Towards A Blockchain-SDN Architecture for Secure and Trustworthy 5G Massive IoT Network

Akram Hakiri
ISSAT Mateur, SYSCOM-ENIT, University of Carthage
Carthage, Tunis, Tunisia
akram.hakiri@enit.utm.tn

Behnam Dezfouli
Internet of Things Research Lab, Santa Clara University
Santa Clara, CA, USA
Email:bdezfouli@scu.edu

ABSTRACT

The emerging 5G mobile network is being seen as a prominent technology for resolving connected Internet of Things (IoT) network related issues, by allowing low power IoT devices to produce high volumes of data that can be transmitted over ultra-reliable, low-latency wireless communication services. Furthermore, incorporating Software Defined Networking (SDN) into IoT systems delivered dramatic improvements in the network agility and flexibility to solve the resource management needs of the IoT environment. However, due IoT systems encounter several security and privacy issues to prevent unauthorized access to IoT nodes and to secure trust-less IoT transactions.

To address these challenges, this paper introduces a novel Blockchain-based architecture that leverages Software Defined Network (SDN) and Network Function Virtualization (NFV) to secure IoT transactions. A novel security appliance is introduced in a form of Virtualized Network Functions (VNFs) to improve the scalability and performance of IoT networks. Then, we introduce a novel Proof-of-Authority (PoA) consensus algorithm to detect and report suspected IoT nodes and mitigate malicious traffic. We evaluate and compare our proposed solution against two well-known consensus algorithms i.e. Proof of Work (PoW) and Proof of Stake (PoS). We highlight the unique feature of our design, lightness, as it offers scalable blockchain-based secure micro-services for easier network deployment and system monitoring, smart contracts analysis and testing. We finally, demonstrate our proposal substantially guarantees a trustworthy IoT communication with privacy-preservation.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

CCS CONCEPTS

• **Networks** → **Programmable networks**; *Security protocols*; *Network privacy and anonymity*; • **Security and privacy** → *Intrusion detection systems*; *Distributed systems security*.

KEYWORDS

SDN, NFV, IoT, Blockchain, Security

1 INTRODUCTION

The emerging 5G mobile broadband network has made a tremendous growth on the Internet of Things (IoT) network [6]. Millions of battery-powered massive IoT devices such as smart cameras, environmental monitoring and smart metering are deployed to serve diverse scenarios, e.g. smart cities, autonomous farming, and smart manufacturing, require delivering high volumes of data over ultra-reliable, low-latency wireless communication services [12]. The forthcoming Ultra-WideBand (UWB) 5G will operate at the highest bandwidth's millimeter wavelength (mmWave) at frequencies of about 28 GHz and 39GHz, will favor this spectrum to meet the demands for fairly low-latency, energy-efficient, high connection density, and high-speed IoT traffic [7], and enabling massive IoT to grow unencumbered in the lower end of the frequency spectrum [8]. However, 5G-enabled massive IoT suffer a lot of security and privacy concerns [32], which hinders the reliability of the involved devices [26]. A compromised IoT devise could be prone to Distributed Denial-of-Service (DDoS) attacks and overwhelm IoT network with malicious traffic. Malignant IoT nodes can join the massive IoT network at any time and overwhelm their resources with malicious traffic to make their services unavailable. Current 5G security models that empower the IoT systems exploit the wireless channel properties [29] to enhance communication security through appropriate coding and signal processing.

Blockchain has opened up a wide range of possibilities for IoT era [24]. It implements a control logic to manage the diverse information coming from various IoT devices to provide them with a secure communication platform in IoT key themes. It also deploys a decentralized security infrastructure for fighting DDoS attacks and eliminate the risk of a

single point of failure [17]. Despite the promise, blockchain can be cost ineffective [5] for massive IoT. For example, Machado et al. [15] introduced two decentralized protocols, i.e. Proof-of-Trust (PoT) and Proof-of-Luck (PoL), which use Fog nodes as a middle layer for integrating IoT devices with the cloud. Lei et al. introduced the Groupchain [13] framework to support Fog-enabled IoT services on public blockchain. Nonetheless, these decentralized algorithms demand higher computing resources and power far beyond the reach of resource-constrained IoT devices, preventing the full adoption of distributed consensus protocols in IoT systems.

Likewise, Liu et al. [14] developed an access control system for IoT on consortium blockchain. Benedict et al. [3] proposed serverless blockchain-enabled IoT architecture for monitoring environment quality in smart city. However, these approaches relies on centralized cloud-hosted security infrastructures to deploy authentication and privacy-preserving schemes. Zhaofeng et al. [36] introduced a decentralized trust management and secure usage control scheme of IoT big data. However, it consumes substantially computation power required by miners to solve a mathematical puzzle known as Proof-of-Work (PoW) problem for creating trusted transactions. Furthermore, scalability and decentralization becomes at odds as massive IoT nodes need to store the entire blockchain transactions, state of account balances, contracts, and storage [11]. Yan et al. [31] proposed replacing traditional blockchain hash and cryptographic functions with specialized hardware component to attest that the running code was set up correctly in a protected environment. Nevertheless, breaking a single piece of trusted hardware enables the attacker to always win the lottery. Second, because smart contracts are immutable by design, upgrading their software code or patching security vulnerabilities becomes difficult and sometimes impossible.

