International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-9 Issue-1, May 2020

Hyperledger Fabric Blockchain for data Security in IOT Devices

Kathayayani N, Jayanthi K Murthy, Vaibhav Nityanand Naik

Abstract*: Data security for IOT devices is very import aspect these days as the world is moving towards digitalization. Consider a smart energy meter which provides a way to monitor the energy consumption at home, data security in such smart meter reading is very important. If the Power reading signals are tampered, then it may cause serious economic loss for the authorities. The personal information infringement of user can occur at the database and may fall in the hands of unethical persons. In order to address these issues in this paper we propose to use a permissioned blockchain network. Blockchain maintains time stamped ledger records that are very hard to tamper. Every transaction is recorded and distributed across many participant nodes, these records are immutable because they have blocks of data which are linked to each other with strong cryptographic hash. The blockchain network is built using hyperledger fabric, where all the participant nodes are registered and only registered nodes involve in consensus process of transaction. In fabric, MSP (membership service provider) identifies the identity of the participant nodes through X.509 digital certificates issued by certificate authority. Along with creation of blockchain network for the application, a mobile client, a web client, an Arduino client and web server is created. The Arduino client is the hardware module that has an energy meter (SDM120) measuring the energy consumption of the user and sends this information serially to NODEMCU. NODEMCU POSTs the read energy details to the web server at particular api, web server POSTs the details to the Blockchain Network, where transactions undergoes consensus to add this information to blockchain ledger. Now data is decentralized and every peer node has the local copy of ledger. The updated information can be queried and seen on the web Client and Mobile client user interfaces. Anonymity-enhanced blockchain has been implemented to avoid the disclosure of personal information or data. Also performance analysis of the application is carried out for number of sequential requests and concurrent requests from many users using different tools.

***Keywords*: Blockchain, distributed system, hyperledger fabric, Internet of Things (IOT).**

I. INTRODUCTION

A strong security is essential to maintain the collected data safely between IOT devices. There are many challenges in implementing data Security for Internet of Things (IoT) devices. The ability of an unauthorized user to access the system needs to be blocked for attacks such as denial of service, and only the authorized users should be allowed to access the data in a secure system without any delay. It is very essential for the communication to be private to make sure that data cannot be changed or viewed during the movement. In an IOT application such as smart energy meter, one should concentrate to avoid any attack due to impersonation leading to serious economic loss. This paper focuses on finding a solution for data security, personal information infringement and tampering of data at the service provider, after receiving data from any IOT device like smart energy meter. Blockchain is found as one of emerging technology to address these issues. The data can be distributed across the systems and the security of these distributed data can be achieved with blockchain. There are lots of transactions that are happening within the system, all these transactions are recorded in blockchain, which are specific & verifiable records. Blockchain contains data in the form of ledger records, which are very hard to tamper. Blocks have transaction information and the blocks are linked to each other cryptographically with strong hash encryptions. Every block has details of the prior block transactions, as the current block includes the hash generated by the previous block transactions, creating a 'chain' of blocks and hence called blockchain. If a hacker tries to modify one block, then he has to modify the prior block and repeat this process to modify the chain entirely, hence blockchain technology is found to be immune to modification. As the blockchain technology is distributed, when data stored in one of the system is crashed, the ledger contents are available in the rest of participating nodes. Thus tampering of data and data loss can be avoided. Permissioned blockchains build a chain that is surrounded by all recognized, identified sources which can be trusted. There are instances where the participants have a similar core, but may not trust each other completely. In such cases permissioned blockchain helps in securing the instructions among participants. There are some consensus protocols in a permissioned blockchain such as Hyperledger fabric framework that does not need the process of mining. These consensus protocol can be either crash fault tolerant (CFT) or byzantine fault tolerant (BFT). Using this type of traditional blockchain avoids any intentional introduction of malicious codes by a participant through smart contract. In such network all actions like submitting a transaction from an application to update ledger, any modifications made to network configuration or deployment of smart contract on peers are recorded. The main objective of the paper is to design an IoT system that is capable of reading and sharing the read energy meter data to application and then uploading it to a blockchain server.

Manuscript received on May 18, 2020.
Revised Manuscript received on May 27, 2020.
Manuscript published on May 30, 2020.

* Correspondence Author

Kathayayani N*, Student, Master of Technology, Department of Electronics and Communication, B.M.S college of Engineering, Bengaluru, India.

Jayanthi K Murthy, Associate Professor, Department of Electronics and Communication, B.M.S college of Engineering, Bengaluru, India.

Vaibhav Nityanand Naik, Software Engineer, Computer Software, Banngalore, India.

