NPTEL ONLINE CERTIFICATION COURSES

Blockchain and its applications

Prof. Sandip Chakraborty

Department of Computer Science & Engineering
Indian Institute of Technology Kharagpur

Lecture 60: Blockchain for Decentralized Marketplace (Part 2)

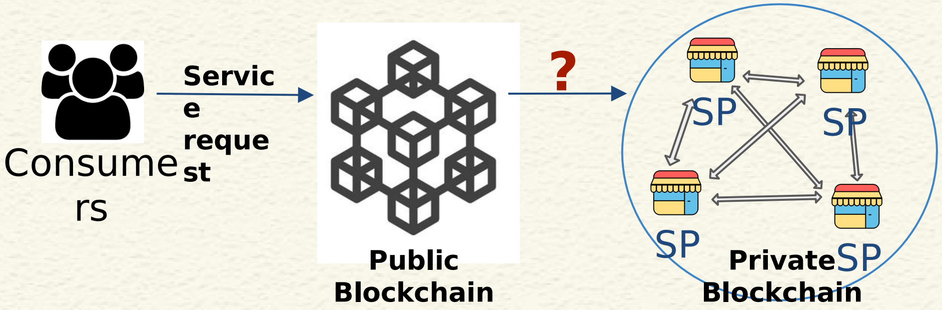
CONCEPTS COVERED

- **Blockchain application for a decentralized marketplace**

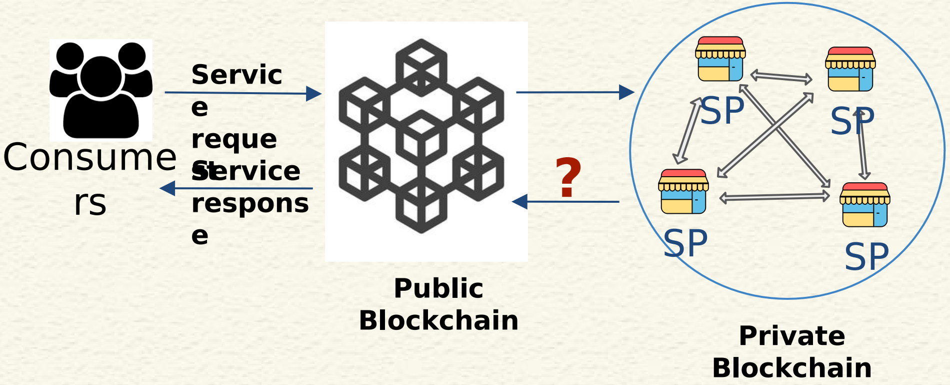
KEYWORDS

- Design a blockchain use-case
- Analyzing the requirements
- Consensus on Consensus

