applied sciences

MDPI

Article

Data Sharing System Integrating Access Control Mechanism using Blockchain-Based Smart Contracts for IoT Devices

Tanzeela Sultana [1], Ahmad Almogren [2,*], Mariam Akbar [1], Mansour Zuair [3], Ibrar Ullah [4] and Nadeem Javaid [1]

[1] Department of Computer Science, COMSATS University Islamabad, Islamabad 44000, Pakistan; tanzeela.sultana@yahoo.com (T.S.); mariam.akbar@gmail.com (M.A.); nadeemjavaidqau@gmail.com (N.J.)

[2] Computer Science Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia

[3] Computer Engineering Department, College of Computer and Information Sciences, King Saud University, Riyadh 11543, Saudi Arabia; zuair@ksu.edu.sa

[4] Faculty of Electrical and Computer Engineering, University of Engineering and Technology Peshawar, Bannu 28100, Pakistan; ibrarullah@uetpeshawar.edu.pk

* Correspondence: ahalmogren@ksu.edu.sa

Received: 7 November 2019; Accepted: 6 January 2020; Published: 9 January 2020

Abstract: In this paper, a blockchain-based data sharing and access control system is proposed, for communication between the Internet of Things (IoT) devices. The proposed system is intended to overcome the issues related to trust and authentication for access control in IoT networks. Moreover, the objectives of the system are to achieve trustfulness, authorization, and authentication for data sharing in IoT networks. Multiple smart contracts such as Access Control Contract (ACC), Register Contract (RC), and Judge Contract (JC) are used to provide efficient access control management. Where ACC manages overall access control of the system, and RC is used to authenticate users in the system, JC implements the behavior judging method for detecting misbehavior of a subject (i.e., user). After the misbehavior detection, a penalty is defined for that subject. Several permission levels are set for IoT devices' users to share services with others. In the end, performance of the proposed system is analyzed by calculating cost consumption rate of smart contracts and their functions. A comparison is made between existing and proposed systems. Results show that the proposed system is efficient in terms of cost. The overall execution cost of the system is 6,900,000 gas units and the transaction cost is 5,200,000 gas units.

Keywords: blockchain; Internet of Things; data sharing; access control; smart contracts; trustfulness; authentication

1. Introduction

Development of the Internet leads to a growing number of devices. The devices are more likely to connect with each other due to the rise of networking and communication technologies (i.e., wifi, ZigBee, etc.). The connection of devices has fastened the growth of the Internet of Things (IoT) networks. IoT is a promising technology that integrates physical world with the Internet. The IoT network is defined as; "the connection of Internet enabled devices that share data, information, and resources with each other". The connection among IoT devices is established without any human intervention. One of the main aspects of IoT is to share data, information, and resources among devices.

The connection of devices extends the applications of IoT networks in various fields. Some of the applications of IoT networks include: vehicular networks (where cars are integrated with an existing entertainment, traffic, and navigation systems), home automation (i.e., smart homes), healthcare (i.e., health data sharing), supply chain management (asset tracking, forecasting, vendor relations, connected fleets, etc.), security systems (i.e., sensors, buzzer connected), and many others [1]. Because of the aforementioned applications, IoT devices are connected globally and the ratio of connectivity is increasing day-by-day.

The connection of devices brings several challenges in the networks. Some of the challenges are: inefficient management of data, unauthorized access, malicious attacks, single point of failure through centralization, and many others [2]. The IoT devices consist of sensitive data; therefore, an efficient management of networks is highly required. As the IoT devices' data is of huge amount; so, centralized storage systems, i.e., cloud and fog are used to store that data. The cloud has the ability to process a huge amount of data in rapid manners. Moreover, it achieves accuracy, efficiency, and speed for data processing. Besides the storage and processing advantages of networks, cloud and fog also bring latency, security, and privacy issues. However, unauthorized access of data is the major issue [3–5].

Many strategies are presented in the literature to overcome issues in the IoT networks. The challenging tasks of IoT networks are efficient data sharing and authorized access control. There are several strategies to manage data sharing and authorized access control of data. Such strategies are based on traditional models, which include: Attribute-Based Access Control (ABAC), Role-Based Access Control (RBAC), and many others. However, there are many limitations in the traditional systems, such as: single point of failure through centralization, untrustworthiness, unauthorized access to data, etc., [6,7].

Therefore, blockchain technology is integrated with access control and data sharing mechanisms, to eliminate the issues in traditional schemes. Blockchain provides solutions to many problems that are more effective to provide data integrity, fairness, authenticity, security, and distribution [8]. Furthermore, Section 1.1 provides an in-depth understanding of blockchain technology.

1.1. Blockchain

The idea of blockchain is proposed by Satoshi Nakamoto in 2008 via a white paper. It is introduced as an underlying technology for bitcoin cryptocurrency. Bitcoin is also known as the first application of blockchain. The blockchain technology is used to secure the financial transactions of digital currency by eliminating the central authority. Blockchain is a networking technology, where nodes are directly connected to each other in a Peer-to-Peer (P2P) manner. It has eliminated the concept of centralization through consensus mechanisms. Decisions in the network are made after the consensus among all nodes. Furthermore, blockchain is also known as a distributed ledger technology. Ledger contains the record of transactions made in the network, and is distributed over all nodes. In Figure 1, the basic blockchain structure is presented.

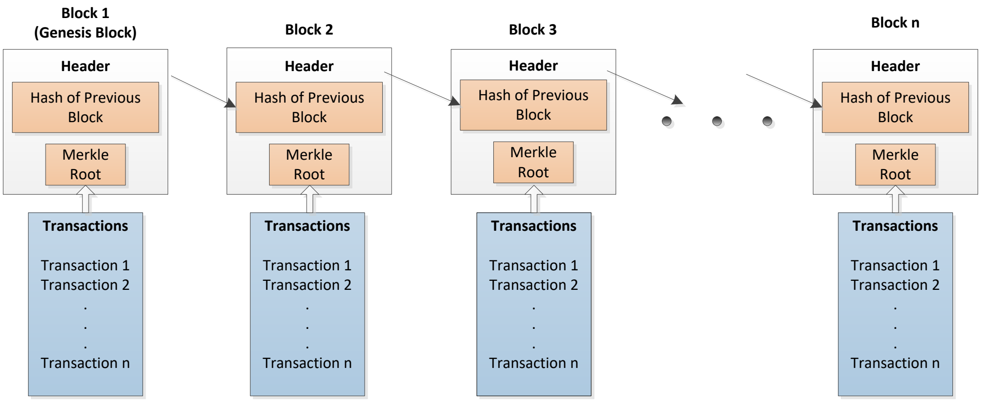

Figure 1. Basic Blockchain Structure.

Moreover, blockchain contains several features that enhance its significance over traditional transaction systems. The features of blockchain that make it more efficient and reliable are security, scalability, immutability, and anonymity [9,10]. The aforementioned properties and features of blockchain technology increase its demand significantly. Therefore, the applications of blockchain are also increased in various fields. For example: it is being integrated with IoT and vehicular networks [11]. Other uses of blockchain technology are: Artificial Intelligence (AI), economy, transportation, health, identity management, supply chain management, and smart contract services [12].

Major features of blockchain that make it distinct from existing systems are:

1.1.1. Ledger

The ledger in blockchain records all transactions that are made in the networks. Blockchain is known as a distributed ledger technology, in which ledger is distributed over all network nodes. Transactions include: financial, health-care, business-related transactions, etc.

1.1.2. Smart Contract

Blockchain technology eliminates the involvement of third-party through smart contracts. Smart contracts are written computer programs. All rules for transaction between two parties are defined in smart contracts. The transaction cannot be done until all the agreements in smart contract are fulfilled. Whenever any transaction is to be made in the blockchain-based networks, smart contract is triggered automatically [13,14].

1.1.3. Consensus Mechanism

The ledger is distributed over all nodes in the network. It is necessary to keep the record synchronized. To maintain data consistency in blockchain, several consensus mechanisms are used. The concept of single authority is also eliminated by consensus mechanisms. Decisions in the network are made after agreement between majority nodes. Moreover, the consensus mechanisms used in blockchain technology include: Proof of Work (PoW), Proof of Concept (PoC), Proof of Authority (PoA), Proof of Stake (PoS), Practical Byzantine Fault Tolerance (PBFT), etc.

