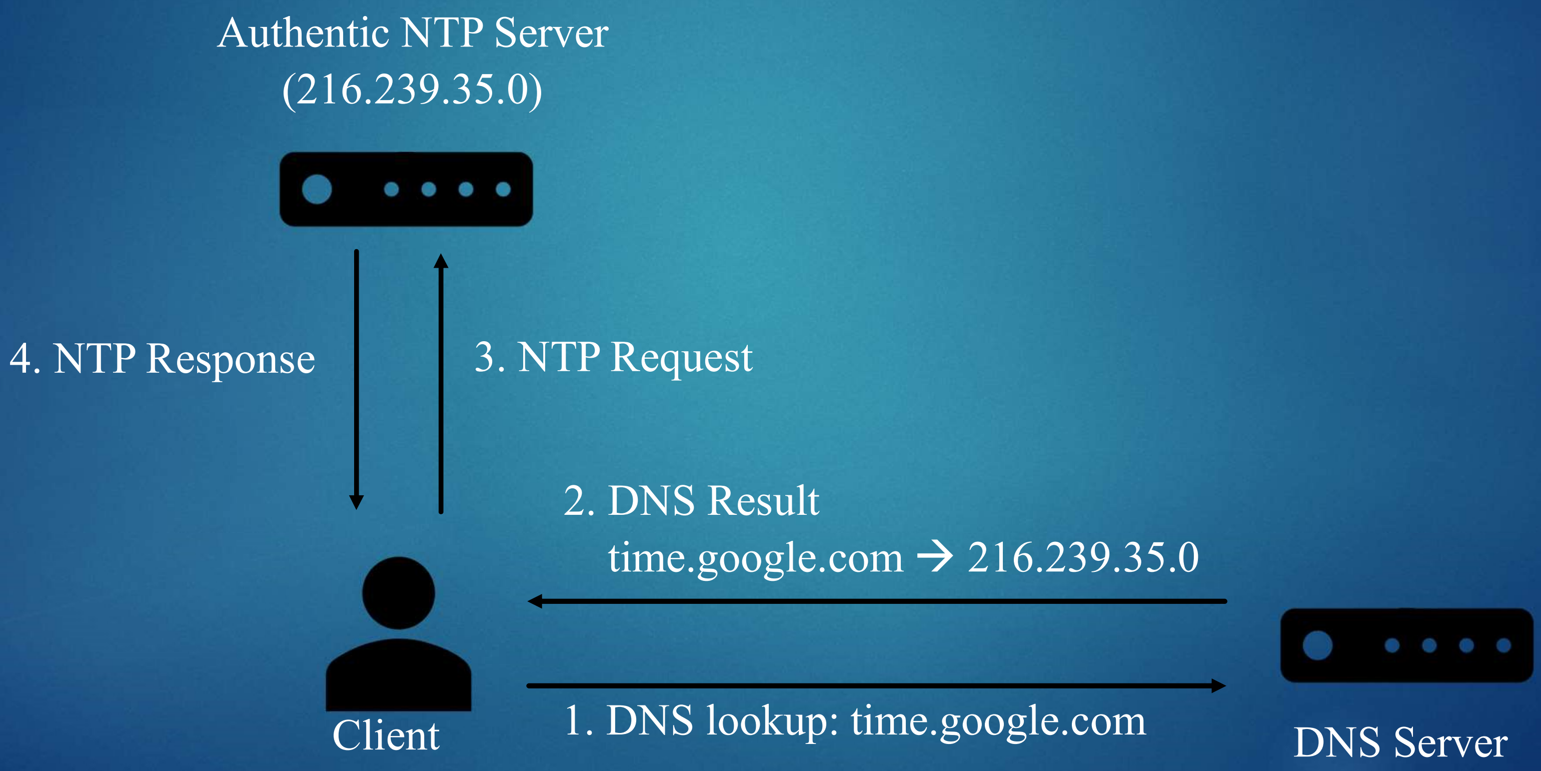
Man-in-the-middle Attacks

- Support symmetric and asymmetric authentication

- Asymmetric authentication
  - Autokey protocol: NTPv4

- On-path Attacks

Man-in-the-middle Attacks



Authentic NTP Server
(216.239.35.0)

4. NTP Response

3. NTP Request

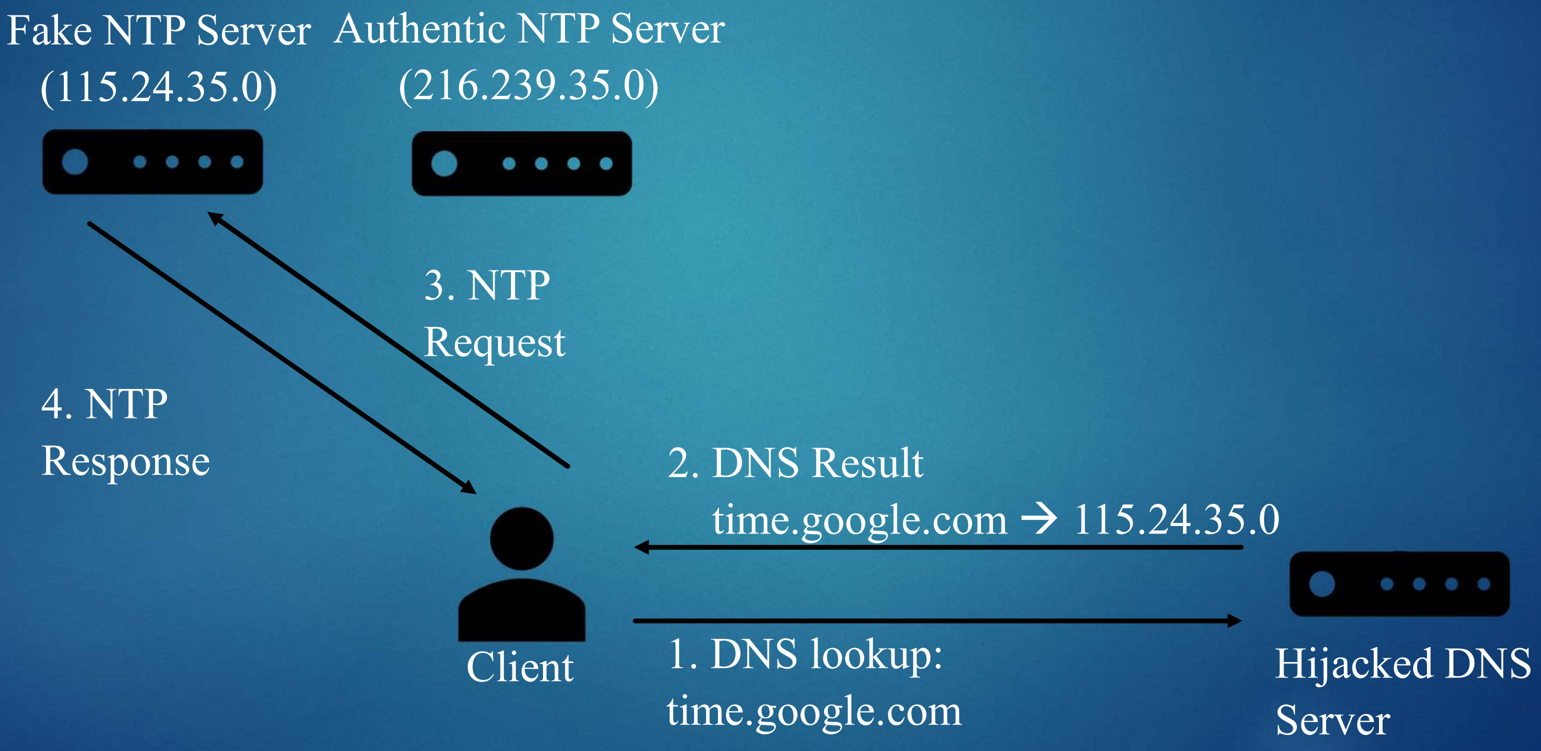
2. DNS Result
time.google.com → 216.239.35.0

Client

1. DNS lookup: time.google.com

DNS Server

Man-in-the-middle Attacks



Fake NTP Server
(115.24.35.0)

Authentic NTP Server
(216.239.35.0)

