Project Report
on

Generating a Decentralized Architecture for Data Privacy and Sharing using Blockchain Technology

submitted in partial fulfillment of the requirement
for the award of the Degree of

Bachelor of Engineering
in
Electronics & Telecommunication Engineering

by

Akash Mahale
Anirvin Vishwanathan
Kanishka Kothari

under the guidance of

Dr. Preetida Jani

Department of Electronics & Telecommunication Engineering
Bharatiya Vidya Bhavan's
Sardar Patel Institute of Technology
Munshi Nagar, Andheri-West, Mumbai-400058
University of Mumbai
October 2018

Certificate

This is to certify that the Project entitled "Generating a Decentralized architecture for Data Privacy and Sharing using Blockchain Technology" has been completed successfully by Akash Mahale, Anirvin Vishwanathan and Kanishka Kothari under the guidance of Dr. Y. S. Rao for the award of Degree of Bachelor of Engineering in Electronics & Telecommunication Engineering from University of Mumbai.

Certified by

Dr. Preetida Jani
Project Guide

Dr. Y. S. Rao
Head of Department

Dr. Prachi Gharpure
Principal

Department of Electronics & Telecommunication Engineering
Bharatiya Vidya Bhavan's
Sardar Patel Institute of Technology
Munshi Nagar, Andheri(W), Mumbai-400058
University of Mumbai
Octobar 2018

Project approval Certificate

This is to certify that the Project entitled "Generating a Decentralised Architecture for Data Privacy and Sharing" by Akash Mahale, Anirvin Vishwanathan and Kanishka Kothari is approved for the award of Degree of Bachelor of Engineering in Electronics & Telecommunication Engineering from University of Mumbai.

External Examiner

(signature)

Name:

Date:

Internal Examiner

(signature)

Name:

Date:

Seal of the Institute

Glossary

KYC - Know Your Customer
DOS- Denial Of Service
DDOS- Distributed Denial Of Service
SHA- Secure Hash Algorithm
POW- Proof of Work
REST- Representational State Transfer
API -Application Program Interface
HTML - Hypertext Markup Language
CSS - Cascading Style Sheets
JSON - java Script Object Notation
IPFS - InterPlanetary File System
DAG - Directed Acyclic Graph

Abstract

KYC (Know Your Customer) is used in many organization such as banks, Aadhaar Card, Mutual Funds, hospital etc. It checks who are people they are dealing with before making establishing relation with them. But the problem with current KYC system is that they are outdated and very time consuming. Also they are centralized system so if the server is down then there will be loss of down time. Also Inter and Intra bank schemes are not followed. All this things leads to very high cost. To overcome all these problem our the proposed system is a decentralized system which uses Blockchain technology where the information are encrypted and can never be decrypted which in turns make it secure. The system is decentralized using IPFS (Interplanetary File System). There is no third Party involved in this system and also its cost efficient.

Contents

List of Figures

1 Introduction

Blockchain is basically a chain of encrypted blocks where each block has information stored in it with hash of the previous block.The information stored in blockchain is secured through cryptography. Hash (SHA-256) basically used to encrypt the input. Each block in blockchain consists of its own hash and the hash of the previous block. This is how chain is formed So if we have to change the information in one block we have to change it in every block. This is the reason the information stored once can never be erased.

Hash functions main advantage is it cannot be decrypted back as there is data loss. SHA-256 (Secure Hash Algorithm) is one of the latest and strongest type of Hash . It generates unique 256 bit for a text. SHA-256 is not a complex algorithm but in fact easy to implement.

The current KYC system is a centralized system. The centralized system has its own disadvantages. If the server is down then the whole system gets crashed. It is vulnerable to DOS (Denial of Service) and D-DOS (Distributed Denial Of Service) attack. Also misuse of the information is possible.

So we implement Interplanetary File System (IPFS) to decentralize the system. IPFS is an open source and Peer to Peer file system for storage system.

2 Objectives

We have tried to implement KYC using Blockchain. Peer-to-Peer file system (IPFS) is also used to decentralize the system and helps in overcome storage limitation.Hence we have implemented serverless architecture. We have tried to overcome challenges like time consumption, allowing customer to submit their documents for verification only once, irrespective of any bank they are. In Short inter and intra bank scheme is followed.

