Design, Applying and Analysis of BlockChain System in e-Biddind

Bachelor Thesis

By

The Bot

A thesis submitted to

IIIT Kalyani

for the partial fulfillment of the degree of

Bachelors of Engineering in Computer Science

in

Department of Computer Science and Information Technology

December, 2018

Certificate

This is to certify that the thesis entitled "**Design, Importing and Analysis of BlockChain Technology in e-Bidding**" being submitted by **The Bot**, an undergraduate student (ID 00000121) in the Department of Computer Science and Information Technology, Indian Institute of Information Technology Kalyani, Nodia, 741235, India, for the award of **Bachelors of Technology** in **Computer Science & Engineering**, is an original research work carried by him under my supervision and guidance. The thesis has fulfilled all the requirements as par the regulation of **IIIT Kalyani** and in my opinion, has reached the standards needed for submission. The works, techniques and the results presented have not been submitted to any other university or Institute for the award of any other degree or diploma.

(**Dr. Imon Mukherjee, Ph.D**)
Assistant Professor
Department of Computer Science and Information Systems
Indian Institute of Information Technology Kalyani
IIIT Kalyani Campus, West Bengal 741235, India. December 2018

To my beloved parents who have supported me and prayed for my success throughout my life.

Acknowledgments

First of all, I would like to take this opportunity to thank my supervisor Dr. Imon Mukherjee without whose effort this thesis would not have been possible. I am so grateful to him for working tirelessly after me, answering my doubts whenever and wherever possible. I am most grateful to Department of Computer Science and Information Technology, Indian Institute of Information Technology Kalyani, West Bengal, 741235, India, for providing me this wonderful opportunity to complete my bachelor thesis. I would like to thank my friends for providing me with help as and when required. I would like to thank Anuj Pathak for being a great motivator and a great friend.

And last but the biggest of all, I want to thank my parents, for always believing in me and letting me do what I wanted, but keeping a continuous check that I never wandered off the track from my goal.

The Bot
Id. No.: 00000121
December, 2018

Abstract

A purely peer-to-peer version of online data would allow online transactions to be sent directly from one node to another without going through a central authority. Digital signatures provide part of the solution. The network timestamps transactions by hashing them into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work. The longest chain not only serves as proof of the sequence of events witnessed, but proof that it came from the largest pool of CPU power. As long as a majority of CPU power is controlled by nodes that are not cooperating to attack the network, they'll generate the longest chain and outpace attackers. The network itself requires minimal structure. Messages are broadcast on a best effort basis, and nodes can leave and rejoin the network at will, accepting the longest proof-of-work chain as proof of what happened while they were gone.

Keywords: Peer-to-Peer network, Proof-of-work, Proof-of-Stake, Distributed Ledger (Hyper-Ledger), Concensus Mechanism, Hash Function, No duplicate Entry.

Contents

Chapter 1

Introduction

This chapter presents the introduction of the thesis that includes the brief description of BlockChain, and the adopted approach to address the problems. This chapter also presents the scope of this thesis and the contributions of the thesis.

1.1 Block-Chain

A Block-Chain can be broadly described as a peer-to-peer network of nodes that makes a collaborative effort in sensing certain specified data around its periphery and thereby controls the surrounding environment. In BlockChains, each node consists of processing capability, it may contain multiple types of memory like program, data and memories, having a Web-Service transceiver, Client-Server processors, and a power source. The nodes communicate with each other using web-services and self-organized.

1.2 Applications of BCTs

Block-Chain Technology provides one major advantage over conventional centralized database system: immunity from unexpected data changes or Hacks, which gives rise to numerous applications. Some of them include

- Crypto-currency: Creating and transferring digital money, Data Mining.
- Military applications: Secure and verified records of Every Military Events and documents.
- Structural health Monitoring
- e-Biding Systems

