BlockTrack-L: A Lightweight Blockchain-based Provenance Message Tracking in IoT

Muhammad Shoaib Siddiqui[1], Toqeer Ali Syed[2], Adnan Nadeem[3], Waqas Nawaz[4], Sami S. Albouq[5]
Faculty of Computer and Information Systems
Islamic University of Madinah
Madinah, Kingdom of Saudi Arabia

***Abstract*—Data tracking is of great significance and a central part in digital forensics. In today's complex network design, Internet of Things (IoT) devices communicate with each other and require strong security mechanisms. In maintaining an audit trail of IoT devices or provenance of IoT device data, it is important to know the origins of requests to ensure certain level of trust in IoT data. Blockchain can provide traceability of records generated from IoT devices in a sensitive environment. In this paper, we present an application layer data provenance model that works on execute-order architecture for cloud based IoT networks. It supports high throughput of transactions on the blockchain network with lightweight security overhead by using outsourced encryption on edge nodes. All communications among the IoT devices are connected to a blockchain network and stored on permissioned blockchain peers. The proposed system is evaluated to have less cryptographic load by offloading the IoT nodes with Edge nodes.**

Keywords—Data provenance; cloud-based IoT; blockchain; attribute based encryption; light-weight signature generation; light-weight authentication

I. Introduction

Digital data tracking is an important concept and has been studied in the past couple of decades for privacy, security and forensics [1]. Data provenance has been examined in many areas for various purposes, such as, copyright of an artistic work, intellectual property, scientific contribution tracking, regulatory and legal considerations etc. Similarly, data tracking is of great significance and a central part in digital forensics [2]. HealthCare has an essential aspect to trace patient records for any kind of conflict resolution [3]. In short, keeping data secure, transparent and building trust among various users on the Internet require a strong record storing and sharing mechanism in today's complex network. The importance of transactional data provenance is high in various real-life applications, such as knowing the digital history of ownership of a car is imperative before its purchase. As in many cases, the ownership history of a used car has an effect on his expected price. In another example of buying a property, it is essential to track digital provenance data related to the ownership of the property. In [4], the author proposes a trusted property registration system on permissioned blockchain to track the digital provenance of the property.

Ragib et al. [5] have given a detailed data provenance mechanism on the application level at traditional operating system design. They have shown how to keep track of data writes at the kernel level, at the file system as well as at application level. However, in today’s complex networks and application design we need to extend these models for other architecture to cover future applications. In today's complex network design, Internet of Things (IoT) communicates with each other that requires strong security mechanisms. However, message tracking among IoT devices is also received great significance. Apart from that, provenance of data/messages has also been studied in databases [6], cloud computing, wireless sensor networks etc. [2]. In fact, message provenance of IoT devices communication on the blockchain network is still an open area of research.

Due to the inherent characteristics of the blockchain, i.e. keeping a history of immutable records, the field of data provenance can greatly benefit from the integration of blockchain technologies. Once *n* number of transactions are recorded as a block inside a blockchain or a distributed ledger and validated by the consensus algorithm, then that block could not be changed or deleted. Any attempt to change or alter the data would be identified by the peer nodes and rejected by the blockchain. The reason the blockchain can ensure immutability of the recorded transaction is because it is computationally reliable and secure [7], [8].

IoT Devices are being used for remote monitoring, surveillance, actuation and control and there are billions of devices already on the Internet, as well as a billion more which are not directly connected to the Internet. These devices produce a lot of data and as this data is an integral part of decision making; therefore, the protection of this data is essential. The biggest threat to this data is the compromise on the integrity of the data. By using immutable record keeping of the blockchain, the data communication between the IoT devices and with the gateway/sink node can be secured. The integration of blockchain and IoT (named BIoT) can result in three-fold advantage. First, it can ensure the integrity of the data communication; and secondly, it can help in back-tracking the malicious senders and intermediate nodes. Finally, BIoT can secure the provenance control packets as well, which are attacked by colluding attackers to compromise the data provenance mechanism using spoofing and man-in-the middle attacks.

However, there is a cryptographic overhead of blockchain, which makes it unsuitable for IoT devices. Fortunately, in a

This work was supported by the Deanship of Research, Islamic University of Madinah, Kingdom of Saudi Arabia, 30/40 ~ 1440-1441).

cloud based IoT environment, the blockchain can be maintained at the cloud level and thus significantly reduce the cryptographic overhead of blockchain on the IoT devices. Every communication made by the IoT nodes can be stored on the blockchain using cloud services. The only issue would be to secure IoT communication with the cloud. This is normally achieved through encryption techniques, which again have a high computational load. In our previous work [2], we have devised a light-weight encryption mechanism which can calculate attribute-based signature by outsourcing encryption load to the Edge node instead of the IoT node, thus significantly reducing the computational load and providing better security services.

In this paper, we present an application layer data provenance model that works on execute-order architecture for cloud based IoT networks. It supports high throughput of transaction on blockchain network with lightweight security overhead by using outsourced encryption on Edge nodes. All communication among the IoT devices is connected on a blockchain network and stored on permissioned blockchain peers, while reducing the load on the IoT nodes. The rest of the paper is articulated as follows.

