Block chain as a secure platform for storing IoT based data

Nikita Raparti[1], Supriya Mokashi[2], Swapnil Kulkarni[3], Himanshu Gadekar[4], Shweta Guja[5]

[1] *Student, Computer Engineering Department, NBN Sinhgad School of Engineering, Maharashtra, India*
[2] *Student, Computer Engineering Department, NBN Sinhgad School of Engineering, Maharashtra, India*
[3] *Student, Computer Engineering Department, NBN Sinhgad School of Engineering, Maharashtra, India*
[4] *Student, Computer Engineering Department, NBN Sinhgad School of Engineering, Maharashtra, India*

ABSTRACT

Block chain, a promising technology is emerging as a boon for many sectors. IoT, at the same time has brought about a tide of innovation with its wide range of applications. However, the problem with IoT is ensuring the security of the sensor based data storage. The possible and feasible solution for this problem can be formulated using Block chain. Block chain has begun to have a significant influence in the Internet of Things by enhancing security, empowering the incorporation of an increasing number of devices into the ecosystem. The improvements in IoT gadget security encourage quicker appropriation of this progressive advancement, and will open up an extensive variety of potential outcomes for endeavors in the days to come. Hereby we aim to propose a new IoT server platform by introducing a block chain. This system can be a radical departure from traditional processes of storing data and it can also remove the bottlenecks associated with the traditional cloud based data storing systems. It aims at retrieving sensitive data from the IoT devices and securely storing as well as maintaining it in the block chain and controlling the access to this data.

Keyword: *Block chain, Ethereum, Smart contract, IPFS, Decentralized storage, Publisher-Subscriber model.*

1. INTRODUCTION

IoT is now taking over the world rapidly, it is estimated that the number of devices connected to the Internet forming the Internet of Things will reach Billions. Despite the benefits of the IoT, the amount of data generated by it often leads to privacy issues. IoT, as it is currently implemented, works on a centralized, client-server based access model in which user data is entrusted with centralized service providers. The clients send requests for the data and the servers provide the required data after communicating with the centralized storage. In most of the scenarios IoT devices and the data they generate can reveal personal information of the users including their behaviors and preferences. This is most likely not appreciated and the users want their data to be kept free from theft. Secure access to this IoT based data is problematic considering security, privacy issues and transparency. This is why in order to develop secure and reliable solutions for storing this data, it requires unprecedented coordination and collaboration between all the pieces of the system. All devices must work together and must be integrated with all other devices. All devices must communicate and interact seamlessly with remote systems and infrastructures in a secure way. Such a solution is possible; however it can be expensive and time consuming. Thus the need of new ideas, new technologies that will converge IoT security towards a decentralized model were felt. Having this huge amount of data, being centralized, and sometimes monitored by one single provider, may create many issues. To introduce privacy for user's data in IoT, the goal is to use Decentralized storage for IoT Data that has privacy built into it by design. Decentralized IoT data management will give users the choice of sharing or selling their sensor data with third party entities without intermediaries. The objective therefore, is to provide a decentralized data access model for IoT, which ensures that user-data is not entrusted to centralized entities or companies, but instead is made the property of the users themselves. Block chains can prove to be crucial in realizing this goal.

2. BACKGROUND

2.1 Block chain

In 2008 Satoshi Nakamoto introduced Bit coin to the world, a completely computerized and decentralized digital currency. With a specific end goal to take care of the twofold spending issue in Bit coin, Block chain- the technology behind crypto currency was presented. It is a shared decentralized dispersed record that is duplicated on all hubs taking part in the framework. The block chain is a peer-to-peer distributed data structure that represents an immutable ledger of transactions. The transactions refer to data exchanges that occur in network. In crypto currency networks, the transactions involve transfer of crypto currency. In a block chain network, the entire ledger is distributed over all the nodes, and every node casts a vote over the validity of every new transaction that is added to the ledger. Since the same copy of the block chain is maintained over all the nodes using peer-to-peer consensus algorithms, there is no central entity entrusted with maintaining transaction records. Therefore, block chains create a trustless environment with accountability built into it.

2.2 Ethereum

Ethereum is the virtual currency developed by Vitalik Buterin, the platform. It is a virtual currency derived from an existing bit coin. While bit coins concentrate on settlement and transaction related systems, Ethereum transparently passes various applications such as contracts, e-mails, electronic voting, as well as transactions and settlement based on the core technology block chain provide extensibility so that it can act on it. As it is based on Block Chain, these will of course be decentralized applications.