Transferring Consensus to the Consortium



Transferring Verifiable Response

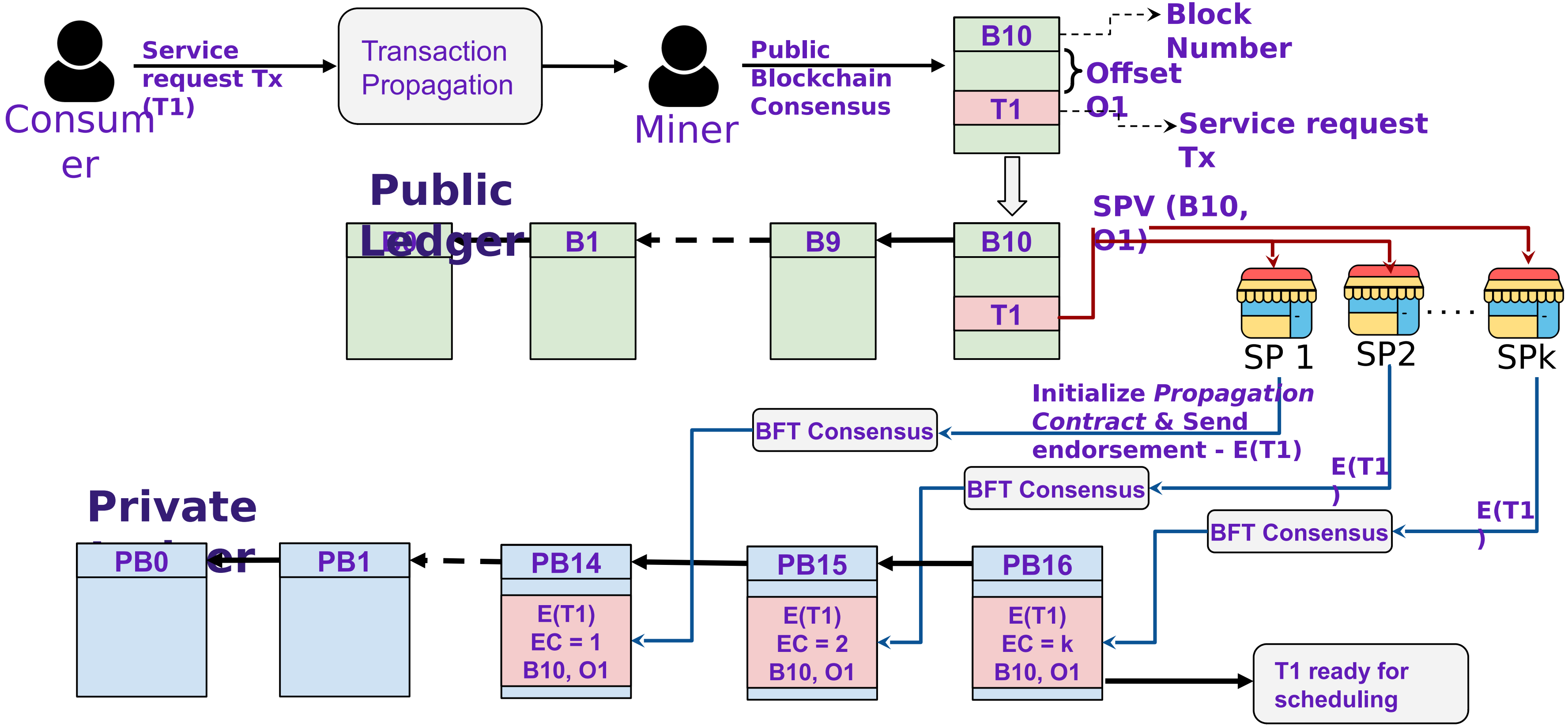


Consensus on Consensus

Consensus propagation from public blockchain to private consortium blockchain

- Each **SP also participates in the public blockchain** to receive service requests from consumers.
- When a transaction is committed in the public ledger, it is verified by the SPs through *Simplified Payment Verification* (***SPV***)[8]
- For each service request, the SPs **collect endorsements** through *Propagation Contract*
- Each endorsement goes through BFT consensus.
- When a service request receives k ≥ ⅔ of the SPs' endorsements, it is marked as confirmed.

Consensus on Consensus



Consumer

Service request Tx (T1)

Transaction Propagation

Miner

Public Blockchain Consensus

B10

T1

Block Number

Offset O1

Service request Tx

Public Ledger

B0

B1

B9

B10

T1

SPV (B10, O1)

SP 1

SP2

SPk

Initialize *Propagation Contract* & Send endorsement - E(T1)

E(T1)

E(T1)

BFT Consensus

BFT Consensus

BFT Consensus

Private Ledger

PB0

PB1

PB14

E(T1)
EC = 1
B10, O1

PB15

E(T1)
EC = 2
B10, O1

PB16

E(T1)
EC = k
B10, O1

T1 ready for scheduling

Verifiable Response Transfer

- Two kinds of information need to be transferred from the consortium to the consumers:

a. Consortium information such as catalog, pricing, etc.. - **not sensitive**
b. Request responses - results of scheduling and processing consumer requests such as a digital document, e.g., access credentials, tickets, invoices, etc. - **sensitive**

- Both kinds of data are generated collectively by SPs through private blockchain's consensus process.
- Consumers being outside the permissioned network cannot verify the correctness of the data.
- Separate protocol required for validation of consortium response by consumers.

