Multimedia Tools and Applications (2021) 80:18901–18915
https://doi.org/10.1007/s11042-021-10715-4

Ethereum for Internet of Things security

Aparna Raj[1] · Kavita Maji[1] · Sujala D. Shetty[1]

Received: 15 July 2020 / Revised: 24 January 2021 / Accepted: 10 February 2021 /
Published online: 20 February 2021

Abstract

The influence of Internet of Things (IoT) is growing exponentially in the consumer patterns and will continue to increase in the coming years. With millions of interconnected devices over the internet, IoT is currently running into its monumental security and centralization issues that can be resolved using Blockchain. This paper aims to demonstrate the use of a smart contract on a private Ethereum Blockchain (also known as permission less Blockchain) to check whether a user holds enough tokens to utilize a service. Private in this context implies a private instance of the Ethereum implementation in which we make use of the same security and principles of the Ethereum main chain on a personal network of connected IoT devices which improves the overall privacy and security of the system than the traditional schemes. The proposed model consists of a target IoT device controlled by a Raspberry Pi, running a client application that makes decisions based on the state of the smart contract. With this environment, we are able to achieve the following goals: (1) no data leaves the private network (2) all services are employed with transparency and accountability (3) only registered devices can issue requests for using a service. Although the demonstration is made on a home network, the model presented in this paper can be utilized in commercial environments where any service can be availed from a wide array of smart devices through a mobile application.

Keywords Internet of things (IoT) · Ethereum · Blockchain · IoT Security · Home automation systems · IoT applications

✉ Aparna Raj
p20190003@dubai.bits–pilani.ac.in

Kavita Maji
h20180002@dubai.bits-pilani.ac.in

Sujala D. Shetty
sujala@dubai.bits-pilani.ac.in

[1] Department of Computer Science, Birla Institute of Technology and Science, Pilani, Dubai Campus, Dubai, United Arab Emirates

1 Introduction

The term IoT was coined by Kevin Ashton in 1999, however it started gaining popularity in the earliest of 2010. By 2013, IoT had developed into an ecosystem that could host multitude of devices ranging from wireless to Micro-Electro-Mechanical Systems (MEMS) to embedded systems. We now have driverless cars equipped with sensors, gyroscopes, cloud infrastructure etc. for sensing and processing huge amounts of traffic data, pedestrian patterns, and road conditions within fraction of a second for making reliable driving decisions. We also have smart homes with the capability of automatically regulating lighting, ambience, locking and temperature facilities and can be secured remotely on the press of a button. Wearables like smart watches can run mobile applications, make calls, inform us of burnt calories and even make online payments.

The technology supporting IoT picked up swiftly with global acceptance, but its security aspects gained attention much later. Critical safety protocols were sidelined to meet the rapidly growing demands. Most manufacturers offered software updates for a shorter period, sometimes none. In addition, unsupported legacy Linux kernels were used, and consumers were exposed to potential attacks using outdated hardware and software. Weak credentials and default passwords make majority of the IoT devices insecure and prone to hacking. Companies providing cloud services harness data from devices and can sell or share consumer information to third parties. Limited computing power, minimal security provisions and centralized architecture made these devices vulnerable to security attacks [10]. Also increased communication between IoT devices requires huge amounts of critical data and sensitive information to be transmitted over the network which further rises the chances of even more vulnerabilities [9].

Early work on Blockchain began in 1991 with the development of cryptographically secured chain of blocks with tamperproof timestamps by Stuart Haber and Scott Stornetta. In 2008, Bitcoin appeared as the first application of Blockchain Technology. Soon many applications sprung up to leverage the capabilities of the digital ledger technology. The second pivotal phase began when Vitalik Buterin proposed Ethereum to utilize Blockchain Technology not just as a payment system but also for storing computer codes which can be used to power up decentralized applications immune to cyber-attacks [8]. In the following years, Hyperledger fabric and IOTA came into existence featuring more complex applications on Blockchain platform. Some of the challenges faced by conventional IoT frameworks are as follows:

- Centralized architecture in which devices are connected using centralized cloud servers which leads to single point of failure
- Constrained resource consumption due to limited memory of IoT devices [21]
- Data security and privacy issues as most of the user privacy data are being transmitted across the network [2]

Blockchain overpowered various security and scalability issues associated with IoT through decentralization. It stores all the transaction histories of smart devices in an immutable manner making it a transparent trusted ledger [1]. Since data is transmitted on a peer-peer basis, it further reduces the overall operational costs and accelerates the data exchanges [12].

The main contributions of the research work include:

- We propose a Blockchain based smart home network architecture to mitigate the security challenges prevailing in centralized network architecture and to counter security attacks
- We implemented the proposed architecture using a decentralized Ethereum Blockchain Technology to deliver different security requirements by developing and testing a smart contract

In this paper, we use Ethereum smart contracts for establishing communication between IoT devices in a private home network. Smart contracts allow users to execute codes on a Blockchain network based on a set of rules in agreement by all the parties thereby eliminating the need of a middleman. A smart phone, LED and a Raspberry Pi are used in the demonstration of secure smart lighting. Smart phone can access the current state of LED through a function deployed in the smart contract. Pi is used by the smart contract to sense the state of LED and send a response to the smart phone. The phone can then initiate a transaction from its Ethereum wallet to change the LED state (ON/OFF). The scenario is aimed at illustrating how a user can remotely control home lighting from his smart phone.

The remainder of this paper is organized as follows. The background (Section 2) explains the components and working principles of Ethereum and IoT systems. The design structure (Section 3) helps to understand the proposed model and why it functions better than MQTT protocol (default for IoT communication). The implementation section (Section 4) describes how to develop and test your own smart contracts for your IoT devices to exchange data securely over the private Ethereum network. We then conclude by evaluating the success of the experiment and setting a milestone for future work.

