UNIVERSIDADE DE ÉVORA

ESCOLA DE CIÊNCIAS E TECNOLOGIAS

DEPARTAMENTO DE INFORMÁTICA

IDENTITY MANAGEMENT IN HEALTHCARE USING BLOCKCHAIN TECHNOLOGY

João Pedro Nunes dos Santos

Orientação Pedro Salgueiro
Vítor Beires Nogueira

Mestrado em Engenharia Informática

Dissertação

Évora, 30 de Dezembro de 2018

UNIVERSIDADE DE ÉVORA

ESCOLA DE CIÊNCIAS E TECNOLOGIAS

DEPARTAMENTO DE INFORMÁTICA

IDENTITY MANAGEMENT IN HEALTHCARE USING BLOCKCHAIN TECHNOLOGY

João Pedro Nunes dos Santos

Orientação Pedro Salgueiro
Vítor Beires Nogueira

Mestrado em Engenharia Informática

Dissertação

Évora, 30 de Dezembro de 2018

To My Family

Acknowledgements

Firstly, I want to thank my dissertation advisors, Pedro Salgueiro and Vítor Beires Nogueira, for being patient with me, for their availability and for their dedication in this project.

I want to thank my family who helped me finish this project, with their unending support, words of wisdom and for always pushing me to do better.

I also want to thank my work colleagues, who always supported me, helped me grow as a professional and person, challenged me to improve, and whom I consider as a second family.

I need to thank my close friends for always believing in me and for always being available when I needed the most, showing they are true friends.

Lastly, every person who supported me in some manner, I truly am grateful for your support.

Contents

List of Figures

List of Tables

Acronyms

EHR *Electronic Health Record*

HL7 *Health Level 7*

DDOS *Distributed Denial of Service*

EU *European Union*

BFT *Byzantine Fault Tolerant*

GDPR *General Data Protection Regulation*

DLP *Distributed Ledger Platform*

EVM *Ethereum Virtual Machine*

SDK *Software Development Kit*

MSP *Membership Service Provider*

HLF *Hyperledger Fabric*

CA *Certificate Authority*

FHIR *Fast Healthcare Interoperability Resources*

API *Application Programming Interface*

JSON *JavaScript Object Notation*

FHIR *Fast Healthcare Interoperability Resources*

CIA *Confidentiality, Integrity, and Availability*

SHA *Secure Hash Algorithm*

TLS *Transport Layer Security*

Abstract

Bitcoin served as the catalyst for creating a solution to secure digital transactions without requiring a trusted third party to be involved. To solve this problem, the mechanisms now associated with a Blockchain were conceptualized and implemented to serve as the backbone for the Bitcoin network. More specifically, it was used as a security tool making Bitcoin a more transparent and reliable form of cash, a digital cryptographic currency. Even tough Bitcoin ended up not fulfilling its intended purpose as a currency, the Blockchain technology has enabled further avenues for innovation and creativity.

Blockchain has since been used as the backbone for various cryptocurrencies networks. Some implementations of this technology allow the execution of code, also known as ”smart contracts”. Smart contracts are executed in an autonomous manner, with no human intervention. These can be used to solve a new set of problems due to their transparent behavior, lack of human intervention and distributed nature.

Blockchain technology allows the creation of systems that introduce a number of benefits over traditional data handling used in today's Healthcare Information Systems. Costs and risks associated with these systems can be reduced and information can become transparent and trustworthy to all participants.

The Hyperledger Fabric Network with true private transactions and advanced security mechanisms was used to serve as the basis for the system proposed in this dissertation. Moreover, a client application was also created that interacts with smart contracts to manipulate the ledger.

The work discussed in this dissertation shows that a Blockchain system based on Hyperledger Fabric is suitable for managing patients identity, in Healthcare. Even tough the feature set of this Blockchain is very focused in privacy and security, some additional measures regarding confidentiality of data had to be implemented. Regardless, a system was built successfully that met the requirements. The implementation of this system would provide transparency, immutability and additional security for patients and medical staff alike.

Keywords: Blockchain, Healthcare, Identity, Hyperledger Fabric, Smart Contracts

Sumário

Gestão de Identidade nos Serviços de Saúde Utilizando Tecnologia Blockchain

A criptomoeda Bitcoin foi essencial para criar uma solução para transacções digitais seguras, sem requerer a participação de um terceiro interveniente fidedigno para ambas as partes. Para resolver este problema, os mecanismos que hoje são associados com a tecnologia Blockchain foram concebidos e implementados para servir como base para a rede da Bitcoin. Mais especificamente, esta foi utilizada como um mecanismo de segurança, de forma a tornar a Bitcoin uma forma de dinheiro mais transparente e estável, uma moeda criptográfica. Mesmo que a Bitcoin não tenha conseguido cumprir o seu propósito original, a tecnologia Blockchain despoletou novas inovações e permitiu maior criatividade.

A Blockchain tem sido, desde então, a base tecnológica de várias criptomoedas. Algumas implementações desta tecnologia permitem a execução de código de uma forma autónoma exactamente como foi programado, sem intervenção humana. Habitualmente chamados *smart contracts*, estes podem ser usados para resolver um novo conjunto de problemas devido ao seu comportamento transparente, ausência de intervenção humana e devido à sua natureza distribuida.

A Blockchain é uma tecnologia que permite a criação de sistemas que introduzem um conjunto de beneficios em relação aos sistemas tradicionais de armazenamento de dados utilizados nos serviços de saúde. Custos e riscos associados a estes sistemas podem ser reduzidos e a informação pode ser mais transparente e fidedigna para todos os participantes.

A rede Hyperledger Fabric com transacções privadas e mecanismos avançados de segurança foi usada como base para a criação do sistema proposto nesta dissertação. Adicionalmente, uma aplicação foi criada que usa *smart contracts* para manipular o *ledger* da Blockchain.

O trabalho apresentado nesta dissertação mostra que um sistema baseado em Blockchain, neste caso em Hyperledger Fabric, é adequado a gerir a identidade de utentes, em organizações prestadoras de cuidados de saúde. Apesar das funcionalidades apresentadas por esta plataforma serem focadas em privacidade e segurança, algumas medidas adicionais em torno da confidencialidade dos dados tiveram de ser implementadas. Independentemente disso, o sistema foi construido com sucesso e conseguiu cumprir os requerimentos que foram definidos. A implementação deste sistema em serviços de saúde traria tranparência, imutabilidade e segurança adicional para utentes e profissionais de saúde.

Palavras chave: Blockchain, Saúde, Identidade, Hyperledger Fabric, Smart Contracts

1

Introduction

This Chapter introduces the main topics and technologies covered by this dissertation. Healthcare and its relationship with technology is presented. The current flaws associated with patients identity data management are described. The Blockchain technology is introduced as a potential solution to some of these problems.

The aim of this dissertation is to create a solution for managing the identity of patients in the Healthcare environment by using Blockchain technology, and in turn, evaluate the use of this technology in this specific use case. Health is intrinsically linked with technology, as new technologies enable safer and better treatments. Nowadays, Healthcare organizations store patients data on a digital format. The Electronic Health Record (EHR) is an abstract concept representing the patients digitally stored clinical data and their identity in a medical and clinical context.

Standards are an important aspect to take into account when designing an information system because they allow interoperability between different organizations. The Health Level 7 (HL7) Fast Healthcare Interoperability Resources (FHIR) standard (see Section 2.4), is being built primarily by the Health Level Seven organization. Over the last few years, it has seen a significant growth in usage. It is also an international standard with partnerships worldwide. HL7 Portugal is now starting its operations and is building a community to support this standard in Portugal [Hea17].

Blockchain is often known as the technology behind the Bitcoin cryptocurrency. Bitcoin depends on two complementary technologies, digital tokens and a Blockchain, that when orchestrated together facilitate trust, immutability and resiliency [Eva16].

A Blockchain runs on a network of computers and has a list of records that are replicated across the participating peers. Blockchain, as we know today, was conceptualized as the public ledger [1] for the Bitcoin cryptocurrency in 2008 by Satoshi Nakamoto [Nak08]. Satoshi Nakamoto is a pen name of, a still unknown to this day, individual or organization of individuals.

Traditional Healthcare databases and architectures are increasingly vulnerable and a target to groups of malicious actors that possess the technical expertise to deny services with Distributed Denial of Service (DDOS) attacks [2] or cause a data breach [3] [TJR18].

Making matters worse other problems spring to mind. The data that comprises the identity of a patient is often fragmented across multiple Healthcare organizations, in such a way that, to get a true overview of the patients history and diagnosis there would be a need to merge all the pieces of information stored in data systems that are hosted in architecturally different Healthcare information systems. Transparency is also a concern, as a patient does not currently possess the means to track how his medical data is being handled.

As more information becomes available, new insights can be extracted by Healthcare professionals that lead to an overall improvement of the patients interaction with the Healthcare ecosystem. However, maintaining a high amount of data secure is a costly and risky matter for every party involved. Security and privacy are a top concern regarding sensitive data.

This dissertation provides an insight into the design and implementation of a Blockchain based system for managing the identity of patients in an Healthcare setting and its subsequent evaluation. The creation of this system and its subsequent evaluation could provide interesting conclusions to medical staff as well as patients, regarding its potential implementation and deployment in the field.

In this document, different Blockchain implementations are explored to get an overview of their feature set and focus. Considering a set of defined requirements a platform is chosen, in order to evaluate the suitability of this technology in the Healthcare field. More precisely, in Chapter 2, a brief introduction to Blockchain and its most prominent implementations is presented. The technology is further explored in Chapter 3 and a number of real world use cases of this technology in the Healthcare field are explored. In Chapter 4 a Blockchain platform is chosen in order to build a prototype system to evaluate the usability of this technology in the Healthcare field. Insight is given into the system design and implementation. In Chapter 5, the system is tested and evaluated. Finally, in Chapter 6 some conclusions are presented and potential future work is discussed.

[1] A ledger is defined as an object in which items are regularly recorded, originally business activities and money received or paid, but in reality, it can be used to store any type of record.

[2] A Distributed Denial of Service attack is an attempt to make an online service unavailable by overwhelming it with traffic from multiple sources.

[3] A data breach is the intentional or unintentional release of secure or private/confidential information to an untrusted environment.

2

Background

This Chapter presents an overview of the Blockchain technology. Some Blockchain implementations are introduced and categorized. Consensus is introduced as a key aspect of this technology. Finally, developments in how the Healthcare industry has handled patients identity data management throughout the years is shown.

While Blockchain is not a new concept at this point, it is an evolving technology that is being used to solve old problems with new approaches, while at the same time creating new application fields and challenging old conventions and methodologies. Blockchain technology is having an environmental and economic impact, as discussed further in this Chapter.

2.1 Brief Introduction to Blockchain Technology

Blockchain can be defined as a collection of cryptographic and network technologies orchestrated to work together. The concept can also be used to refer to the Bitcoin's Blockchain or refer to forks [1] of the Bitcoin's Blockchain called Altchains [Lew15] that share some characteristics but may have different features and purposes. Some forks even improved upon the original premise of the concept, resulting in platforms that allow execution of code in an autonomous manner, exactly as it was programmed, with no human intervention.

A Blockchain is, generally speaking, a continuously growing list of records being written in the ledger. The ledger is a structure where all records are written and stored. This structure is constantly replicated across a network of peers, in opposition to having a single central record history, making it a good example of a distributed database [Bar17].

The purpose of a Blockchain is to establish trust between different participating parties in a network of distributed systems without the need for a trusted middle man [Dre17]. To fulfill this purpose it uses cryptographic techniques and digital signatures to, not only verify the authenticity of records, but also as a way to manage read or write access to the network. These, are also used to create proof that a record was written in the ledger and was never tampered with, creating an immutable history of records.

Unlike a conventional database system running in a server, where only a single entity keeps a copy of the underlying database, the ledger of the Blockchain is constantly replicated across any number of participating nodes in the network, making it a distributed system by design [Lew15]. Depending on the Blockchain implementation, not every participant has the same ability to interact with the ledger and in this respect a Blockchain can be permissionless or permissioned. Generally speaking, in a permissionless Blockchain every node of the network can write to the ledger, whereas in a permissioned Blockchain only a select group of entities have writing access to the ledger. The permissioned alternative is secure by default if the entities who participate the network are considered secure and trustworthy, for example, through a chain of trust similar to Domain Name Systems digital certification schema [Lew15, VS17].