This is done using a Permissioned blockchain Network. A Smart contract which defines set of rules, is utilized to add the data to the Blockchain, thus increasing the reliability of users.

II. RELATED WORK

[1]According to authors the security features should be implemented from the design stage of smart metering application in order to avoid security compromise. The authors have discussed different attacker model like Eavesdroppers, marketing agencies and journalists, customers, novice attackers and active attackers. In [2] the authors have presented threat taxonomy considering: (a) System-level security threats, (b) Services threats and/or theft, and (c) Privacy threats. A group of security and privacy necessities for smart grid metering systems are derived based on the presented threats. Addressing issues of data tempering, man-in-the-middle attack, and blockchain based data records for advanced metering infrastructure is done in [3] using MICAz motes for communication between smart meters. To resist the attacks from semi-honest users and malicious users and to preserve privacy in social networks the authors in [4] propose the use of block chain technology to protect the identity of users and the security of their corresponding keys. In [5] authors have established a secure connection between two IOT devices using ethereum blockchain platform, to show that blockchains can be used for IoT security. They have conducted two experiments to have communication between IoT devices with and without blockchain. The authors of paper [6] concentrate on issues faced by the pub/sub model used for IOT. They try to address issues such as failure of server which is centralized causing a single point failure, data tampering by unethical broker. The authors have developed a pub/sub architecture by using blockchain which preserves confidentiality of private data. In [7] authors have proposed use of Blockchain technology for a distributed method to provide safety in maintaining the medical records of the patient. They have used three steps to provide security one is authentication, the second Encryption and lastly getting access to data using Blockchain. In [8] authors introduce an IoT server platform using blockchain. Authors try to address the vulnerabilities and threats to security in Mysql's Mobius configuration. The authors explained the emphasis on the data collected from the power meter and transmitting this data securely [9]. In [10] authors try to address disclosure of Personal information of a user during the process of proof of work in blockchain IOT environment. The authors prove a concept called as zero knowledge proof. Attributed based access control (ABAC) on Hyperledger Fabric blockchain framework for access control in IOT system is proposed in [11]. The architecture of a blockchain based frame-work using Ethereum to maintain EMR (electronic medical record) was proposed in [12]. The frame-work aims at preserving privacy of the patient data and providing efficient access of medical records to authorized person.

III. PRELIMINARIES

The members in a fabric network are connected through a channel, by which specific set of participants are provided access to specific set of transaction. Hence the privacy and confidentiality both are preserved by allowing the access to the smart contract for only participating nodes. All the existing blockchain system including public/permission less platform have the order-execute architecture. In comparison to order-execute model the execute-order-validate in hyperledger fabric provides: a) Resiliency b) flexibility c) scalability d) performance e) confidentiality.

A. Hyperledger Fabric architecture [13], [14]

The chaincode applications are written and tested by the developers through a structured environment of libraries provided by the Hyperledger Fabric client SDK. Node.js and java are the only two officially supported SDK, where as python, go and REST can be downloaded and tested but they are unofficial. The elements of a blockchain network are peers, ledgers, smart contract, orderer, policies, channels, applications, organizations, identities and membership.

Peers: A blockchain network is formed by a number of peer nodes. Since the ledgers and smart contracts are hosted by peers, they are considered as fundamental elements of blockchain network. The instances of ledger and chaincode are hosted by peer. Any transaction generated by smart contract is recorded immutably in a ledger. In a blockchain network the shared process are encapsulated by smart contract and shared information is encapsulated by ledgers. If the blockchain resources have to be accessed by application and administration, then they should have an interaction with peer since the ledgers and chaincode are hosted by peers. Due to these reasons peers are considered to be basic construction blocks of a hyperledger fabric blockchain network. Peers of organization are connected through channel. A peer performs many roles such as an endorsing peer, committing peer, anchor peer or a leading peer. The endorsing peers involve in executing smart contract during a transaction and they return signed response back to client application. The committing peers involve in validating the blocks of transactions that are orderly arranged and applies block to its local ledger copy. Since all peers store a copy of ledger, hence all peers in the network can take the role of committing peer. An anchor peer will be the first peer in the channel that will be discovered by other organizations on the network. Leading peers involve in communicating with ordering service in situations, where an organization has several peers.

Blockchain ledger: A blockchain ledger contains world state database and blockchain**.** The latest values of a group of ledger states are placed in World state that is nothing but a collection of data. The world state helps the programmer in getting the recent values of the states, instead of finding them by going through entire transaction log. Key-value pairs are used to articulate Ledgers. Blockchain contains transaction information, framed as interlinked blocks. Each block has a series of transactions that indicates updating the world state. Block chain captures all the modifications that settle on the world state. Transactions are accumulated within the blocks that are added to the blockchain. Once data written to the blockchain, it cannot be modified, this architecture is very diverse from world state, where world state always changes based on updates.