Software Defined Network (SDN) [4] showed a significant promise in meeting IoT needs by offloading the computation to Fog infrastructures at the network edge. Aligned with SDN, Network Function Virtualization (NFV) [35] enables scaling IoT capabilities by allowing on-demand service orchestration and management. Scaling up IoT resources could be performed through Virtualized Network Functions (VNF), which in turn can be provisioned inside virtual appliances deployed on a generic hardware. In addition to improving the management of network flows in IoT systems, SDN allows better isolation of data flows and improves resiliency to failure for critical data. Specifically, SDN allows redirecting and balancing IoT flows in case of node or link failure, so that flows will be delivered to their destination while still meeting QoS requirements [33]. That is, by combining Blockchain and SDN/NFV we can optimize the management of IoT flows in response to attacks. We can also enable sophisticated analysis of IoT transactions, improve security, and increase privacy based on global network awareness given by SDN controllers.

In this paper, we introduce the design of a Blockchain-based architecture for enforcing the security of IoT transactions by implementing a SDN-aware Decentralized Application (DApp), which listen to mining nodes, reports suspicious IP addresses, and validate unknown packets. The architecture introduces a Proof-of-Authority (PoA) consensus algorithm that reveals suspected IoT smart devices and report them under smart contract. We also developed an intrusion detection system in a form of virtualized network functions (VNFs) to eliminate malicious flow and enable DDoS detection and mitigation on demand.

The remainder of this paper is organized as follows: Section 2 highlights existing approaches to integrate Blockchain in SDN-enabled IoT systems and points out how SDN operates in blockchain-based IoT networks. Section 3 describes the architecture of our solution on empowering IoT systems with SDN and blockchain. Section 4 qualitatively evaluates the performance and the scalability. Section 5 provides concluding remarks describing potential future directions and open research problems in this realm.

2 RELATED WORK

Since SDN/NFV are being nested into 5G mobile backbone to enable network softwarization and slicing, blockchain becomes a promising paradigm to address issues related to transparency, data encryption, auditability, and immutability [28].

Guo et al. [9] proposed to offload available resource on non-mining IoT devices to edge-cloud servers in order to construct collaborative mining network (CMN) to mining tasks for mobile blockchain. The authors claimed their model obtained the optimal resource price and devices' resource demands, improved mining and ensured maximum profit of edge cloud operators. Okon et al. [18] proposed a unified SDN and blockchain architecture to enhance wireless spectrum management mobile network operators (MNOs). [2] Discussed the security issues in SDN networks, and described the utilization of blockchain technology to provide confidentiality, integrity, and availability to network infrastructure. In addition to enhanced network performance, SDN/NFV and Blockchain have been merged in cluster structure to mitigate some issues such as flexibility, efficiency, availability, and security. Rahman et al. [23] used SDN/NFV and blockchain in symbiosis to offer a reliable condominium communication in smart building networks. The claimed their framework, called DistB-Condo, can robust, and secured platform to meet safety, confidentiality, flexibility, efficiency, and availability requirements needed by IoT networks.

Misra et al. [16] extended IoT security by implementing an encrypted networked clock mechanism to synchronize IoT devices with their Fog network within a private Ethereum blockchain. Hamdaoui et al. [10] implemented a decentralized protocol for enabling secure authentication, registration, and management for participatory IoT devices. The proposed scheme offers fast discovery of IoT resources and secure instantiation of IoT networks-on-demand.

Xie et al. [30] designed a blockchain-based security framework for SDN-enabled vehicular IoT services. They implemented an intelligent transportation system that relay on cloud servers to detect malicious vehicular nodes, perform real-time video reporting and trust management on vehicular messages. Pourvahab et al. [19] presented a cloud-hosted digital forensic architecture using SDN and blockchain to protect their data from unauthorized users. At the core of this architecture is a Secure Ring Verification based Authentication (SRVA) scheme to generate keys using Harmony Search Optimization (HSO) algorithm. Data are encrypted in cloud servers using Sensitivity Aware Deep Elliptic Curve Cryptography (SA-DECC) algorithm. Pourvahab et al. [20] introduced a forensics architecture in SDN-IoT that establishes the Chain of Custody (CoC) in blockchain. The CoC migrates SDN packets from malicious SDN routers to nearby switches. The packets disobeying flow rules will be discarded. Sharma et al. [25] proposed a novel blockchain-based distributed cloud architecture with a SDN-enabled controller fog nodes at the edge of the network to meet the required design principles.

