Proof-of-Stake Consensus Protocol for Cyber Supply Chain Data Provenance

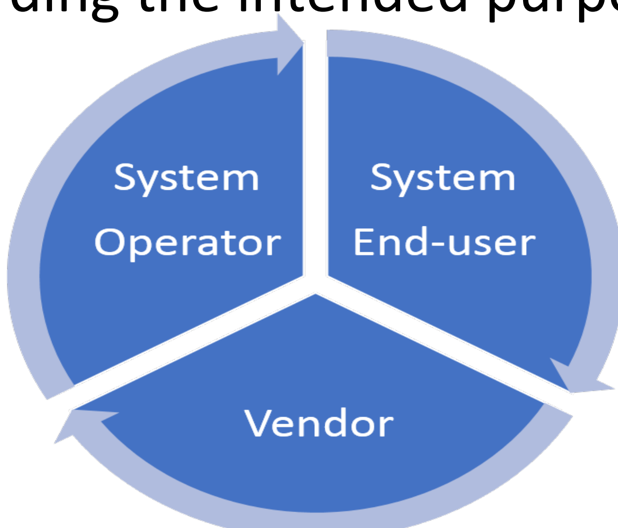
Xueping Liang, Deepak Tosh, **Sachin Shetty**
Old Dominion University

CYBER RESILIENT ENERGY DELIVERY CONSORTIUM

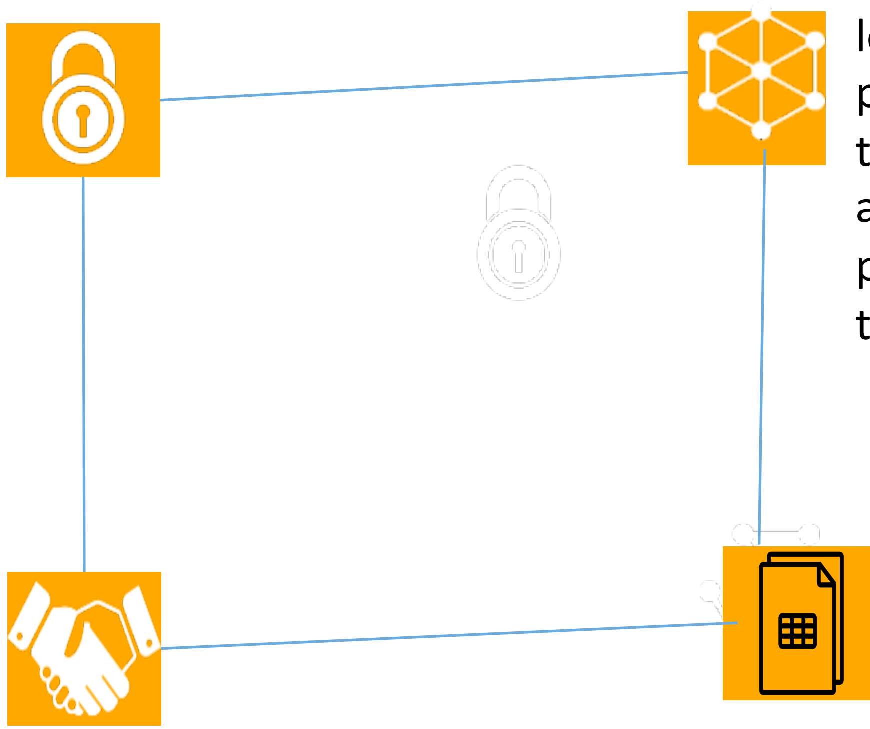
Funded by the U.S. Department of Energy and the U.S. Department of Homeland Security | cred-c.org

Motivation

- Address cyber supply chain risks due to lack of trust in software and firmware developed by third party vendors
- Current solutions, such as, side channel fingerprinting, reverse engineering, deployed at chip level are not scalable to protect entire cyber supply chain and cannot provide near real-time tracking
- **Goal** – Permissioned blockchain-based data provenance framework to ensure processes in the supply chain are functioning according the intended purpose.



Blockchain Overview



Cryptographically Secure
Public/Private signature technology applied to create transactions that establishes a shared truth.

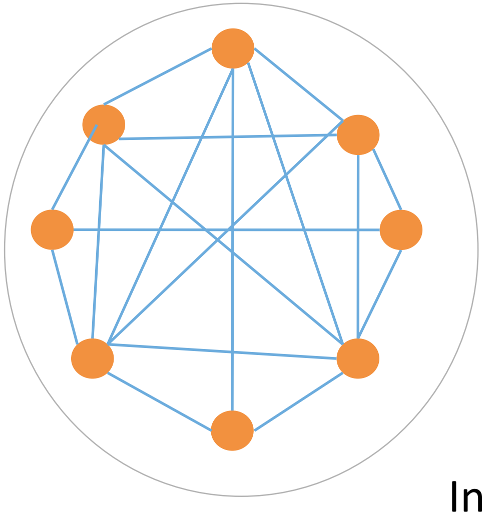
Distributed Network
Replicas of distributed ledger and no single participant owns or can tamper. Consensus among majority participants is needed to update the database

