DistB-OT: A Distributed Secure Blockchain based Online Ticketing System using Hyperledger Fabric

Submitted to the Department of Computer Science and Engineering of Mawlana Bhashani Science and Technology University in partial fulfillment of the requirements for the degree of B.Sc. in CSE.

Submitted by

Md. Tamim Dari Chowdhury (CE-16021)

Mahbub Alam (CE-16042)

Supervised by

Mohd. Sultan Ahammad

Department of Computer Science and Engineering

Department of Computer Science and Engineering

Mawlana Bhashani Science and Technology University

Tangail - 1902, Bangladesh

September, 2021

Dedicated to

Our Parents

Abstract

Meanwhile, Online Ticketing System has become an emerging technology for the rapid growth of the internet and internet devices. Most of the systems use centralized control and database systems. We introduce a hyper ledger-based online ticketing platform for Bus, Railway, and Airway which is a decentralized, permissioned, secured, and highly scalable system using blockchain technology. We use Hyperledger Fabric (A framework by IBM) to design a blueprint of an architecture, which enables ticketing among all participating organizations, where all rights and rules are maintained by author organizations.Retail or vending organizations share access with the same ticket ledger. This platform offers railway, bus, and airway passengers various ways to issue and pay ticket fees for different types of transportation using digital methods as well as legacy systems.We Offer a combined system that is hassle-free and simple to use for passengers and related organizations

Keywords: Blockchain, Hyperledger fabric, Privacy, Security, Ticketing, Railway, Bus, Airway, Distributed system

Acknowledgment

First and foremost, at this very special moment, we would like to express our sincere gratitude to the Almighty Allah for helping us to complete this Bachelor's degree. We are very grateful not only throughout our study time, but also during our lives, for the tremendous blessings the Almighty has conferred upon us.

We have been through the experiences with and assistance of other people in achieving this goal, and would like to express our heartfelt gratitude to those who have contributed to this work.

Md. Tamim Dari Chowdhury
Mahbub Alam
September, 2021

Contents

List of Figures

List of Tables

List of Acronyms

CA - Certificate Authority

CC - Chain Code

Org - Organization

Ord - Orderer

MSP - Membership Service Provider

DLT - Distributed Ledger Technology

EMR- Electronic Medical Records

BCT - Blockchain Technology

HLT - Hyperledger Fabric Technology

UIIP - Uniquely Identifiable Patient Information

GB - Genesis Block

HF - Hash Functions

HIE - Health Information Exchange

API - Application Programming Interface

HTTP - Hyper Text Transfer Protocol

IOT - Internet of Things

M2M - Machine-to-Machine

P2P - Peer-to-Peer

PBFT- Practical Byzantine Fault Tolerance

CFT- Crash Fault Tolerant (CFT)

MQTT - Message Queuing Telemetry Transport

OEM - Original Equipment Manufacturer

OTA- Over The Air

RFID - Radio-Frequency Identification

Chapter 1

Introduction

Chapter 1

Introduction

1.1 Overview

Due to the rapid development of the global internet and the availability of internet devices, the internet is being applied to nearly all aspects of human life. However, more people than ever before are getting connected to the internet. Following the trend, the internet also has been used in the Railway, Bus, and Airline industry across the globe. Digitization of the old legacy system to cope up with the modern era paved the way for online ticketing.

As ticketing entered the online era, passengers have multiple options to buy tickets for transportation. Along with traditional paper ticketing services now passengers can book, validate and retrieve bus tickets through the web and mobile applications of the corresponding bus service providers. Developed countries adopted the online ticketing concept in their railway systems very well. For example, The national railway in the UK introduced smart cards, e-ticket, and m-ticket along with the traditional paper ticket to modernize the ticketing process. Developing countries like Bangladesh are adopting them-ticket similar to the UK where the tickets purchased online are directly sent to mail accounts. Air ticketing also follows a similar process where users buy and cancel tickets through a website or mobile applications.

This research addresses the potential of blockchain-based hyperledger fabric technology as a mechanism for digital ticketing.

1.2 Motivation

A recent survey says that about 40 % of total passengers use an online ticketing system to reach their destination. The number is very low because of the inconvenient experience and low secured volatile database in our ticketing system. The 40 % of users face a poor experience.

Most of them have to face in-place ticket validation problems because of the mismatch of analog and digital systems. The is a big question in terms of transparency and trust issues. And all the ticketing system is scattered. It would be great a platform if all the ticketing system has the same platform to purchase a ticket.

1.3 Significance

The possibility of using blockchain for online ticketing data management has recently raised a lot of attention in both industry and persons. A few operating implementations of a system using blockchain for the control of ticket data have been suggested. In our work, we concentrate on a practical implementation of a system using Hyperledger fabric: a distributed operating system for permitted blockchain together with docker containers to demonstrate a multiple channel that can be incorporated with several transportation systems for the use of virtual to real peers. Hyperledger fabric is also the first blockchain system that runs distributed programs written without systemic dependence on a native crypto-currency in standard, general-purpose programming languages. This stands in sharp contrast to existing block-chain platforms that require ”smart-contracts” to be written in domain-specific languages or rely on a cryptocurrency. Using a portable notion of membership, which can be incorporated with industry-standard identity management, Hyperledger fabric realizes the permitted model. Fabric introduces an entirely new blockchain design to promote such flexibility, and revamps the way blockchains deal with non-determinism, resource exhaustion, and performance attacks. The combination of these technologies enables us to guarantee data security and privacy as well as availability with regard to the patient-defined access control policy. We suggest several scenarios for online ticketing system blockchain based solutions and evaluate current iterations of technology that may be used to bring the scenarios into effect. We developed a prototype, and with further modifications and improvements, the functionality of the prototype is expected to meet the requirements from the perspective of a conventional ticketing platform in a single system.

1.4 Problem Statements

Several researchers have recommended a number of clarifications to improve online ticketing systems, but they can not solve the problem fully. Most of them give a solution for a single platform like railway or bus or air. And most of them are in a single channel that's why several platforms can not share similar data among them. Hyperledger Fabric can be a solution for online ticket systems for all kinds of transportation platforms in a single platform. It is possible to list the challenges as follows:

- In which way can this research improve the existing platform?
- What are the benefits of transport organizations by joining a Blockchain network?
- What are the extra benefits a passenger can get from it?

Multiple blockchain technology, an immutable, encrypted, distributed ledger technology, has recently been used. This offers a unique opportunity to build a stable and efficient blockchain-based online ticket data management. Hyperledger is one of the Blockchain frameworks for this kind of enterprise solutions. In addition, the first fully extensible blockchain system for running distributed applications is Hyperledger Fabric.

1.5 Objectives

The goal of this study is to build a system to buy air, bus and railway tickets in a single platform to lessen the problems of passengers using the Hyperledger fabric that can privately share data across all channels in the network. This provides a unique opportunity to create an optimised, stable and powerful data management and sharing system based on blockchain. The following contributions are found in our work:

- To apply Hyperledger Fabric for more trustable and secured data transfer and management.
- To develop a single platform where all transportation ticketing system are available.
- To develop a Blockchain based system for efficient performance by merging all the databases that were previously used for individual platforms.

1.6 Contributions

In this work, Docker containers are used to create a prototype in the local environment to demonstrate a multiple client network. We followed the principles of Hyperledger fabric that is a distributed operating system for permissioned blockchains. Fabric is a modular and extensible open-source system for deploying and operating permissioned blockchains and one of the Hyperledger projects hosted by the Linux Foundation[1]. In this work, the prime contributions incorporate the following key points:

- Dealing with the implementation challenges to initiate a Hyperledger based blockchain network and generating required cryptographic material
- Analysis of the suitability of hyperledger network in real world implementation and applicability of this implementation to securely maintain and share important and significant medical records.
- Comparative assessment of the existing system or proposed systems for sharing medical records and our proposed hyperledger based medical record sharing

1.7 Organization

The remainder of this research is organized as follows:

- **Chapter 2 Background Study** Introduces the background of this thesis in detail of Blockchain, classification of Blockchain,Function, Advantages of Blockchain, Application of Blockchain HyperLedger fabric, Security Concerns of Blockchain Technology.
- **Chapter 3 Literature Review** Highlights the previous research works on Electronic Medical Record Sharing and proposed blockchain based Medical Record Sharing as well.
- **Chapter 4 Methodology** Illustrates the model of the proposed blockchain based network of this research work and it's working principle in details.
- **Chapter 5 Simulation Design** Introduction with the simulation tools and presents the topology design of proposed network architecture.
- **Chapter 6 Result Analysis and Discussion** Presents simulation results and taken decisions.
- **Chapter 7 Conclusion and Future Work** Concludes the whole research work and focuses on future research directions.

Chapter 2

Background Study

Chapter 2

Background Study

2.1 Introduction

Background Study or Technical Preliminaries has been discussed in this chapter. This chapter includes key technology such as the Blockchain Technology (BCT) and Hyperledger Fabric.

2.2 Blockchain Technology (BCT)

Blockchain was selected as one of the key technologies to lead the fourth industrial revolution era at the 2016 World Economic Forum. Global market research institutes, Gartner, and Deloitte also selected Blockchain as one of the 2017 technology trends [2] . The inherent technology of Bitcoin, which was created in 2008, is the blockchain. A blockchain can be defined as an immutable ledger for recording transactions, maintained within a distributed network of mutually untrusting peers. Because of its existence, it is called a distributed ledger. Ledger, for instance, consists of different blocks of different data transactions that are connected with each other using cryptographic methods such as hashing. The distributed and immutable list of elaborate blocks keeping data as a transaction is Blockchain [3] which has shown in Fig. 2.1 .

Most importantly, a distributed network that ensures accountability among Blockchain users is evenly shared by ledgers. If the ledger has been attached to a block, it will be immutable. If the transaction is shared by every participating node, chains collapse between them.

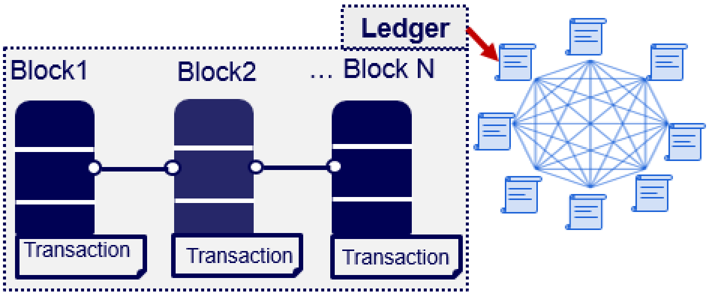


Figure 2.1: Distributed and Immutable Block chain

It is possible to divide Blockchain into two groups, such as public and private Blockchain [4] . In the public blockchain, after being accepted, everyone can connect. The individuality of the participants is at another time. The public blockchains are Bitcoin and Etherum. Due to a great deal of anonymity, it gets slower. That is another reason, in the desired framework, the public Blockchain does not provide strong privacy and confidentiality. On the other hand, in the private Blockchain, invitation or permission is required to connect. It has no anonymity as identification is already known, which creates private Blockchain faster. Above all, by special procedures, privacy and confidentiality can be achieved. On the other hand, in the private Blockchain, invitation or permission is required to connect. It has no anonymity as identification is already known, which creates private Blockchain faster. Above all, by special procedures, privacy and confidentiality can be achieved. Finally, the Blockchain principle as decentralized, open-source, transparent, autonomous, immutable, etc. is concluded. The digital currency (Bitcoin), smart contract, hyperledger, smart cities, grids, houses, and smart building control system are the application of Blockchain technology [5].

2.2.1 How Blockchains Work

A Blockchain is a distributed database system that is replicated and partitioned among the members of a system. Blockchain technology can be described as a trustless and fully decentralized peer-to-peer data storage that is spread over all participants that are often referred to as nodes [6] . The Blockchain is remembered as a log whose records are batched into time-stamped blocks. By its cryptographic hash, each block is marked. Each block refers to the hash of the preceding block. This authenticates a link between the blocks, creating a block chain, or the blockchain shown in Fig.2.2.

