



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 7 Issue: V Month of publication: May 2019

DOI: <https://doi.org/10.22214/ijraset.2019.5132>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com



Secure Financial Application using Block-Chain with Data Protection

Megha J¹, Jhansi Rani P²

¹M.Tech, ²HOD & Professor, Computer Science, CMR Institute of Technology, Bengaluru

Abstract: In an IoT environment using a block-chain, when data authentication or device authentication information is placed in a block-chain, personal information may be disclosed during the job validation process or address lookup. On paper, we apply zero knowledge evidence to a Mobile Financial wallet system to prove that a prover did not disclose information such as the public key, and we investigated how to improve the anonymity of the chain of custody. Blocks for the protection of confidentiality. A secure web browser login system has been set up and implemented using cryptographic techniques for purpose of authentication.

The proof of zero knowledge and AES algorithm is a concept that has been used here to provide cryptographic systems for more authentication. Zero knowledge protocol with AES cryptographic algorithm can be applied on the client side and it works between the client and server. In this it builds an algorithm in the authentication system as firewall or with firewall. This allows a party to prove that he / she knows something (i.e., Credential), without having to send the value of the credential. In this implementation, it will be used to prove the user's password without sending on the current password. The system also does not allow any Password hashes must be stored on the server. The purpose of the implementation is to make the user confidential and authentication password

Keywords: Block-Chain, Zero knowledge proof (ZKP), AES cryptographic algorithm, Authentication, Data protection.

I. INTRODUCTION

A. Overview

The IoT is the abbreviation of the Internet of Things, which enables objects to share and control data between objects because things are connected to the Internet. It is possible to commit malicious attacks, such as data tampering, or privacy infringement, while sharing data on objects over the Internet. This paper introduced a block chain to prevent security threats such as data counterfeiting, which could occur using Mobile Financials. Zero-Knowledge proof, a block chain anonymity enhancement technology and NFC, was introduced to prevent security threats such as personal information infringement through block inquiry. It was proposed to use smart contracts to prevent Mobile Financial wallet data forgery and personal information infringement we suggest.

B. Existing System

Block chains with anonymity include Monroe, Dash, and ZCASH so on. An anonymous block-chain is a chain of blocks that makes it impossible to track an account and the content of a transaction, such as an account, etc., to avoid the violation of personal information. They implemented an anonymous chain of blocks using different security technologies. Monroe applied technology to avoid tracing existing bit currencies with digital assets using the Crypto-note protocol. He used a special encryption technique called Ring Signatures, one-time keys. It is very difficult for a third party to confirm the content of a transaction because the key is mixed in a certain group and a Private Key is required to confirm the transaction.

Disadvantages

- 1) Personal Information may be leaked.
- 2) Proof-of-work process.
- 3) Less efficiency.

C. Problem Statement

The Internet of Things allows objects to share and control data between objects because things are connected to the Internet. It is possible to commit malicious attacks, such as the management of data or the violation of privacy, while dealing with objects over the Internet. There are risks that the machines are hacked by fraudulent transactions. Chain security, certain weak key generations by a certain block-chain program, double spending are other key security concerns.



D. Proposed System

As a proposed system environment, the administrator can control the web server and the Mobile Financial Wallet Application (MFWA) that must be installed on the user's Android mobile phone. The details of the user transaction made in the mobile financial wallet application are sent to the web server and stored in the block chain. The user can log in to the mobile financial wallet application with the help of zero knowledge authentication using NFC (Near Field Communication). To enter MFWA, the user must provide identification and touch the NFC card on the mobile. MFWA that can read the NFC hash code and send it to the web server. According to the user ID, it will search for the hash code (1) and compare it with the hash code already stored (2), if it is authenticated, it will go to the financial main page.

Advantages

- 1) Security
- 2) Immutable
- 3) Efficient

II. LITERATURE SURVEY

The following works were carried out by specific persons in the area of Block-chain:

- A. In a block-chain IoT environment, when the data or the authentication information of the device is placed in a chain of blocks personal information can be filtered through the work test. In this work, we apply Zero-Knowledge test of an intelligent meter system to demonstrate that a prover without revealing information like the public key and have studied how to improve the anonymity of the chain of blocks for privacy protection.
- B. After the global financial crisis of 2008, the world has been making a greater effort to restrict banking and financial activities with stricter regulations. However, the effectiveness of this policy has been controversial since many people believe that policy makers should promote freedom and transparency by allowing the public to directly interfere and change the system for the public interest. This article attempts to synthesize and analyse the available information with a focus on the role of block-chain, a financial tool that can potentially play an important role in the sustainable development of the global economy.
- C. Block-chain (BC) has received significant attention recently. This document presents problems related to the system for BCs for financial applications. This document presents for the first time the design of a BC without taking into account the scenarios of the application, and problems such as performance, security, performance and scalability lead to specific BC designs. The BC sample scenarios are analysed and lead to additional BC designs. Specifically, two new types of BC arise: to store information at the transaction level, to store account information. By dividing traditional BCs into these two BCs, the system can be optimized with respect to scalability and privacy.

