Blockchain-based Authentication Mechanism for Edge Devices in Fog-enabled IoT Networks

Erukala Suresh Babu *, Aguru Aswani Devi *, Ilaiah Kavati *, B K N Srinivasarao †
*Department of Computer Science & Engineering, National Institute of Technology Warangal, India, 506001
†Department of Electronics & Communication Engineering, National Institute of Technology Warangal, India, 506001
esbabu@nitw.ac.in *, aa720086@student.nitw.ac.in *, ilaiahkavati@nitw.ac.in *, srinu.bkn@nitw.ac.in †

***Abstract*—The deployment of fog computing paradigms for securing IoT networks is associated with several advantages, including reduced bandwidth, latency, storage, and computational overhead at cloud servers. However, the fog layer has additional security requirements, such as the establishment of secure channels for key distribution and the overhead of repetitive device authentications. In this paper, we have addressed these issues using a permissioned blockchain-based fog network that validates the edge devices through smart contracts by establishing a mechanism for secure storage and exchange of credentials. The communication between edge devices is carried out through the MQTT protocol, and device registration is performed using a smart contract. The key pairs are generated from the secp256k1 elliptic curve to ensure faster trust-based identity management and authentication of edge devices and gateways. The proposed frameworks ensures that the access to Blockchain network is given exclusively for authenticated devices The implementation of the proposed scheme is performed on private Ethereum 2.0, and the performance is evaluated in terms of node registration time, authentication time, key generation time, and throughput. The implementation results are available on GitHub(https://github.com/Aswani08/tenconresults.git).**

***Index Terms*—Fog network, Edge computing, Blockchain, IoT, Security, MQTT**

I. Introduction

The Revolution of Industrial 4.0 is all around the Ambient Intelligence that enables leading disruptive technologies such as the Internet of Things (IoT), Big-Data, Cloud Computing, 5G, and Artificial Intelligence. Recently, the IoT has been facilitated as a technology integrator and provided intelligent connections of Things, Data, Processes, and People for efficient decision-making without human intervention. Furthermore, the IoT/Edge devices have been deployed at a massive scale in various sectors like the Healthcare industry, Supply chain industry, Smart home systems, Smart city applications, home security systems, and so on. Already 50 billion devices are connected to the internet, increasing exponentially in the near future. However, in traditional Information Technology (IT) compute architecture with IoT requirements, the data processing cannot perform real-time data analysis and feedback when Operational Technologies (OT) devices interact with the IT side. IoT devices will generate data faster, and processing will be done at the IT side (Cloud Environment), which cannot ingest it. Cloud computing has several evolving challenges for IoT networks that cannot handle too much OT data with physical storage shortage, High Latency, increased transfer cost, lacking context awareness; Resiliency is Impractical, etc. Hence, the fog network was introduced as a new computing paradigm that brings the compute power close to the edge devices. Fog computing uses IoT devices known as fog devices and edge devices. The edge devices capture the data that hasn't been processed locally due to their resource-constrained memory, processing, and limited physical security. Consequently, this data is transported to a fog device near the data source and processed locally, filtered, evaluated, and transmitted to the cloud for long-term storage. In other words, the fog network acts as a management layer between the edge devices and the cloud layer. The most significant benefit of fog computing is that it extends the services of the cloud, reduces latency, provides scalability, and improves the efficiency of computing resources and structure. However, the IoT-Fog environment is more vulnerable to various security breaches [17], if not handled properly (1) Malicious edge devices, if not appropriately authenticated, can flood the network (fog devices) with irrelevant requests resulting in denial-of-service attacks. (2) All the aggregated IoT/edge device data goes through the fog node. The data stored at the fog node can tamper if not stored securely, posing a threat to confidentiality. (3) Replay attacks include sending the same request to the receiver as an authenticated user. (4) Man-in-the-middle attacks (MITM) threaten confidentiality and integrity if data is not communicated securely. The traditional authentication schemes presented in [14,19,23], similar to password-based, OAuth, etc., do not provide robust handling techniques. Hence, to prevent these challenging threats, we proposed a blockchain system to secure storage supply credentials-creating a distributed and trusted identity management of the edge nodes, securing data communications, and avoiding the single point of failure vulnerability inherited in a centralized system.