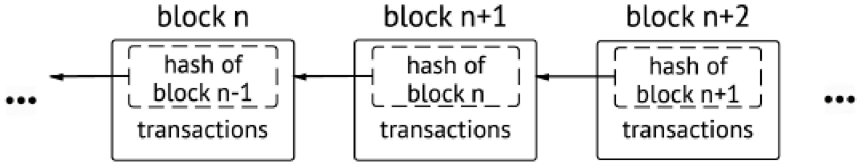


Figure 2.2: Working Procedure of Blockchain

When data is committed to the chain, the blockchain is built to carry immutable information and is thus a decentralized, distributed and immutable ledger. So we can tell Blockchain is organized logically as a sequence of smaller chunks (blocks). Each block $B_{i>0}$ is immutably connected to a single preceding block B_{i-1} through a cryptographic hash function $\mathrm{H}(B_{i-1})$. Changes to B_{i-1} would yield an invalid hash in B_i and all following blocks. The very first block B_0 is called the genesis block and is the only block without a predecessor. Any node that has access to this existing block can read it and determine the global status of the data being replaced on the network. In order to assure the integrity of a block and the data contained in it, respectively, the block is usually digitally signed.

When we explore how a blockchain network operates, we get a deeper understanding of how a blockchain functions. This is a group of nodes (clients) that switch on the corresponding Blockchain via the unique copy that one accommodates. For various different Blockchain users, a node may also serve as an entry point into the tracks, but for creativity, we assume that each user performs via their node on the network. These connections form a peer-to - peer network where :

1) A transaction must happen. Let 's begin with the instance of Amazon's impulsive buy. We go against our best judgement and make a purchase after hastily scrolling through several checkout prompts. In certain instances, a block will theoretically group thousands of transactions together, so our Amazon purchase will also be packaged in the block along with the transaction details of other users.

2) It must check the transaction. Each blockchain network needs to decide certain rules that should be followed by each database transaction. Each Blockchain client processes these application-dependent rules, and then uses them to select whether an incoming operation is legitimate and, therefore, whether or not it should be transmitted to the network. However, with blockchain, the job is left up to a com-

puter network. When we make our Amazon order, the computer network rushes to verify that our transaction occurred in the way we said it did. That is, they confirm the acquisition information, including the time, dollar amount, and participants of the transaction.

3) It is important to store the transaction in a block. It gets the green light after the transaction has been checked as correct. The dollar sum of the purchase, the digital signature, and the digital signature from Amazon are all stored in a block. There, the contract is likely to come with hundreds, or thousands, of others like it.

4) A hash must be given to that block. Once all transactions of a block have been checked, a specific, identifying code called a hash must be given to it. The hash of the most recent block added to the blockchain is also provided to the block. The block can be added to the blockchain when hashed.

When that new block is added to the blockchain, it becomes publicly available for anyone to view.

2.2.2 Types of Blockchain

Blockchain technologies can be roughly divided into four categories.

2.2.2.1 Public Blockchain

The public blockchain is a completely open source. Anybody can connect with as a client, designer, or network part. The Public blockchain is additionally straightforward, any of the public blockchain client can try or demonstrate the exchange subtleties but cannot modify the exchange information data.

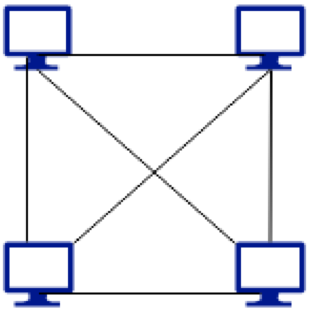


Figure 2.3: Public Blockchain

It is completely decentralized on the grounds that nobody controlling the exchange. Even though anybody can join the public blockchain network paying little mind to area or ethnicity specialists can't close down this record since it is enough hard. The Public blockchain resembles Bitcoin, Ethereum. Fig. 2.3 shows public blockchain.

2.2.2.2 Private Blockchain

In private blockchain, anybody cannot take an interest. It has a few limitations from the specialists to the board or admittance to the information. Fig. 2.4 shows the private blockchain. Private blockchains are valuable implied for organizations who might want to team up and advance information however do not want their specific delicate association data uncovered on the public blockchain. These kinds of chains, by their temperament, will be more focal. That substances working that chain have indispensable control of individuals and afterward administration frameworks. Private blockchains may have a token related with that chain. Private blockchain are utilizing in supply chain management, Digital personality, Hyperledger, Vote counting, Asset license, and so forth.

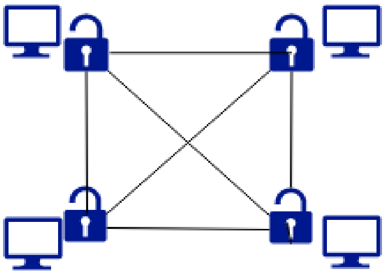


Figure 2.4: Private Blockchain

2.2.2.3 Consortium Blockchain

Picked clients in the consortium can work or run a total node, can make exchanges, and review the blockchain. Here with respect to one out of forcing, you may have different in control. You may have a gathering of companies or maybe partner those individuals returning overall and afterward delivering choices to locate the best bit of advantage of the whole systems administration. These sorts of groupings are called consortium or a federation. In the Fig. 2.5 shows the consortium blockchain. Members on consortium blockchain can comprise of any individual originating from banks, to really government specialists and supply chain management.

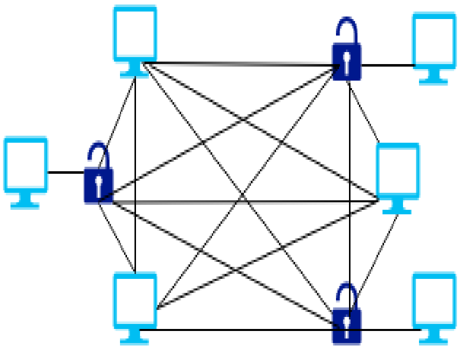


Figure 2.5: Consortium Blockchain

2.2.2.4 Hybrid Blockchain

Hybrid blockchain is the mixer of centralization and decentralization and it is likewise engaged with the various gatherings, information accesses the executives, authorization, information sharing, dependable and rub. The entirety of the blockchain like private, public and consortium experiences some difficulty and preferred position. This blockchain is easier to work the business for his simple usefulness without losing their security and protection. It is likewise more adaptable and straightforward for the business reason to keep the information into the hidden and what kinds of information they need to disclose. In Fig. 2.6 shows the hybrid blockchain. The present Hybrid blockchain is utilizing into the IoT, Supply chain and general information insurance guideline.

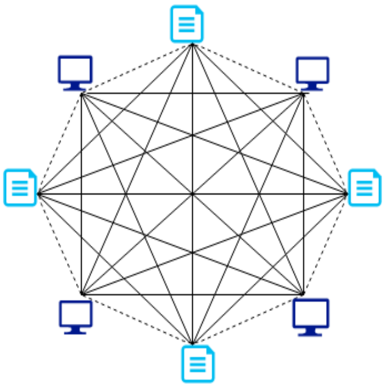


Figure 2.6: Hybrid Blockchain

2.2.3 Benefits of Using Blockchain Technology

- Security: Blockchain is imagined to be an incredibly solid framework because of its advanced stamp and encryption. The framework is explicitly intended to be secure, flexible, and sealed.

- Quack control: A framework that depends on information spared in various spots is protected to programmers; it isn't so easy to secure access to it, and assuming this is the case, any portion of information can be easily recovered.
- Transparency: Banks, just as the customers, are in a flash advised about the culmination of exchanges, which is both open and solid.
- No concealed expenses: Overlook about charges and commitments as the framework is decentralized, there is no compelling reason to pay delegates.
- Access levels: Users need to pick between open Blockchain accessible for anybody and the solidarities expecting authority where every node ought to be engaged first for the client to enter.
- Speed: Sales are devised route quicker than expected as there is no compelling reason to incorporate change frameworks, which lessen the expense and improves the preparing speed.
- Record assuagement: The legitimacy of activities is shown and verified by cooperators, in this way, they additionally avow their own validness.

2.2.4 Application of Blockchain Technologies

BC advances are utilizing in numerous territories appeared in Fig. 2.7 the monetary applications as well as in different businesses.

2.2.4.1 Bitcoin

Bitcoin is a digital currency which also known as crypto-currency not supported by the central bank or government of any nation. Bitcoins can be exchanged for products or services with merchants who recognize Bitcoins as payment. Bitcoin-to-Bitcoin transfers are rendered by digitally swapping secret, strongly secured hash codes through a peer-to-peer (P2P) network. The P2P network tracks and verifies the transmission of bitcoins between users. Each Bitcoins user is held in a technology called a digital wallet, which also contains every address the user sends and receives from Bitcoins, as well as a secret key known only to the user.

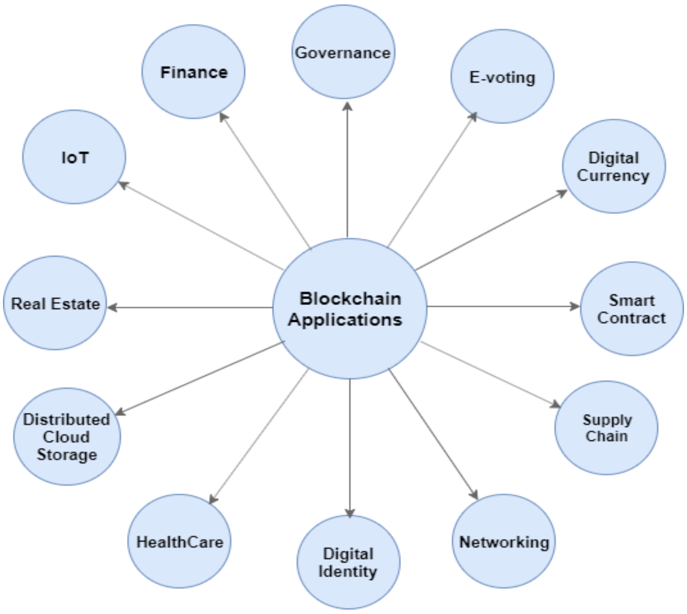


Figure 2.7: Blockchain Applications

2.2.4.2 Blockchain in supply chain management

All physical goods should take a path from work to purchaser and this journey is called supply chain. Supply chains include dynamic producer, manufacturer, dealer, retailer, auditor, and customer networks. The decentralized IT architecture of a blockchain would streamline workflows for both participants regardless of company network scale. The road to the customer is not easy. There are often hundreds of intermediaries involved in this process. Blockchain technology can theoretically enhance accountability and traceability problems within the supply chain of manufacturing through the use of immutable data recording, distributed storage, and managed user access [7] . Additionally, a collaborative infrastructure will allow auditors more insight in the supply chain operations of members.

2.2.4.3 Blockchain in E-Voting

E-casting is a democratic decision that casts an electoral frame where consumers cast their votes in an advanced setting for a secure, accurate and enigmatic Internet voting. Due to the namelessness, protection, unchanging efficiency, respetability and accessibility of E-casting, many countries worldwide use the electoral system. As a decent contestant, the blockchain with its brilliant agreements has increased to improve the security, cost, privacy, ease and simplicity of e-casting for voting [8] . Using Blockchain, the main focus, to make the voting process fair and without any unwanted interference. There are various steps in the administration of the E-casting Voting System. One of them, Ethereum, is an unbounded organization to express those software. The fundamental concern of E-Voting is to ensure the personality of the client while at the same time maintaining the simplicity and integrity of the details. In order to resolve this problem, Ethereum gives clients diverse hash values in the organization from which it is basically difficult to identify the entity, and the transactions made in the Ethereum network are obvious to all in the organization [9] and can be accepted to make it convenient for all hubs in the organization and to preserve the trustworthiness of the organization. Simple knowledge is lost in the distribution of information, in particular in areas that make information permanent and incredibly difficult to monitor.