A blockchain is a series of blocks, each has a set of ordered transactions which are immutable.

Smart contract: It defines a set of rules to access information from blockchain. If a client application has to invoke smart contract, it has to be installed and instantiated on the peers.

Orderer nodes: These nodes involve in receiving the transaction proposal response from the peers, that are packaged as transactions in a block and delivered to peers in order to update the peer's local replica of ledger. An ordering service is a collection of orderer nodes within the network and there will be a single ordering service for a network. The policies of channel and membership information of each member of channel are maintained in channel configuration. Ordering service will have the channel configuration for the network and hence they administer a network.

Network Policies: The certificate authority provides the permission record for organizations to authenticate to the network. The client applications use these certificates to verify transaction proposal and peers use these to approve transaction proposal and append transaction to the ledger.

Channel: Channel is the secure communication link that connects the members of the blockchain network. By creating a particular channel a set of specific member components can communicate, achieving data isolation and maintaining confidentiality. The configuration block evaluates the validity of the channel. In order to perform a transaction through a channel, the member has to authenticate to channel.

Identities and MSP: The actors in a blockchain network have digital identity that is encapsulated in X.509 digital certificate. These identities are used to determine whether the particular actor should be given permission to access resources and information from the blockchain network. An identity is verifiable if it is a trusted authority like MSP. The policies that govern valid identities for organization are defined by an MSP (Membership service Provider). The X.509 certificates are used as identities in implementation of MSP in fabric. The MSP lists the identities to define the members of an organization.

IV. SYSTEM MODEL AND DESIGN

The overall IOT system architecture consists of five major implementation blocks. They are Blockchain Network, Web server, Web client, Mobile client, Arduino client (smart energy meter)

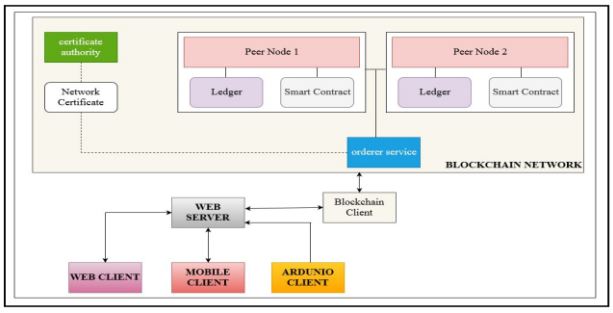

Fig. 1.IOT System overview

Implementation Steps:
1. Create a web server and host APIs to get data from IOT system.
2. Establish communication between the IOT sensor device and the server.
3. Create a web client to monitor and manage admin activities like creating and modifying user, verification, etc..
4. Create a mobile client to monitor activities of IOT devices registered for a user.
5. Setup a blockchain system and establish connection with the web server.

A. Blockchain network:

Blockchain network consists of Certificate Authority (CA) which issues the certificates for actors (peers, ordering node etc) to authenticate to the network. The peers, orderers etc are the active elements of blockchain network which provides/use Network service has digital identities. These identities are enclosed in an X.509 certificate and they specify the permissions for accessing resources of blockchain. X.509 certificates are used as identities in MSP implementation of Hyperledger Fabric. These identities are recognized by MSP (Membership Service Provider) in Fabric to ensure that the request is from a trusted source. The participants of a particular organization of fabric blockchain networks are identified by MSP.

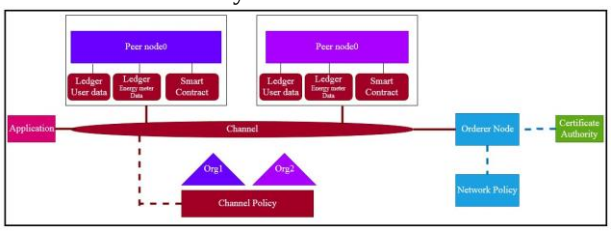

Fig. 2.Blockchain network

In this application the blockchain network is built using hyplerledger fabric frame work [15]. The network is constructed by network policy, an ordering service having single order node and two organizations having one peer each. The certificate authority will issue digital certificate to all these participants. A channel is formed between the peer0 of Org1 and peer0 of Org2. The orderer node manages the channel. The application is also linked to channel by granting permissions. The peers of the network maintain two ledgers one for User Data and other for Energy meter data (Usagedata) associated with channel. A single smart contract with multiple functions runs on peers. Peers use this chain code to access information from blockchain ledger.