3. NTP
Request

4. NTP
Response

2. DNS Result
time.google.com → 115.24.35.0

Client

1. DNS lookup:
time.google.com

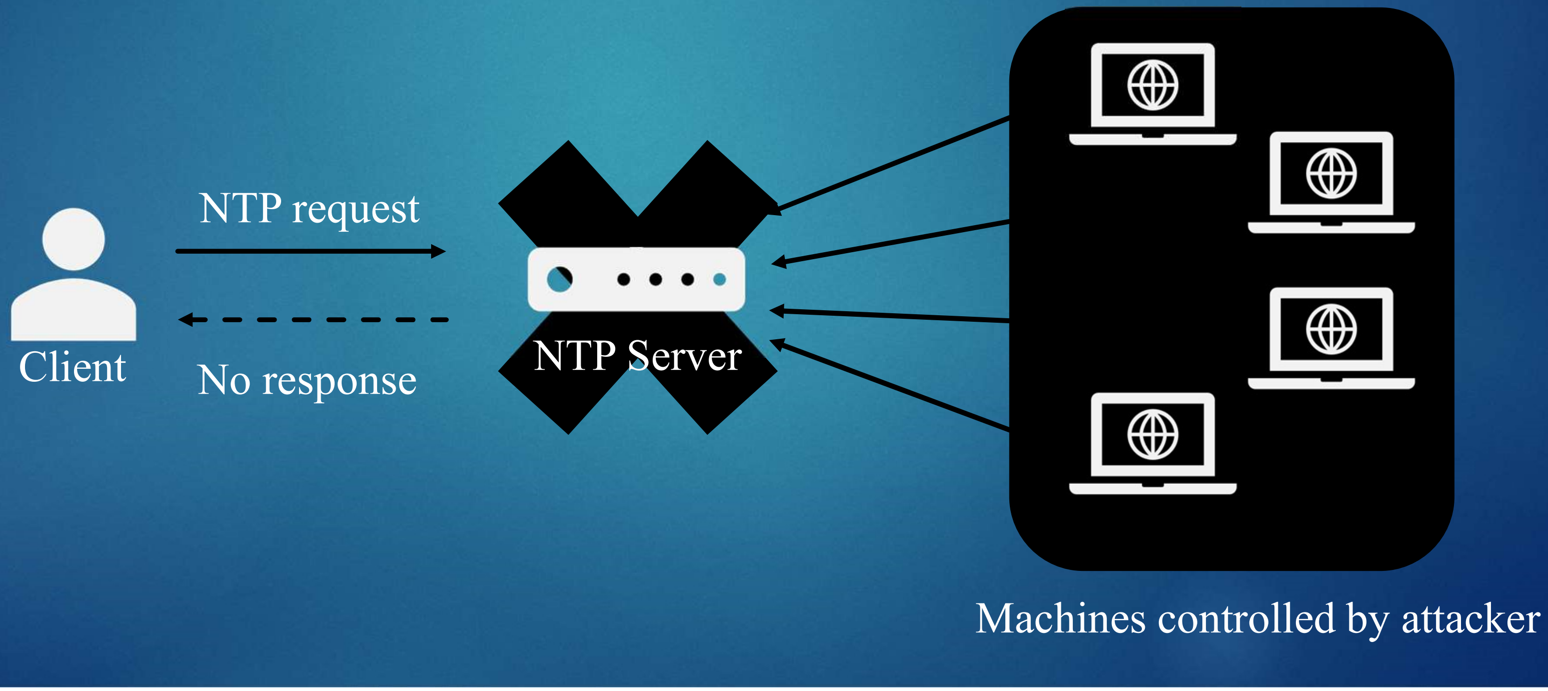
Hijacked DNS
Server

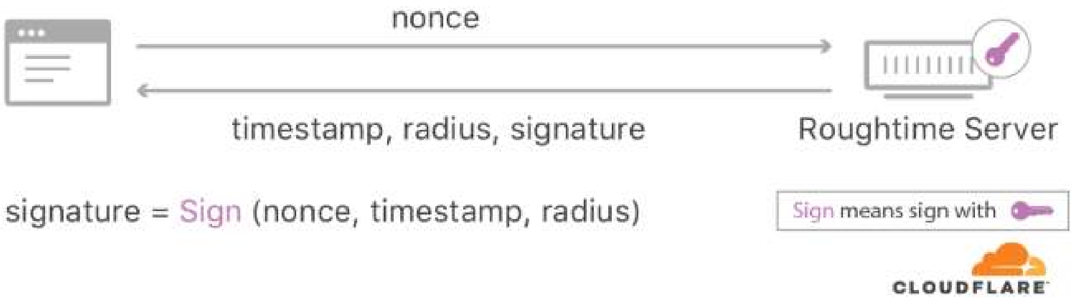
Guarding against wrong time

- 125 ms
- 1000 seconds

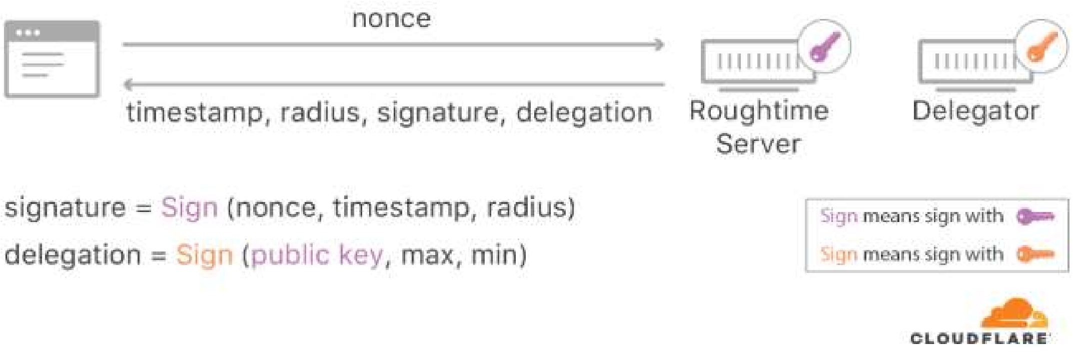
Single point of failure

- Distributed Denial-of-Service Attack




nonce
timestamp, radius, signature
Roughtime Server
signature = Sign (nonce, timestamp, radius)
Sign means sign with
CLOUDFLARE

Simplified Roughtime (without delegation)


nonce
timestamp, radius, signature, delegation
Roughtime Server
Delegator
signature = Sign (nonce, timestamp, radius)
delegation = Sign (public key, max, min)
Sign means sign with
Sign means sign with
CLOUDFLARE

Roughtime with delegation

https://blog.cloudflare.com/roughtime/

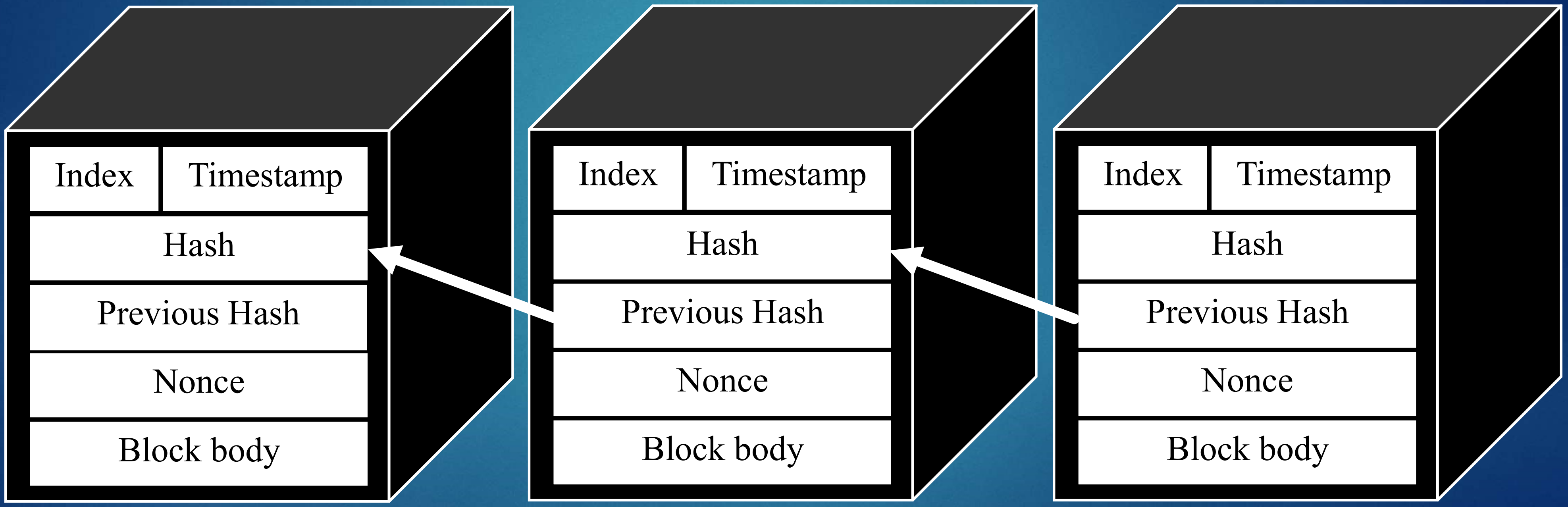
Blockchain

- Decentralized, distributed public ledger

- Appending data
- Verifying data

- Each block are built on top of other blocks

A block



Index Timestamp
Hash
Previous Hash
Nonce
Block body

Index Timestamp
Hash
Previous Hash
Nonce
Block body

Index Timestamp
Hash
Previous Hash
Nonce
Block body

Consensus Algorithm

- Proof of Work (PoW)