2.2.4.4 Blockchain in healthcare issues

The usage of IoT (Internet of Things) and blockchain in clinical consideration is increasingly expanding alongside the production of clinical knowledge on current occasions. To handle safe and trustworthy transfers of information on the sharing and logging of clinical details. Protected Health Information (PHI), which is the fundamental consideration of countless data is made. Solid partnerships focused on Blockchain to facilitate healthy investigation and clinical sensor boarding [10] . Ensure knowledge confidentiality for patients and health professionals by entering into a sound deal. To the point that clinical knowledge is increasingly expanding alongside clinical sensors. Primary management, the continuous gathering of records is central to the system for patient treatment. Blockchain will bring an end to this security and other poor problems in the system of medical services.

2.2.4.5 Smart Contract

A smart contract is a tiny programming program that is concealed in a blockchain. This is an arrangement between two or more individuals in the context of a programming code. They function on the blockchain; programming codes are kept in a shared database and cannot be modified. Blockchain has processed transactions that exist under a smart contract. They can be submitted automatically without a third party.

2.2.4.6 Hyperledger

Hyperledger is a distributed open-source care that has been built into previous blockchain technology. The Foundation is home to the Linux Foundation which comprises supply chains, retail, distribution, IoT, finance and technology executives. Hyperledger can simply be considered as a software application that helps developers around the world to build applications for individual organizations on a blockchain basis.

2.2.4.7 Blockchain in IoT

The Internet of Things (IoT) provides a number of possibilities for remote control surveillance and then connects ordinary artifacts in many domains by means of applications [11] . In smart industries, smart transport, healthcare, military things, frontline things, etc., IoT is commonly used. Another revolution that is transforming our environment is the IoT. Privacy and confidentiality are a big concern of IoT networks because of their oppressive size and dispersed existence. For protected IoT platforms, many frameworks were proposed, such as lightweight instantiation by extracting job proof. Now BC with IoT merging protected IoT networks [12] .

2.2.4.8 Others applications

Besides the above applications and BC used in healthcare, banking, industry, government such as wealth management, e-voting, insurance, supply chain sensors, public community, BC identity and personal recognition etc.

2.2.5 Security Issues and Challenges of Blockchain

In 2008 to work presently, BC is generally utilizing for its security and conveyed framework. In any case, it has hardly any security issues. These security issues are communicated constantly, and Cyber-assaults forces indispensable deficiencies on it [13].

2.2.5.1 Majority attacks of 51%

The ability of mining with the proof of work depends on the miner. Peer to peer working capacity or ability of mining 50% or more while the attacker is 51,%, the attacker running the hash block [13]. If the intruder adjusts transaction records, a twin attack takes place, the transaction ceases immediately and the miner's operation ends. Attacker 25% illicit profits and have bogus G-hash trade of 50%.

2.2.5.2 Fork problems

Another issue is the fork problem. It is part of a decentralized variant of the node contract as automated programs are modified. It is now an important issue across a large variety of blockchains. Fork challenges are classified into two distinct types: hard fork and soft fork. When the new block was written, a consensus algorithm modified the rules. Therefore nodes may be split into two forms of blockchain: old nodes and new nodes. The new nodes consent or do not agree with the deal with the existing nodes. Around the same moment, old nodes approve or do not agree with the exchange with the latest nodes.

- Hard Fork: A hard fork is a major shift inside an incompatible cryptocurrency protocol, a node that does not upgrade the current Node version cannot process or insert any blocks into the cryptocurrency block. It uses a modern, separate protocol and blockchain to modify or enhance an established protocol.
- Soft fork: A soft fork is simply a modification of a back-compatible cryptocurrency specification where the transfer and new blocks to the network may be processed through non-updated nodes.

2.2.5.3 Scalability

In addition to the amount of transactions, the blockchain gets stronger each day. In every node, all transactions must be saved to check whether the root of the recent transaction is normally unused. Each node needs to save all transactions. The Bitcoin network is capable of handling about 7 transactions per second [14] . This is why often minor transactions postponed as mine employees demand that they pay the transactions at a large transaction charge. It cannot control or process the million transactions in real time because the scale of a block volume is small.

2.2.5.4 Time confirmation

Since the p2p carriage transaction of decentralization is Blockchain. Yet there is the matter of time for proof. For validation transactions [15], each transaction took an average period of 20 to 40 minutes. The maximum period used on optimization shows that the usage of algorithms reduces [16] by 71.42of the time.

2.2.5.5 Selfish mining

Another problem of BC technologies is greedy mining. A block is typically susceptible to cheating where a limited amount of hazing is used. The miners retain the mined blocks without transmittal to the networking device during their selfish mining and then create a personal component, which should be broadcast immediately after precise specifications have been met. And in this situation, trustworthy miners invest much time and money, whereas individual chains can be mined by greedy miners [17] .

2.2.5.6 Social Engineering

Social innovation is a challenge for security problems in Blockchain. All sorts of methods of social engineering rely on the vulnerabilities of human psychology. To exploit and trick their victims, scammers take advantage of emotions. Via a multitude of strategies, people's fear, envy, interest, and even their ability to support others are turned against them. Among the multiple sorts of malicious social engineering, phishing is certainly one of the most common and well-known examples. Scareware and Baiting are another known methods in social engineering [18] .

- Phishing: The intruder sends a Bogus URL or website links or social applications to the target customer that you can trust after a business or agency brand, a national bank chain, a reliable online retailer, or an email supplier. They ask your account to have any protection problems that we are, authority people, you should give us your details that links we given urgently or your account will be blocked. They can build friction. If you deliver your information, you've missed something.

- Scareware: Scareware is malware intended to alarm and shock users. They usually include generating false alarms that try to deceive victims into downloading fake, legitimate-looking apps, or visiting a website that infects their device. Such a tactic also depends on users' apprehension of sacrificing their device, enticing them to press a site banner or popup.

- Baiting: Baiting is another social engineering tool that makes many inattentive users problems. It includes utilizing baits to attract victims based on envy or interest. For example, scammers may build a website providing anything free, like music files, images, or books. But to access these archives, users must build an account, providing their personal information. In other instances, no account is required since files are specifically compromised with ransomware that penetrates the victim's operating device and captures their confidential data.

Chapter 3

Literature Review

Chapter 3

Literature Review

3.1 Introduction

In this section, We have mentioned some literature reviews of recent works which are given below:

3.1.1 Blockchain Technology (BCT)

Blockchain is an immutable, encrypted, distributed ledger technology. In this paper Shi-Cho Cha et al [19]. have discussed potential problems compared with other conventional ticketing systems, data integrity, authentication of purchased tickets, and security pitfalls. He also provided a secured and privacy preserving ticketing service. It has some key drawbacks like there is no solution for malicious ticket validators

Problems with existing systems are identified and alternatives based on blockchain technologies are presented as solutions. The author in this paper [20] presented three potential applications for blockchain technology within the railway industry: digital ticketing; logistics and supply chain processes.

Michael Kuperberg et al. have discussed First-of-its-kind blockchain based Billing and prototype implementation for online ticketing system [21]. They addressed the confidentiality, stability, and scalability of the proposed system. The proposed blockchain based approach which simplifies usage billing and enables a train-to-train/machine-to-machine economy. They also addressed the use of blockchain as a life cycle approach for condition based monitoring and predictive maintenance in train operations are outlined.

J. D. Preece et al [22]. proposed a method using IBM's Hyperledger Fabric framework to design an architecture that distributes the tickets across all participating organisations. They noted the potential benefits this platform has. Governing organisations maintain their right to set the rules of the platform and access the data to generate statistics. Vending organisations share access to the same underlying tickets whilst preserving competition. The platform offers passengers a variety of ways to pay for and access their tickets, using a combination of legacy and modern methods. Furthermore, we note the platform has the potential to eradicate paper ticketing and surplus voucher cards.

Hyperledger Fabric is a ”permitted” blockchain architecture that offers a transparent distributed ledger that must all have the same view of its state, shared by a group of peers.It is a distributed operating system which is modular and extensible. F. Benhamouda et al. suggested a solution in which peers encrypt their private data and use safe MPC if such private data is necessary in a transaction, until it is stored on the chain. To help their approach, the authors identified two basic services that should be applied to Hyperledger Fabric [23].

In another paper, Elli Androulaki et al. described Fabric, its architecture, the reasoning behind different design decisions, its most influential implementation aspects as well as the programming model of distributed application. Via implementation and benchmarking a Bitcoin-inspired digital currency, the authors further test fabric. They illustrated that the fabric accomplishes more than 3500 transactions per second per end-to - end throughput with sub-seconds in certain common deployment configurations, scaling well to over 100 peers latency [24].

Table 3.1: Summarized Table of Literature Reviews

Paper reference	Method	Findings	Limitation
Shi-Cho Cha et al. [19]	Etherium blockchain	Compared with other ticketing system and provided a secured and privacy preserving ticketing service	There is no solution for malicious ticket validator
J. D. Preece et al. [20]	Hyperledger, Ethereum	Integrated blockchain within railway industry: digital ticketing; logistics; and data interoperability and distribution	No solution for volatile cryptocurrency market.
Michael Kuperberg et al. [21]	Blockchain	First-of-its-kind blockchain based billing and prototype implementation.	High workload on chain.
J. D. Preece et al. [22]	Hyperledger Fabric	Drawbacks of legacy ticketing systems and offers a better secured and privacy focused railway ticketing system.	Platform only for Railway service and No real time stress testing.

Chapter 4

Methodology

Chapter 4

Methodology

4.1 Introduction

Before talking about the platform architecture and its functions we need to consider some essential conditions. From the passenger's perspective, the platform must have all existing ticketing options. Both online and onsite ticket purchasing options must be preserved. As many passengers may not want to purchase through the online process. Passengers do not need to register in air, bus, and railway separately. That means they will only register once on the platform to be able to purchase tickets of all transportation. Additional information like passport details may be required only when a passenger is willing to book or buy air tickets to travel to a different country. But this must be processed only once. There should be an offline ticket verification process available, when connecting to the World Wide Web(WWW) is not possible. From an organization perspective, passengers must have the option to choose the organization from which they want to purchase bus, railway, or air tickets. Prices of bus tickets for similar routes must be the same for all the organizations depending on the service. For ac service, non-ac service price must be the same. Organizations must have the option to offer discounts to passengers to attract them. For railways, there are no vendor options in Bangladesh.So the price is always same but for railways with vendor organizations prices must be the same. Similar considerations must be taken for air ticketing.The information must be distributed between the organizations, which means one organization must know that another organization already sold the bus or air ticket or not. But the privacy of private information between these organizations must be kept secret.

4.2 Choosing Hyperledger Fabric over traditional blockchain network

Hyperledger Fabric supports distributed ledger solutions on permissioned networks for a wide range of industries. Its modular architecture maximizes the confidentiality, resilience, and flexibility of a network system.Fabric is open-source system for deploying and operating permissioned-blockchains and one of the Hyperledger projects hosted by the Linux Foundation. We must choose between implementations that are permissionless and a permitted blockchain. Below, we present the facts that favor the implementation of a permitted system.

- Permissionless blockchain is public that means virtually everyone can participate. This type of network is fully decentralized across unknown parties. Typically, permissionless blockchains use a ”mined” native cryptocurrency or transaction fees to mitigate this lack of trust in order to provide economic incentives to offset the extraordinary costs of participating in a form of Byzantine fault tolerant consensus based on ”proof of work (PoW).” On the other hand, permitted blockchains operate a blockchain among a set of known, identified and often validated participants operating under a model of governance that yields a certain degree of confidence.

- A permitted blockchain provides a way for a group of entities that have a common goal but may not fully trust each other to secure their interactions. A permitted blockchain can use more traditional crash fault tolerant (CFT) or byzantine fault tolerant (BFT) consensus protocols that do not require costly mining by relying on the identities of the participants.

- In addition, the risk of a participant intentionally introducing malicious code via a smart contract is decreased in such a permitted context. The guilty party can easily be identified and the incident handled in accordance with the terms of the governance model, rather than being completely anonymous.

- As ticketing includes real-time operation, our systems need to be very responsive. Moreover, the system should be able to handle a lot of users so, it needs to be scalable. IBM's hyperledger fabric has all these qualities required for our system design.

- Ticketing data along with NID information of a passenger is very sensitive. So, our system must ensure no information leak and there is some restrictions needed to ensure privacy. Hyperledger fabric is suitable for this type of secure information handling.