3 Current System

KYC system is used in many organization such as banks , Mutual Funds, big corporate offices, Aadhaar Card etc. KYC is mainly needed to know who they are dealing with. For eg- Banks. In banks KYC is needed to know its customers and information regarding them. Basically they are used for authentication.

Initial Process of KYC in banks.
1) Firstly clients and bank establish relationship by client sharing their document in bank such as address proof , photo identity etc.
2) Bank enters these documents in their database and look into clients status and its link with politicians, criminal activity, terrorism etc. If no match is found the bank will proceed further.
3) Bank keeps on repeating this process as the database may change over time.

3.1 Problem with the current KYC system.

1) It is very time consuming.
2) As it is a manual process it is very time consuming.
3) Also many documents need to be submitted which consists of same type of information.Example: Aadhar card, pan card, have address proof and photo proof.
4) Repeated check is done by banks to keep a track on update of information.
5) Inter Bank scheme isnt folowed.
6) Also if we change the bank we have to submit the documents all over again which becomes very tiring process.

4 Literature Survey

1) **K. Bhaskaran et al., "Double-Blind Consent-Driven Data Sharing on Blockchain," *2018 IEEE International Conference on Cloud Engineering (IC2E)*, Orlando, FL, 2018, pp. 385-391.**

Challenges like Dynamic access Control, customer consent , preserving privacy, promoting transparency etc is solved by smart contract and with components and protocol. It shows implementation of HYPER-LEDGER Fabrics and Sharing OF KYC on blockchain by Double Blind Data Share Model.

2) **W. M. Shbair, M. Steichen, J. Franois and R. State, "Blockchain orchestration and experimentation framework: A case study of KYC," *NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, Taipei, 2018, pp. 1-6.**

In this proposed case study deployment of blockchain in Grid5000 shown. It focuses on private blockchain. Once the resources are ready it is sent for deployment in blockchain and its has four tasks. Its also suggest Blockchain KYC POP Implementation by writing smart contract. Peer to Peer file system (IPFS) is used because of storage limitation.

3) **S. Sunkle, D. Kholkar and V. Kulkarni, "Model-driven regulatory compliance: A case study of Know Your Customer regulations," *2015 ACM/IEEE 18th International Conference on Model Driven Engineering Languages and Systems (MODELS)* ,Ottawa, ON, 2015, pp. 436-445..**

Integrated GRC is implemented. They have used existing techniques and for checking and norm changes. Challenges like managing of change in regulation, explanation of proofs, semantic disparity are solved using GRC.Concepts of two vocabularies are used. They have also encoded basic statement from domain knowledge in vocabulary.

