An analysis of the applicability of blockchain to secure IP addresses allocation, delegation and bindings

draft-paillisse-sidrops-blockchain-00

IETF 99 - Prague
July 2017

Jordi Paillissé, Albert Cabellos, Vina Ermagan, Alberto Rodríguez, Fabio Maino
jordip@ac.upc.edu

OPEN OVERLAY ROUTER

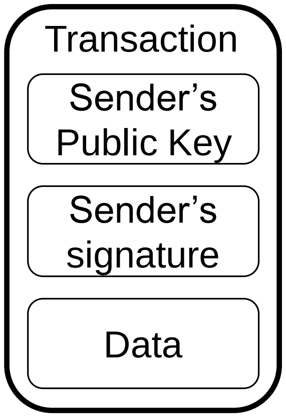
http://openoverlayrouter.org

A short Blockchain tutorial

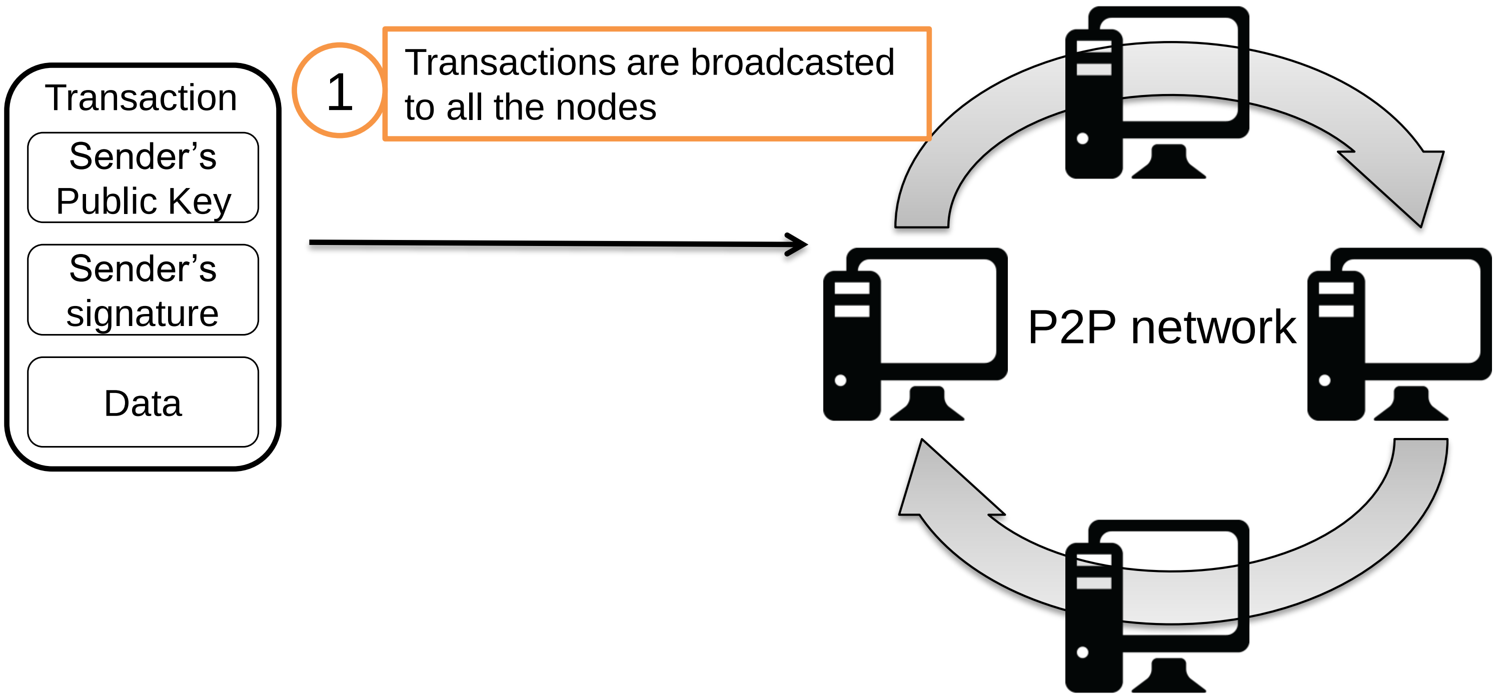
Blockchain - Introduction

- Blockchain:
  – Decentralized, secure and trustless database
  – Token tracking system (who has what)
- Add blocks of data one after another
- Protected by two mechanisms:
  – Chain of signatures
  – Consensus algorithm
- First appeared: Bitcoin, to exchange money
- Other applications are possible

