Dynamic Secure Access Control and Data Sharing Through Trusted Delegation and Revocation in a Blockchain-Enabled Cloud-IoT Environment

Suhair Alshehri, Omaimah Bamasaq, Daniyal Alghazzawi, *Senior Member, IEEE*, and Arwa Jamjoom

***Abstract*—The Internet of Things (IoT) is vulnerable to leakage of private information during data sharing. To avoid this problem, access control and secure data sharing have been introduced in IoT; however, many challenges are faced because of centralized access control and single delegator selection. Additionally, blockchain is integrated into IoT to enhance the security of the environment. For that purpose, this research proposes dynamic secure access control using the blockchain (DSA-Block) model, which performs secure access control and data sharing. Initially, the IoT device attributes and user attributes are registered at a local domain authority (LDA) for generating private and public keys using the hyperelliptic curve cryptography (HECC) algorithm, which ensures the legitimacy of the users and devices. Then, the IoT devices send a request message to the edge nodes (ENs) via a gateway, which performs request filtration by validating the user's authenticity. The filtered requests are sent to the edge server to perform access delegation using rock hyraxes swarm optimization (RHSO), which selects a set of delegator nodes. The access control decision is made by using the Trusted practical Byzantine fault tolerance (PBFT) consensus algorithm. The IoT data are stored in the cloud server for secure storage, in which the data are secured using a differential privacy mechanism. Finally, dual revocations, such as user attribute revocation and user revocation, are used to maintain security. The performance of DSA-Block is evaluated and the results demonstrate that the proposed DSA-Block model achieves superior performance compared to previous works.**

***Index Terms*—Access delegation, blockchain, edge computing, Internet of Things (IoT), practical Byzantine fault tolerance (PBFT) consensus, revocation, secure data sharing.**

I. Introduction

THE EMERGENCE of the Internet of Things (IoT) technology in recent times has provided connectivity between several devices to facilitate various applications, such as smart homes, smart manufacturing, and smart transportation [1], [2]. The privacy and security issues associated with this development necessitate a mechanism for controlling these devices [3]. Access control is found to be the most significant mechanism to control the interventions of illegitimate and unauthorized users [4], [5]. Several access control schemes have been proposed to control the access to resources, but these approaches rely on a centralized entity, which results in a single point of failure [6]. The adversaries in the network compromise the centralized server to redefine the access policies in their favor. Moreover, the data stored in the compromised centralized entity represent serious privacy threats, as the personal information of the IoT users, such as location and surveillance data are exposed to the malicious adversary [7].

The decentralized access control in the IoT network is provided by the blockchain technology which mitigates the limitations of centralized architectures and provides services for large-scale scenarios [8], [9]. Multiauthority-based access control is applied in several existing works to improve secure data sharing, however, it leads to significant privacy issues [10], [11], [12]. Several existing approaches integrate the blockchain to provide decentralized access controls by storing the access policies in the blockchain node and making decisions based on the access policies [13]. In these approaches, the blockchain node acts as a delegator and is used to process the overall resource requirements of the IoT entities [14]. The selection of a single blockchain node to provide access control to devices in the IoT network represents various threats, as follows.

1) *Security Threats:* The single blockchain node that is responsible for validation of all user requests is vulnerable to various Distributed Denial-of-Service (DDoS) attacks such as when the malicious adversary produces a large volume of access requests in a periodical manner to waste the resources of the blockchain node.
2) *Privacy Threats:* The access control is carried out by the single public blockchain node by verifying the access policies and attributes stored in the blockchain, which poses privacy issues for the attribute owners [15], [16].

The privacy issues of the public blockchain have been overcome by the deployment of Hyperledger Fabric, which is an open-source platform that possesses several significant advantages, such as an effective consensus mechanism, decentralized ledgers to protect privacy, and increased

Manuscript received 7 May 2022; revised 10 August 2022 and 21 September 2022; accepted 9 October 2022. Date of publication 26 October 2022; date of current version 20 February 2023. This work was supported by the Institutional Fund Projects Under Grant IFPRC-114-612-2020. Therefore, the authors gratefully acknowledge technical and financial support from the Ministry of Education and King Abdulaziz University, Jeddah, Saudi Arabia. *(Corresponding author: Suhair Alshehri.)*

Suhair Alshehri is with the Department of Information Technology, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 22254, Saudi Arabia (e-mail: sdalshehri@kau.edu.sa).

Omaimah Bamasaq is with the Department of Computer Science, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 22254, Saudi Arabia (e-mail: obamasek@kau.edu.sa).

Daniyal Alghazzawi and Arwa Jamjoom are with the Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah 22254, Saudi Arabia (e-mail: dghazzawi@kau.edu.sa; ajamjoom@kau.edu.sa).

Digital Object Identifier 10.1109/JIOT.2022.3217087

throughput [17]. Edge-based blockchain is implemented to perform access control with low latency and computational complexity using Hyperledger Fabric [18]. However, the proper usage of Hyperledger Fabric for effective access control has not yet been efficiently performed [19]. The revocation process is implemented to revoke malicious users or attributes, to increase security efficiency. However, this can lead to various attacks due to a lack of time-based revocation [20]. In this article, secure access control-based sharing of data is carried out in a decentralized manner.

A. Motivation and Objectives

The major aim of this research work is to perform dynamic access control and data sharing in a decentralized manner. The trust-based selection of delegator nodes is performed to achieve effective consensus. The delegation and revocation of access rights are executed in a periodic way to maintain security. Various issues in previous works have motivated this research and are addressed within it, as follows.

1) *Inefficient Access Management:* The existing approaches have integrated the access control with the blockchain technology to give it a decentralized nature, but the selection of a single delegator (e.g., cloud) increases the vulnerability of users and risks of leakage of private data.
2) *Single Point of Failure:* Blockchain-based access control has been utilized to provide decentralized computation, but several existing approaches have used gateway (GW)-based access control, which results in a single point of failure when the number of requests or users in the network increases.
3) *Complex Cryptography:* The existing approaches have ensured the security of IoT nodes by utilizing cryptographic algorithms such as elliptic curve cryptography (ECC), which possess increased message size and computational complexity that are not suitable for resource-constrained IoT devices.

The major objective of this approach is to ensure security and privacy in the IoT network. This objective is achieved by satisfying a set of subobjectives, as follows.

1) To minimize the effect of adversarial access requests in overloading the entity by performing authentication-based request filtration.
2) To maximize the scalability and performance of blockchain by achieving increased transactional throughput and reduced block validation time.
3) To maximize the reputation of the consensus by considering the trust value in selecting the consensus node.
4) To maximize the security by performing revocation of the attributes and users based on time and behavior.

B. Research Contributions

The proposed dynamic secure access control using the blockchain (DSA-Block) model focuses on secure access control and secure data sharing using blockchain. The major contributions of this research are listed as follows.

1) Authentication of nodes and users is carried out via hyperelliptic curve cryptography (HECC) and the entities are stored in the private local ledger (LL) to enhance the security, which helps to mitigate external attacks.
2) Authentication-based request filtering is performed through a GW by verifying the legitimacy with a timestamp and freshness of the requests, which increases throughput and reduces latency.
3) Access delegation is achieved through the edge server using rock hyraxes swarm optimization (RHSO) by considering trust, energy, load, and resource availability (RA), in which the trust value is evaluated using blockchain, which reduces the block validation time, response time, and consensus time.
4) The data are shared securely manner by uploading the data to the cloud server with the help of a differential privacy mechanism, which increases the attack detection rate.
5) Finally, revocation is performed for both user attributes and users to enhance security. The user attribute revocation is performed by considering expiry time and attributes updating, and user revocation is performed based on the trust value, which also increases the attack detection rate.

The performance of the proposed work is evaluated in terms of consensus time, throughput, block validation time, transaction time, latency, response time, and attack detection accuracy.

C. Article Organization

This section outlines the structure of this article that follows. Section II summarizes the literature review of previous works. Section III explains the preliminaries of this research which is followed by the major problem statement in Section IV. Section V presents the system model of the proposed DSA-Block approach and its experimental results are described in Section VI. Finally, Section VII concludes the research and recommends future work.

II. Literature Survey

This section summarizes the literature review on the existing methods, where different approaches have been used to perform secure access control based on attributes. It also outlines the key research gaps in these works.

A. Access Control Schemes

The decentralized authorization of IoT devices was proposed by [21]. The necessity of updating policies was addressed by implementing a collaborative-based access control approach in which the new access policy collaborated with the old one using a collaborative node. The system model is comprised of entities, such as authority nodes, IoT nodes, and blockchain nodes. Initially, the IoT devices were authenticated by the hash ID using the ECC algorithm. The authorization of nodes was carried out using the access tree, in which the construction and reconstruction of a collaborative node were carried out by verifying the blockchain ledger. The generation

of private and public keys was carried out based on the ECC algorithm, which provided increased security due to its asymmetric nature. However, the increased size of the encryption key affects the complexity of these approaches. The privacy of IoT data stored in the cloud was ensured by proposing attribute-based encryption (ABE) in [22]. The vulnerabilities inherent to data being kept in cloud storage, such as privacy leakage, data tampering, and illegal access were considered, and an effective approach was introduced. The centralized server utilized in the conventional approaches was replaced with the decentralized blockchain approach. The identity of the users in the IoT network was preserved by using the asymmetric encryption algorithm and the keys were generated by the consensus nodes. Revocation of users in the network was carried out periodically and the key updating of the revoked users was executed. When the user requests data kept in the cloud, the cloud performs predecryption by retrieving the keys from the blockchain, which reduces the computational cost of the users. The consensus nodes were responsible for generation of keys and revocation of users in the network, but the selection of consensus nodes in the blockchain is carried out without considering the reputation of the nodes, which affects the credibility of the approach. Symmetric key encryption was used to ensure the authenticity of the IoT users, but the usage of a single key by the IoT nodes makes it vulnerable to being compromised.

A flexible access control mechanism using ABE was proposed in [23]. The limitations of existing nonrevocable access control approaches were addressed by introducing a blockchain-based revocable approach. This system model was comprised of entities, such as trusted authority (TA), publisher, user, and miners. Initially, the registration of users along with their attributes was performed, after which the chameleon hash algorithm was used to generate the hash function for verification. The revocation of attributes was performed by the TA if the user attributes were found to be changed. Based on the new set of attributes, the access control for the respective user was performed. The authors stated that the decentralization of the approach can be achieved by deploying multiple Tas, but the matching of users to a respective TA was not performed, which increases the overload of a particular TA in the network. The authors performed multifactor-based access control of IoT users in [24]. The limitations of existing single-factor access control mechanisms were overcome by utilizing three factors: 1) user password; 2) biometrics; and 3) the keys provided by the TA for access control. Initially, the key generation for TA was computed based on the elliptical curve discrete logarithm, and the registration and enrolment of users and GW nodes were carried out to perform access control. The data sharing was performed after the establishment and verification of the session key through mutual authentication of the GW node and IoT users. Finally, the updating of security parameters was performed in the user's favor. The generation of private and public keys was carried out based on the ECC algorithm, which provided increased security due to its asymmetric nature. However, the increased size of the encryption key increases the complexity of this approach.

A simplified access control approach for an IoT environment was presented in [25]. The limitations of traditional centralized access control approaches, such as single point of failure and data tampering, were overcome by a distributed approach. In this approach, the consortium blockchain was utilized in which the identity of the user is denoted as the address. The generation of public and private keys was carried out via the ECC algorithm. The practical Byzantine fault tolerance (PBFT) consensus was utilized for the verification of access requests and revocation of users was carried out to ensure the security of the approach.

