International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211
Volume 11, Issue 1, January-2023, Impact Factor: 7.429, Available online at: www.ijaresm.com

Implementing security in IoT systems via blockchain

Samarth Sharma, Ram Shararn Goyal, Mahesh Potdar, Hari Chodapuneedi, Shaik Ashiq Hussain, Pavan Kumar R, Arya Abhishek,

---*****************---

ABSTRACT

The Internet of Things (IoT) presents numerous benefits to consumers, and has the potential to change the ways that consumers interact with technology in fundamental ways. In the future, the Internet of Things is likely to meld the virtual and physical worlds together in ways that are currently difficult to comprehend. From a security and privacy perspective, the predicted pervasive introduction of sensors and devices into currently intimate spaces – such as the home, the car, and with wearables and ingestible, even the body – poses particular challenges. As physical objects in our everyday lives increasingly detect and share observations about us, consumers will likely continue to want privacy. A client/server model or a centralized model of networking in employed in IoT devices. These devices also use a single gateway to transfer data between them and connect through a cloudserver. This model has lots of shortcomings like high cost of centralized cloud maintenance and network equipmentand the cost will continue to rise with the increase in the number of devices and the volume of data. Single gateway is not very secure as it allows gaining access to a whole IoT network by compromising a single device. To tackle the issue of security, we propose the inclusion of blockchain technology. A Blockchainis a distributed ledger where data is stored across various nodes all over the world, this eliminatesthe single point of failure. This project proposes the plethora of benefits that blockchain technology offers and aims to tackle the security aspect of IoT systems.

Keywords: Blockchain, Internet of Things, Security

INTRODUCTION

The Internet of Things (IoT) is emerging as the third wave in the development of the Internet. The 1990s' Internet wave connected 1 billion users while the 2000s' mobile wave connected another 2 billion. The IoT has the potential to connect 10X as many (28 billion) "things" to the Internet by 2020, ranging from bracelets to cars. Breakthroughs in the cost of sensors, processing power and bandwidth to connect devices are enabling ubiquitous connections right now. Smart products like smart watches and thermostats (Nest) are already gaining traction as stated inGoldman Sachs Global Investment Research's report. The rise of the Internet of Things (IoT) has already sparked concerns about privacy: now security pros are worried that badly configured gadgets might provide a backdoor for hackers looking to break into corporatenetworks.

IoT devices are poised to become more pervasive in our lives than mobile phones and will have access to the most sensitive personal data such as social security numbers and banking information. As the number of are also exponentially multiplied. A couple of security concerns on a single device such as a mobile phone can quickly turn to 50 or 60 concerns when considering multiple IoT devices in an interconnected home or business. In light of the importance of what IoT devices have access to, it's important to understand their security risk. importance of what IoT devices have access to, it's important to understand their security risk.

IoT's Threats

We can list the threats of IoT under three categories; Privacy, Security and Safety. Experts say the security threats of the Internet of Things are broad and potentially even crippling to systems. Since the IoT will have critical infrastructure components, it presents a good target for national and industrial espionage, as well as denial of service and other attacks. Another major area of concern is privacy with the personal information that will potentially reside on networks, also a likely target for cyber criminals.

One thing to keep in mind when evaluating security needs is that the IoT is still very much a work in progress. Many things are connected to the Internet now, and we will see an increase inthis and the advent of contextual data sharing and autonomous machine actions based on that information, the IoT is the allocation of a virtual presence to a physical object, as it develops, these virtual presences will begin to interact and exchange contextual information, [and] the devices will make decisions based on this contextual device. This will lead to very physical threats, around national infrastructure, possessions [for example, cars and homes], environment, power, water and food supply, etc.

As a variety of objects become part of an interconnected environment, we have to consider thatthese devices have lost physical security, as they are going to be located in inhospitable environments, instantly accessible by the individual who is most motivated to tamper with the controls, attackers could potentially intercept, read or change data; they could tamper with control systems and change functionality, all adding to the risk scenarios.

Threats Are Real …

Among the recent examples, one involves researchers who hacked into two cars and wirelessly disabled the brakes, turned the lights off and switched the brakes full on—all beyond the control of the driver. In another case, a luxury yacht was lured off course by researchers hackingthe GPS signal that it was using for navigation. Home control hubs have been found to be vulnerable, allowing attackers to tamper withheating, lighting, power and door locks, other cases involve industrial control systems being hacked via their wireless network and sensors.