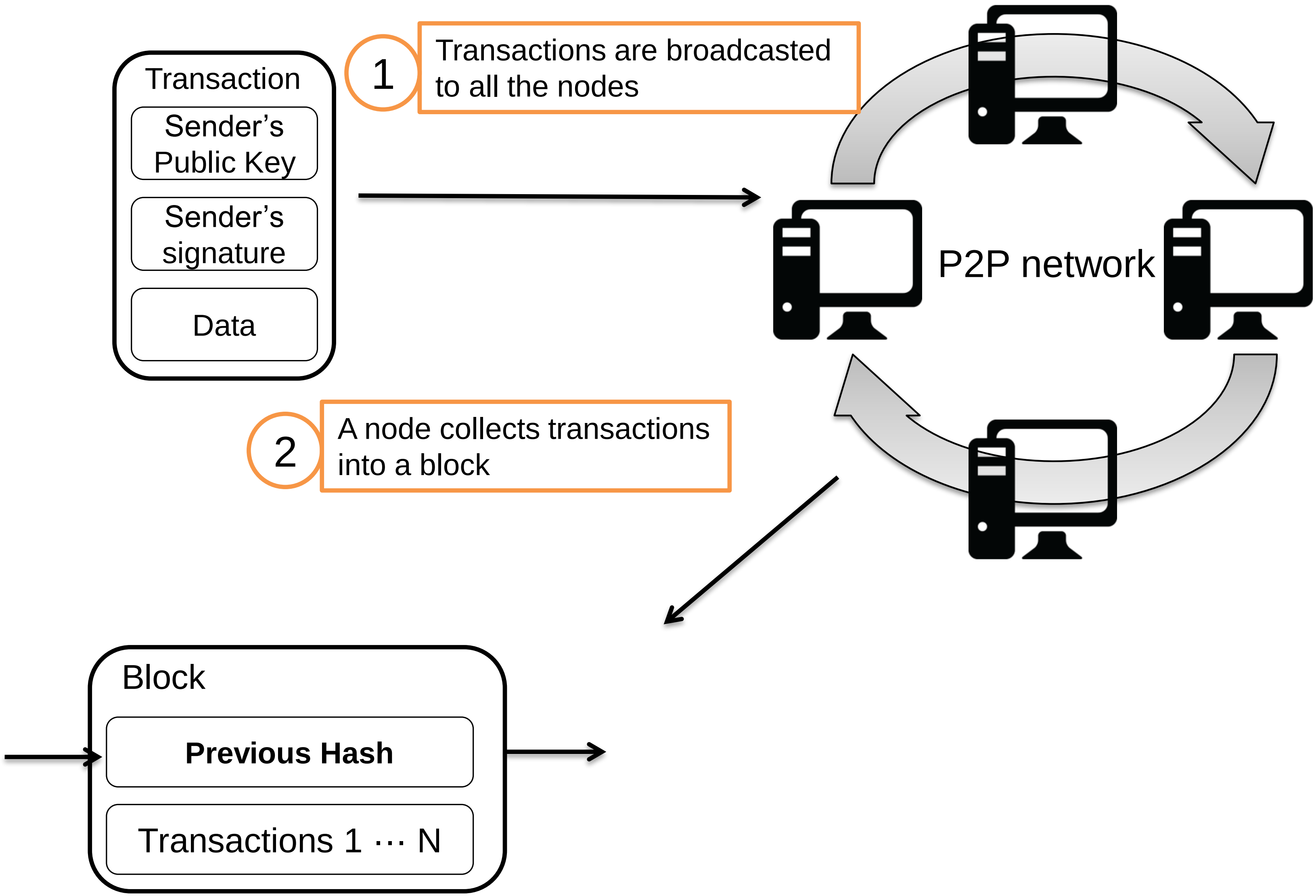
Blockchain - Transactions



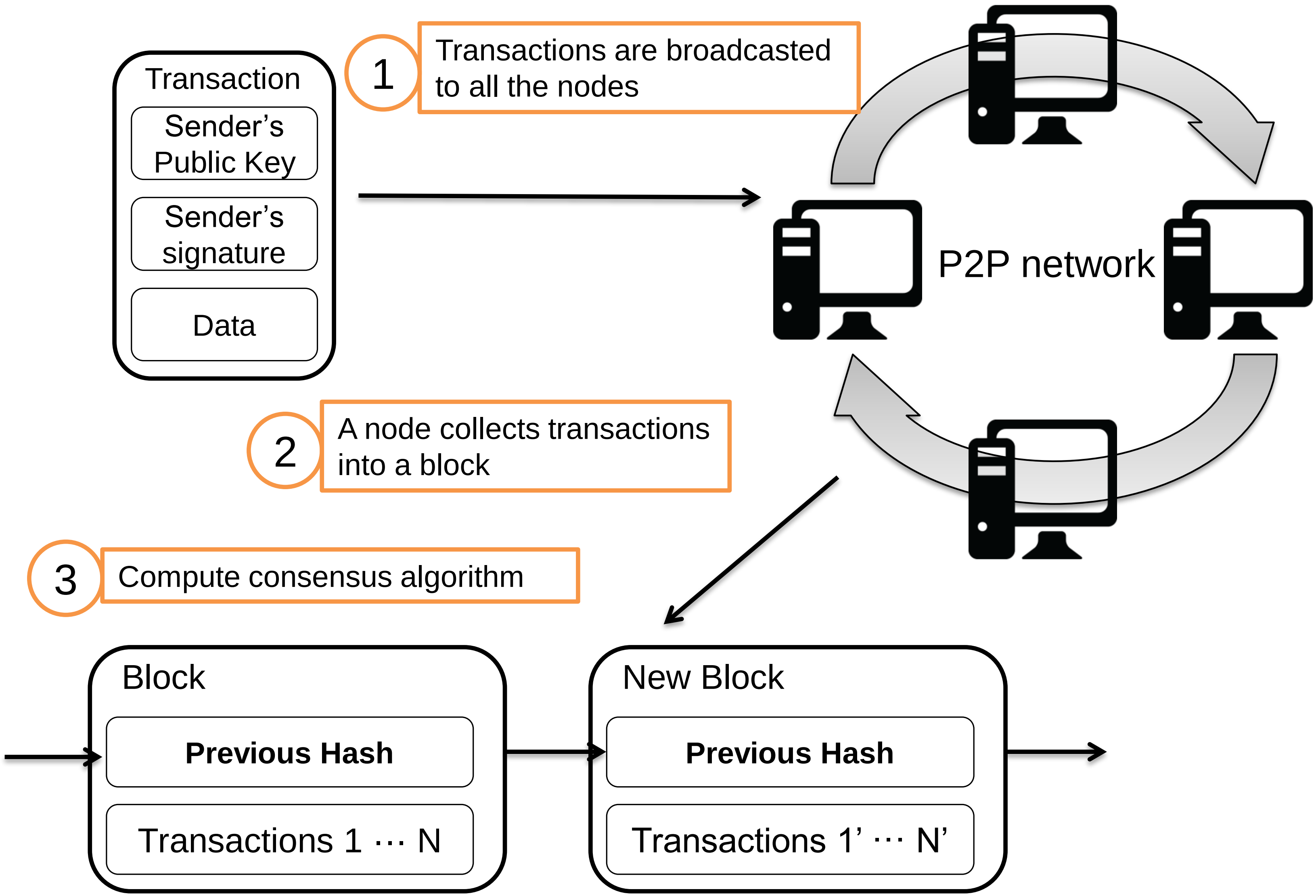
Blockchain - Transactions



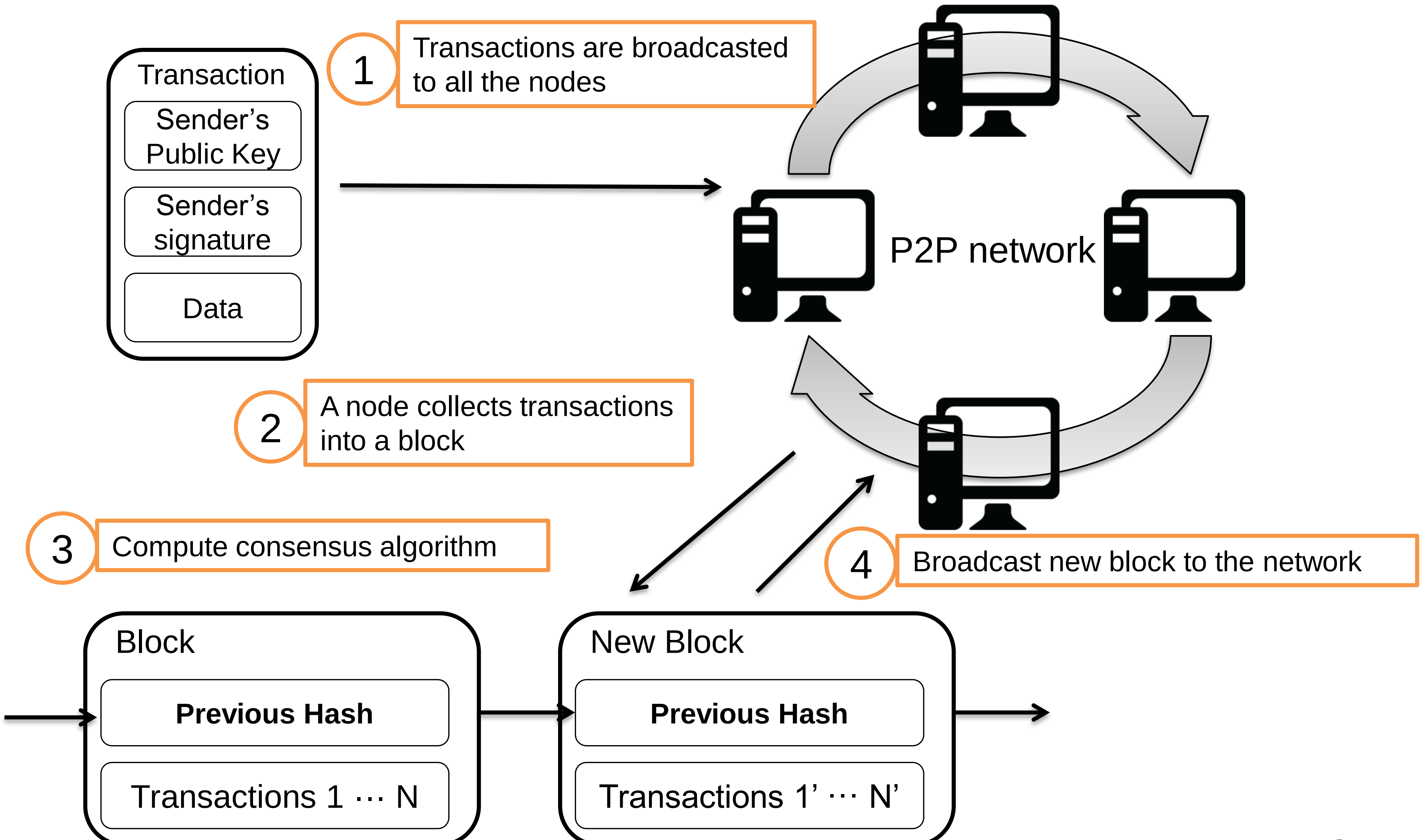
Blockchain - Transactions



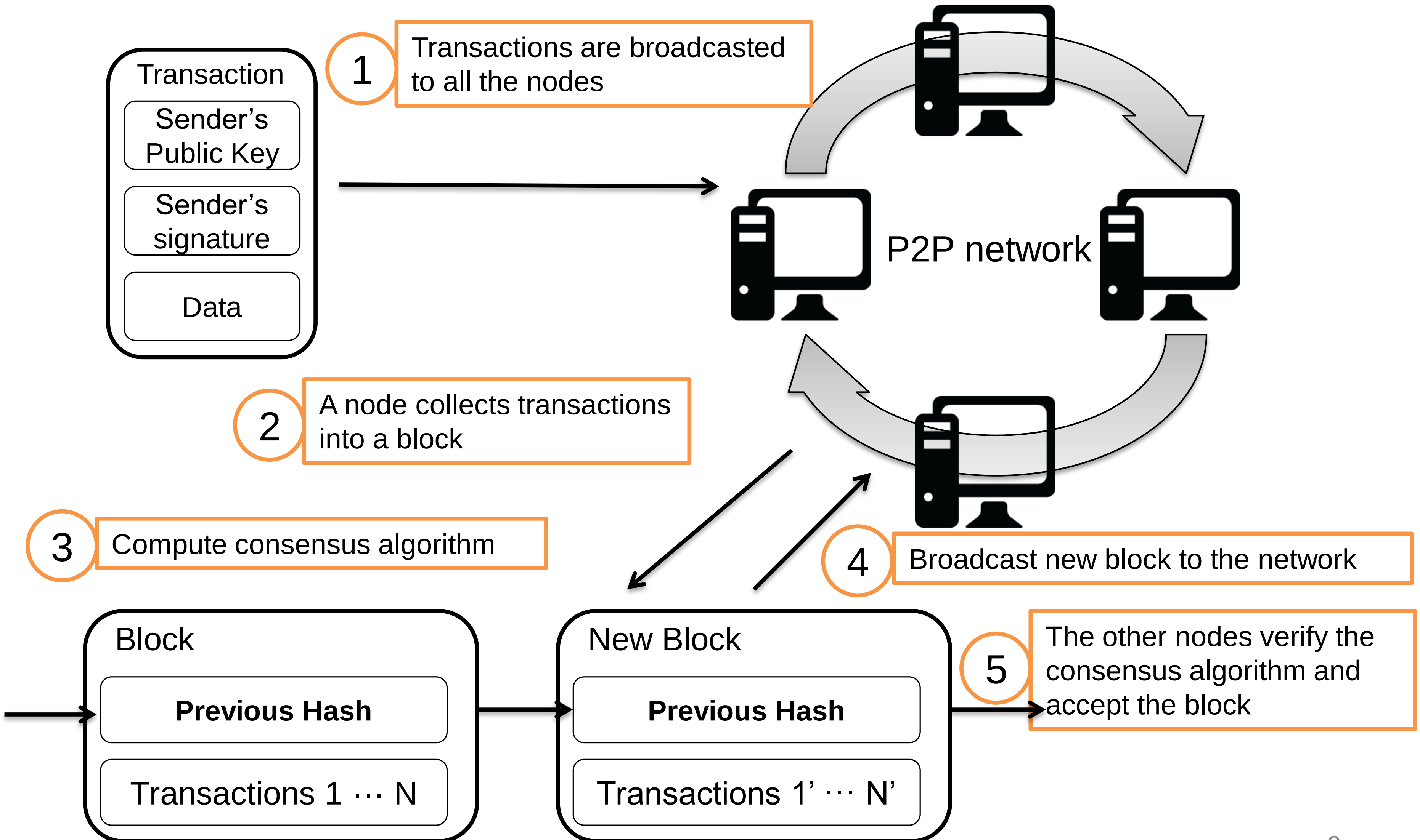
Blockchain - Transactions



Blockchain - Transactions



Blockchain - Transactions



Blockchain - Properties

- Decentralized: all nodes have the entire blockchain
- No prior trust required
- Decouples ownership from identity
- Append-only and immutable: added transactions cannot be modified
- Verifiable

“Blockchain Technology”, Sutardja Center (UC Berkeley)
