A

Seminar Report

Mobile Charger Billing System Using Lightweight Blockchain

Submitted in partial fulfillment of the requirements for the Award of the Degree

of

Master of Computer Applications

of

APJ Abdul Kalam Technological University

Submitted by

PHEBE JOHN

RegNo: TVE16MCA40

Department of Computer Applications

COLLEGE OF ENGINEERING TRIVANDRUM

OCTOBER 2018

DEPARTMENT OF COMPUTER APPLICATIONS

COLLEGE OF ENGINEERING TRIVANDRUM

CERTIFICATE

Certified that this Seminar report entitled, ***" Mobile Charger Billing System Using Lightweight Blockchain "*** *is the paper presented by* ***" Phebe John "(Reg No: TVE16MCA40)*** *in partial fulfillment of the requirements for the award of the degree of Master of Computer Applications of APJ Abdul Kalam Technological University during the year 2018.*

Prof. Baby Syla L.

Co-ordinator

Prof. Jose T Joseph.

Head of the Department

Acknowledgement

First and for most I thank **GOD** almighty and to my parents for the success of this seminar. I owe a sincere gratitude and heart full thanks to everyone who shared their precious time and knowledge for the successful completion of my seminar.

I would like to thank **Dr.Jiji C.V**, Principal, College of Engineering Trivandrum, who helped me during the entire process of work.

I am extremely grateful to **Prof.Jose T Joseph**, HOD, Dept of Computer Applications, for providing me with best facilities and atmosphere for the creative work guidance and encouragement.

I would like to thank my coordinator, **Prof. Baby Syla**, Dept of Computer Applications, who motivated me throughout the work of my seminar.

I profusely thank other Asst. Professors in the department and all other staffs of CET, for their guidance and inspirations throughout my course of study.

I owe my thanks to my friends and all others who have directly or indirectly helped me in the successful completion of this seminar. No words can express my humble gratitude to my beloved parents and relatives who have been guiding me in all walks of my journey.

Phebe John

Abstract

Green transportation such as electric vehicles are emerging as an alternative to the traditional vehicles primarily due to the increasing cost and need of petroleum energy worldwide. These electric vehicles operate by using electric charging and the way to charge an electric car is to use a mobile charger or use a charging infrastructure. Therefore, when a mobile charger is used, a billing system is required through which a user is billed who has charged the electric vehicle. In this paper, I propose a mobile charger billing system that utilizes Blockchain technology. This technology has been applied to achieve more secure online transactions in a peer-to-peer manner. Moreover, it analyzes the requirements of mobile charger for billing and propose a lightweight scheme that can overcome the challenge of data size in existing Blockchain. Current online transaction rely on certain trusted institutions. However, these third party sources can be hacked, manipulated or compromised. There is a need for an electronic payment system for direct transactions between trading partners, without the existence of a trusted third party. Blockchain is a technique that enables reliable electronic transactions by creating computational evidence of the time sequence of transactions using a peerto-peer distributed timestamp server to solve this problem. They explain electronic cash which is dealt in peer-to-peer network so that direct transactions can be made between the two parties without trading through a third trusted institution. A Blockchain is essentially a public ledger that is executed and shared between participants. Once the data has been entered into the block, it is difficult to forge or delete the information. If a malicious user attempts to modify or delete a block, it must also modify all previous blocks as well as the block at that point in time.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Recently, consumption of petroleum energy has increased worldwide, and the amount of petroleum consumed by automobiles has reached up to 30Therefore, it is recommended to find alternatives and use green transportation such as hybrid electric vehicle (HEV) and electric vehicle (EV). According to the global market for electric vehicles in 2015, including EVs, has reached 550,000 units. In North America, most electric vehicles are mainly charged at home. Benz and BMW, for example, are helping to set up a charging points for each home when buying their own electric car. However, in populated places where there are no garage, it is difficult to install a charge point. Also, If garage is shared among a building, the billing must be made to the concerned person, i.e., who has charged his car rather than divide equally. Therefore, in order to grow and disseminate the electric vehicle industry, it is necessary to devise an infrastructure and appropriate methods to solve these aforementioned issues.

Typically, there are two main methods for charging the electric vehicles, one uses a charge point while the other uses a mobile charger. If a charge point used, it will charge the electric vehicle using a grid-based fixed charging infrastructure. In case of mobile charger, the vehicle is charged from a power supply device by using the mobile charger. In the case of using a mobile charger, it is important not only to charge the electric car, but also to impose a cost to the user who charges the electric car. If a user charges an electric vehicle from a power supply at a place other than his / her home, there is probability that the billing is imposed on a person who is the owner of that place and not to the person who has charged the electric vehicle. Furthermore, charging at public places further complicates the scenario in which it becomes almost impossible to identify exactly who charged the electric car. Therefore, the current international standard for

mobile charging for electric vehicles is insufficient and has a number of open issues which need to be addressed. One of them is to establish a policy on how to impose the billing to the actual users. The transaction on the Internet have been dependent on financial institutions. However, if you depend exclusively on financial institutions, it will suffers from the inherent weaknesses of the trust-based model. Thus, there is a need for an electronic payment system for direct transactions between trading partners, without the existence of a trusted third party. In this work, I propose a mobile charging billing service system using Blockchain to solve the aforementioned scenario. Blockchain is a technique that enables reliable electronic transactions by creating computational evidence of the time sequence of transactions using a peer-to-peer distributed timestamp server to solve this problem.