The related work is discussed in Section II, while Section III discusses the proposed mechanism. Section IV discuss the performance evaluation of our scheme and the paper is concluded in Section V.

II. Related Work

Data provenance has been utilized in decentralized systems to identify the source of data and to track records, identify data flows from a subset of the original inputs, and debug data flows [9]. One of the requirements for IoT networks is to ensure trust about data origin and location [10]. Suhail et al. in [11], first, indicated that research in the area of security of IoT does not focus on secure provenance and then highlighted various ways to include secure provenance in IoT based solutions [11].

In [12], the authors have also identified the challenges of secure provenance and identified the potential applications, such as, law, scientific data, digital forensics, regulatory compliance and authorship for secure provenance. In [13], the authors have claimed that secure provenance is the essential of bread and butter of data forensics. They have devised a secure provenance algorithm for cloud storage using authentication, authorization and their provenance tracking algorithm. The technique is based on the bilinear pairings and used the provable security technique [13] to prove its security in the standard model. However, the solution is too complex to be implemented on IoT devices.

In [14], the authors first identified that secure data provenance is vital for cloud data. Realizing this they propose an architecture for secure data provenance in cloud that can enable collection and verification of data provenance. Authors also evaluate the performance of proposed architecture and the results shows that proposed architecture provides data provenance of cloud data with low overhead.

Because blockchains can keep a record of unchanging transactions that are computationally safe and reliable, historical data about communications or transactions between IoT devices can also be recorded in a similar way. Data sourcing is a technique used to provide data traceability from source to destination and are used to ensure the sender's data integrity and authentication. Integrating blockchain mechanisms into your IoT infrastructure will ensure that your data is safe and secure from medium and data spoofing attacks. Therefore, several solutions have been proposed to ensure data source in BIoT environment, such as [15]. In a supply chain scenario, blockchain-based data source solutions can be utilized for asset and commodity tracking [16]. Chronicled is a solution for the secure exchange of physical assets using BIoT [17].

In [15], Ricardo et al. have proposed a blockchain-based approach data provenance, based on the public blockchain known as Ethereum. Ethereum works on a proof-of-work consensus algorithm and is an order execution architecture. However, transactions/smart contracts are designed specifically for cryptocurrencies and are considerably slower in networks that are not suitable for general purpose applications.

For data integrity and cloud audit provisioning, Liang et al. proposed the idea of protecting drone data collection and communication with the public blockchain in [18]. Similarly, in [19], blockchain based solution is proposed to build an immutable data sourcing system for IoT-based networks using a distributed architecture to ensure data integrity. In business driven IoT, end users must share personal information with multiple third parties. To prevent data leakage and to protect user privacy, the authors have proposed a solution to isolate and serve different information retrieval requests for each type of personal information.

In [20], the authors have presented a blockchain framework for IoT networks, which can maintain security of transactions while considering the low computational resources of the IoT nodes. It restricts the number of transactions to be logged inside the global blockchain by using a scalable local ledger on a local peer network, which stores the local transactions and a global ledger for storing global transactions.

Ali et al. have presented a secure data provenance mechanism for cloud based IoT by using smart contracts [21]. Blockchain based smart contracts are used to store the meta-data of the actual communications for maintaining data provenance, while the actual data is stored on the cloud.

Zhang et al. have presented a secure blockchain based architecture for data sharing between IoT nodes [22]. They have used attribute-based signature and encryption for providing access control to implement data provenance. The consensus model used by the authors is the byzantine fault tolerance instead of the proof-of-work, which is used by most of the public blockchains.

In [23], Javaid et al. have presented a blockchain based data integrity and provenance mechanism for securing IoT communication by utilizing physical unclonable functions and Ethereum based permissioned blockchain for their mechanism.

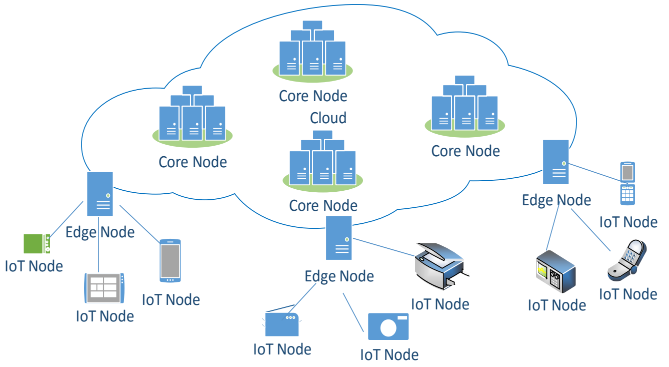


Fig 1. The Communication Model of the Proposed Design System.

Considering the above-mentioned related work, we can conclude that the integration of blockchain with IoT is inevitable and brings along lots of advantages and various solutions for maintaining traceability in generic BIoT systems are proposed in the literature; however, these solutions are heavy on the computationally scarce IoT nodes. The solutions which use public blockchain have a high convergence time as all peer nodes are involved in the consensus algorithm and as most of the nodes are IoT nodes which limited capabilities, the solutions have a high computation footprint. On the other hand, the permissioned blockchain-based solution used encryption-based registration mechanism for authentication, which have computational overhead on the resource constraint IoT nodes. Therefore, in this paper, we have proposed a permissioned blockchain based data provenance mechanism which offloads the computational load of (1) hash calculation for blockchain and (2) cryptographic load of digital signature by outsourcing the mechanism to the Edge nodes. The details of the scheme are discussed in the following sections.