We are already seeing hacked TV sets and video cameras [and] child monitors that have raised privacy concerns, and even hacked power meters which to date have been used to steal electricpower, adds Paul Henry, a principal at security consulting firm VNet Security LLC in Boynton Beach, Fla., and a senior instructor at the SANS Institute, a cooperative research and education organization in Bethesda, MD."A recent article spoke of a 'hacked light bulb,'" Henry says. "I canimagine a worm that would compromise large numbers of these Internet-connected devices and amass them in to a botnet of some kind. Remember it is not just the value or power of the device that the bad guy wants; it is the bandwidth it can access and use in a DDoS [distributed denial-of-service] attack."

The biggest concern, Henry says, is that the users of IoT devices will not regard the security of the devices they are connecting as being of great concern. "The issue is that the bandwidth of acompromised device can be used to attack a third party," he says. "Imagine a botnet of 100,000,000 IoT devices all making legitimate Web site requests on your corporate Web site at the same time."

Experts say the IoT will likely create unique and in some cases complex security challenges for organizations. As machines become autonomous they are able to interact with other machines and make decisions which impact upon the physical world. We have seen problems withautomatic trading software, which can get trapped in a loop causing market drops. The systemsmay have failsafe built in, but these are coded by humans who are fallible, especially when theyare writing code that works at the speed [and] frequency that computer programs can operate.

If a power system were hacked and they turned off the lights in an area of the city. No big deal perhaps for many, but for the thousands of people in the subway stations hundreds of feet underground in pitch darkness, the difference is massive. IoT allows the virtual world to interactwith the physical world and that brings big safety issues.

What Can We Do?

While threats will always exist with the IoT as they do with other technology endeavors, it is possible to bolster the security of IoT environments using security tools such as data encryption,strong user authentication, resilient coding and standardized and tested APIs that react in a predictable manner.

Some security tools will need to be applied directly to the connected devices. "The IoT and its cousin BYOD have the same security issues as traditional computers," says Randy Marchany, CISO at Virginia Tech University and the director of Virginia Tech's IT Security Laboratory. "However, IoT devices usually don't have the capability to defend themselves and might have torely on separate devices such as firewalls [and] intrusion detection/prevention systems. Creating a separate network segment is one option." In fact, the lack of security tools on the devices themselves or a lack of timely security updates on the devices is what could make securing the IoT somewhat more difficult from other types of security initiatives, Marchany says. "Physical security is probably more of an issue, since these devices are usually out in the open or in remote locations and anyone can get physical access to it," Marchany says. "Once someone has physical access to the device, the security concerns rise dramatically."

It doesn't help that vendors providing IoT technologies most likely have not designed security into their devices, Marchany says. "In the long term, IT executives should start requiring the vendors to assert [that] their products aren't vulnerable to common attacks such as those listed in the OWASP [Open Web Application Security Project] Top 10 Web Vulnerabilities," he says. IT and security executives should "require vendors to list the vulnerabilities they know exist on their devices as part of the purchase process.

Security needs to be built in as the foundation of IoT systems, with rigorous validity checks, authentication, data verification, and all the data needs to be encrypted. At the application level, software development organizations need to be better at writing code that is stable, resilient and trustworthy, with better code development standards, training, threat analysis andtesting. As systems interact with each other, it's essential to have an agreed interoperability standard, which safe and valid. Without a solid bottom-top structure we will create more threats with every device added to the IoT. What we need is a secure and safe IoT with privacy protected, tough trade off but not impossible.

METHODOLOGY

Methodology Description

Present day IoT systems are cloud-based infrastructures are not completely secure. The following section illustrates a plausible scenario that can occur in an IoT system which can thereby compromise the security of the system.

In developing the project, we are using ARDUINO UNO which is the heart of the project asshown in Fig3.1.1. There are various other sensors and devices which are interfaced to Arduino UNO which includes ESP 8266 WiFi module and DHT 11. Arduino is an open- source electronics platform based on easy-to-use hardware and software which able to read inputs - light on a sensor, a finger on a button and turn it into an output. The DHT11 is a basic, ultra-low-cost digital temperature and humidity sensor. The ESP8266 WiFi Module is a self-contained SOC with integrated TCP/IP protocol stack that can give any microcontroller access to your WiFi network. ESP 8266 and DTH 11 is calibrated to the Arduino. DTH11 being as an input sensor is used to read the temperature of the surrounding and ESP 8266 is used to connect to the internet. Once entire connection is made DTH reads the temperature of the surrounding and sends it to the Arduino. The Arduino sends the temperature value to the cloud using WiFi module ESP 8266 as it is used to connect to the internet.

