A Blockchain-Oriented SDN-aware Architecture for a Scalable Communication in IoT Network

Akram Hakiri and Behnam Dezfouli

Abstract The widespread growth of the Internet of Things (IoT) systems has motivated the need for trusted IoT transactions, where smart devices can be active participants that share their data with cloud-hosted applications. A compromised IoT device can be prone to vulnerable attacks and overwhelm the whole network with malicious traffic. Recently, Blockchain is being envisioned to enforce security and trustworthiness in diverse IoT environments, including transactive energy auctions, connected vehicles, and trusted healthcare systems. However, blockchain incurs brutal latency, high compute costs and limited storage to process IoT transactions. It also could be cost ineffective, as it consumes substantially computing power and higher energy to process and validate IoT transactions. Additionally, the lack of IoT-focused consensus protocols makes it is difficult to coordinate distributed IoT systems to detect and destroy large-scale botnets. To address these challenges, this chapter presents the architectural design of a novel Blockchain-based IoT network architecture that leverages Software Defined Network (SDN) and Network Function Virtualization (NFV) to secure IoT transactions. We developed an intrusion detection system in a form of Virtualized Network Functions (VNFs) to improve the scalability and performance of IoT networks. We show how the design of our IoT-focused smart contract can prevent Destination Advertisement Object (DAO) induction attacks in distributed IoT network. We introduce a novel Proof-of-Authority (PoA) consensus algorithm to detect and report suspected IoT nodes and mitigate malicious traffic. We also evaluate our solution against voting-based and lottery-based consensus algorithms.

Key words: Blockchain, Distributed Ledger, Internet of Things, SDN/NFV, Ethereum, Security.

Akram Hakiri
SYSCOM-ENIT, Department of Computer Science and Telecommunications, ISSAT Mateur, University of Carthage, Tunisia. e-mail: akram.hakiri@enit.utm.tn

Behnam Dezfouli
Internet of Things Research Lab, Department of Computer Science and Engineering, Santa Clara University, USA. e-mail: bdezfouli@scu.edu

1 Introduction

The growing evolution of the Internet of Things (IoT) with tremendous growth in sensors and actuators motivated the need for trusted transactions due the transformation of IoT devices from smart sensing to being active participants that share their data with cloud-hosted applications. IoT systems encounter several security and privacy concerns to prevent unauthorized access to smart devices and to secure trust-less interactions between devices themselves and with service providers on the Internet [4]. A compromised IoT device could be prone to Distributed Denial-of-Service (DDoS) attacks and overwhelm IoT network with malicious traffic. Malignant IoT nodes can join the network at any time and overwhelm their resources with malicious traffic to make their services unavailable. Current security models [34] for IoT communication such as centralized cloud-based security infrastructures cannot address the IoT's security and privacy concerns because of lacking resources and flexibility, which makes IoT devices susceptible to elevation of privileges and spoofing attacks.

An attractive and more realistic alternative is Blockchain [40], which deploys a decentralized infrastructure for fighting DDoS attacks and eliminate the risk of a single point of failure. Blockchain has been seen as the backbone for diverse IoT applications, such as transactive energy auctions [19], guaranteeing fair payments in smart grids [30], electric vehicles [25], monitoring environment quality in smart city [24], and trusted healthcare systems [67]. Despite these promises, Blockchain can be cost ineffective [9] because it consumes a substantial amount of computation power by miners to solve a mathematical puzzle known as Proof-of-Work (PoW) problem for creating trusted transactions. Besides, scalability and decentralization are at odds since all IoT nodes need to store the entire blockchain transactions, state of account balances, contracts, and storage. As the number of connected IoT devices is forecast to grow to almost 31 billion in the next decade, scalability challenges worsen, especially when it comes to process billions of transactions generated by massive IoT devices [22]. Besides, smart contracts come with many disadvantages when deployed in IoT networks. In particular, because smart contracts are immutable by design, upgrading their software code or patching security vulnerabilities becomes difficult and sometimes impossible. Furthermore, since most IoT devices run over centralized resource-constrained platforms with low memory footprint and computation resources, blockchain can be cost ineffective [9] for massive IoT. Blockchain requires computing resources far beyond the reach of resource-constrained IoT devices, thereby preventing the full adoption of distributed consensus protocols in IoT systems. Thus, Blockchain is not widely adopted in resource constrained IoT systems.

Software Defined Networking (SDN) [8] showed a significant promise in meeting IoT needs by offloading computation to fog infrastructures at the network edge [43]. Aligned with SDN, Network Function Virtualization (NFV) [68] enables scaling IoT capabilities by allowing on-demand service orchestration and management. Scaling up IoT resources can 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 [66]. 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 chapter, we introduce a Blockchain-based architecture for enforcing the security of massive IoT transactions. We developed a SDN-enabled intrusion detection system in a form of virtualized network functions (VNFs) to eliminate malicious flows and enable DDoS detection and mitigation on demand. These VNFs are connected to blockchain Decentralized Application (DApp) instances to maintain all reports about trustworthy and untrustworthy IP addresses. The architecture introduces a Proof-of-Authority (PoA) consensus algorithm that reveals suspected IoT smart devices and reports them under smart contract.

The remainder of this chapter 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 presents three use cases that can benefit from our Blockchain-enabled SDN architecture. Section 5 qualitatively evaluates performance and scalability. Section 6 discusses open challenges that should be addressed in the near future to support other technical requirements for improving advanced trusted smart IoT networks. Section 8 provides concluding remarks describing potential future directions and open research problems in this realm.

2 Related Work

This section draws on the research directions on the convergence of blockchain and the Internet of Things (IoT), and empowering Blockchain-based IoT networks with SDN/NFV.

2.1 Blockchain Integration with IoT

Blockchain has opened up a wide range of possibilities for IoT era as 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. For example, Machado et al. [32] introduced two consensus algorithms, 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. Chen et al. [11] introduced Devify framework to build an interoperable trusted IoT networks in a decentralized fashion. The framework adopts the Web of Things ontology model to develop cloud-hosted Blockchain IoT applications. Similarly, Singh at al. [56] introduced a unique crypto ID called Trust Bit (TB) for decentralized intelligent vehicle (IV) communication. The authors created a reward system to store

Trust bit details and reward trusted IVs by distributing some TBs after successful and trusted inter-IVs communication.

Ellul el al. [14] proposed the Alkyl Virtual Machine (AlkylVM), where an Aryl blockchain agent acts as an interface between IoT network and Ethereum blockchain. AlkylVM continues using the traditional energy intensive Proof-of-Work (PoW) consensus mechanism to validate all transactions on behalf of IoT devices, thereby offloads the computations of resource-constrained IoT devices. Novo [40] introduced a fully decentralized architecture to manage IoT communication in permissioned IoT network. multiple smart contracts, distributed miners add transaction records into the blockchain, then a single smart contract is used to mange consensuses in the entire network. However, transactions processing incurs long delays when a manager node to grant access to trusted nodes or deny access to particular resources in a device. An unauthorized attacker could gain access to restricted information before a manager could validate and secure transaction's data.

Yin et al. [63] proposed a blockchain-based architecture for Machine-to-Machine (M2M) communication in both public and private areas. Walker et al. [60] introduced the PlaTIBART framework for private, fault-tolerant, and transactive blockchain deployment in IoT networks. Likewise, Chen [12] introduced the FlowChain distributed ledger system over peer-to-peer IoT systems. FlowChain provides secure, real-time data exchange model to enforce IoT privacy.

2.2 Blockchain-based IoT networks with SDN/NFV

SDN and Blockchain have been merged to mitigate some issues such as flexibility, efficiency, availability, and security. Salahuddin et al. [52] argue that using SDN in blockchain-based IoT networks could enforce the security of IoT data against malicious traffic analysis. Kataoka [26] integrated SDN and blockchain to automate the process of doubting, verification and trusting of IoT web services to prevent them from attacks. Samaniego et al [53] virtualized IoT resources by combining Blockchain and SDN to enforce permission-based communication during resource provisioning. Additionally, Steichen et al. [58] proposed ChainGuard framework atop of the Floodlight controller to filter and intercept illegitimate packets and prevent malicious behavior from vulnerable sources. Abbasi et al. [1] introduced the VeidBlock framework to generate verifiable identities based on blockchain over distributed SDN infrastructure.

Likewise, Qiu et al. [45] used Dueling Deep Q-Learning approach to achieve low cost, low latency and low-band intensive network computation and optimize the trust features and throughput performance. Rodrigues et al. [51] proposed a Blockchain Signaling System (BSS) for whitelisting and blacklisting IP addresses across multi domains SDN network. Similarly, Hari et al. [23] proposed Internet Blockchain for securing Border Gateway routing Protocol (BGP) sessions and DNS transactions. Sharma et al [54] proposed the DistBlockNet framework to update OpenFlow rules, verify security of flow rule entries, and install updated flow rules to the forwarding SDN-aware IoT devices. Mendiboure et al. [36] introduced a SDN-based Application Trust Index (ATI) to enable authentication and control in Internet-of-Vehicule (IoV) during resource allocation

process. The authors used the Proof-of-Elapsed Time (PoET) consensus algorithm for Hyperledger Sawtooth to prevent high resource utilization and high energy consumption. They used the PoET algorithm by following a fair lottery system to elect (with equal opportunities) SDN controllers for managing the certification process. Participating SDN controllers select random time to win the election and become manager nodes, and the winner controller should indeed complete certain waiting time. Despite the promise, PoET is susceptible to Sybil attacks, where a single attacker can forge multiple node identities to achieve the majority of 51% and take control over the IoT network. Additionally, PoET has the disadvantage to necessary rely on specializing SGX hardware (only available from Intel) which could be a barrier as it runs against the new paradigm of removal of trust in intermediaries.