III. Proposed System

A. *Communication Model*

The considered model for communication is a cloud based IoT network with IoT devices connected to the cloud with the Edge nodes as the first point of contact. As the edge nodes are close to the IoT device, being at the edge of the cloud, each of them is assigned to some IoT nodes as the gateway node. Now, this edge node has relatively higher, power, storage and computational resources, which make it an ideal candidate to perform the computationally heavy tasks of digital-signature generation and validation on behalf of the IoT nodes. Furthermore, the Edge node would publish the provenance data on to the blockchain also. Fig. 1 shows the scenario of the proposed system.

B. *Threat Model*

The objective of the attacker for the proposed system is as follows:

- An attacker can access IoT devices to compromise data integrity as there is no physical security for IoT devices.
- An attacker can pretend to be a legitimate user and mimic as an authorized IoT node. The target of the attacker is to compromise the provenance data accuracy by injecting false data into the system.
- An attacker can violate confidentiality and integrity of messages.
- An attacker would try to compromise the data integrity of the information sent from an IoT node to the cloud and exploit the provenance mechanism.

The system should be able to provide information about the source node from where the data was originated in a similar case. The system should be able to identify data origin, along with the time it was originated, and the path taken by the data packets.

C. *Blockchain based Provenance System*

Blockchains are used to keep a record of unchanging transactions that are computationally safe and reliable, historical data about communications or transactions between IoT devices can also be recorded in a similar manner.

IoT nodes generate significantly large amounts of data. In a conventional provenance mechanism, the data is stored at all or some of the intermediate nodes to avoid attacks on the integrity of the provenance data. However, due to the immutable record keeping of the blockchains, the need to save provenance data at every node or some nodes is not required. With respect to data provenance, there are four steps involved in provenance mechanism:

1) *Data Gathering*
2) *Data Recording & Publishing*
3) *Data Validation*
4) *Database Update*

Data Gathering: Whenever a data packet is sent from an IoT node, it is stored on the blockchain as a transaction. The data is received by the Edge node and forwarded to the blockchain.

Data Recording and Publishing: After the data is received, it is sent to the endorserer node. The endorserer node authenticates the IoT node by verifying the signature. It executes the smart contract related to the device registration and authorization.

The hash is calculated for the data packet along with the timestamp. After that the message is sent to the anchor node. This hash is used by the orderer node to create the block on the blockchain. In our scheme, we use Kafka ordering.

Then the block is broadcasted to the peer nodes by the anchor node. Individual peers then update their local ledger with the latest block. Thus, all the network nodes get the ledger synchronized. On the cloud, the provenance data is received by the provenance auditor, which would save it in the provenance database after the validation process. The working of the proposed mechanism for storing data transactions on the blockchain is given in Fig. 2 and explained as follows:

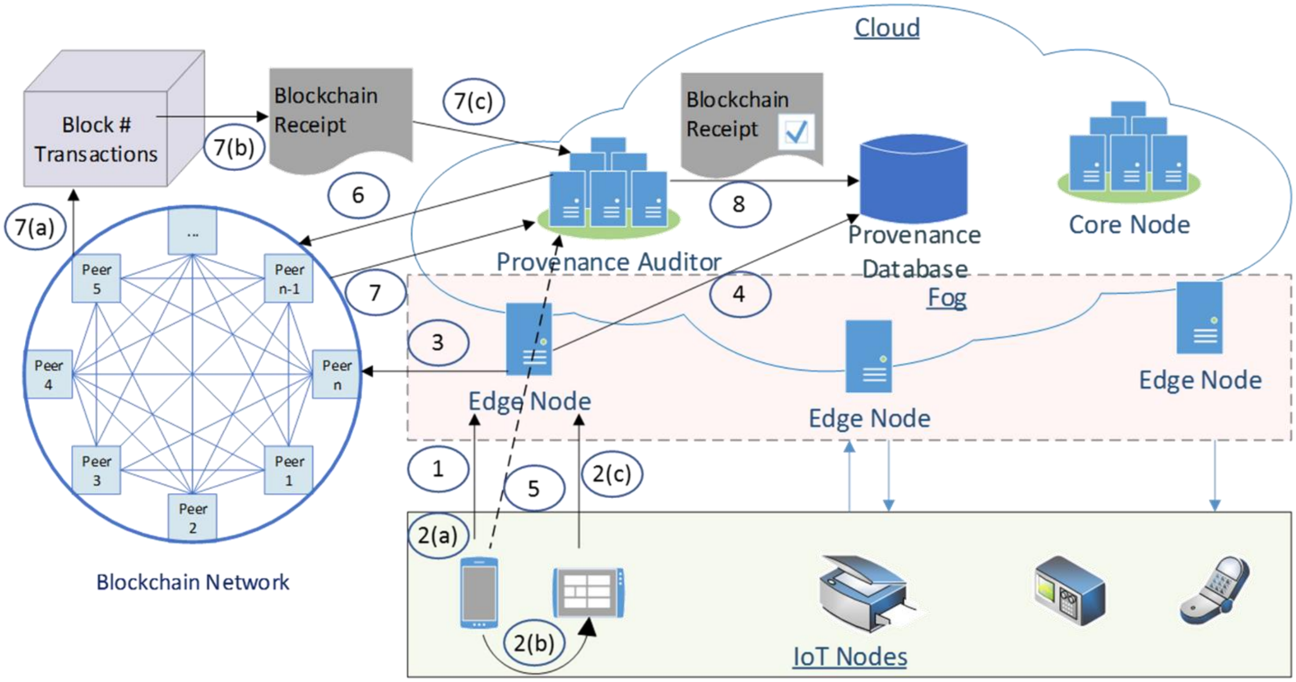


Fig 2. The Blockchain System Integrated with the Cloud based IoT Network.

1. IoT device registers using an authentication key KU_R.

1) Edge node partially signs using outsourced signing algorithm and send to the IoT node

2) IoT Node complete the signature.

2. IoT device sends data to another IoT node. The IoT node signs the data using the partially signature sent by the edge node and sends it back to the Edge node.

3. Edge node forwards the digitally signed data packet to the blockchain

4. Blockchain publishes provenance data on the blockchain Network through provenance auditor and save the block in the Provenance Database.

5. Edge node stores provenance data locally in the Provenance Database through the Provenance Auditor.

Provenance Validation: The validation mechanism for a transaction (Message sent) on the blockchain is given as:

1. IoT node requests data provenance validation from Provenance Auditor through the Edge node.

2. Provenance Auditor validates provenance data from the blockchain Network.

3. Blockchain Network returns validation result to Provenance Auditor.

4. Provenance Auditor updates provenance data validation status at the Provenance Database.

The provenance data is stored on the blockchain with the help of smart contracts. Fig. 3 shows how the data is stored inside the blockchain. It shows the storage during three major operations of data collection, its verification and database update using smart contract. Smart contracts allow to conduct reliable transactions without the involvement of third parties. These transactions are traceable and irreversible. There are three types of smart contracts present in the proposed system. These are initiated when the following operations occur:

1. IoT Device Registration

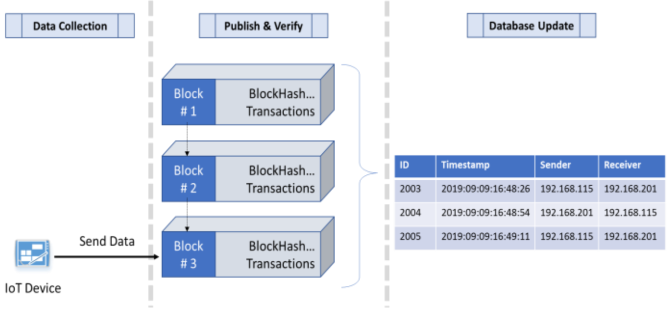
2. Data Transfer (Transaction)

3. Provenance Verification (Validation)

When an IoT node joins the network, the ***IoT Device Registration*** smart contract is initiated. IoT node is assigned a secret key from the authority attribute (present at the cloud). Similarly, a secret key is assigned to the corresponding Edge node. The Edge node calculates a partial signature based on the attributes selected by the attribute authority and sends it to the IoT node. The IoT node creates the complete signature and sends the signed message to the blockchain for registration.

Similarly, when a message is sent from an IoT node to another node, ***Data Transfer*** smart contract is executed. This is responsible for storing the provenance data (source, message and timestamp) on to the blockchain.

Finally, at the time of verifying that the message was sent by the corresponding IoT node, ***Provenance Verification*** smart contract is executed, which is responsible for validating the provenance data. Fig. 3 shows how the data is stored inside the blockchain.



ID	Timestamp	Sender	Receiver
2003	2019:09:09:16:48:26	192.168.115	192.168.201
2004	2019:09:09:16:48:54	192.168.201	192.168.115
2005	2019:09:09:16:49:11	192.168.115	192.168.201

Fig 3. The Blockchain Structure for Storing Data Communication Performed by the IoT Nodes.

D. Light-Weight Authentication System

The blockchain provides an immutable record of each communication performed inside the cloud based IoT network; however, as the hash is stored on a permissioned blockchain, the IoT nodes is required to be authenticated to maintain the data integrity. This section provides the details of the outsourced signature-based mechanism used to provide authentication and data integrity, which is suitable for IoT devices as it offloads the computational load [2].

Ciphertext-Policy Attribute based Encryption (CP-ABE) is utilized to authenticate IoT nodes on the blockchain [24] [25]. Fig. 4 shows the conceptual model of implementing the CP-ABE in IoT based Cloud environment. The details of the algorithm can be seen in our previous work [2], where the authors have implemented the load sharing of cryptographical computation by outsourcing the signing and signature verification to the Edge nodes.

In our outsourced CP-ABE mechanism, Edge node creates a partial semi-signature on behalf of the IoT node; while performing most of the computational task. This semi-signature is received by the IoT node, which perform negligible computation to calculate the complete signature. IoT node can use this signature to authenticate itself while communicating with the other IoT devices and gateway nodes. Data integrity, sender authentication and data accuracy can be ensured by using this algorithm in the proposed system.