5 Proposed System

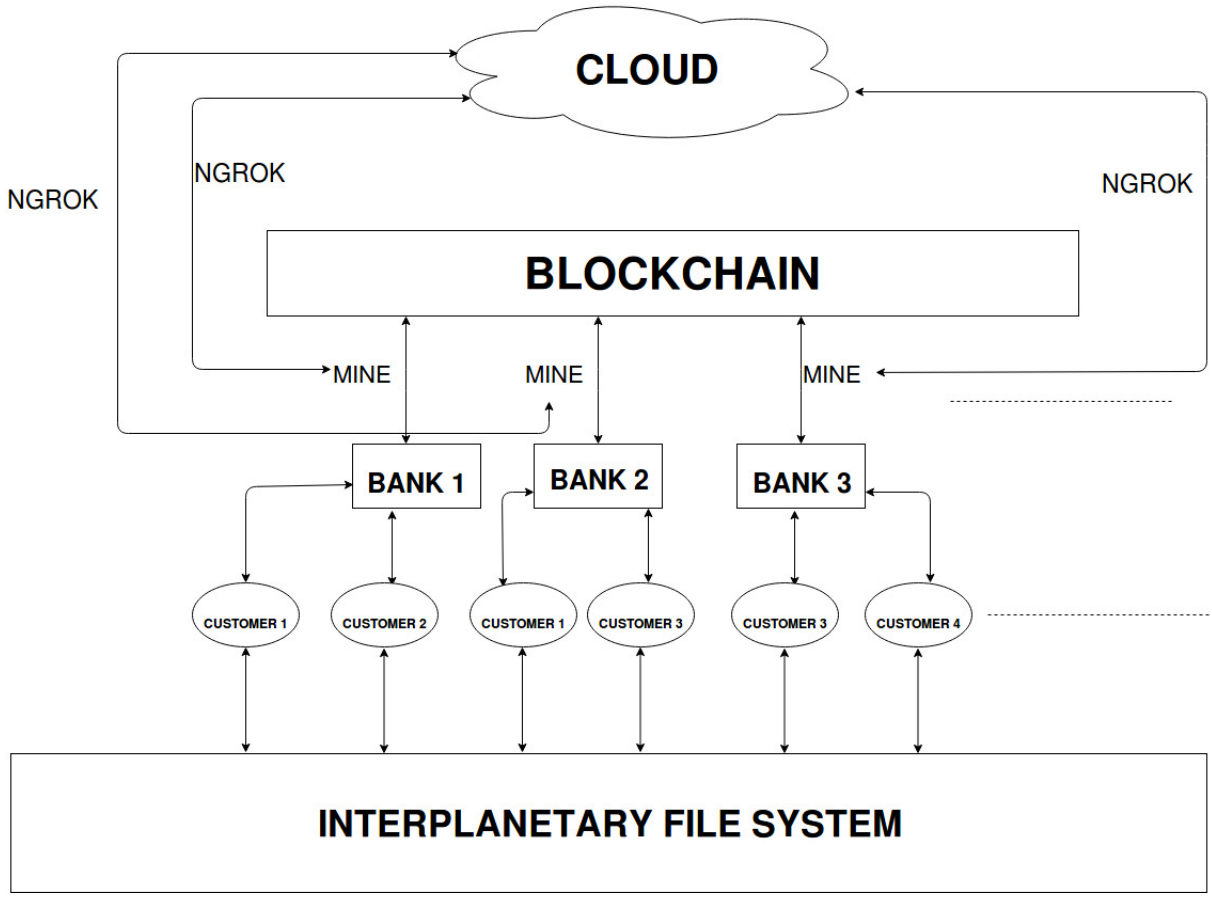

Figure 1: Proposed System.

The proposed system aims at solving the drawbacks of the current KYC system. The new system is blockchain based hence secured,cloud based hence distributed and hosting on IPFS thus ensuring decentralization. Every authorized bank will have the right to update the blockchain via the cloud service. A proper authentication and authorization will take place before updating the information and mining the blockchain. Authentication parameters like biometrics, digital signature, security questions, etc will be carried out. Thus banks will access the blockchain using a public key while updating process will require a private key. The entire application will be decentralized using IPFS. IPFS will serve as a platform to view the user data. IPFS generates a unique hash for every file system, hence this unique hash will the virtual ID to customers who update their KYC. For every change of customer information the unique hash generated will change and the corresponding new hash will stored in a database against the customers' biometrics. Thus every time the customer goes to a bank, he/she needs to provide just a unique virtual ID and biometrics. It will checked in the database and his/her KYC information will be retrieved.

6 Logical Design Of The Application

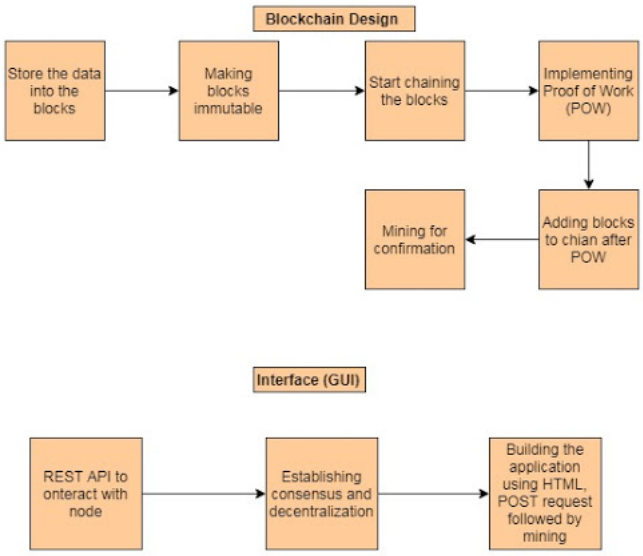

Figure 2: Software Logic Flow.