Guo et al. [17] 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. In addition to enhanced network performance, SDN/NFV and Blockchain have been merged in cluster structure to mitigate challenges such as flexibility, efficiency, availability, and security [3]. Likewise, Liu et al. [28] developed an access control system for IoT on consortium blockchain. Benedict et al. [7] 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. Lu et al. [29] proposed a SDN-based energy Internet distributed energy-trading scheme supported by blockchain. Their design offered a reasonable match of the transaction objects and allowed meeting security and privacy needs in smart grids.

Yan et al. [61] 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.

Rahman et al. [48] proposed a Blockchain-SDN architecture managing a safe and secure data transfer in smart building system. Rahman et al. [47] 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.

Gao et al. [16] enhanced the network performance of Vehicular Ad-Hoc Networks (VANETs) by incorporating SDN into decentralized blockchain infrastructure in order to track malicious activities in the network. Additionally, the authors introduced Deep Reinforcement Learning (DRL) approach to construct the trusted and auto-adjust service function chain (SFC) orchestration architecture [18] and improve resource allocation in SDN/NFV infrastructure. Houda et al. [2] introduced a 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 achieves a high accuracy in detecting illegitimate flows.

Gao et al. [16] enhanced the network performance of Vehicular Ad-Hoc Networks (VANETs) by incorporating SDN into decentralized blockchain infrastructure in order to track malicious activities in the network. Additionally, the authors introduced Deep Reinforcement Learning (DRL) approach to construct the trusted and auto-adjust service function chain (SFC) orchestration architecture [18] and improve resource allocation in SDN/NFV infrastructure. Singh et al. [55] developed a deep-learning-based blockchain to improve SDN reliability and extend the control plane beyond its centralized ecosystem, thus avoiding a single point of failure. Based on voting-based consensus mechanism, the authors proposed to use the blockchain to identify anomalous switch requests, verify and certify trustworthy SDN switches using zero-knowledge proof. Luo et al. [31] proposed to improve the scalability and the flexibility of SDN-based industrial IoT by integrating decentralized blockchain into multi-SDN distributed control plane to handle a large amount of data generated by industrial devices. The authors proposed partially observable Markov decision process (POMDP) and a deep reinforcement learning (DRL) approach to optimize the system energy efficiency, we adaptively allocate computational resources and the batch size of the block. Medhane et al. [35] described a blockchain-based framework that leverages edge-cloud and SDN to support prominent features like continuous confidentiality, authentication, and robustness. A thread detection layer is implemented at the cloud side servers to reduce the overhead of SDN-enabled IoT gateways at the edge layer. Similarly, Qiu et al. [46] proposed dueling deep Q-learning approach based on blockchain decentralized protocol to implement consensus among multiple controllers under complex industrial environments. The authors implemented a analytical model as joint optimization problem to formulate view change, access selection, and computational resource allocation. Misra et al. [38] 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. [21] 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.

2.3 Chapter Contribution

Unlike the aforementioned approaches, our solution delegates blacklisting and whitelisting IP addresses to Virtualized Network Function (VNF) instances inside Docker containers. The VNFs trustworthy maintain all reports about white-listed 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 [44] 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 [58], 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 [45], 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.

3 Architecture Overview

This section delves into the architectural details that support scalable, dynamic, and flexible resource management with our SDN-based framework. We also present algorithms to perform tamper-resistant IoT-on-Blockchain communication in symbiosis with SDN.

3.1 System Design

Figure 1 illustrates the architectural overview of our proposed solution, which comprises four different layers. First, the peer-to-peer blockchain networking layer uses the InterPlanetary File System (IPFS) for storing and sharing data in a distributed file system. Blockchain nodes, i.e., miners and clients, use IPFS to interoperable with smart contracts and blockchain transactions.

Second, both the virtualization layer and the controller network service abstraction layer are described in Figure 1. The former provides a Blockchain on Kubernetes as an Infrastructure-as-code, where applications are maintained inside Docker containers across multiple physical hosts. The virtualization layer also provides many management features to facilitate the orchestration of VNFs. On the one hand, these virtual appliances host distributed blockchain client nodes in a form of lightweight containers (i.e., Pods), which communicate with the Blockchain network and perform agreement-driven decisions between each other.

On the one hand, virtual appliances host distributed 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 agile management features and facilitate the orchestration of VNFs. On the other hand, virtual appliances communicate with blockchain-SDN applications (i.e., DApps) using low-level Application Binary Interface (ABI) calls over Remote Procedure Call (RPC) to interact with smart contracts. The contract object is converted into json interface where all the contract calls are converted into low-level ABI calls over RPC.

The distributed SDN controllers are responsible for distributing security policies among blockchain nodes and IoT network infrastructure. This is achieved by the blockchain decentralized applications (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

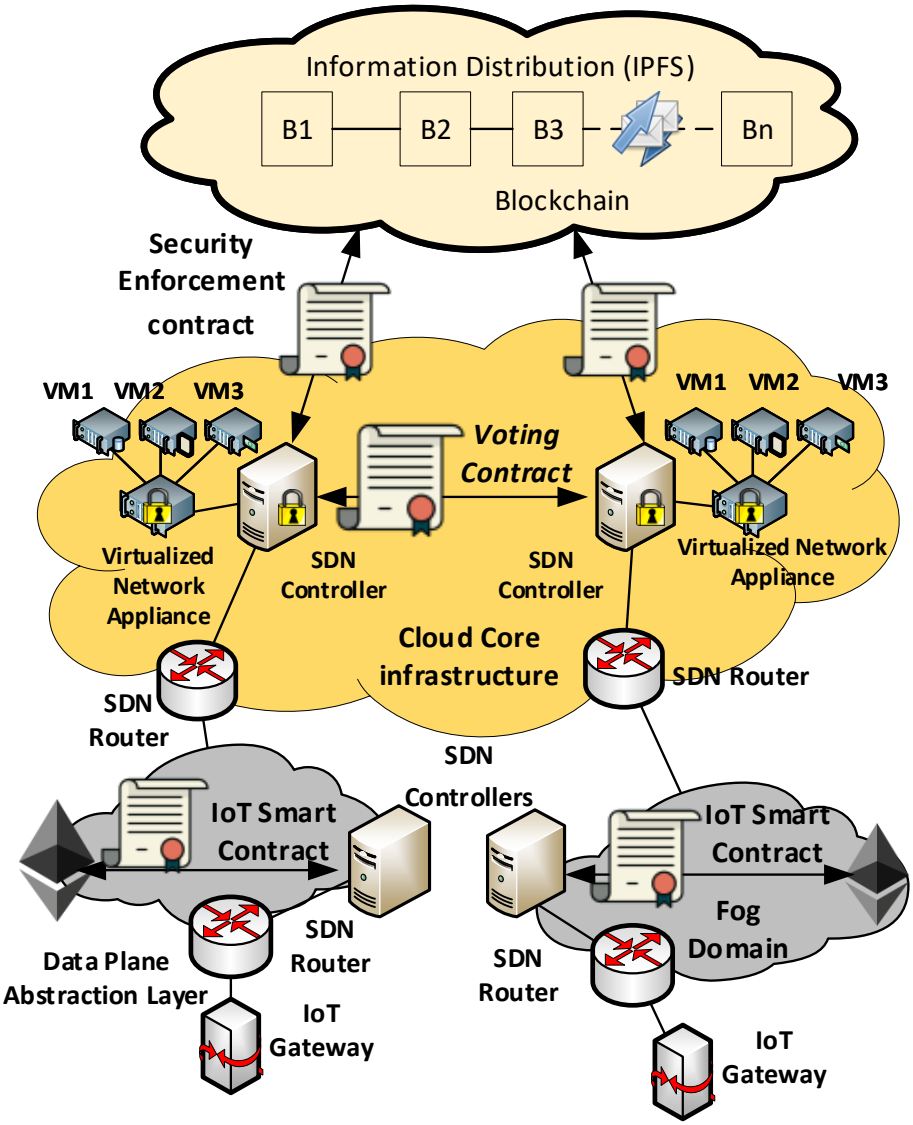


Fig. 1 Overview of the Blockchain-SDN IoT architecture.

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 (e.g., 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. The data plane layer gathers sensing data from IoT gateways, which interfaces 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 Flow Management

Figure 2 depicts flow management details at 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. An IoT node address in the blockchain is inferred from the node's own public keys. We apply Keccak-256 hash algorithm to the node's public key, take the last 20 bytes of the result and add the hexadecimal '0x' at the start of the address. Then, the address is associated with the IoT node's balances and used for sending and receiving transactions.