A. Motivation

The Fog layer nodes can process a large amount of data, including sensitive information, bringing several advantages to the IoT-Cloud Architecture. However, the inherited distributed nature of the fog computing applications results in several security concerns. Amon all, the end user authentication and privacy are the most significant issues. Nevertheless, The Fog nodes can plausibly exploit the properties of decentralized, secure, tamper-proof blockchain technology [12], [18]. Moreover, IoT devices are heterogeneous, scalable and interoperable [20]. But these devices are resource-constrained in terms of their memory, battery energy volume, and computational power and are prone to various cyber-attacks. Therefore, motivated by the above challenges, we explore an efficient solution with improved IoT/Edge-Fog environment architecture.

B. Our Contributions

This paper proposes a blockchain-based fog network for secure communication among IoT-enabled edge devices.

- We initially proposed an improved IoT architecture that empowers the blockchain-based fog network, ensures decentralization, and is highly scalable. The fog nodes maintain the blockchain system and hold a common smart contract to validate edge devices. The participating fog nodes act as a validator, maintain a shared ledger/blockchain and validate the request of the edge nodes.
- Identify and authenticate the fog, gateway, and edge nodes using a permissioned blockchain system. The network participants must register as edge devices, gateway, and fog nodes. Only those authenticated against the registered details in the blockchain will have access to the network services.
- To secure the IoT-enabled fog network, we used an Elliptic curve-based digital signatures scheme that generates the key pairs derived from elliptic curves, which are shorter and faster. Each edge node has a key pair derived from the secp256k1 elliptic curve.

The rest of the paper is organized as follows. Section 2 presents the related work that provides the basis for the proposed work. In section 3, we first provide the system model, description of our proposed system and this section ends with the authentication of edge devices in pBCT network. The implementation and results were analysed in Section 5. Finally, conclude our article with concluding remarks in Section 6.

TABLE I
SUMMARY OF RESEARCH GAPS

Ref.	Problem Addressed	Limitations
[1]	No central entity involved, mutual authentication	Each device is assumed to be a node in the blockchain even though IoT devices are resource-constrained
[2]	Comparison between MQTT and blockchain for data transfer	The device would have to store the smart contract functions, which is a storage overhead
[3]	Device-Gateway-Smart contract architecture based on Hyperledger fabric platform	Mutual authentication between device and gateway is not present and explicitly shown as a mechanism
[4]	Fog nodes form a blockchain and act as validators	No specific mechanism or security analysis is described for authenticity, confidentiality, etc.
[5]	HMAC and blockchain-based authentication and support for multiple authentication methods	Heterogeneity of devices introduces complexity overhead
[6]	Token-based authentication based on fog computing and blockchain	No actual implementation for communication between device, gateway, or controller. In addition, the central entity might lead single point of failure
[7]	Central authority to register and authenticate using hashed messages	Identities sent over a public channel and hence prone to attacks
[8]	The token-based approach based on the Ethereum platform - Remix IDE	Evaluated was restricted to registration phase, addition and removal of fog devices
[9]	Registration credentials are stored at the cloud level, and authentication is done at the fog level	Dependence on the cloud makes it partially centralized. Moreover, no simulation described
[10]	Blockchain-based authentication model for fog devices while considering their dynamic nature.	Location-based validation, an overhead

II. Related Work

This section presents a detailed analysis of the existing works on Blockchain-based authentication techniques. Li et al. [1] and Babu et al. [7] have presented the mutual authentication schemes using a blockchain network. An MQTT-based Blockchain authentication is proposed by Fakhri et al. [2]. A blockchain-based identity management system was presented by BANOUN et al. [3]. A secure data transfer scheme between the fog nodes is proposed by Priyadarshini et al. [4]. A fog-based authentication using Blockchain is presented by Imineet al. [9], Patwary et al. [10] and Fayad et al. [5]. A token-based authentication for IoT devices was studied by Babu et al. [6]. A smart contract-based mutual authentication is presented by Almadhoun et al. [8]. The Table. I presents the addressed problems and limitations of state-of-the-art mechanisms.