Fog computing-based distributed management of keys was carried out in [26]. The limitations of cloud-based access control, such as unreliability, were overcome by this approach. The system model is comprised of hierarchical deployment of entities, such as IoT devices, edge nodes (ENs), a security access manager (SAM), and cloud-based blockchain nodes, respectively. The initialization of the users was performed based on several parameters. The accessed query and access response was performed by the SAM presented in the fog layer. To ensure the security of the IoT environment, periodic updating of keys, and revocation of users was carried out. The generation of blocks was achieved by using proof of work consensus; however, the time required for generation of blocks and reduced throughput provided by this approach restricts the usage of this consensus.

Zhang et al. [27] proposed the key-policy ABE (KP-ABE) scheme using the decisional learning with error (DLWE) problem and combined the KP-ABE with blockchain for secure access management. In KP-ABE, the researchers used probabilistic polynomial time (PPT) algorithms, including setup, keygen, encryption, and decryption. Through the combination of blockchain and KP-ABE scheme, access management was facilitated between the owner and their device and then the user. In the access management scheme, first initialization takes place for the registration process and then the transaction process occurs when a transaction request is received. By using miners, the transaction is verified by a PoS consensus mechanism before the block is added to the blockchain. The blockchain gives access permission to users. Finally, the DLWA method was used to prove the secure access management using the KP-ABE scheme. Here, the authors proposed a KP-ABE scheme for secure access management by using blockchain. The efficiency of the access management was considered, but not the security issues in the ABE scheme, which reduces the degree of security for IoT.

Ali et al. [28] proposed an ABE scheme with a lightweight revoke method. By using a central authority (CA), domain authorities (DAs), cloud service provider (CSP), data owner (DO), and data user (DU), five processes were implemented with PPT algorithms. First, using a cryptographical background, the authors initialized the CA and DAs. In the key delegation process, key generation for the DO and DU was achieved using the access tree. In the data outsourcing process, the partially encrypted data from the DO was encrypted in CSP. Then, the encrypted data was partially decrypted in CSP and fully decrypted in DU during the decryption process. In the user revocation process, the DAs updated the parameters

and the re-encrypted secret key was updated for the users to allow or revoke access. Finally, for the proofing process, the Decisional Bilinear Diffe–Hellman problem was used, with the hardness assumption. Here, the authors proposed a hierarchical ABE scheme for secure access control for IoT. However, there is a probability of data tampering due to low data integrity and user privacy.

B. Blockchain-Based Security Schemes

The secure sharing of data in the IoT environment using identity-based encryption was proposed in [29]. The limitation of an ABE process, such as high time consumption, was addressed by implementing a proxy-based re-encryption approach in which the EN acts as a proxy server to perform re-encryption of the data. The DO is responsible for the generation of keys for each user based on their identity. The users submit their keys along with a request, based on which the user verification was performed by the blockchain-assisted TA. The PBFT consensus was utilized to achieve consensus for block generation in order to reduce the time required for the encryption and decryption processes. However, the generation of keys for each user in the network becomes complex when the number of users in the network increases. The PBFT consensus was utilized in this approach to achieve a consensus about the final decision, but the proposed consensus model was found to have scalability issues and increased block validation time. Kumar et al. [30] proposed a secure machine learning-based framework for ensuring trustworthiness in an IoT environment. The smart city application was adopted, in which three modules, namely trustworthiness, privacy, and intrusion detection, were presented. In the first module, the computation of a reputation score for the IoT devices was carried out and, based on that, the nodes were categorized into three clusters. The reputation score and transaction of the nodes were further stored in a blockchain. The off-chain storage of raw data was carried out in an interplanetary file system (IPFS), in which the privacy of the data was ensured by storing its hash values in the blockchain. The blocks were generated using enhanced proof of work consensus. Finally, intrusion detection was performed by utilizing the XGBoost algorithm. The generation of blocks was carried out using enhanced proof of work consensus, but the time required for block generation and the reduced throughput provided by this approach restricts the usage of this consensus.

A trust-based provision of access to IoT services was presented in [31]. The limitations of combining blockchain technology and trust management were overcome through this approach. The trust values were computed based on processes such as service testing, service monitoring, and service rating. The trust computation was carried out between the IoT users for provision of services. This is facilitated by the trust consensus protocol, in which the trust metric is computed for processes, such as leader selection, block generation, and validation of blocks. The trust computation was based on the service testing, service monitoring, and service rating parameters, but the individual nodal trust of the IoT devices was not considered, which limits the effectiveness of the trust-based consensus protocol.

Ali et al. [32] proposed an approach named xDBAuth for permission delegation and access control in a cross-domain context, by using four operations. Four transactions, such as T.register, T.access, T.delegate, and T.revoke, were used for the following operations. The first operation was domain registration in a global smart contract. In these various domains, the admin sent registration requests to the blockchain manager through the T.register transaction, followed by user or IoT device registration in a local smart contract in the same T.register transaction. In this operation, they assumed that the user had installed TPM and registered the user or IoT device in a local smart contract through the transaction. The third process was to publish the delegation policy through the T.delegate transaction. The last operation was resource access, which was done through the T.access transaction, where the cross-domain authentication was performed by a PoAI mechanism. Here, the authors proposed a PoAI mechanism approach for cross-domain authentication and performing the hash function using a sha-256 algorithm. However, when using this algorithm for hash generation, there is a difficult to find value, which was computed as the liable hash value. Local and global smart contracts were used for blockchain to carry out the transactions, but the delegate policies were stored in the off-chain due to storage limitations in the blockchain. This reduces the storage complexity, but leads to privacy issues and there is a probability of changing or rewriting the delegate policies.

Qashlan et al. [33] proposed an approach to preserve privacy by using blockchain. Register contract and access contract were used in a smart contract. Initially, the end-user sends an access request to the edge server; the server then redirects the user to the smart contract and the user calls the smart contract to check its validity. If valid, the user policies are then checked using ERC20 for attributes and the generating token if its attributes satisfy the policy. By using the token, the user sends the access request to the smart contract via the edge server and gets access to the IoT devices. Finally, the authors used a differential privacy enhancement mechanism with stochastic gradient descent to ensure safety of the data by adding noise samples without considering privacy. Here, the authors used the PoW mechanism in blockchain for consensus; however, this leads to a Sybil attack and other attacks.

Wang et al. [34] proposed a Proof of X-Repute (PoXR) mechanism for the consensus process in blockchain for IoT. For rapid and safe consensus, they used two methods: 1) repute rewards and 2) repute punishments. Initially, the block production method was used to create a new block by considering the initial reputes, miner ID, and other related parameters. Then, the generated block undergoes a verification process by using the fork selection method and reputation module with parameter characteristics of the consensus algorithm. Finally, using the incentive method, reputation status was identified by giving the respective parameters to improve the consensus process in the blockchain. In this work, the PoXR mechanism was used to select the node for consensus to update the blockchain ledger. However, in PoXR, if more nodes are able to perform consensus, this leads to a computational overhead.

Putra et al. [35] proposed an authorization approach for IoT in blockchain by using trust scores and reputation values. Here,

TABLE I
SUMMARY OF THE RELATED WORK

Schemes	References	Objectives	Algorithms/Methods Used	Advantages	Disadvantages
Access Control Schemes	[21]	Decentralized Authorization for IoT devices	Collaborative based access control approach, ECC algorithm	High Security due to hash-based authentication	High complexity due to large key size
	[22]	Attribute-based privacy-preserving of IoT data	Decentralized blockchain approach, symmetric encryption algorithm	Low Computation cost due to pre-decryption	Low credibility by lack of reputation
	[23]	Attribute-based revocation and access control	Chameleon hash algorithm	Flexible access control due to attribute-based encryption	Maximum overloading of TA due to lack of user matching
	[24]	Three-factor based access control	Key generation using elliptical curve discrete algorithm, mutual authentication,	High security due to mutual authentication	High complexity due to key generation method
	[25]	Attribute-based access control in IoT environment	Consortium blockchain, Elliptically curved cryptography algorithm	Low single point of failure due to distributed approach	Large key size increases the complexity
	[26]	Fog computing based access control	Fog layer, blockchain	High reliability due to hierarchical development	High time consumption due to slow block generation
	[27]	Access management by attribute-based encryption	Decisional learning with error, probabilistic polynomial algorithm	High authenticity due to POS consensus	Low security
	[28]	Revocation using attribute-based encryption	Probabilistic polynomial-time algorithm, decisional bilinear Diffe-Hellman problem	Low latency due to partial encryption and decryption	Low data integrity
Blockchain-based Schemes	[29]	Secure data sharing in IoT	Proxy-based re-encryption approach	Low time consumption due to proxy-based re-encryption	Low scalability due to key generation complexity
	[30]	Privacy-preserving in IoT environment	XGBoost Algorithm, Proof of Work consensus	High privacy due to IPFS-based off-chain data storage	High block generation time
	[31]	Secure service provisioning for IoT	Blockchain, trust management, trust consensus protocol	Low complexity	Poor trust management due to lack of individual trust of IoT devices
	[32]	Cross domain-based permission delegation	PoAI mechanism	High security due to cross-domain permission	Low privacy
	[33]	Privacy-preserving	ERC20, stochastic gradient descent, proof of work mechanism	High reliability due to edge-based access control	High-security threats
	[34]	Blockchain-based Consensus for IoT environment	Proof of x-repute mechanism, fork selection method	High data integrity due to safe consensus	High computation overhead when occurring a large number of nodes
	[35]	Authorization for IoT	Asymmetric cryptography	High time consumption due to two blockchains	Lack of privacy

two blockchains were used, public and private blockchains. Initially, service providers (SPs) and service consumers (SC) were registered in an attribute authority (AA) based on their attributes and received key pairs by using an asymmetric cryptography mechanism. In the public blockchain, the trust score was found by considering the previous interactions between the SP and SC, and the reputation value was found by using the trust score. Whenever an access request arrived from SC, the request was validated in the public blockchain based on the trust value and its corresponding reputation. If the request was valid, the attributes were retrieved from the private blockchain a token was created for SC, and the data storage system validated the token and allowed access to the data. Here, the authors proposed an authorization in blockchain approach for IoT using a trust score to achieve privacy. However, at the time of registering, the SC and SP gave the attributes to the AA in an off-chain scenario, which leads to a lack of privacy.

Table I provides a summary of the related works indicating the algorithms/methods used, advantages and disadvantages.

III. PRELIMINARIES

This section describes the preliminaries of this research, which consist of the blockchain technology, Hyperledger Fabric, and PBFT consensus.

A. Blockchain Technology

The security of IoT entities is provided by the decentralized nature of the blockchain technology. The data stored within the blockchain are stored in a hashed manner, as blocks. The communications between each block are stored as transactions. The blockchain integrity is ensured by running consensus for block creation, with the help of miners. The miners validate the transactions and create the block. Once a new block has been created, it is broadcast to other blocks. The generated block contains a header and a body. The header contains versions of the block, the hashed value of the transaction using the Merkle hash root tree, the new block threshold value, nonce, and previous block information. The body of the block contains a transaction counter. The transaction size limits the size of the block. Some of the features of the blockchain are listed as follows.