Furthermore, since each IoT node can be associated with more than one account, called Externally Owned Account (EOA), and has more than one address, these accounts/addresses can be used to identify peers IoT nodes in different IoT scenarios and use cases. Therefore, to further enforce the security of our architecture, our framework implements another module for 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 prevent impersonation, 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 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 also signs and verifies IoT transactions across distributed IoT nodes. Leveraging SDN/NFV enforces the coordination of distributed IoT nodes and increases their performance

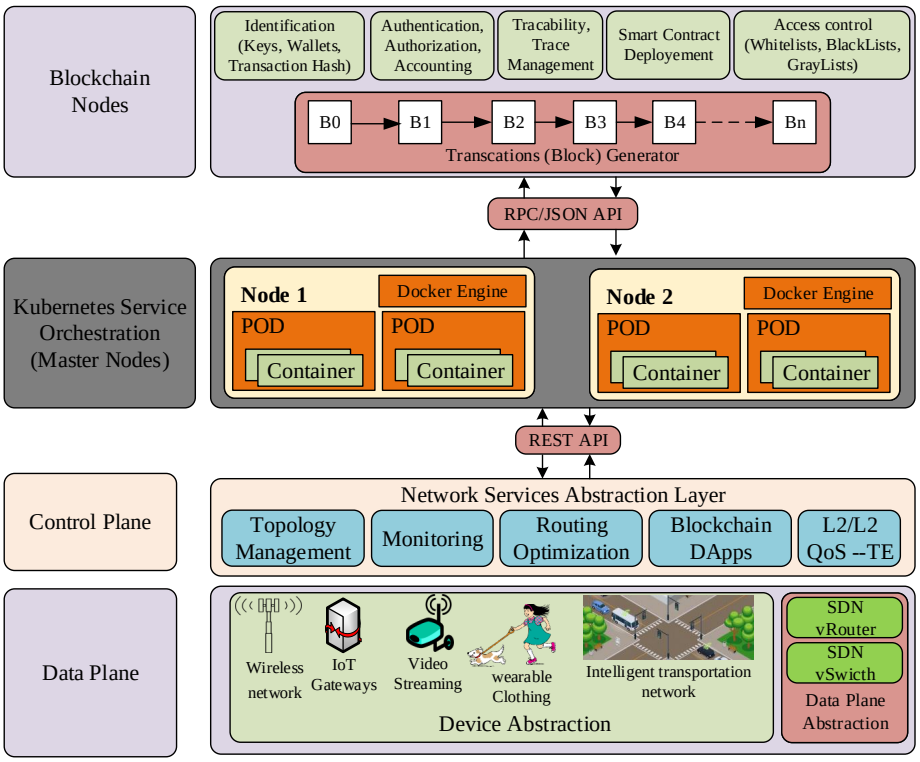


Fig. 2 Blockchain-SDN Applications Framework in Smart City Security.

by creating a modular architecture in which virtual miners can be hosted inside a NFV platform such as the Open Platform for NFV (OPNFV) [68]. 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 Design

Similar to regular transactions generated by massive IoT devices, smart contracts must be validated by distributed nodes. 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 DApps.

Algorithm 1 illustrates the contract deployment to provide a trustworthy mechanism to secure IoT transactions.

This algorithm describes how the SDN controller can enforce security of the network using function *setFlowRulesTurstList* in line 2. Any detected misbehavior is reported not

Algorithm 1: Deploy the Smart Contract.

```
   Input: Contract Address & ABI
   Output: List of available nodes
   // Deploys contract
 1 getInstance(class ContractInterface)
 2 deployContract(ABI, contractAddress)
   // Retreive IoT application by token
 3 nodes ← mycontract(nodeAddress).getAllApps()
 4 for i ← 1 to length(nodes) do
       // Get nodes' details
 5     nodesDetails ← mycontract(getAppsDetails(nodes[i])
       // generate a list of application details
 6     ListApps[apps[i]] ← nodesDetails
 7 end
   // Sending maching packets to SDN controller
 8 for j ← 1 to len(ListApps) do
 9     setFlowRulesTurstList(ListApps[apps[i]] )
10 end
   // Receive SDN controller messages
11 devices ← getAllDevices()
12 for i ← 1 to len (devices) do
       // Get IoT device info
13     deviceInfo ← mycontract(getIoTDevDetails(devices[i]))
       // Get service layer details
14     appDevInfo ← mycontract(getIoTDevApps(detailsIoTDev[1]))
       // List IoT devices by their Apps
15     listDevs[devices[i]] ← (IP, MAC, Apps)
16 end
17 for k ← 1 to length(listDevs) do
18     tmp ← ListApps[k]
         portsDev.Append(tmp["appProtocol"]"-"tmp["appPort"])
19 end
```

only based on its MAC 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 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 global network view including topology state and transactions meta-data. The SDN controller parses every OpenFlow packet exchanged between the IoT devices and the network to identify 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.

Specifically, the controller listens to the incoming IoT traffic and reports suspicious IP addresses before verifying unknown packets. Besides, intrusion detection VNFs (i.e., Firewall as a Service) are deployed inside Kubernetes clusters to take care of malicious flows and enable on-demand DDoS detection and mitigation. The SDN controller triggers storing decisions to VNF instances to maintain reports about white-listed and blacklisted IP addresses. Routing SDN packets among these VNF components is completely managed and controlled by the SDN controller using the *pipework* and the *overlay* mode of Open vSwitch software router. The former allows connecting multiple containers in arbitrarily complex scenarios. The later provides a form of private IP addresses that are only valid internally. Each IP address P identifies a service deployment in a separate chain, so that the SDN controller can program the flow table with the required flow entries F_P to define the following component $B = F_P(A)$ in the chain for which the traffic will be forwarded. The controller creates, for each flow entry F_P, the forwarding table entries that match received packets against the forwarding ports A they should follow with P as the destination address. 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.

Algorithm 2 shows how the SDN control plane prepares two types of lists, i.e., blacklist and whitelist. The blacklist includes suspicious IoT devices with abnormal behaviour such as an attack or an unexpected behavior. The controller uses this list to isolate these devices from sending traffic to the blockchain. The function *servergateway()* is called when an overwhelmed node must be removed the network. The whitelist includes users or devices whose behavior is normal and can continue delivering their content as long as they belong to the blockchain. Line 21 of Algorithm 2 shows that the SDN controller continuously updates the list of whitelisted nodes and establishes a topology graph including discovered trustworthy IoT nodes.

3.4 Consensus Algorithm

We rely on the Proof-of-Authority (PoA) consensus algorithm to select a set of N trusted nodes called the authorities. To enforce 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 more hardware resources (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. The advantages of this approach are twofold: first, it helps in keeping the decentralization more efficient while requiring less computational power.

Second, by relying on a group of authority nodes that are pre-approved validators to verify transactions and build blocks, we ensure that nodes wishing to become authorities and validators 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 a series of rules to be considered trustworthy. For example, they should be elected by at least 51% of existing ones. Figure 3 depicts block creation and validation using the POA mechanism. The time is divided into steps, each of which has

Algorithm 2: BLACKLISTING AND WHITELISTING OF IOT NODES

```
   Input: List IoT devices by Apps
   Output: Update trusted IoT apps by protocol:ports
   // Update list of protocol:ports
 1 results ← list(portsDev) ;
 2 if length(results) ≠ 0 then
 3     for i ← 1 to length(results) do
 4         keys ← results[i] ;
 5         for k ← 1to length(ListApps) do
 6             p ← ListApps[k] ;
 7             if key["appPort"] = p[1] then
                   // Port added
 8                 file.write(p[1] + "," + key["time"]") ;
 9         end
10     end
11 for i ← 1 length(ListDevs) do
12     src_mac ← ListDevs[i] ;
13     src_ip ← key['IP'] ;
14     myapp ← key['myApp'] ;
15     Assert(key in ListDevs) ;
16     for i ← 1 to length(myapp) do
           // Reputation preprocessor to whitelist trusted IoT
              nodes
17         if myapp[i] in ListApps then
18             dst_ip ← ListApps[myapp[i]]['appIPaddr'];
19             dst_port ← ListApps[myapp[i]]['appPort'];
20             proto ← ListApps[myapp[i]]['appProtocol'];
21             source ← ListMAc[key]['dpid'] - ListMAc[key]['dpid'] ;
22             dest ← servergateway;
               // Create SDN topology graph
23             create_graph (source, servergateway, src_mac, scr_ip, dst_ip,
                 dst_port, proto)
24     end
25 end
```

an authority elected as mining leader. In this example, there are 3 authorities with their IDs equal to 1, 2 and 3. The leader of the first step is node 1, then 2, and 3. The leader of a step S is the authority identified by the id $l = SmodN$; where S is the number of steps, and N is the number of authority nodes.