Chapter 2

A Step to Blockchain

2.1 Blockchain data structure

2.1.1 Structure of a block

A blockchain, is a growing list of records, called blocks, which are linked using cryptography. Each block contains a cryptographic hash of the previous block, a timestamp, and transaction data (generally represented as a merkle tree root hash). A block is a container data structure, which brings together transactions for inclusion in the public ledger, known as the blockchain. The block is made up of a header; containing metadata, followed by a long list of transactions. A block can be identified in two ways, either by referencing the block hash, or through referencing the block height. The block header consists of three sets of block metadata. Metadata is data that provides information about other data. Firstly, there is a reference to a previous block hash, which connects this block to the previous block, lying in the blockchain. The second set of metadata relates to the mining competition; namely the difficulty, timestamp and nonce. Lastly, the third piece of metadata is the Merkle Tree root; a data structure used to summarize all the transactions in the block in an efficient manner.

Block headers can be regarded as an example of a dynamic membership multi-party signature (DMSS). A DMSS is a digital signature formed by a set of signers which has no fixed size (Back, Corallo, Dashjr, Friedenbach, 2014).Bitcoins block headers are DMSS because their proof of work has the property that anyone can contribute without undergoing an enrolment process. Furthermore, contribution is weighted by proportional computational power rather

than one threshold signature contribution per party (Back, Corallo, Dashjr, Friedenbach, 2014). This allows anonymous membership without risk of a Sybil attack. A Sybil attack is when one party joins many times and has an uneven, disproportionate input into the signature. Since the blocks are chained together, Bitcoins DMSS is cumulative. A chain of block headers is also a DMSS on its first block, with computational strength equivalent to the sum of the computational strengths of the composing DMSS . Therefore, the key innovation in Blockchain is a signature of computational power, rather than the typical signature of knowledge.

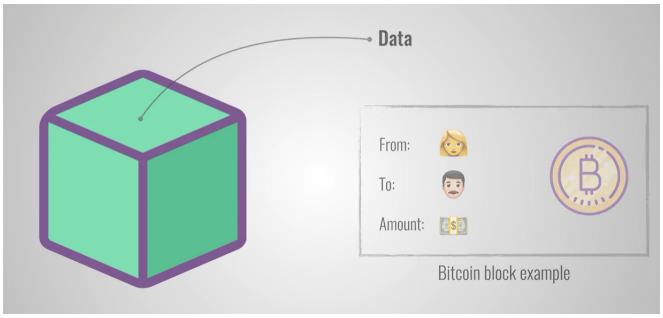

Figure 2.1: Data in the Block

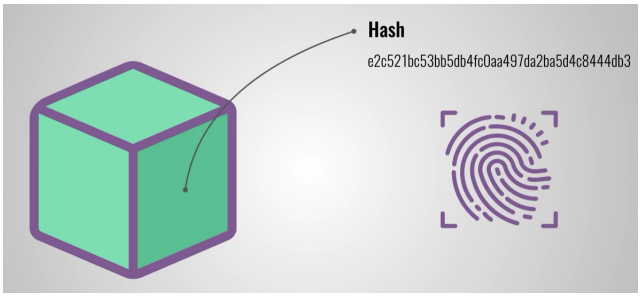

Figure 2.2: Hash in the Block

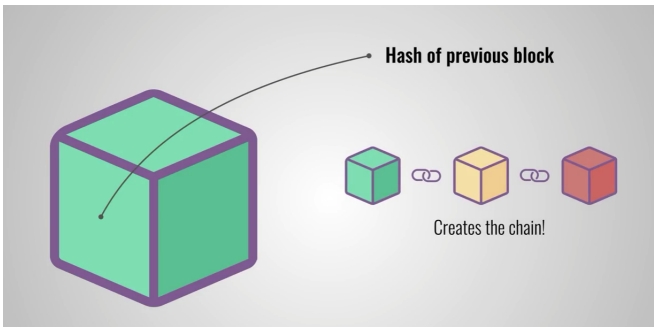

Figure 2.3: Hash of the Previous Block

2.1.2 Block header hash and nodes

Here I have am providing an example, the block hash of the first Bitcoin block ever created will be like 000000000019d7789c085ae165831e934gf763ae46a4a6c172b3f1b60a8ce26f. The block hash identifies a block uniquely, and can be independently derived by any node simply by hashing the block header. A node is a full client. A full client is a client that owns the block chains and is sharing blocks and transactions across the blockchain network. A node is considered to be part of the blockchain infrastructure, and does not necessarily have to be a miner. Each node keeps a complete copy of a totally ordered sequence of events in the form of a blockchain . The blocks hash is computed by each node, as the block is received from the network. The block hash may be stored in a separate database table as part of the blocks metadata, to facilitate indexing and faster retrieval of blocks from disk.

2.1.3 Block height

Block height is another method to identify a block, this time through its position in the blockchain. The first block ever created is at block height 0 (zero), and in the case of Bitcoin, is the same block that was referenced by the block hash of the above block which is 000000000019d7789c085ae165831e934gf763ae46a4a6c172b3f1b60a8ce26f .Each subsequent block added on top of that first block is one position higher in the blockchain, like boxes stacked one on top of the other.Block height does not always identify a particular singular block. It is possible for two or more blocks may have the same block height, both competing for the same position in the blockchain.

2.1.4 Genesis Block

The first block in any blockchain is termed the genesis block. If you start at any block and follow the chain backwards chronologically, you will arrive at the genesis block. The genesis block is statically encoded within the client software, that it cannot be changed. Every node can identify the genesis blocks hash and structure, the fixed time of creation, and the single transactions within. Thus every node has a secure root from which is possible to build a trusted blockchain on.