III. Proposed Work

This section discusses the details of the proposed solution. The proposed work aim is to show a proof of concept to secure the fog-based IoT network with device-to-device communication using a blockchain system. This device-to-device communication exchanges messages between the edge devices, edge devices to the gateway, and edge devices to fog devices via gateway devices. The gateway devices will act as interfaces to communicate between the edge devices and fog nodes. These gateway devices simplify the process like registration, authentication, communication, storing data on the blockchain, retrieving identity data from the blockchain using the smart contract, and relieving the device of holding smart contract functions. It also facilitates localized authentication for the IoT devices using the data stored on the blockchain to reduce latency and increase throughput. The main objective of the proposed framework is to avoid the security breaches such as (1) Malicious edge devices, if not appropriately authenticated, can flood the network (fog devices) with irrelevant requests resulting in denial-of-service attacks. (2) All the aggregated IoT/edge device data goes through the fog node. The data stored at the fog node can tamper if not stored securely, posing a threat to confidentiality. (3) Replay attacks include sending the same request to the receiver as an authenticated user. (4) Man-in-the-middle attacks (MITM) threaten confidentiality and integrity if data is not communicated securely. First, we present the architecture of the proposed blockchain system for secure storage of credentials of edge devices, gateways, and fog nodes. Second, Create a trusted identity management system and authentication of the edge nodes. Finally, we end up with secure data communications and avoid the single point of failure vulnerability inherited in a centralized system.

Abbreviations used in the Proposed Scheme:
Pub_d - Device public key, Prv_d - Device private key, Pub_r - Receiver's public key, $Signature_d$ - Device signature, Enc - Encrypt (), EA - Ethereum address, txn - Transaction instance, Ni - Nonce, dev_{ID} - Device identifier, Pub_g - Gateway public key, Prv_g - Gateway private key, Prv_r - Receiver's private key, $signature_g$ - Gateway signature, Dec - Decrypt (), SC - Smart contract instance, $signedtxn$ -

Signed transaction, TS_i - Timestamp $'i'$, $recv_{ID}$ - Receiver device identifier

Assumptions:

- We assume the gateway has better computation capabilities to generate the ECC key pair for the edge nodes and own key pair generation.
- The fog nodes are the validators of the blockchain that has secure Keystore capabilities

A. System Model

In this proposed work, we consider the fog-based blockchain IoT network that consists of a set of fog nodes (FN) serving K network services for different applications such that$FN = \{F_1, F_2, F_3 \ldots, F_K\}$. Each fog node F_i; $1 \leq i \leq$, maintains the permissioned blockchain network (pBCN) network and holds a common smart contract to validate the edge devices ED=$\{e_1, e_2, e_3 \ldots, e_n\}$. We consider these fog nodes to form a distributed peer-to-peer network. We also considered there would be multiple gateways $G = \{g_1, g_2, g_3 \ldots, g_m\}$. connected to a single fog node, and multiple edge devices are connected to a single gateway. We consider every edge node $e_i \in ED$ registers to one of the selected fog nodes $F_i \in FN$ for data transmission, where ED are the set of edge nodes, and FN are the fog nodes already connected to the fog-based IoT network. Within the time duration $'t'$ each edge node e_i sends the registration request, and each fog node F_i receives the identity messages of their edge devices. ${F_{it}}^{e_i}$.

$$F_i(t) \subseteq \bigcup_{e_i \in ED} e_i(t) \tag{1}$$

Each fog node F_i initiates the transaction (identity messages of the edge nodes) digitally signed with the ECDSA technique, edge and gateway devices can verify the transaction. The fog nodes maintain a permissioned blockchain network (pBCN) and communicate using smart contracts. Further, every fog node collects the identity details (signed transactions) from their edge devices via a gateway. One of the fog nodes(leader) F_l will collect all registration and identity details of the edge devices (signed transactions) from the other fog nodes $F_(n-1)$ with their timestamp ts. The leader fog nodes form the block $B_i$by combining the signed transaction of all edge nodes registration. The leader fog nodes send this B_i To validate and verify their registered edge and gateway devices to all fog nodes. After the validation and verification process, the block B_i is updated in the permissioned blockchain network (pBCN). The proposed approach performs Gateway registration and authentication, Edge device registration and authentication, and secure device-to-device communication in the pBCN network.