	Nonce	Hash
Block content	0001	888B19A43B151683C87895F6211D9F8640F97BDC8EF……
	0002	4FAC6DBE26E823ED6EDF999C63FAB3507119CF3CB……
	0003	446E21F212AB200933C4C9A0802E1FF0C410BBD75F……
	……	
	1234	03AC674216F3E15C761EE1A5E255F067953623C8B38……

How it works

- Each node prepare its own block
- Each node works on PoW
- The node broadcast the block to all nodes reachable
- Nodes that receives a block verify the PoW result and its data (if necessary)
- Block accepted: appends to its chain

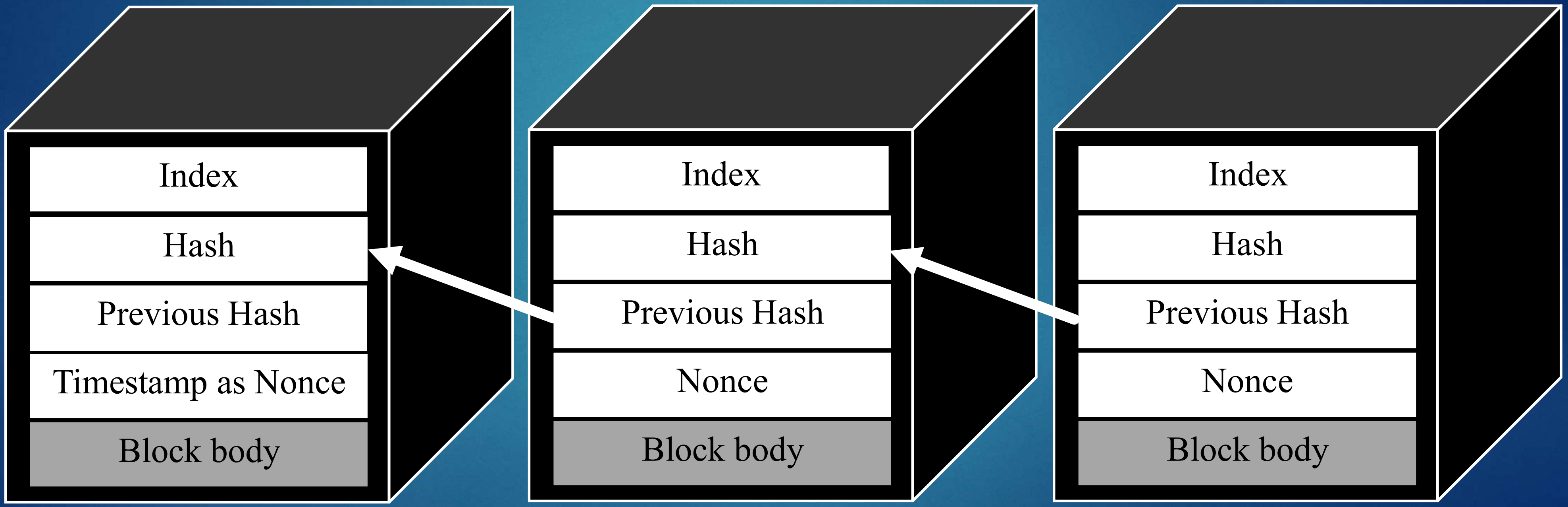
Timechain

- Chain is immutable
- Distributed manner

Time. Why it matters?

- TLS Certificates
  - 6.75% Chrome users have error >24 hours
- Authentication
- Bitcoin