2.1.5 Proof of Work

In Proof of Work, in order for an actor to be elected as a leader and choose the next block to be added to the blockchain they have to find a solution to a particular mathematical problem. Given that the hash function used is cryptographically secure, the only way to find a solution to that problem is by bruteforce (trying all possible combinations). In other words, probabilistically speaking, the actor who will solve the aforementioned problem first the majority of the time is the one who has access to the most computing power. These actors are also called miners.

It has been widely successful primarily due to its following properties:

1. It is hard to find a solution for that given problem
2. When given a solution to that problem it is easy to verify that it is correct.

Whenever a new block is mined, that miner gets rewarded with some currency (block reward, transaction fees) and thus are incentivized to keep mining. In Proof of Work, other nodes verify the validity of the block by checking that the hash of the data of the block is less than a preset number.Due to the limited supply of computational power, miners are also incentivized not to cheat. Attacking the network would cost a lot because of the high cost of hardware, energy, and potential mining profits missed.

2.1.6 Linking blocks in the blockchain

Nodes maintain a copy of the blockchain locally, starting from the genesis block. The local copy of the blockchain constantly updates as new blocks are discovered and subsequently built on the chain. As a node receives information of incoming blocks from the network, it will validate these blocks first, then link them to the existing blockchain.The process to establish a link is as follows; a node will examine the incoming block header and look for the previous block hash. Looking at this incoming block, the node finds the previous block hash field, which contains the hash of its parent block. This hash is known to the node previously. Therefore, the node reasons that this new block is a child of the last block on the chain, and is the legitimate extension of the chain. The node adds this new block to the end of the chain, making the blockchain longer with a new height of the incoming block, now validated.

2.2 Cryptographic Hash Algorithm

The blockchain data structure is a back-linked list of blocks of transactions, which is ordered. It can be stored as a flat file or in a simple database. Each block is identifiable by a hash, generated using the SHA256 cryptographic hash algorithm on the header of the block. Each block references a previous block, also known as the parent block, in the previous block hash field, in the block header.A hash, also known in long form as cryptographic hash function, is a mathematical algorithm that maps data of arbitrary size to a bit string of a fixed size. In the case of SHA 256, the result is a string of 32 bytes. The resultant 32 bytes makes it effectively impossible to reverse the output, since the function was designed to be a one-way function .

The idea behind a hash functions use is to facilitate a thorough means for searching for data in a dataset. The most basic form of hash function is any function that can be used to map data of arbitrary size to data of fixed size. This output is a bit-string known as the hash value, hash sum or hash code. The hash values can be stored in a tabular form known as a hash table and is an efficient indexing mechanism; especially useful in search performance. Hash functions are collision-free too. That means its impossible to find two messages that hash to the same hash value. Therefore, when given a compact hash, one can confirm that it matches a particular input datum. Blocks can be identified from their hash, serving two purposes; identification and integrity verification.

Bitcoin hashing function makes use of the SHA 256, applied twice. It generates an almost-unique, fixed size 256-bit (32-byte) hash security. Large classes of hash functions are based on a building block of a compression function .Each block contains the hash of its parent inside its own header. There lays a chain going all the way back to the first block created, also known as the genesis block, linked together by a sequence of hashes. The previous block hash field is inside the block header and thereby the current block hash is dependent on the parent block hash. The childs own identity changes if the parents identity changes. When the parent is modified in any way, the parents hash changes. The parents changed hash necessitates an alteration in the previous block hash pointer of the child. This in turn causes the childs hash to mutate, which requires a change in the pointer of the grandchild, which in turn alters the grandchild and so on.

This cascading effect ensures that, once a block has many generations succeeding it, it cannot be changed without consequently forcing a recalculation of all the subsequent blocks. Because such a recalculation would require an enormous amount of computation, the existence of a long chain of blocks fortifies the Blockchains deep history to be immutable; a key feature of blockchain technology security.

2.3 Modification of Data

By design, a blockchain is resistant to modification of the data. It is "an open, distributed ledger that can record transactions between two parties efficiently and in a verifiable and permanent way". For use as a distributed ledger, a blockchain is typically managed by a peer-to-peer network collectively adhering to a protocol for inter-node communication and validating new blocks. Once recorded, the data in any given block cannot be altered retroactively without alteration of all subsequent blocks, which requires consensus of the network majority. Although blockchain records are not unalterable, blockchains may be considered secure by design and exemplify a distributed computing system with high Byzantine fault tolerance. Decentralized consensus has therefore been claimed with a blockchain.

Blockchains are incredibly popular nowadays.Here, I have a chain of three blocks. As you can see, each block has a hash and a hash of the previous block.So block 3 points to block 2 and block 2 points to the block 1. Now the block 1 is special. It is the Genesis block and it cannot point to any other block. Now if block 2 is tampered, it changes the hash of the block 2. Now the hash in the block 3 becomes invalid as it does not match with the hash in the previous block. Computers are very fast and they can calculate hash at a very high speed.