Verifiable Response Transfer

- We use the concept of Collective Signing (CoSi) [21]
  - A set of consortium **SPs collectively sign** a valid data to make it verifiable.
  - We utilize **Boneh-Lynn-Shacham (BLS)** cryptosystem for **aggregating** signatures from individual SPs.
  - A BLS signature for message $\mathcal{M}$ is computed as: $\mathbb{S}_i(\mathcal{M}) = \mathcal{H}(\mathcal{M})^{\mathcal{S}_{\mathcal{C}_i}}$

    $\mathcal{H}(.)$ is a cryptographic hash function.

    $\mathcal{S}_{\mathcal{C}_i}$ Is secret key of SP $\mathcal{C}_i$

Aggregated multi signature for n SPs:

$$\mathbb{S}_{1..n}(\mathcal{M}) = \mathcal{H}(\mathcal{M})^{\mathcal{S}_{\mathcal{C}_1}+\mathcal{S}_{\mathcal{C}_2}+..+\mathcal{S}_{\mathcal{C}_n}} = \prod_{i=1}^{n} \mathcal{H}(\mathcal{M})^{\mathcal{S}_{\mathcal{C}_i}}$$

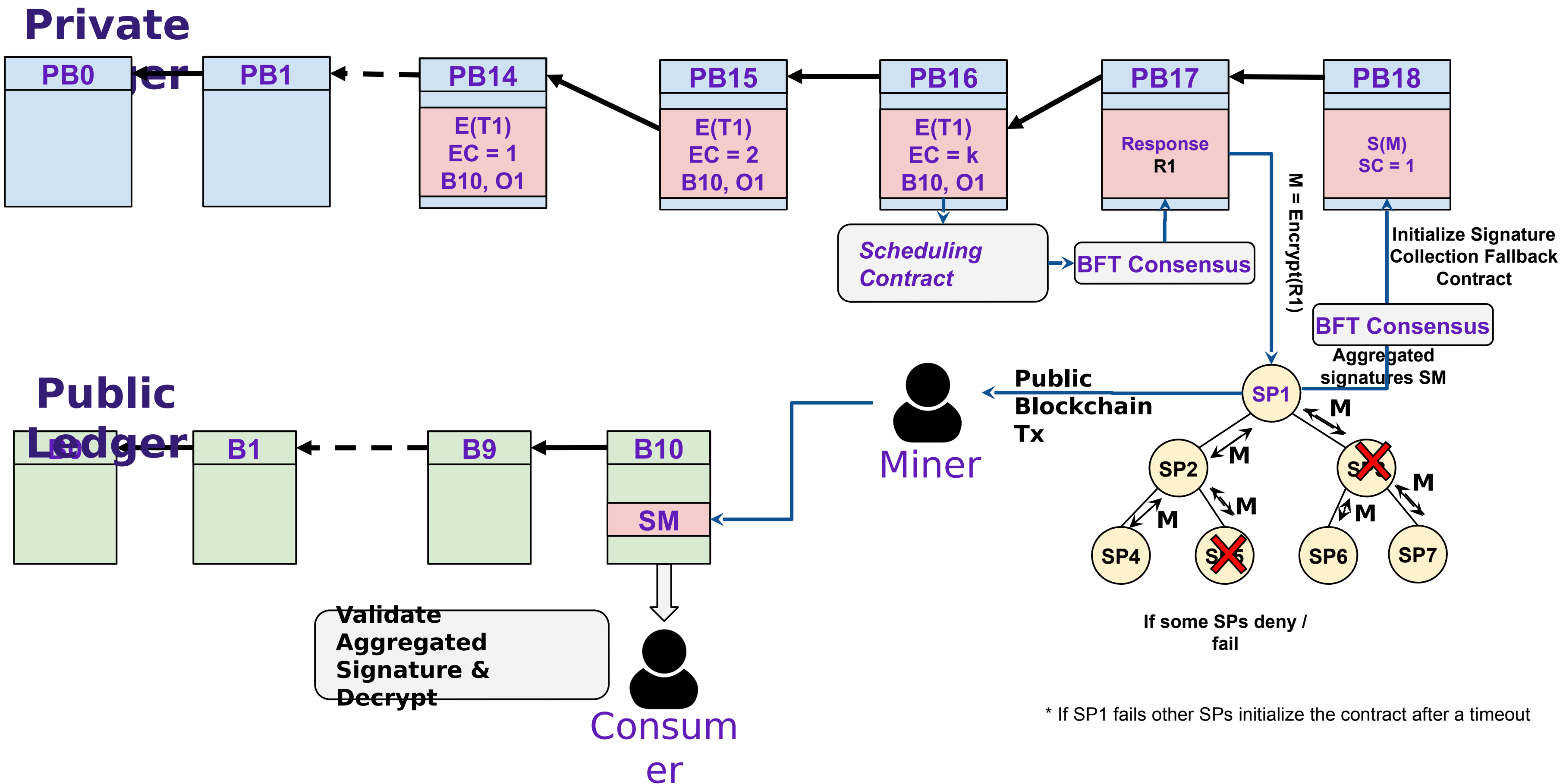
$$= \mathbb{S}_1(\mathcal{M}) \times \mathbb{S}_2(\mathcal{M}) \times .. \times \mathbb{S}_n(\mathcal{M}) = \prod_{i=1}^{n} \mathbb{S}_i(\mathcal{M})$$