2 Background

2.1 Blockchain

Blockchain Technology has drawn immense attention from both industry and academia in the last couple of years. The ability to keep the past and present transactions secure, safe and tamper resistant makes it as a unique technology. Some of the major characteristics of Blockchain are its decentralized nature, auditability, anonymity, and persistency. Blockchain is the technology used for cryptocurrency bitcoin and it was invented by a person (or a group of people) under the name Satoshi Nakamoto in 2008 as a solution to double spending problem. Double spending problem is a scenario in which the same digital currency is spent more than once [35].

Blockchain is an array of linearly connected blocks of information secured with cryptography to prevent the blocks from unauthorized access and alterations. It uses SHA 256 for encryption and, for validating users it makes use of digital signature. Each block contains a block header comprising meta data of the block, hash of the previous block, list of valid transactions, mining statistics for block creation and Merkle tree root for verifying the transactions. Each user has their own pair of public and private key where, the public key is used for identifying the user by verifying signature and private key for signing the message [22] .

Hash acts as a fingerprint to identify whether a block and its contents are unique and any changes to the data in the block, changes the hash value of the block and makes the following blocks invalid. However, computers these days, are fast enough to calculate different hash values of each block and hence can tamper a block and recalculate all the hashes of each blocks in the chain within small amount of time to make it valid. Hashing alone cannot deal with data tampering and to overcome this Blockchain comes up with consensus mechanism making it more secure. It contains a set of rules that decides the contributions of each participants to meet the necessary agreement on a data value or the state of network in a Blockchain. Proof of Work (PoW), Proof of Stake (PoS), Proof of Capacity (PoC) etc. are some examples of consensus mechanism algorithms that slow down the creation of new blocks based on different principles [6]. When a new block is created, it is sent to everyone on the network and then each node verifies the block to determine whether its contents are not being tampered and on successfully completing the verification, each node adds the current block to their own Blockchain. Hence all the nodes have reached on consensus and if any block is tampered, it will be rejected by all other nodes in the network [25]. All these steps encompass the working of a Blockchain and is illustrated in Fig. 1. Other than Bitcoin cryptocurrency, additional domains that can make use of Blockchain includes healthcare applications, creating e-notary, collecting taxes, financial applications, IoT, e-business applications, transportation, and supply chains etc.

2.2 Smart contracts

Blockchains are constantly evolving and smart contracts are one of the recent developments, which are programs stored on Blockchain used to exchange coins automatically when certain predefined conditions are met. It is deployed on Blockchain with a unique address and can hold many individual/correlated accounts. Funds can be held or dispersed in it like an escrow and state variables are maintained. Escrow is a legal notion for protecting and controlling financial assets, in which the assets are owned by a trusted third party who controls the entire process and ensures that all the commitments are met. Here nobody can make use of the money without an agreement from other participants and the funds are released only once all the pre-conditions are met. Users can access them as interfaces to trigger a range of functions by sending transactions to the contract and every node can view the details of smart contract and freely execute the instructions. Every interaction maintains a log that are transparent to all the participants of the network.

2.3 Ethereum

Ethereum was proposed in late 2013 and rolled out in 2014 by Vitalik Buterin. It is a network of computers combined into a single powerful decentralized supercomputer for executing codes to power up decentralized applications called Dapps. Ethereum allows people to connect directly without the need of a centralized authority and its goal is to decentralize the internet. It makes use of Ethereum programming language called Solidity for writing smart contracts and the currency used to incentivize the network is called Ether (ETH). Current Ethereum Blockchain uses consensus algorithm specifically designed for itself called Ethash (Ethash PoW algorithm). Smart contracts are compiled and executed in Ethereum Virtual Machine (EVM) which is a turing complete language, which refers to a system capable of performing any logical step of a computational function. Bitcoin does not require such scripting language as they were written for carrying out single function such as money transfer from one address to another [23]. Every operation that takes part in an Ethereum requires some amount of gas. Gas is a unit that measures the amount of computational effort taken to execute any

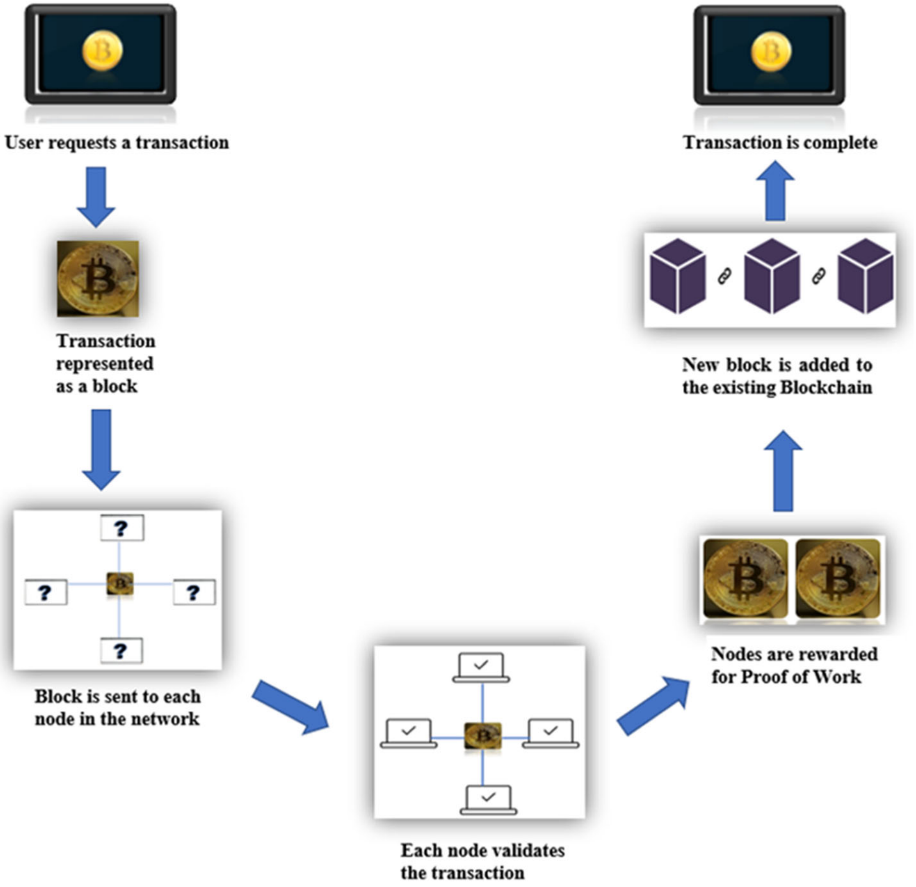

User requests a transaction
Transaction represented as a block
Block is sent to each node in the network
Each node validates the transaction
Nodes are rewarded for Proof of Work
New block is added to the existing Blockchain
Transaction is complete

Fig. 1 Blockchain mechanism

transactions or smart contracts in an EVM [34]. Every block has a gas limit that dictates the maximum amount of gas that can be spent within its operations. Such limits are imposed to encourage quality application codes and to protect Blockchain from Distributed Denial of Service (DDoS) attacks.