The outsourced digital signature generation algorithm is shown in Fig. 5 which has five phases that are: Setup, KeyGeneration, Sign$_{\text{partial}}$, and Sign$_{\text{complete}}$. Sign$_{\text{partial}}$ has most of the computational load, so it's done at the Edge node, while a significantly lower overhead, i.e. signing, is done at the IoT node. At each phase certain computation is required which is discussed here:

Setup is the first phase. The attribute authority (AA), which can be at the cloud or the IoT network or the Edge node itself can implement this phase. The attribute authority must select certain factors which are used as the initialization parameters of the proposed mechanism; which are: a universal set of attributes $\varLambda$, the security parameter υ, and an auxiliary information ε. The setup process provides the master key $\varOmega$ and a public key κ.

Key Generation is the second phase that is executed at the AA, which should exist on the cloud or the fog/edge node. Whenever an IoT node desire to send a data packet, it should consult with the AA first, to obtain a secret key with a particular attribute set $\varLambda_i$. The AA would take the attribute set $\varLambda_i$ and the master key $\varOmega$, and generates a pair of secret keys; K_{IoT} and K_{Fog} for the IoT and edge nodes, respectively. These keys will be used by the IoT and edge nodes to create the digital signature partially by the edge node and complete signature by the IoT node, sequentially.

After that the third phase is **Sign$_{\text{outsourced}}$** in which the partial, outsourced signature is calculated by the Edge node. By using the predicate function $\varPi$, attribute set $\varLambda_i$, and the private key of the Edge node K_{Fog} (provided by the AA), the Edge node generates the partial signature using the outsourced algorithm, which takes up most of the computational overhead. The algorithm to create the partial signature is based on CP-ABE discussed in detail in [2]. After the signature is calculated, it can be used by the IoT device to completely sign a message using its own secret key K_{IoT}, provided by the AA.

After the fourth phase is complete, the IoT node receives the partial signature from the Edge node and performs the fifth phase; **Sign**, which is the last phase for generating digital sign for a message. The inputs for the Sign phase are the message M, the secret key K_{IoT}, the predicate $\varPi$, and the partial signature $\boldsymbol{\sigma_{partial}}$. Using the inputs, the algorithm generates the complete signature $\boldsymbol{\sigma}$ for the message M.

In the end, when the signature is completed by the IoT node, the message M, which contains the data, sender ID, timestamp and the signature is sent to the gateway or another IoT node through the Edge node. The attached signature could be used to ensure that the message is from the authenticated IoT node, data integrity could be maintained and if an attacker node tries to change the data then it can be identified during the verification phase.

Verification steps are performed to verify the integrity of the message and to authenticate the sender (see Fig. 6).

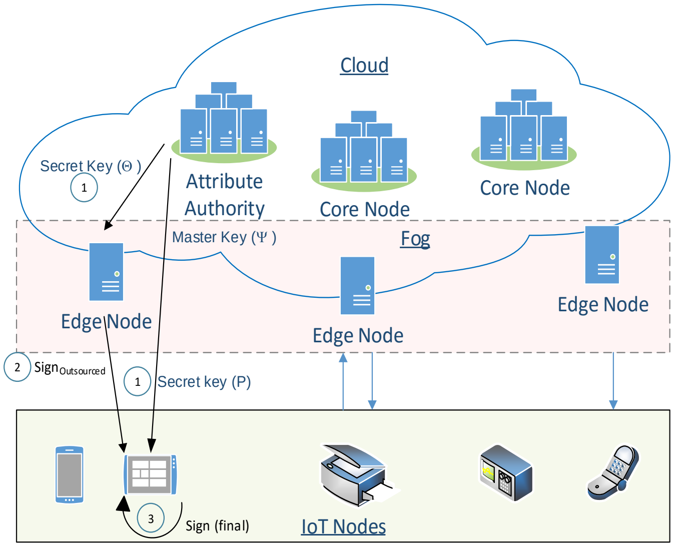


Fig 4. The Process of Signing by the Fog (Outsourced) and IoT (Partial) Nodes.

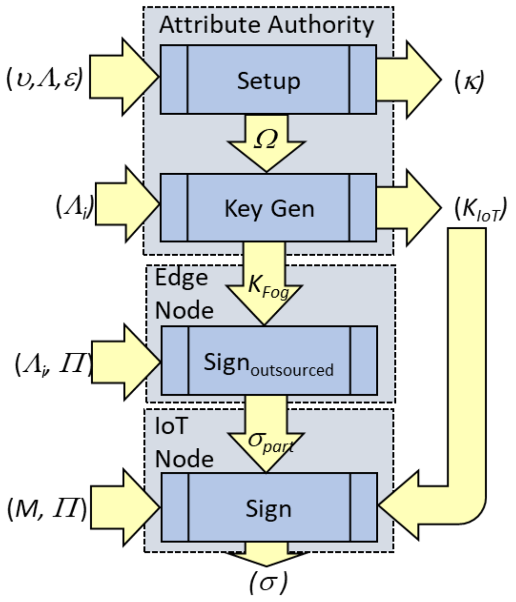


Fig 5. The Partial Signature is Created by the Edge Node and the Signature is Completed by the IoT Node using CP-ABE.