1.1.4. Cryptography

Cryptography brings security to the whole network. The cryptographic techniques are used in blockchain to make the data and transactions more secure and tamper-proof. Blockchain uses several cryptographic methods such as: hashing techniques (Secure Hash Algorithm-256 (SHA-256), Keccak256), digital signatures (Rivest-Shamir-Adleman (RSA), Digital Signature Algorithm (DSA), Elliptic Curve Digital Signature Algorithm (ECDSA)), and encryption techniques (Attribute-Based Encryption (ABE), Data Encryption Standard (DES)), and many others [15,16].

Due to the rapid progress of blockchain technology, its usage is increasing. It is used in almost every field. Therefore, the blockchain-based data sharing and access control system is used to provide authenticated and trustworthy sharing and access control of IoT devices' data. Sharing of data is done between two peers, which are: subject (request sender) and object (service provider). Smart contracts are used to manage data sharing and access control. Furthermore, behavior of the users is also monitored. Additionally, some permission levels are defined for the subject to access services of the object.

1.2. Motivation

A lot of work was done in the literature for efficient use and communication of devices in the IoT networks. Several strategies use blockchain technology for data sharing and access control management among devices. Some work only considered access control, whereas others focused on sharing of data and services. The work in [1] is based on access control management of IoT devices' data. Smart contracts are used to ensure trustfulness of the system. The author in [2] proposed an access control system to prevent single point of failure and unauthorized access to the networks. Moreover, many systems are presented to manage sharing of data among IoT devices. In [7], the authors proposed a trust-based sharing system. In this system, data sharing is integrated with access control to achieve trustfulness and authentication. Additionally, the authors in [11] presented a blockchain-based sharing model for the vehicular networks. The objectives of the system are to prevent unauthorized and unauthenticated data sharing, and to achieve high quality sharing. By considering the aforementioned work, a blockchain-based data sharing and access control system is proposed. This system is intended to achieve trustfulness and authentication for data sharing among IoT devices. Additionally, the authorized and privileged access to data, and efficiency of the system in terms of cost are also considered.

1.3. Problem Statement

The dramatic growth of the IoT networks results in numerous challenges like: illegitimate data sharing, unauthorized access control, unauthenticated users, single point of failure due to centralization, and many others. To solve the aforementioned issues, many systems are presented in the existing literature. In these systems, blockchain technology is used that is beneficial due to its decentralized, immutable, transparent, and secure nature. The authors in [1] proposed a smart contracts-based access control framework to provide distributed and trust-based access to the data. However, authorization and authentication of users are not considered, which affect security of the IoT networks. Furthermore, cost and complexity of the system are increased due to multiple smart contracts. Furthermore, the authors in [7] proposed a blockchain-based key management scheme for access control. This scheme is intended to achieve privacy, efficiency, decentralization, and scalability in IoT network access. However, unauthorized access affects validity, reliability, trustfulness, and authentication of the system. Several schemes are proposed for data sharing in IoT networks. The authors in [9] proposed a blockchain-based service sharing system to protect IoT terminals from unauthorized service providers. Despite authorization, trust-based sharing is not taken into consideration, which highly affects the reliability and validity of data sharing. Furthermore, the authors in [10] proposed a blockchain-based data sharing in AI-powered networks. This system works on trust-based sharing mechanism and is managed by smart contracts. However, the use of the system in all sharing scenarios is a challenging task. Furthermore, it does not provide trustfulness, and authentication-based data sharing. Hence, the aforementioned literature work lack to provide authenticated, authorized, and trust-based data sharing and access control among IoT devices.

1.4. Contributions

To overcome the aforementioned issues discussed in Section 1.3, a blockchain-based data sharing and access control system is proposed in this work. The proposed system aimed to achieve authorized, authenticated, permissioned and trust-based access control in the IoT networks. Furthermore, efficiency of the system is determined in terms of cost. The proposed system is an extension of the work in [17].

The main contributions of this work are as follows:

- a blockchain-based system is proposed for efficient data sharing, trustworthy and authorized access control among IoT devices' users,
- multiple smart contracts are used for efficient, secure, authorized, and trust-based access management of users in the network,
- secure and reliable data sharing is also achieved through smart contracts,
- one main smart contract; i.e., Access Control Contract (ACC) is used to manage the overall access control and sharing among users and also to enhance efficiency of the system in terms of cost,
- authentication of users in the network is maintained by Register Contract (RC) through users' registration and the record is maintained in the user registration table,

- misbehavior of users is checked by JC; after that, the corresponding penalty is determined,
- misbehavior of each user is recorded in the system by creating a subject misbehavior record table,
- different permission levels are set to provide permissioned access rights to users, and to achieve the trustful and reliable sharing,
- in addition, both transaction and execution costs of smart contracts and their functions are calculated, and
- at the end, a comparison is made between existing and proposed systems, which is given in Table 1.

Table 1. Cost Consumption Comparison of Benchmark and Proposed Systems.

Systems	Execution Cost	Transaction Cost
Access control system [1]	5,484,074 gas units	-
Data sharing model [10]	458,761 gas units	662,673 gas units
Proposed data sharing and access control system	5,200,000 gas units	6,900,000 gas units

Rest of the paper is organized as follows. In Section 2, a detailed literature review is presented. In Section 3, objectives of the system are presented. Furthermore, in Section 4, the proposed system is discussed in detail with its workflow. After that, the simulation results are described in Section 5. In Section 6, a comparison of existing and proposed system is given. In Section 7, the whole paper is concluded.

2. Related Work

Blockchain technology is intended to provide efficient management of data sharing and access control in the IoT networks. Several schemes are presented in the literature that integrate blockchain with IoT network.

The authors in [1] presented a smart contracts-based access control system to provide trustfulness and validation of users. However, this system lacked to provide the direct interaction between IoT devices and also compromised in terms of high cost. The authors in [2] presented a blockchain-based distributed access management architecture for IoT devices to achieve high mobility, concurrency, accessibility, and resiliency towards attacks. However, the proposed architecture does not provide authorized and authenticated access. The authors in [3] proposed an ABAC system for IoT networks in order to achieve less communication and computational overhead and enhanced flexibility, and efficient maintenance of system. Therefore, the consensus mechanism, i.e., PoC used in the system, performed efficiently for some parts of the system. The authors in [4] presented a blockchain consensus-based access control scheme to authenticate users through their provided Channel State Information (CSI). However, the system lacked to provide trustfulness and does not perform efficiently in non-cooperative environments. The authors in [5] designed a multiple blockchain-based cross-chain framework for access control management in IoT networks. The main objective of the framework is to achieve IoT devices' security. However, the privacy and trustfulness of users' information are not guaranteed efficiently. The authors in [6] proposed a novel Blockchain-based Distributed Key Management Architecture (BDKMA) to overcome the single point of failure issue. The system achieves scalability; however, the blockchain technology is not fully used. The authors in [7] presented an off-chain-based sovereign blockchain to ensure the security and effectiveness in access control. Therefore, the system does not provide trustful transactions, when it is integrated with industrial use cases.

The authors in [8] proposed a blockchain-enabled efficient data collection and secure sharing scheme to provide high quality sharing, safe and reliable environment for MTs, and resiliency towards attacks. However, the trustfulness and security of nodes are not ensured. The authors in [9] designed a novel blockchain-based service provisioning mechanism for Lw clients. The validity of on-chain and off-chain services is maintained through smart contracts. Therefore, the system lacked to provide authorized access control and secure sharing. The authors in [10] presented an AI-based trusted sharing network for mobile communications by implementing hyperledger fabric. The authors achieved trusted sharing through smart contracts and fine-grained access control. However, the scheme does not perform efficiently in all sharing environments. The authors in [11] proposed a reputation-based data sharing scheme. The goal is to enhance data sharing quality among network nodes, and to ensure security of data storage. However, the system is only efficient for small area networks. The authors in [12] designed a blockchain-based infrastructure for security and privacy-oriented service sharing system in IoT. The system is scalable that reduces its efficiency.