[21] Syta, Ewa, Iulia Tamas, Dylan Visher, David Isaac Wolinsky, Philipp Jovanovic, Linus Gasser, Nicolas Gailly, Ismail Khoffi, and Bryan Ford. "Keeping authorities" honest or bust" with decentralized witness cosigning." In 2016 IEEE Symposium on Security and Privacy

Verifiable Response Transfer

- Consortium response is accepted as **valid only if it has ≥ ⅔ of the SPs' signatures**.
- For preserving confidentiality, a response to a consumer is encrypted using its public key.
- **Signature Collection:**
  - Multisignature collection is carried out **off-chain** to improve latency.
  - A **communication tree** is formed along which the singing request and the signatures are exchanged.
  - Each node of the tree aggregates signatures collected from its descendants.
  - **Fallback** to smart contract based signature collection in case of denial of service attack by some SP.

Verifiable Response Transfer

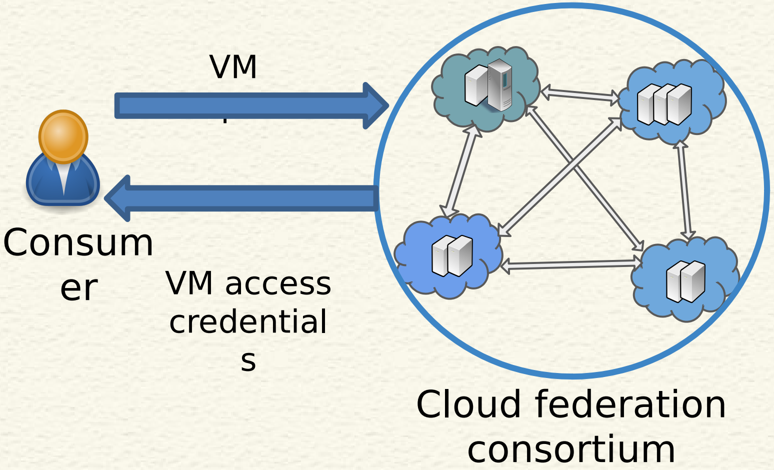


* If SP1 fails other SPs initialize the contract after a timeout

Use Case Implementation: Cloud Federation

- Consortium of **cloud service providers** (CSPs).
- Provide cloud infrastructure resources to end-users (IaaS).
- Implemented a **fair scheduling algorithm** for allocation of consumer requests among SPs:
  - Each SP will be allocated the number of consumer requests proportional to its infrastructure contribution in the federation.
- Test bed implementation using **Ethereum** and Hyperledger **Fabric**, and Hyperledger **Burrow**.
- Mininet emulation for evaluating scalability.