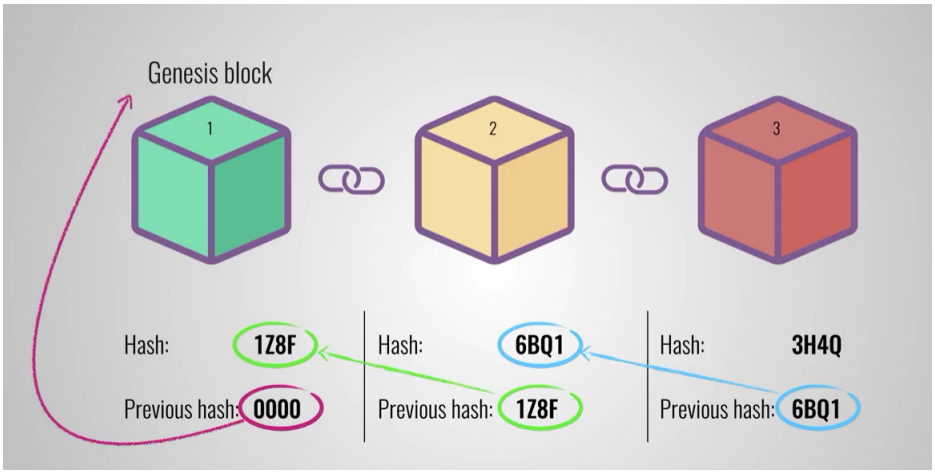

Figure 2.4: Chaining the blockchain

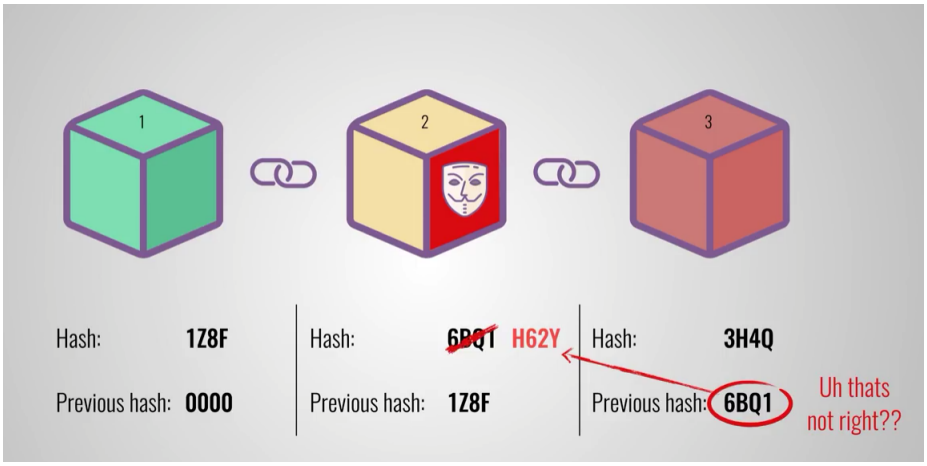

Figure 2.5: Denial on Data Modification

2.4 Types of blockchains

- Public blockchains : A public blockchain has absolutely no access restrictions. Anyone with an internet connection can send transactions[disambiguation needed] to it as well as become a validator (i.e., participate in the execution of a consensus protocol). Usually, such networks offer economic incentives for those who secure them and utilize some type of a Proof of Stake or Proof of Work algorithm.Some of the largest, most known public blockchains are Bitcoin and Ethereum.

- Private blockchains : A private blockchain is permissioned. One cannot join it unless invited by the network administrators. Participant and validator access is restricted.This

type of blockchains can be considered a middle-ground for companies that are interested in the blockchain technology in general but are not comfortable with a level of control offered by public networks. Typically, they seek to incorporate blockchain into their accounting and record-keeping procedures without sacrificing autonomy and running the risk of exposing sensitive data to the public internet.

- Consortium blockchains **:** A consortium blockchain is often said to be semi-decentralized. It, too, is permissioned but instead of a single organization controlling it, a number of companies might each operate a node on such a network. The administrators of a consortium chain restrict users reading rights as they see fit and only allow a limited set of trusted nodes to execute a consensus protocol.

2.5 Applications

- The Food Industry
- Cyber Security
- Voting
- Land Registry
- Smart Contracts
- Banks
- Insurance
- Internet-of-Things (IoT)
- Smart Appliances
- Supply Chain Sensors

Chapter 3

Related Work

BlockChain (BC) is a distributed database that maintains a growing list of blocks that are chained to each other.. BC has been shown to possess a number of salient features including security, immutability and privacy and could thus be a useful technology to address the aforementioned challenges BC is managed distributedly by a peer to peer network. Each node is identified using a Public Key (PK). All communications between nodes, known as transactions, are encrypted using PKs and broadcast to the entire network[2]. Every node can verify a transaction, by validating the signature of the transaction generator against their PK. This ensures that BC can achieve trustless consensus, meaning that an agreement between nodes can be achieved without a central trust broker, e.g. Certificate Authority (CA).

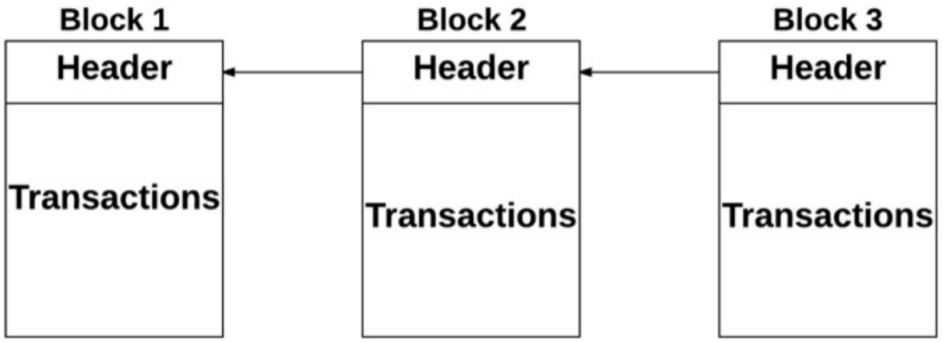

Figure 3.1: Structure of a Blockchain

A node will periodically collect multiple transactions from its pool of pending transactions toform a block, which is broadcast to the entire network. The block is appended to the local copy of the BC stored at a node if all constituent transactions are valid. A consensus algorithm

such as Proof of Work (PoW), which involves solving a hardto- solve easy-to-verify puzzle, is employed to control which nodes can participate in the BC. Once a block is appended, it (or the constituent transactions) cannot be modified, since the hash of each block is contained in the subsequent block in the chain, which ensures immutability. A node can change its PK (i.e. identity) after each transaction to ensure anonymity and privacy.