B. Proposed Architecture

The proposed architecture scenario is outlined in Figure. 1. This proposed architecture shows a proof of concept to secure the fog-based IoT network with device-to-device communication using a blockchain system. The following describes the role of each component in the proposed architecture.

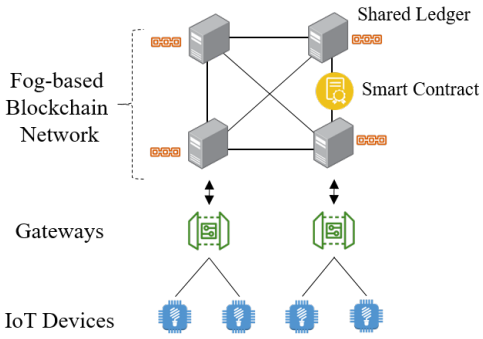

Fig. 1. Proposed System Architecture

1) IoT Edge Device: Each edge device is an IoT device with high resource constraints such as Limited Memory, Limited Bandwidth, Limited Processing, and Limited energy. These devices will communicate to the fog node, Via the gateway. The edge device communicates with the gateway using the MQTT protocol using the publish/subscribe model, i.e., the device publishes onto a topic to which the gateway is subscribed to send a message and vice-versa.

2) Gateway: Each gateway manages multiple IoT edge devices, and a single fog node contains it. The gateway communicates with the edge device using MQTT protocol and invokes necessary smart contract functions. The services performed by the gateway are Device *Registration*, *Authentication* and *Secure Communication*.

3) Fog Nodes: The fog nodes form a peer-to-peer network and hold a common smart contract that runs on each node. These nodes are relatively more potent than the edge device or the gateway. Therefore, they can maintain a shared ledger/blockchain and participate in consensus mechanisms by signing and validating the transactions received from other fog nodes as part of the consensus algorithm. These nodes relieve edge devices and the gateway's computation and storage capacity burden.

4) Smart Contract: Smart contract has an essential role in the proposed architecture. It allows us to add custom logic to the pBCN network. It will enable storage of the devices' identities, gateways, and mapping of gateways to the list of devices associated with it on the pBCN network. The smart contract functions allow registration, retrieving the credentials that aid authentication, messages for the devices, and other access control functionalities. The transactions performed through smart contract is binding and immutable. This ensures the integrity and tamper-proof of the data stored.

C. Edge Device Registration

Suppose a device wants to join the fog-based IoT network to communicate with the fog node. Then the edge device must register to be part of the pBCN network Via the gateway. The gateway listens for registration requests from edge devices and accepts the device identity and its public key. We assume that the gateway's public key is pre-distributed, i.e., the edge device has the gateway's public key that is already connected with it. The gateway and edge device communicate with each other using MQTT (Message Queuing Telemetry Transport) protocol using the publish/subscribe model, i.e., the device publishes onto a topic to which the

gateway is subscribed to send a message and vice-versa. The edge device registration process is presented in Algorithm. 1

Algorithm 1 Edge Device Registration

mqtt.connect($< BrokerURL >$)
mqtt.Subscribe($< devID >$)
$signature_d \leftarrow Sign(TS_0)$
$cipher_1 \leftarrow encrypt(Pub_g, devID, Pub_d, TS, signature_d)$

mqtt.Publish($cipher_1$) {Request sent from device to gateway}
$response_1 \leftarrow$ mqtt.onMessage() {Response from gateway}
status $\leftarrow decrypt(Prv_d, response_1)$
if status == true **then**
  "Registration is successful."
else
  "Registration failed."
end if