Zhang et al. [33] proposed an SDN-based integrated IP source address validation architecture (ISAVA) for intra-domain and the inter-domain areas. The proposed framework relies on an SDN incremental deployment scheme which can achieve IP prefix (subnet)-level validation granularity with minimum SDN devices deployment. While among ASes, ISAVA sets up border server and establishes a vouch mechanism between allied ASes for signing outbound packets so as to achieve AS-level validation granularity. Houda et al. [1] introduced SDN-based framework, called Cochain-SC, for intra-domain and inter-domain DDoS mitigation. Cochain-SC relies on Ethereum's smart contracts to facilitate the collaboration among SDN-based large-scale domains and showed high accuracy in detecting illegitimate flows.

Unlike the aforementioned approaches, our solution delegates blacklisting and whitelisting IP addresses to Virtualized Network Functions (VNFs) instances inside Docker containers. The VNFs trustworthy maintain all reports about whitelisted and black-listed IP addresses. VNF instances can be dynamically deployed to meet changing conditions and accommodate to higher traffic demand or more stringent service requirements. Furthermore, rather than using energy intensive PoW as in [21] or the Sybil-vulnerable POET, we introduced a Proof-of-Authority (PoA) consensus algorithm to select a pre-qualified number of IoT nodes for validating transactions according to strict rules.

Compared to [27], we implemented a Blockchain Decentralized Application (DApp) as a SDN northbound network application to enforce trust on IoT transactions. The DApp can list and report suspicious IoT nodes and validate (or not) unknown blocks. Moreover, compared to permissioned Blockchain approach in [22], we employ state machine replications in a form of VNF appliances to deal with existing cloud-hosted Byzantine nodes, and enable DDoS detection and mitigation-on-demand. That is, in our architecture distributed SDN controllers are aligned with distributed blockchain nodes to avoid unified IoT vulnerability attacks and emphasis geographical distribution of Fog computing nodes and thereby latency reduction.

3 SYSTEM MODEL

This section delves into the architectural overview of our Blockchain-SDN framework for flexible resource management tamper-resistant massive IoT communication.

3.1 Overview of the Blockchain-SDN enabled Architecture

The system design of our proposed framework is described in Figure 1, which comprises four different layers. The first layer illustrated in Figure 1 is the blockchain networking layer, which allows storing and sharing data in a distributed file system. Validating nodes maintain a full copy of every block of transactions in the blockchain, check (approve or reject blocks) and confirm them against the consensus rules (see Section 3.4), which involves nodes relaying data to one another. Specifically, each validating node will act as service provider in the platform that can interact with other nodes in the blockchain through distributed smart Service Level Agreement (SLA) that can guarantee the trustworthy of involved IoT transactions. These smart SLA are blockchain smart contracts, which are self-executing contract objects that make it easy to interact with blockchain nodes to exchange data in a trusted, conflict-free manner. Trusted transactions to be executed by a network of mutually distrusted nodes using consistent distributed agreements, without the need for a central arbitration authority. Thus, integrating the SLA into blockchain smart contract makes it possible to verify that delivered service fulfill the required Quality of Service (QoS).

The second layer involves both the virtualization layer and the controller network service abstraction layer. The Blockchain virtualization is performed using Kubernetes as an Infrastructure as code (IaC), where virtualized appliances are deployed inside small execution unit called pods, which maintain lightweight Docker containers in a form of microservices. On the one hand, virtual appliances host distributed

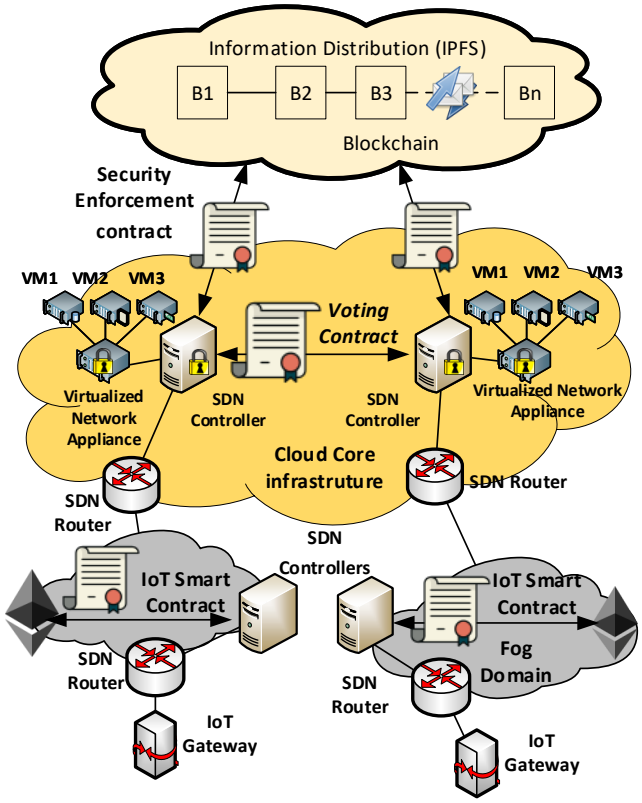

Figure 1: Overview of the Blockchain-SDN IoT architecture.

ledger nodes and communicate with the main blockchain network and perform agreement-driven decisions between each other. These appliances are running across multiple physical hosts to offer management features and facilitate the orchestration of VNFs. On the other hand, they communicate with blockchain applications (i.e. DApps) using low level Application Binary Interface (ABI) calls over remote procedure call (RPC) API to interact with smart contracts. Thanks to JSON interface that converts contract agreements, i.e. ABI, into RPC calls without relying on a third party authority.

