Blockchain based Data Integrity Framework for Internet of Things

Poornima M. Chanal (poornima.chanal@yahoo.com)
Basaveshwar Engineering College (Autonomous), Bagalkot, Karnataka, India
Mahabaleshwar S. Kakkasageri
Basaveshwar Engineering College (Autonomous), Bagalkot, Karnataka, India

Research Article

Keywords: Internet of Things, Data integrity, Blockchain

Posted Date: May 16th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1641782/v1

Title: Blockchain based Data Integrity Framework for Internet of Things

Authors: Poornima M. Chanal*, Mahabaleshwar S. Kakkasageri**

Affiliation and Address of the Authors:
Department of Electronics and Communication Engineering
Basaveshwar Engineering College (Autonomous)
Bagalkot-587102, Karnataka, India

*ORCID ID: 0000-0002-6906-6548

**ORCID ID: 0000-0001-6687-1093

E-mail ID: *pmcec@becbgk.edu, **mskec@beck.edu

Contact numbers: *+91 9538061707, **+91 9482861933

***Corresponding Author:**
Poornima M. Chanal
E-mail ID: pmcec@becbgk.edu
Contact number: +91 9538061707

Blockchain based Data Integrity Framework for Internet of Things

Abstract

The Internet of Things (IoT) is one important digital revolution in both the academic world and commerce. It brings convenience to people's daily lives; but, the challenges of security and privacy have become a very big task in IoT. Attaining safety and data integrity verification of information in IoT networks is one of the hot topics for future IoT applications. The provision of security to any information in an IoT network is a big challenge, which must be given the top priority for many current and future applications of IoT. Traditional data integrity verification approaches only use encryption algorithms to secure information, depending on Third-Party Auditors (TPAs). The blockchain's basic principle is that information produced by users or nodes is tested for accuracy and cannot be altered once it is updated on the blockchain. Blockchain based data integrity schemes can effectively overcome TPA's issues. In this paper, we propose a blockchain based data integrity technique with a bilinear design for IoT information. We achieve data integrity according to the characteristics of bilinear design in the form of blockchain communications. The proposed blockchain based framework for data integrity verification consists of different entities such as client, Key Generation Centre (KGC), cloud storage server, and blockchain. Proposed data integrity verification operates in three stages: Setup stage, Processing stage and Verification stage. Simulation results show that the performance of the proposed blockchain approach for data integrity in IoT is better as compared with the blockchain based cloud data integrity verification scheme with high efficiency (DICF). The results demonstrate that our scheme is better interms of signature generation time, end to end delay, memory utilization as compared to DICF.

Keywords: Internet of Things, Data integrity, Blockchain

1 Introduction

The Internet of Things (IoT) has become one of the very important technologies in the past decade. IoT devices can give limitless benefits to computational systems. Many IoT devices are embedded with the electronic chipset, software, sensors, and actuators has the capability to gather, exchange and transfer information through Internet without human intervention. IoT aims to build smart and self-conscious independent devices for smart cities, smart hospitals, intelligent transportation systems, etc.,

[1]-[3]. In the business field, IoT gives a good insight for different sectors like IoT developers, integrators, network operators and software dealers [4]-[6]. Scalability and other constraints on IoT device potential conclude that traditional cryptography algorithms, security and privacy schemes are insufficient [7] [8].

The data integrity verification concept with false data filtering protocol was proposed by Deswarte [9]. Integrity may assure the information cannot be altered by unauthorized or accidental intervention with communication networks during information transfer. Only the authorized users should get accurate information. Blockchain technology is a decentralized ledger, which increases security, trust and transparency in IoT networks. The blockchain is one chain structure consisting of original data, hash function blocks, timestamps, merkle tree and consensus protocols. In chain structure, a block consists of its hash function and merkle tree nodes. The chain structure is formed by transferring the hash function from the previous block into the new block's header [10]. In the blockchain, cryptography based digital signature algorithms provide security to user transactions. The consensus protocols protect the data from unauthorized people and gives secure data.

The problem with IoT based applications are the lack of a data integrity verification technique. In this paper, we propose a blockchain based technique that operates in the edge layer which can assure trust and very suitable environment for performance. The edge layer consists of more advanced devices than the IoT device layer. Blockchain is one of the promising decentralized and distributed technologies, which is paying attention to its security, transparency, and immutability. We can execute data integrity verification service in blockchain using decentralized manner, where transactions can be performed with no need for a trusted third-party auditor (TPA). We present the merkle tree structure, which allows efficient and secure verification of content in a large set of data. Merkle tree structure is used to verify the data integrity and analyze the performance of the system.

Our contributions to the proposed scheme are as follows:

- Blockchain based data integrity verification mechanism for IoT framework solves the problem of untrustworthy. In this framework, clients (data owners) and cloud servers that do not trust each other can interact.
- Merkle tree and bilinear mapping techniques are used for data integrity verification.
- The proposed data verification scheme provides better transparency regarding the history of transactions. All IoT devices have a copy of the same ledger, which is updated with every transaction. Modification of a single record in the data slice leads to further changes in the following data slices.
- Blockchain based data verification solution removes the need for intermediaries and other third parties for establishing trust.

Blockchain is one of the most promising technique and is attracted for its clearness, security, immutability, and decentralization. The technique is a combined multifield structure, it includes cryptosystem, and mathematics, relating peer-to-peer systems and using various algorithms to solve traditional record synchronize problems. Blockchain is an immutable digital ledger of transactions that distributes

and processes records of transactions across the entire network. Blockchain uses Distributed Ledger Technology (DLT) for each transaction using a cryptographic signature called a hash [11].

Merkle Tree (MT) is also called as hash tree, which encodes the data of blockchain in very secure and systematic way. Hash tree gives very quick verification response of data, as well as quick movement of data from one node to the other on P2P blockchain network. Each transaction of the blockchain consists hash value. So, hash values are stored in the form of a tree like structure not in sequential order in the block. In tree structure every hash value linked to its parent, like parent child tree like relation. All the transaction hashes in the block are also hashed result in a MT [12]. MT structure is popular for data integrity verification.

In MT each non-leaf node is assigned as the hash function of its children nodes and each leaf node is labeled as the hash value of a data block. The rootR is the final hash function shown in figure 1, which consists eight leaf nodes, MT = $(\mathrm{A}_i|A_i = \mathrm{h}(X_i, 1 \leq i \leq 15))$, where $\mathrm{h}(.)$ represents a hash function, e.g., Secure Hash Algorithm (SHA-1) etc. The value of the non-leaf node A_i is h $(A_l^i \,||\mathrm{A}_r^i)$, where A_l^i and A_r^i represent left child and right child respectively. Given a node A_i, the smallest ordered node set $\Omega_i = (A_i^1 >> \mathrm{A}_i^2 >> \cdots)$ that can be used by A_i to compute the root node root R is called Auxiliary Authentication Information (AAI). For example, in figure 1, the AAI of the node A_2 is $\Omega_2 = (\mathrm{A}_1 >> A_{10} >> A_{14})$.