B. Web server:

Web server is a system program that responds to clients request by providing the client with web pages which uses Hypertext Transfer Protocol to send the files that serve Web pages to users. The web server processes and provides a web page to the client. A web server for RESTful API is built using NodeJS. The framework used is Express 4 with its router and ejs template engine.

It is then deployed into an azure cloud. The system requirements will change as the size of the blockchain changes with the course of time.

Minimum system requirements of azure cloud.

1. 2 core CPU
2. 4GB Memory
3. 10 GB of HDD/SSD
4. Linux based OS

The HTTP routes are defined as provided by the express npm module. The application is divided into UI routes and API routes. The API routes start with the path /API. Following are the routes defined in the application

POST, GET: /api/user - Creates user and fetches users list

POST, GET: /api/usage/ - Create usage entry for user and fetches usages list

GET: /api/usage/:userId - Fetches usage info for a user ID

GET: /api/user/:userId - Fetches user info for a user ID

This application is dependent on the blockchain module that was developed as part of this project. The blockchain module is packaged as a javascript module and is imported using RequireJS pattern.

All the activities described in the smart contract are hooked with necessary javascript constructs and exported as functions in the blockchain module. The respective REST APIs are programmed to handle the queries and invocation requests to the blockchain by registering respective handlers provided by the module.

C. Web client and Mobile client:

They fetch the information from the server at particular API. They provide user interface. Mobile application was developed as a part of this project in android studio using kotlin language [16]. Mobile client can only fetch information of a particular user. The Web client is provided with access to view all users information and also with access for creation of new users. A User ID is generated for every new user created. Using this User ID to generate transactions avoids account data or personal data getting revealed.

D. Arduino client:

The Node MCU acts as an Arduino client, which reads the energy meter data through serial port and POSTs this data to the web server. SDM120M is used as the energy meter which is capable of measuring the Voltage in Volts(V), current in amperes(A), power in Watts(W), frequency in Hertz(Hz), energy in KWh, power factor etc. of the connected load.

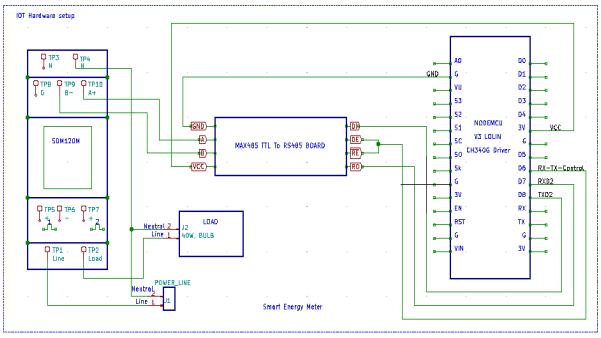

Fig. 3.Arduino Client

SDM120M has a mod bus communication interface to read the measured value. SDM120 has the option of an RS485 communication facility to communicate with systems using the Modbus RTU Protocol. It uses a MAX485 TTL - RS485 board which is a converter that provides two way serial communication signal conversion between the RS485 to TTL and vice versa.

V. RESULTS AND ANALYSIS

Fig 4 describes the details of one of transaction of USER0004 stored in peer0 of Org2. It can be observed that voltage, current, time, frequency, power and energy along with user ID are stored.

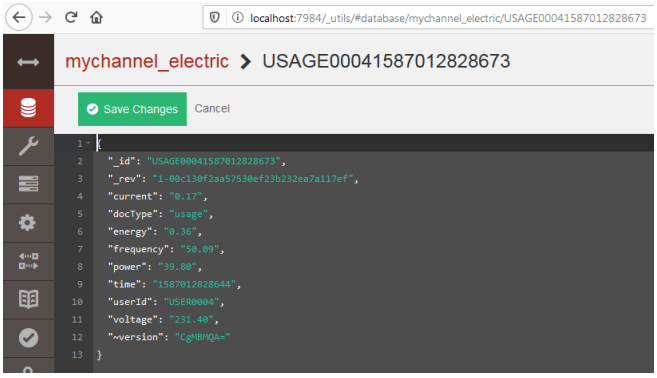

Fig. 4.Details of data in one of the transaction

The detailed transaction record of USER0000 reflected in peer0 of Org1 is shown in Fig 5 and peer0 of Org2 is shown in Fig 6. Both peers have the same data, hence indicating that data is decentralized and distributed.