2.3 Smart Contract

The first block chain based smart contract is a bit coin script. In this, transactions are automatically executed according to the conditions for creating and sending scripts in OPCODE of source language for bit coin transactions. However, bit coin scripts can't use loops, and there is a limit that can't manage information other than the balance of bit coins. Due to the unique structure of the block chain when allowing loops with bit coin scripts, if an infinite loop occurs on the door during script conditions, the entire network can be stopped. Ethereum is a smart contract specialized block chain platform that has come up to overcome the limitations of these bit coin script systems. Here, a fee is generated every time each line is executed, the limit of the fee on the network is set, and the infinite loop is prevented.

2.4 IPFS-Decentralized System

The Interplanetary File System (IPFS) is a protocol started by Protocol Labs to create a new way to server information on the web. Currently the Internet works off a location based addressing where you go to a URL like medium.com which has an IP of X.X.X.X and then you get served your articles. These URL's are pointed to certain servers around the world. Instead, what IPFS does is it serves information based on what it is as opposed to where it is located. With their routing algorithms, you can choose where you get your content from and you can set your privacy of what peers/nodes you trust to receive your files which is quite interesting.

2.5 Publisher-Subscriber Model

There are many data management and data sharing protocol: Direct block chain access, Server Client access, Publisher Subscriber access. Based on the study of these protocols, we decide to use publisher subscriber mechanism in our solution. Publisher-Subscriber mechanism is very efficient in terms of filtering Data and introducing intermediates to handle electrical power for technologies such as the Block chain. There are two study steps in this model; the first step is to simplify this architecture by working on the block chain related part only, thus the communication between the publisher nodes and subscriber nodes regardless of the users connected to it. The second step is to focus on the end user part, its connection to the subscribers, and the IoT devices and how they are connected to publishers, thus how data is being sent and organized. In the next section, we will go into details of the main components of the system, as well as the proposed solution including the protocol and the smart contracts design.

3. ARCHITECTURE OF PROPOSED SYSTEM

In order to realize a successful implementation of Block chain technology in the Internet of things context, Smart Contracts should be the corner stone of the system. Also, the publisher contract and the subscriber contract must be deployed. After describing the contracts for publishers as well as subscribers, this part describes how the publisher node acts upon receiving new data generated by an IoT device connected to it.
The steps that should occur are:
(1) Storing the data in the off-chain database (IPFS in our case)
(2) Generate the hash pointing to the location in the database, then
(3) Sending that hash to every node on the block chain that is subscribed to this publisher.
The transaction is sent through the block chain client, which is an ethereum client in our case. A block chain database is managed autonomously using a peer-to-peer network and a distributed time stamping server. They are authenticated by mass collaboration powered by collective self-interests. The result is a robust workflow where in the participants are least concerned about the security. It confirms that each unit of value was transferred only once, solving the long-standing problem of double spending. Block chains have been described as a value-exchange protocol. This block chain-based exchange of value can be completed more quickly, more safely and more cheaply than with traditional systems.

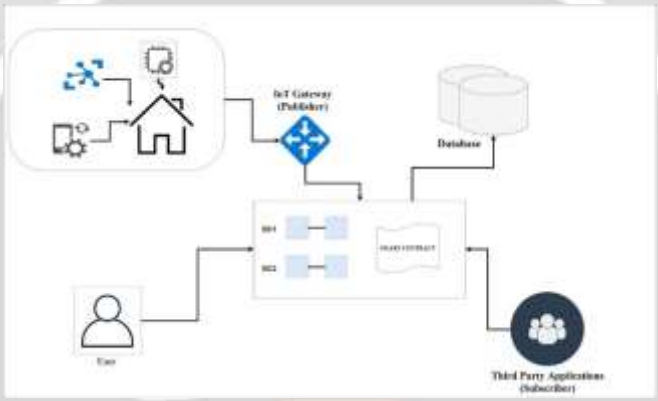

Figure 1- Architecture of Proposed system

4. MANAGING IOT DEVICE WITH ETHEREUM

4.1 User registration and login in contract

This contract is deployed for the new user registration and Login of previously registered user for accessing and managing device rights. User login is compulsory for accessing device data and granting the access to Known third party requesters.

```
function createUser(bytes32  hashValue) public{

    require(userAddress[hashValue] == 0x0);
    address newUser = new main_contract(msg.sender);
    userAddress[hashValue] =newUser;
    //contractAddress.push(newUser);
}

function signIn(bytes32 hashValue) public view returns(address) {
    require(userAddress[hashValue]!=0x0);
    return userAddress[hashValue];

}

// function getContractAddress()public view returns(address[]){
//     return contractAddress;

// }
```