1) Faster execution.
2) Immutability.
3) Consensus.
4) Distributed ledgers.
5) Decentralized.
6) Highly secure.

Based on the size of the user data, they can be stored either in the blockchain or the off-chain. The nontransactional data are stored in the off-chain, being too large to store in the

blockchain. Off-chain storage is lower cost, and ensures better privacy than the blockchain. The off-chain distributed system for storing off-chain data is IPFS.

B. Hyperledger Fabric

Hyperledger Fabric is the prefabricated architecture in blockchain that permits the IoT nodes to run consensus based on a plug-play mechanism for preserving privacy. The Hyperledger Fabric compromises the following components: chain code, peer node, channel, membership SP, devoted assembling service, and ledger. The three main components of Hyperledger Fabric are the devoted ordering service, membership SP, and peer nodes.

1) *Devoted Assembling Service:* The assembler in the network is responsible for effective communication which periodically checks the state of the ledger. Any of the consensuses in the Hyperledger Fabric are proven by the assembler to sustain the order of transactions.
2) *Membership Service Provider:* Membership SPs are responsible for ensuring the authenticity of the authorized blockchain network, which also generates private channels for communication among blockchain nodes.
3) *Peer Nodes:* Peer nodes are responsible for upholding the ledgers. Generally, the Hyperledger Fabric contains broadcaster peer nodes and supporter peer nodes. The broadcaster peers are distributed in the network and broadcast the block to other peers, whereas supporter peers are used to validate user requests.

The Hyperledger Fabric contains policies to maintain the nodes in a decentralized manner to reduce the scalability and mitigate security threats. Two types of policies, namely, implicit meta-policy and signature policy, provide effective access control via consortium.

C. Practical Byzantine Fault Tolerance Consensus

The PBFT is a consensus algorithm based on a voting procedure; it can allow unauthorized nodes, i.e., mischievous nodes to take part in the consensus algorithm. The PBFT consensus maintains safeness and liveliness among the blockchain nodes. Safeness refers to the case where, if any of the services are repeated by some nodes, it must satisfy the correctness condition, and liveliness refers to nodes' behavior based on the request. Every node in the PBFT network communicates through cryptographic methods to prove its integrity, and also to reach the state of consensus. The steps involved in PBFT consensus are listed as follows.

1) Demand.
2) Preprepare.
3) Prepare.
4) Obligate.
5) Block formation.

Table II presents a comparison of conventional consensus and PBFT consensus. Fig. 1 presents the block formation in the blockchain-based on the PBFT consensus mechanisms in which information in the workings of the PBFT consensus is also provided.

TABLE II
COMPARISON OF EXISTING CONSENSUS VERSUS PBFT

Characteristics	**PoW**	**PoS**	**PBFT**
Throughput	< 100	> 1000	< 2000
Rate of transaction	Low	High	High
Energy redeemable	No	Partial	Yes
Speed of Verification	$> 100s$	$< 100s$	$< 10s$
Level of trust	Untrusted	Untrusted	Semi-trusted
Block creation speed	Low	High	High
Scalability of transaction	Low	Low	High
Overhead during computing	High	Low	Low
Example	Bitcoin	Ethereum	Hyperledger

IV. PROBLEM STATEMENT

The security and privacy of the IoT users are a major concern, and several existing approaches have executed various schemes to achieve the requirements. However, they possess several limitations that affect the achievement of security and privacy in the IoT environment. The major problems with these approaches are described as follows.

The scalable access control mechanism based on blockchain for a resource-constraint IoT environment was proposed in [36]. The limitations in the selection of a single delegation node for access delegation were addressed by implementing a lightweight blockchain scheme in several IoT nodes. The major problems with this research are defined as follows.

1) The identity-based authentication of devices, users, and ENs was carried out, but the parameters considered for authentication possess a low degree of security, which affects the secure authentication of nodes and users.
2) The selection of policy decision points was performed based only on the parameters such as task execution and use of resources, but the lack of consideration of trust-based parameters affects the integrity of the final access decision.

The management of access control in an IoT environment based on Hyperledger Fabric was presented in this article [37]. The limitations of the centralized access control mechanisms were addressed by this approach. The hierarchical structure of access control in the IoT environment was proposed in this article [38]. The limitations of traditional access control mechanisms in providing flexible access to IoT resources were addressed by utilizing an ABE mechanism. The major problems with these researches are explained as follows.

1) The Hyperledger Fabric utilized Kafka consensus for block generation due to its performance, but the Kafka consensus does not provide resistance against the malicious nodes present in the network.
2) In the Fabric-IoT approach, the access policy of the resources was delegated by the devices, but the revocation and updating of users were not performed, which affects the adoption of this approach.
3) The data consumer sends the access requests directly to the centralized blockchain node, but the increased number of access requests generated by the malicious users will cause an overload of the blockchain, which affects the proper provision of access.

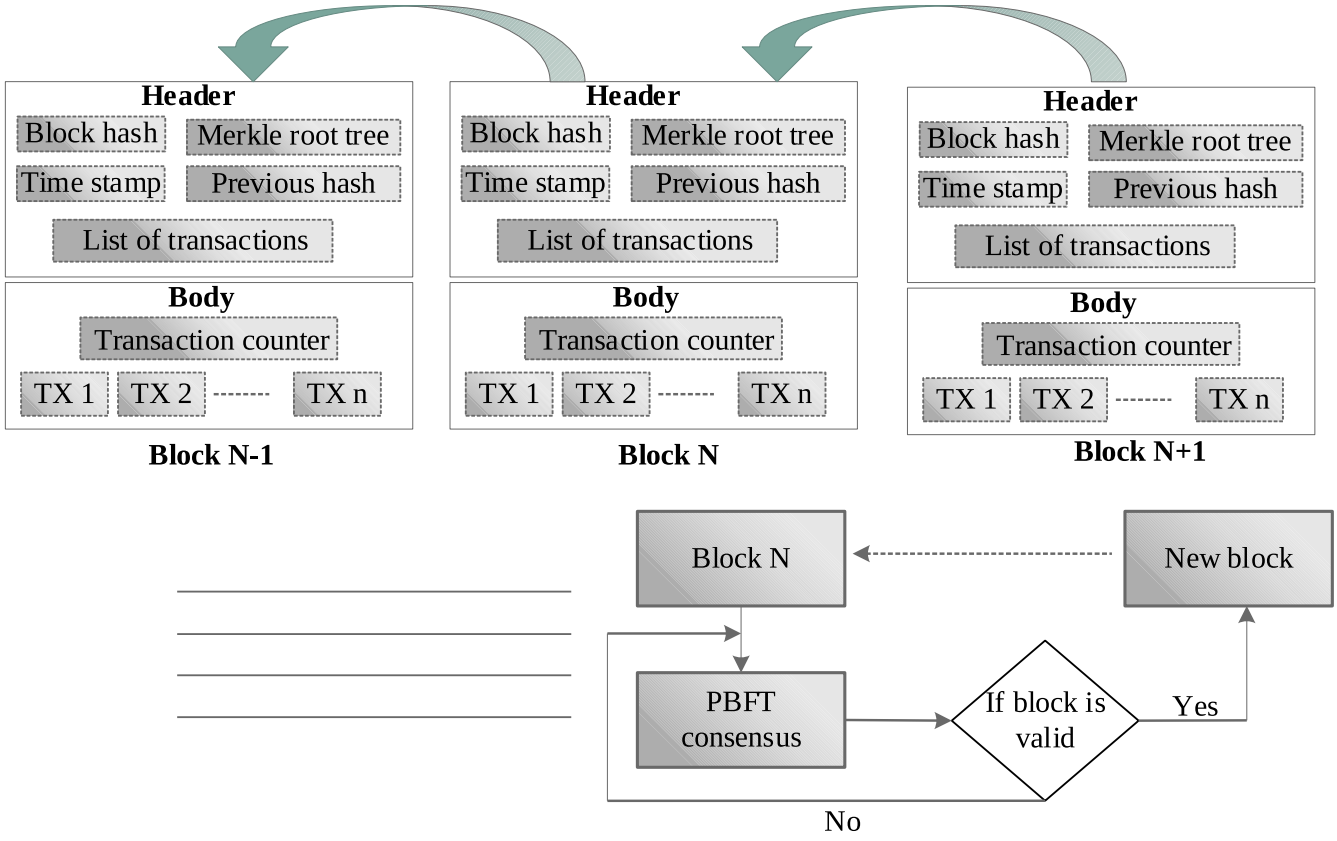

Fig. 1. Blockchain-based PBFT consensus.

The GW-based architecture for providing access control in the IoT environment was proposed in this article [39]. The limitations of the traditional access control approaches were analyzed and a decentralized approach named BorderChain for management of access to the IoT domains was proposed. The major problems with this research are listed as follows.

1) The proposed GW-based architecture performed well in ensuring the security of the IoT users and devices within a domain, due to the trust provided by vendors and ISPs, but the single point of failure occurring in these entities affects the usefulness of this approach.
2) In addition, the IoT GW was utilized as the blockchain node, but the presence of malicious users in the network creates an increased number of access requests, which wastes the resources of the blockchain nodes.

The blockchain-based delegation of access for the IoT users was carried out in this article [40]. The major problems with this research are defined as follows.

1) The delegator node, which is considered the broker, was responsible for delegation of access to the users, but this node possesses a high risk of a single point of failure, as attackers in the network use it to affect the proper delegation of access.
2) The provision of access based on user attributes was carried out in this approach, but the changing nature of user attributes was not considered which affects the access provision process.
3) Access provision was performed based on the smart contract, but the process involved in access provision and consensus of both the public and private blockchain was not mentioned.

A. Research Solutions

The aforementioned problems are addressed by providing the following solutions.

1) The authentication of devices, users, GWs, and ENs was carried out based on several significant user attributes and devices attributes that increase the degree of security of the nodes and users. Private and public keys are generated by using the HECC algorithm, which enables reduced key size without any compromise on security. This algorithm is suitable for resource-constraint IoT environments.
2) The hierarchical architecture-based approach is implemented, in which the decentralized management of authorization is performed by using both a global domain authority (GDA) and a local domain authority (LDA).
3) The delegator nodes are selected by using the RHSO algorithm for the trust value, energy level, traffic load, and RA; this contributes to the effective selection of nodes.
4) Access control is executed by the consensus operation using the selected delegator nodes, which increases the network scalability. The burden of the GW is reduced by initially filtering the requests.
5) The Trusted PBFT is utilized, in which trusted nodes are selected for consensus. The number of nodes participating in the consensus is restricted to a particular value

based on the number of nodes. This provides resistance to malicious nodes and also reduces the block validation time.

6) The dual revocation is executed, in which the revocation of attributes is carried out based on the attribute expiry time, and the revocation of users is performed based on the trust threshold value.
7) The overloading and resource wastage of blockchain nodes is mitigated by filtrating incoming requests. The authenticity and timestamps of the requests are validated to ignore the malicious requests.

V. System Model

This article focuses on providing security and privacy of the data and devices in the IoT environment. The proposed DSA-Block model comprises the following entities: cloud node (CN), EN, GWs, IoT users (U^I), and IoT devices (D^I). The Hyperledger Fabric is utilized in this approach due to its private and permissioned nature. Blockchain is used to select the multiple IoT delegator nodes by running the PBFT consensus mechanism.

The IoT nodes are categorized into domains and each domain possesses an individual LL. The LDA is placed in each domain and the GDA is placed in the cloud. The entities are used in the DSA-Block model are summarized as follows.