Consensus
Consensus among majority participants is needed to update the database. Leverages validation rules provided by smart contract (“Business Logic”)

IMMUTABLE LEDGER
Append only database that holds immutable record of every transaction

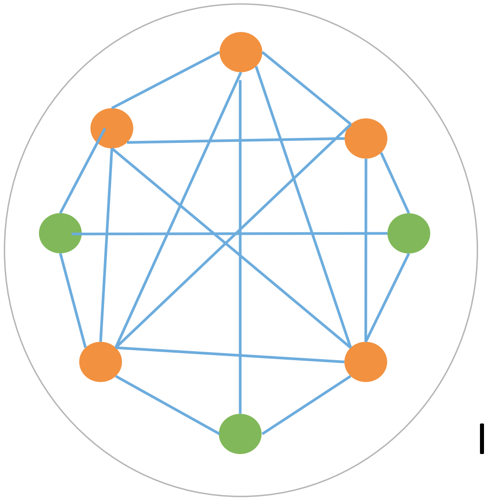
Blockchain Overview

- Permissionless Blockchain Infrastructures
  - Open access on the Internet
  - Anonymous validators
  - Proof of Work consensus
  - Public network



Internet

- Permissioned Blockchain Infrastructures
  - Private network
  - Participation by members only
  - Trusted validators
  - Customized consensus protocol



Intranet

Consensus Protocols

- Proof of Work
  - Carry out large computation and prove that computation was successfully
  - No additional work to check the proof
  - Limits the rate of new blocks and expensive to add invalid blocks
  - Aids in deciding between competing chains
- Proof of Stake
  - Achieve consensus by eliminating expense proof of work
  - Block creation tied to amount of stake
- Byzantine Fault Tolerance
  - Trusted entities work together to add records
  - Voting process for accepting a block on the chain