Figure 2: User Registeration

4.2 Device detail contract

This contract is deployed during the addition of the device to Block chain and deletion (prohibition) of the device from the chain. This contract has works by calling the add device function to add device using the details given by the registered user.

```
deviceInfo[] public deviceData;
Request[] public requests;


function main_contract(address creator) public{
    user_address =creator;
}

function addDevice(string name ,string mac)public{
    deviceInfo memory newDevice= deviceInfo({
       name: name,
       mac: mac,
       access: true
    });
```

Figure 3: Device Details

4.3 Request contract

Request contract is deployed during the initialization of request by the third party and after adding a particular Device by the registered user. This contract handles the request access of the data.

```
function removeDeviceAccess(uint index)public restricted{
    deviceData[index].access =false;
}

function requestAccess(uint index)public restricted {
    requests[index].access =true;

}

function sendRequest(string description ,string companyName) public{
    Request memory newRequest = Request({
       description: description,
       sendersAddress: msg.sender,
       companyName: companyName,
       access:false
    });

    requests.push(newRequest);

}
function getSummary() public view returns(uint,uint){
 return(
        deviceData.length,
        requests.length
    );
}
```

Figure 4: Request Contract

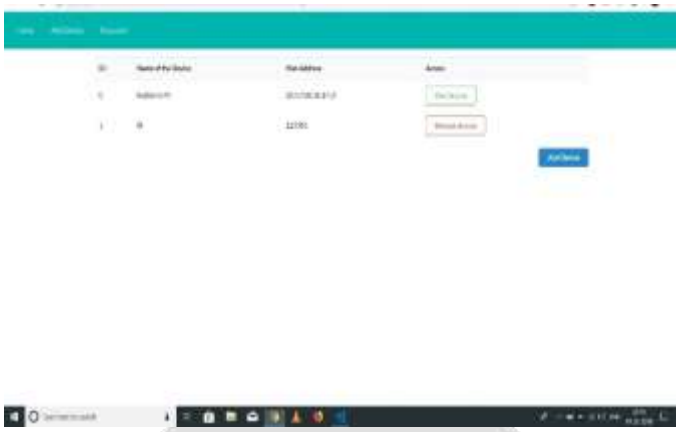

Figure 5: Application screenshot home

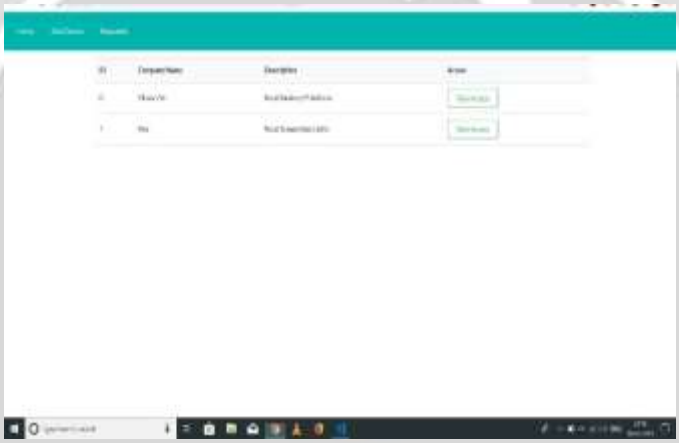

Figure 6: Application screenshot request

5. CONCLUSIONS

Thus network architecture is suitable for providing IoT data privacy via block chains and IPFS. In this architecture, block chain smart contracts perform access control, while providing accountability for both the data owners and the third parties whom the users allow access to. This architecture consists of two existing block chain application platforms. Performance analysis of the block chain platforms provided insights into the architecture's feasibility and further considerations for deploying a usable implementation. This Architecture is implemented using a software stack of block chain smart contracts and peer-to-peer file storage, to give IoT users authority over their data, and to eliminate the need for centralized IoT data management.

6. REFERENCES

[1]. "Block chain: The Invisible Technology That's Changing the World", PC Magazine, Feb 6, 2017, http://www.pcmag.com/article/351486/blockchain-the-invisible-technology-thats-changingthe-world.
[2]. D. Wilson and G. Ateniese, "From Pretty Good to Great: Enhancing PGP using Bit coin and the Block chain," CoRR, vol. abs/1508.04868, 2015.
[3]. G. Zyskind, O. Nathan, and A. Pentland, "Decentralizing Privacy: Using Block chain to Protect Personal Data," in IEEE Symposium on Security and Privacy Workshops.
[4]. Y. Symey, S. Suresh, S. Nurafifah, "Application of Smart Technology for Mobile Patient Appointment System," International Journal of Advanced Trends in Computer Science and Engineering, vol. 2, No. 4, 2013, pp. 74 - 85.
[5]. Ethereum Whitepaper, https://github.com/ethereum/wiki/wiki/White- Paper.