But then, how does a permissionless Blockchain maintain security if every participant in the network has access to writing on it, including potentially malicious parties? Given that participating nodes in a public network can belong to different and often competing parties, there is no implied trust between them. Blockchain provides a mechanism to ensure the integrity of the ledger and prevent malicious meddling from interested parties, while at the same time, avoiding the need for a central authority [Bar17]. For example, the Bitcoin Blockchain uses a peer-to-peer network and manages to avoid the requirement of a third party being involved in a financial transaction such as a financial institution or a middle man, in order to see it through [Nak08].

The mechanism employed by the Blockchain to solve this problem is called consensus. Even though consensus mechanisms can behave vastly different, depending on its implementation and purpose, they are at the core, a solution to create immutability and ensure resiliency and transparency by ensuring the majority of the network agrees upon the sequence of events. For example, in the Bitcoin's Blockchain case, consensus is reached by the longest chain rule where the longest chain of blocks not only serves as proof of the sequence of events witnessed, but as proof that it came from the largest pool of computing power. This is due to the fact that the Bitcoin's Blockchain uses a Proof-of-Work consensus algorithm that relies on brute force to solve a complex mathematical puzzle, making the longest chain of blocks the one with the most computing power behind it and therefore agreed upon by the majority of the network [Baa16, Woo17], making that chain the most likely to be the one that represents the sequence of events witnessed.

[1] In this context, a fork is a condition whereby the state of the Blockchain diverges into one or more valid paths forward, where a part of the network has a different perspective than a different part of the network. The fork can be either with regards to a network's transaction history or a new rule in deciding what makes a transaction valid. Since a fork can create a chain with different rules a new Blockchain variation can be created.

There have been however, environmental and economic reasons to replace Proof-of-Work consensus algorithms. Nowadays the cryptocurrency mining, forms a billion United States Dollar industry with an estimated consumption of 288 megawatts in 2017 and in 2016, 70% of the Bitcoin's computational power was located in China [d'A17]. Unfortunately, the vast majority of electricity in the country is produced by burning coal, resulting in one of the biggest carbon footprints in the world. In response to this environmental and eventual economic concern over the sustainability of the mining incentives, there have been a few alternative algorithms that have eventually appeared. Proof-of-Stake, for example, is a consensus algorithm that was first suggested on the Bitcointalk forum on July 11, 2011 [For11]. It is currently in use in various currencies as their consensus algorithm and the first digital currency to use this method was Peercoin in 2012.

Rather than requiring the peers to perform a certain amount of computational work, a Proof-of-Stake system requires the validators to show ownership of a certain amount of currency. Any participating peer in the network can become a validator by sending a special type of transaction that locks some of their currency in a deposit, defined as the stake. The creator of a new block is chosen in a deterministic way, depending on its wealth and other factors, determined by the specific Proof-of-Stake implementation. Validators then participate in the process of creating and agreeing to new blocks. Proof-of-Stake based consensus algorithms have proven to be difficult to implement [But14], leading some Blockchain platforms to consider implementing a mix of Proof-of-Work and Proof-of-Stake algorithms.

In the case of a permissioned Blockchain implementation, interactions are made among a set of known, identified participants who have a common goal, but do not fully trust each other. By relying on the identities of peers, a permissioned Blockchain can use a more traditional Byzantine Fault Tolerant (BFT) consensus algorithm [SBV18, Wik18a].

While the Blockchain, we now know today, was conceptualized as the public ledger for the Bitcoin cryptocurrency in 2008 by Satoshi Nakamoto and implemented in 2009, many are now using it as a foundation across a variety of application areas such as traceability, asset management and Healthcare [SWP16].

2.2 Blockchain as a Platform

Due to Bitcoin getting extensive media coverage, the average public awareness in cryptocurrencies is shown to be rising [Bov17]. While Blockchain is used as a means to increase the resiliency of the Bitcoin cryptocurrency network from malicious parties, a token is used to represent the coin.

Just like a Euro, it has no value by itself, it only has value because we agree to trade goods and services in exchange for a higher amount of the currency under our control and we believe others will do the same [D'A16]. Through the years Blockchain has evolved to be capable of being an independent development platform using the token as a means to reward those who maintain the consensus by spending electricity and computation power in the network. In some networks, Ethereum and Hyperledger Fabric for example, one can build upon the network to create Decentralized Applications[2] (Đapps) that allow logic to be executed in an autonomous manner [Woo17].

In the same manner that the Bitcoin Blockchain can be seen as an adding machine, the Ethereum and Hyperledger Fabric Blockchain (see Section 2.3) can be seen as computers able to execute programs designed for it [Woo15].

[2] In this thesis context, Decentralized Applications are applications that run on a peer to peer network of computers rather than a single computer. They are a type of software program designed to exist on a network or multiple networks in a way that is not controlled by any single entity. Decentralized applications consist of the whole package, from backend to frontend. The smart contract is only the backend of these type of applications.

Ethereum is an open-source platform based on the Blockchain technology that enables developers to build and deploy Đapps. Ethereum is being developed by the Ethereum Foundation and was first discussed by Buterin in 2013. Ethereum intends to provide a Blockchain with a built-in programming language that is used to create smart contracts [Woo17], defined in the following paragraphs. Many Blockchain implementations nowadays use this concept.

A Blockchain that supports Bitcoin style transactions enables asset transfers between parties that do not trust each other. A Blockchain that supports smart contracts however, takes this further and allows for multi-step interactions to occur between mutually distrustful parties. Nick Szabo introduced this concept in 1994 [CD16] and defined a smart contract as "a computerized transaction protocol that executes the terms of a contract".

Smart contracts can translate contractual clauses into a piece of code, embedding it into property hardware, or software that can self-enforce these. Smart contracts are designed in order to minimize the need for trusted intermediaries between transacting parties, as well as, the occurrence of malicious or accidental exceptions.

In a Blockchain, smart contracts are scripts that describe the logical backend of a Decentralized application and are stored on the Blockchain where they execute as "autonomous agents" and where they can be instantiated and invoked as needed after achieving consensus. Since they reside on the network, they have a unique address. A smart contract is invoked by addressing a transaction to it. It then executes independently and automatically in a prescribed manner, according to the data that was included in the invoking transaction. Smart contracts allow general purpose computations on the chain. Smart contracts offer an abstract layer of interaction with the ledger, doing away with a required background in coding cryptography and mathematics, in order to program Blockchain applications [Woo17, Blo17a].

The Ethereum Blockchain is a permissionless Blockchain, and thus, it must have a consensus mechanism to ensure the validation process of every record and, in turn, ensure resiliency and immutability. While other implementations of the Blockchain have different consensus mechanics, in Ethereum's case, all participants have to reach consensus over the order of all transactions that have taken place. If a definitive order cannot be established then a double-spend [3] might have occurred and the transaction is rejected [Woo17].

2.3 Blockchain for Enterprise

Hyperledger Fabric (HLF) is part of the Hyperledger project started in December 2015 by the Linux Foundation. It is an open-source developer-focused community with the common goal of advancing the development of enterprise-grade, open-source Blockchain-based solutions. Fabric is an implementation of a Distributed Ledger Platform (DLP) under the Hyperledger umbrella [Cac16].

Hyperledger Fabric's initial commit was contributed by IBM and written in the Go programming language. It is a permissioned Blockchain and its main design goal was to surpass previous Blockchain implementation limitations, such as, lack of true private transactions and confidential contracts.

These goals are achieved thanks to assigning peers in the network three distinct roles and by offering the ability to create channels each with its own private ledger. A peer has the role of endorser, committer or consenter or multiple roles. Hyperledger Fabric is intended as a foundation for developing applications in a modular fashion, opting for a plug-and-play approach to its various components as well as its consensus mechanism [Hyp17b].

[3] Double-spending is a potential flaw in a digital cash scheme in which the same single digital token can be spent more than once. This is possible because a digital token consists of a digital file that can be duplicated or falsified. As with counterfeit money, such double-spending leads to inflation by creating a new amount of fraudulent currency that did not previously exist. This devalues the currency relative to other monetary units, and diminishes user trust as well as the circulation and retention of the currency.

Hyperledger Fabric, as discussed, also allows the creation of smart contracts. Fabric's key operational requirement is privacy, featuring true private transactions and confidential contracts. As such it fits in a business environment where sensitive information must be handled with care and disclosed on a case by case basis. Thanks to its modular approach, consensus protocols are no longer hard-coded and trust models can be repurposed, for example using Hyperledger Burrow.

Hyperledger Burrow is also part of the Hyperledger project and its development started in 2014 by Monax and sponsored by Intel. It is a permissionable smart contract machine written in Go and offers a modular Blockchain client with a permissioned smart contract interpreter built, in part, to the specification of the Ethereum Virtual Machine (EVM) with the client having, essentially, three main components, the consensus engine, the permissioned EVM and the Remote Procedure Call gateway [KMBD17, Hyp17a].

Hyperledger Burrow has its own Consensus Engine, the Byzantine fault-tolerant Tendermint protocol. The Tendermint protocol is an open-source effort that allows high performance in solving the consensus problem and also has a flexible interface for building arbitrary applications above the consensus, as well as, a suite of tools for deployments and their management [Buc16].

Hyperledger Indy is an open-source distributed ledger, purpose-built for decentralized identity. Indy uses a modified version of Redundant Byzantine Fault Tolerance called Plenum. Indy provides tools, libraries, and reusable components for creating and using independent digital identities distributed ledgers. Indy provides a software ecosystem where the users are in charge of decisions about their own privacy and disclosure of such information. Indy can be used to define connection contracts, revocation and curated reputation, for example. Hyperledger Indy was not used in this thesis as it was at the time in incubation phase. Documentation was lacking and the platform was not feature complete. With this said, Hyperledger Indy could be used as future work, as explained on Section 6.2.

The first network built on Indy was deployed on July 31, 2017, running version 1.0 of Indy. The Indy Software Development Kit (SDK) was released in August of the same year. The SDK supports common programming languages like Python, Java, Go, Node.js and Rust for interacting with the Indy ledger, running as Sovrin. iOS support for Indy is mature, and Android support is planned. Institutions currently have several incentives to adopt a solution similar to Indy, one being regulation. GDPR, and other legal requirements are forcing companies to adopt some measures in how they handle data pertaining to their clients and employees. Privacy and user data control standards are being demanded by governments and institutional organizations worldwide.

2.4 Developments in Healthcare

Records of a patient were originally stored in paper, a physical format. Thanks to the advent of the computers more and more records are stored on a digital format and the Electronic Health Record (EHR) was created [Mar17]. The digitalization of this data benefits handling of information between the patient and the medical professionals and medical institutions[ONC17].

But what is defined as identity? Identity is a construct that depends on the context. Identity is often defined as the characteristics determining who or what something is. In this thesis context, identity is defined as the set of characteristics that determine who a patient is, in the given Healthcare ecosystem they belong to, such as the name, the age, the cellphone number, the gender and the birth date of the patient.

Electronic Health Records encapsulate this information in digital format. Unfortunately, they are usually represented in a format according to the Information System they were designed to work with, meaning that they are not created according to any established standard. The increasing limitations of paper-based records, the

potential benefits of Electronic Health Records and the acknowledged challenges of delivering these in practice have stimulated a considerable investment in research and development of this solution. Between 1991 and 1998 the European Union provided considerable direct funding support to related research projects [Kal06].

To enable interoperability, standards for EHR were sought after and considerable research has been undertaken since 1996 to develop architecture formalisms to capture Healthcare data comprehensively, however, many of them have had no significant adoption, failing to bring the much needed consensus [EAR+06] that was required for enabling interoperability between different Information Systems in Healthcare institutions.

Healthcare professionals increasingly require access to detailed and complete health records in order to manage a safe and effective delivery of their care service, as well as, being able to share this information with their team in an efficient way. Patients nowadays should also be able to access their own information to an extent that allows them to play an active role in the management of their health related data. These requirements are becoming increasingly urgent as the focus of Healthcare delivery shifts progressively from specialist centres to community settings and to the patient's personal environment.

Health Level 7 organization has done much work to be recognized internationally and their standard for Healthcare data interoperability is being implemented in many countries to allow for joint efforts between medical and clinical institutions. As of 2017, HL7 has an official presence in 34 countries such as United Kingdom, Spain, France, Germany, Russia and China [Org16]. The organization has been keen on expanding their standard internationally as work continues on official adoption in other countries. Adding to this, HL7 has been providing several events and discussion meetings, as well as, developing its learning ecosystem in the form of certifications and web streams, for example. In 2017, HL7 has partnered with the Health Services Platform Consortium [Con18], composed of more than a dozen clinical professional societies, committing resources for creating and testing clinical data models and thus beginning to lay a semantic foundation for achieving interoperability.