2.4 Internet of things (IoT)

IoT eco-system comprises of different devices such as sensors, actuators, processors etc. connected with each other. IoT when combined with devices, gateway and internet can perform any operations in a smarter manner and assists in making decisions from the acquired data [18]. It bridges the gap between physical and digital world. Anyone can access data irrespective of their locations, hence saving time and money. Data can be processed either locally on the device itself or on a fog device, or even on a distant cloud infrastructure. IoT devices communicate with each other using different technologies (Wi-Fi, Bluetooth, ZigBee, Z-Wave etc.) and protocols (Constrained Application Protocol, Message Queuing Telemetry Transport etc.) [33].

Blockchain when combined with IoT provides the users a trusted sharing service, with enhanced transparency, security, and comfort levels. IoT captures real time data from its surrounding environment using different sensors/actuators. Mostly a user's private and sensitive data are being sent over the network, Blockchain is used to enhance data security during the transit and to prevent unauthorized access to this data by third parties. This reduces any chances of mistrust and theft in the network and provides a secure storage. Blockchain being a decentralized ledger and every device in the network being registered within it, safeguards the system from spoofing attacks [28]. Combining Blockchain with IoT has emerged as one of the key trends of current times. Some of the advantages of combining IoT and Blockchain are increased interoperability among IoT devices, increased traceability and reliability of data, increased security using cryptographic algorithms and automated system interactions without trusted third parties [14]. Some of the Blockchain in IoT real world applications are in supply chain and logistics, Smart homes, Automotive industries, Pharmacy etc. to name a few.

2.5 Home automation systems

Home automation systems deal with automating different home devices, thereby improving the comfort, ease, security, and quality of life. In the past decade, these systems achieved greater popularity and were massively used for ambient assisted living. They also ease the life of elderly and disabled people, keep track of their activities and provide necessary support [4]. These devices can be effectively monitored and controlled using different devices such as smartphones, laptops, tablets etc. from a remote location.

Home automation systems offer a wide range of services such as controlling appliances, thermostats, lighting, real time video surveillance and text alerts, entertainment systems, security systems etc. Some of the popular smart home systems are Amazon Echo, Nest Learning Thermostat, Samsung Smart-Things Hub, Google Nest, August Wi-Fi Smart Lock etc. However, the security and privacy of homeowner is a challenging issue as more and more private data are shared among these IoT devices [16]. Blockchain can be combined with these IoT devices which decentralizes the communication between these devices and shields them from cyber-attacks [11].

3 Related works

This section depicts a brief synopsis on state-of-the-art usage of Ethereum for IoT applications security. In [17], authors have proposed a mechanism to manage IoT devices such as air conditioner, lightbulb and smart mete-r using Ethereum Blockchain platform. They have written three smart contracts, one for tracking the smart meter value and others for saving the power usage of both air conditioner and lightbulb by switching onto energy saving mode when the power consumption hits a certain value. Since all the transactions were stored and executed via consensus algorithm, it was not viable for the attackers to forge or tamper data easily, which made the system stand against Denial-of-Service attacks and forgery attacks. But the proposed mechanism requires high transaction time which makes it difficult to apply for time sensitive domains. It also requires larger storage which makes it infeasible for small IoT devices with constrained storage capacity.

In [3], authors have demonstrated the use of Ethereum Blockchain in IoT by building a system for detecting floods in a reservoir. The system consists of three nodes and each node updates its water level to the smart contract based on which flood detection mechanism is carried out. When a certain threshold value is reached flood is acknowledged, and the discharging pump is activated. Here the authors make use of Proof of Authority (PoA) mechanism to highlight the possible latency and network stability problems faced by the real-world applications. But more blocks required to be added and tested to precisely find the problems that hinder the integration of Blockchain and IoT. In [7], authors have presented a methodology using private Blockchain for securing and maintaining the privacy of Smart Home System (SHS) operations. SHS is an integration of different home appliances consisting of lighting, heating, home security, air conditioning, health care systems etc. along with sensors. The Ethereum Blockchain allows the owner to keep track of all the transaction history done within their SHS through different policy mechanisms. Here, the owner is only the authorized person who can access and monitor the data within SHS. Major challenge faced by the proposed approach is the higher transaction time required by Ethereum Blockchain which makes it difficult to use for time sensitive applications demanding immediate responses.

In [32], authors have introduced an emergency service system called Home Service Provider (HSP) for SHS based on Ethereum Blockchain. The sensor manager collects the environmental data and if the data value crosses a pre-defined threshold, it automatically sends an emergency call to HSP and the HSP staffs responds to the call. For authentication, the system makes use of Interplanetary File System (IPFS) with digital signature and Public Key Cryptography (PKI) for responding to the emergency calls that protects the system from Distributed Denial of Service (DDoS) attack from any rouge IoT devices (Table 1).