http://scet.berkeley.edu/wp-content/uploads/BlockchainPaper.pdf

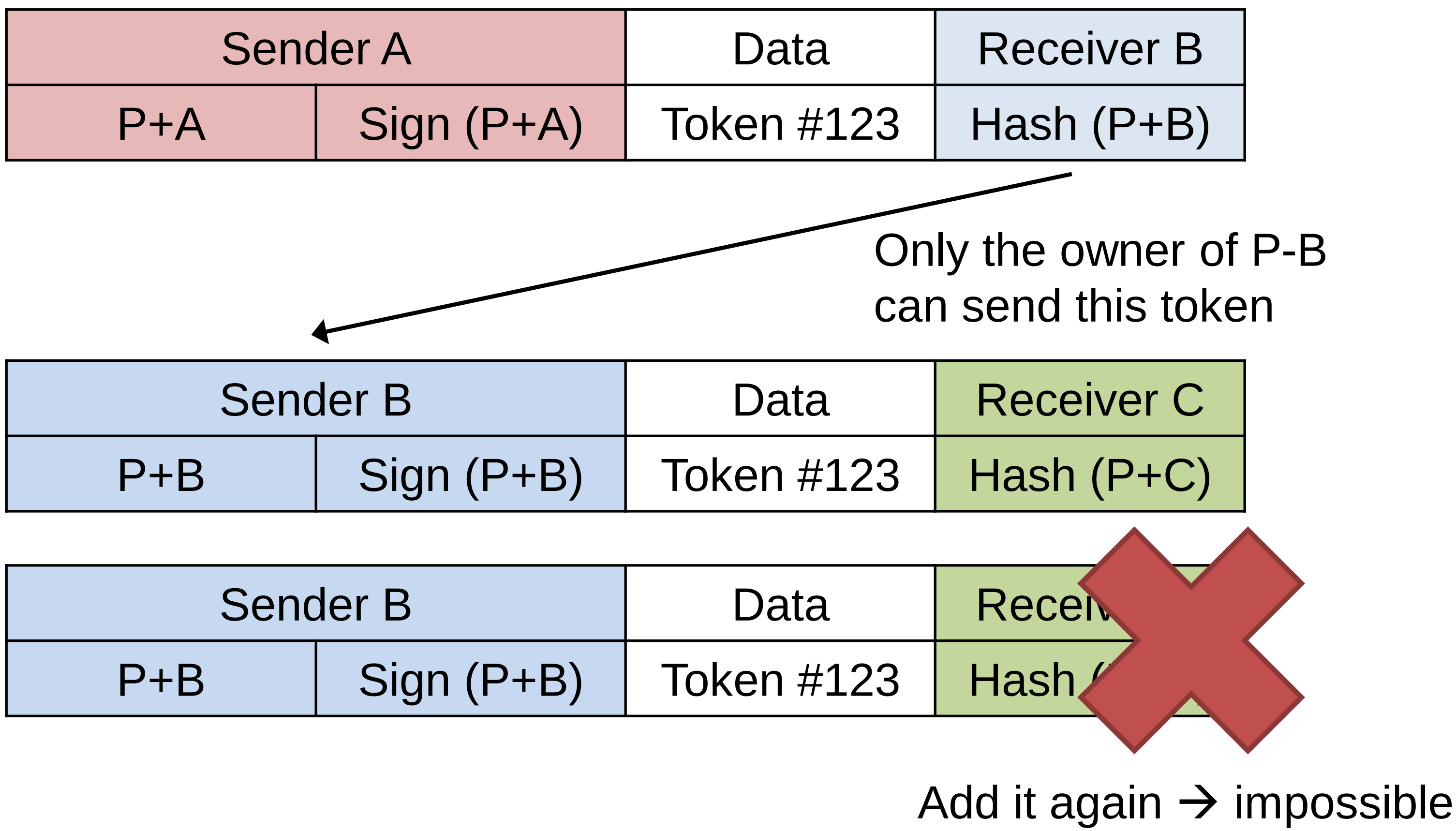
Chain of signatures

Sender A		Data	Receiver B
P+A	Sign (P+A)	Token #123	Hash (P+B)

Only the owner of P-B
can send this token

Sender B		Data	Receiver C
P+B	Sign (P+B)	Token #123	Hash (P+C)

Chain of signatures



Consensus algorithm

- Central part of blockchains
- Controls addition of blocks
- Defines what is consensus
- Most common:
  - Proof of Work, e.g. Bitcoin
  - Proof of Stake, e.g. Ethereum (shorty)

Proof of Work

- Perform a large number of calculations
- Eg: find nonce so that:

Bruteforce!!