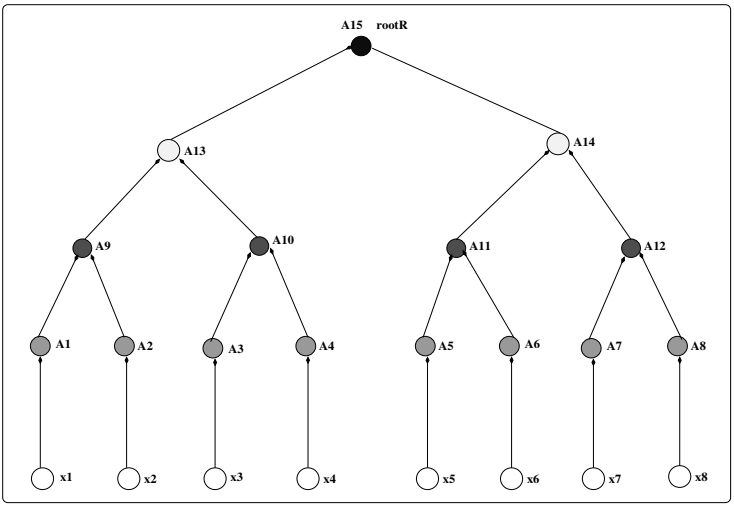

Fig. 1 Merkel Tree Structure

A bilinear mapping is a method of combining different characterized elements of two vector spaces to obtain an element of a third vector space. In cryptography, bilinear mapping is used to combine elements from two cryptographic groups to form a third group [13].

Rest of the research paper is organized as follows. The related work is discussed in section 2. The proposed blockchain based data integrity scheme is discussed in

section 3. The simulation model, simulation inputs, performance metrics, result analysis, and result discussion are presented in section 4. The conclusion and future work are presented in section 5.

2 Related Works

In this section, we present a comprehensive study of blockchain in IoT environment. Blockchain is a transparent, trusted, distributed, and decentralized ledger on a P2P network. Blockchain has a data unit called as transaction, and many correct transactions are bundled in a block [14]. A decentralized blockchain structure is constructed with all guaranteed blocks. In the distributed ledger, a block is connected to the previously agreed block using a cryptographic hash function of the block [15]. Blockchain assurances efficient and firm operations with the benefits of tamper conflicts and minimize single point of failure vulnerabilities. Blockchain transfers trust into the system via a consensus algorithm. The consensus algorithm is the process of coordinating the decentralized ledger through all the nodes in the blockchain network. The five layers framework of blockchain along with the integrity, immutability and security is presented in [16] [17].

Security and privacy challenges for IoT are addressed with blockchain mechanism is discussed in [18]. The mechanism exhibits a guarantee for IoT data. A blockchain approach for data integrity verification scheme in IoT is presented to overcome the traditional data integrity problems in [19]. Protection of data integrity using blockchain technique in IoT framework to address the security challenges are discussed in [20]. Privacy preservation schemes such as confidentiality, private contract, differential authentication, mixing and private contract are also discussed. Blockchain based data integrity verification model for decentralized edge cloud storage is described in [21].

Blockchain based certificate-less public verification scheme against procrastinating auditors is presented in [22]. The scheme has constant communication and computational overheads. The 5G Vehicular Ad-hoc Network Software-Defined Networking (VANET-SDN) enabled architecture along with the scheduling protocols of blockchain to ensure security and trustworthiness of vehicular IoT environment by preserving privacy is discussed in [23]. Blockchain based Distributed Key Management Architecture (BDKMA) with fog computing and multi blockchain mechanisms are discussed in [24].

The mobile agent approach for distributed virtual machines deployed in cloud computing is described in [25]. The virtual machine agent enables multi-tenant to ensure data trust verification. Blockchain technology is used in IoT networks to assure data integrity. The main aim of the network that provides a comprehensive, immutable log and allows easy access to the devices deployed in various domains is presented in [26]. Smart digital contracts of blockchain technology and its applications are discussed in [27].

A blockchain framework is designed to give data integrity in fish farming is discussed in [28]. The main goal of the scheme is to avoid data tampering and to provide secure storage. Data integrity verification in semi-trusted storage for resource

constrained Cyber-Physical Systems (CPS) in IoT is described in [29]. Data authentication system using a Merkle Tree (hash) concept to ensure data integrity is presented in [30]. The author discussed some analysis to enhance the security and reliability of the outsourced data. Merkle tree structure with a hash digital signature algorithm for data integrity verification scheme in cloud storage is discussed in [31].

In IoT networks, fundamental concepts like data sharing, computing, and providing security to the data are at the highest priority. The anomaly detection technique uses Mobile Data Collectors (MDCs) to identify the malicious activities before sending them to the Base Station (BS). The detection technique guarantees data integrity between Leader Nodes (LNs) and MDC is described in [32] [33].

A unique secure data aggregation scheme using the homomorphic encryption system in a wireless sensor network is presented in [34]. The scheme uses symmetric key homomorphic encryption and homomorphic signature to protect the privacy of information and check the aggregation data integrity respectively. A protected cloud computing protocol using a homomorphic algorithm for data integrity is mentioned in [35] [36].

A hybrid homomorphic encryption process and Rivest Shamir Adleman (RSA) algorithm to provide improved security and confidentiality of the information stored in the cloud servers is discussed in [37]. The blockchain based secure data aggregation mechanism for wireless sensor networks to guarantee data confidentiality and integrity is presented in [38].

A decentralized blockchain network is added to replace traditional centralized auditing in order to certify verification results. The cuckoo filter is used to reduce the computational burden of the user verification is proposed in [39].

Existing data integrity verification schemes are presented in table 1. Some of the drawbacks of current research works on blockchain based data integrity verification in IoT are as follows: limited utilization of blockchain to enhance cloud storage security, missing different privacy tasks in block chain technique, usage of common consensus algorithm, high computational delay and difficult communication methods. The rapid technology development in IoT needs intelligent techniques to provide security and maintain data integrity to achieve good performance. Hence, we proposed blockchain technology based data integrity verification scheme with Merkle tree structure in IoT network.