If any attack such as man-in-the-middle attack takes place then the temperature that was read by the DTH 11 and was send to Arduino will be intercepted by anattacker causing itself to interject in the process and manipulating the data being sent from the Arduino to the cloud via WiFi Module. For example, as shown above DTH 11 reads temperature as 30 degree Celsius from the surrounding and sends it to the Arduino. Arduino passes the temperature value as 30 to the cloud via WiFi Module. However, if the attack takesplace in between then it manipulates the data and sends the false data as 40 degree Celsius tothe cloud. This is how the attack can take place while passing over the data to the cloud. We propose a functional blockchain that is coded using C++ programming language. This model blockchain as proposed by us demonstrates the processes with regards to security nodes in a blockchain network. In this implementation of a blockchain, we include objects that are IoT components like sensors. The blockchain tests the possibility of data tampering dur to an attacker and the same is explained further in section

Algorithms used

Blockchain technology works on hashing as opposed to encryption. Due to this, it is extremely difficult to break, and the hashing algorithm used is SHA256.

SHA256 Hashing Algorithm

Blockchains use the SHA-256 hashing algorithm as their hash function. A cryptographic hash is a 'signature' type of text or data file. The SHA-256 hashing produces a signature thatdiffers from the 256-bit (32-byte) text provided. The hash is not 'encrypted' - it cannot be encrypted back to the original text (it is a cryptographic function of 'one-way', and it is the default size of any source text size).

This makes it appropriate to compare 'quick' versions of texts, as opposed to encrypting text to get the original version. Such applications include hash tables, integrity verification, challenge handshake challenge, digital signatures, etc.

- A ‘handshake challenge’ (or a ‘hash proof authentication challenge’) avoid transferring lostpasswords ‘clearly’ a client can send a hash password online to be verified by the server without the risk of the original password being intercepted.
- The anti-tamper link message hash is real, and the recipient can reply to the message and compare it with the given hash: if it is the same, the message has not been changed; this canbe used to ensure that there is no data loss in the

transfer

- Digital signatures are very involved, but in reality, you can sign the document hash by encrypting it with your private key, producing a digital document signature.

Anyone else can check if you have verified the text by removing the encryption with your public key to get the actual hash again, and then compare it with their text hash.

SHA-256 is one of the following hash functions in SHA-1 (collectively called SHA-2) and is one of the most powerful hash functions available. SHA-256 is not much more complicatedto install than SHA-1, and it has not been disturbed in any way. The 256-bit key makes it a great work for AES partners. Described in the NIST (National Institute of Standards and Technology) standard 'FIPS 180-4'. NIST also provides multiple test vectors to ensure implementation accuracy.

Modelling Design

This paper proposes a method and design that a decentralized one, whereas the existing nature of IoT systems is centralized. With this introduction of a decentralized nature as opposed to a centralized one, the aspect of security is dealt with. Further, a single point failure cannot take place as there are several other nodes present on the network. For the case of data manipulation, in case of a centralized system, it is possible for an attacker to tamper with dataand not get noticed. Whereas in a decentralized system, due to the presence of multiple nodes,if the same scenario were to occur, the blockchain would compare the data of the other nodesand maintain the correct data by correcting itself. Blockchains work on the concept of hashing and not on encryption, thus, there is no possibility of decrypting the data. The hashes generated for every transaction are unique and even the most minute change or tampering ofthe data being transacted will result in a new hash being generated.

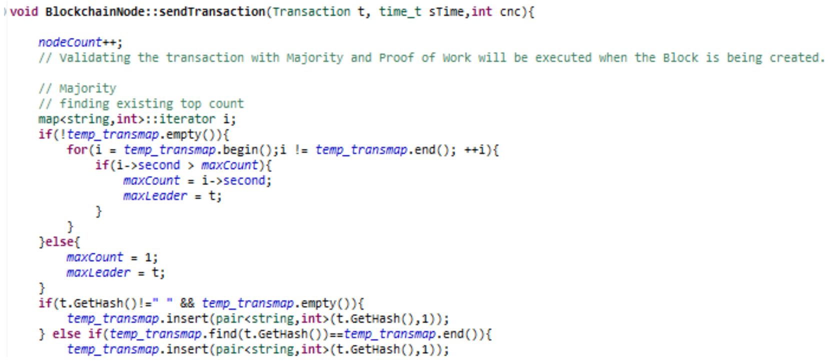
This property of blockchain technology renders it highly secure. Another aspect that is integral to blockchains is that even if a new hash is generated in a node,the blockchain will then correct that node based on the presence and validation of the data of majority of the nodes on the block. For the purpose of this paper, we will be considering a weather monitoring system for our proposed methodology and data manipulations. Accordingly, components such as temperature and humidity sensing via dht11 sensor are considered.