1) *IoT Devices (D^I):* The D^I are the DOs, in which they are classified into LDA or GDA. The LDA is used for providing authenticity to the IoT device and IoT users. The GDA is for storing the LDA transactions and providing delegation or revocation to the IoT users.
2) *IoT Users (U^I):* All the U^I are data customers, they are requesting access to D^I. Based on their respective requests, access is provided to U^I.
3) *Gateway:* GWs are located between U^I, D^I, and ENs, which ensure the legitimacy of the U^I by storing the U^I and D^I attributes in the LDA thereby mitigating DDoS attacks.
4) *Edge Nodes:* The ENs provide access delegation and revocation to the U^I by running PBFT consensus based on trust; they also manage the network.
5) *Cloud Node:* The CNs are used for storing the data of D^I, which contains GDA for managing the D^I data.

Fig. 2 presents the architecture of the proposed DSA-Block model, which shows all the processes of the proposed work in a detailed manner.

A. System Initialization

The entities in the IoT network are initialized to perform secure data sharing. There are several entities in the network, including IoT users (U^I), IoT devices (D^I), GW nodes, and ENs inside the domains, which are initialized by submitting their parameters to the LDA. The user attributes (U^{Ia}) include user ID (U^{ID}), position (pos), role (role), etc. The device attributes (D^{Ia}) include Device ID (D^{ID}), MAC address (MAC), IMEI number (IMEI), etc., GW nodes, and ENs are submitted to the respective LDA.

Algorithm 1 Pseudocode of the System Initialization

Begin
  Initialize $\mathrm{U_i^I}$, $\mathrm{D_i^i}$, $\mathrm{GW_i}$, $\mathrm{EN_i}$, LL, LDA Where $i = 1, 2, 3 \ldots n$
  Initialize ($\mathrm{U^{Ia}}$ = {$\mathrm{U^{ID}}$, pos,role }, $\mathrm{D^{Ia}}$ = {$\mathrm{D^{ID}}$, MAC, IMEI}
  For all $\mathrm{U_i^I}$, $\mathrm{D_i^i}$, $\mathrm{GW_i}$, $\mathrm{EN_i}$ **do**
    Register Attributes to LDA ();
    Store Attributes in LL;
    // Key Generation
    Select and $\acute{\upsilon}$ // prime and divisor
    $I_R \in S$;
    $\mathrm{K_S} \leftarrow [I_R]\acute{\upsilon}$;
    Return I_R and κ //private and public key
    Generate attribute code using I_R and κ
  End For
End

The LDA stores these attributes in the private LL and generates the public and private keys for the users and devices using HECC. Initially, consider $\hat{\mathrm{G}}$ as the field and Ç as algebraic closure for $\hat{\mathrm{G}}$ with genus υ, and it is ≥ 1 for the hyperelliptic curve. The polynomial $k(m) \in \hat{\mathrm{G}}[m]$ of a degree greater than or equal to υ, and monic polynomial $f(m) \in \hat{\mathrm{G}}[m]$ of degree $2\upsilon + 1$. The hyperelliptic curve (∂) is formulated as follows:

$$\partial : \Phi^2 + k(m)\Phi = f(m). \tag{1}$$

Solution set generation $(b, d) \in \text{Ç} * \text{Ç}$ is performed using the curve points ∂ known as Jacobian with the quotient $\lambda = \acute{}\upsilon / \Phi$, where $\acute{}\upsilon$ denotes the divisor. It consists of a summation of points Þ$\in \partial$. The curve is nonsingular if it does not have many pairs of $(b, d) \in \text{Ç} * \text{Ç}$. In addition, it satisfies the curve equation with a Partial Differential Equation, which is expressed as follows:

$$2\Phi + k(m) = 0 \tag{2}$$

$$k'(m)\Phi - f'(m) = 0. \tag{3}$$

The LDA further generates the attribute code for the attributes along with the expiry time for respective users based on the private and public keys of the users. The system initialization is briefly explained by the pseudocode, which is described in Pseudocode 1.

B. Authentication-Based Request Filtration

Users requesting real-time access to the IoT devices send their request message to the EN through the GW. The GW performs the request filtration process to mitigate DDoS attacks caused by illegitimate user requests. Initially, the GW verifies the authenticity of the users by which the requests have been generated using filters. At first, the filter starts with an empty array with m bits. It returns false when the element is verified for membership in the filter. To add the element to the filter, the element is first accepted through the hash function. Here, the SHA-256 algorithm is used to perform hashing. The hash function result is used to take a position in the array of the filter. The position of the bit is assigned to one. To verify the element in the filter, the hash functions are used to create

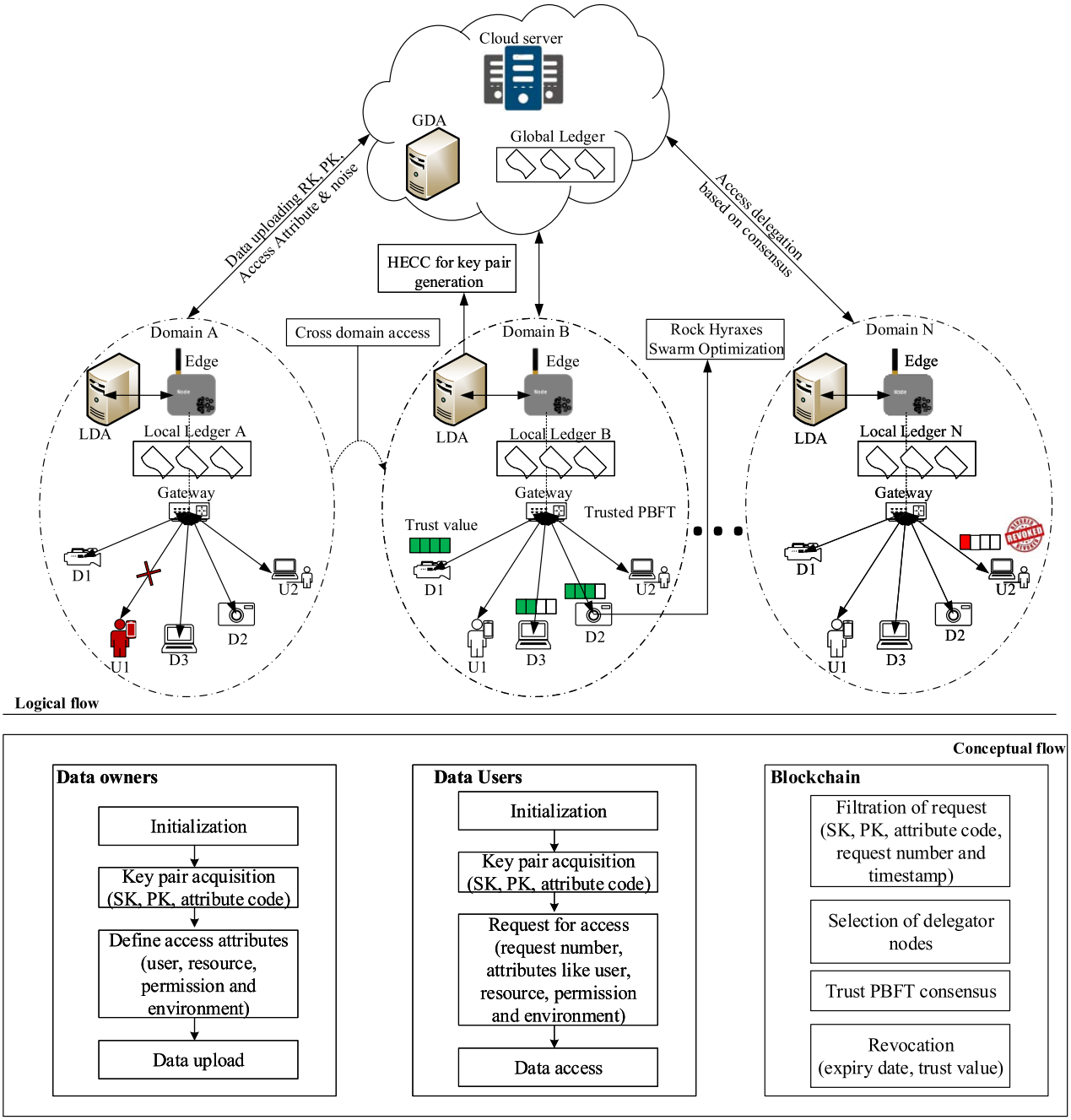

Fig. 2. Architecture of the proposed DSA-block model.

the positions in the filter. If the entire position in the array is one, the filter shows that the request is legitimate, otherwise, it is illegitimate. Hence, the hash function is used to verify the entries in the array. In addition, it is also used to ensure the legitimacy of the request. For huge values of m, the false positive rate (μ) of the filter is defined as follows:

$$\mu = \left(1 - e^{-pq/m}\right)^p \tag{4}$$

where q represents the number of elements in the filter, m and p represent the false positive rate, m represents the nearest value of power 2, and p is rounded to the nearest integer value. Based on the filter, we filter only legitimate requests. If the user is found to be an authenticated user, the timestamp and freshness of the incoming packets are verified. By doing so, the illegitimate access requests causing DDoS attacks will be filtered and ignored. Only the legitimate access requests are processed, which reduces the burden on access delegation.

C. Trust-Based Access Delegation

The filtered legitimate requests are forwarded to the edge server, which is responsible for access delegation. The disadvantage of a single delegator node is overcome by selecting the multiple delegators using blockchain to ensure security. The selection of delegator nodes is carried out by using the RHSO

algorithm based on the parameters, such as trust value (T), energy level (E), load (L), and RA. The trust value of a node is provided by other nodes in the domain. First, the population is initialized for leaders and members. Hence, the leader selects the best place for observing the remainder of the group. Here, the best nodes among all nodes are known as the leader, which is used to select the optimal delegator. Then, the leader updates the current location based on its previous location, which is defined as follows:

$$\text{loc} = R_1 * \text{lpp}(\text{lcp}, j) \tag{5}$$

where R_1 represents a random value between [0, 1], lpp represents the leader's previous position, lcp represents the current position of the leader, and j represents the reduction. After updating the position of the leader, all the members update their current position based on their previous position. The fitness value of the new position is calculated as follows:

$$F = w(T, E, L, \text{RA}) \tag{6}$$

where F represents the fitness function and w represents the weight values of the parameters. If the new position is better than the leader, then the member changes and updates its position

$$\text{lpp}(i, j) = (x(i, j) - \text{Cir} * \text{lpp}(i, j) + \text{loc}) \tag{7}$$

where Cir represents the motion that imitates the circle system, which is defined as follows:

$$\varepsilon_1 = R_2 * \cos(\vartheta) \tag{8}$$

$$\varepsilon_2 = R_2 * \sin(\vartheta) \tag{9}$$

$$\text{Cir} = \text{SQRT}\left(\varepsilon_1^2 + \varepsilon_2^2\right)\Big) \tag{10}$$

where R_2 represents the radius and it is a random value between [0, 1], ϑ is also a random value between [0, 360] and it represents the movement angle, which is updated in every iteration. The updating of the angle is based on the upper bound (u_b) and lower bound (l_b) of the variables, which are defined as follows:

$$\text{Dalta} = \text{Random}[l_b, u_b) \tag{11}$$

$$\vartheta = \vartheta + \text{Dalta}. \tag{12}$$

After completed the updating process, they continue searching the food in the locality, otherwise, the current new position is discarded. In this way, optimal delegators are selected by ENs. The energy levels of the IoT nodes are considered to address its energy-constraint nature. The number of selected delegator nodes is restricted to a limited number based on the total number of nodes in the domain. The Trusted PBFT consensus is utilized to achieve consensus between the nodes to provide the final access decision. The selection of more trusted nodes from the set of trusted nodes reduces the block validation time, which is an advantage of Trusted PBFT. The access decision is carried out based on user attributes, resource attributes, permission attributes, and environmental attributes. If the user requests resource in another domain, then the request along with the respective user attributes are shared to the EN in the respective domain via a secure channel from which the delegation of access will be performed.