SHA-256 (transactions +
hash (prev. Block) +
nonce) = 00000000xxxxxxxxxxxxx

- Change data → redo Proof of Work
- Accumulate computing power

0	1	2	...	...	300	301	302	303	Last block

impossible hard easy

- Not necessarily performed by the users of the blockchain

Proof of Stake

- Any owner of tokens can add a block
- Selected randomly
- Users with more tokens are more likely to be selected
  - Reduced incentive to attack (because they use the blockchain)
- Attacks are different than PoW

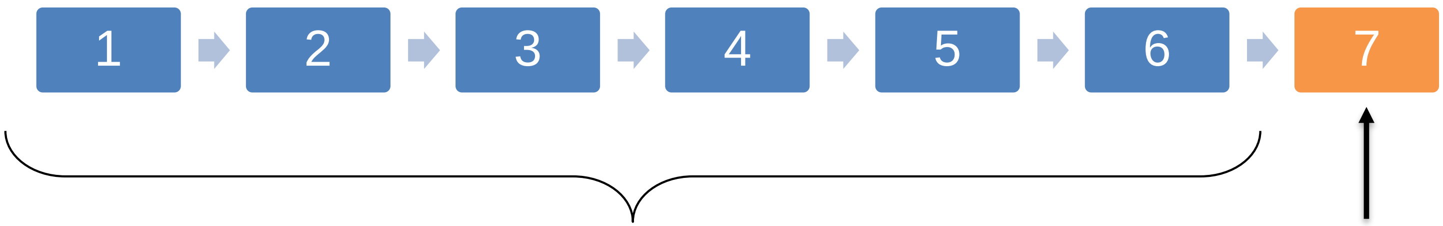
Proof of Stake

1 → 2 → 3 → 4 → 5 → 6

List all stake

Holder	# tokens
A	124
E	110
B	87
D	75
F	54
C	3

Proof of Stake



List all stake

Holder	# tokens
A	124
E	110
B	87
D	75
F	54
C	3

Sign new block
(E priv. key)

E

Weighted
random selection

Summary of features

vs. traditional PKI systems

Advantages

- Decentralized
- No CAs
- Simplified management
- Simple rekeying
- Limited prior trust
- Auditable
- Censorship-resistant

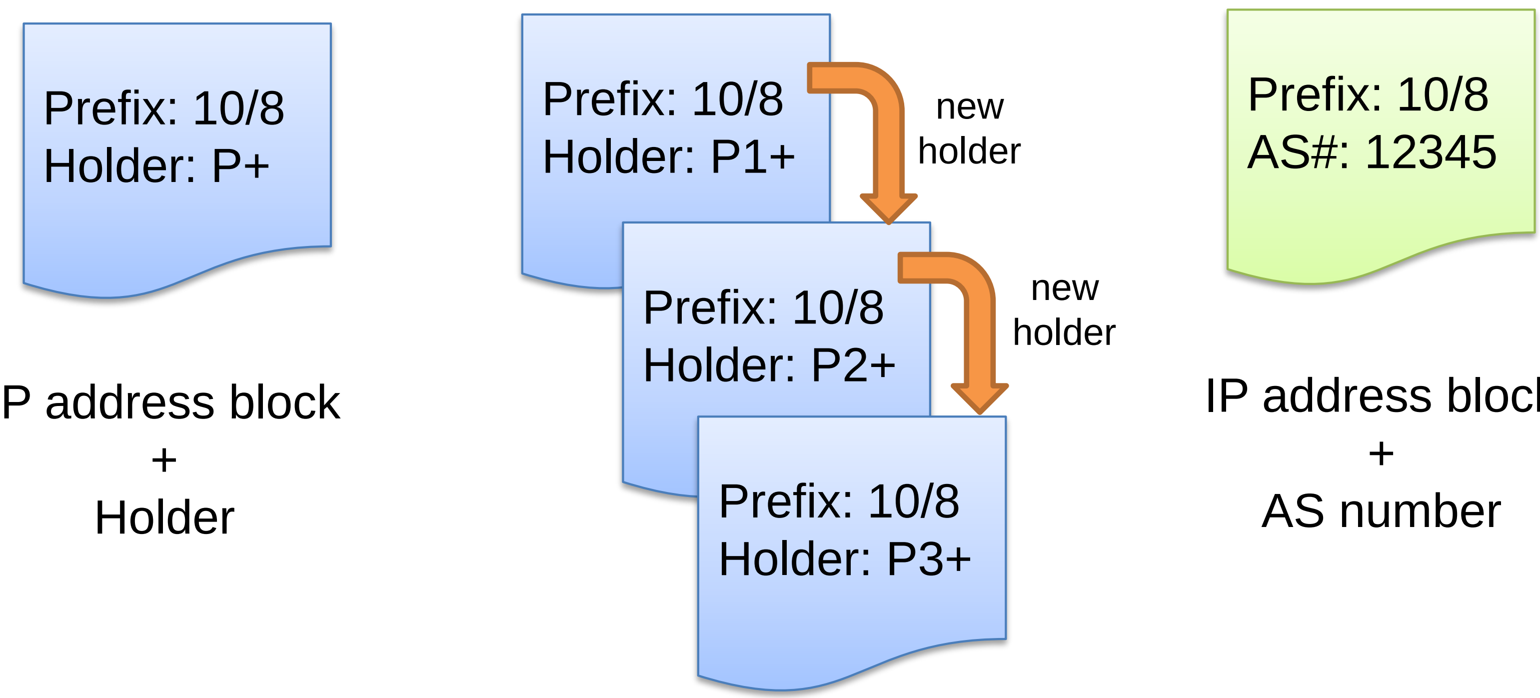
Drawbacks

- No crypto guarantees
- Large storage
- Costly bootstrapping

Blockchain for IP addresses

Data in the blockcahin

We want to store:



IP address block
+
Holder

Chain of allocations
and delegations

IP address block
+
AS number

IP addresses vs. coins

- IP addresses = coins
- Similar properties:
  - Unique
  - Transferrable
  - Divisible
- Exchange blocks of IP addresses just like coins

Which consensus algorithm?

- PoW presents some drawbacks:
  - Parties that add blocks do not necessarily use the blockchain
  - Takeover if enough computing power
  - Hardware dependency
  - Energy inefficiency

AntMiner S7

Advertised Capacity:
4.73 Th/s

Power Efficiency:
0.25 W/Gh

Weight:
8.8 pounds

Guide:
Yes

Price:
$479.95

Buy from amazon.com

Appx. BTC Earned Per Month:
0.1645

https://www.bitcoinmining.com/

Which consensus algorithm?

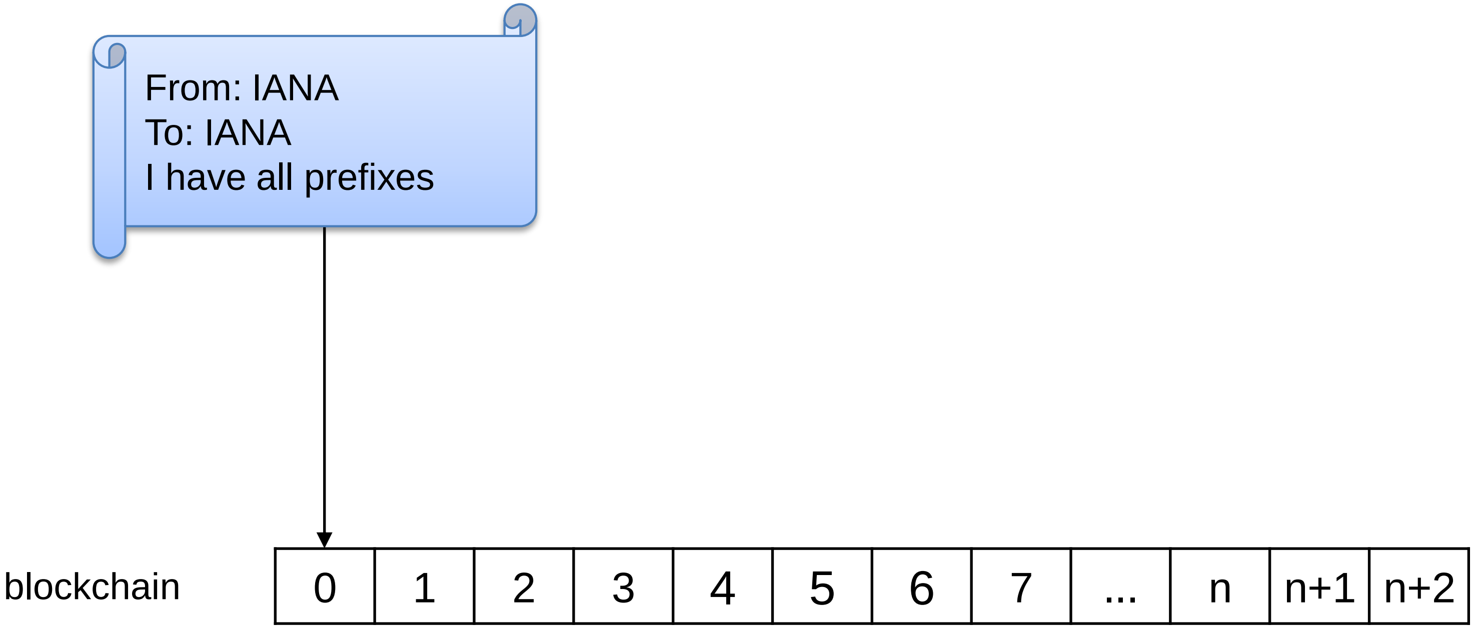
- PoS appears to be more suitable for this scenario:
  - No special hardware
  - No expensive computations
  - Parties with more IP addresses control the blockchain
  - Users of the blockchain maintain it