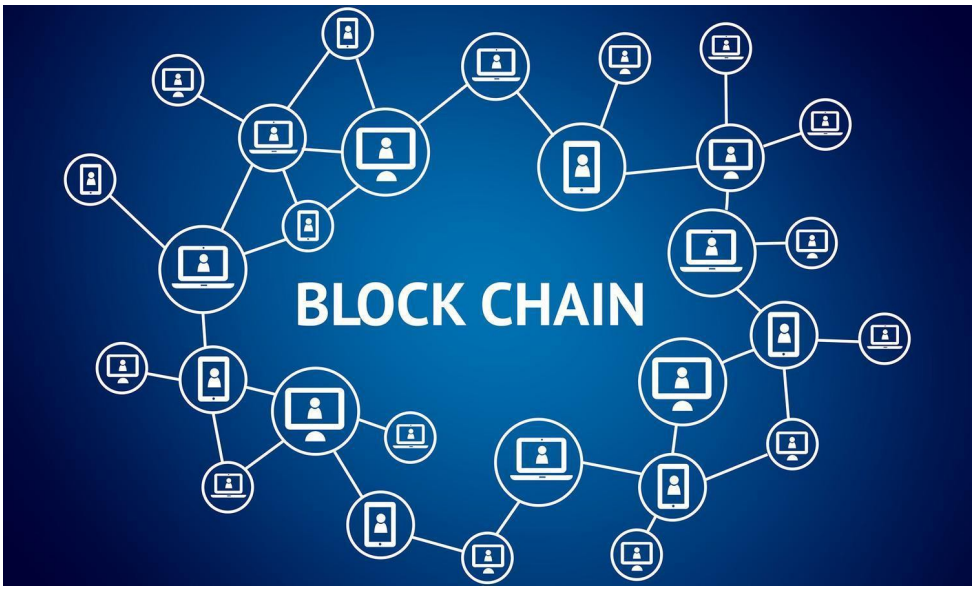

Figure 1.1: Overview of Block-Chain Technology (BCTs).

- Election System or e-Voting Systems
- Selling Records and other Commercial Applications
- Music Copyright Verification System

1.3 Security and Integrity in BCTs

As the data is not sitting on a single data server so there is no security issue for Server Hacking. And also the hash-Chain with Cryptography makes it near to impossible to figure out or change previous data block in the blockchain. Before adding any data block with the help of consensus mechanism the blocks are verified with the digital signature of the node and some solution of nonce.

1.4 Scope and Adopted Approach

1.5 Thesis Contribution

The main contributions of the thesis includes

1. Proposes an user anonymity-preserving algorithm to be a part of e-Biding.
2. Formally analyzes the security of the newly designed protocol as well as its performance.

3. The scheme, as compared to the existing schemes, not only authenticates the users but, also establishes a session key between the user and the System after successful mutual authentication.

4. The scheme provides many security and robustness features of user authentication and Block Processing scheme for BCTs.

1.6 Roadmap of the Thesis

The structure of the thesis is as follows:

1. The Chapter 1 is an introductory part which discusses the scope of the thesis, about the contribution of this thesis and the motivation for writing it.

2. The Chapter 2 provides the background of BCTs, applications of BCTs, security aspects of BCTs.

3. The Chapter 3 discusses in details the basis of ECC, some definitions for security and some mathematically intractable problems.

4. The Chapter 4 introduces the proposed authentication framework after highlighting the motivations behind this work.

5. The implementation of Chapter 5, where an informal implementation of the proposed protocol has been discussed.

6. The Chapter 6 comprises of the conclusion and further work of the Project in future.

Chapter 2

Background

Commerce on the Internet has come to rely almost exclusively on financial institutions serving as trusted third parties to process electronic payments. While the system works well enough for most transactions, it still suffers from the inherent weaknesses of the trust based model. Completely non-reversible transactions are not really possible, since financial institutions cannot avoid mediating disputes. The cost of mediation increases transaction costs, limiting the minimum practical transaction size and cutting off the possibility for small casual transactions, and there is a broader cost in the loss of ability to make non-reversible payments for non-reversible services. With the possibility of reversal, the need for trust spreads. Merchants must be wary of their customers, hassling them for more information than they would otherwise need. A certain percentage of fraud is accepted as unavoidable. These costs and payment uncertainties can be avoided in person by using physical currency, but no mechanism exists to make payments over a communications channel without a trusted party.

What is needed is an electronic payment system based on cryptographic proof instead of trust, allowing any two willing parties to transact directly with each other without the need for a trusted third party. Transactions that are computationally impractical to reverse would protect sellers from fraud, and routine escrow mechanisms could easily be implemented to protect buyers. In this paper, we propose a solution to the double-spending problem using a peer-to-peer distributed timestamp server to generate computational proof of the chronological order of transactions. The system is secure as long as honest nodes collectively control more CPU power than any cooperating group of attacker nodes.

Chapter 3

Details

The process underlying the block-chain technology are the folowing:

- Creating Account and Be a Part of the Network with an hash id
- Creating own Data block
- Request the BCT System to add it by sending it to the open Network in secure or en-Crypted way
- The system then send the block to other nodes for verification
- The other nodes can verify it with solving some nonce and report to system
- After being verified the System will add the new block to the chain and update it to every peer's copy of the hyper-ledger
- Data mining is useful when to check again and again for the integrity of data and the performance of the system

The following is the overview of the BitCoin implemention using concepts of blockchain,

3.1 Transaction

We define an electronic coin as a chain of digital signatures. Each owner transfers the coin to the next by digitally signing a hash of the previous transaction and the public key of the next owner and adding these to the end of the coin. A payee can verify the signatures to verify the chain of ownership.