Use Case Implementation: Cloud Federation



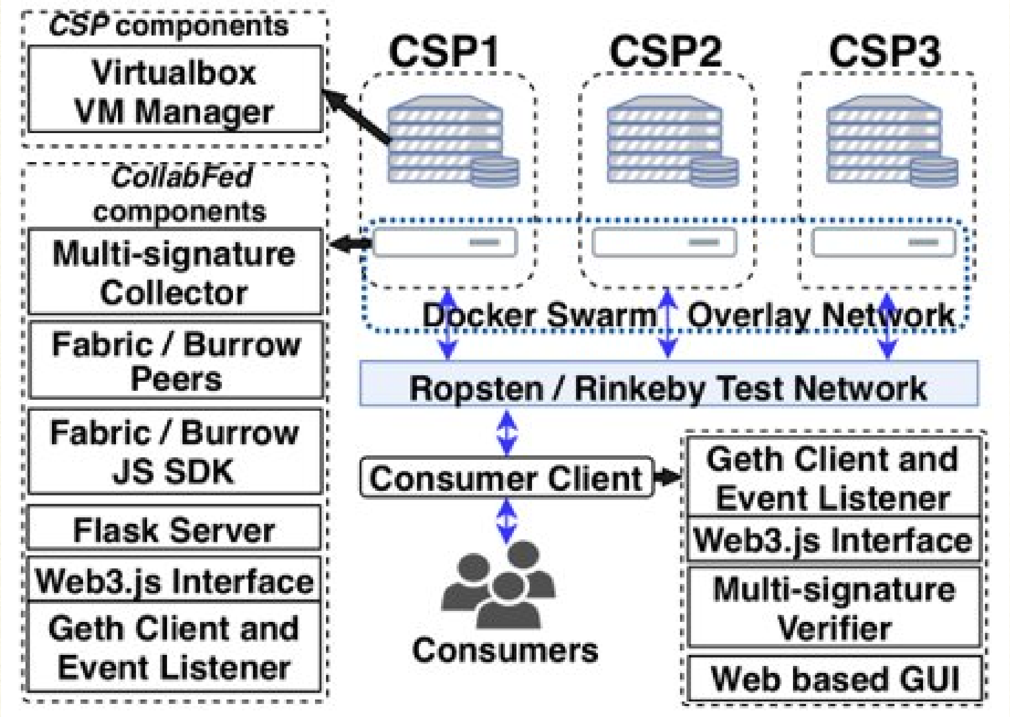
VM

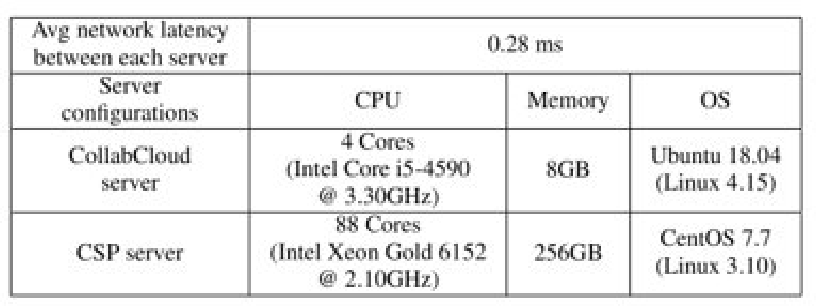
Consum
er

VM access
credential
s

Cloud federation
consortium

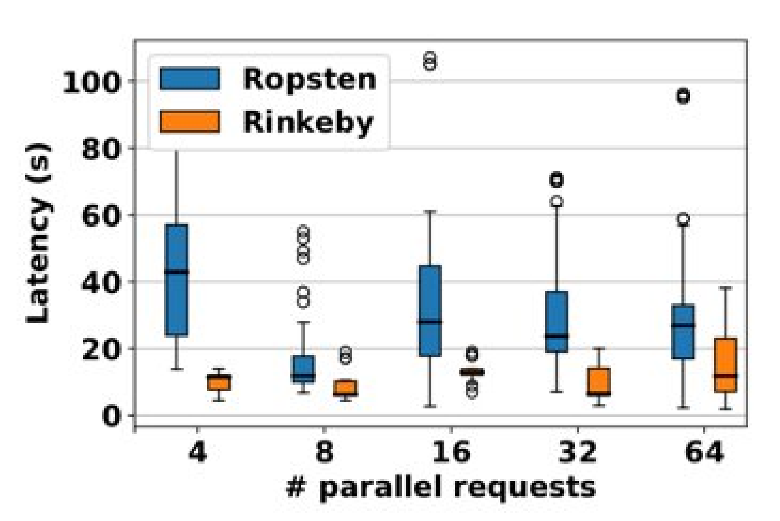
Testbed Setup



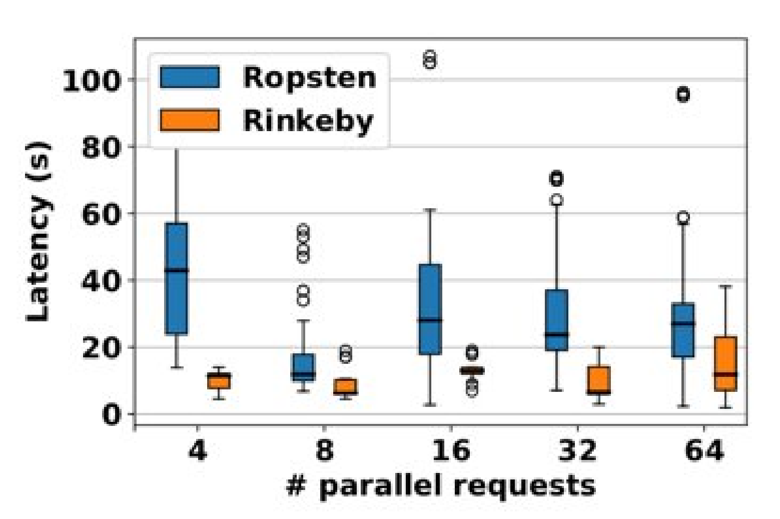


Avg network latency between each server	0.28 ms		
Server configurations	CPU	Memory	OS
CollabCloud server	4 Cores (Intel Core i5-4590 @ 3.30GHz)	8GB	Ubuntu 18.04 (Linux 4.15)
CSP server	88 Cores (Intel Xeon Gold 6152 @ 2.10GHz)	256GB	CentOS 7.7 (Linux 3.10)