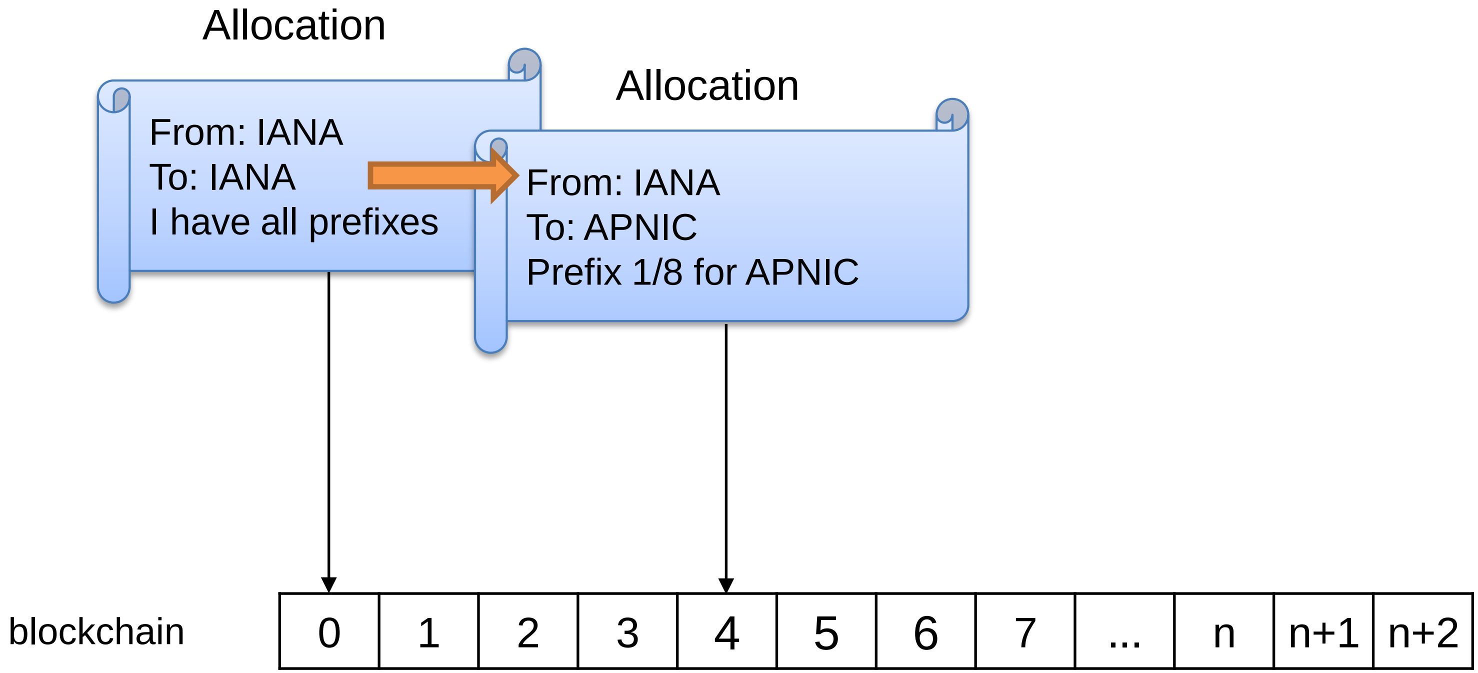
Why Proof of Stake?

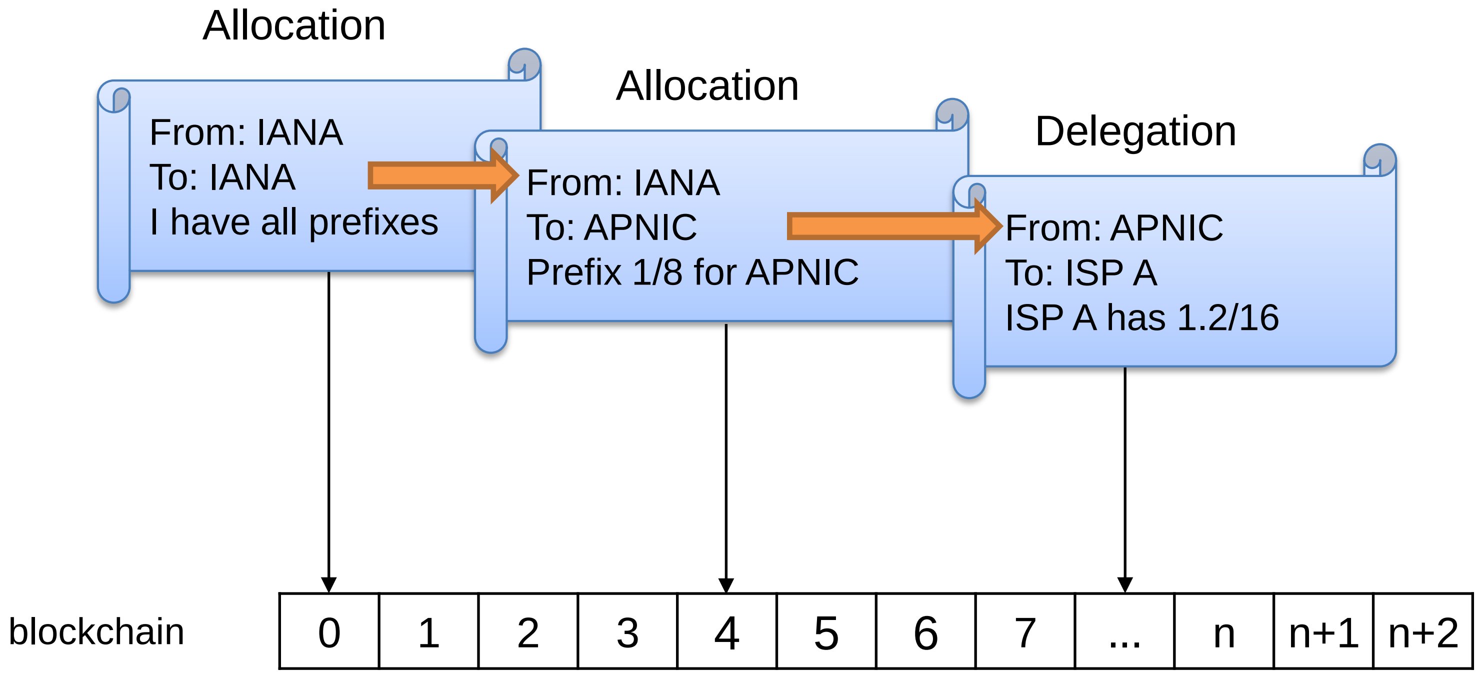
- PoS appears to be more suitable for this scenario:
  - Takeover requires accumulating a large amount of IP blocks
  - Participants do not have an incentive to sell IP blocks to an attacker