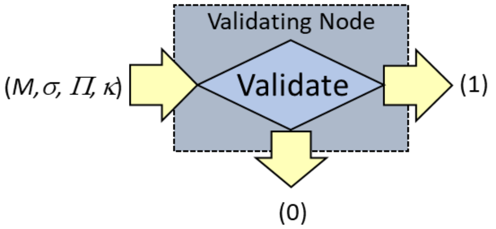


Fig 6. The Verifying Node can endorse the Signature by using the Proposed CP-ABE based Validation Technique.

The four phases are used to generate the digital signature, while the fifth phase is the named **Validate.** This phase is performed by the endorser node in the blockchain, who ensures that the message received is from the mentioned IoT node and not by a pretender. The endorserer node receives the message *M*, which includes the *ID* of the sender node and the digital signature σ. The endorserer node also receives the public key κ, and the predicate Π from the attribute authority AA. It validates if the signature is correct or invalid using the CP-ABE scheme.

If the validation process fails, the message is not authenticated and an attack to compromise the integrity of data is identified. In such case, the sender node could be asked to resend the data. If the validation step is successful, data about the message / packet is stored at the blockchain before that data is passed to the receiving node. Before data / messages are forwarded, the data is stored at the blockchain along with source information and timestamp for each data packet / message, as shown in Fig. 3.

IV. Evaluation

For the performance evaluation of our system, we have developed our blockchain using Hyperledger Fabric [26] v.1.1.0 with 6 peer nodes, which are hosted on a single machine. The peer nodes are configured to have 2.0 GHz processor with 4 GB RAM. We have used BFT-SMaRt, which is a “high-performance Byzantine fault-tolerant state machine replication library developed in Java with simplicity and robustness as primary requirements” [27]. Four of the peer nodes are configured as the orderer nodes. Kafka orderer is used as the single ordering service. Byzantine fault-tolerant is used as the consensus algorithm.

We simulated to have 50 IoT nodes, capable of sending data. The IoT nodes are associated with one of the Edge nodes in the cloud. The Edge nodes are relatively computationally powerful nodes as compared to the IoT nodes, with configuration of 1.0 GHz processor and 2GB RAM. The core nodes in the cloud are 4.0 GHz with 8GB RAM.

Cipher-text Policy Attribute based Encryption (CP-ABE) based signatures are used to authenticate the IoT nodes and maintain data integrity. We use the light-weight scheme, which reduce the signing load from the IoT node with Edge nodes [2].

For calculating the throughput, we increase the number of IoT nodes who are sending data, starting from 1 node to finally all 50 nodes generating data at the rate of 2 messages per seconds. The rate of data packet generation was also increased with a constant number of nodes (30 nodes) from 10 packet per second to 300 packets per seconds to identify the effect of high load on specific edge nodes.

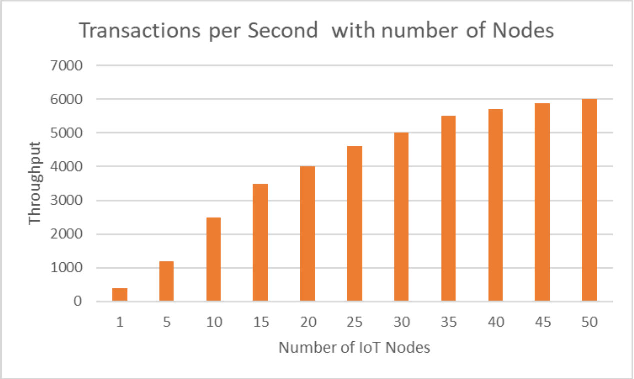


Fig 7. Throughput in Terms of Transactions per seconds for Varying Number of IoT Nodes.

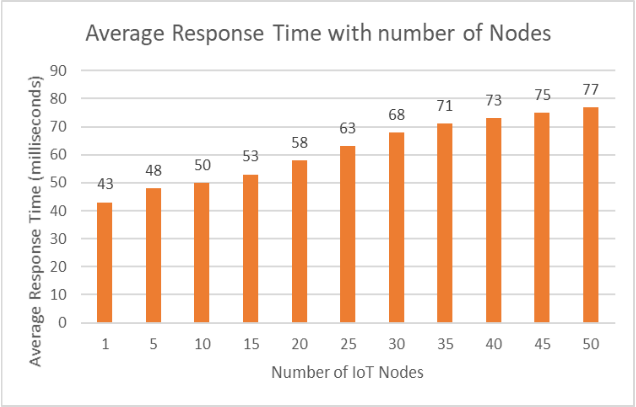


Fig 8. Average Response Time for the Blockchain Network for Varying Number of Data Sending Nodes.

The throughput is shown in Fig. 7. With 10 nodes and 300 packets per second data rate, we achieve a throughput of almost 2500 transactions per second for block-size of 2MB and latency of 50 milliseconds. The higher throughput was achieved with higher number of nodes. The limitation to 2500 transaction per second with 10 nodes is due to the fact that the load was mostly on some edge nodes while other edge nodes were idle, as there was no load on these edge nodes. The IoT nodes attached to these edge nodes were not generating any packets. When the number of IoT nodes were increased, we were able to attain a higher throughput as more edge nodes were involved and the load was being shared by all the edge nodes. However, if we closely monitor the curve of throughput with respect to the increase in the number of nodes, the curve shows algorithmic nature. This is due to the fact that the increase in data nodes increases the number of packets, which increases the load on the edge nodes as well as on the blockchain nodes.