3.1 Blockchain

Current online transaction rely on certain trusted institutions. However, these third party sources can be hacked, manipulated or compromised[5] . Thus, novel secure schemes are required. In 2008, Nakamoto Satoshi proposed the Blockchain technology to solve the above problems. They explain electronic cash which is dealt in peer-to-peer network so that direct transactions can be made between the two parties without trading through a third trusted institution. A Blockchain is essentially a public ledger that is executed and shared between participants.

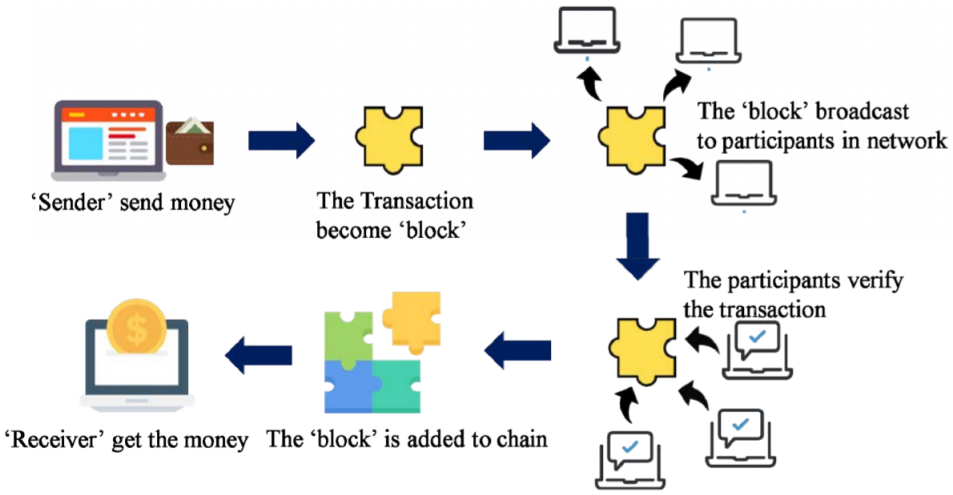

Figure 3.2: How Blockchain Works

Once the data has been entered into the block, it is difficult to forge or delete the information. If a malicious user attempts to modify or delete a block, it must also modify all previous blocks as well as the block at that point in time. The Figure shows an example of an online transaction using Blockchain. Each Blocks constituting a Blockchain consist of a 'block header' and a 'block body'. The block header includes the hash value of the previous block header. In addition, each

block is linked by a linked list method such as a chain. Block bodies may contain different values depending on its service.

3.2 Simplified Payment Verification - SPV

Note that all nodes do not possess the capability to store the full Blockchain especially the resource constrained devices, e.g., space and power-constrained devices cannot maintain the full Blockchain. Therefore, for such devices, a simplified payment verification (SPV) is used to operate without the full Blockchain. SPV nodes download only the block header rather than the complete chain. Therefore, they do not know about the transactions. SPV nodes can verify the transactions using a different method. They verify transactions by reference to their depth in the Blockchain instead of the height. SPV nodes will verify the chain of all blocks and link that specific chain to the transaction of interest . The SPV node will establish a link between the transaction and the block that contains it, using a Merkle Path.

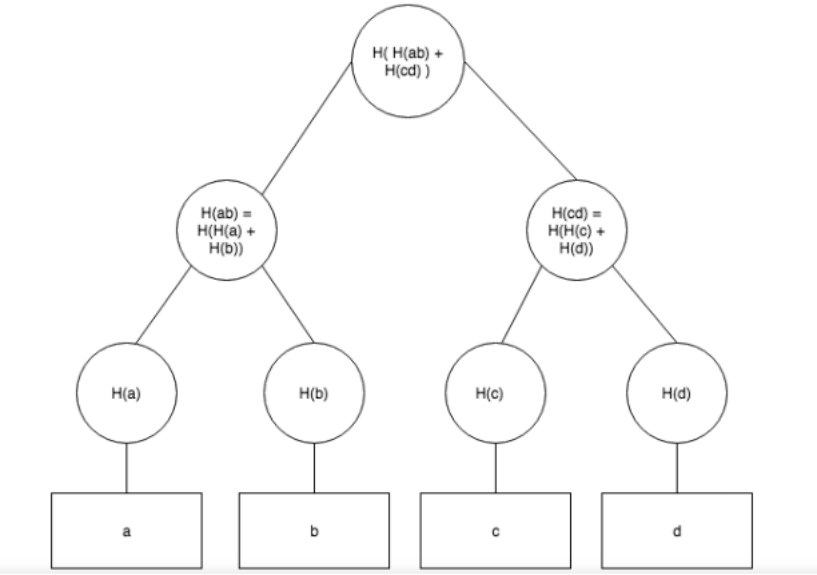

Figure 3.3: Merkle Path

a, b, c, and d are some data elements (files, public/private keys, JSON, etc) and H is a hash function. If you are unfamiliar, a hash function acts as a digital fingerprint of some piece of data by mapping it to a simple string with a low probability that any other piece of data will map to the same string. Each node is created by hashing the concatenation of its parents in the tree.

You will notice that the Merkle tree here is a binary tree, most Merkle Trees are binary, but there are non-binary Merkle Trees employed in platforms like Ethereum. Here I will just cover the binary case as it is by far the most common.