Table 1 Summarization of Existing Data Integrity Verification Mechanisms

Authors	Blockchain Type	Algorithms	Tools/Simulator	Contribution	Outcome	Remarks
Haiyan Wang, et. al [9]	Public Blockchain	Consensus Algorithm	Hyperledger Fabric, Pairing Cryptography	Blockchain and Bilinear mapping scheme for large-scale IoT	Minimized communication overhead and verification delay	Data recovery solutions in large scale IoT not addressed
Haiping Si, et. al [18]	Private Blockchain	Byzantine Fault Tolerant (BFT)	MATLAB	Lightweight IoT information sharing security framework	–	Effectively presented light weight IoT information sharing scheme.
Yu-Jia Chen, et. al [19]	Private Blockchain	Consensus Algorithm	MATLAB, Platform	A stochastic blockchain based data checking scheme	Minimized computational delay and communication overhead.	Effectively presented chain structure in stochastic blockchain technique
D. Yue, et. al [20]	Ethereum Blockchain	Proof of Work	Node.js, Solidity, Ethereum	Blockchain based data integrity verification scheme with cloud storage	Minimized latency and computational overhead	optimization of blockchain and edge cloud storage required
Yuan Zhang, et. al [22]	Public Blockchain	Proof of Work	C language and MIRACL	First certificate less public data integrity scheme is discussed.	Minimized communication overhead and computation overhead	Effective usage of blockchain technology to enhance cloud server in terms of security
Lixia Xie, et. al [23]	Public Blockchain	Proof of Work and Proof of stake	OMNeT++ 4.5, and crypto++ Cryptography	SDN enabled 5G VANET using blockchain scheme	Minimized communication overheads	Setting parameters for blockchain has not been discussed
Mingxin Ma, et. al [24]	Public Blockchain	Proof of Work	–	Distributed key management	–	Not explored feedback mechanism for SAMs and cloud managers.
Lei Hang, et. al [26]	Public Blockchain	Byzantine Fault Tolerant (BFT)	Hyper-ledger Fabric and Raspberry Pi	IoT platform using Blockchain technique	Minimum processing time	Effectively presented blockchain technique and consensus algorithms
Lei Hang, et. al [28]	Public Blockchain	Proof of Work	Hyper-ledger Fabric	Blockchain based fish platform	–	Consent usage not applied for different application domains

3 Proposed Work

This section present network architecture and proposed model of blockchain based data integrity verification mechanism for IoT.

3.1 Network Architecture

IoT network is facing many challenges like security, reliability and performance due to less computations interms of data management and power processing. The combination of the IoT with the edge nodes and cloud is the right way to solve security and privacy issues. The general IoT architecture shown in figure 2, which consists Physical layer, Edge layer, Network layer and Cloud layer (Cloud Service Provider).

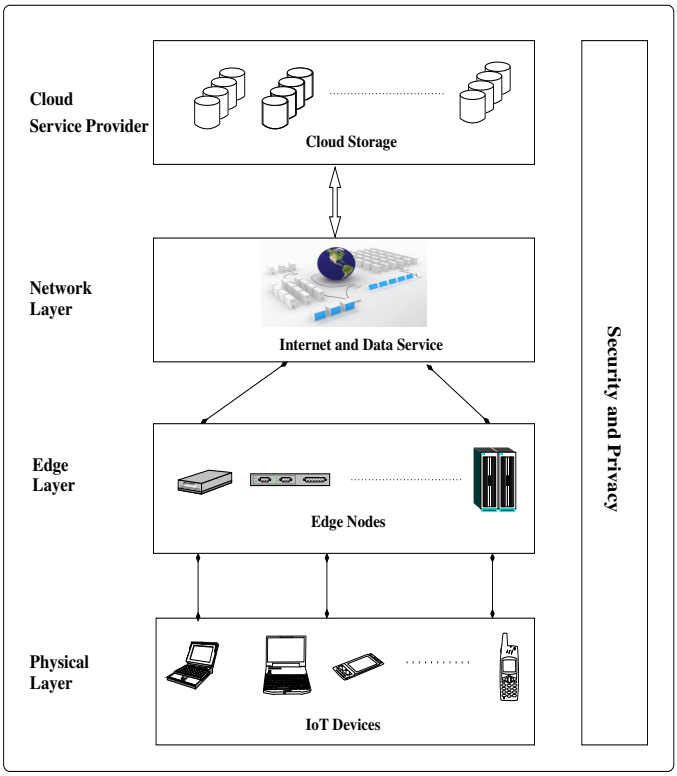

Fig. 2 IoT Architecture

The cloud service provider (CSP) is the top layer of the architecture, which is used to store the users information, transaction information and so on. The second layer is the network layer, which is used to transfer the information collected from the IoT devices to CSP. The edge layer consists of many base stations and embodies software modules and embedded operating systems. It is able to analyze the gathered information collected from physical layer and thus make decisions locally. A base

station has strong computing and storage capabilities, hence blockchain maintains consistency through entire network. Edge node communicate to the cloud through the Internet. The closeness of the edge nodes to the IoT devices helps in solving the latency difficulties and reduce the unnecessary communications between the cloud storage provider and IoT devices in real-time. In the edge layer, blockchain mechanism verifies the integrity of data sent by the IoT devices. The physical layer consists of different IoT devices like, cell phones, computers, household devices, etc. IoT devices are formed into clusters and every device is connected to one of the edge nodes i.e. base station.

3.2 Preliminaries

Some of the preliminaries are used in our proposed mechanism are as follows:

3.2.1 Bilinear Mapping Technique

This technique is represented as $e : G \times G \rightarrow G_T$, where G is a Gap Diffie-Hellman (GDH) group and G_T is multiplicative cyclic group of prime order p with the properties given below [37]:

(i) Computability: An efficiently computable algorithm for computing e and e is a non-degenerate bilinear mapping $e : G \times G \rightarrow G_T$, where $|G| = |G_T| = p$

(ii) Bilinear: For all $h_1, h_2 \in G$ and $a, b \in Z_p, e(h_1^a, h_2^b) = e(h_1, h_2)^{ab}$

(iii) Non degenerate: $e(g, g) \neq 1$, where g is a generator of G

3.2.2 KeyGen

KeyGen is represented as$(1^k) \longrightarrow (\mathrm{Pb}_k, Pr_k)$, it defined by probabilistic key generation algorithm. 1^K is the input security parameter selected by client. Outputs are Pb_k and Pr_k. Client publishes the public key Pb_k and keeps private key Pr_k.

3.2.3 SignGen

SignGen is represented as $(\mathrm{Pr}_k, D) \longrightarrow (\delta, Sign_{Pr_k}(H(R)))$. The algorithm runs at the client side by taking private key Pr_k and file D as input. The files are slice as blocks $\{d_i\}$ and the signature set δ, which is ordered collection of signature $\{S_i\}$ on $\{d_i\}$. It is also output metadata signature $Sign_{Pr_k}(H(R))$ of the rootR of merkle hash tree. In our proposed scheme the leaf nodes of the Merkle tree are hashes of $H(d_i)$.

3.2.4 GenProof

GenProof is represented as $(D, \delta, chlng) \longrightarrow (P)$. The cloud storage server compute the GenProof. It includes input file D, its signature δ and a challenge $chlng$. It produces a data integrity proof P for the blocks specified by $chlng$.