In [24], authors have proposed a Proof of Concept (PoC) based Blockchain mechanism for verifying the authenticity and integrity of the firmware prior downloading them to the IoT devices. This allows each IoT device to track and verify that they are downloading software's from legitimate service providers. If there is any modification in the firmware during transit, the model successfully identifies it and makes the software modifications invalid. Authors in [29], suggested that Publish-Subscribe mechanism could be efficiently used in an IoT-Blockchain ecosystem where the publisher node receives the data generated by the connected IoT device and stores it in an off-chain database. Later it generates the hash of the database location and sends it to each node on the Blockchain network which are subscribed to that publisher. The Ethereum client intercepts this transaction containing the hash and then the subscriber verifies the address which sent the transaction to validate the sender and thwart the attempts of any spam which makes the system more effective. But the proposed system does not consider the time taken by the publisher and subscriber during the mining and block validation mechanisms.

In [36], authors have proposed a novel security mechanism based on human centric computing by combining machine learning and various privacy protection policies. But the method efficiently works only in cases having large training data sets. In [27], authors presented a privacy preserving approach for secure data storage in a home network. But it requires symmetric key cryptographic principles to be deployed. In [5], authors evaluated the impact of various security attacks on smart home for further analysis and to device appropriate solutions. In [30], authors proposed a smart home use case scenario for improving secured communication using Ethereum Blockchain. But even least errors turn out to be time consuming and inexpensive. In [31], authors applied Proof of Authority (PoA) mechanism to manage appliances in a home network without using any third party cloud services. Even though the system provides higher performance, security and privacy is a major concern. In

Table 1 Comparative study on using Ethereum for IoT applications and its key features

Article	Application	Methodology	Key features	Disadvantages
Huh S et al. [11]	Home automation.	3 Smart Contracts for better privacy and security.	Improved data security, Prevents DoS and forgery attacks.	Requires higher transaction time and larger storage.
Alrubei et al. [17]	Detecting floods in a reservoir.	Proof of Authority mechanism.	PoA mechanism used for providing network stability.	More blocks needed to be added and tested for accuracy.
Aung et al. [3]	Smart home system.	Authentication mechanism.	Only authorized persons can access and monitor data in the network.	Cannot be used for time sensitive applications.
Tantidham et al. [7]	Home service provider.	Public key cryptography.	Provides higher security against rouge systems.	No access control mechanism.
Rifi et al. [24]	Home automation.	Publish subscribe mechanism.	Only authorized persons can access the network.	Mining and block validation mechanism requires higher time.
Yin et al. [29]	Home automation.	Machine Learning approach.	Highly secured network, maintains privacy.	Requires large training data sets.
Poh Sen et al. [36]	Home network.	Symmetric key cryptography.	Highly secured network.	More time required.
Selim et al. [5]	Smart home.	Proof of Concept mechanism.	Secured communication.	Time consuming.
Singh et al. [30]	Smart home.	Proof of Authority mechanism.	High performance.	Security and privacy.
Maode et al. [31]	Home automation.	Public key cryptography.	Easily contol home appliances.	More ethers to be spend.
Hossain et al. [20]	Home devices	Private access control mechanism.	Prevents unauthorized access of devices.	Limited number of transactions done in a time.
Vrunda Paunikar et al. [26]	Home automation	Authentication mechanism.	Highly scalable and secure.	Computationally expensive.

[20], authors simulated a smart home application using Ethereum Blockchain. But for more transactions to be done more ethers needed to be expended and the system also requires higher hardware cost. In [15], authors combined Ethereum with Blockchain for IoT devices for safeguarding and concealing them from unauthorized access. But the system is limited with the number of transactions that can be made in each time. Also, once the smart contract is made least errors in the cryptogram turn out to be time consuming and expensive. In [26] authors proposed an Ethereum based user authentication mechanism for smart home systems. Even though it is highly scalable and secure, it consumes greater number of resources which is challenging for small scale home networks.

A potential smart home user who needs their sensitive data to be confidential within their home network does not trust the third-party service providers. Traditional solutions to this problem make use of a centralized framework which exposes the user network to numerous malicious assaults. Being a single point of attack and if one of the nodes is conceded, it fails the entire system which limits the entire systems scalability, reliability, and security. Therefore, such solutions cannot be adequately applied for decentralized smart home services. Hence a light weight Blockchain mechanism is advisable for maintaining the confidentiality and security of a smart home network.

To overcome such problems the proposed approach in this paper, uses Ethereum to connect with most of the home IoT devices with minimum configuration resulting in fewer resource consumption. Also, it is not required by the owner to depend on external service providers for the upgradation of the communication system between these devices. This is achieved by including the compatibility settings within the smart contract which acts as the gateway of communication/broker between the connected devices. Only the owner can monitor and manage all the activities within the home network which makes it more secure. This mechanism is illustrated via the communication between the user's smart phone and LED and could be easily extrapolated by adding more complex devices to the home network such as homecare, access control management, utility management etc. Hence Ethereum renders the association amongst these devices more trustworthy, secure, agile and transparent.

4 System design

In this experiment, we have used a Raspberry Pi (RPi) 3 with a microSD of 16 GB, a smart phone (features: Android 8.1 OS, 64 GB memory, 6 GB RAM) with a stable internet connection to receive email notifications and a 5mm Infrared LED. Ethereum client is installed on the RPi and the phone. This acts as the wallet from which a transaction can be initiated and in order to invoke the application in phone, we can use any freely available Ethereum wallets such as mist or others and subsequently set the address and Application Binary Interface (ABI) of the contract in the Ethereum wallet which allows to use the functions of the contract. We can also use truffle console or a JavaScript file for connecting and controlling our Ethereum contract and its functions.

The Mining process is done on a personal computer instead of the smartphone node due to its processing capacity. The mining process updates the state of the transactions issued and enables the smart contract to track the latest valid instructions. Miner node is synchronized with the RPi and smartphone nodes. Communication is tested by transferring ethers between accounts of all nodes using eth.sendTransaction command. We then deploy a smart contract on our Private Ethereum network. This smart contract has three functions:

1. deposit Tokens: To deposit tokens from one address to another.
2. withdraw Tokens: To withdraw tokens from one address to another.
3. get Tokens: To retrieve the number of tokens available for an address.