6.1 Software Design for Blockchain

a) Store the data into the block: Each block will store of the customers like name, age, address, biometrics, retina scans, etc.

b) Making blocks immutable: The blocks will be encrypted using SHA 256 algorithm. Hash function will be used making the system very safe and very secured.

c) Chaining the blocks: The blocks containing user data will be chained one after the other. Each block will contain the hash of the previous block. If by any chance the content of the previous block changes, errors will occur in the chain, thus blockchains are immutable, i.e. only new data can be appended but the old data cannot be deleted. The first block is called the genesis block.

d) Implementing Proof of Work (POW): As the discussed, the data in previous block cannot be changed, still if someone tries to do it, the hash of the block can be recomputed, but it will be computed in a different way. A certain constraints like the hash should start with two leading zeros. This is called nonce. It means, a number will keep changing until we get a hash that fulfills the constraints.

e) Adding block to the chains : Once POW is verified, only then the blocks will be added to the chain.

f) Mining : The data of the customer will be initially will stored in a pool of unconfirmed data set i.e blocks. Only when POW will be verified only then they will be mined, meaning they will be added to the blockchain.

6.2 Front End Interface:

a) REST API to interact with node : Flask library in Python is used. An endpoint to submit the data has been created. REST API is an application program interface (API) that uses HTTP or HTTPS requests to GET, PUT, POST and DELETE data. A REST API is based on representational state transfer (REST) technology, an architectural style to communicate, used in web services development and deployment.

b) Establishing consensus and decentralization : Creation of multiple nodes, across the system. It will select the longest possible chain. Everyone can update and move on to mine the data.

c) Building the application : Basic HTML,CSS, JSON is used. A POST request to a connected node is made to add data in the unconfirmed pool which is to be further mined. A refresh button will reload the blockchain with new block being added.

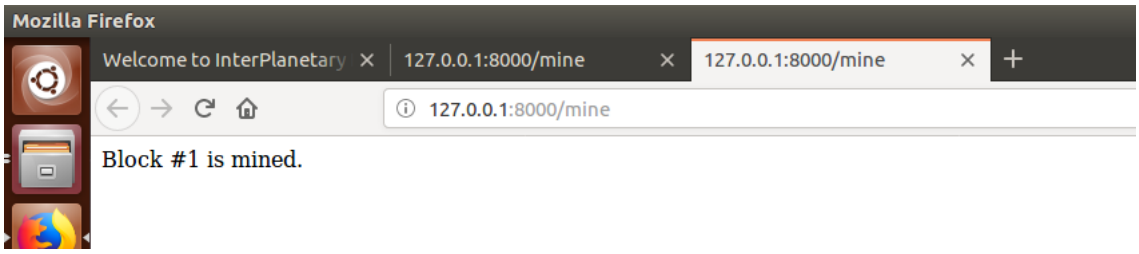

Figure 3: Mining of the First Block.