1) The gateway starts listening on a topic "$< GatewayPublickey > /register$", which accepts registration requests, while the edge device listens on " $< Device - ID >$" to get responses or messages.
2) The edge device sends a request containing its device identifier and public key to the gateway device with its initial timestamped signature to prove that it's a legitimate registration request and encrypted with the gateway's public key.
3) Upon receiving the request, the gateway decrypts the request message, verifies the signature, and creates a transaction containing the device's identifier and public key.
4) The gateway signs the transaction with its private key and sends it to the pBCN network fog node smart contract.
5) The smart contract verifies whether the gateway is registered, marks the device as registered, and maps it to this gateway.
6) The gateway receives the receipt upon success and sends an encrypted and authenticated status about the registration of the device.

D. Gateway Registration

This section presents the registration process of Edge Devices and Gateways in the pBCN network. All network participants must register themselves, such as edge devices, gateway, and fog nodes. Only those authenticated against the registered details in the pBCN will access the network services. The gateway and fog node will simplify the process, like the registration process, by storing the device's credentials on the blockchain, retrieving identity data of the device from the pBCN using the smart contract, and relieving the device of holding smart contract functions. However, to perform the above process, The gateway must first register in the pBCN network. It can authenticate and accept communication requests from the registering edge devices. The registration process of the gateway on the pBCN network is presented in Algorithm. 2.

Algorithm 2 Gateway Registration

mqtt.connect($< BrokerURL >$)
SC $\leftarrow$ web3.Connect($< FogNodeURL >$)
mqtt.Subscribe()
$request_1 \leftarrow$ mqtt.onMessage()
$\{devID, Pub_d\} \leftarrow decrypt(Prv_g, request_1)$
$txn \leftarrow$ new Transaction(register-device(devID, Pub_d))
signed-$txn \leftarrow$ sign(Prv_g, txn)
(receipt, error) $\leftarrow$ SC.send(signed-txn)
if error == null **then**
  status $\leftarrow$ TRUE
else
  status $\leftarrow$ FALSE
end if
$cipher_1 \leftarrow$ encrypt(Pub_d, status)
mqtt.Publish($cipher_1$)

E. Device Authentication

Once all the network participants registered in the pBCN network. Only authenticated devices registered in the pBCN network will access the network services. The smart contract of the pBCN network verifies the credentials, whether the raised gateway/fog node is registered with the associated edge device, before providing access to the gateway/fog node. The authentication mechanism acts as a general boilerplate for processing requests made by the edge device.

1) The gateway and edge device will connect to the MQTT broker to communicate with each other.
2) The edge device starts by sending a request containing its $< Device - ID >$ and Timestamp encrypting using the gateway's public key. The nonce for the edge device is stored in the pBCN network against device identity, and if changes happen at any time, then authentication will occur.
3) The gateway decrypts the request using its private key and then:
   - Through smart contract, it verifies whether the device was previously registered using the device id and timestamp of the request. Retrieves the nonce corresponding to the device from the pBCN network to be sent to the device.
4) The gateway then creates a signature using its private key that is hashed together with the retrieved nonce and latest timestamp to maintain freshness. It also encrypts the device's public key and sends it to the edge device.
5) The edge device, upon receiving the response of the gateway:
   - Decrypts the response using its private key.It verifies the signature of the gateway and timestamp. If the timestamp is old, the device doesn't proceed further. Retrieves the nonce.
6) The edge device then creates a signature with the received nonce, encrypts a signature using the gateway public key, and sends it to the gateway.
7) The gateway decrypts and verifies whether the signature was created using the same nonce as stored

in the pBCN network against the device identity and produces the device status.

8) The gateway sends the authentication status to the edge device if there is a successful notification.

IV. Implementation and Performance Analysis

The section presents our proposed work's experimental setups, results, and security analysis. We assume that the participants cannot be trusted and the path of communication may be lossy, unreliable and with potential delays.

A. System Setup

This subsection presents the implementation details of proposed framework.