A JavaScript client is used to continuously monitor calls made from the phone to the Ethereum client. It decides which function to be invoked in the smart contract based on a predetermined number of tokens sent by the phone. Based on this selection, the current state of the LED is fetched, or the LED is turned on and off. This whole process is depicted in Fig. 2. Every call made to the RPi is recorded on the Blockchain and the address of the sender and receiver is strictly checked before any operation is permitted.

The transition from MQTT layer to Ethereum makes the communication system impenetrable to external entities. MQTT is a widely used lightweight protocol for resource constrained and IoT devices. It works based on a publish-subscribe model, in which an IoT device acts as client, connects to a proxy server and publishes messages on certain topics.

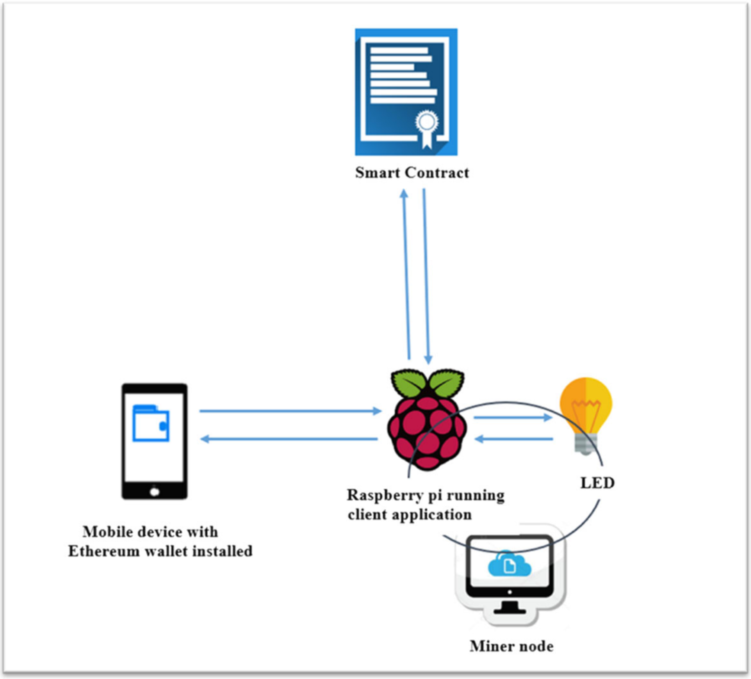


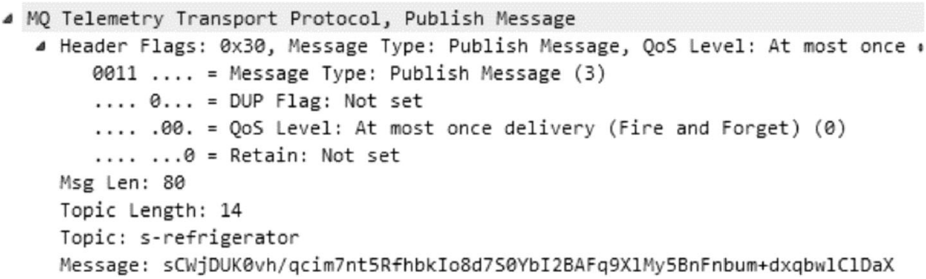
Fig. 2 Working procedure of the Ethereum network connected along with a home device

These messages are then viewed by Subscriber IoT devices. MQTT is supported by TCP protocol which is an unsecured protocol by default. It also utilizes TCP handshakes at the beginning of each communication which results in battery overconsumption. Therefore, we have eliminated the use of MQTT in our experiment.

Algorithm

1. A JavaScript client application runs permanently on the RPi to check incoming requests.
2. Mobile device sends a request to RPi wallet to turn on LED. Miner node mines a block, and this transaction is recorded on the Blockchain.
3. RPi receives the tokens and checks smart contract to figure out which action to perform.
4. Smart contract states that if you receive so and so request from this registered device address, then perform the action: "Toggle current state of bulb".
5. RPi turns on/off LED.
6. RPi sends email/message notification to smartphone indicating task completion.

The numerical comparison of MQTT vs. Blockchain is represented in [13]. Here the MQTT protocol is replaced with an Ethereum Blockchain network and a smart contract. Ethereum acts as the platform for the Blockchain network and the smart contract acts as the intermediary among the connected IoT devices which helps to store and retrieve data within the user home network. The reason why we are using this method instead of MQTT protocol is described using Figs. 2, 3, and 4.



```
◢ MQ Telemetry Transport Protocol, Publish Message
  ◢ Header Flags: 0x30, Message Type: Publish Message, QoS Level: At most once
      0011 .... = Message Type: Publish Message (3)
      .... 0... = DUP Flag: Not set
      .... .00. = QoS Level: At most once delivery (Fire and Forget) (0)
      .... ...0 = Retain: Not set
    Msg Len: 80
    Topic Length: 14
    Topic: s-refrigerator
    Message: sCWjDUK0vh/qcim7nt5RfhbkIo8d7S0YbI2BAFq9X1My5BnFnbum+dxqbwlClDaX
```

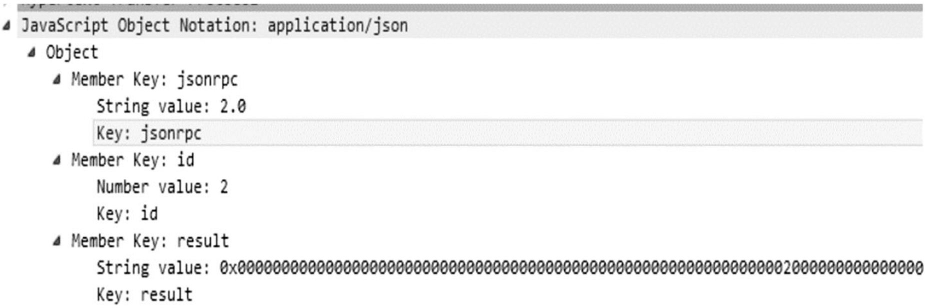
Fig. 3 Is a display of the process of publishing messages done by the smart device. The capture of this process is dangerous because it contains the length of the message to be published, the name of the topic, and the contents of the message itself. The capture of the process on this communication can threaten the integrity of the data. This causes the data to be unsafe and vulnerable to abuse

5 Implementation and testing

For testing scenario, QEMU simulator [19] along with a physical RPi is used. Then Ethereum node is installed on our Raspberry Pi (using Geth command) and personal computer (features: 64-bit Windows 10 OS, Core i7 7700HQ, 16 GB RAM and 256 GB internal memory). Afterwards, initialize the private Blockchain using genesis.json file. Every node in the network uses its own data directory to store database and wallet information and each node will be initialized with the same genesis file. Then the miner is launched. A sample of genesis file is given in Fig. 5.