The working of the blockchain technology. First, a transaction is requested; this is the initial step. Once a transaction has been requested. A block is then created that holds data (along with other fields such as timestamp, hash, and previous hash vales). Now this block is sent to all the nodes present in the network for validation. Once validated, the block is successfully added to the chain; and all nodes on the network are now updated with the latest block. This completes the transaction. Due to three essential features- hashing, proof-of-work, and distributed nature of the blockchain which allows for peer-to-peer communication; the blockchain has immense security and is extremely difficult to break.

With the addition blockchain technology, the central cloud server in the above-mentioned system is eliminated and in place of it, a blockchain system is introduced as shown in the data and interactions will pass through this blockchain and get stored on it.

Different aspects in our code fit together. The DHT11 sensor entity sends the temperature and humidity data via a transaction. This the moves into a block when then is a part of a blockchain. This is contained with a Blockchain node. The sensed data is sent to the three closest nodes. The other two remaining nodes do not receive the transactions butdue to the process of synchronization, the other nodes also get updated eventually after all the validation is done. The blocks are all individual

This paper presents the working of a model blockchain that consists of 5 nodes and the transactions are being sent to the three closest nodes. As the scenario of weather monitoring system is considered, the code consists of corresponding class of DHT11 sensor that will send readings of temperature and humidity. These readings are then encapsulated into a "transaction" that are then sent over to the nodes of the blockchain. Here, the three closest nodes have been selected to receive the transactions. Fig. 4.4 shows the main initializations set. The Sensor constructor is set to transmit the sensor readings. The whole blockchain is setas a list and the hashes with their counters are key-value pairs in a mapping. As the nodes, sensors are part of a network, they have IP addressed that are set as shown in Fig. 4.4. The sensor is a separate entity that only sends the data over a transaction to the blockchain.

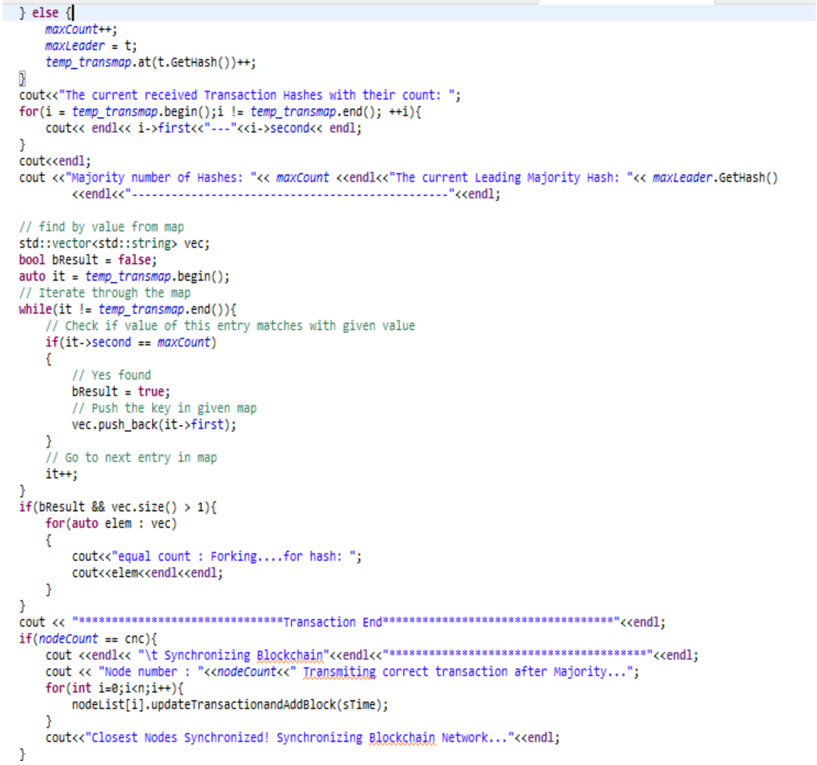


```
void BlockchainNode::sendTransaction(Transaction t, time_t sTime,int cnc){

    nodeCount++;
    // Validating the transaction with Majority and Proof of Work will be executed when the Block is being created.

    // Majority
    // finding existing top count
    map<string,int>::iterator i;
    if(!temp_transmap.empty()){
        for(i = temp_transmap.begin();i != temp_transmap.end(); ++i){
            if(i->second > maxCount){
                maxCount = i->second;
                maxLeader = t;
            }
        }
    }else{
        maxCount = 1;
        maxLeader = t;
    }
    if(t.GetHash()!=" " && temp_transmap.empty()){
        temp_transmap.insert(pair<string,int>(t.GetHash(),1));
    } else if(temp_transmap.find(t.GetHash())==temp_transmap.end()){
        temp_transmap.insert(pair<string,int>(t.GetHash(),1));
```