There is a greater need for digital security solutions, as the amount of data grows in a connected age, where access to the world wide web is easily available and every device is part of the Internet of Things [Wik18b]. Various industries and the general public are becoming interested in solutions to solve problems in this field. Blockchain started as a security solution and is now laying the foundation for a change in the underlying flow of our economic and social systems [Zag18, Mar18, Lon18]. Some companies have already started developing Blockchain applications in the Healthcare field and established some key partnerships.

Guardtime has fully deployed their system in 2008, started cooperating in 2011 and in 2016 announced a partnership with the Estonian Government, where a million patient records are now secured by the system. Guardtime uses a system that shares the same background as the Bitcoin but is not based on Bitcoin. The Keyless Signature Infrastructure system [Tec18] proves the resilience of Blockchain related concepts, as well as, other advances in cryptography. Large companies like Verizon are becoming interested in Blockchain technology for their own purposes [Gua18, Est16].

Another company, Gem, is collaborating with Phillips Healthcare to explore options in this area[Pri16], and is opting to solve the interoperability problem with an additional layer of abstraction they call GemOS [Gem18]. Factom, another Blockchain-based service, has also announced a partnership with a major US medical services provider HealthNautica [Blo17b, Fac17].

In 2018, a platform called Medichain was introduced. This platform is also based on Blockchain technology and it allows patients to store their own data in a secure way and give anonymized access to this data to specialists. Giving data rewards users with tokens that represent value [Med18], effectively allowing patients to knowingly monetize their Healthcare data.

Even though many Blockchain based solutions are still very early on development or deployment and many projects do not actually materialize, the disruption potential of this technology is clear. All over the world many efforts are being made to regulate the high amount of data that is being generated by digital services.

For example, in the European Union (EU), the EU General Data Protection Regulation (GDPR) is officially in effect as of May 25, 2018. The aim of this regulation is to protect all EU citizens from privacy and data breaches in an increasingly data-driven world that is vastly different from when the preceding 1995 directive was established. GDPR defines a set of points that organizations and enterprises must comply regarding the handling of personal data. Since GDPR is a regulation, not a directive, it does not require national governments to pass any enabling legislation and is directly binding and applicable. A violator of this regulation may be fined up to **20 million € or up to 4% of the annual worldwide turnover** of the preceding financial year in case of an enterprise, whichever is greater. Although the key principles of data privacy still hold true to the previous 1995 directive, terms of usage as well as consent to use data has become more clear to the user of a data-driven service. Therefore, any solution proposed in the Healthcare field must also take this legal landscape into consideration which may be a problem. One of the key features of Blockchain is immutability which means that data, once written, is impossible to delete or tamper with. This causes a practical clash between this technology and the regulation, even tough they both are aligned in ideological concepts as both seek to return control of data usage and handling to their rightful owners. With this in mind it becomes clear that to comply with this regulation some additional steps need to be taken by this technology or eventual amendments need to be added to the regulation.

3

Blockchain: A Practical Overview and Use Cases

This Chapter presents a more technical overview of the Blockchain technology. A more in-depth exploration of the Ethereum and Hyperledger Fabric platform characteristics is also shown. Smart contracts and use cases of this technology in Healthcare are also shown.

As discussed in Chapter 2, Blockchain implementations are an emerging structure for distributed computing systems that provide an immutable history of records written to a ledger, even when there is no implicit trust relationship between the parties involved [Bar17]. The origin of this technology can be traced back to the realization that full centralization should be avoided in critical services.

3.1 Trust in a Network

Banks used to keep track of their financial transactions by writing on a book usually located at the central bank. This book was often called ledger. Whenever a transaction occurred someone wrote the record of the transaction on the book, permanently adding information to the book. In short, the ledger acts as a permanent mean of storing all the transaction details between the bank and other entities.

Nowadays, banks do not use the ledger in a book format. Instead, the ledger is the structure that holds all transaction information the bank possesses. It is a structure that keeps the original purpose of recording all the transactions that are made.

Imagine the following situation, Bob is on vacation and needs to borrow money from Alice, his wife. Bob calls Alice to ask for some money and Alice tells him it will send the money right away. Alice then proceeds to use her homebanking system to transfer some of her money to Bob. Finally Alice calls Bob to tell him that she made the request to send money to him. As seen on Figure 3.1 Bob and Alice need to use and trust the the bank as a middle man in order to complete this transaction. If the bank was ever to be unavailable, the bank's database was corrupted or if someone with privileged access was able to intercept the transactions from inside the bank then all transactions between Bob and Alice would fail creating additional costs to all parties involved.

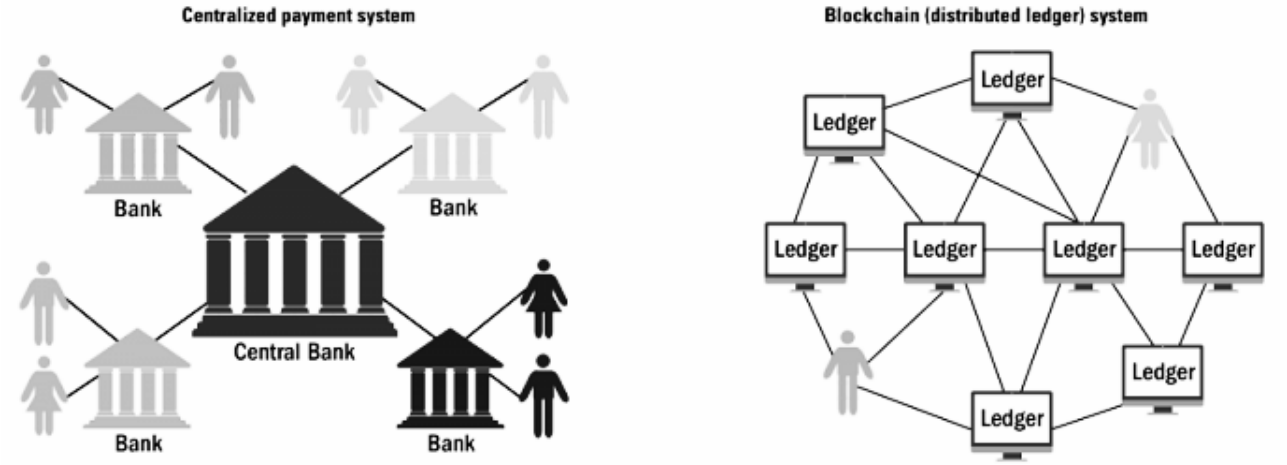

Figure 3.1: A comparison between a Centralized Banking System and a Distributed Ledger. (Source: Finance & Development, 2016)

For a long time it was necessary, to establish trust between two entities, a middle-man with a neutral stake. While the ledger is also at the core of the Blockchain, this technology aims to solve the dependency placed upon third parties using decentralization and aims to make two different entities trust each other through constant replication of the ledger, a security mechanism called consensus.

For example, in the Bitcoin's Blockchain case, Alice initializes this process in the Blockchain by signing a transaction that describes some of her money is being sent to Bob's digital wallet address. The transaction is placed in a pool of unconfirmed transactions and broadcasted to every node. Every 10 minutes a miner [1] solves a hash[2] problem, collects some of these unconfirmed transactions, packages them in a block and broadcasts that he found the solution. The output of the hash function is easily verified, and every miner confirms if the block is considered valid. If consensus is reached by the nodes, the block is added to the Blockchain (written in the ledger) and the new state is persisted on all peers through replication of the ledger. At this point the transaction is confirmed, requiring no middle man to ensure the transaction is trustworthy and valid. The transaction authentication process is shown on Figure 3.2

[1] Bitcoin miners help keep the Bitcoin network secure by approving transactions and then writing in the Blockchain ledger. In a Proof-of-Work consensus based Blockchain, like Bitcoin, miners use special software and hardware to solve complex mathematical problems and are issued a certain number of Bitcoins in exchange, if a solution is found, as an incentive to mining.

[2] A cryptographic hash function allows one to easily verify that some input data maps to a given hash value. However, if the input data is unknown, it is deliberately difficult to reconstruct it by knowing the stored hash value. The cryptograhic puzzle that miners need to solve is to find an input value, often called nonce, that produces an output hash value that satisfies a defined condition by the mining software.

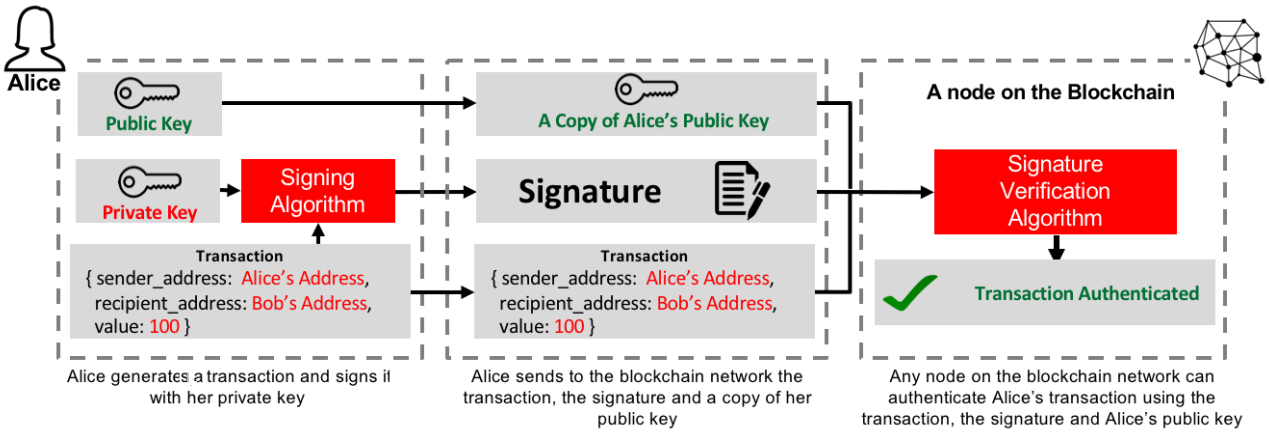

Figure 3.2: Bitcoin Transaction Authentication Process (Source: Adil Moujahid, 2018)

While consensus has a system performance impact due to the necessary replication of data and wasteful energy and computing power in the mining process, it is a mechanism that establishes a set of rules that defines if a sequence of transactions is considered valid. Different Blockchain implementations often use a variety of consensus protocols to balance this trade-off.

3.2 Permissionless and Permissioned Blockchain Implementations

There are three types of computer systems, as seen on Figure 3.3.

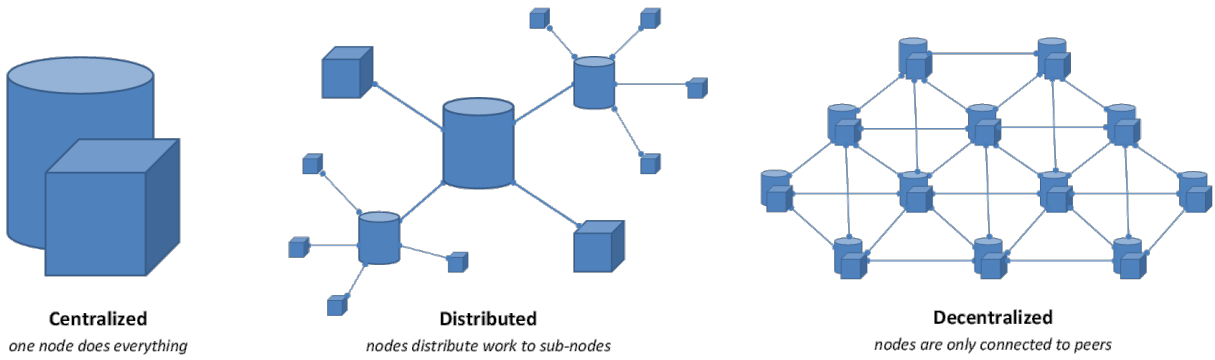

Figure 3.3: A comparison between different types of computing systems. (Source: Eric Grange, 2016)

Put simply, a centralized system is one that is governed by a hierarchical authority, for example, banks and credit card companies. If one wants to use a Visa card, one must request access from Visa and be approved. At any time the access to that line of credit and associated funds may be made unavailable and access permanently revoked [Dre18].

In contrast, distributed systems are based upon the philosophy that processing is shared across multiple nodes even if the decisions themselves may still be centralized and use complete system state knowledge of the network. Finally, a decentralized system is one where no single node can make a decision individually, instead relying on the other participants to reach an agreement and make a decision, as no single node has a complete system state knowledge. With this in mind, a decentralized system is seen as a subset of a distributed system.