Later default accounts are created for each of the nodes and we can see the account under the default data directory of the node. Then a miner is built executing a bat script (shell script for Linux). We do not use RPi for the mining process due to its limited computing power. Once mining begins, we can see ethers in the default mining accounts and subsequently test the connection between the nodes by transferring ethers amongst these accounts. We can see some balance in the RPi account while making a transfer from the node of our computer.

We have used Truffle to develop and deploy the smart contract, which is a development framework for Ethereum. To prepare for deployment, update truffle-config.js to include our port and network ID changes. Before proceeding for deployment, make sure that all miners are



```
◢ JavaScript Object Notation: application/json
  ◢ Object
    ◢ Member Key: jsonrpc
        String value: 2.0
        Key: jsonrpc
    ◢ Member Key: id
        Number value: 2
        Key: id
    ◢ Member Key: result
        String value: 0x000000000000000000000000000000000000000000000000000000000000002000000000000000
        Key: result
```

Fig. 4 Here it is seen that Wireshark can capture the JSON format from the implementation of the smart contract performed. From this capture can be observed there is a string value variable whose contents are very long hexadecimal numbers. The hexadecimal number is obtained from a very complicated process in the form of hashing and encoding performed by Ethereum. These numbers are very difficult to be converted back into their original form due to the Blockchain encryption mechanism. Also, any changes in the numbers included in the Blockchain can be easily detected because every block in the network are connected to each other

started up and running. Finally run the command "truffle migrate" to compile and deploy contract on the Blockchain. For testing the contract find the address where our contract is deployed using Remix command as depicted in Fig. 6. This allows to enter the address of a smart contract and to test its functions directly.

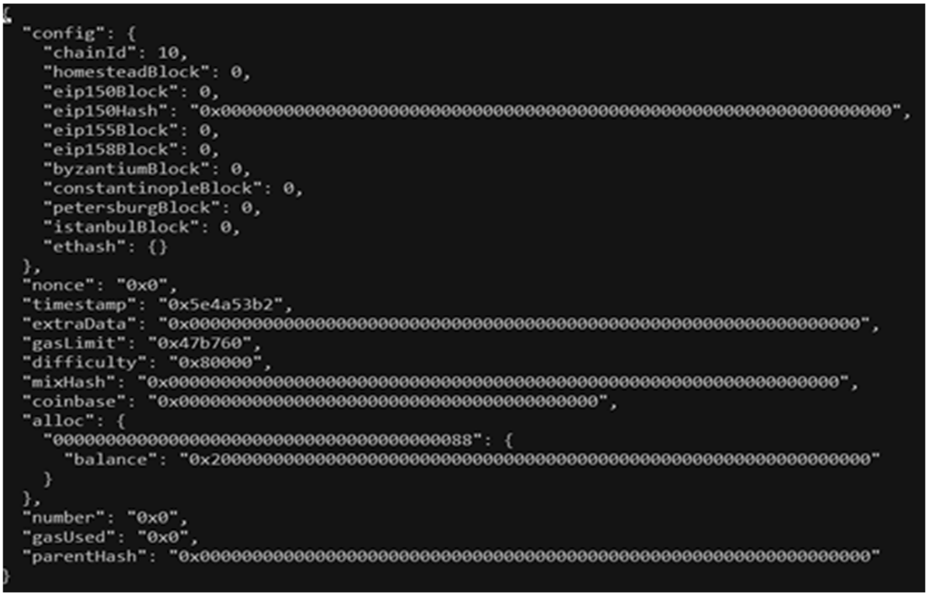
Let us test this system first with a registered device on our private network.

1. Send 5 tokens from Mist wallet on the mobile phone to the RPi account address.
2. RPi receives the tokens and checks smart contract to figure out which action to perform against the given request.
3. Based on conditions written in smart contract, the RPi toggles the current state of the LED.
4. The mobile phone receives an email notification from RPi denoting task completion.

Now let us test this system with an unregistered device.

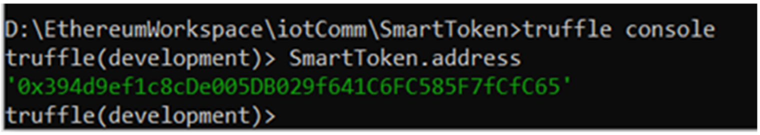
1. Send 5 tokens from a new wallet on an unregistered device (unregistered here means not a connected node of the private IoT network) to the RPi account address. Remember wallet addresses cannot be faked/duplicated on Ethereum network.
2. RPi receives the tokens and checks the smart contract to figure out which action to perform against the given request.
3. The new address fails the trusted device check in the smart contract and the RPi performs no action.