Timechain block



Index

Hash

Previous Hash

Timestamp as Nonce

Block body

Index

Hash

Previous Hash

Nonce

Block body

Index

Hash

Previous Hash

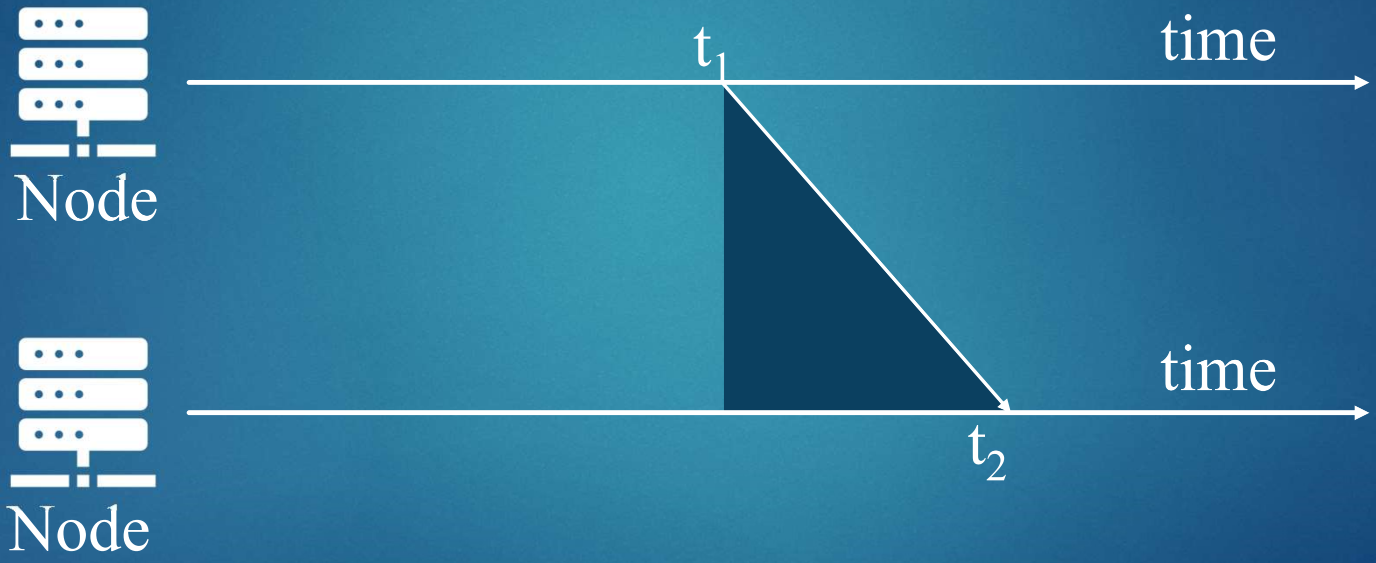
Nonce

Block body

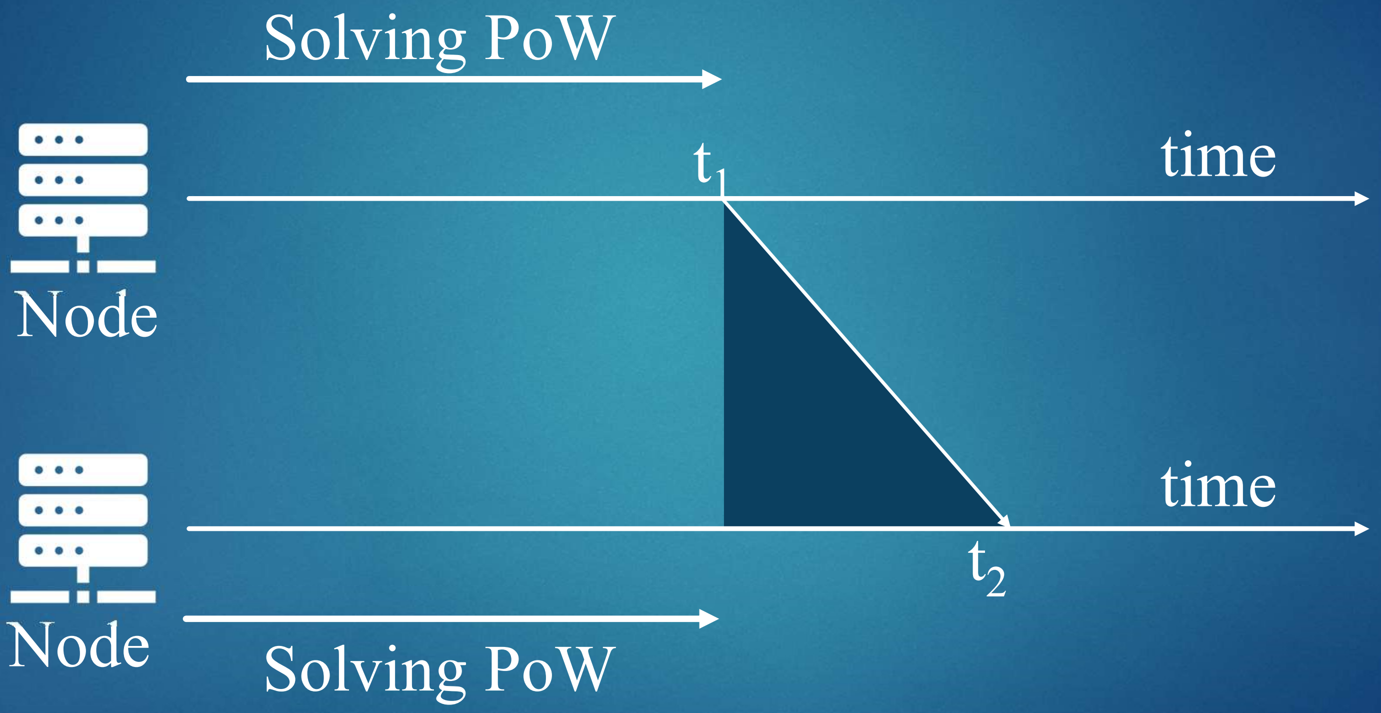
Timechain block

	Timestamp	Hash
Block content	13:24:01.89392 24/12/2018	888B19A43B151683C87895F6211D......
	13:24:01.89393 24/12/2018	4FAC6DBE26E823ED6EDF999C63......
	13:24:01.89394 24/12/2018	446E21F212AB200933C4C9A0802......
	......	
	13:24:05.29348 24/12/2018	03AC674216F3E15C761EE1A5E25......