Third, validating IoT transactions relies on a mining rotation schema to fairly distribute the responsibility of block creation among authorities. Authorities are assumed to be asynchronous and all of them are allowed to propose blocks in each computation step. The current step is calculated based on a formula that combines the block number

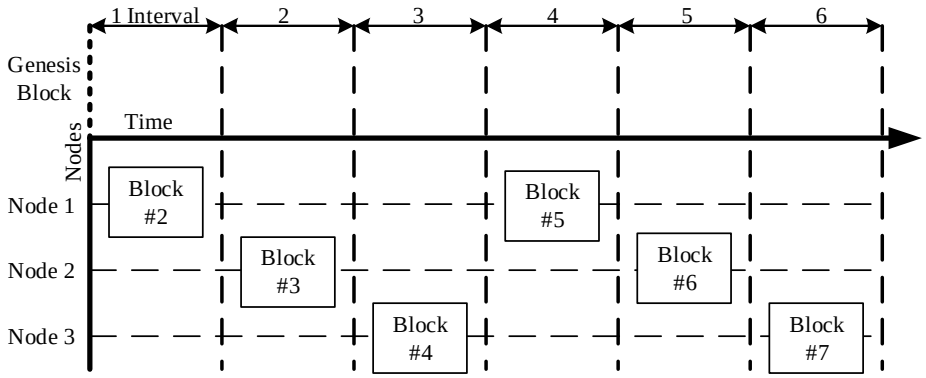


Fig. 3 Block creation in PoA Consensus Algorithm

and the number of authorities. To prevent an authority from monopolizing the network resources (e.g., proposing a block when it is not allowed), each authority node is only allowed to propose a block every $N/2+1$ blocks. That is, at any point of time a maximum number of $N-(N/2+1)$ authorities allowed to propose a block. If an authority node acts maliciously, it can be voted out and removed by other nodes from the list of legitimate authorities if a majority is reached.

4 Use cases

4.1 Blockchain-SDN enabled Internet of Vehicles

Figure 4 depicts an Internet of Vehicles (IoV) scenario, where distributed networks interconnect various IoT systems, such as connected cars, pedestrians, roads, and parking systems. IoV is envisaged to improve the safety of vehicles and are foreseen to use 5G mobile networks.

Figure 4 shows how SDN can address the challenges related to frequent node topology changes, high node mobility, and dynamic topology changes caused by frequent node disconnections. Specifically, SDN controllers exploit information obtained from Road Side Units (RSUs) to find optimal paths to connected vehicles and route messages across shortest paths within the VANET. SDN can also extend RSU coverage by coordinating their communication with other RSUs and with neighbor wireless access points. The SDN controller collects routing information from the VANET nodes to create a global view map of the connected vehicles and handle various topological changes in the VANET. Furthermore, combined with NFV, the controller significantly improves scalability, performance and Quality of Service (QoS). Specifically, SDN/NFV enable the generation of flow rules to support dynamic resource allocation, network slicing and orchestration, and mobility management. RSUs parse SDN packets received from the controller layer to decide the actions to perform for packet forwarding either to connected

vehicles or push them down to other RSUs. It allows them to exchange their information more efficiently with infrastructures using Vehicular Ad Hoc Networks (VANETs).

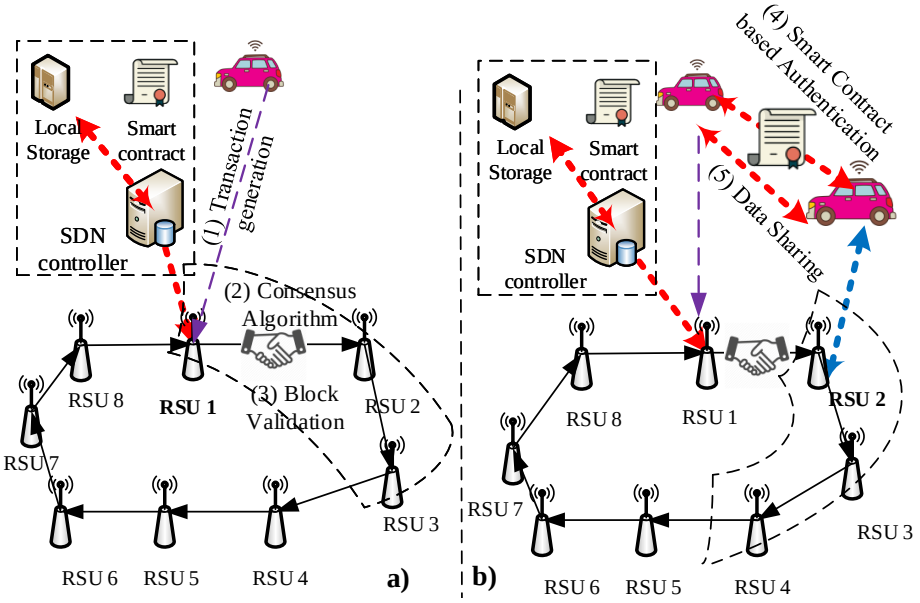


Fig. 4 Secure message dissemination in SDN-enabled VANET

Additionally, Blockchain distributed ledgers coupled with consensus mechanisms can guarantee the preservation of trustworthy data. Figure 4 shows two consecutive steps and how current leader and authorities allowed proposing blocks change. There are $N = 8$ authorities (i.e., RSU1 to RSU8), hence $N - (N/2+1) = 3$ authorities allowed to propose a block at each step, with one of them acting as leader (the bold node in Figure 4). In the first time step in Figure 4(a), RSU1 is the leader while RSU2 and RSU3 are allowed to propose blocks. Next, in next time step as depicted in Figure 4(b), RSU1 is not allowed anymore to propose a block (it was in the previous step, so it has to wait $(N/2)+1$ steps), while RSU4 is now authorized to propose a new block and RSU2 is the current leader.

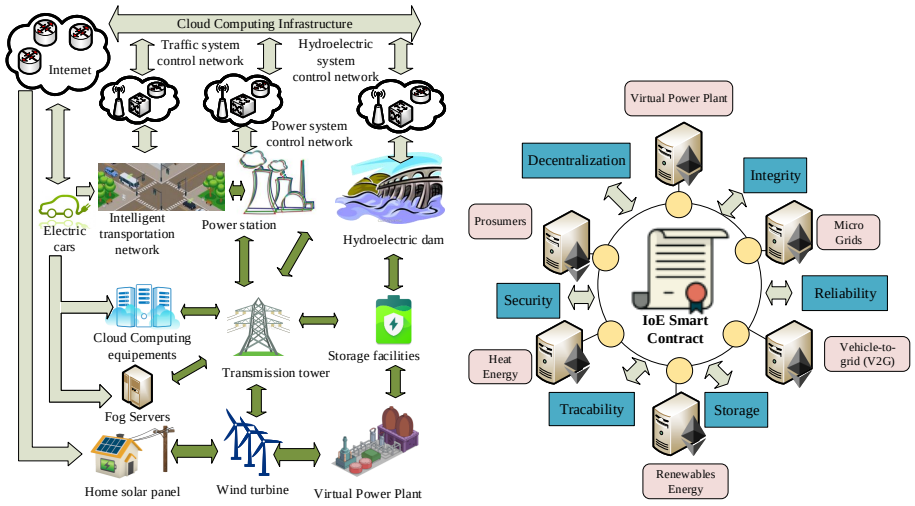
The combination of SDN and Blockchain can effectively and efficiently manage and control the operations of VANET systems. Blockchain distributed ledgers record transactions generated in VANET nodes and maintain these records in a transparent, immutable and secure infrastructure. RSUs nodes can be pre-selected to create blocks and perform lightweight mining. For example, a voting process can be established between these pre-qualified nodes to validate transactions and verify the correctness of exchanged blocks. Various messages exchanged between RSUs can be recorded as the evidence of the trustworthiness of received data. In such an approach, falsified transactions can be easily detected by the shortlisted cluster of VANET nodes and decisions can be provided to sender nodes to report any detected intrusion. Thus,

Blockchain can handle blocks concurrently with SDN to ensure an efficient, agile and flexible network management while preventing malicious activities.

Besides, as modern vehicles contain hundred Electronic Control Units (ECUs) to make them able to support Vehicle-to-Everything (V2X) communication technologies and connect to external infrastructure, connected vehicles encounter several security and privacy concerns. Since VANETs do not rely on a third-party security server, malicious nodes from the network or compromised OnBoard Units (OBU) can cause security vulnerabilities, such as jamming, eavesdropping, and tampering, and overwhelm the network with malicious traffic. In this context, Blockchain distributed ledgers, coupled with consensus mechanisms, can guarantee the preservation of trustworthy data.

4.2 Internet of Energy (IoE)

Internet of Energy (IoE) [10] has recently received significant attention from energy manufacturers and producers to reduce fossil energy and address environmental pollution concerns [33]. At its core, IoE is a peer-to-peer, distributed, interconnected, open and intelligent network architecture for upgrading and automating electricity infrastructure, including renewable energy, energy harvesting devices, micro-grids, and Virtual Power Plant (VPP). In particular, VPP is a cloud native network architecture that embraces message brokers, network virtualization, and edge computing with edge telemetry and command/control of edge devices to aggregate the capacities of heterogeneous distributed energy resources.