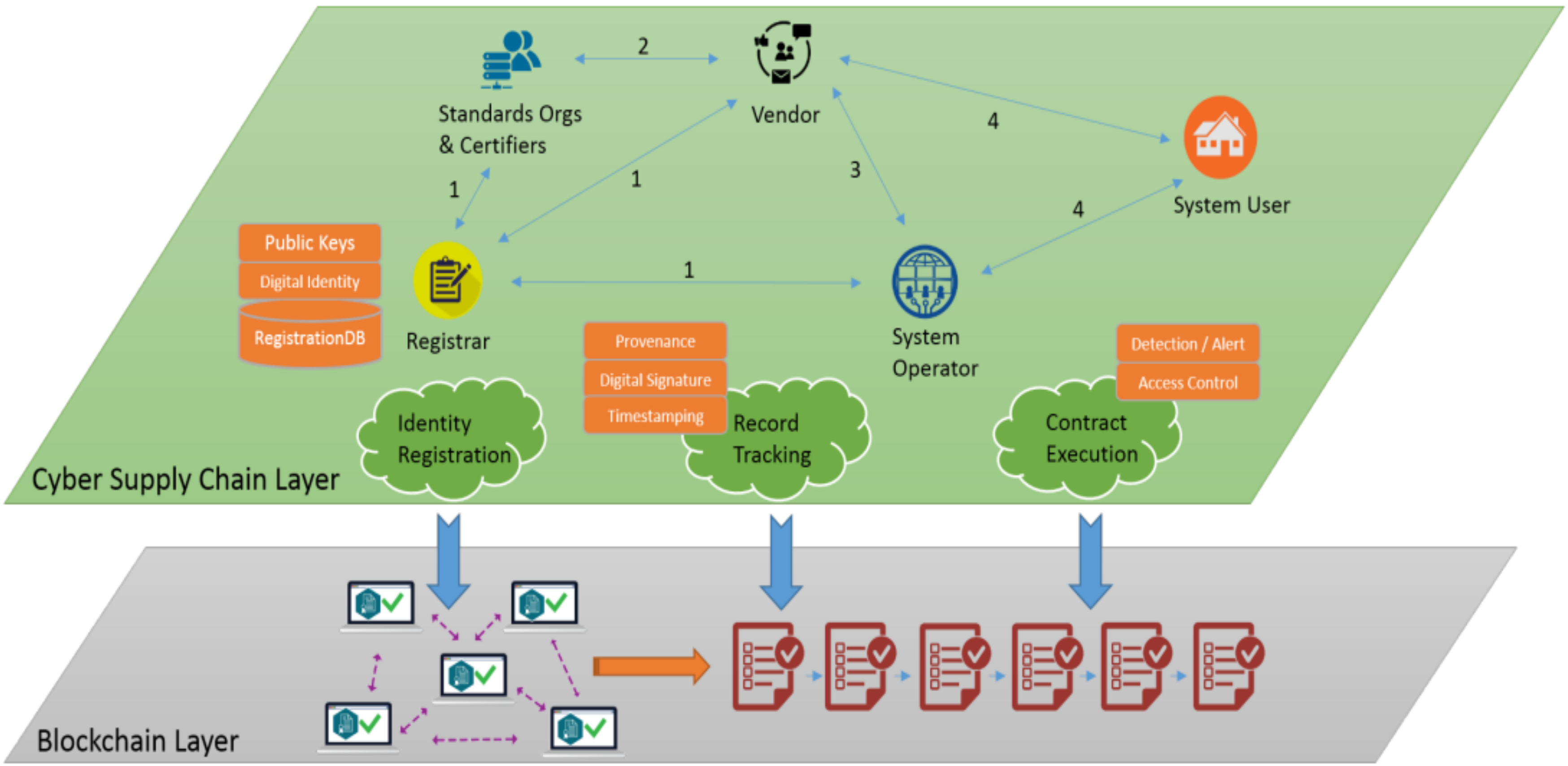
Consensus Protocols

- GHOST
  - Weigh subtrees to resolve conflicts
- Bitcoin-NG
  - Leader election to append microblocks for increasing throughput and decreasing latency
- Parallelization
  - BlockDAG
- Eliminate communication and resource overhead
  - Stellar, XFT, CheapBF(trusted hardware)
- Randomized BFT
  - Probability vs deterministically
  - BFT design framework (http://www.vukolic.com/700-Eurosys.pdf)
- Mix of PoW and BFT (SCP)
  - PoW for identity management
  - BFT for agreement

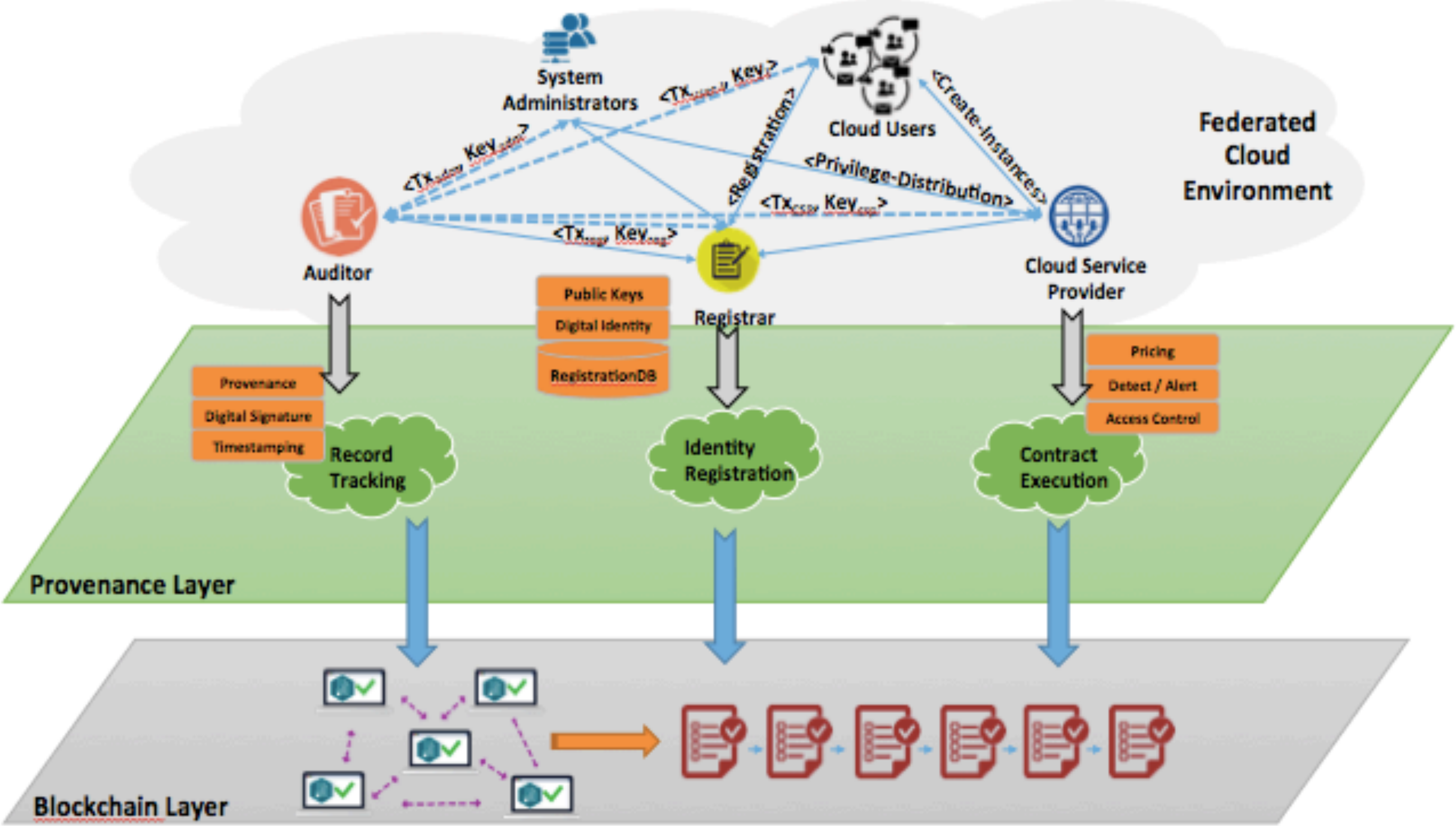
Approach

- Blockchain empowered cyber supply chain framework
  - Cyber Supply Chain System Entities
    - System Operator, end-user and vendor
  - Cyber Supply Chain System Processes
    - Procurement and Operational Phases
  - Cyber Supply Chain Attacks
    - Manufacturer Source Code, vendor remote access
- Proof-of-stake consensus protocol to balance tradeoff between scalability and resilience