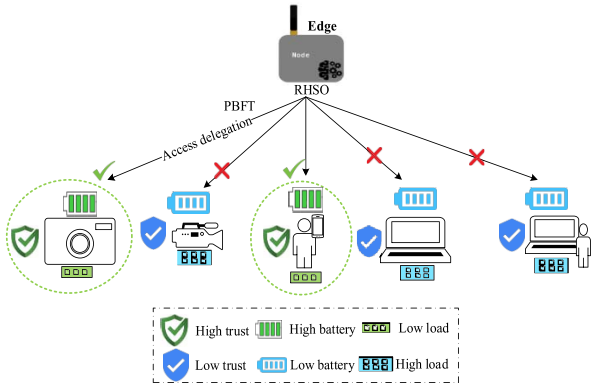

Fig. 3. Trust-based access delegation mechanism using RHSO and PBFT.

1) *Trust Value (T):* It is calculated for every delegator to increase security. Here, trust values are given by neighboring nodes in the network. The calculation of the trust value is defined as follows:

$$T = \sum_{i=1}^{n} F_n \tag{13}$$

where F_n represents the feedback of the delegator, which is provided by the neighbor nodes.

2) *Energy Level (E):* It is used to determine the current energy level of the delegators. A high-energy level delegator is selected as the optimal delegator to perform access delegation. The calculation of energy level is defined as follows:

$$E = \xi - £ \tag{14}$$

where ξ represents the total energy of the node and £ represents the utilized energy of the node.

3) *Load (L):* It used to evaluate the load of the node. If the node has a lower load, then it is selected as the optimal delegator for performing access delegation. The evaluation of load is defined as follows:

$$L = \sum_{i=1}^{n} w_n \tag{15}$$

where w_n represents the amount of work performed by the delegator node. If the node has less work then it also has a lower load level.

4) *Resource Availability:* It is used to calculate the available resources of the node. If the node has high RA, then it is selected as the optimal delegator for access delegation. The calculation of RA is defined as follows:

$$\text{RA} = \eta - \zeta \tag{16}$$

where η represents the total resources of the node and ζ represents the resources utilized by the node.

Fig. 3 represents the trust-based access delegation mechanism using RHSO and PBFT.

D. Privacy-Aware Data Sharing

The IoT devices upload their data to the cloud server for secure storage. In this scenario, the IoT devices act as DOs,

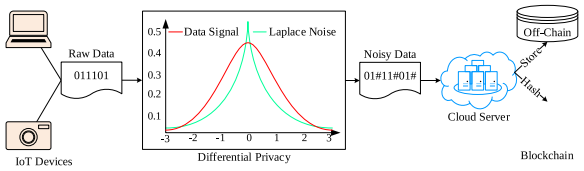

Fig. 4. Privacy data sharing using differential privacy mechanism.

and the users requesting access to the data are considered data customers. The DOs upload their data to the cloud storage by providing a random key, its public key, and the access attributes. The privacy of the data is further improved by implementing the differential privacy mechanism, in which the noisy data is encoded with the original data by the DOs. The Laplace method is used in this work for adding noisy data to the original data shared by the DOs. The data shared by the $D^I = (D^{I1}, D^{I2}, \ldots, D^{In})$ contains privacy information $D^I_{pv} = (D^1_{pv}, D^2_{pv}, \ldots, D^n_{pv}$, the Laplace function with mean and variance is given as follows:

$$\begin{cases} \text{Variance}, & \frac{\nabla D^I_{pv}}{\exists} \\ \text{Mean}, & 0. \end{cases} \tag{17}$$

The probability distribution function is expressed as follows:

$$\text{PDF}(z\ \forall) = \frac{1}{2\forall} \exp\left(-\frac{|z|}{\forall}\right) \tag{18}$$

where z denotes the explicit variable and $\forall = ([\nabla D^{I1}]/[\exists])$, respectively. If the data D^{I1} is proportional to the $\exp(([\exists\forall(D^I_{pv}, \text{CN})]/[2\nabla\forall])$, (D^I_{pv}, CN) denotes the privacy data exported from D^I to the CN, the $\exists$ indicating that differential privacy protection is applied, which can be formulated as follows:

$$B\left(D^I_{pv}\ \forall\right) = \left\{\text{CN} : \text{PDF}(D^I \in \text{CN}) \propto \exp\left(\frac{\exists\forall\left(D^I_{pv}, \text{CN}\right)}{2\nabla\forall}\right) + \text{Lap}(\partial(\exists_1))\right\} \tag{19}$$

where $\text{Lap}(\partial(\exists_1))$ denotes the noise by the Laplace function. The above equation is the differential scoring function, in which a higher score provides greater privacy. Any adversary who tries to compromise the message, will not be able to do so.

The uploaded data are stored off-chain and their respective hash values are stored in the blockchain. The access of data by the users is performed by submitting a request to the cloud server through the EN. The CN requests access delegation to the respective EN, which provides access decisions through a consensus mechanism. Once the cloud receives the access decision, it provides the access accordingly. Fig. 4 presents the differential privacy mechanism-based data sharing from DOs to cloud.

E. Dual Revocation

The dynamic access control is provided by timely revocation of the user attributes and on-demand revocation of users. The user attributes are revoked based on the expiry time and the updating of attributes is carried out. The revoked users cannot upload the data to the cloud server. If the attributes are revoked, then a request message is sent to the private LL to get a new key based on attributes for further processes. The ledger verifies the attributes using blockchain for new key generation, if it is correct, then it generates a new key, otherwise the request is rejected. This type of revocation process increases security.

The revocation of users is used to maintain the security of the domain. The trust values of the IoT users vary depending on the behavior of the users. At the time of attribute revocation, the malicious users have an opportunity to compromise the users. If a user is compromised by a malicious user its trust value will be reduced. Once the trust value is reduced below the threshold value the revocation of users from the domain is carried out. Here, the threshold value is generated based on the Shannon entropy, which is defined as follows:

$$H(T) = -\sum_{i=1}^{n} P(T_i) \log P(T_i) \tag{20}$$

where H represents the threshold and T represents the trust value of the user. If the threshold is less than 0.5 then user revocation is performed, otherwise, revocation is not performed.

TABLE III
SYSTEM SPECIFICATIONS

Hardware Specifications	RAM	6GB
	Hard disk	500GB
Software Specifications	Simulation Tool	NS-3
	Operating System (OS)	Ubuntu 14.04LTS

VI. EXPERIMENTAL RESULTS

This section describes the experimentation carried out with the proposed DSA-Block method to analyze its performance. The experimental results show that the proposed DSA-Block method achieves high security and privacy based on the efficient access control method. This section consists of four subsections, such as the simulation setup, application scenario, comparative analysis, research summary, and security analysis.

A. Simulation Setup

This section illustrates the simulation of the proposed DSA-Block method. The performance analysis of the proposed DSA-Block method was simulated by using the Network Simulator version 3.26 (NS-3) simulation tool. The specifications of the NS-3 tool are closely relevant to the proposed DSA-Block method. The proposed DSA-Block method was simulated in an 1000 m $\times$ 1000 m environment. The system specifications for performing simulation are represented in Table III and the simulation parameters are denoted in Table IV.

The simulation environment of the proposed DSA-Block method is illustrated in Fig. 5. It consists of numerous IoT devices with several GWs, ENs, and LDA. In addition, the GDA, cloud layer, and blockchain are also included. The IoT devices gather data from the environment and transmit it to the LDA through GW and ENs. All the LDAs transfer the aggregated data to the GDA for further access. The ENs act as

TABLE IV
SIMULATION PARAMETERS

Parameters	Values
Number of IoT Devices	150
Number of Users	15
Number of Edge Nodes	3
Number of Gateways	3
Number of LDA	3
Number of GDA	1
Blockchain	1
Cloud Server	1
Placement of nodes	Random
Network layer protocol	IPV4
MAC layer protocol	IEEE 802.11b
Request expiry time	350ms
Block size	8 bytes
Data hash	32 bytes
Timestamp	4 bytes
Transaction size	4 bytes
Simulation time	100s
Simulation area	1000m * 1000m

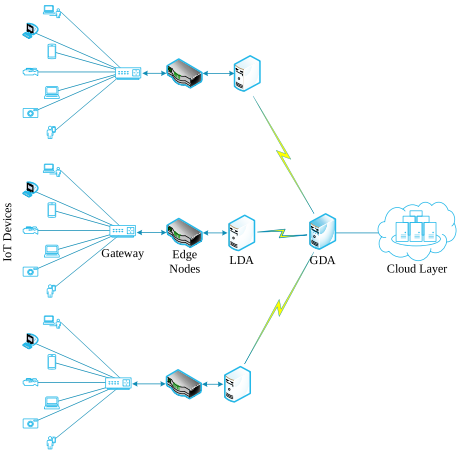

Fig. 5. Simulation environment.

mediators for the IoT users and the cloud layer. Data security is ensured by hashing the data and storing it in the blockchain.

B. Application Scenario

Recently, IoT has been used for numerous real-time applications, such as smart homes, smart hospitals, smart schools, etc., with rapidly emerging technologies, especially in smart hospitals (i.e., health care). Fig. 6 provides a diagrammatic representation of the smart hospital application flow. The personal details of the patients, such as name, age, sex, previous medical records, emergency details, such as contact details, allergies, blood group, etc., are maintained in a secure manner for further purposes and access is provided to authorized persons (i.e., doctors and healthcare workers). After consulting the doctor, the private notes (i.e., personal reports) of the patients are managed securely without any duplication.

Entry access to the restricted area and to visit resident patients is provided only for managers, doctors, and healthcare workers. The medication reports of the patients are given high privacy and access is provided for doctors and pharmacies only. Access is provided to visitors only during visiting hours, otherwise, access is restricted. Resident patients should

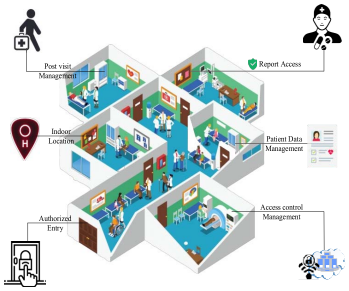

Fig. 6. Use case (smart hospital).

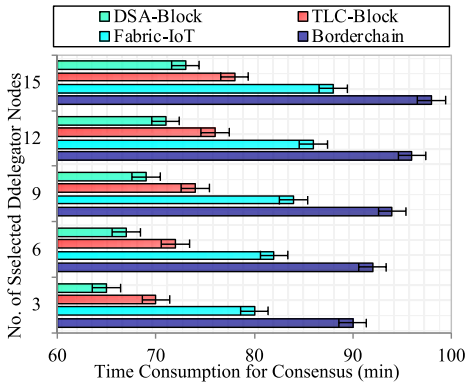

Fig. 7. Time consumption versus number of selected delegated nodes.

not have any access to the personal information of other residents. Similarly, based on the identity of the doctors, some access is restricted for certain actions. The doctors' post-visits (i.e., when doctors attend patients homes to treat them) are recorded and stored on the server securely. The doctors have access that enables them to retrieve the emergency details of the resident patients for further treatments and medications. If any unauthorized persons try to perform a specific operation for which they do not have permission, this is referred to the managers for further actions. Fig. 6 represents the use case of the proposed work in a smart hospital scenario.