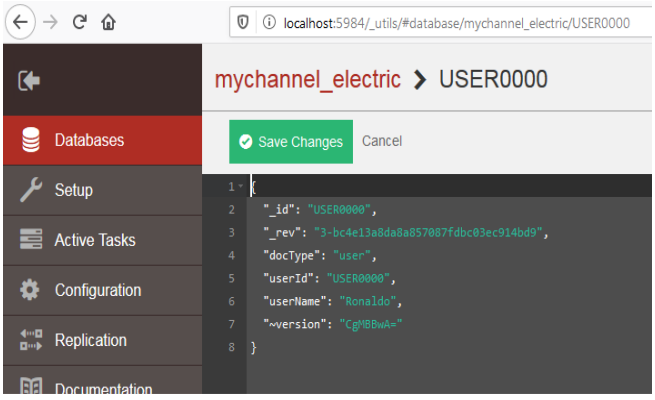

Fig. 5.The detailed transaction record of USER0000 reflected in peer0 of Org1

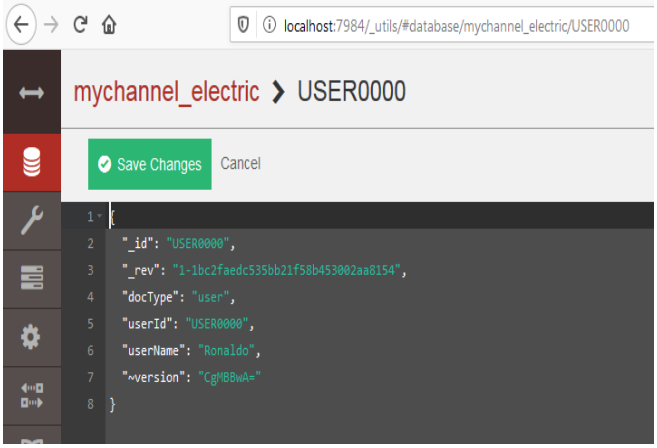

Fig. 6.The detailed transaction record of USER0000 reflected in peer0 of Org2

It can be observed from Fig 7 and Fig 8 that even if one of peer data is tampered, the other peer has original information, thus ensuring the safety of the information.

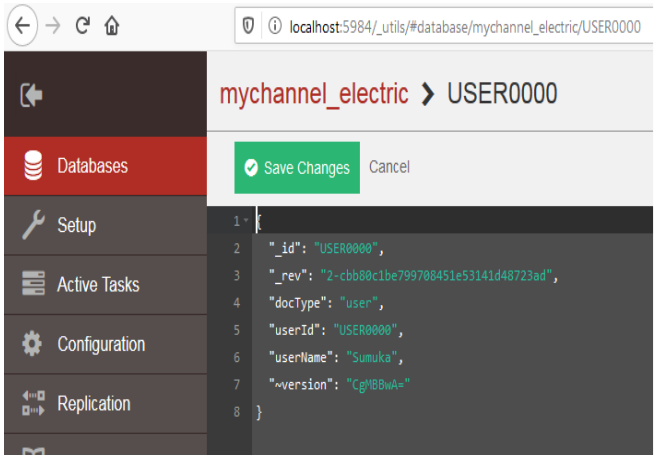

Fig. 7.Transaction details in peer0 of Org 1 after modifying username.

From Fig 7 one can observe that, a data in one of the transactions in the peer0 of org1 related to registration of USER0000 is changed. Here changes are made in username i.e changed from Ronaldo to Sumuka.

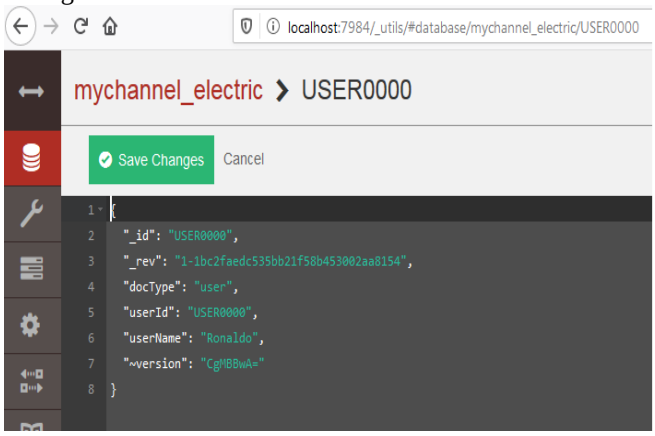

Fig. 8.Transaction details in peer 0 of Org 2 after modifying username in peer0 of Org 1

The results obtained on mobile app named Electric Bill for a Get request to a particular user is shown in Fig 9

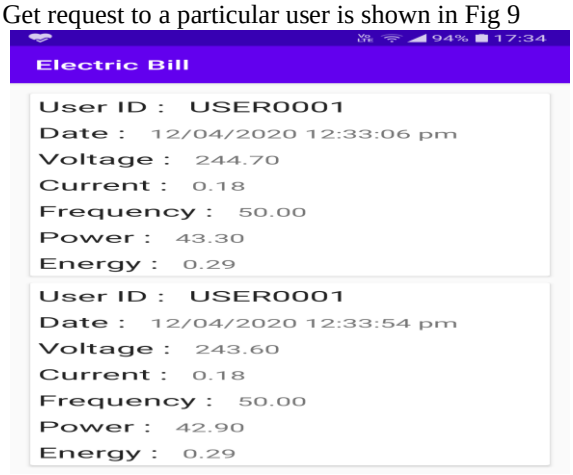

Fig. 9. Mobile client sending a GET request and obtaining a response from web server.