The authors in [13] proposed a blockchain-based secure data sharing system for vehicles to provide efficient incentives and to discard fake announcements by vehicles. Therefore, the ethereum blockchain used in the system lacked to achieve high throughput. The authors in [14] presented an encryption-based data sharing system using blockchain. The system maintained the integrity, privacy, and non-repudiation of data and achieved better encryption. However, it does not perform well for decryption. The authors in [15] proposed a Data security Sharing and Storage system based on Consortium Blockchain (DSSCB). The system achieved efficiency; however, authentication and security sharing do not perform better in real time evaluation. The authors in [16] proposed an Electronic Health Records (EHRs) system to address issue of sensitive medical data leakage and to ensure secure data access. Therefore, the real-world implementation of the system is not provided. The authors in [18] presented a bubble of trust mechanism to provide trustworthiness and confidentiality of data, identification and authentication of devices. The system is efficient in terms of cost; however, the communication between nodes in a bubble is not controlled. Furthermore, compromised devices are not eliminated. The authors in [19] presented a blockchain-based identity management system that works on a key management protocol based on Self Certified public Key-Based System (SCKBS). The system ensures several security requirements such as: authentication, confidentiality, and auditability. However, access mechanisms are not used efficiently. The authors in [20] presented a blockchain-based smart toy data exchange system to provide decentralized, secure, trusted, fair, and reliable data exchange. The system is efficient and ensures transparency and data confidentiality. Therefore, the data delivery time that is increased due to logs and high throughput are not handled. The authors in [21] presented an access control system based on eXtensible Access Control Markup Language (XACML) policies. The system ensures auditability of resources through generalized access control. However, privacy and auditability are not provided properly. Furthermore, permissioned blockchain used in the system does not work well. The authors in [22] proposed a blockchain-based mutual authentication system for Industry 4.0. The system aimed to provide efficient and secure access control. However, real implementation of the system in smart factory is not presented.

Furthermore, the authors in [23,24] addressed security issues in the networks. To overcome the issues, the authors proposed a blockchain-based trusted system for nodes' routing and recovery. The insecurity and untrustworthiness of data is also identified by the authors in [25–29]. To overcome the issues, blockchain-based systems are presented. The systems also provide the efficient use of devices data. Additionally, the authors in [30–32] presented a blockchain-based system to achieve trustfulness and authentication of data in the networks.

The related work is summarized in Table 2.

Table 2. Summary of Related Work.

Techniques	Problems Addressed	Contributions	Evaluations	Limitations
Blockchain [1]	Critical access control	Smart contracts-based access control	Gas price and misbehavior	Cost and overhead
DTLS protocol [2]	Security, centralization	Access management architecture	Throughput, latency, and response time	Single management hub
PoC, AES-128, PBFT [3]	Complexity and security	ABAC and hash operation	Overhead	Consensus mechanism not efficient
PBFT, CNN [4]	Unauthorized access	D2D underlying cellular networks	Channel rate and spectral efficiency	Non-cooperative scenarios
PoW, PBFT [5]	Data management and security	Decentralized access	Security and cost consumption	User privacy
PoW [6]	Untrustworthiness and centralization	Distributed key management system	Auditability and scalability	Not persistency
ECDSA, PoC [7]	Security and data mismanagement	Sovereign blockchain	Execution time	Inefficient in industries
DRL [8]	Limited MT resources and security	Secure sharing	Energy consumption and data collection ratio	Trustfulness and security
PoA, PoW [9]	Security and untrustworthy	Service provisioning mechanism	Delay, throughput, and gas consumption	Un-authorization
PBFT [10]	Complex system and AI bottleneck	Data sharing system	Security, privacy, and scalability	Lacks efficiency
ECDSA [11]	Constrained resources and trust	Reputation-based data sharing	Efficiency and security	Not suitable for large area networks
MEC tier, NBT, SHA-256 [12]	Insecure sharing	Certificate-less aggregate signature scheme	Tests on different scenarios	Lacks efficiency
PBFT [13]	Security and un-authorization	Secure data sharing	Security analysis and efficiency	Ethereum does not perform well
FBSS [14]	Bottleneck of ABE	Encryption-based sharing	Security analysis	Lack efficiency
PBFT, SHA-256 [15]	Untrustworthiness and communication overhead	Data security sharing and storage system	Safety analysis	Real-time authentication and message verification
PoW, SHA-256 [16]	Privacy leakage	Permission-based access	Execution time	Real world implementation
ECDSA [18]	Un-authentication and high cost	Decentralized authentication mechanism	Time and financial cost	Compromised devices
SCKBS [19]	Security risks	Identity-based access control mechanism	Energy consumption, requests per second	Inefficient access mechanisms
Chain-code algorithm [20]	Security, reliability, and privacy leakage	Data exchange system	Request speed and transaction time	Delivery time and throughput
XACML, PoC [21]	Resource protection	XACML-based access control system	Monetary cost, resource and time	Performance lacks
PBFT, cryptographic algorithms [22]	Security problems	Mutual authentication system	Transaction time	No real implementation

3. Objectives of the Proposed Access Control and Data Sharing System

In this section, the objectives of the proposed system for access control and data sharing among IoT devices' users are presented. Such objectives are given below.

- **Trustfulness:** The trust of IoT devices' users is maintained through a misbehavior-judging method, which is implemented by JC. The users who misbehave are known as untrusted users and are penalized. Only the trusted users are allowed to access their required services.
- **Authentication:** In this system, authentication of users is done by RC. Any user, who becomes part of the system is registered by RC. Furthermore, a record is maintained in the user registration table.
- **Authorized access:** In this system, only authorized users are allowed to access their required services. The users' authorization is maintained through permission levels.
- **Efficient data sharing:** The data sharing among IoT devices' users is maintained through the ACC. Which maintains the sharing of services among users. Furthermore, the permission-based sharing is also achieved through the permission levels.
- **Understanding of user behavior** [33]**:** In this system, the behavior of users is checked by implementing behavior judging method in JC. Misbehavior is conducted by subjects by sending too many requests. After identifying misbehavior, the requests of subjects are halted for a particular time. That is why, only the users who behave well are provided with the requested services.
- **Less cost consumption:** The system consists of three smart contracts, as in [1]. In the previous scheme, ACC is generated by users, for every transaction. It will increase the overall cost of the system. However, in this system, single ACC is used to control the access of the system. It also reduces the cost in terms of gas consumption of smart contracts, which is given in the comparison section.

4. Proposed System Model

A blockchain-based data sharing and access control system for IoT is proposed in this work, which is shown in Figure 2. This system is designed after getting motivation from work in [1,10]. Moreover, comparisons of the proposed and existing systems are made. The first comparison is made on the basis of scenarios, platforms, and tools used for simulations, as shown in Table 3. Another comparison is made between parameters of existing and proposed systems, which is shown in Table 4. In this system, data sharing is done between two network's users or peers called subject and object. The subject is a user that actually wants the data services. Whereas, the object contains services that are required by the subject. Services include: data, file, program, etc. Furthermore, three smart contracts are used to manage data sharing and access control among subjects and objects.

The components used in this system are: IoT devices, smart contracts (i.e., ACC, RC, and JC) as in [1], misbehavior judging method, data access permission. The detailed discussion of components is presented in Sections 4.1–4.3. Additionally, workflow of the system is also discussed in Section 4.4, which shows the overall working of the proposed system model.

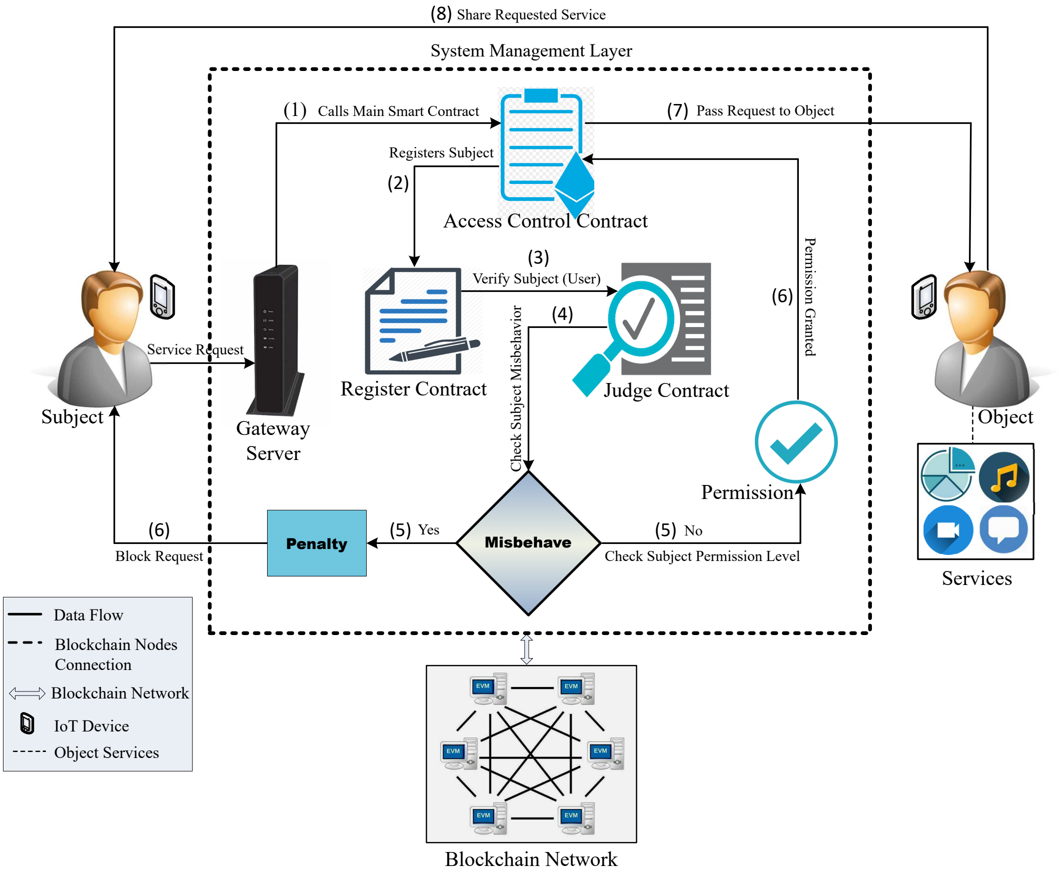

Figure 2. Proposed System Model.