Additionally, distributed SDN controllers are responsible for distributing security policies between blockchain nodes and IoT network infrastructure. Thanks to the decentralized applications (DApps) running inside these controllers, which trigger the generation of transactions data from different IoT nodes. All transactions are cryptographically secured using hash functions and embedded inside blocks of data. Then, consensus-driven decisions are made between DApps to validate blocks generated by different IoT nodes. Once validated, blocks are immutable and their content will not be altered, modified or deleted during the process. Furthermore, the SDN control plane in Figure 1 encompasses softwarized agile, flexible, and communication layer that translates Blockchain decisions (i.e. transactions and blocks validations) into flow rules to program the underlying SDN routers according to the application requirements. Specifically, the controller listens to the incoming IoT traffic and reports suspicious IP addresses before validating unknown packets. Besides, intrusion detection VNFs (i.e. Firewall as a Service) are deployed inside Kubernetes clusters to take care of malicious flows and enables DDoS detection and mitigation-on-demand. The SDN controller triggers storing decisions to VNF instances to maintain reports about whitelisted and blacklisted IP addresses. The Kubernetes manager can dynamically scale up and down clustered VNFs to meet changing conditions and accommodate higher traffic demand or more stringent service requirements.

Finally, the data plane abstraction layer in Figure 1 contains both SDN virtual routers and switches as well as the abstraction device layer. It gathers sensing data from IoT gateways, which interface remote sensors and actuators. SDN controllers implement security policies to protect the underlying virtual routers and switches against eventual intrusion. As the SDN routers are directly connected to the blockchain, data are encrypted before being transmitted to remote participants.

3.2 Blockchain as a Service

Figure 2 depicts the details of flow management through different layers. First, the Blockchain layer is composed of four modules: 1) the identification module manages the user/node access using the private and public keys. Indeed, IoT node addresses are inferred from their own public keys in the blockchain (i.e. a node address is the last 20 bytes string from the 32-byte string public key after dropping 12 of these bytes), which is also associated with node balances and used for sending and receiving transactions.

Furthermore, since each IoT node could have one or multiple accounts (i.e. called Externally Owned Account (EOA)), it should have different identification scenarios for each EOA. Therefore, the framework implements another module for the Authentication, Authorization and Accounting (AAA) with the Blockchain. Thus, an IoT node can access the infrastructure service using a given account for a given scenarios, and interact with the blockchain through API calls to reserve the required resources and execute the transactions. The authentication is based on identity to ensure impersonation prevention, protect the control and data planes against intrusion, and ensure that malicious attacks do not tamper with the controller configuration.

Similarly, the traceability module offers the ability to trace the entire lifestyle of a transaction, from its originating node to every processing on the blockchain infrastructure. The

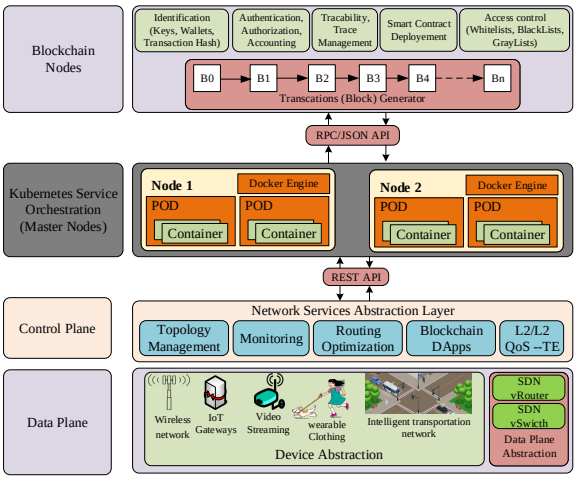

Figure 2: Blockchain-SDN Applications Framework in Smart City Security.

smart contract deployment module allows the interaction between contract functions and IoT nodes from their creation to their deployment. Finally, the access control module (will be discussed in Section 3.4) implements the functions for enforcing trust on transactions by listening to mining nodes and reporting suspicious IP addresses.

In the meanwhile, Kubernetes orchestration layer allows creating a set of network functions that can be deployed into software packages, assembled and chained to create the services required by IoT nodes. It also coordinates and orchestrates the virtual appliances (i.e. containers) either when predefined resource limits are being reached or after receiving trigger events from the underlying SDN controller. The latter will also sign and verify IoT transactions across distributed IoT nodes in which data could be signed and verified in near real-time. Leveraging SDN/NFV enforces the coordination of distributed IoT nodes and increases their performance by creating a modular architecture in which virtual miners can be hosted inside a NFV platform such as the Open Platform for NFV (OPNFV) [34]. On the other hand, the SDN controller network abstraction layer can enforce the security policies and configuration of the data plane by protecting flow table rules inside virtual SDN routers from intentional or unintentional tampering.