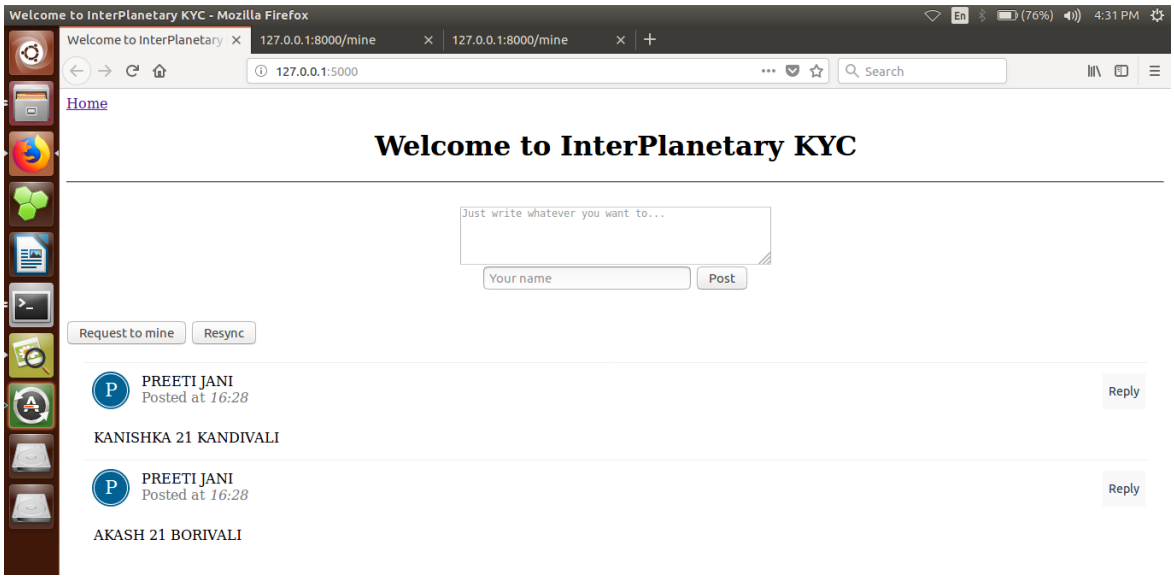

Figure 4: Front End GUI.

6.3 Achieving Distribution via Cloud:

Ngrok is an open source cloud service for hosting web application. Ngrok is an open source tunnelling platform, a reverse proxy software that helps to establish a secure tunnel from locally running network interface to global or public domain URL. It captures all the traffic for inspection and replay.

Before starting ngrok, we need to expose a localhost application to public domain. We need to deploy the local application in the DMZ and configure NATing in the firewall. DMZ prevents outside users from getting direct access to a server. Ngrok will serve the purpose of distribution via cloud so that all the authorized banks will be able to update the blockchain after verification of documents. Thus, ngrok will be a web hosting cloud service to make the localhost application accessible from public internet.