(a) Traditional Virtual Power Plant (b) Blockchain SDN-enabled IoE

Fig. 5 Internet of Energy: Traditional Model vs Blockchain-enabled Model

Figure 5 depicts the current traditional IoE model (i.e., Figure 5(a)) and Blockchain-enabled model supported by our approach (i.e., Figure 5(b)). Our Blockchain-enabled model has the potential to meet flexibility, security, resiliency, and flow isolation requirements of the IoE networks [29]. The computer nodes in Figure 5(b) describe distributed ledgers that mediate data dissemination between IoE stakeholders [49]. For example, in modern VPPs the ledgers hold a SDN control layer to offer fast flow rerouting and low latency capabilities, build resilient network communication, and abstract and integrate the distributed utility resources. Similar to mobile cellular operators, this approach enables supporting different virtual energy operators and power traders that share a decentralized cloud-native architecture of the power grid. Specifically, from traditional main heat and nuclear plants, modern VPPs interconnect decentralised energy production units of renewable sources such as biogas, wind, and solar with commercial and industrial power consumers and power-storage systems. The decentralized aspect of Blockchain fits in this universal vision of IoE by enabling the participation of connected and distributed electricity infrastructures in validating transactions [39]. Concretely, blockchain enables peer-to-peer energy transactions between different electric utilities without the need for a third-party [37]. Energy transactions are publicly validated by all participants to enable secure energy trading. Thus, blockchain addresses not only the challenging issues of energy trading in the above use case, but also satisfies data integrity and ensures data security and privacy-preservation of number of use cases such as decentralised marketplaces, electric vehicle charging and e-mobility [5] .

4.3 Improving Security of IoT Gateways

Blockchain can improve SDN-enabled IoT gateways in a dynamic fashion. First, adopting distributed ledgers can enforce the trust in IoT networks by enabling security mechanisms between IoT gateways and IoT services distributed among Fog nodes at the network edge. Figure 6 shows how IoT gateways can be connected to our SDN controller using our developed Blockchain IoT service layer. The SDN controller implements a Python-based decentralized application that integrates with Ethereum Web3 API to filter the traffic and detect suspicious IoT nodes. It provides a collaborative mechanism for whitelisting or blacklisting suspicious IoT gateway IP addresses. Our approach delegates storing blacklisted and white listed IP addresses to VNFs. VNF instances can be dynamically deployed to meet changing conditions and accommodate to higher traffic demand or more stringent service requirements.

That is, our approach enhances scalability, flexibility, agility, resiliency, and dynamic resource management and enforces trust on the IoT-on-the-blockchain network. Additionally, it enables new types of trust-less interactions for empowering IoT communications and brings more transparency and performance by reducing deep packet inspection of SDN-enabled IoT traffic. Specifically, securing individual IoT devices becomes a challenging issue because their limited processing, memory, power and defense capabilities and resources. The symbiosis of distributed blockchain ledger and SDN/NFV enforces the coordination of individual IoT devices in destroying large-scale botnets. Decentralized blockchain security functions for IoT are deployed in a form of

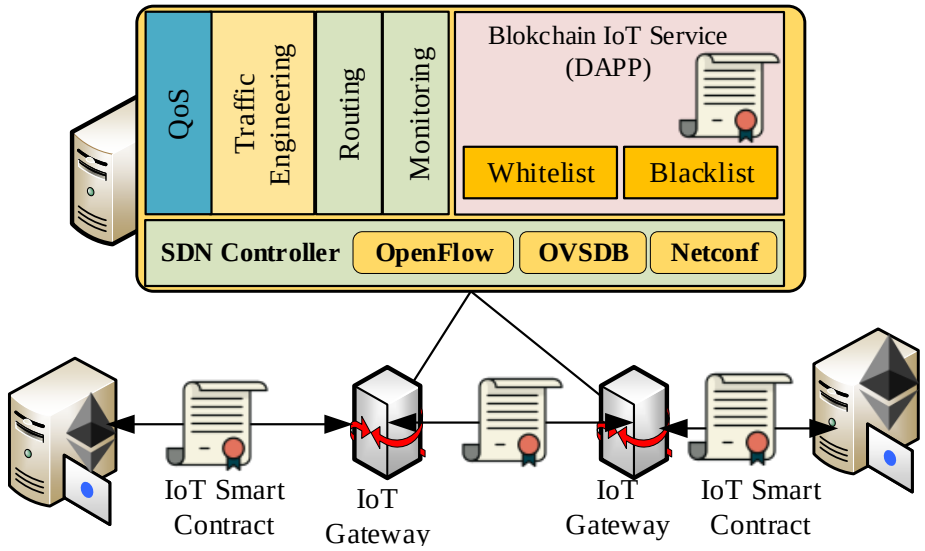


Fig. 6 Ensuring Security and Interoperability between IoT Gateways.

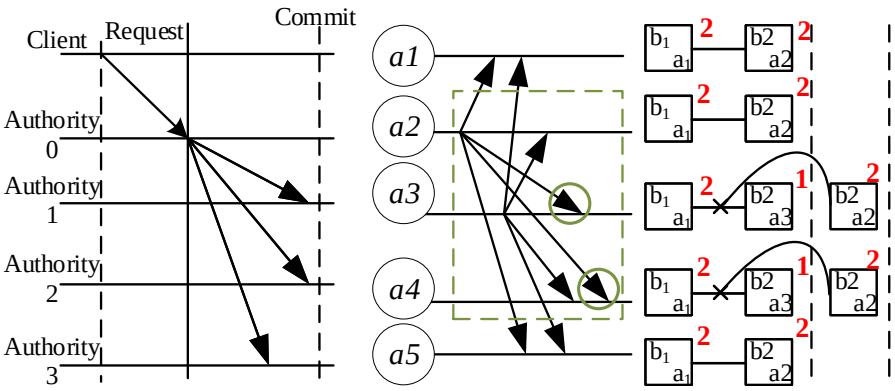
firewall VNF appliances (i.e. micro-services running inside lightweight containers) to mitigate malicious traffic.

5 Performance Evaluation

In this section, we qualitatively evaluate the performance and scalability of our approach to determine the effectiveness and fitness of the consensus algorithm. The performance refers to transaction latency and throughput. An IoT transaction is not considered valid until it is committed out to the blockchain. Performance is bounded by a combination of block interval – time between publishing subsequent blocks – and block size. These parameters establish an upper bound on transaction throughput. We define scalability as the ability of the blockchain network to improve or degrade the workload versus the number of nodes.

We implemented a prototype including 20 nodes that act as blockchain miners, where each node runs our leader election consensus algorithm. Then we compared our solution against three well-known consensus algorithms [69], i.e., Proof of Work (PoW), Proof of Elapsed Time (PoET), and Proof of Stake (PoS). The POW consensus algorithm involves solving cryptographically hard mathematical puzzles by using miners computational resources. PoS avoids using complex and unnecessary calculations used by the PoW. Instead of miners, there are validators that their own resources as pledge to become candidates to create and validate blocks. The PoET is a random leader election consensus introduced by Intel in which a separate random timer that operates independently at every node to spread the chances of winning equally across network

participants. This randomization gives every single node the same chance of likely to be the winner.



(a) PoA Message exchanges (b) Fork occurring in PoA

Fig. 7 Latency during Message Exchange in PoA

Figure 7(a) shows a leader node (Authority 0) receiving a client request to validate an IoT transaction block. The leader then broadcasts the block to a group of pre-approved authority nodes (authority nodes 1, 2 and 3) to validate IoT transactions and commit it to the blockchain. In terms of transaction's latency, our election-based IoT node validation process shows lower latency compared against the other consensus algorithms. Specifically, our solution approach requires only one round to validate and commit a new block to the blockchain. Because our approach relies on PoA, which is a communication-oriented consensus mechanism that does not involve extensive computation, it assumes bounded latency expressed in terms of time steps. The block is committed at once, hence the latency in terms of the number of message rounds is ones.

Figure 7(b) illustrates a specific scenario in which a leader node a_2 broadcasts a new block b_1 to the blockchain and at the same time another no-leader authority node a_3 broadcasts also another block b_2. The first new created block b_1, which precedes the block b_2, reaches nodes a_1 and a_5 before b_2 arrives to these nodes. However, b_2 reaches nodes a_3 and a_4 before they receive the first created block b_1. The right side of Figure 7(b) shows a fork operation performed by each node in the blockchain when blocks in different miners become misaligned and the network becomes desynchronized. Authority nodes a_3 and a_4 decide to continue using block b_1 as the first block, reference it as a previously reacted block, and b_2 as the next arriving block.

Figure 8 depicts the message exchange performed by the Practical Byzantine Fault Tolerance (PBFT) voting-based consensus used in the PoW algorithm. The PoW consensus algorithm, 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. Additionally, in PoW all unverified IoT transactions are put together in a poll, then all miners work to check that those transactions are legitimate by solving a complex mathematical puzzle. Thus, the PoW consensus algorithm is the

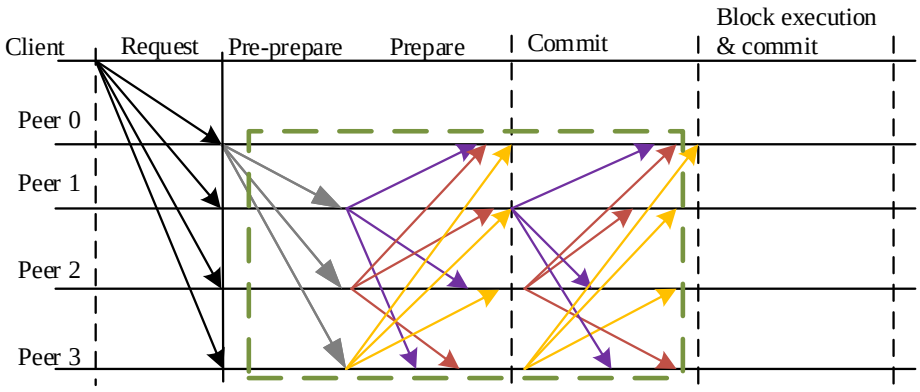


Fig. 8 Latency during Message Exchange in voting-based PBFT

most reliable and secure among the three algorithms. However, it has scalability issues because the block size is very small to sustain thousands of transactions, which limits the throughput performance of PoW.