1) Implementation Details of pBCN Blockchain Network: We implemented the proposed pBCN network for the IoT edge devices using the private Ethereum 2.0. The dependencies and tools for private Ethereum implementation are Node.js (v18.15.0), Ganache, Truffle Framework (v5.0.2), Solidity (v0.5.0), Metamask and web3.js.

- **geth (Go-Ethereum):** The Fog Nodes/Validators run on different ports of the pBCN network. All Fog Nodes will remain fully synchronized with the latest updates to the pBCN network. Each fog node has a key pair stored in a secure Keystore file.
- **Smart Contract:** The smart contract was implemented in solidity, an object-oriented language for writing smart contracts.

2) Gateway: The IoT devices interact with the fog nodes via this gateway identified by its Ethereum address. It was implemented in JavaScript on NodeJS and uses the web3 library to connect to a particular fog node (geth client) and invoke smart contract functions. MQTT protocol is used to communicate with these devices.

3) IoT Edge Device: The edge device was simulated using javascript in the browser and interacted with a particular gateway using MQTT protocol for a registration request, authentication, and sending messages to other devices which may or may not be under the same gateway.

B. Simulated Fog based IoT Environment

We simulated 500 edge devices, 12 gateways, and 15 fog nodes to test the network. Further, we used 256-bit ECDSA and ECDHE algorithms to encrypt messages from fog nodes to edge devices and device-to-device communication and showed performance variation. However, for simplicity, we will show the instances for simulated two IoT devices, each under a different gateway, and each one of the gateways connected to another fog node and part of the same network. The network of two fog nodes, each running has a different URL, such as Fog node 1: 127.0.0.1: 8545 and Fog node 2: 127.0.0.1: 8546.

C. Security Analysis

Confidential and Authentication: In our work, the authenticated gateway will only execute the transactions for those devices that are registered under it and the smart contract enforces this across all the gateways and fog nodes thereby maintaining confidentiality among different groups of devices. The gateway requests device data like nonce for a device not registered under it. Since smart contract contains all the device mappings it does not reveal confidential data of other devices to unauthorized gateways. Additionally, the fog nodes and the ledger is distributed among the fog nodes. The network is resistant to single point of failure. The security analysis against the reply attack is available on GitHub. (https://github.com/Aswani08/tenconresults.git)

V. Performance Analysis

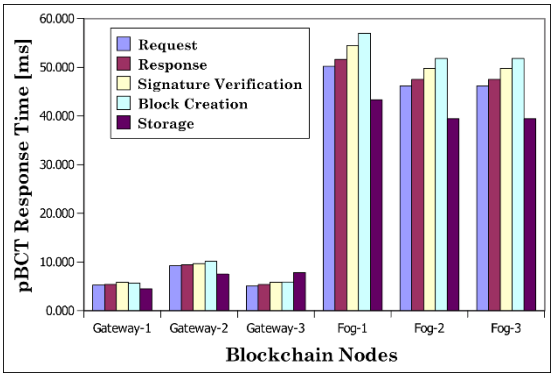

Fig. 2. Average Response Times for Node Registration

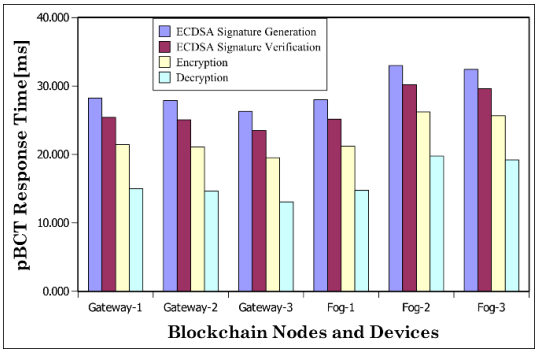

Fig. 3. Average Response Times for Key Generation in pBCT Network

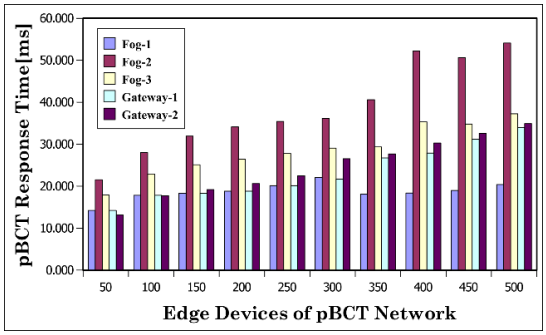

Fig. 4. Average Response Times for Edge Devices with Authentication Process in the pBCT network