A Blockchain is a distributed by design. However there are two major implementation categories as discussed briefly on Chapter 2.

Permisionless Blockchain implementations were the first to appear and while some industries saw benefits in using the technology, some saw drawbacks to adopting it in enterprise-grade systems [Gop16] due to this unregulated nature. They do have some advantages however compared to traditional systems.

Permisionless Blockchain implementations, like the Bitcoin's and Ethereum's Blockchain for example, have no barrier to entry. This means that anyone can, in theory, participate in the network, write into it as a result of mining and store data in the ledger sharing the work needed to maintain the network. Permisionless implementations have some strengths. These are completely open and transactions are transparent while also being able to offer anonymity or pseudo-anonymity. They also take away the need for system administrators or central servers since the network is based exclusively on peer-to-peer technology and decisions are made by every participant, creating reduced costs to maintain and deploy Decentralized applications. On the other hand, these implementations are slower than traditional systems because every node must participate in consensus creating an overhead before a transaction is considered verified. Due to this there is also a time cost associated because of the need to wait until verification of the transaction. They operate without clear legal rules and are trust-free, meaning that there is no responsible entity if data loss or damages affect systems based on this implementation.

Permissioned Blockchain implementations have some clear advantages for enterprise. They are faster because consensus is done by a set of nodes instead of the entire network, can fall back on the legal system because it features an identity service. This means the platform is auditable and that there is a legal responsible entity or entities that manage the network. However, when compared to the permissionless variant, costs are higher due to having the need for a system administrator and servers to manage the network, featuring a private membership meaning that they are closed to the general public and managed by a set of entities and are a compromise between the original vision of a completely decentralized network and enterprise needs and concerns.

Enterprises benefit greatly from the immutability of the Blockchain architecture, in that all records cannot be changed. By adding authorized identity services onto Blockchain, they can meet the regulatory needs of their industries, by allowing the network to be auditable and assets to be traceable, falling back to laws or regulations if a dispute between participating entities occurs [Bar17].

3.3 A Decentralized Open Platform - Ethereum

Ethereum is a permissionless Blockchain implementation. It is a platform that lets anyone build and use decentralized applications commonly named Đapps. It is an open-source project developed primarily by the Ethereum Foundation and was designed to be adaptable and flexible, in contrast to Bitcoin's Blockchain that only records financial transactions [Com16], as discussed in Section 2.2.

It features a friendly programming language called Solidity that is influenced by C++, Python and Javascript. It is designed to allow an easy way for developers to create new applications on the Ethereum platform with code of arbitrary algorithmic complexity in a turing complete language. Smart Contract application code targets the Ethereum Virtual Machine (EVM), presented shortly, and is then deployed to the Blockchain via a local Ethereum node [Woo17, Bar17].

At the heart of Ethereum is the Ethereum Virtual Machine, as seen on Figure 3.4. Ethereum also includes a peer-to-peer network protocol, as does any Blockchain. The Ethereum Blockchain database is maintained and updated by many nodes connected to the network. Each and every node of the network runs the EVM and

executes the same instructions in order to maintain consensus across the entire Blockchain. Decentralized consensus gives Ethereum a high degree of fault tolerance, ensures zero downtime, and makes data stored on the Blockchain forever unchangeable and censorship-resistant [Com16].

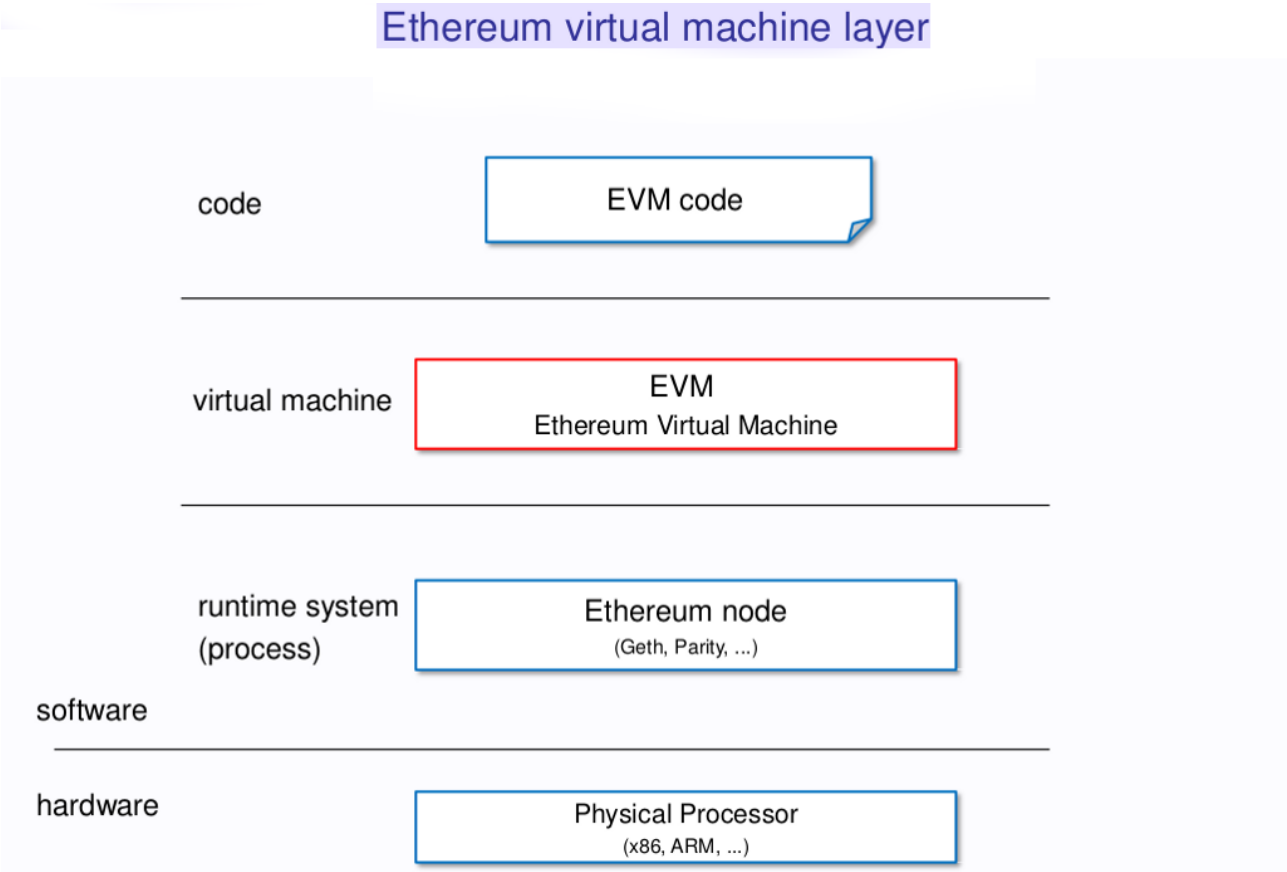

Figure 3.4: A diagram of where the Ethereum Virtual Machine fits into the Ethereum Platform (Original: Vaibhav Saini, 2018)

Users must pay a small transaction fee to the network each time they execute a transaction. This protects the Ethereum Blockchain from frivolous or malicious computational tasks, like Distributed Denial of Service attacks or an infinite loop in smart contract logic. The sender of a transaction must pay for each step of the "program" they activated, including computation and memory storage. These fees are paid in amounts of Ethereum's native value-token, ether, and then these transaction fees are collected by the nodes that validate the network commonly called miners. Miners are nodes in the network that receive, propagate, verify, and execute transactions. Ethereum currently uses a Proof-of-Work based consensus algorithm but plans to change to a Proof-of-Stake based algorithm due to environmental and financial concerns as well as reduced centralization risks [Com16, Ray18].

Ethereum has a live production network called "mainnet" available for any developer to deploy applications to, as well as three test networks. "Ropsten" is based on a Proof-of-Work algorithm while "Rinkeby" and "Kovan" are based on Proof of Authority [3]. All of these are publicly available and free to use [Bar17, Ken18].

[3] In Proof of Authority based networks, transactions and blocks are validated by approved accounts, known as validators. Validators run software allowing them to put transactions in blocks. The process is automated and does not require validators to be constantly monitoring their computers. It does, however, require maintaining the authority node uncompromised.

Ethereum has had some unforeseen problems along the way, namely the Digital Decentralized Autonomous Organization heist where a hacker took advantage of a bug in a smart contract to steal a great sum of money requiring a hard fork of the network to a point before the incident [Lei17]. Also, with Ethereum frequently reaching full transaction capacity, scaling solutions are the next big investment and focus [Kum18].

There are a few proposed solutions by Buterin. For example, **sharding** is a solution that aims to avoid every node processing all data in order to verify and process a transaction. When transactions are initiated they will not be directed to all the nodes but would instead only be directed to those depending on the shard in question. Another solution is **off chain computation** where a layer apart from the Blockchain is created and where all the computation or solving of a complex mathematical equation takes place. This would not only take the load off the Ethereum Blockchain but also help decrease the cost of transaction verification and processing. This mechanism would ensure that the tasks that account for slower transaction speeds on the Ethereum's Blockchain do not affect the whole network. Finally, to avoid every node having the need to download the entirety of the Blockchain's data, the complete picture can be stored on cloud and each node only has to store and load relevant data [But18].

3.4 A Permissioned Distributed Ledger Platform - Hyperledger Fabric

As discussed in Section 2.3, Hyperledger Fabric is a platform for distributed ledger solutions featuring a modular architecture. It provides developers with a permissioned platform targeted at business and enterprise use cases that supports pluggable implementations of different components to accommodate the complexity and intricacies that exist across the economic ecosystem. It is an open source project initially committed by IBM and established under the Linux Foundation, being developed by over 44 organizations and more than 250 members [Hyp17b, Hyp18].

It supports the creation of smart contracts, **commonly called chaincode in Fabric**, that are authored in general-purpose programming languages such as Java, Go and Node.js rather than constrained domain specific languages.

Chaincode in Fabric consists of two components. The **code itself**, which describes the logic of the program running in the execution phase, and the **endorsement policy** that describes how a specific chaincode transaction is validated. For example, a typical endorsement policy lets the chaincode specify the endorsers for a transaction in the form of a set of peers that are necessary for endorsement and subsequent successful validation [ABB+18]. Chaincode runs in a container isolated from the peer process which consented its installation providing additional control over information dissemination.

At the heart of Fabric is the permissioned distributed ledger that provides a way to secure the interactions among a group of entities that have a common goal but which may not fully trust each other. By relying on the identities of the participants, a permissioned ledger platform can use a more traditional Crash Fault Tolerant or Byzantine Fault Tolerant consensus protocols that do not require mining or an associated currency in order to achieve consensus.

Fabric introduces the execute-order-validate Blockchain architecture as shown on Figure 3.5 and does not follow the standard order-execute design illustrated on Figure 3.6 [ABB+18]. In this architecture, a client sends transactions to the peers specified by the endorsement policy. Each transaction is then executed and the output is recorded. After execution, transactions enter the ordering phase. An ordered sequence of transactions grouped into blocks are produced using the consensus mechanism. Then, these blocks are broadcast to all peers. Fabric orders the transaction outputs computed during the execution phase. Each peer then validates state changes according to the endorsement policy and the consistency of the execution in the validation phase. All peers validate the transactions in the same order and validation is deterministic. In this sense, Fabric intro-

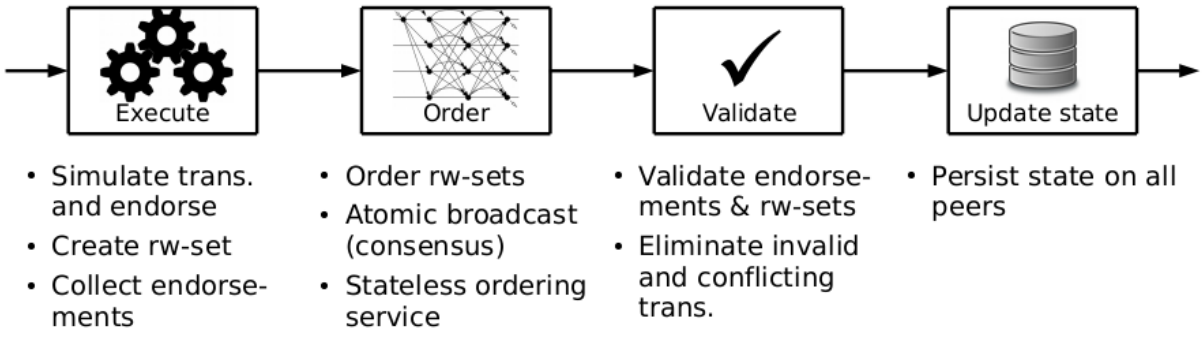

Figure 3.5: Execute-order-validate architecture of Fabric (Source: IBM, 2018)

duces a novel hybrid replication paradigm in the Byzantine model [ABB+18]. This model combines passive replication, which is the pre-consensus computation of state updates, with active replication, the post-consensus validation of execution results and state changes.