4.2.1 Hyperledger Fabric approaches in a network system

In fabric information is updated or deleted in a transaction scheme. It introduces a new architecture for transactions that we call execute-order-validate. It identifies the issues of resilience, adaptability, usability, efficiency and integrity faced by the order-execute model by dividing the transaction flow into following three stages.

1. Perform a transaction and verify its correctness, thus endorsing it

2. Order transactions through a consensus protocol and

3. Validate transactions against an application-specific endorsement policy before committing them to the ledger

Fabric is the first blockchain technology that enables use of standard programming languages. Hyperledger Fabric has been specifically architected to have a modular architecture.At a high level, the following modular components are comprised of fabric:

- A pluggable ordering service creates agreement on the order of transactions and then transmits blocks to peers.

- A provider of pluggable membership services is liable for associating network entities with cryptographic identities.

- Smart contracts (”chaincode”) operate for isolation within a container environment (e.g. Docker). They can be written in standard programming languages but do not have direct access to the ledger state.

- The ledger can be configured to support a variety of DBMSs.

Several research papers have been published to study and test Hyperledger Fabric performance capabilities [25].

4.2.2 An example of sample network to understand process of hyperledger based network system

To understand the hyperledger fabric approach for a simple network following steps with figures are provided.

- **Creating the Network:** Let's start at the beginning by creating the basis for the network:

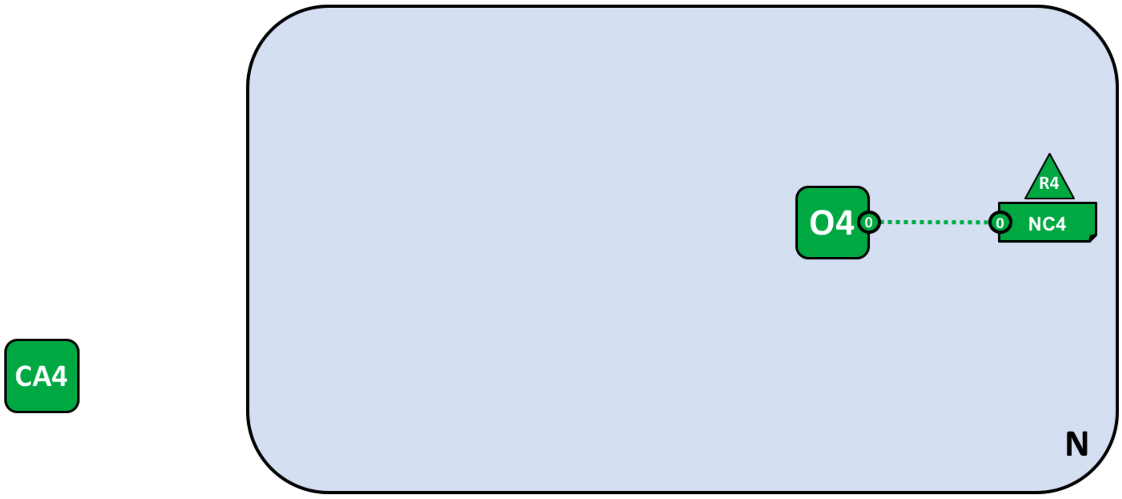


Figure 4.1: Network base

The network is formed when an orderer is started. In our example network, N, the ordering service comprising a single node, O4, is configured according to a network configuration NC4, which gives administrative rights to organization R4. At the network level, Certificate Authority CA4 is used to dispense identities to the administrators and network nodes of the R4 organization.

- **Certificate Authorities**: We can see a Certificate Authority, CA4, which is used to issue certificates to administrators and network nodes. CA4 plays a key role in our network because it dispenses X.509 certificates that can be used to identify components as belonging to organization R4. Certificates issued by CAs can also be used to sign transactions to indicate that an organization endorses the transaction result – a precondition of it being accepted onto the ledger.

- **Adding Network Administrators:** NC4 was initially configured to only allow R4 users administrative rights over the network. In this next phase, we are going to allow organization R1 users to administer the network. Let's see how the network evolves:

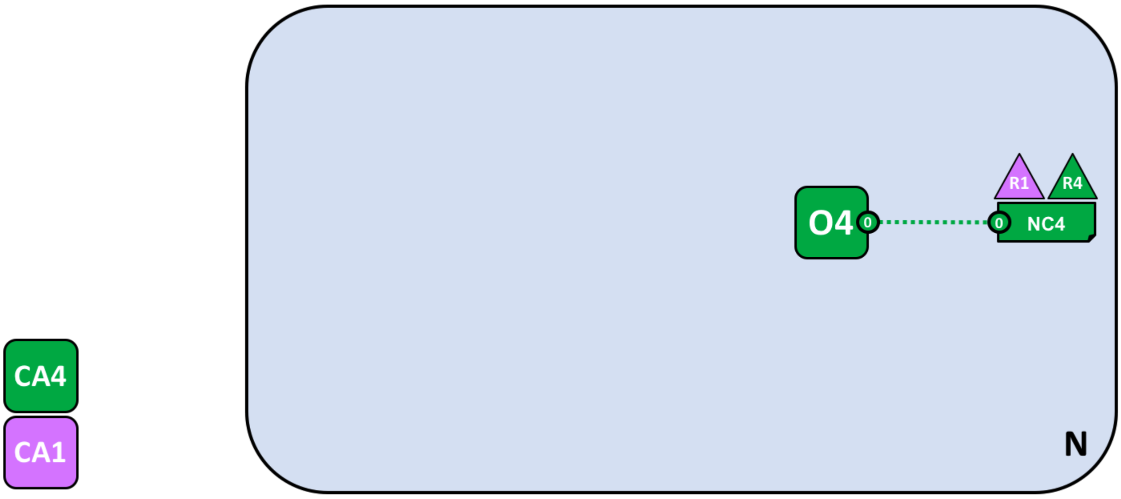


Figure 4.2: Adding network administrators

  Organization R4 updates the network configuration to make organization R1 an administrator too. After this point R1 and R4 have equal rights over the network configuration.

- **Defining a Consortium**: Although the network can now be administered by R1 and R4, there is very little that can be done. The first thing we need to do is define a consortium. This word literally means "a group with a shared destiny", so it's an appropriate choice for a set of organizations in a blockchain network. Let's see how a consortium is defined:

  A network administrator defines a consortium X1 that contains two members, the organizations R1 and R2. This consortium definition is stored in the network configuration NC4, and will be used at the next stage of network development. CA1 and CA2 are the respective Certificate Authorities for these organizations.

- **Creating a channel for a consortium:** So let's create this key part of the Fabric

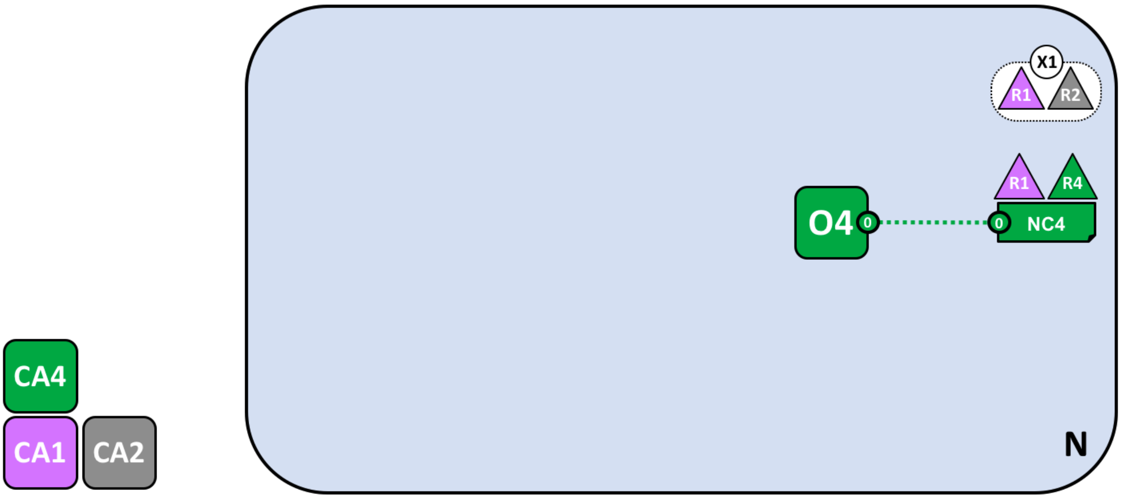


Figure 4.3: Defining a Consortium

blockchain network – a channel. A channel is a primary communications mechanism by which the members of a consortium can communicate with each other. There can be multiple channels in a network, but for now, we'll start with one. Let's see how the first channel has been added to the network:

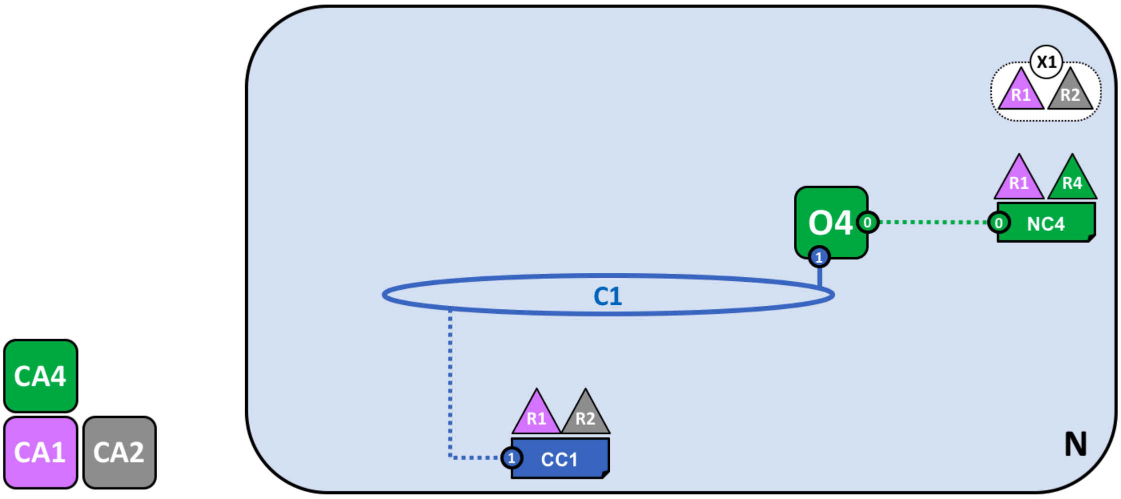


Figure 4.4: Creating a channel for a consortium

A channel C1 has been created for R1 and R2 using the consortium definition X1. The channel is governed by a channel configuration CC1, completely separate to the network configuration. CC1 is managed by R1 and R2 who have equal rights over C1. R4 has no rights in CC1 whatsoever.

- **Peers and Ledgers:** Let's now start to use the channel to connect the blockchain network and the organizational components together. In the next stage of network development, we can see that our network N has just acquired two new components, namely a peer node P1 and a ledger instance, L1.

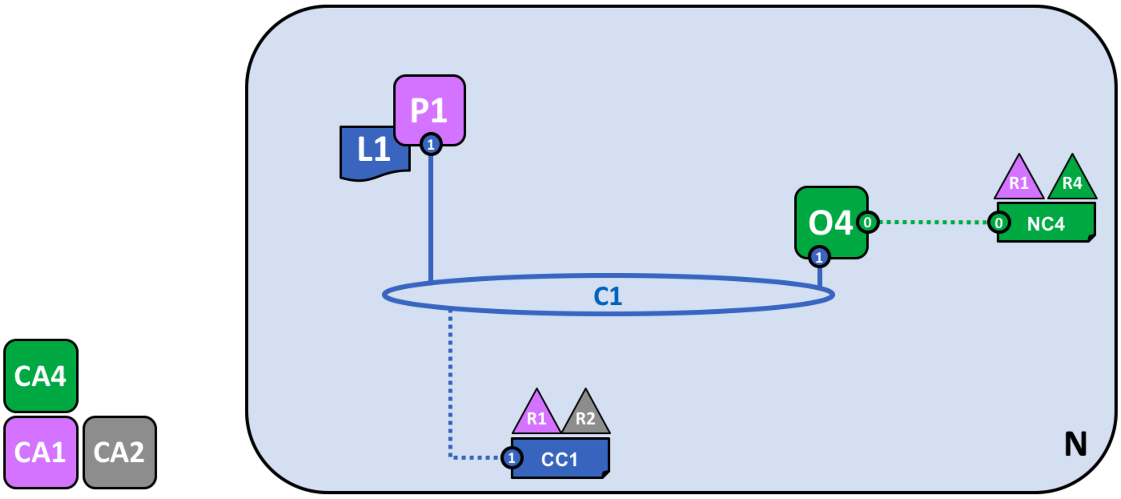


Figure 4.5: Peers and ledgers

A peer node P1 has joined the channel C1. P1 physically hosts a copy of the ledger L1. P1 and O4 can communicate with each other using channel C1.