Table 3. Comparison with Existing Works.

System Models	Scenarios	Implementations	Platforms	Simulation Environments
Access control system [1]	Access control in IoT	Remix IDE, Go language	Ethereum, raspberry pi	Javascript
Access control system [5]	Cross-chain based access control in IoT	-	IOTA and Tangle	Ubuntu 16.04
Access control [6]	Access control with key management in IoT	-	Multiple public blockchains	OMNeT++ 5.4.1
Secure data sharing [8]	Data sharing among MTs using deep learning	Go	Ethereum	Python
Data sharing model [10]	Data sharing in AI powered network	-	Hyperledger fabric	-
Proposed data sharing and access control model	Access control and data sharing	Solidity language	Ethereum, spyder	Remix IDE, python

Table 4. Parameters Comparison with Other Works.

Systems	Trustfulness	Authorization	Authentication	Validation	Decentralized	Reliability
Access control system [1]	Yes	No	No	Yes	Yes	No
Access control system [5]	No	No	No	No	Yes	No
Access control system [6]	No	Yes	No	No	Yes	Yes
Data sharing system [8]	No	No	Yes	Yes	Yes	Yes
Data sharing system [10]	Yes	No	No	No	Yes	No
Proposed data sharing and access control system	Yes	Yes	Yes	Yes	Yes	Yes

4.1. Smart Contracts

In the proposed system, multiple smart contracts are used to manage data and service sharing among network users. The smart contracts are: ACC, RC, and JC. Where ACC manages the overall access control of the system, the RC is used to register users (subjects and objects) in the system. It also generates a registration table, which stores the information of users. Users' authentication and authorization are also maintained by the registration table. Furthermore, behavior of subjects is determined by the JC. It checks if any misbehavior is conducted by the subject or not. When a subject sends too many requests or cancels the generated request, it is considered to be misbehavior. After the misbehavior conduction, the penalty is imposed on subject by the JC. Hence, trustfulness of the subject is determined by its behavior. If no misbehavior is done by the subject, then permission levels are checked. After that, the request of subject is passed by ACC to the corresponding object, to get the required service. Moreover, a detailed description of ACC, RC, and JC is given below.

4.1.1. ACC

It is the main smart contract that manages the access control between IoT devices. Whenever the subject requires any service from the object, it sends a request to the system. After that, ACC maintains the access control for subject. It also increases the performance efficiency of system. Whereas, in the benchmark system [1], multiple ACCs are deployed by objects for each request. In this system, access control is done by the user instead of the system itself. Each time a request is generated by the subject, an ACC is deployed by the object in response to that request. In a result, complexity and cost of the system are increased. Moreover, processing time of the system is also effected. However, in the proposed data sharing and access control system, access control among users is managed by a single ACC. When a subject sends service request into the system, ACC is executed. Other smart contracts are executed for registration and authorization of the users. After registration of the user, ACC forwards the request of subject to the object by checking corresponding permission level.

4.1.2. RC

The users who are intended to access the services should be legitimate. For that, the authentication of users is done by RC, by registering them in the system. For registration, RC maintains the registration table, as in [1]. In the table, all information of the users is stored. The registration table generated by RC is shown in Table 5. The information of users stored in the registration table are: subject, object, service, and time. Moreover, verification and authentication of the subjects are also maintained by registration table. The registration table consists of the following information:

- *subject*: the particular user that sends the service request,
- *object*: user that contains required services and entertains the request of the subject,
- *service*: particular data or service, which is requested by the subject, and
- *time*: the time at which a request is generated.

Table 5. User Registration Table.

Subject	Object	Service	Time
Subject A	Object X	File-1	2019/5/17 11:12
Subject B	Object Y	Program-2	2019/6/14 1:15
Subject C	Object Z	File-3	2019/8/8 3:00

4.1.3. JC

A judging method is implemented by the JC, which judges the behavior of users in the system. When a subject sends service request in the system, its behavior is verified by the JC. The misbehavior is always conducted by the subject. When the subject sends frequent and too many requests for a service, it is considered to be misbehaving. Additionally, if the subject cancels its generated request, it is also known as misbehavior. After that, corresponding penalty is determined for a subject who misbehaves. In penalty, the state of subject is turned off for a particular time. When the state of a subject is off, it cannot send request to the system. On the other hand, if the subject has not misbehaved, then the permission levels are checked. The request of subject is granted or denied, according to its behavior and permission levels. After that, alert messages are generated by the JC.

The alerts generated for access are as follows (! indicates the alert messages):

- *Access Authorized !*
- *Requests are Blocked !*
- *Static Check Failed !*
- *Misbehavior Detected !*
- *Static Check failed & Misbehavior Detected !*

If no misbehavior is conducted by the subject, then access is granted. JC generates an alert message of access authorization. Whereas, if there is any misbehavior conducted by the subject, other alert messages such as: permission denied or access blocked are generated by the JC. These alerts show that permission is not granted to the subject and a misbehavior is conducted.

4.2. Misbehavior

In the data sharing system, misbehavior is conducted by subjects. When a subject sends too frequent access requests, it is considered as misbehavior. As a result, trust of the system is highly effected. Furthermore, misbehavior highly affects the performance and efficiency of the system. If multiple requests are sent by one user, then all network traffic is only consumed by that particular user. Other issues may occur in IoT networks like: congestion, latency, etc. Several types of misbehaviors that are done by the subject are:

- subject sends too frequent service access requests,
- subject sends multiple access requests for service(s) in a particular time, i.e., 5 requests in 10 min, and
- the request is canceled after generation.

The misbehavior of subject is determined by a misbehavior judging method, implemented by JC. Which maintains trustfulness of the system. Furthermore, a misbehavior field is maintained by the JC, which is shown in Table 6. The field records all misbehaviors conducted by subjects. In the result of misbehavior, penalty for subjects is determined by the JC. In penalty, requests of a subject are halted. In the halted state, subject is no more able to send requests in network for a certain time period.

Table 6. Subject Misbehavior Record.

Subject	Object	Misbehavior	Misbehavior-Time	Penalty
Subject-A	Object-X	Multiple requests in 3 min	2019/3/27 11:12	Request halted for 2 h
Subject-B	Object-Y	Canceled request	2019/4/1 1:05	Request halted for 10 min
Subject-C	Object-Z	Frequent requests	2019/4/3 3:09	Request halted for 3 h

4.3. Data Permission Control

The proposed system is intended to provide authorized access control and permission-based sharing of services. Several permission levels are set for subject to access services of the object. The permission levels determine privileges for the user to access services. These levels are defined according to the sensitivity level of data. Moreover, the permission is granted to subject based on its privilege level; what type of data is requested by the subject. The data permission levels are as follows:

- *L0: Data is not accessible,*
- *L1: Data can be used in aggregated computation without revealing the raw data [10],*
- *L2: Data is partly allowed, and*
- *L3: Data or service is accessible.*

Hence, all of the above-mentioned levels determine the acceptance or denial of users' access request. L0 defines that the subject is not legitimate to avail the requested service. Such services can be personal information of object. Legitimacy of the subject is determined by the object. In L1 and L2, service is shared with some restrictions of usage. In L3, service is fully accessible to subject.

4.4. Workflow of System Model

The logical flow of the proposed system is presented, which is shown in Figure 3. The data sharing system works as follows:

- at fist, a subject sends request for any service (i.e., data, file, storage unit) in the IoT networks,
- further, communication and access management for subject is managed by the smart contracts in blockchain,
- when the request of subject is generated, ACC (main smart contract) is executed to control the overall access management,
- after that, authentication of the users is done by the RC, which registers the users' information and maintains records of users via a registration table,
- then, trustfulness of the system is maintained by JC, which checks the behavior of a subject,
- if any type of misbehavior is conducted by the subject, JC determines penalty for the subject and halt its state in the system,
- in another case, if no misbehavior is done, then permission level for the subject is checked,
- then ACC forwards the request of subjects to corresponding objects,
- after that, request of the subject is fulfilled by the object, and
- at last, transaction is stored into the blockchain.