We have not used an optimization tool as we are merely checking whether the expected devices are able to issue requests and the correct action is being performed. However, this



```
{
  "config": {
    "chainId": 10,
    "homesteadBlock": 0,
    "eip150Block": 0,
    "eip150Hash": "0x0000000000000000000000000000000000000000000000000000000000000000",
    "eip155Block": 0,
    "eip158Block": 0,
    "byzantiumBlock": 0,
    "constantinopleBlock": 0,
    "petersburgBlock": 0,
    "istanbulBlock": 0,
    "ethash": {}
  },
  "nonce": "0x0",
  "timestamp": "0x5e4a53b2",
  "extraData": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "gasLimit": "0x47b760",
  "difficulty": "0x80000",
  "mixHash": "0x0000000000000000000000000000000000000000000000000000000000000000",
  "coinbase": "0x0000000000000000000000000000000000000000",
  "alloc": {
    "0000000000000000000000000000000000000088": {
      "balance": "0x200000000000000000000000000000000000000000000000000000000000000"
    }
  },
  "number": "0x0",
  "gasUsed": "0x0",
  "parentHash": "0x0000000000000000000000000000000000000000000000000000000000000000"
}
```

Fig. 5 Genesis block sample which is used for initiating a private Blockchain network



```
D:\EthereumWorkspace\iotComm\SmartToken>truffle console
truffle(development)> SmartToken.address
'0x394d9ef1c8cDe005DB029f641C6FC585F7fCfC65'
truffle(development)>
```

Fig. 6 Locating deployed Smart Contract address using remix which is an online IDE used for Ethereum

system can be improved by adding a blocking address feature if a new address is encountered issuing requests without an entry present in the smart contract. Based on the above tests, it is proven that IoT with Blockchain mechanism can be used to solve the security problems that arise in communication between IoT devices thus ensuring the system integrity.

6 Conclusion and future work

In this paper, we design an Ethereum based communication system between two smart devices. The RPi validates the address of the sender thereby blocking any attempts of impersonation. Only the holder of the smartphone can issue instructions to the RPi. The smart contract has the IP address of the RPi, which is connected as a node on the network. Nobody can imitate the RPi as a node or send instructions to it directly on the Blockchain as the smart contract also validates the address of the sender. Higher number of nodes offer higher security. In real-world applications, there are millions of transactions per minute which dictate the need for sufficiently higher number of miner nodes. If there are more nodes, there will be more data on the Blockchain; hence a greater number of blocks will be mined. Many Blockchains consider the longest chain to be the true version of the ledger. In a Selfish mining attack, a miner can try to keep building blocks in stealth mode on top of the existing chain, and once he leads in block creation by one or two blocks than the current chain in the network, he can publish his private fork, which will be accepted as the new truth since it is the longest chain. He can make fraudulent transactions in the public network and then publish his longer stealth chain to reverse the transactions (double spending problem). Hence, it is advisable to have several devices connected to the private Ethereum network. There is no data leaving the home network since all processing is done on the home computer itself rather than on the cloud, so there is high availability and even higher privacy and security. This concept can be very useful in large scale industrial applications where different kinds of functions (including but not limited to switching device on and off, changing modes, reducing, and increasing speed/temperature/pressure) can be performed to regulate a huge volume of devices through a mobile application.

Blockchain is still in nascent stage and research is being conducted to manifest its true potential. The number of transactions per minute on Blockchain is lower than systems using traditional MQTT protocols. However, it is still more secure than most of the communication protocols being used today. Different aspects of our experimental setup can be switched and tested to fine tune the system efficiency. For instance, the staking mechanism of the Ethereum instance can be changed from PoW to PoS. The number of nodes and miners can be increased or decreased to detect the acceptable number of devices that can be safely connected without hampering system performance. More research is needed to determine how Ethereum performs against competitors like Hyperledger and IBM Blockchain in the field of IoT security.