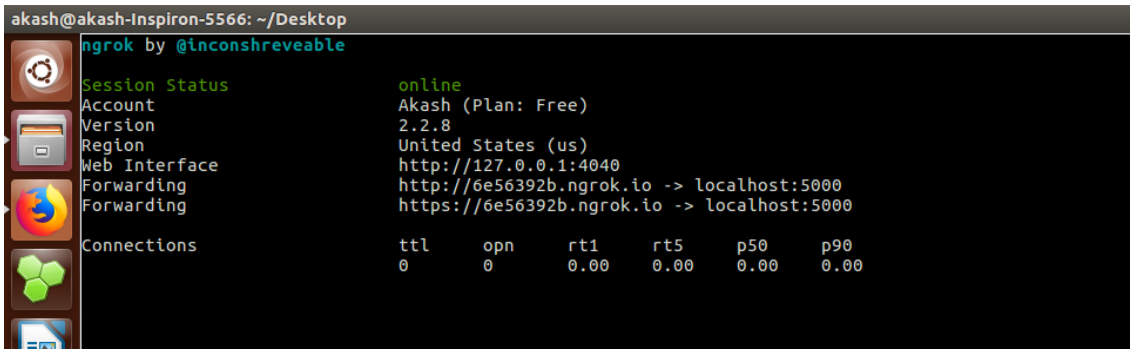

Figure 5: Ngrok setup

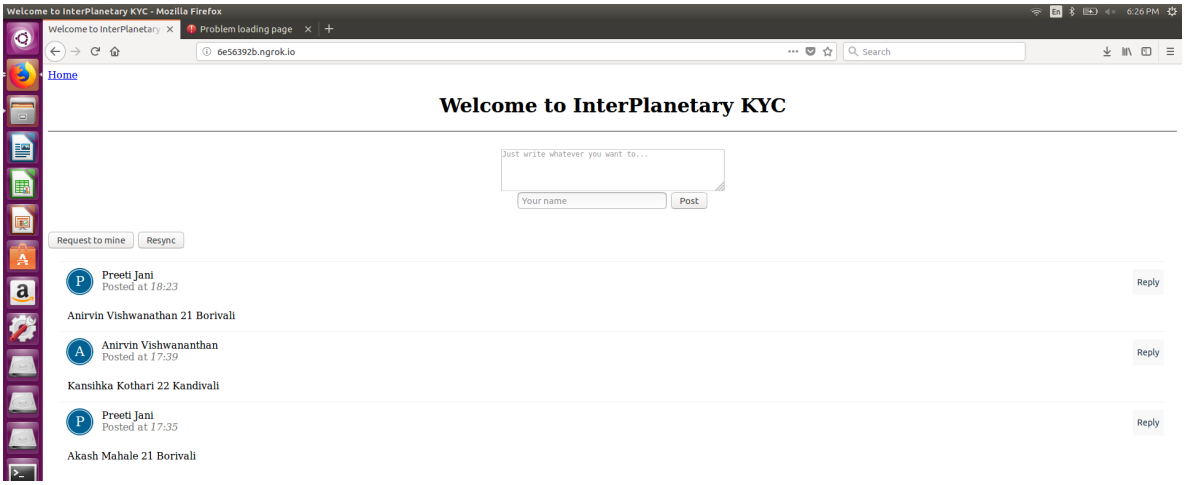

Figure 6: Accessing using ngrok URL

6.4 Achieving Decentralization:

Decentralization of the system is achieved using InterPlanetary File System(IPFS). InterPlanetary File System (IPFS) is a recent protocol which is designed to create a peer-to-peer method of storing and sharing data in a distributed file system architecture. IPFS is an open-source project developed with help from the community.IPFS is a peer-to-peer distributed file system that connects all information systems with the same system of files. IPFS is similar to the www (World Wide Web).

IPFS gives high performance, block data storage model. This forms Merkle directed acyclic graph (DAG). IPFS combines a distributed hash table. IPFS rarely has a chance of failure and information systems do not require any consensus to communicate with each other. It saves bandwidth, thus making the process fast and prevents DOS attacks which server architecture deals with.

Using IPFS we are provide every customer with a customer ID which will be the unique hash generated by the IPFS system also known as the fingerprint to identify the file. This ID can be used by customer to check their details in the database.

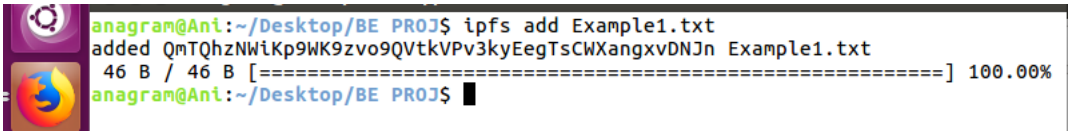

Figure 7: Unique hash generated by IPFS.

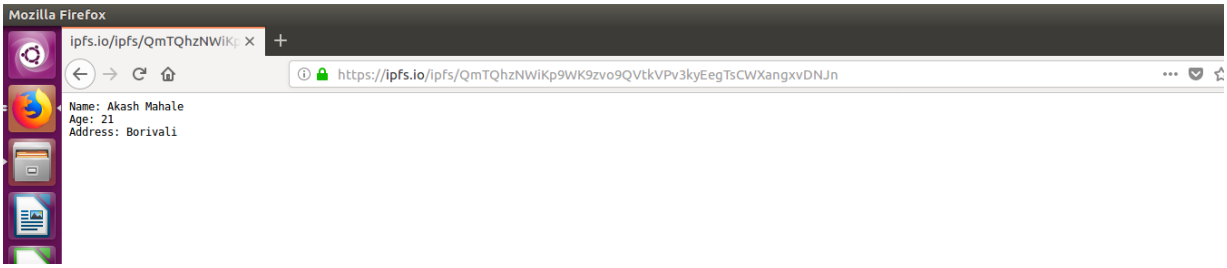

Figure 8: Globally hosted IPFS.

References

[1] K. Bhaskaran et al., "Double-Blind Consent-Driven Data Sharing on Blockchain, " *2018 IEEE International Conference on Cloud Engineering (IC2E)*, Orlando, FL, 2018, pp. 385-391.

[2] S. Sunkle, D. Kholkar and V. Kulkarni, "Model-driven regulatory compliance: A case study of Know Your Customer regulations," *2015 ACM/IEEE 18th International Conference on Model Driven Engineering Languages and Systems (MODELS)*, Ottawa, ON, 2015, pp. 436-445.

[3] P. C. Mondal, R. Deb and M. N. Huda, "Know your customer (KYC) based authentication method for financial services through the internet," *2016 19th International Conference on Computer and Information Technology (ICCIT)*, Dhaka, 2016, pp. 535-540.