3.3 Smart Contract

The smart contract is a software program stored on the Blockchain infrastructure, which needs to be validated by distributed nodes like any other regular transaction generated by IoT devices. However, smart contracts have a specific built-in account in the blockchain without any private key. They are stored and managed as special transactions that can be used to interact with D-APPs. Our approach uses the Ethereum Blockchain platform to build the smart contract in the Solidity language, which are than compiled into "*bytecode*", read and executed over a portable execution environment called the "*Ethereum virtual machine*" (EVM). Algorithm 1 illustrates the contract deployment to provide security while transaction and reduce surplus transaction costs.

Algorithm 1 describes how the SDN controller can enforce trust of the network using function *setFlowRulesTurstList* in line 2. Any detected misbehavior is reported not only based on its MA and IP addresses, but also by identifying the business application impacted by possible intrusion. A blockchain validator is introduced to check the validity of IoT devices connected over the blockchain. The validator parses the OpenFlow messages to identify the source and destination of incoming traffic. The SDN controller uses the information contained in the OpenFlow packet headers to create a wide network view including topology state and transactions meta-data.

By expecting and parsing every OpenFlow packet exchanged between the IoT devices and the network, the SDN controller can identify every abnormal behavior in the network. That is, if an attacker wants to take control of any IoT device, the changes of device ownership in the network will be visible in the topology viewer module within the SDN controller.

This method allows the SDN control plane to distinguish two types of lists, i.e. blacklisted devices and whitelisted ones. Algorithm 2 is responsible for updating blacklisted and white listed nodes. The former's are suspicious users whose behavior is abnormal (i.e. representatives of malicious attack or unexpected behavior) so the controller should isolate them from sending traffic on the blockchain. The function *servergateway()* is called when overwhelmed node should be removed the whole network. Whitelisted nodes are users or devices whose behavior is normal, and they could continue delivering their content as they belong to the blockchain.

3.4 Consensus Agreement

We rely to the Proof-of-Authority (PoA) consensus algorithm (described in Figure 3) to select a set of N trusted nodes called the authorities. To enforce the network security, the PoA selects a pre-qualified number of IoT nodes for validating transactions according to strict rules. First, nodes are elected based on their QoS parameters, i.e. higher bandwidth link, lower latency, and higher hardware resources performance (CPU, Memory, link quality). These nodes can themselves elect a limited number of leaders which have a set of authorities to maintain and keep the network working.

By leveraging the identity of pre-selected nodes, e.g. nodes ***a1*** and ***a2*** in Figure 3, our framework gives more importance to a node's reputation. The advantages of this approach are

Algorithm 1: Deploy the Smart Contract.

Input: Contract Address & ABI
Output: List of available nodes
// Initialize contract address & ABI
ABI ← ABI
contractAddress ← contractAddress
// Deploys contract
get_instance(class ContractInterface)
deployContract(ABI, contractAddress)
// Retreive IoT application by blockchain Keys:Accounts
nodes ← mycontract(nodeAddress).getAllApps()
for *i ← 1* **to** *length(nodes)* **do**
    // Get nodes details
    getNodesDetails ← mycontract(getAppsDetails(nodes[i])
    // generate a list of application details
    ListApps[apps[i]] ← nodesDetails
// Sending maching packets of trusted nodes to controller for further processing
for *j ← 1* **to** *lenListApps* **do**
    setFlowRulesTurstList(ListApps[apps[i]])
// Receive controller messages about connected IoT devices
devices ← getAllDevices()
for *i ← 1* **to** *len devices* **do**
    // Get IoT device info
    deviceInfo ← mycontract(getIoTDevDetails(devices[i]))
    // Get business application layer details for connected IoT device
    appDevInfo ← mycontract(getIOTDevApps(DetailsIOTDev[1]))
    // List IoT devices by their Apps
    listDevs[devices[i]] ← (IP, MAC, Apps)
// Secure trusted IoT apps by protocol:ports
for *k ← 1* **to** *length(ListApps)* **do**
    tmp ← ListApps[k]
    portsDev.Append(tmp["appProtocol"]"-"tmp["appPort"])

Algorithm 2: UPDATE LIST OF TRUSTED NODES

// Update list of ports
common ← list(set(ports) & set(PortsDev)) ;
remove ← list(set(ports) - set(common));
addition ← list(set(ports1) - set(common)) ;
if *length(addition) ≠ 0* **then**
    for *i ← 1* **to** *length(addition)* **do**
        key ← addition[i] ;
        for *k ← 1***to** *length(ListApps)* **do**
            p ← ListApps[k] ;
            if *key["appPort"] = p[1]* **then**
                // Port added
                file.write(*p*1 + "," + *key*"*time*"") ;
for *i ← 1 ListApps length(ListDevs)* **do**
    src_mac ← ListDevs[i] ;
    src_ip ← key['IP'] ;
    myapp ← key['myApp'] ;
    if *key in ListMAc* **then**
        for *i ← 1* **to** *length(myapp)* **do**
            if *myappi in ListApps* **then**
                dst_ip ← ListApps[myapp[i]]['appIPaddr'] ;
                dst_port ← ListApps[myapp[i]]['appPort'] ;
                proto ← ListApps[myapp[i]]['appProtocol'] ;
                source ← ListMAc[key]['dpid'] - ListMAc[key]['dpid'] ;
                dest ← servergateway ;
                // Create permenant graph
                create_graph (source, *servergateway*, *src_mac*, *scr_ip*, *dst_ip*, dst_port, *proto*)