III. THEORETICAL BACKGROUND

A. Block-Chain

The entire Block-chain refers to an encrypted and distributed database, which is a public repository of information that cannot be reversed and is incorruptible. In other words, a chain of blocks can be defined as a distributed public ledger or a database of records of each transaction that has been made and shared among the participants in the network. Each transaction or digital event in the public ledger must be authenticated through the agreement of more than half of the participants in the network. This implies that no participant or user as an individual can modify any data within a chain of blocks without the consent of other users (participants). It could be clearly seen that the technological concept behind the block chain is very similar to that of a database. Block-chain makes it possible for the participants for the first time to reach an agreement on how a specific transaction or digital event can occur without a control authority being necessary. This technology (Block-chain technology) is unique in the sense that it reduces the role of the intermediary.

Blocks are data structures whose purpose is to group sets of transactions and distribute them to all nodes in the network. The blocks are created by the miners. Blocks contain a block header, helps to verify the validity of a block.

Typical block metadata contains

- 1) Version - the current version of the block structure
- 2) Previous block header hash - the reference of the main block of this block
- 3) Root hash: a cryptographic hash of all the transactions included in this block.
- 4) Time - the time this block was created
- 5) n-Bits: the current difficulty that was used to create this block
- 6) Nonce ("number used once") - a random value that the creator of a block can manipulate, however, so choose it

These 6 fields constitute the block header. The rest of a block contains transactions that the miner has chosen to include in the block he created. The structure of the Block-chain is shown in Fig. 1.

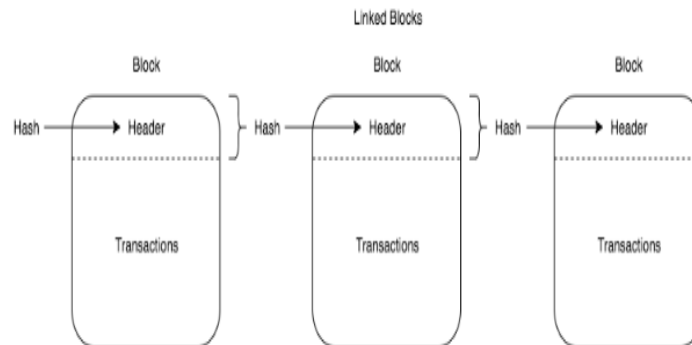


Fig. 1 Block-Chain

B. Zero Knowledge Proof (ZKP)

In cryptography, a zero knowledge proof or zero knowledge proof protocol is a method by which one party (the prover) can prove to another party (the verifier) that it knows a value x , without transmitting information knowing the value x . The essence of a proof of zero knowledge is that it is trivial to prove that someone has knowledge of certain information by simply revealing it. The challenge is to justify such possession without disclosing the information itself or additional information.

A proof of zero knowledge must satisfy the following three parameters:

- 1) *Completeness*: If the statement is true, the honest auditor - the one who follows the protocol correctly - will be convinced by an honest prover.
- 2) *Soundness*: If the statement is false, no cheating prover can convince the honest verifier that it is true, except for a low probability.
- 3) *Zero Knowledge*: If the statement is true, no auditor learns anything except the fact that the statement is true. In other words, it is enough to know the statement (not the secret) to imagine a scenario showing that the prover knows the secret. To do this, each auditor has a simulator capable of producing a transcript that "resembles" an interaction between the honest prover and the auditor. The simulator should be able to produce the transcript, while having access to the statement to prove and not to the prover itself.

Completeness and soundness are the properties of more general interactive proof systems. The addition of zero knowledge is what turns the verification process into proof of zero knowledge.

Zero-knowledge proof are not proofs in the mathematical sense of the term, because there is a small probability, the soundness error, that a fraud prover can convince the verifier of a misrepresentation. In other words, zero-knowledge proofs are probabilistic rather than deterministic proofs. However, some techniques can reduce the reliability error to negligible values.