References

1. Ali J, Ali T, Musa S, Zahrani A (2020) Towards secure IoT communication with smart contracts in a blockchain infrastructure
2. Ali A, Zhu Y, Chen Q et al (2019) Leveraging spatio-Temporal patterns for predicting citywide traffic crowd flows using deep hybrid neural networks. Proc Int Conf Parallel Distrib Syst - ICPADS 2019-December:125–132. https://doi.org/10.1109/ICPADS47876.2019.00025
3. Alrubei S, Rigelsford J, Willis C, Ball E (2019) Ethereum blockchain for securing the internet of things: Practical implementation and performance evaluation. 2019 Int Conf Cyber Secur Prot Digit Serv Cyber Secur 2019 1–5. https://doi.org/10.1109/CyberSecPODS.2019.8885029
4. Asadullah M, Raza A (2016) An overview of home automation systems. 2016 2nd Int Conf Robot Artif Intell ICRAI 2016 27–31. https://doi.org/10.1109/ICRAI.2016.7791223
5. Ashari A, Shouran Z, Kuntoro Priyambodo T (2019) Internet of Things (IoT) of smart home: privacy and security. Artic Int J Comput Appl 182:975–8887. https://doi.org/10.5120/ijca2019918450
6. Atlam HF, Wills GB (2019) Technical aspects of blockchain and IoT, 1st ed. Elsevier Inc, Amsterdam
7. Aung YN, Tantidham T (2017) Review of Ethereum: Smart home case study. Proceeding 2017 2nd Int Conf Inf Technol INCIT 2017 2018-Janua:1–4. https://doi.org/10.1109/INCIT.2017.8257877
8. Buterin V (2014) A next-generation smart contract and decentralized application platform. White paper 3, no. 37
9. Chen H, Pendleton M, Njilla L, Xu S (2020) 67 A Survey on Ethereum systems security: vulnerabilities, attacks, and defenses. ACM Comput Surv 53. https://doi.org/10.1145/3391195
10. Cheng J, Xie L, Tang X et al (2020) A survey of security threats and defense on Blockchain. Multimed Tools Appl. https://doi.org/10.1007/s11042-020-09368-6
11. Dorri A, Kanhere SS, Jurdak R, Gauravaram P (2017) Blockchain for IoT security and privacy: The case study of a smart home. 2017 IEEE Int Conf Pervasive Comput Commun Work PerCom Work 2017:618–623. https://doi.org/10.1109/PERCOMW.2017.7917634
12. Dorri A, Kanhere SS, Jurdak R, Gauravaram P (2019) LSB: A Lightweight Scalable Blockchain for IoT security and anonymity. J Parallel Distrib Comput 134:180–197. https://doi.org/10.1016/j.jpdc.2019.08.005
13. Fakhri DMutijarsa K (2019) ISESD 2018 - Int Symp Electron Smart Devices Smart Devices Big Data Anal Mach Learn10.1109/ISESD.2018.8605485Fakhri D, Mutijarsa K (2019) Secure IoT communication using blockchain technology. ISESD 2018 - Int Symp Electron Smart Devices Smart Devices Big Data Anal Mach Learn. https://doi.org/10.1109/ISESD.2018.8605485
14. Hassija V, Chamola V, Saxena V et al (2019) A survey on IoT Security: application areas, security threats, and solution architectures. IEEE Access 7:82721–82743. https://doi.org/10.1109/ACCESS.2019.2924045
15. Hossain S, Waheed S, Rahman Z et al (2020) Blockchain for the security of Internet of Things: a smart home use case using Ethereum. https://doi.org/10.35940/ijrte.E6861.018520
16. Hsu HT, Jong GJ, Chen JH, Jhe CG (2019) Improve IoT security system of smart-home by using support vector machine. 2019 IEEE 4th Int Conf Comput Commun Syst ICCCS 2019, 674–677. https://doi.org/10.1109/CCOMS.2019.8821678
17. Huh S, Cho S, Kim S (2017) Managing IoT devices using blockchain platform. Int Conf Adv Commun Technol ICACT 464–467. https://doi.org/10.23919/ICACT.2017.7890132
18. Jeyanthi N, Thandeeswaran R, Global IGI (2017) Security breaches and threat prevention in the Internet of Things. i:276. https://doi.org/10.4018/978-1-5225-2296-6
19. Jones MT Platform emulation with Bochs. http://www.ibm.com/developerworks/library/i-bochs/, January-2011
20. Ma M, He Z, Xu Q, Li XJ (2019) Design and development of smart home sensing supported by blockchain technology. ACM Int Conf Proceeding Ser 525–530. https://doi.org/10.1145/3377170.3377281
21. Makhdoom I, Abolhasan M, Abbas H, Ni W (2019) Blockchain's adoption in IoT: The challenges, and a way forward. J Netw Comput Appl 125:251–279
22. Mohanta BK, Jena D, Panda SS, Sobhanayak S (2019) Blockchain technology: A survey on applications and security privacy Challenges. Internet of Things 8:100107. https://doi.org/10.1016/j.iot.2019.100107
23. Monti M, Rasmussen S (2017) RAIN: a bio-inspired communication and data storage infrastructure. Artif Life 23:552–557. https://doi.org/10.1162/ARTL_a_00247
24. Mtetwa N, Tarwireyi P, Adigun M (2019) Secure the Internet of Things Software Updates with Ethereum Blockchain. Proc – 2019 Int Multidiscip Inf Technol Eng Conf IMITEC 2019:1–6. https://doi.org/10.1109/IMITEC45504.2019.9015865
25. Nguyen DC, Pathirana PN, Ding M, Seneviratne A (2019) Blockchain for 5G and beyond networks: a state of the art survey. J Netw Comput Appl 102693. https://doi.org/10.1016/j.jnca.2020.102693

26. Paunikar VL, Dewalkar VV, Tambekar NS, Dighore RG, Paunikar NO (2020) A user authentication scheme of Iot devices using blockchain-enabled fog nodes. International Journal of All Research Writings 1, 11:19–22
27. Poh G, Sen, Gope P, Ning J (2019) PrivHome: privacy-preserving authenticated communication in smart home environment. IEEE Trans Dependable Secur Comput PP:1–1. https://doi.org/10.1109/tdsc.2019.2914911
28. Reyna A, Martín C, Chen J et al (2018) On blockchain and its integration with IoT. Challenges and opportunities. Futur Gener Comput Syst 88:173–190. https://doi.org/10.1016/j.future.2018.05.046
29. Rifi N, Rachkidi E, Agoulmine NTaher NC (2018) Towards using blockchain technology for IoT data access protection. 2017 IEEE 17th Int Conf Ubiquitous Wirel Broadband, ICUWB 2017 - Proc 2018-Janua: 1–510.1109/ICUWB.2017.8251003Rifi N, Rachkidi E, Agoulmine N, Taher NC (2018) Towards using blockchain technology for IoT data access protection. 2017 IEEE 17th Int Conf Ubiquitous Wirel Broadband, ICUWB 2017 - Proc 2018-Janua:1–5. https://doi.org/10.1109/ICUWB.2017.8251003
30. Selim M, Khwaja H, Ali Y et al (2020) Blockchain for the security of Internet of Things: A smart home use case using Ethereum system and security view project software developement view project. Int J Recent Technol Eng :2277–3878. https://doi.org/10.35940/ijrte.E6861.018520
31. Singh PK, Singh R, Nandi SK, Nandi S (2019) Managing smart home appliances with proof of authority and blockchain. In: Communications in Computer and Information Science. Springer Verlag, Berlin, pp 221–232
32. Tantidham T, Aung YN (2019) Emergency service for smart home system using ethereum blockchain: system and architecture. 2019 IEEE Int Conf Pervasive Comput Commun Work PerCom Work 2019 888–893. https://doi.org/10.1109/PERCOMW.2019.8730816
33. Thakore R, Vaghashiya R, Patel C, Doshi N (2019) Blockchain - based IoT: A survey. Procedia Comput Sci 155:704–709. https://doi.org/10.1016/j.procs.2019.08.101
34. Wohrer M, Zdun U (2018) Smart contracts: Security patterns in the ethereum ecosystem and solidity. 2018 IEEE 1st Int Work Blockchain Oriented Softw Eng IWBOSE 2018 - Proc 2018-Janua:2–8. https://doi.org/10.1109/IWBOSE.2018.8327565
35. Wright CS (2019) Bitcoin: a peer-to-peer electronic cash system. SSRN Electron J. https://doi.org/10.2139/ssrn.3440802
36. Yin C, Zhou B, Yin Z, Wang J (2019) Local privacy protection classification based on human-centric computing. Human Centric Comput Inf Sci 9:1–14. https://doi.org/10.1186/s13673-019-0195-4