A NodeJs Approach in Blockchain Technology

Dr. Archana Sahai
Asst. Professor
Amity University,Uttar Pradesh,
Lucknow, asahai@amity.edu

Dr. Rajiv Pandey
Asst. Professor, Senior Member IEEE
Amity University,Uttar Pradesh
Lucknow, rpandey@lko.amity.edu

ABSTRACT

A blockchain is fundamentally a distributed database of records or public ledger of all transactions ordigital events that have been executed and shared among all joint or contributing parties. Each transaction inthe public ledger is confirmed by agreement of a majority of the participants in the system. Oncethe information is arrived it can never be deleted or erased. This Distributed ledger approach of Blockchain has guaranteed immutability of data. Blockchain ensures high levelof security as the transactions which take place are entirelyanonymous.The chain starts from the genesis block and keeps adding blocks starting from the genesis block based on records of transactions over a given time frame. The hash of each transaction is further represented and many such transaction after due consensus are included in the block of the chain. The Merkel tree approach enables the mathematical resultant of each transaction into the block. This paper explores the implementation aspects of the Blockchain using Node Js where by the Time stamp and hash sequences are demonstrated and relevance brought into context.

Keywords- Blockchain, Immutability, Data. Node JS

I. INTRODUCTION

Immutability means that something is unalterable over time or remains fixed once data has been written to a blockchain. No one, including a system administrator, can change it once it is written. This is beneficial for the purpose of audit. The data provider can easily prove that the data is not altered,and the recipient of data can be sure that the data hasn't been altered. These are beneficial for databases of financial transactions.

Immutability has to be maintained, it is not automatic. However there is no control mechanism making the data immutable. With a private database, an end-user may have read-only access. In which they will be able to view the contents but cannot change the contents of a row in that database. However, somebody with higher privileged access like a systems administrator may have the right to change the data. In such caseswhere administrator can change the database, logs may be stored on another system which is owned and managed by someone else. These organizational systems are put in place to prevent that individual from making the changes.

The quantity of digital data has reached astounding amounts universally. A survey report claims that 90% of the world's data was generated over the past 2 years (Brandtzæg, 2013). In the age of Big Data

and Artificial Intelligence, this data is regularly being collected, scrutinized and altered, thus raising serious data breaches. Centralized organizationsof public and private sectors collect enormous amount of sensitive personal data and information. Data as we know is an intimate part of global headlines presently.The most famous being the Facebook-Cambridge Analytica Scam (Solon, 2018). The scam generated a global chaos with people doubting and questioning the amount of control and confidentiality they have on their sensitive or personal information. WannaCry ransomware is anotherwell-known incident that was covered widely by the public media. It has infected millions of computers globally (Jones, 2017). Cybercriminals have becomemore improved and daringday by day.This upgrading and perfection in cybercriminals iscreatingmoreissuesfor people and organizations.Now a days, data has become one of the most valued assets of an economy (Economist, 2017) and its safety is anessential requirement as it was never before.

This paper contains seven sections, Section I. "Introduction" describes the organization of the paper, Section II, "Blockchain" section deliberates features and presents a brief analysis of the same. Section III. "Security in Blockchain highlights the relevance of Blockchain and its application in the problem's realm", Section-IV. "NodeJs Relevance to Blockchains" section explores the benefits of using NodeJs as the platform for Blockchain implementation. Section- V. "Blockchain Security Advantages" section of the paper demonstrates the implementation of the solution proposed via a NodeJs approach.

.

II. BLOCKCHAIN

"*Blockchain is a shared, trusted, public ledger of transactions that everyone can inspect but which no single user controls, it is a distributed database that maintains a continuously growing list of transactions data records, cryptographically secured from tampering and revision."*.[1]

Theblockchain technology consist of following key characteristics like:

a. Decentralization
b. Persistency
c. Anonymity
d. Auditability.

Withthese traits, blockchain can greatly save the cost and improve the efficiency[11].

Working of a Blockchain:

- Each node receives a replica of all transactions.
- All nodes intercept fresh transactions into block.
- Every node seeks to evolve a consensus for its block.
- If a node encounters a consensus, it transmits it to every node on the network.
- All transactions are evaluated for their validity and subsequently committed. (Fig 7)
- Upon validation of the Hash values the Nodes render their acceptance by appending it as the next block.

III. BLOCKCHAIN SECURITY

Blockchain uses the notion of distributed ledger to enhance security and privacy. Implementing blockchain can offer higher security as compared to storageof all data in a central repository.This way damage from attacks on a database can be prevented.Furthermore, due to the

openness characteristic of blockchain, it provides transparency in data whenapplied to an area demanding the disclosure of data [12]. Due to thesestrong points, it can be utilized in miscellaneousfields including the financial sector and the Internet of Things (IoT) environment and its applicationsare estimated to increase over time [13–17].