- **Applications and Smart Contract chaincode:**

Now that the channel C1 has a ledger on it, we can start connecting client applications to consume some of the services provided by workhorse of the ledger, the peer! Notice how the network has grown:

A smart contract S5 has been installed onto P1. Client application A1 in organization R1 can use S5 to access the ledger via peer node P1. A1, P1 and O4 are all joined to channel C1, i.e. they can all make use of the communication facilities provided by that channel.

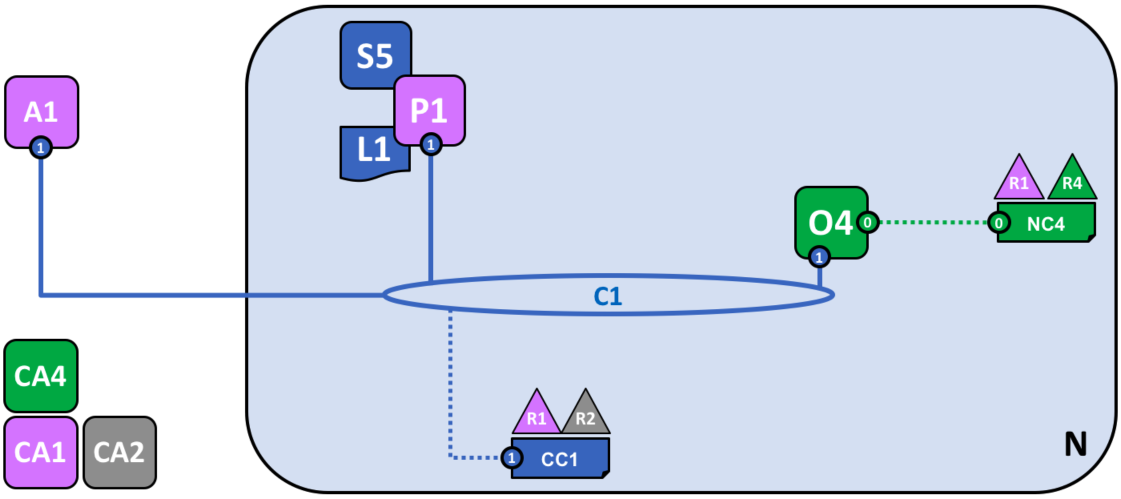


Figure 4.6: Applications and smart contract chaincode

- **Installing a chaincode package:**

  After a smart contract S5 has been developed, an administrator in organization R1 must create a chaincode package and install it onto peer node P1. This is a straightforward operation; once completed, P1 has full knowledge of S5. Specifically, P1 can see the implementation logic of S5 – the program code that it uses to access the ledger L1. We contrast this to the S5 interface which merely describes the inputs and outputs of S5, without regard to its implementation.

- **Defining a chaincode:** Although a chaincode is installed on the peers of individual organizations, it is governed and operated in the scope of a channel. Each organization needs to approve a chaincode definition, a set of parameters that establish how a chaincode will be used on a channel. An organization must approve a chaincode definition in order to use the installed smart contract to query the ledger and endorse transactions. In our example, which only has a single peer node P1, an administrator in organization R1 must approve a chaincode definition for S5.

- **Endorsement policy:** The most important piece of information supplied within the chaincode definition is the endorsement policy. It describes which organizations must approve transactions before they will be accepted by other organizations onto their copy of the ledger. In our sample network, transactions can only be accepted onto ledger L1 if R1 or R2 endorse them.

- **Invoking a smart contract:** Once a smart contract has been installed on a peer node and defined on a channel it can be invoked by a client application. Client applications do this by sending transaction proposals to peers owned by the organizations specified by the smart contract endorsement policy. The transaction proposal serves as input to the smart contract, which uses it to generate an endorsed transaction response, which is returned by the peer node to the client application. We can see that organization R1 is fully participating in the network. Its applications – starting with A1 – can access the ledger L1 via smart contract S5, to generate transactions that will be endorsed by R1, and therefore accepted onto the ledger because they conform to the endorsement policy.

- **Network completed:** Recall that our objective was to create a channel for consortium X1 – organizations R1 and R2. This next phase of network development sees organization R2 add its infrastructure to the network. Let's see how the network has evolved:

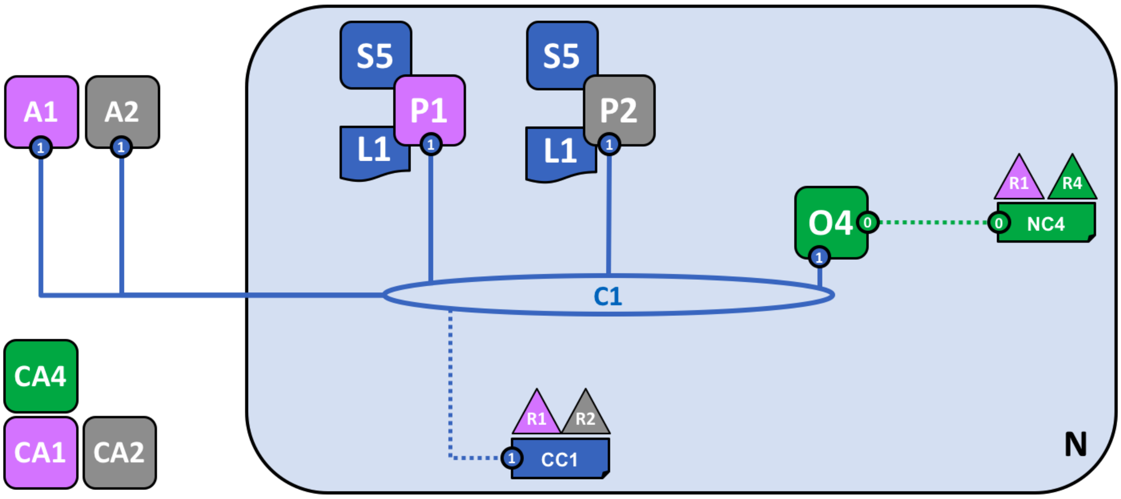


Figure 4.7: Complete Network

The network has grown through the addition of infrastructure from organization R2. Specifically, R2 has added peer node P2, which hosts a copy of ledger L1, and chaincode S5. R2 approves the same chaincode definition as R1. P2 has also joined channel C1, as has application A2. A2 and P2 are identified using certificates from CA2. All of this means that both applications A1 and A2 can invoke S5 on C1 either using peer node P1 or P2.

4.3 Proposed Methodology

In this section we provided the platform architecture and fuctioning along with data flow diagram of the system network.

4.3.1 Platform Architecture

IBM's commercial distribution of Hyperledger Fabric is a platform for distributed ledger solutions. From figure 1 we can see the network architecture, which uses the Hyperledger Fabric framework.

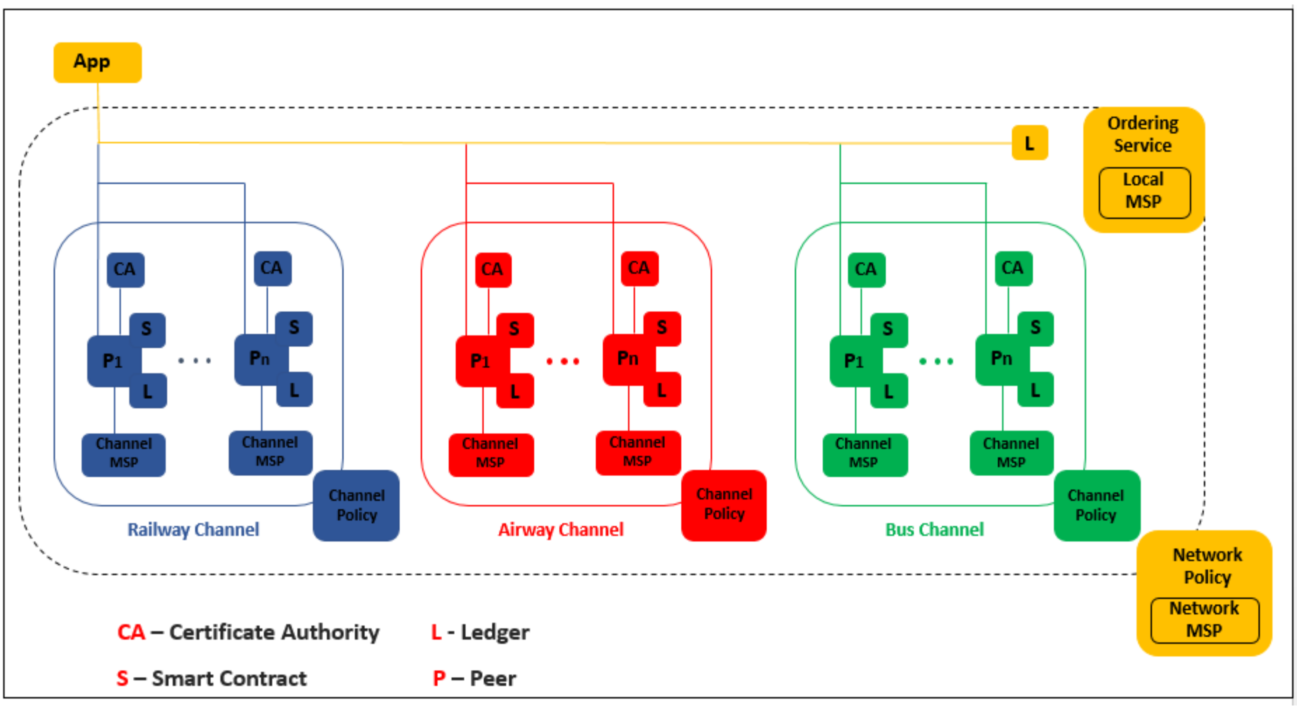


Figure 4.8: Platform architectre

At first, a participating organization initiates the network. This organization is most likely to have the governing power of the whole network. Let's say the railway is the initial organization to have all the governing power. It then adds Airway and Bus Transport corporation as two other governing nodes. Now all three organizations share equal administrative power. These organizations then initiates,

- a network policy, which hand over governing power to these three initial organizations and an MSP(Membership Service Provider)

- a CA(certificate authority), which provides a digital identity in the network
- an ordering node who do the job of ordering and packaging transactions into blocks of the blockchain. There can be multiple ordering nodes based on the network workload.

Each time they need to update any of the initial governing policies, all of them must agree on that change by giving consensus among them. After initializing the network with policies, CA, and ordering nodes these initial organizations (Airway, Railway, and Bus transport Corporation) create three-channel for each of them. That is Airway, Bus transportation, and Railway will have control over there corresponding channels.

Now every organization can add peers (organizations) to their channels. From figure 1 we can see that there can be multiple peers to each channel. Channels allow data isolation and confidentiality by acting as a sub-network inside the hyper ledger fabric network. All peers in a channel hold the same information, that is they share the same ledger. Peers invoke smart contracts which are executables logics that are necessary for a transaction to take place and so that it can be added to the ledger.

Each peer organization also can provide an MSP to sub-peers (in this case the passengers). MSP of a Legit organization is accepted by all other peers taking part in a channel. Channels also have the ability to share information with other channels but in a limited way. That's why when once a passenger registers through a valid organization they don't need to provide the same information to others. From figure 1 we can see that each peer organization can have its own web or mobile applications and passengers can use these to buy tickets from them.