C. Comparative Analysis

In this section, the proposed DSA-Block framework is compared with existing works, such as TLC-Block [36], Fabric-IoT [37], and Borderchain [39] in terms of the consensus time consumption, throughput, block validation time, transaction time, latency, response time, and attack detection rate.

1) Consensus Time Consumption Comparison: The consensus algorithm ensures the integrity of the node. The time consumption for consensus is defined as the amount of time consumed to ensure the integrity of the node by running the consensus algorithm. Fig. 7 shows the comparison of consensus time against the number of selected delegator nodes for the proposed DSA-Block with existing works. From the figure, when the number of delegator nodes increases, the time consumption for the consensus also increases, though our proposed DSA-Block achieves less time consumption for the consensus algorithm due to the utilization of the PBFT

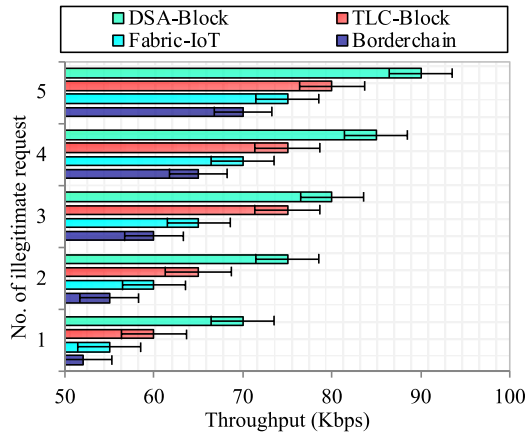

Fig. 8. Throughput versus number of illegitimate requests.

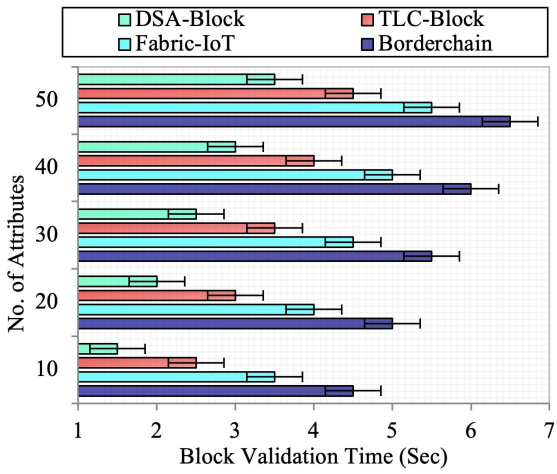

Fig. 9. Block validation time versus number of attributes.

consensus algorithm, in which a limited number of nodes are running consensus, which reduces the time consumption for a consensus algorithm. The existing work Fabric-IoT framework performs access delegation in a GW and Kafka consensus is utilized, which leads to a single point of failure and security threats, respectively.

In the proposed work, consensus takes 73-min when the number of delegator nodes is 15 while the existing works TLC-Block, Fabric-IoT, and Borderchain require more time for consensus as 78, 88, and 98 min, respectively. The average time consumption for consensus when the number of delegator nodes increases to 15 is 69 min, whereas in the existing works, it takes 74, 84, and 94 min, respectively.

2) Throughput Comparison: Throughput (T^p) is defined as the amount of data or number of requests by the users to the size of the data or request by the nodes and users, respectively, which can be formulated as follows:

$$T^p = \frac{\text{Amount of data}}{\text{Size of data}}. \tag{21}$$

Fig. 8 represents the comparison of throughput versus the number of illegitimate requests for the proposed DSA-Block and existing works. From the figure, when the number of illegitimate requests increases, throughput also increases. The proposed DSA-Block achieves high throughput. This is due to the authentication-based request filtration process and privacy-aware data-sharing process. In authentication-based request filtration, the legitimate user request is filtered out by filters in the GW for further processes, and in the privacy-aware data-sharing process, the data of the trusted nodes are sent to the cloud via the differential privacy mechanism method. In addition, the initialization and trust-based access delegation process also increases throughput, thereby improving the overall system performance. The existing works Fabric-IoT and Borderchain are limited with scalability issues, which lead to a single point of failure, thereby reducing throughput.

The proposed work achieves a throughput of 90 kb/s when the number of legitimate user requests is 5, while the existing works TLC-Block, Fabric-IoT, and Borderchain achieve lower throughputs of 80, 75, and 70 kb/s, respectively. The average throughput when the number of legitimate user requests is 5 is 80 kb/s, whereas in the existing works it is 71, 65, and 60.4 kb/s, respectively.

3) Block Validation Time Comparison: The amount of time taken to validate a block (deciding whether to accept or revoke the block) is known as the block validation time. Fig. 9 presents a comparison of block validation versus the number of attributes for the proposed DSA-Block and existing works. From the figure, when the number of attributes increases, block validation time also increases. The proposed DSA-Block, however, achieves a low block validation time. This is due to the trust-based access delegation process, in which the trusted delegator nodes are selected based on several metrics using the RHSO algorithm through, which a set of trusted nodes is obtained. From the obtained set of trusted nodes, the PBFT consensus is utilized for access delegation, which reduces the block validation time, whereas in existing works TLC-Block, Fabric-IoT, and Borderchain, this is limited to a single delegator node section, which means block validation takes more time.

The proposed work achieves a block validation time of 3.5-s when the number of attributes is 50, while the existing works TLC-Block, Fabric-IoT, and Borderchain have higher block validation times of 4.5, 5.5, and 6.5 s, respectively. The average block validation time of the proposed work is 2.5-s when the number of attributes is 50, whereas the existing works have average block validation times of 3.5, 4.5, and 5.5 s, respectively.

4) Transaction Time Comparison: The time taken to complete the verification of transaction time through the consensus algorithm is known as the transaction time. Fig. 10 presents a comparison of the transaction time of the proposed DSA-Block with existing works in which, as the number of transactions increases, transaction time also increases. The proposed work achieves a shorter transaction time due to faster verification of node transactions by using the PBFT consensus algorithm, which takes less time for verification thereby increasing the block creation speed and reducing the transaction time. In addition, the RHSO algorithm also contributes to reducing the transaction time by optimally selecting delegator nodes, whereas Fabric-IoT utilizes the Kafka consensus algorithm,

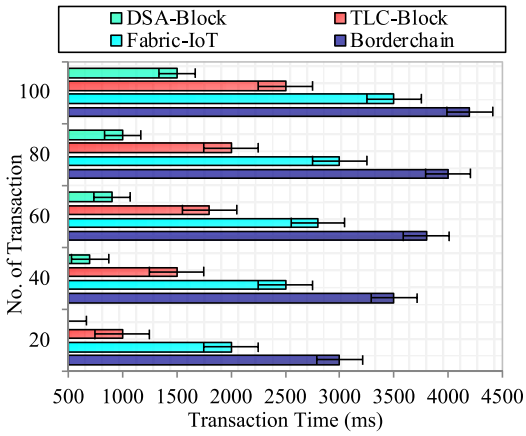

Fig. 10. Transaction time versus number of transactions.

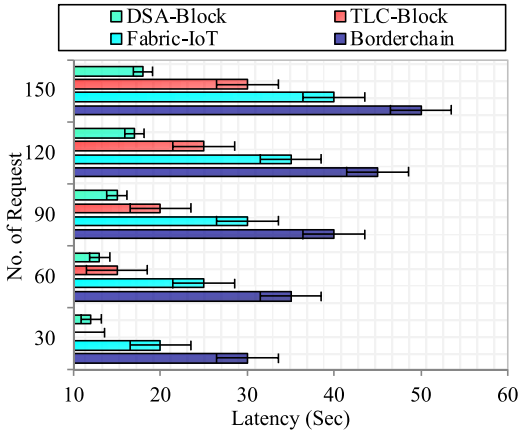

Fig. 11. Latency versus number of requests.

which takes more time for verification and leads to increased block creation and transaction time.

The proposed work achieves a transaction time of 1500-ms when the number of transactions increases to 100, while the existing works TLC-Block, Fabric-IoT, and Borderchain have high transaction times of 2500, 3500, and 4200 ms, respectively. The average transaction time achieved by the proposed DSA-Block when the number of transactions increases to 100 is 920 ms, whereas the existing works show high average transaction times of 1760, 2760, and 3700 ms, respectively.

5) Latency Comparison: Latency (Lat) is defined as the amount of time taken to access the user request from the total time, which can be formulated as follows:

$$\text{Lat} = \text{tot}^T - R^A \tag{22}$$

where tot^T denotes the total time and R^A denotes the accessed request, respectively. Figs. 11 and 12 present the latency comparison of the proposed DSA-Block work with existing works in terms of number of requests and transaction rate, respectively.

From Fig. 11, when the number of user requests increases, latency also increases, and our proposed work achieves lower latency due to the authentication-based request filtration. In the

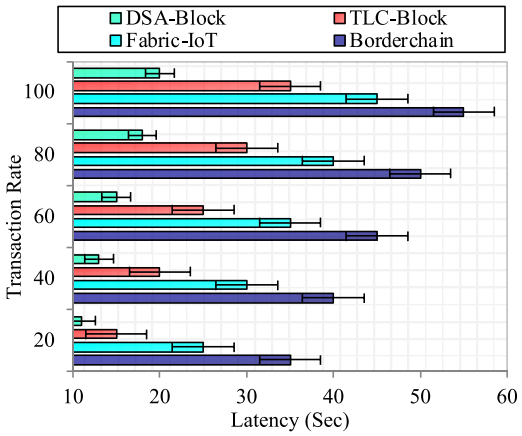

Fig. 12. Latency versus transaction rate.

authentication-based filtration process, the authenticated users are further filtered out based on the time stamp and freshness of the data, which allows only filtered trusted requests for trust-based delegation thereby reducing latency. By contrast, the existing works TLC-Block, Fabric-IoT, and Borderchain perform access delegation for all requests, which leads to increased security threats and increased latency. The proposed work achieves a latency of 18-s when the number of requests increases to 150, while the existing works have higher latencies of 30, 40, and 50 s, respectively. The average latency achieved by the proposed work, when the number of requests is increased to 150 was 15 s, whereas for the existing works, this was 20, 30, and 40, respectively.

Fig. 12 shows that low latency was achieved when the rate of transaction increased, due to the authentication-based request filtration and trust-based access delegation. The authentication-based request filtration process filters the legitimate users based on the time stamp and freshness of data and the trust-based access delegation process utilizes the RHSO algorithm to perform access delegation based on trust, energy level, and load. The PBFT consensus is used to reduce the block validation time and check the integrity of the requests. This process reduces latency by providing the access delegation to only legitimate filtered users even though the transaction rate is high. The proposed work achieves latency of 20-s when the transaction rate increases to 100, while the existing works achieve high latencies of 35, 45, and 55 s, respectively. The average latency achieved by the proposed work when the transaction rate increased to 100 was 15.4 s, whereas for existing works this was 25, 35, and 45 s, respectively.