3.2.5 CheckingProof

CheckingProof is represented as $(Pb_k, chlng, P) \longrightarrow \{Accept, Reject\}$. The CheckingProof algorithm operate under the blockchain. It include input public key

$Pb_k, chlng$ and proof P returned from the cloud storage server. The file is accepted if the integrity of the file F is verified as correct or reject otherwise.

Table 2 shows list of the notations that we considered in proposed scheme.

Table 2 Notations in the Framework

Notations	Description
Pb_k	Public Key
Pr_k	Private Key
D	Data File
$\{d_i\}$	i^{th} file block
A_l^i	A^i left child
A_r^i	A^i right child
x_i	Tag of the i^{th} file block
δ	Set of Signature
$\{S_i\}$	Collection of Signature
h	Hash Function
rootR	Root hash of Merkle Tree
chlng	Challenge
$\{\Omega_i\}$	The Integrity Path
A	File tag for D

3.3 Proposed Scheme

The proposed blockchain based framework for data integrity verification is as shown in the figure 3. It consists of different entities such as client, Key Generation Centre (KGC), cloud storage server, and blockchain. Proposed data integrity verification operates in three stages: Setup Stage, Processing Stage and Verification Stage.

3.3.1 Setup Stage

This stage consists IoT devices (client) and Key Generation Center (KGC). In KGC, the k is the input security parameter selected by the client. Two outputs are the private key (Pr_k) and public key (Pb_k). Every IoT device has a (Pr_k), which is used to create tags of data files $H(d_i)$, and (Pb_k) is used to check the integrity of stored data files. The IoT devices generates private key and public key as given in equation (1) and (2) respectively.

$$Pr_k = (\alpha, SPr_k) \tag{1}$$

$$Pb_k = (u, SPb_k) \tag{2}$$

where, $\alpha \rightarrow G$ random value and computed $u \leftarrow g^{\alpha}$

Assume that a device consists big data file to be stored in the cloud storage server for data preservation and computation. If a client takes data file D (as shown in equation 3) and splitted into data shared of same length $\{d_1, d_2, d_3, \cdots d_n\}$

$$D = (d_i) \tag{3}$$

Where (i = 1, 2, $\cdots n$)

After slicing data as d_i, client generates digital signature with private key using EiGmal algorithm. This algorithm gives high security due to probabilistic in nature. The signature is hash function type and is also called as digest of the data to be encrypted using a private key. The digest data is added with original data as digital signature of the transmitted data. The digital signature (S_i) for each data block d_i, $(i = 1, 2, \cdots n)$ is given in equation (4).

$$S_i = (H(d_i) \cdot v^{d_i})^{\alpha} \tag{4}$$

Where $H(d_i)$ is a tag file for data block d_i, random element $v \leftarrow G$, signature S_i denotes the signature set $\delta = S_i, (1 \leq i \leq n)$ for different data blocks in the network.

Client obtain rootR based on Merkle Tree (MT). In MT, leaf nodes generates the set of hash values of data file tags $H(d_i)$ where $(i = 1, 2, \cdots n)$. Then client signs the rootR using private key α and is given as shown in equation 5.

$$\alpha = Sign_{Pr_k}(H(R)) \tag{5}$$

Where $Sign_{Pr_k}(H(R))$ is the digital signature created by the private key on the MT root node.

Client construct an uploading data file $\{D, A, t_s, \delta, Sign_{Pr_k}(H(R)\}$ to smart contract and then forward to the server. Time stamp is represented as t_s.

3.3.2 Processing Stage

This stage operates under blockchain technology, which verifies the data integrity and gives challenge to the cloud service provider. Before challenging, blockchain will first check the signature on A using public key. If the checking fails it is rejected by releasing as $false$, otherwise recovers as v. Let $A = name \| n \| v \| Sign_{Pr_k}$, where $(name \| n \| v)$ is the file tag for D.

The blockchain verifier gives the challenge *chlng* to the cloud server i.e., prover by selecting random c element subset as shown in equation 6.

$$J = \{s_1, \cdots, s_c\} \; offset \; [1, n] \tag{6}$$

Where for $s_1 \leq \cdots \leq s_c$, $i \in J$, the blockchain select a random element $u_i \leftarrow Z_p$.

The message *chlng* specifies the position of the blocks to be checked in this processing stage. The blockchain send the $chlng\{(i, v_i)\}$ $s_1 \leq i \leq s_c$ to the server. Upon receiving the challenging message *chlng* from the verifier, the prover computes and generates the proofs as given in euations (7) and (8). In addition, prover gives auxiliary information $\{\Omega_i\}$ $s_1 \leq i \leq s_c$, which are the node siblings on the path from the leaves to the rootR of the MT.

$$\mu = \sum_{i=s_1}^{s_c} u_i d_i \in Z_p \tag{7}$$

and

$$\sigma = \prod_{i=s_1}^{s_c} S_i^{u}{}_i \in G \tag{8}$$

The prover responds to the verifier by generating $proofP$ as given in equation (9).

$$proofP = \{\mu, \sigma, \{H(d_i), \Omega_i\}\ s_1 \leq i \leq s_c, Sign_{Pr_k}(H(R))\} \tag{9}$$

3.3.3 Verification Stage

After accepting the response from the prover, the verifier produce rootR using $H(m_i), \Omega_i\}\ s_1 \leq i \leq s_c$ and verifies it by checking the equation (10). If the verification fails, the verifier rejects. Otherwise the verfier verifies again as shown in the equation (11) if the output accepts, otherwise rejects.

$$e(Sign_{Pr_k}(H(R)), g) = e(H(R), g^{\alpha}) \tag{10}$$

$$e(\sigma, g) = e(\prod_{i=s_1}^{s_c} H(d_i)_i^{v}.v^{\mu}, v) \tag{11}$$

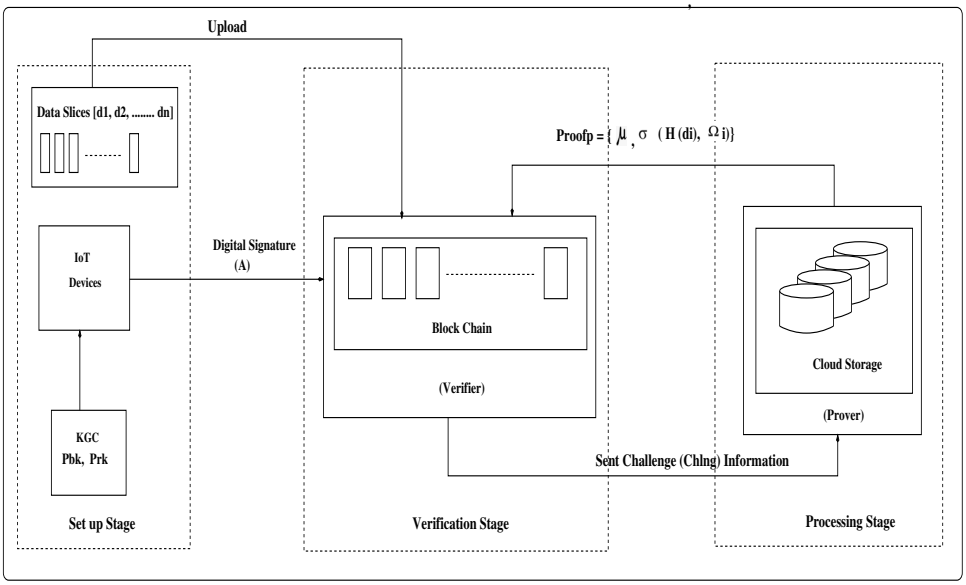

Fig. 3 Proposed Blockchain based Framework for Data Integrity Verification Scheme

3.4 Algorithms

The proposed data integrity verification scheme using blockchain scheme in IoT is mentioned in algorithms (1) to (4).