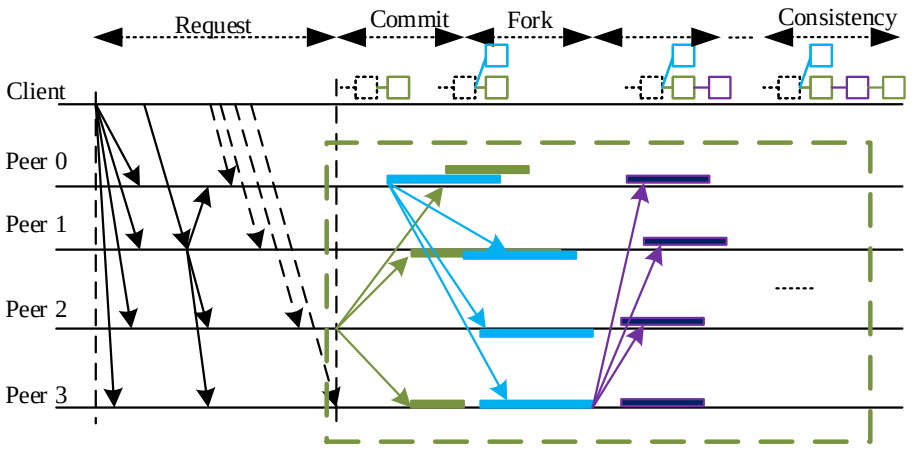


Fig. 9 Latency for Message Exchange in lottery-based PoET

Finally, Figure 9 illustrates the message exchange for IoT transaction's flow using the PoET consensus algorithm. The PoET lottery-based style of consensus algorithms needs five message rounds – Request, Commit, Fork, Resolve, and Consistency – to validate a block.

Our approach achieves an average latency of 30ms for validating an IoT transaction. The voting-based PoW consensus achieved an average latency of 10 minutes in Bitcoin blockchain to validate a transaction. The PoS consensus validates a new IoT transaction in average time delay of 12 seconds in Ethereum blockchain. The PoET algorithm

commits a transaction within 25 seconds and needs additional 10 seconds to validate it. Therefore, our approach outperforms these approaches and archives better transaction throughput.

6 Open Challenges and Directions for Future Work

Blockchain is a nascent technology with a set of complex and open issues that should be addressed to develop infrastructure, agreements and safety mechanisms to overcome them. This section delves into the key technical and government challenges and discusses some initial solutions to address them. These issues then become the key directions for conducting future research.

6.1 Scalability issues

Despite the fact that Blockchain provides high-performance, transparent, immutable, and cryptographically verifiable transactions, scalability is still a significant concern to support low-latency communication in ultra-dense IoT systems [50]. Indeed, current Blockchain platforms may take seconds (e.g., Ethereum) or minutes (e.g., Bitcoin) for a transaction to be included in a block [62]. The main reason for this slow speed is that every block validation and computation should be performed at every Blockchain node in the network, which increases the computational power and the transaction confirmation time of the entire network to the processing capability of a single node. Smart city devices, such as Fog servers and IoT gateways have limited processing, memory, and computation capabilities and currently cannot be used as miners to process consensus algorithms. On the other hand, for large amounts of transfer, it is recommended to wait more time (about an hour) to confirm the transaction and verify its integrity because it must outweigh the cost of a double spend attack, while the same processing takes seconds at most with IoT gateways.

6.2 Storage

Blockchain nodes typically store transitions and blocks in text and meta-data files, where block size depends on the synchronization mode activated at each involved participant node. For example, for an IoT node using the "*fast sync*" mode in Ethereum a complete copy of the Blockchain size is closer to 50 GB and this amount tends to grow by 14 GB each year. Although reducing the block size could improve storage, it however could impact their scalability. Scaling the blockchain is not as simple as increasing the block size or accelerating block generation. Specifically, although a larger block can store more transactions, raise the throughput and make the blockchain more efficiently accessed, it, however, increase the block propagation time and rise the node storage

requirements which may lead to centralization of nodes based on the higher cost of storage equipment's.

Therefore, there is an urgent need to rethink innovative compression algorithms to make blockchain storage scalable perhaps by inspiring approaches from modern big data and data-intensive science advances. Another promising approach could be by improving data accessibility through the use of novel and advanced APIs that facilitate automated calls and notification, to notify user applications every time transactions or blocks are created on the network.

6.3 Privacy Leakage

Double spending 51% attack could become a security challenge in blockchain as one mining entity could grab control of the overall blockchain. This issue stems from the fact that centralizing the mining power into only few large mining pools who control the majority of the transaction recording. Another important security issue concerns the vulnerability of smart contracts themselves. A smart contract is an autonomous software code running across distributed nodes that agree to its content. Attackers can unleash a malicious intermediate contract and invoke it repeatedly to perform memory-intensive work, thereby dramatically slowing down the overall blockchain network. This could happen when smart contracts are poorly designed and suffer from reinjection vulnerability.

A possible alternative to overcome this limitation is to make the design of smart contracts more difficult for bugs to appear. This will need a formal verifiable language and model checking tools for exploring all possible states and transitions in the smart contract models, which in turn requires domain-specific abstraction techniques to perform formal verification.

7 Potential Future Research Opportunities

7.1 Off-chaining Models

To meet the scalability needs of future IoT networks using blockchain, off-chaining computations could be a promising approach. Computationally heavy tasks such as state transition functions are executed on off-chain nodes while only the state's outcomes are stored on the blockchain. Heavy computation can be delegated to another layer on top of the blockchain that performs heavy, compute-intensive work thereby using the blockchain resources effectively. Off-chaining computation could be centralized, distributed across a group of nodes, or outsourced in a side-chain. In order to succeed in such an approach, IoT nodes will not only retrieve the results and the proof of correctness for the outsourced operations, but also IoT devices will be able to verify the proof of correctness themselves without consuming substantial computing power. Off-chaining

model can help to scale up public main chain while limiting and isolating any damages to the private side chain to prevent the main chain from any dramatic damage.

7.2 Data Analytics

Big data analytics have been successfully applied to various smart city themes such as air quality, smart transportation, energy internet, and climate pattern analysis [57]. It also has been the successful driving force behind the evolution of various artificial intelligence, data mining, machine learning and statistical analysis-based solutions in smart cities [41]. Smart city big data are stored in silos within different IoT infrastructure including IoT gateways, Fog computing servers, and cloud computing. Integrating Big data with Blockchain can transform smart city by ensuring data integrity and supporting data auditability without the need for centralized third party auditors (TPAs) [65]. Additionally, when big data meets blockchain, it could help in tracking IoT transactions by creating incentive systems in the form of IoT node managers that delve into the interactions between IoT devices to uncover hidden interaction patterns among them. Specifically, Big data could reshape the data structure in smart city services as it covers a huge amount of data gathered from climate monitoring tools, intelligent transportation systems, connected self-driving vehicles, thereby making more predictable decisions.

Furthermore, combining Blockchain and big data will change the way smart city data can be consumed, which means data will be more structured, abundant and complete, making it more suitable for analytics. In particular, it avoids data fragmentation as all involved parties in IoT transactions have access to the same data and share the same and complete overview of these transactions from their creation to their finish, without needing access to multiple silo systems, while at the same time each involved party can manage and control its own data without any third party authority or centralized repository. For example, BigchainDB [1] is an open-source distributed storage system that builds on top of Big Data for deploying a decentralized Blockchain to avoid any hard limit on the transaction size. This is interesting for smart cities to disseminate trained machine learning models and statistical analysis solutions to all involved parties. Similarly, Amazon Quantum Ledger Database (QLDB) [2] builds on top of a new type of fully managed ledger database to provide complete and cryptographically verifiable history of IoT transactions, which provides multiple smart city stakeholders with full data lineage within a centralized and trusted entity.

7.3 Artificial Intelligence

In light of recent advances in blockchain technology, artificial intelligence (AI) could help in solving challenges such as energy consumption, scalability, security, privacy,

[1] https://github.com/bigchaindb/bigchaindb

[2] https://aws.amazon.com/qldb/

efficiency, and mining. DeepMind AI [13] has been proven to be very efficient in optimizing energy consumption as it consistently achieved a 40 percent reduction in the amount of energy used for cooling in Google data center. We believe that similar results could be achieved with blockchain as well [27]. AI could also help in producing decentralized intelligence (either on-chain for basic information or off-chain in case of extra attachments) by introducing decentralized learning system such as federated learning and supporting sharding techniques to make the system more efficient.