IV. SYSTEM REQUIREMENTS

A. Hardware Requirements

- 1) Processors : Pentium IV
- 2) RAM : 4GB
- 3) Storage : 20GB or higher
- 4) Keyboard : Standard keyboard

B. Software Requirements

- 1) Platform : Windows XP/7
- 2) IDE/tool : Eclipse Galileo
- 3) Coding : Java (Jdk 1.7)
- 4) Web Technology: Servlet, JSP
- 5) Web Server : TomCat6.0
- 6) Database : MySQL5.0

V. SYSTEM ANALYSIS

A. System Architecture

The structure and behaviour of a system is the conceptual design which is defined by system architecture. The architecture description can be said as the formal description of a system, which is organized in a way to support reasoning about the structural properties of a system. It provides a proper plan by which a project is developed. It also include definitions about the system components or building blocks of the overall system.

The Fig. 2 shows the system architecture of the proposed system. As a proposed system the administrator can control the web server and the Mobile Financial Wallet application (MFWA) that must be installed on the user's Android mobile phone. The details of the user transaction made in the mobile financial wallet application are sent to the web server and stored in the block chain. The user can log in to the mobile financial wallet application with the help of zero knowledge authentication using NFC (Near Field Communication). To enter MFWA, the user must provide identification and tap the NFC card on the mobile. MFWA that can read the NFC hash code and send it to the web server. According to the user ID, it will search for the hash code (1) and compare it with the hash code already stored (2), if it is authenticated, it will be directed to the financial main page

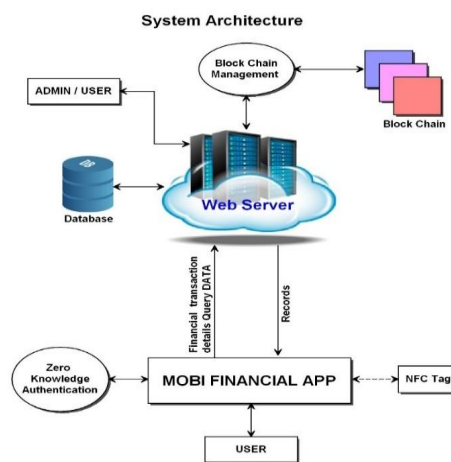


Fig. 2 System Architecture

B. Data Flow Diagram

The Data flow diagram Fig. 3 depicts that the admin is responsible for user settings i.e., to add/view users in the admin web application. The user login by tapping NFC card onto mobile in the mobile financial application. The NFC sensor read the credential from NFC card and gives to zero knowledge authentication protocol which is responsible to validate the credentials with that stored in the block-chain. If the authentication is successful, user is directed to home page and able to initiate transactions related to financial details.

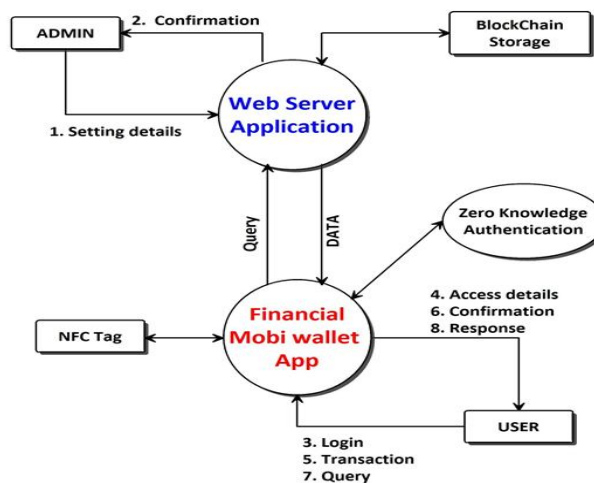


Fig. 3 Data Flow Diagram

C. Class Diagram

A class diagram is a Unified Modelling Language. It provides a structure that explains the working of the system it explains using classes and subclasses. It also highlights the number of attributes used by the system classes and it explains the relationship between different classes.