Algorithm 1 Data Integrity Verification using Blockchain Algorithm

1: *Input : File* (D)*, Generator* (g)*, Hash Function* (h)*, Public Key* (Pb_k)*, Private Key* (Pr_k)
2: *Output : File is accepted if integrity is verified otherwise rejected*
3: **Begin**
4: Step:1 File (D) shards into n number of blocks (Algorithm-2)
5: Step:2 Signature generation $SignGen(Pr_k, D)$ (Algorithm-3)
6: Step:3 Verification of data integrity $Verify()$ to check whether accept or reject (Algorithm-4)
7: **End**

Algorithm 2 File (D): Division of File Shards into n Number of Blocks Algorithm

1: **Begin**
2: Step:1 Obtaining n blocks from File (D)
3: **for** i←0 *to* n **do** $F \leftarrow \mathrm{d}_1, d_2, d_3, \cdots d_n$
4: **end for**
5: Step:2 Use generator and hash function for each n blocks
6: **End**

Algorithm 3 $SignGen(Pr_k, D)$: Signature Generation Algorithm

1: **Begin**
2: Step:1 (Setup Phase) Obtaining Pb_k, Pr_k from probabilistic algorithms using signing key pair (SPr_k, SPb_k) generated from client $Pr_k = (\alpha, SPr_k)$ $Pb_k = (u, SPb_k)$
3: Step:2 (Sign Generation Phase) Signature (δ) generation using D and Pr_k , For each block n generate Signature Set (δ) and obtain collection of Signature σ_i
4: **for** i←0 *to* n **do** $\delta = S_i$
5: **end for**
6: Step:3 Client obtain rootR based on MT, where leave nodes are ordered set of Hashes $H(m_i)$ $\alpha : Sign_{Pr_k}(H(R)) \leftarrow (H(R))^{\alpha}$
7: Step:4 Store the above data in the verifier as $D, A, \delta, Sign_{Pr_k}(H(R))$
8: **End**

4 Simulation

The proposed scheme has been simulated using python programming language. In this section, we present simulation inputs, performance parameters and result analysis of the proposed work.

4.1 Simulation Inputs

To illustrate the results of the proposed scheme, the simulation input parameters are summarized in table 3.

Algorithm 4 $Verify()$: Data Integrity Verification Algorithm

Begin
Step:1 (Challenge Phase) checking the data file using Pb_k
if Pb_k is not verified **then** file is rejected
else
    for i$\leftarrow 0$ *to* $\{s_1, \cdots, s_c\}$ **do** $chlng_i \leftarrow \{(i, u_i)\}$
    end for
end if
Step:2 (Proof Generation Phase) Server challenges the proof of verification
for i$\leftarrow 0$ *to* chlng **do** $\mu = \sum_{i=s_1}^{s_c} u_i d_i$ and $\sigma = \prod_{i=s_1}^{s_c} S_i^u{}_i$
end for
for $i \leftarrow 0$ *to* $\{s_1, \cdots, s_c\}$ **do** Obtain auxilary information $\{\Omega_i\}$
end for
Step:3 Server responds the verifier with Proof(P)
$P = \{\mu, \sigma, H(d_i), \Omega_i, Sign_{Pr_k}(H(R))\}$
Step:4 Verification of data integrity by blockchain
if e(SignPr$_k(H(R))$,g) = e($H(R)$, g^{α}) **then** file is accepted
else file is rejected
end if
End

Table 3 Simulation Inputs

Sl. No.	Input parameters	Specifications
01	No. of IoT Devices	10-50
02	Communication Range	upto 50 m
03	Data Size	100 Bytes-1024 Bytes
04	Memory	up to 8 GB (E flash memory)
05	Signature Set	128 bits
06	Average Tranasaction Length	256 Bytes
07	Blockchain Network	Hyperledger Fabric

4.2 Performance Parameters

To test the performance evaluation of the proposed scheme, some of the performance parameters evaluated are as follows.

- **Signature Generation Delay**: It is the time required to generate a digital signature using a private key to provide data security in IoT network. It is represented in terms of milliseconds..
- **Markle Tree Generation Delay**: The time taken by the proposed scheme to generate markle tree (hash tree), it is used for data verification and synchronization in blockchain. It is expressed in terms of milliseconds (ms).
- **Memory Utilization**: Memory utilization refers to the amount memory consumed by the IoT devices during the processing and verification stage derived from available memory in use at a given time. It is represented in terms of bytes.

- **Control Overheads**: The control overhead is extra information put on top of the payload of the data packet for guaranteed delivery. It is defined as the ratio of the total number of control messages to the total number of data packets generated to perform data integrity in the network. Expressed in terms of bytes.
- **Accuracy Rate**: Data accuracy refers to overall error free, completeness, and consistency of data achieved by the proposed scheme and it is expressed in terms of percentage.
- **End to End Delay**: The end-to-end delay is a sum of all delays experienced by the packet on the way to the destination with the highest level of integrity. It is articulated in terms of milliseconds (ms).

4.3 Result Analysis

To test the operative effectiveness of the proposed scheme to achieve the highest level of integrity for the data in IoT environment using the blockchain mechanism, we have used 50 data blocks and analyzed some of the performance parameters which are mentioned in the underneath graphs. If we consider more than 50 data blocks, it will take more attempts to recover key and complexity of the system will increase, hence signature generation delay, end to end delay will be comparatively more. Further increase in block size will consume more memory of the IoT devices but the data will be more secure. Simulation results show that the proposed algorithm is efficient as compared with the existing blockchain based cloud data integrity verification scheme with high efficiency [39]. Author proposed Cuckoo Filter based Data Integrity scheme (DICF scheme) to provide security in the IoT network.