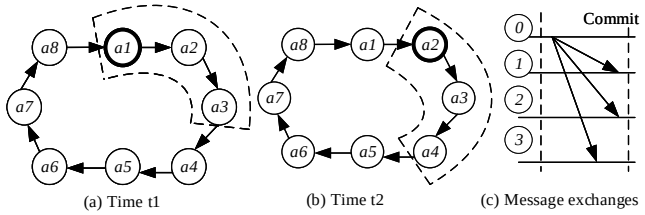

Figure 3: Proof of Authority Consensus Mechanism

twofold: first, it helps in keeping the decentralization more efficient while requiring less computational power. Second, by relying on a group of pre-approved authority nodes to verify transactions and build blocks, we ensure that nodes wishing to become validating should disclose their identity. A dedicated data-store is used to keep the list of pre-approved nodes, and new active nodes who wish to join the group of authorities, should comply with series of rules to be considered

trustworthy, i.e. should be elected by at least 51% of existing ones.

4 RESULTS

We consider two key properties, i.e. the performance and scalability, to determine the effectiveness and fitness of the consensus algorithm. The performance refers to transaction latency and throughput, i.e. a transaction is not considered valid until it is committed out to the blockchain. The performance is bounded by a combination of block interval (i.e. time between publishing subsequent blocks) and block size. These parameters establish an upper bound on transaction throughput.

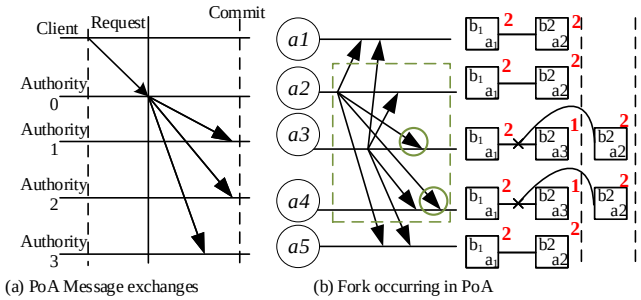

Figure 4: Message Exchange in PoA

Figure 4 shows the message exchange at each step, where each leader node broadcasts a block to all other authorities, which in turn commit it to the blockchain. Additionally, in terms of transaction's latency, i.e the time between submitting a transaction t by a participating node and its commit of a block including t by a leader node, PoA performs better latency compared to other consensus algorithms. PoA is communication oriented consensus mechanism, that does not involve relevant computation, and assumes bounded latency expressed in terms of time steps, rather than CPU-bound (e.g. PoW). PoW algorithm performs 10 minutes average latency in Bitcoin blockchain and PoS carries out 12 seconds in Ethereum. The average transaction latency of our PoA blockchain system was around 30 ms.

As shown in Figure 5, the PoW consensus algorithm is the most reliable and secure among the three algorithms. However, it is not scalable because it has limited transaction per second (TPS) performance. The PoW requires four message rounds to commit a block, which means that before a new IoT transaction block is confirmed, it should be verified and approved by most network nodes. Similarly, the PoS shows better performance than the PoW as it needs three message rounds per IoT transaction to validate a block. However, this difference is not very impressive and the PoS failed to solve the scalability problem. This where the PoA performs better

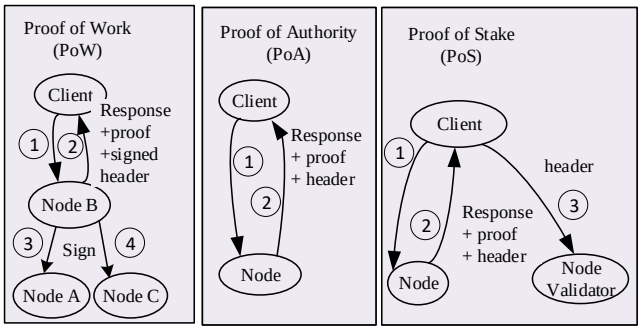

Figure 5: Transaction's Throughput of the three algorithms

transaction's throughput. In PoA each block proposal requires only one round for leader to send the proposed block to all other authorities. The block is committed at once, hence the latency in terms of message rounds is 1. PoA requires less message exchanges and hence performs better transaction's throughput.