Blockchain empowered cyber supply chain framework



Blockchain empowered cyber supply chain framework in a distributed system



Blockchain empowered cyber supply chain framework

- Procurement Phase
  - Identify and document cyber security risks during designing and developing processes.
  - Prevent attacks resulting from procuring and utilizing vendor devices or software, as well as vendor transitions.
- Operational Phase
  - Record regular practices to maintain the system functionality and performance, including security check, periodic assessment, logging and monitoring.
  - Conduct software updates from vendors either for performance improvement or security-related enhancement

Blockchain empowered cyber supply chain framework

- Procedures
  - Identity Establishment
  - Product Authenticity and Verification
  - Access Control Management
  - Contract Negotiation and Execution
  - Logging, Monitoring and Auditing

- Challenges
  - Identity protection
  - Integrity protection
  - Fine-grained access control management
  - Automated contract execution
  - Tamper-resistant record keeping

Requirements for consensus protocols

- Efficiency
  - Time to achieve agreement
  - Transaction processing time
- Security
  - Deterministic agreement
  - Resilient to partial node failure
- Scalability
  - Number of validating nodes
  - Transaction Processing

Distributed Consensus Protocol

- Traditional PoW suffers from large consensus delay and high computational requirement
- State-of-the art Proof of Stake consensus works well for cryptocurrencies
- Mechanism for allocating resources should balance tradeoff between resilience and scalability
- No formal work on defining stake in distributed systems

Distributed Consensus Protocol

- Audit data-related operations in cyber supply chain in near real-time
- PoS based Energy-efficient consensus protocol
  - Validators who commit transactions offer securities in the form of stakes
  - Opportunistic use of under-utilized resources for realizing the consensus in energy-efficient way
  - Reward of dedicating resources to maintain consensus
  - Malicious actions in consensus are prevented through penalizing stake

Threat Model

- Validators' agility (may enter and exit the consensus process anytime)
- Validators may behave erratically or even disappear in between an ongoing epoch
- Permitting any user to be validator can widen attack surface through nothing-at-stake problem
- Reputation of validators matters otherwise greediness may drive the consensus toward maliciousness

Defining Stakes

- In cryptocurrency, stakes are nothing but tokenized form for the currencies

- In cloud computing perspective, stakes can be
  - CPU power or the number of CPU slices/cores provided by the CSP (C_i)
  - Amount of memory allocated for program execution and temporary buffer (S_i)
  - Network data rate (D_i)
  - Secondary storage etc.
- Stake of a validator i can be a tuple $X_i = <X_{C_i}, X_{S_i}, X_{D_i}>$ that is selected out of total allocated resources $R_i = <C_i^{\max}, S_i^{\max}, D_i^{max}>$
  - Given current reso >, the greediness parameter (γ) drive

$$\mathcal{X}_{C_i} = \gamma^i_{cpu}(C_i^{max} - \tilde{C}_i)$$
$$\mathcal{X}_{S_i} = \gamma^i_{mem}(S_i^{max} - \tilde{S}_i)$$
$$\mathcal{X}_{D_i} = \gamma^i_{nw}(D_i^{max} - \tilde{D}_i)$$

Incentives for participation

- Consensus cannot survive with no participation
  - Motivation requires incentivization
- Rewarding consensus validators should be through
  - Transaction fees
  - Transferring resources to the leader's account
  - Discounting leasing costs

- Who offers the reward?
  - Choice to make: Service provider or clients?

- If R_{total} turns out to be the benefit of service for a total of z epochs, then reward R_{total}/z /epoch should be dedicated
- Leader-followers' reward distribution needs to be agreed !!!

PoS based Energy-efficient consensus protocol

a. Stake Determination
   - Stake for validator $i = X_i = f(R, R^u, \gamma) = \gamma(R - R^u)$, γ is greediness parameter

b. Resource staking and confirmation
   - VMCREATE$(\langle X_{C_i}, X_{S_i}, X_{D_i} \rangle$, Shared_Sec$) \rightarrow (\Delta_i, \mathrm{txID}_i)$, $\forall i \in N$
   - VMVERIFY$(\Delta_i) \rightarrow \{0, 1\}$

c. Stochastic leader election based on proportion of staked resources
   - Probability of *i* being a leader is defined as: $p_i = \|X_i\| / \sum_{k=1}^{N} \|X_k\|$

d. Block replication and verification
   - Leader's block gets broadcasted and verified before commit otherwise re-election occurs

e. Reward distribution for participation in consensus
   - Extra resource as incentive, or reduced resource leasing cost as incentive