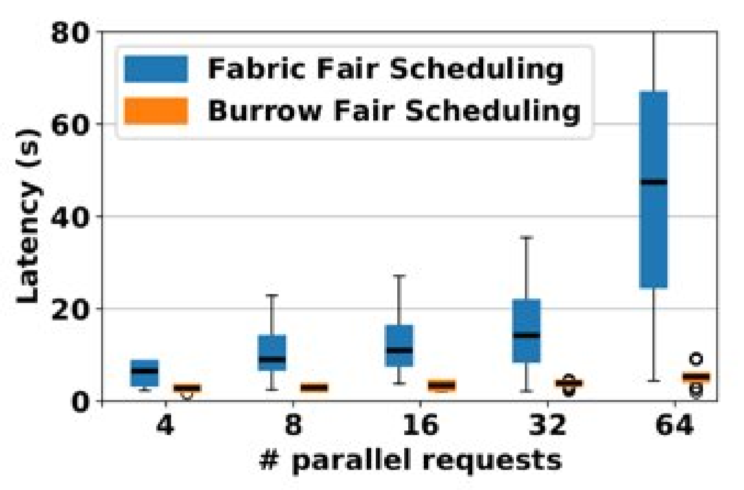
Results


Ropsten
Rinkeby
Latency (s)
0
20
40
60
80
100
4
8
16
32
64
# parallel requests

Results

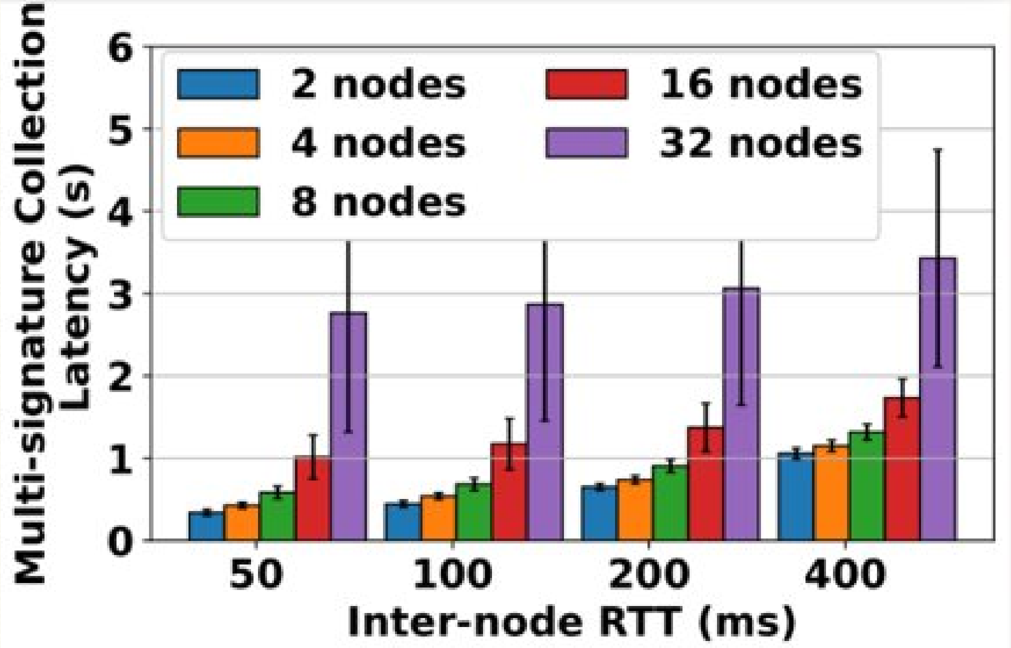


Ropsten
Rinkeby
Latency (s)
0
20
40
60
80
100
4
8
16
32
64
# parallel requests



Fabric Fair Scheduling
Burrow Fair Scheduling
Latency (s)
0
20
40
60
80
4
8
16
32
64
# parallel requests

Results



Multi-signature Collection Latency (s)
0
1
2
3
4
5
6
2 nodes
4 nodes
8 nodes
16 nodes
32 nodes
50
100
200
400
Inter-node RTT (ms)

Conclusion

- There are interesting research/design problems in the blockchain space
  - You need to think of applying the right technology at the right place!

- Remember the fundamental questions that we talked about earlier
  - Network, participants, assets, transactions
  - Keys – how to obtain and share
  - Trusted third party – do we have any?
  - Why people will join your blockchain network

Thank you