The tree can be constructed by taking nodes at the same height, concatenating their values, and hashing the result until the root is reached. A special case needs handled when only one node remains before the tree is complete, but other than that the tree construction is somewhat straightforward (more on this in the implementation section).

Once built, data can be audited using only the root hash in logarithmic time to the number of leaves (this is also known as a Merkle-Proof). Auditing works by recreating the branch containing the piece of data from the root to the piece of data being audited. In the example above, if we wanted to audit c (assuming I have the root hash), I would need to be given H(d) and H(H(a) + H(b)). I would hash c to get H(c), then concatenate and hash H(c) with H(d), then concatenate and hash the result of that with H(H(a) + H(b)). If the result was the same string as the root hash, it would imply that c is truly a part of the data in the Merkle Tree.

In a case such as torrenting, another peer would provide the piece of data, c, H(d), and H(H(a) + H(b)). If youre concerned about the security of this approach, recall that in a hash function it is computationally infeasible find some e such that H(e) = H(c). This means that so long as the root hash is correct, it would be difficult for adversaries to lie about the data they were providing.

Outputting the authentication path of some data is as simple as recreating the branch leading up until the root. Traversing the entire tree to produce the leaves and their respective authentication data becomes important when using the Merkle Tree in digital signature schemes, and this can actually be accomplished in under logarithmic time.

Chapter 4

Scenarios and System Architecture

The system consists of a power supplier, a service provider, and a mobile charger. It is assumed that both the Service Provider and the mobile charger parent node are Full Block. The remaining mobile chargers utilize SPV. In this section, I propose a data structure of a mobile charger for charging through a mobile charger, and discuss how to make lightweight Blockchain.

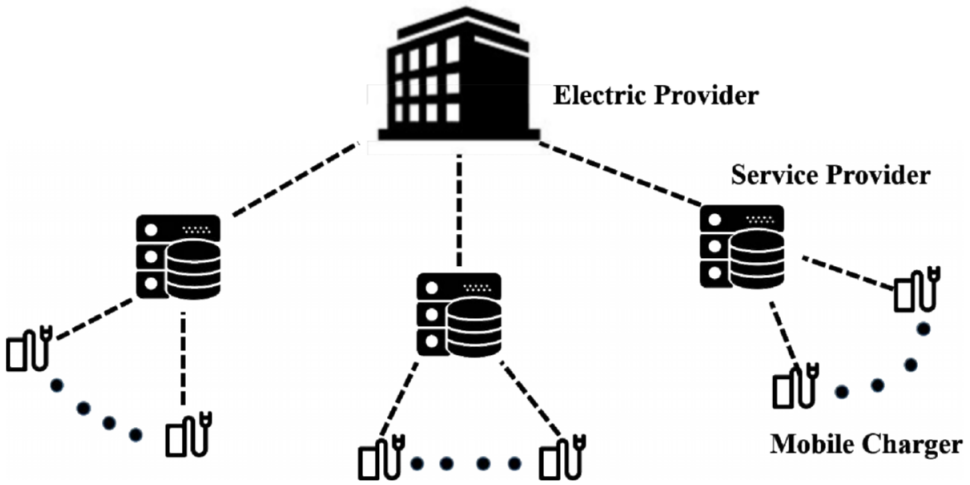

Figure 4.1: System Model

4.1 Mobile charger packet information for billing

Before referring to the mobile charger packet information for billing according to its charging, it is assumed that each mobile charger knows the IP address of its own service provider. If each mobile charger is operated on the network for the first time, it can only participate if it searches

other nodes in the network. As this time, each mobile charger can obtain the information of the current block and the neighboring node through the service provider.

It shows the message type and data type of the mobile charger. Table 4.1 shows the message type whereas the data types are specified in Table 4.2. If some users make charging group and pay the charging fee at the same time, it is much more effective to impose one charger group than to charge for each mobile chargers. Furthermore, some person group can pay the charging fee at the same time, this case is same as like the previous case. Therefore, the mobile charger can have a groupId value with a unique identifier. If certain mobile chargers are grouped together, they can be grouped by passing their groupId value to their service provider.

Table 4.1: Message Type

Message	**Description**
Register	The mobile charger registers itself by transferring its idTag to the Service Provider Server.If grouping is required, the groupId value can be passed along with the idTag value
RegisterAck	Response message to Register.
CheckAuth	If a new mobile charger is added to the group, the Service Provider server forwards this message to the parent node of the group. After receiving this message, the mobile charger confirms that the idTag value of the charger to be added belongs to its own group.
CheckAuthAck	Response message to CheckAuth
Authorize	A mobile charger participating in a group requests permission to join the group by sending its idtag.
AuthorizeAck	Ack Response message to Authorize

Table 4.2: Data Type

Type	**Description**
idTag	Mobile Charger unique identifier
idTagInfo	It is delivered after registration. There are Interval, currentTime, status fields. status field is used only when making a group, and if it has an Accepted, it means the mobile charger get authority to the group
Interval	Cycle to send ChargeProfile
currentTime	The current time in the Service Provider. It is used to synchronize the mobile charger's internal clock
ChargeProfile	Charge history of mobile charger. It consists of idtag and each charge history. Charging history includes start time, maximum output power, and end time.

Each message and description is shown in Table 4.1. Each groupId value is unique, and the first mobile charger to register groupId serves act as the parent node of the group. In the proposed system, the parent mobile charger is a full node(all blocks are held), and the other mobile charger in the group is an SPV node. The messages inlcudes Register, RegisterAck, CheckAuth, CheckAuthAck, Authorize.