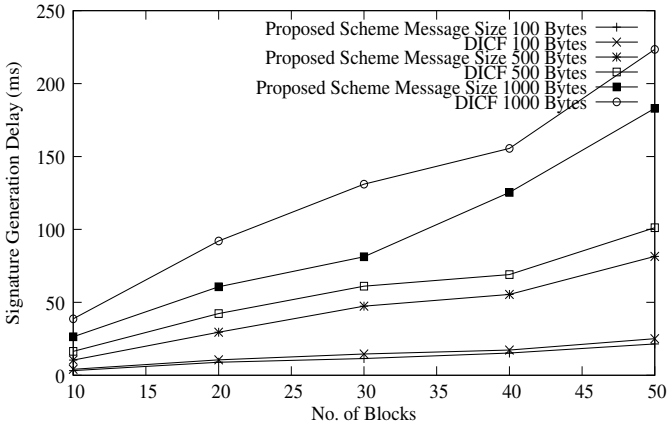

Fig. 4 No. of Blocks Vs. Signature Generation Delay

Figure 4 shows number of blocks vs. signature generation delay for different data size. As the number of blocks increases the signature generation delay also increases. DICF protocol exhibits more delay for the generation of keys than the

proposed scheme. Since DICF scheme has many exponential operations like multiplication operation, hash operation, and repeated calculations with high computational overhead are involved in the signature generation.

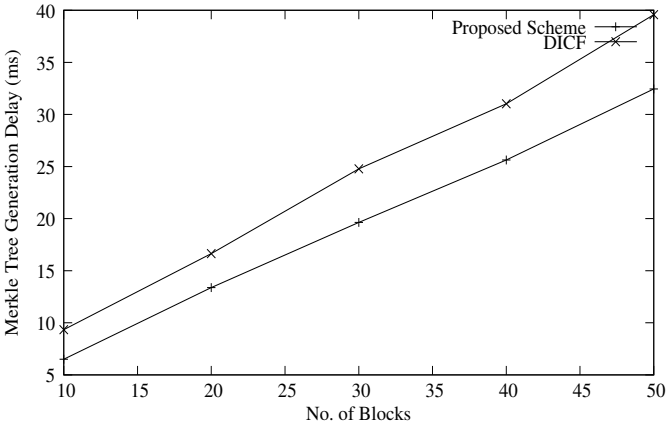

Fig. 5 No. of Blocks Vs. Markle Tree Generation Delay

In figure 5, as number of blocks increases the time taken by the system to generate markle tree also increases, the proposed scheme consumes less time for markle tree generation as compared with DICF protocol. In our proposed scheme blocks are further splited into shards to provide more security to the data. Figure 6 presents that as the number of shards increases the time taken to generate markle tree also increases. The proposed scheme has less dealy for markle tree generation as compared with DICF. DICF scheme consists of cuckoo filter along with blockchain technique, hence DICF scheme exhibits more delay than the proposed scheme.

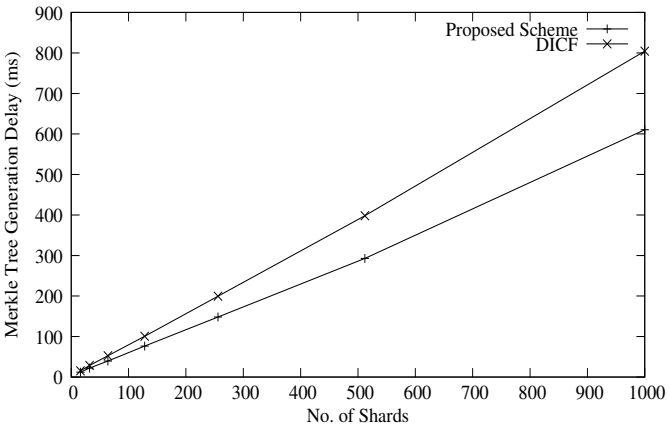

Fig. 6 No. of Shards Vs. Markle Tree Generation Delay

Figure 7 shows the performance of the proposed scheme is better in terms of end-to-end delay w.r.t. DICF scheme, it is observed that the as data size and number of blocks increases the dealy also increases. The proposed scheme has less E2E delay than DICF. Our scheme involves in only bilinear mapping technique, the amount of data to be calculated is small, hence performs better.

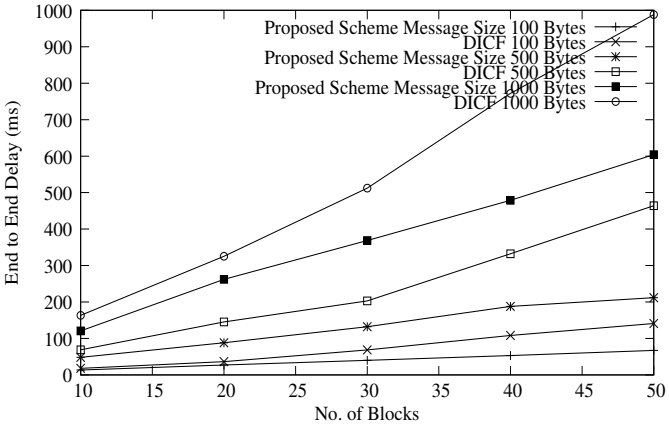

Fig. 7 No. of Blocks Vs. End to End Delay

Figure 8 outlines the comparison of memory utilization for the proposed scheme and DICF scheme. As number of blocks increases with different data sizes the memory consumption also increases this is due to need of more memory to process and verify the data to achieve maximum integrity.

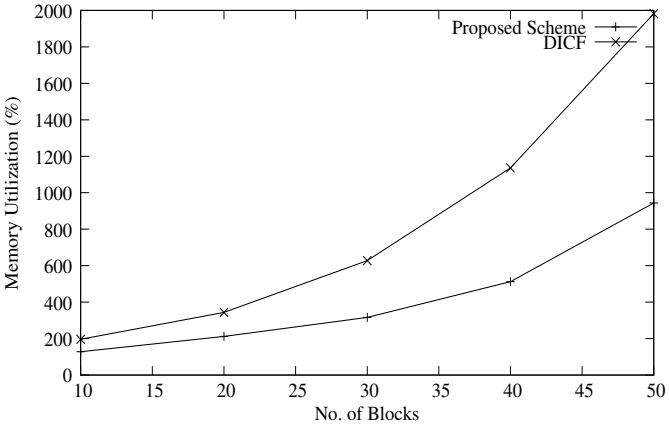

Fig. 8 No. of Blocks Vs. Memory Utilization

From figure 9, we observe that control overhead increases w.r.t. increase in number of blocks and shards. It is observed that DICF scheme is slightly inferior in terms of control overhead packets.