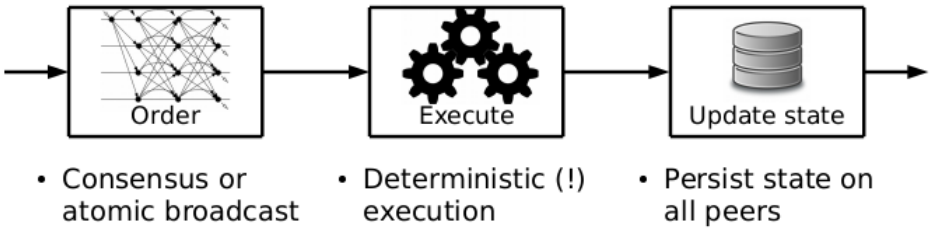

Figure 3.6: Order-execute architecture in Replicated Services Like Ethereum (Source: IBM, 2018)

On the other hand, the order-execute architecture is conceptually simple, leading it to be currently widely implemented in replicated services such as Blockchain. In this architecture the transactions are executed sequentially on all peers which limits the maximum number of simultaneous transactions that can be achieved. Additionally a Denial of Service attack can be mounted just by deploying a slow performing smart contract or one with an infinite loop to the network since the Blockchain forms a distributed computing engine. To cope with this issue, public programmable Blockchains with an associated cryptocurrency, account for the execution cost of executing of the program.

In Fabric all nodes that participate in the network have an identity, as provided by a modular Membership Service Provider (MSP). A MSP is a component that aims to offer the abstraction of a membership operation architecture meaning that all identities are only allowed to participate if verified to be considered trustworthy. The MSP maintains the identities of all nodes in the system and is responsible for issuing credentials that are used for node authentication and authorization. A MSP may define their own notion of identity, and the rules by which those identities are governed and authenticated using signature generation and verification [Hyp17b].

Fabric also assigns different roles to peers. Nodes in Fabric network can have one or more, of these three roles:

- Clients submit transaction proposals for execution, help orchestrate the execution phase, and broadcast transactions for ordering.

- Peers execute transaction proposals and validate transactions. All peers maintain the ledger, where all transactions are recorded in the form of a hash chain, as well as the state, a succinct representation of the latest ledger state. Not all peers execute all transactions.
- Ordering Service Nodes or orderers are the nodes that collectively form the ordering service. In short, the ordering service establishes the total order of all transactions in Fabric, where each transaction contains state updates and dependencies computed during the execution phase, along with cryptographic signatures of the endorsing peers defined in the endorsing policy of the transaction. Orderers are entirely unaware of the application state, and do not participate in the execution nor in the validation of transactions. This design choice renders consensus in Fabric as modular as possible and simplifies the replacement of consensus protocols in Fabric.

Looking ahead, Hyperledger Fabric will continue to focus on privacy and confidentiality with v1.2 being recently released, v1.3 and 1.4 expected to be out this year with further emphasis on these aspects in a regular quarterly cadence [Gut18].

3.5 An Overview of Blockchain Platforms

In this section a brief comparison is made between the platforms introduced in this Chapter. As seen in Table 3.1, different Blockchain implementations have different characteristics and focus.

Characteristics	**Bitcoin**	**Ethereum**	**Fabric**
Permission Restrictions	Permissionless	Permissionless	Permissioned
Access to Data	Public	Public or Private	Private
Consensus	Proof-of-Work	Proof-of-Work (Ethash)	Practical Byzantine Fault Tolerant
Governance	Low, decentralized decision making by community/miners	Medium, core developer group, community improvement proposals	Low, open-governance model based on Linux model
Native Currency	Yes, Bitcoin	Yes, Ether	No
Scripting	Limited possibility	Turing-complete virtual machine, Solidity DSL language	Turing-complete scripting chaincode, multiple general purpose language support
Focus	Financial Transactions	General Purpose	Enterprise Focused

Table 3.1: Characteristics Comparison between Bitcoin, Ethereum and Fabric.

4

Designing and Building the System

This Chapter describes the requirements for a system built upon Blockchain technology. The requirements where chosen in order to create a system that is interesting to an organization while still respecting the patients data access rights. Ethereum and Hyperledger Fabric advantages and disadvantages were weighted for this use case. Finally, insight into the design and implementation of the system is given.

The goal of this dissertation is to create a solution to manage the identity of patients in an Healthcare organization by conceptualizing, implementing and evaluating a Blockchain based system that fulfils this role. In order to fulfill this goal, the development part of the work described in this dissertation was primarily divided into four steps with each step building upon the previous ones. The development workflow spans the conceptualization and its associated challenges and ends in the implementation of said system.

4.1 First Step - Defining Requirements and Choosing a Platform

After investigating the various Blockchain platforms some criteria was needed to serve as reference. As such, the first step consisted in defining a set of key points that the built system had to fulfill. Defining the requirements proved helpful to choose the most appropriate platform for the objectives as explained later.

4.1.1 Requirement Definition

The requirements for this work were deemed to be as follows:

I. The system must allow a patient to opt into the network and register as a participant.

II. The system must allow a patient to record his medical data under the approval of an administrator.

III. The system must keep data confidential, transparent and have high availability.

IV. The system must provide the patient with the ability to share his data with another entity, for example sharing information with a doctor.

V. The system must allow the deletion of a patient's data in some manner, if he wishes to do so, in order to comply with European privacy laws, discussed in Section 2.4.

These requirements were chosen in order to create a system that is interesting to an organization while still respecting the patients data and their access right to it.

After defining the requirements it was necessary to choose the Blockchain platform that best fulfils these requirements.

4.1.2 Choosing a Platform

Even though Blockchain platforms normally originate from the realization that full centralization has major drawbacks, they often have different goals. These translate into architectural differences and different development focuses. These range from open networks, such as Ethereum which anyone can join and use, to permissioned distributed ledgers, which can be run publicly or privately but are only open to access and participation through a membership service, such as Hyperledger Fabric and Hyperledger Indy.

Ethereum has a growing learning ecosystem and community. It is easy to start interacting with the network as anyone is able to simply download a client and connect to it. Thanks to the Solidity (see Section 3.3) smart contract language being targeted for the specific purpose of authoring smart contracts it only allows for a deterministic program to be written, thus avoiding potential conflicts in the execution of these, for example, in the production network. Since Solidity is a Domain Specific Language it is platform easy to develop for since it provides a well thought out and well organized documentation with an easy to use library of operations.

Ethereum is being used in a great deal of projects around the world proving its stability and suitability in a wide variety of use cases. On the other hand, handling patients medical data is a great responsibility due to the private and personal nature of this data. Also hospitals and Healthcare clinics must obey the regulatory laws regarding privacy and usage of this data.

It is also worth noting that while Ethereum can handle private data exchange by building upon it, as shown by Barclay in his dissertation, it was not designed with this intent in mind, therefore these middle ground solutions can prove to be unwise to use at scale given Ethereum's and the whole Blockchain's ecosystem past problems with scalability.

Fabric, like Ethereum, was built with the intention of being a general purpose use Blockchain. It provides developers with the tools needed to build any system they can imagine. The latter is clearly focused on making organizations feel more at ease by being auditable. It is auditable because it offers an identity service by using a membership service provider and a private certificate authority that emits certificates specific for the Fabric

network. These concerns allow this platform to avoid the same fate Internet of Things devices have had in the Healthcare field [TFH17] where the lack of security regulations and ambiguity in how data was being collected by these devices has limited their usage and prevented the widespread usage of these devices in the Healthcare field.

Fabric also has a good amount of development tools that are now maturing and a good learning environment with ample documentation about every aspect important for a developer looking to get started into it. Fabric is being backed by the Linux Foundation and IBM, lending credibility to the project and ensuring that this platform is supported and developed into the foreseeable future, as it is being governed by a diverse technical steering committee and by a diverse set of maintainers from multiple organizations. In regards to performance, the focus in this area is clear, as the Hyperledger community has appointed a Performance and Scale working group to improve performance as well being tasked to implement benchmarking framework for Hyperledger projects called Hyperledger Caliper [Org17].

Regarding Fabric's features, it lends itself very well to fulfill the project requirements. Fabric's channels and private data segregation at peer level make a clear statement that privacy is important in this platform, which is in line with the requirements that were laid out for the work described in this dissertation. It is also worth noting that many upcoming Blockchain based projects in the Healthcare field are using permissioned networks due to these same concerns regarding the patients privacy while retaining the key benefits of Blockchain such as immutability and decentralization. The fact that Fabric has no associated currency also means there is no required mining incentives to maintain the network, even tough it does require some additional infrastructure to set up the network, leading to a higher initial investment in a solution based on this platform.

Both are very interesting platforms, but ultimately it was decided to use Hyperledger Fabric as the platform on which to build upon. This decision was taken in part because Fabric was purpose built for a very regulated environment and is focused on privacy and scalability which are required in the Healthcare field. Also, this technology is relatively recent and there is still a great lack of knowledge available to the general public, making it a more interesting choice from an theoretical standpoint.

4.2 Hyperledger Fabric Components and Administration

After selecting Hyperledger Fabric as the work platform, it became necessary to understand in further detail what are the main components that form a network and the tools available to manage these components. This Section discusses the main components of a Fabric network and the tools required to create and maintain a Fabric network. These components often interact with one another and provide the technical infrastructure that comprises this technology.

4.2.1 Hyperledger Fabric Components

An Hyperledger Fabric (HLF) network is defined as the technical infrastructure that provides ledger and smart contract services to applications. Smart contracts are used to generate transactions and interact with the ledger. The network is comprised of several components, which are described in the following paragraph.

The ledger is a central component of a HLF network. The ledger is composed by a world state and a Blockchain as seen on Figure 4.1. The world state is a database that holds the current values of ledger states. States are, by default, expressed as key-value pairs. The world state is useful because it makes it easy for a smart contract to get the current value of these states, instead of having to traverse the entire transaction log. The Blockchain holds the transaction logs that record the history of changes that have resulted in the current world state.

Finally, transactions are collected and recorded in an immutable sequence of blocks, in which each block contains a set of ordered transactions by the orderer service.

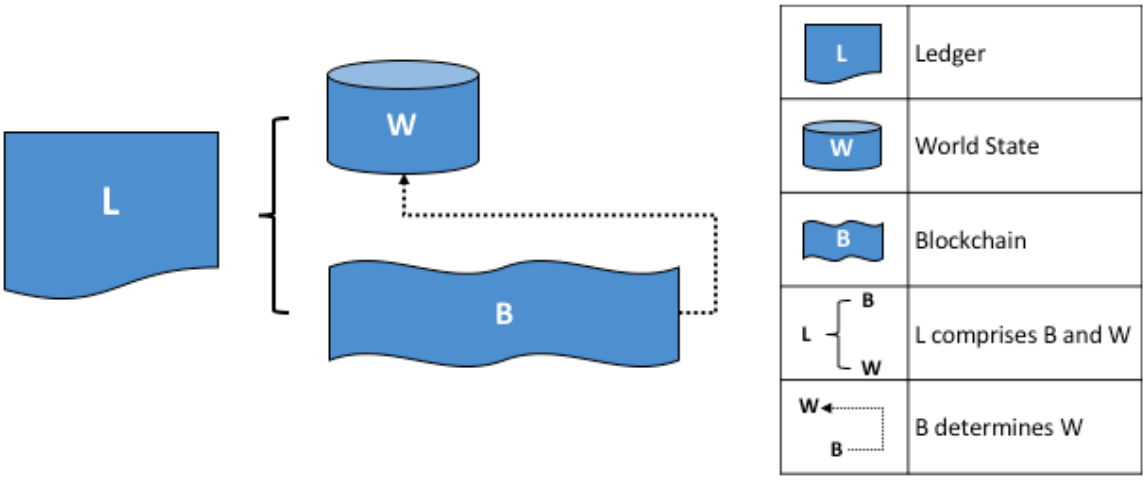

Figure 4.1: Fabric's Ledger Overview (Source: HLF Fabric Documentation)

Another component is the set of peers participating in the network. A peer is a node that hosts a copy of the multiple ledgers and smart contracts. There is one logical ledger in a Hyperledger Fabric network, even tough, in reality the network maintains multiple copies of a ledger that are synchronized through consensus. HLF opts to allow multiple ledgers in a network to achieve different goals of a greater purpose. This allows the creation of channels of information between trusted parties, for example, a channel of secure and private information between the clinical staff of an hospital and a patient as discussed on Chapter 2, in which every channel has a ledger.