We simulated 500 edge devices, 12 gateways, and 15 fog nodes to test the network. Further, we used 256-bit ECDSA and ECDH algorithms to encrypt messages from fog nodes to edge devices and device-to-device communication and showed performance variation. The proposed framework is evaluated in terms of average response time per transaction in seconds (latency) and successful transactions per second,

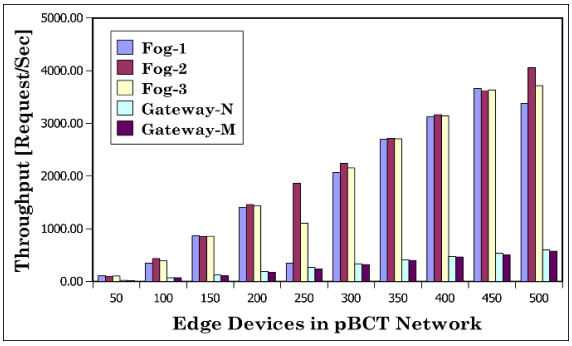

Fig. 5. Average Throughput

called average throughput. The Figure. 2 The average time the edge devices take to register in the pBCT network shows. it is observed that the gateway takes the minimum 8.67 ms response time to process the request and response back to the edge devices. However, the gateways do not store the transaction blocks. The same figure also shows the average 57.43 ms response time for registering the edge devices, creating a new block, and adding to the pBCT network. The Figure. 3 shows in the average response while generating the keys from the pBCT network. The maximum response time to generate the key pair for the gateways is 12.45ms, signature creation is 28.17ms, and signature verification is 26.89ms by the gateways. While fog nodes take the maximum response time for signature creation is 32.45ms, and signature verification is 30.29ms. These fog nodes must wait for proper validation for edge devices coming from other fog devices. The Figure. 4 shows the average response times of edge devices with the Authentication Process in the pBCT network. The average response time for authenticating the edge devices varying from 50 to 500 devices by the m-gateways is 23.088ms, and n-gateways is 25.517ms. At the same time, the average response time for authenticating the edge devices varying from 50 to 500 different fog nodes is 26.723ms, 28.474ms, and 28.598ms, respectively. The Figure. 5 shows average throughput achieved by the varying 50 to 500 edge devices. The pBCT network achieved better throughput with increased edge device registration. The average throughput achieved by the fog node-2 is 2049.48 requests/sec, and the average throughput achieved by the gateway-1 is 301.34 requests/sec.

VI. Conclusion

In this paper, we proposed a fog-based blockchain network that authenticates the edge devices using a private blockchain system, which reduces IoT edge devices' storage overheads and computations by shifting its operations to gateways and fog devices. The main objective of the proposed network is to allow multiple devices to communicate with each other. Hence, each device may communicate with a vast number of devices, and the associated gateways must coordinate those exchanges with the help of the proposed blockchain network. We also performed the security analysis of the proposed communication network from unauthorized malicious activity and protection of IoT edges devices, resilient from sniffing the exchange of messages between the network participants, resilient against replay attacks, and ensuring the confidentiality, integrity, authenticity, and availability of the edges devices.

References

[1] Li, D., Peng, W., Deng, W., & Gai, F. (2018, July). A blockchain-based authentication and security mechanism for IoT. In 2018 27th International Conference on Computer Communication and Networks (ICCCN) (pp. 1-6). IEEE.

[2] Fakhri, D., & Mutijarsa, K. (2018, October). Secure IoT communication using blockchain technology. In 2018 international symposium on electronics and smart devices (ISESD) (pp. 1-6). IEEE.