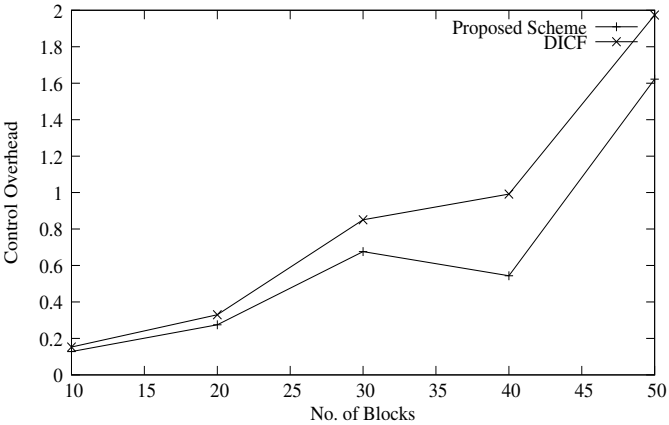

Fig. 9 No. of Blocks Vs. Control Overhead

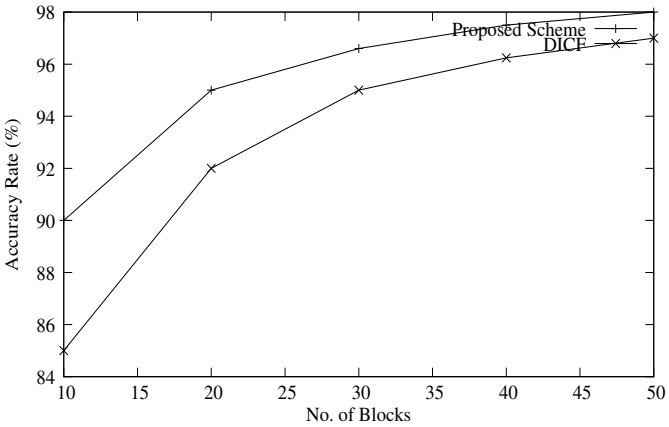

Fig. 10 No. of Blocks Vs. Accuracy Rate

Figure 10 exhibits the overall accuracy of the proposed scheme with considering different data sizes and block rates. The proposed blockchain-based data integrity framework for IoT is better w.r.t. DICF. Since the DICF scheme uses exponential operations with high computational overhead.

The obtained results show that in IoT network data integrity can be achieved considering parameters key size, data sizes, block rates and number of shards. Fig. 4 - 10 shows that the proposed scheme is better compared with DICF protocol. The

proposed algorithm progressively performs at a good rate and the network stability can be maintained by providing data security.

5 Conclusion

In this paper, we proposed blockchain technique for providing data integrity in IoT networks. We have designed a blockchain based data integrity framework considering bilinear mapping technique. The blockchain techniques are efficient in providing data security in ubiquitous IoT applications. The proposed scheme operates in three stages are i) Setup stage ii) Processing stage and iii) Verification stage. In the setup stage, slicing of the data into shards is done, and a digital signature key is generated for each shard to provide security to data. In the processing stage, data shards are verified and combined at different levels in blockchain and sent to the cloud service provider. Then cloud server (prover) gives auxiliary information and responds to the verifier with proof. In the verification stage, the verifier receives the response from the prover and verifies the data integrity. If the data is unaltered then it is accepted at the receiver end otherwise it is rejected. Simulation results demonstrate that the proposed scheme is better in terms of signature generation time, end to end delay, memory utilization as compared with DICF scheme. Further, the research work can be extended to a higher level by considering multiple heterogeneous IoT networks for making secure communication in remote areas.

6 Compliance with Ethical Standards

Conflict of Interest: Author Mrs. Poornima M. Chanal declares that she has no conflict of interest. Author Dr. Mahabaleshwar S. Kakkasageri declares that he has no conflict of interest.

Ethical Approval: This article does not contain any studies with human participants or animals performed by any of the authors.

Funding Details: Not Applicable.

Authors Contribution Section: Both authors equally contributed.

References

[1] Poornima M. Chanal, Mahabaleshwar S. Kakkasageri, “Security and Privacy in IoT: A Survey”, *Journal of Wireless Personal Communication*, Springer, Vol. 115, No. 3, pp. 1667–1693, 2020

[2] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions” *Journal of Future Generation Computer Systems*, Elsevier, Vol. 29, No. 7, pp. 1645–1660, 2013

[3] D. Miorandi, S. Sicari, F. De Pellegrini, I. Chlamtac, “Internet of Things: Vision, Applications and Research Challenges”, *Journal of Ad Hoc Networks*, Elsevier, Vol. 10, No. 7, pp. 1497– 1516, 2012

[4] D. Yang, F. Liu, Y. Liang, "A Survey of Internet of Things", *Proc. of the International Conference on E-Business Intelligence (ICEBI'10)*, Vol. 978, pp. 78–99, China, 2010

[5] "Vision and Challenges for Realizing the Internet of Things", Cluster of European Research Projects on the Internet of Things, European Commission Information Society and Media, 2010

[6] H. Suo, J. Wan, C. Zou, J. Liu, "Security in the Internet of Things: A Review", *Proc. of the IEEE International Conference on Computer Science and Electronics Engineering*, Vol. 3. pp. 648–651, China, 2012

[7] Poornima M. Chanal, Mahabaleshwar S. Kakkasageri, Sunilkumar S. Manvi, "Security and Privacy in Internet of Things: Computational Intelligent Techniques based Approaches", Contributed chapter in book titled, *Recent Trends in Computational Intelligence Enabled Research*, Elsevier pp. 111–127, 2021

[8] G. Yang, J. Xu, W. Chen, Z. H. Qi, H. Y. Wang, "Security Characteristics and Technology in the Internet of Things", *Journal of Nanjing University of Posts and Telecommunications*, Vol. 30, No. 4, 2010

[9] "Haiyan Wang, Jiawei Zhang, "Blockchain Based Data Integrity Verification for Large Scale IoT Data", *Journal of IEEE Access*, Vol. 7, pp. 164996–165006, 2019

[10] Xu Wang, Xuan Zha, Wei Ni, Ren Ping Liu, Y. Jay Guo, Xinxin Niu, Kangfeng Zheng, "Survey on Blockchain for Internet of Things", *Journal of Computer Communications*, Elsevier, Vol. 136, pp. 10–29, 2019

[11] Md. Ashraf Uddin, Andrew Stranieri, Iqbal Gondal, Venki Balasubramanian, "A Survey on the Adoption of Blockchain in IoT: Challenges and Solutions", *Journal of Blockchain: Research and Applications*, doi.org/10.1016/j.bcra.2021.100006, 2021

[12] Jian Mao, Yan Zhang, Pei Li, Teng Li, Qianhong Wu, Jianwei Liu, "A Position Aware Merkle Tree for Dynamic Cloud Data Integrity Verification", *Journal of Soft Computing*, Vol. 21, pp. 2151–2164, 2017

[13] Furukawa J, Imai H, "An Efficient Group Signature Scheme from Bilinear Maps", *Proc. of the International Conference on Information Security and Privacy*, Berlin, 2005

[14] Junqin Huang, Linghe Kong, Guihai Chen, Min-You Wu, Xue Liu, Peng Zeng, "Towards Secure industrial IoT: Blockchain System with Credit based Consensus Mechanism", *IEEE Transactions on Industrial Informatics*, Vol. 15, No. 6, pp. 3680–3689, 2019