Through a peer connection, applications execute chaincode that queries or updates a ledger. Peers have at least one of the three different roles assigned to them, as seen on Section 3.4. Applications always connect to peers when they need to access ledgers and smart contracts. Every peer in the network is assigned a digital certificate by an administrator from its owning organization. The mapping of a peer's identity in an organization is provided through the membership service provider.

Peers, applications, end users (clients), administrators, channels and organizations must have an identity provided by the Membership Service Provider (see Section 3.4) in order to be able to interact with the network. Each of these actors has a digital identity encapsulated in an X.509 digital certificate standard which must be unique to every entity. These determine the exact permissions these have over resources and access to information in the network. The MSP issues these certificates through the built-in Certificate Authority (CA) component, the Fabric CA. The Fabric CA is a private root CA provider that consists in a CA server and a CA client. The MSP also supports Certificate Revocation Lists as seen in Figure 4.2.

4.2.2 Administrating a HLF Networks

As discussed, a HLF network must have an administrator. HLF provides the *cryptogen*, *configtxgen*, *configtxlator* and *peer* tools that are used to configure the network to suit different needs and use cases.

The *cryptogen* tool generates cryptographic data consuming the file *crypto-config.yaml*.

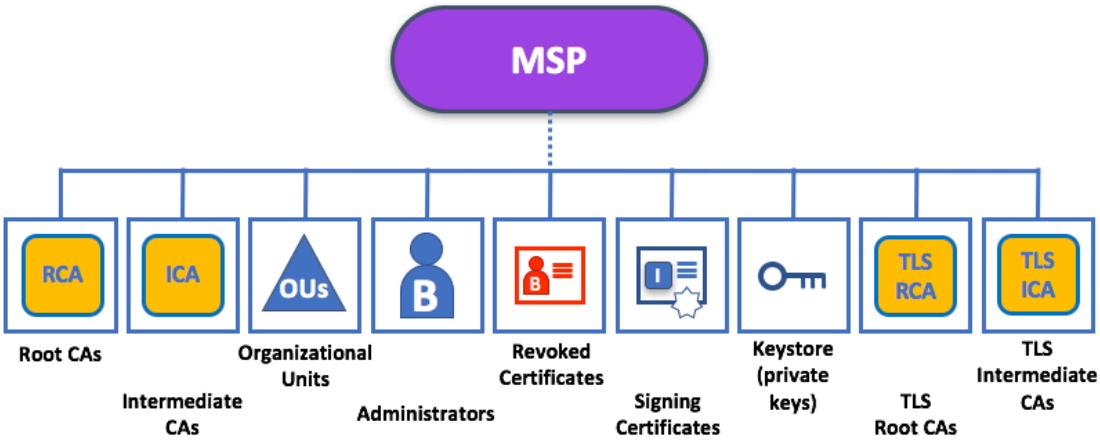

Figure 4.2: Fabric's Membership Service Provider Components Overview (Source: HLF Fabric Documentation)

The *configtxgen* tool generates the genesis block for the orderer services (see Section 3.4) and the initial transactions. This tool consumes the file *configtx.yaml* that defines configuration parameters for channels, the genesis block and the orderer service.

The *configtxlator* tool is also used to generate channel configurations. Finally the *peer* tool is used to manage the participating peers in the HLF network.

These tools are used to create and maintain the topology of the network and are invoked when a change to the network is made, for example, when permissions to certain records are changed or a new user is enrolled in the network and are very much intertwined with the Fabric Certificate Authority (CA) discussed in subsection .

4.3 Building the System

After considering the project goals of investigating the suitability of a Blockchain based system to manage patients identity data in Healthcare, the third step was to build a prototype of a system that would provide a simulation of the production network, albeit on a smaller scale. The insights gained from developing a simple working system would enable benefits and risks of the approach to be identified, and opportunities for further research to be laid out.

In order to build a solution, the research done before hand was taken into account and allowed a global overview of how architecturally a system could be built with the components available. After some consideration some approaches were reached, which are hereby presented in the following sections.

4.3.1 Conceptualization and Design

After an analysis of the defined requirements and the platform chosen, it was determined that the information that defines the patient's identity is a key requirement to build a system that recognizes patients across the Healthcare environment, as discussed in Chapter 2. An asset could be created through chaincode that represents the concept of the patient's identity in this network. This asset is stored in a Fabric network as a key-value pair. The key for this kind of asset could be a string. This string could be composed by the string 'Patient_', followed by a patient identification number assigned when the data is entered into the Blockchain. Since opti-

mization is not a key concern to build a simple prototype in the context of the work described in this dissertation, the patient's identification number was defined to simply be the order of the data entry starting at number one. This ensures that the key is unique and it is easily computable. In short, it was decided that the key is formed by the string 'Patient_' followed by the mentioned patient identification number. This key is used to query and access the patients data.

To aid in interoperability with other systems, the Fast Healthcare Interoperability Resources (FHIR) standard by the Health Level 7 organization was used as the basis for the fields in the structure, used to represent the patient's identity in the Healthcare domain. Each field of the patient's identity structure, defined in a smart contract, would be linked to a field of the patient structure as presented in the FHIR standard [Org18].

The most simple case of an interaction in an Healthcare service is the interaction between a patient and a doctor. In HLF, this situation translates to two organizations and two peers. Each peer belongs to an organization, one organization represents the patients while the other represents the hospital where the doctor works.

To establish a communication between the two participating peers, a channel is created between the peers, ensuring information exchanged between the two is private and does not exist on the rest of the network. If a third organization with another peer representing another health clinic joined the network, then another channel could be created between the patient's organization and this new organization. If the patient inserts his data in the channel then the clinic would be able to view it anytime they wished.

Starting in version 1.1 of Fabric the MSP allows Attribute Based Access Control meaning the access to the data can depend on the value of certain attributes of the certificate. Also it is possible to encrypt data and insert it into the channel and then require a key to decrypt the data. In this case the patient could give a key to the doctor to be able to access only his data. In the current version of Fabric, version 1.2, private data collections were introduced [Gut18], meaning that some data can be marked as private on a channel while other data can be public.

4.3.2 Implementation

Applications for Hyperledger Fabric can be developed in any language as long as there is a Software Development Kit (SDK) supporting the chosen language. At the time of writing this document, the Node.js SDK and the Java SDK are available with additional language support being worked on [Gut18].

To create an interactive system that can manage the patients identity in an Healthcare environment, an application was built using Node.js that the user interacts with. Node.js is a JavaScript runtime built on Google Chrome's V8 JavaScript engine. Node.js has an event-driven architecture focused on asynchronous input/output which optimizes the throughput and scalability of web applications with many input/output operations. In short, Node.js allows JavaScript to be run in the Node.js environment allowing for the creation of a command line interface tool, for example.

In regards to developing smart contracts or chaincode, Go was the first programming language to get support and since then additional languages were added to the list of officially supported languages. At the time of writing this document, the Go language SDK and Node.js SDK are available and the Java SDK was also recently released.

After some research, the Node.js SDK for developing chaincode had similar features in comparison with the Go SDK, in regards to API and features. However, documentation was more sparse and harder to find for the Node.js SDK.

On the other hand JavaScript was more approachable in contrast to Go, in order to implement the system due to the author's familiarity with the Node.js environment. Another important consideration is that, the application and the smart contract could both be built in this language which was considered an advantage as it simplifies dependency management.

Ultimately, this application interfaces with chaincode through the Hyperledger Node.js Fabric Software Development Kit. The chaincode was developed using the Hyperledger Fabric Shim for Node.js.

To avoid the need for multiple machines being created in order to form the network, a Docker Compose file was used that defines, and orchestrates the main components of the network through the Docker Engine. Docker is a platform that allows containerization [1]. The Docker Engine is an open source containerization technology offering a workflow for building and containerizing applications. A Docker Compose file specifies the topology of the service stack and allows orchestration of the services therein defined. With this in mind, each component of the network consists in one or more containers, with one container defined to be used as a command line interface to interact with the network using the peer tool, if needed for administrative purposes. Docker was used as the containerization technology because it is officially supported by Hyperledger and is currently the most popular containerization tool as of 2018 [Hec18, Dia18].

To build the desired network configuration for the prototype, the configuration file for the *cryptogen* tool was modified to allow the network to have two organizations, each of them with a peer associated. The configuration file for the configtxgen tool was also edited to allow a channel of information between the two organizations to be created. Each peer would serve as the anchor peer [2] in each of the organizations.

An application was also built that allows for user enrolment to create a new identity in the network. The application is run by a user and uses the available SDK to call upon the operations that the smart contract makes available. When a new user of the application enters the network, a function in the smart contract initializes the creation of the patient's data and writes the patient's Healthcare information to the ledger as a new asset, and also manages the ledger state through transactions as well as the world state. The overview of the architecture for this system is represented on Figure 4.3. Due to the security mechanisms these transactions are signed and endorsed by the administrator of the network and verified by the CA servers.

The assets loaded contain the necessary fields to identify a patient in an Healthcare context, such as its name and birth date, for example, as well as some other information necessary to manage this data as discussed previously.

These operations form an Application Programming Interface (API) as seen in Figure 4.4 that returns a payload in JavaScript Object Notation (JSON) format with information from the network. This API allows a query to be made to the network that returns the patients information, changing incorrect or outdated information, for example. It also allows an administrator to disable the identity structure of someone who is not participating in the network actively in a given moment, in order for that information to be read-only from that point on, with more operations available. This system architecture leads to a modular as well as extensible approach, regarding the availability of new operations that become available as soon as new versions of the smart contract are deployed.

[1] Containerization, refers to an operating system feature in which the kernel allows the existence of multiple isolated user-space instances without launching an entire virtual machine for each application. These instances, called containers, share the host operating system and hold only the application related binaries and libraries, thus being more lightweight and faster than virtual machines.

[2] Used to initiate communication between peers from different organizations. The anchor peer serves as the entry point for another organization's peer on the same channel to communicate with each of the peers in the anchor peer's organization.

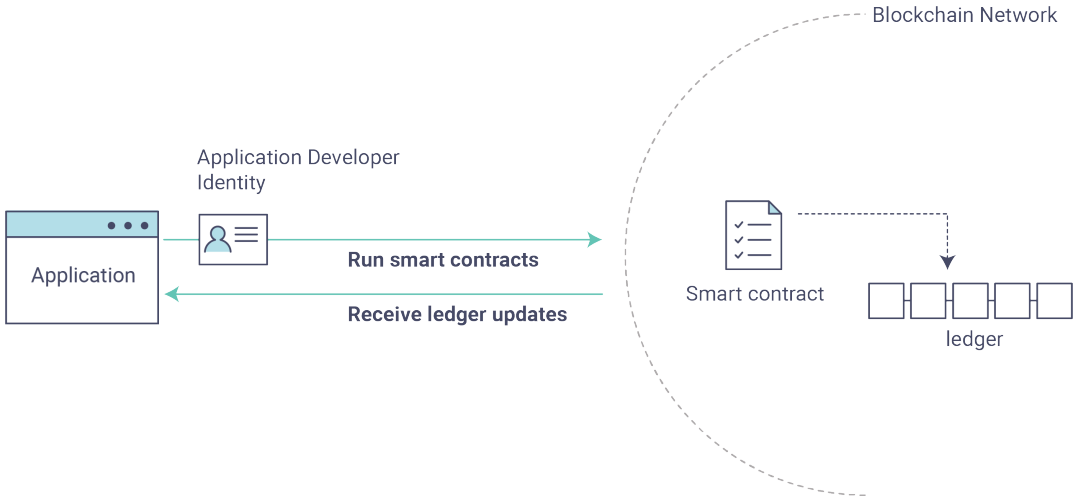

Figure 4.3: An Overview of the System Architecture (Source: HLF Fabric Documentation)

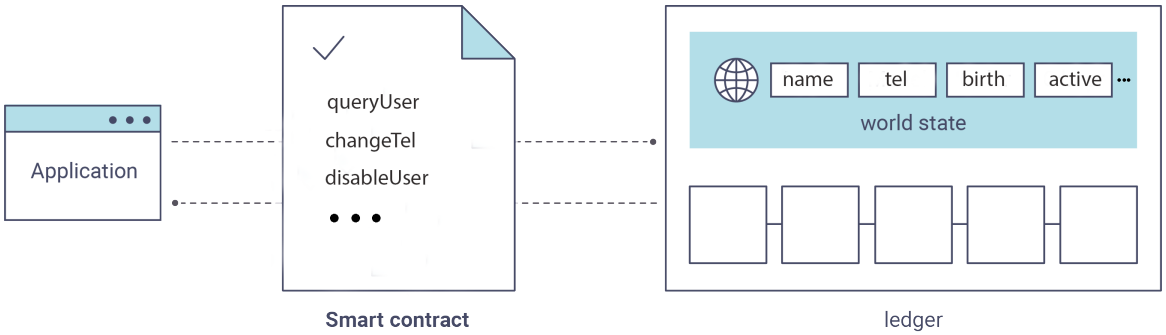

Figure 4.4: Smart Contract Operations Example (Original: HLF Fabric Documentation)

The network was brought up using the technologies mentioned in the previous Section and an administrator was enrolled into the network. This step is required because every action must be verified through a chain of trust and the administrator is the root CA in the network.