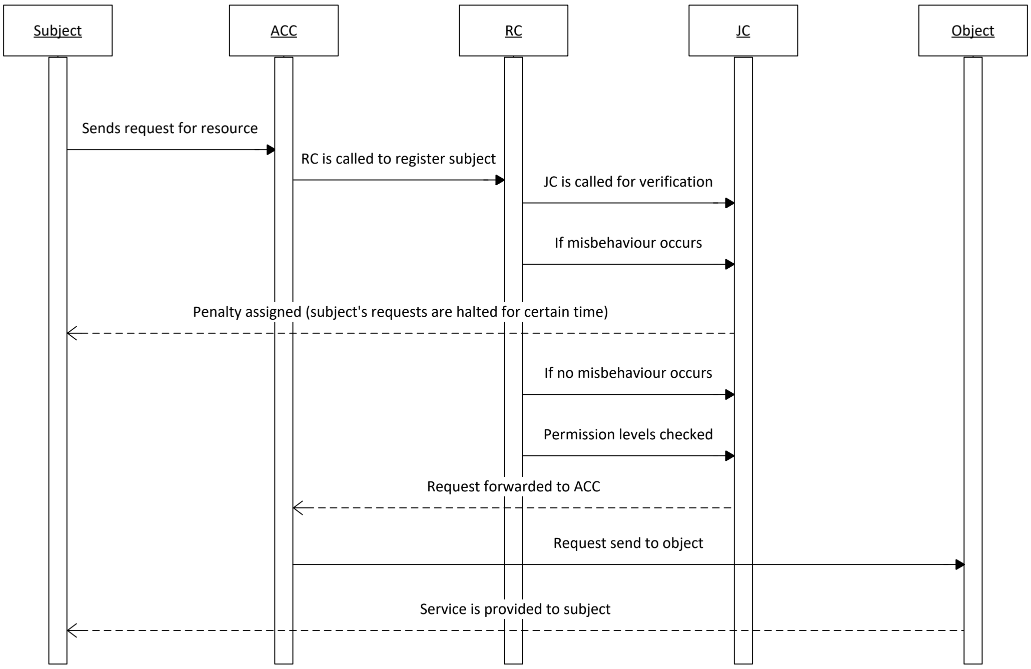
Subject
ACC
RC
JC
Object
Sends request for resource
RC is called to register subject
JC is called for verification
If misbehaviour occurs
Penalty assigned (subject's requests are halted for certain time)
If no misbehaviour occurs
Permission levels checked
Request forwarded to ACC
Request send to object
Service is provided to subject

Figure 3. Flow Diagram of Proposed System Model.

5. Simulations and Results

In this section, simulation results for cost consumption are discussed. Ethereum blockchain is used in this system; for that, cost of smart contracts and their functions is calculated in terms of gas usage. Furthermore, the ether value is also checked for each gas unit. Further sections describe the simulation environment and the cost consumption of each smart contract and its functions.

5.1. Simulations Setting

All the simulations are performed on Intel Core i5, CPU 2.50 GHz with 4 GB RAM, running on Windows 10. The smart contracts are written in solidity language. Solidity is an object-oriented programming language, used to implement smart contracts on blockchain platforms, mostly in ethereum blockchain. Ethereum is an open-source, public blockchain platform for the execution of smart contracts. In ethereum, all computational tasks are performed by smart contracts [16].

Furthermore, truffle environment is used for smart contracts development. Truffle is the development environment and testing framework, used for ethereum blockchain. It makes an easy compilation, deployment, and management of the smart contracts. Moreover, the graphs for cost consumption of smart contracts are taken in python language using spyder platform.

5.2. Cost Consumption

The cost consumption of system is calculated in terms of gas units consumed by smart contracts. Gas is a measurement unit, which measures the computational power for execution of transactions in ethereum platform. Gas price is defined by the miners at start of the transaction and is measured in Gwei. Moreover, gas units are calculated for execution of smart contracts and their functions. In ethereum blockchain, gas units are converted into the ether value (also written as eth) or fiat money. Eth is the fuel of ethereum blockchain.

The gas units are converted into eth value. The gas price is set as 20 Gwei. Furthermore, the cost of Gwei is calculated by multiplying gas units with gas price. After that, the amount is divided by the unit of single ether, i.e., 1 ether = 1,000,000,000 Gwei.

Transaction cost: It is the cost of sending smart contract's code to the ethereum blockchain. It depends on the size of smart contract. The size of smart contract is based on the computational tasks it performs. For example: if the smart contract contains high computational tasks; then it is large in size and the transaction cost is also high. Furthermore, the transaction cost consists of: transaction cost, contract deployment cost, and transaction data cost.

Execution cost: It is the cost of storing global variables and method calls of the smart contracts. It also depends on the computational operations performed in terms of transaction execution.

The gas is calculated by three things: gas cost, gas price, and gas limit. Gas cost is the number of units that are needed to perform any action in the ethereum network. Gas price is the value of one unit, which is measured in ether. Gas limit is the amount of gas, which is paid by participants of the network.

5.2.1. **Smart Contracts Cost**

The transaction and execution costs of smart contracts such as: ACC, RC, JC are illustrated in Figure 4. As it is shown in the aforementioned figure, gas units consumed by ACC are higher than other smart contacts, i.e., RC and JC. It is obvious that ACC is the main smart contract that manages the overall access control and transactions among two IoT devices and it performs more computational tasks. Tasks performed by the ACC are access control management, granting service permission to users, and transactions management. Therefore, more gas units are consumed by the ACC. After that, RC consumes more gas units to perform user registration task and also creates user registration table. All users' records are maintained in the user registration table. RC only manages registration; that is why it has less cost consumption than ACC. Furthermore, JC has less cost consumption rate. As the purpose of JC is to check the behavior of users and to assign the penalty to users who misbehave. The misbehavior is recorded in the form of table. Moreover, the gas units are converted into ether, as it is the cryptocurrency of ethereum blockchain. The gas to eth conversion is done by the criteria given above in Section 5.2.

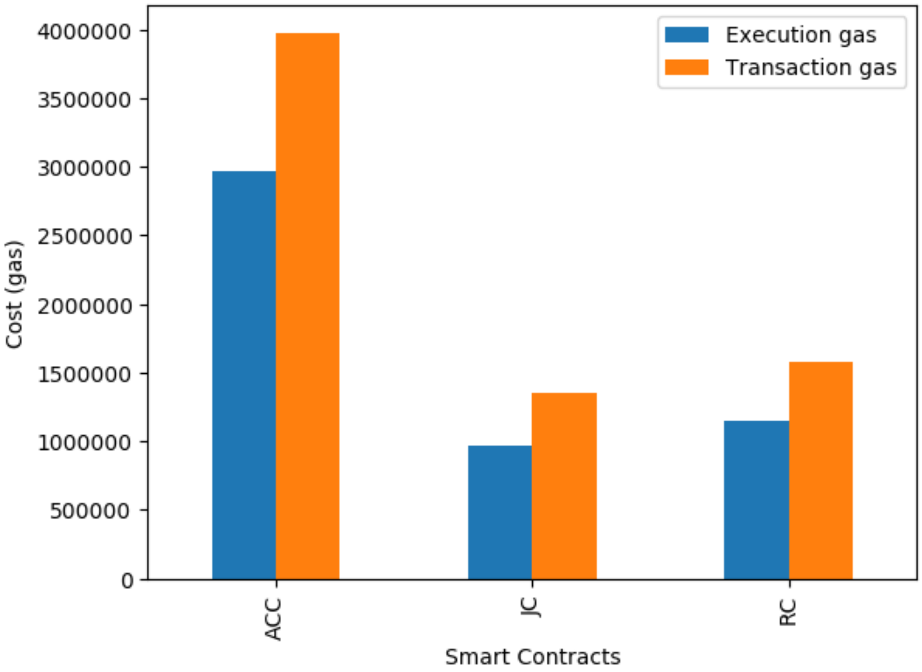

Figure 4. Smart Contracts Cost.