Algorithm

Algorithm 1: PoS Procedure run by a *validator* i at *epoch* t

Input: *Epoch* (t), List of TXs ($\mathcal{L}\{\mathsf{tx}(Key \rightarrow Val)\}$), and blockchain ($\mathcal{B}_{t-1}$) until *epoch* $t-1$

Result: Updated blockchain state $\mathcal{B}_t$

```
1  Initialize a temporary block b_i, where,
   b_i ← H(B_{t−1})||timestamp||M_root||t||L{tx};
2  Define amount of stake (X_i(t)) for epoch t, as
   < X_{C_i}(t), X_{S_i}(t), X_{D_i}(t) >;
3  SS ← create_SharedSecret({pu_i : i ∈ N});
4  Allocate virtual instance that consumes resources
   equivalent to stake (X_i(t)) by invoking
   (Δ_i, txID_i) ← VMCREATE(< X_{C_i}, X_{S_i}, X_{D_i} >, SS);
5  Distribute stake confirmation (txID_i) and resource
   identifier (Δ_i) to other peers;
6  [status_j] ←VMVERIFY(Δ_j) ∀j ∈ N\{i};
7  if Σ_{j=1}^{N} status_j=N then
8  |   leader(t) ← selectLeader({X_i : i ∈ N});
9  |   if leader(t) = i then
10 |   |   Update the blockchain B_t ← B_{t−1}||b_i;
11 |   |   Broadcast the block b_i to other peers in the
   |   |   network;
12 |   else
13 |   |   Listen to brodcast of block b_{leader(t)} from the
   |   |   selected leader;
14 |   |   Update the blockchain B_t ← B_{t−1}||b_{leader(t)};
15 |   end
16 else
17 |   Possible malicious validator and restart the
   |   consensus for epoch← t + 1;
18 |   goto Step 1.
19 end
```