Third parties and Central Organizations validate transactions through their own servers with only a small knowledge of legitimacy but in a Blockchain environment, a P2P network of nodes functioning on the same Blockchain protocol do all the legitimate work and through consensus or Majority vote.

Blockchains are supposed to be decentralized and protected on a P2P network. Hashing is implementedfor this. Hashing permits to input data of any size and always get in return a string of a determined length only. This feature makes it very easy to work with data since the same input will always yield in return the same output. To get the same input with an output using a hash is nearly impossible. Hence increasing the security.

Another method Mining is used to make blockchains more secure, especially in the case of cryptocurrency. It allows implementing a proof of work protocol, which checks the validity of the block mined and only validates once anagreement (consensus)has reached by the nodes on the P2P network. The flow of blocks into the chain is also controlled by mining. Thereby increasing the difficulty level will in turn increase the time needed to effectively mine a block. As we will not be using a P2P network for our blockchain, the mining aspect will simply affect the time needed to add a block to the chain, no consensus needed.

Consensus uses Smart Contracts to maintain legitimacy.

Consensus mechanism or algorithms are primarily:

- Proof of Work (PoW)
- Proof of Stake (PoS)
- Proof of Elapsed Time (PoET)

Therefore, pre-defining consensus rules for approving transactions on peer to peer network enforces legitimate transactions by all the participants in the network making the Blockchain immutable.

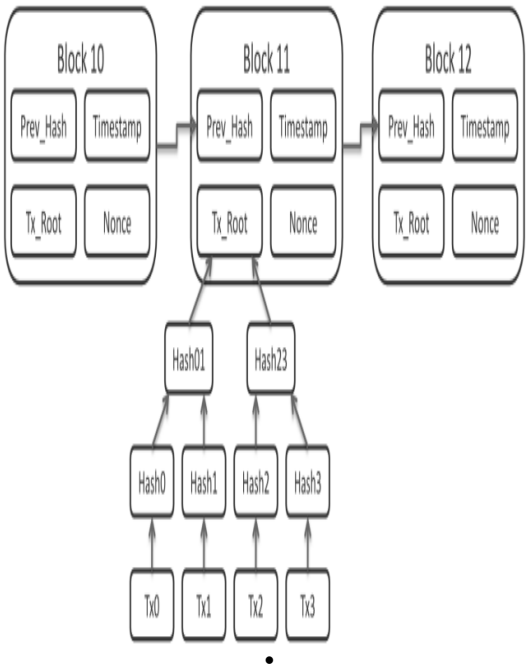

Fig 1: Bitcoin block data[licensed under Creative Commons Attribution-Share-Alike 3.0 Unported, retrieved from Wikipedia]

Fig 1 clearly highights the various components of a block namely,

- Previous Hash: It is the hash value associated with the previous block.
- Timestamp: It ensures immutability via the clause of stating that the block existed at the given time T.
- Tx_Root: It is the Merkel tree of all the transactions in the block.
- Nonce: It is the actual data.

The above components are elucidated in detail in the Node JS code snippet.

IV. NODE JS RELEVANCE TO BLOCKCHAINS

"As an asynchronous event driven JavaScript runtime, Node is designed to build scalable network applications" [https://nodejs.org/en/about/] NodeJs is a server side scripting platform built atop Google Chrome's V8 Javascript engine. It can be used to build fast and scalable network applications. NodeJs provides a rich library of various Javascript modules. Some of these modules have been specifically designed for Blockchain development. This paper in particular utilizes the crypto-js module to implement a Blockchain.

V. BENEFITS OF BLOCKCHAIN SECURITY

Blockchain represents anessential technology thatsupports users to maintain a collective, reliable and a decentralized database. This is a scattered database that maintains a continuously growing list of records that is cryptographically secured from any sort of malicious manipulations (Shermin Voshmgir, 2017). A typical Blockchain system contains a series of blocks which are connected in a specific order. The Blockchain derives its name from its chained representation. The users in the system, are commonly known as nodes. These nodes perform the job of authenticating, validating and storing the data in the form of blocks. The Proof of Work (PoW) algorithm is a consensus algorithm that is used to confirm transactions and add new blocks to the chain.

Each Block contains a hash value of the previous block generated through cryptographic function, a timestamp and transaction related information (Telegraph, n.d.). This cyclic behavior ensures the integrity of the previous block, all the way back to the original genesis block (Nirupama Devi Bhaskar, 2015).