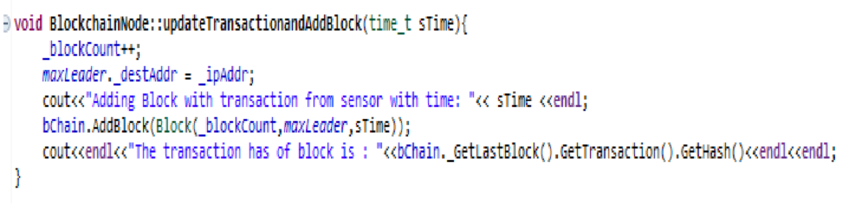
Majority algorithm



```
} else {
    maxCount++;
    maxLeader = t;
    temp_transmap.at(t.GetHash())++;
}
cout<<"The current received Transaction Hashes with their count: ";
for(i = temp_transmap.begin();i != temp_transmap.end(); ++i){
    cout<< endl<< i->first<<"---"<<i->second<< endl;
}
cout<<endl;
cout <<"Majority number of Hashes: "<< maxCount <<endl<<"The current Leading Majority Hash: "<< maxLeader.GetHash()
        <<endl<<"------------------------------------------------"<<endl;

// find by value from map
std::vector<std::string> vec;
bool bResult = false;
auto it = temp_transmap.begin();
// Iterate through the map
while(it != temp_transmap.end()){
    // Check if value of this entry matches with given value
    if(it->second == maxCount)
    {
        // Yes found
        bResult = true;
        // Push the key in given map
        vec.push_back(it->first);
    }
    // Go to next entry in map
    it++;
}
if(bResult && vec.size() > 1){
    for(auto elem : vec)
    {
        cout<<"equal count : Forking....for hash: ";
        cout<<elem<<endl<<endl;
    }
}
cout << "********************************Transaction End*************************************"<<endl;
if(nodeCount == cnc){
    cout <<endl<< "\t Synchronizing Blockchain"<<endl<<"****************************************"<<endl;
    cout << "Node number : "<<nodeCount<<" Transmiting correct transaction after Majority...";
    for(int i=0;i<n;i++){
        nodeList[i].updateTransactionandAddBlock(sTime);
    }
    cout<<"Closest Nodes Synchronized! Synchronizing Blockchain Network..."<<endl;
}
```

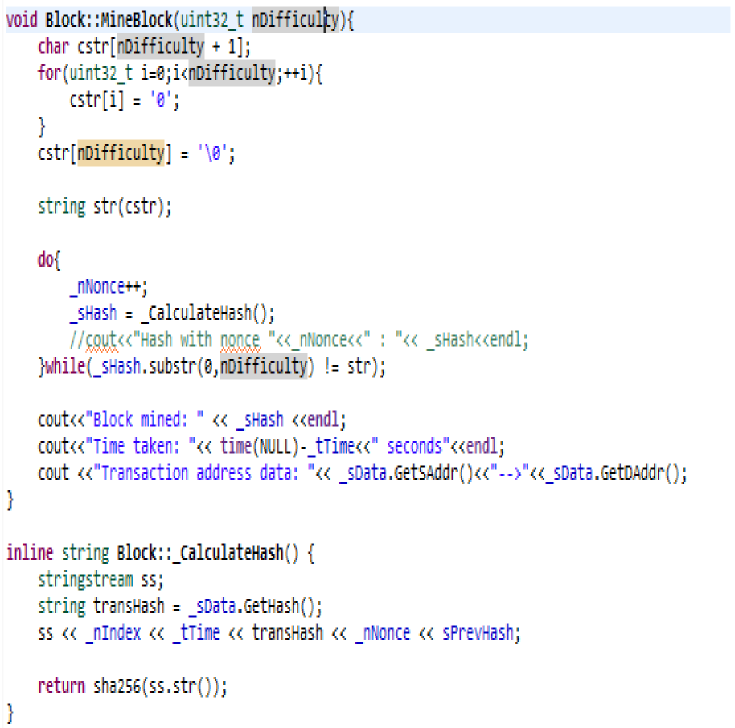
Majority algorithm



```
void BlockchainNode::updateTransactionandAddBlock(time_t sTime){
    _blockCount++;
    maxLeader._destAddr = _ipAddr;
    cout<<"Adding Block with transaction from sensor with time: "<< sTime <<endl;
    bChain.AddBlock(Block(_blockCount,maxLeader,sTime));
    cout<<endl<<"The transaction has of block is : "<<bChain._GetLastBlock().GetTransaction().GetHash()<<endl<<endl;
}
```