6) Response Time Comparison: The amount of time taken to transmit a user request, process the request, and return a response is known as the response time *rp*, which can be formulated as follows:

$$rp = E^R - S^R \tag{23}$$

where S^R and E^R denote the start and end of user requests, respectively. Fig. 13 presents a comparison of the response time of the proposed DSA-Block with existing works, in which when the number of user request increases, response time also

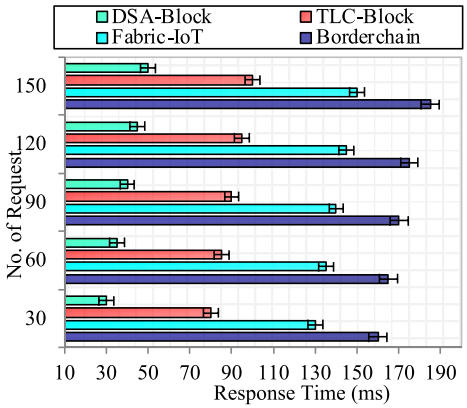

Fig. 13. Response time versus number of requests.

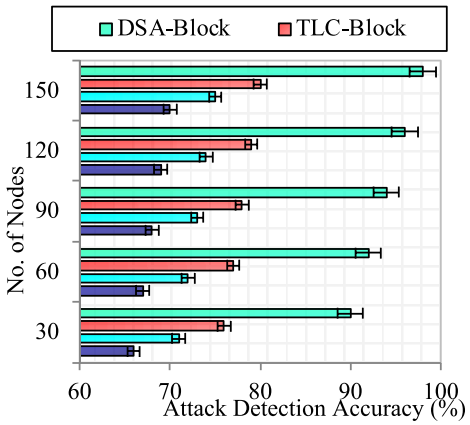

Fig. 14. Attack detection accuracy versus number of nodes.

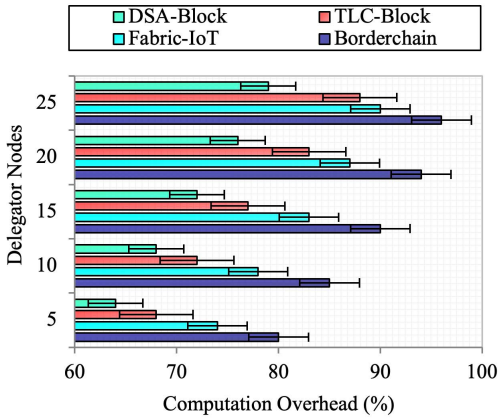

Fig. 15. Computation overhead versus delegator nodes.

increases. Our proposed work achieves a low response time due to system initialization, authentication-based filtration, and trust-based access delegation. The system initialization process utilizes the HECC algorithm to ensure the legitimacy of the user request, authentication-based request filtration further filters the legitimate user requests to reduce the latency of the network, and the trust-based access delegation process utilizes the RHSO algorithm to provide access delegation to only requests with high trust, a high energy level, and low load, respectively. The PBFT is used to reduce the block validation time and ensure the integrity of the user, thereby reducing the response time of the user request. By comparison, the existing works TLC-Block, Fabric-IoT, and Borderchain allow the entire request and also user-centralized methods, which leads to a high response time thereby affecting the performance of the system.

The proposed work achieves response time of 50-ms when the number of requests increases to 150, while the existing works achieve high response times of 100, 150, and 180 ms, respectively. The average response time achieved by the proposed work when user requests increased to 150 was 40 ms, whereas the average response times in existing works were 90, 140, and 171 ms, respectively.

7) Attack Detection Accuracy Comparison: The measure of attack detection as the number of IoT nodes increases is known as the attack detection rate, (D_r^a) which can be formulated as follows:

$$D_r^a = \frac{\text{detected attacks}}{\text{IoT nodes}}. \tag{24}$$

Fig. 14 presents a comparison of the attack detection accuracy for the proposed DSA-Block with existing works. From the figure, it is shown that when the number of nodes increases, the attack detection rate also increases. From that, our proposed work achieves a high attack detection rate due to authentication-based request filtration, privacy-aware data sharing, and the dual revocation process. In authentication-based request filtration, the DDoS attacks are mitigated by filtering the legitimate node/user requests based on the time stamp and freshness, the privacy-aware data-sharing method utilizes the differential privacy mechanism, which reduces the man-in-the-middle attacks, and dual revocation uses the Shannon entropy method based on trust, which further ensures the security through the revocation mechanism. The existing work TLC-Block performs authentication without considering proper authentication metrics, which leads to an increased attack detection rate.

The proposed work achieves a high attack detection rate 98% when the number of nodes increases to 150, while the existing works achieve 80%, 75%, and 70%, respectively. The average attack detection rate achieved by the proposed work when nodes are increased to 150 is 94%, whereas for existing works this is 78%, 73%, and 68%, respectively.

8) Computation Overhead Comparison: The amount of time taken by the entity to complete a process of input data is known as the computation overhead (Co_{oh}^{mp}), which can be formulated as follows:

$$Co_{oh}^{mp} = \frac{\text{receiving time}}{\text{processing time}}. \tag{25}$$

Fig. 15 presents the computation overhead comparison for the proposed and existing works, in which computation overhead increases as the number of delegator nodes increases. The proposed work achieves a lower computation overhead than the existing works. This is due to the adoption of the RHSO algorithm for delegator selection. The RHSO-based trusted delegator is selected based on the increased energy level, the less load, and the high trust. The adoption of an optimization

TABLE V
NUMERICAL RESULTS

Metrics		Proposed DSA-Block	TLC-Block	Fabric-IoT	Borderchain
Consensus time consumption (min)		69	74	84	94
Throughput (Kbps)		80	75	65	60.4
Block validation time (sec)		2.5	3.5	4.5	5.5
Transaction time (ms)		920	1760	2760	3700
Latency (sec)	No. of request	15	20	30	40
	Transaction rate	15.4	25	35	45
Response time (ms)		40	90	140	171
Attack detection accuracy (%)		94	78	73	68
Computation overhead (%)		71.8	77.6	82.4	89

algorithm for the delegator selection reduces the time processing time of the request, thereby reducing the computation overhead. By contrast, the existing works TLC-Block, Fabric-IoT, and Borderchain lack the intelligent packet processing methods.

Fig. 15 shows that when the number of delegator node increases to 25, the proposed work achieves a lower computation overhead of 79%, whereas the existing works Fabric IoT, Borderchain, and TLC-Block achieve high computation overhead of 96%, 90%, and 88%, respectively.

D. Research Summary

The proposed DSA-Block was evaluated using several performance metrics, which are consensus time consumption (69 min), throughput (80 kb/s), block validation time (2.5 sec), transaction time (920 ms), latency (15 s for the number of requests) and (15.4 s for transaction rate), response time (40 ms), and attack detection accuracy (94%), these results are shown in Figs. 7–14. Initially, the proposed DSA-Block system stores both user and device attributes to the LDA for key generation, which increases the security of the environment and results in high throughput and low latency during attack detection. Authentication-based request filtration is used to increase attack detection accuracy. Trust-based access delegation increases the security and throughput, and reduces response time due to considering only filtered requests. Privacy-aware data sharing also increases the throughput and reduces latency by introducing a differential privacy mechanism for data sharing, which increases data security. Dual revocation increases the security of the environment by performing both attribute and user revocation. Here, the blockchain used the PBFT consensus, which reduces block validation time, transaction time, and consensus time. Table V presents the results of the proposed and existing works in terms of comparison metrics. The highlights of the proposed DSA-Block results are defined as follows.

1) The authentication of the entities in the network is carried out by utilizing the HECC algorithm, which has reduced key size and increased security, thereby mitigating the complexity of conventional cryptographic algorithms.
2) The authentication-based filtration of requests mitigates the DDoS attacks and reduces the burden of the IoT GW. The timestamp and freshness of the requests are also validated.
3) The selection of delegator nodes in the domain is performed by using RHSO based on several significant parameters, through which consensus on access delegation is achieved through Trusted PBFT.
4) Privacy-aware data sharing is performed, in which the encrypted data to be uploaded to the cloud server is encoded with noise via the differential privacy mechanism, which ensures data privacy.
5) The dual revocation of attributes and users is executed to ensure the security of the network, in which the attribute revocation is performed based on expiry time and user revocation is performed.

E. Security Analysis

The security analysis of the proposed DSA-Block method is described in this section. Providing access control and data sharing in the IoT environment addresses critical security and privacy threats represented by various forms of attacks, hence, we proposed the DSA-Block method to improve the security of the IoT environment. In doing so, we mitigate four types of attacks, which are listed as follows.

1) *DDoS Attacks:* This attack sends out duplicate requests to create high Internet traffic for the server, which increases the risk of a single point of failure. In the proposed DSA-Block method, this is mitigated by performing authentication based on request filtration to mitigate this attack, which reduces the risk of the single point of failure.
2) *Sybil Attacks:* This type of attack, reduces the reputation value of the IoT nodes by generating a huge number of fake identities. This attack is mitigated by performing the PBFT consensus based on user, resource, permission, and environment attributes in the proposed DSA-Block method.
3) *MITM Attacks:* A man-in-the-middle attack occurs between the IoT device and cloud storage during data sharing by deleting or modifying the data, which increases both security and privacy threats. To overcome these threats, a differential privacy mechanism was implemented in this research prior to data sharing, which mitigates this kind of attack.
4) *Phishing Attacks:* This type of attack appears as reputed nodes that send fake requests to perform malicious actions, which reduces the security. In this research, this kind of attack is mitigated by performing dual revocation based on the expiry time and behavior of the users.

VII. Conclusion

Dynamic secure access and data sharing are achieved in the proposed DSA-Block model, which improves the security and privacy of the IoT environment. All the user and device attributes are registered in the LDA for key generation, which increases the legitimacy of both users and devices. The IoT devices and users send an access request message to the edge server using a GW that only allows legitimate requests for access delegation. Here, access delegation is performed by the edge server by selecting a set of access delegators using RHSO, which optimally selects access delegators. This increases the security of the environment, which also increases the throughput and attack detection rate. Then, legitimate data are stored in the cloud server using blockchain via a differential privacy mechanism, which encodes the noise to the original data to increase privacy. The legitimate data are stored in the off-chain to enhance security; the hash values of the off-chain data are stored in the blockchain. The blockchain uses the PBFT consensus algorithm to create and add new blocks, which reduces transaction time, block validation time, and consensus time. Finally, revocation is performed for both user attributes and users in order to maintain the security of the environment. The simulation results showed that the proposed DSA-Block model achieves superior performance compared to other state-of-the-art works. In future, we plan to mitigate various types of attacks by using a modified blockchain that increases security and processing speed and reduces energy consumption during data sharing. In addition, the deduction of data availability for data customers via a differential privacy mechanism will be further investigated.

References

[1] S. Bhatt, T. K. Pham, M. Gupta, J. Benson, J. Park, and R. Sandhu, "Attribute-based access control for AWS Internet of Things and secure industries of the future," *IEEE Access*, vol. 9, pp. 107200–107223, 2021.

[2] Y. Liu et al., "Capability-based IoT access control using blockchain," *Digital Commun. Netw.* vol. 7, no. 4, pp. 463–469, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352864820302844

[3] S. A. Chaudhry, K. Yahya, F. Al-Turjman, and M.-H. Yang, "A secure and reliable device access control scheme for IoT based sensor cloud systems," *IEEE Access*, vol. 8, pp. 139244–139254, 2020.