5 CONCLUSION

In this study, we have proposed a distributed secure blockchain-based SDN architecture for secure and tamper-resistant massive IoT settings. We present a a Proof-of-Authority (PoA) consensus mechanism to select pre-qualified IoT devices to validate their transactions and verify the correctness of exchanged blocks and perform lightweight mining. An election-based process is established between these pre-qualified nodes to validate transactions and verify the correctness of exchanged blocks. Our results confirm our claims that the solution we propose can readily be used to detect and eliminate falsified IoT transactions. Using the PoA consensus protocol, our mining IoT blocks offers lower-latency by up to 66% compared against the two other approaches. In the future, we will demonstrate a systematic approach to improve SDN-Blockchain with Federated Machine Learning (FedML) and solve the issue of data ownership and privacy, and in the process also demonstrate the novel application of blockchain and distributed consensus in AI/ML.

ACKNOWLEDGMENTS

This work was funded by the NGI Explorers Program under the Horizon 2020 Research and Innovation Framework (H2020), Grant Agreement number: 825183, Call identifier: H2020-ICT-31-2018. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of NGI or H2020.

REFERENCES

[1] Z. Abou El Houda, A. S. Hafid, and L. Khoukhi. 2019. Cochain-SC: An Intra- and Inter-Domain Ddos Mitigation Scheme Based on Blockchain Using SDN and Smart Contract. *IEEE Access* 7 (2019), 98893–98907.

[2] T. Alharbi. 2020. Deployment of Blockchain Technology in Software Defined Networks: A Survey. *IEEE Access* 8 (2020), 9146–9156.

[3] S. Benedict. 2020. Serverless Blockchain-Enabled Architecture for IoT Societal Applications. *IEEE Transactions on Computational Social Systems* 7, 5 (Oct 2020), 1146–1158.

[4] S. Bera, S. Misra, and A. V. Vasilakos. 2017. Software-Defined Networking for Internet of Things: A Survey. *IEEE Internet of Things Journal* 4, 6 (Dec. 2017), 1994–2008.

[5] Francesco Buccafurri, Gianluca Lax, Serena Nicolazzo, and Antonino Nocera. 2017. Overcoming Limits Of Blockchain For IoT Applications. In *Proceedings of the 12th International Conference on Availability, Reliability and Security (ARES '17)*. Article 26, 6 pages.

[6] L. Chettri and R. Bera. 2020. A Comprehensive Survey on Internet of Things (IoT) Toward 5G Wireless Systems. *IEEE Internet of Things Journal* 7, 1 (2020), 16–32.

[7] I. B. F. de Almeida, L. L. Mendes, J. J. P. C. Rodrigues, and M. A. A. da Cruz. 2019. 5G Waveforms for IoT Applications. *IEEE Communications Surveys Tutorials* 21, 3 (2019), 2554–2567.

[8] J. Ding, M. Nemati, C. Ranaweera, and J. Choi. 2020. IoT Connectivity Technologies and Applications: A Survey. *IEEE Access* 8 (2020), 67646–67673.

[9] S. Guo, Y. Dai, S. Guo, X. Qiu, and F. Qi. 2020. Blockchain Meets Edge Computing: Stackelberg Game and Double Auction Based Task Offloading for Mobile Blockchain. *IEEE Transactions on Vehicular Technology* 69, 5 (May 2020), 5549–5561.

[10] B. Hamdaoui, M. Alkalbani, A. Rayes, and N. Zorba. 2020. IoTShare: A Blockchain-Enabled IoT Resource Sharing On-Demand Protocol for Smart City Situation-Awareness Applications. *IEEE Internet of Things Journal* 7, 10 (Oct 2020), 10548–10561.

[11] R. Han, V. Gramoli, and X. Xu. 2018. Evaluating Blockchains For Iot. In *2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS)*. 1–5.

[12] H. Hejazi, H. Rajab, T. Cinkler, and L. Lengyel. 2018. Survey of platforms for massive IoT. In *2018 IEEE International Conference on Future IoT Technologies (Future IoT)*. 1–8.

[13] K. Lei, M. Du, J. Huang, and T. Jin. 2020. Groupchain: Towards a Scalable Public Blockchain in Fog Computing of IoT Services Computing. *IEEE Transactions on Services Computing* 13, 2 (March 2020), 252–262.

[14] H. Liu, D. Han, and D. Li. 2020. Fabric-iot: A Blockchain-Based Access Control System in IoT. *IEEE Access* 8 (2020), 18207–18218.

[15] C. Machado and A. A. Medeiros Fröhlich. 2018. IoT Data Integrity Verification for Cyber-Physical Systems Using Blockchain. In *2018 IEEE 21st International Symposium on Real-Time Distributed Computing (ISORC)*. 83–90.

[16] S. Misra, A. Mukherjee, A. Roy, N. Saurabh, Y. Rahulamathavan, and M. Rajarajan. 2021. Blockchain at the Edge: Performance of Resource-Constrained IoT Networks. *IEEE Transactions on Parallel and Distributed Systems* 32, 1 (Jan 2021), 174–183.

