Access Control For Distributed Ledgers In The Internet Of Things: A Networking Approach (2021)

Andrew Cullen, Pietro Ferraro, William Sanders, Luigi Vigneri and Robert Shorten
Presentation Contents

1. Introduction to Internet of Things
2. Distributed Ledger Technology
3. Important Concepts
4. Motivation and problem statement
5. Contributions
   a. Model of access control problem
   b. Access Control Algorithm
   c. Extensive Simulations
6. Conclusion
The Internet of Things

Physical objects that communicate over the Internet

Resource constrained environment

Requires high throughput and low latency

Networks of large scale

Security and Integrity is vital

Examples: Health data monitoring or Intelligent Vehicles
Distributed Ledger Technology

Decentralized database
- E.g. Bitcoin, Ethereum, IOTA

No owner platform
- E.g. no Google, Amazon

Trust in cryptography/code

Immutable data record
- Stored in all nodes
- Single source of truth

Src: https://openclipart.org/detail/277506/Centralized-Decentralized-and-Distributed-Networks
Example of Peer-to-Peer IoT network
Access Control for DLTs (Sybil protection)

Determines who gets to write new transactions

Protects from *Sybil attacks*

- Attacker creates multiple identities to gain majority vote
Classical networks vs DLT networks

Classical networks may rely on trusted nodes

In DLTs no individual node may be trusted
Blockchain vs DAG

DAG = Directed Acyclic Graph

Each transaction can be linked to more than one transaction

Allows parallel processing

No throughput limit
PoW vs Reputation-based Sybil Protection

PoW access control
- Solve a computationally difficult puzzle
- Prove possession of computing resources
- High energy consumption

Reputation-based access control
- Does not rely on computing resources
- Allocates a portion of the network resources based on reputation
- Reputation gained through honest participation
- Might be more suitable in constrained environments
Motivation and Problem Statement

Propose access control algorithm

- Find alternative to PoW
- Full utilisation of network resources
- Fair access based on nodes’ reputations
- Function in constrained environment (IoT)

Maximise the rate of dissemination of transactions, while achieving:

- Consistency
- Fairness in dissemination rate
- Fairness in latency
- Security
Overview of Contributions

To model the IoT access control problem

Present an access control algorithm for DAGs

Components:

- Scheduling algorithm
- Rate setting algorithm
- Buffer management scheme

Extensive simulations to prove robustness
Access Control Model

**Notation for node and network model.**

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\mathcal{M}$</td>
<td>set of all nodes in the network</td>
</tr>
<tr>
<td>$\mathcal{M}^*$</td>
<td>set of all honest nodes</td>
</tr>
<tr>
<td>$\mathcal{N}_m$</td>
<td>set of nodes that are neighbours of node $m$</td>
</tr>
<tr>
<td>$\text{rep}_m$</td>
<td>reputation of node $m$</td>
</tr>
<tr>
<td>$\mathcal{V}_m$</td>
<td>set of transactions visible to node $m$</td>
</tr>
<tr>
<td>$\mathcal{L}_m$</td>
<td>set of finalised transactions in node $m$’s ledger</td>
</tr>
<tr>
<td>$\mathcal{D}$</td>
<td>set of all disseminated transactions</td>
</tr>
<tr>
<td>$\mathcal{D}_i$</td>
<td>set of disseminated transactions issued by node $i$</td>
</tr>
<tr>
<td>$\lambda R$</td>
<td>dissemination rate (all transactions)</td>
</tr>
<tr>
<td>$\lambda R_i$</td>
<td>dissemination rate (transactions issued by node $i$)</td>
</tr>
<tr>
<td>$\nu$</td>
<td>global transaction writing power</td>
</tr>
<tr>
<td>$\lambda_m$</td>
<td>issue rate of node $m$</td>
</tr>
<tr>
<td>$\tilde{\lambda}_m$</td>
<td>assured issue rate of node $m$</td>
</tr>
</tbody>
</table>
Assured Issuing Rate

Fair allocation of writing power permits each node:

\[ \tilde{\lambda}_m = \frac{\nu \cdot rep_m}{\sum_{i \in M} rep_i}. \]

Based on node reputation relative to total reputation in the network
Paper defines 4 modes of node operation

Inactive Node: \( \lambda_m = 0 \)

Content Node: \( \lambda_m \leq \tilde{\lambda}_m \)

Best-Effort Node: \( \lambda_m > \tilde{\lambda}_m \)

Malicious Node: \( \lambda_m \gg \tilde{\lambda}_m \)
Goals for Access Control Algorithm

Scheduling Algorithm
- Transactions issued by honest nodes should not experience congestion
- In presence of congestion, the rate should be set to be proportional to the nodes reputation

Rate Setting Algorithm
- Allow best-effort nodes to issue transactions without congestion or delay

Buffer Management Scheme
- Ensure that only malicious transactions are dropped
- Ensure that capacity is not reached
Scheduling Algorithm

Based on Deficit Round Robin (DRR)

- Each flow (buffer) of transactions are visited in round robin cycle
- Deficit assigned to the flow
- A flow needs sufficient deficit to schedule a transaction
- BUT, modified for DLT networks:
  - High variance in reputation
  - Bursts of network traffic
- Instead allow flows to gain deficit up to some limit
- Saving rather than going in to debt
Algorithm 1 DRR—Scheduler

Repeat for $i \in M$ in a round robin cycle:

1: if $DC_m(i) < DC_{max}$ then
2: $DC_m(i) \leftarrow DC_m(i) + Q_i$
3: end if
4: while $|Inbox_m(i)| > 0$ do
5: $tx \leftarrow$ oldest transaction in $Inbox_m(i)$
6: if $DC_m(i) \geq |tx|$ then
7: Schedule $tx$
8: $DC_m(i) \leftarrow DC_m(i) - |tx|$
9: Wait $|tx|/\nu$ seconds
10: else
11: break
12: end if
13: end while

---

TABLE II
SCHEDULING ALGORITHM PARAMETERS.