Majority algorithm

Fig. 4.5, 4.6 and 4.7 elucidate the majority algorithm. It is due to this that any tampering of data carried out by an attacker can be disregarded. As an attacker will have to follow proof of work mechanism it is seemingly not possible for an attacker to tamper with more than half the transactions at once. Thus, we take into account the majority of transactions with the same hash value and validate this to be the actual data. Section 5 delves more deeply into this concept.



```
void Block::MineBlock(uint32_t nDifficulty){
    char cstr[nDifficulty + 1];
    for(uint32_t i=0;i<nDifficulty;++i){
        cstr[i] = '0';
    }
    cstr[nDifficulty] = '\0';

    string str(cstr);

    do{
        _nNonce++;
        _sHash = _CalculateHash();
        //cout<<"Hash with nonce "<<_nNonce<<" : "<< _sHash<<endl;
    }while(_sHash.substr(0,nDifficulty) != str);

    cout<<"Block mined: " << _sHash <<endl;
    cout<<"Time taken: "<< time(NULL)-_tTime<<" seconds"<<endl;
    cout <<"Transaction address data: "<< _sData.GetSAddr()<<"-->"<<_sData.GetDAddr();
}

inline string Block::_CalculateHash() {
    stringstream ss;
    string transHash = _sData.GetHash();
    ss << _nIndex << _tTime << transHash << _nNonce << sPrevHash;

    return sha256(ss.str());
}
```

Proof of Work

A blockchain has multiple nodes and all the nodes have the same stake of the blockchain. While adding a new block in the blockchain, to differentiate between a proper and maliciousnode, proof of work is used. Proof of work is a consensus algorithm. Here, all the nodes which are competing with each other to add the new block to the blockchain have to do somecomputing work. All of them are given the generated hash by SHA-256.For example if our difficulty is 4, they will try to iterate and calculate more hashes with the same hash and noncewhich is an iterating number. They keep on iterating until they get a randomly generated hashwith all the values which they are passing in and this randomly generated should have its firstfour digits to be zero, since our difficulty level is 4. If these passes, then they are successfullyvalidated and now can add the block. This is useful in detecting the malicious node as if thereare 3 blocks competing to add the block, blockchain will only select the block which did themost computational work and will disregard others because they have done it in less computational work therefore concluding them to be the malicious block. The more computational work tells the blockchain how secure the block is.

