ISRaft Consensus Algorithm for Autonomous Units

Linir Zamir, Aman Shaan and Mehrdad Nojoumian
Department of Electrical Engineering and Computer Science
Florida Atlantic University
Boca Raton, FL
*f*lzamir2016, ashaan2019, mnojoumian*g*@fau.edu

***Abstract*—Consensus protocols are a key feature in decentralized systems where multiple unreliable nodes operate, e.g., in Blockchain technologies with many worldwide applications such as supply chain management, cryptocurrencies and information sharing. ISRaft is a consensus protocol built upon Raft, a previously developed protocol that is used for replicated state machines when a group of nodes is required to achieve a consensus related to the state of the machine. This paper therefore proposes an alternative version of the ISRaft consensus protocol to allow communication among nodes in a secured fashion while maintaining the security features of the original ISRaft algorithm even in the presence of adversarial nodes. The proposed model utilizes a trust parameter to enforce cooperation, i.e., a trust value is assigned to each node to prevent malicious activities over time. This is a practical solution for autonomous units with resource-constrained devices where a regular encrypted communication method can negatively affect the system performance.**

***Index Terms*—Consensus Algorithm, Autonomous System, Blockchain Technology, Trust Model, Raft**

I. Introduction

Autonomous units can operate under many different operations and models. Much like a human, each unit is required to make decisions and be able to communicate and act upon it. Comprising many small individual nodes, autonomous units in decentralized fashion have no centralized authority to guide them. Instead, they rely on individual communications in order to complete tasks, achieve consensus and share information

to vN0(v)becaus[(N0(v))-344(sitymp [(N0(v)mmunications)-250(v)at)-490(v)ae v
thbetwe -361(uize)]T649an1
tonagement,thSom-457(5)15(v)xamp [-547(the)-548(5)15(v)xplotion, thsur15(e)-hllce.[1ltoeor03e2tite Oustt39521fenthbckchain 0echnologi

The major goal of this research is to solve the problem of secured and immutable communication in P2P networks by using a decentralized consensus protocol solution. Consensus protocols, such as PoW, use mining procedure as a way to guarantee the soundness of the block, where miners are also compensated for their works; see [5] for details of Bitcoin system, the nodes are not chosen based on their rankings or levels of trust, which provides protection for malicious nodes.

The implementation of blockchain among autonomous units provides many benefits as shown in the aforementioned papers. However, in most cases, malicious nodes are not considered. Furthermore, in the majority of these papers, proof-of-work is utilized that is not efficient enough. For instance, Strobel et al. [13] propose a method of managing these byzantine nodes through a reputation value and utilization of proof-of-work. However, if a malicious node has a high hash-rate, he can still compromise the system.

We intend to address these issues by utilizing the Raft consensus protocol. Due to the nature of this protocol and its periodical leader elections, malicious nodes will not be able to compromise the system regardless of their computational powers unless they become a leader. To further prevent a malicious node from becoming a leader, we utilize a trust pa-

- *ApproveCommit* - After a new block has been added to the node, he sends this RPC back to the leader.

These simple RPCs are what makes this consensus protocol easy to implement and operate in resource-constrained devices.

B. Leader Election

For this protocol to work, a leader must be elected. This process is done automatically when no heartbeat message is being sent over *election timeout*. After that period of time, the *follower* node changes its state to a *candidate* and he begins the election process. During this time, the node sends *RequestVote* RPC concatenate with the latest block number signed with his private-key signature to all nodes in the cluster. A follower or a candidate node that receives the RPC will then send a vote if and only if the following conditions are met:

- The node has not received any heartbeat messages from the current leader.
- The latest block number is at least equal to the current node's term plus 1.
- The *RequestVote* RPC is signed with a valid candidate's private-key.
- The trust value T of the sender is at least $0{:}5$.

A node that receives the first *RequestVote* RPC will hold the first vote until the end of the *election timeout* regardless of the candidate's trust value, as long as the candidate fulfills the conditions. However, in the case where more than one *RequestVote* arrives, the node will choose who to vote for based on the trust value. A candidate with a higher trust value T will have a higher chance of getting the vote. A leader is elected when he receives the votes of the majority of the cluster. At that point, the elected leader sends out an RPC that includes the signed vote messages that he received from the nodes in the cluster. This acts as a proof of election and also

V. Conclusions and Future Work

ISRaft is a consensus protocol that can be implemented among autonomous units in a secure data sharing environment. This protocol makes it possible to achieve consensus in the presence of adversarial nodes. Moreover, by using trust and reputation values, it becomes possible to validate the authenticity and correctness of the shared data. This can be easily implemented on resource-constrained devices and our model works perfectly in any infrastructure where regular encrypted communication methods can negatively affect the performance of the system.