7.4 Smart Contracts

Smart contracts are software codes where bugs could exist and may be vulnerable to malicious activity. The DAO contract suffers from the reinjection vulnerability which was exploited by hackers to withdraw existing funds repeatedly. Thus, formal verification of smart contracts [59] [20] is important to detect any irregularities in its design and behavior. Model-Checking approaches [42], formally verifiable language [6], and formal verification tools [15] are necessary to verify that the smart contract implementation complies with its specification, verify its security properties, formalize it by a set of temporal logic propositions [64].

8 Conclusion

In this chapter, we introduced a novel architecture to bring network softwarization and virtualzation to the blockchain nodes and enable dynamic and on-demand service orchestration. This architecture also improves security and privacy-preservation in order to meet various requirements of scalability, performance, seamless distribution, and transparency. Besides, we proposed a novel consensus algorithm based on the proof of authority to deliver fast transactions by increasing the speed at which the authorities validate transactions. Thus, malicious transactions can be detected, signaled, and removed from the network. A decentralized application (DApp) detects malicious IoT nodes and triggers remote miner to invalidated IoT transactions by blacklisting their originating IP addresses. Our solution can be used in various scenarios and use cases such IoV, securing IoT gateways, and audit traceability, thereby reinventing the Blockchain.

Since Blockchain is a nascent technology, we argue that processing billions of transactions in few seconds is challenging for scenarios like IoV and smart cities. We believe that combining Blockchain Big Data Analytics, and Deep Reinforcement Learning could change the way how IoT data can be consumed, which means data will be more structured, abundant and complete, making it more suitable for analytics. Our future research directions will focus on big data analytics and Graph Neural Networks (GNN) to avoid IoT data fragmentation and developing a Federated Machine Learning (FedML) models to solve the issue of data ownership and privacy by training statistical security models inside Fog nodes, while keeping data samples localized inside Fog nodes.

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. Abbasi, A.G., Khan, Z.: Veidblock: Verifiable Identity Using Blockchain And Ledger In A Software Defined Network. In: Companion Proceedings of the10th International Conference on Utility and Cloud Computing. pp. 173–179. UCC '17 Companion (2017)
2. Abou El Houda, Z., Hafid, A.S., Khoukhi, L.: Cochain-sc: An intra- and inter-domain ddos mitigation scheme based on blockchain using sdn and smart contract. IEEE Access **7**, 98893–98907 (2019)
3. Alharbi, T.: Deployment of blockchain technology in software defined networks: A survey. IEEE Access **8**, 9146–9156 (2020)
4. Ali, M.S., Vecchio, M., Pincheira, M., Dolui, K., Antonelli, F., Rehmani, M.H.: Applications of blockchains in the internet of things: A comprehensive survey. IEEE Communications Surveys Tutorials pp. 1–1 (2018). https://doi.org/10.1109/COMST.2018.2886932
5. Andoni, M., Robu, V., Flynn, D., Abram, S., Geach, D., Jenkins, D., McCallum, P., Peacock, A.: Blockchain technology in the energy sector: A systematic review of challenges and opportunities. Renewable and Sustainable Energy Reviews **100**, 143–174 (2019)
6. Antonino, P., Roscoe, A.W.: Formalising and verifying smart contracts with solidifier: a bounded model checker for solidity. ArXiv **abs/2002.02710** (2020)
7. Benedict, S.: Serverless blockchain-enabled architecture for iot societal applications. IEEE Transactions on Computational Social Systems **7**(5), 1146–1158 (Oct 2020)
8. Bera, S., Misra, S., Vasilakos, A.V.: Software-defined networking for internet of things: A survey. IEEE Internet of Things Journal **4**(6), 1994–2008 (Dec 2017)
9. Buccafurri, F., Lax, G., Nicolazzo, S., Nocera, A.: Overcoming Limits Of Blockchain For IoT Applications. In: Proceedings of the 12th International Conference on Availability, Reliability and Security. pp. 26:1–26:6. ARES '17 (2017)
10. CAO, Y., LI, Q., TAN, Y., LI, Y., CHEN, Y., SHAO, X., ZOU, Y.: A comprehensive review of energy internet: basic concept, operation and planning methods, and research prospects. Journal of Modern Power Systems and Clean Energy **6**(3), 399–411 (May 2018)
11. Chen, J.: Devify: Decentralized Internet Of Things Software Framework For A Peer-to-peer And Interoperable Iot Device. In: In proceedings of the Workshop on Advances in IoT Architecture and Systems (Jun 2017)
12. Chen, J.: Flowchain: A Distributed Ledger Designed For Peer-to-peer IoT Networks And Real-time Data Transactions. In: In proceedings of the 2nd International Workshop on Linked Data and Distributed Ledgers (May 2017)
13. DeepMind: Deepmind ai reduces google data centre cooling bill by 40% (2016), https://deepmind.com/blog/article/deepmind-ai-reduces-google-data-centre-cooling-bill-40
14. Ellul, J., Pace, G.J.: Alkylvm: A Virtual Machine For Smart Contract Blockchain Connected Internet Of Things. In: 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS). pp. 1–4 (Feb 2018)
15. Frank, J., Aschermann, C., Holz, T.: ETHBMC: A bounded model checker for smart contracts. In: 29th USENIX Security Symposium (USENIX Security 20). pp. 2757–2774. USENIX Association (Aug 2020), https://www.usenix.org/conference/usenixsecurity20/presentation/frank

16. Gao, J., Obour Agyekum, K.O., Sifah, E.B., Acheampong, K.N., Xia, Q., Du, X., Guizani, M., Xia, H.: A blockchain-sdn-enabled internet of vehicles environment for fog computing and 5g networks. IEEE Internet of Things Journal **7**(5), 4278–4291 (2020)
17. Guo, S., Dai, Y., Guo, S., Qiu, X., Qi, F.: Blockchain meets edge computing: Stackelberg game and double auction based task offloading for mobile blockchain. IEEE Transactions on Vehicular Technology **69**(5), 5549–5561 (May 2020)
18. Guo, S., Dai, Y., Xu, S., Qiu, X., Qi, F.: Trusted cloud-edge network resource management: Drl-driven service function chain orchestration for iot. IEEE Internet of Things Journal **7**(7), 6010–6022 (July 2020)
19. Hahn, A., Singh, R., Liu, C.C., Chen, S.: Smart Contract-based Campus Demonstration Of Decentralized Transactive Energy Auctions. In: 2017 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT). pp. 1–5 (Apr 2017)
20. Hajdu, Á., Jovanovic, D., Cretu-Ciocarlie, G.F.: Formal specification and verification of solidity contracts with events. ArXiv **abs/2005.10382** (2020)
21. Hamdaoui, B., Alkalbani, M., Rayes, A., Zorba, N.: Iotshare: A blockchain-enabled iot resource sharing on-demand protocol for smart city situation-awareness applications. IEEE Internet of Things Journal **7**(10), 10548–10561 (Oct 2020)
22. Han, R., Gramoli, V., Xu, X.: Evaluating Blockchains For Iot. In: 2018 9th IFIP International Conference on New Technologies, Mobility and Security (NTMS). pp. 1–5 (Feb 2018)
23. Hari, A., Lakshman, T.V.: The internet blockchain: A distributed, tamper-resistant transaction framework for the internet. In: Proceedings of the 15th ACM Workshop on Hot Topics in Networks. pp. 204–210. HotNets '16, Atlanta, GA, USA (2016)
24. Ibba, S., Pinna, A., Seu, M., Pani, F.E.: Citysense: Blockchain-oriented Smart Cities. In: Proceedings of the XP2017 Scientific Workshops. pp. 1–5. XP '17 (2017)
25. Kang, J., Yu, R., Huang, X., Maharjan, S., Zhang, Y., Hossain, E.: Enabling Localized Peer-to-peer Electricity Trading Among Plug-in Hybrid Electric Vehicles Using Consortium Blockchains. IEEE Transactions on Industrial Informatics **13**(6), 3154–3164 (Dec 2017)
26. Kataoka, K., Gangwar, S., Podili, P.: Trust List: Internet-wide And Distributed Iot Traffic Management Using Blockchain And Sdn. In: 2018 IEEE 4th World Forum on Internet of Things (WF-IoT). pp. 296–301 (Feb 2018)
27. Li, Y., Wen, Y., Tao, D., Guan, K.: Transforming cooling optimization for green data center via deep reinforcement learning. IEEE Transactions on Cybernetics **50**(5), 2002–2013 (2020). https://doi.org/10.1109/TCYB.2019.2927410
28. Liu, H., Han, D., Li, D.: Fabric-iot: A blockchain-based access control system in iot. IEEE Access **8**, 18207–18218 (2020)
29. Lu, X., Shi, L., Chen, Z., Fan, X., Guan, Z., Du, X., Guizani, M.: Blockchain-based distributed energy trading in energy internet: An sdn approach. IEEE Access **7**, 173817–173826 (2019)
30. Lundqvist, T., de Blanche, A., Andersson, H.R.H.: Thing-to-thing Electricity Micro Payments Using Blockchain Technology. In: 2017 Global Internet of Things Summit (GIoTS). pp. 1–6 (Jun 2017)
31. Luo, J., Chen, Q., Yu, F.R., Tang, L.: Blockchain-enabled software-defined industrial internet of things with deep reinforcement learning. IEEE Internet of Things Journal **7**(6), 5466–5480 (June 2020)
32. Machado, C., Fröhlich, A.A.M.: Iot data integrity verification for cyber-physical systems using blockchain. In: 2018 IEEE 21st International Symposium on Real-Time Distributed Computing (ISORC). pp. 83–90 (May 2018). https://doi.org/10.1109/ISORC.2018.00019
33. Mahmud, K., Khan, B., Ravishankar, J., Ahmadi, A., Siano, P.: An internet of energy framework with distributed energy resources, prosumers and small-scale virtual power plants: An overview. Renewable and Sustainable Energy Reviews **127**, 109840 (2020). https://doi.org/10.1016/j.rser.2020.109840, http://www.sciencedirect.com/science/article/pii/S1364032120301349
34. Massonet, P., Deru, L., Achour, A., Dupont, S., Levin, A., Villari, M.: End-to-end security architecture for federated cloud and iot networks. In: 2017 IEEE International Conference on Smart Computing (SMARTCOMP). pp. 1–6 (May 2017)