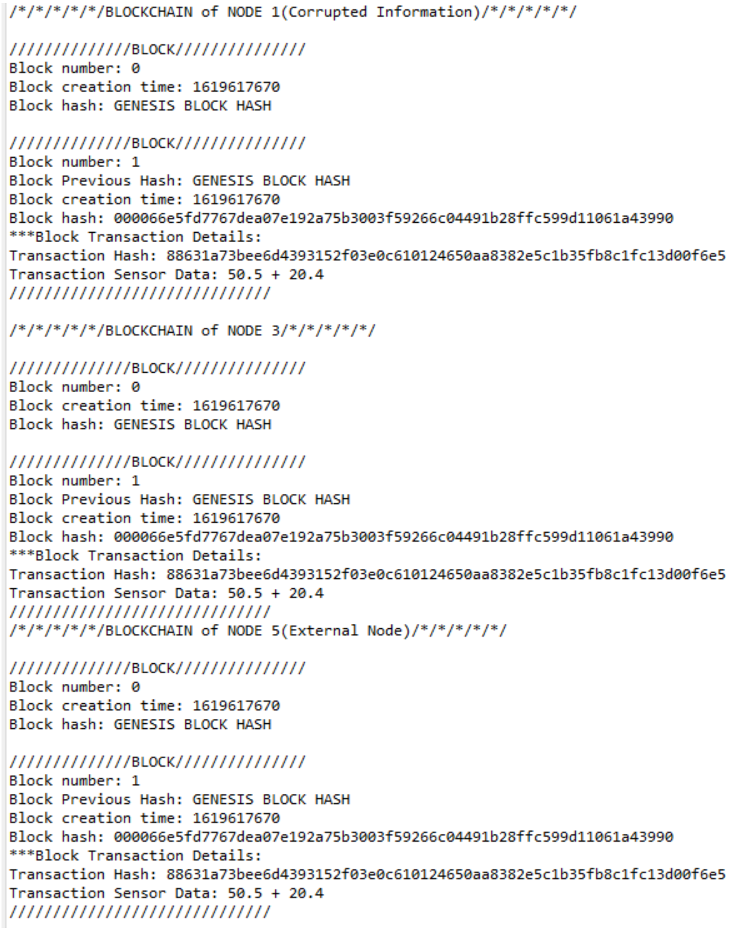
Highlighting the security aspect of the blockchain, the situation where tampering of data or corruption can happen has been considered and illustrated in Section 5 of this paper.

EXPERIMENTS, RESULTS AND DISCUSSION

Case 1: Node 3 is corrupted

Here, the transaction is sent to three closest nodes 1, 2, and 3. While the transactions are being sent to nodes 1, 2 and 3; nodes 4 and 5 do not receive a transaction as it relatively farther away. Now, here, all three nodes receive the transaction at the same time, but the correct data is sent to nodes 1 and 2 while node 3 is receiving corrupted data that may have been altered by a hacker. This is set here by the attacker spoofing as the sensor to the blockchain.

Here, again based on majority, the transaction as received by nodes 2 and 3 are accepted, thesame is updated on node 1. Next, all the nodes in the blockchain are updated with the newlyvalidated transaction as shown in Fig.5.3.3.



```
/*/*/*/*/*/BLOCKCHAIN of NODE 1(Corrupted Information)/*/*/*/*/*/

//////////////BLOCK//////////////
Block number: 0
Block creation time: 1619617670
Block hash: GENESIS BLOCK HASH

//////////////BLOCK//////////////
Block number: 1
Block Previous Hash: GENESIS BLOCK HASH
Block creation time: 1619617670
Block hash: 000066e5fd7767dea07e192a75b3003f59266c04491b28ffc599d11061a43990
***Block Transaction Details:
Transaction Hash: 88631a73bee6d4393152f03e0c610124650aa8382e5c1b35fb8c1fc13d00f6e5
Transaction Sensor Data: 50.5 + 20.4
//////////////////////////////

/*/*/*/*/*/BLOCKCHAIN of NODE 3/*/*/*/*/*/

//////////////BLOCK//////////////
Block number: 0
Block creation time: 1619617670
Block hash: GENESIS BLOCK HASH

//////////////BLOCK//////////////
Block number: 1
Block Previous Hash: GENESIS BLOCK HASH
Block creation time: 1619617670
Block hash: 000066e5fd7767dea07e192a75b3003f59266c04491b28ffc599d11061a43990
***Block Transaction Details:
Transaction Hash: 88631a73bee6d4393152f03e0c610124650aa8382e5c1b35fb8c1fc13d00f6e5
Transaction Sensor Data: 50.5 + 20.4
//////////////////////////////
/*/*/*/*/*/BLOCKCHAIN of NODE 5(External Node)/*/*/*/*/*/

//////////////BLOCK//////////////
Block number: 0
Block creation time: 1619617670
Block hash: GENESIS BLOCK HASH

//////////////BLOCK//////////////
Block number: 1
Block Previous Hash: GENESIS BLOCK HASH
Block creation time: 1619617670
Block hash: 000066e5fd7767dea07e192a75b3003f59266c04491b28ffc599d11061a43990
***Block Transaction Details:
Transaction Hash: 88631a73bee6d4393152f03e0c610124650aa8382e5c1b35fb8c1fc13d00f6e5
Transaction Sensor Data: 50.5 + 20.4
//////////////////////////////
```

Output 3

Fig. 5.3.4 shows the blockchains of nodes 1,3 and 5. It can be seen that the blockchains of these nodes are now identical following the synchronization. In spite of receiving tampered or corrupted data, node 1 now contains the actual data. Also, even though node 5 never received the transaction, upon synchronization it has been updated. The hold the same sensor data and hash values for that transaction. From the above experiments carried out, it can be inferred that the blockchain mechanism is robust, and due to hashing, proof of work and majority algorithm, the blockchain is reliable and secure. From the multiple cases shown, the possibility of an attacker trying to tamper and send altered data is also eliminated.