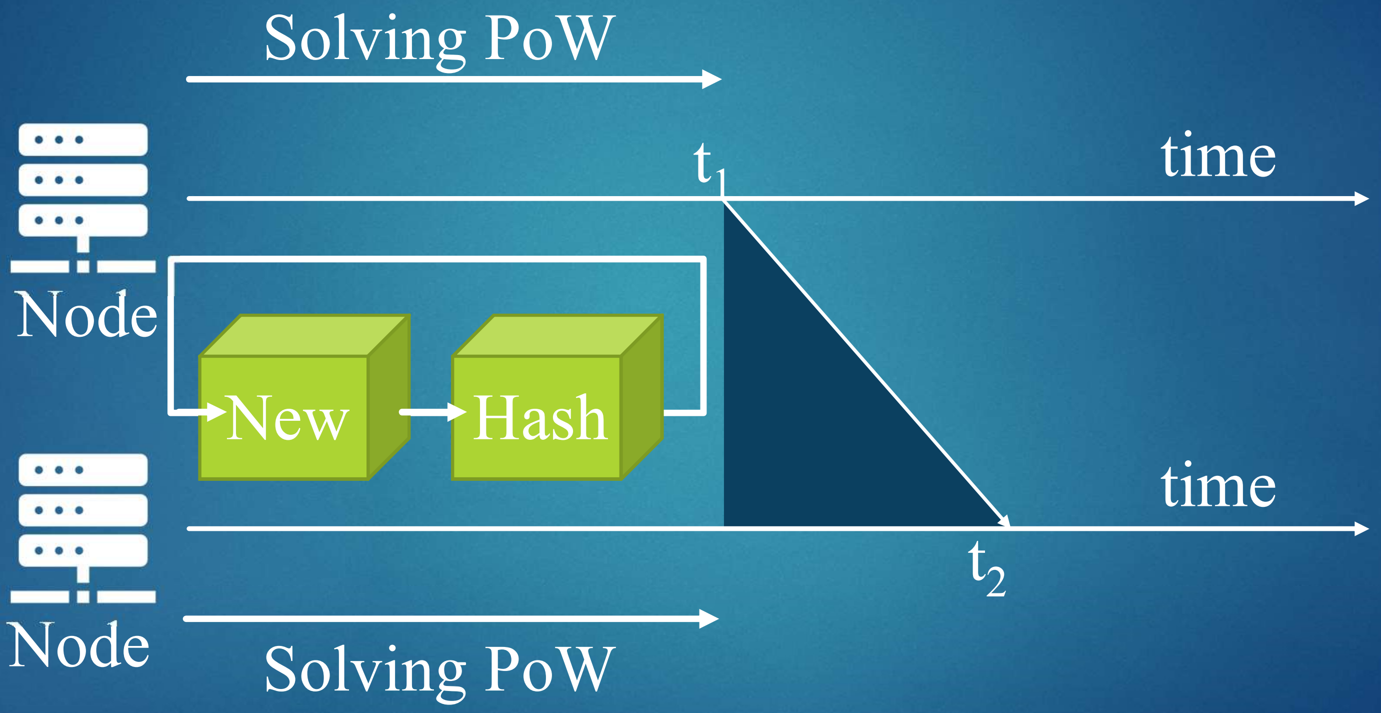
Consensus



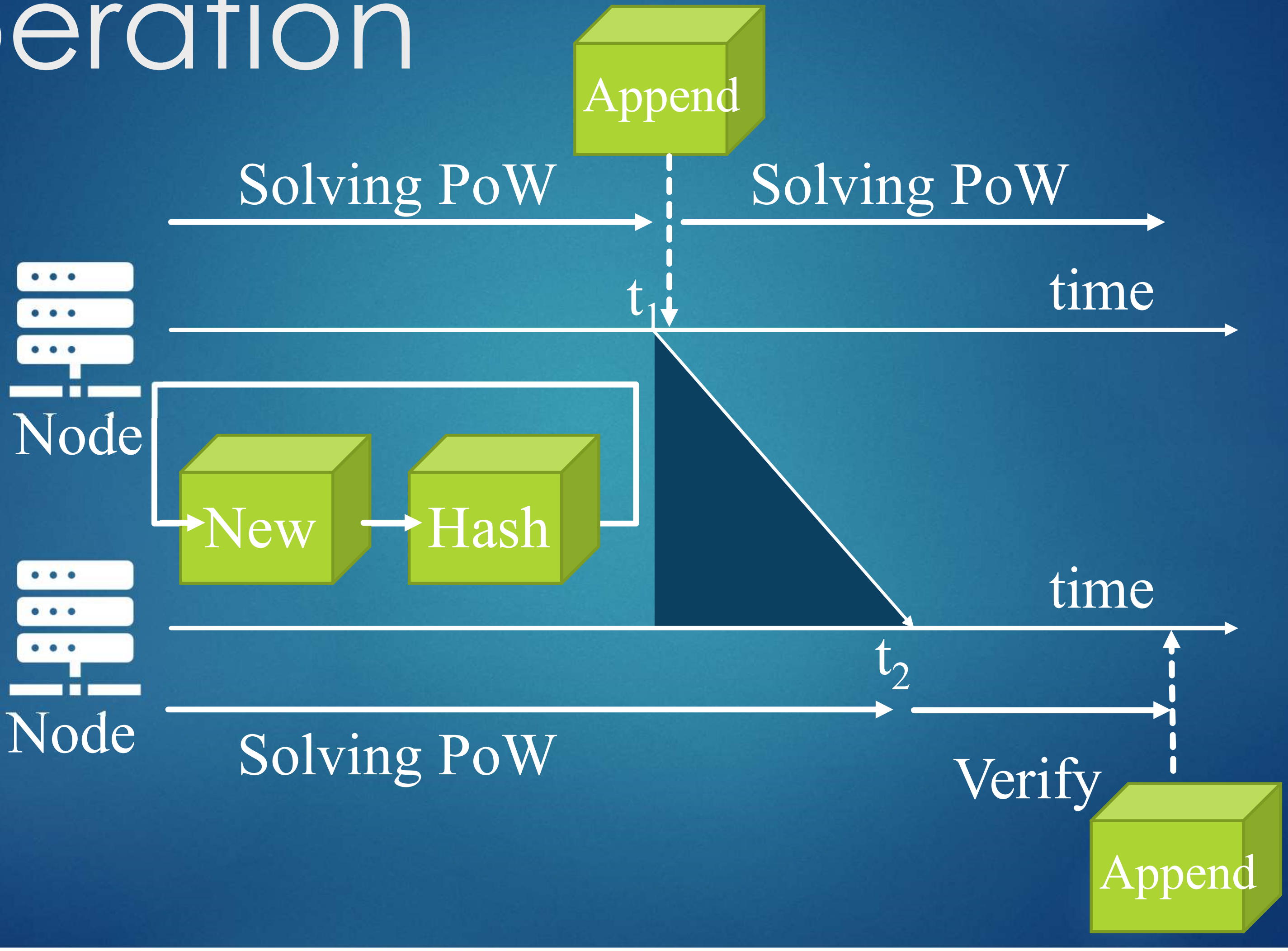
Operation



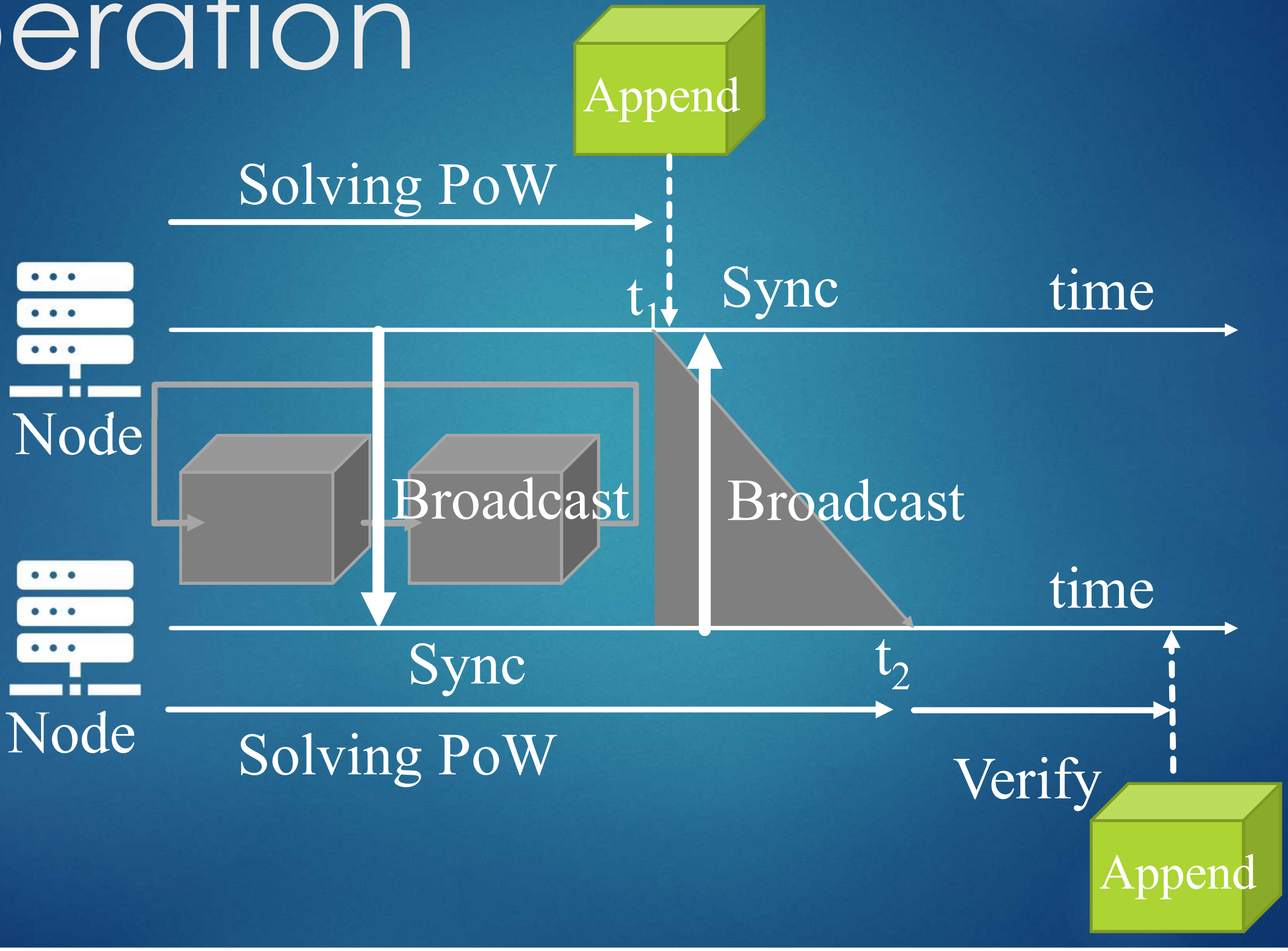
Operation



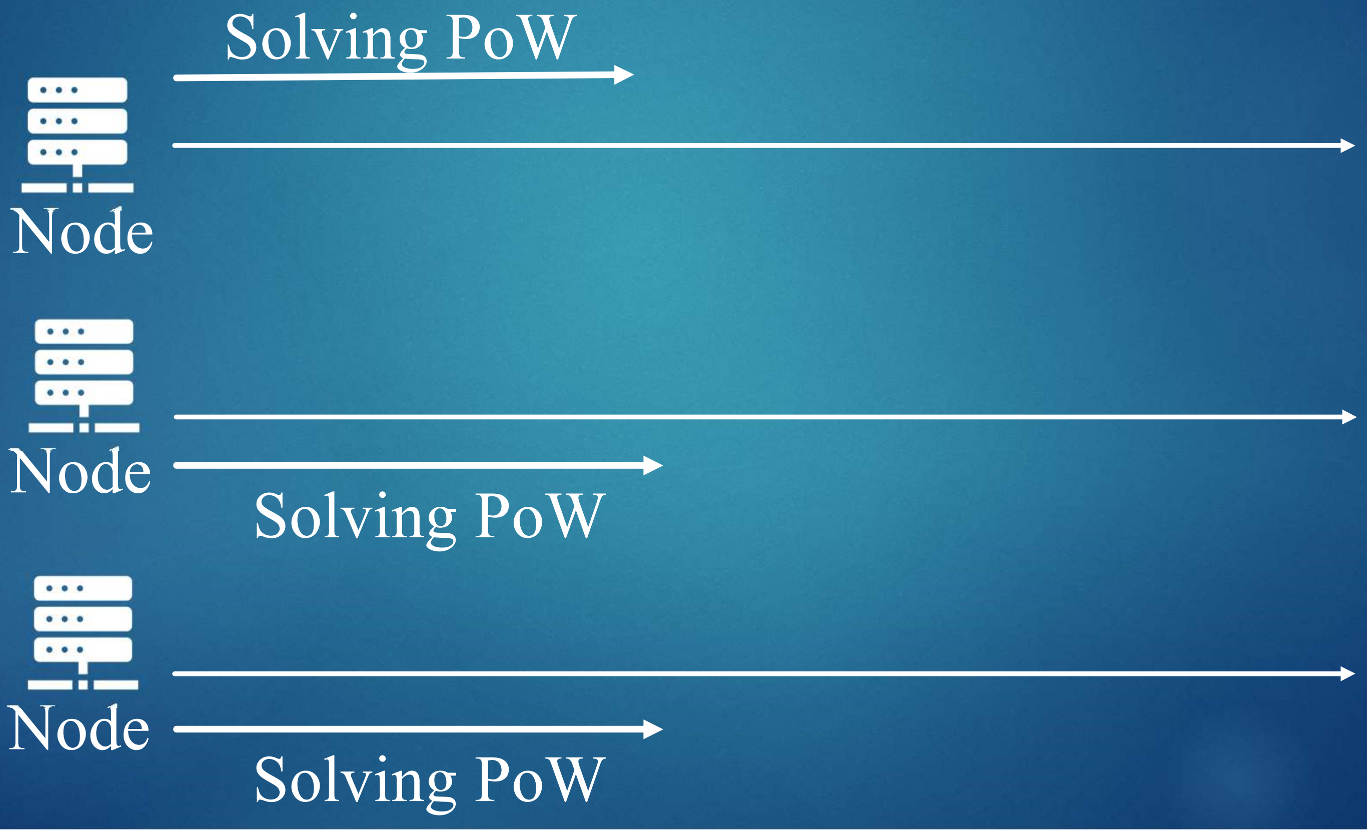
Operation


Append
Solving PoW
Solving PoW
t_1
time
Node
New
Hash
time
Node
t_2
Solving PoW
Verify
Append