The various advantages of Blockchain from the security perspective are given below.

- Privacy Protection:

Blockchain implements Peer to Peer networking system which eradicates the necessity of a centralized database for storageof confidential information. This removes centralized points that a hacker might target tohamper or steal valuable information. Thus, Blockchain is more robust as it does not have a central point of failure as compared to centralized networking systems. Blockchains uses Asymmetric Cryptography where each user has 2 keys:

1. A public key that is visible to everyone on the network and is used to encrypt messages/transactions for the user.
2. A private key that can only be used to decrypt the message encrypted via the user's public key.

A user's public key has no relation to his/her public address hence computing a user's private key from his/her public key is not possible in any case. This way anonymity and privacy of the user is maintained in Blockchain.

- Crash Recovery:

Data on a Blockchain is not kept at a central location as in normal databases instead it is distributed among peer nodes.Each user on the Blockchain has the equal right to create and maintain a completereplica of the data. Although this results in redundancy of data. This replica and redundancy immenselyincrease the reliability and fault tolerance of the network. Even if the nodes are attacked or compromised, it will not cause damage to

rest of the network and data will not be lost.

- Immutability:

Data once written in blockchain then it cannot be altered. Blockchain has a distinctive data writing mechanism which stops the alteration of data in a block. This mechanism involves the generation of a timestamp the

instant a new record is created. (B. Gipp, 2015) Updation of data is forbidden. Even recording of a new transaction is decided using anagreement mechanism which normallyneeds the mutual agreement of more than 50% of the users of the network.

The snippets that follow elucidate how NodeJs can be used in Blockchain:

```
class Block
{
  constructor(index, timestamp, data, previousHash = '')
  {
    this.index = index;
    this.timestamp = timestamp;
    this.data = data;
    this.previousHash = previousHash;
    this.hash = this.calculateHash();
  }

  calculateHash()
  {
    return SHA256(this.index+this.previousHash+this.timestamp+JSON.stringify(this.data)).toString();
  }
}
```

Fig 2: Code representing a block in the Blockchain

This class represents a block in the Blockchain. It has 2 member functions namely constructor and calculate hash. Constructor has index, timestamp, data and previous hash as parameters. Index represents the address of the block. The timestamp contains information related to when the block was created. Data can contain any information a user wishes to secure by uploading it on the Blockchain. The previous hash parameter stores the hash value of the previous block except in the case of genesis block where the hash value is NULL. The calculate hash function calculates the hash value for the current block using SHA256 encryption module.

```
createGenesisBlock()
{
  return new Block(0,"01/01/2017","Genisis","0000")
}
```

Fig 3: Code for creating a chain of blocks.

```
getLatestBlock()
{
  return this.chain[this.chain.length-1];
}
```

Fig 4: Code to get the latest block in the Blockchain.

```
37  chainIsValid()
38  {
39    for(var i=0;i<this.chain.length;i++)
40    {
41      if(this.chain[i].hash !== this.chain[i].calculateHash())
42        return false;
43
44      if(i > 0 && this.chain[i].previousHash !== this.chain[i-1].hash)
45        return false;
46    }
47    return true;
48  }
49
```

Fig 5: Chain Validity (Immutability)

The code snippet in fig 5 demonstrates the consensus aspect of Blockchains and distributed ledgers, this code shall be executed to verify the validity of the block, only if found valid will it be added to the chain. The Consensus algo of either POW or POS shall of course be embedded for detailed implementation of the blockchain

```
addBlock(newBlock)
{
  if(this.chainIsValid())
  {
    newBlock.previousHash = this.getLatestBlock().hash;
    newBlock.hash = newBlock.calculateHash();
    this.chain.push(newBlock);
    console.log("Chain is valid new block added.");
  }
}
```

Fig 6: Adding a New Block


```
Chain is valid new block added.
{
   "chain": [
      {
         "index": 0,
         "timestamp": 1532999648,
         "data": "Genisis",
         "previousHash": "0000",
         "hash": "a8667a19a3c12fa07764e76f4476e1cb2e8929616378bf1fc7a7453c61f
782b8"
      },
      {
         "index": 1,
         "timestamp": 1532999648,
         "data": {
            "amount": 4
         },
         "previousHash": "a8667a19a3c12fa07764e76f4476e1cb2e8929616378bf1fc7a
7453c61f782b8",
         "hash": "1e7bd411e0228ec65e44f3d996f385ec5a7a6c4e5bfc7d10f0fc26107fa
64da5"
      },
      {
         "index": 2,
         "timestamp": 1532999648,
         "data": {
            "amount": 14
         },
         "previousHash": "1e7bd411e0228ec65e44f3d996f385ec5a7a6c4e5bfc7d10f0f
c26107fa64da5",
         "hash": "f0abaf20fe15b912521349e8ef9f487548348fb46b8387816f7918de792
701cd"
      },
```