Stake Determination

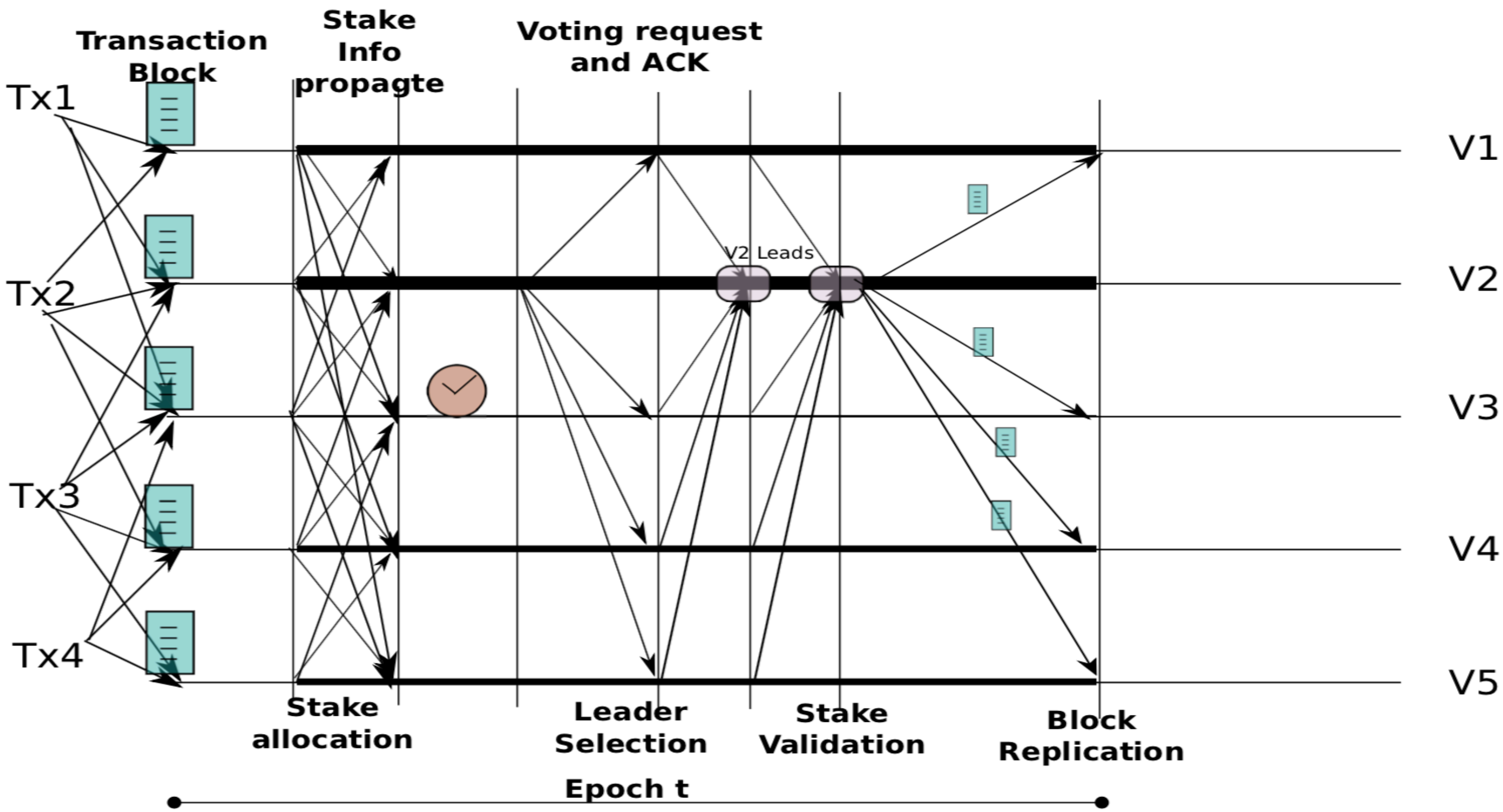
Stake Allocation

Stake Verification

Leader Selection

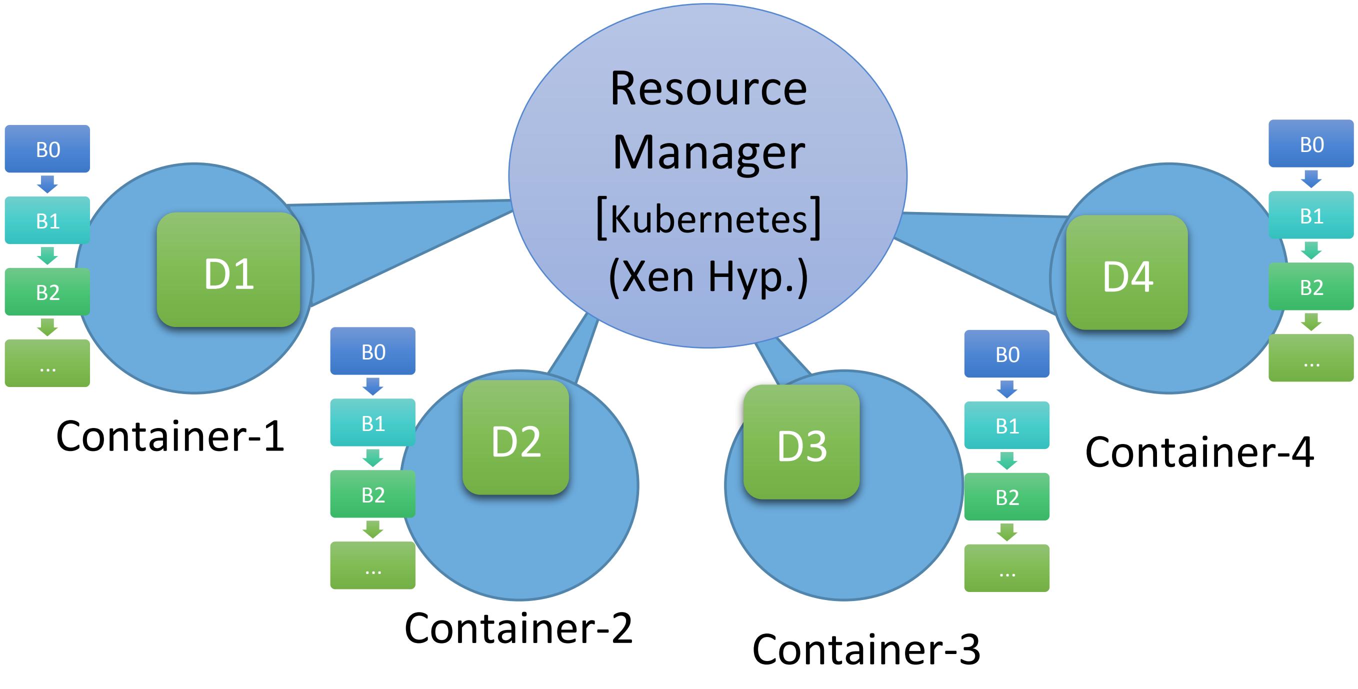
Block Propagation

PoS Consensus Timeline



Experimental Testbed

- ❑ Testbed environment is based on a local cluster of physical machines managed by a Xen Hypervisor
- ❑ Elasticity resource management is done through Kubernetes and Docker is used for containerized services in the VMs