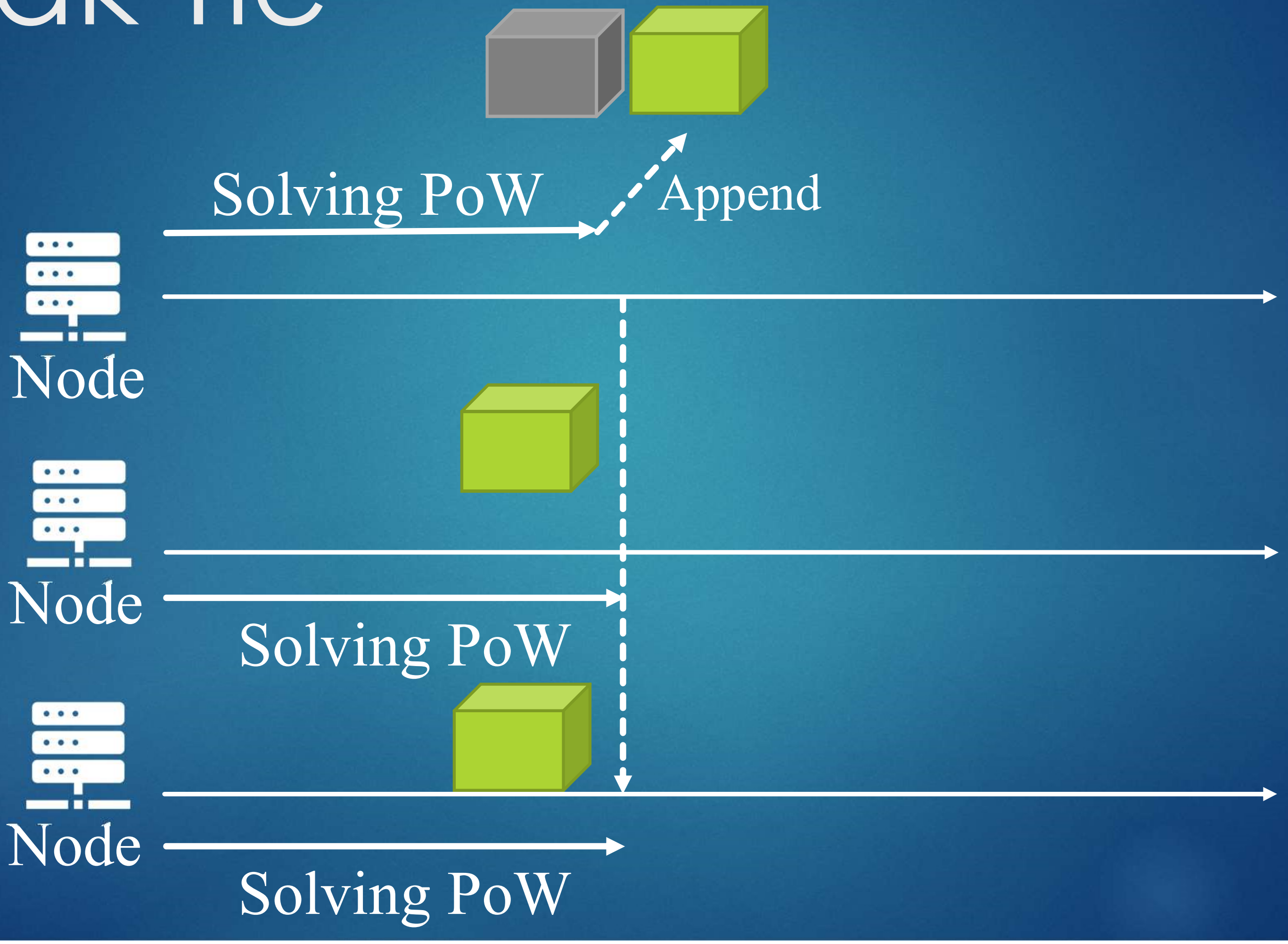
Operation


Append
Solving PoW
t_1
Sync
time
Node
Broadcast
Broadcast
time
Node
Sync
t_2
Solving PoW
Verify
Append