The average response time is also calculated with the various block sizes, number of transactions, and number of peers. The average response time is shown in Fig. 8. The data generation rate is 100 packets per seconds, with block size of 2 MB, which is roughly 40 transactions in a block. The response is measured as the time taken to store the message on the blockchain and a response is received from the provenance authority. As the number of nodes are increased, the response time is again increased due to the load on the edge nodes and the blockchain peers.

For evaluating the CP-ABE based outsourced signature generation scheme, the overhead (in terms of milliseconds) of generating the public and private keys and the signature of the message with increased number of messages, with or without Edge nodes are measured, and shown in Fig. 9. We have increased the number of packets, each signed by the sending IoT node, and measure the overhead of the signing algorithms. The normal scenario, where the whole signing is done by the IoT node and the proposed mechanism, in which the partial signature if first generated by the edge node and the signature is completed by the IoT node. When the number of packets is less, then the advantage is not significant, but, as the number of packets increases then the overhead (time) is decreased significantly.

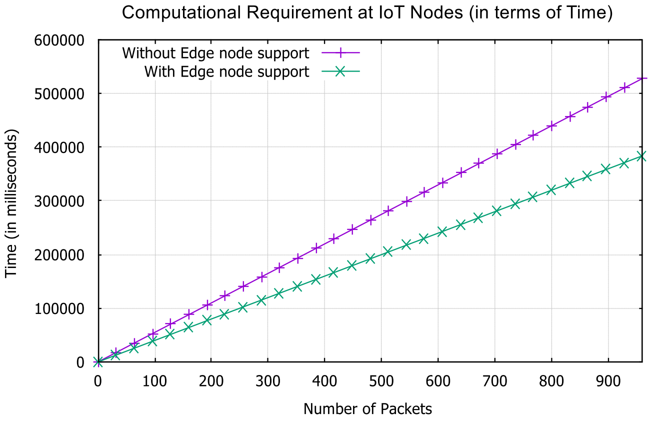


Fig 9. Computational Requirement at IoT Node with and without Outsourced Signature Scheme.

V. Conclusion

The use of blockchain technologies can solve the issue of data provenance in cloud based IoT environments due to its inherent properties of maintaining immutable records about each transaction. The literature review encourages the integration of blockchain and IoT (BIoT); however, public blockchain uses consensus algorithm which needs high computation and is not suitable for IoT devices. Although permissioned blockchain can provide fast convergence and use computationally lightweight mechanism, there is an issues of node authentication and data integrity, as IoT nodes are not capable of implementing authentication mechanisms. In this paper, we have provided a permissioned blockchain solution for maintaining secure data provenance which utilizes the outsourced attribute-based encryption. Our scheme reduced the overhead of authentication and blockchain mechanism from the IoT node by offloading it to the Edge node by using partial signatures. Thus, providing secure communication between the IoT node and the blockchain.

Acknowledgment

This work was supported by the Deanship of Research, Islamic University of Madinah, Kingdom of Saudi Arab [Project Title: “BlockTrack: Towards Blockchain-based Provenance Message Tracking in IoT Mesh Network”, 30/40 ~ 1440-1441.

References

[1] Peter Buneman, Sanjeev Khanna, and Wang Chiew Tan. Data provenance: Some basic issues. In Proceedings of the 20th Conference on Foundations of Software Technology and Theoretical Computer Science, FST TCS 2000, pages 87–93, London, UK, UK, 2000. Springer-Verlag

[2] Muhammad Shoaib Siddiqui, Atiqur Rahman, Adnan Nadeem and Ali M. Alzahrani, “Secure Data Provenance in Internet of Things based Networks by Outsourcing Attribute based Signatures and using Bloom Filters” International Journal of Advanced Computer Science and Applications (IJACSA), 10(5), 2019. http://dx.doi.org/10.14569/IJACSA.2019.0100529

[3] Syed, Toqeer Ali, Ali Alzahrani, Salman Jan, Muhammad Shoaib Siddiqui, Adnan Nadeem, and Turki Alghamdi. "A Comparative Analysis of Blockchain Architecture and its Applications: Problems and Recommendations." IEEE Access 7 (2019): 176838-176869.

[4] Toqeer Syed; Adnan Nadeem; Ali Alzahrani; Salman Jan, “A Transparent and Trusted Property Registration System on Permissioned Blockchain”, accepted for publication in proceedings of IEEE International Conference on Advances in the Emerging Computing Technologies (AECT), Islamic University of Madinah, February 10-12, 2020.

[5] Ragib Hasan, Radu Sion, and Marianne Winslett. “Preventing history forgery with secure provenance”. ACM Transactions on Storage (TOS), 5(4):12, 2009.

[6] T. McConaghy, A. Marques, Rodolphe, D. De Jonghe, T. McConaghy, G. McMullen, R. Henderson, S. Bellemare, and A. Granzotto, “Bigchaindb: a scalable blockchain database," white paper, BigChainDB, 2016.