35. Medhane, D.V., Sangaiah, A.K., Hossain, M.S., Muhammad, G., Wang, J.: Blockchain-enabled distributed security framework for next-generation iot: An edge cloud and software-defined network-integrated approach. IEEE Internet of Things Journal **7**(7), 6143–6149 (July 2020)
36. Mendiboure, L., Chalouf, M.A., Krief, F.: Towards a blockchain-based sd-iov for applications authentication and trust management. In: Internet of Vehicles. Technologies and Services Towards Smart City. pp. 265–277. Springer International Publishing (2018)
37. Miglani, A., Kumar, N., Chamola, V., Zeadally, S.: Blockchain for internet of energy management: Review, solutions, and challenges. Computer Communications **151**, 395–418 (2020)
38. Misra, S., Mukherjee, A., Roy, A., Saurabh, N., Rahulamathavan, Y., Rajarajan, M.: Blockchain at the edge: Performance of resource-constrained iot networks. IEEE Transactions on Parallel and Distributed Systems **32**(1), 174–183 (Jan 2021)
39. Mollah, M.B., Zhao, J., Niyato, D., Lam, K., Zhang, X., Ghias, A.M.Y.M., Koh, L.H., Yang, L.: Blockchain for future smart grid: A comprehensive survey. IEEE Internet of Things Journal pp. 1–1 (2020)
40. Novo, O.: Blockchain Meets Iot: An Architecture For Scalable Access Management In Iot. IEEE Internet of Things Journal **5**(2), 1184–1195 (Apr 2018)
41. Nuaimi, E.A., Neyadi, H.A., Nader, M., Al-Jaroodi, J.: Applications of big data to smart cities. Journal of Internet Services and Applications **6**(25), 1–15 (December 2015)
42. Osterland, T., Rose, T.: Model checking smart contracts for ethereum. Pervasive and Mobile Computing **63**, 101129 (2020)
43. Powell, C., Desiniotis, C., Dezfouli, B.: The fog development kit: A platform for the development and management of fog systems. IEEE Internet of Things Journal **7**(4), 3198–3213 (2020). https://doi.org/10.1109/JIOT.2020.2966405
44. Pradip Kumar Sharma and Jong Hyuk Park: Blockchain based hybrid network architecture for the smart city. Future Generation Computer Systems **86**, 650 – 655 (2018)
45. Qiu, C., Yu, F.R., Yao, H., Jiang, C., Xu, F., Zhao, C.: Blockchain-based software-defined industrial internet of things: A dueling deep q-learning approach. IEEE Internet of Things Journal pp. 1–1 (2018)
46. Qiu, C., Yu, F.R., Yao, H., Jiang, C., Xu, F., Zhao, C.: Blockchain-based software-defined industrial internet of things: A dueling deep Q -learning approach. IEEE Internet of Things Journal **6**(3), 4627–4639 (June 2019)
47. Rahman, A., Islam, M.J., Rahman, Z., Reza, M.M., Anwar, A., Mahmud, M.A.P., Nasir, M.K., Noor, R.M.: Distb-condo: Distributed blockchain-based iot-sdn model for smart condominium. IEEE Access **8**, 209594–209609 (2020)
48. Rahman, A., Nasir, M.K., Rahman, Z., Mosavi, A., S., S., Minaei-Bidgoli, B.: Distblockbuilding: A distributed blockchain-based sdn-iot network for smart building management. IEEE Access **8**, 140008–140018 (2020)
49. Rehmani, M.H., Davy, A., Jennings, B., Assi, C.: Software defined networks-based smart grid communication: A comprehensive survey. IEEE Communications Surveys Tutorials **21**(3), 2637–2670 (2019)
50. Reyna, A., Martín, C., Chen, J., Soler, E., Díaz, M.: On blockchain and its integration with iot. challenges and opportunities. Future Generation Computer Systems **88**, 173 – 190 (2018)
51. Rodrigues, B.B., Bocek, T., Lareida, A., Hausheer, D., Rafati, S., Stiller, B.: A Blockchain-based Architecture For Collaborative Ddos Mitigation With Smart Contracts. In: AIMS (2017)
52. Salahuddin, M.A., Al-Fuqaha, A., Guizani, M., Shuaib, K., Sallabi, F.: Softwarization Of Internet Of Things Infrastructure For Secure And Smart Healthcare. Computer **50**(7), 74–79 (2017)
53. Samaniego, M., Deters, R.: Using Blockchain To Push Software-defined Iot Components Onto Edge Hosts. In: Proceedings of the International Conference on Big Data and Advanced Wireless Technologies. pp. 58:1–58:9. BDAW '16 (2016)
54. Sharma, P.K., Singh, S., Jeong, Y.S., Park, J.H.: Distblocknet: A Distributed Blockchains-based Secure Sdn Architecture For Iot Networks. IEEE Communications Magazine **55**(9), 78–85 (2017)
55. Singh, M., Aujla, G.S., Singh, A., Kumar, N., Garg, S.: Deep-learning-based blockchain framework for secure software-defined industrial networks. IEEE Transactions on Industrial Informatics **17**(1), 606–616 (Jan 2021)
56. Singh, M., Kim, S.: Trust Bit: Reward-based Intelligent Vehicle Commination Using Blockchain Paper. In: 2018 IEEE 4th World Forum on Internet of Things (WF-IoT). pp. 62–67 (Feb 2018)

57. Soomro, K., Bhutta, M.N.M., Khan, Z., Tahir, M.A.: Smart city big data analytics: An advanced review. WIREs Data Mining and Knowledge Discovery **9**(5), e1319 (2019)
58. Steichen, M., Hommes, S., State, R.: Chainguard : A firewall for blockchain applications using sdn with openflow. In: 2017 Principles, Systems and Applications of IP Telecommunications (IPTComm). pp. 1–8 (Sept 2017)
59. Tolmach, P., Li, Y., Lin, S.W., Liu, Y., Li, Z.: A survey of smart contract formal specification and verification (2020)
60. Walker, M.A., Dubey, A., Laszka, A., Schmidt, D.C.: Platibart: A Platform For Transactive Iot Blockchain Applications With Repeatable Testing. In: Proceedings of the 4th Workshop on Middleware and Applications for the Internet of Things. pp. 17–22. M4IoT '17 (2017)
61. Yan, W., Zhang, N., Njilla, L.L., Zhang, X.: Pcbchain: Lightweight reconfigurable blockchain primitives for secure iot applications. IEEE Transactions on Very Large Scale Integration (VLSI) Systems **28**(10), 2196–2209 (Oct 2020)
62. Yasaweerasinghelage, R., Staples, M., Weber, I.: Predicting latency of blockchain-based systems using architectural modelling and simulation. In: 2017 IEEE International Conference on Software Architecture (ICSA). pp. 253–256 (2017)
63. Yin, S., Bao, J., Zhang, Y., Huang, X.: M2m Security Technology Of Cps Based On Blockchains. Symmetry **9**(9) (2017)
64. Yoo, J., Jung, Y., Shin, D., Bae, M., Jee, E.: Formal modeling and verification of a federated byzantine agreement algorithm for blockchain platforms. In: 2019 IEEE International Workshop on Blockchain Oriented Software Engineering (IWBOSE). pp. 11–21 (2019)
65. Yu, H., Yang, Z., Sinnott, R.O.: Decentralized big data auditing for smart city environments leveraging blockchain technology. IEEE Access **7**, 6288–6296 (2019)
66. Zhang, C., Hu, G., Chen, G., Sangaiah, A.K., Zhang, P., Yan, X., Jiang, W.: Towards a sdn-based integrated architecture for mitigating ip spoofing attack. IEEE Access **6**, 64–77 (2018)
67. Zhang, P., Walker, M.A., White, J., Schmidt, D.C., Lenz, G.: Metrics For Assessing Blockchain-based Healthcare Decentralized Apps. In: 2017 IEEE 19th International Conference on e-Health Networking, Applications and Services (Healthcom). pp. 1–4 (Oct 2017)
68. Zhang, T.: Nfv platform design: A survey. arXiv: Networking and Internet Architecture (2020)
69. Zhao, W., Yang, S., Luo, X.: On consensus in public blockchains. In: Proceedings of the 2019 International Conference on Blockchain Technology. p. 1–5. ICBCT 2019, Association for Computing Machinery, New York, NY, USA (2019)