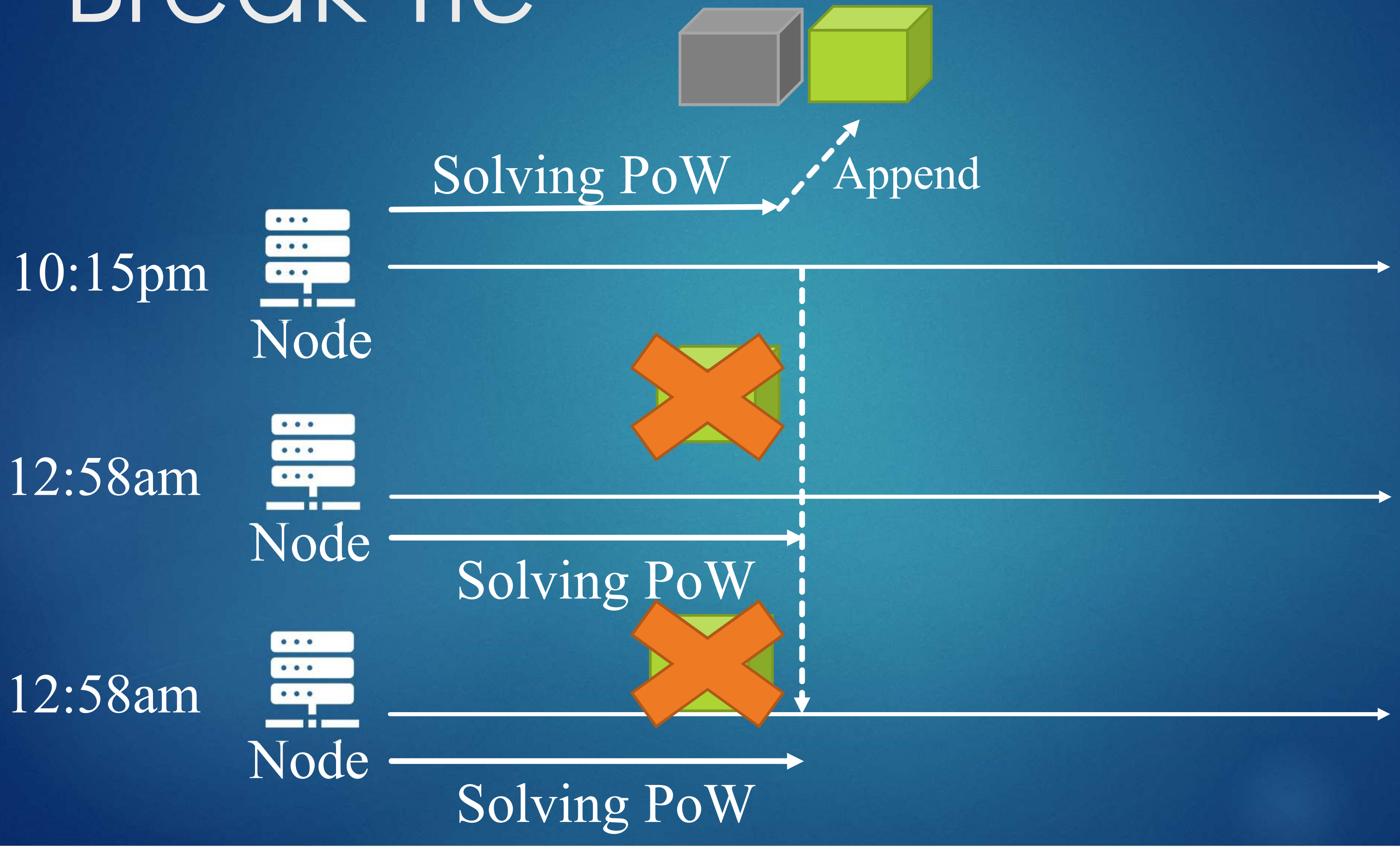
Break Tie



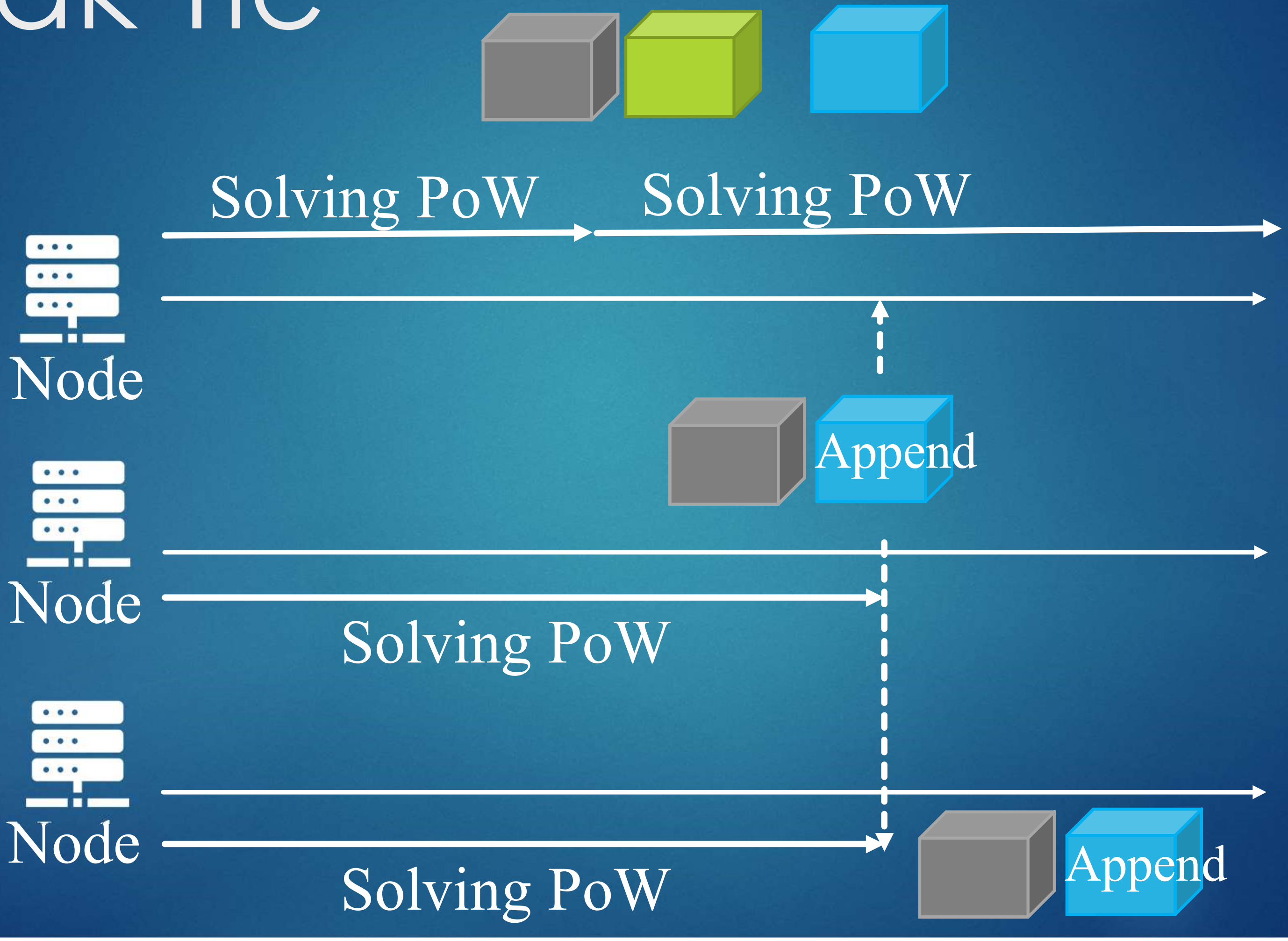
Break Tie



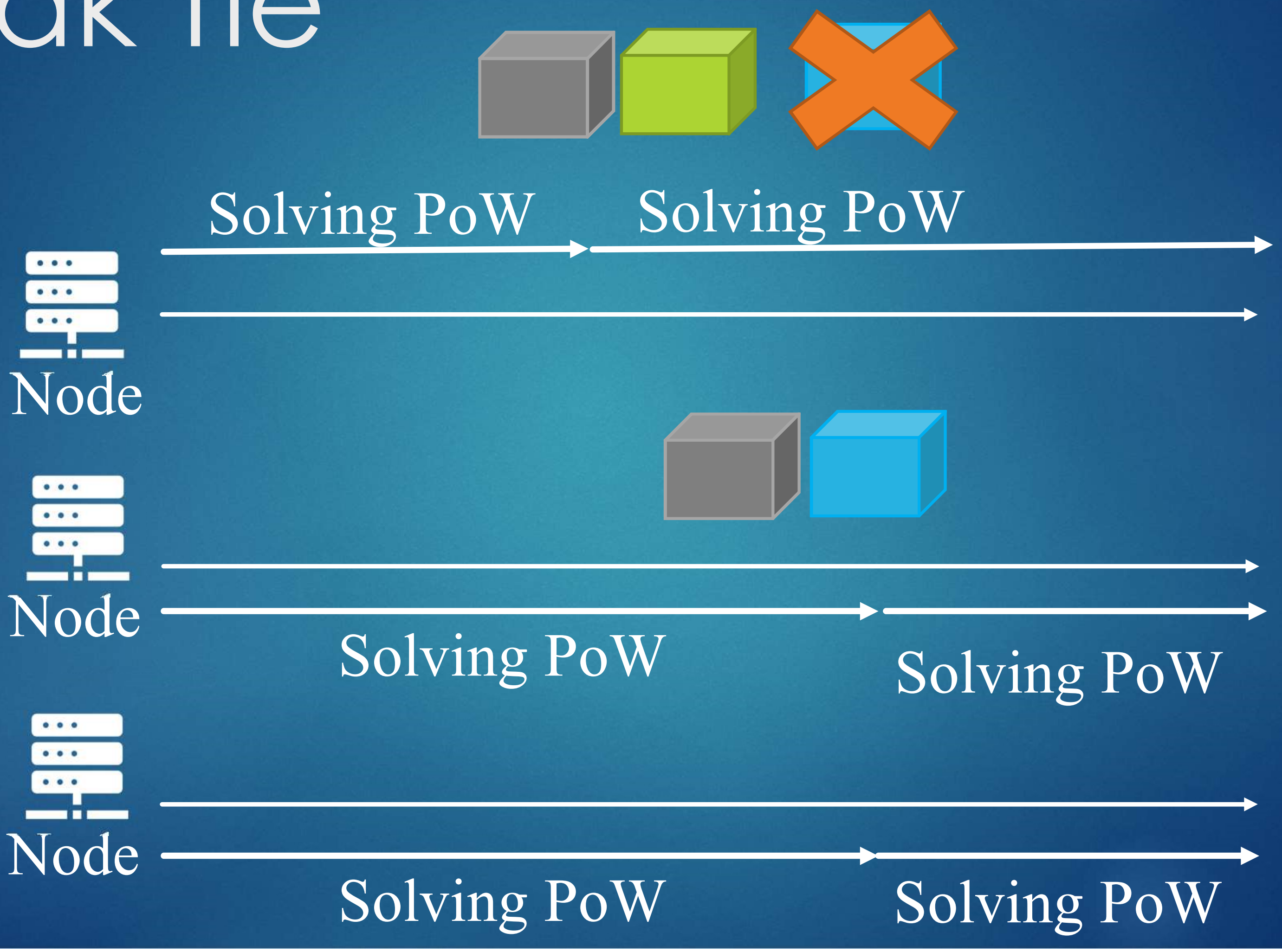
Break Tie


Solving PoW
Append
10:15pm
Node
12:58am
Node
Solving PoW
12:58am
Node
Solving PoW