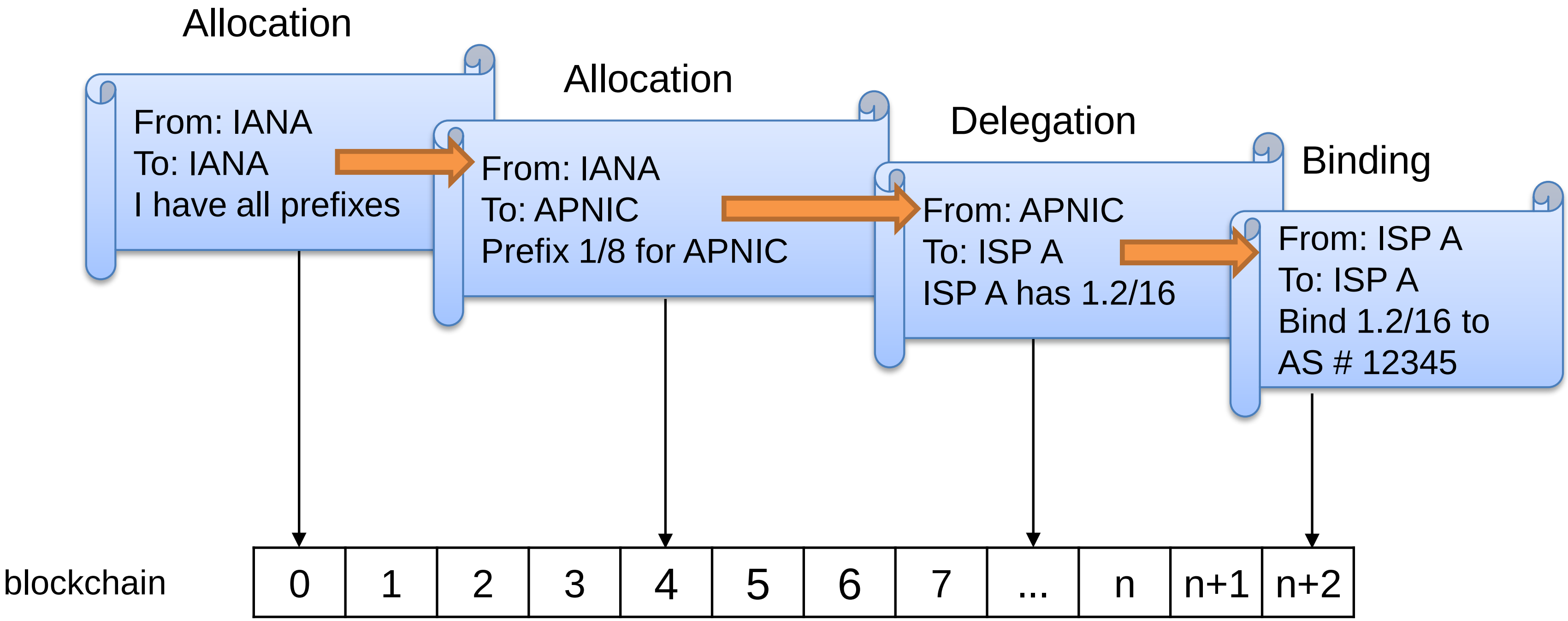
Example

Allocation

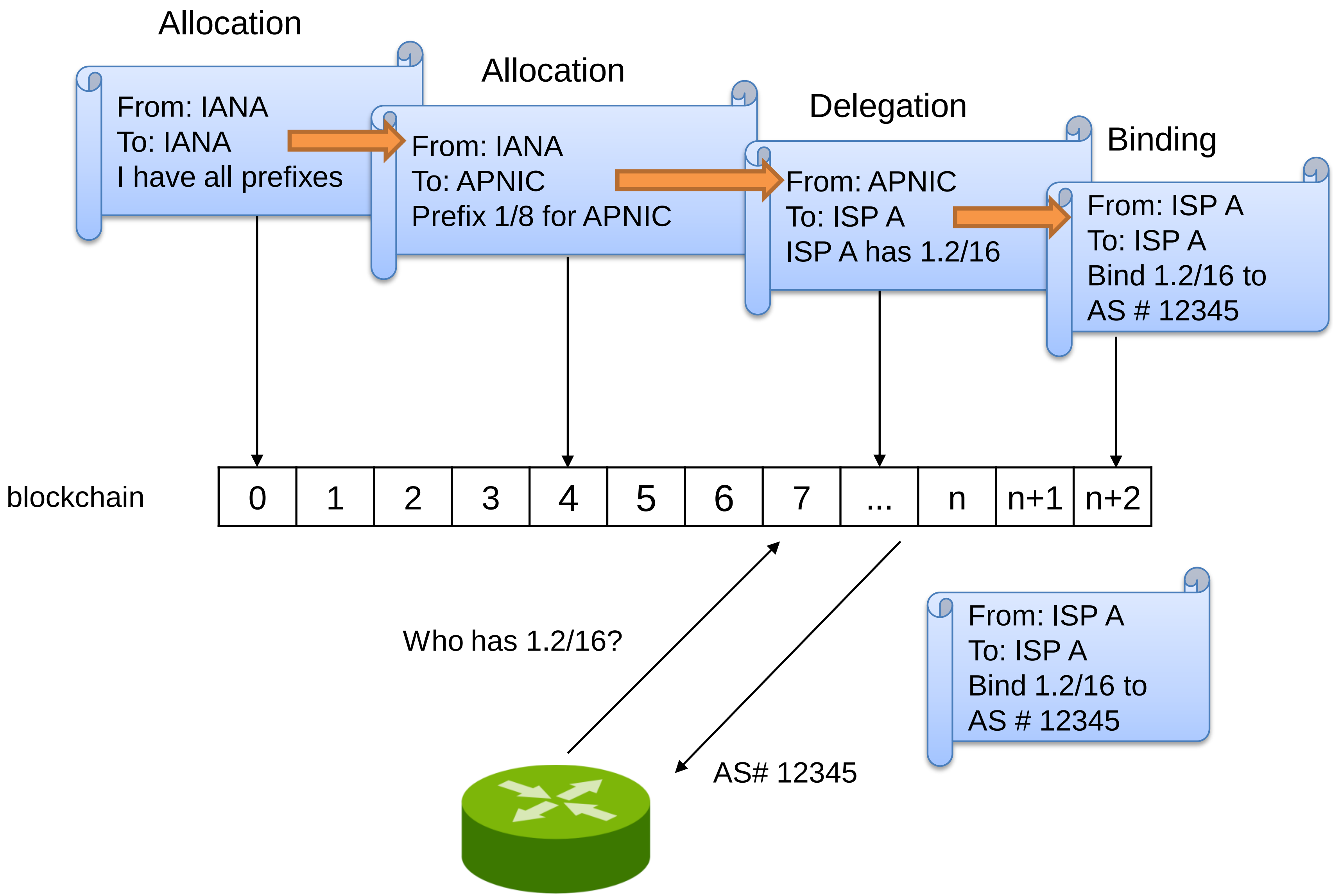

From: IANA
To: IANA
I have all prefixes
blockchain
0
1
2
3
4
5
6
7
...
n
n+1
n+2

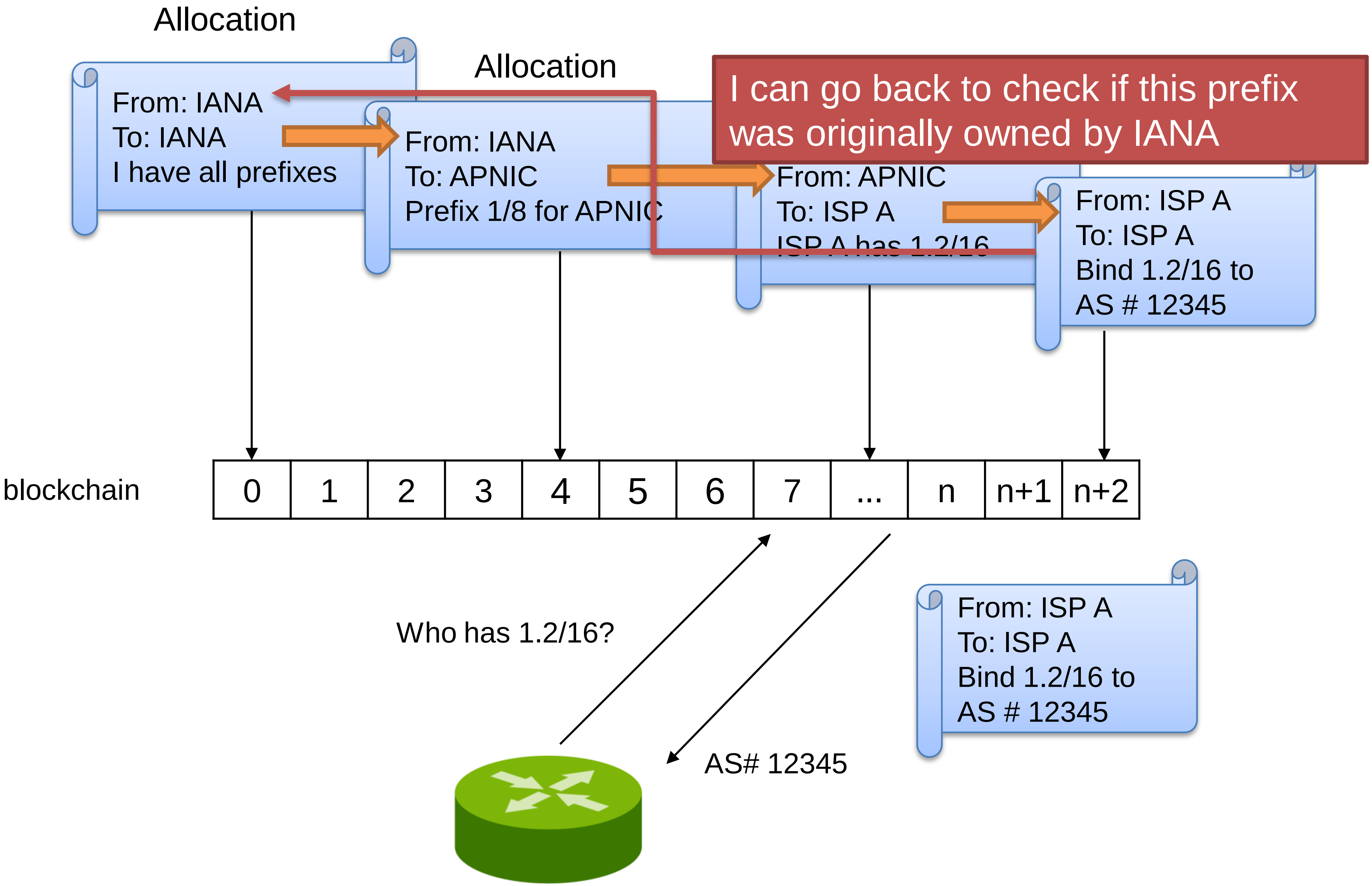

Allocation
From: IANA
To: IANA
I have all prefixes
Allocation
From: IANA
To: APNIC
Prefix 1/8 for APNIC
blockchain
0
1
2
3
4
5
6
7
...
n
n+1
n+2


Allocation
From: IANA
To: IANA
I have all prefixes
Allocation
From: IANA
To: APNIC
Prefix 1/8 for APNIC
Delegation
From: APNIC
To: ISP A
ISP A has 1.2/16
blockchain
0
1
2
3
4
5
6
7
...
n
n+1
n+2



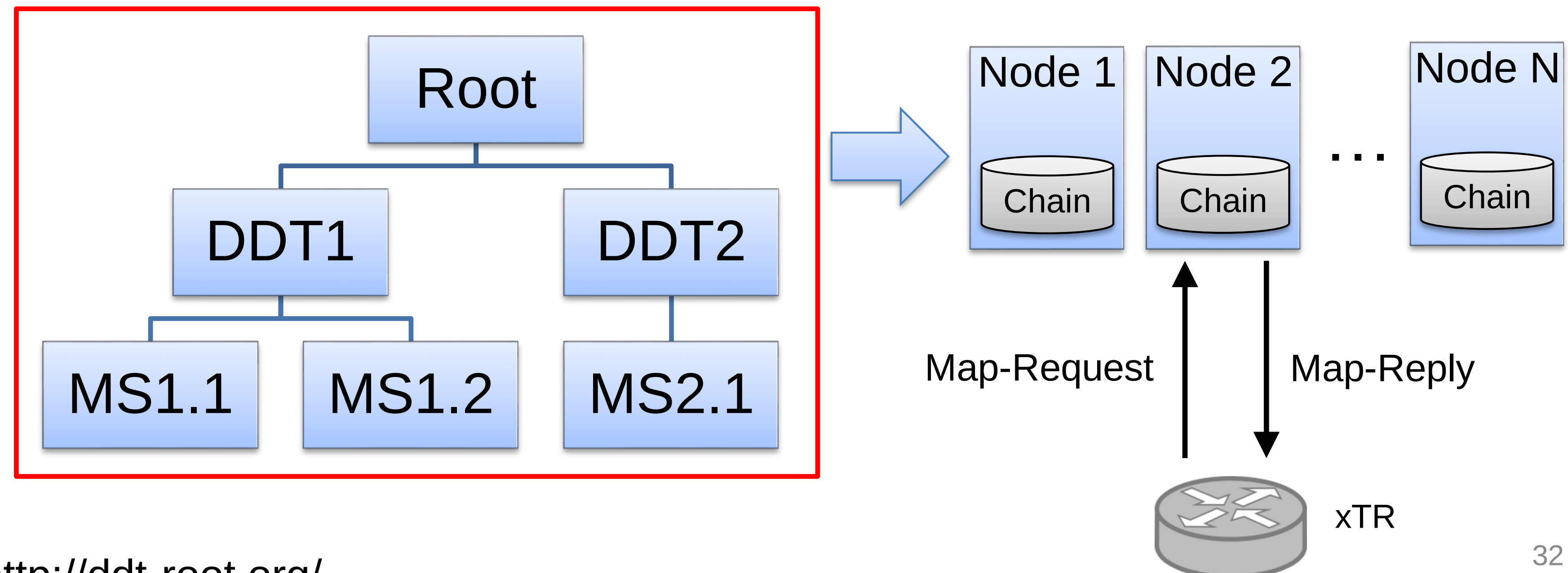
Allocation
From: IANA
To: IANA
I have all prefixes
Allocation
From: IANA
To: APNIC
Prefix 1/8 for APNIC
Delegation
From: APNIC
To: ISP A
ISP A has 1.2/16
Binding
From: ISP A
To: ISP A
Bind 1.2/16 to
AS # 12345
blockchain
0
1
2
3
4
5
6
7