Transaction cost: The transaction cost of smart contracts: ACC, RC, and JC is shown in the above figure. It is clear from the figure that the transaction cost of ACC is much more than RC and JC. It manages the overall access control of the system and it contains more functions than other smart contracts. Therefore, more cost is used according to the size of smart contract. The transaction cost of ACC is 4,000,000 gas units, which is equal to 0.08 eth. The transaction cost of RC is 1,600,000 gas units, which is equal to 0.032 eth. The cost of RC is not much high, because it only performs registration operations of users and creates a user registration table. Additionally, the transaction cost of JC is 1,300,000 gas units, which is equal to 0.026 eth. JC consumed less transaction cost as it implements the judging methods for users.

Execution cost: The execution cost of smart contracts is also shown in the figure. Likewise the transaction cost, the execution cost of ACC is also higher than other smart contracts. The cost is 3,000,000 gas units, which is equal to 0.06 eth. ACC contains more computational operations, function calls and global variables to control access of the system. After ACC, the execution cost of RC is high, i.e., 1,200,000 gas units that is equal to 0.024 eth. JC consumes less cost as compare to other smart contracts, which is 1,000,000 gas units and is equal to 0.02 eth.

5.2.2. **Functions Cost**

Each smart contract contains various functions to manage access control among IoT devices. In this system, both transaction and execution cost for the functions of each smart contract are calculated. The costs are calculated to show the number of gas units consumed by each function of smart contracts. The cost consumption rate of each function is discussed below.

Functions of ACC: The transaction cost and execution cost for each function of ACC are shown in Figure 5. It contains more function calls as compare to other two smart contracts, because ACC manages the access control of whole system. However, the gas consumption is calculated for main functions of the smart contract. The functions of ACC that perform several operations include user registration, generating permission levels for the subject, and data access function. The transaction and execution costs of the functions are given below.

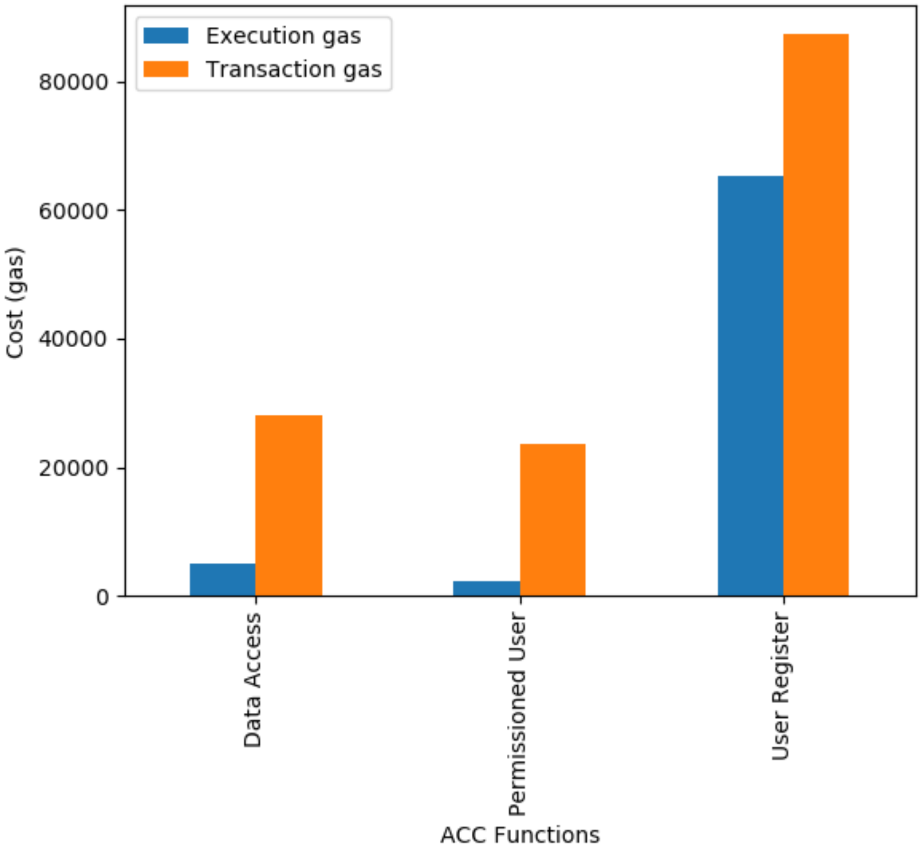

Figure 5. ACC Function Cost.

Transaction cost: As it is illustrated in the aforementioned figure, transaction cost of the user registration function is higher than other functions. The user register function consumes 89,000 gas units, which are equal to 0.00178 eth value. All users are registered in the network for authentication. Furthermore, the data access function consumes 30,000 gas units, which are equal to 0.0006 eth values. In the access control function, requests of subjects are passed by ACC to the corresponding objects. After that, gas consumed by permission level function is 25,000 gas units, which is 0.0005 eth. This function is called after judging behavior of the users. The privileges are checked for users to access the required service, for this reason, it consumed very less cost.

Execution cost: The execution cost of ACC functions is also given in the above-mentioned figure. The execution cost of user register function is 65,000 gas units and its ether value is 0.0013 eth. The registration function also consumes more execution cost. Furthermore, the function that grants data access to the users consumes 9,000 gas units that are equal to 0.00018 eth value. Moreover, the function for setting permission levels for the users consumes very less cost, i.e., 5,000 gas units that are equal to 0.0001 eth. Hence, the functions with high cost consumption are executed more than the functions that cost less.

Functions of RC: The transaction and execution costs of RC functions are shown in Figure 6. The objective of RC is to manage the registration of users in the network, and to maintain a registration table to record the users' information. The functions of RC are user registration and registration table creation.

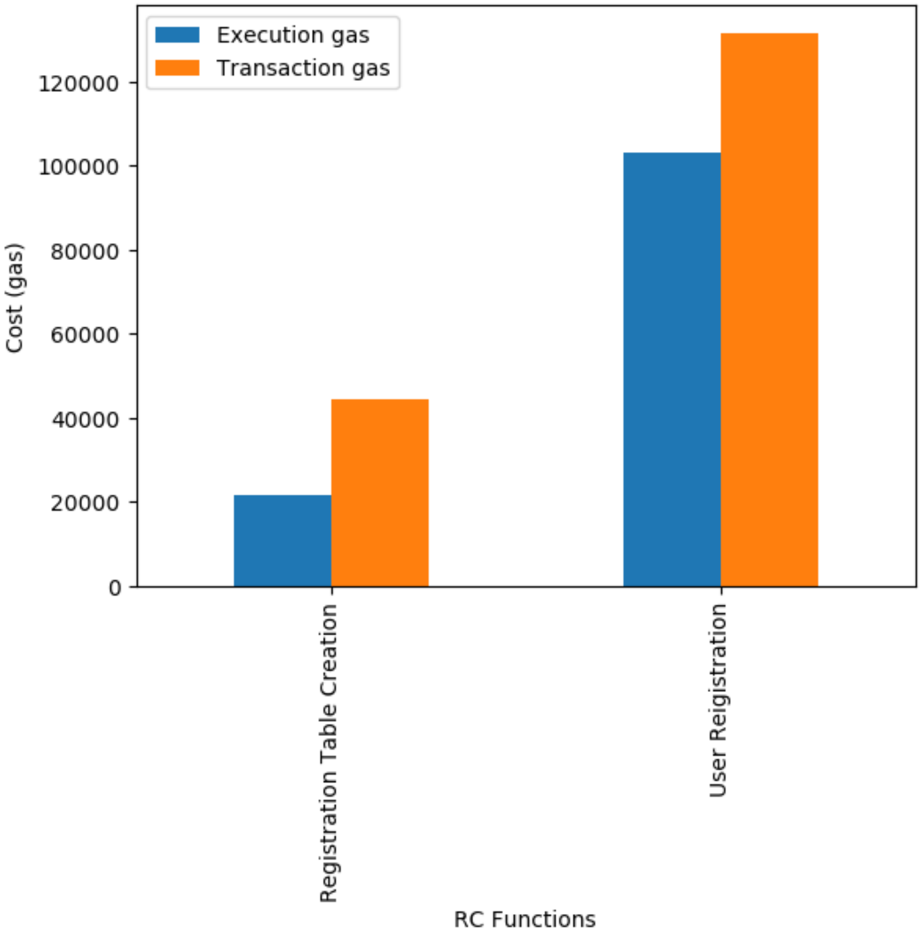

Figure 6. RC Function Cost.

Transaction cost: The transaction cost of RC functions shows that user registration function consumes more transaction cost. The cost consumed is 130,000 gas units that is equal to 0.0026 eth. The function registers the users in the network. Furthermore, the registration table is maintained by the register table function. The cost of registration table function is 45,000 gas units, which is equal to 0.0009 eth values.