[4] L. Liu, H. Wang, and Y. Zhang, "Secure IoT data Outsourcing with aggregate statistics and fine-grained access control," *IEEE Access*, vol. 8, pp. 95057–95067, 2020.

[5] Q. Yang, M. Zhang, Y. Zhou, T. Wang, Z. Xia, and B. Yang, "A non-interactive attribute-based access control scheme by blockchain for IoT," *Electronics*, vol. 10, no. 15, p. 1855, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/15/1855

[6] A. Kousalya, K. Sakthidasan, and A. Latha, "Reliable service availability and access control method for cloud assisted IoT communications," *Wireless Netw.*, vol. 27, pp. 881–892, Feb. 2021.

[7] K. M. Hossein, M. E. Esmaeili, T. Dargahi, A. Khonsari, and M. Conti, "BCHealth: A novel blockchain-based privacy-preserving architecture for IoT Healthcare applications," *Comput. Commun.*, vol. 180, pp. 31–47, Dec. 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0140366421003054

[8] Z. Li, J. Hao, J. Liu, H. Wang, and M. Xian, "An IoT-applicable access control model under double-layer blockchain," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 68, no. 6, pp. 2102–2106, Jun. 2021.

[9] U. Khalid, M. Asim, T. Baker, P. C. K. Hung, M. A. Tariq, and L. Rafferty, "A decentralized lightweight blockchain-based authentication mechanism for IoT systems," *Clust. Comput.*, vol. 23, pp. 2067–2087, Feb. 2020.

[10] S. Xiong, Q. Ni, L. Wang, and Q. Wang, "SEM-ACSIT: Secure and efficient multiauthority access control for IoT cloud storage," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2914–2927, Apr. 2020.

[11] S. Banerjee et al., "Multi-authority CP-ABE-based user access control scheme with constant-size key and ciphertext for IoT deployment," *J. Inf. Security Appl.*, vol. 53, Aug. 2020, Art. no. 102503. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214212619310178

[12] M. Dammak, S.-M. Senouci, M. A. Messous, M. H. Elhdhili, and C. Gransart, "Decentralized lightweight group key management for dynamic access control in IoT environments," *IEEE Trans. Netw. Service Manag.*, vol. 17, no. 3, pp. 1742–1757, Sep. 2020.

[13] P. Chinnasamy, P. Deepalakshmi, A. K. Dutta, J. You, and G. P. Joshi, "Ciphertext-policy attribute-based encryption for cloud storage: Toward data privacy and authentication in AI-enabled IoT system," *Mathematics*, vol. 10, no. 1, p. 68, 2022. [Online]. Available: https://www.mdpi.com/2227-7390/10/1/68

[14] N. Tapas, F. Longo, G. Merlino, and A. Puliafito, "Experimenting with smart contracts for access control and delegation in IoT," *Future Gener. Comput. Syst.*, vol. 111, pp. 324–338, Oct. 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167739X18326979

[15] S. Pal, T. Rabehaja, M. Hitchens, V. Varadharajan, and A. Hill, "On the design of a flexible delegation model for the Internet of Things using blockchain," *IEEE Trans. Ind. Informat.*, vol. 16, no. 5, pp. 3521–3530, May 2020.

[16] C.-A. Lin and C.-F. Liao, "User-managed access delegation for blockchain-driven IoT services," in *Proc. Int. Comput. Symp. (ICS)*, 2020, pp. 462–467.

[17] S. S. Panda, D. Jena, B. K. Mohanta, S. Ramasubbareddy, M. Daneshmand, and A. H. Gandomi, "Authentication and key management in distributed IoT using blockchain technology," *IEEE Internet Things J.*, vol. 8, no. 16, pp. 12947–12954, Aug. 2021.

[18] W. Yang, Z. Guan, L. Wu, X. Du, and M. Guizani, "Secure data access control with fair accountability in smart grid data sharing: An edge blockchain approach," *IEEE Internet Things J.*, vol. 8, no. 10, pp. 8632–8643, May 2021.

[19] Y. Meshcheryakov, A. Melman, O. Evsutin, V. Morozov, and Y. Koucheryavy, "On performance of PBFT blockchain consensus algorithm for IoT-applications with constrained devices," *IEEE Access*, vol. 9, pp. 80559–80570, 2021.

[20] S. Khan et al., "An efficient and secure revocation-enabled attribute-based access control for eHealth in smart society," *Sensors*, vol. 22, no. 1, p. 336, 2022. [Online]. Available: https://www.mdpi.com/1424-8220/22/1/336

[21] Y. Zhang, B. Li, B. Liu, J. Wu, Y. Wang, and X. Yang, "An attribute-based collaborative access control scheme using blockchain for IoT devices," *Electronics*, vol. 9, no. 2, p. 285, 2020. [Online]. Available: https://www.mdpi.com/2079-9292/9/2/285

[22] S. Liu, J. Yu, Y. Xiao, Z. Wan, S. Wang, and B. Yan, "BC-SABE: Blockchain-aided searchable attribute-based encryption for cloud-IoT," *IEEE Internet Things J.*, vol. 7, no. 9, pp. 7851–7867, Sep. 2020.

[23] G. Yu et al., "Enabling attribute revocation for fine-grained access control in blockchain-IoT systems," *IEEE Trans. Eng. Manag.*, vol. 67, no. 4, pp. 1213–1230, Nov. 2020.

[24] S. Mandal, B. Bera, A. K. Sutrala, A. K. Das, K.-K. R. Choo, and Y. Park, "Certificateless-signcryption-based three-factor user access control scheme for IoT environment," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3184–3197, Apr. 2020.

[25] S. Ding, J. Cao, C. Li, K. Fan, and H. Li, "A novel attribute-based access control scheme using blockchain for IoT," *IEEE Access*, vol. 7, pp. 38431–38441, 2019.

[26] M. Ma, G. Shi, and F. Li, "Privacy-oriented blockchain-based distributed key management architecture for hierarchical access control in the IoT scenario," *IEEE Access*, vol. 7, pp. 34045–34059, 2019.

[27] J. Zhang, Y. Xin, Y. Gao, X. Lei, and Y. Yang, "Secure ABE scheme for access management in blockchain-based IoT," *IEEE Access*, vol. 9, pp. 54840–54849, 2021.

[28] M. Ali, M.-R. Sadeghi, and X. Liu, "Lightweight revocable hierarchical attribute-based encryption for Internet of Things," *IEEE Access*, vol. 8, pp. 23951–23964, 2020.

[29] K. O.-B. O. Agyekum, Q. Xia, E. B. Sifah, C. N. A. Cobblah, H. Xia, and J. Gao, "A proxy re-encryption approach to secure data sharing in the Internet of Things based on blockchain," *IEEE Syst. J.*, vol. 16, no. 1, pp. 1685–1696, Mar. 2022.

[30] P. Kumar, G. P. Gupta, and R. Tripathi, "TP2SF: A trustworthy privacy-preserving secured framework for sustainable smart cities by leveraging blockchain and machine learning," *J. Syst. Archit.*, vol. 115, May 2021, Art. no. 101954. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1383762120302071
[31] B. Shala, U. Trick, A. Lehmann, B. Ghita, and S. Shiaeles, "Blockchain and trust for secure, end-user-based and decentralized IoT service provision," *IEEE Access*, vol. 8, pp. 119961–119979, 2020.
[32] G. Ali et al., "xDBAuth: Blockchain based cross domain authentication and authorization framework for Internet of Things," *IEEE Access*, vol. 8, pp. 58800–58816, 2020.
[33] A. Qashlan, P. Nanda, X. He, and M. Mohanty, "Privacy-preserving mechanism in smart home using blockchain," *IEEE Access*, vol. 9, pp. 103651–103669, 2021.
[34] E. K. Wang, R. Sun, C.-M. Chen, Z. Liang, S. Kumari, and M. Khurram Khan, "Proof of X-repute blockchain consensus protocol for IoT systems," *Comput. Security*, vol. 95, Aug. 2020, Art. no. 101871. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167404820301449
[35] G. D. Putra, V. Dedeoglu, S. S. Kanhere, R. Jurdak, and A. Ignjatovic, "Trust-based blockchain authorization for IoT," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 2, pp. 1646–1658, Jun. 2021.
[36] S. Sun, R. Du, S. Chen, and W. Li, "Blockchain-based IoT access control system: Towards security, lightweight, and cross-domain," *IEEE Access*, vol. 9, pp. 36868–36878, 2021.
[37] H. Liu, D. Han, and D. Li, "Fabric-iot: A blockchain-based access control system in IoT," *IEEE Access*, vol. 8, pp. 18207–18218, 2020.
[38] H. Xu, Q. He, X. Li, B. Jiang, and K. Qin, "BDSS-FA: A blockchain-based data security sharing platform with fine-grained access control," *IEEE Access*, vol. 8, pp. 87552–87561, 2020.
[39] Y. E. Oktian and S.-G. Lee, "BorderChain: Blockchain-based access control framework for the Internet of Things endpoint," *IEEE Access*, vol. 9, pp. 3592–3615, 2021.
[40] S. Pal, T. Rabehaja, A. Hill, M. Hitchens, and V. Varadharajan, "On the integration of blockchain to the Internet of Things for enabling access right delegation," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 2630–2639, Apr. 2020.

Suhair Alshehri received the Ph.D. degree in computing and information sciences from the Golisano College of Computing and Information Sciences, Rochester Institute of Technology, Rochester, NY, USA, in 2014.

She is currently an Assistant Professor with the Information Technology Department, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia. Her main research interests include security and privacy in computer and information systems and applied cryptography.

Omaimah Bamasaq received the Ph.D. degree in computer science, electronic information security from The University of Manchester, Manchester, U.K., in 2006.

She is a Professor of Cybersecurity with the Department of Computer Science, Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia, the Dean of Community Services and Continuing Education with the University of Johannesburg, Johannesburg, South Africa, and a Visiting Researcher at Massachusetts Institute of Technology, Cambridge, MA, USA.

Daniyal Alghazzawi (Senior Member, IEEE) received the Ph.D. degree in computer science from the University of Kansas, Lawrence, KS, USA, in 2007.

He is a Professor of Cybersecurity with the Computing Information Systems Department and the Head of the Information Security Research Group, King Abdulaziz University, Jeddah, Saudi Arabia. He served in a variety of administrative and leadership roles and was awarded the Leadership Management International Certificate. In 2010, he was appointed Honorary Lecturer with the University of Essex, Colchester, U.K. He has worked as a Consultant for a number of companies, assisting them in developing information security policies and obtaining certifications, such as ABET, ISO27001, and ISO22301. In the disciplines of smart e-learning, cybersecurity, and artificial intelligence, he has authored multiple scholarly papers and patents.

Dr. Alghazzawi has also served as a reviewer and an editor for a number of local and international conferences, journals, workshops, and contests. He has organized both domestic and international seminars and conferences. Department of Information Systems, Faculty of Computing and Information Technology, King Abdulaziz University.

Arwa Jamjoom received the master's degree in computer science from the University of Southern California, Los Angeles, CA, USA, in 1997, and the Doctoral degree in computer science from the University of Surrey, Guildford, U.K., in 2011.

She is an Associate Professor with the Department of Information System, King Abdulaziz University, Jeddah, Saudi Arabia. Her research interests lie in data analytics and decision support system.