Fig 10 shows the results obtained on web client for GET request for all users.

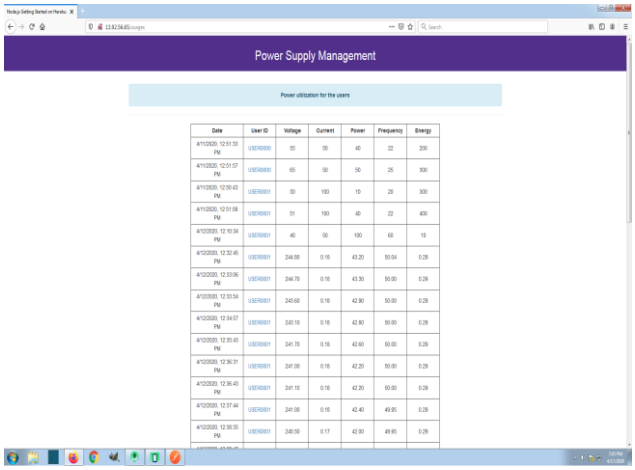

Fig. 10.Web Client sending a GET request and obtaining a response from the web server.

Fig 11 shows the blockchain height before write operation and Fig 12 shows after two write operation on usage details of USER0009. It can be observed that the blockchain height increases after each write operation.

Fig. 11.blockchain hash information before updating the usage details of USER0009

Fig. 12.Blockchain hash information after updating the usage details of USER0009

Performance analysis: The blockchain application that was created using hyperledger fabric is analyzed for its performance by carrying out POST request and GET request. Performance testing on the application is done. Following results were obtained from different application platforms.

Table- I: Types of tests conducted for Performance analysis using different tools.

Test	criteria	Tool	Request	Results
Sequential request test	100 requests within 30 Sec	Postman	POST: /api/usage/<user ID>.	Successfully handled. Test results in Fig 13.
Concurrent Request test	Concurrent request by max 1000 users.	Loadium	POST: /api/usage/<user ID>.	The average throughput is 228.53hits/sec and the average Response is found to be 3.574 seconds. Test results in Fig 14 and Fig 15.

Concurrent Request test	Concurrent request by max 20 users.	Blazemeter	GET: /api/usage.	The average throughput is 7.78hits/sec and the average Response time is found to be 2.12 seconds. Test results in Fig 16 and Fig 17.

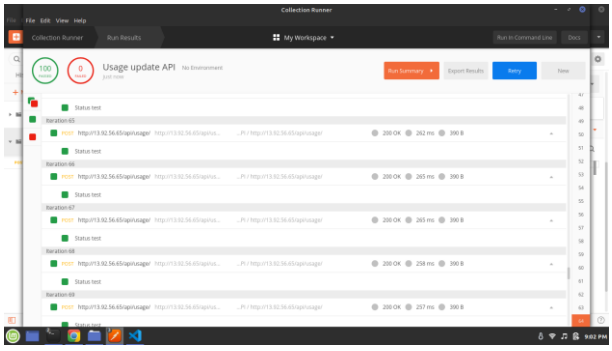

Fig. 13.sequential test results for 100 POST requests within 30 seconds using postman.