<table>
<thead>
<tr>
<th>$DC_j(i)$</th>
<th>deficit counter for transactions in $Inbox_j(i)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q_i$</td>
<td>quantum added to $DC_j(i)$, $\forall j$ in each round ($\propto rep_i$)</td>
</tr>
<tr>
<td>$DC_{max}$</td>
<td>maximum deficit for an empty queue</td>
</tr>
</tbody>
</table>
Rate Setting Algorithm

- Inspired by TCP (Transmission Control Protocol)
- Each node uses AIMD rules
  - Additive Increase Multiplicative Decrease
- In DLT, congestion in one node means congestion elsewhere
- Locally set rate increase parameter:

$$\alpha_m \leftarrow A \cdot \frac{rep_m}{\sum_i rep_i}$$

- Updated every time a transaction is scheduled
Algorithm 2 AIMP Rate Setter (Best-effort Mode)

Repeat each time a transaction $tx$ is scheduled:
1: if $|Inbox_m(m)| > W \cdot rep_m$ then
2: \hspace{1em} $\lambda_m \leftarrow \lambda_m \cdot \beta$
3: \hspace{1em} Pause issuing and rate setting for $\tau$ seconds
4: else
5: \hspace{1em} $\lambda_m \leftarrow \lambda_m + \alpha_m \cdot |tx|$
6: end if

TABLE III

RATE SETTING ALGORITHM PARAMETERS.

<table>
<thead>
<tr>
<th>$A$</th>
<th>global additive increase parameter</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\beta$</td>
<td>global multiplicative decrease parameter</td>
</tr>
<tr>
<td>$\tau$</td>
<td>wait time parameter</td>
</tr>
<tr>
<td>$W$</td>
<td>inbox work threshold</td>
</tr>
</tbody>
</table>
Buffer Management Scheme

- Drops transactions in the buffer fairly
- Based on the nodes reputation
- Ensures that malicious nodes can not fill the buffers

Algorithm 3 Buffer Management

1: while $|Inbox_m| > W_{\text{max}}$ do
2: \hspace{1em} $d \leftarrow \arg \max_{i \in \mathcal{M}} \frac{|Inbox_m(i)|}{\text{rep}_i}$
3: \hspace{1em} Drop transaction from head of $Inbox_m(d)$
4: end while
Evaluation

Honest environment simulations

- Only inactive, content and best-effort nodes

Malicious environment simulations

- Includes malicious nodes

50 nodes with 4 randomly selected neighbours
Honest Environment Simulations

Fig. 6. Reputation distribution follows a Zipf distribution with exponent 0.9. Nodes are Content, Best-effort, or Inactive as indicated by each bar’s colour.
Honest Environment Simulations

Fig. 7. Dissemination rate and mean latency for each node.
Honest Environment Simulations

Fig. 8. Maximum time in transit, measured as time since issue for all undisseminated transactions, demonstrating that consistency is achieved.
Fig. 9. Dissemination rate and scaled dissemination rate of each node. The bottom plot of scaled dissemination rate demonstrates that fairness in dissemination rate is achieved.
Honest Environment Simulations

Fig. 10. Dissemination rate and scaled dissemination rate of each node. The highest reputation content node (purple) switches to best-effort after 90 seconds and other best-effort nodes must adapt their rates.
Fig. 11. Cumulative distribution of latency for each node for DRR scheduler and DRR—scheduler. It is shown that only approximate fairness in latency is achieved, but DRR—performs far better than standard DRR in this respect.
Malicious Environment Simulations

Fig. 12. Reputation distribution following a Zipf distribution with exponent 0.9. Nodes are Content, Best-effort, Inactive or Malicious as indicated by the colour of each bar.
Malicious Environment Simulations

Fig. 13. Maximum time in transit for transactions issued by honest nodes, measured as time since issue for all undisseminated transactions, demonstrating that consistency is achieved.
Fig. 14. Dissemination rate and scaled dissemination rate for each node. The bottom plot of scaled dissemination rate demonstrates that fairness in dissemination rate is achieved for honest nodes, while malicious nodes are penalised by the buffer management and experience lower dissemination rates.
Malicious Environment Simulations

Fig. 15. Cumulative distribution of latency for each node. Malicious nodes are shown to experience higher latency, while approximate fairness in latency is retained for honest nodes.
Comparison to state-of-the-art PoW access control

PoW access control:

- There is a set difficulty for the computational puzzle
- Determines how rapidly nodes can issue transactions
- Transactions scheduled in FIFO order

Paper demonstrated 3 cases of PoW:

1) Active computing power lower than estimated
2) Active computing power matches estimate
3) Active computing power higher than estimated
Comparison to state-of-the-art PoW access control

Fig. 22. Dissemination rate as a percentage of maximum scheduling rate, $\nu$, and mean latency for cases 1)–3) of PoW access control, shown alongside our algorithm with parameters given in Table IV.
Conclusions

Paper presents an access control algorithm for DAG based DLTs

Resources allocated based on reputation

Suitable for IoT

High dissemination rate and low latency

The algorithm is currently tested on IOTA development network

Will later be deployed to the IOTA main network
Questions?