4.3.2 Platform Functioning

Any passenger who wants to purchase a ticket must register to the platform and have an identity. The platforms stores the passenger identity as an X.509 digital certificate. Let's take an example to understand digital certificates. Say, a person belongs to a country and this person has a government identity card that provides information about him/her. The government identity card is proof that that person is a citizen of the country. The digital certificate used in this platform works similarly to that government identity card which is used to prove key facts about a passenger on the platform. Cryptography, a mathematical technique is used to present passenger certificates to others(organizations) to prove his/her identity so long as the other party trusts the certificate issuer(CA). After storing identity a smart contract provided by the governing organization is revoked to validate the identity creation. It checks whether a passenger's information matches the information requirement of the channel or not. Applications used by the peer organization can invoke the smart contract by giving a request to the governing organization. Governing organization stores the MSP of the recent joining passenger. Channel-to-channel MSP sharing shares this MSP to other channels. Thus the same identity of a passenger is shared across the entire network. Passengers having network digital identity can purchase tickets by requesting through applications used by the peer organizations. If the request fills the requirement set through smart contracts of the peer organization, this information is then recorded as a valid transaction on the channels distributed ledger. Tickets are sent to the passenger's digital identity. The ordering node maintains the requested order and helps to record information into the ledger utilizing blocks holding transactions. While performing validation applications are used and a transaction request is sent to peer organizations. Again smart contracts are invoked for the validity check of a transaction in a channel ledger. In simple words, whether a seat of any transportation is available or not, whether a passenger has a valid ticket or not etc.

4.3.3 Data Flow of the Platform

To understand our platform better we provide a data flow diagram of our platform. Data flow follows the following steps,

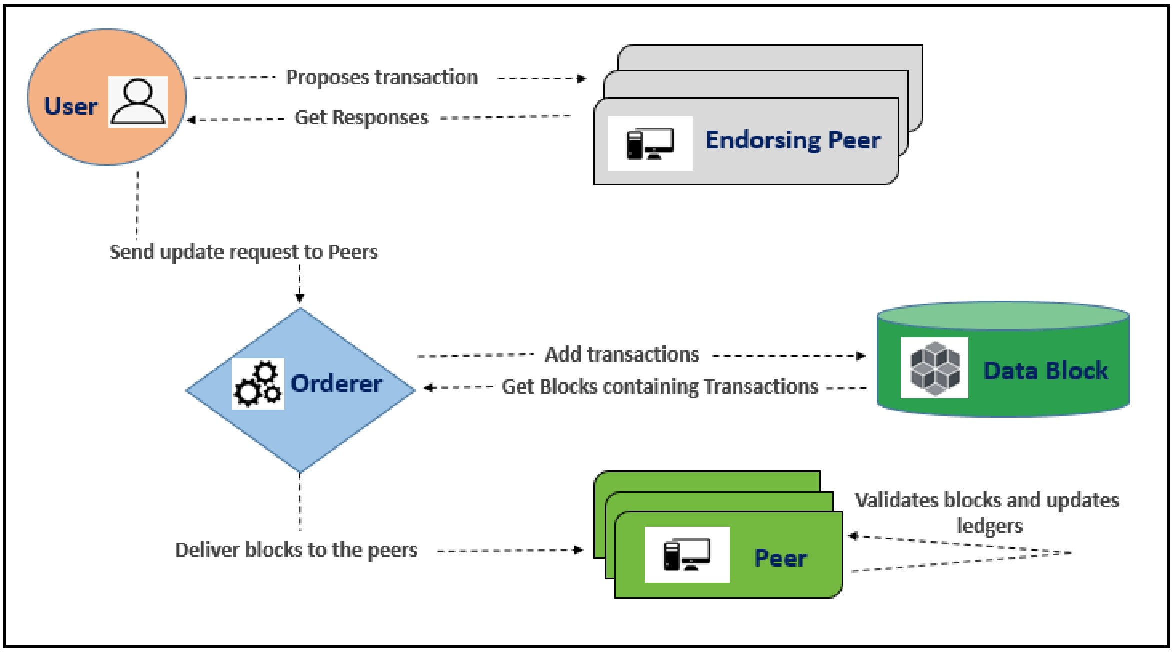

User
Proposes transaction
Get Responses
Endorsing Peer
Send update request to Peers
Orderer
Add transactions
Get Blocks containing Transactions
Data Block
Deliver blocks to the peers
Peer
Validates blocks and updates ledgers

Figure 4.9: Platform architectre

- User proposes transaction to endorsing peers of the network.
- Each endorsing peer responses whether transaction request is valid or not, user takes this response and makes a transaction request to the network.
- Orderer works as a middleman and gets this transaction request. After that it makes blocks containing transaction requests and delivers blocks to the committing peers.
- Committing peers validates received blocks and updates ledgers.

Chapter 5

Running Fabric Test Network

Chapter 5

Running Fabric Test Network

5.1 Introduction

We used a test network provided by IBM Hyperledger Fabric and did a test run. This test network is not a template for deploying a production network but it shows that our platform architecture will work if we use Hyperledger Fabric. It includes,

- It includes two peer organizations and an ordering organization.
- For simplicity, a single node Raft ordering service is configured.
- To reduce complexity, a TLS Certificate Authority (CA) is not deployed. All certificates are issued by the root CAs.
- The sample network deploys a Fabric network with Docker Compose. Because the nodes are isolated within a Docker Compose network, the test network is not configured to connect to other running Fabric nodes.

5.2 Prerequisites

To run our test network, first we need to set up our environment. The following prerequisites are required to run a Docker-based Fabric test network on our local machine.

- **Git:** The latest version of git if it is not already installed.
- **cURL:** The latest version of cURL
- **Docker:** The latest version of Docker Desktop and the docker image of test network provided by the Hyperledger fabric. Docker Desktop must be launched to complete the installation so be sure to open the application after installing it.

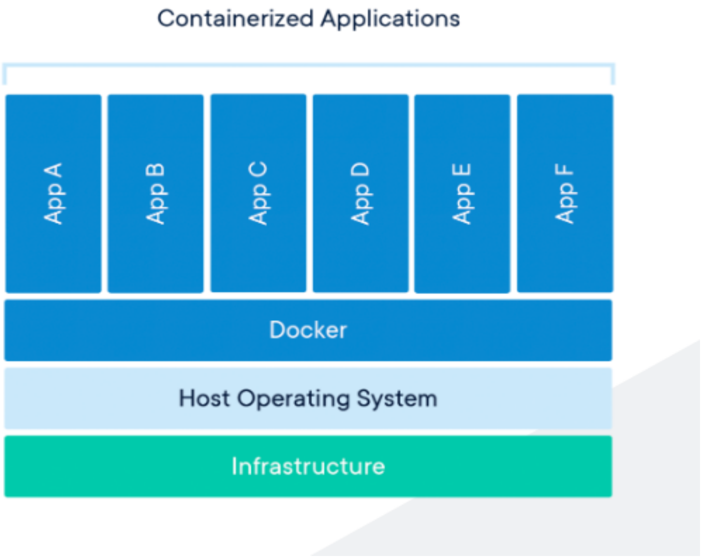


Figure 5.1: Docker Conceptual View

We need a basic concept of docker. A docker container is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another. A Docker container image is a lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, system libraries and settings.

5.3 Deploying the network

Lets bring up the test network and test basic implementations.

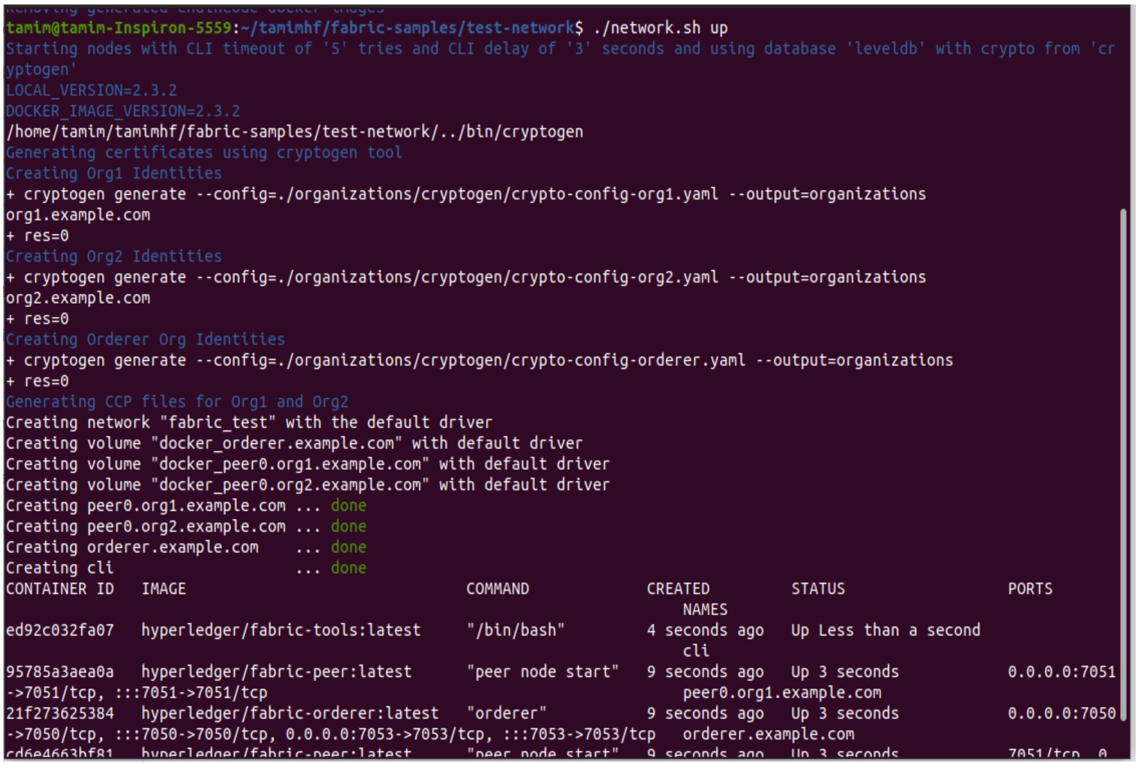
5.3.1 Bringing Up the Network

At first we run the following command to clone the required version of the hyperledger/fabric samples repository and checkout the correct version tag. The command also installs the Hyperledger Fabric platform-specific binaries and config files for the version into the /bin and /config directories of fabric-samples.

- curl -sSL https://bit.ly/2ysbOFE — bash -s – 2.2.2 1.4.9

First we go to the sample networks repository which is ”fabric-samples/test-network” and run the following command to bring up the test network.

- ./network.sh up



```
Removing generated chaincode docker images
tamim@tamim-Inspiron-5559:~/tamimhf/fabric-samples/test-network$ ./network.sh up
Starting nodes with CLI timeout of '5' tries and CLI delay of '3' seconds and using database 'leveldb' with crypto from 'cr
yptogen'
LOCAL_VERSION=2.3.2
DOCKER_IMAGE_VERSION=2.3.2
/home/tamim/tamimhf/fabric-samples/test-network/../bin/cryptogen
Generating certificates using cryptogen tool
Creating Org1 Identities
+ cryptogen generate --config=./organizations/cryptogen/crypto-config-org1.yaml --output=organizations
org1.example.com
+ res=0
Creating Org2 Identities
+ cryptogen generate --config=./organizations/cryptogen/crypto-config-org2.yaml --output=organizations
org2.example.com
+ res=0
Creating Orderer Org Identities
+ cryptogen generate --config=./organizations/cryptogen/crypto-config-orderer.yaml --output=organizations
+ res=0
Generating CCP files for Org1 and Org2
Creating network "fabric_test" with the default driver
Creating volume "docker_orderer.example.com" with default driver
Creating volume "docker_peer0.org1.example.com" with default driver
Creating volume "docker_peer0.org2.example.com" with default driver
Creating peer0.org1.example.com ... done
Creating peer0.org2.example.com ... done
Creating orderer.example.com    ... done
Creating cli                    ... done
CONTAINER ID   IMAGE                               COMMAND             CREATED         STATUS                    PORTS
                                                                         NAMES
ed92c032fa07   hyperledger/fabric-tools:latest     "/bin/bash"         4 seconds ago   Up Less than a second
                                                                         cli
95785a3aea0a   hyperledger/fabric-peer:latest      "peer node start"   9 seconds ago   Up 3 seconds              0.0.0.0:7051
->7051/tcp, :::7051->7051/tcp                                            peer0.org1.example.com
21f273625384   hyperledger/fabric-orderer:latest   "orderer"           9 seconds ago   Up 3 seconds              0.0.0.0:7050
->7050/tcp, :::7050->7050/tcp, 0.0.0.0:7053->7053/tcp, :::7053->7053/tcp   orderer.example.com
cd6e4663bf81   hyperledger/fabric-peer:latest      "peer node start"   9 seconds ago   Up 3 seconds              7051/tcp, 0
```

Figure 5.2: Network up

This command creates a Fabric network that consists of two peer nodes, one ordering node. No channel is created when we run above command.