Performance Evaluation

- Each validator's stake value is designed as a value between 0 and 100

- Validators stake remains unchanged for a fixed duration

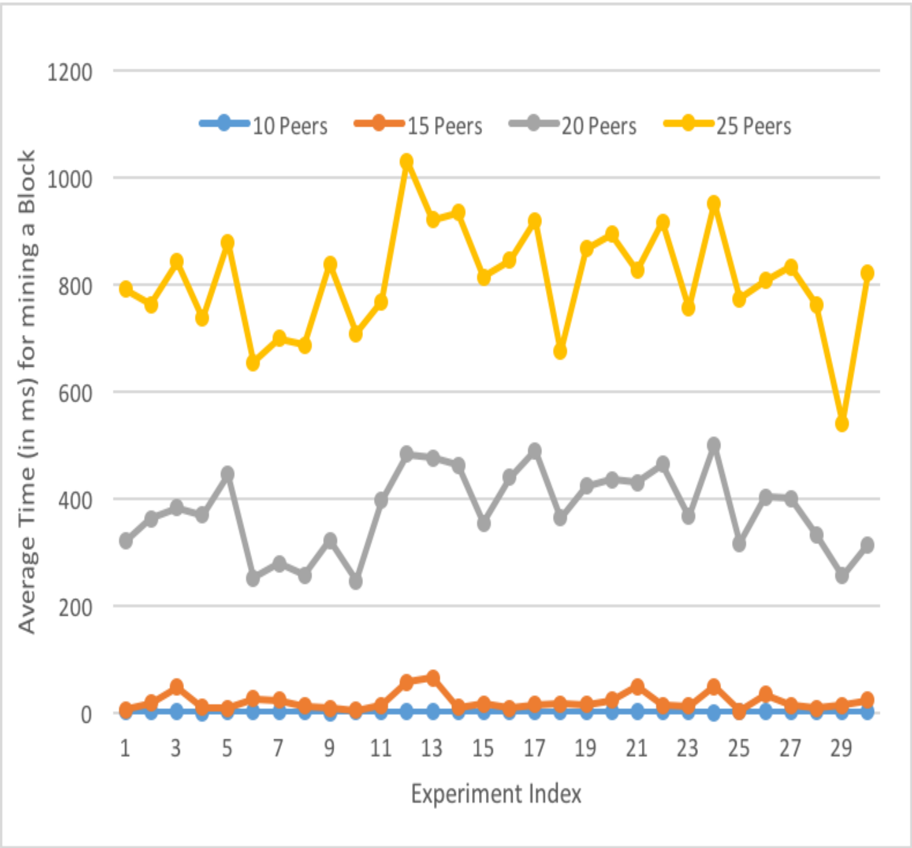
- Network latency is considered to be normally distributed between 1 and 5ms

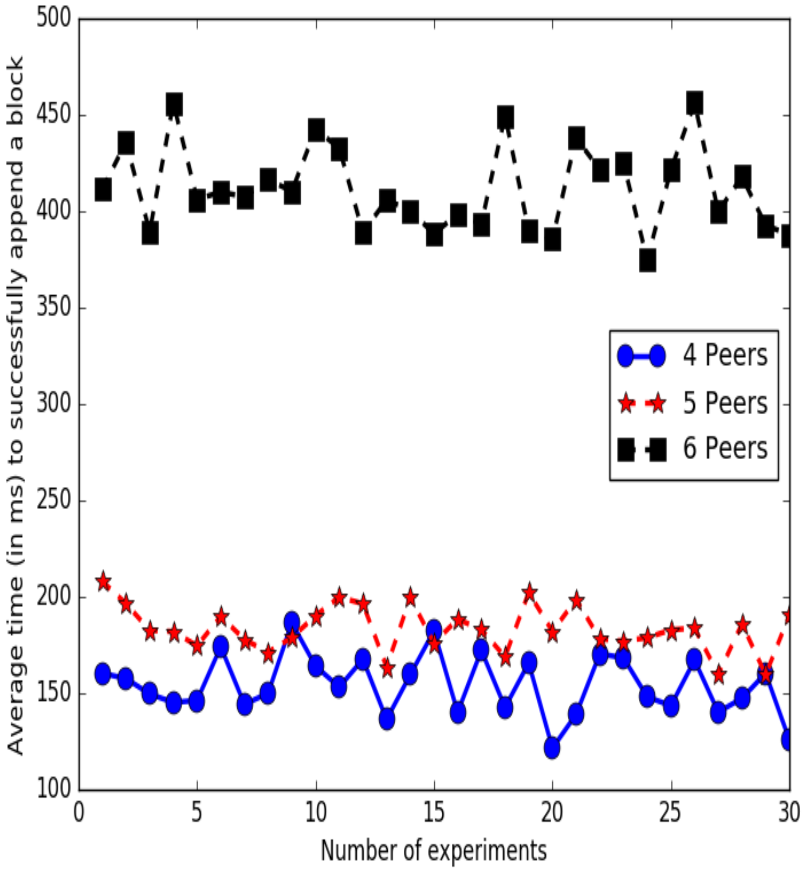
- Time for block mining consists of time taken to verify transactions and stakes of the leader

Evaluation Metrics

- Average and total times each validator was the leader
- Total number of times a leader was selected as validator but did not have the highest stake amount
- Average, max/min time in milliseconds to make progress and extend the Blockchain with a new block