...
n
n+1
n+2


Allocation
From: IANA
To: IANA
I have all prefixes
Allocation
From: IANA
To: APNIC
Prefix 1/8 for APNIC
Delegation
From: APNIC
To: ISP A
ISP A has 1.2/16
Binding
From: ISP A
To: ISP A
Bind 1.2/16 to
AS # 12345
blockchain
0
1
2
3
4
5
6
7
...
n
n+1
n+2
Who has 1.2/16?
AS# 12345
From: ISP A
To: ISP A
Bind 1.2/16 to
AS # 12345


Allocation
From: IANA
To: IANA
I have all prefixes
Allocation
From: IANA
To: APNIC
Prefix 1/8 for APNIC
I can go back to check if this prefix was originally owned by IANA
From: APNIC
To: ISP A
ISP A has 1.2/16
From: ISP A
To: ISP A
Bind 1.2/16 to
AS # 12345
blockchain
0
1
2
3
4
5
6
7
...
n
n+1
n+2
Who has 1.2/16?
AS# 12345
From: ISP A
To: ISP A
Bind 1.2/16 to
AS # 12345

Our use case

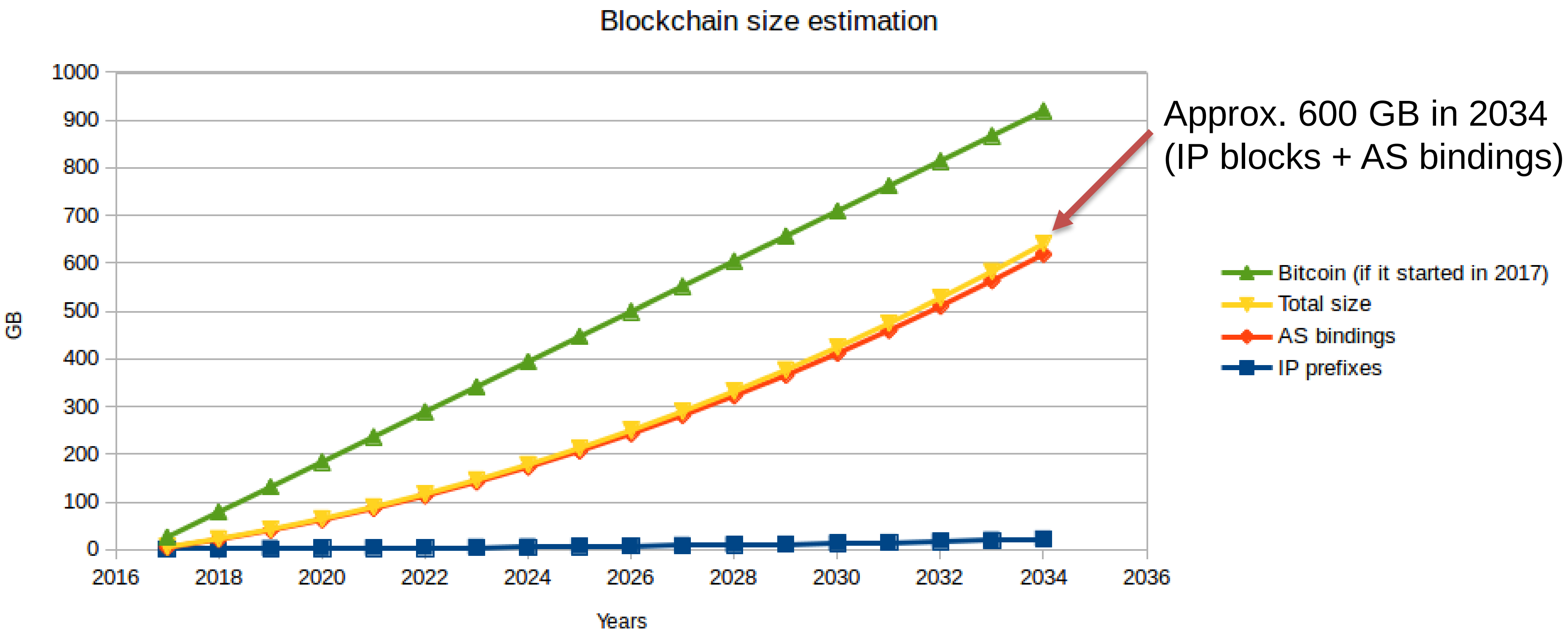
- LISP Beta Network
- Uses LISP-DDT*
- Full mapping system in the blockchain