Execution cost: Furthermore, the execution cost of RC functions is also calculated. The cost of registration function is 130,000 gas units that is equal to 0.0026 eth. The registration table consumes 23,000 gas units, which are equal to 0.00046 eth value.

Functions of JC: In Figure 7, the transaction and execution costs of JC functions are given. The JC is used to maintain trustworthiness of the system by checking behavior of each user. The behavior judging method is implemented by JC, which checks if any misbehavior is conducted by the subjects or not. After that, the penalty is determined for subject. The functions of JC are misbehavior calculation function and misbehavior judge function.

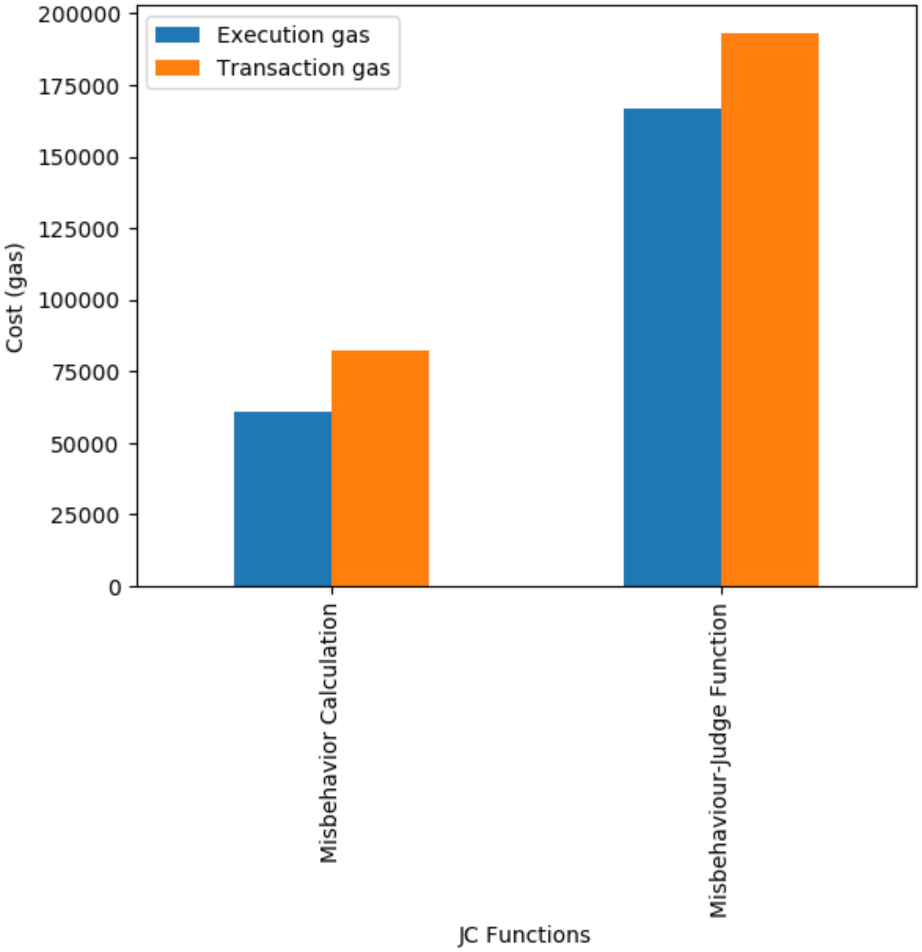

Figure 7. JC Function Cost.

Transaction cost: The aforementioned figure illustrated the transaction cost for functions of JC. The cost of misbehavior judging function is more than other functions, which is 195,000 gas units. The gas units are equal to 0.0039 eth values. As the misbehavior of users and penalty is determined by this function; that is why it costs more. Furthermore, the transaction cost of misbehavior calculation function is 80,000 gas units. The gas units are equal to 0.0016 eth value.

Execution cost: Moreover, the execution cost of JC functions is also shown in the above figure. The misbehavior judge function also consumes more execution cost. The cost is 165,000 gas units that are equal to the 0.0033 eth value. The cost of misbehavior calculation function is 60,000 gas units, which is equal to 0.0012 eth.

6. Analysis and Comparison

In this section, a comparison is made to evaluate the performance of proposed system in terms of cost consumption of smart contracts.

6.1. Comparison

In this section, a comparison is made between existing and proposed systems. One of the objective of this system is to reduce the cost for execution of the smart contracts. The cost is given in the form of gas consumption of ethereum smart contracts. In the above Section 5, the execution and transaction costs of each smart contract and its functions are calculated. However, in this section, comparison is made between cost consumption of smart contracts in [1] and in the proposed system. The cost consumption rate is given in Table 1, which illustrates the transaction and execution costs of smart contracts. The table shows the overall cost of systems for a single transaction. As it is given in the table that less cost is obtained by the smart contract in [10]. The reason is that the system contains a single smart contract that manages the sharing among users. Therefore, in [1] and proposed system, three smart contracts are used to manage the access control among users. The smart contracts in [1] consumes more cost. Because ACC is generated by the user (subject or object) for each transaction. Whenever ACC is generated, the cost for executing the smart contract is used, which increase the overall system cost. The execution cost of ACC for a single transaction is 2,543,479, as given in [1]. For multiple transactions, the overall cost of the system will increase. Also the overhead of the system is increased. However, in the proposed system, the ACC is not created for each transaction. It is a single smart contract that manages the service sharing among users. Furthermore, the overhead of system is also reduced.

The proposed system is efficient and performs better in terms of different parameters.

- **Availability:** Any user can easily access the system, whenever it requires any service. As the public blockchain is open and available. The users can easily become part of the system.
- **Trustfulness:** The trustfulness of users is maintained through the JC smart contract. JC implements the behavior judging method that judges the behavior of each subject. If the misbehavior is conducted by subject, it is known as untrusted and its requests are halted for a particular time period. In another case, if the misbehavior is not conducted, then subject is considered as trusted.

7. Conclusions

The major challenges in IoT networks are the provisioning of authenticated, trusted data sharing, and authorized access control among IoT devices. Therefore, in this paper, blockchain technology is integrated with the data sharing and access control system. Main objectives of this system are to achieve authentication, authorization, and trustfulness. With the integration of blockchain technology, several benefits are brought into the networks. To achieve the aforementioned objectives, multiple smart contracts are used in this system. The smart contracts are ACC, RC, and JC. Where ACC maintains the access control between IoT devices and manages the sharing of data among them. Furthermore, RC is called to manage the authentication of users in the system by registering them. The users' information is recorded in the user registration table. Moreover, the misbehavior judging mechanism is implemented in this system. JC is used to check the behavior of users. Whenever a misbehavior is conducted by the subject, the corresponding penalty is determined. Besides that, if no misbehavior is conducted by the subject, the privileges are set for that and access is granted. Furthermore, simulations are done in terms of cost consumption of smart contracts and their functions. The cost is calculated in terms of gas units. After that, a comparison is made between the existing and proposed systems. Results show that the system is efficient in terms of cost, as the access control is managed by ACC. The proposed system is less complex for access control management among IoT devices.

Author Contributions: Tanzeela Sultana and Nadeem Javaid proposed and implemented the main idea. Ahmad Almogren and Mariam Akbar wrote the simulation section. Mansour Zuair and Ibrar Ullah organized and refined the manuscript. All authors worked together and responded to the honourable reviewers' comments. All authors have read and agreed to the published version of the manuscript.

Acknowledgments: This work was supported by the Deanship of Scientific Research at King Saud University under Grant RGP-1437-35.