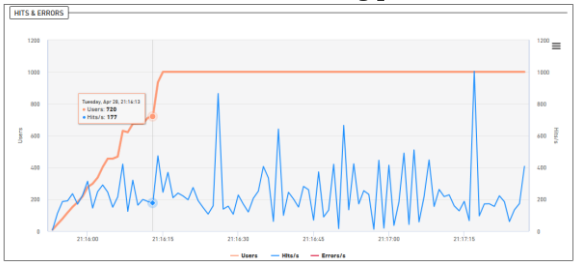

Fig. 14.Load test graph for concurrent POST request using Loadium tool

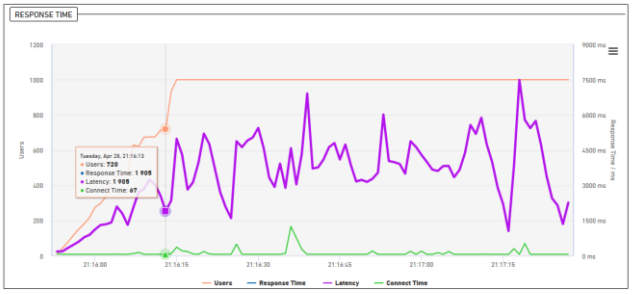

Fig. 15.Response time graph for concurrent POST request using Loadium tool

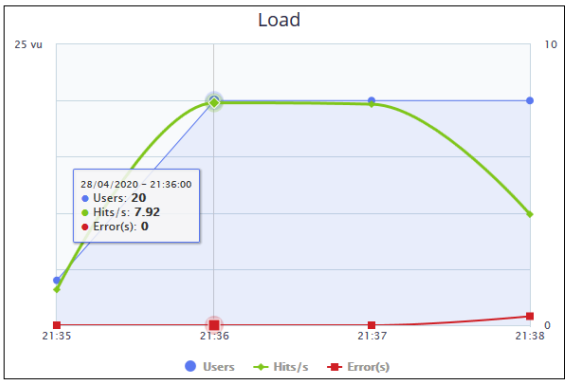

Fig. 16.Load test graph for concurrent GET request using Blazemeter tool

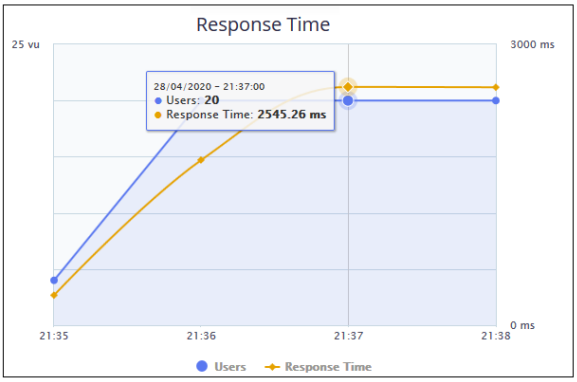

Fig. 17.Response time graph for concurrent GET request using Blazemeter tool

Comparing performance with ethereum blockchain where algorithm used is Proof of Work (PoW) and Proof of Assignment (PoA) [17], the consensus algorithm for hyperledger fabric, has the latency ranges from 1.5-7.5 seconds.

Reasons to use multiple test tools are

- Postman tools have the capability to provide precise information that was required for testing the update request.
- It is a sequential execution of the requests and since the application was launched as a single threaded application, Postman worked for the intended testing which was to measure average time taken for a transaction.
- To understand the behaviour of the application for concurrent requests, Jmeter based loadium tool was used.
- Using blazemeter which is also based on Jmeter, was intended towards monitoring the data requests. Usage of similar tools was to remove the ambiguity of skewed results.

VI. CONCLUSION AND FUTURE SCOPE

The complete setup provides a visualization of an IOT ecosystem that has both trusted and non-trusted parties. The integrity of data is maintained across the ecosystem with this tamper-proof system. The results of the performance tests show that the normal functioning and usability of the system is fairly under acceptable terms. The comparative performance analysis of the framework within the system with other framework shows a positive outcome. Along with data security, the very basic element of this system i.e. distributed system provides availability, scaling, backup and recovery which is an ever-growing need in IOT.

Overall, the complete setup proves to be a highly tamper proof and suitable for a sustainable IOT system. The authors would like to continue this work with the introduction of data security while sending data from Arduino client to server. To make data more secure some encryption techniques can be used at the Arduino client, and use decryption techniques at the server side. The work can be extended to a blockchain network in which the peers can notify orderer and other peers in the blockchain network regarding the modifications made to its local copy.

This data can be used to penalize the peer or take regulatory action to strengthen security. In this application, the experiments are conducted for two organizations with one peer node each. In continuation to the above the network can be scaled to assist many IOT devices and applications.