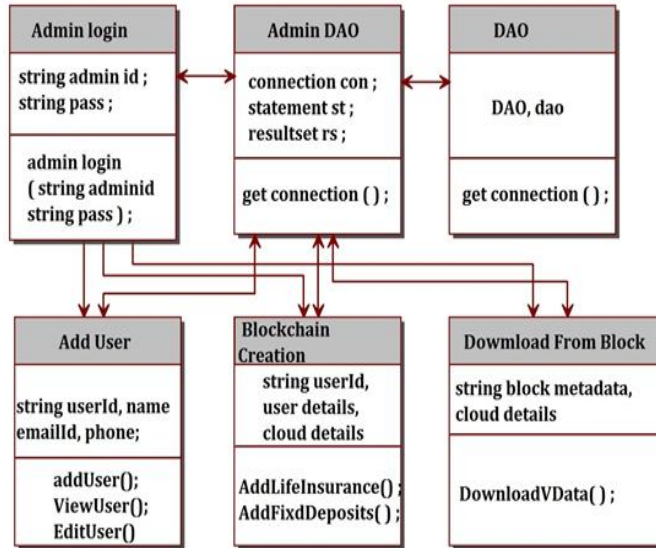


Fig. 4 Class Diagram

The Fig. 4 shows the class diagram of the system in which the admin is able to Add, View or Edit user. The user can add his personal information in which it will be stored in the form of block. All the information in the block are encrypted and the metadata of the block will be stored in the cloud.

D. Sequence Diagram

A sequence diagram is a Unified Modelling Language; it represents the interactions between the processes that work with co-operation and it shows the order in which the interaction process starts and ends.

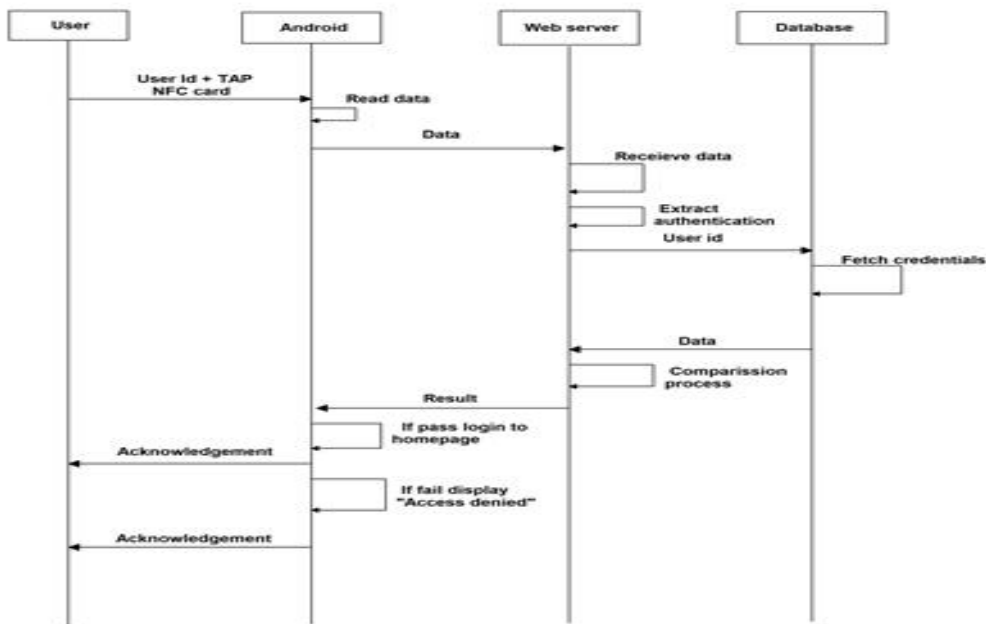


Fig. 5 Sequence Diagram

The above fig. 5 shows the sequence diagram for financial application.

- 1) The user login using user ID and tap the NFC card onto the mobile.
- 2) The NFC sensor read the data from NFC card and sends it to the web server.
- 3) The web server receives the data and initiates the authentication process by fetching details from the database.
- 4) The database receives the query and response it by sending data to the web server.
- 5) The server compares the credentials and send result to the application.
- 6) If authenticated successfully, then user will be directed to the home page, else an error message will be displayed to the user.

VI. IMPLEMENTATION

A. Modules

- 1) *Admin Module:* The administrator must log in with his username and password. Once logged in, admin can add users and view the details of the user. He can also edit when adding a user, the admin is also responsible for creating a hash code for that user.
- 2) *NFC Writing Process:* In Admin android application, the admin is responsible for writing the user information onto the NFC tag, which is then used as a password to authenticate the user.
- 3) *NFC Reading Process:* In this user module user has to login using user-id, if authentication is correct it will navigate to the home page. Then the user can store their personal details.
- 4) *Zero Knowledge Authentication:* In this section, when the user stores his personal data, he creates metadata and stores it in a database. Only these data allow us to find the personal data of the user.
- 5) *Creation of Block-Chain:* In this module, the user's personal data will be stored in the cloud in an encrypted format. When the user wishes to download this data, he must decrypt them and display them to the user

B. Algorithms

- 1) *Advanced Encryption Standard (AES):* The Advanced Encryption Standard (AES) algorithm is the most popular and widely adopted symmetric encryption algorithm. It is found at least six times faster than the triple DES.