The next step was registering the patient and the doctor. Both the patient and the doctor invoked the register function of the application and were asked for a password. After successful register they were presented their assigned patient number in the format described previously.The password required by the register function was also stored in the certification store, as well as the patient number assigned to them. At this point they became active participants in the network. In this case, the assets that represent their identity were not created, because they had been exceptionally created before hand by the administrator to simplify the process. The normal flow would be the creation of both assets on registration.

4.3.3 Data Confidentiality in Fabric

During the implementation of the system it became clear that additional measures to secure data was needed. Even though Fabric is focused towards privacy and confidentiality, data inserted in a channel between a patient and a doctor, for example, would be stored in plain text by default. This means that inside the channel any entity would be able to access all data and see its contents if the required key to query a specific piece of data was obtained. To make information confidential some form of obfuscation or encryption could be used.

Fabric also features Attribute Based Access Control meaning that the chaincode logic could be altered to check if some attribute was present on the doctor's certificate that indicated that he had access to certain patients information. This could work for application users if there was no other way to access a network, however if users used a tool like Hyperledger Explorer they could see the data since the data is stored as plain text, as discussed in the previous paragraph.

After considering the different possibilities and making some tests it was considered necessary to use data encryption, as discussed in Section 5.1. The flow of the operations is designed to be simple. When the patient's data is registered he receives a notification to keep note of a data key which is used to encrypt data using a traditional SHA-256 encryption algorithm [Wik18c]. The data was encrypted using symmetric key encryption [Cen18] with the generated key being stored on the X.509 certificate [Wik18d] of the respective client securely. If a patient wanted to share his information with someone he would give the other entity his data key and that would allow him to decrypt the encrypted patient's data. To complete this system the key would need to be set to expire after a set amount of time and refresh itself, as discussed in Section 6.2. The key could always be accessible to the patient by accessing the data store where Fabric security mechanisms would ensure the certificate attributes are only accessible to the rightful owner.

With this system in place, the last requirement was fulfilled, and the system is now considered to be complete regarding the goals of this dissertation.

5

Experiments and System Evaluation

This Chapter presents the experiments using the solution created for managing patients identity data in a Healthcare context. Then the solution is evaluated against a security model and goals set by this dissertation are evaluated.

To properly evaluate this solution a number of experiments were conducted, as follows. First some data would be present on the channel when the user interacts that represents his identity in a certain clinic. The patient would query the Blockchain for his data and receive his data if everything worked accordingly. The second experiment was making the patient share his data with the doctor in the channel. The last experiment consisted in the patient trying to query data of another patient that was inserted at the genesis of the network and seeing if the data was encrypted or was easily readable. The outcomes of these experiments can shape the development of the solution as it could take these results into consideration and highlight possible problems.

5.1 Testing the Built Solution

With the network in place and the peers set up and registered the experiments proposed have now their requirements fulfilled.

The patient used the function provided by the application to query the network for his information. He searched for his patient number and was shown his information successfully. This shows that the information was recorded with success when the chaincode was deployed. The simple way to query personal information with an assigned patient number also proved successful and shows that this system can be used to store patient's identity data and retrieve it.

Then the patient had to share his information with the doctor. To proceed, it was necessary to assume that the patient had given his patient number to the doctor so that he could use the application built to query for that patient number. The doctor queried the network for the patient's information and was able to access it successfully. This proves that this platform allows, to very easily share information between a patient and a doctor using a smart contract in a simple way.

Finally the patient tried to access another patient's data. It was necessary to assume that he was given the patient number by the respective patient. When he queried the network for that patient's data it became clear what already had arose suspicions in previous experiments. He was actually able to access that data without a problem. This would be okay if the number was willingly given to him. However if the number was obtained unwillingly it could prove a problem. This meant that the solution currently, did not meet the requirement of the information being confidential that was defined previously, even tough it is transparent and has high availability since the information was spread through multiple peers and could be on multiple channels. It became clear that some additional data security measures was needed. After implementing data encryption (see Section 4.3.3), data was effectively encrypted and could not be seen, fulfilling the original goal.

5.2 Evaluation of the Built Solution

It was determined that to evaluate the effectiveness of this system in regards to security, a standard for these types of solution was needed.

After careful consideration, the international standard for information security known as the Confidentiality, Integrity, and Availability (CIA) triad model was used and the solution was evaluated against this standard, in order to draw further conclusions and evaluate how secure the built system is, in regards to data security, which is a critical concern in this particular field.

The three pillars that form this standard are the preservation of confidentiality, information availability and ensuring information integrity. The evaluation of the system against this model is presented over the following Section.

5.2.1 Confidentiality

Confidentiality of the information stored in the network was considered a key requirement when the requirements were presented. The Hyperledger Fabric was a prime candidate for building the solution upon due to its focus on privacy and a more enterprise approach to Blockchain development. While Hyperledger Fabric offers many features such as channels that truly do segregate information in a way that many equivalent platforms cannot do at the moment it is also true that by default data will be stored in plain text.

To solve this problem it was necessary to implement data encryption on top of the network using chaincode. This way, even if someone was able to access the underlying database or if someone used a tool like Hyperledger Explorer to explore the network, all it would see is encrypted data that would require a key to decrypt and become human readable. With these considerations in mind, it can be said that the built system provides a confidential data storage.

5.2.2 Integrity

One of the key aspects of a Blockchain system is the immutability of data. This means that once information is written, it cannot be changed or erased. The transaction logs assure that the specific version of that asset is recorded permanently in the network. In order to comply with privacy regulations some data can become only visible as an hash but it still remains there. Therefore the integrity of data on this Blockchain platform and solution is also preserved.

5.2.3 Availability

Even though Fabric is a permissioned Distributed Ledger Platform and as such it is administrated by an administrator it is also distributed and therefore avoids having a single point of attack. By default, it is more available than a simple informational system that is centralized. In this aspect it can be said that, the more the network scales, the more robust it becomes and therefore more availability it provides as information redundancy also increases.

5.2.4 Review of the System Goals

Using Hyperledger Fabric a system was built that successfully can create, manage and disable patients data. Information can be shared in a secure manner and interoperability eases organization into adopting this system. This system provides benefits to the medical staff as well as the patients due to transparency in how data is handled and secured.

However the costs of deploying this system in a production ready environment would be higher compared to a more traditional approach. Since this system is built upon a Permissioned Platform, machines to host the central services need to be acquired and an administrator of the platform is necessary for the necessary maintenance. As the network grows it would become more resilient and additional servers could be used to expand the core availability of Blockchain components.

There is also the question of scalability. Even though a Permissioned Blockchain is always faster in relation to a Permissionless variant it still is far from matching the scalability and performance of the Electronic Payment Management System created by SIBS [1] for banking transactions, for example. If this system was intended for global use, then additional approaches would need to be taken regarding this matter.

With this said, the pace of development has been relatively fast with new releases on a quarterly basis that focus on the issues of scalability and privacy, two important features pertaining to the system this Thesis proposed.

[1] The Sociedade Interbancária de Serviços is a company that manages all the debit card payment system in Portugal and that operates with all banks. The company is responsible for the Multibanco network. The network is comprised by the store payment machines and the automated banking machines that offer money withdrawal and payment services, for example. As of December 2014, the network had an average of more than 75 million operations every month.

6

Conclusions and Future Work

This Chapter presents some remarks about the conclusions observed during the course of the work described in this dissertation. Some possible future work is also presented based on the findings shown on this document.

In the context of this dissertation a system was built that is capable of managing patients identity using Blockchain technology. As this is a relatively new technology, there was the need to create simple system in order to reach some conclusions. The prototype system built leverages this technology in the Healthcare field. The platform chosen was the Hyperledger Fabric Distributed Ledger Platform.

While some difficulties were encountered developing a system in this technology, ultimately, a system using Hyperledger Fabric was successfully designed and built and some conclusions were taken into account. The platform was evaluated against the Confidentiality, Integrity and Availability triad, the international standard for information security.

6.1 Conclusions

Hyperledger Fabric was one of the first projects that were built under the Hyperledger umbrella, and it is a general purpose Distributed Ledger Platform.

Using Hyperledger Fabric a system was built that leverages the features of this platform to manage the identity of patients in a Healthcare environment. This system was designed and implemented successfully. The system was evaluated against the CIA triad model and was able to meet the desired requirements.

It was shown that it is possible to create a system for the purpose mentioned with this technology and that it provides several advantages in relation to more traditional systems such as data transparency to the patient, immutability of data and decentralization.

The patient benefits from transparency because he will be able to see his data anytime he wishes to do so, creating a greater degree of trust with the Healthcare organizations than what is possible nowadays. Immutability is important to maintain this trust because data ends up being altered in any way, then the record of that tampering would be forever recorded and could be traced back to the malicious actor due to this platform supporting auditability with the Membership Service Provider that issues client's identity certificates through the built-in CA service. Decentralization provides additional resiliency as it avoids having a single point of failure that could be targeted.

This work described in this dissertation proves that this technology has many applications in this field, and that it can be used more often as the platform becomes more mature and complete.

This research is especially important because security must be a key focus of the Healthcare industry for the next few years, as expertise will increase in the digital space providing more opportunities for malicious parties to use potential flaws in the information systems deployed in the Healthcare organizations.

In the Healthcare field, patients data must be treated with the utmost care because health information is a sensitive and personal matter for each patient. The privacy rights of the patients must be respected and, as Healthcare becomes a more digital industry, technology will need to provide additional means to help the medical professionals ensure this.

6.2 Future Work

There are many approaches that can be taken in future work. Blockchain technology is certainly interesting and other platforms can be explored to evaluate their suitability in Healthcare. As this platform matures and new features are added future work could be built upon the new features added or new Hyperledger projects such as Hyperledger Indy.

Hyperledger Indy was a platform that was not available when this dissertation was initially discussed. The platform is an interesting choice because of its focus on identity. Some research on this platform could be made to map potential similarities to Fabric in ways that could show their common points and differences. It could be shown, for example, which platform would be more suited for this task or the different use cases they excel in could be noted.

The prototype project could be expanded with a graphical interface, instead of the current command line interface, leading to a more intuitive usage of this system and additional platforms to be targeted naturally. Further optimization efforts could be made regarding channels, encryption methods, data segregation and scalability. To improve security, the key generated when an entity registers in the network should refresh periodically, to avoid potential identity theft.

Fabric roles could be explored to distinguish between a doctor and a nurse in the same Healthcare organization. Access to information could be regulated using the Role Based Access Control added to Fabric. Hyperledger Burrow could be used to improve compatibility between Ethereum and Hyperledger Fabric and target two platforms with similar smart contract code. Finally this concept could be expanded further and Blockchain could serve as a secure access key store that enables secure transmission of access keys to external protected servers that store a big amount of data that would not be suited for the Blockchain .

Bibliography

[ABB+18] Elli Androulaki, Artem Barger, Vita Bortnikov, Christian Cachin, Konstantinos Christidis, Angelo De Caro, David Enyeart, Christopher Ferris, Gennady Laventman, Yacov Manevich, Srinivasan Muralidharan, Chet Murthy, Binh Nguyen, Manish Sethi, Gari Singh, Keith Smith, Alessandro Sorniotti, Chrysoula Stathakopoulou, Marko Vukolić, Sharon Weed Cocco, and Jason Yellick. Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains. 2018.

[Baa16] Djuri Baars. Towards Self-Sovereign Identity using Blockchain Technology. *University of Twente*, 2016.

[Bar17] Iain Barclay. Innovative Applications of Blockchain Technology in Crime and Security. 2017.