*http://ddt-root.org/

Thanks for listening!

Scalability



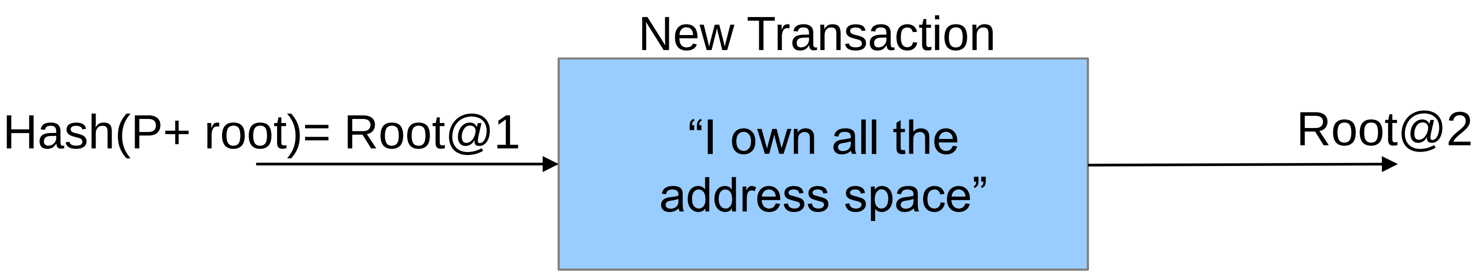
- One AS <> prefix binding for each block of /24 IPv4 address space
- Growth similar to BGP churn*
- Each transaction approx. 400 bytes
- Only IP Prefixes: worst case + BGP table growth*: approx. 40 GB in 20 years
- With PoS, storage can be reduced

*Source: http://www.potaroo.net/ispcol/2017-01/bgp2016.html

Transaction examples

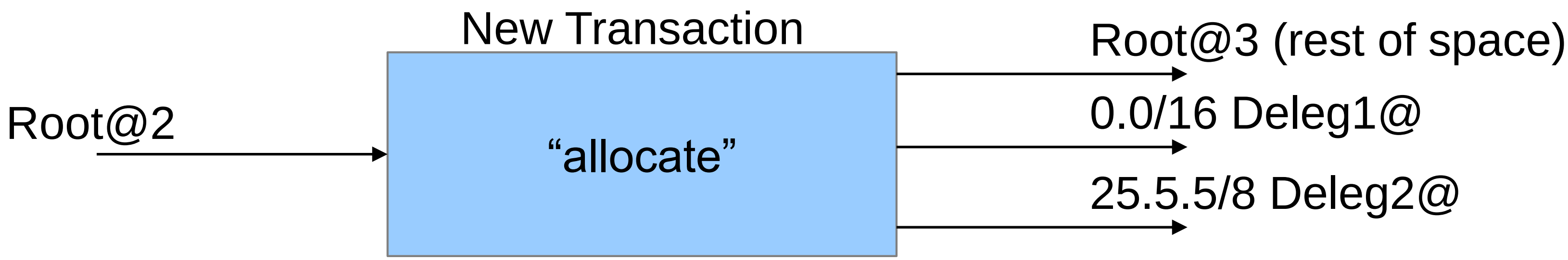
First transaction

- Users trust the Public Key of the Root, that initially claims all address space by writing the genesis block
- Root can delegate all address space to itself and use a different keypair

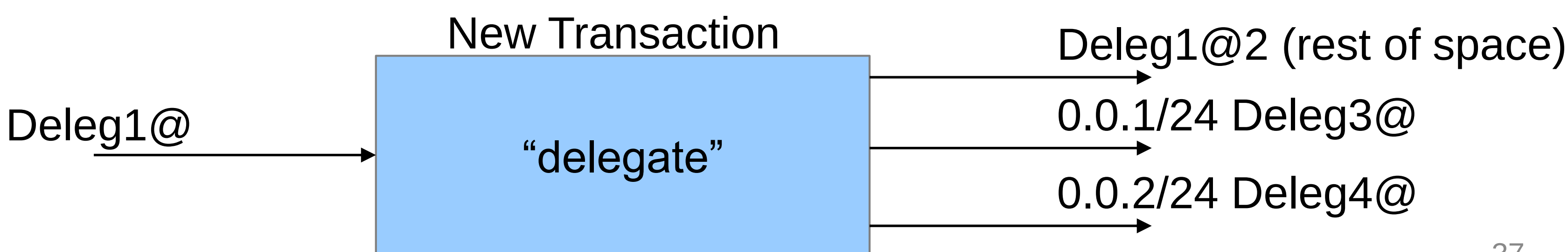


Prefix allocation and delegation

- Root allocates blocks of addresses to other entities (identified by Hash(Public Key)) by adding transactions

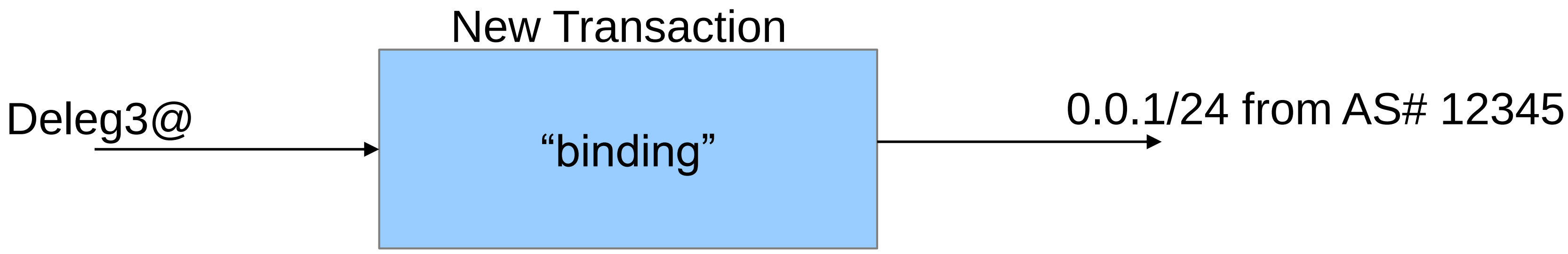


- Holders can further delegate address blocks to other entities



Writing AS bindings

- Just like delegating a prefix, but instead of the new holder, we write the binding



Rekeying

- Delegating the block of addresses to itself using a new key set.
- Simpler than traditional rekeying schemes
- Can be performed independently, i.e. each holder can do it without affecting other holder
- Same procedure for AS number bindings

External server authentication

- Some information may not be suitable for the blockchain, or changes so fast it is already outdated when added into a block
- A public key from an external server can also be included in the delegations
- Since blockchain provides authentication and integrity for this key, parties can use it to authenticate responses from the external server

FAQ

- Does it grow indefinitely?
  - Yes
- Do all nodes have the same information?
  - Yes
- When answering a query, do you have to search the entire blockchain?
  - No, you can create a separate data structure only with the current data
- If I lose my private key, do I lose my prefixes also?
  - Yes, watch out!