Encryption Process:

In a typical series of AES encryption, each round includes four sub-processes. The process of the first round is described below in fig. 6

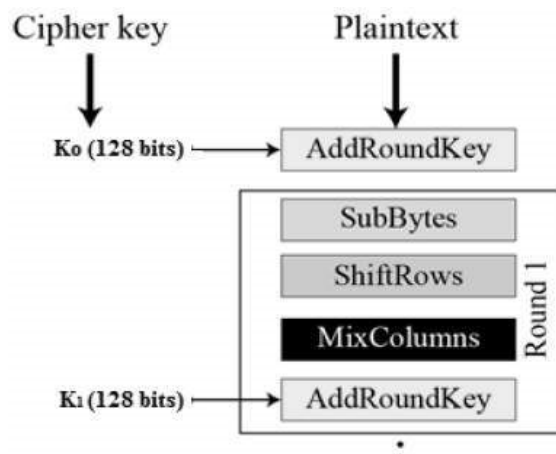


Fig. 6 AES Encryption Process

- a) *Byte Substitution (SubBytes):* The 16 bytes of input are replaced by the search for a fixed table (S-box). The result is a matrix of four rows and four columns.
- b) *Shift rows:* Each of the four rows of the matrix is shifted to the left. All entries that "fall" are reinserted to the right of the line. The offset is performed as follows -
 - i) The first row is not shifted.
 - ii) The second row is shifted one byte to the left.
 - iii) The third row is shifted two positions to the left.
 - iv) The fourth row is shifted three positions to the left.

The result is a new matrix composed of the same 16 bytes but offset with respect to each other.

- c) *Mix-Columns*: Each four-byte column is now transformed using a special math function. This function takes the four bytes of a column as input and generates four entirely new bytes, which replace the original column. The result is another new matrix of 16 new bytes. It should be noted that this step is not performed in the last round.
 - d) *Addroundkey*: The 16 bytes of the matrix are now considered as 128 bits and are XORed to the 128 bits of the round key. If this is the last round then the output is the cipher text. Otherwise, the resulting 128 bits are interpreted as 16 bytes and we begin another similar round.
- 2) *Decryption Process*: The process of decrypting an AES encrypted text is similar to the encryption process in reverse order. Each cycle includes the four processes conducted in reverse order:
- a) Add a round key
 - b) Mix the columns
 - c) Stagger lines
 - d) Byte substitution

Since the sub-processes of each cycle are inverted, unlike Feistel encryption, the encryption and decryption algorithms must be implemented separately, although they are very closely related.

- 3) *Message Digest 5 (MD5)*: The extended MD5 algorithm takes a random length message as input and outputs a 128-bit message digest of the input. This algorithm ensures that it is impossible, from a computational point of view, to produce two messages having the same message digests or produce any message having a given predetermined target message digest.

MD5 algorithms includes the following steps:

- i) *Step 1: Append padded bits*
 - 1. The message is filled so that its length is congruent to 448 modulo 512.
 - 2. A single "1" bit is added to the message, then the "0" bits are added so that the length in bits equals 448 modulo 512.
- ii) *Step 2: Append Length*
 - 1. A 64 bit representation of b is added to the result of the previous step.
 - 2. The resulting message has a length that is an exact multiple of 512 bits.

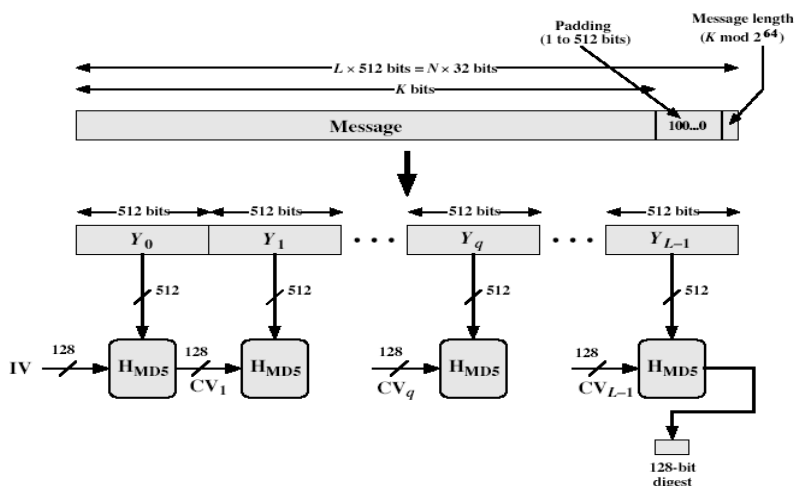


Fig. 7 MD5 Overview

- iii) *Step 3: Initialize MD Buffer*
 - 1. A four-word buffer (A, B, C, D) is used to calculate the message digest.
 - 2. Here, each of A, B, C, D, is a 32-bit register.
 - 3. These registers are initialized at the following values in hexadecimal:
 - Word A: 01 23 45 67
 - Word B: 89 ab cd ef
 - Word C: fe dc ba 98
 - Word D: 76 54 32 10