[Blo17a] BlockGeeks Ethereum Guide. **https://blockgeeks.com/guides/ethereum/**, 2017. [Online; Accessed November 29, 2017].

[Blo17b] Is Blockchain the Answer to Healthcare's Big Data Problems? **https://healthitanalytics.com/news/is-blockchain-the-answer-to-healthcares-big-data-problems**, 2017. [Online; Accessed December 1, 2017].

[Bov17] Charles Bovaird. Top 5 factors driving bitcoin higher this year. **https://www.forbes.com/sites/cbovaird/2017/12/22/top-5-factors-driving-bitcoin-higher-this-year/**, December 2017. [Online; Accessed August 5, 2018].

[Buc16] Ethan Buchman. Tendermint: Byzantine Fault Tolerance in the Age of Blockchains. 2016.

[But14] Vitalik Buterin. Slasher ghost, and other developments in proof of stake. **https://blog.ethereum.org/2014/10/03/slasher-ghost-developments-proof-stake/**, 10 2014. [Online; Accessed August 30, 2018].

[But18] Vitalik Buterin. Ethereum scalability research and development subsidy programs. **https://blog.ethereum.org/2018/01/02/ethereum-scalability-research-development-subsidy-programs/**, 01 2018. [Online; Accessed August 17, 2018].

[Cac16] Christian Cachin. Architecture of the hyperledger blockchain fabric. *IBM Research*, July, 2016.

[CD16] Konstantinos Christidis and Michael Devetsikiotis. Blockchains and Smart Contracts for the Internet of Things. *IEEE Access*, 4:2292–2303, 2016.

[Cen18] IBM Knowledge Center. Symmetric cryptography. https://www.ibm.com/support/knowledgecenter/en/SSB23S_1.1.0.14/gtps7/s7symm.html, 12 2018. (Accessed on 12/29/2018).

[Com16] Ethereum Community. What is ethereum? — ethereum documentation. http://ethdocs.org/en/latest/introduction/what-is-ethereum.html, 2016. [Online; Accessed August 16, 2018].

[Con18] Healthcare Services Platform Consortium. Hspc website. https://www.hspconsortium.org/, 12 2018. [Online; Accessed December 29, 2018].

[D'A16] Michele D'Aliessi. How does the blockchain work?, medium. https://medium.com/@micheledaliessi/how-does-the-blockchain-work-98c8cd01d2ae, 1 2016. [Online; Accessed August 13, 2018].

[d'A17] Frisco d'Anconia. Bitcoin mining 'wastes vast amounts of energy, harms environment'. https://cointelegraph.com/news/bitcoin-mining-wastes-vast-amounts-of-energy-harms-environment, 08 2017. [Online; Accessed August 30, 2018].

[Dia18] Diamanti. 2018 container adoption benchmark survey. https://diamanti.com/wp-content/uploads/2018/07/WP_Diamanti_End-User_Survey_072818.pdf, 07 2018. [Online; Accessed October 14, 2018].

[Dre17] Daniel Drescher. *Blockchain Basics*. Apress, Berkeley, CA, 2017.

[Dre18] Dustin Dreifuerst. Permissioned vs. permissionless blockchains, medium. https://medium.com/@dustindreifuerst/permissioned-vs-permissionless-blockchains-acb8661ee095, 3 2018. [Online; Accessed August 13, 2018].

[EAR+06] Marco Eichelberg, Thomas Aden, Jörg Riesmeier, Asuman Dogac, and Gokce B Laleci. Electronic Health Record Standards - A Brief Overview. *ITI International Conference on Information & Communications Technology*, 2006.

[Est16] Estonian Government Adopts Blockchain To Secure 1 Mln Health Records. https://cointelegraph.com/news/estonian-government-adopts-blockchain-to-secure-1-mln-health-records, 2016. [Online; Accessed February 25, 2018].

[Eva16] Philip Evans. Blockchain and digital tokens: A strategic perspective. https://www.bcg.com/blockchain/thinking-outside-the-blocks.html, 12 2016. [Online; Accessed August 12, 2018].

[Fac17] Factom's Latest Partnership Takes on US Healthcare. https://cointelegraph.com/news/factoms-latest-partnership-takes-on-us-healthcare, 2017. [Online; Accessed December 1, 2017].

[For11] Bitcoin Forums. Proof of stake instead of proof of work. https://bitcointalk.org/index.php?topic=27787.0, 06 2011. [Online; Accessed December 29, 2018].

[Gem18] Gem. Gemos | gem. https://enterprise.gem.co/gemos/, 09 2018. [Online; Accessed September 29, 2018].

[Gop16] Ramesh Gopinath. Checking the ledger: Permissioned vs. permissionless blockchains. https://www.ibm.com/blogs/think/2016/07/checking-the-ledger-permissioned-vs-permissionless-blockchains/, 07 2016. [Online; Accessed August 14, 2018].

[Gua18] Verizon to Use KSI Blockchain Technology Developed for Estonia. https://www.sdxcentral.com/articles/news/verizon-use-ksi-blockchain-technology-developed-estonia/2018/02/, 2018. [Online; Accessed February 25, 2018].

[Gut18] Carlo Gutierrez. Hyperledger fabric v1.2: What's new and the roadmap for 2018. https://www.altoros.com/blog/hyperledger-fabric-v1-2-whats-new-and-roadmap-for-2018/, 08 2018. [Online; Accessed August 18, 2018].

[Hea17] Health Level 7 Webpage. https://hl7.org/, 2017. [Online; Accessed January 15, 2018].

[Hec18] Lawrence Hecht. Steady docker adoption leads to jump in hiring - the new stack. https://thenewstack.io/steady-docker-adoption-leads-to-jump-in-hiring/, 06 2018. [Online; Accessed October 14, 2018].

[Hyp17a] Hyperledger Burrow Github Page. https://github.com/hyperledger/burrow, 2017. [Online; Accessed November 29, 2017].

[Hyp17b] Hyperledger Fabric Documentation. https://hyperledger-fabric.readthedocs.io/en/release/, 2017. [Online; Accessed November 29, 2017].

[Hyp18] Hyperledger. Hyperledger passes 250 members with addition of 9 organizations. https://www.hyperledger.org/announcements/2018/07/31/hyperledger-passes-250-members-with-addition-of-9-organizations, 08 2018. [Online; Accessed August 29, 2018].

[Kal06] Dipak Kalra. Electronic Health Record Standards. *IMIA Yearbook of Medical Informatics*, 2006.

[Ken18] Hu Kenneth. Ethereum test networks. https://medium.com/coinmonks/ethereum-test-network-21baa86072fa, 05 2018. [Online; Accessed August 16, 2018].

[KMBD17] Casey Kuhlman, Dan Middleton, Benjamin Bollen, and Silas Davis. Hyperledger Burrow (formerly eris-db). 2017.

[Kum18] Utsav Kumar. Understanding ethereum — pertinent problems,scalability, and possible solutions. https://medium.com/coinmonks/understanding-ethereum-pertinent-problems-scalability-and-possible-solutions-eb4fec0405be, 06 2018. [Online; Accessed August 17, 2018].

[Lei17] Matthew Leising. Ether thief remains a mystery a year after a 55 million usd digital heist. https://www.bloomberg.com/features/2017-the-ether-thief/, 06 2017. [Online; Accessed September 30, 2018].

[Lew15] Antony Lewis. A gentle introduction to Blockchain Technology. *Bits On Blocks*, 2015.

[Lon18] Jonathan Long. 12 startups utilizing blockchain technology in new ways. https://www.entrepreneur.com/article/310373, March 2018. [Online; Accessed August 5, 2018].

[Mar17] Gabby Marquez. The history of electronic health records | elation health. https://www.elationhealth.com/clinical-ehr-blog/history-ehrs/, August 2017. [Online; Accessed August 5, 2018].

[Mar18] Bernard Marr. 35 amazing real world examples of how blockchain is changing our world. https://www.forbes.com/sites/bernardmarr/2018/01/22/35-amazing-real-world-examples-of-how-blockchain-is-changing-our-world/#3f13103843b5, January 2018. [Online; Accessed August 5, 2018].

[Med18] MEDICHAIN - The Medical Big-Data Platform. https://medichain.online/, 2018. [Online; Accessed February 25, 2018].

[Nak08] Satoshi Nakamoto. Bitcoin: A Peer-to-Peer Electronic Cash System. *https://bitcoin.org/bitcoin.pdf*, 2008.

[ONC17] ONC. Electronic health records infographic. https://www.healthit.gov/infographic/electronic-health-records-infographic, September 2017. [Online; Accessed August 5, 2018].

[Org16] HL7 Organization. Hl7 2016 annual report. http://www.hl7.org/documentcenter/public_temp_6FBE48E5-1C23-BA17-0C93EDE774C2450A/HL7/HL7%202016%20Annual%20Report_FINAL_WEB.pdf, 2016. [Online; Accessed August 5, 2018].

[Org17] Hyperledger Organization. Hyperledger announces performance and scalability working group – hyperledger. https://www.hyperledger.org/blog/2017/06/08/hyperledger-announces-performance-and-scalability-working-group, 06 2017. [Online; Accessed September 26, 2018].

[Org18] HL7 Organization. Patient - fhir v4.0.0. http://www.hl7.org/fhir/patient.html, 12 2018. [Online; Accessed December 29, 2018].

[Pri16] Giulio Prisco. The blockchain for healthcare: Gem launches gem health network with philips blockchain lab. https://bitcoinmagazine.com/articles/the-blockchain-for-heathcare-gem-launches-gem-health-network-with-philips-blockchain-lab-1461674938/, 04 2016. [Online; Accessed September 29, 2018].

[Ray18] James Ray. Proof of stake faqs - ethereum wiki. https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQs#what-are-the-benefits-of-proof-of-stake-as-opposed-to-proof-of-work, 08 2018. [Online; Accessed August 16, 2018].

[SBV18] Joao Sousa, Alysson Bessani, and Marko Vukolic. A Byzantine Fault-Tolerant Ordering Service for the Hyperledger Fabric Blockchain Platform. *Proceedings - 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2018*, 2018.

[SWP16] David Shrier, Weige Wu, and Alex Pentland. MIT - Blockchain & Infrastructure (Identity , Data Security). 2016.

[Tec18] KSI Technology. Industrial scale blockchain. https://guardtime.com/technology, 12 2018. [Online; Accessed December 29, 2018].

[TFH17] Jonas Tana, Maria Forss, and Thomas Hellstén. The use of wearables in healthcare – challenges and opportunities. pages 1–8, 2017.

[TJR18] McCoy TH, Jr, and Perlis RH. Temporal trends and characteristics of reportable health data breaches, 2010-2017. *JAMA*, 320(12):1282–1284, 2018.

[VS17] Martin Valenta and Philipp Sandner. Comparison of Ethereum, Hyperledger Fabric and Corda. (June), 2017.

[Wik18a] Wikipedia. Byzantine fault tolerance. https://en.wikipedia.org/wiki/Byzantine_fault_tolerance, 12 2018. [Online; Accessed December 29, 2018].

[Wik18b] Wikipedia. Internet of things. https://en.wikipedia.org/wiki/Internet_of_things, 12 2018. [Online; Accessed December 29, 2018].

[Wik18c] Wikipedia. Sha-2. https://en.wikipedia.org/wiki/SHA-2, 12 2018. (Accessed on 12/29/2018).

[Wik18d] Wikipedia. X.509. https://en.wikipedia.org/wiki/X.509, 12 2018. (Accessed on 12/29/2018).

[Woo15] Gavin Wood. Blockchain what and why? https://www.slideshare.net/gavofyork/blockchain-what-and-why, 1 2015. [Online; Accessed August 13, 2018].

[Woo17] Gavin Wood. Ethereum: a Secure Decentralised Generalised Transaction Ledger. 2017.

[Zag18] Matteo Gianpietro Zago. 50+ examples of how blockchains are taking over the world. https://medium.com/@matteozago/50-examples-of-how-blockchains-are-taking-over-the-world-4276bf488a4b, May 2018. [Online; Accessed August 5, 2018].

UNIVERSIDADE DE ÉVORA
INSTITUTO DE INVESTIGAÇÃO
E FORMAÇÃO AVANÇADA

Contactos:
Universidade de Évora
Instituto de Investigação e Formação Avançada — IIFA
Palácio do Vimioso | Largo Marquês de Marialva, Apart. 94
7002 - 554 Évora | Portugal
Tel: (+351) 266 706 581
Fax: (+351) 266 744 677
email: iifa@uevora.pt