Each node and user that interacts with a Fabric network needs to belong to an organization in order to participate in the network. The test network includes two peer organizations, Org1 and Org2. Peers are the fundamental components of any Fabric network. Peers store the blockchain ledger and validate transactions before they are committed to the ledger. Peers run the smart contracts that contain the business logic that is used to manage the assets on the blockchain ledger.Every peer in the network needs to belong to an organization. In the test network, each organization operates one peer each, peer0.org1.example.com and peer0.org2.example.com.The network also includes a single orderer organization that maintains the ordering service of the network.

5.3.2 Creating a Channel

We used network.sh script a open-sourced bash script to create a channel between Org1 and Org2 and join their peers to the channel.

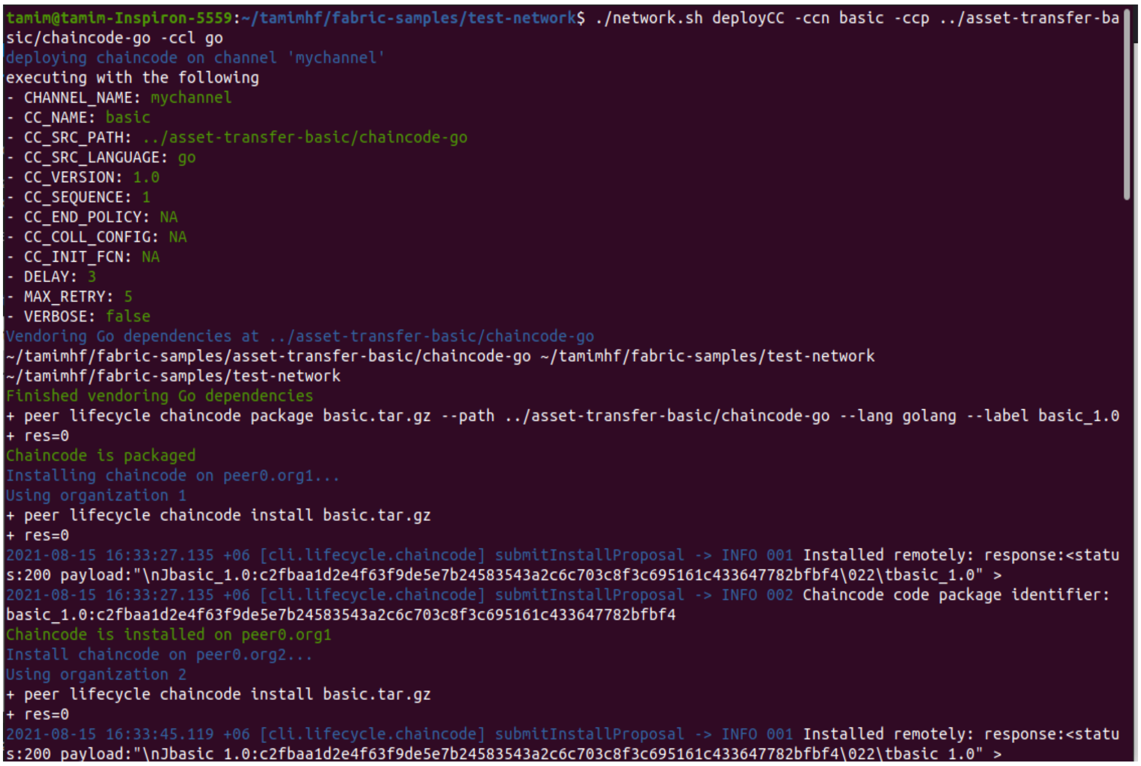
- We can create a defaul channel by using the, ./network.sh createChannel command
- We can also create a channel with our provided name if we use, ./network.sh createChannel -c channelName command. We can add multiple channels to our network by using this command. Like: channel1, channel2 etc.

5.3.3 Deploying a Chaincode

We can now start using smart contracts to interact with the channel ledger. Smart contracts contain the business logic that governs assets on the blockchain ledger. Applications run by members of the network can invoke smart contracts to create assets on the ledger, as well as change and transfer those assets. Applications also query smart contracts to read data on the ledger.

If we use the following command, now we can start a chaincode on the channel:

- ./network.sh deployCC -ccn basic -ccp ../asset-transfer-basic/chaincode-go -ccl go



```
tamim@tamim-Inspiron-5559:~/tamimhf/fabric-samples/test-network$ ./network.sh deployCC -ccn basic -ccp ../asset-transfer-ba
sic/chaincode-go -ccl go
deploying chaincode on channel 'mychannel'
executing with the following
- CHANNEL_NAME: mychannel
- CC_NAME: basic
- CC_SRC_PATH: ../asset-transfer-basic/chaincode-go
- CC_SRC_LANGUAGE: go
- CC_VERSION: 1.0
- CC_SEQUENCE: 1
- CC_END_POLICY: NA
- CC_COLL_CONFIG: NA
- CC_INIT_FCN: NA
- DELAY: 3
- MAX_RETRY: 5
- VERBOSE: false
Vendoring Go dependencies at ../asset-transfer-basic/chaincode-go
~/tamimhf/fabric-samples/asset-transfer-basic/chaincode-go ~/tamimhf/fabric-samples/test-network
~/tamimhf/fabric-samples/test-network
Finished vendoring Go dependencies
+ peer lifecycle chaincode package basic.tar.gz --path ../asset-transfer-basic/chaincode-go --lang golang --label basic_1.0
+ res=0
Chaincode is packaged
Installing chaincode on peer0.org1...
Using organization 1
+ peer lifecycle chaincode install basic.tar.gz
+ res=0
2021-08-15 16:33:27.135 +06 [cli.lifecycle.chaincode] submitInstallProposal -> INFO 001 Installed remotely: response:<statu
s:200 payload:"\nJbasic_1.0:c2fbaa1d2e4f63f9de5e7b24583543a2c6c703c8f3c695161c433647782bfbf4\022\tbasic_1.0" >
2021-08-15 16:33:27.135 +06 [cli.lifecycle.chaincode] submitInstallProposal -> INFO 002 Chaincode code package identifier:
basic_1.0:c2fbaa1d2e4f63f9de5e7b24583543a2c6c703c8f3c695161c433647782bfbf4
Chaincode is installed on peer0.org1
Install chaincode on peer0.org2...
Using organization 2
+ peer lifecycle chaincode install basic.tar.gz
+ res=0
2021-08-15 16:33:45.119 +06 [cli.lifecycle.chaincode] submitInstallProposal -> INFO 001 Installed remotely: response:<statu
s:200 payload:"\nJbasic_1.0:c2fbaa1d2e4f63f9de5e7b24583543a2c6c703c8f3c695161c433647782bfbf4\022\tbasic_1.0" >
```

Figure 5.3: Chaincode initializing

5.3.4 Interacting With the Network

We can use the peer CLI to interact with the network. The peer CLI allows us to invoke deployed smart contracts, update channels, or install and deploy new smart contracts from the CLI. But first we need to run the following command to add those binaries to our CLI Path:

- export PATH=$PWD/../bin:$PATH
- export FABRIC_CFG_PATH=$PWD/../config/

We can now set the environment variables that allow us to operate the peer CLI as Org1:

# Environment variables for Org1

- export CORE_PEER_TLS_ENABLED=true
- export CORE_PEER_LOCALMSPID="Org1MSP"
- export CORE_PEER_TLS_ROOTCERT_FILE=$PWD/organizations/peerOrganizations/org1.ex

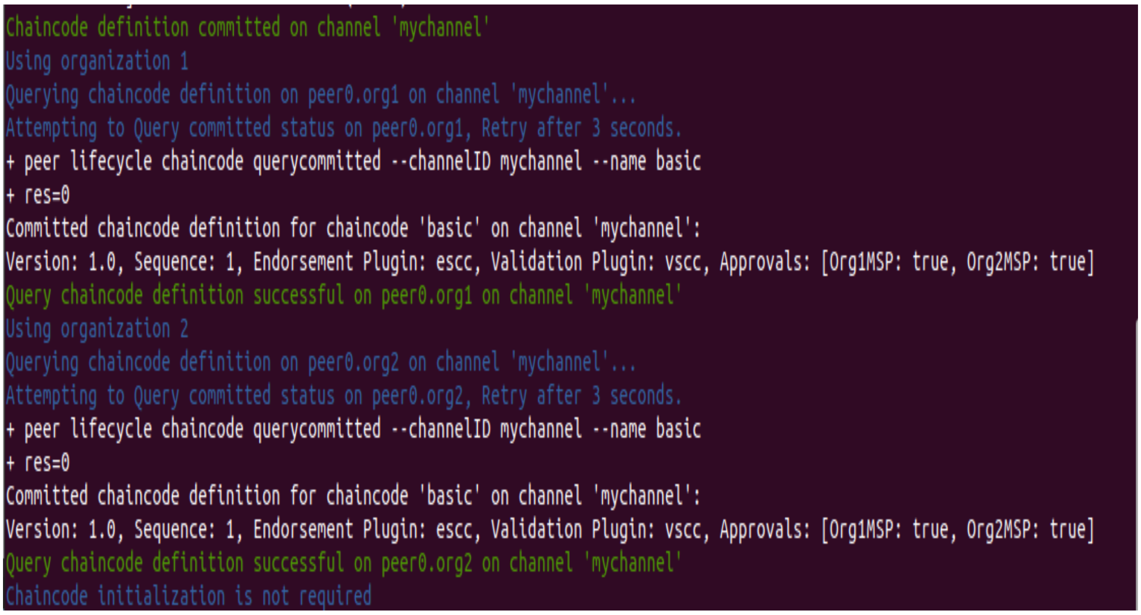
- export CORE_PEER_MSPCONFIGPATH=$PWD/organizations/peerOrganizations/org1.examp

- export CORE_PEER_ADDRESS=localhost:7051

CORE_PEER_TLS_ROOTCERT_FILE and CORE_PEER_MSPCONFIGPATH environment variables point to the Org1 crypto material dependency folders. localhost:7051 points to the hosting address of the core peer.

We can also invoke the deployed chaincode by running a command to initialize the ledger with assets. Also, we can now query the ledger from your CLI. By Running the following command to get the list of assets that were added to the network channel ledger:

- peer chaincode query -C mychannel -n basic -c ’”Args”:[”GetAllAssets”]’



```
Chaincode definition committed on channel 'mychannel'
Using organization 1
Querying chaincode definition on peer0.org1 on channel 'mychannel'...
Attempting to Query committed status on peer0.org1, Retry after 3 seconds.
+ peer lifecycle chaincode querycommitted --channelID mychannel --name basic
+ res=0
Committed chaincode definition for chaincode 'basic' on channel 'mychannel':
Version: 1.0, Sequence: 1, Endorsement Plugin: escc, Validation Plugin: vscc, Approvals: [Org1MSP: true, Org2MSP: true]
Query chaincode definition successful on peer0.org1 on channel 'mychannel'
Using organization 2
Querying chaincode definition on peer0.org2 on channel 'mychannel'...
Attempting to Query committed status on peer0.org2, Retry after 3 seconds.
+ peer lifecycle chaincode querycommitted --channelID mychannel --name basic
+ res=0
Committed chaincode definition for chaincode 'basic' on channel 'mychannel':
Version: 1.0, Sequence: 1, Endorsement Plugin: escc, Validation Plugin: vscc, Approvals: [Org1MSP: true, Org2MSP: true]
Query chaincode definition successful on peer0.org2 on channel 'mychannel'
Chaincode initialization is not required
```

Figure 5.4: Assets

After each run we bring down the network by simply running ./network.sh down command, to remove the node and chaincode containers, delete the organization crypto material, and remove the chaincode images from the Docker Registry. Otherwise there will be trouble running for the second time.