iv) Step 4: Process Message in 16-word block

1. Four auxiliary functions that take as input three of 32 bits words and output a 32-bit word.

$$F(X,Y,Z) = XY \vee \text{not}(X) Z$$

$$G(X,Y,Z) = XZ \vee Y \text{not}(Z)$$

$$H(X,Y,Z) = X \text{ xor } Y \text{ xor } Z$$

$$I(X,Y,Z) = Y \text{ xor } (X \vee \text{not}(Z))$$

2. If the bits of X, Y and Z are independent and without bias, each bit of F (X, Y, Z), G (X, Y, Z), H (X, Y, Z) and I (X, Y, Z) will be independent and impartial.

v) Step 5: Output

1. The output message summary is A B C D.
2. In other words, the output starts with the low byte of A and end with the most significant byte of D.

VII. RESULTS

The following snapshots define the results or outputs that are obtained after step by step execution of all modules of the system.

A. Admin Web App

The fig. 8 depicts the login page of the web app of the admin which contains Admin ID, password and sign in button, which is used to authenticate the Admin. The Admin ID and password are predefined in the database. Once the Admin is authenticated then he is allowed to move to the next page.

B. Add User

The fig. 9 depicts the view user module in which the admin can view list of all users. It also contains edit and delete buttons through which admin can edit or delete the user.

C. View User

The fig. 10 depicts the view user module in which the admin can view list of all users. It also contains edit and delete buttons through which admin can edit or delete the user.

D. Delete User

The fig. 11 depicts the delete user module in which admin can delete the user by providing user id and name of the user.

E. User Android Login Page

The fig. 12 depicts the login page of the User android app which contains user ID and login button, which is used to authenticate the User. The User ID is predefined in the database and added by the admin. Once the User is authenticated then he is allowed to move to the next page.