Break Tie

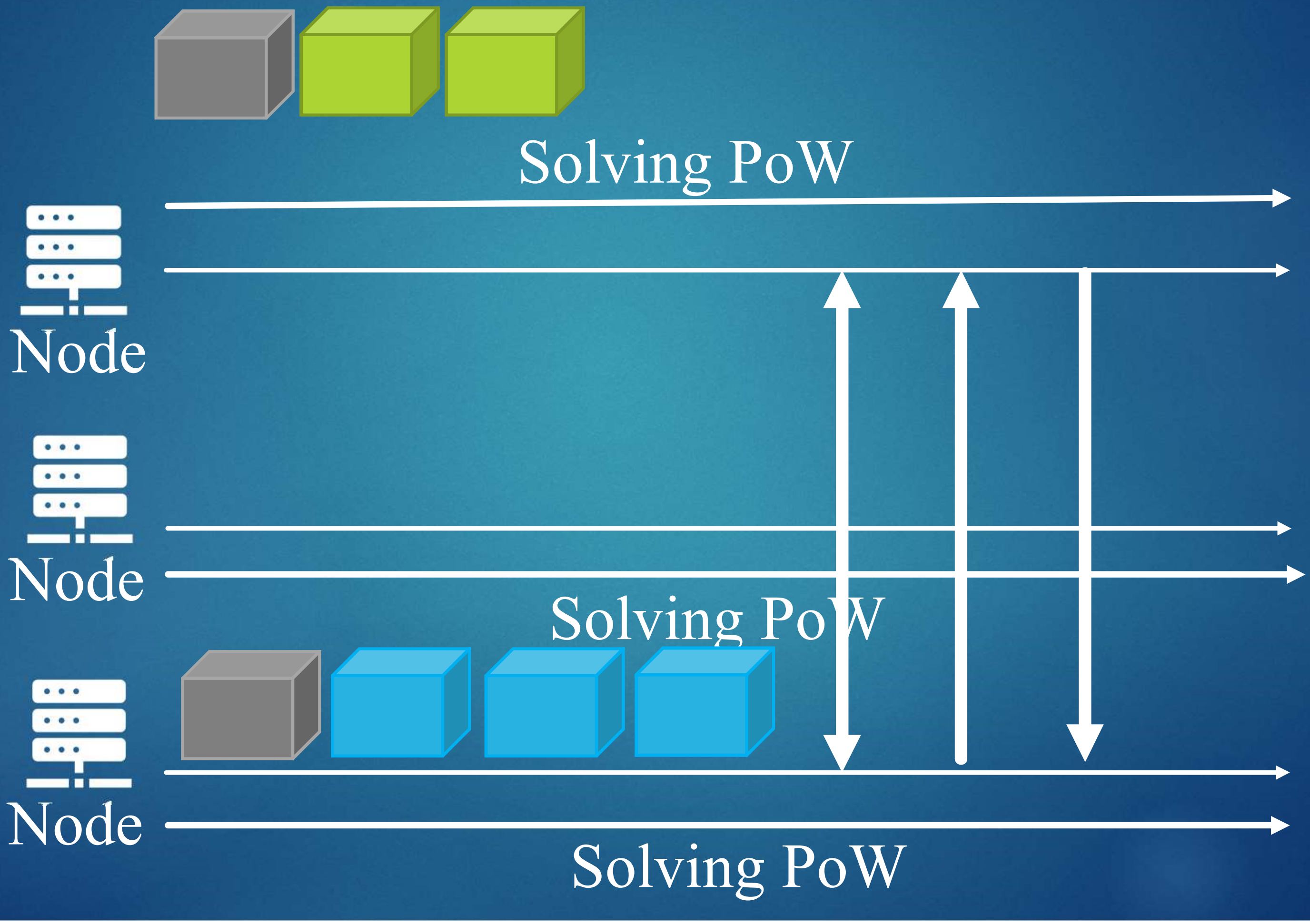

Solving PoW
Solving PoW
Node
Append
Node
Solving PoW
Node
Append
Solving PoW

Break Tie

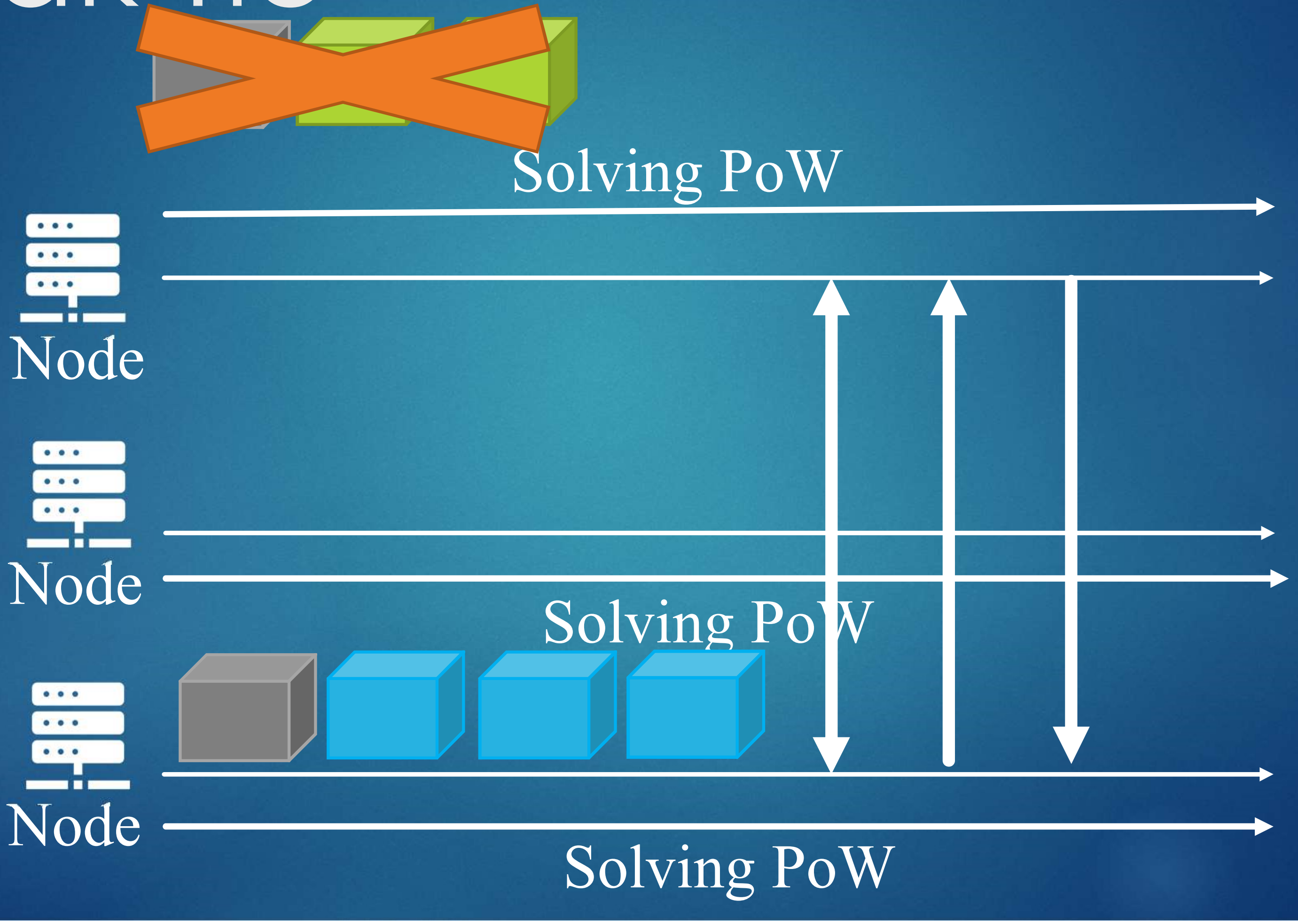


Solving PoW
Solving PoW
Node
Node
Solving PoW
Solving PoW
Node
Solving PoW
Solving PoW

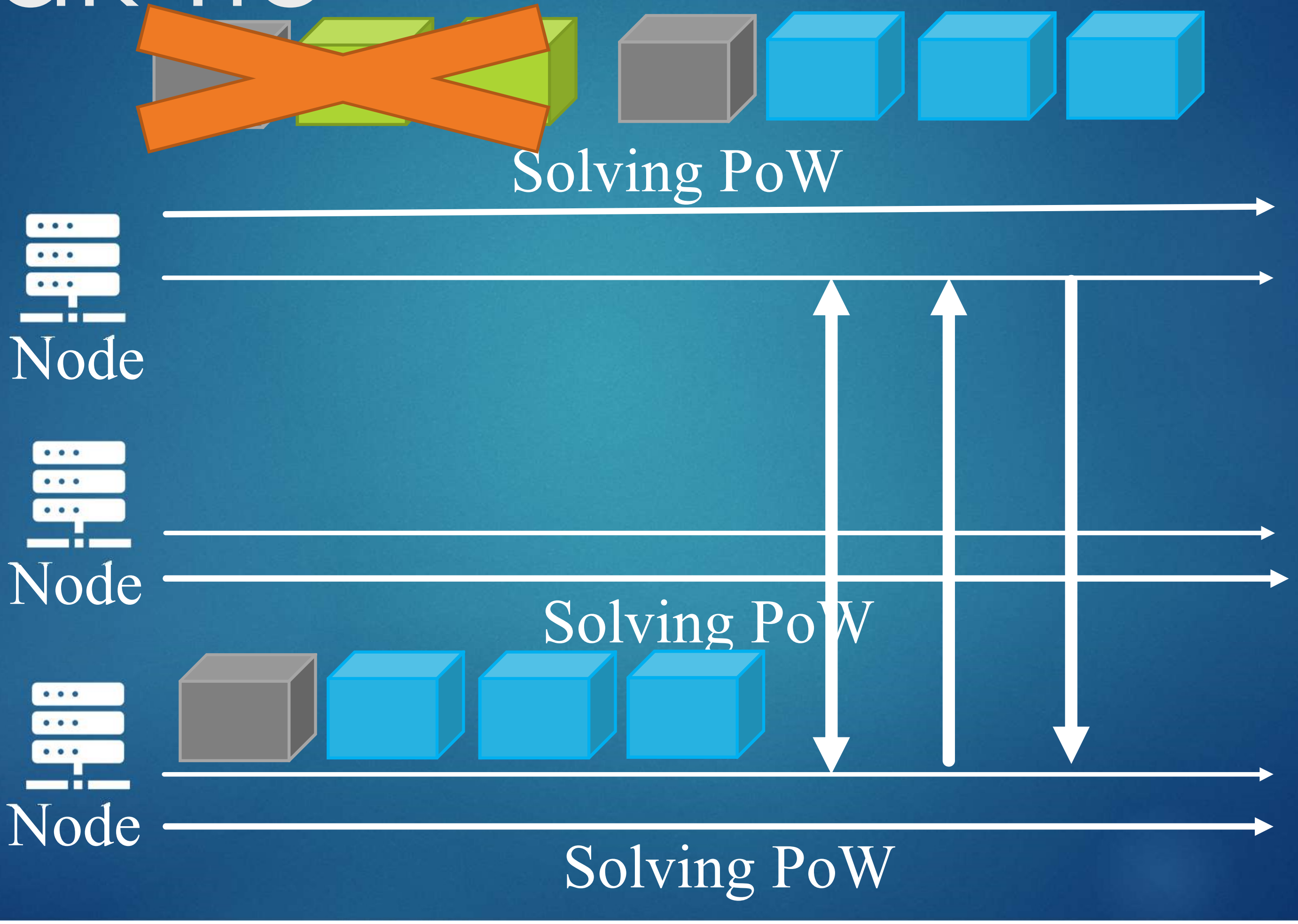
Break Tie



Break Tie



Break Tie



Deploy

- In GoLang
- Creating blockings, chaining up, hashing
- Broadcast blocks, updating chains