Chapter 6

Evaluation and Performance Analysis

Chapter 6

Evaluation and Performance Analysis

6.1 Introduction

To measure the capability of our proposed platform, we considered some case studies. These studies sum up the whole system. The passenger's request for the ticket, governing organizations regulate the rules and regulations. To simulate the behavior of hypothetical architecture technologies, we performed some sub-testing through mock data inputs and outputs.

6.1.1 Case Study: Multiple Channels

Three hypothetical channels are considered to achieve the whole transportation system. Blue, Red, and Green are these three channels respectively representing the Railway, Air, and Bus ticketing platform.

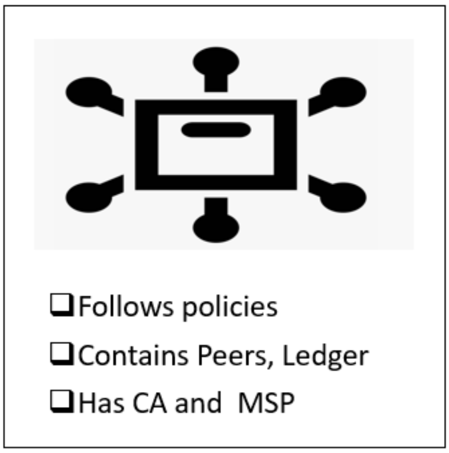


Figure 6.1: Channel Case

Each channel will have different policies, contract systems, and local MSPs in the network. As a channel is a part of the network, they can access the shared ledgers, ordering services, and MSPs of the network. To make it work all channels should fulfill the criteria of the network policy and must be the member of Network MSP.

6.1.2 Case Study: Governing Organization

There is four Governing organization from every three of the channels and a system organization is required to initiate the network. System organization (we may think as the Ministry of Communication) will initialize the network, and provide the rules and platform governance. And rest of the three organizations belonging to their respective channels (Railway, Bus, and Airway) will analyze tickets at their perspective channel. The use case of multiple channels is as follows:

Figure 6.2: Governing organization

- System organization P will initiate the network.
- System organization P provides certificate authority, an ordering node, and a network MSP.
- Organization P provides smart contracts on their peer node of each channel.

Furthermore, Organization P needs access to the entire database of the system to provide a statistical overview of ticketing data. It distributes the profit between each channel organization and stakeholders.

6.1.3 Case Study: Ticket Vendor

There may be multiple organizations in each of the channels. There will be a single or multiple ticket vending organization, an organization for validating and issuing tickets.

□Sells tickets
□Can Host Peers
□Participates in consensus

Figure 6.3: Ticket vendor

Let's assume Organization A is responsible for selling tickets to the passengers. Then it hosts one or multiple peers on the channel, consents to the other transaction, and issues new tickets to passengers.

- Organization A adds one or more than one peer's node to contract with the channel.
- Organization A provides an application to passengers to purchase tickets.
- It will provide consensus for other transactions on the respective channel.

Let's assume another Organization B on the channel that validates and issues tickets. Besides, it will handle other responsibilities to the network if needed.

- Organization B adds a minimum of one peer node to the channel.
- Organization B issues tickets for the customers.
- It will provide consensus for other transactions on the respective channel. It connects the passenger's organization to identify and validate the passengers.
- Organization B provides consensus for all other transactions for the respective channel.

6.1.4 Case Study: Passenger

Let's say a passenger is 30 years old and wants to travel. Luckily that man/woman is Eligible for a 30 discount. This passenger proceeds to purchase a ticket online or offline. The test for the passenger is as follows:

- A passenger creates his/her identity and purchases a Ticket-card from an organization.
- He/She purchases a ticket from an organization on that certain service.
- He/she validates a ticket from another organization on that particular service.

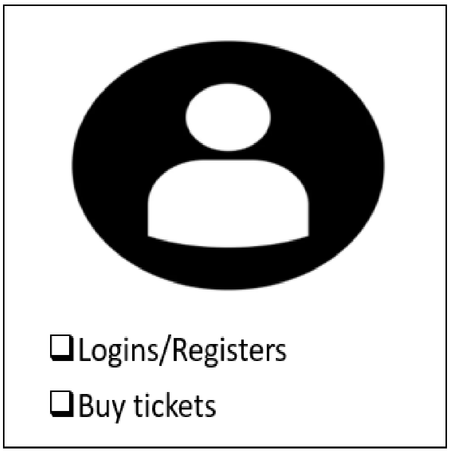


Figure 6.4: Passenger

If someone loses his/her ticket after purchasing. He/she can change the digital identity to prevent anonymous use of the card.

- A passenger purchases a ticket from an organization.
- Passenger validates a ticket from an organization.
- Passenger loses the provided physical ticket and cancels identity using another organization's application.

We have implemented hyper ledger fabric to simulate the process of channels and how all the things perform as a whole in a network. And we have successfully implemented it in a small manner due to the lack of hyper ledger fabric compatibility issues and resources. And we are looking forward to the implementation.

Chapter 7

Conclusion and Future Work

Chapter 7

Conclusion and Future Work

7.1 Introduction

This chapter discusses the conclusion of our research. Furthermore, it conveys the completion of proposed technologies. This section provides the future direction of the next study as well.

7.2 Conclusion

Blockchain in the ticketing system is not a novel approach anymore. But combining multiple ticketing systems into one network using distributed hyperledger fabric system is new. This platform does not exclude the currently available ticketing methods, instead, it works alongside the legacy system to provide a better experience to both organizations and passengers. This system has some technological benefits due to its secured system. Which improves the experiences of passengers in an online ticketing system keeping data secured.

7.3 Future Research Directions

In the future, the proposed architecture will be appended with more options for customers properly. Then we will be able to measure the performances of the presented architecture in numerous parameters correctly.

Bibliography

[1] E. Androulaki, A. Barger, V. Bortnikov, *et al.*, "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proceedings of the Thirteenth EuroSys Conference*, ser. EuroSys '18, Porto, Portugal: Association for Computing Machinery, 2018, ISBN: 9781450355841. DOI: `10.1145/3190508.3190538`. [Online]. Available: `https://doi.org/10.1145/3190508.3190538`.

[2] "Gartner: Blockchain and connected home are almost at the peak of the hype cycle," in *Available frommml: http://prwire.com.au/pr/62010//*, Gartner, Inc. (NYSE: IT), 2017.

[3] A. Rahman, M. K. Nasir, Z. Rahman, A. Mosavi, S. S., and B. Minaei-Bidgoli, "Distblockbuilding: A distributed blockchain-based sdn-iot network for smart building management," *IEEE Access*, vol. 8, pp. 140 008–140 018, 2020. DOI: `10.1109/ACCESS.2020.3012435`.

[4] T. M. Fernández-Caramés and P. Fraga-Lamas, "A review on the use of blockchain for the internet of things," *Ieee Access*, vol. 6, pp. 32 979–33 001, 2018.

[5] I.-C. Lin and T.-C. Liao, "A survey of blockchain security issues and challenges.," *IJ Network Security*, vol. 19, no. 5, pp. 653–659, 2017.

[6] F. Knirsch, A. Unterweger, and D. Engel, "Implementing a blockchain from scratch: Why, how, and what we learned," *EURASIP Journal on Information Security*, vol. 2019, no. 1, p. 2, 2019.

[7] S. Abeyratne and R. Monfared, "Blockchain ready manufacturing supply chain using distributed ledger," *International Journal of Research in Engineering and Technology*, vol. 05, Sep. 2016.

[8] E. A. Yavuz, A. Koç, U. C. Çabuk, and G. Dalkiliç, "Towards secure e-voting using ethereum blockchain," *2018 6th International Symposium on Digital Forensic and Security (ISDFS)*, pp. 1–7, 2018.

[9] S. Pareek, A. Upadhyay, S. Doulani, S. Tyagi, and A. Varma, "E-voting using ethereum blockchain," 2018.

[10] K. N. Griggs, O. Ossipova, C. P. Kohlios, A. N. Baccarini, E. A. Howson, and T. Hayajneh, "Healthcare blockchain system using smart contracts for secure automated remote patient monitoring," *Journal of medical systems*, vol. 42, no. 7, p. 130, 2018.

[11] S. Sankaran, S. Sanju, and K. Achuthan, "Towards realistic energy profiling of blockchains for securing internet of things," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, IEEE, 2018, pp. 1454–1459.

[12] A. Dorri, S. S. Kanhere, R. Jurdak, and P. Gauravaram, "Blockchain for iot security and privacy: The case study of a smart home," in *2017 IEEE international conference on pervasive computing and communications workshops (PerCom workshops)*, IEEE, 2017, pp. 618–623.

[13] H. Wang, Y. Wang, Z. Cao, Z. Li, and G. Xiong, "An overview of blockchain security analysis," in *China Cyber Security Annual Conference*, Springer, Singapore, 2018, pp. 55–72.

[14] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *2017 IEEE international congress on big data (BigData congress)*, IEEE, 2017, pp. 557–564.

[15] A. Reyna, C. Martın, J. Chen, E. Soler, and M. Dıaz, "On blockchain and its integration with iot. challenges and opportunities," *Future generation computer systems*, vol. 88, pp. 173–190, 2018.

[16] J. E. Pazmiño and C. Rodrigues, "Simply dividing a bitcoin network node may reduce transaction verification time," *The SIJ Transactions on Computer Networks & Communication Engineering (CNCE)*, vol. 3, no. 2, pp. 17–21, 2015.

[17] A. P. Joshi, M. Han, and Y. Wang, "A survey on security and privacy issues of blockchain technology," *Mathematical foundations of computing*, vol. 1, no. 2, p. 121, 2018.

[18] W. I. S. Engineering? "What is social engineering?," 2020.

[19] S.-C. Cha, W.-C. Peng, T.-Y. Hsu, C.-L. Chang, and S.-W. Li, "A blockchain-based privacy preserving ticketing service," in *2018 IEEE 7th Global Conference on Consumer Electronics (GCCE)*, 2018, pp. 585–587. DOI: `10.1109/GCCE.2018.8574479`.

[20] J. Preece and J. Easton, "A review of prospective applications of blockchain technology in the railway industry," *Preprint submitted to Int. J. Railw. Technol*, pp. 1–22, 2019.

[21] M. Kuperberg, D. Kindler, and S. Jeschke, "Are smart contracts and blockchains suitable for decentralized railway control?" *arXiv preprint arXiv:1901.06236*, 2019.

[22] J. Preece and J. Easton, "Blockchain technology as a mechanism for digital railway ticketing," in *2019 IEEE International Conference on Big Data (Big Data)*, IEEE, 2019, pp. 3599–3606.

[23] F. Benhamouda, S. Halevi, and T. Halevi, "Supporting private data on hyperledger fabric with secure multiparty computation," *IBM Journal of Research and Development*, vol. 63, no. 2/3, 3:1–3:8, 2019. DOI: `10.1147/JRD.2019.2913621`.

[24] E. Androulaki, A. Barger, V. Bortnikov, *et al.*, "Hyperledger fabric: A distributed operating system for permissioned blockchains," in *Proceedings of the thirteenth EuroSys conference*, 2018, pp. 1–15.

[25] C. Gorenflo, S. Lee, L. Golab, and S. Keshav, "Fastfabric: Scaling hyperledger fabric to 20, 000 transactions per second," *CoRR*, vol. abs/1901.00910, 2019. arXiv: `1901.00910`. [Online]. Available: `http://arxiv.org/abs/1901.00910`.