Conflicts of Interest: The authors declare no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript:

ABAC	Attribute-Based Access Control
ABE	Attribute-Based Encryption
ACC	Access Control Contract
AES	Advanced Encryption Standard
AI	Artificial Intelligence
CNN	Conventional Neural Network
CSI	Channel State Information
D2D	Device-to-Device
DES	Data Encryption Standard
DRL	Deep Reinforcement Learning
DSA	Digital Signature Algorithm
DSSCB	Data security Sharing and Storage system based on Consortium Blockchain
DTLS	Datagram Transport Layer Security
ECDSA	Elliptic Curve Digital Signature Algorithm
EHRs	Electronic Health Records
FBSS	Fair Blind Signature Scheme
IDE	Integrated Development Environment
IoT	Internet of Things
JC	Judge Contract
Lw	Lightweight
MTs	Mobile Terminals
NBT	Naive Bayes Theorem
P2P	Peer-to-Peer
PBFT	Practical Byzantine Fault Tolerance
PoA	Proof of Authority
PoC	Proof of Concept
PoS	Proof of Stake
PoW	Proof of Work
RC	Register Contract
RSA	Rivest-Shamir-Adleman
SCKBS	Self Certified public Key Based System
SHA	Secure Hash Algorithm
XACML	eXtensible Access Control Markup Language

References

1. Zhang, Y.; Kasahara, S.; Shen, Y.; Jiang, X.; Wan, J. Smart Contract-Based Access Control for the Internet of Things. *IEEE Internet Things J.* **2018**, *6*, 1594–1605.
2. Novo, O. Scalable Access Management in IoT Using Blockchain: A Performance Evaluation. *IEEE Internet Things J.* **2018**, *6*, 4694–4701.
3. Ding, S.; Cao, J.; Li, C.; Fan, K.; Li, H. A Novel Attribute-Based Access Control Scheme Using Blockchain for IoT. *IEEE Access* **2019**, *7*, 38431–38441.
4. Lin, D.; Tang, Y. Blockchain Consensus Based User Access Strategies in D2D Networks for Data-Intensive Applications. *IEEE Access* **2018**, *6*, 72683–72690.
5. Jiang, Y.; Wang, C.; Wang, Y.; Gao, L. A Cross-Chain Solution to Integrating Multiple Blockchains for IoT Data Management. *Sensors* **2019**, *19*, 2042.
6. Ma, M.; Shi, G.; Li, F. Privacy-Oriented Blockchain-Based Distributed Key Management Architecture for Hierarchical Access Control in the IoT Scenario. *IEEE Access* **2019**, *7*, 34045–34059.
7. Sifah, E.B.; Xia, Q.; Agyekum, K.O.-B.O.; Amofa, S.; Gao, J.; Chen, R.; Xia, H.; Gee, J.C.; Du, X.; Guizani, M. Chain-based big data access control infrastructure. *J. Supercomput.* **2018**, *74*, 4945–4964.
8. Liu, H.C.; Lin, Q.; Wen, S. Blockchain-enabled Data Collection and Sharing for Industrial IoT with Deep Reinforcement Learning. *IEEE Trans. Ind. Inform.* **2018**, *15*, 3516–3526.

9. Xu, Y.; Wang, G.; Yang, J.; Ren, J.; Zhang, Y.; Zhang, C. Towards Secure Network Computing Services for Lightweight Clients Using Blockchain. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 1–12.
10. Zhang, G.; Li, T.; Li, Y.; Hui, P.; Jin, D. Blockchain-Based Data Sharing System for AI-Powered Network Operations. *J. Commun. Inf. Netw.* **2018**, *3*, 1–8.
11. Kang, J.; Yu, R.; Huang, X.; Wu, M.; Maharjan, S.; Xie, S.; Zhang, Y. Blockchain for secure and efficient data sharing in vehicular edge computing and networks. *IEEE Internet Things J.* **2019**, *6*, 4660–4670.
12. Rahman, A.; Rashid, M.; Hossain, M.S.; Hassanain, E.; Alhamid, M.F.; Guizani, M. Blockchain and IoT-Based Cognitive Edge Framework for Sharing Economy Services in a Smart City. *IEEE Access* **2019**, *7*, 18611–18621.
13. Zhang, L.; Luo, M.; Li, J.; Au, M.H.; Choo, K.-K.R.; Chen, T.; Tian, S. Blockchain based secure data sharing system for Internet of vehicles: A position paper. *Veh. Commun.* **2019**, *16*, 85–93.
14. Wu, A.; Zhang, Y.; Zheng, X.; Guo, R.; Zhao, Q.; Zheng, D. Efficient and privacy-preserving traceable attribute-based encryption in blockchain. *Ann. Telecommun.* **2019**, *74*, 401–411.
15. Zhang, X.; Chen, X. Data Security Sharing and Storage Based on a Consortium Blockchain in a Vehicular Ad-hoc Network. *IEEE Access* **2019**, *7*, 58241–58254.
16. Chen, L.; Lee, W.-K.; Chang, C.-C.; Choo, K.-K.R.; Zhang, N. Blockchain based searchable encryption for electronic health record sharing. *Future Gener. Comput. Syst.* **2019**, *95*, 420–429.
17. Lin, C.; He, D.; Huang, X.; Choo, K.-K.R.; Vasilakos, A.V. BSeIn: A blockchain-based secure mutual authentication with fine-grained access control system for industry 4.0. *J. Netw. Comput. Appl.* **2018**, *116*, 42–52.
18. Sultana, T.; Ghaffar, A.; Azeem, M.; Abubaker, Z.; Gurmani, M.U.; Javaid, N. Data Sharing System Integrating Access Control Based on Smart Contracts for IoT. In Proceedings of the International Conference on P2P, Parallel, Grid, Cloud and Internet Computing, Antwerp, Belgium, 7–9 November 2019; Springer: Cham, Switzerland, 2019; pp. 863–874.
19. Hammi, M.T.; Hammi, B.; Bellot, P.; Serhrouchni, A. Bubbles of Trust: A decentralized blockchain-based authentication system for IoT. *Comput. Secur.* **2018**, *78*, 126–142.
20. Ren, Y.; Zhu, F.; Qi, J.; Wang, J.; Sangaiah, A.K. Identity Management and Access Control Based on Blockchain under Edge Computing for the Industrial Internet of Things. *Appl. Sci.* **2019**, *9*, 2058.
21. Yang, J.; Lu, Z.; Wu, J. Smart-toy-edge-computing-oriented data exchange based on blockchain. *J. Syst. Arch.* **2018**, *87*, 36–48.
22. Maesa, D.D.F.; Mori, P.; Ricci, L. A blockchain based approach for the definition of auditable Access Control systems. *Comput. Secur.* **2019**, *84*, 93–119.
23. Mateen, A.; Javaid, N.; Iqbal, S. Towards Energy Efficient Routing in Blockchain Based Underwater WSNs via Recovering the Void Holes. Master's Thesis, COMSATS University Islamabad (CUI), Islamabad, Pakistan, 2019.
24. Khan, R.J.H.; Javaid, N.; Iqbal, S. Blockchain Based Node Recovery Scheme for Wireless Sensor Networks. Master's Thesis, COMSATS University Islamabad (CUI), Islamabad, Pakistan, 2019.
25. Naz, M.; Javaid, N.; Iqbal, S. Research Based Data Rights Management Using Blockchain Over Ethereum Network. Master's Thesis, COMSATS University Islamabad (CUI), Islamabad, Pakistan, 2019.
26. Noshad, Z.; Javaid, N.; Imran, M. Analyzing and Securing Data using Data Science and Blockchain in Smart Networks. Master's Thesis, COMSATS University Islamabad (CUI), Islamabad, Pakistan, 2019.
27. Ali, I.; Javaid, N.; Iqbal, S. An Incentive Mechanism for Secure Service Provisioning for Lightweight Clients Based on Blockchain. Master's Thesis, COMSATS University Islamabad (CUI), Islamabad, Pakistan, 2019.
28. Samuel, O.; Javaid, N.; Awais, M.; Ahmed, Z.; Imran, M.; Guizani, M. A Blockchain Model for Fair Data Sharing in Deregulated Smart Grids. In Proceedings of the IEEE Global Communications Conference (GLOBCOM), Big Island, HI, USA, 9–13 December 2019.
29. Rehman, M.; Javaid, N.; Awais, M.; Imran, M.; Naseer, N. Cloud based Secure Service Providing for IoTs using Blockchain. In Proceedings of the IEEE Global Communications Conference (GLOBCOM), Big Island, HI, USA, 9–13 December 2019.
30. Javaid, A.; Javaid, N.; Imran, M. Ensuring Analyzing and Monetization of Data Using Data Science and Blockchain in LoT Devices. Master's Thesis, COMSATS University Islamabad (CUI), Islamabad, Pakistan, 2019.

31. Kazmi, H.S.Z.; Javaid, N.; Imran, M. Towards Energy Efficiency and Trustfulness in Complex Networks Using Data Science Techniques and Blockchain. Master's Thesis, COMSATS University Islamabad (CUI), Islamabad, Pakistan, 2019.
32. Zahid, M.; Javaid, N.; Rasheed, M.B. Balancing Electricity Demand and Supply in Smart Grids using Blockchain. Master's Thesis, COMSATS University Islamabad (CUI), Islamabad, Pakistan, 2019.
33. T.; Pota, H.; Chu, C.-C.; Gadh, R. Real-time renewable energy incentive system for electric vehicles using prioritization and cryptocurrency. *Appl. Energy* **2018**, *226*, 582–594.