REFERENCES

1. Obaid Ur-Rehman, NatasaZivic, ChristophRuland," Security issues in smart metering systems", IEEE International Conference on Smart Energy Grid Engineering (SEGE) , 2015
2. Pardeep Kumar, Yun Lin , Guangdong Bai ,Andrew Paverd, Jin Song Dong , Andrew Martin, "Smart Grid Metering Networks: A Survey on Security, Privacy and Open Research Issues", IEEE Communications Surveys & Tutorials ,Volume: 21 , Issue: 3 , 2019.
3. Mohsin Kamal , Muhammad Tariq, "Light-Weight Security and Blockchain Based Provenance for Advanced Metering Infrastructure", IEEE Access (Volume: 7), 2019, INSPEC Accession Number: 18826750
4. Ruiguo Yu, Jianrong Wang, Tianyi Xu , Jie Gao , Yongli An , Gong Zhang , Mei Yu , "Authentication With Block-Chain Algorithm and Text Encryption Protocol in Calculation of Social Network ", IEEE Access (Volume: 5), 09 November 2017
5. DinanFakhri, KusprasaptaMutijarsa, "Secure IoT Communication using Blockchain Technology", International Symposium on Electronics and Smart Devices (ISESD), 2018, INSPEC Accession Number: 18374691
6. Pin Lv , Licheng Wang , Huijun Zhu , Wenbo Deng , LizeGu, "An IOT-Oriented Privacy-Preserving Publish/Subscribe Model Over Blockchains", IEEE Access (Volume: 7), march 2019, INSPEC Accession Number: 18576298
7. Mary Subaja Christo, AnigoMerjora A, ParthaSarathy G, Priyanka C and Raj Kumari M, "An Efficient Data Security in Medical Report using Block Chain Technology", International Conference on Communication and Signal Processing (ICCSP), 2019
8. Jin HyeongJeon ; Ki-Hyung Kim ; Jai-Hoon Kim, "Block chain based data security enhanced IoT server platform", International Conference on Information Networking (ICOIN), 2018, INSPEC Accession Number: 17720930
9. XiPeiyu,ZhangQian,WangHaining ,ZhaoHaoyue ,WangChunyan , "Exploration of Block chain Technology in Electric Power transaction", International Conference on Power System Technology (POWERCON), 2018, INSPEC Accession Number: 18392665.
10. Chan Hyeok Lee , Ki-Hyung Kim, "Implementation of IoT system using block chain with authentication and data protection", International Conference on Information Networking (ICOIN), 2018, INSPEC Accession Number: 17720922.
11. Han Liu ; Dezhi Han ; Dun Li, "Fabric-iot: A Blockchain-Based Access Control System in IoT", IEEE Access (Volume: 8 Page(s): 18207 – 18218), January 2020,Electronic ISSN: 2169-3536.
12. Eman-Yasser Daraghmi, Yousef-AwwadDaraghmi, Shyan-Ming Yuan, "MedChain: A Design of Blockchain-Based System for Medical Records Access and Permissions Management", IEEE Access (Volume: 7 , Page(s): 164595 - 164613), November 2019,INSPEC Accession Number:19144264.
13. https://hyperledger-fabric-ca.readthedocs.io/en/release-1.4/users-guide.html
14. https://hyperledger-fabric.readthedocs.io/en/release-2.0/key_concepts.html
15. https://hyperledger-fabric.readthedocs.io/en/release-2.0/build_network.html
16. https://kotlinlang.org/docs/reference/android-overview.html
17. Markus Schäffer,Monika di Angelo and GernotSalzer, "Performance and scalability of private ethereum Blockchains", International conference on process Management, August 2019, Online ISBN 978-3-030-30429-4

AUTHORS PROFILE

Kathayayani N, Student pursuing Master of Technology in the department of Electronics and Communication at B. M. S college of Engineering, Bengaluru, India. Graduated as Electronics and Communication Engineer in the year 2009 from SJC Institute of Technology , Chickballapur, Karnataka, India. After completing her degree she is been serving in Department of Technical education since 2011 and working in a Government Polytechnic as Lecturer. She has handled subjects such as ARM Controller, Analog and Digital Communication, Verilog, Analog and Digital Electronics, Embedded Systems for Diploma students. She has also guided many students in completion of their final year diploma projects.

Dr. Jayanthi K Murthy , is presently serving as an Associate Professor in Department of Electronics and Communication, BMS College of Engineering, Bangalore, India with 25 years of experience. She completed her B.E and M.E. from Bangalore University and received her Ph.D in Wireless Sensor Networks from VTU. Her main areas of interest are in Wireless communication, networking and security. She has authored more than 20 research papers in international conferences and repute journals. She was also the co convenor for the first National Conference-NEWS 2010. She has got
grants of 20 lakhs from Vision Group of Science and Technology, Dept. of IT BT, Govt. of Karnataka, 45 lakhs through TEQIP II funding and is actively involved in setting up a BMSCE- Keysight Co-Funded Advanced Communication lab worth about 2.50 Crores.

Vaibhav Nityanand Naik, has completed his graduation in Electronics and Communication Engineering from SDM Institute of Technology, Ujire, India, affiliated to Visvesvaraya Technological University. He started his career in embedded programming focusing on device drivers and communication applications. He is having a fair experience working on communications protocols for web applications and desktop application. He has worked as a full stack web developer and collaborated on projects for setting up web development team. He has worked with a startup on logistics applications and headed projects on web technology. He has been passionately contributing to technological advancements in the industry for past five years and mentored many during the journey.