Each data type and description is shown in Table 4.2. Each idTag value is unique, which represents the unique identifier. The data type inlcudes idTag, idTagInfo, Interval, currentTime,ChargeProfile.

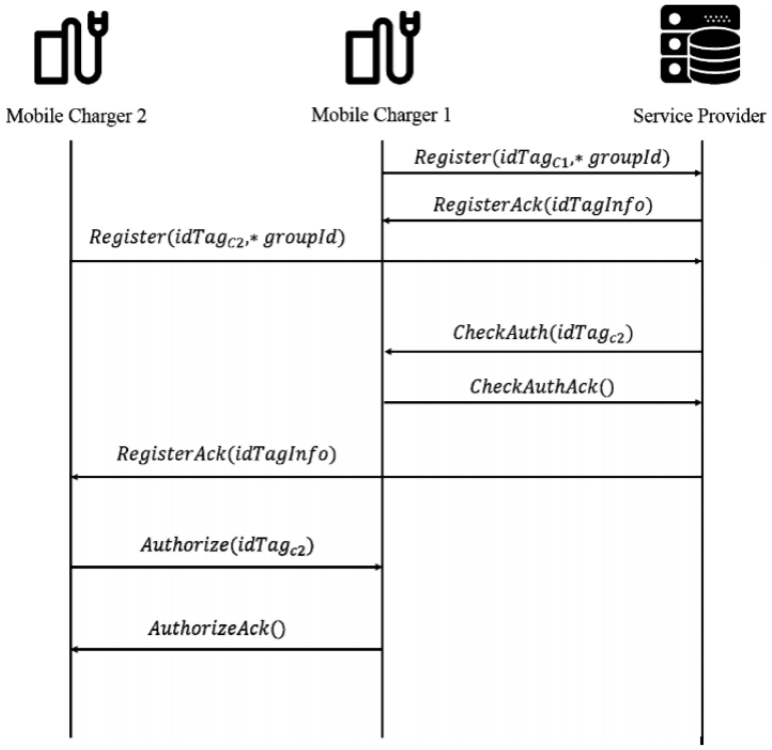

Figure 4.2: Sequence of the mobile charger registration

After completing the registration sequence with the Service Provider, the mobile charger transmits its Charging Profile to the Service Provider after completing the charging process. If the service provider receives 'Charging Profile' from mobile charger, it sends the 'Charging profile' to chargers belonging to other group. The mobile charger delivers all its charging profiles which occurred within the interval value from idTagInfo to the parent node.

The parent node that receives the profile of the group generates a block containing the contents of all profiles and sends it to the service provider. The Service Provider forwards the block to all groups. If you get more than half the correct validation results for a transaction, the service provider adds the block to the existing Blockchain. Then Service Provider pass the block to all nodes. The service provider checks the block information for billing and transmits the charge information according to the charge amount for each group.

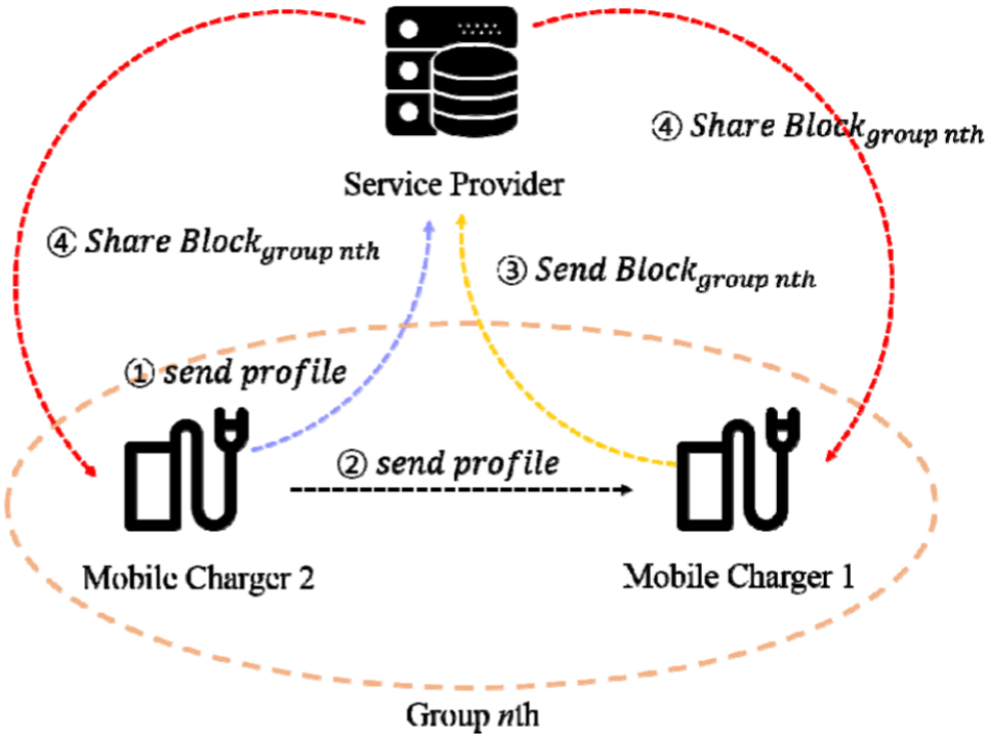

Figure 4.3: Sequence of Transaction Communication

4.2 Lightweight Blockchain data

Currently, the Blockchain technique has some limitations. One of the biggest challenge in it is the size of the data. The Blockchain will continue to accumulate previous data records, so the size of the data over time will increase. Moreover, in Blockchain architecture the data is generated after every predetermined period and this data is broadcasted to all node that belong to the network. So, if the number of nodes are increased, the data size will grow exponentially. Thus the cost of maintaining this data will also increase accordingly.