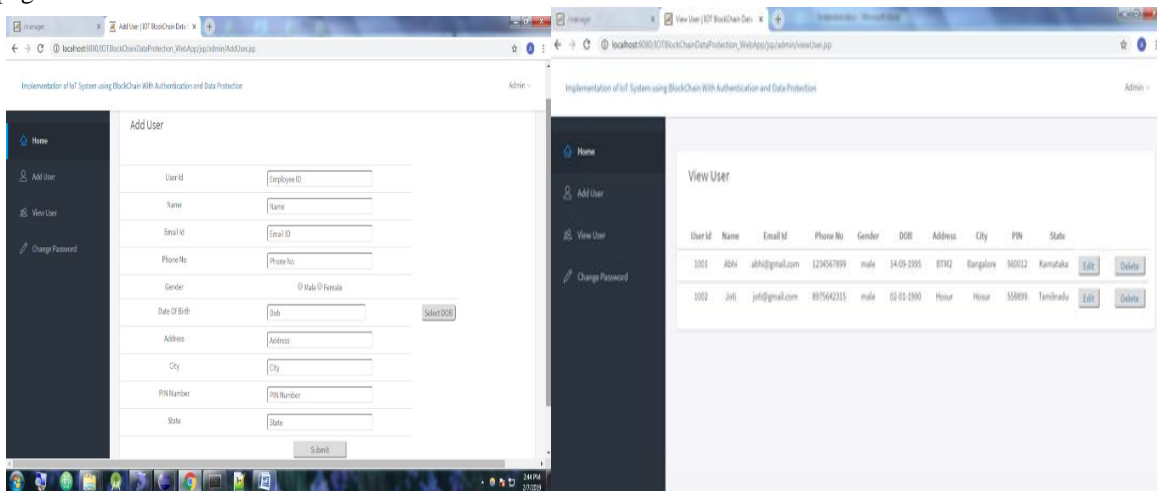


Fig.8 Admin Web App

Fig.9 Add User

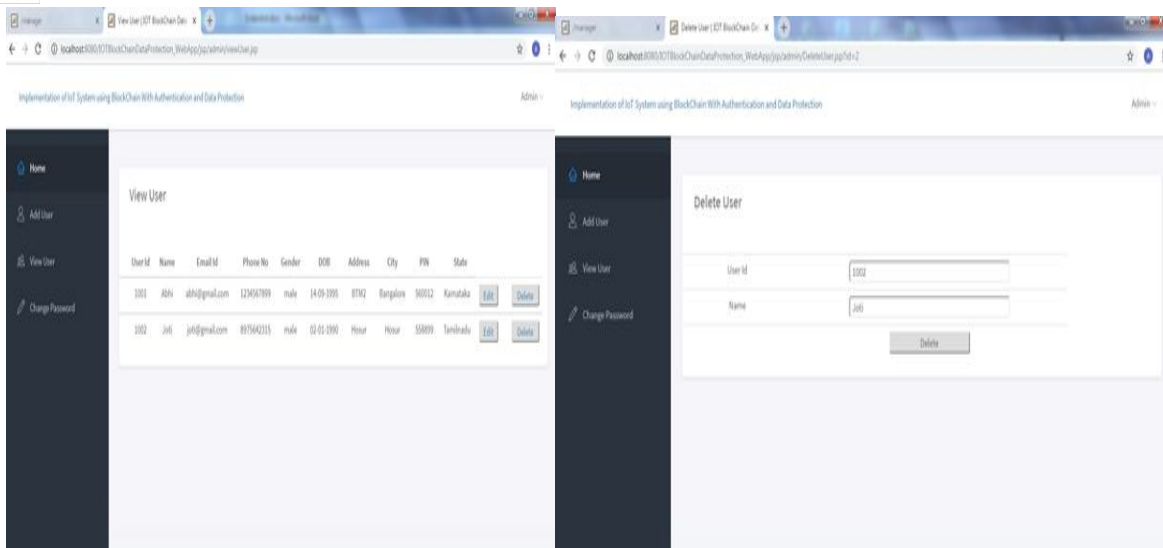


Fig. 10 View User

Fig. 11 Delete User

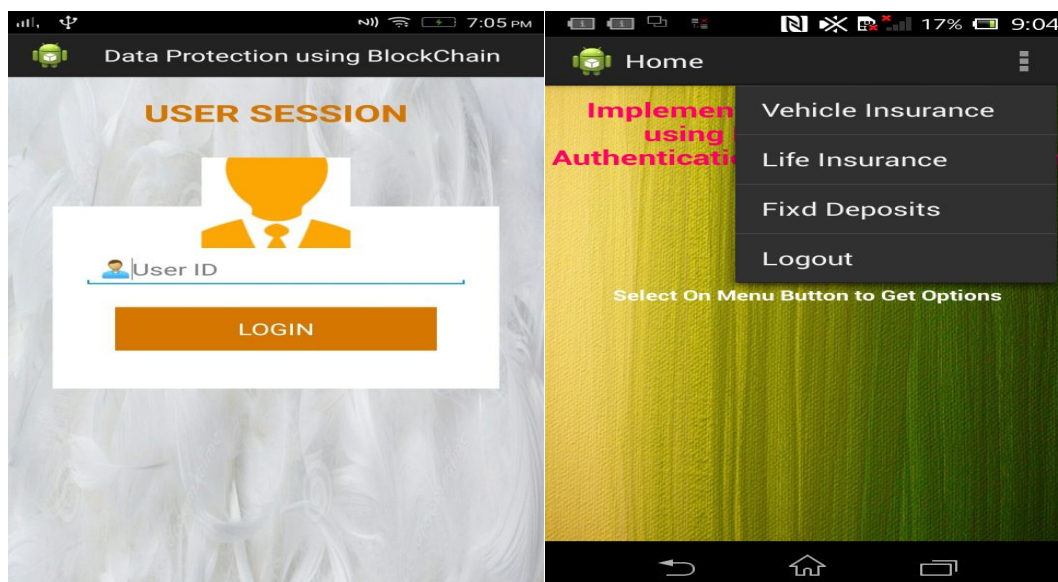


Fig. 12 User Android Login Page

Fig. 13 Home Page

F. Home Page

This fig. 13 shows the default home page of the user android application which has options like life insurance details, vehicle Details etc., when clicked on any one of the option, it will display either to view the stored information or to add the new details of the user.

G. Add Details

This fig. 14 shows Add page of the user android application which has options like life insurance details, vehicle details etc., when clicked on any one of the option, it will display the options to store the information of the FD or the vehicle details respectively.