[7] A. Reyna, C. Martin, J. Chen, E. Soler, and M. Diaz, “On blockchain and its integration with iot. challenges and opportunities," Future Generation Computer Systems, 2018.

[8] K. Christidis and M. Devetsikiotis, “Blockchains and smart contracts for the internet of things," IEEE Access, vol. 4, pp. 2292-2303, 2016.

[9] Pasquier, Thomas; Lau, Matthew K.; Trisovic, Ana; Boose, Emery R.; Couturier, Ben; Crosas, Mercè; Ellison, Aaron M.; Gibson, Valerie; Jones, Chris R.; Seltzer, Margo (5 September 2017). "If these data could talk". Scientific Data. 4: 170114. doi:10.1038/sdata.2017.114.

[10] Muhammad Naveed Aman, Kee Chaing Chua, Biplab Sikdar, “Secure Data Provenance for the Internet of Things”, Proceedings of the 3rd ACM International Workshop on IoT Privacy, Trust, and Security, April 02-02, 2017, Abu Dhabi, United Arab Emirates

[11] Sabah Suhail, Zuhaib Uddin Ahmad, Choong Seon Hong, "Introducing Secure Provenance in IoT: Requirements and Challenges" International Workshop on Secure Internet of Things (SIoT 2016), Sep. 26-30, 2016, Heraklion, Crete, Greece

[12] Ragib Hasan, Radu Sion, Marianne Winslett, “Introducing secure provenance: problems and challenges”, Proceedings of the 2007 ACM workshop on Storage security and survivability, October 29-29, 2007, Alexandria, Virginia, USA

[13] Rongxing Lu, Xiaodong Lin, Xiaohui Liang, Xuemin (Sherman) Shen, “Secure provenance: the essential of bread and butter of data forensics in cloud computing”, Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security, April 13-16, 2010, Beijing, China

[14] Liang, Xueping, Sachin Shetty, Deepak Tosh, Charles Kamhoua, Kevin Kwiat, and Laurent Njilla. "Provchain: A blockchain-based data provenance architecture in cloud environment with enhanced privacy and availability." In Proceedings of the 17th IEEE/ACM international symposium on cluster, cloud and grid computing, pp. 468-477. IEEE Press, 2017.

[15] Ricardo Neisse, Gary Steri, and Igor Nai-Fovino. A blockchain-based approach for data accountability and provenance tracking. In Proceedings of the 12th International Conference on Availability, Reliability and Security, page 14. ACM, 2017.

[16] H. M. Kim and M. Laskowski, “Toward an ontology-driven blockchain design for supply-chain provenance," Intelligent Systems in Accounting, Finance and Management, vol. 25, no. 1, pp. 18- 27, 2018.

[17] Chronicled, https://chronicled.com/, 2018, online; accessed February 2020.

[18] Liang, Xueping, Juan Zhao, Sachin Shetty, and Danyi Li. "Towards data assurance and resilience in iot using blockchain." In MILCOM 2017-2017 IEEE Military Communications Conference (MILCOM), pp. 261-266. IEEE, 2017.

[19] Liang, Xueping, Sachin Shetty, Deepak K. Tosh, Juan Zhao, Danyi Li, and Jihong Liu. "A Reliable Data Provenance and Privacy Preservation Architecture for Business-Driven Cyber-Physical Systems Using Blockchain." International Journal of Information Security and Privacy (IJISP) 12, no. 4 (2018): 68-81.

[20] Biswas, Sujit, Kashif Sharif, Fan Li, Boubakr Nour, and Yu Wang. "A scalable blockchain framework for secure transactions in IoT." IEEE Internet of Things Journal 6, no. 3 (2018): 4650-4659.

[21] Ali, Saqib, Guojun Wang, Md Zakirul Alam Bhuiyan, and Hai Jiang. "Secure data provenance in cloud-centric internet of things via blockchain smart contracts." In 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/ SCALCOM/ UIC/ ATC/CBDCom/IOP/SCI), pp. 991-998. IEEE, 2018.

[22] Zhang, Yunru, Debiao He, and Kim-Kwang Raymond Choo. "BaDS: Blockchain-based architecture for data sharing with ABS and CP-ABE in IoT." Wireless Communications and Mobile Computing 2018 (2018).

[23] Javaid, U., Aman, M.N. and Sikdar, B., “Blockpro: Blockchain based data provenance and integrity for secure iot environments”. In Proceedings of the 1st Workshop on Blockchain-enabled Networked Sensor Systems (pp. 13-18).

[24] Robert Ikeda and Jennifer Widom. “Data lineage: A survey”, Technical report, Stanford University, 2009.

[25] Y. Cui and J. Widom. “Lineage tracing for general data warehouse transformations”, VLDB Journal, 12(1), 2003.

[26] Cachin, Christian. "Architecture of the hyperledger blockchain fabric." In Workshop on distributed cryptocurrencies and consensus ledgers, vol. 310, p. 4. 2016.

[27] Sousa, J., Bessani, A. and Vukolic, M., 2018, June. A byzantine fault-tolerant ordering service for the hyperledger fabric blockchain platform. In 2018 48th annual IEEE/IFIP international conference on dependable systems and networks (DSN) (pp. 51-58). IEEE.