3.2 Timestamp

The solution we propose begins with a timestamp server. A timestamp server works by taking a hash of a block of items to be timestamped and widely publishing the hash, such as in a newspaper or Usenet post [2-5]. The timestamp proves that the data must have existed at the time, obviously, in order to get into the hash. Each timestamp includes the previous timestamp in its hash, forming a chain, with each additional timestamp reinforcing the ones before it.

3.3 Proof of Work

To implement a distributed timestamp server on a peer-to-peer basis, we will need to use a proof- of-work system similar to Adam Back's Hashcash [6], rather than newspaper or Usenet posts. The proof-of-work involves scanning for a value that when hashed, such as with SHA-256, the hash begins with a number of zero bits. The average work required is exponential in the number of zero bits required and can be verified by executing a single hash. For our timestamp network, we implement the proof-of-work by incrementing a nonce in the block until a value is found that gives the block's hash the required zero bits. Once the CPU effort has been expended to make it satisfy the proof-of-work, the block cannot be changed without redoing the work. As later blocks are chained after it, the work to change the block would include redoing all the blocks after it.

3.4 Network

The steps to run the network are as follows:

1. New transactions are broadcast to all nodes.
2. Each node collects new transactions into a block.
3. Each node works on finding a difficult proof-of-work for its block.
4. When a node finds a proof-of-work, it broadcasts the block to all nodes.
5. Nodes accept the block only if all transactions in it are valid and not already spent.
6. Nodes express their acceptance of the block by working on creating the next block in the chain, using the hash of the accepted block as the previous hash.

3.5 Incentive

By convention, the first transaction in a block is a special transaction that starts a new coin owned by the creator of the block. This adds an incentive for nodes to support the network, and provides a way to initially distribute coins into circulation, since there is no central authority to issue them. The steady addition of a constant of amount of new coins is analogous to gold miners expending resources to add gold to circulation. In our case, it is CPU time and electricity that is expended.

3.6 Claiming Memory

Once the latest transaction in a coin is buried under enough blocks, the spent transactions before it can be discarded to save disk space. To facilitate this without breaking the block's hash, transactions are hashed in a Merkle Tree, with only the root included in the block's hash. Old blocks can then be compacted by stubbing off branches of the tree. The interior hashes do not need to be stored.

3.7 Simplified Payment Verification

It is possible to verify payments without running a full network node. A user only needs to keep a copy of the block headers of the longest proof-of-work chain, which he can get by querying network nodes until he's convinced he has the longest chain, and obtain the Merkle branch linking the transaction to the block it's timestamped in. He can't check the transaction for himself, but by linking it to a place in the chain, he can see that a network node has accepted it, and blocks added after it further confirm the network has accepted it.

3.8 Combining and Splitting Value

Although it would be possible to handle coins individually, it would be unwieldy to make a separate transaction for every cent in a transfer. To allow value to be split and combined, transactions contain multiple inputs and outputs. Normally there will be either a single input from a larger previous transaction or multiple inputs combining smaller amounts, and at most two outputs: one for the payment, and one returning the change, if any, back to the sender.

3.9 Privacy

The traditional banking model achieves a level of privacy by limiting access to information to the parties involved and the trusted third party. The necessity to announce all transactions

publicly precludes this method, but privacy can still be maintained by breaking the flow of information in another place: by keeping public keys anonymous. The public can see that someone is sending an amount to someone else, but without information linking the transaction to anyone. This is similar to the level of information released by stock exchanges, where the time and size of individual trades, the "tape", is made public, but without telling who the parties were.

Chapter 4

Proposals

Using the concepts of Block-Chain (Bitcoin) we propose our algoorithms for e-Biding D-APP system.

We will use Truffle framework with Metamask extension with Ganache api to build an e-Bidding System in Solidity Programming language.

Chapter 5

Implementation

Chapter 6

Conclusion and Further Work

This thesis provides a detailed description of an Practical implementation of Online Biding system which provides User Anonymity and Secure Key Exchange and agreement.

Further Work

It would be an interesting area of research to look into formal analysis of authentication protocols. Any formal analysis till date does not talk about security against various specific attacks. Any adversary even if he know the messages cannot break the ciphers from the function as he doesn't have the authority to do so.

In future we will try to apply these works in the other way like central server decentralized app. Also we can implement new apps for other fields also like music copyright verification or healthcare systems.