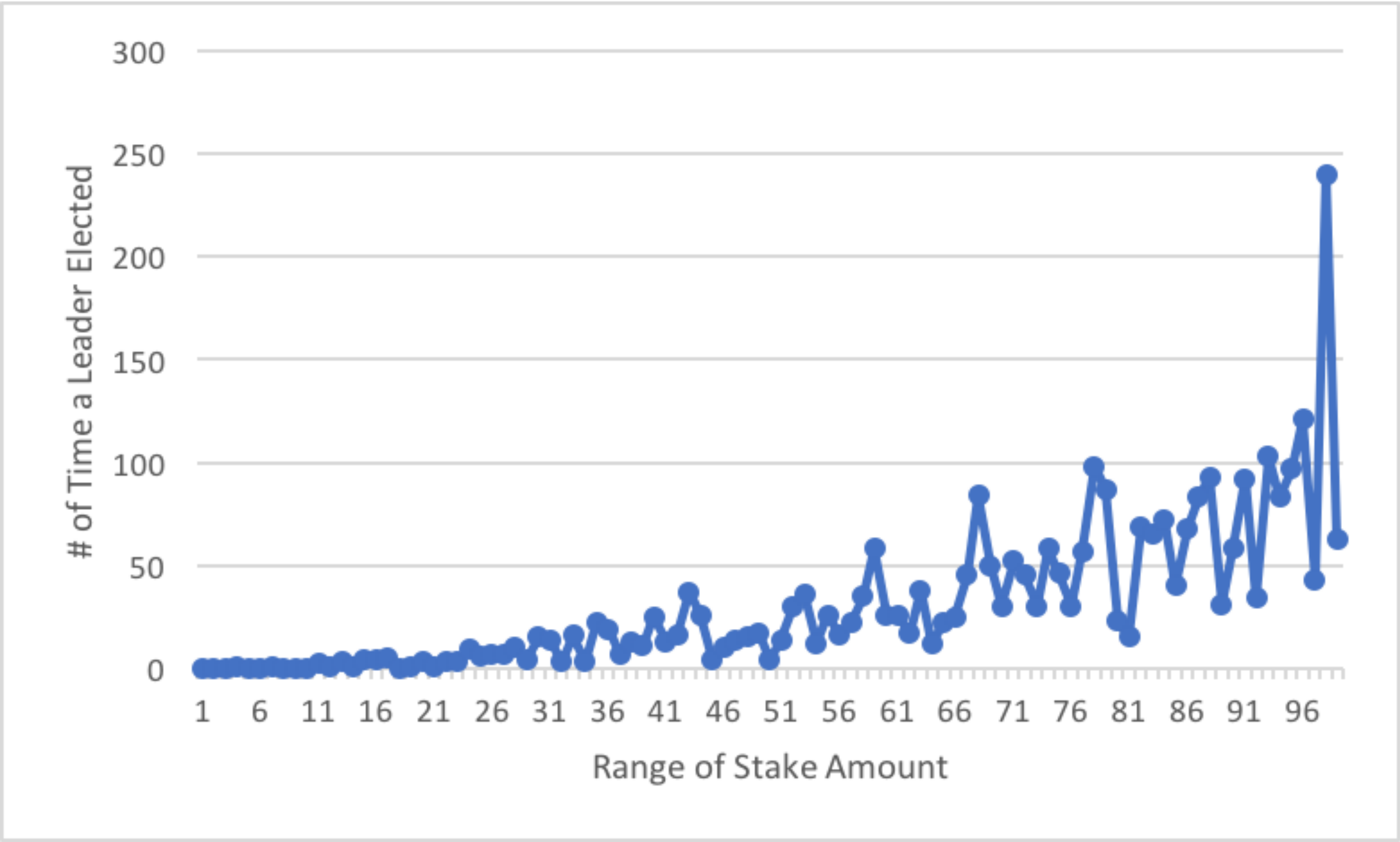
Average time to extend Blockchain with a new block





(In Presence of Network Delay)

Average # of times a leader elected based on stake amount



Higher the stake, chances of becoming leader is high

Ongoing and Future Work

- Formal Analysis of the Proof-of-Stake protocol to evaluate scalability and resilience to attacks
- Development of Blockchain-based Cyber Supply Chain Prototype in Hyperledger Fabric
- Development of simulator to aid in engineering Blockchain solutions for cyber supply chain
  - Quantitative insights into choice of platforms (public/private/public-private), consensus protocols (Proof-of-Work, Proof-of-Stake, Proof of Elapsed Time, Practical Byzantine Fault Tolerance), factors impacting scalability (validating nodes, bootstrap time) and resilience (network/node failures)

Related Publications

- Xueping Liang, Sachin Shetty, Deepak Tosh, Yafei Ji, Danyi Li, "Towards a Reliable and Accountable Cyber Supply Chain in Energy Delivery System using Blockchain", 14th EAI International Conference on Security and Privacy in Communication Networks (SecureComm), August 2018
- Xueping Liang, Sachin Shetty, Deepak Tosh, Charles Kamhoua, Kevin Kwiat, Laurent Njilla, "ProvChain: A Blockchain-based Data Provenance Architecture in Cloud Environment with Enhanced Privacy and Availability", The 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), May 2017.
- Deepak Tosh Sachin Shetty, Xueping Liang, Charles Kamhoua, Kevin Kwiat, Laurent Njilla, "Security Implications of Blockchain Cloud with Analysis of Block Withholding Attack", 17th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID), May 2017.

Thank You !

Questions?