H. View Details

The fig. 15 depicts the View details module in which the user can view the details. The information stored in the cloud will be in encrypted format. When click on to view, the file from the cloud will be downloaded to the mobile, then decrypt the information and display it to the user.

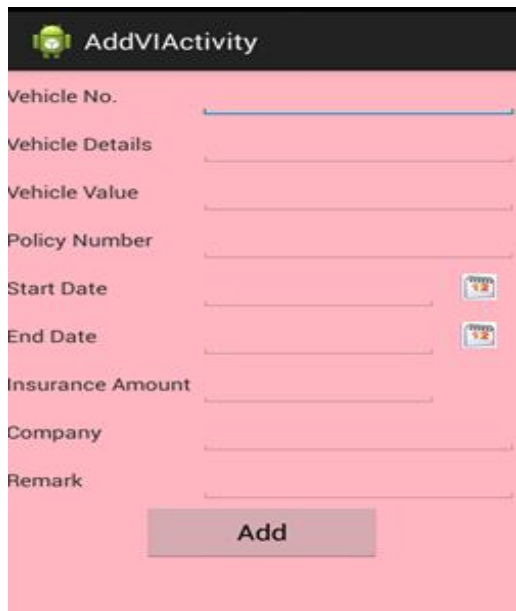
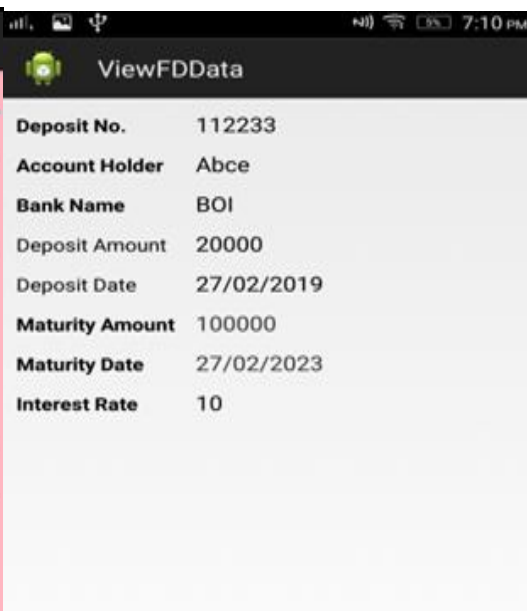


Fig. 14 Add Details



Deposit No.	112233
Account Holder	Abce
Bank Name	BOI
Deposit Amount	20000
Deposit Date	27/02/2019
Maturity Amount	100000
Maturity Date	27/02/2023
Interest Rate	10

Fig. 15 View Details

VIII. CONCLUSION

In this project, the intelligent contract system uses zero knowledge evidence to protect the data. IoT data is stored in the block chain, which can prevent authentication of IoT devices and data falsification. Zero Knowledge proof technology is applied to prevent third parties from verifying the user's original data via block recovery. The current system for measuring and billing the quantity of electricity via the smart meter applies a chain of blocks because there are various problems such as forgery and alteration of data and errors in the calculation of costs, as well as by through smart contracts with zero knowledge. The evidence makes it possible to carry out transactions such as car chargers, to presume that the energy trade is convenient and safe.

REFERENCES

- [1] Gungor, V. Cagri, et al. "A survey on smart grid potential applications and communication requirements." *Industrial Informatics*, Vol.9, No.1, 2013, pp. 28-42.
- [2] Gangale, Flavia, Anna Mengolini, and Ijeoma Onyeji., "Consumer engagement: An insight from smart grid projects in Europe.", *Energy Policy*, Vol.60, 2013, pp.621-628.
- [3] Luan, Shang-Wen, et al. "Development of a smart power meter for AMI based on Zig-Bee communication", *Power Electronics and Drive Systems*, 2009. PEDS 2009. International Conference on. IEEE, 2009..
- [4] Common Criteria for Information Technology Security Evaluation, Version 3.1, CCMB, Sep.2006.
- [5] Youngu Lee, A Study for PKI Based Home Network System Authentication and Access Control Protocol, KICS '10-04Vol.35No. 4
- [6] Kepco, Prosumer Power Trading, <http://home.kepco.co.kr>
- [7] Andreas M, Masteing Bitcoin: Unlocking Digital Cryptocurrencies, pp.49-68, O'REILLY, 2015
- [8] Sung-Hoon Lee, Device authentication in Smart Grid System using Blockchai, KAIST, 2016.
- [9] Satoshi Nakamoto, Bitcoin: A Peer-to-Peer Electronic Cash System, 2008.
- [10] Nick Szabo, Smart Contracts, 1994.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)