To reduce the size of the data, a simple way is to delete the old block data which are no longer needed and not required to be maintained. However, this method may cause another issue relating to the loss of the data value at a certain time. For example, since the previous charge record is deleted, the user may deny the its bill even though the charge was actually made by the user.

Algorithm 1 : Block Data Size decrease

Input: ***Whole Block***
Output: ***Charge List Block***
1. **Check Charging Data for each group**
2. **for number of group do**
3. *CurrentBlockCharge* per each group is '0'
4. **end for**
5. **do**
6. **for number of group do**
7. $i = 1$
8. $Charge_{G_i} = Charge_{G_i} + CurrentBlockCharge_{G_i}$
9. **End for**
10. *CurrentBlockCharge* move to next Block
11. **while(End of Block)**
12. **Generate Charge List Block**
13. **for number of group do**
14. Add $Group_{id}$ + $Charge_{G_i}$ to *Block data*
15. **end for**
16. Add $BlockHeader_{lastest}$ and *Block data*
17. Send *Charge List Block* to each group parent node

Figure 4.4: Algorithm - Block Data Size Decrease

In my proposed scheme, parent mobile charger maintains the full Blockchain and other mobile chargers in the group are SPV node. In such case, any charger may be reluctant to become a parent node as the parent mobile charger have to maintain an additional overhead of the full block. So, I propose a novel method to reduce Blockchain data size for parent node. The proposed method makes more efficient management compared to the block data. The proposed method is a re-construction of a block into a new type of block which is called Charge List Block. The service provider receives block for each group transaction from the parent node, and it reconstructs the block body part.

The service provider periodically checks the size of the Blockchain data received from each group. If the size of the Blockchain data exceeds a certain size, it check the billing profile for the last transaction for each group. The Service Provider checks the height of the block from

the first block to the block containing the last transaction contents. After this process, the Service Provider creates a Charge List Block by listing the idTag and charge usage of all mobile chargers in the group. The block header of the Charge List Block is generated in the same way as the existing block header part, and is transmitted for each group. The group parent mobile chargers receive the newly created block form a new Blockchain starting from the reconstructed block. Through this method, the parent mobile charger can be maintained the lightweight full Blockchain.

4.3 Performance Analysis

In this section, I present the results of the analysis for the proposed method of mobile chargers and the lightweight Blockchain technique.If any malicious user wants to change the existing charging record, he or she must change not only that block but also all blocks after that block. In fact, it is impossible to change the record because it is shared by all nodes participating in the network.Since the parent node receives the profile in the group of every interval, the malicious user may attempt to change the value of the profile in the group on the parent node.

This is because, if the parent node changes the value to generate a block, the nodes belonging to the other group receive the existing profile contents in advance, and can confirm whether or not the profile is changed through validation. If the contents of the profile are different, the block containing the profile is not tied to the existing Blockchain. Therefore, it is impossible to attack a profile change with malicious parent. The proposed lightweight Blockchain scheme can reduce the size of existing Blockchain data.

Existing blocks must include a signature and a secret key for each transaction. However, if the proposed method is used, the data size can be greatly reduced because only the idTag, the charging amount of the mobile charger and the hash of the last transaction are maintained even if the number of transactions increases.

Chapter 5

Conclusion and Future Scope

In this paper, I propose a mobile charger billing system using Blockchain. In order to provide efficient charging according to the charge details of the mobile charger, the mobile charger can be grouped by utilizing the groupId. In addition, using Blockchain technology, appropriate billing for charging can be generated. Moreover, I propose a technique to reduce the size of block data, and solve the problem of accumulating data size of existing Blockchain.

However, in the proposed system, there is still remains a challenge that needs to be addressed. It is possible for a particular attacker to pretend to have a lot of mobile chargers. Therefore, when validating a block, an attacker can send a lot of messages that block is not valid. In such a case, the block may not be added to the chain, even if the block is valid, according to the principle of a block chain. Therefore, I intend to further study more techniques to address this problem as future research course.

Bibliography

[1] Nam Ho Kim, Sun Moo Kang, Choong Seon Hong *"Mobile charger billing system using lightweight Blockchain"*, Sep. 2016.

[2] Ali Dorri, Marco Steger, Salil S. Kanhere, and Raja Jurdak *"BlockChain: A Distributed Solution to Automotive Security and Privacy"*, IEEE Communications Magazine, Vol. 55 , Issue. 6 , Dec. 2017

[3] Narayanan, Arvind; Bonneau, Joseph; Felten, Edward; Miller, Andrew; Goldfeder, Steven *"Bitcoin and cryptocurrency technologies: a comprehensive introduction"*, IEEE Communications Magazine, 2016

[4] Leonardo Aniello , Roberto Baldoni , Edoardo Gaetani , Federico Lombardi , Andrea Margheri *"A Prototype Evaluation of a Tamper-Resistant High Performance Blockchain-Based Transaction Log for a Distributed Database"*,European Dependable Computing Conference (EDCC), Sep. 2017

[5] Michael Crosby et al, *B*lockchain Technology: Beyond Bitcoin, AIR (Applied Innovation Review), No. 2 ,June 2016

[6] Kenji Saito , Hiroyuki Yamada *"Whats So Different about Blockchain? Blockchain is a Probabilistic State Machine"*,International Conference on Distributed Computing Systems Workshops, Jun. 2016

[7] Quoc Khanh Nguyen *"Blockchain - A Financial Technology for Future Sustainable Development"*, Conference on Green Technology and Sustainable Development (GTSD) Nov. 2016