[17] O. Novo. 2018. Blockchain Meets IoT: An Architecture For Scalable Access Management In IoT. *IEEE Internet of Things Journal* 5, 2 (April 2018), 1184–1195.

[18] A. A. Okon, I. Elgendi, O. S. Sholiyi, J. M. H. Elmirghani, A. Jamalipour, and K. Munasinghe. 2020. Blockchain and SDN Architecture for Spectrum Management in Cellular Networks. *IEEE Access* 8 (2020), 94415–94428.

[19] M. Pourvahab and G. Ekbatanifard. 2019. Digital Forensics Architecture for Evidence Collection and Provenance Preservation in IaaS Cloud Environment Using SDN and Blockchain Technology. *IEEE Access* 7 (2019), 153349–153364.

[20] M. Pourvahab and G. Ekbatanifard. 2019. An Efficient Forensics Architecture in Software-Defined Networking-IoT Using Blockchain Technology. *IEEE Access* 7 (2019), 99573–99588.

[21] Pradip Kumar Sharma and Jong Hyuk Park. 2018. Blockchain based hybrid network architecture for the smart city. *Future Generation Computer Systems* 86 (2018), 650–655.

[22] C. Qiu, F. R. Yu, H. Yao, C. Jiang, F. Xu, and C. Zhao. 2018. Blockchain-Based Software-Defined Industrial Internet of Things: A Dueling Deep Q-Learning Approach. *IEEE Internet of Things Journal* (2018), 1–1.

[23] A. Rahman, M. J. Islam, Z. Rahman, M. M. Reza, A. Anwar, M. A. P. Mahmud, M. K. Nasir, and R. M. Noor. 2020. DistB-Condo: Distributed Blockchain-Based IoT-SDN Model for Smart Condominium. *IEEE Access* 8 (2020), 209594–209609.

[24] R. Sekaran, R. Patan, A. Raveendran, F. Al-Turjman, M. Ramachandran, and L. Mostarda. 2020. Survival Study on Blockchain Based 6G-Enabled Mobile Edge Computation for IoT Automation. *IEEE Access* 8 (2020), 143453–143463.

[25] P. K. Sharma, M. Chen, and J. H. Park. 2018. A Software Defined Fog Node Based Distributed Blockchain Cloud Architecture for IoT. *IEEE Access* 6 (2018), 115–124.

[26] Sabrina Sicari, Alessandra Rizzardi, and Alberto Coen-Porisini. 2020. 5G In the internet of things era: An overview on security and privacy challenges. *Computer Networks* 179 (2020), 107345.

[27] M. Steichen, S. Hommes, and R. State. 2017. ChainGuard : A firewall for blockchain applications using SDN with OpenFlow. In *2017 Principles, Systems and Applications of IP Telecommunications (IPTComm)*. 1–8.

[28] M. Tahir, M. H. Habaebi, M. Dabbagh, A. Mughees, A. Ahad, and K. I. Ahmed. 2020. A Review on Application of Blockchain in 5G and Beyond Networks: Taxonomy, Field-Trials, Challenges and Opportunities. *IEEE Access* 8 (2020), 115876–115904.

[29] Y. Wu, A. Khisti, C. Xiao, G. Caire, K. Wong, and X. Gao. 2018. A Survey of Physical Layer Security Techniques for 5G Wireless Networks and Challenges Ahead. *IEEE Journal on Selected Areas in Communications* 36, 4 (2018), 679–695.

[30] L. Xie, Y. Ding, H. Yang, and X. Wang. 2019. Blockchain-Based Secure and Trustworthy Internet of Things in SDN-Enabled 5G-VANETs. *IEEE Access* 7 (2019), 56656–56666.

[31] W. Yan, N. Zhang, L. L. Njilla, and X. Zhang. 2020. PCBChain: Lightweight Reconfigurable Blockchain Primitives for Secure IoT Applications. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 28, 10 (Oct 2020), 2196–2209.

[32] R. Yugha and S. Chithra. 2020. A survey on technologies and security protocols: Reference for future generation IoT. *Journal of Network and Computer Applications* 169 (2020), 102763.

[33] C. Zhang, G. Hu, G. Chen, A. K. Sangaiah, P. Zhang, X. Yan, and W. Jiang. 2018. Towards a SDN-Based Integrated Architecture for Mitigating IP Spoofing Attack. *IEEE Access* 6 (2018), 22764–22777.

[34] Tianzhu Zhang. 2020. NFV Platform Design: A Survey. *arXiv: Networking and Internet Architecture* (2020).

[35] T. Zhang, H. Qiu, L. Linguaglossa, W. Cerroni, and P. Giaccone. 2020. NFV Platforms: Taxonomy, Design Choices and Future Challenges. *IEEE Transactions on Network and Service Management* (2020), 1–1.

[36] M. Zhaofeng, W. Lingyun, W. Xiaochang, W. Zhen, and Z. Weizhe. 2020. Blockchain-Enabled Decentralized Trust Management and Secure Usage Control of IoT Big Data. *IEEE Internet of Things Journal* 7, 5 (May 2020), 4000–4015.