[3] BANOUN, N., & DIARRA, N. (2021, May). IoT-BDMS: securing IoT devices with hyperledger fabric blockchain. In CS & IT Conference Proceedings (Vol. 11, No. 6). CS & IT Conference Proceedings.

[4] Priyadarshini, R., & Malarvizhi, N. (2021). Secured Data Transfer Between Fog Nodes Using Blockchain. In Proceedings of the 2nd International Conference on Computational and Bio Engineering (pp. 417-422). Springer, Singapore.

[5] Fayad, A., Hammi, B., & Khatoun, R. (2018, October). An adaptive authentication and authorization scheme for IoT's gateways: a blockchain based approach. In 2018 Third International Conference on Security of Smart Cities, Industrial Control System and Communications (SSIC) (pp. 1-7). IEEE.

[6] Babu, E. S., Kavati, I., Nayak, S. R., Ghosh, U., & Al Numay, W. (2022). Secure and transparent pharmaceutical supply chain using permissioned blockchain network. International Journal of Logistics Research and Applications, 1-28.

[7] Babu, E. S., Dadi, A. K., Singh, K. K., Nayak, S. R., Bhoi, A. K., & Singh, A. (2022). A distributed identity-based authentication scheme for internet of things devices using permissioned blockchain system. Expert Systems, e12941.

[8] Almadhoun, R., Kadadha, M., Alhemeiri, M., Alshehhi, M., & Salah, K. (2018, October). A user authentication scheme of IoT devices using blockchain-enabled fog nodes. In 2018 IEEE/ACS 15th international conference on computer systems and applications (AICCSA) (pp. 1-8). IEEE.

[9] Imine, Y., Kouicem, D. E., Bouabdallah, A., & Ahmed, L. (2018, August). MASFOG: an efficient mutual authentication scheme for fog computing architecture. In 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE) (pp. 608-613). IEEE.

[10] Patwary, A. A. N., Fu, A., Battula, S. K., Naha, R. K., Garg, S., & Mahanti, A. (2020). FogAuthChain: A secure location-based authentication scheme in fog computing environments using Blockchain. Computer Communications, 162, 212-224.

[11] Tuli, S., Mahmud, R., Tuli, S., & Buyya, R. (2019). Fogbus: A blockchain-based lightweight framework for edge and fog computing. Journal of Systems and Software, 154, 22-36.

[12] Nakamoto,Satoshi. Bitcoin: A peer-to-peer electronic cash system. 2008.

[13] Borgohain, T., Borgohain, A., Kumar, U., & Sanyal, S. (2015). Authentication systems in internet of things. arXiv preprint arXiv:1502.00870.

[14] Pustišek, M., & Kos, A. (2018). Approaches to front-end IoT application development for the ethereum blockchain. Procedia Computer Science, 129, 410-419.

[15] Ali, J., Ali, T., Musa, S., & Zahrani, A. (2020). Towards secure IoT communication with smart contracts in a blockchain infrastructure. arXiv preprint arXiv:2001.01837.

[16] Mishra, B., & Kertesz, A. (2020). The use of MQTT in M2M and IoT systems: A survey. IEEE Access, 8, 201071-201086.

[17] Zheng, Z., Xie, S., Dai, H. N., Chen, X., & Wang, H. (2018). Blockchain challenges and opportunities: A survey. International journal of web and grid services, 14(4), 352-375.

[18] Babu, E. S., Srinivasarao, B. K. N., Kavati, I., & Rao, M. S. (2022). Verifiable authentication and issuance of academic certificates using permissioned blockchain network. International Journal of Information Security and Privacy (IJISP), 16(1), 1-24.

[19] Ruoti, S., Andersen, J., & Seamons, K. (2016). Strengthening password-based authentication. In Twelfth Symposium on Usable Privacy and Security (SOUPS 2016).

[20] Aguru, Aswani Devi, Erukala Suresh Babu, Soumya Ranjan Nayak, Abhisek Sethy, and Amit Verma. 2022. "Integrated Industrial Reference Architecture for Smart Healthcare in Internet of Things: A Systematic Investigation" Algorithms 15, no. 9: 309. https://doi.org/10.3390/a15090309