CONCLUSION

Blockchain and Internet of things are two of the most talked about technologies right now. IoT system right now uses client/server model or a centralized model of networking. They all use a single gateway to transfer data. This has been utilized for a long time, but is not suitable for our current needs. It has high-cost maintenance and the cost will increase as we increase the number of connected devices. The single gateway used is not trustworthy, as it *25* allows gaining access to the whole network by compromising a single device. It has become a big challenge to identify, authenticate and secure these devices. The current model is expensive to scale and manage. In this paper we have proposed blockchain mechanism which is more robust and due to hashing, proof of work and majority algorithm, it is more reliable and secure than the traditional model. Therefore, from the above experiments carried out, itcan be inferred that the proposed method is more reliable and secure than the traditional model. From the multiple cases shown, we can conclude that the possibility of an attacker trying to tamper and send altered data is also eliminated. Blockchain technology can become the future for IoT systems making it very secure and can be used in applications like smart cities.

REFERENCES

[1]. Minoli, D. and Occhiogrosso, B., 2018. Blockchain mechanisms for IoT security. Internet of Things, 1,pp.1-13.

[2]. Hassan, M.U., Rehmani, M.H. and Chen, J., 2019. Privacy preservation in blockchain based IoT systems: Integration issues, prospects, challenges, and future research directions. Future Generation Computer Systems, 97, pp.512-529.

[3]. Sultan, A., Mushtaq, M.A. and Abubakar, M., 2019, March. IOT security issues via blockchain: a review paper. In Proceedings of the 2019 International Conference on Blockchain Technology (pp. 60-65).

[4]. Agrawal, R., Verma, P., Sonanis, R., Goel, U., De, A., Kondaveeti, S.A. and Shekhar, S., 2018, April. Continuous security in IoT using blockchain. In 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 6423-6427). IEEE.

[5]. Dai, H.N., Zheng, Z. and Zhang, Y., 2019. Blockchain for Internet of Things: A survey. IEEE Internet of Things Journal, 6(5), pp.8076-8094.

[6]. Abou-Nassar, E.M., Iliyasu, A.M., El-Kafrawy, P.M., Song, O.Y., Bashir, A.K. and Abd El- Latif, A.A., 2020. DITrust chain: towards blockchain-based trust models for sustainable healthcare IoT systems. IEEE Access, 8, pp.111223-111238.

[7]. Hang, L. and Kim, D.H., 2019. Design and implementation of an integrated iot blockchain platform for sensing data integrity. Sensors, 19(10), p.2228.

[8]. Andersen, M.P., Kolb, J., Chen, K., Fierro, G., Culler, D.E. and Popa, R.A., 2017. Wave: A decentralized authorization system for iot via blockchain smart contracts. University of California at Berkeley, Tech. Rep.

[9]. Fakhri, D. and Mutijarsa, K., 2018, October. Secure IoT communication using blockchain technology. In 2018 International Symposium on Electronics and Smart Devices (ISESD) (pp. 1-6). IEEE.

[10]. Alotaibi, B., 2019. Utilizing blockchain to overcome cyber security concerns in the internet of things: A review. IEEE Sensors Journal, 19(23), pp.10953-10971.

[11]. Huckle, S., Bhattacharya, R., White, M. and Beloff, N., 2016. Internet of things, blockchain and shared economy applications. Procedia computer science, 98, pp.461-466.

[12]. Sicari, S., Rizzardi, A., Grieco, L.A. and Coen-Porisini, A., 2015. Security, privacy and trust in Internet of Things: The road ahead. Computer networks, 76, pp.146-164.

[13]. Christidis, K. and Devetsikiotis, M., 2016. Blockchains and smart contracts for the internet of things. Ieee Access, 4, pp.2292-2303.

[14]. Roman, R., Zhou, J. and Lopez, J., 2013. On the features and challenges of security and privacy in distributed internet of things. Computer Networks, 57(10), pp.2266-2279.

[15]. Ali, M.S., Dolui, K. and Antonelli, F., 2017, October. IoT data privacy via blockchains and IPFS. In Proceedings of the seventh international conference on the internet of things (pp. 1- 7).