Fig 7: Execution of code in CMD

The above figure illustrates 2 blocks in a Blockchain. The first block, namely Genesis block has an index value of '0' implying that it is the first in the chain of blocks. The previous hash of the Genesis block is "0000" further asserting the fact that it is indeed the first block as the previous hash cannot be zero except in the case of the first block in the Blockchain. The Genesis block also contains a hash value that can be verified by everyone present on the Blockchain. The next block has an "index" value equal to '1' implying that it is the second block in the chain. Its previous hash parameter contains the exact same hash value of the previous (Genesis) block.

VI. CONCLUSION

This paper explores the architecture of a typical blockchain whereby the components of a block have been explained with intent of implementing the same through the Node JS approach. The immutability feature of a blockchain is the key factor for suitability in all business environments where security and privacy is the foremost criteria.

The future scope of extendibility of the said NodeJS Applications shall demonstrate the integration with DAPPs and other web applications.The paper presents the demonstrative approach of a Blockchain and not a complete code by itself. The code needs to be extended for complete implementation of the blockchain and is beyond the scope of this paper.

VII. REFERENCES

1. Shermin Voshmgir, . . (2017, September 30). https://blockchainhub.net/blockchain-intro.

2. B. Gipp, N. M. (2015),"Decentralized Trusted Timestamping using the Crypto Currency Bitcoin",iConference. Newport Beach, CA.

3. Brandtzæg, P. B. (2013, May 22). "Big Data, for better or worse: 90% of world's data generated over last two years".

https://www.sciencedaily.com/releases/2013/05/130522085217.htm

4. Economist, T. (2017, May 6). https://www.economist.com/leaders/2017/05/06/the-worlds-most-valuable-resource-is-no-longer-oil-but-data

5. Jones, S. (2017, May 14). "Global alert to prepare for fresh cyber-attacks". https://www.ft.com/content/bb4dda38-389f-11e7-821a-6027b8a20f23

6. Nirupama Devi Bhaskar, D. L. (2015). "Handbook of Digital Currency. Elsevier".

7. Solon, O. (2018, April 4). https://www.theguardian.com/technology/2018/apr/04/facebook-cambridge-analytica-user-data-latest-more-than-thought

8. Tapscott, D. &. (2016). "BLOCKCHAIN REVOLUTION: How the Technology Behind Bitcoin is Changing Money, Business, and the World".

9. Telegraph, C. (n.d.). "How Blockchain Technology Works. Guide for Beginners". https://cointelegraph.com/bitcoin-for-beginners/how-blockchain-technology-works-guide-for-beginners#nodes

10. https://blockgeeks.com/guides/what-is-blockchain-technology/

11. Zibin Zheng1, Shaoan Xie1, Hongning Dai2, Xiangping Chen4, and Huaimin Wang3, "An Overview of Blockchain Technology:Architecture, Consensus, and Future Trends", IEEE 6th International Congress on Big Data, 2017.

12. Park, Jin & Park, Jong. (2017). "Blockchain Security in Cloud Computing: Use Cases, Challenges, and Solutions", Symmetry. 9. 164. 10.3390/sym9080164.

13. Beikverdi, A.; JooSeok, S., June 2015, "Trend of centralization in Bitcoin's distributed network". In Proceedings of the 2015 16th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), Takamatsu, Japan, 1–3.

14. Bonneau, J.; Miller, A.; Clark, J.; Narayanan, A.; Kroll, J.A.; Felten, E.W. Sok, 17–21 May 2015, " Research perspectives and challenges for bitcoin and cryptocurrencies". In Proceedings of the 2015 IEEE Symposium on Security and Privacy (SP), San Jose, CA, USA,.

15. Christidis, K.; Michael, D., 2016,"Blockchains and Smart Contracts for the Internet of Things", IEEE Access 2016, 4, 2292–2303.

16. Huang, H.; Chen, X.; Wu, Q.; Huang, X.; Shen, J., 2016, "Bitcoin-based fair payments for outsourcing computations of fog devices. Future Gener". Comput. Syst. 2016.

17. Huh, S.; Sangrae, C.; Soohyung, K., 2017, "Managing IoT devices using blockchain platform", In Proceedings of the2017 19th International Conference on Advanced Communication Technology (ICACT), Bongpyeong, Korea,19–22 February 2017.