[15] Yong Yu, Yannan Li, Junfeng Tian, Jianwei Liu, "Blockchain-Based Solutions to Security and Privacy Issues in the Internet of Things", *Jouranal of IEEE Wireless Communications*, pp. 12–18, 2018

[16] M. Ul Hassan, M.H. Rehmani, J. Chen, "Privacy Preservation in Blockchain based IoT Systems: Integration Issues, Prospects, Challenges, and Future Research Directions", *Journal of Future Generation Computer Systems*, Vol. 97, pp. 512–529, 2019

[17] Javelino F Zorzo, Henry C Nunes, Roben C Lunardi, Regio A Michelin, Salil S Kanhere, "Dependable IoT using Blockchain based Technology", *In 2018 Eighth Latin-American Symposium on Dependable Computing (LADC)*, pp, 1–9, 2018

[18] Haiping Si, Changxia Sun, Yanling Li, Hongbo Qiao, Lei Shi, "IoT Information Sharing Security Mechanism based on Blockchain Technology", *Journal of Future Generation Computer Systems*, Vol. 101, pp. 1028–1040, 2019

[19] Yu-Jia Chen, Li-Chun Wang, Shu Wang, "Stochastic Blockchain for IoT Data Integrity", *Journal of IEEE Transactions on Network Science and Engineering*, Vol. 7, No. 1, 373–384, 2020

[20] D. Yue, R. Li, Y. Zhang, W. Tian, Y Huang, "Blockchain based Verification Framework for Data Integrity in Edge Cloud Storage", *Journal of Parallel and Distributed Computing*, Vol. 146 pp. 1–14, 2020

[21] Peng Ceng Wei, Dahu Wang, Yu Zhao, Sumarga Kumar Sah Tyagi, Neeraj Kumar, "Blockchain Data-based Cloud Data Integrity Protection Mechanism", *Journal of Future Generation Computer Systems*, Vol. 102, pp. 902–911, 2020

[22] Yuan Zhang, Chunxiang Xu, Xiaodong Lin, Xuemin, "Blockchain-Based Public Integrity Verification for Cloud Storage against Procrastinating Auditors", 2018

[23] Lixia Xie, Ying Ding, Hongyu Yang, Xinmu Wang, "Blockchain-Based Secure and Trustworthy Internet of Things in SDN-Enabled 5G-VANETs", *Journal of IEEE Access*, Vol. 7, pp. 56656–56666, 2019

[24] Mingxin Ma, Guozhen Shi, Fenghua Li, "Privacy-Oriented Blockchain-Based Distributed Key Management Architecture for Hierarchical Access Control in the IoT Scenario", *Journal of IEEE Access*, Vol. 7, pp. 34046–34059, 2019

[25] H. F. Atlam, R. J. Walters, G. B. Wills, A. Alenezi, A. Alharthi, "Integration of Cloud Computing with Internet of Things: Challenges and Open Issues", *Proc. of The IEEE International Conference on Internet of Things and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pp. 670–675 Exeter, UK, 2017

[26] Lei Hang, Do Hyuen Kim, "Design and Implementation of an Integrated IoT Blockchain Platform for Sensing Data Integrity", *Journal of Sensor*, Vol. 19, No. 10, 2019

[27] Athina Styliani Kleinaki, Petros Mytis-Gkometh, George Drosatos, "A Blockchain-Based Notarization Service for Biomedical Knowledge Retrieval", *Journal of Computational and Structural Biotechnology*, Vol. 16, pp. 288–297, 2018

[28] Lei Hang, Israr Ullah, DoHyeun Kim, "A Secure Fish Farm Platform based on Blockchain for Agriculture Data Integrity", *Journal of Computers and Electronics in Agriculture* Vol. 170, 2020

[29] Caciano Machado, Antonio Augusto Frohlich, "IoT Data Integrity Verification for Cyber-Physical Systems Using Blockchain", *Proc. of the International Symposium on Object Oriented Real Time Distributed Computing*, Singapore, 2018

[30] Dongyoung Koo, Youngjoo Shin, Joobeom Yun, Junbeom Hur, "Improving Security and Reliability in Merkle Tree-Based Online Data Authentication with Leakage Resilience", *Journal of Applied Science*, Vol. 8, pp. 1–29, 2018

[31] A. R. Arunachalam, G. Michael, "Merkle Hash Tree with Hash based Digital Signature for Cloud data Confidentiality and Security", *Journal of Pure and Applied Mathematics*, Vol. 119 No. 12, pp. 12233–12242, 2018

[32] Quazi Mamun, Rafiqul Islam, Mohammed Kaosar, "Ensuring Data Integrity by Anomaly Node Detection during Data Gathering in WSNs", *Journal of Security Communication*, Vol. 127, pp. 367–379, 2013

[33] May Altulyan, Lina Yao, Salil. S .Kanhere, Xianzhi Wang, Chaoran Huang, "A Unified Framework for Data Integrity Protection in People-centric Smart Cities", *Journal of Multimedia Tools and Applications*, Vol. 79, pp. 4989–5002, 2020

[34] Poornima M. Chanal, Mahabaleshwar S. Kakkasageri, "Preserving Data Confidentiality in Internet of Things", *Journal of SN Computer Science*, Springer, Vol. 02, No. 01, pp. 1-12, 2021.

[35] Soufiene Ben Othman, Abdullah Ali Bahattab, Abdelbasset Trad, Habib Youssef, "Confidentiality and Integrity for Data Aggregation in WSN Using Homomorphic Encryption", *Journal of Wireless Personal Communication*, Vol. 80, pp. 867–889, 2015

[36] Prakhar Awasthi, Sanya Mittal, Sibeli Mukherjee, Trupil Limbasiya, "A Protected Cloud Computation Algorithm Using Homomorphic Encryption for Preserving Data Integrity", *Proc. of the* 5^{th} *International Conference on Advanced Computing, Networking and Informatics*, pp. 509–517, Goa, 2018

[37] D. Chandravathi, P. V. Lakshmi, "A New Hybrid Homomorphic Encryption Scheme for Cloud Data Security", *Journal of Advances in Computational Sciences and Technology*, Vol. 10, No. 5, pp. 825–837, 2017

[38] Yun Liu, Chaoran Li, Jing Zhang, Qing Liu, "A Homomorphic MAC-based Secure Data Aggregation Scheme for Wireless Sensor Networks", *Journal of Internet Technology* Vol. 19, No.7, pp. 2069–2077,2018

[39] Gaopeng Xie, Yuling Liu, Guojiang Xin, Qiuwei Yang, "Blockchain-Based Cloud Data Integrity Verification Scheme with High Efficiency", *Journal of Security and Communication Networks*, Hindawi, pp. 01–15, 2021